Empirical Evaluation of the Resistance of Novel Non-Algebraic AES S-Boxes to Power Side-Channel Attacks

Samuele Yves Cerini[1,2,*], Gianluca Roascio[2,*,†], Nicolò Maunero[1,2,*] and Paolo Prinetto[2,*]

[1]Scuola IMT Alti Studi Lucca, Piazza S. Ponziano 6, 55100 Lucca, LU, Italy
[2]CINI Cybersecurity National Lab, Via Ariosto 25, 00185 Roma, RM, Italy

Abstract

In the area of hardware security, the exploitation of Side-Channel Analysis (SCA) to attack hardware devices has become a major issue in the last 20 years. The study of effective countermeasures is crucial, as this class of attacks reaches higher rates of effectiveness with respect to classical cryptanalysis. While implementation-level countermeasures are achieving promising results, the academic community has recently focused on solutions that can reduce leakage from the cryptographic mathematical layer, regardless of the underlying hardware/software architecture. In the field of symmetric encryption schemes (such as AES), novel substitution structures have been proposed, claiming an improved side-channel resistance without any additional costs in terms of area, performance or power consumption. To the best of our knowledge, most of these solutions have been studied only from a mathematical point of view, and are still lacking practical experimentation on resource-constrained devices.

This paper provides an empirical evaluation of the latest S-Box proposals. The necessary data has been collected in a reference scenario with limited noise effects, targeting an unprotected software implementation of the AES-128 algorithm running on an 8-bit microcontroller. The results indicate that, despite claims of enhanced resistance to Side-Channel Analysis, these new countermeasures do not offer a meaningful improvement over the standard AES implementation and are insufficient to prevent a successful attack. Moreover, when compared to widely known implementation-level countermeasures such as masking and hiding, these novel approaches appear far less promising: established techniques provide substantially stronger protection in practice for implementations capable of supporting them.

Keywords

Side-Channel Analysis, Hardware Attacks, Power Analysis, S-Boxes, AES, Cryptography, ChipWhisperer,

1. Introduction

The widely known *Spectre* [1] and *Meltdown* [2] attacks demonstrated that Side-Channel Analysis (SCA) [3] can be exploited to target any computing device, from consumer-grade personal computers, up to large scale mainframes, passing through embedded and IoT devices. Due to their physical consistency, hardware components unintentionally release certain "clues", such as power consumption, execution time, electromagnetic waves, and even microsounds. These metrics, when combined with other techniques that exploit knowledge of the internal components or the adopted algorithm, could pose a critical threat to information security. The hardware layer of any modern device plays a primary role in Information System security: it represents, by construction, *the last line of defense* against intrusions [4]. Hardware is in fact the base which all the other layers rely on. Therefore, a hardware-targeting attack may render useless all the defences implemented in the upper layers, like the ones related to system and application software [5].

The research for viable countermeasures against Side-Channel Analysis presents a multitude of methodologies and tradeoffs to evaluate. Two main approaches have been developed over time, depending on the level of abstraction adopted: *implementation-level* and *cryptographic-level* countermeasures.

Joint National Conference on Cybersecurity (ITASEC & SERICS 2026), February 09-13, 2026, Cagliari, IT
*Corresponding author
†Activities carried out while at Politecnico di Torino, currently employed at Italy's National Cybersecurity Agency (ACN)
samuele.cerini@imtlucca.it (S. Y. Cerini); g.roascio@acn.gov.it (G. Roascio); nicolo.maunero@imtlucca.it (N. Maunero); paolo.prinetto@cybersecnatlab.it (P. Prinetto)
0009-0001-4961-8915 (S. Y. Cerini); 0000-0003-0457-0855 (G. Roascio); 0000-0002-4331-1066 (N. Maunero); 0000-0003-2400-8245 (P. Prinetto)

The first approach introduces patches that can be applied directly in the physical implementation (hardware or software), trying to limit information leakage straight into the component where it occurs. Masking [6, 7], hiding [8], switched capacitances or digital signature attenuation [9, 10] are just some of the most common examples in this category. In general, the following scenario can be observed: the greater the security provided by a particular solution, the greater its impact on the device performance. Implementation-level countermeasures often incur a significant overhead, with studies suggesting a twofold or even threefold increase in both performance cost and code size [11, 12, 13]. Such penalties are hardly affordable for standard embedded cryptographic systems, which operate under stringent memory and power consumption constraints.

The second approach, of far more recent development, explores possible tweaks in the encryption schemes at mathematical and algorithmic level. The aim is to identify and eliminate those features of current algorithms that may favor the proliferation of leakage within the implementation layer. Among the advantages that make these countermeasures particularly interesting, there is the possible absence of performance impacts on the underlying device, in terms of execution time, area and energy consumption. Most of the recent proposals in this domain are focusing on the design of novel substitution boxes (S-Box) with improved resistance to side-channel analysis. The ultimate goal is to modify widely-known standard symmetric encryption schemes, such as AES [14], by replacing their original S-Box with structures with better SCA-resistance properties.

To the best of our knowledge, while research on the theoretical aspects of these new countermeasures is progressing rapidly, a similar evolution in finding empirical benchmarks and real-world results has not been observed. Many of the latest proposals have never left the simulation domain. Therefore, a series of extensive empirical tests are needed to prove the theoretical claims about side-channel resistance, comparing their results with those obtained by testing an original and unprotected software implementation of AES-128.

The present paper proposes such an evaluation on a selected set of S-Box designs. Conducted experiments leverage the use of the *ChipWhisperer™* platform [15]. An unprotected software implementation of AES-128 has been targeted, implemented on the XMEGA 8-bit microcontroller included in the *ChipWhisperer™* -Lite kit.

The remainder of the paper is organized as follows. Section 2 provides a general background on power side-channel attacks and major countermeasures; Section 3 introduces some basic metrics to evaluate the resistance against cryptanalysis and side-channel analysis; Section 4 analyses current state of the art about non-algebraic S-Boxes proposals; Section 5 outlines the subjects and the methodology of our study; Section 6 reports experimental results obtained; Section 7 concludes the paper.

2. Background

2.1. Power Analysis

Power analysis capitalizes on the observation that the power consumption of an integrated circuit is subject to variation in accordance with the toggling activity of its individual transistors [16]. The exploitation of such behavior requires the attacker to tamper with the circuit board where the device under attack is located. The cryptographic activity of the device is recorded by collecting a multitude of power traces. In such phase, also called *online analysis*, the malicious actor creates as many plaintexts as necessary (assuming an encryption operation) to be encrypted by the device. For each sent plaintext P_i, a power trace T_i related to that specific execution is recorded. At the end of the process, the actor is in possession of N $[P_i, T_i]$ plaintext – trace pairs, where N corresponds to the number of encryption operations completed by the device. All the data obtained from this online phase are then analyzed during the *offline analysis*.

This second stage of the attack targets the recovery of the secret key processed by the device during execution. To this end, several techniques have been developed, including Differential Power Analysis (DPA) and Correlation Power Analysis (CPA) [17, 18], with more recent approaches leveraging deep-learning-based methods [19, 20].

