A REVIEW ON HIGH SPEED ADVANCE ENCRYPTION STANDARD (AES)

Pushpinder Kaur, Preeti Gupta, Sanjiv Kumar
UIET, PU, Chandigarh, India

ABSTRACT - AES is one of the algorithms that are used for data security transmission. Using AES we can secure our data while sending from one person to another person. Our data can be in the form of text data, image data, video data, animated data etc. Need of AES is to stop the cyber-thief that could stole your keystrokes and access your passwords, credit card numbers and important information. AES is a specification for the encryption of electronic data. Established by the National Institute of Standards and Technology (NIST) in 2001, it has been adopted by the US government and other nations to protect confidential data and information. Pipeline algorithm is one of the algorithms which used to increase the speed and operational frequency. AES can be implemented on FPGA to improve security and to validate the results. Many authors are trying to achieve maximum frequency and low slices due to which speed will be increased to secure data.

Keywords - Advance Encryption Standard, Cryptography, Field Programmable Gate Array, Substitution box

I. INTRODUCTION

Cryptography is a method of transmitting and storing data so that only that person can read the information which is authenticated. Cryptography is used for secret data transfer that is in the form of symmetric and asymmetric key. In symmetric there is only one type of key i.e. common key for both encryption and decryption called as secret key cryptography. In Asymmetric key cryptography uses two types of keys for encryption and decryption. It uses public key for encryption and private key for decryption. This is called public key cryptography. Cryptography AES algorithm is Established by the National Institute of Standards and Technology (NIST) in 2001, it has been adopted by the US govt. and other countries to protect confidential data and information [1]. The criteria characterized by NIST for choosing AES fall into three regions i.e. Security, Cost and Implementation. Hardware and software both implementations are done in AES. In AES they are processing data bits of 128 bits by using cipher key of length 128, 192 or 256 bits [2]. Hardware implementation is used for better physical security and speed as compare to software implementation. High speed is also required for secure communication for one to other person. Fig 1: shows Some Symmetric key cryptographic algorithms.

![Fig 1: Types of cryptography algorithms](image)

After analysis of these entire algorithm AES was selected as best depending upon security, cost and implementation by NIST. There are some attacks or flaws on AES that are Brute-Force attacks, Unauthorized access, Side channel effects, Complexity due to S-box, DPA attacks etc. and can be improved by pipeline algorithm, Cryptography algorithm, Evolutionary algorithm etc. General assaults that were uncovered against concentrated rounds versions of Rijndael are Square Attack, Improved Square Attack, Impossible Differential Attack and Reversed Key Schedule Attack, however none of the assaults are possible in reality[9].

![Fig 2: Basic AES Process](image)
II. WORKING OF AES

Basic process of AES is that when we send some information from one person to other person that information in form of plain text and using key we encrypt that data called as cipher text while sending the information. The person who receives that information can decrypt cipher text to plain text using same secret key. Encryption algorithm consists of following steps:

A. Convert State array
Input given to the AES is in the form of text message and converts that message in the form of blocks. Then convert that block in the form of state array

B. Transformations and their inverse
   a) Add Round Key: In the state array there are rows and columns. Each column represents the byte. XOR operation is done in each round of state array and round key which gives us new state array with Add Round Key, as shown in fig. 3.

   b) Sub-bytes: The S-box (Substitution box) [7] is non-liner substitution of AES. The Sub-Byte is a non-linear byte substitution process, utilizing a substitution table (S-box), which is composed by augmentation, shown in Fig. 4.

   c) Shift Rows: State array rows are shifted cyclically. Fig. 5 below explains the shifting process. First row is not shifted by none byte. Second row is shifted by one byte; third row is shifted by two bytes and so on.
d) **Mix columns**: Fourth step after Shift Rows is Mix columns. Mix column is linear process where state matrix columns are independent on each other and new columns of state matrix are created by shuffling four bytes of each column, shown in fig. 6.

![Mix columns](image)

> Fig 6: Mix columns of AES [1]

C. Key Expansion

Key expansion is to increase the security so that no one hack the key. Key Expansion is done so that in each round uses a new round key.

Decryption is just inverse of encryption. Decryption is used to convert the ciphertext to plaintext using decryption algorithm. Fig. 7 shows the whole process of AES algorithm.

![AES algorithm](image)

> Fig 7: General AES block and rounds of AES [2]
III. LITERATURE REVIEW:

Mustafa. E. H et al describes that using standard symmetric encryption algorithm some attacks can be prevented. This paper still required some improvement for encryption of data and to prevent attacks [1]. The author Yu. W and S. Köse describes an implementation of modified lightweight in FPGA. AES algorithm such as mix columns, shift rows, substitution byte all are come in parallel manner. AES is more secure in hardware implementation and also have less cost and minimized hardware utilization [10]. The author proposed a false key based AES algorithm to prevent leakage of stored key from substitution box under CPA (correlation power analysis) attacks without significant power[11]. The authors Bri. S and Oukili. S describe the high speed for efficient AES implementation. After analysis conclude that it consumes low area and gives better throughput using pipelining technique [2]. For Unauthorized access speed of AES must be high and have strong cryptography. Authors S. Bri and S. Oukili used 5-stage pipeline design for maximum frequency and used to increase speed [12]. This paper explores the switching Capabilities of BRAM and by changing duty cycle of input clock. Work done by this author is by using pipeline design and saved area by 59.01% [13]. The non-pipelined plans [14–16] concentrated on the reduced plan and are more proficient when AES works in the input mode, where the new plaintext should be first included with the already encoded cipher text. The main outline choice is the quantity of pipelining stages in each figure round. In [17–21], different quantities of stages were investigated, prompting diverse throughput. The second outline choice is the usage of S-box in Sub-Bytes. The composite-field calculation based S-box accomplished littler territory contrasted with the LUT based usage in ASIC executions [22, 23]. For FPGA with four-input LUTs, the composite-field usage can likewise decrease zone. Nonetheless, for late FPGA with six-input LUTs, the circumstance is changed, and the LUT-based usage of S-box has littler region and lessened deferral. We will see the related exploratory outcomes in the accompanying area. In this way we pick the main answer for actualize S-box. It was specified in [19, 20, 21] that the LUT in the first arrangement constrains additionally parcelling of pipelining stages in each round of AES. This restriction is settled in the display paper by partitioning S-box into sub-boxes. J.-Y. Park et al. [24] chipped away at strategies for viable white-box cryptography. In this assaults are considerably more grounded then the discovery display. The fundamental confinement of this plan was switching of look into table which is quick and solid if there should arise an occurrence of the white box and considered for the future research. Gaspar et al. [25] chipped away at effective AES S-boxes execution for non-usable FPGA. They proposed a proficient strategy for the execution of AES byte substitution work (S-box). The proposed an answer which requires less space and is speedier than the one executing entire S-encloses the rationale region. The principle confinement of this plan was FPGA can’t be utilized for the low battery purposes, Selimis et al. [26] chipped away at applying low power procedure in AES MixColumn or InvMixColumn transformation. They research the utilization of low power assets which expands the security needs and effectiveness. Along these lines, the information ways which are of no utilization for the framework are deactivate in parallel manner. AES is more secure in hardware implementation and also have less cost and minimized hardware utilization. Mustafa.E.H et al describes that using standard symmetric encryption algorithm some attacks can be prevented. This paper still required some improvement for encryption of data and to prevent attacks [1]. The author Yu. W and S. Köse describes an implementation of modified lightweight in FPGA. AES algorithm such as mix columns, shift rows, substitution byte all are come in parallel manner. AES is more secure in hardware implementation and also have less cost and minimized hardware utilization. From Table I, we can conclude that using pipelining method we can achieve high throughput in which key expansion and other steps will be done in parallel processing. We can use composite field arithmetic, direct mapping from LUT or field usage can likewise decrease zone. Nonetheless, for late FPGA with six-input LUTs, the circumstance is changed, and the LUT-based usage of S-box has littler region and lessened deferral. We will see the related exploratory outcomes in the accompanying area. In this way we pick the main answer for actualize S-box. It was specified in [19, 20, 21] that the LUT in the first arrangement constrains additionally parcelling of pipelining stages in each round of AES. This restriction is settled in the display paper by partitioning S-box into sub-boxes. J.-Y. Park et al. [24] chipped away at strategies for viable white-box cryptography. In this assaults are considerably more grounded than the discovery display. The fundamental confinement of this plan was switching of look into table which is quick and solid if there should arise an occurrence of the white box and considered for the future research. Gaspar et al. [25] chipped away at effective AES S-boxes execution for non-usable FPGA. They proposed a proficient strategy for the execution of AES byte substitution work (S-box). The proposed an answer which requires less space and is speedier than the one executing entire S-encloses the rationale region. The principle confinement of this plan was FPGA can’t be utilized for the low battery purposes, Selimis et al. [26] chipped away at applying low power procedure in AES MixColumn or InvMixColumn transformation. They research the utilization of low power assets which expands the security needs and effectiveness. Along these lines, the information ways which are of no utilization for the framework are deactivate in parallel manner. AES is more secure in hardware implementation and also have less cost and minimized hardware utilization. Mustafa.E.H et al describes that using standard symmetric encryption algorithm some attacks can be prevented. This paper still required some improvement for encryption of data and to prevent attacks [1]. The author Yu. W and S. Köse describes an implementation of modified lightweight in FPGA. AES algorithm such as mix columns, shift rows, substitution byte all are come in parallel manner. AES is more secure in hardware implementation and also have less cost and minimized hardware utilization.

<table>
<thead>
<tr>
<th>Author</th>
<th>Tech. used</th>
<th>Devices</th>
<th>Output Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wang et al [19]</td>
<td>GP-method</td>
<td>Virtex</td>
<td>3927.6 (GHz)</td>
</tr>
<tr>
<td>Yi Wang et al [15]</td>
<td>Pipeline</td>
<td>Virtex XC6V LX24</td>
<td>6784</td>
</tr>
<tr>
<td>Liu et al [12]</td>
<td>Pipeline</td>
<td>Virtex 5</td>
<td>3121</td>
</tr>
<tr>
<td>Sharma et al [22]</td>
<td>Pipeline</td>
<td>Virtex-5</td>
<td>5759</td>
</tr>
<tr>
<td>Liu et al [24]</td>
<td>Pipeline</td>
<td>Virtex-4</td>
<td>5900</td>
</tr>
<tr>
<td>Wang and Ha [21]</td>
<td>Pipeline</td>
<td>Virtex-4</td>
<td>5613</td>
</tr>
</tbody>
</table>

Table I. Comparison
REFERENCES


