Lightweight Hardware Architectures for Fault Diagnosis Schemes of Efficiently-Maskable Cryptographic Substitution Boxes

Mehran Mozaffari Kermani, *Senior Member*, *IEEE*, and Reza Azarderakhsh, *Member, IEEE*

***Abstract*—As we are close to the advent of quantum computing capability, the potential use of resistant algorithms, i.e., code-based, lattice-based, hash-based, multivariate-quadratic-equations, and symmetric-key cryptographic algorithms, depends on many sensitive factors including resistance against natural and malicious faults. Active side-channel analysis attacks (SCAs) such as fault analysis attacks and passive ones, e.g., power analysis attacks, have been effective to compromise algorithmically-secure crypto-solutions including the emerging lightweight block ciphers. Nevertheless, one can provide fault diagnosis approaches to ameliorate the former and use efficient masking mechanisms for the latter. There has been recent work to account for power analysis attacks resistivity before developing new lightweight block ciphers; nonetheless, in this paper, we present error detection approaches for such ciphers and present insights towards future directions for potential, combined power and fault analysis attacks resistivity as a major deciding factor in developing SCA-resistant lightweight block ciphers. Through error simulations, the theoretical back-bone of the presented error detection scheme for a lightweight block cipher case study is benchmarked. The proposed design factor can be tailored based on the required security, fault resistivity, and overhead tolerance of both classical and post-quantum cryptography.**

I. INTRODUCTION

Cryptographic protocols such as elliptic curve digital signature algorithm (ECDSA) and elliptic curve Diffie–Hellman (ECDH), or the ones based on Rivest-Shamir-Adleman (RSA) will be no longer algorithmically-secure[1] in post-quantum era. Various lightweight hash functions and block/stream ciphers have been developed to account for tight constraints of different usage models including implantable and wearable medical devices, smart infrastructures, and the like. Not only are such symmetric-key cryptographic algorithms used as stand-alone security providers, but they are also utilized in other post-quantum resistant approaches, e.g., hash functions and stream ciphers used in stateless hash-based post-quantum scheme SPHINCS [1]. Nevertheless, it is well-known that through active or passive side-channel attacks (SCAs), such cryptographic architectures may leak sensitive key-related information.

M. Mozaffari Kermani is with the Department of Electrical and Microelectronic Engineering, Rochester Institute of Technology, Rochester, NY 14623, USA (e-mail: m.mozaffari@rit.edu).

R. Azarderakhsh is with the Department of Computer and Electrical Engineering and Computer Science and is an I-SENSE Fellow, Florida Atlantic University, Boca Raton, FL, USA (e-mail: razarderakhsh@fau.edu).

[1]We use algorithm security and implementation security to accurately differentiate different types of security.

Protecting lightweight block ciphers against SCAs incurs performance and complexity overheads which are undesirable. Thus, embedding SCA resistivity as a deciding factor for design (it is referred to as design-for-SCA-resistivity hereafter) beforehand and not as an aftermath is a natural alternate. There has been previous work on designing (or tweaking the previously-devised designs) to account for power analysis attacks utilizing "masking" as an efficient embedded countermeasure [2]. We would like to emphasize that such designs have lead to better performance and implementation metrics compared to the ones based on adding the same masking countermeasures to the already-designed block ciphers.

In this paper, we present error detection schemes for efficiently-maskable substitution boxes. We take into account the nonlinear substitution boxes of (lightweight) block ciphers, protected against power analysis through masking, and propose reliability approaches for thwarting natural and malicious faults. This work takes into account two lightweight block ciphers (one efficiently-maskable and the other not) for proposing efficient error detection schemes. We note that natural (hardware failures, for instance, single event upsets, electromagnetic waves, or external radiations) and malicious (fault analysis attacks) error detection has been considered in the previous work, see, for instance, [3], [4], [5], [6], [7]. We also note that there has been previous work on investigating the resistivity of fault attack countermeasures to power analysis attacks and on applying power analysis attacks to fault analysis attack-resistant architectures [8], [9], [10], [11], [12].