The objective of both DPA and CPA, which are statistical attacks in nature, is to establish a correlation between the power traces that have been previously collected during the online phase and the intermediate data that is anticipated to be calculated within the encryption/decryption operations performed by the device. The generation of such data, contingent upon the targeted operation within the encryption algorithm (under the assumption of an AES-128 software implementation), constitutes the direct product of an interaction among one byte of the plaintext and one byte of the key. A *divide-and-conquer* approach is used, which consists of recovering the entire secret by sequentially cracking one byte at a time. At each iteration of the attack, a feasible exhaustive search is performed, leveraging the side-channel information acquired. For each of the 256 possible subkey candidates, a correlation score is computed, indicating *how well* a hypothesis correlates with the data embedded in the power traces, *de facto* showing *how close* this hypothesis is to the actual subkey. The attacker sets up a ranking of the best performing candidates by listing all the possible key bytes from the most likely to the least likely. The ability to isolate the correct subkey candidates depends on the number of $[P_i, T_i]$ pairs collected during the online phase. The higher the leakage data available, the higher the probability of the attack to retrieve the secret key. The distinguishing factor between DPA and CPA lies in the methodology employed to establish a correlation between the intermediate data and the acquired power traces. The algorithm used to calculate the score assigned to each subkey candidate changes accordingly.

The *divide-and-conquer* approach just mentioned is what makes power analysis a shortcut attack [21]. In fact, instead of testing 2^{128} possible keys (considering AES-128), the complexity of the problem is reduced to testing all the 256 possible values (from `0x00` to `0xFF`) of each subkey for 16 total subkeys, reducing the exploration space to only 2^{12} possible combinations. This reasoning also demonstrates how power analysis makes insecure (and therefore useless) even possible upgrades to longer variants of AES, such as AES-256, whose complexity is reduced from 2^{256} to 2^{13} possible combinations.

2.2. Device-level Countermeasures

As for possible countermeasures, a crucial observation has to be made: side-channel (power) attacks cannot be entirely prevented due to the power-consuming nature that characterizes any operating VLSI device [22, 23]. As a consequence, any statistical attack like DPA will eventually succeed, with an ideally infinite number of power traces acquired. Although not completely removable, leakage from a microprocessor can be reduced, and side-channel analysis can be made so complex and costly as to make it unattractive, leading the attacker to spend more resources than what can be gained from a successful attack. In general, countermeasures against statistical attacks adhere to 2 main principles: (i) reduce as much as possible the level of information leaked, and (ii) introduce as much random noise as possible to mask and "scramble" the remaining portion of leakage.

As summarized in [16], reducing the leakage can be achieved by using *multi-bit data representations* and *balanced transitions*, so as to complement and nullify the effect of the inevitable transitions produced by the normal activity of the processor. However, adding transitions on pre-existent hardware increases the overall power consumption of the device. Moreover, in the worst case, the adoption of multi-bit data representations may require application designers to switch to a larger and more expensive device.

Similarly, masking-based countermeasures can be adopted to make the inevitable leaking information not related to the secret data internally treated by the device. These solutions enable the physical implementation to resist side-channel analysis by periodically changing the internal representation of the secret parameters [16]. The attacker is required to considerably increase the number of traces to be collected and to rely to *high-order* DPA techniques to successfully complete an attack [24]. This class of countermeasures guarantee the highest security among all those still implemented at the device level. However, the impact on both performance and memory requirements is non-negligible, and can prevent their adoption, especially in the case of IoT devices and other embedded systems like smartcards and hardware tokens [25, 7, 26, 27].

2.3. Cryptographic-level Countermeasures

As introduced in [16], countermeasures against side-channel analysis can also be implemented at the protocol level, the same in which cryptographic algorithms are specified. Such approach consists in device-independent solutions, therefore providing efficient countermeasures even in the case of low-cost microcontrollers.

A novel class of cryptographic-level countermeasures is targeting the design of substitution boxes (S-Boxes). These structures are a critical component of block ciphers specifications such as AES and DES. They are the main source of *confusion* and *non-linearity* and already exhibit resistance against classical cryptanalysis like linear [28] and differential [29] cryptanalysis. Therefore, it is reasonable to include in their design additional criteria indicating resistance to side-channel attacks. The choice of targeting these structures is also dictated by the flexibility they provide. In fact, S-Boxes are just one of the many primary operations that compose block cipher algorithms. As long as their cryptographic properties are still guaranteed, swapping one structure for another (i.e., simply changing the numeric content of their software Look-Up-Tables (LUTs)) can allow improving the quality of the corresponding algorithm without having any impact on the underlying software implementation and physical device, as anticipated in [30].

3. Related Metrics

The present section is intended to give the reader some fundamental definitions about metrics for the evaluation of substitution boxes and side-channel attacks performance.

3.1. Theoretical Metrics for Classical Cryptanalysis

To define properties and qualities of an S-Box design, many parameters have been defined over time. With respect to classical cryptanalysis, three main properties are used in literature:

1. **Bijection**: an S-Box $S(n, m)$ can be viewed as a look-up-table (LUT) that maps an n-bit value x to an m-bit output value y. When $n = m$ and the S-Box outputs all 2^m possible values exactly once, the mapping constitutes a permutation of the input space and is therefore bijective. A bijective S-Box admits a well-defined inverse mapping, allowing each output value to be uniquely mapped back to its corresponding input – an essential property for decryption;

2. **Non-linearity**: the overall non-linearity property of an S-Box structure is determined by the non-linearity of the Boolean functions that defines it. Since linear transformations are cryptographically weak, the higher the nonlinearity score, the better the S-Box resilience to linear cryptanalysis. AES S-Box has a non-linearity value of 112 [30];

3. **Input/Output XOR Distribution**: this property indicates resistance to differential cryptanalysis. To compute it for the entire substitution structure, a Differential Distribution Table (DDT) has to be built. If an S-Box has a low and equiprobable input/output XOR distribution, the structure is considered immune to a differential attack. In the case of AES, the maximum value of differential propagation probability is of $4/256$ (or 2^{-6}) [30].

3.2. Theoretical Metrics for Side-Channel Analysis

As for classical cryptanalysis, new properties have been defined over the years to address and formally define the resistance of a software/hardware implementation against side-channel analysis:

1. **Transparency Order**: defined in [11], the Transparency Order (TO) can quantify the resistance of an S-Box to DPA attacks: the smaller the TO of an S-Box, the higher its resistance. As demonstrated by the author, the higher the non-linearity of a Boolean function, the lower its

resistance against DPA. In [31], authors prove that highly non-linear Boolean functions like the ones that compose the S-Box of AES have "very bad transparency orders";

2. **Improved Transparency Order**: defined in [32], the lower the value of the Improved Transparency Order (ITO), the lower the success probability to extract the secret key based on leakages associated to the S-Box;

3. **Confusion Coefficient**: defined in [33], the Confusion Coefficient (CC) measures the discrepancy between the hypothesis of an intermediate state using the correct (secret) key and any wrong key assumption [34]. In [35, 6], authors claim that the higher the CC metric, the higher the resistance of the S-Box against side-channel attacks.

Fairly recent publications observed an opposition among properties associated to classical cryptanalysis and the feasibility of side-channel attacks. For instance, Guilley *et al.* proved that the more an S-Box is protected against linear cryptanalysis, the more vulnerable it is to side-channel attacks such as DPA [36]. In [34], it is observed that having an S-Box more resilient against SCA will make it potentially more vulnerable to classical cryptanalysis, without however excluding possible trade-offs, similarly to what already proved in [11].

As a consequence of these findings, it is evident that side-channel resistance mandates the reach for better trade-offs with classical cryptanalysis. The search for different design methodologies may result in a wider exploration of the solution space. Moreover, such methodologies could potentially lead to S-Boxes with slightly worse resistance to classical cryptanalysis, but with a drastically improved resistance to side-channel attacks.

3.3. Empirical Metrics for Power Attacks Evaluation

The following parameters have been defined to summarize the achievements and qualities of side-channel attacks such as correlation power analysis.

