International Journal of Research

Available at https://edupediapublications.org/journals

P-ISSN: 2348-6848

E-ISSN: 2348-795X

Volume 07 Issue05May 2020

Design of Light Weight Encryption Algorithm for Data Privacy in Light Weight Devices Using 128 Bit Key

Korada Yougeswari Devi[1] (Research Scholar), Dr. V.V.S.S.S. Chakravarthy[2] (Professor)

RAGHU INSTITUTE OF TECHNOLOGY AUTONOMOUS, DAKAMARRI, VISAKHAPATNAM, Andhra Pradesh - 531162

ABSTRACT

Internet of things (IoT), internetworking of smart devices, embedded with sensors, software, electronics and network connectivity that enables to communicate with each other to exchange and collect data through an uncertain wireless medium. Recently IoT devices are dominating the world by providing it's versatile functionality and real-time data communication. Apart from versatile functionality of IoT devices, they are very low battery powered, small and sophisticated, and experience lots of challenges due to unsafe communication medium. Despite the fact of many challenges, the energy issue is now becoming the prime concern. Optimization of algorithms in terms of energy consumption has not been explored specifically; rather most of the algorithms focus on hardware area to minimize it extensively and to maximize it on security issue as possible. But due to recent emerge of IoT a device, the main concern is shifting to moderate security and less energy consumption rate. This paper presents AES, a lightweight version of Advanced Encryption Standard (AES) which meets the demand and simulation results were verified in xilinx12.3 ISE Tool.

Keywords—AES, IoT and Resource Constraint Environments (RCEs).

I. INTRODUCTION

Internet of Things (IoT) is the following revolution of the internet which brings profound effect on our ordinary lives. IoT is the extension of the Internet to attach just about the entirety on earth. This includes real and physical gadgets ranging from family add-ons to business engineering. As such those "matters" which can be connected to the Internet will be able to take actions or make decisions primarily based on the information they collect from the Internet without or with human interaction.

In addition, additionally they replace the Internet with actual-time statistics with the assist of numerous sensors. IoT works with resource-constraint components consisting of sensor nodes, RFID tags and so on. These additives have low computation functionality, confined reminiscence potential and strength assets, and susceptibility to physical seize. Also, they communicate thru the Wi-Fi communication channel which is not secured and transmit actual-time facts thru the treacherous Wi-Fi medium. In certain applications,confidentiality,authentication, records freshness, and facts integrity might be extraordinarily essential. Encrypting statistics the usage of standardized cryptographic algorithms may also devour more strength which notably reduces the lifetime of the components. Two predominant approaches are followed to layout and implement protection primitives which might be equipped with extremely constraint devices. Firstly, designing new light-weight cryptosystem. For example, are some recently proposed lightweight cryptographic algorithms? Secondly, enhancing the present fashionable cryptosystem in a lightweight style. Possible examples of the second one method are modification of the Advanced Encryption Standard Algorithm (AES) , SHA-256 and so on.

With recognize to the security issue and implementation complexity, AES is taken into consideration as one of the strongest and efficient algorithms. Despite that like different symmetric encryption algorithms, the name of the game key distribution remains considered as a critical problem. Again to encrypt or decrypt a single block (128-bit) of statistics, an important quantity of computational processing must be accomplished which consumes great battery energy. As components of IoT have useful resource-constraint characteristics, consuming tremendous power may additionally reason expiration of such additives.

Analyzing related work, we come to realize that Substitution Layer is the maximum strength consuming part of AES within the round based totally design.

II. LITERATURE SURVEY

AES, especially to utilize beneath low electricity intake, high protection, higher performance and stepped forward efficiency. The implementation feasibility in VLSI environment is likewise studied and analyzed extensive. 25 Farhadian.A. A and Aref.M.R (2009) supplied green technique for simplifying and approximating the s-boxes primarily based on electricity functions [29]. In this paper cipher algorithms, strength functions over finite fields and unique inversion functions have an important position within the S-container design structure.

A new systematic green method is brought to crypt analyze such S-bins. This method is quite simple and does no longer need any heuristic strive and may be taken into consideration as a brief criterion to discover a few easy approximations. Using this new method, approximations can be received for advanced encryption trendy (AES) like S-containers, which includes AES, Camellia, and Shark and so forth. Finally as an application of this approach, a simple linear approximation for AES S-field is offered. Akashi Satoh, Sumio Morioka, Kohji Takano, and Seiji Munetoh (2011) supplied a Compact Rijndael Hardware Architecture with S-Box Optimization. Encryption and decryption records paths are combined and all mathematics components are reused.

