iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 Issue: XI Month of publication: November 2019

DOI: http://doi.org/10.22214/ijraset.2019.11096

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue XI, Nov 2019- Available at www.ijraset.com

Proposed Work for Modification of AES Mix Column

Nihar Sahu[1], Vidyadhari R. Singh[2]
[1]*BE, Computer Engineering, Thakur College of Engineering and Technology, Mumbai, India*
[2]*Assistant Professor, Computer Engineering, Thakur College of Engineering and Technology, Mumbai, India*

Abstract: AES is most popular and widely used symmetric encryption algorithm. It follows the widely deployed block cipher approach by iterating a single round function multiple times. This paper proposes implementation of AES mix column operation. This paper tries to provide a compact architecture of mix column operation without following the conventional method.

Keywords: AES, Mix column

I. INTRODUCTION

AES (Advance Encryption standard) is an adaptation of Rijndael algorithm by National Institute of Standards and Technology in the year 2001. Due to failure of DES (Data Encryption Standard) because of small key size it was considered vulnerable against exhaustive key search attack. AES began to replace DES immediately after its selection and implementation. AES is better than DES due to its improved long term security because of its larger key sizes (128,192 and 256 bits).For each key length the round function is iterated different number of times (10,12 and 14 respectively). A 128 bit plaintext is treated as 4x4 byte matrix, where each byte represents a value in GF (2^8).

AES is a cryptographically secure encryption algorithm. A brute force attack requires 2^{128} trials for the 128-bit key size. In addition, the structure of the algorithm and the round functions used in it ensure high immunity to linear and differential cryptanalysis.

Attacks against AES haven't been successful till now and it is the current encryption standard. The AES design can be used in any application that requires protection of data during transmission through the communication network, including applications such as electronic commerce transactions, ATM machines, and wireless communication.

The AES is a universal encryption standard. It can be used for encryption of any type of data, text and other media alike. The AES finds special applications in encryption of images and other media like audio and video. Image encryption using the AES is done generally in Cipher Block Chaining mode to prevent clusters appearing in the image due to similar cipher text. Media file encryption also generally follows on similar lines.

II. AES ALGORITHM

There are 4 different stages, one is permutation and other three are substitution:

Substitute bytes: Uses an S-box to perform a byte-by-byte substitution of the block

Shift Rows: A simple permutation

Mix Columns: A substitution that makes use of arithmetic over GF (2^8)

Add Round Key: A simple bitwise XOR of the current block with a portion of the expanded key

In AES, a state is a 4 * 4 matrix and block is a 16-byte length array. The block to state conversion is done column by column. The state to block conversion is also done column by column.

The key that is provided as input is expanded into an array of forty- four 32-bit words, w[i]. Four distinct words (128 bits) serve as a round key for each round. The decryption algorithm makes use of the expanded key in reverse order. However, the decryption algorithm is not identical to the encryption algorithm.

