International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

IMPLEMENTATION FPGA BASED IMPLEMENTATION OF 128-BIT AES ALGORITHM USING VHDL

ATTAR MEHARAJ BANU[1], SYED NOORULLAH[2]
[1]DECS 202T1D3801, ASHOKA WOMENS ENGINEERING COLLEGE, KURNOOL, A.P.
[2]Assistant Professor – ASHOKA WOMENS ENGINEERING COLLEGE, KURNOOL, A.P

ABSTRACT
In current world of computations, data encryption is of prominent importance. Many algorithms were developed for data encryption and decryption to prevent hacking. The Advanced Encryption Standard (AES) is one of the data encryption techniques. Two famous kinds of hardware implementation techniques are pipelining and loop-unrolling techniques. In pipelining, registers are inserted between each combinational processing element so that each input data block can be processed simultaneously in each processing element. In this work, a pipelined implementation of AES encryption algorithm is developed. The number of rounds of AES-128 encryption is 10 and an architecture implementing this cipher is called fully pipelined, when all data blocks of 10 rounds can be processed simultaneously. In the loop-unrolling technique one or multiple rounds of the algorithm are processed in the same clock cycle. Here only one round of the algorithm is implemented as a combinational processing element and a data register is also used to store the result obtained in the previous clock cycle. In this work, a128-bit AES is implemented and for each round of AES encryption, a different sub-key is used as the round key, which is produced by the key schedule algorithm based on the loop-unrolled technique, to produce the required sub-key for each round is done. The no of I/O, Slices in the proposed work are 386, 229 respectively with Spartan6 fpga with a minimum period of 5.813ns,maximum Frequency of 172.031MHz, minimum input arrival time before clock of 4.823ns,maximum output required time after clock of 5.588ns, throughput of 2.2Gbps. The no of I/O, Slices are 387, 264 respectively with Artix7 fpga with a minimum period of 3.397ns, maximum Frequency of 294.366MHz, minimum input arrival time before clock of 1.649ns, maximum output required time after clock of 1.669ns, throughput of 3.77Gbps.
Key Words - 128-bit AES, Encryption, Decryption, Pipelining, Loop-Unrolling.

1. Introduction

The Security is one of the biggest concerns in the developing world. It is important to ensure a safe transfer of information between communicating parties, protecting them from attacks. Many standards and developed encryption protocols are avail- able as resources and are used based on the requirements. In this thesis, we propose a customized encryption algorithm and an authentication scheme to safely transfer information. The algorithm is a variation of Advanced Encryption Standard (AES) and is carried out between multiple devices. AES uses only one private key (sym- metric key) to encrypt the data. Cryptography is the science of information security. Cryptography includes techniques such as microdots, merging words with images and other ways to hide information in storage or transit. Cryptography is most often associated with scrambling plaintext (ordinary text, sometimes referred to as clear text) into ciphertext (a process called encryption), then back again (known as decryption). A Cryptographic system that uses two keys – a public key known to everyone and a private or secret key known only to the recipient of the message. Individuals who practice this field are known as cryptographers. Symmetric cryptography can be split into block ciphers and stream ciphers. Fig. 1depicts the operational differences of a block cipher and stream cipher. Bits are encrypted individually in stream ciphers. A bit from a key stream is added to a plain text to achieve this. An entire block of plaintext is encrypted with the same key in block ciphers.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

The encryption of the plaintexts in any given block is related to the encryption of another plaintext of the same block. Practically, the majority of block ciphers have a block length of 128 bits(16 bytes) like Advanced Encryption Standard (AES), or a length of 64 bits (8 bytes) like Data Encryption Standard (DES) or Triple DES (3DES) algorithm. AES and DES are examples of the symmetric algorithm. But, these are some of the many symmetric algorithms. Hundreds of algorithms have been proposed over the years. Even though some of these proposed algorithms were deemed insecure, many other cryptographically secure ones exist in the market.

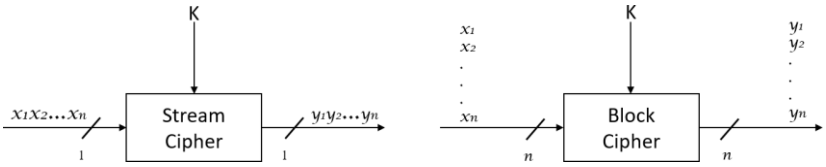

Figure 1: Principles of encrypting n bits with stream and block ciphers

Advanced Encryption Standard (AES) is the prominent choice of algorithm for en- cryption which will be discussed further on. The AES cipher is similar to block cipher Rijndael. Rijndael with a block length of 128 bits is known as the AES algorithm.

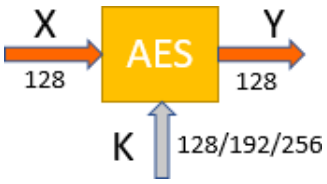

Figure 2: Basic AES block

AES has survived decades of brute force attacks with its three high key lengths of 128, 192 and 256 bits shown in Fig. 2. Any analytical attacks with a reasonable chance of success are still unknown. AES was a result of an open competition. There were four other potential strong finalist algorithms, Mars, RC6, Serpent, and Twofish. These were cryptographically strong and quite fast, particularly in software. Mars, Serpent, and Twofish are royalty-free. Triple DES often denoted as 3DES is an alternate to AES or the AES finalist algorithms. Three subsequent DES encryption with different keys are present in 3DES. 3DES does not support software as efficiently as it supports hardware. Financial applications and protection of biometric information in electronic passports use 3DES. The DES is made much more resistant against exhaustive key searches by this simple modification.