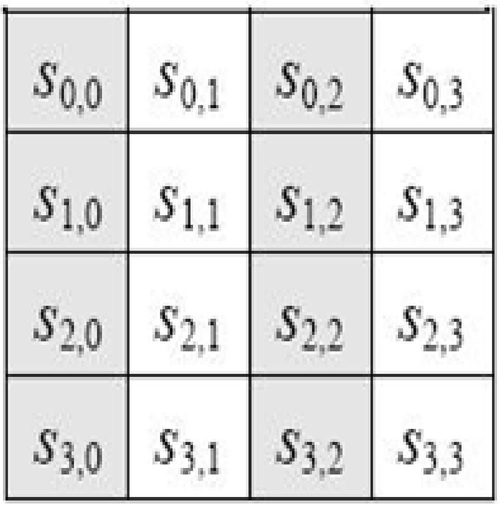
An extraordinarily small length of five.4 K gates is acquired for a 128-bit key Rijndael circuit using a 0.Eleven-μm CMOS trendy. In order to minimize the difficult-ware size, the orders of the mathematics features were modified, and the encryption-decryption statistics paths are correctly combined with respect to cell library.

Logic optimization strategies consisting of factoring were carried out to the arithmetic additives, and gate counts had been reduced.

III. AES (ADVANCED ENCRYPTION STANDARD):

The Advanced Encryption Algorithm (AES), a symmetric block cipher algorithm that can tactics blocks of 128-b, the usage of cipher keys with lengths of 128,192 and 256-bits. The enter and output for the AES set of rules each encompass collection of 128-b (digit with values of zero or 1). These sequences will occasionally seek advice from as blocks and the quantity of bits they include will be referred to as their length.

Internally, the AES algorithm's operation is are done on a -dimensional array of bytes called the state as proven within the Figure.1. The nation includes 4 rows of bytes, every containing Nb bytes, where Nb is the block length. Here Nb = four, which reflects the number of 32-b words (wide variety of columns) in the state.



$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

Fig1: State Array for AES set of rules, the duration of the cipher key, K, is 128,192 or 256 bits. The key length is represented by way of Nk = four, 6, or eight, which reflects the wide variety of 32-b phrases (variety of columns) in the cipher key. The number of rounds to be done all through the execution of the set of rules is established on the key size. The wide variety of rounds is represented via Nr, in which Nr = 10 when Nk = 4, Nr = 12 while Nk = 6 and Nr = 14 in which Nk = eight. The handiest key-block-spherical aggregate that confirms to the same old is given in Figure.2. Here we consider Nr = 10 and Nk = four for design of the architecture.

The Figure.3 shows the whole structure of AES algorithm (each encryption and decryption procedure). In FIPS 197 trendy, the block is depicted as square matrix of bytes. The block is copied into state array, which is changed at each degree of encryption or decryption approaches. Similarly, the 128-b key's depicted as a square matrix of bytes. The key is then extended into an array of key time table phrases, every phrase is of four-bytes and the overall key scheduled is forty four phrases for the 128-b key input. The Cipher and Decipher technique is defined in the pseudo code in Figure.4 and Figure.5.

The person transformation – Byte substitution, Shift row, Mix column and Add spherical key- methods the nation and is described within the following subsection. As proven within the Figure. Four, all Nr rounds are identical apart from the very last round, which does now not encompass Mix column transformation and as such equal in the decipher manner shown inside the Figure. Five. Let us now see the designated description of every of the 4 stages used in AES. For each degree each encryption and decryption transformation may be defined. This is accompanied with the aid of a discussion of key enlargement process utilized in AES.

