International Journal of Scientific Engineering and Technology Research

ISSN 2319-8885
Vol.04,Issue.33,
August-2015,
Pages:6553-6557

www.ijsetr.com

Least Complex S-Box and Its Fault Detection for Robust Advanced Encryption Standard Algorithm

P. LAHARI[1], V. TEJU[2], K. VENKATESWARLU[3]

[1]PG Scholar, Dept of ECE, Jawaharlal Nehru Institute of Technology, Hyderabad, TS, India, E-mail: laharipadala@gmail.com.
[2]Assoc Prof, Dept of ECE, Jawaharlal Nehru Institute of Technology, Hyderabad, TS, India, E-mail: teju_51@yahoo.com.
[2]Assoc Prof & HOD, Dept of ECE, Jawaharlal Nehru Institute of Technology, Hyderabad, TS, India, E-mail: ecehodjnit2@gmail.com.

Abstract: Cryptography is a method that has been developed to ensure the secrecy of messages and transfer data securely. The Advanced Encryption Standard (AES) is the newly accepted symmetric cryptography standard for transferring block of data securely. However, the natural and malicious injected faults reduce its reliability and may cause confidential information leakage. The objective of this paper is to find optimized fault detection schemes for reaching reasonable fault coverage in the high performance AES implementations. In order to provide low cost complexity signature, two sets of error indication flag is used. This structure can be applied to both look-up tables and logic gate for the implementation of S-box and inverse S-box and their parity predictions. Defects in the logic gates caused either by the natural faults or malicious injected faults that are detected independent of the method the S-box is implemented Moreover, the overhead costs, including space complexity and time delay of the proposed schemes are analyzed. Finally, our simulation results show the error coverage of greater than 99 percent for the proposed schemes.

Keywords: AES Algorithm, Composite Field S-Box, Error Coverage, Galois Field, Parity Based Fault Detection.

I. INTRODUCTION

In today's digital world, encryption is emerging as a disintegrable part of all communication networks and information processing systems, for protecting both stored and in transit data. There are other important drawbacks in software implementation of any encryption algorithm, including lack of CPU instructions operating on very large operands, word size mismatch on different operating systems and less parallelism in software. In addition, software implementation does not fulfill the required speed for time critical encryption applications. Thus, hardware implementation of encryption algorithms is an important alternative, since it provides ultimate secrecy of the encryption key, faster speed and more efficiency through higher levels of parallelism. Cryptography is a method that has been developed to ensure the secrecy of messages and transfer data securely. In digital communications the data is sent through the wires or air and thus it is not protected from eavesdropping. Therefore, confidentiality of the transferring data is of extreme importance. Encryption is a process which transforms the data that is aimed to be sent to an encrypted data using a key. The encryption process is not confidential but the key is only known to the sender and receiver of data. The receiver transforms the received data using the decryption process to obtain the original data. Symmetric key cryptography is a form of cryptosystem in which encryption and decryption are performed using the same key. It has been utilized for secure communications for long period of time. Symmetric key cryptography comprises two different methods for encryption and decryption. It can either use stream cipher or block cipher method of encryption/ decryption.

The National Institute of Standards and Technology initiated a process to select a symmetric key encryption/ decryption algorithm in 1997. Finally, Rijndael algorithm was accepted among other finalists as the Advanced Encryption Standard (AES) in 2001. It is noted that before the acceptance of Rijndael algorithm, DES and its improved variant 3DES were used as symmetric key standards. DES has 16 rounds and encrypts and decrypts data using a 64-bit key. 3DES has hardware implementation that doesn't produce efficient software code and three times as many rounds as DES so correspondingly slower. Both DES and 3DES use a 64-bit block size. To satisfy both efficiency and security, a larger block size is desirable. AES-128 has 10 rounds where data is encrypted and decrypted in 128-bit blocks using a 128-bit key. It is a very good performer in both hardware and software across a wide range of computing environments. The objective in using AES is to transfer the data so that only the desired receiver with a specific key would be able to retrieve the original data. However, the natural and malicious injected faults reduce its reliability and may cause confidential information leakage. This can be either due to:

- Natural faults caused by defects in gates or,
- Malicious injected faults to retrieve the key and break the system.

