Side-Channel Attacks from Static Power: When Should we Care?

Santos Merino Del Pozo[1], François-Xavier Standaert[1], Dina Kamel[1], Amir Moradi[2].

[1] ICTEAM/ELEN/Crypto Group, Université catholique de Louvain, Belgium.
[2] Horst Görtz Institute for IT Security, Ruhr University Bochum, Germany

***Abstract*—Static power consumption is an increasingly important concern when designing circuits in deep submicron technologies. Besides its impact for low-power implementations, recent research has investigated whether it could lead to exploitable side-channel leakages. Both simulated analyses and measurements from FPGA devices have confirmed that such a static signal can indeed lead to successful key recoveries. In this respect, the main remaining question is whether it can become the target of choice for actual adversaries, especially since it has smaller amplitude than its dynamic counterpart. In this paper, we answer this question based on actual measurements taken from an AES S-box prototype chip implemented in a 65-nanometer CMOS technology. For this purpose, we first provide a fair comparison of the static and dynamic leakages in a univariate setting, based on worst-case information theoretic analysis. This comparison confirms that the static signal is significantly less informative than the dynamic one. Next, we extend our evaluations to a multivariate setting. In this case, we observe that simple averaging strategies can be used to reduce the noise in static leakage traces. As a result, we mainly conclude that (a) if the target chip is working at maximum clock frequency (which prevents the previously mentioned averaging), the static leakage signal remains substantially smaller than the dynamic one, so has limited impact, and (b) if the adversary can reduce the clock frequency, the noise of the static leakage traces can be reduced arbitrarily. Whether the static signal leads to more informative leakages than the dynamic one then depends on the quality of the measurements (as the former one has very small amplitude). But it anyway raises a warning flag for the implementation of algorithmic countermeasures such as masking, that require high noise levels.**

I. Introduction

The possible exploitation of static power consumption through side-channel analysis against cryptographic implementations has been put forward a while ago. Previous works have provided concrete evidence that it can lead to successful key recoveries, based on simulated analyses and actual measurements [2], [7], [11], [16]. For example, in the latter reference, Moradi described successful experiments of static leakage based side-channel analysis against FPGA implementations. So qualitatively, this source of leakage is well known and has already triggered research on countermeasures [1], [8], [24].

In this paper, we aim to study the problem from a more quantitative point-of-view. In particular, we are interested in the question whether there exist situations where focusing attacks on the static leakage could provide concrete gains compared to the standard approach exploiting dynamic power. Answering this question requires to perform fair comparisons between both types of leakages, which (to the best of our knowledge) has been left out of the analysis in previous works (essentially because ad hoc attacks were sufficient for proving qualitative statements). For this purpose, we relied on the evaluation framework proposed at Eurocrypt 2009 [20], and performed worst-case information theoretic and security analyses of an AES S-box implemented in a 65-nanometer CMOS technology. Our main conclusions are twofold.

First, in cases the target implementation runs at (or close to) maximum frequency, the information provided by static power remains substantially lower than that of dynamic leakage – so is essentially useless. This is because its signal amplitude is significantly lower and a limited number of samples are available to exploit it. Second, in cases the adversary can control (and reduce) the clock frequency of a target device, the information provided by static power can be the weak point – so become a critical source of leakage. This is because reducing the clock frequency may significantly increase the portion of the traces containing only static leakage, hence allow reducing its noise via simple averaging strategies. Such observations naturally fit with the standard intuition in low-power design that it is the integration of static power over (long) time(s) that makes it energy consuming. In view of simulated analyses available from smaller technologies, we believe this intuition is valid for most current implementations.

These results have important consequences. First and very concretely, they confirm that clock control may be a significant advantage for side-channel adversaries (as already hinted in previous works on the topic [17]). Second, they raise warning flags for the security of different algorithmic countermeasures against side-channel attacks, since many of them have as primary requirement that the measurements should be "sufficiently noisy". This is the case, e.g. of masking [18] (that we briefly discuss in conclusion of this work) and shuffling [23]. Eventually, they exhibit a context where the "only computation leaks" paradigm (that is frequently used in formal works aiming to prevent side-channel attacks [15]) is not respected.