2. Literature Survey

Palm The data or information may be in any form such as text, audio, image, video, or others, making the data unreadable to attackers (Suresh & Ajai, 2016). Data encryption (Mishra et al., 2019) is done by encryption methods and the reverse of encryption is defined as decryption. To replace the data encryption standard (DES) for advanced technologies implementation, the advanced encryption standard (AES) method is proposed. There are special optimization techniques also followed in this hardware implementation, like lookup tables, pipelining (Swetha et al., 2017), etc. Alternatively, FPGA is mainly known for its security, high speed, flexibility, and low maintenance. These implementations are mainly used in block ciphers (Pandey et al., 2018). The AES algorithm analyzed in this review has been implemented in VLSI architecture to provide security and fast processing (Hameed et al., 2018). The AES algorithm is a symmetric encryption algorithm of cryptography. It is a block cipher algorithm discovered by Joan Daemen and Vincent Rijimen (Landge & Mishra, 2016). The AES 128-bit functions on a 4 4 matrix form array of array of bytes and size of block referred as a state. In this state, the encryption and decryption process is defined

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

in this state (Abdullah, 2017). This process is iterated in a number of rounds based on AES algorithm key length (number of bits in key) and mainly is used to transmit clear data into unclear data. Ten rounds of repetition used in 128-bit keys, for 192 -bit keys, 12 rounds, and for 256-bit keys, 14 rounds are used by AES based on key size. Multiple rounds of AES are used in hardware implementation of VLSI architecture. Each round is comprised of sub-bytes, shift rows, mix columns, and add round key (Siddesh & Shruthi, 2017).

3. Implementation

The proposed system of implementing 128-bit AES algorithm using HDL. Fig. 3 describes the structure of AES. It comprises of three layers, Key Addition layer, Byte Substitution Layer and Diffusion Layers (Shiftrow and Mixcolumn) respectively. Each layer manipulates 128 bits of the data. Before going through the process of how AES converts the plain text into cipher text, one has to know the standard properties exhibited by the AES fields, which are used in every layer of the algorithm. AES implementations are not limited to software applications. It is also used in hardware implementations such as FPGA's

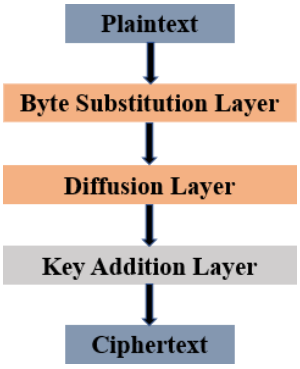

Figure 3. Structure of AES.

An overview of a pipelined implementation of AES encryption algorithm is depicted in the following figure 4, where the round-i depicts the i^{th} round of AES encryption algorithm.

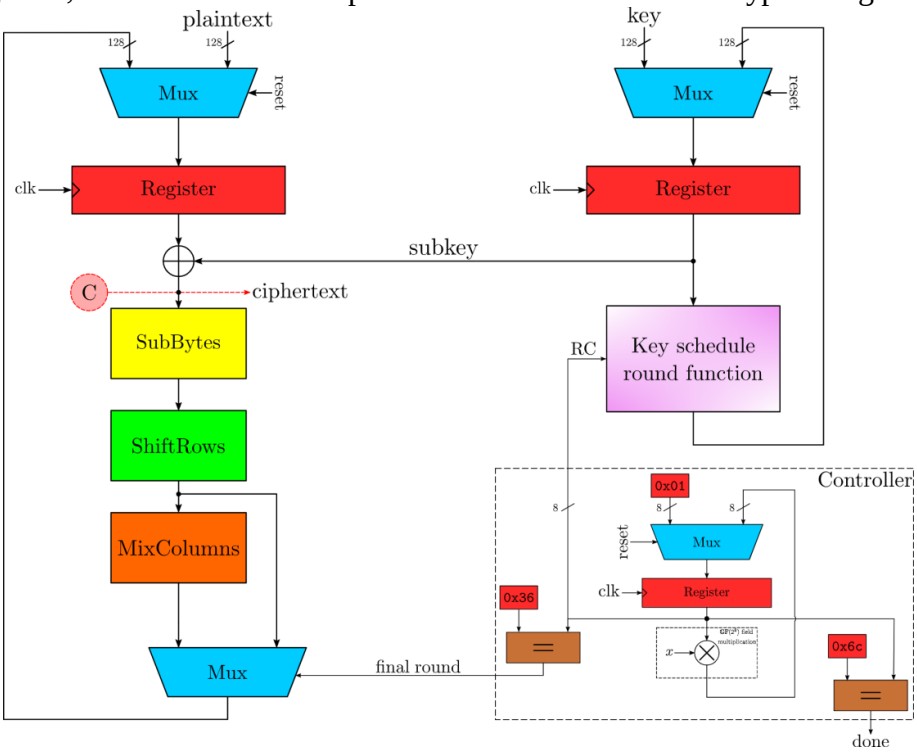

Figure 4. Pipelined implementation of AES encryption algorithm.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

The number of rounds of AES-128 encryption is 10 and an architecture implementing this cipher is called fully pipelined, when all data blocks of 10 rounds can be processed simultaneously. For a fully pipelined implementation of AES-128 ten 128-bit data registers are needed. The more data block we want to process simultaneously, the more registers and therefore, the more area we need for implementation. In contrast to pipelining, in a loop-unrolling technique one or multiple rounds of the algorithm are processed in the same clock cycle. In the smallest case of a loop-unrolled implementation of AES which is depicted in the following figure 5, only one round of the algorithm is implemented as a combinational processing element and a data register is also used to store the result obtained in the previous clock cycle.

