### Research Article Design of Low Complexity Fault Detection Scheme for AES using Composite Field Arithmetic

<sup>1</sup>G.I. Shamini and <sup>2</sup>Sunil Raj

<sup>1</sup>Department of Electronics and Communication Engineering, Sathyabama University, Chennai, <sup>2</sup>Department of Electronics and Communication Engineering, Government Engineering College, Idukki, India

**Abstract:** The Advanced Encryption Standard (AES) is the symmetric cryptography standard that can be used to protect the electronic data. The natural and malicious injected faults may cause confidential information leakage and also reduce its reliability. In this study, we have explained a low complexity fault detection schemes for the AES architecture. The proposed work is low-complexity fault detection schemes using composite fields in polynomial basis for the AES encryption and decryption. These schemes are independent of the existing S-box and inverse S-box constructed. Here we have developed a new technique for the fault detection of subbyte and inverse subbyte using multiplicative inversion and affine transformation of the S-box and the inverse S-box. These are constructed in S-box and the inverse S-box. So this scheme can be used for the S-boxes and the inverse S-boxes in composite fields subbyte and inverse subbyte and using ROM. The proposed AES Fault detection scheme is coded in VHDL (Very High Speed Integrated Circuits Hardware Description Language), synthesized and simulated using EDA (Electronic Design Automation) tool-XilinxISEVirtex FPGA (http://www.xilinx.com/.). Finally the results are compared with Conventional ROM based subbyte and inverse subbyte to show the significant improvement in its efficiency in terms of path delay, speed and area.

Keywords: Advanced Encryption Standard (AES), composite field, decryption, encryption, fault detection, polynomial basis, S-box

### **INTRODUCTION**

The Advanced Encryption Standard (AES) is the symmetric key cryptography standard that can encrypt and decrypt the electronic data. In encryption, AES accepts a plaintext (which is limited to 128 bits) and a key for generating the ciphertext. The key can be specified to be 128 bits (AES-128). In AES-128, the ciphertext is generated after 10 cycles of repetition. For encryption, each round, except the final round, consists of four transformations which includes Sub Bytes (which is implemented by 16 S-boxes), Shift Rows, Mix Columns, AddRoundKey. The decryption transformations are the reverse of the encryption transformations which is utilised to obtain original plain text from the cipher text. Among the transformations, the nonlinear ones are the S-boxes in the encryption and the inverse S-boxes in the decryption. It occupies much of the total AES encryption or decryption area.

There exist many schemes for detecting the faults in the AES hardware implementation, see for example (Karri *et al.*, 2002; Rijmen, 2000; Satoh *et al.*, 2001; Satoh *et al.*, 2008; Mozaffari-Kermani and Reyhani-Masoleh, 2008). Among them, the schemes presented in Karri *et al.* (2001) and Maistri and Laveugle (2008) are independent of the ways the S-box and inverse S-box in the hardware implementation. The fault detection schemes using memories (ROMs) for the S-box and the inverse S-box are there. Further rmore, a fault tolerant scheme which is resistant to fault attacks is presented in Moratelli *et al.* (2008).

Either the parity-based scheme proposed in Bertoni et al. (2002) or the duplication approach is implemented to protect the combinational logic blocks used in the four transformations of the AES. Moreover, for storing the expanded key and the state matrix, either the Reed-Solomon error correcting code or Hamming code is utilized for protecting the memories. Our proposed scheme is only applied to the S-box and inverse S-box in composite field polynomial basis. While, the scheme presented in Bertoni *et al.* (2003); Wolkerstorfer *et al.*, (2002) uses memories. But for high performance, using ROMs are not preferable. Thus, for high performance AES, the S-box and the inverse S-box are implemented using logic gates in composite fields (Canright, 2005; Yen and Wu, 2006).

Thus the schemes suitable for the S-box and the inverse S box in composite field implementation are

Corresponding Author: G.I. Shamini, Department of Electronics and Communication Engineering, Sathyabama University, Chennai, India

This work is licensed under a Creative Commons Attribution 4.0 International License (URL: http://creativecommons.org/licenses/by/4.0/).

obtained in Kermani and Reyhani-Masoleh (2006) and Mozaffari-Kermani and Reyhani-Masoleh (2008). The approach in Kermani and Reyhani-Masoleh (2006); Karpovsky et al. (2004); Wu and Yen (2006) is based on using the parity-based fault detection method for a specific S-box in composite field and polynomial basis for covering all the single malicious faults. For the multiplicative inversion of the S-box, two specific composite fields are treated. Though the transformation and affine matrices are excluded in this approach. Furthermore, in Cohen (2007) Zhang and Parhi (2004, 2006), the fault detection scheme for the multiplicative inversion of a S-box in composite field polynomial basis, the systematic method including predicted parities have been used. The transformation matrices are also advised. Finally, in the parity-based approach in Mozaffari-Kermani and Reyhani-Masoleh (2008), through exhaustive search among all the fault detection S-boxes utilizing five predicted parities using polynomial basis, utmost compact one is obtained. The main objective of the work is to obtain low complexity fault detection schemes using composite field and the result is compared with conventional ROM to get efficient path delay, speed and area.

**AES encryption:** In this section, we briefly explain about the four transformations used in the AES encryption and decryption (National Institute of Standards and Technologies, 2001). In the AES-128 (128-bit key) transformation implementations, the irreducible polynomial of  $P(x) = x^8 + x^4 + x^3 + x + 1$  is used for constructing the binary field GF(2<sup>8</sup>). Each transformation in every round acts on its 128-bit input denoted as the state. The states are considered as  $4 \times 4$ matrices whose entries are 8 bits. For example, the input state S with its 8-bit entries, i.e., s <sub>r,c</sub>,  $0 \le r$ ,  $c \le 3$ , is represented as follows:

$$S = [s_{r,c}]^{3}_{r,c} = 0$$
(1)

Considering (1) as the input state of an encryption round. The transformations in each round, except the final round, are as follows:

**SubBytes:** In each round the first transformation is the bytes substitution (SubBytes) which is implemented by 16 S-boxes. Let the 8-bit input and output of each S-box bes<sub>r,c</sub>€GF(2<sup>8</sup>) and s'<sub>r,c</sub> € GF(2<sup>8</sup>) respectively. The S-box consists of a multiplicative inversion, i.e., s<sup>-1</sup>r, c € GF(2<sup>8</sup>), followed by an affine transformation consisting of the matrix  $\Gamma$  and the vector  $\gamma$  to generate the output as:

$$\begin{array}{c} s'_{r,c} = \Gamma s^{-l}_{r,c} + \gamma = \\ \hline \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \end{array} \right] s^{-l}_{r,c} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ \end{pmatrix}$$
(2)

The 8-bit outputs of 16 S-boxes are used to obtain the output state of the SubBytes transformation as:

$$s' = [s'_{r,c}]_{r,c=0}$$
(3)

**Shift rows:** In the second transformation, it cyclically shifts the 4 bytes of the rows of the input state to the left and the first row is left unchanged to obtain the output state, i.e., SR(S'), as:

$$SR(S') = \begin{cases} s'0,0 \quad s'0,1 \quad s'0,2 \quad s'0,3 \\ s'1,1 \quad s'1,2 \quad s'1,3 \quad s'1,0 \\ s'2,2 \quad s'2,3 \quad s'2,0 \quad s'2,1 \\ s'3,3 \quad s'3,0 \quad s'3,1 \quad s'3,2 \\ \end{cases}$$

$$= [S'_{(r,c)} \mod d_{J}^{3}_{r,c} = 0 \qquad (4)$$

**Mix columns:** In the third transformation, multiplying a constant matrix with the output state of ShiftRows, SR(S') in (4), to obtain the output state of MixColumns, i.e., the matrix S", as:

$$S'' = [s''_{r,c}]^{3}_{r,c=0} = \begin{pmatrix} \{2\}h & \{3\}h & \{1\}h & \{1\}h \\ \{1\}h & \{2\}h & \{3\}h & \{1\}h \\ \{1\}h & \{1\}h & \{2\}h & \{3\}h \\ \{3\}h & \{1\}h & \{2\}h & \{3\}h \\ \{3\}h & \{1\}h & \{1\}h & \{2\}h \end{pmatrix}$$
(5)

$$SR(S') = \begin{pmatrix} s'0,0 & s'0,1 & s'0,2 & s'0,3 \\ s'1,3 & s'1,0 & s'1,1 & s'1,2 \\ s'2,2 & s'2,3 & s'2,0 & s'2,1 \\ s'3,3 & s'3,2 & s'3,3 & s'3,0 \end{pmatrix}$$
(6)

AddRoundKey: The final transformation is *AddRoundKey* in which the input state is added (modulo-2) with the key of the round. Considering the round key input state as the matrix K = [kr, c]3r, c = 0, with entries kr; c,  $0 \le r$ ,  $c \le 3$ , the output state of the AddRoundKey transformation, i.e., O, is obtained as:

$$O = [o_{r,c}]^{3}_{r;c} = 0 = S'' + K$$
(7)

$$O = [o_{r,l}]_{r,c}^{\beta} = \begin{pmatrix} \{0e\}h & \{0b\}h & \{0d\}h & \{09\}h \\ \{09\}h & \{0e\}h & \{0b\}h & \{0d\}h & \{0d\}h \\ \{0d\}h & \{00\}h & \{0e\}h & \{0b\}h \\ \{0b\}h & \{0d\}h & \{00\}h & \{0e\}h \end{pmatrix}$$
(8)

#### FAULT DETECTION SCHEME

The systematic fault detection scheme for the multiplicative inversion of s-box and inverse s-box: This scheme explains the 8-bit input of the multiplicative inversion is multiplied by the 8-bit output. Also the n-bit result  $(1 \le n \le 8)$  of the multiplication is compared with the actually obtained n-bit result, i.e.,  $1 \in GF(2^8)$ . If  $s \ne 0$  and  $0 \in GF(2^8)$ . If s = 0 because the multiplicative inversion is also used in the inverse S-box, the same scheme can be used for the inverse S-box.



Fig. 1: The scheme based on multiplication for the fault detection of the multiplicative inversion

We present a systematic method for the fault detection scheme for the multiplicative inversion by deriving the matrix-based formulations for the multiplicative inversion in the S-box and inverses-box. We use the following theorem from Mentens *et al.* (2005) to obtain the multiplication of field elements  $A = \sum_{i=0}^{m-1} a_i a_i$  and  $B = \sum_{i=0}^{m-1} b_i a_i$  in the finite field  $GF(2^m)$  constructed by the irreducible polynomial of P(x) with the primitive root of  $a_i$ .

Let  $s = s_7 \alpha^7 + s_6 \alpha^6 + s_5 \alpha^5 + s_4 \alpha^4 + s_3 \alpha^3 + s_2 \alpha^2 + s_1 \alpha + s0$  and  $s^{-1} = s_7^{-1} \alpha^7 + s_6^{-1} \alpha^6 + s_5^{-1} \alpha^5 + s_4^{-1} \alpha^4 + s_3^{-1} \alpha^3 + s_2^{-1} \alpha^2 + s_1^{-1} \alpha^1 + s_0^{-1}$  be the 8-bit input and output of the multiplicative inversion in the binary field GF(2<sup>8</sup>), respectively. Considering the fact that the result of the multiplication of the 8-bit input s,  $s \neq 0$  and the output s<sup>-1</sup> of the multiplicative inversion is the unity polynomial  $1 \in GF(2^8)$ , the following is derived from Theorem 1 for the relation between s and s<sup>-1</sup>.

**Corollary 1:** Let the vectors corresponding to the input and output of the multiplicative inversion be  $s = [s_0, s_1]$ ,  $s_2, s_3, s_4, s_5, s_6, s_7]^T$  and  $s^{-1} = [s_7^{-1}, s_6^{-1}, s_5^{-1}, s_4^{-1}, s_3^{-1}, s_2^{-1}]$ ,  $s_1^{-1}, s_0^{-1}]^T$ . Then, the matrix formulation of the S-box multiplicative inversion (respectively, the inverse Sbox) is as follows in Fig. 1.

