PAPER NAME

VLSI IMPLEMENTATION IN HSM BASED ON AES METHOD.pdf

WORD COUNT

3696 Words

CHARACTER COUNT

20380 Characters

PAGE COUNT

36 Pages

FILE SIZE

2.4MB

SUBMISSION DATE

Apr 15, 2024 7:14 PM GMT+5:30

REPORT DATE

Apr 15, 2024 7:15 PM GMT+5:30

20% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 7% Internet database
- 3% Publications database
- Crossref database
- Crossref Posted Content database
- 19% Submitted Works database

Excluded from Similarity Report

- Bibliographic material

PROJECT REPORT

VLSI IMPLEMENTATION IN HARDWARE SECURITYMODULE BASED ON AES ENCRYPTION METHOD

20ECPJ701- PROJECT PHASE - I

Submitted by

AKASH A (412520106007)

JAIGANESH P (412520106053)

MAHIZHAN M (412520106085)

in partial fulfillment for award of the degreeof

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

SRI SAI RAM ENGINEERING COLLEGE

(An Autonomous Institution; Affiliated to Anna University, Chennai - 600 025)

ANNA UNIVERSITY :: CHENNAI 600 025

DECEMBER 2023

SRI SAIRAM ENGINEERING COLLEGE

(An Autonomous Institution; Affiliated to Anna University, Chennai -600 025)

BONAFIDE CERTIFICATE

Certified that this project report on **"VLSI IMPLEMENTATION IN HARDWARE SECURITY MODULE BASED ON AES ENCRYPTION METHOD"** is the bonafide work of **"AKASH A (412520106007), JAIGANESH P (412520106053) and MAHIZHAN M (412520106085)"** who carried out the **20ECPJ701- PROJECT PHASE - I** Work under my supervision.

SIGNATURE

Dr J Raja

HEAD OF THE DEPARTMENT

Department of Electronics and Communication Engineering,
Sri Sairam Engineering College,
(Autonomous) Chennai - 600 044

SIGNATURE

Mr. K Srinivasan

SUPERVISOR

Associate Professor, Department of Electronics and Communication Engineering,
Sri Sairam Engineering College,
(Autonomous) Chennai - 600 044

Submitted for VIVA-VOCE EXAMINATION held on:

INTERNAL EXAMINER

EXTERNAL EXAMINER

ABSTRACT

As ecommerce and internet applications have grown in popularity, cryptography has become increasingly vital for data security and integrity. However, it is often overlooked in many circumstances due to the additional memory and other needs required for implementation. The primary objective of this project is to employ Verilog to create aes algorithms. Cryptographic algorithms are implemented to safeguard data, such as electronics. AES parallel design can decrease the latency associated with each encryption round. Using the suggested high- performance design, we reduce power consumption and critical path latency. Although security concerns have grown in importance over time, the initiative's primary purpose is to maximize data flow. The employment of encryption and decryption techniques inside VLSI has lately risen since cryptography can transform plaintext to cipher and vice versa. The most current advances in cryptography technology will be utilized in the hardware security module by simultaneously developing a large number of HDL modules. The primary goal is to send and receive data securely while preventing data from being hacked, as well as to increase the performance of a certain parameter. It is crucial to remember that all system blocks will securely handle digital data because methods of encryption operate in digital environments.

JUSTIFICATION FOR SDG & SAP

SDG No: 9

: Industry, Innovation and Infrastructure

SAP No: SAP090C

This is implementing effective strategies to promote research and development initiatives is crucial for the sustainable growth of emerging economies.

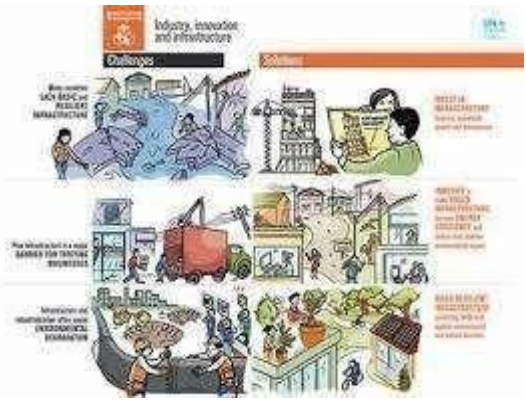


TABLE OF CONTENTS

List of Figures

CHAPTER-1
INTRODUCTION

1.1 OBJECTIVE:

Security is becoming a crucial component of contemporary warfare Cyberattacks account for the majority of attacks. Cyberattack against Systems, Aircraft, Drones, and other Military Equipment, etc.

The initiative's main objective is to increase data flow, but as time goes on, security issues have taken on more significance. Since cryptography can transform plaintext into cypher and vice versa, its use inside VLSI has lately increased. A large number of HDL modules will be simultaneously written in order to implement the most recent advancements in cryptography technology in the hardware security module. The major goal is to send and receive data securely without allowing data to be hacked, as well as to boost the efficiency of a certain parameter. Verilog code was used as the technique in this system. Analog and digital platforms are offered by Xilinx to support the design of both analogue and digital circuits. Interesting fact: Any encryption algorithm will function.

1.2 Motivation:

The goal of the project is to increase information security. To that end, we would use our adaptation of AES encryption to strengthen the hardware security module's security in an environment centered around network-centric warfare. Additionally, to strengthen the security of defense technologies from cyber-attacks.

1.3 Relevance of the project:

Security is becoming a crucial component of contemporary warfare. Cyberattacks account for the majority of attacks. Cyberattack against Systems, Aircraft, Drones, and other Military Equipment, etc. These seriously harm a nation's collateral. In order to achieve our goal of improving system security, we would use our adaptation of AES encryption to strengthen the security of the hardware security module, which is an HSM (Hardware Security Module) chip implanted in the system controller.

CHAPTER-2

LITERATURE SURVEY

S.NO	TITLE	AUTHOR	PUBLISHED IN	INFERENCE
1	JESIT-15	K. Kalaiselvi, H. Mangalam	**2022**	The implementation of this algorithm has shown, that with the increasing demand for secure data transmission, processes like key expansion are crucial. By incorporating this technique into the AES algorithm, significant advancements in both power efficiency and data processing speed can be achieved
2	VLSI implementarion of AES Algorithm	Surabh Kumar	**2022**	This paper offers Historically, cryptography was limited to the use of secret keys for encryption and decryption; currently, it encompasses a variety of operations, such symmetric-key and asymmetric-key encipherment.

3	VLSI Implementation of Cryptographic Algorithms & Techniques	Favin Gauravi	**2021**	The movement and flow of information has risen dramatically over time, as have the security risks connected to it. Recent advances in VLSI technology have allowed the usage of encryption and decryption techniques in cryptography, permitting the encoding and conversion of plaintext into cipher text and vice versa.