As a result, finding a suitable fault detection scheme has always been an issue in the AES. FPGAs are most flexible implementation to produces high performance with low cost. FPGA provides more physical security with parallelism.

A. Key Based Approach

Different versions of AES algorithm exist today (AES128, AES196, and AES256) depending on the size of the encryption key. In this project, a hardware model for implementing the AES128 algorithm was developed using the Verilog hardware description language. A unique feature of the design proposed in this project is that the round keys, which are consumed during different iterations of encryption, are generated in parallel with the encryption process.

B. Language

The hardware model was then completely verified using a test bench, which took advantage of the Verilog, is programming feature, by constructing random test objects and providing them to the model. Then, the verified model was synthesized using the Synopsis Design-Compiler tool to get an estimate of the number of gates, area and timing of the hardware model. Finally, the performances of software and hardware implementations were compared.

C. Finite Fields

In this section, the preliminaries on finite fields (also known as Galois fields) used in the subsequent sections are presented. The detailed description of these fields can be found in a number of publications, see for example. According to Lin and Costello, the definition of a finite field is as follows. Let F be a set of elements on which two binary operations of addition and multiplication, shown by " + " and " · ", respectively, are defined. Then, the set F and these operations construct a finite field if the following conditions are satisfied:

- The set F be commutative under addition. The identity element in addition is zero.
- The non-zero elements of set F be commutative under multiplication. The identity element in multiplication is one.
- Multiplication be distributive over addition, i.e., for a, b and c in set F we have $a \cdot (b + c) = a \cdot b + a \cdot c$.
- The number of elements in the field is finite. In the AES, the irreducible polynomial of P(x) = x^8 + x^4 + x^3 + x + 1 is used to construct GF (2^8). Each element in GF (2^8) is represented by a polynomial of degree 7, having 8 coefficients in GF (2). Furthermore, all the field operations are carried out using the above mentioned irreducible polynomial.

D. Cryptosystems And Public Key Cryptography

Cryptography is the process of encrypting the plain text into an incomprehensible cipher text by the process of Encryption and the conversion back to plain text by process of Decryption. Most encryption algorithms are based on 2 general principles,

- Substitution, in which each element in plain text is mapped to some other element to form the cipher text
- Transposition, in which elements in plaintext are rearranged to form cipher text.

E. Number of Keys Used

If both the sender and the receiver use a same key then such a system is referred to as Symmetric, single-key, secret-key or conventional encryption. If the sender and receiver use different keys, then such a system is called Asymmetric, Two-key, or public-key encryption Processing of Plain text: A Block cipher processes the input one block at a time, producing an output block for each input block. A Stream cipher processes the input elements continuously producing output elements on the fly. Most of the cryptographic algorithms are either symmetric or asymmetric key algorithms.

1. Secret Key Cryptography: This type of cryptosystem uses the same key for both encryption and decryption. Some of the advantages of such a system are

- Very fast relative to public key cryptography
- Considered secure, as long as the key is strong

Symmetric key cryptosystems have some disadvantages too. Exchange and administration of the key becomes complicated. Non-repudiation is not possible. Some of the examples of Symmetric key cryptosystems include DES, 3-DES, RC4, RC5 etc.

2. Public Key Cryptography: This type of cryptosystems uses different keys for encryption and decryption. Each user has a public key, which is known to all others, and a private key, which remains a secret. The private key and public key are mathematically linked. Encryption is performed with the public key and the decryption is performed with the private key. Public key cryptosystems are considered to be very secure and supports Non-repudiation. No exchange of keys is required thus reducing key administration to a minimum. But it is much slower than Symmetric key algorithms and the cipher text tend to be much larger than plaintext. Some of the examples of public key cryptosystems include Diffie-Hellman, RSA and Elliptic Curve Cryptography.

II. ADVANCED ENCRYPTION STANDARD

A. AES Algorithm

AES is an iterated block cipher with a fixed block size of 128 and a variable key length. It has variable number of rounds, which is fixed according to key length. AES performs four transformations in each round in order to provide high level of security.