II. Background

Notations. In the rest of the paper, we use capital letters for random variables and small caps for their realizations. Vectors are denoted with bold notations, functions with Greek letters or sans serif fonts and sets with calligraphic ones.

A. Target chip and setup

Our analysis is based on actual power traces obtained from the execution of an AES Rijndael S-box, full-custom designed in a low-power 65-nanometer CMOS technology, measured under a 1.2V supply voltage. We used an area-optimized architecture based on composite field arithmetic, described in [14], of which the design is detailed in [10]. Measurements were performed on a prototype chip implementing this S-box, made of 1,530 transistors and with a maximum logic depth of 22. This leads to a delay of 3 ns at 1.2V supply voltage, hence tolerating operating frequencies up to 200 MHz (taking a security margin of 2 ns). Yet, since our goal was to investigate static leakages, our measurements were running the chip at a much lower 2 MHz frequency. Concretely, we monitored the voltage drop on a resistor introduced in the supply circuit of the chip, using a high sampling rate 8-bit oscilloscope. For illustration, Figure 1 shows an exemplary trace where the samples corresponding to the dynamic and static power consumption are marked. However, in order to exploit the full quantization range of our oscilloscope, these static and dynamic parts have been measured separately (see the figures in the next section). Furthermore, and in order to limit the memory requirements of our experiments, we measured 256×1,000 traces at a sampling rate of 2 GS/s corresponding to the 256 S-box inputs for the dynamic parts of the traces, and 256×100 traces at 1 GS/s corresponding to the 256 S-box inputs for the (much longer) static parts of the traces. In the following, we denote the i^{th} leakage trace corresponding to the intermediate value $y = x \oplus k$ with $\mathsf{S}(y) \rightsquigarrow \boldsymbol{l}_y^i$. We further use $\boldsymbol{l}_y^{i,\text{dyn}}$ and $\boldsymbol{l}_y^{i,\text{stat}}$ for the dynamic and static parts of the traces. Eventually, whenever accessing the j^{th} time sample of these traces, we will use the notations $\boldsymbol{l}_y^{i,\text{dyn}}(j)$ and $\boldsymbol{l}_y^{i,\text{stat}}(j)$. These subscripts and superscripts will be omitted when not necessary and clear from the context.

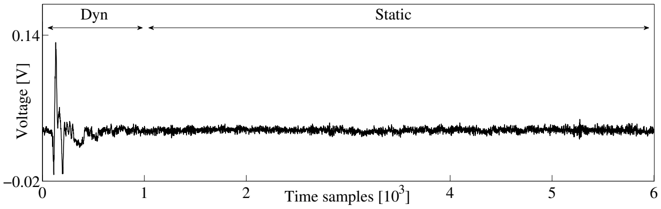

Fig. 1. Leakage trace with dynamic and static parts of the power consumption.

B. Evaluation tools and metrics

We now describe the tools and metrics used in our experiments. In order to mitigate biases due to incorrect a-priori choices of leakage models, we rely on profiled distinguishers. Namely, we estimate the Signal-to-Noise Ratio and the Perceived Information of our implementation, based on Gaussian Templates, possibly enhanced by Principal Component Analysis.