**Theorem 1 (Mentens** *et al.*, **2005):** Let  $C = \sum_{i=0}^{m-1} c_i a_{ibe}$  the multiplication of A and  $B \in GF(2^m)$ . Then, the coordinates of C can be obtained from:

$$[c_0, c_1, c_2....c_{m-1}] = (L + Q^T U)b$$
(9)

where,  $b = [b_0, b_1, b_2, \dots, b_{m-1}]^T$ 

L = *a*0 0 0 a1 <u>a</u>0 0 0 0 a0 0 **a**1 0 a1 a0 . . a2 a1 a0 am - 2 am - 3. . am - 2<u>a</u>0 am-1(10)



And the (m-1xm) binary matrix Q is obtained as follows:

$$[\alpha_{m} \alpha_{m+1} \dots \alpha_{12m-2}]T = Q [1, \alpha, \alpha_{2} \dots \alpha_{m-1}]^{T} mod(p(x))$$
 (12)

$$ZS^{-1} = u \tag{13}$$

u = [u' 0 0 0 0 0 0] where u' is obtained by logical OR operations of all inputs and outputs, u' =  $(s_0 \lor s_1 \lor s_2 \lor s_3 \lor s_4 \lor s_5 \lor s_6 \lor s_7) \lor (s_7^{-1} \lor s_6^{-1} \lor s_5^{-1} \lor s_4^{-1} \lor s_3^{-1}$  $\lor s_2^{-1} \lor s_1^{-1} \lor s_0^{-1})$  Moreover, the modulo-2 additions (XOR operations) of the coordinates of s are shown with commas in indices, e.g.,  $s_{7,0} = s_7 + s_0$ :

7	_	
1.	=	
_		

$\sim$							~	
s0	<i>s</i> 7	<i>s</i> 6	<i>s</i> 5	<i>s</i> 4	s7,3	s7,6,2	s6,5,1	
<i>s</i> 1	s7,0	s7,6	s6,5	s5,4	s7,4,3	s6,3,2	s7,5,2,1	
<u>s</u> 2	<i>s</i> 1	s7,0	s7,6	s6,5	s5,4	s7,4,2	s6,3,2	
<i>s</i> 3	s7,2	s6,1	s7,5,0	s7,6,4	s7,6,5,3	s7,6,5,4,2	s7,6,5,4,3,1	
<i>s</i> 4	s7,3	s7,6,2	s6,5,1	s7,5,4,0	s6,4,3	s5,3,2	s7,4,2,1	
<i>s</i> 5	<i>s</i> 4	s7,3	s7,6,2	s6,5,1	s7,5,4,0	s6,4,2	s5,3,2	
<i>s</i> 6	<i>s</i> 5	<i>s</i> 4	s7,3	s7,6,2	s6,5,1	s7,5,4,0	s6,4,3	
<u>s</u> 7	<i>s</i> 6	<i>s</i> 5	<i>s</i> 4	s7,3	s7,6,2	s6,6,1	s7,5,4,0 /	
								(14)

**Proof:** We prove (13) for two cases of s = 0 and  $s \neq 0$  separately. Let the input ( $s \neq 0$ ) be a nonzero field element in  $GF(2^8)$  generated by  $P(x) = x^8 + x^4 + x^3 + x + 1$ . Then, the multiplicative inversion should generate s-1. Using (12) in Theorem 1 and considering the irreducible polynomial of P(x), the (7×8) matrix Q can be obtained as:

$$Q = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$
(15)

This matrix is obtained by using the representations of  $\alpha^8 \alpha^9$ .....  $\alpha^{14}$  with respect to the polynomial basis for different rows of Q. Considering A= s  $\neq$  0 and B = s<sup>-1</sup> in Theorem 1, the matrices L and U in (10) and (11) are functions of the 8-bit input vector s as:

Substituting Q, L and U (15)-(17) from (9) and denoting  $Z = L + Q^{T}U$ . Since  $s \neq (0 \ 0, \ 0, \ 0... \ 0) \in GF(2^{8})$ , u = 1 and the result of multiplication is:

C = A.B mod P(x) = 1  $\in$  GF(2<sup>8</sup>) i.e., c = [c<sub>0</sub>, c<sub>1</sub>, c<sub>7</sub>]<sup>T</sup> = [1 0... 0]<sup>T</sup>

Therefore using (9) one can prove that (13) is valid for  $s \neq 0$ . moreover, for s = 0, the output of the multiplicative inversion generates  $0 = (0 \ 0.... \ 0)$ . Thus, all entries of the matrix Z and hence, all eight entries of the left-hand side vector of (13) are equal to zero. In such a case, the vector  $u = [0 \ 0... \ 0]^T$  since the result of the OR operation among all *sis* and *s<sup>-1</sup> i s* are zero, i.e., u = 0. Therefore, the proof is complete.

One can figure out that implementation (13) needs 64 ANDs, 15 ORs and 143 XOR gates. Also it is noted that XOR gates can be reduced to 84, if sub expression sharing is used. If one implements the S-box using the composite field presented in Breveglieri *et al.* (2007), it requires 36 and gates and 123 XOR gates for the original S-box implementation. Then, adding this fault detection scheme would require approximately 91% area overhead. Also the silicon area of an AND is 0.6 that of an XOR gate and is derived assuming that an XOR gate is implemented by 10 transistors.

The proposed fault detection scheme for the S-Box and the inverse S-Box: If the SubBytes implementation in the AES is using LUTs, there will be no means of entry to the output of the multiplicative inversion. Thus, the aforementioned scheme cannot be used. We propose a new scheme which is independent of the way the implementation of S-box and the inverse S-box. First, we obtain the matrix-based S-box formulations as follows:

**Theorm 2:** Let  $s = s_7 \alpha^7 + s_6 \alpha^6 + s_5 \alpha^5 + s_4 \alpha^4 + s_3 \alpha^3 + s_2 \alpha^2 + s_1 \alpha + s_0 \alpha n d s' = s'_7 \alpha^7 + s'_6 \alpha^6 + s'_5 \alpha^5 + s'_4 \alpha^4 + s'_3 \alpha^3 + s'_2 \alpha^2 + s'_5 \alpha^5 + s'_5 \alpha^6 + s'_5 \alpha^5 + s'_5$ 

 $s'_{l}\alpha^{l} + s'_{0}$  be the 8-bit input and output of the S-box. Thus the relation between the input and output of the S-box can be obtained as:

$$Ms' + m = u' \tag{18}$$

Moreover, the  $(8 \times 8)$  matrix M is denoted as:

	M =							
1	s6,5,6	s5,4,1	s7,5,3,0	s6,4,2	s7,5,3,1	s7,6,5,2,0	s7,6,5,4,1	s7,6,3,0
	s7,5,3,2,0	s6,4,2,1	s7,6,5,4,3,1	s6,6,5,4,3,1 s	7,6,5,4,3,2,1	s5,3,2,1	s4,2,1,0	s6,4,3,1
	s6,4,3,1	s7,5,3,2,0	s7,6,5,4,2	s7,6,5,4,3,1 s	7,6,5,4,3,2,0	s6,4,3,2	s5,3,2,1,	s7,5,4,2,0
	s7,6,4,0	s6,5,3	s6,0	s7,5	s6,4	s6,4,3,2,0	s7,5,3,2,1	s7,5,1
	s7,6,2,1	s7,6,5,1,0	s5,3,1	s4,2,0	s3,1	s6,4,3,2,1	s7,5,3,2,1,0	s7,2,3
	s7,3,2	s7,6,2,1	s6,4,2,0	s5,3,1	s4,2,0	s7,5,4,3,2	s6,4,3,2,1	s4,3,0
	s4,3,0	s7,3,2	s7,5,3,1	s6,4,2,0	s5,3,1	s6,5,4,3,0	s7,5,4,3,2	s5,4,1
	s5,4,1	s4,3,0	s6,4,2	s7,5,3,1	s6,4,2,0	s7,6,5,4,1	s6,5,4,3,0	s6,5,2
,								
								(19)
W	here,	u' =	= [ <i>u'</i> ,0	,0,0,0,0	,0,0] <i>T</i>	u' =	$(s_0 \lor s)$	$_{l} \vee s_{2} \vee s_{3}$
V	$s_4 \lor s_5$	$\lor s_6 \lor s$	7) $\vee (s'_{7})$	$\vee \overline{s'_6} \vee \overline{s'_5}$	$\forall s'_4 \forall s$	$'_{3} \vee s'_{2} \vee$	$\overline{s'_{l}} \lor s'_{0}$	j and <i>m</i>

 $= [s_{6;0}, s_{7;6;1}, s_{7;2;0}, s_{6;3;1}, s_{7;6;4;2}, s_{7;5;3}, s_{6;4}, s_{7;5}]^T.$ 

**Proof:** We prove (18) for two cases of  $s \neq 0$  and s = 0 separately. Let 8-bit input s be a nonzero field element in GF(2<sup>8</sup>). Considering (2), one can obtain:

$$s^{-l} = \Gamma^{-l} s' + \Gamma^{-l} \gamma = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} s^{l}_{r,c} + \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$
(20)

By substituting s-1 from (20) into (13), one reaches  $Z\Gamma^{-1} s'+Z\Gamma^{-1}\gamma$ . Now, let us denote  $Z\Gamma^{-1} = M$  and  $Z\Gamma^{-1} \gamma = m$ . Then, the left-hand side of (18) is obtained. Since s  $\neq 0 = (0, 0... 0) \in GF(2^8)$  u' = 1. i.e., the result of multiplication C = AB mod P(x) = 1  $\in GF(2^8)$ . This implies that the left-hand side of (18) be  $Zs^{-1} = [1 \ 0... 0]$  T = u'. Furthermore, because we have  $Zs^{-1} = Ms'+m$  one can prove that (18) is valid for  $s\neq 0$ . Moreover, according to (2), for the input  $s = 0 = (0, 0... 0) \in GF(2^8)$ .

We have the output as  $\mathbf{s}' = [\mathbf{s}'_0 \ \mathbf{s}'_1 \dots \ \mathbf{s}'_7]^T = [1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0]^T$  which corresponds to the field element  $\mathbf{s}' = \{63\}$ h = (0 1 1 0 0 0 1 1)  $\in$  GF(2<sup>8</sup>). From the Theorm 2, u' =  $[0 \ 0 \dots \ 0]^T$  since we have u' =  $(\mathbf{s}_0 \lor \mathbf{s}_1 \lor \mathbf{s}_2 \lor \mathbf{s}_3 \lor \mathbf{s}_4 \lor \mathbf{s}_5 \lor \mathbf{s}_6 \lor \mathbf{s}_7)$  $\lor (\mathbf{s}'_7 \lor \mathbf{s}'_6 \lor \mathbf{s}'_5 \lor \mathbf{s}'_4 \lor \mathbf{s}'_3 \lor \mathbf{s}'_2 \lor \mathbf{s}'_1 \lor \mathbf{s}'_0)$ . Then the vector  $[0 \ 0 \dots \ 0]^T$  = u'. Therefore, the proof is complete (Fig. 2).

Let us consider (18) for the input  $s = 0 = [0 \ 0... \ 0] \ 2 \in GF(2^8)$ . For this input, the correct output is  $s' = \{63\}_h = (01 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1) \in GF(2^8)$ . If the erroneous

Res. J. Appl. Sci. Eng. Technol., 12(1): 19-26, 2016



Fig. 2: The proposed fault detection scheme of the S-box

output is not s' =  $\{63\}h = (0\ 1\ 1\ 0\ 0\ 1\ 1) \in GF(2^8)$  in the right hand side of (18), we have u' = 1, whereas the left-hand side is zero and therefore, the wrong output is detected.

