AES 128 ENCRYPTION

REPOPRT

PROFESSOR: Mr. BISHNU DAS

TEACHING ASSISTANT: ANU VERMA

TEACHING ASSISTANT: ADABALA VENUGOPAL

PROJECT-TEAM

SHASHANK SONI (20116090) - O8

DISHANT MEENA (20116030) - 06

SAINATHA RAO P (20116094) - O8

ADITYA TADIPARTHI (20116098) - O8

YASHRAJ (20116110) - O8

ABSTRACT

In this report, we Implement the efficient **Advanced Encryption Standard (AES)** Algorithm. This is world's most secure encryption algorithm. In this implementation, Substitute byte, mix column, shift row and several other operations are implemented using composite field arithmetic. In the algorithm of this AES-128 has a plaintext of 128 bits and random cipher-key of 128 bits size. In this algorithm for 128-bit size cipher key there are 10 rounds excluding initial round. For each round we need input keys so we expand our cipher input key. **Modelsim starter edition 20.2** software is used for simulation and optimization of the synthesizable VERILOG code.

INTRODUCTION

The Advanced Encryption Standard is a symmetric block cipher. It is chosen by the U.S. government to protect the classified and sensitive information. AES is implemented in software and hardware both. It is crucial for government cybersecurity, computer security, and electronic data protection. In this algorithm, we can have a plain text usually 128-bit size, and produces a corresponding output block of the same size. We also need a second input key known as secret key or cipher-key. It is important to know that the cipher-key can be of any size (in our case, we have used 128 bit) and that AES can use three different key sizes: 128, 192 and 256 bits.

DESIGN PROCEDURE

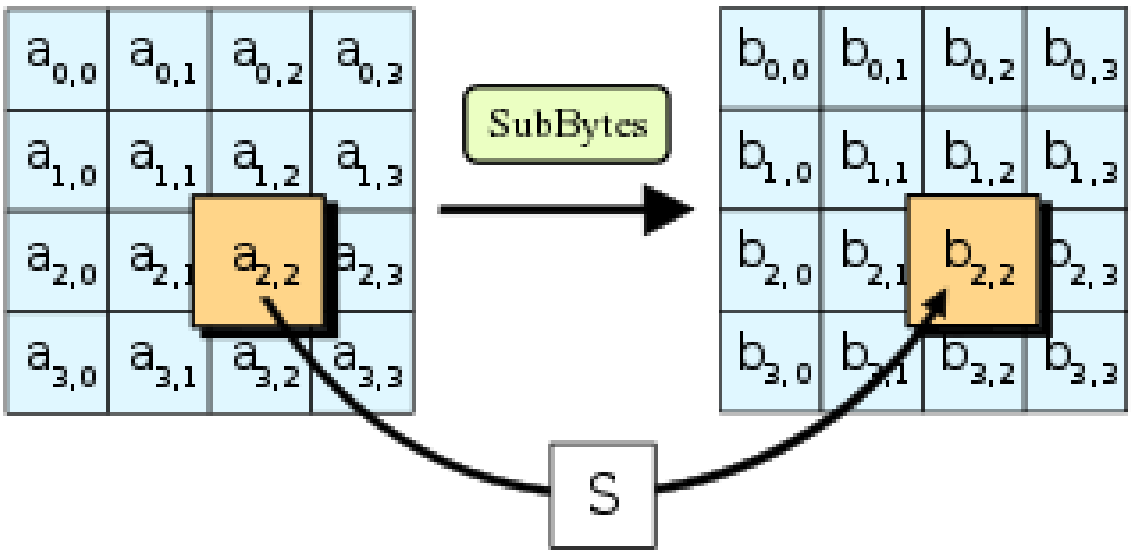
In this AES algorithm, we follow three parts**: INITIAL ROUND, INTERMIDIATE ROUNDS** and **FINAL ROUND**. The AES algorithm works on 128-bit data blocks using a cipher key of possible lengths of 128bit size (key length can be based on our choice). Each INTERMIDIATE ROUND consists of a set of transformations as follows: **Substitute Bytes, Shift Rows, Mix Columns, AddRoundKey.**

I. Substitute bytes

In the state matrix each cell has 8 bits. (a_7 a_6 a_5 a_4….a_0).