1) Gaussian Templates (GT): Introduced by Chari et al. [5] at CHES 2002 as a powerful attack against cryptographic implementations, GT essentially extract secret information based on probabilistic leakage models (next denoted as $\hat{\Pr}_{\text{model}}$) – which can typically be used to estimate the Perceived Information metric (see below). For this purpose, GT assume that the leakages can be interpreted as the realizations of a random variable which generates samples according to a Gaussian distribution. For example, when the target intermediate value is a key addition (as in the following), we have that $\hat{\Pr}_{\text{model}}[\boldsymbol{l}_y|x,k] \approx \hat{\Pr}_{\text{model}}[\boldsymbol{l}_y|x \oplus k] \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_y, \hat{\Sigma}_y^2)$, with $\hat{\boldsymbol{\mu}}_y$ and $\hat{\Sigma}_y^2$ the mean vector and covariance matrices corresponding to the target intermediate value y. The latter simplifies to $\hat{\Pr}_{\text{model}}[l_y|x \oplus k] \sim \mathcal{N}(\hat{\mu}_y, \hat{\sigma}_y^2)$ in the case of univariate leakages. Hence, this approach only requires the estimation of the sample means and (co)variances corresponding to each $y = x \oplus k$ from a set $\mathcal{L}_Y^p$ of N_p profiling traces. We denote this profiling step as $\hat{\mathsf{Pr}}_{\text{model}} \leftarrow \mathcal{L}_Y^p$. In order to recover the key, GT then estimates probabilities for each candidate k^*:

$$p_{k^*} = \prod_{i=1}^{q} \hat{\mathsf{Pr}}_{\text{model}}[k^*|x, \boldsymbol{l}_y^i],$$

which corresponds to the DPA setting where the input x is given to the adversary. Note that for this second step, the traces are taken from a new set $\mathcal{L}_Y^t$ with N_t test traces.

2) Principal Component Analysis (PCA): In order to maximize the amount of information extracted from leakage traces while keeping the advantage of low-dimensional data spaces, dimensionality reduction techniques such as PCA can be applied, as introduced in the context of side-channel analysis by Archambeau et al. [3] at CHES 2006. PCA projects the traces into a subspace of small dimensionality while optimizing the inter-class variance. When applied to mean traces in the context of unprotected implementations (as in the following), this corresponds to maximizing the so-called "leakage signal" (which corresponds to the variance taken over the mean traces). Concretely, we will apply PCA to the dynamic parts of the traces, and denote the reduced samples as $\mathsf{PCA}(\boldsymbol{l}_y^{i,\text{dyn}})$.

3) Signal-to-Noise Ratio (SNR): Introduced by Mangard [12] at CT-RSA 2004, the SNR of a sample point in a power trace can be defined as:

$$\hat{\mathsf{SNR}} = \frac{\hat{\mathsf{var}}_y\left(\hat{\mathsf{E}}_i(L_y^i)\right)}{\hat{\mathsf{E}}_y\left(\hat{\mathsf{var}}_i(L_y^i)\right)},$$

with $\hat{\mathsf{E}}$ and $\hat{\mathsf{var}}$ denoting the sample mean and variance operators. Contrary to Perceived Information which relies on GT (hence requires two sets of profiling and test traces), the SNR is estimated from a single set of N_t test traces.

4) Perceived Information (PI): The information theoretic metric introduced by Standaert et al. [20] at Eurocrypt 2009 aims at quantifying the leakage of an implementation by measuring the Mutual Information (MI) between the sensitive values manipulated by the device and the leakages it produces:

$$\mathsf{MI}(K; X, \boldsymbol{L}) = \mathsf{H}[K] - \sum_{k \in \mathcal{K}} \mathsf{Pr}[k] \sum_{x \in \mathcal{X}} \mathsf{Pr}[x] \cdot \sum_{\boldsymbol{l}_y^i \in \mathcal{L}_Y} \mathsf{Pr}_{\text{chip}}[\boldsymbol{l}_y^i|k, x] \cdot \log_2 \mathsf{Pr}_{\text{chip}}[k|x, \boldsymbol{l}_y^i].$$

In practice, the real leakage probability density function is unknown, hence the adversary can only approximate the MI based on the adversary's chip model (e.g. using the GT described above). This results in computing the PI introduced by Renauld et al. [19] at Eurocrypt 2011, which is given by:

$$\hat{\mathsf{PI}}(K;X,\boldsymbol{L}) = \mathsf{H}[K] - \sum_{k\in\mathcal{K}} \mathsf{Pr}[k] \sum_{x\in\mathcal{X}} \mathsf{Pr}[x] \cdot \sum_{\boldsymbol{l}_y^i\in\mathcal{L}_Y^t} \mathsf{Pr}_{\text{chip}}[\boldsymbol{l}_y^i|k,x]\cdot\log_2 \hat{\mathsf{Pr}}_{\text{model}}[k|x,\boldsymbol{l}_y^i].$$