CHAPTER-3

EXISTING AND PROPOSED SYSTEM

3.1 EXISTING SYSTEM:

The volume of data and its transmission has expanded dramatically over the years, as have the security challenges that accompany it. Cryptography has advanced in recent years, with the advent of Encryption and Decryption procedures, which allow for the translation and conversion of plaintext into cypher text and vice versa. This study reviews many elements of VLSI's encryption and decryption implementations. Finally, using this overview, a fundamental grasp of several VLSI approaches for encryption and decryption may be examined and used. This is the current system of the project.

.

3.2 PROPOSED SYSTEM:

Verilog coding is the method used in this system. We would examine the most recent version of AES encryption first, modify it to reach the algorithm's optimum efficiency, and then implement it in a Hardware Security Module. After that We would implement into a Controller for further study of the Security of the Controller System. This is our proposed methodology of our Project.

CHAPTER-4

REQUIREMENT SPECIFICATION

4.1 Hardware requirements:

4.1.1 Hardware security module:

Strong authentication, digital signature encryption and decryption, digital key management, and other cryptographic functions are all performed by hardware security modules (HSMs), which are actual computer devices. Historically, a computer or network server may be directly connected to these modules via an external device or as a plug-in card. With one or more safe crypto processor chips, a hardware security module is assembled.

HSMs with safety-proof features include safety resistance, which makes intrusion difficult but does not render the HSM ineffectual, safety responsiveness, which removes keys upon detecting tampering, and visible indicators of manipulation, logging, and alarms. Each module contains either one or more secure crypto processing chips to guard against tampering and bus probing, or a mix of chips protected by counterfeit evident, tamper resistant, or counterfeit responsive packaging.

Figure 4.1

Uses of HSM:

Any application that makes use of digital keys can benefit from using a hardware security module. Since the keys are usually highly valuable, the owner would lose a lot of money if they were compromised.

The working process of an HSM are:

- Generates safe cryptography keys on-board.
- Master keys, or highest-level, are storedsafely onboard.
- Key personnel.
- For jobs involving digital signatures and decryption, use sensitive data and cryptography.
- Application servers that are offloaded for complete symmetric and asymmetric cryptography.

Transparent data encryption keys for storage devices and databases, such as disks and tapes are also managed by HSMs. Logically and physically, HSMs protect these resources—including cryptographic keys—from prying eyes, unsanctioned access, and potential enemies. Asymmetric key pairs and certificates The cryptographic materials utilized in some applications, such digital signatures and certificate authorities, come from public-key cryptography. For other uses, including financial payment systems or data encryption, symmetric keys make up the majority of cryptographic material.

Figure 4.2

4.1.2 FPGA:

Field Programmable Gate Arrays (FPGAs) are semiconductor devices consisting of a matrix of programmable logic blocks (CLBs) coupled via programmable interconnects. FPGAs can be reconfigured once they are constructed to meet specific feature or application needs. FPGAs differ from Application Specific Integrated Circuits (ASICs), which are custom-built to fulfil specific design specifications, in that they have this feature. Although certain FPGAs are only needed to be programmed once, the most widely available models are SRAM-based and may be programmed again when the design changes.

Application:

- Defense & Aerospace: Intellectual property and radiation-tolerant FPGAs for waveform generation, image processing, and SDR reconfiguration.
- Embedded software verification and SoC system modeling may be completed more quickly and accurately using FPGA-based ASIC prototyping.

- Broadcast & AV - High-end professional broadcast systems' design platforms and solutions enable quicker response to changing requirements and longer product lifetimes.

- Full-featured consumer applications including digital good panel displays convergence phones, and residential set-top boxes are made possible by our affordable technology. The use of a data centre is to enhance cloud deployments by providing good bandwidth .

- For NAS, SAN, and storage systems, we provide good performance computing and data storage ideas. Industrial: Higher degrees of flexibility, quicker time-to-market, and lower total non-securing engineering costs (NRE) are made possible by Xilinx FPGAs and targeted design platforms for Industrial, Scientific, and Medical (ISM) applications. The processing, display, and I&O interface needs for medical applications, such as diagnosis, monitoring, and therapy, may be met by the Vivado FPGA and Spartan FPGA families.

• Xilinx offers solutions for safety systems, surveillance, and access control, among other security-related applications. Video & Image Processing: A variety of video and imaging applications can benefit from more flexibility, a quicker time to markets and cheapest total NRE thanks to Xilinx FPGA and tailored designplatforms.

- Wireless Communications: This category includes base band, RF, networking, connectivity, and transport solutions for wireless devices that comply with WiMAX, HSDPA, and other standards.

Figure 4.3

4.2 Software Requirements:

4.2.1 VIVADO:

Vivado is a comprehensive design suite created by AMD for developing and implementing designs on Adaptive SoCs and FPGA. It gives a variety of tools and features that are streamline the entire design flow, from design entry to implementation and verification.

Capabilities:

- Design Input: Vivado supports various design entry formats, including Verilog, VHDL, System Verilog, and IP Integrator.
- Synthesis: Converts HDL code into a netlist that represents the logic gates and interconnections of your design.
- Place and Route: Maps the synthesized netlist onto the FPGA fabric, optimizing placement and routing for performance and timing closure.
- Verification/Simulation: Provides tools for simulating and verifying your design at various levels of abstraction, ensuring its functionality before implementation.
- System-on-Chip (SoC) Design: Offers advanced features for designing and implementing complex SoC systems, including IP integration, power analysis, and floor planning.
- High-Level Synthesis (HLS): Enables C/C++ code to be converted into hardware for faster prototyping and design exploration.
- Timing Closure: Provides a comprehensive set of tools for analyzing and optimizing timing performance, ensuring your design meets timing constraints.
- Methodology Support: Supports various design methodologies, including Agile, Waterfall, and IP-centric design.

Benefits:

- Improved Productivity: Streamlines the design flow with a unified interface and advanced automation features, leading to faster design cycles.
- Enhanced Performance: Optimizes designs for performance and timing closure, enabling efficient implementation on FPGAs.
- Reduced Design Errors: Comprehensive verification tools help identify and eliminate errors early in the design process.
- Increased Flexibility: Supports various design entry formats and methodologies, offering flexibility for different design styles.
- IP-Centric Design: Enables efficient integration and reuse of intellectual property (IP) cores, accelerating design creation.

Figure 4.4

CHAPTER-5

ALGORITHM

5.1 Advanced Encryption Standard:

. Secret data is protected by the US military using a identical block cipher called the AES.

Encrypting important data is a global usage of AES in hardware and software. Cybersecurity, electronic data protection, and military computer security all depend on it.