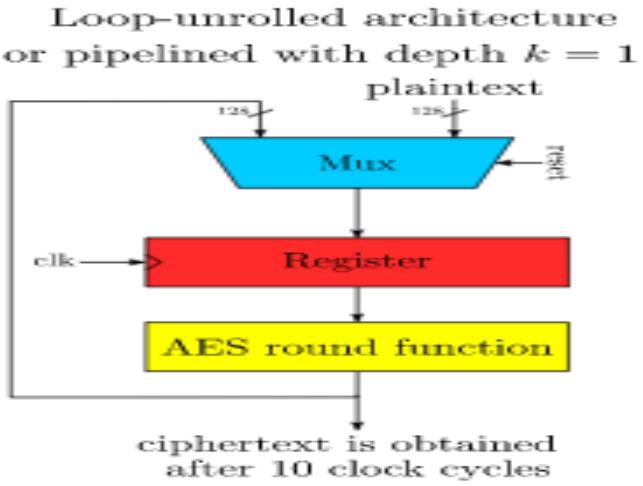

Figure 5 Loop-Unrolled architecture of AES encryption algorithm.

For each round of AES encryption, a different sub-key is used as the round key, which is produced by the keyschedule algorithm. The following figure 6 represents one round of keyschedule algorithm.

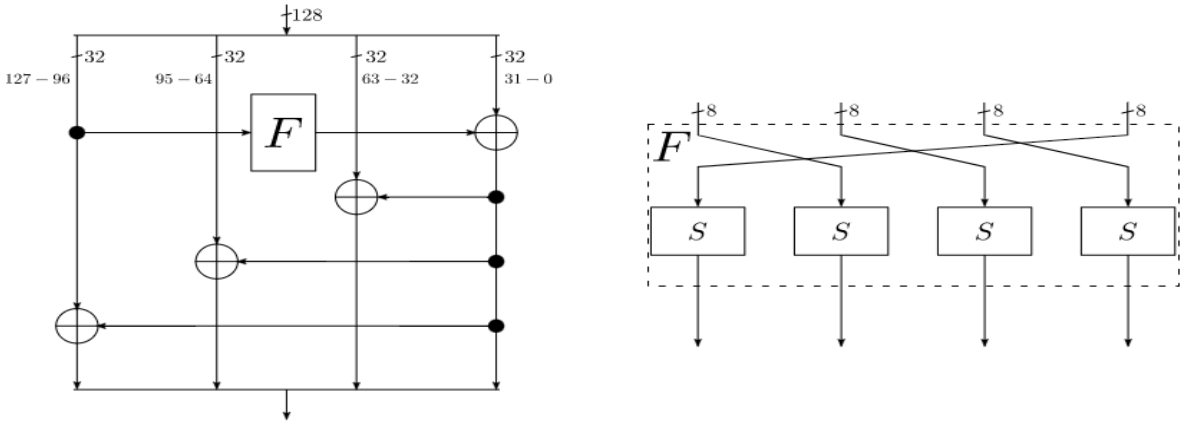

Figure 6. Process of one round of key-schedule algorithm.

If someone wants to use a fixed key, it is preferable to calculate all sub-keys once and use a lookup table to store sub-keys, instead of implementing keyschedule algorithm and recalculate the sub-keys frequently. This strategy is especially suitable for software implementations, where memory is not as constrained as hardware implementations. The keyschedule is implemented based on the loop-unrolled technique to calculate the sub-keys on the fly. In other words, there is a dedicated part implementing keyschedule algorithm based on the loop-unrolled technique, to produce the required sub-key for each round on the fly. It is shown in below figure 7.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