Intuitively, the MI quantifies the worst-case security level of a leaking device, while the PI (computed with the "best-available" model) quantifies its best concrete estimate. In the following, we will compute the PI separately for $\boldsymbol{L}^{\text{dyn}}$ or $\boldsymbol{L}^{\text{stat}}$. Besides being applicable to multivariate leakages, one advantage of this metric is that it embeds a test for the quality of the adversary's model, which may avoid some shortcomings of the SNR metric, as will be shown next. Indeed, the PI essentially computes how well the true distribution $\mathsf{Pr}_{\text{chip}}[\boldsymbol{l}_y^i|k,x]$ can be "explained" by the estimated distribution $\hat{\mathsf{Pr}}_{\text{model}}[k|x,\boldsymbol{l}_y^i]$. Thanks to this feature, and in order to obtain accurate PI values, our experiments exploit 10-fold cross-validation as suggested by Durvaux et al. [6] at Eurocrypt 2014.

Note that we considered the PI for a single chip (i.e. we profiled and attacked the same chip), which can be substantially easier than profiling one chip and attacking another one, e.g. in case of variability in the power measurements [19]. The motivation for considering this (worst) case is the same as the one for using the PI metric in general. That is, our goal is to evaluate the security of an implementation independent of the adversary exploiting it, and without relying on the possibly incomplete knowledge of the leakage model. That is, we stick with the standard cryptographic setting where only the key is considered as secret knowledge (whereas considering inaccurate leakage models, although sometimes justified in practice, amounts to relying on security by obscurity).

III. Experimental results

Preliminary results presented at CHES 2014 showed the feasibility of attacks using the static power on FPGA platforms [16]. We now present complementary studies targeting the CMOS S-box implementation described in the previous section.

A. Leakage traces, signal, noise and SNR

As a preliminary step, we represented the 256 mean leakage traces corresponding to the dynamic and static power consumption of our chip (see Figure 2-left, and Figure 3-left, respectively). While the dynamic traces present clearly visible data-dependent power variations, allowing us to detect points-of-interest by visual inspection, this is not the case for the static measurements which exhibit a much smaller signal. These observations are confirmed by computing the respective SNR of these traces' samples, included in the right part of the figures. The SNR of the dynamic power indeed exhibits clear peaks. By contrast, the SNR estimations for the static leakages are more questionable. On the one hand, they are positive for all samples. On the other hand, they look very noisy compared to their dynamic counterpart. Hence, one could naturally suspect that this signal could come from estimation errors in the mean traces $\hat{\mathsf{E}}_i(L_y^i)$, hence creating variance in the term $\hat{\mathsf{var}}_y\left(\hat{\mathsf{E}}_i(L_y^i)\right)$ that would not correspond to exploitable information. Such doubts are in fact typical of the possible shortcomings discussed in [6] for metrics (such as the SNR) that do not allow cross-validation. In the next section, we show evidence that these doubts can be encountered, by trying to confirm these SNRs with a fair(er) PI-based evaluation.

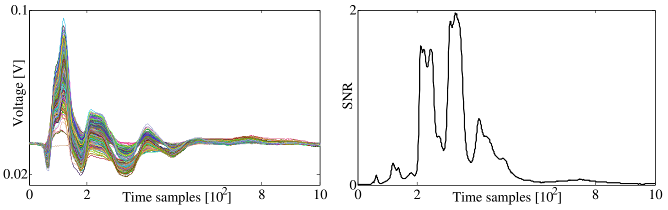

Fig. 2. Dynamic power consumption: mean leakage traces and SNR.