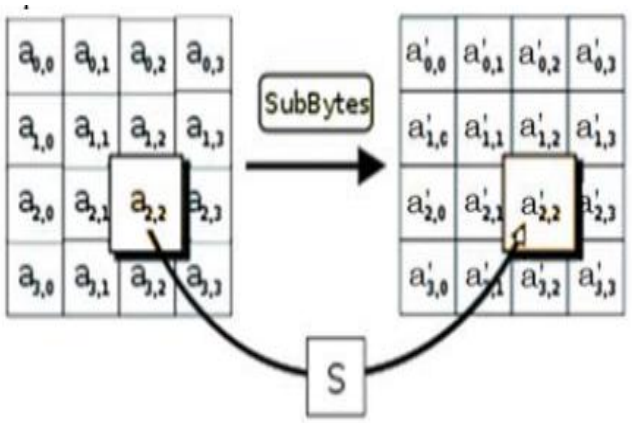

Fig.1. Sub Bytes.

B. Transformations

AES performs four transformations in each round in order to provide high level of security. This involves the properties of confusion and diffusion to provide frustrating statistical cryptanalysis. The transformations in each round of encryption except for the last round are as follows:

1. Sub Bytes: It is a non-linear substitution step where each byte is replaced with another according to a lookup table. The look table is known as S-Box which is generated by applying affine transform to multiplicative inverse of input as shown in Fig.1.

2. Shift Row: It is a transposition step where each row of the state is shifted cyclically a certain number of steps to the left. For AES, the first row is left unchanged. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively. Similarly for decryption rows are shifted right as shown in Fig.2.

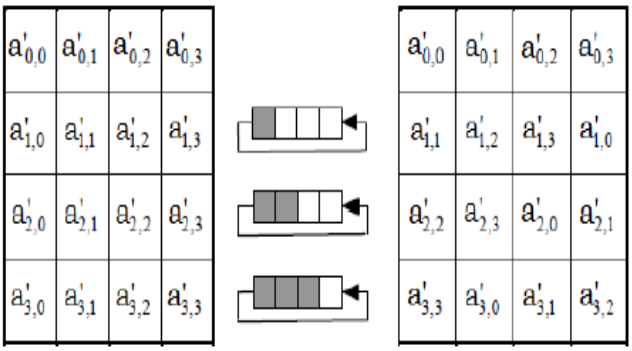

Fig.2. Shift Row.

3. Mix Column: It is a mixing operation which operates on the columns of the state, combining the four bytes in each column using an invertible linear transformation as shown in Fig.3.

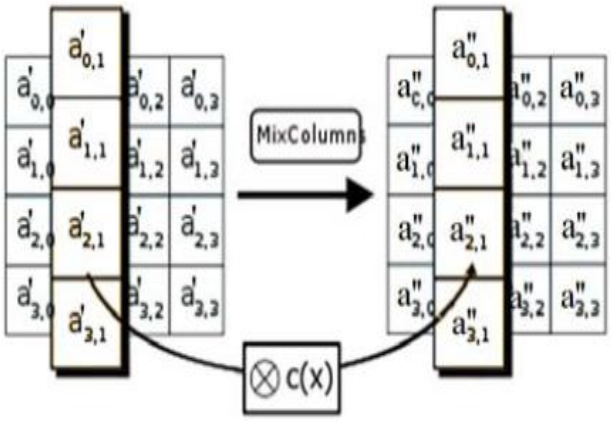

Fig.3. Mix Column.

During this operation, each column is multiplied by the known matrix that for the 128 bit key is

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \quad (1)$$

4. Add Round Key: In this, each byte of the state is combined with the round key; each round key is derived from the cipher key using a key schedule. The round Key is added to the state before starting the loop. In the Add Round Key step, each byte of the state is combined with a byte of the round sub key using the XOR operation as shown in Fig.4.