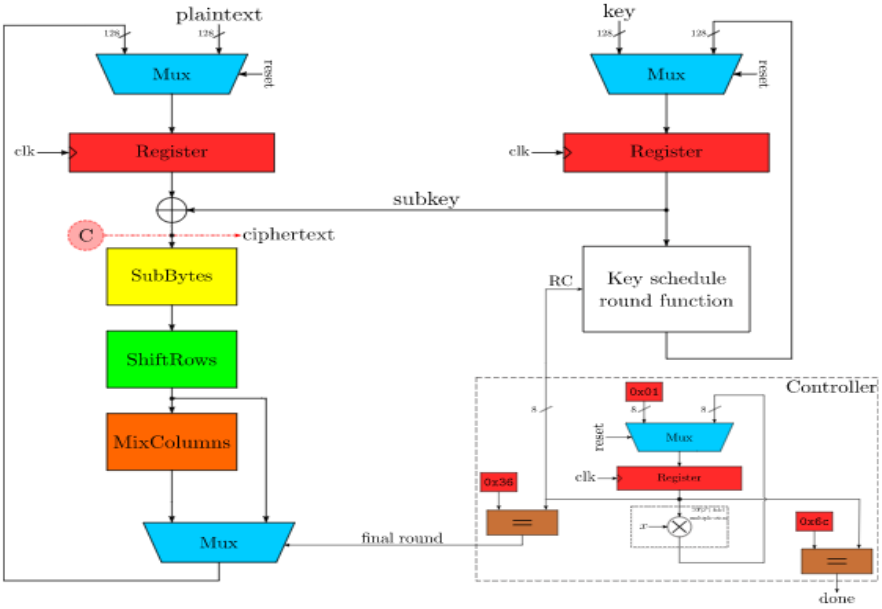

Figure 7 Architecure view of algorithm implemented.

The byte value in AES is represented as a set of bits (0 or 1) and is represented as the collection of bits separated by comma as {b7, b6, b5, b4, b3, b2, b1, b0}. These bytes are interpreted as finite field elements using polynomial representation as

b7x7 + b6x6+ b5x5+ b4x4 + b3x3+ b2x2+ b1x+ b0 (1)

All the operations performed in AES are modulo-2 operations. These Operations are not as the same operations used in general Number System. The basic operations on which the entire math of the AES algorithm is based are Addition, Multiplication.

4. RESULTS

In order to implement 128-bit AES algorithm, keys we used are as listed below for plaintext: x"2a179373117e3de9969f402ee2bec16b",

key: x"3c4fcf098815f7aba6d2ae2816157e2b" ciphertext: x"97ef6624f3ca9ea860367a0db47bd73a", The timing waveforms are shown in below figures