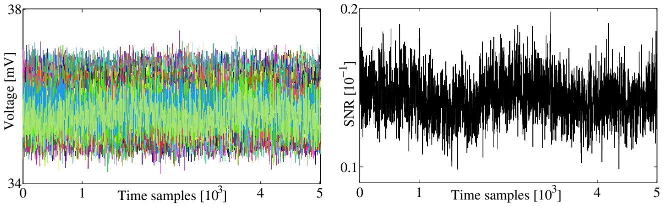

Fig. 3. Static power consumption: mean leakage traces and SNR.

B. Fair (univariate) comparison using the PI

In order to confirm/infirm the previous SNR-based evaluations, we computed PI estimates from Gaussian templates using 10-fold cross-validation. First, we evaluated this PI for the dynamic and static power traces across all time samples. The result of this analysis is given in Figure 4, where we can observe that for the dynamic power (left part of the figure), the PI curves nicely match with the conclusions derived in the previous section, hence confirming the soundness of the SNR metric when the number of measurements available is enough to estimate the signal and noise components accurately. By contrast, for the static measurements (right part of the figure) one can observe that the PI is negative for each time sample, hence confirming that the SNR obtained in the previous section is due to estimation artifacts. On the one hand, this proves that when limited to univariate analysis, the amount of information provided by our static leakage samples is significantly lower than dynamic ones. In our experiments, it was even too small to lead to any exploitable information (i.e. a positive PI). We believe that such an observation (i.e. that the univariate PI is larger for the dynamic leakages than for static ones) generally holds for any current CMOS technology. On the other hand,

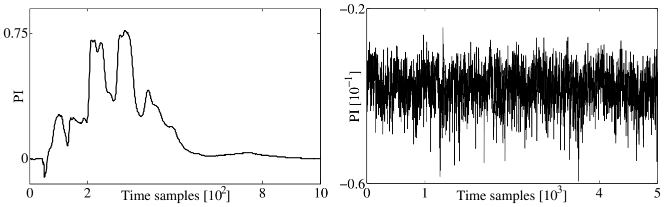

Fig. 4. PI for all time samples: dynamic (left) and static (right) power.

it is also likely that the negative PI comes from a very weak SNR and would simply require more measurements to exhibit exploitable leakage and contrast our conclusions. Therefore, we investigate this scenario in the next subsection.

C. Multivariate analysis

In order to reduce the noise in the static part of the power traces, a natural option is to perform more measurements and to average them. Nevertheless, since we expect the static power consumption to be constant, a much more efficient option is to average these traces across the time samples. We will denote this averaging process with $\mathsf{AVG}(\boldsymbol{l}_y^{\text{stat}})$. It actually corresponds to the strategy exploited by Moradi in [16]. Quite naturally, and in order to keep our comparisons (somewhat) comparable (see the following discussion), it also means that we need to move to a multivariate setting for the dynamic leakages as well. As a starting point, we considered the PCA described in Section II-B for this purpose, and projected our dynamic traces onto a 1-dimensional principal subspace $\mathsf{PCA}(\boldsymbol{l}_y^{\text{dyn}})$. The result of these multivariate attacks are given in Figure 5. As expected, the left part of the figure shows that PCA leads to an improved PI (compared to the one obtained with the best Point-of-Interest). More importantly, we observe that averaging greatly increases the amount of information leaked through the static power consumption in the right part of the figure. In our setting, the averaging was limited to 5,000 samples (i.e. the size of our static leakage traces) but was enough (a) to reach positive PI values (meaning exploitable leakages), hence confirming the results of Moradi, and (b) to exhibit some saturation in the information gain provided by the averaging. We next discuss the interpretation of these results.

