Journal of Engineering Sciences

Vol 11, Issue 7,July/2020

ISSN NO:0377-9254

DESIGN OF HIGH SPEED MODIFIDED ADVANCED ENCRYPTION STANDARD FOR THE SECURE COMMUNICATIONS

P.Charishma[1], A.Maheswar Reddy[2]

[1]*ECE Department, JNTUA (AITK) India, paidikalva94@gmail.com*

[2]*ECE Department, JNTUA (AITK) India, a.maheswarreddy@gmail.com*

***Abstract*— The Advanced Encryption Standard (AES) algorithm has become the default choice for different types of security services in so many applications. Internet of things(IoT),working 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. IoT devices are dominating the world by providing it's versatile functionality and real-time data communication. A part 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. But due to recent merge of IoT devices, the main concern is shifting to moderate security and less energy consumption rate. In existing MAES a new 1-dimensional Substitution Box is implemented by formulating a novel equation for constructing a square matrix in affine transformation phase of MAES where the area hardware is more. In proposed method the mix column is replaced by random permutation. The random permutation stage performs the Xor operation between the LFSR and state matrix from shift rows. this will improve the security level and also the area and delay will be reduced when compare to existing method.**

***Keywords*— AES, IoT, Energy Consumption, Resource Con- straint Environments (RCEs), TelosB, Cryptography.**

1. INTRODUCTION

Internet of Things (IoT) is the next revolution of the internet which brings profound impact on our everyday lives. IoT is the extension of the Internet to connect just about everything on the planet. This includes real and physical objects ranging from household accessories to industrial engineering As such these “things” that are connected to the Internet will be able to take actions or make decisions based on the information they gather from the Internet with or without human interaction. In addition, they also update the Internet with real-time information with the help of various sensors. IoT works with resource-constraint components such as sensor modes, RFID tags etc. These components have low computation capability, limited memory capacity and energy resources, and susceptibility to physical capture. Also, they communicate through the wireless communication channel which is not secured and transmit real-time information through the treacherous wireless medium. In certain applications, confidentiality, authentication, data freshness, and data integrity might be extremely important. Therefore, encryption of data is becoming a major concern. But due to resource-constraint nature of the components, short-term security can meet the demand. Encrypting data using standardized cryptographic algorithms may consume more energy which drastically reduces the lifetime of the components. Two main approaches are followed to design and implement security primitives which are fitted with extremely constraint devices. Firstly, designing new lightweight crypto system. For instance, are some recently proposed lightweight cryptographic algorithms. Secondly, modifying the existing standard cryptosystem in a light weight fashion. Possible examples of the second approach are modification of the Advanced Encryption Standard Algorithm (AES), SHA-256 etc.

With respect to the security aspect and implementation complexity, AES is considered as one of the strongest and efficient algorithms. Despite that like other symmetric encryption algorithms, the secret key distribution is still considered as a critical issue. Again to encrypt or decrypt a single block (128-bit) of data, an essential amount of computational processing has to be done

which consumes enormous battery power. As components of IoT have resource-constraint characteristics, consuming immense power may cause expiration of such components. Analyzing related work, we come to know that Substitution Layer is the most energy consuming portion of AES in the round based design. Considering energy consumption of resource-constrained components of IoT, we are proposing MAES, a lightweight version of AES where we reduce the computation of Substitution Box (S-Box) of AES

The existing work consists of 1-dimensional Substitution Box (S-Box) which is constructed by formulating a novel equation for constructing a square matrix in affine transformation phase of MAES. In this work they have implemented both original AES and MAES algorithms to compare the work. After analyzing the result of our experiment we conclude that MAES is well efficient than milliseconds in terms of number of packet transmission and latency, respectively. In existing system they have modified the S-box in this proposed system they have modified the mix column in order to get the better area and achieve high speed.. In the proposed system they have used the LFSR which generates pseudo random numbers which will improve the security level of the MAES.

2. RELATED WORKS

The Advanced Encryption Standard (AES), a symmetric key block which is published by the National Institute of Standards and Technology (NIST) in December 2001. It is a non-Feistel block cipher that encrypts and decrypts a fixed data block of 128-bits. There are three different key lengths. The encryption/decryption consists of 10 rounds of processing for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys.AES performs several rounds where each round is made of several stages. A data block is transformed from one stage to another. Before and after each stage, the data block is referred to as a state. Each round, except the last, performs four transformations which are invertible. The last round implements the rest three transformations except the Mix Columns stage. Figure 1 shows the AES cipher structure.