1. **Success Rate**: defined in [37], the Success Rate (SR) is generally computed empirically from the measurements related to a particular device [38]. In the case of an algorithm like CPA (i.e., a side-channel key recovery adversary operating in a *divide-and-conquer* fashion), the Success Rate of order 1 refers to the probability of having the correct key sorted first by the algorithm. A Success Rate of order n implies that the adversary still has a maximum of n key candidates to test after the attack [37];

2. **(Partial) Guessing Entropy**: defined in [39], the Partial Guessing Entropy (PGE) metric is extremely similar to the Success Rate and can be used, in most cases, as an alternative to it [37]. As an example, a PGE of 0 indicates the correct subkey is known, whereas a PGE of 2 indicates that 2 wrong key candidates were ranked incorrectly higher than the candidate representing the correct key guess [40];

3. **Correlation Plot**: this kind of graph allows to observe how much the implementation under attack hinders (or not) the separation of correlation traces related to correct keys from traces related to wrong candidates;

4. **Minimum Traces to Disclosure**: an empirical metric derived from graphic reading a correlation plot. The Minimum Traces to Disclosure (MTD) metric indicates the point (expressed in terms of number of traces) in which the average correlation trace of the key isolates itself from the correlation traces of the wrong key candidates.

4. Related Work

Many different S-Box design methodologies have been proposed over the years. The most known fall into the following categories: *algebraic methods*, *pseudo-random generation*, *heuristic methods*, and, more

recently, *dynamic structures*. Algebraic methods are able to produce structures with excellent properties against linear and differential cryptanalysis. Design methodologies based on finite field inversion led to the proposal of many widely-known S-Boxes [41] based upon Nyberg's structures [42], such as the original Rijndael AES S-Box. On the other hand, pseudo-random methods leverage the injection of random values into the substitution structure. However, as claimed in [43] and [44], most of the designs obtained with this methodology lack of relevant cryptographic properties.

4.1. Heuristic Methods

In many practical settings, heuristic algorithms constitute the only feasible strategy for tackling problems of NP-hard complexity. A representative example arises in the design of substitution boxes (S-Boxes). For an 8×8 bijective S-Box – such as that employed in AES – the total number of possible structures is extraordinarily large. Specifically, an S-Box corresponds to a permutation of the 256 possible byte values, yielding $256!$ distinct bijective tables. This combinatorial explosion makes exhaustive exploration entirely intractable, thereby motivating the use of heuristic and evolutionary methods to navigate such vast design spaces.

In [43], the authors adopted a novel hybrid heuristic method based on *Leaders and Followers* and *hill-climbing* algorithms obtaining "excellent results for confusion coefficient variance". In addition, authors leverage the cost function proposed in [44] to improve the nonlinearity score of their findings, obtaining values always greater or equal than 100. The structures proved to have better theoretical side-channel resistance than many real-life S-Boxes used in algorithms like AES and PICARO [7], while satisfying nonlinearity and differential properties. Three of the most promising structures proposed in [43] have been selected and analyzed, the discussion can be found in the following sections.

4.2. Chaos-based Methods

Chaos-based approaches introduce a new rationale for S-Box design by exploiting chaotic systems as sources of randomness, including chaotic maps and hyper-chaotic systems [45, 46, 47, 48, 49]. Achieving S-Boxes with strong cryptographic properties generally requires the use of optimization algorithms capable of selecting the most suitable values from those produced by the chaotic source [50].

However, despite recent efforts focusing primarily on demonstrating resistance to classical cryptanalysis, chaos-based S-Box proposals rarely claim robustness against side-channel analysis [51]. As previously noted, due to the inherent tension between these two domains, existing works appear, to the best of our knowledge, to strive for a compromise between resistance to cryptanalysis and resistance to side-channel attacks. For instance, in [51], the author shows that current chaos-based S-Boxes still exhibit weaker cryptographic properties than the AES S-Box designed by Daemen and Rijmen, achieving a nonlinearity of 106 (compared to 112 for AES [30]). Likewise, while the maximum differential propagation probability for the AES S-Box is $4/256$, the best value attained by existing chaos-based structures is $10/256$ [51].

4.3. Dynamic and Key-Dependent S-Box Structures

A fundamentally different strategy for improving resistance to both cryptanalysis and side-channel attacks (SCA) is the adoption of dynamic, key-dependent S-Box architectures. Unlike the standard AES S-Box, which relies on a fixed lookup-table (LUT) access pattern, dynamic S-Box constructions overcome this rigidity by generating a new substitution function for each session or key [52]. Their primary SCA countermeasure is randomization: by continually altering the substitution function based on a dynamic perturbation or seed, these designs impede profiled side-channel attacks. Adversaries attempting to construct fixed statistical templates or Deep Learning models for a single, known S-Box function are inherently disadvantaged, as the leakage characteristics change with every dynamic instantiation [52, 53].

However, this strategy introduces an important engineering challenge. The security burden shifts from protecting a static LUT access pattern to securing the dynamic S-Box generation circuitry itself,

which inevitably leaks information. Consequently, side-channel analysis may still reveal either the generation parameters or the secret key, thereby necessitating strong physical (implementation-level) protections for the S-Box generation unit.

4.4. Theoretical Side-Channel Evaluation on S-Boxes

In the past years, many publications have evaluated the *theoretical* side-channel resistance of S-Boxes. To the best of our knowledge, most consisted in CAD simulations leveraging synthesis tools, without implementing empirical attacks on real devices. In [54], simulation-based CPA attacks were conducted on multiple AES implementations involving different S-Box structures. The authors gave a first glance on using S-Boxes as countermeasures against power analysis, without recurring to device-level solutions and thus without making use of any added logic. In [36], Guilley *et al.* investigate the theoretical resistance of the AES S-Box against differential power analysis. In [55], the authors analyzed several ciphers, highlighting the fact that, in some cases, metrics like the confusion coefficient and the transparency order lack accuracy when evaluating the resistance of S-Boxes against side-channel analysis. The group finally suggested the need for further work to affine such metrics. In [56], the authors empirically tested 3 different S-Boxes: the first being the standard AES substitution structure while the remaining consisting in two chaos-based S-Boxes taken from Özkaynak's research [51]. The chaos-based structures denoted an improved resistance to side-channel attacks. However, as the number of synchronously-captured traces increases above 30, the attack can be considered successful no matter the S-Box used. This final result implies the need of further investigation and characterization of novel proposed structures [56].

5. Our Contribution

This paper presents an empirical evaluation of several recent substitution boxes that have been proposed in the literature, focusing on those that claim to offer side-channel resistance. With the exception of the original AES S-Box, all of the structures under evaluation have been designed using non-algebraic methodologies, such as heuristic and chaos-based ones. The aim of this study is to test the claims supporting the proposed structures, verifying the overall efficacy of theoretical metrics like the confusion coefficient. The final goal is to prove the adequacy of these structures in real-world attack scenarios.

5.1. Selected S-Boxes

The attack is conducted on six different S-Box structures, including the original AES S-Box, which is used as a reference. Another structure, proposed by Hussain et al. [57] and referred to in the original work as 'S-Box-6', exhibits cryptographic properties very similar to the AES S-Box, despite being constructed through a completely different process. Including this S-Box in our evaluation allows us to further characterize the behavior of "classic" S-Boxes that lack explicit countermeasures against side-channel attacks. Finally, the remaining four structures were selected among the latest publications – claiming some sort of side-channel resistance – for both chaos-based and heuristic-based methodologies. Three of these come from the work presented by Freyre *et al.* in [43]. The last S-Box taken into consideration has been initially proposed by Özkaynak in [51] and tested in [56]. The structure, in the original publication, is referred to as "Proposal 2". Unfortunately, despite claiming some form of side-channel resistance, no value of the confusion coefficient was reported by the authors.

A summary of the cryptographic properties of the six structures under attack is presented in Table 1.