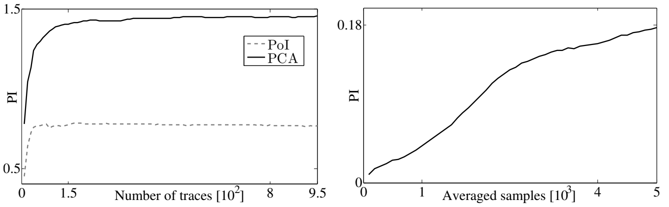

Fig. 5. Left: PI for the dynamic power, computed from the best PoI and PCA. Right: PI for the static power after averaging across time samples.

D. Interpretation/discussion

In view of the results of Figure 5, a natural question to ask is "*how far can we go?*". Clearly, we observed that the PI of the static leakage traces can be increased thanks to averaging. But in our setting, it remained lower than its dynamic counterpart. The most important point to acknowledge here is that in theory, the PI of the static power could reach a full leakage of the secret key – and in fact, this only depends on the quality of the measurement setup. Indeed, the most important feature of static leakages is that their noise can be arbitrarily reduced thanks to intra-trace averaging (whereas dynamic leakages can mostly benefit from inter-trace averaging). So the value of the PI eventually reached in this case only depends on the amplitude of the static signal and quality of the acquisition devices (with typically 8 bits to 12 bits of quantization). Our setup was limited to a PI of approximately 0.2. Designing better setups (e.g. with low-noise high-bandwidth amplifiers) able to improve this value is an interesting open problem.

Yet, we would like to insist that such improvements have limited impact on our main conclusion. First, one should note that a similar problem generally holds for dynamic leakages as well. For example, higher PIs could be reached for this part of the measurements, e.g. by trying different dimensionality reduction techniques, or by better exploiting time-based leakages due to glitches [9] (e.g. through synchronization [22]). But in fact, it will anyway remains that for (a) univariate attacks, dynamic power will remain the target of choice, because of larger signal, and (b) for multivariate attacks, static power provides a very relevant alternative for adversaries controlling the clock frequency, because of (possibly much) smaller noise. For concreteness, the next subsection exhibits a couple of attack results (in order to translate the previous PI values into a number of measurements to recover the key). We then conclude the paper by briefly discussing the impact of our findings for algorithmic countermeasures such as masking.

E. Security analysis based on worst-case attacks

Previous experiments quantified the dynamic and static information leakages by means of the PI, which allows comparing them. In this section, we complement this analysis by providing the success rates of several attacks exploiting the same models as used to estimate the PI. For this purpose, we used sets of $N_p = 100$ traces to profile the template of each of our 256 intermediate values $y = x \oplus k$. We performed both GT and CPA attacks. For the dynamic traces, we additionally investigated the efficiency of PCA (compared to the best PoI). For the static traces, averaging was used anyway (since necessary to obtain non negligible success rates). Each success rate curve was computed from a set of 1,000 experiments (randomly sampled from our set of traces). These results are reported in Figure 6 from which a couple of observations can be extracted. In the left part of the figure (i.e. for dynamic power), we mainly see the improved success rate enabled by PCA. In the right part of the figure, we further confirm that the number of traces to perform successful key recoveries thanks to static power is increased by a factor corresponding to its reduced PI. Eventually, we note that GT and profiled CPA provide very similar results in both cases, as predicted by [13][1].

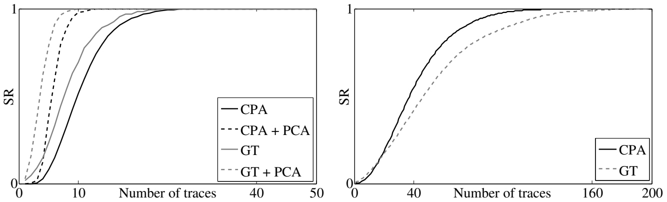

Fig. 6. Success rates of GT attacks and profiled CPA attacks exploiting dynamic power consumption (left) and static power consumption (right).