Divide it into two parts i.e.., a_7 to a_4(let say x) and a_3 to a_0(let say y). The first four bits (a_7-a_4) represents the row number whereas the other four bits(a_3-a_0) represent the column number in S-box

Replace each cell of state matrix with the corresponding S-box cell.

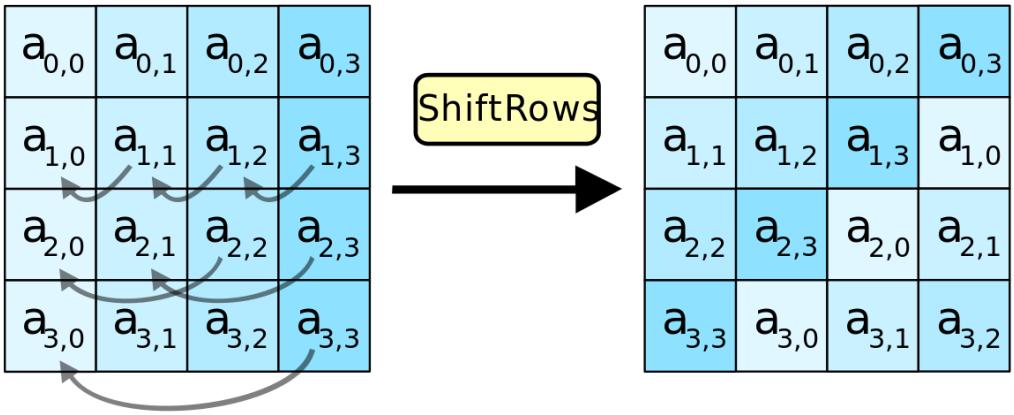


II. Shifting Rows

The shifting concept is as following-

- First row is not shifted.
- Second row is shifted one unit.
- Third row is shifted two units.
- Fourth row is shifted three units.

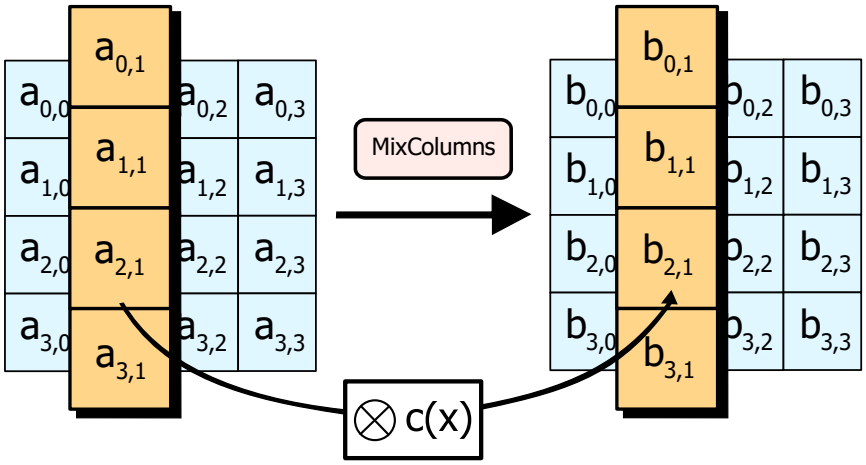
Traverse through the four rows and take the j value for each row in $S_{i,j}$ notation of state matrix. Now take the first j units from the left end and insert at the right after the row ends and the final array results in the following way