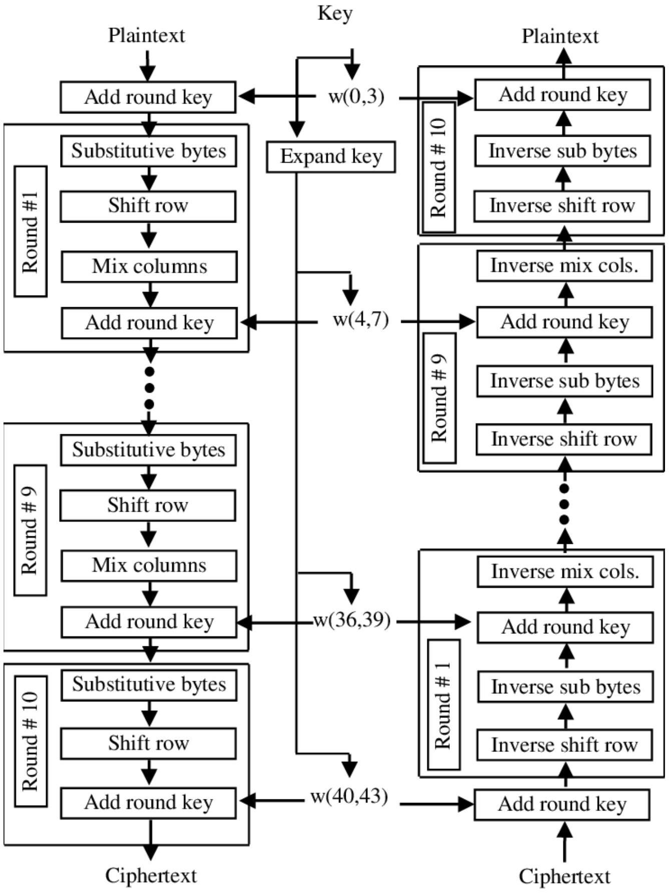
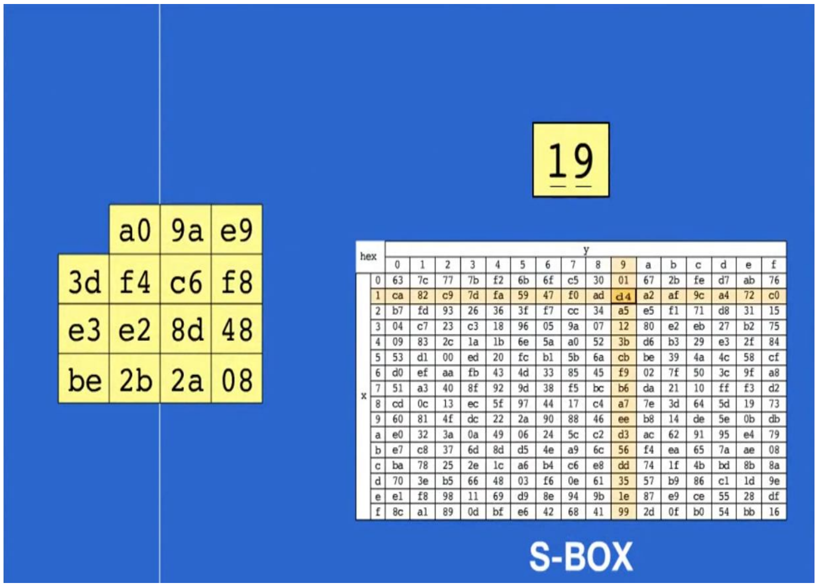


Fig. 1.AES encryption and decryption

A. *Substitute Bytes*

It is a nonlinear byte substitution. It has two sub transformation; multiplicative inverse of Galois field and affine transformation. In many case these two sub transformation is combined into a single lookup table called s-box. Byte Substitution provides confusion in the algorithm.



	a0	9a	e9
3d	f4	c6	f8
e3	e2	8d	48
be	2b	2a	08

hex									y								
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
x	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

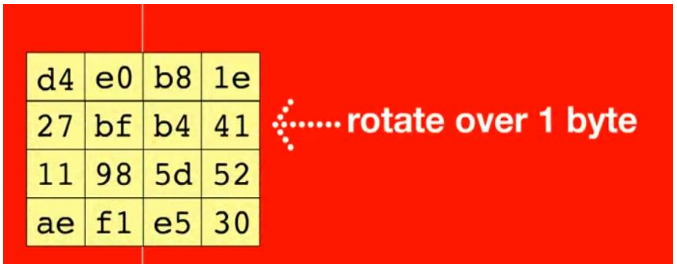
Fig. 2.The sunstitution byte transformation

B. Shift Rows

This is a simple permutation process operating on individual rows. The shift row permutation is as follow:

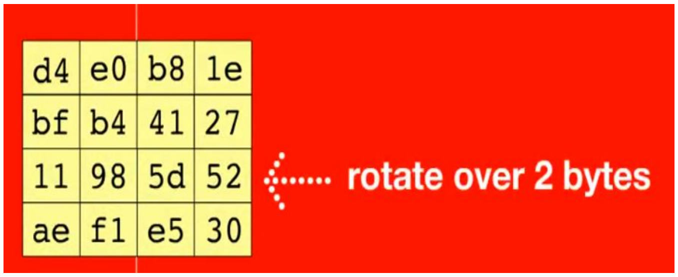
1) 1st row: no shift,
2) 2nd row: 1 byte circular left shift,
3) 3rd row: 2 byte circular left shift,
4) 4th row: 3 byte circular left shift.

Shift row transformation provides diffusion in algorithm.



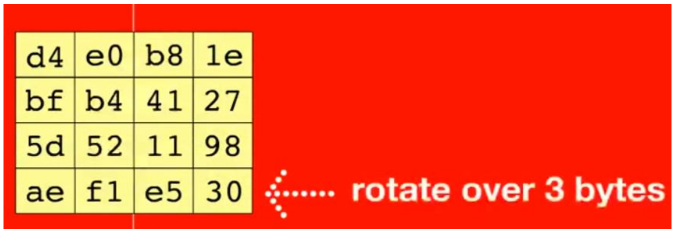
d4	e0	b8	1e
27	bf	b4	41
11	98	5d	52
ae	f1	e5	30

Fig. 3.2nd Round of shift row



d4	e0	b8	1e
bf	b4	41	27
11	98	5d	52
ae	f1	e5	30

Fig. 4.3rd Round of shift row

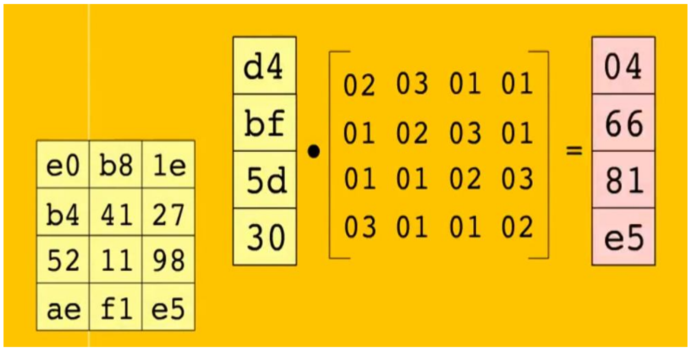


d4	e0	b8	1e
bf	b4	41	27
5d	52	11	98
ae	f1	e5	30

Fig. 5.Last round of shift row

C. Mix Column Transformation

In this transformation column vector (in GF (2^8)) is multiplied by a fixed static matrix where bytes are treated as polynomials of degree less than 4.



e0	b8	1e
b4	41	27
52	11	98
ae	f1	e5

$$\begin{array}{|c|}\hline d4 \\ \hline bf \\ \hline 5d \\ \hline 30 \\ \hline\end{array} \bullet \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} = \begin{array}{|c|}\hline 04 \\ \hline 66 \\ \hline 81 \\ \hline e5 \\ \hline\end{array}$$

Fig. 6.Mix column transformation

D. AddRound Key

In the AddRoundKey step, the subkey is combined with the state. For each round, a subkey is derived from the main key using Rijndael's key schedule; each subkey is the same size as the state. The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.

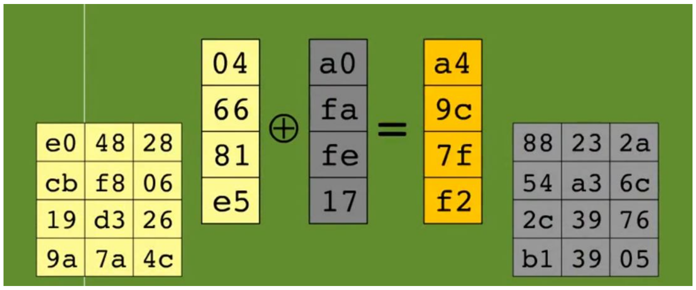


Fig. 7.Add round key trasformation

Features that make AES stand out than other block cipher is that it is a symmetric key symmetric block cipher, stronger and faster than DES and Triple DES, provide full specification and design details, software implacable in C and java.