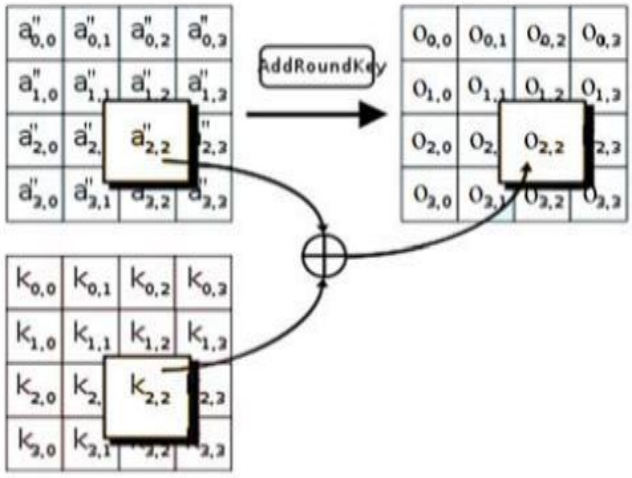

Fig.4. Add Round Key.

III. EXISTING SYSTEM

There has been many fault detection schemes proposed till this date to avoid the possibility of suffering from various attacks such as natural faults caused by defects in gates ,injection of fault by attackers to retrieve the key. Some of the majorly contributed schemes follow as:

A. A 16-Bit Key Parity Method

In this scheme, the output parity bits of each transformation in every round are predicted from the inputs. Then, the comparisons between the predicted and the actual parities (obtained using the actual parity block or predetermined parity block) can be scheduled so that the desired error coverage is obtained. Since the 128 bit input is represented in 4X4 matrix 16 parity bits corresponding to each 1 byte are compared which is presented. It has drawback that requires two blocks of 256 x 9 memory cells (S-boxes and parity predictions box). So it has relatively high area complexity for the parity predictions of Mix Columns in the AES encryption. This is even more for Inv Mix Columns in the AES decryption.

B. Redundancy-Based Technique

The redundancy-based solution for implementing fault detection in the encryption module is based on the idea of performing a test decryption immediately after the encryption and then checking whether the original data block is obtained. The redundant unit fault detection scheme is used where algorithm-level, round-level, or operation-level fault detections are considered. The schemes pays time penalty either to decrypt a data block or for the comparison.

C. Double Time Transformation Technique

In, the scheme uses same transformations twice in an AES round for the same data to detect the transient errors. It is time consuming and hence increases delay overhead.

However, this method suffers from permanent internal faults or the malicious injected faults lasting for a long period.

D. Multiplication-Based Scheme

In scheme, the result of the multiplication of the input and the output of the multiplicative inversion is compared with the predicted result of unity.

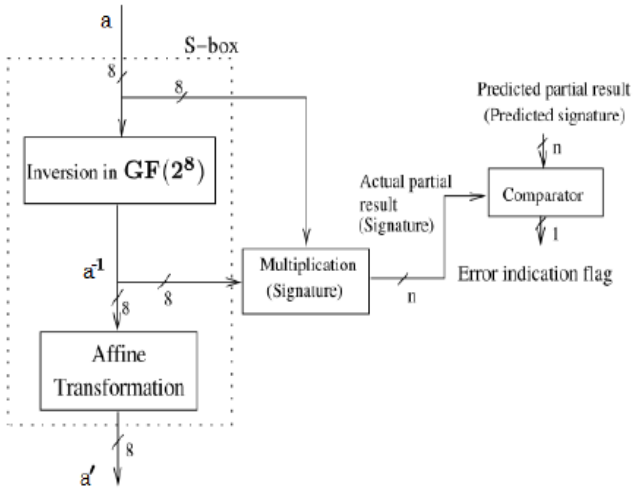

Fig.5.The multiplication-based scheme for the fault detection of the multiplicative inversion.

Since S-box is generated by applying affine transform to multiplicative inverse of an input, there is no access to the output of the multiplicative inversion as shown in Fig.5. Therefore, this scheme is not suitable for the S-boxes and inverse S-boxes implemented using lookup tables (LUTs).

E. Pipelined Structure

Under this, pipelined distributed memories for the LUT-based S-boxes and inverse S-boxes are used to increase the design speed and the overall frequency of AES. There are three architecture to speed up, namely pipelining, sub pipelining, and loop unrolling. Among these approaches, the sub pipelined architecture can achieve maximum speed up and optimum speed–area ratio in non-feedback modes. Sub pipelining inserts rows of registers among combinational logic not only between but also inside each round unit which is presented. Non-LUT-based approaches can be used to avoid the unbreakable delay of LUTs which involve inversions in Galois Field, which may have high hardware complexities.