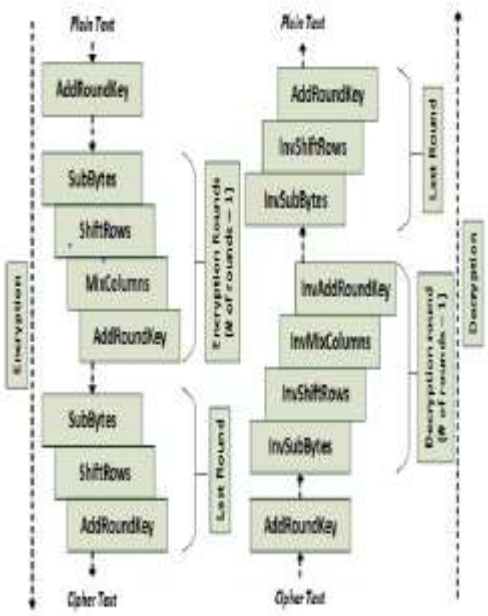

Fig.1.General design of AES encryption and decryption

4 stages of each round are:

1) *Substitute Bytes:* The first transformation, Sub Bytes, is used at the encryption site. It is a non linear byte substitution that operates independently on each byte of the state using a substitution table (S-Box). All the 16 bytes of the state are substituted by the corresponding values which are found from the lookup table. In decryption, Inv Sub Bytes is used. Bytes of a state are substituted from Inv Sub Bytes table.

Figure 2 shows the Sub Bytes operation.

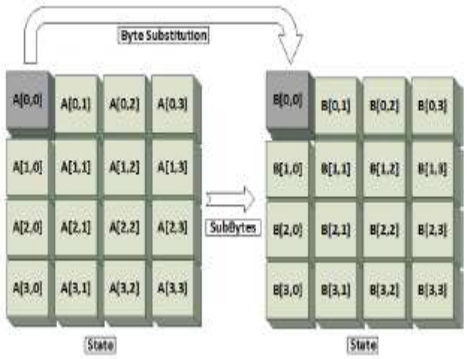

Fig. 2. Sub Bytes

2) Shift Rows: In the encryption, the state bytes are shifted left in each row. It is called Shift Rows

operation. The number of the shifts depends on the row number (0, 1, 2 or 3) of the state matrix. Row 0 bytes are not shifted and row 1, 2, 3 are shifted to 1, 2, 3 bytes left accordingly. Figure 3 shows the Shift Rows operation.

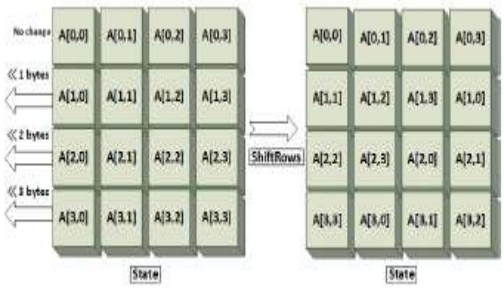

Fig. 3. Shift Rows

3) *Mix Column:* The Mix Columns transformation operates at the column level. It transforms each column of the state to a new column. The transformation is actually the matrix multiplication of a state column by a constant square matrix. All the arithmetic operations are conducted in the Galois Field(Finite Field). The bytes are treated as polynomials rather than numbers. Figure 4 shows the Mix Columns operation.

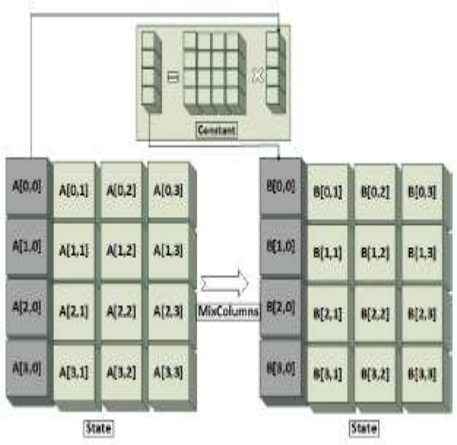

Fig. 4. Mix Columns

4) Add Round Key: Add Round Key proceeds one column at a time. It is similar to Mix Columns in this respect. Add Round Key adds a round keyword to each column matrix. Matrix addition operation is performed in the Add Round Key stage. Figure 5 shows the Add Round Key operation.