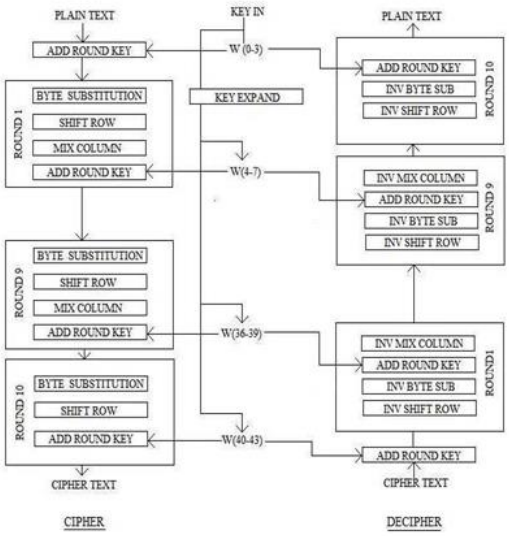
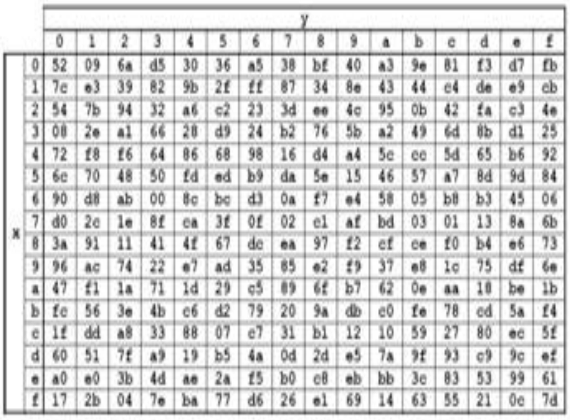


Fig2: AES Encryption and Decryption

A) Byte Substitution

The alternative byte transformation, known as the byte sub, is a easy table lookup. The system is shown within the Figure.3. AES defines a 16X16 matrix of byte values, called an S-Box that incorporates a permutation of all viable 256 8-b values which is shown in the Table.1. Each person byte of the state is mapped into a brand new byte inside the following way: The leftmost four-b of the byte cost is used as a row value and the rightmost 4-b are used as a column value. These row and column values serve as indexes into the S-Box to choose a precise 8-b output price. For example, the hexadecimal fee 95 references row nine, column 5 of the S-Box, which consists of the value ad. Accordingly, the price 95 is mapped into the fee advert. The manner of byte substitution is identical for the decryption method however it makes use of inverse S-Box as shown within the Table.2 that is carried out to every bite of the state.

Table 1: AES S-Box



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

FIG .3 AES Byte Substitution Operation

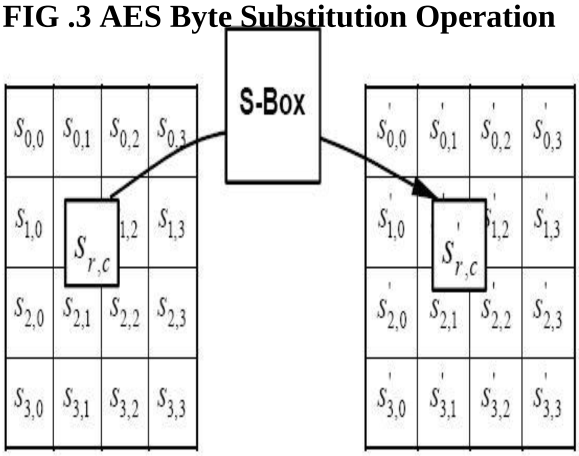
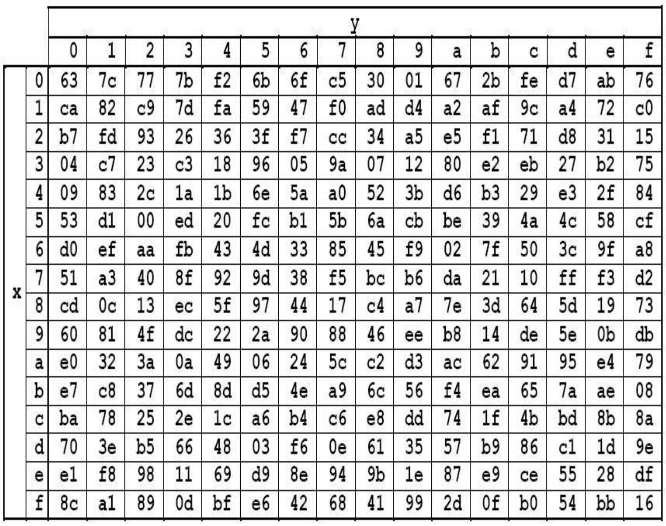


Table.2 AES Inverse S-Box



		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	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
	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

B) Shift Row

In this step, a regular left circular shift operation is carried out where within the first row isn't always altered and the subsequent 3 rows are moved via 1, 2, 3 bytes respectively. Conceptually, this is shown in the Figure.4

decryption transformation, the rows are subjected to right round shift wherein the first row is untouched and row 2, three and four are shifted by 1, 2 and 3 bytes respectively. This process is shown in the Figure.5.