II. PRELIMINARIES

In what follows, we present preliminaries on masking, substitution boxes, and the block ciphers ICEBERG and Zorro.

Masking. Masking [13] is a widely-used method for thwarting power analysis attacks through randomizing the internal state of the implementations so that information leakage about the intermediate values is obstructed. As such, knowing d internal values and computations is ineffective when we deal with d-th order SCA implementation security.

Substitution boxes. Nonlinear substitution boxes, e.g., S-boxes of block ciphers, are known to be the most difficult to mask, e.g., the time complexity of their masking grows at least quadratically with the order d. Devising block ciphers which are designed to be effectively masked has been the objective of previous works, see, for instance, [14]. Non-bijective

978-1-5090-6113-6/16/$31.00 ©2016 IEEE

substitution boxes usually lead to simple non-profiled attacks; yet, non-profiled attacks do not exist against bijective (having one-to-one correspondence) substitution boxes. Efficiently-maskable bijective substitution boxes can be constructed using smaller substitution boxes, e.g., KHAZAD [15] and ICEBERG [16], or through exhaustive search of permutations over $GF(2^n)$ which is computationally infeasible for large n. It is noted that typically, 4-bit and 8-bit S-boxes are utilized in block ciphers. The former is used in lightweight ciphers through look-up tables or logic gates and the latter, like the one in the Advanced Encryption Standard, can be implemented through composite fields, e.g., $GF((2^2)^2)^2$ or $GF(2^2)^4$, or through memory macros implemented on application-specific integrated circuit (ASIC) or field-programmable gate array (FPGA).

Block ciphers ICEBERG and Zorro. ICEBERG operates on 64-bit blocks and uses a 128-bit key. It is an involutional iterative block cipher based on the repetition of identical round functions which are key-dependent. This block cipher consists of two nonlinear substitution layers, a permutation layer, and key-addition/linear/matrix multiplication layers. For more details, which are not presented for the sake of brevity, one can refer to [16]. Through the aforementioned approaches, Zorro (an efficiently-maskable block cipher) has been developed. Block size and key size of Zorro are 128 bits, with iterative 24 rounds and 6 steps (combination of 4 rounds). In the substitution transformation within Zorro, only 4 substitution boxes are applied to the 4 bytes of the first row in the state matrix (comparing 24 rounds $\times$ 4 substitution boxes in Zorro with 10 rounds $\times$ 16 substitution boxes in the Advanced Encryption Standard shows roughly half substitution boxes for Zorro).

III. Proposed Error Detection Approach

In this section, we first present the error detection approaches for the block cipher ICEBERG which has a relatively-similar architecture (not in terms of being maskable though) to Zorro. We consider different linear and nonlinear layers of this block cipher and for the sake of brevity, we do not present all of our derived computations. Then, we compare the results with Zorro which has been constructed to be efficiently-maskable, i.e., power analysis attack resistant. We note that there is no limitation in generalizing this comparison for other block ciphers.

A. Linear and Nonlinear Layers of ICEBERG

As we are interested in error detection complexity, we examine different layers in function γ independently to conclude the cost[2]. Fault diagnosis for the substitution boxes s_0, s_1 of function γ can be performed through concurrent error detection and signatures. We propose using signatures, e.g., parities for detecting odd faults and single stuck-at faults and interleaved parities to be able to detect burst faults as well, and storing them with the original values in (a) distributed memories using look-up tables or block memories on FPGAs, or (b) synthesized memories or memory macros on ASICs. The results of our derivations are presented in Tables I-IV for s_0, s_1, respectively. We note that parallel application of s_0, s_1 (and thus parallel error detection of which) creates substitution layers S_0, S_1 such that for $0 \leq j \leq 1$, $S_j : \mathbb{Z}_{2^4}^{16} \rightarrow \mathbb{Z}_{2^4}^{16} : x \rightarrow y = S_j(x) \Leftrightarrow y_i = s_j(x_i)$, where $0 \leq i \leq 15$. Thus, detection of the error in any of s_0, s_1 would cause the error indication flag of the entire S_0, S_1 to be asserted.