In encryption, Sub Bytes, Shift Rows, Mix Columns, and Add Round Key are performed in all rounds except the last round.

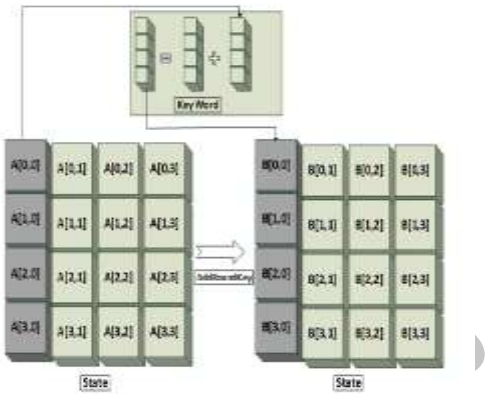

Fig. 5. Add Round Key

Mix Columns transformation operation is not performed in the last round of encryption. The decryption process essentially follows the same structure as the encryption, in addition to the nine rounds of Inverse Shift Rows, Inverse Sub Bytes, Inverse Add Round Key and Inverse Mix Columns Transformation. In the final round, Inverse Mix Columns is no longer performed.

3. EXISTING METHOD

MODIFIED ADVANCED ENCRYPTION STANDARD:

According to previous research observation, we have found out that S-Box and Mix Columns are the most energy consuming stages in encryption and decryption process. We have analyzed the S-Box generation process of the Rijndael AES. The 16x16 2-dimensional lookup table is formed through the multiplicative inverse phase and affine transformation phase in the original AES. We are proposing a new 1-dimensionallookup table as S-Box. It also follows the same generation

process as the original one. However, substitution of one complete byte requires two times substitution from the SBox. First four bits of the state byte is replaced first then the remaining four bits are substituted from the S-Box.

A. Rijndael S-Box Generation Method: The Rijndael S-Box is a square matrix which is used in the Rijndael cipher. The S-Box serves as a lookup table. It is generated by determining the multiplicative inverse for a given number in GF(28) and then

transforming the multiplicative inverse using affine transformation.

1) Multiplicative Inverse Phase: In multiplicative inverse phase, the input byte is inversed by substituting value from multiplicative inverse table.

2) Affine Transformation: Selection of the irreducible polynomial and the designated byte are the two most important factors of affine transformation phase. In Rijndael AES,x8 + x4 + x3 + x + 1 is used as the irreducible polynomial and as the constant column matrix 0x63 specially designated byte is chosen. Basically, the affine transformation consists of two operations. Firstly, 8x8 square matrix's multiplication and secondly, 8x1 constant column matrix addition. The 8x8 square matrix is constructed using the following

di=
bi_b(i+4)%8_b(i+5)%8_b(i+6)%8_b(i+7)%8_Ci (1)

bi =ith bit of multiplicative inverse of input byte (2)

Ci = ith bit of a specially designated byte (3)

Figure 6 illustrates the generation process of Substitution Box (S-Box) of the original AES.

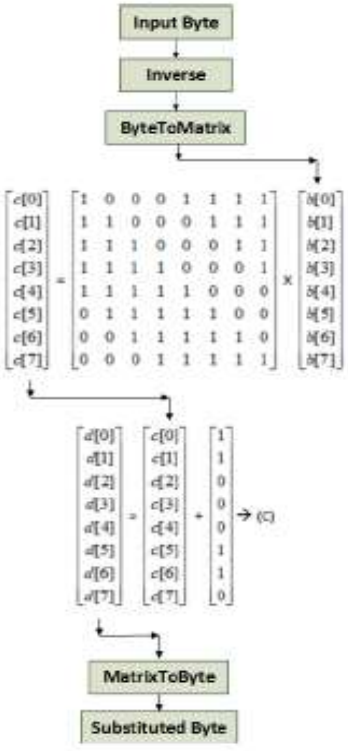

Fig.6. Original S-Box generation process

B. Modified AES S-Box Generation

Our modified AES S-Box generation process follows the construction procedure of the original AES. The whole process differs only in the selection of the irreducible polynomial and specially designated byte.