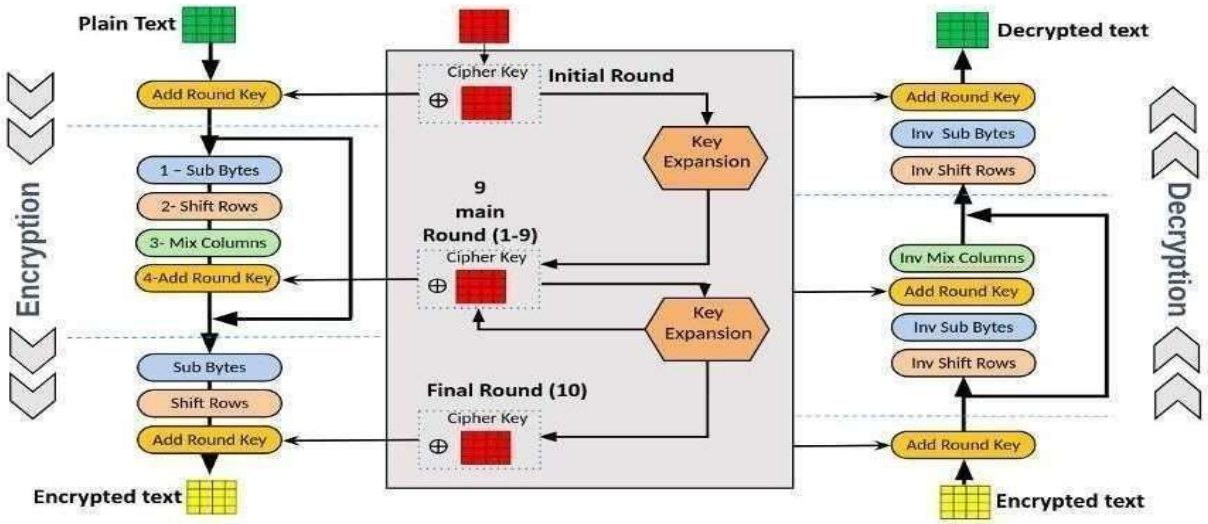


Figure 5.1

Working of AES:

AES includes three block ciphers:

- Using a 128 bit key, AES 128 bit encrypts and decrypts a block of messages. AES 192 is used to encrypt and decrypt a block of communications with a 192 bit key length.
- AES 256 encrypts and decrypts message blocks with a 256 bit key length. Cryptographic keys with lengths of 128 bits, 192 bits, and 256 bits, respectively, are used by each cypher to encrypt and decrypt data in blocks.

Symmetric cyphers, often known as secret key cyphers, employ a single key for both encryption and decoding. Both the sender and the recipient must be aware of and able to utilise the secret key.

Multiple encryption rounds are used to the AES encryption method. It may even go in this manner through 9, 11, or 13 rounds.

The instructions following are the same for every round.

- Split the information into chunks.
- Expansion of the key.

- Include the spherical key.
- The bytes are changed or substituted.
- Rearrange the rows.
- Stir the columns together.
- Reapply a circular key.
- Re-do everything.

The algorithm will go through one more round after the last one. With the exception of step 6, the algorithm will complete steps 1 through 7.

Since the sixth step would not work at this time, it is modified. Recall that it has already gone through this process several times.

Consequently, there's no need to repeat step 6. The data won't be much changed; therefore, it just isn't worth the processing effort needed to mix the columns once again.

Data stored in an array may be modified in a number of ways using the AES algorithm. The data is first placed in array as part of that cipher's first stage. Afterward, many encryption cycles are completed by repeating the cipher's alterations.

Using this substitution table, data substitution is the initial alteration in the AES algorithm. all data rows are rearranged in the second transformation. Columns are combined in the third phase. Using a separate part of the encryption key for each column, the last transformation is applied. Longer keys take more rounds to complete.

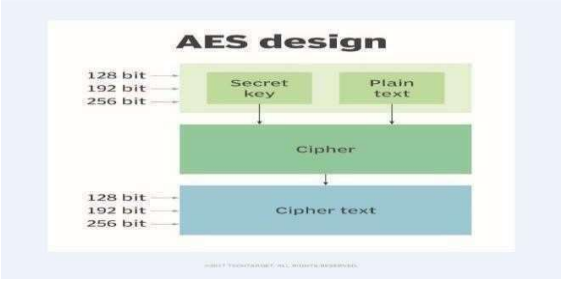


Figure 5.2

Features of AES:

NIST required that the recently created AES algorithm be a block cypher capable of handling 128 bit blocks with keys with 128, 192, and 256 bit widths.

When selecting the next AES algorithm, the following factors were also taken into account.

Security: Competing algorithms were to be assessed based on how well they could withstand attacks, relative to other cyphers. Security was intended to be the main focus of the tournament.

Cost: Following an evaluation of their memory and computational efficiency, the potential algorithms were to be made publicly available, nonexclusively, and without restriction on use.

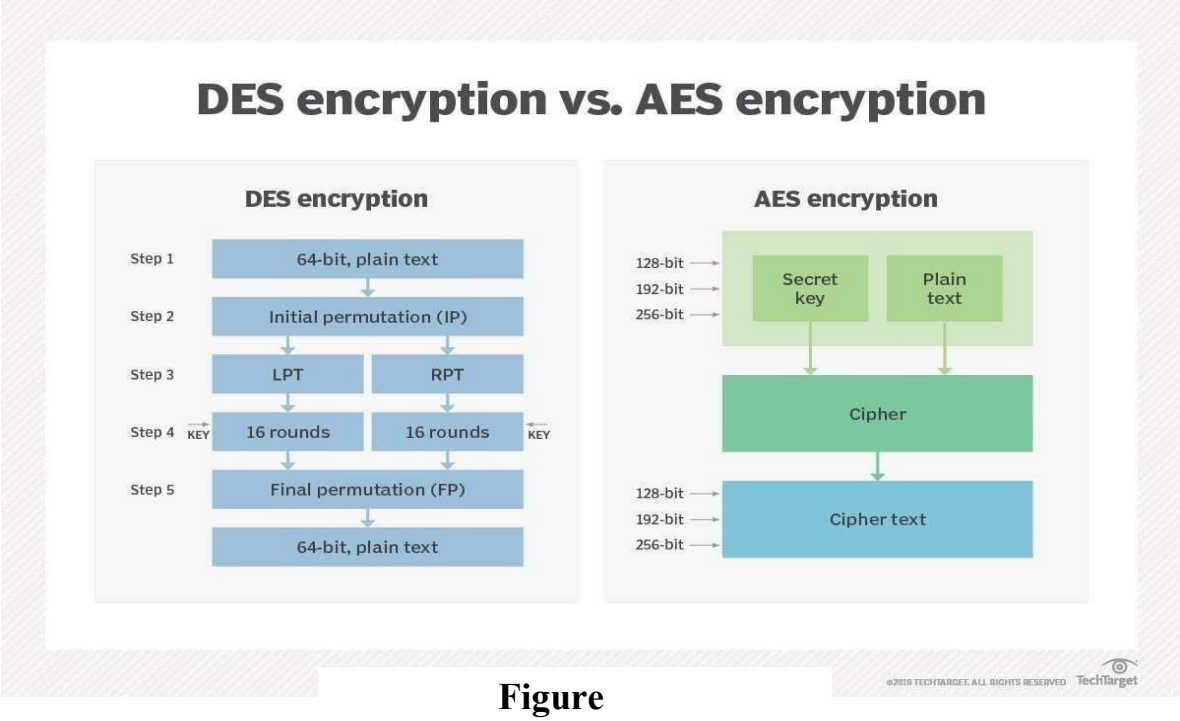
Consider the method's general simplicity, versatility, and adaptability to various hardware and software when putting it into practice.