Table I
INTERLEAVED PARITY PREDICTION FOR s_0.

Row, Col.	Binary interleaved parity pairs							
1, 1-8	00	01	01	00	10	11	11	10
2, 1-8	01	10	00	01	11	11	00	01
1, 9-16	10	11	11	10	01	01	01	00
2, 9-16	00	10	11	01	11	00	10	10

Table II
PREDICTED PARITY FOR s_0 FOR SINGLE/ODD FAULTS.

Row, Col.	Binary interleaved parity pairs							
1, 1-8	0	1	1	0	1	0	0	1
2, 1-8	1	1	0	1	0	0	0	1
1, 9-16	1	0	0	1	1	1	1	0
2, 9-16	0	1	0	1	0	0	1	1

Permutations of function γ do not change the signatures. Permutation layer $P8$ consists of the parallel application of 8 permutations $p8$ to the state, where $p8$ consists of bit permutations on 8-bit blocks of data such that $P8 : \mathbb{Z}_{2^8}^8 \rightarrow \mathbb{Z}_{2^8}^8 : x \rightarrow y = P8(x) \Leftrightarrow y(8i + j) = x(8i + p8(j))$ for $0 \leq i, j \leq 7$. In other words, we have $(\textcircled{S}x) = (\textcircled{S}y)$, where $\textcircled{S}$ denotes signatures such as interleaved parity and the like. Finally, we achieve the error detection of the entire function γ through the corresponding signatures as $\textcircled{S}\gamma : \mathbb{Z}_2^{64} \rightarrow \mathbb{Z}_2^{64} \Longrightarrow (\textcircled{S}S_0) \bigcirc (\textcircled{S}P_8) \bigcirc (\textcircled{S}S_1) \bigcirc (\textcircled{S}P_8) \bigcirc (\textcircled{S}S_0)$, where $\bigcirc$ is used to show that the operations are done consecutively.

Modulo-2 addition with the key is through XOR gates (σ_k). Moreover, the matrix multiplication (denoted as M) in the linear layer ϵ_k is a linear step through which, for the 4-bit entries x_i where $0 \leq i \leq 15$, we perform a multiplication with the 4×4 matrix $V = [0, 1, 1, 1; 1, 0, 1, 1; 1, 1, 0, 1; 1, 1, 1, 0]$. We derive parities and interleaved parities for this linear step with low cost as follows.

Remark 1. The predicted parities for the matrix multiplication are equal to actual parities and there is no cost for their derivation.

Adjacent fault are very relevant for both natural and malicious faults. Natural faults can happen as single event upset (SEU) and multiple event upset (MEU), and malicious faults are usually injected as transient faults which are preferred to be single stuck-at faults; however, due to technological constraints, adjacent faults may incur.

Remark 2. The interleaved parities are derived for this linear sub-block simply through $x_0 + x_1$ and $x_2 + x_3$, where $+$ denotes modulo-2 addition.

The critical path delay of the predicted parities in Remark 1 is zero and the one for the interleaved parity in Remark 2

[2]Throughout this paper, small and capital letters are used with different intentions for different substitutions, permutations, and other functions.

Table III
INTERLEAVED PARITY PREDICTION FOR s_1.

Row, Col.	Binary interleaved parity pairs							
1, 1-8	00	01	01	00	10	11	11	10
2, 1-8	10	11	00	00	00	01	11	10
1, 9-16	10	11	11	10	01	01	01	00
2, 9-16	01	11	01	10	00	11	10	01

Table IV
PREDICTED PARITY FOR s_1 FOR SINGLE/ODD FAULTS.

Row, Col.	Binary interleaved parity pairs							
1, 1-8	0	1	1	0	1	0	0	1
2, 1-8	1	0	0	0	0	1	0	1
1, 9-16	1	0	0	1	1	1	1	0
2, 9-16	1	0	1	1	0	0	1	1

is T_X (delay of an XOR gate).