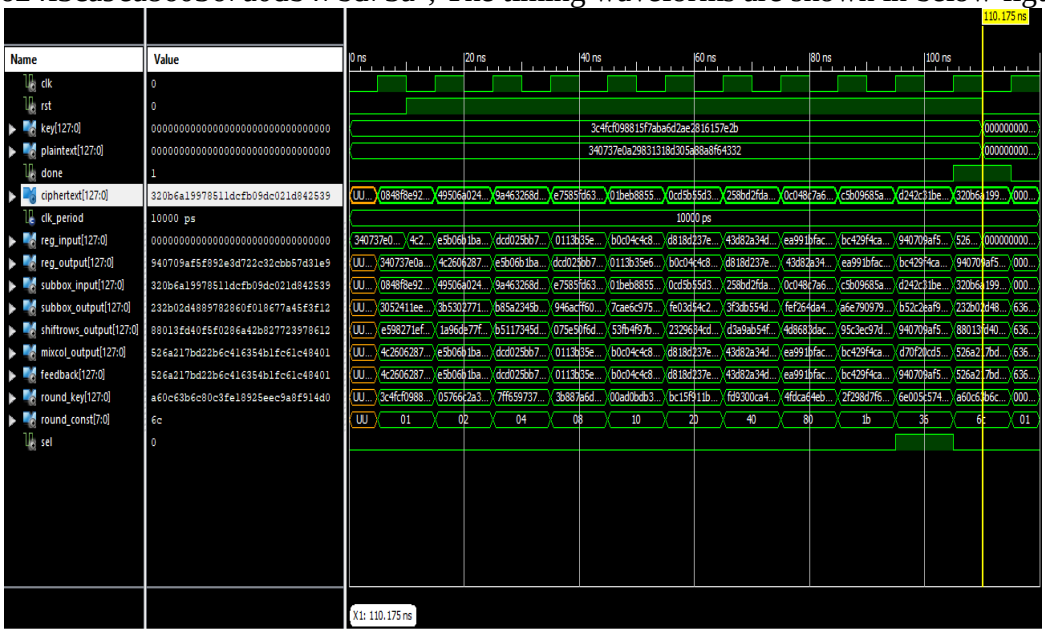

Figure 8. AES_ENC Simulated waveform

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

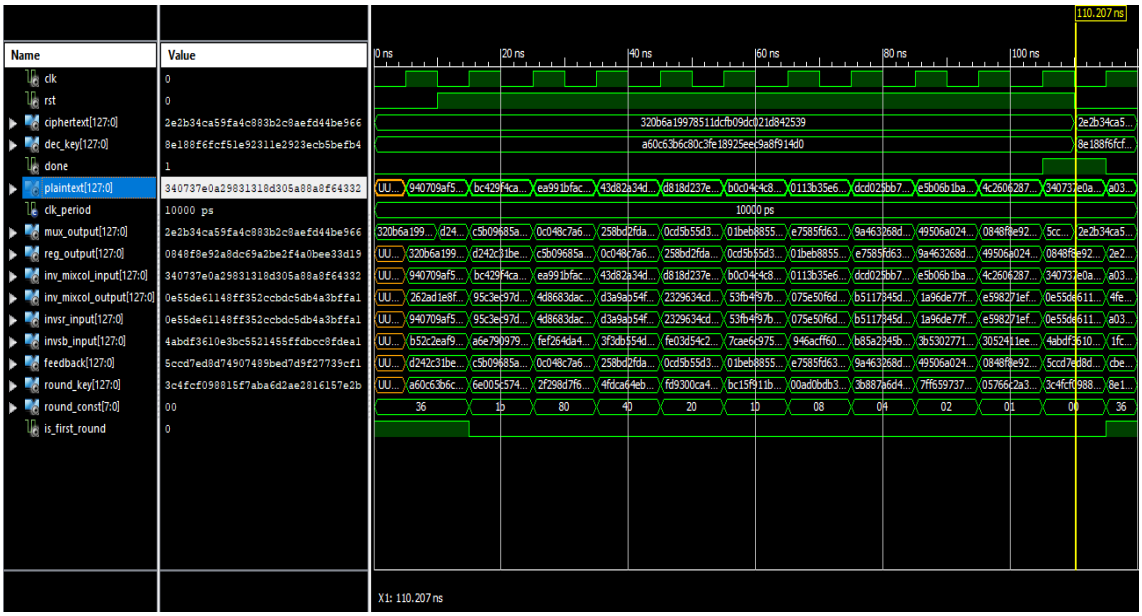

Figure 9 AES_DEC Simulated waveform

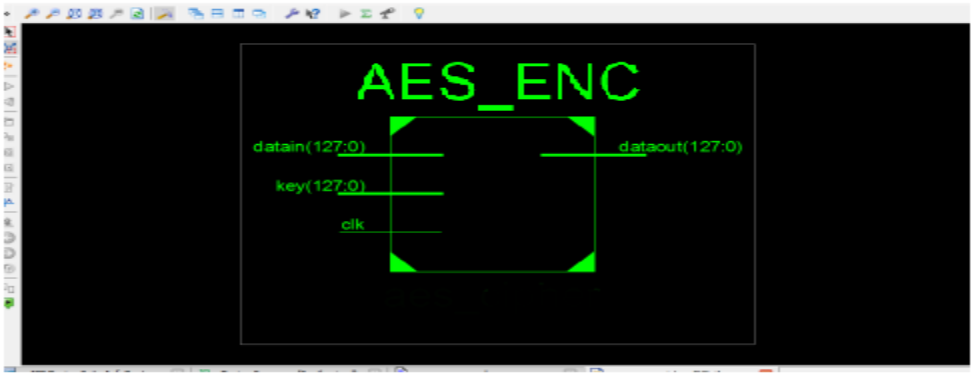

Figure 10. RTL view of AES_ENC.

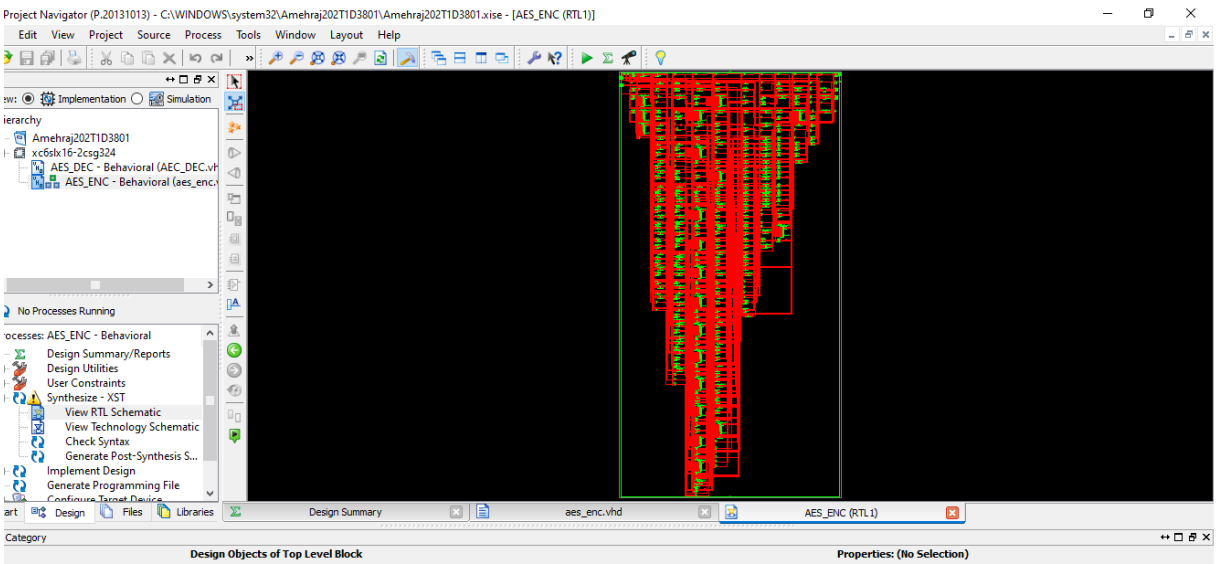

Figure 11. RTL schematic view of AES_ENC.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