5.2. Capture Instrumentation

The experiments in this study were carried out using the *ChipWhisperer™* platform [15]. The *ChipWhisperer™* -Lite board integrates all essential instrumentation – including trigger, oscilloscope, probe, and the target board – into a single device. Its acquisition hardware supports a capture technique not

Table 1
Properties of S-Box Structures Under Test.

S-Box	Non-linearity	Diff. Unif.	CC
AES [30] [35]	112	4/256	0.111
Freyre #1 [43]	100	8/256	4.500
Freyre #2 [43]	100	8/256	4.492
Freyre #3 [43]	102	8/256	1.934
Hussain [57]	112	4/256	n.d.
Özkaynak [51]	106	10/256	n.d.

typically found in commercial oscilloscopes, known as synchronous sampling. This technique maintains a fixed phase relationship between the clocks of the capture board and the target device, thereby relaxing the sampling-rate requirements compared to asynchronous sampling. For example, in [58], the authors present an attack against a SASEBO-GII board, claiming that a synchronous sampling at 96 MS/s produces similar results obtained by an asynchronous sampling done at 2 GS/s. Such reductions in required sampling frequency stem from the elimination of jitter and phase variations between the two clocks: conditions that cannot be guaranteed when the clocks are independent. By reducing jitter, synchronous capture substantially lowers noise in the acquired traces, increases SNR, and consequently decreases the number of traces needed to mount DPA/CPA attacks.

For all the necessary acquisitions required for this study, a synchronous sampling technique has been used, with a sampling frequency 4 times higher than the clock frequency of the device under attack (29.5 MHz and 7.4 MHz, respectively). The first sampling point is acquired in conjunction with the rising edge of the device clock. The ADC gain has been set to 24.8 dB. Each trace is composed by 5000 sampling points, with no decimation involved.

5.3. Attack procedure

The procedure used for this study consists of a series of non-profiled Correlation Power Analysis (CPA) attacks performed on each of the six selected structures. As previously mentioned, software implementations are the type of deployment that benefits most from this kind of countermeasures. Since S-Boxes are coded as Look-Up Tables (LUT), replacing an S-Box with a new one only requires changing the numerical values in memory, without making any further change to the original algorithm, nor affecting the performance of the device.

The reference device under attack (DUA) is a 8-bit XMEGA AVR microcontroller [59]: a perfect representative for low-power embedded systems. This microcontroller is bundled together with the *ChipWhisperer*™ -Lite kit, on a separate PCB, which can be connected to the probes of the main capture board. The source code of the unprotected AES-128 software implementation, `avrcryptolib`, can be found in [60]. Both the online and the offline analysis are performed by exploiting a `python3` script able to send commands and receive acquisition data from the *ChipWhisperer*™ board, leveraging its open-source software APIs.

The online analysis launched by the script is divided into multiple batches. Each batch is defined by a certain number of traces to be acquired: the first batch acquires 80 traces, the second 100, the third 200, the fourth 300 and the fifth 500. In each batch, all available S-Boxes are tested in sequence. For each S-Box, 20 acquisitions are repeated: each acquisition captures the number of traces that characterizes the current batch. The traces captured from one acquisition to another are all different: the input plaintexts sent to the microcontroller change randomly when a new trace is captured. Summarizing in an example: the first batch acquires, for each S-Box, a total of $20 \cdot 80 = 1600$ different traces. Considering all the S-Boxes to be tested, the first batch captures $6 \cdot 20 \cdot 80 = 9600$ total traces. When switching from one S-Box to the other, the script modifies the source code of the software implementation by swapping the current structure with the following one to be tested. The C code is compiled and then flashed onto the microcontroller. The main *ChipWhisperer*™ board is instructed by the script and acts as a middleman

able to flash the target device. It is important to remember that the secret key used by the device for each encryption operation is kept constant for the entire online analysis.

Once the online analysis completed, CPA attacks are performed during the subsequent offline analysis.

Similar to what was done during the data collection phase, the attacks are divided into multiple batches, each batch referring to a specific number of traces. In each of them, S-Boxes are attacked sequentially. For each structure, 20 different CPA attacks are performed. Given the independence among the data collected for each of the 20 iterations, the script is able to parallelize the computations by adapting to the number of available CPU cores. It is important to note that, *in all cases*, a single CPA attack would have been sufficient for a malicious actor to reveal the secret key. The 20 iterations are only needed to collect multiple scenarios, so as to calculate a statistically representative average. This allows to mediate the presence of outliers that would not have represented the most common behavior a real attacker may face.

The data obtained from all 20 attacks are then used to calculate the average performance of an attack made on that precise S-Box structure. On the data related to this result, three graphs are created. The first one represents the average trend (and standard deviation) of the Partial Guessing Entropy (PGE) of each of the 16 key bytes as the number of traces (and the leakage data available to the algorithm) increases. This type of graph allows to understand how many traces are required by the attack to reveal the secret key. We expect that an implementation with a SCA-resistant S-Box will require more traces than an implementation having a "classical" unprotected S-Box. The second graph represents the average correlation (and standard deviation) scores associated to each correct key byte and to all wrong key candidates. The traces related to the wrong guesses are decimated with a factor of 10 to facilitate the interpretation of the graph. Finally, the third graph, extremely similar to the first one, allows us to retrieve the Minimum Traces to Disclosure (MTD) metric.

A visual examination of the graphs obtained permits the estimation of the minimum number of traces necessary for an attack to reveal the secret key. The predominant approach employed in the extant literature derives the Minimum Traces to Disclosure (MTD) value from the mean correlation graph. However, it was observed that the application of this method yields results that are highly approximate, encompassing the entire process from the creation of the graph to the visual identification of the MTD point. To obtain less approximate results, we defined four new approaches, ranging from the most optimistic to the most conservative one (from an attacker's point of view). The reliability of these estimates is supported by the observations of the graph representing the average trend (and standard deviations) of the PGE metric:

(A) $PGE_{avg} \leq 4$ [mean]: the attack is considered successful when all the average traces referring to each of the 16 bytes of the secret key crosses the threshold $PGE \leq 4$. From that point on, all 16 bytes that make up the key are at most 4 positions away from the top of the leaderboard, possibly preceded by incorrect candidate bytes;

(B) $PGE_{avg} \leq 4$ [std_dev]: similar to the first case, but making sure that also the standard deviations related to the trends of each of the 16 correct subkeys have crossed the threshold. We expect this method to return slightly higher values than the first one;

(C) $PGE_{avg} < 1$ [mean]: the exact same scenario of the first case is repeated but lowering the threshold strictly below $PGE = 1$. By doing so, all the 16 correct subkeys are now on top of their respective rankings, meaning the attack has found the secret key in its entireness;

(D) $PGE_{avg} < 1$ [std_dev]: similarly to the third case, but making sure that also the standard deviations related to the trends of each of the 16 correct subkeys have crossed the threshold $PGE < 1$. We expect this method will yield the highest values in terms of number of traces, thereby becoming the most conservative technique among those considered.

The choice of a specific PGE threshold is inherently arbitrary, as it reflects a trade-off between the complexity of the side-channel attack and the reliability of the resulting key recovery. A higher PGE

threshold forces the attacker to eliminate the remaining incorrect candidates for each subkey through a final, feasible brute-force search. Conversely, a lower PGE threshold leaves fewer "uncertain" subkeys to be evaluated. The ideal scenario for an attacker is $PGE = 0$, which indicates that all subkeys were correctly identified in a single attempt, eliminating the need for any additional brute-force search and minimizing the time and computational resources required for the entire operation.

Finally, it is worth noting that choosing PGE values very close to zero may be counterproductive for this type of analysis. Certain countermeasures may prevent complete key recovery except asymptotically, requiring an impractically large number of traces. In such situations, it is reasonable to assume that an attacker would instead stop after collecting a smaller set of traces and complete the remaining search via brute force, thereby choosing the strategy that minimizes the overall effort.