Although checking the formulation of (18) detects all errors in the output of the S-box, its implementation is very costly (Proposition 1). To reduce the overhead of the fault detection scheme (Fig. 2), we have obtained the single-bit parity for the formulation of (18). In Fig. 2, this is obtained in order to compare only 1 bit for an 8-bit data to detect any combination of odd number of erroneous bits at the result of the left-hand side of (18). Thus, one can check the parity of two sides of (18) to obtain 1-bit equation for checking the S-box as follows:

**Theorem 2:** Let  $s = s_7 \alpha + s_6 \alpha^6 + s_5 \alpha^5 + s_4 \alpha^4 + s_3 \alpha^3 + s_2 \alpha^2 + s_1 \alpha + s_0$  and  $s' = s'_7 \alpha^7 + s'_6 \alpha^6 + s'_5 \alpha^5 + s'_4 \alpha^4 + s'_3 \alpha^3 + s'_2 \alpha^2 + s'_1 \alpha^1 + s'_0$  be the 8-bit input and output of the S-box. The equation holds for all the possible patterns of s and s' is as follows:

$$P(M s'+m) = s_0(s'_b+s'_c)+s_1s'b+s_2s'd+s_3s'4+s_4(s'c+s'_3)+s_5s'_a+s_6(s'_d+s'_6)+s_7(s'_5+s'_4) = u'$$

where

$$s'_{a} = s'_{0} + s'_{2} + s'_{3} + s'_{5.}, s'_{b} = s'_{a} + s'_{7}, s'_{c} = s' + s'_{4} + s'_{6}$$
  
and  $s'_{d} = s'_{2} + s'_{7}$  (21)

**Proof:** The parity of two sides of (18) as obtained and we have:

$$P_{(M\,s'+m)} = Pu' = u' \tag{22}$$

where, M, m and u' are presented in Theorem 2. Considering the fact that parity is a linear operation, P(M s'+m) = PM s'+P m. Then, using M and m defined in Theorem 2 one can obtain:

$$P_{Ms'} = s_a s'_0 + s_b s'_1 + s_c s'_2 + s'_3 (s_a + s_4) + s'_4 (s_b + s_3 + s_7) + s'_5 (s_a + s_7) + s'_6 (s_b + s_6) + s'_7 (s_5 + s_c)$$

And  $P_m = s_6+s_7$ , where  $s_a = s_0+s_1+s_5$ ,  $s_b = s_0+s_4$ ,  $s_c = s_a+s_2+s_6$ , after rearranging, the proof is complete.

**Corollary 2:** For the fault detection of the inverse Sbox, one can use by changing the place of the input and output, i.e., swapping the coordinates of s with s'.

#### SIMULATION RESULTS

Here we have considered both the single and multiple stuck-at errors for the proposed scheme. And these models covers both natural faults and fault attacks. In the AES encryption or decryption rounds, if exactly 1 bit error appears at the output, this proposed scheme detects it, the error coverage is about 100%. Because in this case, one of the 8-bit four error indication flags in alarms the error. However, multiple stuck-at errors are also considered. Because multiple bits will actually be flipped due to the reason an attacker cannot be able to flip exactly 1 bit in a single stuck at error to gain more information by some technical constraints.

The AES algorithm and low complexity fault detection scheme for composite s-box was described in VHDL and we used the *Modelsim* 6.3 g\_b1 tool to simulate the code. We analyzed the area and internal and external fault in AES. The fault detection schemes of sub byte in existing and proposed compare the performance in Table 1. Figure 3 to 6 shows the simulation results.

The implementation of s-box requires large number of gates in traditional (LUT) Look up table. Also the unbreakable delay is longer than that of the total delay. Also it is not suitable for resource constrained use

Res. J. App	l. Sci. Eng	. Technol	l., 12(1):	19-26, 2016
-------------	-------------	-----------	------------	-------------

Table 1: Comparision of s-box

Design	Area	Delay	Power
LUT-Based	262144	31.824 ns	35 mw
Composite field based	28514	8.129 ns	34 mw

Anter yosen, dect Vesk, ask, ask, ask, ask     anter yosen, dect Vesk, ask, ask, ask, ask     anter yosen, dect Vesk, ask     a	D- Jen D- Jen	norgottern "dweit/dack vervenken "diestlijkken "dest vervenken "diestiken» "im	1 COLUEID+HICKETREMANECCIOED+ COLUEID+HICKETREMANECCIOED+	717171717171 2011221044598877 2001020304059807	STANDS CONTRACTOR			
American	0-5 /m 5 /m 5 /m	novpran, decklopher_text anyotan, decklopher_text anyotan, decklophik_mex_add	DCRIF <del>FFFF</del> ICIALIZME (DEBENGRIE) D 1	DRC CODEA 7040	DISCUELTED VOE 40250	DC32 4471	13C9A12496 20100	9502037
Operation (active (int.))         0001000000000000000000000000000000000	🔶 /ets	scryption_dect/feult	4	1				
D00         D01         D00         D01           D01         D02         D01	D	scruption declary sur	0001001004060607060904380000025FD644	20011020304050507	DB090A0E0C0D0E0P0	SAA74C02AF72F	DAAS70F 106A576	E56920*d85
Description         Solid         Dot           Description         Solid         Solid         Solid           Description         Solid         Solid         Solid         Solid           Description         Solid         Solid         Solid         Solid         Solid           Description         Solid         Solid         Solid         Solid         Solid         Solid           Description         Solid         Solid <td>D-1 (m)</td> <td>scryption_dect/sub_eleft_f</td> <td>DOG</td> <td>000</td> <td></td> <td>200</td> <td></td> <td></td>	D-1 (m)	scryption_dect/sub_eleft_f	DOG	000		200		
Description         Description <thdescription< th=""> <thdescription< th=""></thdescription<></thdescription<>	D. 100	Trutton dectree add f	001	000		6011		
Bits of specing         Period	D	structure should be	DOD 10/01/2-02000/700000AUDOC DOD PC/6AA	000020104050602	SOMULAR CONTRACT	544740028F7.84	CALL RETURNER	PERFORMENT FORM
Anticipation pectral provide a provide a provide and a provide a provid		the watch and an	FRIDAR BADSDER/7000904080C0D0E/Fd	00 20 20 20 20 40 50 50 20	BORGADEDCOCOCOPOED	P0 x020 30 40	5050 2350 2340 200	COFOFO
Bit How yours _ dectrit	D	trotter methic school	DCC/45754095 \$2051C060F0F /bA70F185	5 K 48 204795 1 05	CHERTE MARKETS	BCLAR ZING	S 105 X DALE	48.70E150
Operating (active) (activ	D	stration order and so	RESERVICE ORCOTERSIBACACOET	\$15 TO \$ 100	CO 705 75 IEAC ADDE 7	facts energy	AOP 104CD 20075 10	AT ADVET
C         Percention         C         Percention         <	D- 100	ticoter Decis Tencoul	WARDER FOR THE OWNER AND THE OWNER	1273641663E424	A REPORT OF A	Fail of Data and a	- 40 V A AF & MI	0000018
Displanet part / 2 public         2004 102517000013001001200000         20070000         20070000000000000000000000000000000000	n	Andrew decisi attaut	NUMBER OF STREET STREET STREET	STATISTICS ACTIN	CTUDAT ACTUDED	General Care Care Care Care Care Care Care Care	SACESS TO DAYD	812884
Display         Ant: LASS/24/01/2016/01/20100/01/2016/000/0000000000	-	setutation marily 2 material	256A 35255 WEIRIASTINASTINAST PUTATION	3 76 11 2 14 19 70 19 19	1940146116097969	D99416250	NE DE MODERN LA ELL	6(97969
Description_meth2_monait         PEDDIADD/PEDDIADD/2003.0003.07.0005.07.072         PEDDIADD/2003.07.0005.07.072         PEDDIADD/2003.07.0005.072         PEDDIADD/2003.07.072         PEDDIADD/2003.07.072         PEDDIADD/2003.072         PEDDIA		tunder Detail shifted	CONTRACTOR OF THE PARTY OF THE	A THE LANDSTATISTICS	CALCULATE AND THE A	Cooper La ADDO	MOTORIACOLEON	POARA 1
	24	unumber meth 7 minut	PEDDINGTINGAN TANKS BETTING STORET	Constant special	Lat. 11 (C + 275) (V/V)	E 213 + 21 / 21	1107200 87 781	07861
Contraction Contraction Contraction Contraction Contraction	12/10/20	anisting design and an	ALTER WATE COLOR THE STOLEN AND	1117200101000000	CACAGARIA MODAL REAL	(a)/5/20/0	000000000000000000000000000000000000000	WINT C
and a second	and the second	and an and a south	A CONTRACT OF A	AN 122340, 22C 20104	CHEMINE ADDREED	Descover.	CROST A030537.007	6.7837
2403 st 2403 st 2403 st 2403 st 2000 s	-	Non	2400 pe	Street street.	CTTT LETTE	111111111	Contract Contract Contract	Mine 1