III. Mix-column operation:

Perform matrix multiplication between state matrix and a constant matrix

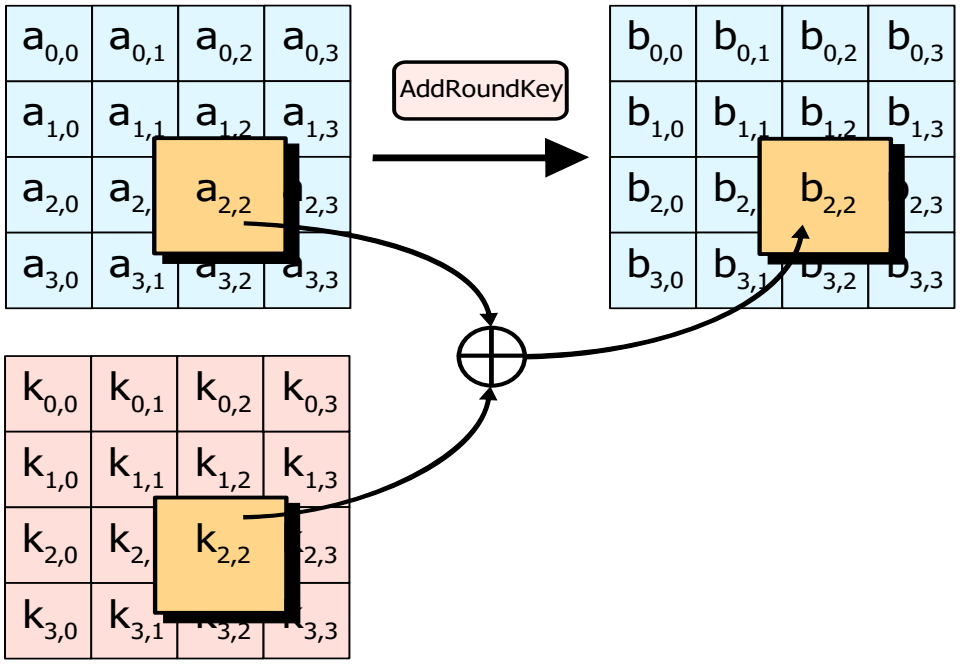
Example: $b_{0,1} = 02 \times a_{0,1} + 03 \times a_{1,1} + 01 \times a_{2,1} + 01 \times a_{3,1}$



02	03	01	01
01	02	03	01
01	01	02	03
03	01	01	02

IV. Add Round key

Perform XOR operation of each byte with corresponding bytes in round key. This modified state matrix is then further encrypted in next rounds.