Difference between AES encryption and DES encryption:

Up until 1999, when researchers employed a distributed computer system to crack the algorithm's 56-bit key, DES served as the heart of govt encryption. The US government made the decision to use AES in 2000 to protect sensitive data. In certain situations, DES is still used for backward compatibility.

Symmetric block ciphers are used by both protocols, while AES is theoretically more effective. AES's primary advantage is its key length.

An encryption technique's breaking time is closely correlated with the size of the key used to secure the message. Consequently, AES's 56-bit keys are orders of magnitude weaker than DES's. Because AES encryption is faster, it is a great choice for firmware, programs, and devices that require high throughput or low latency.



Figure

AES 256 Encryption:

We are aware that encryption techniques mix up the data they are meant to secure, creating an unpredictable confusion.

All encryption is based on the basic idea that every data unit is replaced with a new one according to the security key.

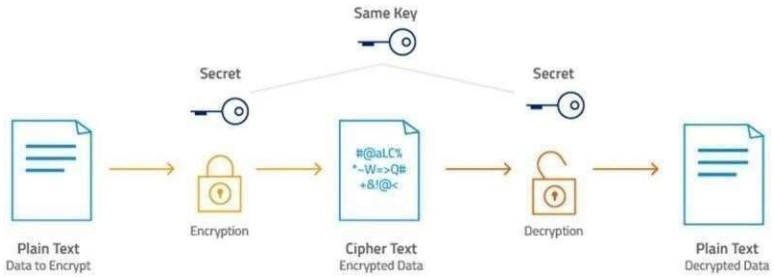


Figure 5.4

This encryption procedure consists of many rounds.

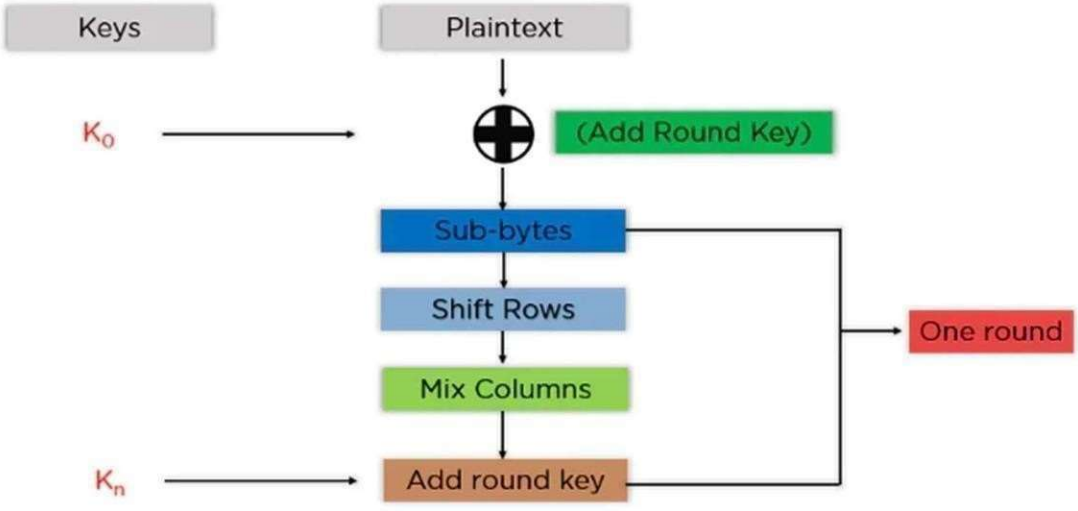


Figure 5.5

Add Round Key: An XOR method is used to combine the initial key created with the block data in the state array. The subsequent step receives the resultant state array as an input.

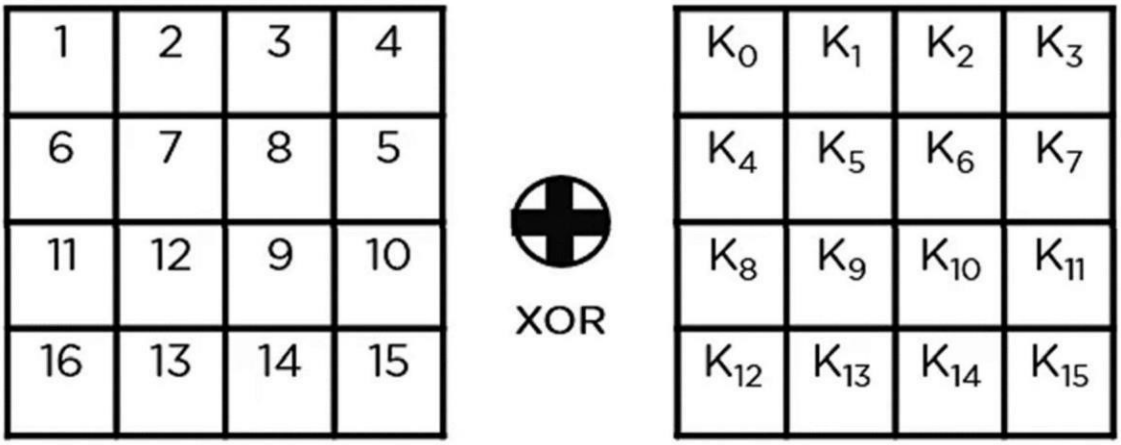


Figure 5.6

Sub-Bytes: The state array's bytes are now divided into two equal parts and transformed to hexadecimal. These rows and columns are mapped to the final state array so that a substitution box (S-Box) can be used to create new values.

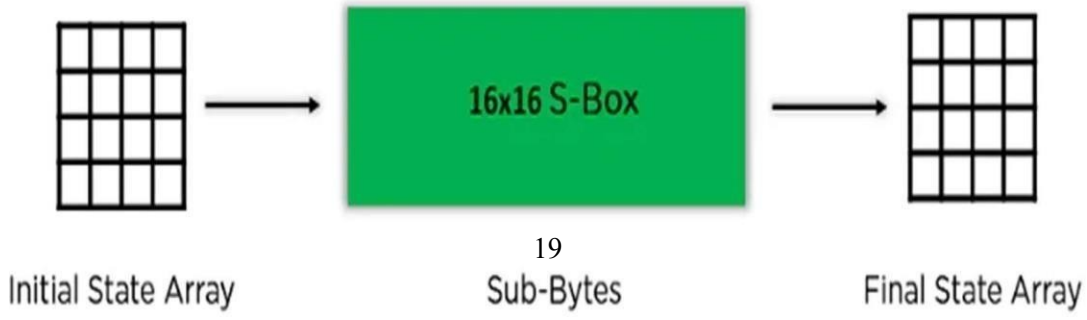


Figure 5.7

Shift Rows: Row items are switched. It avoids the initial row. It shifts the parts in the second row to the left by one position. it shifts the items in the last row three positions to the left and the third row's contentstwo places to the left.

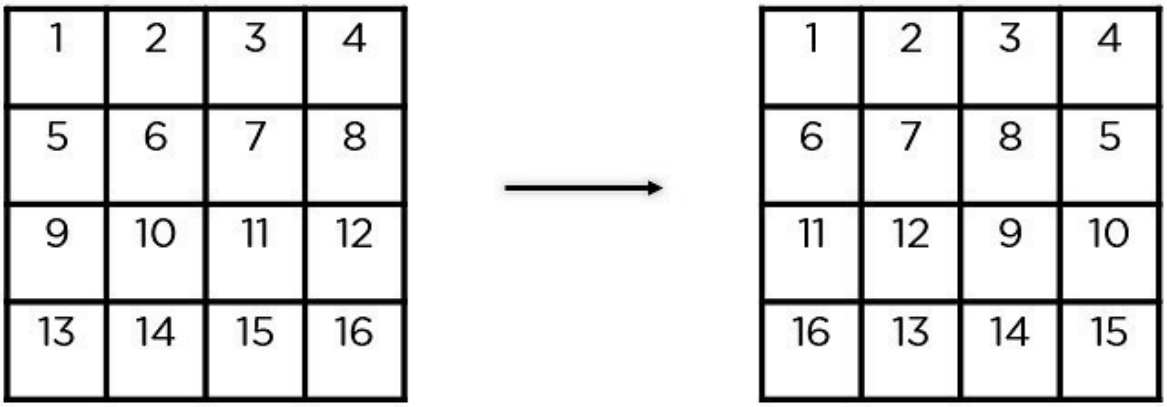


Figure 5.8

Mix Columns: To create a new column for the following state array, it multiplies a constant matrix by each column in the state array. You'll have your state array for the next step after multiplying each column by the same constant matrix. In the last round, this specific step is not to be finished.

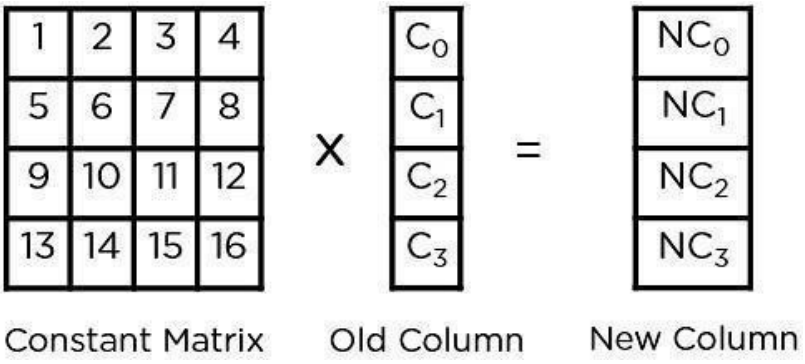


Figure 5.9

Add Round Key: The round's key is combined with the state array that was formed in the previous phase. If this is the last round, the resultant state array becomes the ciphertext for the designated block; if not, it becomes the new state array input for the following round.