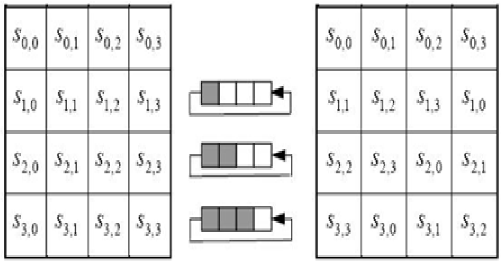


Fig4: AES Shift Row operation

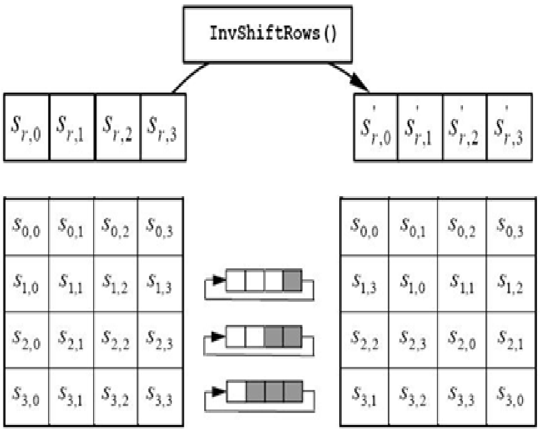


Fig5: AES Inverse Shift Row operation

C) Mix Column

The mix column transformation operates on the state column-by-column, treating each column as a four term polynomial. The column are considered as polynomial over GF (2^8) and multiplied with modulo x^4+1 with a fixed polynomial a(x), given by

$a(x) = \{03\}x^3+\{01\}x^2+\{01\}x+\{02\}$

Let s`(x) = a(x) x s(x):

result of this multiplication, the four bytes in a column are replaced by the following: And the illustrates of Mix column transformation is shown in the Figure.6 for the encryption process. Figure.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

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

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

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

$$s'_{3,c} = (\{03\} \bullet s_{0,c}) \oplus s_{1,c} \oplus s_{2,c} \oplus (\{02\} \bullet s_{3,c}).$$

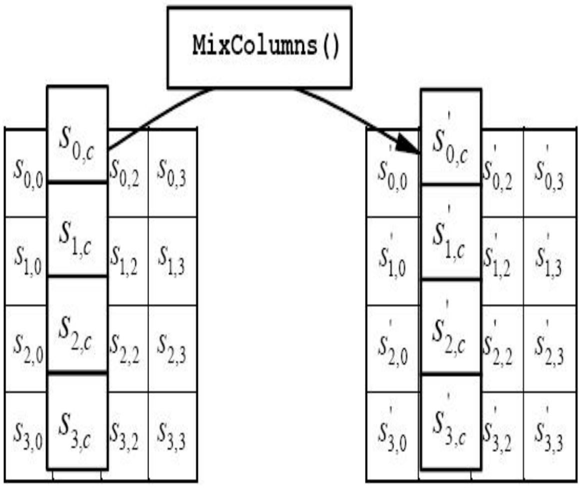


Fig6: AES Mix Column Transformation

While for the decryption process inverse of the mix column transformation is the Inverse mixture column , where in it is multiplied with fixed polynomial

$a^{-1}(x)$ $a^{-1}(x)$ $=\{0b\}x^3+\{0d\}x_2+\{09\}x+\{0e\}$

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Let s`(x) = a^{-1}(x) x s(x):

$$s'_{0,c} = (\{0e\} \bullet s_{0,c}) \oplus (\{0b\} \bullet s_{1,c}) \oplus (\{0d\} \bullet s_{2,c}) \oplus (\{09\} \bullet s_{3,c})$$

$$s'_{1,c} = (\{09\} \bullet s_{0,c}) \oplus (\{0e\} \bullet s_{1,c}) \oplus (\{0b\} \bullet s_{2,c}) \oplus (\{0d\} \bullet s_{3,c})$$

$$s'_{2,c} = (\{0d\} \bullet s_{0,c}) \oplus (\{09\} \bullet s_{1,c}) \oplus (\{0e\} \bullet s_{2,c}) \oplus (\{0b\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{0b\} \bullet s_{0,c}) \oplus (\{0d\} \bullet s_{1,c}) \oplus (\{09\} \bullet s_{2,c}) \oplus (\{0e\} \bullet s_{3,c})$$

D) Add Round Key

In the Add round key transformation, the 128-b of the state is bitwise XORed with128-b of the round key. As shown in the Figure.7, the operation is viewed as a column wise operation between the 4-bytes of a state column and one word of the round key; it can also be viewed as a byte level operation. The Add round key operation in decryption transformation is same, as XOR operation is its own inverse.