Based on the aforementioned remarks and considering the linear layer $\epsilon_k : \mathbb{Z}_2^{64} \rightarrow \mathbb{Z}_2^{64} : \epsilon_k \equiv P64 \bigcirc P4 \bigcirc \sigma_k \bigcirc M \bigcirc P64$, where $P64, P4$ are permutations (and thus do not affect the signatures) whose definitions are not presented for the sake of brevity, error detection for key schedule is the combination of those for the aforementioned transformations whose error detection can be performed through union of the methods presented.

The following remark presents the fault diagnosis scheme for the key selection unit of ICEBERG.

Remark 3. The key selection function of ICEBERG includes a compression function that selects 64 bytes of K^i having odd indices. A 4×4 key selection box is applied in parallel. We present the interleaved parities for the selection box as follows: $\hat{P}_1 = [(x_0 + x_1 + x_2).sel \vee (x_0 + x_1).\overline{sel}] + [(x_0 + x_2 + x_3).sel \vee (x_2 + x_3).\overline{sel}], \hat{P}_2 = [(x_1 + x_2).sel \vee x_1.\overline{sel}] + [(x_0 + x_3).sel \vee x_3.\overline{sel}]$.

1) Intelligent Attacker Assumption: We note that the fundamental difference between security attacks and random faults is the intelligent-attacker assumption. Injection of random faults mimics errors happening due to natural causes. In contrast, the intelligent adversary running fault-based cryptanalysis will carefully determine the fault (s)he is going to inject and perform injection right at the calculated position and point of time. Consequently, we note that just trying random faults will not be helpful in breaking most ciphers. As such, in addition to the presented work here, we provide error detection approaches based on recomputing with encoded operands for both transient and permanent faults. The merit of these approaches is that they can detect both transient and permanent faults and unlike (interleaved) parities, they are not confined to certain fault models, e.g., random faults, as derived by our simulations.

2) Data Reliability vs. *Availability:* One can use pipeline-registers to sub-pipeline the structures to break the timing path to approximately equal halves. This trend (which can be scaled to n stages) is consecutively executed for normal (N) and encoded (E) operands for the n stages. Through such approach, low degradation in the throughput at the expense of more area overhead. We utilize Fig. 1 to show a possibility for such a scheme. Depending on the requirements in terms of reliability *vs.* availability, one can tailor these approaches to fulfill such constraints, e.g., in Fig. 1, an illustrative compromise is seen where three sub-segments are considered (three encoded [rotated or shifted] sub-operands preceded by three normal sub-operands).

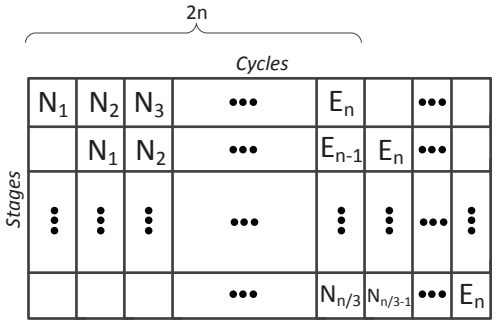

Figure 1. Data reliability *vs.* availability.

B. Comparison with Zorro

Zorro and ICEBERG are two instances of block ciphers in which larger substitution boxes are constructed from smaller ones. In ICEBERG, three applications of 4-bit substitution box layers, interleaved with a bit permutation, are used. However, although it is algorithmically-secure, the cost of six 4-bit substitution boxes to compute an 8-bit substitution box is not suitable for low-complexity masking. This has been resolved in Zorro with a Feistel network of substitution boxes along with an invertible 8×8 binary matrix which is shown in Fig. 2. Moreover, Zorro can be constructed through $GF(2^7)$ substitution boxes as shown in Fig. 2. Using signatures for these two block ciphers with respect to their 8-bit substitution boxes would result in the same complexity. This is also the case for recomputing with encoded operands. For signature-based approaches, based on the hardware platform, one needs to store the resulting signatures in look-up tables or memory blocks and the cost is relative to the size of substitution boxes. However, if one considers 4-bit substitution boxes which are combined to create an 8-bit box, Zorro has clear advantage simply because it has less complexity.