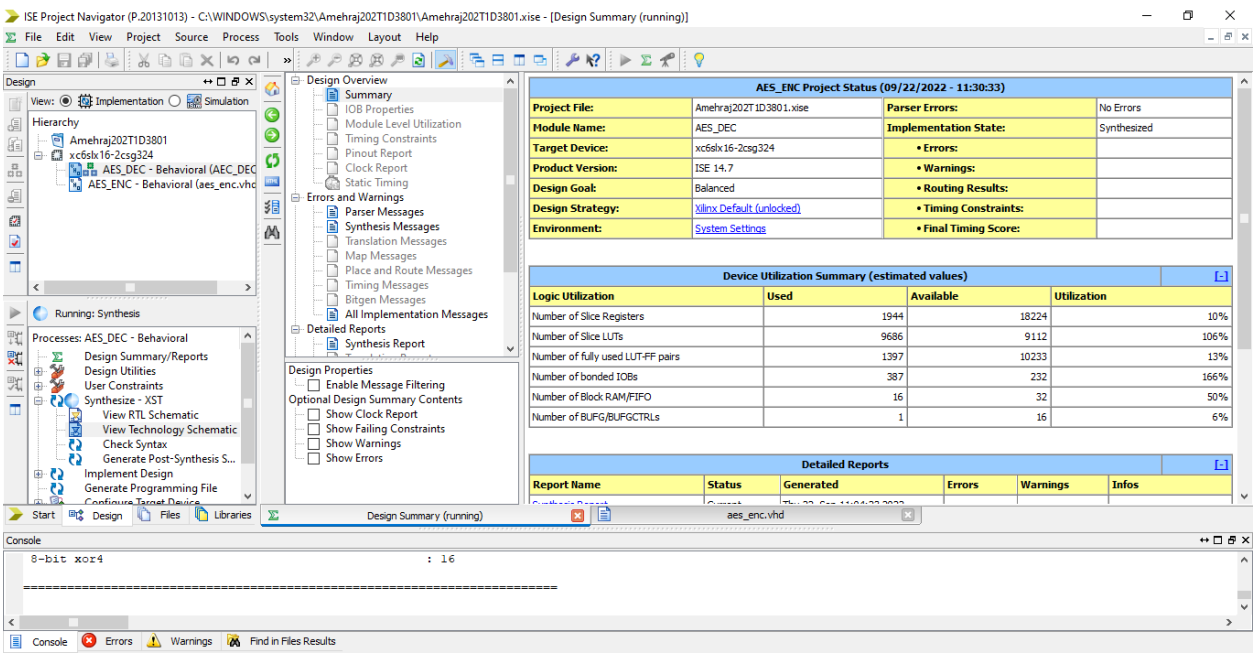

Figure 12. Design Summary of AES_ENC.

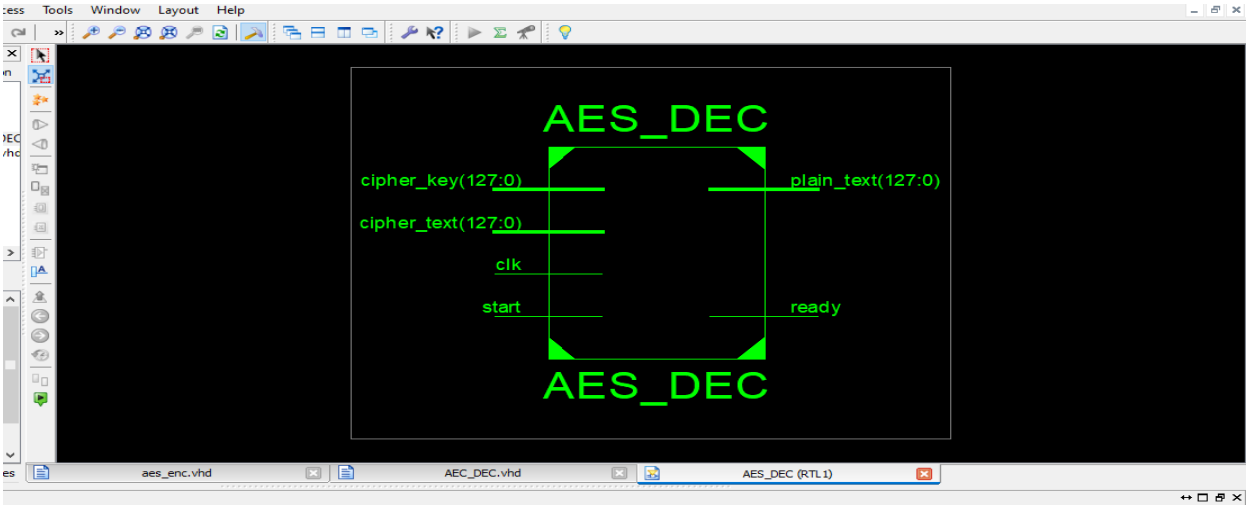

Figure 13. RTL VIEW of AES_DEC.