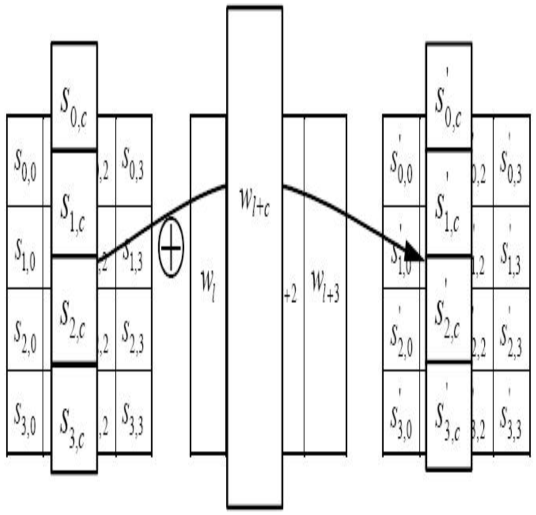


Fig7: AES Add Round Key Transformation

E) Key Expansion

The AES key expansion algorithm takes as input a 4-word (16-byte) key and produces a linear array of 44 words (176 bytes). This is sufficient to provide a four-word spherical key for the preliminary Add Round Key degree and every of the 10 rounds of the cipher. The key is copied into the first 4 words of the expanded key.

The rest of the expanded key is filled in four phrases at a time. Each introduced word w[i] rely on the right away previous word, w[i-1], and the word 4 positions back, w[i-4]. In three out of four cases, an easy XOR is used. For a phrase whose role inside the w array is a couple of 4, a greater complex function is used. Figure.11 illustrates the generation of the first eight phrases of the elevated key, the usage of the image g to constitute that complicated feature. The feature g consists of the following sub functions:

1. Rot Word performs a one-byte circular left shift on a word. This means

that an input word [b0, b1, b2, b3] is transformed into [b1, b2, b3, b0].

2. Sub Word performs a byte substitution on each byte of its input word, using the S-box.

3. The result of steps 1 and 2 is XORed with a round constant, Rcon[j] which is given in the Table.3.

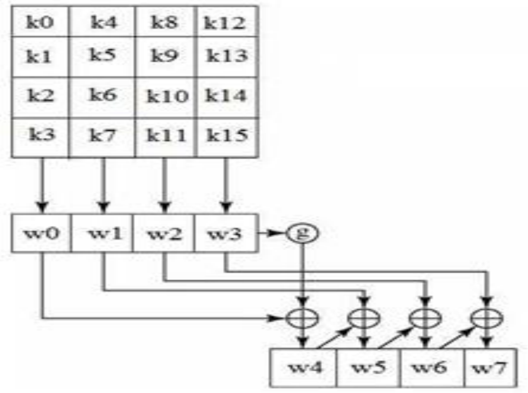
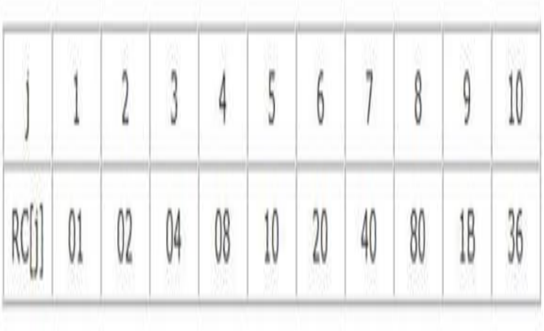


Fig8: AES Key Expansion



j	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36

Table3: Round Constant Rcon[j]