IV. IMPLEMENTATION RESULTS

The S-box and inverse S-Box are designed using logic gates for reducing the hardware complexity and detecting the faults in it. The multiple faults can be applied to the S-box and inverse S-box and the output is noted. Then the S-box and inverse S-box is tested using FPGA kit. The simulated result can be obtained using Model Sim SE plus 6.4c and synthesis is carried out using Xilinx ISE. Then the delay and error coverage can be calculated using the simulated results. They are explained below.

A. Delay and Area

The number of LUTs and slices used to design the S-box and inverse S-box is calculated from the simulation results. Table-I represents the comparison of the number of LUTs and slices used for designing the S-box and inverse S-box.

TABLE I: Comparison of Luts and Slices

	No. of 4 input LUTs	No. of slices
LUT based S-box	250	158
LUT based inverse S-box	250	158
Low power S-box	87	46
Low power inverse S-box	84	44
Proposed S-box	71	41
Proposed inverse S-box	69	39

TABLE II: Gate Delay an D Net Delay

	Gate delay	Net Delay	Total Delay
Proposed S-Box	11.851 ns	12.988 ns	24.839 ns
Proposed inverse S-Box	11.581 ns	12.008 ns	23.589ns

TABLE III: Error Coverage

Faults	Error Coverage
Single fault in S-box	100%
Single fault in inverse S-box	100%
Multiple faults in S-box	99%
Multiple faults in inverse S-box	98%

From Table-I, the proposed S-box contains less number of LUTs and slices when compared to low power S-box and S-box using LUTs. The gate delay and net delay of the proposed S-box and inverse S-box are given in Table-II.

B. Error Coverage

The proposed S-box and inverse S-box is able to find the Stuck-at faults and random faults in the circuit. The faults are injected in any blocks of the S-box and inverse S-box using logic gates. In case of multiple faults injected in S-box, 100 faults have been injected in the circuit. Out of these, 99 faults have been recognized. In case of multiple faults in inverse S-box, out of 100 faults 98 have been recognized. Table-III represents the error of S-box and inverse S-box. Using the simulation results, the error coverage is calculated. From Table-III, the error coverage in S-box and inverse S-box is approximately 99.25%.

C. Simulation Result

The S-box and inverse S-box are designed and simulated using Modelsim. The simulated result for the S-box and inverse S-box is revealed in Fig. 6 and Fig. 7. The simulated output for fault detection in S-box when there is no fault in the circuit is as shown in Fig.8. From the figure, we conclude that there is no fault in the S-box circuit where all the error indication flag of the five blocks in the S-Box is zero.

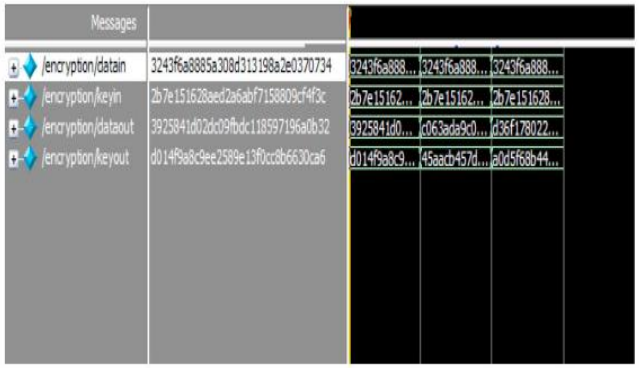

Fig.6. Simulated output for AES Encryption.

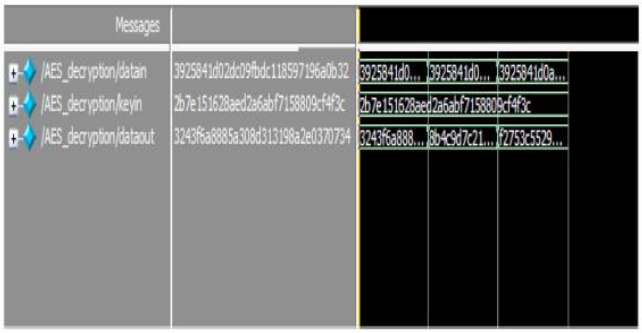

Fig.7. Simulated output for AES Decryption.