III. MIX COLUMN IMPLEMENTATION

Mix column operation is done on every single column individually. Each byte of column is multiplied by elements of static multiplicative matrix suggested by AES.

The transformation can be defined by following matrix multiplication.

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} s_{00} & s_{01} & s_{02} & s_{03} \\ s_{10} & s_{11} & s_{12} & s_{13} \\ s_{20} & s_{21} & s_{22} & s_{23} \\ s_{30} & s_{31} & s_{32} & s_{33} \end{pmatrix} = \begin{pmatrix} s'_{00} & s'_{01} & s'_{02} & s'_{03} \\ s'_{10} & s'_{11} & s'_{12} & s'_{13} \\ s'_{20} & s'_{21} & s'_{22} & s'_{23} \\ s'_{30} & s'_{31} & s'_{32} & s'_{33} \end{pmatrix}$$

Every element in multiplicative matrix is sum of products of elements of one row and one column. In conventional method individual addition and subtraction is done in GF (28).

The mix column transformation on single column of

i(0≤ i≤ 3) of state can be expressed as follow:

$$s'0,\ i = (02*s0,\ i) \oplus (03*s1,\ i) \oplus s2,i \oplus s3,i$$

$$s'1,\ i = s1,\ i \oplus (02*s1,\ i) \oplus (03*s2,i) \oplus s3,i$$

$$s'2,\ i = s0,\ i \oplus s\ 1,\ i \oplus (02*s2,i) \oplus (03*s3,i)$$

$$s'0,\ i = (03*s0,\ i) \oplus s1,\ i \oplus s2,i \oplus (02*s3,i)$$

Using identity {03}.x= {02 ⊕ 01}.x can be written as follow {03}.x= {02.x} ⊕ x

We can rewrite the above equation as follow:

$$s'_{0,\ i} = (02*s_{0,\ i}) \oplus (02*s_{1,\ i}) \oplus s_{1,\ i} \oplus s_{2,i} \oplus s_{3,i}$$

$$s'_{1,\ i} = s_{1,\ i} \oplus (02*s_{1,\ i}) \oplus (02*s_{2,i}) \oplus s_{2,i} \oplus s_{3,i}$$

$$s'_{2,\ i} = s_{0,\ i} \oplus s_{\ 1,\ i} \oplus (02*s_{2,i}) \oplus (02*s_{3,i}) \oplus s_{3,i}$$

$$s'_{0,\ i} = (03*s_{0,\ i}) \oplus s_{0,\ i} \oplus s_{1,\ i} \oplus s_{2,i} \oplus (02*s_{3,i})$$

$$x*f(x)=\left\{\begin{array}{ll}(b_6b_5b_4b_3b_2b_1b_00) & \text{if } b_7=0\\ (b_6b_5b_4b_3b_2b_1b_00)\oplus(00011011) & \text{if } b_7=1\end{array}\right\}$$

Multiplication by 02 can be done as the method shown above. For example:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}\begin{bmatrix} d4 \\ bf \\ 5d \\ 30 \end{bmatrix} = \begin{bmatrix} 04 \\ 66 \\ 81 \\ e5 \end{bmatrix}$$

$$r_1 = (02*d4) \oplus (03*bf) \oplus (01*5d) \oplus (01*30) \qquad (1)$$

Using identity we can write

$03*x = (02\oplus01)*x$

$03*x = (02*x) \oplus (01*x)$

$$03*x = (02*x) \oplus x \qquad (2)$$

This can be further simplified as follow:

$r_1 = (02*d4) \oplus (02*bf) \oplus bf \oplus 5d \oplus 30$ From (1)&(2)

to get the product of (02*d4) and (02*bf)

$$x*f(x)=\left\{\begin{array}{ll}(b_6b_5b_4b_3b_2b_1b_00) & \text{if } b_7=0\\ (b_6b_5b_4b_3b_2b_1b_00)\oplus(00011011) & \text{if } b_7=1\end{array}\right\} \qquad (3)$$

Since $(d4)_8 = (11010100)_2$
d4<<1 = 10101000
As the most significant bit is 1 hence we need to add it with $(1b)_8 = (00011011)_2$
Therefore,

$$(02*d4) = 10101000 \oplus 00011011 \qquad (4)$$
=10110011

Since $(bf)_8 = (10111111)_2$
bf << 1 = 01111110
As the most significant bit is 1 hence we need to add it with $(1b)_8 = (00011011)_2$
Therefore,

$$(02*bf) = 01111110 \oplus 00011011 = 01100101 \qquad (5)$$

Substituting values of equation (4) and (5) in r_1

Hence r_1 = 10110011 ⊕ 01100101 ⊕ 10111111 ⊕ 01011101 ⊕ 00110000
r1 = $(00000100)_2 = (04)_8$

$r_2 = (01*d4) \oplus (02*bf) \oplus (03*5d) \oplus (01*30)$ (6)

This can be further simplified using equation (2) as follow:
$r_2 = d4 \oplus (02*bf) \oplus (02*5d) \oplus 5d \oplus 30$
from equation (3) we can calculate the product value as follow, $(bf)_8 = (10111111)_2$
bf << 1 = 01111110
As the most significant bit is 1 hence we need to add it with $(1b)_8 = (00011011)_2$
Therefore,

(02*bf) = 01111110 ⊕ 00011011=01100101 (7)

Since $(5d)_8 = (010111101)_2$

5d << 1 = 101111010 (8)
As the most significant bit is 0 hence we do not need to add it with $(1b)_8 = (00011011)_2$
Substituting values of equation (7) and (8) in r_2 we get,
r_2 = 10110100 ⊕ 01100101 ⊕ 101111010 ⊕ 010111101 ⊕ 00110000
$r_2 = (01100110)_2 = (66)_8$