6. Experimental Results

This section reports and comments on the results obtained from the previously described offline analysis phase.

Table 2 summarizes the final results obtained from the observations of the various graphs created by the script. For each S-Box tested, and for each of the modalities reported in the previous Section (A) (B) (C) and (D), intervals representing the number of traces needed to disclose the key are reported. The annotation of these events was done manually by observing graphs such as in Figures 3 and 4. For each plot produced, the 4 points of interest were noted, reporting the closest interval of traces in which each of them occurred. The choice was to report a trace range rather than a punctual graphical estimation in order to avoid as much as possible individual bias that could have altered the perception of the data obtained.

Once all the intervals collected (for each S-Box and for each of the available batches), data for Table 2 were computed. To do so, the intersection of all the intervals was performed for each cell in the Table, given the corresponding batch and S-Box. It is expected that, at each intersection, the interval relative to the recorded event increasingly narrows, allowing to obtain – at the end of the process – an average representative range for all the measurements acquired. In all those cases where the intersection between two ranges does not exist (such as the case $\{20..40\}$ and $\{60..80\}$, for example), a conservative approach was preferred by taking their union ($\{20..80\}$), rather than arbitrarily accepting one of the two ranges and erroneously forcing a personal bias into our analysis.

Table 2 highlights the similarities among the original AES S-Box, the one proposed by Hussain and the one designed by Özkaynak. As expected, the original AES software implementation proved to be weak against CPA, revealing nearly the whole key after just 30 different traces. All 4 structures share comparable trace ranges, for all 4 events, with Özkaynak's one able to stretch a little more the number of traces necessary for case (D). The analogies between these three structures can also be appreciated in their corresponding correlation graphs. Only the correlation graph for the original AES S-Box is shown in Figure 1, since the corresponding plots for the Hussain and Özkaynak structures exhibit virtually identical behavior.

The results discussed up to now allow us to draw two initial conclusions: (i) from the point of view of resistance to SCA, Hussain's S-Box – as expected – obtains results that are virtually identical to those obtained from the S-Box of AES; (ii) Özkaynak's S-Box showed no improvement over AES S-Box, highlighting the presence of insufficient protection provided by the current chaos-based structures, contrary to the authors' initial claims in [51]. We consider the previously mentioned results achieved in (D) as absolutely insufficient to prevent a real-world attack.

Turning to the observation of the results obtained from the structures proposed by Freyre in [43], it can be noticed both from the Table 2 and the graph in Figure 2, that two of the three S-Boxes tested (Freyre #1 and Freyre #2) report a more marked resistance to side-channel analysis. Both designs increase the number of traces necessary to disclose the complete key up to 100, in the best case scenario. The improvement, with respect to the original unprotected AES implementation, goes from 1.25x to 2x. A way smaller improvement is obtained by the third S-Box, Freyre #3, that mimics in most

cases the performance of the structure found in the AES specifications. The difference among the structures proposed by Freyre can be explained by observing their respective values of the Confusion Coefficient (CC). As mentioned in Section 3, the higher the value of the confusion coefficient, the better the resistance against side-channel analysis attacks. The first and the second S-Box proposed by the author have such value approaching 4.5, whereas Freyre #3 only reaches 1.9. The correlation plots, as seen in Figures 1 and 2, show how tangled the correlations of the correct keys are with the ones related to the wrong key candidates: the higher the entanglement the better the S-Box is able to thwart SCA attacks. Both Freyre #1 and Freyre #2 are able to keep a belt of wrong key candidates near the belt related to the correct key guesses: the closer, the better. Such behavior does not recur in the case of Freyre #3 (see Figure 5), which is able to divide the wrong key candidates traces in only two main belts, far from the correct key guesses, hence the reduced resistance to attacks.

Table 2
Integer intervals representing the number of traces to disclosure.

S-Box	Case (A)	Case (B)	Case (C)	Case (D)	MTD
AES [30]	{30..32}	{32..40}	{32..40}	{40..48}	{8..16}
Freyre #1 [43]	{30..32}	{48..56}	{48..56}	{72..100}	{8..16}
Freyre #2 [43]	{32..40}	{50..70}	{60..80}	{60..100}	{8..16}
Freyre #3 [43]	{30..32}	{32..40}	{32..40}	{32..50}	{8..16}
Hussain [57]	{30..32}	{32..40}	{32..40}	{40..48}	{8..16}
Özkaynak [51]	{30..32}	{32..40}	{32..40}	{32..50}	{8..16}

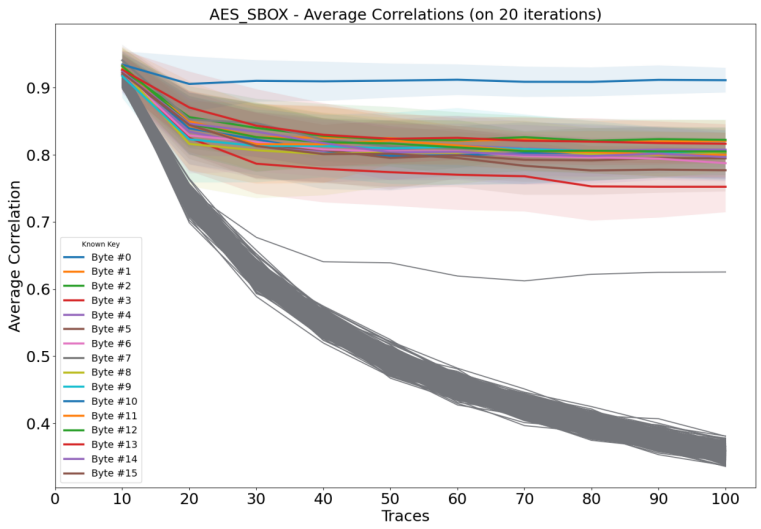

Figure 1: Average correlation traces (w/ standard deviation) for the original AES S-Box [100 traces].

7. Conclusions

In this paper, an empirical evaluation on a selected set of novel S-Box structures for software implementations of the AES-128 algorithm has been proposed. The goal is to test the capabilities of such structures in preventing physical attacks, such as side-channel power analysis, and to verify the possible adoptability of these cryptographic-level countermeasures in real-world usage scenarios.

Six different structures have been tested on an unprotected AES-128 software implementation, running on an 8-bit microcontroller. The extensive trace acquisition phase involved a synchronous capture leveraging the *ChipWhisperer*™ platform.

The obtained results indicate that some of the tested structures exhibit a degree of resistance to

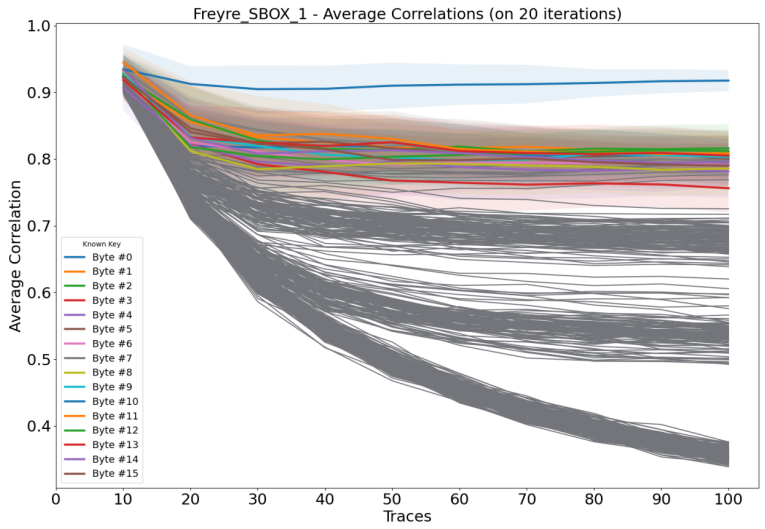

Figure 2: Average correlation traces (w/ standard deviation) for Freyre #1 S-Box [100 traces].