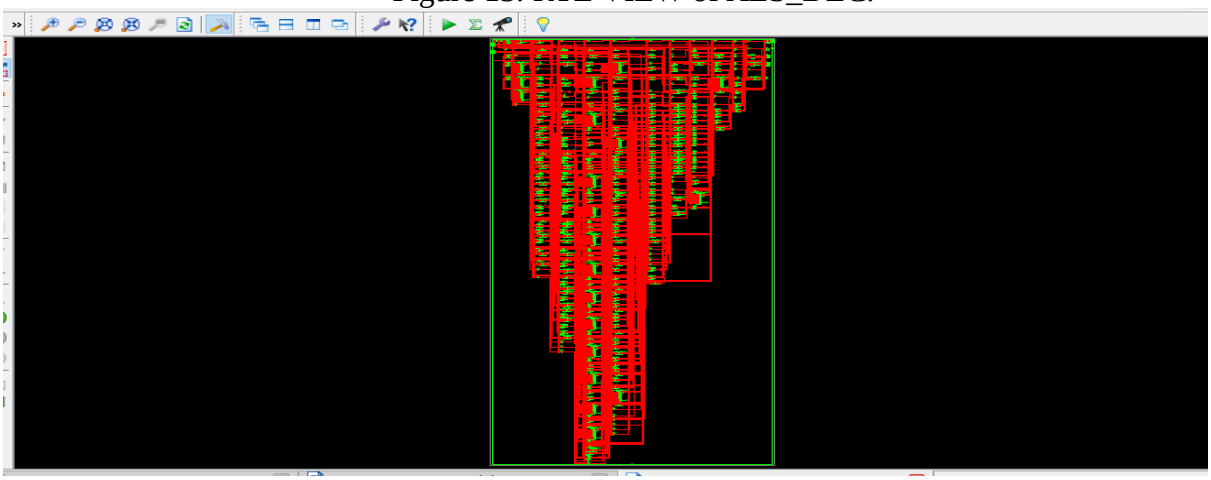

Figure 14. RTL schematic VIEW of AES_DEC.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

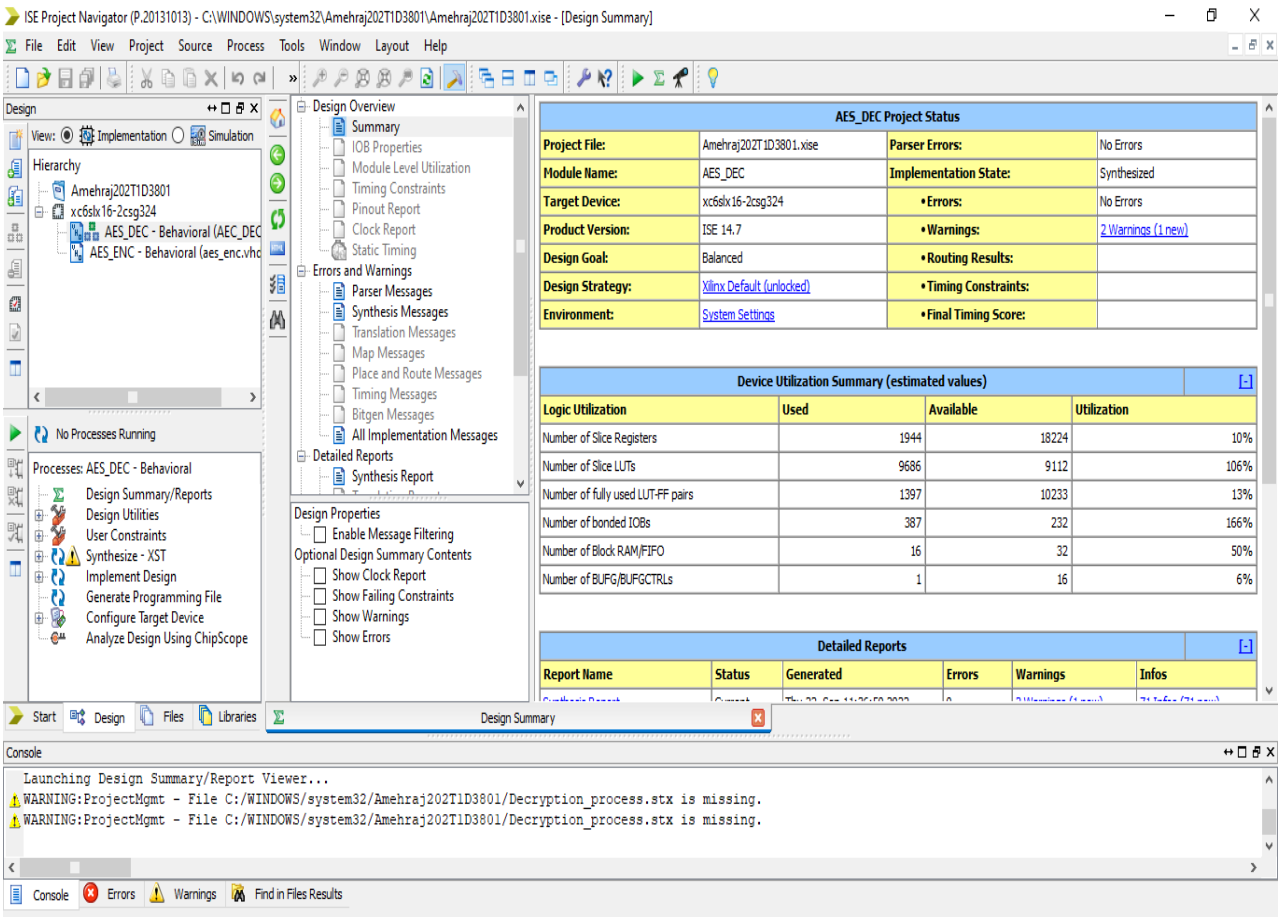

Figure 15 Design Summary of AES_DEC.

In the reference [10] the number of slice register used is 954. In reference [11] researchers used two FPGA for implementing the AES design, for Virtex-5 FPGA the number of slice register used is 255. Below tables represent the comparison of existing system and proposed system results.