C. Error Simulations

Using signatures (interleaved parity prediction as our case study without losing of generality), we have derived the error detection capability of block cipher ICEBERG. Throughout this paper, both single and multiple stuck-at faults have been considered. Interleaved parity bits are stored in the look-up tables of the substitution boxes of ICEBERG and for the linear multiplication, we have used one-bit signatures with no cost as derived in the previous section. Both stuck-at zero and stuck-at one faults are injected in multiple locations. Finally, any error detected in these structures (iterated based on the number of rounds) leads to error indication flag assertion for the entire ICEBERG block cipher.

For single-bit and multiple-bit fault injection, we generated 10,000 faults for both types of stuck-at zero and stuck-at

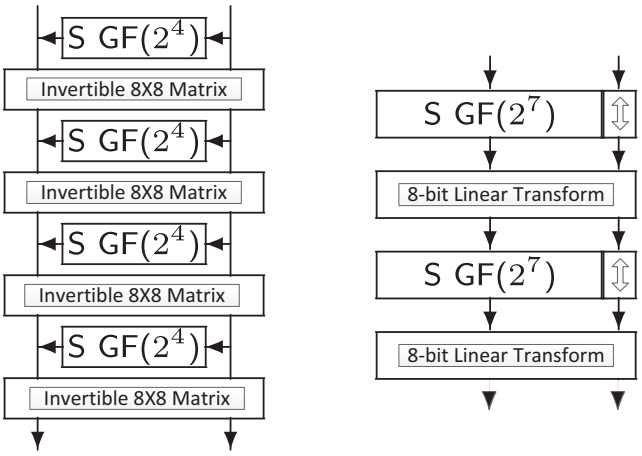

Figure 2. Creating substitution boxes in Zorro through smaller ones and through $GF(2^7)$ constructions.

one faults. The proposed approaches detect both transient and permanent faults and the results of our simulations show that the detection capability is 99.99% (with very low percent of false alarms, i.e., the errors that are detected in the intermediate rounds and are not transferred to the output of the entire ICEBERG and thus falsely alarm the errors without having erroneous final output for the block cipher).

IV. Conclusions

The main constraint in low-cost maskability is to have combined algorithmic/implementation security and low-overhead power analysis attack resiliency. This is, in-large, in-line with design-for-error-detection as similarly we need to have algorithmically-secure designs (for instance, substitution boxes of Zorro) and of course high error detection resiliency. Suitable error detection approaches for general VLSI systems can be vulnerable to fault analysis attacks. Furthermore, the choice of error detection scheme, e.g., storing the signatures in the substitution box implemented as memory blocks on FPGAs or deriving them by logic gates in composite fields on ASICs, affects the approach for design-for-SCA-resistivity. On the other hand, this choice is dependent on the eventual objectives of the designs, e.g., if the bottleneck is performance, recomputing with encoded operands is not justifiable. Thus, a useful insight is to choose secure schemes (both algorithmically-secure and secure against fault analysis attacks) which comply with the eventual design objectives.

In this paper, we have explored error detection approaches for efficiently-maskable substitution boxes for the hardware implementations of symmetric-key cryptography and, in particular, block ciphers. We have taken into account the non-linear substitution boxes of block ciphers (with emphasis on lightweight ones), protected against power analysis through masking, and presented insight to achieve reliability approaches. The case studies of Zorro and ICEBERG block ciphers have been selected and insights for future, potential combined maskability and error detection have been presented. With future, emerging directions towards post-quantum cryptography era, these considerations open new research areas on side-channel analysis resiliency. Finally, we would like to emphasize that many of the currently-under-investigation authenticated encryption schemes (CAESAR competition [17]) contain substitution boxes which can be designed for thwarting power analysis attacks (not as aftermath) and thus can be benefited from the presented insights.

Acknowledgments

This work was performed under the U.S. federal agency award 60NANB16D245 granted from U.S. Department of Commerce, National Institute of Standards and Technology (NIST).