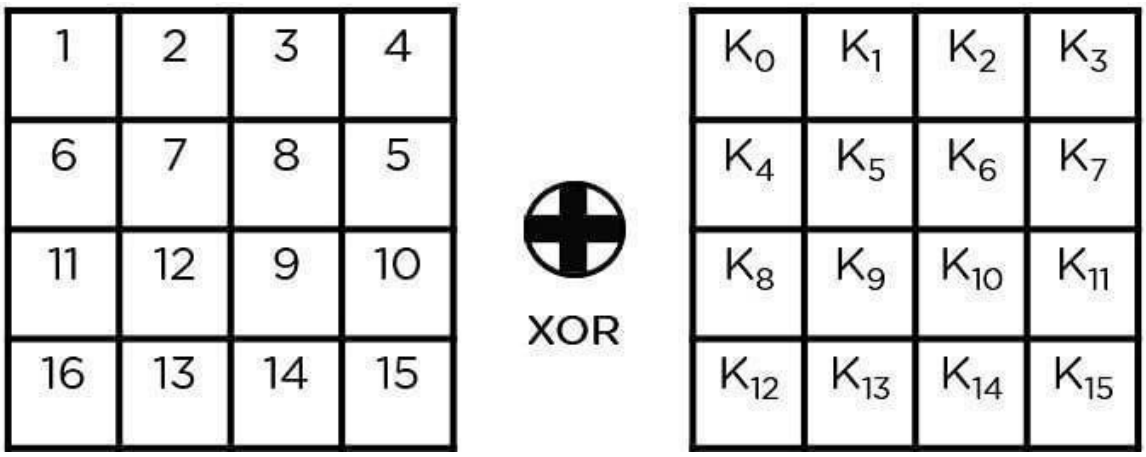


Figure 5.10

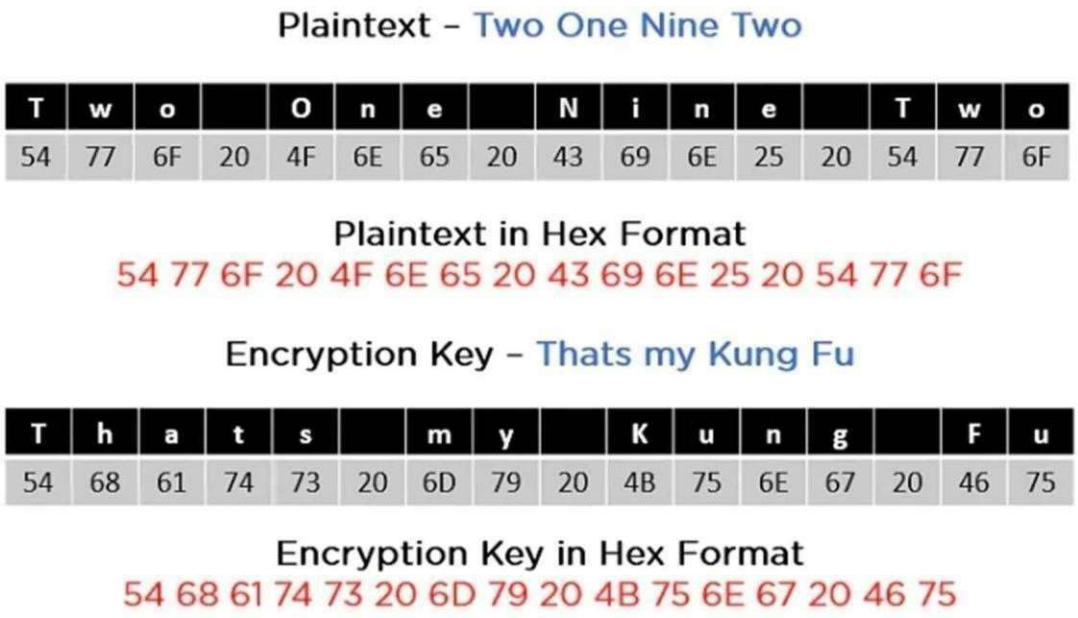


Figure 5.11

Before beginning any actions, plaintext and encryption first convert keys to hexadecimal format, as shown in the image above. As a result, you can create keys for the next ten rounds, as seen below.

The above-described procedures must be repeated in the order to extract the state array and send it through input to the round. Here are the steps to follow.

Put a circular key there.

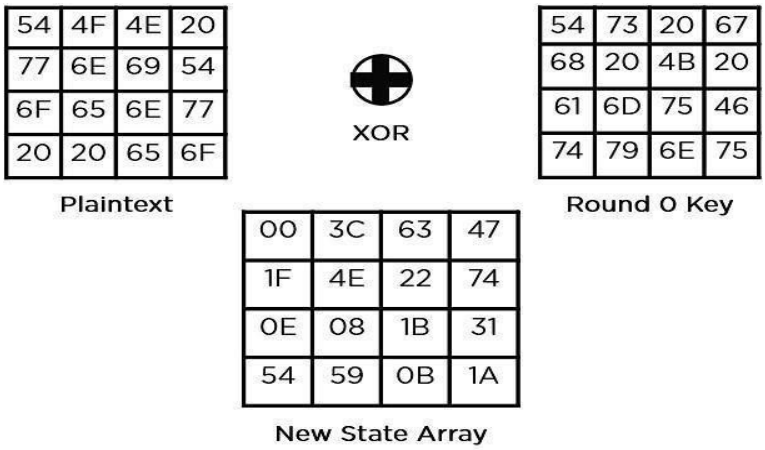


Figure 5.12

Sub-Bytes: To get a whole new state arrays, the elements are passedthrough a 16x16 S-Box.