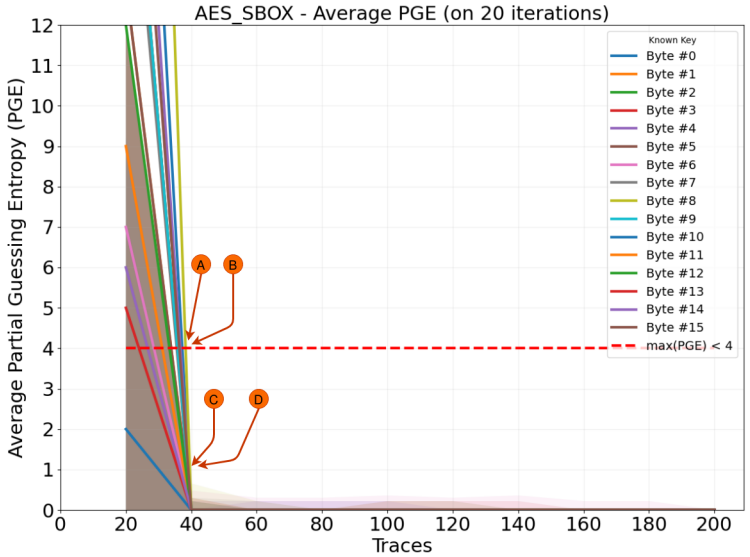

Figure 3: Average PGE traces (w/ standard deviation) for the original AES S-Box [200 traces]. The graph is annotated with each of the four events defined in 5.3.

side-channel attacks, albeit very limited. This effect was observed in the S-Boxes with the highest confusion coefficient (CC) values, highlighting and confirming the practical relevance of this metric. In contrast, standard structures designed solely to withstand classical cryptanalysis displayed poor resistance to physical attacks, as expected. This behavior is characteristic of both AES S-Box and Hussain's S-Box [57]. Additionally, these structures confirmed empirically, on a real microcontroller, what already claimed by many previous publications: properties like non-linearity and differential uniformity provide no protection against SCA.

Other chaotic constructions, such as Özkaynak's S-Box [51], yielded results nearly identical to those of the unprotected AES implementation. Despite claims of enhanced security, this structure demonstrated no significant resistance to side-channel attacks.

The heuristic-based structures proposed by Freyre in [43] reached the highest level of side-channel resistance among all the tested S-Boxes. On average, the number of traces required to successfully recover the secret key increased by up to a factor of two. Moreover, some interesting observations relating the impact of the confusion coefficient can be made. As expected, the higher the value of the confusion coefficient, the higher the number of traces required to achieve the attack. Moreover, as

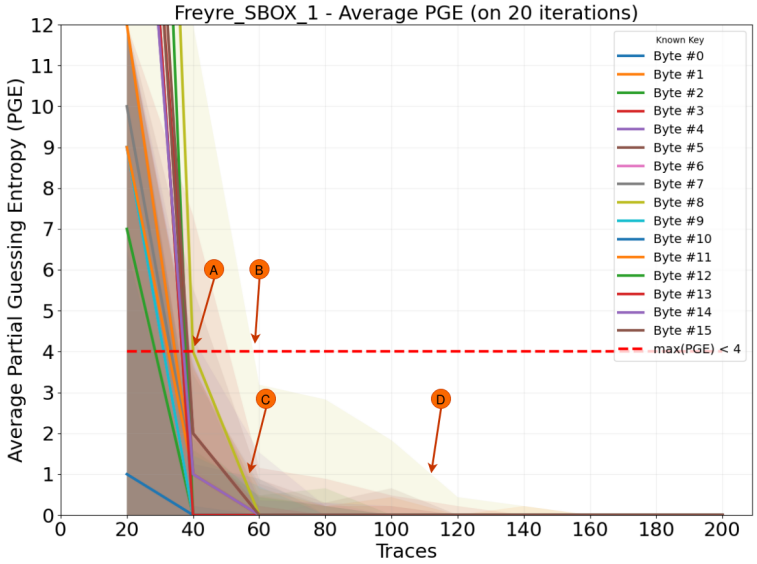

Figure 4: Average PGE traces (w/ standard deviation) for Freyre #1 S-Box [200 traces]. The graph is annotated with each of the four events defined in 5.3.

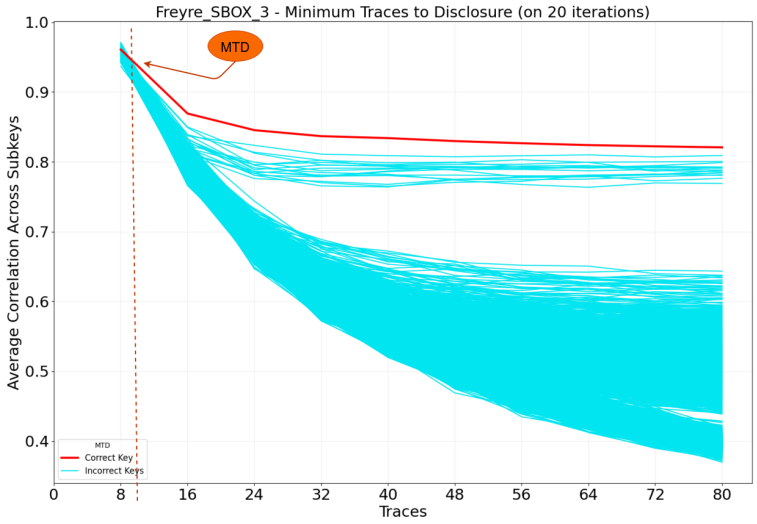

Figure 5: Observation of the Minimum Traces to Disclosure metric for Freyre's S-Box #3.

visible in Figures 2 and 5, a high CC value implies an increased "entanglement" among the various correlation traces. These plots perfectly picture the definition given in [34]. As mentioned in the previous Section, both *Freyre #1* and *Freyre #2* acted similarly, with a CC of 4.5 and 4.492, respectively. Conversely, the *Freyre #3* structure, with a CC of 1.934, showed some sign of correlation confusion in the corresponding graph 5, but not significant enough to increase considerably the number of traces required to achieve an attack.

Despite the apparently promising results of some of the tested S-Boxes, especially the ones designed by Freyre *et al.* in [43], the protection against power side-channel analysis provided by these new structures is insufficient and only slightly better than the one provided by a standard unprotected AES implementation. It is important to point out that the acquisition of 80 traces, with the use of *ChipWhisperer™* , takes place in about 2 seconds (about 46 traces are captured per second). Then, offline analysis takes about 25 seconds on average on a laptop with an *Intel™ i5 Broadwell* dual-core processor. Therefore, the combination of online and offline analysis can take less than 5 minutes in such a best case scenario for an attacker (with prior knowledge of the device, considering synchronous sampling possible, not taking into account the time needed to tamper with the device). It is also

extremely important to remind that, with such a setup, the secret key can be entirely recovered with just one iteration of the attack. An attack carried out by doubling the captured traces (e.g., 200 traces to be conservative) will require less than 5 seconds for acquisition and an average of 35 seconds to process them.

Albeit being extremely different in logistics and overall attack complexity, real-life scenarios still require countermeasures that can significantly increase the number of traces required to break a device, far over a 2x factor. Despite the presence of a clear improvement, further studies are needed to refine the properties of the proposed structures and make them suitable to deal with real attack scenarios. The results obtained emphasize, once again, the importance of empirical tests on real existing devices, as only such kind of analysis can give clear insights into the real-world feasibility of academic solutions.

