IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 1, January 2021

DOI: 10.17148/IARJSET.2021.8107

A Critical Review on Design and Implementation of AES Algorithm

Geethashree A[1], Suchitra M[2], Praveena K S[3]

Associate Professor, Department of ECE, Vidyavardhaka College of Engineering, Mysuru, India[1]

Professor, Department of ECE, Vidyavardhaka College of Engineering, Mysuru, India[2]

Assistant Professor, Department of ECE, Vidyavardhaka College of Engineering, Mysuru, India[3]

Abstract: Cryptography plays a vital role in securing confidential data of the user from being accessed by a third person. It is used to protect and secure businesses corporate secrets, classified information of the government and personal information of people and guards against identity theft. The Advanced Encryption Standard (AES) algorithm is a symmetric encryption algorithm which is approved by the U.S government as a standard for encrypting sensitive and private data. This paper summarizes the different techniques used to implement Advanced Encryption Standard (AES) Algorithm. The software implementation of the AES algorithm is cost effective and easier to implement. However, hardware implementation has greater speed and security.

Keywords: AES algorithm, ASIC implementation, Cryptography, Encryption, FPGA implementation, SubBytes transformation, S-Box.

I. INTRODUCTION

People are constantly using computer and internet to transmit private or confidential data to one another. It is crucial to ensure that this data is secured against any unauthorized access. Encryption is the process of encoding a data, message or information in such a way that only authorized members can access it. Earlier, DES and DES3 algorithm was used to perform encryption. However, its encryption and decryption speed were slow and also had security issues like linear and differential cryptanalysis attacks and Bruce force attack. There was a necessity for a much faster and secure algorithm and thus, AES algorithm was adopted. It is a widely used algorithm and is considered as a standard for securing file transfer protocols like HTTPS, SFTP, FTPS, AS2, OFTP etc. The standard key sizes used in the AES Algorithm can be 128, 192 or 256 bits. It has greater encryption speed, decryption speed and simulation speed when compared to previous algorithms. It also consumes less power and can be implemented on hardware as well as software.

II. AES ALGORITHM

The plain text or the data that needs to be encrypted is converted into binary data. The algorithm considers 128 bits of binary data at a time for processing. The 128 bits of data is converted into a 4 × 4 matrix with each element of the matrix containing 8 bits of data. The data undergoes a Pre-Round transformation where the data matrix is xor-ed with the key matrix. The size of the key can be 128, 192 or 256 bits and the number of rounds of transformation varies accordingly as 10, 12 and 14. The transformation comprises of four functions namely, SubBytes, ShiftRows, MixColumns and AddRoundKey as shown in Figure 1. For the last round, the MixColumns function is not performed.

A. SubBytes Transformation

In the SubBytes step, the substitution box (S-Box) unit performs substitution on each element of the matrix by performing multiplicative inverse and applying an affine transform in the finite field GF (28) to generate an output byte. Out of all the blocks implemented in the AES algorithm this block is non-linear and complex. The design of S-Box can be implemented in two ways.

Implementation of S-Box Using ROM: The output of the S-Box remains constant for a given input. Thus, a static look-up-table (LUT) can be designed using a fully custom set of 8-bit ROMs that can be used to store the output substitution values for each of the combination of inputs [1][3]. A secure ROM is designed as shown in [2] to combat the capacitance variation from bit-line to bit-line and word-line to word-line in an unsecure ROM cell. The worst-case power consumption in the decode logic is reduced through the symmetry of the decode operation and a pulse-mode logic style [2]. This also ensures that the power dissipation is data independent. Although the design of S-box using ROM is easier to implement and handle, it leads to delay in execution. It does not support pipelining and also increases the possibility of hacking [4].

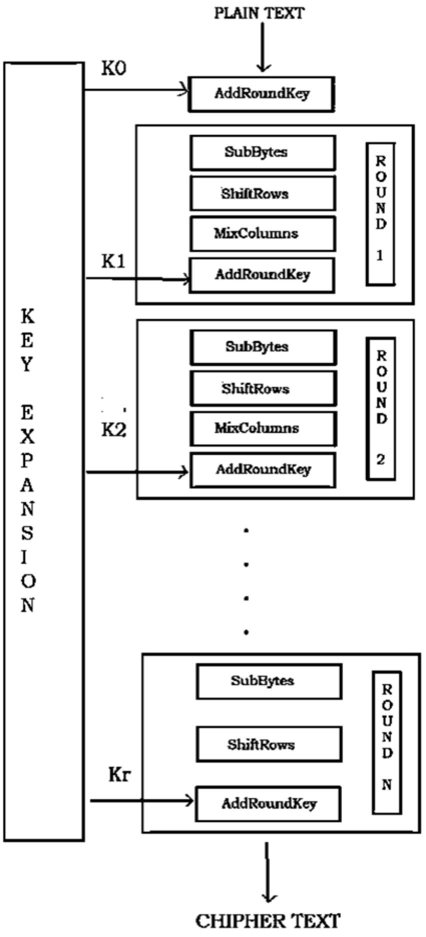

Figure 1. The AES Algorithm Block Diagram

Implementation of Customized S-Box: Due to the above-mentioned disadvantages of designing S-Box using ROM, it is recommended to design a custom block that calculates the multiplicative inverse of each 16-bit data of the state and finds its affine transform. Both the operations are performed in finite field GF (28). The design of these blocks is quite complex and requires mathematical knowledge about finite field. However, it produces output efficiently [4][5].

B. ShiftRows Transformation

The first row of the 4 × 4 matrix is kept unchanged. The second row is of the matrix is shifted to the left once, the third row is left shifted twice and the last row is left shifted thrice. This is performed to ensure the data is thoroughly scrambled.

MixColumns Transformation: In this step, the matrix is multiplied by a irreducible GF(28) polynomial which can be represented in matrix as,

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

The multiplication performed in this step is in Galois field. It is not a regular matrix multiplication.

AddRoundKey Transformation: The matrix is xored with the key matrix which is obtained through a key expansion. The key chosen to perform this operation is unique and known only to the sender and receiver.

III. Literature Survey

There are different parameters on which AES algorithm can be analyzed. This literature survey classifies the AES algorithm on the basis on its implementation techniques.

A. Software Implementation

DOI: 10.17148/IARJSET.2021.8107

One of the major advantages of AES algorithm is that it can easily be implemented on software. The program can be written using software tools like C, HTML, Java, MATLAB and Verilog/VHDL and run of various processors. Different processors take different execution time to encrypt and decrypt the data. A modification to Gladman implementation was proposed in paper [7] to obtain greater encryption/decryption speed. Paper [9] revisits the concepts of AES algorithm and presents the results of paper [7] expressing the concern of delay in execution speed due to look up table. The execution speed can be improved for applications such as smart cards by designing AES algorithm without lookup table. The Leon3 processor is a 32-bit open source core processor which is synthesized using VHDL and used for faster applications due to its multiprocessing configuration [6]. Speed and resource requirement are mainly considered to select the architecture for the application. The speed optimization of a 32-bit processor is made possible by rearranging the inner operations of the AES rounds and also the byte-operations involved in it. The inner operations are computed with respect to other ones and are then grouped in such a way as to fit well in processors having 32-bits words. This optimization leads to better utilization of the resources by the processor and ensures improvement in timing performances with respect to the standard implementation of the AES algorithm [7][8]. The implementation can be made more secure and less prone to side channel attacks and zero attacks with low computation cost and memory requirements by using log and antilog tables for arithmetic computations in Galois fields directly on masked data [10]. Software encryption tools like VeraCrypt, AES Crypt use AES algorithm for encrypting files and disks.
Advantages of Software Implementation are:

- Easier to make parameter changes (like key size) when compared to hardware implementation.
- Cost of development is relatively low.
- Easier to test and verify the software designs

Disadvantages of Software Implementation are:

- It is dependent on a processor to execute the stages.
- Execution time taken is more as it utilizes general purpose processor to execute the stages of AES algorithm. The processing time of software implementation is almost 100 times more than hardware implementation. [11]

B. Software Implementation Using Intel AES-NI Instruction Set

A set of processor-based instructions are introduced by Intel from 2010 Intel® Core™ processors family Intel. These instructions are called AES New Instruction Set. These instructions can be used to implement AES encryption/decryption with great ease without the need of Look up Table (LUT). The encryption/decryption speed is very high speed. It is capable of protecting the data against cache-access based side channel analysis. The instruction set consists of six instructions which provide high performance encryption and decryption of the data. The instructions AESIMC/ AESKEYGENASSIST are used to support the AES key expansion [13][14].

- Advantages of AES-NI Implementation:
  - Almost 100 times faster than fast AES implementation.
  - Provides full hardware support for any mode of operation and key length
- Disadvantages of AES-NI Implementation:
  - Available in only Intel family processors.
  - Can’t be used for variety of applications.

C. FPGA Implementation

Field Programmable Gate Array (FPGA) is an integrated circuit (IC) which is reconfigurable and can be bought off the shelf by designers [15]. If the hardware implementation is done using pipelining structure and parallel processing then the speed of encryption increases and the mode of data transmission is modified so that the chip size can be reduced [16]. If the aim of the design is to reduce the hardware structure and optimize the area then a simple controller is used to identify complete 128 bits data [16]. In order to enhance the security of the encryption, the State matrix values are multiplied with a pseudorandom noise sequence. The key sequence required for the encryption/decryption is also generated using pseudorandom sequence generator. This technique also improves the accuracy of decryption [19]. Boolean masking is implemented to design a 1st order mask which can resist 1st order differential (or correlation) power attack [18]. The table I compares various AES implementation using FPGA.

- Advantages of FPGA Implementation
  - It is reconfigurable, i.e., if the parameters of the AES algorithm like key size, data size changes then the design could be reconfigured.
  - Suitable for lesser units of production.
  - It has lesser latency and high throughput over software implementation
  - Faster than software implementation
  - Security is high

- Disadvantages of FPGA Implementation:
  - It is expensive for mass production.
  - Power consumed is more in FPGA when compared to an ASIC
  - Timing delay in FPGA is more when compared to an ASIC
  - Occupies more area than ASIC

TABLE I. COMPARING VARIOUS AES IMPLEMENTATION USING FPGA

Reference	FPGA Devices	Number of slices/ hardware resources	Maximum Frequency (in MHz)
[17]	XC6SLX451	3854	153.3
[18]	XC2V3000	4308	132.5
[19]	XC4VLX60	20,818	214.48
	XC6SLX150	5566	237.45
	XC7VX690T	697	372.98
	XC6VLX240T	4095	463.42
	XC5VSX240T	3420	199.18
	XC5VLX110T	3788	232.30
[22]	Spartan XC3S100E-5VQ100	930	81.74

D. Implementation Using ASIC

An Application Specific Integrated Circuit (ASIC) is a type of integrated circuit chip which is customized for a particular application. It is suitable for mass production. The area can be optimized by adopting an iterative round-looping structure which maps sub modules to lower data paths of 8 and 32 bit [4]. Further, customized Sbox can be designed instead of ROM based look up table (LUT). Mathematical operations can be performed to reduce Galois Field GF (28) to GF (2((2)(23))). The design is complex but the throughput increases. There are very few papers that focusses on ASIC Implementation of AES algorithm. The technology and number of gates used in this implementation is as mentioned in Table II. However, this type of implementation has more potential and possesses more advantages over other methods of implementation[25].

- Advantages of ASIC implementation:
  - Customized for a specific application and thus has zero unutilized elements
  - Requires less area, power and timing
  - High throughput
- Disadvantages of ASIC implementation:
  - Cost per unit is high
  - Design implementation is complex
  - Design cannot be modified once manufactured. It is not reusable.
  - Non-Recurring Cost is high

TABLE II. COMPARING ASIC PARAMETERS FOR IMPLEMENTING AES ALGORITHM

Reference	[22]	[3]	[1]	[23]
Technology	130 nm	180 nm	65nm	110 nm
Number of gates	7229	58445	-	21337

IV. CONCLUSION

This paper reviews various design and implementations of AES algorithm. The software implementation is easy and cost effective but lacks security and speed. The FPGA implementation has high speed and security. It also has less latency and high throughput over that of the software implementation which can be observed in Table III. However, the FPGA implementation has unutilized registers, pins, logic elements and memory which increases timing, area and power. The ASIC implementation is customized for a specific application and thus has zero unutilized elements. It requires less area, power and timing. It also provides high throughput. Comparing all the implementations, we have decided to design and implement the AES algorithm using ASIC with a customized S-Box. The customized S-box provides better speed and security over ROM bases S-box. The customized standard cells provide better area, power and timing over the standard cells available in the pre-existing library.

TABLE III. THROUGHPUT OF VARIOUS AES IMPLEMENTATIONS

Implementation	Reference	Specification	Throughput (in Gbps)
Software Implementation	[6]	Leon Processor, 128-bit key	0.00134
		Leon Processor, 192-bit key	0.00135
		Leon Processor, 256-bit key	0.00136
FPGA Implementation	[17]	XC6SLX451	1.57
	[19]	XC4VLX60	2.74
		XC6SLX150	3.03
		XC7VX690T	4.34
		XC6VLX240T	5.93
		XC5VSX240T	25.5
		XC5VLX110T	29.73
	[18]	XC2V3000, 128-bit key XC2V3000, 192-bit key XC2V3000, 256-bit key	1.005 0.96 0.704
ASIC Implementation	[2]	Secure ROM S-Box implementation (S-ROM) in 65 nm technology	6.15
		Unsecure ROM S-Box implementation (U-ROM) in 65 nm technology	13.27
		Synthesized, combinational logic S-Box (U-logic) in 65 nm technology	5.387
		Synthesized, combinational logic S-Box (U-logic) scaled to 90 nm technology	4.341
	[21]	130 nm technology	0.13
	[1]	65nm technology	275
	[3]	180 nm technology, 128-bit key	1.6
		180 nm technology, 192-bit key	1.33
		180 nm technology, 256-bit key	1.14

REFERENCES

[1] Burak Erbagci, Nail Etkin Can Akkaya, Craig Teegarden, Ken Mai "A 275 Gbps AES Encryption Accelerator Using ROM-based SBoxes in 65nm" Institute of Electronics and Electrical Engineering Journals and Magazines -2015.