Table 1: Existing System Results.

Resources	Used	Available
SR	3987	126800
LUT	4115	63400
I/O	269	300
BUFG	1	32

As seen in table-1, the existing work requires 4115 LUTs for the implementation of the AES algorithm. But for proposed work shown table-2 it is 1397 LUTs which effective utilization.

Table2: Proposed System.

Resources	Used	Available
SR	264	93296
LUT	1104	46648
I/O	387	408
BUFG	1	16

From the two tables the proposed system is occupying lesser area when compared to existing system 50% of area. With this result we can infer that the proposed one is much better than the existing system.

International Journal of Engineering Technology and Management Sciences
Website: ijetms.in Issue: 1 Volume No.7 January - February – 2023
DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

Figure 16. Test Bench wave form of AES_ENC

In the above test bench the encryption of plaintext is x"2a179373117e3de9969f402ee2bec16b", under the key is x"3c4fcf098815f7aba6d2ae2816157e2b", which has to produce the cipher text as x"97ef6624f3ca9ea860367a0db47bd73a".

Figure 17. Test Bench wave form of AES_DEC

In the above test bench the decryption of cipher text 32e2b34c959f94c883b2c8aefd44be9ee inputted is obtained as 8e188f6fcf51e92311e2923ecb5befb4.

DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

5. Conclusion

The 128-bit AES encoder, decoder are successfully implemented and tested for performance comparison. Thus the hardware architectures are much more dependent on the varying bit size than software implementation. This effect is enhanced with the employment of several cores in FPGA with smaller bit sizes. Although an additional core on FPGA might slightly decrease the maximum clock speed, it is overcome by the additional computational power provided by this extra point processor. The no of I/O, Slices in the proposed work are 386, 229 respectively with Spartan6 fpga with a minimum period of 5.813ns,maximum Frequency of 172.031MHz, minimum input arrival time before clock of 4.823ns,maximum output required time after clock of 5.588ns, throughput of 2.2Gbps. The no of I/O, Slices are 387, 264 respectively with Artix7 fpga with a minimum period of 3.397ns, maximum Frequency of 294.366MHz, minimum input arrival time before clock of 1.649ns, maximum output required time after clock of 1.669ns, throughput of 3.77Gbps. The algorithm can be implemented securely and efficiently in a wide variety of platforms and applications (e.g., 8-bit processors, ATM networks, voice & satellite communications, HDTV, B-ISDN, etc.). The algorithm can be implemented as a stream cipher, message authentication code (MAC) generator, pseudorandom number generator, hashing algorithm, etc.

References

1. Suresh, A., & Ajai, A. R. (2016). VLSI implementation of text to image encryption algorithm based on private key encryption. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 4879–4881). IEEE. https://doi. org/10.1109/ICEEOT.2016.7755647.
2. Mishra, Z., Ramu, G., & Acharya, B. (2019). Hight speed low area VLSI architecture for LEA encryption algorithm. In Proceedings of the third international conference on microelectronics, computing and communication systems (pp. 155–160). Springer
3. Swetha, P. B., Sonti, V. K., & Murali, A. (2017). VLSI design for efficient RSD-Based ECC processor using Karatsuba algorithm. International Journal of Engineering & Technology, 7(1.5), 164–169. https://doi.org/10.14419/ijet.v7i1.5.9140.
4. Pandey, J., Gurawa, A., Nehra, H., & Karmakar, A. (2016). An efficient VLSI architecture for data encryption standard and its FPGA implementation. In 2016 International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA) (pp. 1–5). IEEE. https://doi.org/10.1109/VLSI-SATA.2016.7593054
5. Hameed, M. E., Ibrahim, M. M., & Abd Manap, N. (2018). Review on improvement of advanced encryption standard (AES) algorithm based on time execution, differential cryptanalysis and level of security. Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 10(1), 139–145.
6. Landge, I. A. G., & Mishra, B. (2016). Iterative architecture AES for secure VLSI based system design. In 2016 Symposium on Colossal Data Analysis and Networking (CDAN) (pp. 1–4). IEEE. https://doi.org/10.1109/CDAN.2016.7570938
7.Abdullah, A. M. (2017). Advanced encryption standard (AES) algorithm to encrypt and decrypt data. Cryptography and Network Security, 16
8.Siddesh, G., & Shruthi, J. (2017). AN EFFICIENT VLSI ARCHITECTURE FOR AES AND its FPGA IMPLEMENTATION. International Research Journal of Engineering and Technology, 4(6).
9. Keshava Kumar, “A Design Implementation and Comparative Analysis of Advanced Encryption Standard” 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) Amity University, Noida, India. June 4-5, 2020.
10. K. P. Singh, and S. Dod. "An Efficient Hardware Design and Implementation of Advanced Encryption Standard (AES) Algorithm." IACR Cryptology ePrint Archive 2016 (2016): 789.
11. U. Farooq r, and M. F. Aslam. "Comparative analysis of different AES implementation

DOI:10.46647/ijetms.2023.v07i01.019 ISSN: 2581-4621

techniques for efficient resource usage and better performance of an FPGA." Journal of King Saud University, Computer and Information Sciences 29, no. 3 (2017): 295-302.