Fig. 3: Output of encryption for composite field s-box without error

🗉 🔶	/encryption/plain_text	00112233445566778899AABBCCDDEEFF	00112233445566778899AABBCCDDEEFF
<b>B-</b>	/encryption/key_in	000102030405060708090A0B0C0D0E0F	000102030405060708090A0B0C0D0E0F
	/encryption/clock	1	
<b>B-</b>	/encryption/cipher_text	69C4E0D86A780430D8CD878070B4C55A	69C4E0D86A7B0430D8CDB7807DB4C55A
<b>B-</b>	/encryption/key_out	000102030405060708090A0B0C0D0E0FD6AA	000102030405060708090A0B0C0D0E0FD6AA74FDD2AF72FAD
<b>—</b>	/encryption/key	000102030405060708090A0B0C0D0E0FD6AA	000102030405060708090A0B0C0D0E0FD6AA74FDD2AF72FAD
<b>B-</b>	/encryption/pre_out	00102030405060708090A0B0C0D0E0F0	00102030405060708090A0B0C0D0E0F0
<b>B-</b>	/encryption/r1_subout	63CAB7040953D051CD60E0E7BA70E18C	63CAB7040953D051CD60E0E7BA70E18C
<b>B-</b>	/encryption/r1_shiftout	6353E08C0960E104CD70B751BACAD0E7	6353E08C0960E104CD70B751BACAD0E7
<b>B-</b>	/encryption/r1_mixout	5F72641557F5BC92F7BE3B291DB9F91A	5F72641557F5BC92F7BE3B291DB9F91A
<b>B-</b> *	/encryption/r1_addout	89D810E8855ACE682D1843D8CB128FE4	89D810E8855ACE682D1843D8CB128FE4
<b>B-</b>	/encryption/r2_subout	A761CA9B97BE8B45D8AD1A611FC97369	A761CA9B97BE8B45D8AD1A611FC97369
<b>B-</b>	/encryption/r2_shiftout	A78E1A6997AD739BD8C9CA451F618B61	A7BE1A6997AD739BD8C9CA451F618B61
<b>B-</b>	/encryption/r2_mixout	FF87968431D86A51645151FA773AD009	FF87968431D86A51645151FA773AD009
<b>B-</b>	/encryption/r2_addout	4915598F55E5D7A0DACA94FA1F0A63F7	4915598F55E5D7A0DACA94FA1F0A63F7
<b>B-</b>	/encryption/r3_subout	3B59CB73FCD90EE05774222DC067FB68	3B59CB73FCD90EE05774222DC067FB68
<b>B-</b>	/encryption/r3_shiftout	38D92268FC74FB735767CBE0C0590E2D	38D92268FC74FB735767CBE0C0590E2D
<b>B-</b>	/encryption/r3_mixout	4C9C1E66F771F0762C3F868E534DF256	4C9C1E66F771F0762C3F868E534DF256
<b>B-</b> *	/encryption/r3_addout	FA636A2825B339C940668A3157244D17	FA636A2825B339C940668A3157244D17
<b>B-</b>	/encryption/r4_subout	2DFB02343F6D12DD09337EC75B36E3F0	2DFB02343F6D12DD09337EC75B36E3F0
	Now	1400 pc	
	Now	1400 ps	ps 200 ps 400 ps 600 ps 8

Fig. 4: Output of encryption for composite field s-box with error

/decryption/cipher_text	69C4E0D86A7B0430D8CDB78070B4C55A	69C4E0D86A7B0430D8CDB78070B4C55A
🔶 /decryption/clk	1	
E-//decryption/key_in	000102030405060708090A080C0D0E0F	000102030405060708090A0B0C0D0E0F
m	00112233445566778899AABBCCDDEEFF	00112233445566778899AABBCCDDEEFF
E- /decryption/key	000102030405060708090A080C0D0E0FD6AA3	000102030405060708090A0B0C0D0E0FD6AA74FDD2
/decryption/pre_out	7AD 5FDA789EF 4E272BCA 100B3D9FF 59F	7AD5FDA789EF4E272BCA100B3D9FF59F
m	7A9F102789D5F50B2BEFFD9F3DCA4EA7	7A9F102789D5F50B2BEFFD9F30CA4EA7
decryption/r 1_subout	BD6E7C3DF2B5779E0B61216E8B10B689	BD6E7C3DF2B5779E0B61216E8B10B689
m	E9F74EEC023020F61BF2CCF2353C21C7	E9F74EEC023020F61BF2CCF23\$3C21C7
m	54D990A16BA09AB596BBF40EA111702F	54D990A16BA09AB596BBF40EA111702F
m	5411F4B56BD9700E96A0902FA1BB9AA1	5411F4B56BD9700E96A0902FA1BB9AA1
m	FDE3BAD 205E5D0D73547964EF1FE37F1	FDE38AD205E500D73547964EF1FE37F1
m	BAA03DE7A1F9B56ED5512CBA5F414D23	BAA03DE7A 1F9B56ED 5512CBA 5F414D23
m	3E1C22C0B6FCBF768DA85067F6170495	3E1C22C0B6FC8F768DA85067F6170495
decryption/r3_shiftout	3E175076B61C04678DFC2295F6A8BFC0	3E175076B61C04678DFC2295F6A8BFC0
decryption/r3_subout	D1876C0F79C4300AB45594ADD66FF41F	D 1876C0F79C4300AB45594ADD66FF41F
m	C57E1C159A9BD286F05F4BE098C63439	C57E1C159A9BD286F05F4BE098C63439
Image: maintent of the second seco	B458124C68B68A014B99F82E5F15554C	B458124C68B68A014B99F82E5F15554C
#/decryption/r4_shiftout	B415F8016858552E48B6124C5F998A4C	B415F8016858552E48B6124C5F998A4C
/decryption/r4_subout	C62FE109F75EEDC3CC79395D84F9CF5D	C62FE109F75EEDC3CC79395D84F9CF5D

Fig. 5: Output of decryption for composite field s-box without error

Res. J. Appl. Sci. Eng. Technol., 12(1): 19-26, 2016

Alexy prior, decl/duck	1 DRC 4722 EEA 7004302 RCL 0 TRU 702-6E 154.		43008(1878)7	094CSSA			
D-* Alectronic section sections for D-* Alectronic dections and	E25725C2E0F33FC343E300F8C152762 D001003040506070809040B1C0DE2F04443		67737793A4080	CODEDTF	SUSSICIONE (FI	ACTIF LDSAE AV	C 192782 1269207086
/slearyption_dect/fault	1	the strongers	and the second			Carl Contractor Carl Section	
Men within , dect/fluid, _BHTsu /decrysteric dect/fluid, _eccl_ro	6 A						
C- veel vpoor_cectment_sto_t	1 100	2m			1000		
C	an (	200			2001		
E- ideo votion_dectAey	100102030405060708090A096C0D069F06AA3	0003121204150	003352904060	CIDOEDFDSAA7	400018F72FADA	A671F 100A576F	25652070380
D-1 Main yolion_sectione_suit	DACOPULATINES - ECTLER: A LODGORD BY SIR	0.409-04/898F	#27.88(A108)	D/#FS#	DAD/SFDA789EF	#27/AC410083	D9FF39F
B. Alebyption_dect#1_shiftput	149F 10229900F 50805EFFD 9F30CA4EA7	7ASF 1027690 St	Satury Corp.	DCA4EA7	2049 20279905	106280710/21	CARA7
the start price and a second	A 302 YO REP 285779203613 300081309039	BOREN DE 265	7%E0861216E8	8106689	AMEX OF 35	7790386121663	108683
E- Steproption dectr ( addaut	F7F74E6033030F4 #F3C6F2383C2107	274PC0/50	OF 6 LBP 20CF 25	5302307	P7774EE00230	016 25 200723	3C21E7
Charles the state of the state	EDUTOS TIMEADIAE SSIEDER 402 A LI 170 PT	C43660456880	ALCOND. ST.	111202	EUGORIZ BEAD	ARCINETER-MEA	11212
The Alexandre destal shifts a	FOULF-ROMETZ/DOFPLACING FAILUPIA TI	54117-015001001	TOP SEATON PA	INFIGALI	POLIP-PORTAZ	TOPSPACES PA	INFIGA 71
P. d statistics chain 2 admit	AND TRAILINGS THAT TAKES AND TRACK	The second state	11/2/24/24/24	100 276 1	WE SHALLYING	10011164781461	100 2000
The Manual And A should be and	TANK PROPERTY AND A DESCRIPTION OF A DES	THE REPORT AND	A 100 100 100 100 100	1.1.2011	Service and the service of the	CONTRACTOR CONTRACTOR	A 5075
D-* identigation_siently2_proput	35MP12P921565629176751751579657	EXCLOSED	F NEOLESIE A	6170-86	55F0F9C19	GENF17E75075	\$579857
	v 1200 ps	11 1	17 51	Trat. B	(Erm 30	178	des 1700

Fig. 6: Output of decryption for composite field s-box with error

because it costs a large area. Thus the composite field arithmetic is used to solve these problems. The fault detection scheme implemented by (LUT) look up table in VHDL code synthesised using Xilinix 9.1ISE and get the report of gate count. The gate count value of LUT based fault detection scheme is 262144 logic gates. This fault detection scheme requires 4 times greater than the proposed one. We have to compare it with proposed once and our aim is to reduce the gate count to maximum possible extent.

Composite field implementation of s-box needs less number of gates. We can describe in VHDL and perform synthesis using Xilinix 9.1 In our synthesis report we got a comparatively small value of value of 28514 numbers of gates. Our aim is to reduce the gate count to maximum possible extent. We got gate count almost one tenth of look up table implementation of s box.

# Encryption output for composite field s-box without error:

Plain text: x"00112233445566778899aabbccd deeff" Key: x"000102030405060708090a0b0c0d0e0f" Cipher text: x"69c4e0d86a7b0430d8cdb78070b 4c55"

Figure 3 shows the output for composite field sbox without error any for a particular input. Theta, eta, gamma, sigma values are obtained and the fault output value is zero.

**Encryption output for composite field s-box without error:** Figure 4 shows the Output for encryption for composite field s-box without error for a particular 128bits input and128-bit input key. Fault detection scheme implemented by composite field s-box and detect a internal fault. We will consider the initial round pre out output value and replace output of pre\_out with another 128-bit value and stimulate with modelsim we will get faulty output Theta, eta, gamma, sigma values are obtained and the fault output value is obtained. In this waveform fault occur in add roundkey transformation in initial round.

# Decryption output for composite field s-box without error (Fig. 5):

**Decryption output for composite field s-box with error:** Figure 6 shows the Output of decryption for composite field s-box with error any for a particular 128-bits cipher input and 128-bit input cipher key of each round. Fault detection scheme implemented by composite field s-box and detect a internal fault. We will consider the internal inverse round key output value and replace output of inverse add round key with another 128-bit value and stimulate with modelsim we will get faulty output Theta, eta, gamma, sigma values are obtained and the fault output value is obtained. In this wave form fault occur in add round key transformation of initial round.

### CONCLUSION

We have presented a high performance low complexity parity based fault detection scheme for the AES. These schemes are constructed using the S-box and the inverse S-box using composite fields. We have obtained the least complexity S-boxes and inverse Sboxes including their fault detection circuits. The new fault detection schemes are independent of the structures of the S-boxes and the inverse S-boxes. So that we have used parity based method in s box. Therefore it improves the fault coverage to a greater extent because here the error detection at s box takes two times of it. This simulation results shows that the proposed structure-independent schemes have the highest efficiencies with acceptable error coverage. It shows reasonable area also the time complexity overheads.

### REFERENCES

- Bertoni, G., L. Breveglieri, I. Koren, P. Maistri and V. Piuri, 2002. A parity code based fault detection for an implementation of the advanced encryption standard. Proceeding of the 17th IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT, 2002), pp: 51-59.
- Bertoni, G., L. Breveglieri, I. Koren, P. Maistri and V. Piuri, 2003. Error analysis and detection procedures for a hardware implementation of the advanced encryption standard. IEEE T. Comput., 52(4): 492-505.
- Breveglieri, L., I. Koren and P. Maistri, 2007. An operation-centered approach to fault detection in symmetric cryptography ciphers. IEEE T. Comput., C-56(5): 534-540.
- Canright, D., 2005. A very compact S-box for AES. In: Rao, J.R. and B. Sunar (Eds.), CHES, 2005. LNCS 3659, Springer, Berlin, Heidelberg, pp: 441-455.
- Cohen, A.E., 2007. Architectures for cryptography accelerators. Ph.D. Thesis, University of Minnesota, Twin Cities.
- Karpovsky, M.G., K.J. Kulikowski and A. Taubin, 2004. Differential fault analysis attack resistant architectures for the advanced encryption standard. In: Quisquater, J.J., P. Paradinas, Y. Deswarte and A.A. El Kalam (Eds.), Smart Card Research and Advanced Applications VI (CARDIS, 2004). Kluwer Academic Publishers, Amsterdam, 153: 177-192.
- Karri, R., P. Mishra, K. Wu and K. Yongkook, 2001. Fault-based side-channel cryptanalysis tolerant Rijndael symmetric block cipher architecture. Proceeding of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT, 2001), pp: 418-426.
- Karri, R., P. Mishra, K. Wu and Y. Kim, 2002. Concurrent error detection schemes for fault-based side-channel cryptanalysis of symmetric block ciphers. IEEE T. Comput. Aid. D., 21(12): 1509-1517.
- Kermani, M.M. and A. Reyhani-Masoleh, 2006. Paritybased fault detection architecture of S-box for advanced encryption standard. Proceeding of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT, 2006), pp: 572-580.
- Maistri, P. and R. Leveugle, 2008. Double-data-rate computation as a countermeasure against fault analysis. IEEE T. Comput., 57(11): 1528-1539.

- Mentens, N., L. Batina, B. Preneel and I. Verbauwhede, 2005. A systematic evaluation of compact hardware implementations for the Rijndael S-box. Proceeding of the Cryptographers' Track at the RSA Conference (CT-RSA, 2005), pp: 323-333.
- Moratelli, C., F. Ghellar, E. Cota and M. Lubaszewski, 2008. A fault-tolerant, DFA-resistant AES core. Proceeding of the IEEE International Symposium on Circuits and Systems (ISCAS, 2008), pp: 244-247.
- Mozaffari-Kermani, M. and A. Reyhani-Masoleh, 2008. A lightweight concurrent fault detection scheme for the AES S-boxes using normal basis. Proceeding of the International Workshop Cryptographic Hardware and Embedded Systems (CHES '08), pp: 113-129.
- National Institute of Standards and Technologies (NIST), 2001. Announcing the Advanced Encryption Standard (AES). FIPS Publication 197, National Institute of Standards and Technologies, Washington, DC, pp: 51.
- Rijmen, V., 2000. Efficient implementation of the Rijndael S-box. Department of ESAT, Katholieke Universiteit Leuven, Leuven, Belgium.
- Satoh, A., S. Morioka, K. Takano and S. Munetoh, 2001. A compact Rijndael hardware architecture with S-box optimization. Proceeding of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT, 2001), pp: 239-254.
- Satoh, A., T. Sugawara, N. Homma and T. Aoki, 2008. High-performance concurrent error detection scheme for AES hardware. Proceeding of the CHES, pp: 100-112.
- Wolkerstorfer, J., E. Oswald and M. Lamberger, 2002. An ASIC implementation of the AES S-boxes. In: Preneel, B. (Ed.), CT-RSA, 2002. LNCS 2271, Springer-Verlag, Berlin, Heidelberg pp: 67-78.
- Wu, S.Y. and H.T. Yen, 2006. On the S-box architectures with concurrent error detection for the advanced encryption standard. IEICE T. Fund. Electr., E89-A(10): 2583-2588.
- Yen, C.H. and B.F. Wu, 2006. Simple error detection methods for hardware implementation of advanced encryption standard. IEEE T. Comput., 55(6): 720-731.
- Zhang, X. and K.K. Parhi, 2004. High-speed VLSI architectures for the AES algorithm. IEEE T. VLSI Syst., 12(9): 957-967.
- Zhang, X. and K.K. Parhi, 2006. On the optimum constructions of composite field for the AES algorithm. IEEE T. Circuits-II, 53(10): 1153-1157.