IV. RESULTS

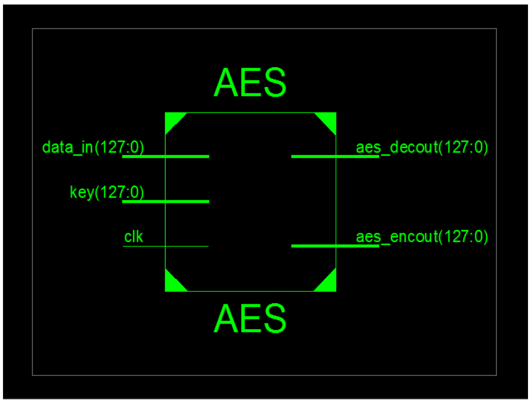


Fig 9: RTL Schematic view

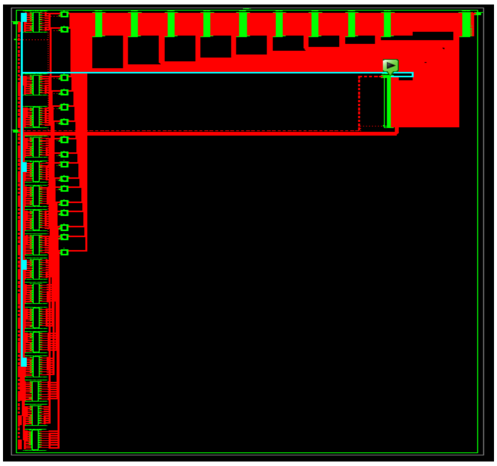


Fig 10: View Technology Schematic

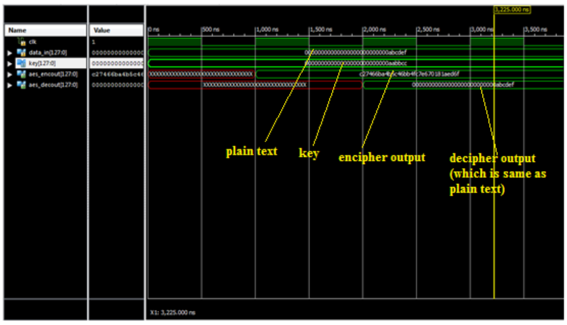


Fig 11: simulated waveforms

V. CONCLUSION AND FUTURE SCOPE

In this paper present a AES lightweight algorithm to provide security for data privacy that is core of IoT services for Resource-Constraint Environments. Design consumes less delay of 45.490ns and uses 20743 numbers of Luts. One notable feature of MAES is extending the battery life of low powered devices by consuming less amount of energy. In future, the security issue and space complexity will be considered to make the proposed modification more applicable. Also, plan to investigate multipath routing scheme while transmitting the encrypted data to the sink node. It can be used for the development of more advanced cryptographic algorithms in the field of light weight encryption technology, especially Elliptic-curve cryptography

(ECC) to achieve comparable efficiency in terms of number of packet transmission and latency with better security.

REFERENCES

[1] Madakam, Somayya, R. Ramaswamy, and Siddhartha Tripathi. "Internet of Things (IoT): A literature review." Journal of Computer and Communications 3, no. 05 (2015): p.164.

[2] Wang, Yong, Garhan Attebury, and Byrav Ramamurthy. "A survey of security issues in wireless sensor networks." IEEE Communications Surveys Tutorial (2006).

[3] Veeramallu, B., S. Sahitya, and Ch. LavanyaSusanna. Veeramallu, B., S. Sahitya, and Ch. LavanyaSusanna. "Confidentiality in Wireless sensor Networks." International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January 2013.

[4] Eisenbarth, Thomas, and Sandeep Kumar. "A surveyoflightweightcryptographyimplementations." IEEE Design & Test of Computers 24.6 (2007).

[5] Banik, Subhadeep, Andrey Bogdanov, and Francesco Regazzoni. "Exploring energy efficiency of lightweight block ciphers." International Conference on Selected Areas in Cryptography. Springer, Cham, 2015.

[6] Bogdanov, Andrey, et al. "PRESENT: An ultra-lightweight block cipher." CHES. Vol. 4727. 2007.

[7] Borghoff, Julia, et al. "PRINCEa low-latency block cipher for pervasive computing applications." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2012.

[8] Beaulieu, Ray, et al. "The SIMON and SPECK lightweight block ciphers." Design Automation Conference (DAC), 52nd ACM/EDAC/IEEE. IEEE, 2015.

[9] Suzuki, Tomoyasu, et al. "TWINE: A Lightweight Block Cipher for Multiple Platforms." Selected Areas in Cryptography. Vol. 7707. 2012

.[10] Li, Wei, et al. "Security analysis of the LED lightweight cipher in the internet of things." Jisuanji Xuebao (Chinese Journal of Computers) 35.3 (2012): p.434-445.