63	EB	9F	A0
C0	2F	93	92
AB	30	AF	C7
20	CB	2B	A2

New State Array

Figure 5.13

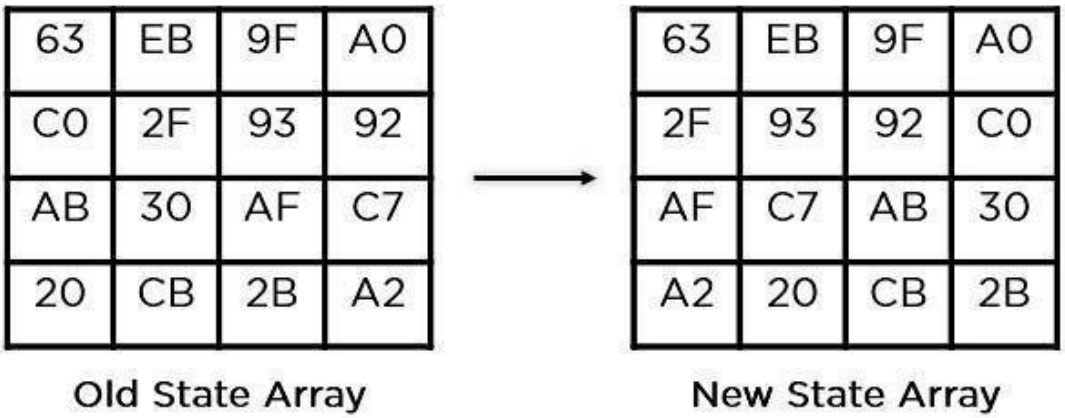


Figure 5.14

Mix columns,

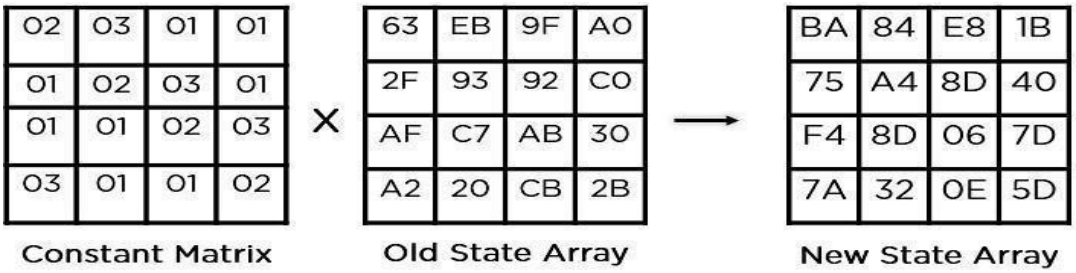


Figure 5.15

Add round key,

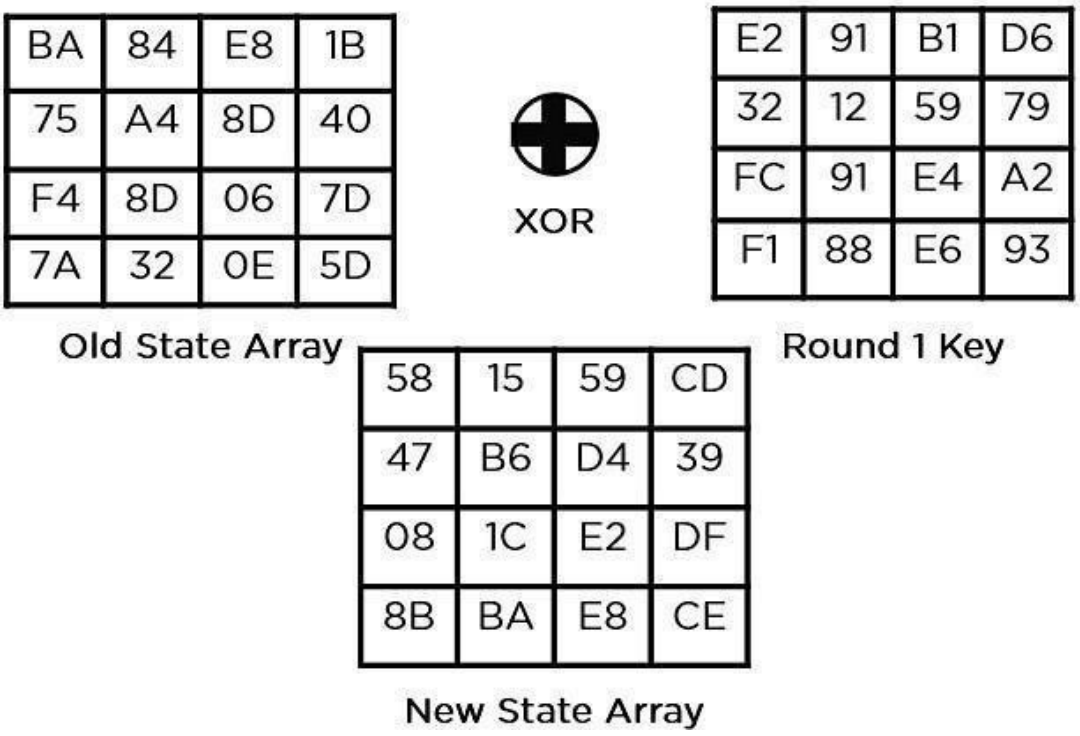


Figure 5.16

This state array now holds the final ciphertext for this round. This is the cycle's input the next time around. Depending on the length of the key, you will receive the final ciphertext after round 10 if you follow the previously specified stages.

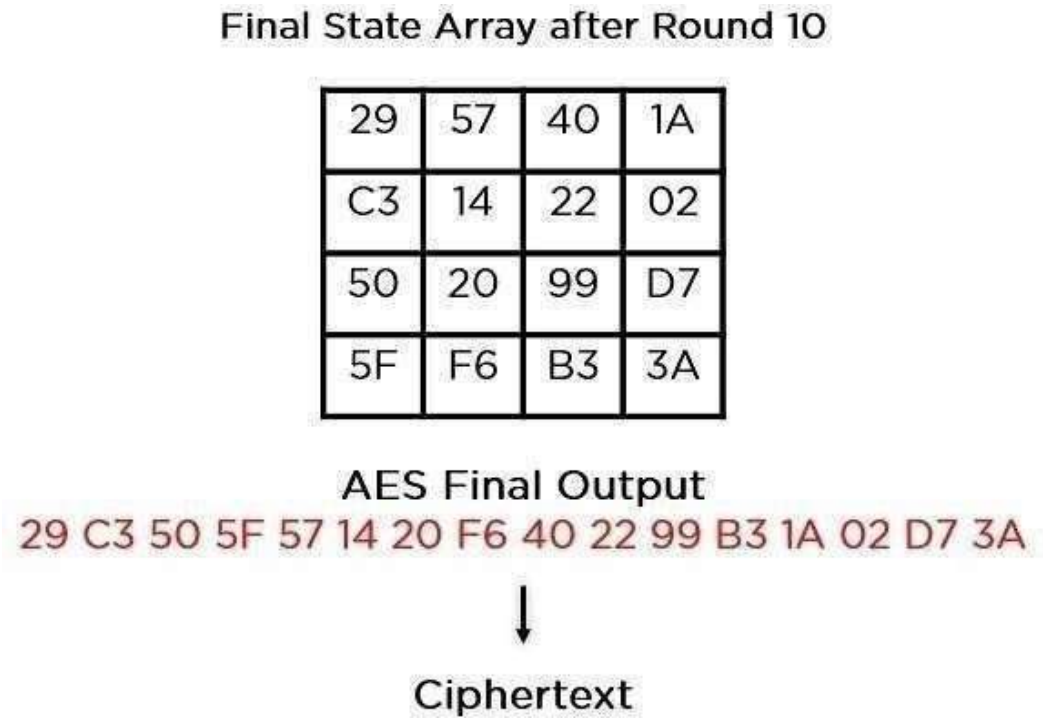


Figure 5.17

Application of AES:

- AES is used in wireless networks for client and router authentication. Firmware software and all- encompassing security measures built on this technique are utilized by Wi-Fi networks and are currently in widespread usage.
- Secure website server authentication is guaranteed by AES encryption on both the client and server ends. This method helps SSL/TLS encryption methods to guarantee the best level of security and privacy when surfing by utilizing both symmetrical and asymmetrical mode.
- AES is frequently used for business as well as encrypted data transfers between partners. Legal documents, family photographs, and chat discussions can all contain encrypted content.
- Processor Security Many processor manufacturers use hardware-level encryption, like as AES encryption, to improve security and lower the likelihood of meltdown failures, among other low-profile issues. chip Security: To improve security and lower the likelihood of meltdowns, several chip manufacturers include hardware-level encryption, such as AES encryption, in addition to other low-profile problems.

CHAPTER – 6

EXPERIMENTS OUTPUTS

Encryption:

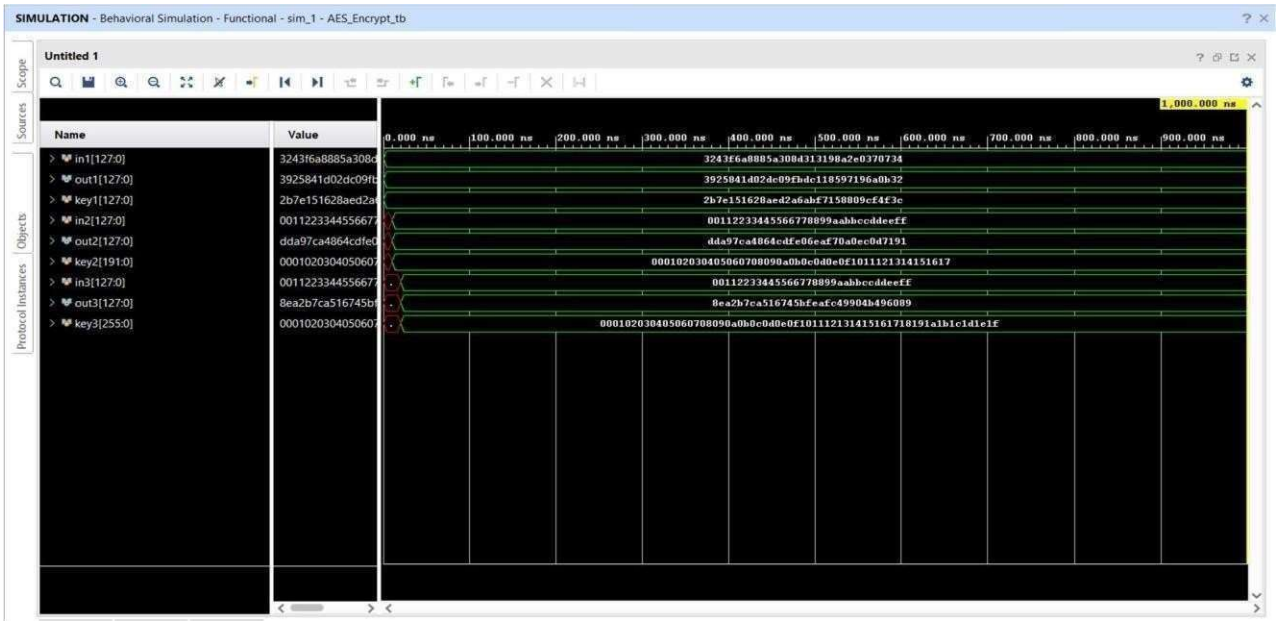


Figure 6.1

Decryption:

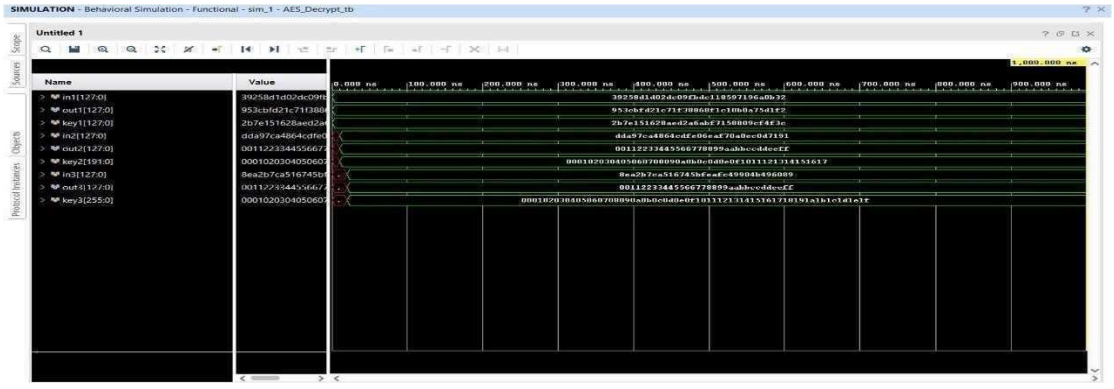


Figure 6.2

CHAPTER – 7

CONCLUSION AND FUTURE SCOPE

Conclusion:

This leads us to the conclusion that we have successfully run the Verilog code of aes 128, 192 and 256 bits in vivado software

Future scope:

We intended to enhance this project through IOT applications which can embedded with the PLCs for the industrial applications. And also, we can use this in military applications.

REFERENCES

1. Kumar, Pramod, T. V. Narendra, and N. A. Vinay. "Short Hand Recognition using Canny Edge Detector." International Journal 7, no. 5 (2017).
2. Kumar, Mamatha MS Pramod, and M.Mamatha. "FPGAImplementation Of Low Area Single Precision Floating Point Multiplier."International Journal of Science Technology and Engineering, Vol.2, no. 2 (2016): 560-566.
3. M.Natheera Banu, FPGA Based Hardware Implementation of Encryption Algorithm, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 8958, Volume-3, Issue-4, April 2014.
4. Deguang Le, Jinyi Chang, Xingdou Gou, Ankang Zhang ,Conglan Lu , Parallel AES Algorithm for Fast Data Encryption on GPU, IEEE journal on AES 2010.
5. K. Xinmiao Zhang, High speed VLSI architectures for the AES algorithm, IEEE transactions on VLSI systems, Tech. Rep., sep2004.
6. National Institute of Standards and Technology, Advanced Encryption Standard, Federal Information Processing Standards 197, November 2001.
7. M.Pitchaiah, Philemon Daniel, Praveen, Implementation of Advanced Encryption Standard Algorithm, International Journal of Scientific Engineering Research.

20% Overall Similarity

Top sources found in the following databases:

- 7% Internet database
- Crossref database
- 19% Submitted Works database
- 3% Publications database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Submitted on 1690504213462 Submitted works	3%
2	Sri Sairam Engineering College on 2024-04-13 Submitted works	2%
3	Sri Sairam Engineering College on 2024-04-01 Submitted works	2%
4	Middlesex University on 2023-04-30 Submitted works	1%
5	IPEKA International Christian School on 2023-01-16 Submitted works	1%
6	Sabanci Universitesi on 2021-10-07 Submitted works	1%
7	Webster University on 2022-10-14 Submitted works	<1%
8	Kingston University on 2023-12-18 Submitted works	<1%

9	ijettjournal.org Internet	<1%
10	Kingston University on 2022-01-09 Submitted works	<1%
11	software.fujitsu.com Internet	<1%
12	University of Maryland, University College on 2021-09-06 Submitted works	<1%
13	Queen's University of Belfast on 2022-03-01 Submitted works	<1%
14	Webster University on 2022-10-11 Submitted works	<1%
15	Kingston University on 2023-12-18 Submitted works	<1%
16	Toronto Business College on 2023-09-02 Submitted works	<1%
17	Rivier University on 2015-08-03 Submitted works	<1%
18	The Manchester College on 2022-03-23 Submitted works	<1%
19	Indian Institute of Technology on 2024-02-16 Submitted works	<1%
20	dspace.cus.ac.in Internet	<1%

21	Sri Sairam Engineering College on 2024-04-15 Submitted works	<1%
22	kipdf.com Internet	<1%
23	pdfs.semanticscholar.org Internet	<1%
24	ijert.org Internet	<1%
25	Loughborough University on 2022-04-01 Submitted works	<1%
26	PSB Academy (ACP eSolutions) on 2022-08-27 Submitted works	<1%
27	Sri Sairam Engineering College on 2024-04-15 Submitted works	<1%
28	Sri Sairam Engineering College on 2024-04-15 Submitted works	<1%
29	Swarnima Rai, Vaaruni Choubey, Suryansh, Puneet Garg. "A Systemati... Crossref	<1%
30	University of Wales Swansea on 2023-05-04 Submitted works	<1%
31	ir.juit.ac.in:8080 Internet	<1%
32	DeVry University Onlline on 2015-09-21 Submitted works	<1%

33 DeVry, Inc. on 2021-09-27
Submitted works <1%

34 National University on 2015-02-16
Submitted works <1%

35 School of Business & Computer Science Limited on 2023-05-26
Submitted works <1%