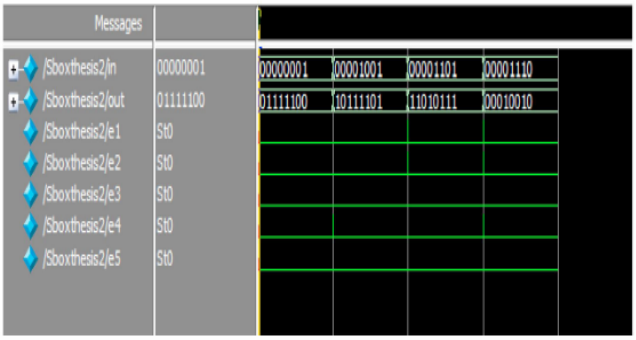

Fig.8. Simulated output for Fault Detection in S-box.

V. CONCLUSION

In our project we perused the concept of Cryptography including the various schemes of system based on the kind of key and a few algorithms such as RSA and AES. We studied in detail the mathematical foundations for AES based systems, basically the concepts of rings, fields, groups, Galois finite fields and their properties. The various algorithms for the computation of the scalar product of a point were studied and their complexities were analyzed. The advantage of this over the other Fault detection systems are proved by parameters .The key strength of this systems in comparison to other is fault detection is implanted in all levels of algorithm implementation and this will increase reliability. The proposed system is able to find the round and its corresponding transformation in which fault occurred. Thereby optimized hardware is achieved by modifying the structure using sub expression sharing. Hence the reduced number of gates is required in the implementation of AES. The slice overheads are less than those for the other schemes which have the same error coverage. Thus, this scheme has the highest efficiencies, showing reasonable area and time complexity overheads. Hence the proposed schemes outperform the previously reported ones.

VI. REFERENCES

[1] G. Alisha Evangeline, Mrs. S. Krithiga, Mrs. S. Sheeba Rani Gnanamalar, “Least Complex S-Box and Its Fault Detection for Robust Advanced Encryption Standard Algorithm”, IEEE 2013.

[2] Subashri T, Arunachalam R, Gokul Vinoth Kumar B, and Vaidehi V, "Pipelining Architecture of AES Encryption and Key Generation with Search Based Memory," International journal of VLSI design & Communication Systems (VLSICS), You, No.4, December 2010.

[3] J. Vijaya and M. Rajaram, "High Speed Pipelined AES with Mix Column Transform," European Journal of Scientific Research, ISSN 1450-216X Vol.61 No.2 (2011), pp. 255-264.

[4] Priyanka Pimpale, Rohan Rayarikar, Sanket Upadhyay, "Modifications to AES Algoritlun for Complex Encryption," IJCSNS International Journal of Computer Science and Network Security, Vol.11 No.1 0, October 2011.

[5] Aluned. H. Sawahneh, "Hardware Design of AES S-box using pipelining structure over GF ((24i)".

[6] K.Rahimwmisa, Dr. S. Sureshkumar, and K.Rajeshkumar, "Implementation of AES with New S-Box and Performance Analysis with the Modified S-Box," International Conference on VLSI, Communication & Instrumentation (ICVCI) 20J J Proceedings published by International Journal of Computer Applications® (IJCA).

[7] MooSeop Kim, Juhan Kim, and Yongje Choi, "Low Power Circuit Architecture of AES Crypto Module for Wireless sensor Network," World Academy of Science, Engineering and Technology 8, 2007.

[8] M.Pitchaiah, Philemon Daniel, and Praveen, "Implementation of Advanced Encryption Standard Algorithm," International Journal of Scientific & Engineering Research, Volume 3, Issue 3, March -2012 1 ISSN 2229-5518.

[9] Zine EI Abidine, Alaoui Ismaili, and Aluned MOUSSA, "Self-Partial and Dynamic Reconfiguration Implementation for AES using FPGA," IJCSI International Journal of Computer Science, Issues, Vol. 2, 2009 ISSN (Online): 1694-0784 ISSN (Print): 1694-0814.