References

[1] D. J. Bernstein, D. Hopwood, A. Hülsing, T. Lange, R. Niederhagen, L. Papachristodoulou, M. Schneider, P. Schwabe, and Z. Wilcox-O'Hearn, "SPHINCS: Practical stateless hash-based signatures," in *Proc.* EUROCRYPT, 2015, pp. 368-397.

[2] B. Gérard, V. Grosso, M. Naya-Plasencia, and F.-X. Standaert, "Block ciphers that are easier to mask: How far can we go?," in *Proc. CHES,* 2013, pp. 383-399.

[3] M. Mozaffari Kermani, R. Azarderakhsh, C.-Y. Lee, and S. Bayat-Sarmadi, "Reliable concurrent error detection architectures for extended Euclidean-based division over $GF(2^m)$," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst*., vol. 22, no. 5, pp. 995-1003, 2014.

[4] X. Guo, D. Mukhopadhyay, C. Jin, and R. Karri, "Security analysis of concurrent error detection against differential fault analysis," *J. Cryptographic Engineering,* vol. 5, no. 3, pp. 153-169, 2015.

[5] M. Mozaffari Kermani, R. Azarderakhsh, and A. Aghaie, "Fault detection architectures for post-quantum cryptographic stateless hash-based secure signatures benchmarked on ASIC," *ACM Trans. Embedded Computing Syst.* (special issue on Embedded Device Forensics and Security: State of the Art Advances), to appear in 2016.

[6] M. Mozaffari Kermani, R. Azarderakhsh, C.-Y. Lee, and S. Bayat-Sarmadi, "Reliable concurrent error detection architectures for extended Euclidean-based division over $GF(2^m)$," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst*., vol. 22, no. 5, pp. 995-1003, 2014.

[7] M. Mozaffari Kermani and A. Reyhani-Masoleh, "A lightweight high-performance fault detection scheme for the Advanced Encryption Standard using composite fields," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 1, pp. 85-91, Jan. 2011.

[8] F. Regazzoni, T. Eisenbarth, L. Breveglieri, P. Ienne, and I. Koren, "Can knowledge regarding the presence of countermeasures against fault attacks simplify power attacks on cryptographic devices?" in *Proc. IEEE Int. Symp. Defect and Fault Tolerance in VLSI Systems (DFT),* 2008, pp. 202-210.

[9] F. Regazzoni, T. Eisenbarth, J. Großschädl, L. Breveglieri, P. Ienne, I. Koren, and C. Paar, "Power attacks resistance of cryptographic S-Boxes with added error detection circuits" in *Proc. IEEE Int. Symp. Defect and Fault Tolerance in VLSI Systems (DFT),* 2007, pp. 508-516.

[10] T. Schneider, A. Moradi, and Tim Güneysu, "ParTI - Towards combined hardware countermeasures against side-channel and fault-injection attacks," to appear in CRYPTO 2016.

[11] H. Pahlevanzadeh, J. Dofe, and Q. Yu, "Assessing CPA resistance of AES with different fault tolerance mechanisms," in *Proc. Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2016, pp. 661-666.

[12] K. J. Kulikowski, M. G. Karpovsky, and A. Taubin, "DPA on faulty cryptographic hardware and countermeasures," in *Proc. FDTC,* 2006, pp. 211-222.

[13] L. Goubin and J. Patarin, "DES and differential power analysis," in *Proc. CHES*, 1999, pp. 158-172.

[14] G. Piret, T. Roche, and C. Carlet, "PICARO - a block cipher allowing efficient higher-order side-channel resistance," in *Proc. ACNS*, 2012, pp. 311-328.

[15] P. Barreto and V. Rijmen, "The KHAZAD legacy-level block cipher," Primitive submitted to NESSIE, p. 4, 2000.

[16] F.-X. Standaert, G. Piret, G. Rouvroy, J.-J. Quisquater, and J.-D. Legat, "ICE-BERG: An involutional cipher efficient for block encryption in reconfigurable hardware," in *Proc. FSE*, 2004, pp. 279-299.

[17] CAESAR competition, http://competitions.cr.yp.to/caesar.html, accessed June 2016.