Multiplicative Inverse Table: In the Rijndael AES, all the arithmetic operations are performed over the Galois Field. In our work, the Galois Field is considered. The number of irreducible polynomials of degree 4 over GF are x4 + x + 1, x4 + x3 + x2 + x + 1 and x4 + x3 + 1. All the generated values of the multiplicative inverse table and substitution box depend on the selection of irreducible polynomial. For our experiment purpose, we choose x4+x+1 as our irreducible polynomial but we can select any of the irreducible polynomials which are mentioned above. Following the Extended Euclidean Algorithm, 1-dimensional multiplicative inverse table is formed. Figure 7 illustrates the multiplicative inverse table of the proposed algorithm.

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	1	9	E	D	A	7	6	F	2	C	5	A	4	3	8

Fig.7. Multiplicative inverse table

2) Affine Transformation: This affine transformation process also follows two phases. Firstly, 4x4 square matrix's multiplication and secondly, 4x1 constant column matrix addition. The 4x4 square matrix is constructed following the equation1 and equation 2 refers to the value of di:

di = bi _ b(i+2)%4 _ b(i+3)%4^Ci (4)

Ci = ith bit of a specially designated byte which is hexadecimal of 3; 8; 10; 13; 15 as they

don't generate any fixed points. (5)

Selection of the constant value is a little bit precarious. As we are calculating over the GF (24) where the value of the constant column matrix ranges from 0x00 to 0x0F, we can only select 5 values from there as these values do not generate any fixed point after transformation. The fixed point refers to the generation of the output value

same as the input value. Figure 8 shows the generation process of proposed MAES.

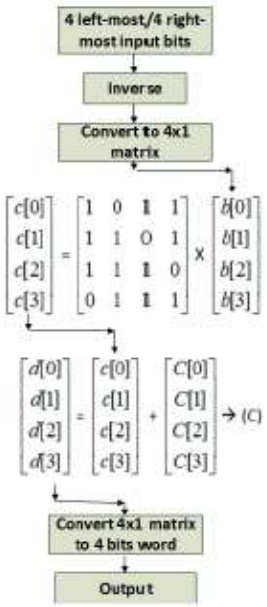

Fig.8.Proposed MAES S-Box generation process

Different S-Boxes and inverse S-Boxes for different values of the constant value C is given below from figure 9 to 13:

Case-1: When C = 0x03

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
3	4	F	B	2	1	7	0	C	D	5	9	6	E	A	8

Fig.9. Case-1: When C = 0x03

Case-2: When C = 0x08

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	F	4	0	9	A	C	B	7	6	E	2	D	5	1	3

Fig.10. Case-2: When C = 0x08

4. PROPOSED METHOD

Modifying the AES algorithm has been done several times before, each researcher limits his concerns to one characteristic to improve. The main characteristics that the researchers concentrate on are the speed of the AES. In existing method they have modified the s-box for the better performance but the area and delay are complex. Our proposed design will improve the performance in terms of area and delay.

Increasing the speed of the AES algorithm, while keeping the security level high is a vital for a lot of applications that require high security level with limited resources. The AES algorithm explained in the previous method suffers from consumption of unnecessary time to achieve the necessary complexity needed to meet the security level. A new modification for the AES algorithm (MAES) is done changing the s-box as well as by replacing the Mix Columns stage with random number generator. This will increase the speed of the algorithm without a decrease in the security of the AES algorithm. In addition, the security of the MAES algorithm can be enhanced using the permutation stage which uses random number generator.

The design of the MAES algorithm will ensure the following:

1. Speed up the AES algorithm by replace Mix Columns stage with simple xor operations.

2. To the xor operation the first input is state matrix coming from shift rows.

3. The second input for the Xor operation is random number generator like linear feedback shift register.

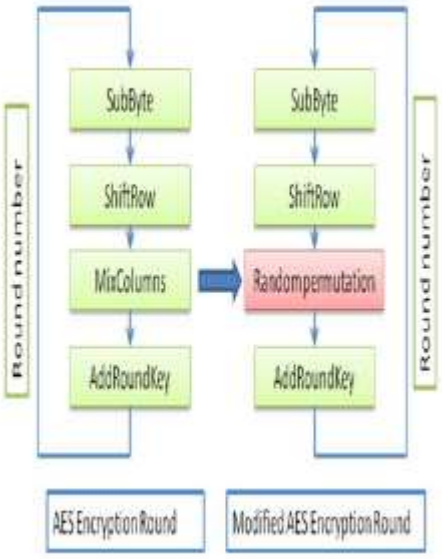

Figure 11.1 the proposed MAES algorithm compared with AES algorithm design.