IV. CONCLUSION

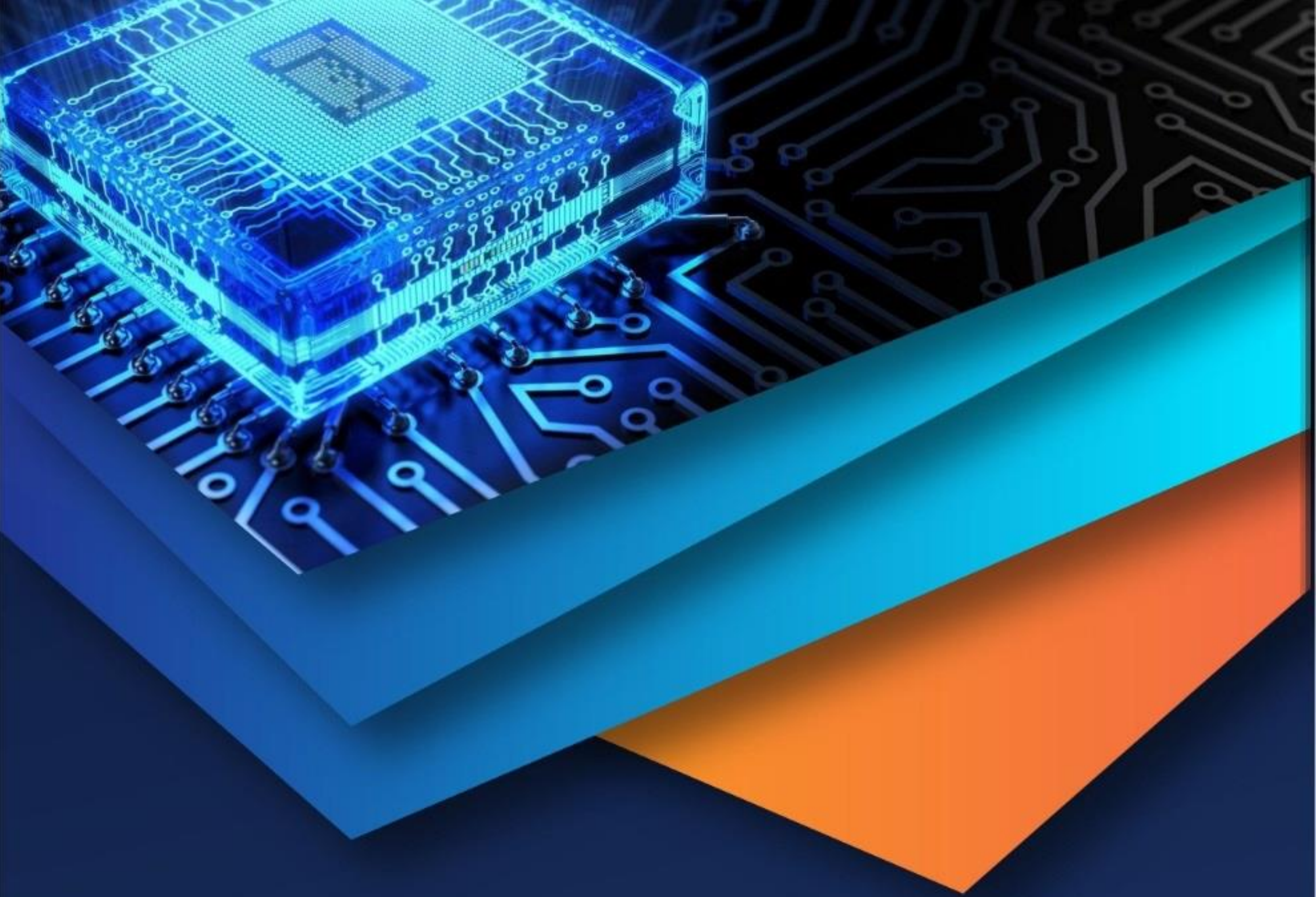
Mix column has been chosen from the space of 4-byte to 4- byte linear transformation according to following criteria:

A. Inevitability
B. Linearity in GF(2)
C. Relevant diffusion power
D. Speed on 8- bit processor
E. Symmetry
F. Simplicity of descriptor.

In this paper we have proposed an alternative design for mix column operation in the AES. The comparison indicate that the proposed mix column design have less complexity than previous relevant work in clock cycles. This design prevents timing attack on mix columns as the resultant columns take the same duration not depending on multiplicand.

REFRENCES

[1] William Stallings, Cryptography and Network security(Fourth edition),2005.
[2] Joan Daemen, Vincent Rijmen, AES proposal: Rijndael,2002
[3] Alex Biryukov and Dmitry Khovratovich, Related-Key Cryptanalysis of the FullAES-192 and AES-256, Advances in Cryptography, proceedings of ASIACRYPT,2009
[4] Behnam Bahrak, Mohammad Reza Aref, A Novel Impossible Differential Cryptanalysis of AES, proceedings of the Western European Workshop on Research in Cryptology 2007
[5] Joan Daemen, Vincent Rijmen, AES Proposal: Rijndael, NIST AES proposal, 1998.
[6] Joan Daemen, Vincent Rijmen The design of Rijndael: AES — the Advanced Encryption Standard, Springer-Verlag, 2002.



10.22214/IJRASET

INDEX COPERNICUS
45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL
IMPACT FACTOR:
7.429