An In-Depth Evaluation of Externally Amplified Coupling (EAC) Attacks — a Concrete Threat for Masked Cryptographic Implementations

Ofek Gur, Tomer Gross, Davide Bellizia, François-Xavier Standaert, and Itamar Levi

***Abstract*—Masking is a systematic countermeasure to achieve side-channel security for cryptographic algorithms. However, its secure implementation relies on an independence assumption that can be violated by signal coupling. It has been established that coupling induced within a device can be detrimental. It was demonstrated on a 1^{st}-order secure design (i.e., with two shares) that an adversary who can manipulate the design's power-measurement setup can externally induce significant coupling. It can thus concretely reduce the "effective-security-order", i.e., make 1^{st}-order leakages as significant as 2^{nd}-order ones with fewer measurements. This paper explores the impact of such external amplification phenomena on fabricated hardware test cases for the first time. We designed a dedicated ASIC to extend the empirical results for demonstrating impact up to the 4^{th} order. We have systematically evaluated factors related to adversarial control, e.g., the external measurement resistance. We also investigated their relative influence compared to intra-design ones, i.e., internal power-grid resistance and transistors' inherent resistance. Our study demonstrates that externally amplified coupling scales up to concrete masked hardware designs with various amounts of shares and is not very sensitive to intra-design parameters. Therefore, providing experimental evidence that such coupling should be considered during masking validation.**

***Index Terms*—Coupling, Effective Security Order, Externally-Amplified-Coupling, EAC, Masking, Side-channel analysis**

I. Introduction

Masking is used to prevent side-channel attacks by splitting all sensitive variables of an implementation into d shares. Subsequently, computations are performed over these shared values. One key assumption behind masked designs is the *independence assumption*, which requires that leakages produced during computations depend on at most one share each from each secret variable. Alternatively, the total leakage can be written as a linear function of the leakages from computation of the shared values. In practical terms, if there is sufficient noise in the measurements, then an adversary will be forced to extract secrets from higher-order statistical moments of the leakage distribution. The latter task's cost in data complexity increases exponentially with increasing numbers of shares [1]–[3]. The lowest key-dependent moment of the leakage distribution is usually denoted as the (statistical) *security order*. However, it is well-known that implementing masking schemes in a way that fulfills the independence assumption is not a trivial task. There is abundant literature on the challenges relating to signal 'glitches' [4], memory-recombination [5]–[8], and *composition* issues. These problems can be handled by verification tools, such as FullVerif [9] and extensions [10], MaskVerif [11], and SILVER [12], as illustrated in Fig. 1(a). This paper focuses specifically on signal coupling challenges. *These are more complex since they cannot be handled within a simple logical or mathematical abstraction* [13], [14]; rather, they must be handled on the physical abstraction. To date, there are no masking-specific industrial tools to verify physical features associated with the implementation physics, as illustrated in Figures 1(a) and 1(b). An interesting and positive direction for tackling these issues is the ELMO simulator, which promotes building leakage templates from the device itself [15]. Clearly, such tools should also be directed at evaluating and modeling leakage from extreme signal coupling scenarios, as highlighted here.

Electrical energy transfer between circuits or electronic components is termed electronic coupling. In this paper we mostly relate to what is denoted by resistive and capacitive based coupling which highlights the dominating electronic element through-which such energy is transferred. We specifically discuss passive elements which exist either deliberately or parasitically in integrated circuits (ICs) and power distribution grids of ICs, as illustrated in Fig. 1(b). Potentially, sensitive information such as masked shared values (signals), manipulated or stored in one circuit, can couple to another via (e.g.,) device *internal* power-grids. In this work, we consider adversaries which might further amplify internal coupling by *external* means, such as external passive elements connected outside the device. Therefore, our taxonomy is *external*- or *internal*-coupling relating to these two scenarios.

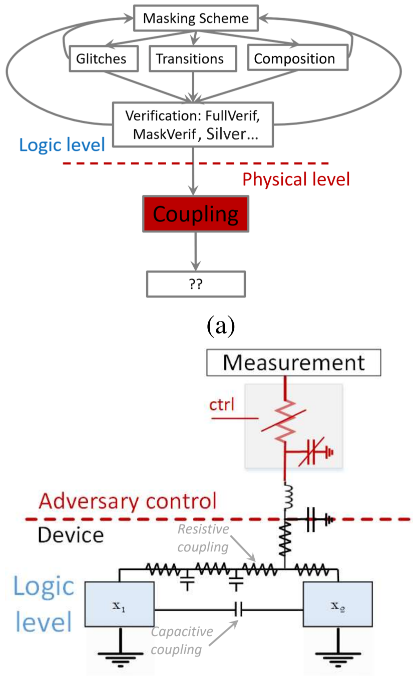
In this paper, logic- and storage-circuitry, associated with masked shared-variables, is denote by *shares*, i.e., we assume it is clear from the context if *shares* represent the shared logic values or the physical entities which store and process them.

Internal coupling induced by shares circuits proximity on FPGA hardware can be a threat if not resolved correctly in the design stages [16], [16]–[18]. This can be mainly attributed to current dividers from the main device's current source to underlying shares. Such current dividers make the leakage of specific manipulated shared values dependent on others [19].

External setup manipulations which induce coupling were first discussed by Moradi and Mischke [20] , as illustrated in the timeline of coupling-related research progress in Fig. 2. The study revealed how a specific case of serial processing of shared values could reduce the effective security

T. Gross, O. Gur and I. Levi are with the Faculty of Engineering, Bar-Ilan University, Ramat Gan, Israel (e-mail: {first.last}@biu.ac.il)

D.Bellizia and F-X. Standaert are from Université catholique de Louvain, ICTEAM/ELEN/Crypto Group, Belgium (e-mail: {first.last}@uclouvain.be)



(a)

(b)

Fig. 1: A general perspective of challenges with masked designs: (a) challenges that can be handled in logic layers, such as glitches, transitions, and compositions, and challenges like coupling, which must be considered by utilizing information from physical layers (b) a simplified EAC illustration.

order. Basically, by adding an external capacitor[1], leakage coupling of different shared values, processed in different time samples, could be induced. [19] showed that a side-channel adversary could inflict resistive coupling by simply adding sufficiently large resistors to a well-controlled power measurement setup. A linear device, i.e., a resistor, was enough to induce considerable amplitude coupling through current dividers when shared values were processed concurrently (or on parallel hardware). Coupling induced by all possible adversary's *external* manipulations were denoted as externally amplified coupling (EAC). However, the evaluation in [19] was analyzed for an architecture masked with only two shares and not over an ASIC platform, only a microcontroller and an FPGA which are different. In [21], the authors examined a masked software implementation and demonstrated how dangerous its unknown factors could be. They mainly focused on logical recombinations (glitches) with combinational elements, stemming from a barrel shifter, and also exemplified EACs in some processor operations (such as *move*/*store*). Important findings on *internal* coupling also appear in [16] where coupling induced by internal power grid elements were simulated over first-order secured designs. This paper focuses on hardware implementation scenarios. However, we believe the general conclusions and findings related to this threat should be taken as a cautionary note and be applied to software-based platforms.

[1]A DC blocker circuitry with a capacitive dominating element.

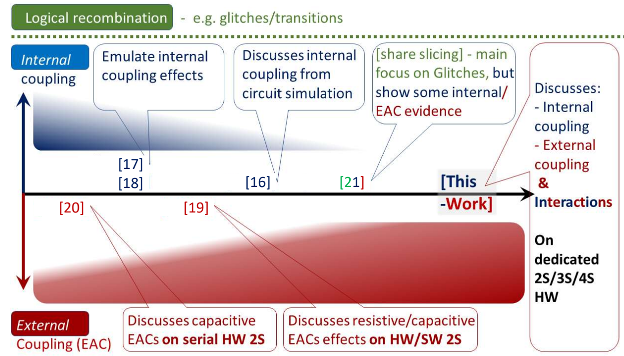


Fig. 2: Progress in coupling effects of masked designs.

Clearly, the above mentioned findings leave considerable room for further evaluation of designs protected by higher masking orders. The main goal of this work is to tackle this open challenge experimentally. Our carefully designed evaluation framework is built in a way that allows a parametric evaluation of several ASIC constructs with 2, 3, and 4 shares designs. It explores the following questions:

1) **Scaling up to more shares:** Can an adversary always lower the effective security order to the 1^{st} moment? The 2^{nd}? Does it incrementally progress with d?
2) **Data Complexity:** Does signal-coupling allow lowering the data complexity (number of required traces) and computational complexity of the SCA adversary, i.e., in which moment sensitive information is captured?
3) **Internal *vs.* external coupling:** How significant are technological features such as the device's *internal* power-grid resistance shared among different shares and transistors' drive strengths, as compared to *external* ones?

Our dedicated evaluation framework aims to provide experimental results, extracted from minimal testing constructs targeted to examine these substantial technology-dependent effects with a parametric nature. We tailored a specialized 65nm ASIC as a case study, allowing us to validate the exploitability of amplified leakages in orders $< d$.

Given this highly generic evaluation framework and the obtained results , our answer to the question on scaling-up of the EAC phenomena is positive. As for the second question, we provide data-complexity examples. We also answer the third question: EAC factors are more important for the exploitability of lower order leakages than internal ones on hardware test-cases, extending threats considered in [16] for the external adversary context, higher masking orders and measured-data. This paper highlights that EACs are also more dangerous than internal couplings because they are entirely under the adversary's control. Our results exhibit the difference between the theoretical and the *effective* security order. We conclude by discussing the impact of this phenomenon, the limitations of the analysis, the remaining open problems with respect to further generalization of our conclusions to noisy designs, hiding-protected designs, such as [22]–[24], and possibly other dedicated countermeasures for future evaluation.

The manuscript begins with a short background, including a general perspective and some necessary information. In Section III, we present our design implementation constructed for EAC evaluation, as well as the rationale for the specific analysis that follows, and provide examples of leakages. Section IV evaluates our parametric 2-shares masked design, the *internal* vs. *external* parameters of the physical implementation, and the leakage distributions.In Section V, we scale up to evaluate the EAC extent with 3 and 4 shares. We provide insight relating to the leakage model and its physical implementation, and quantitatively explore the questions raised above. Finally, we discuss the possible consequences, the limitations of our analysis, and open research in Section VI.

II. Background

This section includes some recent findings relating to coupling and an overall perspective. We then detail the Domain Oriented Masking, DOM, elements we utilize and embed in our ASIC device and the tool we use for evaluation.

We specifically consider masked implementations via Boolean masking where each sensitive variable s is represented by $d-1$ random variables $(s_1, s_2, ..., s_{d-1})$ and one more variable s_d, which complies with:

$$s = \oplus_{i=1}^{d} s_i, \tag{1}$$

where $\oplus$ is a group addition operation in a finite-field (in the case of binary values in GF(2), it represents the XOR operation between bits). If the secret is a vector of d-bits we utilize bold-face, $\boldsymbol{s}$), the operations are performed bitwise. The secret variable is never processed within the hardware, only its d shares are. For this purpose and to implement a cryptographic architecture, one must be able to perform logic operations securely on the shared values. In general, any logical function can be represented by multiplications (AND gates/operations) and additions (XOR gates/operations). The implementation of linear operations has a low computational cost, as they can be performed share by share. Special care should be taken for multiplications (see, for example, [25]) since they *recombine* values of different shares (logically). In this work, we examine a popular architecture to perform secure multiplications in the hardware context, as detailed below.

A. A General Perspective

Theoretical works have established that if the masking assumptions are met, given d shares, information will only leak on the d^{th} statistical moment of the leakage, as illustrated in Fig. 3(b-d)[2], where $f(l|s)$ denotes the conditional probability distribution of the leakage in the shared value. Note that to extract this information from measurements, e.g., by computing the d^{th} standardized statistical moment of the leakages, ΔSM^d, the required number of samples increases exponentially with d, as illustrated in Fig. 3(a).

B. Domain-Oriented Masking

Domain-oriented masking (DOM) is a masked hardware implementation strategy of masked gates proposed by Gross

[2]This refers to an ideal Hamming weight leakage function of each share.

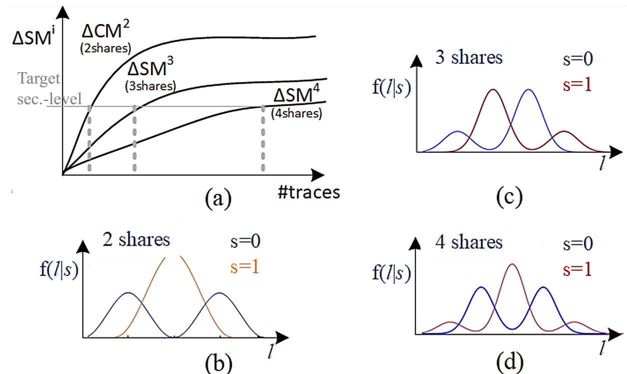


Fig. 3: Theoretical masking performance: (a) the statistical order capturing leakage in 2, 3, and 4 share designs versus data/time complexity (b-d) ideally modeled, Gaussian, and Hamming weight leakages for 2, 3, and 4 shares, respectively.

et al. [26]. It reduces the implementation cost by enforcing a certain separation of signals-paths and randomness compared to the naive software-oriented implementations of primitives in the original algorithm (created by Ishai, Sahai, and Wagner [25]). In this work, we make use of a masked AND shared variables. The approach consists of splitting the shared values into *domains* corresponding to the shares' indices (e.g., $\{x_0,y_0\}$ or $\{x_1,y_1\}$ for variables *x* and *y* shared in two), and to keep the shares from each domain independent of the shares of other domains. This is achieved by adding randomness to the combinational paths, before the mixed values are sampled. This independence guarantees $(d-1)^{th}$-order security, as illustrated in Fig. 4 adapted from [26]. Despite recent works revealing composition-level issues for this scheme (e.g., see discussions in [4]), our conclusions persist. This is maintained since we focus on designs without output-to-input feedback and show that coupling makes univariate leakages informative at low orders, while exploiting a composition flaw for such designs would require a multivariate analysis[3].

C. Welch's T-test for Higher Statistical Moments

In this paper, we test for the existence of leakages in high statistical moments up to the d^{th} moment, denoted as $\hat{M}_s^d$. Moments are computed on a subset of the leakage samples.

[3]Concretely, the DOM solution was also state-of-the-art when taping out the ASIC investigated in this paper. We refer to [9] for a composable solution.

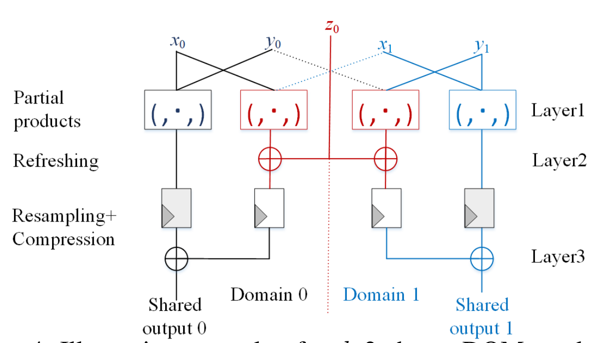


Fig. 4: Illustrative example of a d=2 shares DOM masking.

The samples are grouped by an internally processed secret value s (either '0' or a '1'); i.e., over l_s^t, the leakage time sample ($t \in \{0, ..., \#\text{samples}\}$), corresponding to different outcome manipulation of s. For the 2^{nd}-order, the second-order central-moment, $CM_s^{2,t} = E((l_s^t - \mu)^2)$, is used instead of the raw moment, M; and for higher orders ($d > 2$), the standardized moment is used, $SM_s^{d,t} = E((\frac{l_s^t - \mu}{\sigma})^d)$. Where, μ and σ are the populations' means and standard deviations, respectively; μ and σ operate on the entire vector of observations in a set per time sample $\mathbf{l}_s^t$.

Our analysis is based on the Test Vector Leakage Assessment procedure from Cryptography Research, CRI [27], [28]. The popular leakage detection approach utilized is the traditional univariate method, based on Welch's (two-tailed) T-test [29]. It is computed on two input sequences (Set_0 and Set_1). In this work, we compare two classes of leakages with so-called specific "*fixed* vs. *fixed*" [30], [31] tests to detect leakages using the following T-test statistic:

$$T_{value} = \left(\mu(SM^i_{Set_0}) - \mu(SM^i_{Set_1})\right) \Big/ \sqrt{\sigma^2(SM^i_{Set_0})/|Set_0| + \sigma^2(SM^i_{Set_1})/|Set_1|}, \quad (2)$$

We use the generalization in [32] for higher-orders.

Our analysis below is performed on small test-constructs, i.e., with very small number of input bits and no rounds or iterations of computations such as ones which appear in a full permutation (say AES). Therefore, and considering some known limitations of the tests as generally pointed out in [30], and more specifically in [31] (E.g., Section 1), we perform the "*fixed* vs. *fixed*" test. One of the motivations to use a "*fixed* vs. "*random*" test is to identify non-specific leakages which are hard to find when (e.g.,) some countermeasures are present such as *shuffling* with specific tests ("*fixed* vs. *fixed*"). However, in our case we can easily capture all "*fixed* vs. *fixed*" scenarios, showing worst-case leakages owing to the circuit's logic simplicity. The nice feature of such a test is that it is translated directly to concretely exploitable SCA attacks.

Thanks to our controlled hardware setup, we can conclude that detected leakages correspond to specific sensitive operations so that they would naturally translate into concrete attacks (E.g., consider discussions in [33]). Since such key recoveries have already been demonstrated in [19], we do not repeat them here. However, we have also performed "*fixed* vs. *random*" tests. These, as expected, yielded similar but slightly worse results, owing to averaging of moments in the random set, and are therefore less meaningful/interesting. The threshold for the tests was set at $|4.5|$ (the standard in SCA-related literature). However, this value depends on the degrees of freedom (df) selection and the number of traces. In all experiments below, the confidence levels were computed, and the thresholds were verified to hold while computing p-values.

D. A note on EACs and different computational platforms

The key factors affecting the significance of the EAC phenomenon are power-grid impedance and the existence/ characteristics of power-regulation circuitry. Basically, different computational platforms (a microprocessor, FPGA or ASIC) may exhibit very different power-grid impedance characteristics. Concretely, with FPGA technologies, by-design internal capacitive loads over the power-grid are very significant. This is a result of plurality of switching elements connected, regardless if they are logically utilized or not. Electronically, this is a non-issue because it helps prevent voltage *droops* [34]. However, it negatively affects capacitive coupling. A microcontroller is likely to have a lighter capacitive load over the power-grid path to the source IO. In ASIC designs, the situation is very flexible and application dependent, therefore dangerous, which highlights the importance of this research. Signal coupling, owing to simultaneous (i.e., parallel) shared values manipulation, are more dangerous in scenarios characterized by low capacitive loads of the internal power grid and large internal (or external) resistance. Thus, ASICs are the most problematic candidates: (1) they can potentially exhibit minimal capacitive loads on the power grid and (2) the internal grid resistance is a parameter set by physical/back-end designers, and in the context of EACs, very large *effective* resistance can be reflected anyway from externally by adversaries.

To date, EACs have been demonstrated on: (1) FPGAs - in [19], [20] and (2) microcontrollers - in [19], [21]. However, no work has been reported on ASICs with high-orders analysis providing measurement results. Another aspect is the interaction of EAC attacks with power regulators. Design details of regulators in FPGAs and microcontrollers are typically protected by IPs. In [19], EACs were shown over a microcontroller that included an on-chip regulator. In this sense, our work is mainly aimed to evaluate the effectiveness of EACs without a regulator. I.e., without assumptions on the protection such regulators may provide or not.

As discussed in the introduction, in [21], the authors examined how special operations on a processor with unknown implementation details may induce: (1) logical-recombinations (glitches) and (2) internal coupling / EAC in some processor operations. More specifically, their goal was to show that when designing for an unknown software environment (processor implementation), it is very hard to guarantee security. This is due to micro-architectural effects *which inherently falsify masking assumptions*. They demonstrated combinational logic-recombination of shared values, i.e., *glitches* on dedicated combinational barrel shifter hardware.

Especially relevant to this paper, further evidence for concrete capacitive/resistive coupling was presented in [21]: *move/store* ARM-processor instructions were shown to leak for several clock cycles past execution, which is expected from global routing. In addition, they interacted with other processed shared values that were supposed to be manipulated at different time samples. The outcome was a leakage in lower orders than the theoretical d. We see no explanation for this result, other than coupling (capacitive/resistive) since no "logical" interactions are expected. In the context of the [21] paper, our contribution and focus are:

- In this manuscript, our main agenda is to stress that there are concrete issues other than masking assumptions being overlooked in logical layers, even if the main difficulty is that the hardware is unknown. What was shown in [21] is an issue that can be handled with logical analysis and tools. In this work, we focus on issues that must be handled with hardware design tools, verification, and experimentation/falsification. Interestingly, although different mechanisms of leakage were explored, that is, **logical** glitches and **electronic** coupling, very similar results of security order reduction were demonstrated in both of our works: main leakage in the 2^{nd} moment for a 4^{th} order design and main leakage in the 1^{st} moment for a 2^{nd} order design.

- Another interesting observation reported by Gao et al. [21] is that on their platform, when both significant logical-recombinations (glitches) and EACs exist, glitches affect the effective order reduction more severe.

E. Adversary and Threat Model

In our attack we assume an adversary has physical access to a device by which measurements can be performed, he has knowledge on the whereabouts of the power-supply pin and ability to mount external passive device on this power-supply path. Either the adversary has these abilities or he can get access to such abilities, e.g., by remotely controlling power-regulation on say a server-CPU or utilizing temperature or voltage sensing infractructure on server cores [35]–[37], regardless of the applicability of such scenario. We have no assumptions about the internals of a device, nor does the attack require any physical information on the internal power grid.

III. Externally Amplified Coupling and Design-for-Evaluation

As illustrated in Fig. 5(a), internal signal coupling in a device occurs when the leakage associated with the activity of one shared value affects the leakage originating from the associated hardware of other shared value. The severity of internal coupling is primarily affected by the impedance of the power delivery network. The current drawn from one share, e.g., s_1, may induce a shared voltage level for both s_1 and s_2. Depending on the IR drop of the main voltage supply over the internal power grid resistance, $R_{int} = \epsilon$ yields a total internal current, $I_{int} = f'(R_{s_1}, R_{s_2}, \epsilon)$, where f' is the joint leakage function of the circuit. In standard manufacturing technologies, ϵ is typically designed to be much smaller than the internal effective resistances of the shares circuitry, R_{s_1} and R_{s_2}. In turn, it implies that the internal voltage level V_{int} is lightly affected by the shares circuitry resistance, i.e., equals V_{ext}. However, ϵ is never 0. Therefore, some internal coupling always exists. Yet, when ϵ is very small relative to the shares circuitry resistance, the total leakage contains only a negligible joint or multiplicative factor of the independent shares' leakages. This implies that the summation of shares' circuitry leakages does not contain such joint-factors, i.e., $I_{int} \sim f(R_{s_1}, \epsilon) + f(R_{s_2}, \epsilon)$, where, in this case, f represent the internal leakage function of each of the shares. This finding was explored and supported by an approximated model in [19]. A simplified example is illustrated in Fig. 5(c). In this sub-figure, a simple RC circuitry models a scenario where two shared values are stored (i.e. it abstracts a storage cell or a flip-flop). When no EACs are modeled (i.e., by R_{ext}), the current depends linearly on the shared values, $I_{int} = V_{ext}(R_{s_1}^{-1} + R_{s_2}^{-1})$. However, when EACs are modeled, non-linearity appear, $I_{int} = V_{ext}/(R_{ext} + (R_{s_1}^{-1} + R_{s_2}^{-1})^{-1})$. In fact, due to the inverse components in the equation, a sum-of-products with all multiplied powers of R_{s_i} factors will exist.

TABLE I: Parameters and corresponding sections. #sh denotes the number of shares evaluated.

Evaluation	Parameter	Value range	#sh	Sec.
Internal-2s	R_{int}	{1,... ,8} ϵ	2	4
Internal-2s	Dup (emulating R_L)	{1, 5, 20}	2	4
External-2s	R_{ext}	{0,... ,100} Ω	2	4
Int/ext-3s	R_{int}, Dup, R_{ext}	all as above	3	5
Int/ext-4s	R_{int}, Dup, R_{ext}	all as above	4	5

Externally induced signal coupling depends on the external manipulations an adversary can induce on the (e.g.,) power supply path of a device. That is, regardless of the internal technological parameters of a design, an adversary may affect the impedance of the measurement path. In this case, shares leakages can recombine on passive physical elements. For example, on our simplified schematic representation (Fig. 5(a)), an increase of R_{int} can be externally mimicked by introducing an external resistance R_{ext} over the measurement path. This was demonstrated in [19] over a FPGA and microcontroller environment. Internal coupling effects were also discussed in [16] on a simulated environment. In practice, the power delivery network is complex, introducing actual impedance and not only resistive elements, but also capacitive and, in some scenarios, non-negligible inductive elements. All of which contribute to a very complex network and very hard to analyze/argue leakage functions. In this work, our intention is to demonstrate that even such a simplified view of EAC already provides a level of modeling and understanding which can be utilized to analyze the EAC mechanism. Clearly, as demonstrated below, masked designs suffer from such EACs. Therefore, and regardless of the complexity of the model used for intuition, they pose a concrete threat.

In this research, we explore how the coupling phenomenon affects the distribution of the leakage. But more importantly, we study the extent to which it is possible to externally amplify this coupling using different mechanisms and how would technological parameters affect it. Therefore, considering Fig. 5(b), theoretically, we aim to explore design parameters and adversarial external parameters. The list of evaluated parameters is detailed next. Table I , lists their value ranges, and the sections in the paper where each parameter is evaluated.

1) Internal parameters:
   a) Internal resistance emulated through internal power grid devices (power-gating) with controllable effective resistances, denoted by R_{int} in Fig. 5(b).
   b) Drive-strength of transistors emulated through duplication of logic elements), denoted by R_L.
2) External parameters: e.g., external resistance, R_{ext}.

Our starting point is that EACs were not evaluated for more than 2 shares. As explained, the physical effects of EAC mechanisms are very complex at the electronic level, and the answers are not easy to obtain from a model[4]. Therefore,

[4]However, note that in [19] a model is provided, which shows that EACs will be present for all d-share designs.

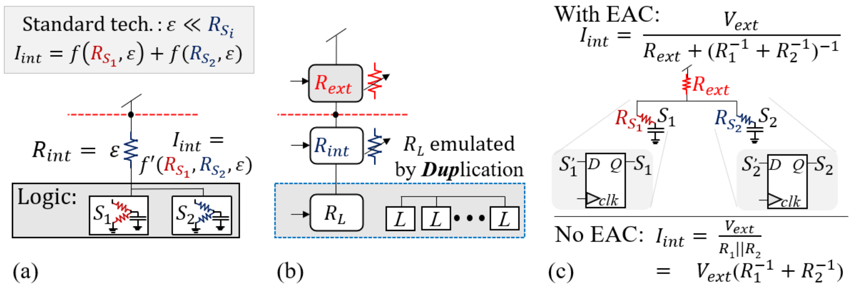


Fig. 5: Illustration of the device's: (a) internal coupling and (b) internal and external coupling parameters, and (c) simplified example of resistive EAC which breaks the *independence* assumption.

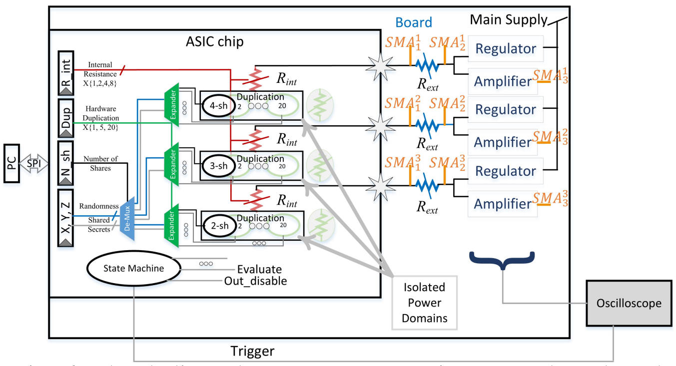


Fig. 6: The dedicated measurement environment, board, and a high-level view of the internal ASIC based test-bed.

we aim to evaluate their effectiveness with 3 and 4 share designs through experimentation. The effects of some internal parameters were discussed in the literature [17], [18], and in [16] on a circuit simulated setting, leaving ample room for additional questions related to how external manipulations affect coupling? and the importance of internal parameters.

Following this motivation, a carefully tailored evaluation ASIC was designed in a standard $65nm$ process technology, accompanied by a supporting evaluation board, as illustrated in Fig. 6. Three blocks containing constructs to evaluate $\{2, 3, 4\}$ share designs were embedded on the chip. Each was placed in its own power domain provided with an isolated and independent power measurement port. In the chip, special circuitry was embedded (details below) on the internal power lines of each design to emulate different R_{int}s' in the ASIC, schematically illustrated by a red variable resistor. The value was programmed by setting appropriate bits in the R_{int} dedicated 3-bit register. The effective resistance of the logic layers, R_{s_i} was emulated and controlled by parametric *duplication* of the logic, which was programmed by setting the 3-bit Dup register. In accordance with the Dup value, inputs were duplicated and assigned only to $\{1, 5, 20\}$ duplicated elements, where the rest were tied to '0'. The logical constructs to achieve this are denoted by an *Expander* green block on the scheme. The shared values, X and Y, and the required randomness, Z, were assigned only to the block under evaluation (with or without duplication) from corresponding registers. The current number of shares of the design under evaluation was set by the N_{sh} parameter, controlled by an appropriate register (i.e., containing d). It set the block under evaluation and demultiplexed these values to the required destination (denoted in blue). A state machine controlled the triggering mechanisms to measurement equipment and whether to disable inputs assignment to the blocks after a predefined number of cycles. It also controlled whether or not to send the computation outputs to the user for verification in order not to induce large and unwanted leakage. On the board, dedicated power regulators per block were embedded as well as amplifiers. Low-capacitance and resistance traces were implemented on the PCB with dedicated ports for R_{ext} changing (denoted by blue resistors on the figure). Communication was handled through an SPI channel supported by dedicated logic on the ASIC. All X, Y and Z values needed internally for the computation, are computed/generated on a different device, stored and sent via SPI to the ASIC for processing while leakage is measured. For a more detailed explanation, one example block of the design with two shares is schematically illustrated in Fig. 7:

1) Each block contained a DOM-indep. multiplier with 2, 3, or 4 shares, duplicated up to a maximum of 20 elements.
2) The emulated and controlled R_{int} was embedded utilizing an always-ON, low-resistance power-gate power gate element (a standard cell from the power-management kit). In a parallel resistance connection, 4 elements were connected with different sizing to control the internal power-grid resistance of the shared design. This sizing provides con-

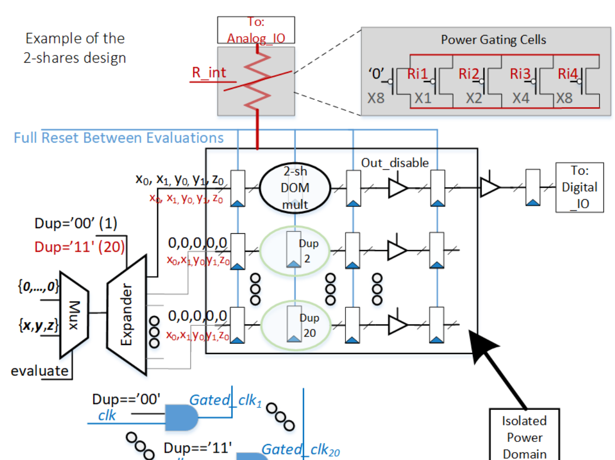


Fig. 7: Example: low-level view of the internal ASIC construct of the two shares design.

figuration of linearly spaced R_{int} values ($\epsilon, 2\epsilon, 3\epsilon, ..., 8\epsilon$).

3) In any case where Dup was not full (i.e., 20 duplicated elements), all elements that were not considered were assigned '0' inputs on all ports. Additionally, they were reset prior to the execution, including all internal registers, and the clock ports associated with them were gated, as illustrated on the bottom part of the figure.
4) For leakage evaluation, we wanted to keep internal signals local, connected only to small routing traces (small energy and capacitance footprint), and therefore our design supports the 'outputs disable' setting.
5) The design support a mechanism to prevent inputs assignment by an embedded predefined counter. Consequently, we were able to assign inputs over one clock cycle or more. The *evaluate* signal was controlled by that counter.

A. Example leakages - 2 shares

We begin the analysis and evaluation part of the manuscript with an example of leakages taken on our evaluation environment. For that purpose, we start with a standard-case evaluation of $R_{ext} = 0$ (non-EAC), with the 2 shares block. An SMA connection voltage measurement point on the board is probed by a *true* 12-bit ADC resolution PicoScope oscilloscope. The device internal clock was set to 6MHz. The sampling frequency of the oscilloscope was set to 200MHz (approximately 30 samples per cycle). Our first goal was to understand how clean was our measurement environment (regardless of masking). Therefore, we have performed a T-test evaluation on known and pre computed internal layers of the multiplier, i.e., **unshared** internal computations. That is, as we know exactly the shared values of X, Y, and Z (,A, B, and C on the figure, respectively), we are able to compute the internal values computed within our DOM multiplier (see Fig. 4, relating to the indicated layers): **Layer 0** - randomness input, i.e., known value z. **Layer 1** - all partial products (inner- and cross-domain), denoted by $a_i \otimes b_j,\ i.j \in 0, 1$. **Layer 2** - randomness addition / register-values, $a_0 \otimes b_1 \oplus z$ and $a_1 \otimes b_0 \oplus z$. **Layer 3** - output compressed values, $(a_0 \otimes b_0 \oplus (a_0 \otimes b_1 \oplus z))$ and $(a_1 \otimes b_1 \oplus\ (a_1 \otimes b_0 \oplus z))$.

To evaluate the leakage from this set of layered computation, a T-test of 1^{st} statistical moment of the leakage and the 2^{nd} centralized statistical moment with sets classification was performed in accordance with each of the computations (9 internal values as listed above), as discussed in Sub-section II-C. The results from the 1^{st} and 2^{nd} moments are presented in Fig. 8(a) and Fig. 8(b), respectively. Several observations are:

Causality: Leakage only appears following inputs assignment, marked by a dashed gray vertical line in the plots.

Leakage magnitude: A significant leakage was already revealed within the 1^{st} statistical moment with a small number of traces (in the range of 20 to 50). The T-values of the 2^{nd} moment were far greater, in the scale of hundreds.

Complex leakages: The leakage distribution is very complex and is clearly not a trivial Gaussian leakage- by observing the T-values of the precomputed internal variables, a very significant second order leakage appear for the un-masked variables. This is clearly not the case of a simple leakage distribution (illustrated by examples displayed in Fig. 12).

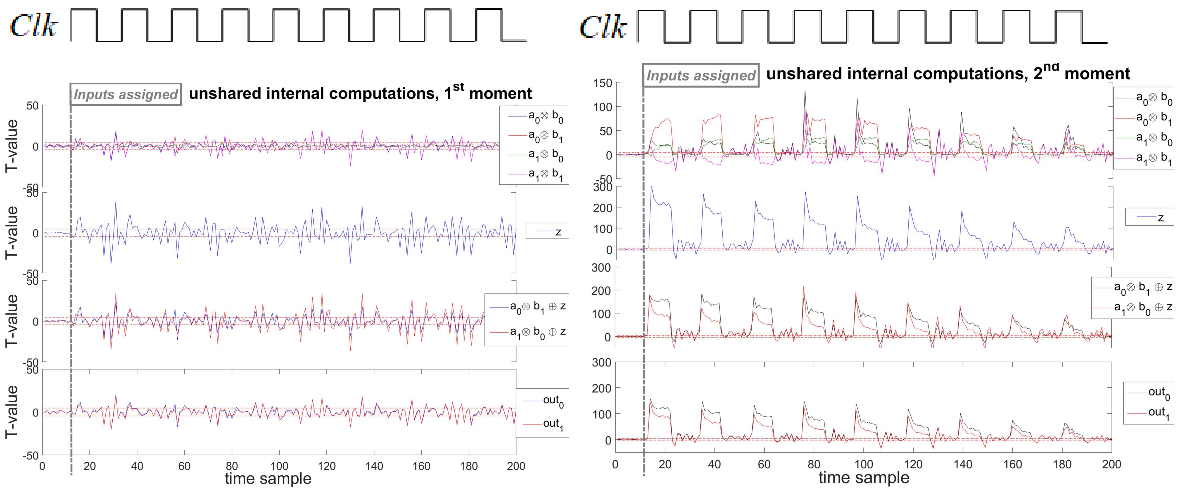
Unbalanced shares' leakages: That is, two logically-identical and symmetric shares leak differently, depending on the electronics, as opposed to the 'ideal' way we typically model the leakage, as shown for example by the different curves in the top of Fig. 8(b). While this outcome is expected, it raises the question of the extent to which it affects security. As we show below, we did not find any evidence of information leaking from our masked designs in lower statistical moments than d without considering the EAC scenario, implying no practical impact relating to security order reduction.

Decaying nature and filtering: inputs were assigned to the block for one clock cycle. The outcome leakage illustrates a decaying nature over the complex power-grid of the block-device-measurement path. This was most prominent in the 2^{nd} moment by a decaying pattern up to time sample 200. This behaviour is rather expected, especially as the design does not include algorithmic noise. The decay will show to be less prominent in following sections.

IV. Externally-Amplified Coupling - 2 shares design, a detailed evaluation

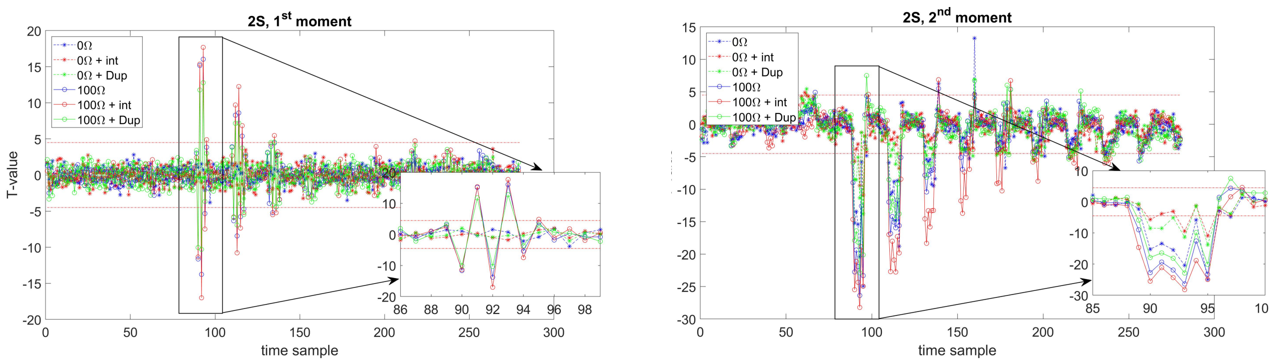
In this section, we first evaluate the extent of EACs over a design with two shares in extreme conditions, i.e., while considering very large external resistor values and the magnitude of internal coupling versus EACs. EACs were evaluated by assigning precision surface-mount (SMD) resistors from a 'short-circuit' (0 Ω) to large 100 Ω. Notably, for our regulated 1.2 V nominal supply voltage, resistors starting from around 200 Ω induced voltage drops larger than 400 mV, causing faults. Therefore, aggressive values of 100 Ω, inducing voltage drops of approximately 150 mV, were established as safe and conservative for illustrating the extent of EACs. Typically, SCAs current measurements through voltage-drops across a resistor utilize small resistors in the range of only few Ohms. Generally, it is known that increasing this resistor in the standard (non-masked) SCA context, can increase the SNR owing to larger signals, differentially generated across larger resistors. However, if this resistor value is enlarged too much, at some stage the side-channel SNR will starts to decrease, owing to reduced *on*/*off* current ratio of the underlying logic elements. In the EAC context, this balance is different, as illustrated in Fig. 5(c): increasing the value of this resistor, induce coupling between shares' leakages which are then measured; enlarging the resistor will increase the generated coupling, and will not only improve the noise-sensitivity or the measurement resolution.

To evaluate internal technological parameters related to coupling, we vary the internal power-gating resistance in the range of $\{1,\ldots,8\}\ \epsilon$, where ϵ is estimated at a maximum of 10 Ohms. The resistance of the logic elements within shares (i.e., the maximum switching resistance of a standard gate) is estimated in the range of 0.1 to 1 kΩ. To emulate a technology with reduced resistance, as discussed above, we duplicate cells a maximal duplication of 20, reducing the effective resistance induced by logic elements by a factor of approximately 20. Starting with an evaluation of the 1^{st} statistical moment



(a) (b)

Fig. 8: DOM-indep. multiplier, leakages over our 65nm ASIC evaluation environment with $\{R_{ext}, R_{int}\} = 0, 0$. T-test of: (a) 1^{st} moment (b) 2^{nd} centralized moment. In both panels: **Top** - Layer 1 partial products, 2^{nd} **from top** - randomness input, 3^{rd} **from top** - Layer 2 randomness addition / register-values, **Bottom** - Layer 3 output compression.



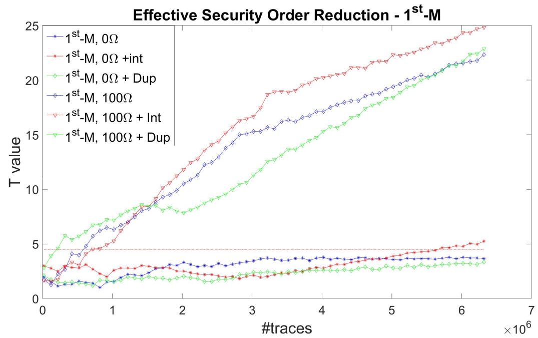
(a) (b)

Fig. 9: T-values of 2 shares AND output with $R_{ext} = \{0, 100\}\ \Omega$, and with and without $R_{int} = 8\ \epsilon$ and *Dup*=20 versus time sample: (a) 1^{st} moment and (b) 2^{nd} moment.

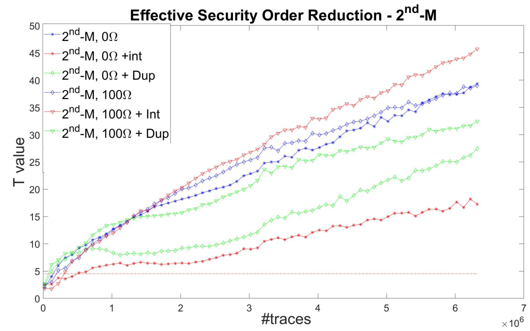
leakage over time, Fig. 9(a) shows the results of the T-test, with sets grouped in accordance with the value of the AND output, whereas Fig. 9(b) shows the results of the 2^{nd} central moment. In each figure, results are plotted for $R_{ext} = \{0, 100\}\ \Omega$ with a maximum internal resistance $R_{int} = 8\ \epsilon$ (annotated by '+*int*') and a maximum duplication of 20 (annotated by '+*Dup*'). A zoom-in view of the maximum leakage point-of-interest is provided. For this figure, the #traces $3.5{\cdot}10^6$ were used. Next, we explore trends and evaluations with more traces.

Notably, the first set of leakages associated with $R_{ext} = 0$ do not compromise the secret value within the 1^{st} moment view. However, all evaluation scenarios of $R_{ext} = 100\Omega$ do so with confidence, reaching considerable detection values of close to 20, far above the threshold. Within the results of the 2^{nd} central moment detection, all scenarios compromise the secret, as expected. Causality is clearly illustrated in the figures, where leakages appear when inputs are assigned at around time-sample 90. The decaying nature of the leakage is also observable, owing to the power-grid's RC-like nature. A more detailed investigation of the results in the zoomed-in subplots shows that, as expected, in each of the R_{ext} states, increasing R_{int} increases coupling in the 1^{st} and 2^{nd} moments leakages and increasing the Dup factor reduces it. The latter is due to the lesser dominance of $R_{int} + R_{ext}$.

The first fundamental question that we tackle is whether EACs can induce leakages in lower statistical moments than the expected d^{th} moment. This question has a positive answer. However, a more quantitative question we face is: does that provide the adversary with a concrete advantage? This is manifested in lower data and processing complexity. To answer this question, Fig. 10 depicts the T-value results versus the number of leakage samples used. Note that henceforth, all the figures in this manuscript presenting T-values vs. #traces depict the maximum absolute value. The first main observation from Fig. 10(a) is that the $R_{ext} = 0\ \Omega$ case does not provide a 1^{st}-order advantage. That is, internal coupling does not play a significant role with up to about $4{\cdot}10^6$ traces. By contrast, EACs do reveal 1^{st}-order information already with approximately $150 \cdot 10^3$ traces and $550 \cdot 10^3$ traces with no duplications. An interesting observation is that the "100 Ω + Dup" curve crosses the "100 Ω" and "100 Ω + *int*" curves. We hypothesize that it is due to the spatial distribution of the duplicated elements, generating



(a)



(b)

Fig. 10: T-values of 2 shares AND output with $R_{ext} = \{0, 100\}$ Ω, with and without $R_{int} = 8$ ϵ and *Dup*=20 versus #traces at POI: (a) 1^{st} moment (b) 2^{nd} moment.

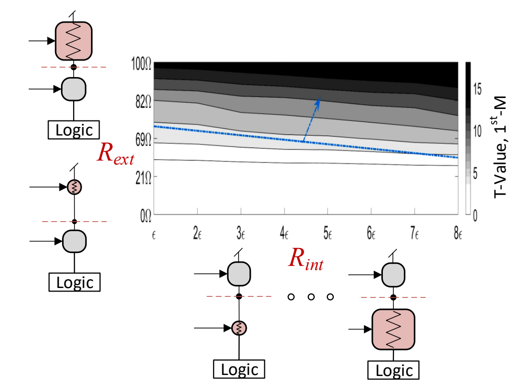
some variance factor, which, in turn, requires more samples to stabilize with statistical confidence. Next, considering the 2^{nd}-moment results, all scenarios show significant detection already with about $100 \cdot 10^3$ to $600 \cdot 10^3$ traces. That is, we demonstrate that, for the design with two shares, EACs reduce the effective security order but not concretely the data-complexity of an adversary. This picture significantly changes for 3 and 4 share designs. It is hard to establish why this occurs only for the 2 shares design, as it is the result of a complex leakage distribution over a complex power-grid. Nevertheless, the appearance of leakage in the 1^{st} moment validates our assumptions on the EAC mechanisms, and internal parameters versus ECS impact is observed.

A. EAC - R_{int} vs. R_{ext}

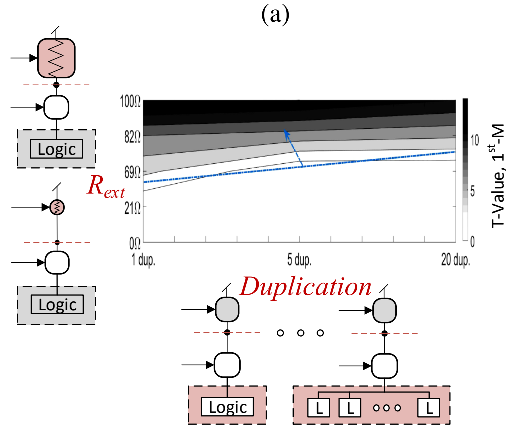
In this subsection, we investigate the influence of internal parameters on the leakage. The investigation is important, as it is aimed at clarifying conceptual questions regarding manufacturing technologies: Would a more/less resistive power delivery network or a standard-cell library with stronger/larger devices considerably affect coupling ? That is, what are the design factors that exert a negative influence on the independent leakage assumption? Figure 11(a) shows the maximum detected T-value of the 1^{st}-order leakage in a gray-scale color map, where the x-axis represent the value of R_{int} configured on the device, illustrated below the axis by a scheme with a small/large internal resistor. The y-axis represents the value of R_{ext} connected to the device, illustrated to the left of the axis by a scheme with a small/large external resistor. The external resistors set used was about {0,21,69,82,100} Ω. The figure clearly demonstrates a trend where the increase in both factors increases the leakage. However, the contours reveal that external resistance is the dominant factor.

B. EAC - Logic Elements Resistance (Dup) vs. R_{ext}

We now consider Fig. 11(b), which shows the maximum detected T-value of the 1^{st} order leakage, where the x-axis represents the value of the *Dup* register configured on the device, illustrated below the axis by a scheme with a no/maximal logic duplication. The y-axis represents the value of R_{ext} as above. The figure clearly shows a trend of leakage increase with both anincrease in external resistor and a decrease in the duplication factor. However, the contours again reveal that external resistance is the dominant factor and that the internal resistance of cells is very hard to bias. That is, it would be hard for designers to make EAC attacks hard by changing the dimensions of the logic-devices utilized. Alternatively, it would be hard to generate significant internal coupling effects to a point of significant leakage. Moreover, values of R_{ext} exhibiting a significant/sufficient EAC range from 21 to 69 Ω with little dependence on the *Dup* factor.



(a)



(b)

Fig. 11: T-values of 1^{st}-moment over a 2 shares AND output with #traces $= 3.2 \cdot 10^6$ at POI: (a) R_{ext} vs. R_{int}, (b) R_{ext} vs. *Dup*. Blue lines indicate trends of crossing a T-value = 5.

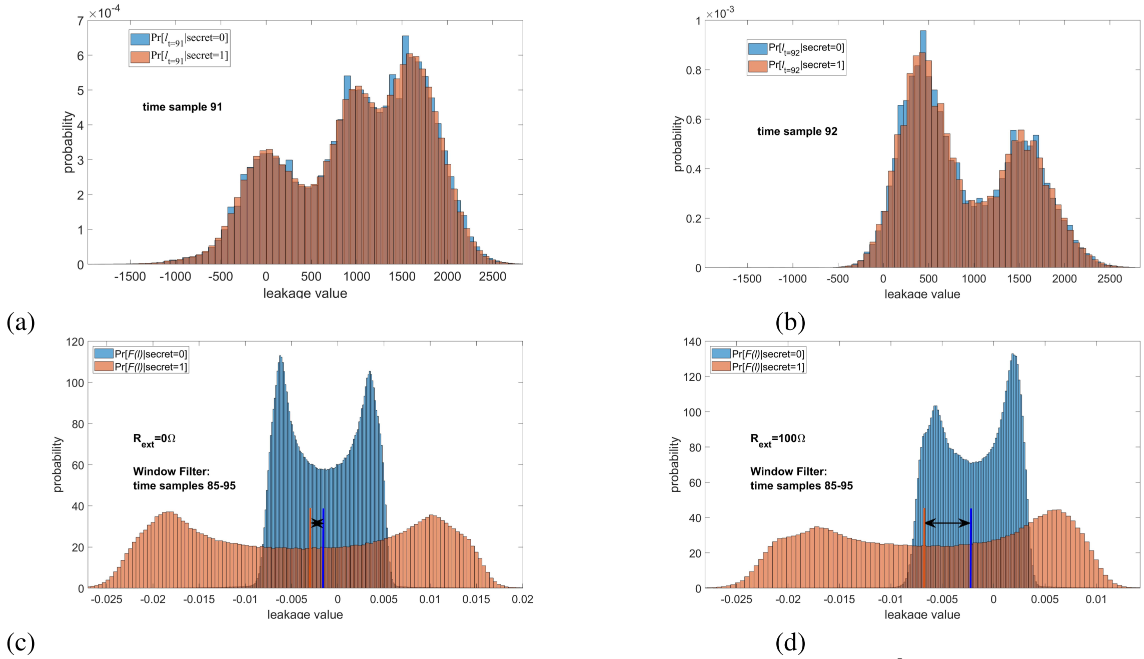


Fig. 12: Leakage distribution of a 2 shares AND output with $\#\text{traces} = 3.2 \cdot 10^6$ at (a) time sample 91 (b) time sample 92, and with window filtering over R_{ext} = (c) 0 Ω (d) 100 Ω. Leakage values (x-axis) represent the sampled leakage ADC scale directly or after the window filtering.

C. Complex leakage distributions

An interesting question relates to the shape of the leakage distribution in this rather complex ASIC environment. One important feature of ASIC technologies is that signals change rapidly and the leakages of neighboring computations overlap, due to the rapid propagation of signals and the impedance of the power delivery network and filtering effects. This makes it hard to isolate leakages from specific internal computations. Evaluating the leakage distribution of two neighboring leakage samples can show how fast internal and other computations are and the decay rate of other parasitic effects. In Fig. 12 (a-b), the probability distribution of the leakages is plotted when grouped to shared-output = '0'/'1' in time samples 90 and 91 (from Fig. 9). Clearly, the shape of the distribution varies considerably between these two close time samples.

Several computations affect the leakage simultaneously, including toggling of combinational elements, which takes place in proximate time samples. Their toggling is manifested in a power-grid current with different propagation time constants. Note that in our environment, the on-chip power delivery network parasitic capacitance is at its minimum. In the real world, the situation is worse (especially in FPGAs), leading to larger effects of capacitive coupling, as noted in [20]. Regenerating such distributions for multiple scenarios/designs/parameter cases, and in various time samples reveals that we are not able to observe the obvious and ideal distributions. I.e., ones which only exist with modeled Gaussian distributions with Hamming weight modeling, consider Fig. 3. Nevertheless, these distributions reveal considerable leakage with significant confidence, as shown above. Therefore, considering the profile of the filtering effects over the power-grid (recall the decaying nature of the leakage), we post-processed the leakages with a rather trivial time-domain Hamming-window filtering (convolution) with a 10-sample width, while taking into account the periodic leakage segments we obtained. The distribution of the point-of-interest that revealed the maximum information is illustrated in Fig. 12 (c-d) for two cases; in (c) for R_{ext} =0 Ω and in (d) for R_{ext} =100 Ω. As shown, in case (c), the difference between the means of the blue and the orange distributions is very small (in fact, it was not detected). However, the difference in the variance is clearly visible and resembles more to what we would expect from theory. In case (d), the difference between the means is highly significant, while clearly, the variance also conveys a great deal of information on the secret value. This filtered view bears a much closer resemblance to previous studies with EACs over a FPGA and microcontroller environments [19].

V. EAC Extent with More Shares

This section explores the remaining question of the extent of EAC with high(er) orders of masking. We first present a simplified model developed in [19] (Section 2.2), where I_i represents the leakage current of share i, and I'_j is the approximated leakage current of share j. The total current flowing through the main supply can then be approximated assuming a significant external resistance R_{ext} as shown in Eq. 3[5]. Where V_{DS_j} represents the drain-source voltage over the pull-up network of transistors in share j. Within a switching activity, its maximum value for all shares is a constant, V_{DD}. Basically, Eq. 3 implies that in addition to the d^{th} order leakage (the first summation), there always exists EAC

[5]With several approximations; the transistors' conductance ranges from 10^4 to 10^6 Siemens, expanding the *Taylor* series (of $1/(1+x) \approx 1 - x + x^2 - x^3 ...$), and I_i is linearly approximated in one summation to get I'_i.

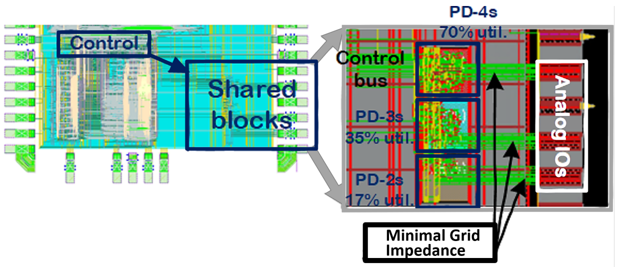


Fig. 13: Design layout highlighting power domain utilization, minimal power-grid impedance, and Analog-IO connections.

elements in the leakage of the $d/2^{th}$ order if d is even (the second summation) and there exists a mixture of joint factors of shared values leading to leakages in lower statistical orders (represented by different powers and colors). However, eventhough the gradually increasing powers combine leakages, their magnitude changes: the coefficients which are multiplied are smaller; hence, they will be harder to distinguish. Note that, as discussed above, a real-life system is excessively complex while a simplified modeling attempt may not reveal all the details. Nevertheless, we show below that our model succeeded to relate well to the outcome of resistive EACs.

$$I'_{\text{supply}} \approx \underbrace{\sum_i I_i}_{d^{th}-M} - \frac{R_{\text{ext}}}{V_{DS_j}} \cdot \underbrace{\sum_i I_i \left[\sum_j I'_j\right]}_{\lfloor d/2 \rfloor^{th}-M} + \\ + \frac{R_{\text{ext}}}{V_{DS_j}}^{\mathbf{2}} \cdot \sum_i I_i \left[\sum_j I'_j\right]^{\mathbf{2}} - \frac{R_{\text{ext}}}{V_{DS_j}}^{\mathbf{3}} \cdot \sum_i I_i \left[\sum_j I'_j\right]^{\mathbf{3}} + ... \tag{3}$$

As shown in Fig. 13, the three blocks placed on the ASIC chip have the same area footprint and more importantly, similar power-grid impedance characteristics. All the power connections to power domains (PDs) are connected on all the possible geometries to reduce power-grid resistance. They are placed with minimal spacing from the IO-ring to reduce power-grid capacitance as much as possible, where they are finally connected to power-isolated Analog-IOs. In comparative terms, all 2, 3, and 4 share designs have exactly the same chip-internal power-grid characteristics. However, the blocks internal impedance varies, as does the area utilization of $\{17, 35, 70\}\%$, illustrated in the figure. This implies increased logic-resistance while switching as the #shares increase.

We start with the 3 shares design implementation. A measurement set of about $20 \cdot 10^6$ traces is collected with several R_{ext}, R_{int}, Dup values. However, after evaluation, internal factors are shown to only slightly affect results as compared to the more dominant R_{ext} factor. The left side of Fig. 14 indicates the T-values of the first three statistical moments (as mentioned above, the third was standardized) versus the number of samples for R_{ext} =0 Ω. On the right side of Fig. 14, the same evaluation repeats for R_{ext} =100 Ω. The first significant observation is that it requires around $17 \cdot 10^6$ traces to extract information from the third statistical moment when R_{ext} =0 Ω. The data-complexity increase is expected (recall the discussion related to Fig. 3). Note also that we did not find any observable leakage or concrete deviation from theory on this test case. However, the right side of the figure where R_{ext} =100 Ω, calls for three important comments:

- Leakage in the d^{th} moment: the leakage in the 3^{rd} moment appears at as few as $9 \cdot 10^6$ traces, implying that EACs make it easier to extract information even from the theoretical security-order moment.
- The most significant effect emerges for the 2^{nd} moment, where EACs reveal information with as few as $4 \cdot 10^6$ traces (compared to $17 \cdot 10^6$ without EAC). Thus it manifests a concrete reduction in the effective security order, which translates into actual data complexity gains.
- Information is also pushed down statistically to the 1^{st}-order moment. However, perhaps due to noise or the complexity of the actual leakage function induced by the EAC, this effect is only observed when a very high number of traces is used, thus providing no advantage.

Let us now look at the 4 shares implementation. A measurement set of approximately $50 \cdot 10^6$ traces was collected with several R_{ext}, R_{int}, Dup values. However, as mentioned above, only significant results dominated by R_{ext}, are discussed here. On the left side of Fig. 15, the T-values of the first four statistical moments are presented versus the number of samples for R_{ext} =0 Ω. On the right side of Fig. 15, the same evaluation repeats for R_{ext} =82 Ω. The R_{ext} value is reduced since the current drawn from this block is larger, thus inflicting a larger voltage drop. As shown, it requires approximately $35 \cdot 10^6$ samples to extract information from the fourth statistical moment when R_{ext} =0 Ω, and the significance of the detection increases rapidly from that point. We list several observations:

- Leakage in the d^{th} moment: 4^{th}-moment leakage appears at approximately $34 \cdot 10^6$ traces, implying no significant change due to EACs from the theoretical security-order.
- The most significant effect emerges for the 2^{nd} moment, again, where EACs reveal information with as few as approximately $11 \cdot 10^6$ traces (compared to $35 \cdot 10^6$ without EAC), thereby exhibiting a concrete reduction of the effective security order, and actual data complexity gains.
- Information is also pushed down statistically to the 3^{rd} moment in this case, and not to the 1^{st} moment as was observed for the design with three shares.

Hence, these two experiments suggest that it is possible to identify some statistical links between the leakages in EAC scenarios pushed down to lower even moments (the closest even $d-1$), when the number of shares in the implementation is odd, and leakages in EAC scenarios pushed down to lower even moments (the closest even $d-2$) when the number of shares in the implementation is even. At this stage, our dedicated testing ASIC environment does not provide circuitry to evaluate designs with more shares. To concretely make claims about such connections and links, further research and more complex ASIC tapeouts instantiating more masked designs are indeed required for that porpuse. Eq. 3 indicates that in the 4 shares design, the 2^{nd} moment is more severe than the 4^{th} moment leakage. In the case of the 3 shares design

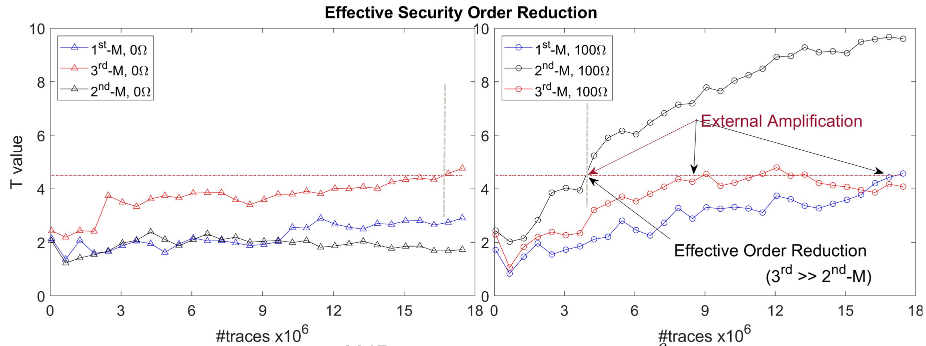


Fig. 14: T-values of three moments over a 3-share AND with $\#\text{traces} = 18 \cdot 10^6$ at POI: (a) $R_{ext} = 0\ \Omega$, (b) $R_{ext} = 100\ \Omega$.

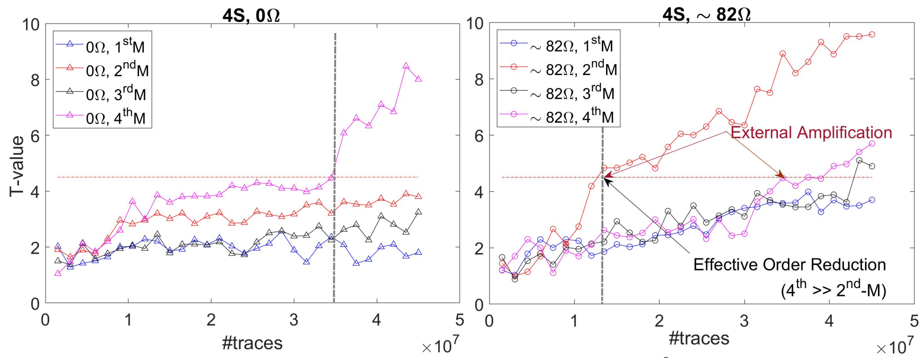


Fig. 15: T-values of four moments over a 4-share AND with $\#\text{traces} = 45 \cdot 10^6$ at POI: (a) $R_{ext} = 0\Omega$, (b) $R_{ext} =\sim 82\Omega$.

due to individual share leakage joint overlapping products, the leakage emerges as early as the 1^{st} order leakage, although it is less severe than the EAC'ed 2^{nd}-order moment.

VI. Discussion

The discussion below focuses on the impact of the EAC phenomenon, the limitations of the proposed analysis, and the remaining open challenges to further generalize our conclusions to noisy and already hiding-protected designs. Since in this research coupling exhibited a concrete threat, we discuss how to best utilize dedicated hardware for its prevention, such as regulators and sensors. Although such mechanisms are restrictive from a cryptographic standpoint in the sense that assumptions are needed about an adversary's inabilities to manipulate them, we believe they are key ingredients.

Low-noise - our environment is very low-noise with respect to the measurement environment, the construction of the board, the chip's internal power-grid, and the isolated power domains design. However, noise is expected to *shift* all results relatively, which is not likely to trigger significant *relative* changes in a noisier design: the foundations of masking rely on noise amplification. It has been previously modeled by Chari et al. [1] that the number of measurements needed to distinguish 1-bit with HW model (i.e., $\Pr(L|y=0)$ from $\Pr(L|y=1)$) is $n_{attack} \le \sigma_n^{d+4\cdot log(\alpha)/log(\sigma_n)} \overset{\alpha=1}{=} \sigma_n^d$,assuming a Gaussian noise, a probability of α, and shares-independence. The attack Success-Rate (SR) of a key-recovery with m measurements is also bounded by $SR^{kr} \le 1-(1-(\sigma_n^2)^d)^m$ [3]. Therefore, we can expect that increasing the inherent underlying noise will render it exponentially harder to capture information from a non-EAC'd design ($n_{attack} \le \sigma_n^d$). If EAC attacks makes leakages apparent in some lower-order, $d-j$ moment, giving some set of j shared-values leakages are dependent, it will imply: $n_{attack}^{EAC} \le \sigma_n^{d-j}$. Consequently, it seems that capturing it will require even fewer traces with noise-scaling (factor β), i.e., $\frac{n_{attack}}{n_{attack}^{EAC}} \le \frac{(\beta\sigma_n)^d}{(\beta\sigma_n)^{d-j}}$. However, we stress that such extrapolations should be treated with great care and require further investigations.**Low algorithmic noise** - in our environment, algorithmic noise is not present by design. The goal is to provide a clear evaluation of the EAC phenomenon without artifacts. Similar to the considerations on physical noise, we do not believe the trends observed (for ASICs) will vary greatly in a more large-scale system.

Large R_{ext} values - we captured rather large external values to show the extremes of EACs. We also captured the effects with much lower values than the maximum shown in the figures (e.g., see Fig. 11) for the range of detection-threshold passing values. Nevertheless, in some systems, it might be hard to connect such large resistors since they may induce high probability faults caused by large voltage drops.

There are several natural mitigation tactics for EAC attacks:

- **Natural countermeasures** - in general, it is expensive to utilize masking alone to provide security. Therefore, combining randomization mechanisms prior to masking to achieve the desired security level with a smaller d is an interesting option to investigate, as detailed in [22], supported by a concrete low-cost countermeasure. We expect that countermeasures that randomize the power-grid impedance in a low-cost and localized way, will make EACs more difficult, while at the same time reduce the overall cost.
- **Power regulation** - exists today in most commercial devices. Although EACs can even be captured through a

power-regulated software implementation [19] (and perhaps also in [21]), we anticipate that tailoring the regulator's properties to attenuate EACs' fingerprint is possible, e.g., as explored in [23], [38]).

- **Sensors** - One of the natural mechanisms to handle EACs is through a dedicated power-grid impedance sensor or even more advanced machine-learning based detection [39]. Despite the typical difficulties associated with sensing SCAs due to the large impedance changes, in the case of EACs, we believe these solutions may be efficient.

Future work will concentrate on both the limitations of our analysis and the mitigation tactics put forward above.

VII. Conclusion

Masking countermeasure is deployed and treated as a viable mechanism to reach a given security target. Signal coupling in hardware and software implementations is composed of natural electronic interactions that can breach masking's underlying assumptions. It has been demonstrated on a 1^{st}-order secure design over both software and hardware AES benchmarks that a designer who can manipulate the power-measurement setup of a design can externally induce significant coupling that can concretely reduce the "effective security order" of a design regardless of the intra-design. In this study, we analyzed the scaling-up of this external amplification phenomenon on hardware test cases. For that purpose, we first considerably extended earlier empirical results by showing EAC attacks threat remains significant even for higher orders of masking (2, 3, and 4 shares designs). To do so, we designed a dedicated ASIC. The main contribution of this work is a systematic evaluation of factors relating to the adversary's control, such as external measurement resistance. Additionally, our work contributed to several research aspects: (1) intra-design parameters, e.g., internal power-grid resistance with tailored programmable circuit-constructs embedded in the ASIC, and (2) device/transistor resistance emulated on hardware through the programmable duplication of devices. We demonstrated that externally amplified coupling, which is in complete control of the adversary, poses a significant threat, and that they scale-up to concrete masked designs (3 and 4 shares). Although often neglected, we show that externally amplified coupling should be evaluated early in the design stages and when validating masked designs. We discuss exploitability in a noisy environment, and embedding countermeasures.

Acknowledgments

This research was funded by Israel Science Foundation (ISF) grant number 2569/21. François-Xavier Standaert is supported by the FNRS-F.R.S. and in part by the European Union (ERC SWORD project 724725).

References

[1] S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi, "Towards sound approaches to counteract power-analysis attacks," in *Annual International Cryptology Conference*. Springer, 1999, pp. 398–412.

[2] E. Prouff and M. Rivain, "Masking against side-channel attacks: A formal security proof," in *in Advances in Cryptology - EUROCRYPT 2013*. Springer, 2013, pp. 142–159.

[3] A. Duc, S. Faust, and F. Standaert, "Making masking security proofs concrete - or how to evaluate the security of any leaking device," in *Advances in Cryptology - EUROCRYPT 2015*, 2015, pp. 401–429.

[4] T. Moos, A. Moradi, T. Schneider, and F.-X. Standaert, "Glitch-resistant masking revisited," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 256–292, 2019.

[5] S. Mangard, T. Popp, and B. M. Gammel, "Side-channel leakage of masked CMOS gates," in *Cryptographers' Track at the RSA Conference*. Springer, 2005, pp. 351–365.

[6] S. Mangard, N. Pramstaller, and E. Oswald, "Successfully attacking masked AES hardware implementations," in *International workshop on cryptographic hardware and embedded systems*. Springer, 2005, pp. 157–171.

[7] J.-S. Coron, C. Giraud, E. Prouff, S. Renner, M. Rivain, and P. K. Vadnala, "Conversion of security proofs from one leakage model to another: A new issue," in *International Workshop on Constructive Side-Channel Analysis and Secure Design*. Springer, 2012, pp. 69–81.

[8] J. Balasch, B. Gierlichs, V. Grosso, O. Reparaz, and F.-X. Standaert, "On the cost of lazy engineering for masked software implementations," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2014, pp. 64–81.

[9] G. Cassiers, B. Grégoire, I. Levi, and F.-X. Standaert, "Hardware private circuits: From trivial composition to full verification," *IEEE Transactions on Computers*, 2020.

[10] G. Cassiers and F.-X. Standaert, "Provably secure hardware masking in the transition-and glitch-robust probing model: Better safe than sorry," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 136–158, 2021.

[11] G. Barthe, S. Belaïd, G. Cassiers, P.-A. Fouque, B. Grégoire, and F.-X. Standaert, "Maskverif: Automated verification of higher-order masking in presence of physical defaults," in *European Symposium on Research in Computer Security*. Springer, 2019, pp. 300–318.

[12] D. Knichel, P. Sasdrich, and A. Moradi, "Silver–statistical independence and leakage verification," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2020, pp. 787–816.

[13] S. Nikova, V. Rijmen, and M. Schläffer, "Secure hardware implementation of nonlinear functions in the presence of glitches," *Journal of Cryptology*, vol. 24, no. 2, pp. 292–321, 2011.

[14] S. Faust, V. Grosso, S. Pozo, C. Paglialonga, and F.-X. Standaert, "Composable masking schemes in the presence of physical defaults & the robust probing model," 2018.

[15] D. McCann, E. Oswald, and C. Whitnall, "Towards practical tools for side channel aware software engineering:'grey box'modelling for instruction leakages," in *26th {USENIX} Security Symposium ({USENIX} Security 17)*, 2017, pp. 199–216.

[16] D. Šijačić, J. Balasch, and I. Verbauwhede, "Sweeping for leakage in masked circuit layouts," in *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2020, pp. 915–920.

[17] T. D. Cnudde, B. Bilgin, B. Gierlichs, V. Nikov, S. Nikova, and V. Rijmen, "Does coupling affect the security of masked implementations?" in *Constructive Side-Channel Analysis and Secure Design - 8th International Workshop, COSADE 2017, Paris, France, April 13-14, 2017, Revised Selected Papers*, 2017, pp. 1–18.

[18] T. D. Cnudde, M. Ender, and A. Moradi, "Hardware masking, revisited," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2018, no. 2, pp. 123–148, 2018.

[19] I. Levi, D. Bellizia, and F. Standaert, "Reducing a masked implementation's effective security order with setup manipulations and an explanation based on externally-amplified couplings," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2019, no. 2, pp. 293–317, 2019.

[20] A. Moradi and O. Mischke, "On the simplicity of converting leakages from multivariate to univariate," in *Cryptographic Hardware and Embedded Systems - CHES 2013*, ser. Lecture Notes in Computer Science, G. Bertoni and J. Coron, Eds., vol. 8086. Springer, 2013, pp. 1–20.

[21] S. Gao, B. Marshall, D. Page, and E. Oswald, "Share-slicing: Friend or foe?" *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 152–174, 2020.

[22] I. Levi, D. Bellizia, D. Bol, and F.-X. Standaert, "Ask less, get more: Side-channel signal hiding, revisited," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2020.

[23] D. Das, S. Maity, S. B. Nasir, S. Ghosh, A. Raychowdhury, and S. Sen, "Asni: Attenuated signature noise injection for low-overhead power side-channel attack immunity," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 10, pp. 3300–3311, 2018.

[24] W. Yu and S. Köse, "A lightweight masked aes implementation for securing iot against cpa attacks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 11, pp. 2934–2944, 2017.

[25] Y. Ishai, A. Sahai, and D. Wagner, "Private circuits: Securing hardware against probing attacks," in *Annual International Cryptology Conference*. Springer, 2003, pp. 463–481.

[26] H. Gross, S. Mangard, and T. Korak, "An efficient side-channel protected AES implementation with arbitrary protection order," in *Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14-17, 2017, Proceedings*, 2017, pp. 95–112.

[27] J. Cooper, E. D. Mulder, G. Goodwill, J. Jaffe, G. Kenworthy, and P. Rohatgi, "Test vector leakage assessment (TVLA) methodology in practice (extended abstract)," ICMC 2013.

[28] B. J. Gilbert Goodwill, J. Jaffe, P. Rohatgi *et al.*, "A testing methodology for side-channel resistance validation," in *NIST non-invasive attack testing workshop*, vol. 7, 2011, pp. 115–136.

[29] B. L. Welch, "The generalization of 'students' problem when several different population variances are involved," *Biometrika*, vol. 34, no. 1/2, pp. 28–35, 1947.

[30] F.-X. Standaert, "How (not) to use welch's t-test in side-channel security evaluations," in *International conference on smart card research and advanced applications*. Springer, 2018, pp. 65–79.

[31] F. Durvaux and F. Standaert, "From improved leakage detection to the detection of points of interests in leakage traces," in *Advances in Cryptology - EUROCRYPT 2016*, 2016, pp. 240–262.

[32] T. Schneider and A. Moradi, "Leakage assessment methodology - extended version," *J. Cryptographic Engineering*, vol. 6, no. 2, pp. 85–99, 2016.

[33] A. A. Ding, L. Zhang, F. Durvaux, F.-X. Standaert, and Y. Fei, "Towards sound and optimal leakage detection procedure," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2017, pp. 105–122.

[34] K. A. Bowman, S. Raina, J. T. Bridges, D. J. Yingling, H. H. Nguyen, B. R. Appel, Y. N. Kolla, J. Jeong, F. I. Atallah, and D. W. Hansquine, "A 16 nm all-digital auto-calibrating adaptive clock distribution for supply voltage droop tolerance across a wide operating range," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 1, pp. 8–17, 2015.

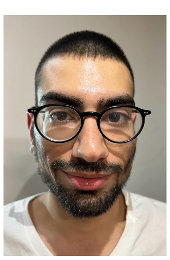
[35] C. Ramesh, S. B. Patil, S. N. Dhanuskodi, G. Provelengios, S. Pillement, D. Holcomb, and R. Tessier, "FPGA side channel attacks without physical access," in *2018 IEEE 26th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. IEEE, 2018, pp. 45–52.

[36] F. Schellenberg, D. R. Gnad, A. Moradi, and M. B. Tahoori, "Remote inter-chip power analysis side-channel attacks at board-level," in *2018 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. IEEE, 2018, pp. 1–7.

[37] S. Moini, S. Tian, D. Holcomb, J. Szefer, and R. Tessier, "Power side-channel attacks on BNN accelerators in remote FPGAs," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 11, no. 2, pp. 357–370, 2021.

[38] W. Yu and S. Köse, "A voltage regulator-assisted lightweight aes implementation against dpa attacks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 8, pp. 1152–1163, 2016.

[39] F. Kenarangi and I. Partin-Vaisband, "Exploiting machine learning against on-chip power analysis attacks: Tradeoffs and design considerations," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 2, pp. 769–781, 2018.



Ofek Gur received his B.Sc. from the Faculty of Engineering Bar-Ilan University in 2020, following which he has started his M.Sc. studies. Ofek's interests are embedded-systems, hardware-security and digital electronic design.

Tomer Gross received the B.Sc. degree in Computer Engineering from Bar-Ilan University, in 2020. Currently pursuing his M.Sc. in Cybersecurity at NYU Tandon School of Engineering. His research interests include a few areas of cyber security, including reverse engineering, malware analysis, and embedded and cyber-physical systems.

Davide Belizzia received the M.Sc. degree (summa cum laude) and the Ph.D. degree in electronics engineering from the University La Sapienza of Rome, Italy, in 2014 and 2018, respectively. In 2017, he joined to the Crypto-Group UCLouvain, Belgium, as a Post-Doctoral Researcher. His research interests include SCA countermeasures, design and test of cryptographic ICs, VLSI design, and DSP. In 2014, he received the Laureato Eccellente Award.

Francois-Xavier Standaert received the Electrical Engineering degree and PhD degree from the Universite catholique de Louvain, respectively in 2001 and 2004. In 2004-2005, he was a Fulbright visiting researcher at Columbia University, Crypto Lab (hosted by Tal G. Malkin and Moti Yung) and at the MIT Medialab (hosted by Neil Gershenfeld). From 2005 to 2008, he was a post-doctoral researcher of the Belgian Fund for Scientific Research (FNRS-F.R.S.) at the UCL Crypto Group. Since 2008 (resp. 2017), he is associate researcher (resp. senior associate researcher) of the FNRS-F.R.S. Since 2013 (resp. 2018), he is associate professor (resp. professor) at the UCL Institute of Information and Communication Technologies, Electronics and Applied Mathematics (ICTEAM). In 2010, he was program co-chair of CHES. In 2021, he was program co-chair of EUROCRYPT.In 2011, he was awarded a Starting Independent Research Grant by the European Research Council. In 2016, he has been awarded a Consolidator Grant by the ERC. From 2017 to 2022, he is a board member (director) of the International Association for Cryptologic Research (IACR). His research interests include cryptographic hardware and embedded systems, physical security including side-channel & fault attacks, and the design analysis of cryptographic primitives.

Itamar Levi received his B.Sc. and M.Sc. degrees in Electrical and Computer Engineering as a part of a direct excellence student track from Ben-Gurion University in 2012 and 2013, respectively. He completed his Ph.D. at Bar-Ilan University in 2017. He was a research-associate in UCLouvain, Belgium until 2019 with the UCLouvains Crypto-Group and currently he is a Computer-Engineering Faculty member at Bar-Ilan University, in Ramat Gan, Israel. He is also a member of Emerging Nanoscale Circuits and Systems Labs (EnICS), at BIU. Dr. Levi's current research interests are digital circuit design, embedded systems security, security evaluation analysis for cryptographic devices, side-channel and fault-injection countermeasures, and cryptographic implementations. He has (co-)authored over 50 journal articles and international conference papers and 7 patent applications, he co-authored a book on "Dual-Mode-Logic: A New Paradigm for Digital IC Design" and serves in several Technical Committees of IEEE CAS Society and Hardware Security journals and conferences.