[2] Craig Teegarden, Mudit Bhargava, Ken Mai "Side-Channel Attack Resistant ROM-Based AES S-Box" IEEE International Symposium on Hardware-Oriented Security and Trust (HOST) -2010.

[3] P.V. Sriniwas Shastry, Amruta Kulkarni, Mukul S.Sutaone "ASIC Implementation of AES" Annual IEEE India Conference (INDICON) -2012.

[4] V.Nanuku Naik, P. Michael Cholines "Design of Combinational S-Box Implemented AES Algorithm for Multiple Fault Detection Scheme" International Journal of Advance Research in Science and Engineering Vol. No. 4, Issue No. 12, December 2015.

[5] Neenu Shaji, Bonifus P.L "Design of AES architecture with area and speed tradeoff" International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST - 2015).

[6] Afef Kchaou, Wajih El Hadj Youssef, Rached Tourki "Software Implementation of AES Algorithm on LEON3 Processor" 15th international conference on Sciences and Techniques of Automatic control & computer engineering - STA'2014, Hammamet, Tunisia, December 21-23, 2014.

[7] Guido Bertoni1, Luca Breveglieri, Pasqualina Fragneto, Marco Macchetti and Stefano Marchesin "Efficient Software Implementation of AES on 32-bits Platforms" Cryptographic Hardware and Embedded Systems - 4th International Workshop Redwood Shores, CA, USA, August 13–15.

[8] Abhilasha CP, Nataraj KR "Software Implementation of AES Encryption Algorithm" International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 5, May, 2016.

[9] Eashwar Thiagarajan and Madhuri Gourishetty "Study of AES and its Efficient Software Implementation" International Journal of Science and Technology, March 2017.

[10] E. Trichina and L. Korkishkoternopo "Secure and Efficient AES Software Implementation for Smart Cards" Academy of National Economy, Ukraine.

[11] Thanapol Hongsongkiat, Prabhas Chongstitvatana "AES Implementation for RFID Tags: The Hardware and Software Approaches" International Computer Science and Engineering Conference (ICSEC)-2014.

[12] R. Velayutham And D. Manimegalai "Analysis of AES hardware and software implementation" Oriental Journal of Computer Science & Technology Vol. 3(1), 83-88 (2010).

[13] Shay Gueron "White box AES using Intel's New AES Instructions" 10th International Conference on Information Technology: New Generations-2013.

[14] Guang-liang Guo, Quan Qian, Rui Zhang "Different Implementations of AES Cryptographic Algorithm" IEEE 17th International Conference on High Performance Computing and Communications (HPCC)-2015.

DOI: 10.17148/IARJSET.2021.8107

[15] Ashwini M. Deshpande, Mangesh S. Deshpande and Devendra N. Kayatanavar “FPGA Implementation of AES Encryption and Decryption” International Conference On “Control, Automation, Communication and Energy Conservation -2009, 4th-6th June 2009.

[16] Sonali A. Varhade, N. N. Kasat “Implementation of AES Algorithm Using FPGA & Its Performance Analysis” International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Volume 4 Issue 5, May 2015.

[17] Ye Yuan, Yijun Yang, Liji Wu, Xiangmin Zhang “A High-Performance Encryption System Based on AES Algorithm with Novel Hardware Implementation” June,2015.

[18] Yuwen Zhu, Hongqi Zhang, Yibao Bao “Study of the AES Realization Method on the Reconfigurable Hardware” International Conference on Computer Sciences and Applications-2013.

[19] Harshali Zodpe, Ashok Sapkal “An efficient AES implementation using FPGA with enhanced security features” Journal of King Saud University – Engineering Sciences-23rd July 2018.

[20] Mohini Mohurle and Vishal V. Panchbhai “Review on Realization of AES Encryption and Decryption with Power and Area Optimization” 1st IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES-2016).

[21] Thanapol Hongsongkiat, Prabhas Chongstitvatana “AES Implementation for RFID Tags: The Hardware and Software Approaches”, International Computer Science and Engineering Conference (ICSEC)-2014.

[22] Neenu Shaji, Bonifus P.L “Design of AES architecture with area and speed tradeoff” International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST - 2015).

[23] Akashi Satoh, Sumio Morioka, Kohji Takano, and Seiji Munetoh “A Compact Rijndael Hardware Architecture with S-Box Optimization” Springer-Verlag Berlin Heidelberg 2001.

[24] Krithika Sharma N. "A Review on Design and Application of CORDIC algorithm." IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), vol. 10, no. 1, 2020, pp. 01-08.