The observations gained with this study can be taken as a starting point for future work, aiming to define a value for the confusion coefficient beyond which a given S-Box is guaranteed to empirically provide adequate resistance to side channel analysis. Such value may eventually provide an extremely valuable feedback to the cryptographic community. The analysis described in this paper can also be extended to consider additional theoretical metrics indicating side-channel resistance, such as the (improved) transparency order.

Declaration on Generative AI

During the preparation of this work, the authors used DeepL in order to: Text translation and to improve the writing style. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] P. Kocher, J. Horn, A. Fogh, D. Genkin, D. Gruss, W. Haas, M. Hamburg, M. Lipp, S. Mangard, T. Prescher, et al., Spectre attacks: Exploiting speculative execution, in: 2019 IEEE Symposium on Security and Privacy (SP), IEEE, 2019, pp. 1–19. doi:10.1145/3399742.

[2] M. Lipp, M. Schwarz, D. Gruss, T. Prescher, W. Haas, A. Fogh, J. Horn, S. Mangard, P. Kocher, D. Genkin, et al., Meltdown: Reading kernel memory from user space, in: 27th USENIX Security Symposium (USENIX Security 18), 2018, pp. 973–990. doi:10.1145/3357033.

[3] Y. Li, M. Chen, J. Wang, Introduction to side-channel attacks and fault attacks, in: 2016 Asia-Pacific International Symposium on Electromagnetic Compatibility (APEMC), volume 01, 2016, pp. 573–575. doi:10.1109/APEMC.2016.7522801.

[4] R. Baldoni, R. D. Nicola, P. Prinetto, The future of Cybersecurity in Italy: strategic focus areas, Consorzio Interuniversitario Nazionale per l'Informatica - CINI, 2018. ISBN: 9788894137330.

[5] P. Prinetto, G. Roascio, Hardware security, vulnerabilities, and attacks: A comprehensive taxonomy, in: ITASEC, 2020, pp. 177–189.

[6] K. Stoffelen, Intrinsic side-channel analysis resistance and efficient masking, Ph.D. thesis, Master's thesis, Radboud University, 2015.

[7] G. Piret, T. Roche, C. Carlet, PICARO–a block cipher allowing efficient higher-order Side-Channel resistance, in: International Conference on Applied Cryptography and Network Security, Springer, 2012, pp. 311–328. doi:10.1007/978-3-642-31284-7_19.

[8] T. Güneysu, A. Moradi, Generic side-channel countermeasures for reconfigurable devices, in: International workshop on cryptographic hardware and embedded systems, Springer, 2011, pp. 33–48. doi:10.1007/978-3-642-23951-9_3.

[9] A. Ghosh, D. Das, J. Danial, V. De, S. Ghosh, S. Sen, 36.2 An EM/Power SCA-Resilient AES-256 with Synthesizable Signature Attenuation Using Digital-Friendly Current Source and RO-Bleed-Based Integrated Local Feedback and Global Switched-Mode Control, in: 2021 IEEE International Solid-State Circuits Conference (ISSCC), volume 64, IEEE, 2021, pp. 499–501. doi:10.1109/ISSCC42613.2021.9365978.

[10] A. Singh, M. Kar, S. Mathew, A. Rajan, V. De, S. Mukhopadhyay, A 128b aes engine with higher resistance to power and electromagnetic side-channel attacks enabled by a security-aware integrated all-digital low-dropout regulator, in: 2019 IEEE International Solid-State Circuits Conference-(ISSCC), IEEE, 2019, pp. 404–406. doi:10.1109/ISSCC.2019.8662344.

[11] E. Prouff, DPA attacks and S-Boxes, in: International Workshop on Fast Software Encryption, Springer, 2005, pp. 424–441. doi:10.1007/11502760_29.

[12] K. Schramm, C. Paar, Higher Order Masking of the AES, in: Cryptographers' track at the RSA conference, Springer, 2006, pp. 208–225. doi:10.1007/11605805_14.

[13] J. W. Bos, M. Gourjon, J. Renes, T. Schneider, C. van Vredendaal, Masking kyber: first- and higher-order implementations, IACR Cryptol. ePrint Arch. 2021 (2021) 483. URL: https://api.semanticscholar.org/CorpusID:233329820. doi:10.46586/tches.v2021.i4.173-214.

[14] N. I. of Standards, T. (NIST), FIPS 197, Advanced Encryption Standard (AES), https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf, 2001. Accessed 22 April 2025.

[15] C. O'Flynn, A. Dewar, J.-P. Thibault, ChipWhisperer White Paper – NewAE, http://media.newae.com/appnotes/NAE0010_Whitepaper_CW305_AES_SCA_Attack.pdf, October 2020. Accessed 27 March 2025.

[16] P. Kocher, J. Jaffe, B. Jun, P. Rohatgi, Introduction to Differential Power Analysis, Journal of Cryptographic Engineering 1 (2011) 5–27. doi:10.1007/s13389-011-0006-y.

[17] P. Kocher, J. Jaffe, B. Jun, Differential Power Analysis, in: Annual international cryptology conference, Springer, 1999, pp. 388–397. doi:10.1007/3-540-48405-1_25.

[18] E. Brier, C. Clavier, F. Olivier, Correlation Power Analysis with a leakage model, in: International workshop on cryptographic hardware and embedded systems, Springer, 2004, pp. 16–29. doi:10.1007/978-3-540-28632-5_2.

[19] B. Timon, Non-profiled deep learning-based side-channel attacks, Cryptology ePrint Archive (2018). doi:10.13154/tches.v2019.i2.107-131.

[20] S. Karayalçin, M. Krcek, S. Picek, Sok: Deep learning-based side-channel analysis trends and challenges, Cryptology ePrint Archive (2025). URL: https://eprint.iacr.org/2025/1309.

[21] K. Sakiyama, Y. Sasaki, Y. Li, Security of Block Ciphers: from Algorithm Design to Hardware Implementation, John Wiley & Sons, 2016.

[22] C. Mead, L. Conway, Introduction to vlsi systems, 1980.

[23] J.-S. Coron, D. Naccache, P. Kocher, Statistics and secret leakage, ACM Transactions on Embedded Computing Systems (TECS) 3 (2004) 492–508. doi:10.1007/3-540-45472-1_12.

[24] A. A. Ding, L. Zhang, Y. Fei, P. Luo, A statistical model for higher order DPA on masked devices, in: International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2014, pp. 147–169. doi:10.1007/978-3-662-44709-3_9.

[25] S. Chari, C. S. Jutla, J. R. Rao, P. Rohatgi, Towards sound approaches to counteract power-analysis attacks, in: Annual International Cryptology Conference, Springer, 1999, pp. 398–412. doi:10.1007/3-540-48405-1_26.

[26] M. Mayhew, R. Muresan, An overview of hardware-level statistical power analysis attack countermeasures, Journal of Cryptographic Engineering 7 (2017) 213–244. doi:10.1007/s13389-016-0133-6.

[27] C. Liptak, S. Mal-Sarkar, S. A. Kumar, Power analysis side channel attacks and countermeasures for the internet of things, in: 2022 IEEE Physical Assurance and Inspection of Electronics (PAINE), IEEE, 2022, pp. 1–7. doi:10.1109/PAINE56030.2022.10014854.

[28] M. Matsui, Linear cryptanalysis method for DES cipher, in: Workshop on the Theory and Application of of Cryptographic Techniques, Springer, 1993, pp. 386–397.

[29] E. Biham, A. Shamir, Differential cryptanalysis of the full 16-round DES, in: Annual International Cryptology Conference, Springer, 1992, pp. 487–496. doi:10.1007/3-540-48071-4_34.

[30] J. Daemen, V. Rijmen, AES proposal: Rijndael (1999).

[31] C. Carlet, On highly nonlinear S-Boxes and their inability to thwart DPA attacks, in: International Conference on Cryptology in India, Springer, 2005, pp. 49–62. doi:10.1007/11596219_5.

[32] K. Chakraborty, S. Sarkar, S. Maitra, B. Mazumdar, D. Mukhopadhyay, E. Prouff, Redefining

the Transparency Order, Designs, codes and cryptography 82 (2017) 95–115. doi:10.1007/s10623-016-0250-3.

[33] Y. Fei, A. A. Ding, J. Lao, L. Zhang, A statistics-based success rate model for DPA and CPA, Journal of Cryptographic Engineering 5 (2015) 227–243. doi:10.1007/s13389-015-0107-0.

[34] C. Carlet, A. Heuser, S. Picek, Trade-offs for S-Boxes: cryptographic properties and Side-Channel Resilience, in: International Conference on Applied Cryptography and Network Security, Springer, 2017, pp. 393–414. doi:10.1007/978-3-319-61204-1_20.

[35] S. Picek, K. Papagiannopoulos, B. Ege, L. Batina, D. Jakobovic, Confused by confusion: systematic evaluation of DPA resistance of various S-Boxes, in: International Conference on Cryptology in India, Springer, 2014, pp. 374–390. doi:10.1007/978-3-319-13039-2_22.

[36] S. Guilley, P. Hoogvorst, R. Pacalet, Differential power analysis model and some results, in: Smart Card Research and Advanced Applications Vi, Springer, 2004, pp. 127–142. doi:10.1007/1-4020-8147-2_9.

[37] F.-X. Standaert, T. G. Malkin, M. Yung, A unified framework for the analysis of side-channel key recovery attacks, in: Annual international conference on the theory and applications of cryptographic techniques, Springer, 2009, pp. 443–461. doi:10.1007/978-3-642-01001-9_26.

[38] S. Guilley, A. Heuser, O. Rioul, A key to success, in: International Conference on Cryptology in India, Springer, 2015, pp. 270–290. doi:10.1007/978-3-319-26617-6_15.

[39] J. L. Massey, Guessing and entropy, in: Proceedings of 1994 IEEE International Symposium on Information Theory, IEEE, 1994, p. 204. doi:10.1109/ISIT.1994.394764.

[40] C. O'Flynn, Z. Chen, Synchronous sampling and clock recovery of internal oscillators for side channel analysis and fault injection, Journal of Cryptographic Engineering 5 (2015) 53–69. doi:10.1007/s13389-014-0087-5.

[41] M. T. Sakallı, B. Aslan, E. Buluş, A. Ş. Mesut, F. Büyüksaraçoğlu, O. Karaahmetoğlu, On the algebraic expression of the AES S-Box like S-Boxes, in: International Conference on Networked Digital Technologies, Springer, 2010, pp. 213–227. doi:10.1007/978-3-642-14292-5_23.

[42] K. Nyberg, Differentially uniform mappings for cryptography, in: Workshop on the Theory and Application of of Cryptographic Techniques, Springer, 1993, pp. 55–64. doi:10.1007/3-540-48285-7_6.

[43] A. Freyre-Echevarría, I. Martínez-Díaz, C. M. L. Pérez, G. Sosa-Gómez, O. Rojas, Evolving nonlinear S-Boxes with improved theoretical resilience to power attacks, IEEE Access 8 (2020) 202728–202737. doi:10.1109/ACCESS.2020.3035163.

[44] S. Picek, M. Cupic, L. Rotim, A new cost function for evolution of S-Boxes, Evolutionary computation 24 (2016) 695–718. doi:10.1162/EVCO_a_00191.

[45] G. Tang, X. Liao, Y. Chen, A novel method for designing S-Boxes based on chaotic maps, Chaos, Solitons & Fractals 23 (2005) 413–419. doi:10.1016/j.chaos.2004.04.023.

[46] K. M. Ali, M. Khan, Application based construction and optimization of substitution boxes over 2D mixed chaotic maps, International Journal of Theoretical Physics 58 (2019) 3091–3117. doi:10.1007/s10773-019-04188-3.

[47] F. Özkaynak, Chaos based substitution boxes as a cryptographic primitives: challenges and opportunities, Chaotic Model. Simul. 1 (2019) 49–57.

[48] M. M. Dimitrov, On the design of chaos-based S-Boxes, IEEE Access 8 (2020) 117173–117181. doi:10.1109/ACCESS.2020.3004526.

[49] S. H. Tolpa, M. A. Abdelhamed, E.-S. S. A. Said, M. Y. I. Afi, A novel chaos-based approach for constructing lightweight S-Boxes, Scientific Reports 15 (2025) 34112. doi:10.1038/s41598-025-20019-4.

[50] F. Artuğer, A method for designing substitution boxes based on chaos with high nonlinearity, Wireless Personal Communications 135 (2024) 1077–1092. doi:10.1007/s11277-024-11104-4.

[51] F. Özkaynak, Construction of robust substitution boxes based on chaotic systems, Neural Computing and Applications 31 (2019) 3317–3326. doi:10.1007/s00521-017-3287-y.

[52] X. Hou, W. Wang, Lightweight dynamic advanced encryption standard encryption based on S-Box reconfiguration and real-time key expansion for secure Over-the-Air communication, Electronics

14 (2025) 3274. doi:10.3390/electronics14163274.

[53] T.-H. Tran, D.-T. Dam, T.-K. Dang, D.-H. Le, T.-T. Hoang, C.-K. Pham, Countering side-channel attacks with a dynamic S-Box based on affine transformations and gold sequences, IEEE Transactions on Very Large Scale Integration (VLSI) Systems (2025). doi:10.1109/TVLSI.2025.3595897.

[54] Z.-l. Liu, X. Guo, Y.-c. Chen, Y. Han, X.-c. Zou, On the ability of AES S-Boxes to secure against correlation power analysis, in: International Conference on Information Security Practice and Experience, Springer, 2007, pp. 43–50. doi:10.1007/978-3-540-72163-5_5.

[55] L. Lerman, O. Markowitch, N. Veshchikov, Comparing S-Boxes of ciphers from the perspective of Side-Channel attacks, in: 2016 IEEE Asian Hardware-Oriented Security and Trust (AsianHOST), IEEE, 2016, pp. 1–6. doi:10.1109/AsianHOST.2016.7835556.

[56] M. Ş. Açikkapi, F. Özkaynak, A. B. Özer, Side-channel analysis of chaos-based substitution box structures, IEEE Access 7 (2019) 79030–79043. doi:10.1109/ACCESS.2019.2921708.

[57] I. Hussain, T. Shah, M. A. Gondal, H. Mahmood, An efficient approach for the construction of LFT S-Boxes using chaotic logistic map, Nonlinear Dynamics 71 (2013) 133–140. doi:10.1007/s11071-012-0646-1.

[58] C. O'Flynn, Z. Chen, A case study of side-channel analysis using decoupling capacitor power measurement with the OpenADC, in: International Symposium on Foundations and Practice of Security, Springer, 2012, pp. 341–356. doi:10.1007/978-3-642-37119-6_22.

[59] Atmel, Atmel 8 and 16 bit AVR microcontrollers – ATxmega, http://ww1.microchip.com/downloads/en/devicedoc/atmel-8135-8-and-16-bit-avr-microcontroller-atxmega16d4-32d4-64d4-128d4_datasheet.pdf, 2016. Accessed 15 March 2025.

[60] NewAE, avrcryptolib - AES, https://github.com/newaetech/chipwhisperer/tree/develop/firmware/mcu/crypto/avrcryptolib/aes, August 2024. Accessed 5 September 2025.