The first goal of the MAES scheme is to increase the speed of the MAES algorithm. The Mix Columns stage is the most calculation demanding stage in the AES design and therefore it consumes most of the time needed for encryption and decryption. In the MAES design, the Mix Columns

stage is replaced with a xor operation between the input state and random vector which uses LFSR to generate random numbers

In the above figure 11.1 mix columns is replaced by the random permutation. The random permutations consist of xor operation between the state matrix and the random number generator. LFSR is used as a random number generator.

RANDOM NUMBER GENERATOR:

Linear-feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state. The most commonly used linear function of single bits is exclusive-or (XOR). Thus, an LFSR is most often a shift register whose input bit is driven by the XOR of some bits of the overall shift register value. An LFSR is a class of devices known as state machine. It is a shift register whose input bit is a linear function of its previous state. The only linear functions of single bits are XOR and XNOR. Thus it is a shift register whose input bit is driven by XOR or XNOR of some bits of overall shift register value. LFSR used in this work is 128-bit with maximum length feedback polynomial $x^{128} + x^{127} + x^{126} + x^{121} + 1$ for which 2^{128}-1 = 429, 49, 67,295 random outputs.

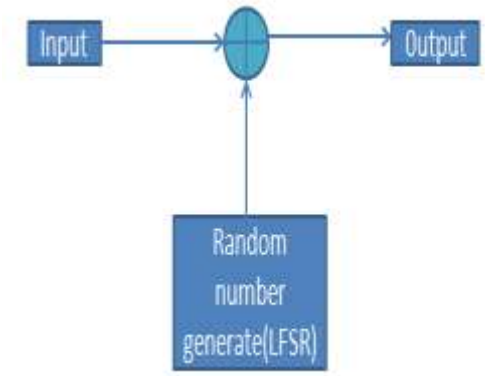

Figure 11.2: Random permutation stage

The above figure 11.2 shows the random permutation stage which consists of one input from the shift rows and another input from the 128 bit LFSR .The xor operation is performed from this two inputs and generate the random output which will act as input to the add round stage.

Comparation table:

Parameters	Existing method	Proposed method
Area(LUT's)	4281	2626
Delay(ns)	30.263	18.436

5. CONCLUSION

One of the widely used algorithms is advance encryption standard (AES), this algorithm suffer from consuming unnecessary time to achieve the complexity requirements needed for the encryption process specially for the real time application. Several modifications have been done on the algorithm to reduce the consuming time or to increase the complexity of the algorithm, but all the modification concentrate on one purpose which is decrease the consumption time or increase the complexity of the algorithm. In existing system a new s-box is proposed .In this proposed work, a new modification is applied on the AES algorithm by replacing the mix column with random permutation. The new algorithm which is called MAES can increase the speed of the algorithm processes and decreasing the area , while maintaining the complexity of the encryption as high as possible when compare to existing method.

6. REFERENCES

[1] Madakam, Somayya, R. Ramaswamy, and Siddharth Tripathi. "Internetof Things (IoT): A literature

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

[3] Veeramallu, B., S. Sahitya, and ChLavanyaSusanna. Veeramallu, B., S. Sahitya, and ChLavanyaSusanna. "Confidentiality in Wireless sensor Networks." I

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

[5] Banik, subhadeep, AndreyBogdanov, and Francesco Regazzoni. "Exploring energy efficiency

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

[7] Borgoff, Julia, et al. "PRINCEa low-latency block cipher for pervasive computing applications.

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

[9] Suzaki, 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."

[11] Shibutani, Kyoji, TakanoriIsobe, HarunagaHiwatari, Atsushi Mitsuda, Toru Akishita, and TaizoShirai. "Piccolo: An ultra-lightweight block cipher." In CHES, vol. 6917, pp. 342-357. 2011.

[12] Wu, Wenling, and Lei Zhang. "LBlock: a lightweight block cipher." In Applied Cryptography and Network Security, pp. 327-344. Springer Berlin/Heidelberg, 2011.

[13] Daemen, Joan and Rijmen, Vincent. "The design of Rijndael: AES-theadvanced encryption standard.", Springer Science & Business Media, 2013.

[14] Descriptions of SHA-256, SHA-384, and SHA-512. http://csrc.nist.gov/groups/STM/cavp/documents/shs/sha256-384-512.pdf.