SIMULATION AND MEASURED RESULTS

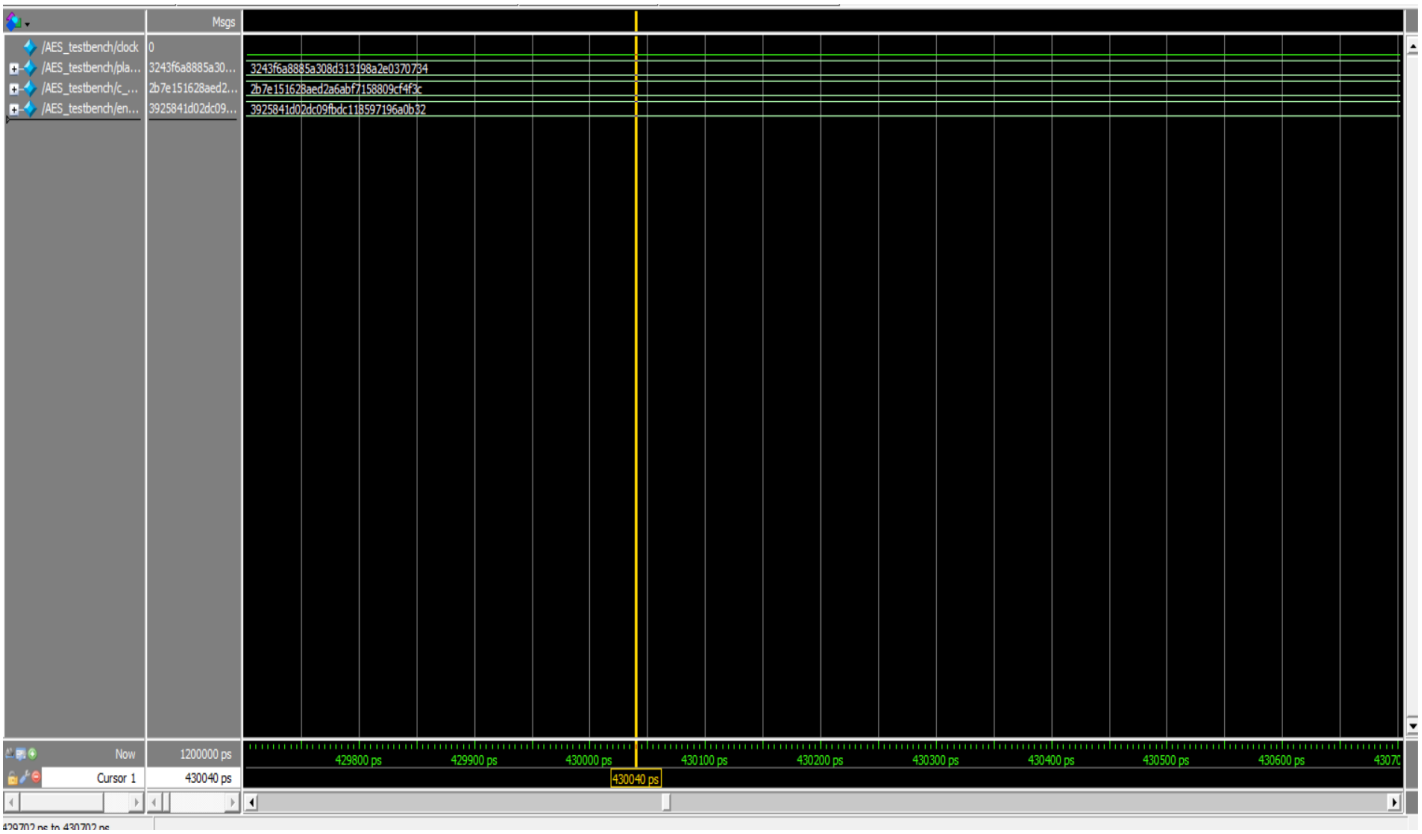
The simulation of AES encryption is done using Modelsim starter edition 20.2.1

First simulation result:

Plain-text: 3243f6a8885a308d313198a2e0370734

Cipher-key: 2b7e151628aed2a6abf7158809cf4f3c

Ciphertext: 3925841d02dc09fbdc118597196a0b32

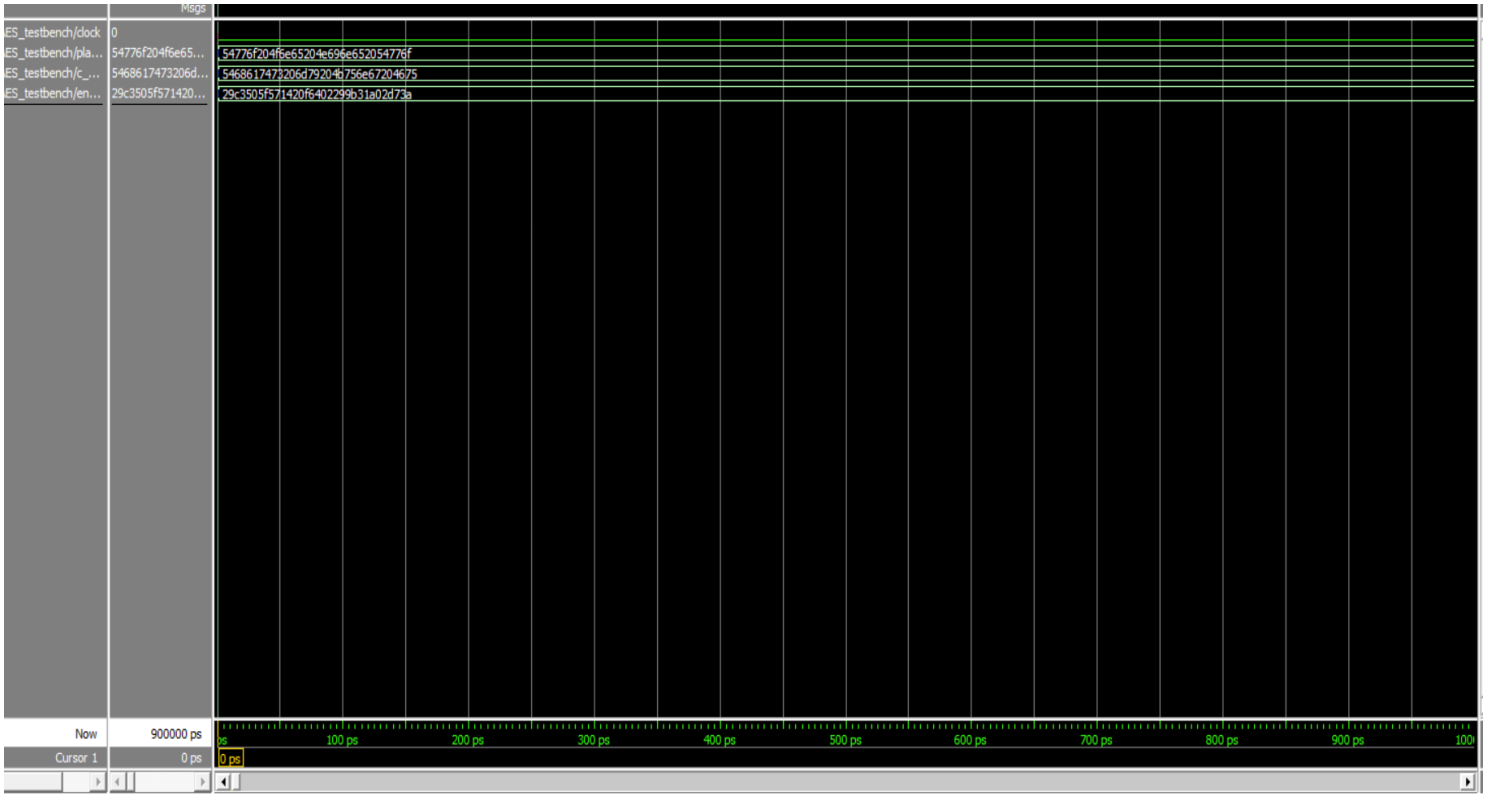


Second simulation result:

Plain-text: 54776F204F6E65204E696E652054776F

Cipher-key: 5468617473206D79204B756E67204675

Ciphertext: 29c3505f571420f6402299b31a02d73a

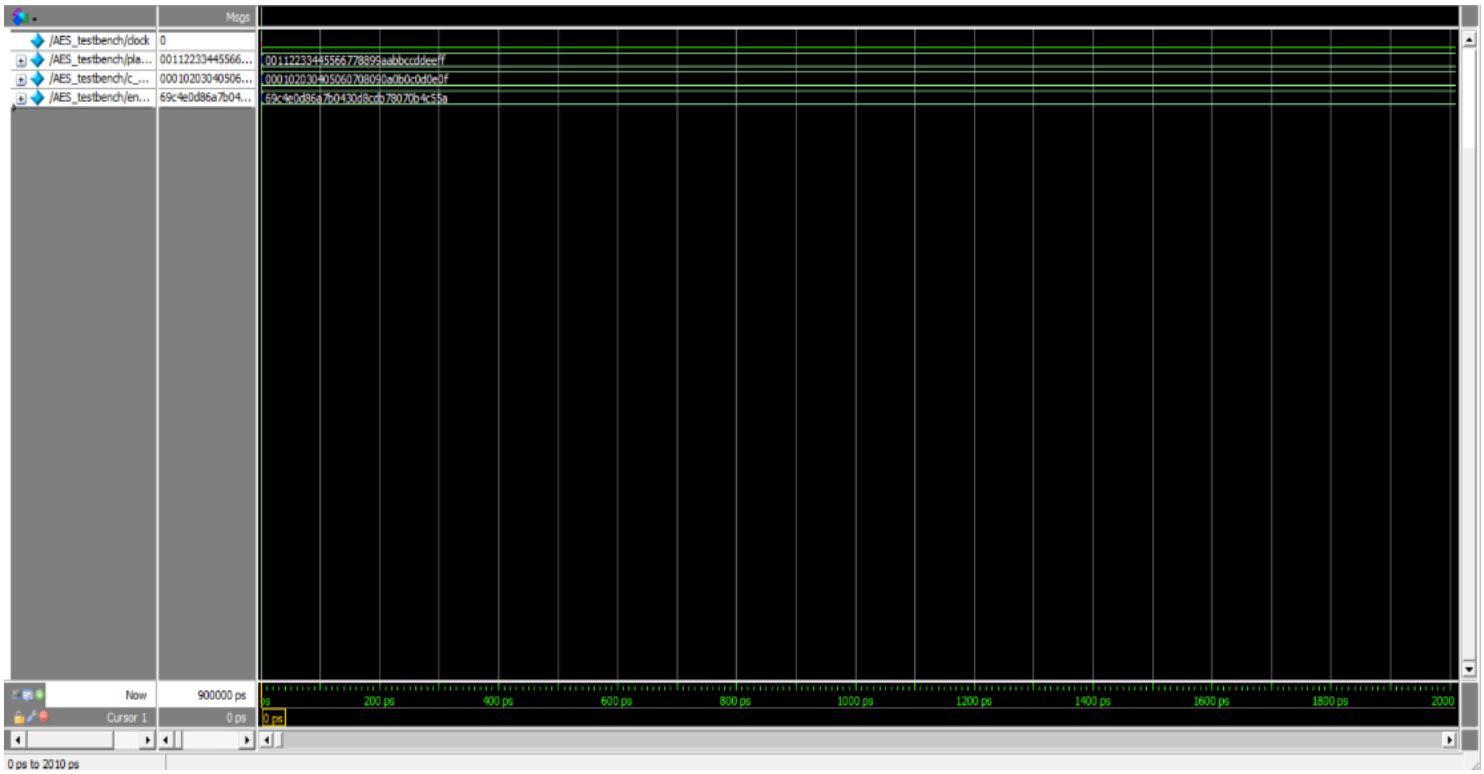


Third simulation result:

Plain-text: 00112233445566778899aabbccddeeff

Cipher-key: 000102030405060708090a0b0c0d0e0f

Ciphertext: 69c4e0d86a7b0430d8cdb78070b4c55a



CONCLUSION:

The Advanced Encryption Standard algorithm is a symmetric block cipher that can process data blocks of 128 bits through the use of cipher keys. These cipher keys can be of 128, 192, and 256-bits size. In this paper we have presented the efficient encryption AES technique. We have developed Improved and optimized VERILOG code for the different rounds of this AES algorithm. Using composite field arithmetic operations, we have implemented sub round operations like Sub byte, shift round etc. For this implementation we have used Modelsim starter edition 20.2.1.

REFRENCES

https://www.kavaliro.com/wp-content/uploads/2014/03/AES.pdf

AESAlgorithmpaper2017AKOMAbdullah.docx.pdf

https://www.researchgate.net/post/How-to-do-RTL-design-of-AES-128-bit-implementation-for-securing-voice-communication-using-fpga

https://www.educative.io/edpresso/what-is-the-aes-algorithm

https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf