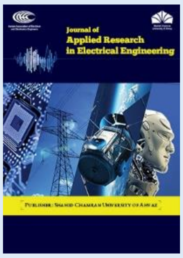
Shahid Chamran University of Ahvaz

Iranian Association of Electrical and Electronics Engineers

Journal of Applied Research in Electrical Engineering

E-ISSN: 2783-2864
P-ISSN: 2717-414X
Homepage: https://jaree.scu.ac.ir/



Research Article

A Fault-Resistant Architecture for AES S-box Architecture

Mahdi Taheri[1], Saeideh Sheikhpour[2], Mohammad Saeed Ansari[3], and Ali Mahani[2,*]

[1] *Department of Computer Systems, Tallinn University of Technology, Tallinn 19086, Estonia*

[2] *Department of Electrical Engineering, Shahid Bahonar University of Kerman, Kerman 7616913439, Iran*

[3] *Eideticom Computational Storage, Calgary, AB, Canada*

* *Corresponding Author: amahani@uk.ac.ir*

Abstract: This paper introduces a high-speed fault-resistant hardware implementation for the S-box of AES cryptographic algorithm, called HFS-box. A deep pipelining for S-box at the gate level is proposed. In addition, a new Dual Modular Redundancy-based (DMR-based) countermeasure is exploited in HFS-box for fault correction. The newly introduced countermeasure is a fault correction scheme based on the DMR technique (FC-DMR) combined with a version of the time redundancy technique. In the proposed architecture, when a transient random or malicious fault(s) is detected in each pipeline stage, the error signal corresponding to that stage increases. The control unit holds the previous correct value in the output of the proposed DMR voter in the other pipeline stages as soon as it observes the value '1' on the error signal. The previous correct outputs will be kept until the fault effect disappears. The presented low-cost HFS-box provides a high capability of fault resistance against transient faults with any duration by imposing low area overhead compared with similar fault correction strategies, i.e., 137%, and low throughput degradation, i.e., 11.3%, on the original S-box implementation.

Keywords: Fault-resistant, advanced encryption standard (AES), S-box, high-speed.

Article history
Received 30 December 2020; Revised 28 April 2021; Accepted 11 June 2021; Published online 30 June 2021.

How to cite this article
M. Taheri, S. Sheikhpour, M. S. Ansari, and A. Mahani, "A fault-resistant architecture for AES S-box architecture," *J. Appl. Res. Electr. Eng.*, vol. 1, no. 1, pp. 86-92, 2022. DOI: 10.22055/jaree.2021.36230.1020

Check for updates

1. Introduction

Dependable applications, like secure information systems, remote security services, online banking, etc., play an important role in our daily lives. Secure storage and communication are critical requirements of these applications. Nowadays, cryptography is extensively used in dependable applications to meet these critical requirements, thereby preventing unauthorized access to secure information. Another important requirement of a dependable application is reliability. Therefore, in many cases, a fault resilient approach is incorporated with original hardware implementation [1].

The Advanced Encryption Standard (AES) [2] was standardized by the National Institute of Standards and Technology (NIST) in 1997. Since then, AES has been one of the most common symmetric cryptographic algorithms. Many hardware implementations of AES have so far been proposed with different characteristics [3-6], each of which is suited for different applications with different constraints. Recently, many faults injection attacks have been proposed on AES [7-9]. In a fault attack, attackers inject malicious faults into the VLSI design of cryptographic primitives to extract secure information (i.e., cryptographic key).

On the other hand, with transistor size downscaling, reducing power supply voltage level, increasing operating frequencies, and reducing noise margins, VLSI hardware designs will be more and more sensitive to random faults occurrence [10]. All random faults that occur in VLSI designs can be grouped into transient and permanent faults.

Various fault resilient hardware implementations of AES were proposed to thwart the random and/or malicious faults effect [11-14]. AES includes four basic operations, i.e., SubByte, ShifRows, MixColumns, and AddRoundKey. The hardware implementation of SubByte operation is realized with 16 S-Boxes that are nonlinear mapping in which each byte of state array is replaced with another byte. It also occupies much of the total AES hardware implementation area [15]. In a fault injection attack, an injected fault changes specific bits or bytes during a special round of the encryption and produces certain differences [16-19]. The nonlinear operations, namely, S-Boxes of the block ciphers, are commonly the target of DFAs. In those DFAs that faults are injected during the encryption process, the fault propagation patterns denote some relations between the input and output difference of the specific S-boxes. In almost all block ciphers, including AES, the S-box values are known, so an attacker can simply conclude the difference distribution table of the utilized S-box. The inputs of S-boxes are mainly combined with the round Keyes's chunk through some mixing operations. The attacker can reduce the search space of some secret information, i.e., a part of the key, exploiting the difference distribution table and the relations between the difference of input and output. This divide-and-conquer method is used to extract the whole cryptographic key of most block ciphers quite efficiently [20]. So, integrating its hardware implementations with an efficient fault resilient scheme is crucial for making AES robust to random and/or malicious faults. There are many online error detection schemes for SubByte implementation of AES; see, for example, [21-22].

Just a few studies among previous research works have addressed fault correction. In fact, most of the previous studies have only considered the detection task, so extra corrective operations should be employed for their solutions. In [23], a hybrid redundancy is proposed in which hardware redundancy and time redundancy are combined for fault correction in S-box. Their proposed S-box architecture can tolerate single faults. It is worth noting that the fault-tolerant S-box in [23] provides a high level of reliability against the natural faults due to the essence of electronic devices, not the malicious faults in the fault attacks.

The present paper is mainly aimed to propose a high-throughput fault-attack resistant hardware implementation of AES S-box. We propose a correction scheme at the hardware level so that the circuit frequency is not significantly affected. In this paper, a high-speed design is considered. In fact, we exploit the features of gate-level implementation of S-box, allowing the pipeline technique to speed up the hardware implementation of SubByte operation of AES. The proposed technique is also practical for any generic cipher block.

We also implement the traditional fault-tolerant configurations, triple time redundancy, and triple module redundancy of the AES S-box and compare the implementation results of the proposed architecture to both of them.

N-tuple modular redundancy (NMR) [24] is a well-known fault-tolerant scheme based on hardware redundancy. Dual modular redundancy (DMR) is the most famous realization of NMR for performing error detection task. Another special case of NMR is triple modular redundancy (TMR) in which three identical units execute the same operation and the output is deduced from the majority voter [25, 26].

Time redundancy is achieved by re-computation of an operation using the same hardware multiple times, saving results, and comparing them for the error correction or detection tasks. N-tuple temporal redundancy is a generic fault-tolerant configuration of time redundancy. The triple time redundancy (TTR) is a special form of N-tuple temporal redundancy. In this scheme, the same input data is processed through the same unit three times. The majority voter generates the output of TTR by the majority vote of these three consecutive processes [26-27].

The main contributions of this paper are as follows:

- We present an implementation of a high-throughput and lightweight S-box in the gate level for high-speed AES encryption.
- We propose a fault-attack resistant technique, i.e., FC-DMR, for real-time applications which cannot tolerate high running time and require a high-speed process. The proposed technique could generally be used in all digital functional units.
- We design a new DMR voter that is composed of standard library components and could be implemented on any digital platform, such as FPGA and ASIC.
- Finally, we implement the AES S-box in TMR and TTR configurations in the same situation as HFS-box for design metrics comparison.

The rest of the paper is organized as follows. Section 2 presents a brief background of the S-box of the AES algorithm and its implementation. Section 3 presents the proposed fault-attack resistant technique (FC-DMR) besides our DMR voter model. It also describes the HFS-box architecture. We evaluate the proposed architecture's architectural characteristics in terms of area, frequency, and throughput in Section 4. Finally, Section 5 concludes the paper.

2. S-box Implementation

In this subsection, we describe the S-box operation and its utilized architecture. The proposed S-box architecture using composite-field in [28] is employed in this paper. The S-box operation, which is believed to be most resource-consuming among other AES operations, is a nonlinear mapping on each state array byte. This nonlinear mapping is nothing but finding a multiplicative inverse over $GF(2^8)$, i.e., Galois field, which is arithmetic in a finite field (a field containing a finite number of elements) contrary to arithmetic in a field with an infinite number of elements, like the field of rational numbers. $x^{-1}\epsilon\, GF(2^8)$ is followed by an affine transformation. In other words, if $y = SB(x)$ and $X\epsilon GF(2^8)$ and $Y\epsilon GF(2^8)$, then we have:

$$y = Ax^{-1} + b = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} x^{-1} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \tag{1}$$

Since direct multiplicative inversion of S-box computation is costly, multiplicative inversion in composite fields is preferred [29]. This implementation leads to lower complexity and smaller implementation area.

The S-box implementation using composite-field and polynomial basis is illustrated in Fig. 1.

As shown in this figure, the 8-bit input of multiplicative inversion, i.e., $X = \sum_{i=0}^{7} \alpha_i x_i$in the binary field *GF* (2^8) using the transformation matrix δ, transforms to composite-field $GF(2^8)/GF(((2^2)^2)^2)$. In turn, the output of the multiplicative inverse from composite-field transforms back to binary field *GF* (2^8) by the inverse transformation matrix δ^{-1} to obtain X^{-1}. The hierarchical composite-field decomposition, i.e., $GF(((2^2)^2)^2) \rightarrow GF((2^2)^2)$, $GF((2^2)^2) \rightarrow GF(2^2)$, and $GF(2^2) \rightarrow GF(2)$, can be made using the irreducible polynomials of $x^2 + x + \lambda$, $x^2 + x + \varphi$ and $x^2 + x + 1$, respectively. As shown in Fig. 1, the output of the S-box, i.e., Y, is obtained using the affine transformation after inverse transformation (δ^{-1}) [27]. The S-box is composed of the multiplications, squaring, and inversion all of which are over $GF((2^2)^2)$. Besides these arithmetic blocks, the S-box includes modulo-2 addition that is realized by XOR gates (see Fig. 1). Considering this figure, the output of the S-box can be formulated as follows:

$$\sigma_h = ((\xi_h + \xi_l)\xi_l + \xi_h^2\lambda)^{-1}\xi_h \qquad (2)$$

$$\sigma_l = ((\xi_h + \xi_l)\xi_l + \xi_h^2\lambda)^{-1}(\xi_h + \xi_l) \qquad (3)$$

where ξ and σ are the input and output of the multiplicative inversion, respectively. It is also worth mentioning that we have used the proposed architecture for different parts of the S-box, i.e., adder and multiplier in [28].

3. Proposed Fault Correction Structure (FC-DMR)

3.1. FC-DMR

We propose a correction technique in the DMR implementation of a digital circuit (FC-DMR) depicted in Fig. 2. The proposed FC-DMR protects the operation of both combinational and sequential parts of a digital circuit in each pipeline stage. Fig. 2 depicts an instance pipeline stage *i* in the intended circuit. As depicted in this figure, our FC-DMR consists of the following elements:

- *Pipeline Logici (original)*: a part of the system's combinational logic utilized to process data in the original mode in the i^th^ pipeline stage.

- *Pipeline Logic$_i$ (redundant)*: a redundant copy of the original i^th^ pipeline stage utilized to process data in the redundant mode in the *i^th^* pipeline stage.

- *Register stage$_i$*: the register or sequential part of the *i^th^* pipeline including DMR register and two DMR voters to preserve the correct state in the presence of a fault.

- *DU*: the fault detection unit, which is actually implemented using a comparator must provide the output error signal *err^i,^* which indicates differences in the DMR register in the i^th^ pipeline stage occur.

- *CU*: the control unit producing *Err*, which is a general error signal and indicates fault occurrence in the system (any pipeline stage), i.e., a fault is detected.

The input of each pipeline stage is processed by the pipeline logic and its redundant unit.

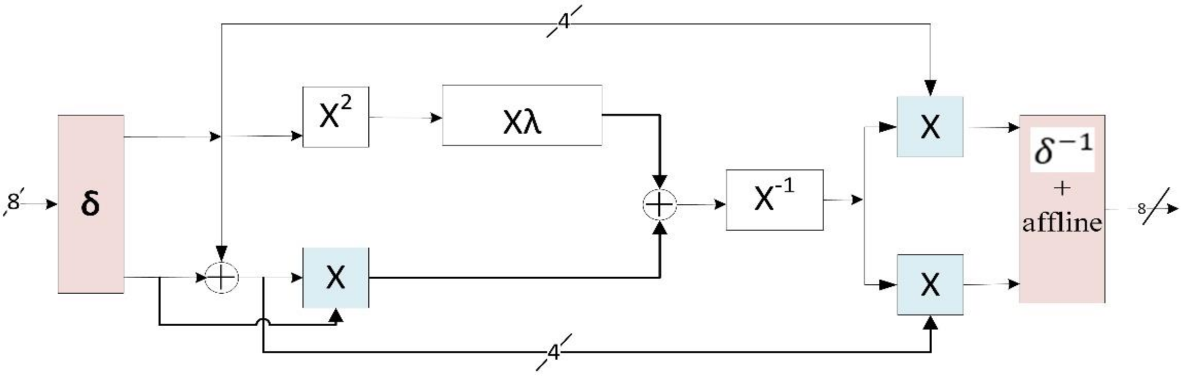


Fig. 1: Composite field-based S-box architecture.

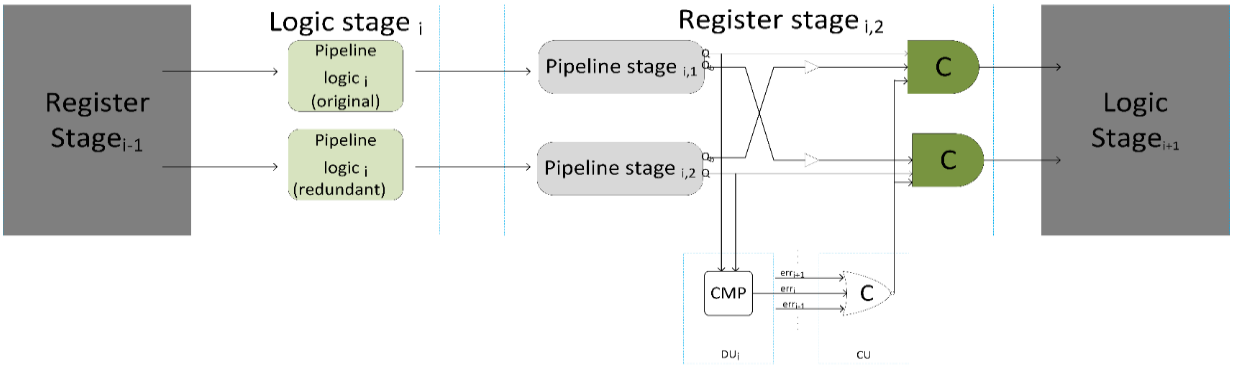


Fig. 2: The proposed fault correction technique in DMR implementation (FC-DMR).

The corresponding output of the original and redundant pipeline logic units are stored in the register stages, i.e., pipeline register$_1$ and pipeline register$_2$, respectively. If the register's contents are identical, no fault is detected. Otherwise, the comparator *CMP*^{i}'s output in *DU*^{i}, i.e., *err*^{i}, will be activated. Two DMR voters are employed to protect both combinational and sequential part of the system. The proposed technique can correct any transient fault that occurs in a single S-box. When a fault is detected in any pipeline stage components, either in the logic stages, in the pipeline registers, or in the *DU*, the *CU* will reset its output, i.e., *Err*, and later it will prevent loading the incorrect state on the output of DMR voters.

Hence, the pipeline logics process the previous correct state till the fault effect disappears. When the fault effect disappears, the next correct state is processed without any problem. This solution may put a negligible delay overhead on the critical path due to the comparison and voting.

3.2. Proposed Voter

The employed voter does the two tasks of a majority voter in the DMR technique, i.e., holding the previous state when facing a mismatch and changing the vote signal's value when both modules produce the same output (Fig. 3).

In fact, when the outputs of the two replicas are not the same as each other, which means an error has occurred, the voter holds the previous value until the two replicas' outputs become similar. Besides, our design has a delay module that is useful if the comparator faces a mismatch. This delay makes it possible to affect the enable signals. Enables are provided to control internal wires not to send the faulty signals to voter's output, which means that the pipeline stage is unchanged until the correct value gains and the sequence in our pipeline design remains unaffected.

3.3. HFS-box

The main contribution of this paper is proposing high-throughput fault-attack resistant hardware implementations of S-box.

We propose a full pipeline implementation of S-box in the composite field approach, which leads to the reduction of the circuit critical path. In fact, this solution enables us to enhance the frequency of clock signal in our proposed method and also makes it suitable for meeting the high-speed application requirements.

The proposed pipeline S-box is depicted in Fig. 4. We place pipeline registers into this schema, which are illustrated by the dotted lines. As depicted in this figure, the proposed S-box architecture (shown in Fig. 1) is divided into five stages. These pipeline registers are inserted into S-box architecture so that the critical path is optimally pipelined. This architecture is integrated with the proposed FC-DMR to achieve fault tolerance for any transient fault in both combinational and sequential parts in any pipeline stage of a single S-box, named HFS-box. In HFS-box, each DMR implementation of pipeline logic is lied between two register stages to check against fault occurrence, as depicted in Fig. 2.

4. Implementation Result

To evaluate the proposed HFS-box, we compare it with the TMR and TTR implementation of S-box, as traditional fault-tolerant structures with high fault correction capability. We report the synthesis result by using the TSMC 180 nm CMOS. We employ Verilog as the design entry description language and Synopsys DC as the synthesis tool. It should be noted the 8-bit SubByte operation is considered, so a single S-box is needed in each structure.

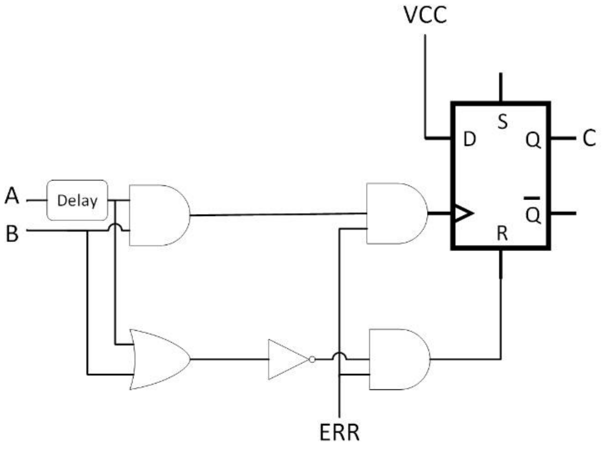


Fig. 3: Proposed voter in gate level.

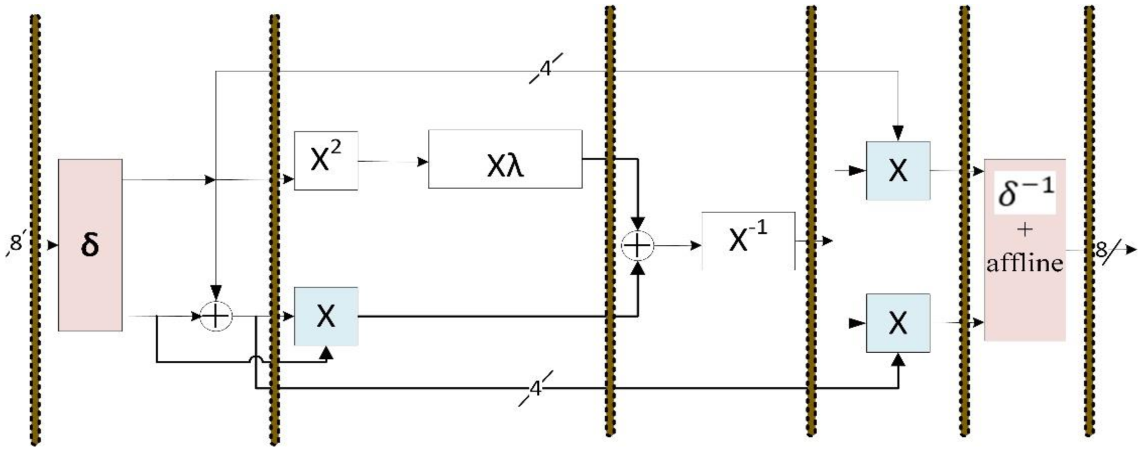


Fig. 4: The architecture of the S-box with the 5-stage pipeline.

Table 1: Throughput, maximum frequency, area result.

Design metric		Original	TMR	TTR	HFS-box
Area	GE	212.42	673.31	279.02	503.46
	% of area overhead	-	216	31.35	137
Freq.	MHz	555	525	519	492
	% of reduction	-	-5.4	-6.4	-11.3
Throuput	Mbps	4440	4200	1384	3936
	% of reduction	-	-5.4	-68.8	-11.3
Fault Tolerance	Transition	×	✓	✓	✓
	Permanent	×	✓	×	×
Security against fault attack		×	✓×	✓××	✓

In this section, the ASIC implementation results of all fault-tolerant S-box implementations are reported and compared. The design features that we consider include area, area overhead, frequency, and frequency overhead. Table 1 presents the implementation results of all fault resilient designs.

In this table, we use (4) to calculate- the-cost overhead.

$$Overhead = \frac{C_{FT} - C_O}{C_O} \tag{4}$$

where C_O is the original implementation cost (area, frequency, throughput, etc.), and C_{FT} is the cost of the fault tolerant implementation. It can be seen that TTR has the lowest area overhead (44.5% and 58.54% reduction compared to HFS-box and TMR, respectively) and, at the same time, lower throughput (64.83% and 67.04% worse than HFS-box and TMR, respectively). HFS-box requires about 503 NAND gate equivalences (GEs). Actually, it puts more area overhead than TTR but still is much better than TMR (25.22% better than TMR). However, TMR achieves the best throughput among all fault resilient architectures, its security and reliability against fault attacks is lower than our HFS-box, and also it puts much more area overhead on the original S-box than HFS-box.

The security of TMR and TTR is overshadowed by the majority voter. In fact, the majority voter is a bottleneck for these traditional fault-tolerant schemes. There are many works focused on introducing fault-tolerant majority voter. But, we do not address them because those research works are out of the scope of this paper. In addition, according to the proposed structure for the voter, it can be seen that the proposed design can detect symmetric transient errors in a very short period. However, TMR and TTR configurations do not have such capability. So, the proposed HFS-box can offer a higher level of reliability and better security against fault attacks, as mentioned in Table 1.

In fact, the proposed low-cost HFS-box can continue its proper task without a considerable negative impact on the system speed or even any traditional recovery scheme. It is a suitable fault-tolerant technique for resource-constrained applications that require a high level of security.

5. Conclusion

In this paper, we proposed a lightweight high-throughput fault-attack resistant architecture for composite field S-box implementation of AES, which consumes the largest space in AES, named HFS-box. The proposed fault-attack resistant technique is based on fault correction in DMR implementation (FC-DMR) combined with a temporal redundancy technique. It can correct transient faults, which may occur in S-box naturally or maliciously. Our solution is valid for any digital circuit implementation (especially block cipher hardware implementation) with different levels of pipelining. HFS-box uses five pipeline stages to meet the real-time application requirements for speed and throughput. Indeed, we inserted pipeline registers in optimal places in the S-box architecture. Furthermore, we introduced a compatible DMR voter with our FC-DMR. The proposed HFS-box and two well-known methods with high fault-tolerant ability, i.e., TMR and TTR, were implemented on ASIC using TSMC 180nm CMOS technology, and their area, frequency, and throughput were derived and reported. The synthesis results pointed out that the HFS-box had a low area overhead (137%) and low throughput degradation (11.3) compared with other fault-tolerant schemes.

CRediT Authorship Contribution Statement

Mahdi Taheri: Conceptualization, Formal analysis, Methodology, Resources, Roles/Writing - original draft, Writing - review & editing. **Saeideh Sheikhpour:** Formal analysis, Writing - review & editing. **Mohammad Saeed Ansari:** Conceptualization**,** Formal analysis. **Ali Mahani:** Project administration, Supervision**,** Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The ethical issues; including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, redundancy has been completely observed by the authors.

References

[1] S. Patranabis, and D. Mukhopadhyay, *Fault tolerant architectures for cryptography and hardware security*. Berlin: Springer, 2018.

[2] *Announcing the advanced encryption standard (AES)*, Federal Information Processing Standards Publication 197, no. 1-51, 3-3., Nov. 2001.

[3] D. Bui, D. Puschini, S. Bacles-Min, E. Beigné and X. Tran, "AES datapath optimization strategies for low-

power low-energy multisecurity-level internet-of-things applications", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 12, pp. 3281-3290, 2017.

[4] D.-S. Kundi, A. Aziz, N. Ikram, "A high performance ST-Box based unified AES encryption/decryption architecture on FPGA", *Microprocessors and Microsystems*, vol. 41, no. 1, pp. 37-46, 2015.

[5] S. S. Priya,, P. Karthigaikumar, N. M.Siva-Mangai , P. K. Gaurav-Das, "An-efficient hardware architecture for high throughput AES encryptor using MUX based sub pipelined S-box", *Wireless Personal Communications*, vol. 94, no. 4, pp.2259-2273, 2017.

[6] S. Shanthi Rekha and P. Saravanan, "Low-cost AES-128 implementation for edge devices in iot applications", *Journal of Circuits, Systems and Computers*, vol. 28, no.4, pp.1950062, 2019.

[7] E. Biham, A. Shamir, "Differential fault analysis of secret key cryptosystems", in *Advances in Cryptology (CRYPTO '97), FLEXChip Signal Processor (MC68175/D),* 1997 pp. 513-525.

[8] T. Fuhr, E. Jaulmes, V. Lomné, and A. Thillard, "Fault attacks on AES with faulty ciphertexts only", *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography*, Santa Barbara, CA, 2013, pp. 108-118.

[9] P. Dusart, G. Letourneux, and O. Vivolo, "Differential fault analysis on AES", Cryptology ePrint Archive: Report 2003/010, 2003, [online] Available: http://www.iacr.org.

[10] S. S. Mukherjee, J. Emer, and S. K. Reinhardt, "The soft error problem: an architectural perspective", in *11th International Symposium on High-Performance Computer Architecture*, San Francisco, CA, USA, 2005, pp. 243-247.

[11] X. Guo and R. Karri, "Recomputing with permuted pperands: A concurrent error detection approach", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 10, pp. 1595-1608, 2013.

[12] M. Mozaffari-Kermani, A. Reyhani-Masoleh, "Concurrent structure independent fault detection schemes for the advanced encryption standard", *IEEE Transaction on computers*, vol. 59, no. 5, pp. 608-622, 2010.

[13] H. Mestiri, F. Kahri, B. Bouallegue, and M. Machhout, "A high-speed AES design resistant to fault injection attacks", *Microprocessors and Microsystems Journal Elsevier*, vol. 41, pp. 47-55, 2016.

[14] M. Bedoui, H. Mestiri, B. Bouallegue, M. Marzougui, M. Qayyum, and M. Machhout, "An improved and efficient countermeasure against fault attacks for AES", *2017 2nd International Conference on Anti-Cyber Crimes (ICACC),* Abha, 2017, pp. 209-212.

[15] S. Morioka and A. Satoh, "An optimized s-box circuit architecture for low power aes design", *Cryptographic Hardware and Embedded Systems-CHES*, 2003, pp. 271–295.

[16] N. Liao, X. Cui, T. Wang, K. Liao, D. Yu, and X. Cui, "A high-efficient fault attack on AES S-box", *International Conference on Information Science & Technology*, pp. 210-215, 2016.

[17] Y. Liu, X. Cui, J. Cao, and X. Zhang, "A hybrid fault model for differential fault attack on AES", *2017 IEEE 12th International Conference on ASIC (ASICON),* 2017, pp. 784-787.

[18] J. Park, S. Moon, D. Choi, Y. Kang, and J. Ha, "Fault attack for the iterative operation of AES S-Box", *5th International Conference on Computer Sciences and Convergence Information Technology*, 2010, pp. 550-555.

[19] C.-N. Chen and S.-M. Yen, "Differential fault analysis on AES key schedule and some countermeasures", *In Information Security and Privacy, LNCS Springer 2003*, volume 2727, pages 118-129, 2003.

[20] S. Ali, X. Guo, R. Karri, and D. Mukhopadhyay, "Fault attacks on AES and their countermeasures", *in Secure System Design and Trustable Computing*, 2015, pp. 163-208.

[21] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "A lightweight high-performance fault detection scheme for the advanced encryption standard using composite fields", *in IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 1, pp. 85-91, Jan. 2011.

[22] I. Koren, S.Y.H. Su, "Reliability analysis of N-modular redundancy systems with intermittent and permanent faults", *in IEEE Transactions on Computers*, vol. C-28, no. 7, pp. 514-520, July 1979.

[23] F. Flammini, N. Mazzocca, V. Vittorini, and S. Marrone, "A new modeling approach to the safety evaluation of n-modular redundant computer systems in presence of imperfect maintenance", Re*liability Eng. Syst. Safety (RESS),* vol. 94, no. 9, pp. 1422-1432, 2009.

[24] S. Sheikhpour, A. Mahani, and N. Bagheri, "Reliable advanced encryption standard hardware implementation: 32-bit and 64-bit data-paths", Microprocessors and Microsystems, vol. 81, p.103740, 2021.

[25] M. Mozaffari-Kermani, A. Reyhani-Masoleh, "Fault detection structures of the S-boxes and the inverse S-boxes for the advanced encryption standard", Journal of Electronic Testing, vol. 25, no. 4, pp. 225-245, 2009.

[26] T. An, L. A. d. B. Naviner, and P. Matherat, "A low cost reliable architecture for S-boxes in AES processors", *2013 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS)*, 2013, pp. 155-160.

[27] M. Taheri, S. Sheikhpour, MS. Ansari, and A. Mahani, "DMR-based technique for fault tolerant AES S-box architecture", arXiv preprint arXiv:2009.05329. 2020.

[28] N. Ahmad, S. M. Rezaul Hasan, "Low-power compact composite field AES S-box/Inv S-box design in 65 nmCMOS using Novel XOR Gate", *Integration*, vol. 46, no. 4, pp. 333-344, 2013.

[29] S. Sheikhpur, M. Taheri, M.S. Ansari, and A. Mahani, "Strengthened 32-bit AES implementation: Architectural error correction configuration with a new voting scheme", *IET Computers & Digital Techniques*, 2021.

BIOGRAPHY

Mahdi Taheri received his B.Sc. degree in electronic engineering from the Khaje Nasir University of Technology (KNTU), Tehran, Iran, in 2017 and his M.Sc. degree in Electronic Engineering from the Shahid Bahonar University of Kerman, Kerman, Iran, in 2020. Since then, he was with the RSS Lab at the Shahid Bahonar University of Kerman for 1 year, and now, he is studying his Ph.D. at the Tallinn University of Technology (TalTech). His research interests focus on hardware assessment and reliability of neural networks, fault-tolerant designs, and FPGA-based accelerators.

Saeideh Sheikhpour received her Ph.D. in Electrical Engineering from the Shahid Bahonar University of Kerman, Kerman, Iran, in 2019. Her current research interests are fault-tolerant designs, reliable and secure cryptographic hardware designs, circuit and system reliability and quality, soft computing, artificial intelligence, and sensor networks.

Mohammad Saeed Ansari received the BSc and MSc degrees in electrical and electronic engineering from the Iran University of Science and Technology, Tehran, Iran, in 2013 and 2015, respectively, and the PhD degree in integrated circuits and systems from the University of Alberta, Edmonton, AB, Canada, in 2019. He is presently a digital design engineer at Eideticom Computational Storage, Calgary, AB, Canada. His research interests include approximate computing and developing hardware accelerator IP cores for data compression/decompression, neural networks, and digital signal processing applications.

Ali Mahani received his B.Sc. degree in Electronic Engineering from the Shahid Bahonar University of Kerman, Iran in 2001, and his M.Sc. and Ph.D. degrees both in Electronic Engineering from the Iran University of Science and Technology (IUST), Tehran, Iran in 2003 and 2009, respectively. Since then, he has been with the Electrical Engineering Department of the Shahid Bahonar University of Kerman, where he is currently an associate professor. His research interests focus on fault-tolerant designs, FPGA-based accelerators, approximate digital circuits, stochastic computing, and networked systems.