IV. Conclusions

The main conclusion of this work is that static leakage can be a useful information source for side-channel adversaries able to control the clock signal. In these cases, reducing the frequency can make the static parts of the traces (as depicted

[1] The slight performance penalty of GT in the right part of the figure is due to the limited number of measurements in this case, which did not allow perfectly accurate estimates of the noise variance (that is not exploited by CPA).

in Figure 1) arbitrarily long, hence allowing to use these long traces to reduce/remove the noise via averaging. While this can be an issue in unprotected implementations, we note that the impact of static leakages can become much more critical, e.g. in (Boolean) masked implementations. Indeed, the main goal of masking is to "amplify" the noise. Information theoretic plots (such as detailed in [21]) provide a simple intuition for this amplification. As illustrated in Figure 7, masking leads to an exponential security increase (compared to an unprotected implementation) given that the noise is large enough. By contrast, for low noise levels, such a protection is essentially ineffective (because sufficient information can be gathered on the different shares), leading to a plateau'ed region in the information theoretic curves. As a result, static leakage signal averaging could be directly exploited to move from the "effective masking" zone to the "ineffective" one. Note that attacks against masked implementations exploiting static leakage have already been put forward by Moradi in [16]. An interesting scope for further research would be to study such attacks more quantitatively (as we did in this paper for unprotected implementation). In particular, it would be interesting to put forward an example of masked implementation where static leakage allows significant reductions of the side-channel attacks' complexity. As mentioned in introduction, such a risk can be extended to any algorithmic countermeasure requiring high noise levels to be effective. Eventually, it is worth recalling that controlling a cryptographic implementation usually allows adversaries to take advantage of other physical defaults, fault insertion being a typical example of this concern (see [4] for examples of vulnerabilities in a similar technology as used in this paper). We naturally make no claim regarding what is the best attack path in general in such challenging contexts.

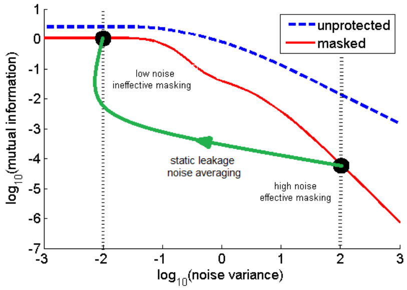

Fig. 7. Potential impact of static leakage averaging on masking.

Acknowledgements. François-Xavier Standaert is a research associate of the Belgian Fund for Scientific Research (FNRS-F.R.S.). This work has been funded in parts by the European Commission through the ERC project 280141 (CRASH).

References

[1] Massimo Alioto, Simone Bongiovanni, Milena Djukanovic, Giuseppe Scotti, and Alessandro Trifiletti. Effectiveness of leakage power analysis attacks on DPA-resistant logic styles under process variations. *IEEE Trans. on Circuits and Systems*, 61-I(2):429–442, 2014.

[2] Massimo Alioto, Luca Giancane, Giuseppe Scotti, and Alessandro Trifiletti. Leakage power analysis attacks: A novel class of attacks to nanometer cryptographic circuits. *IEEE Trans. on Circuits and Systems*, 57-I(2):355–367, 2010.

[3] Cédric Archambeau, Eric Peeters, François-Xavier Standaert, and Jean-Jacques Quisquater. Template attacks in principal subspaces. In Louis Goubin and Mitsuru Matsui, editors, *CHES*, volume 4249 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2006.

[4] Alessandro Barenghi, Cédric Hocquet, David Bol, François-Xavier Standaert, Francesco Regazzoni, and Israel Koren. A combined design-time/test-time study of the vulnerability of sub-threshold devices to low voltage fault attacks. *IEEE Trans. Emerging Topics Comput.*, 2(2):107–118, 2014.

[5] Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi. Template attacks. In Burton S. Kaliski Jr., Çetin Kaya Koç, and Christof Paar, editors, *CHES*, volume 2523 of *Lecture Notes in Computer Science*, pages 13–28. Springer, 2002.

[6] François Durvaux, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. How to certify the leakage of a chip? In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT*, volume 8441 of *Lecture Notes in Computer Science*, pages 459–476. Springer, 2014.

[7] Jacopo Giorgetti, Giuseppe Scotti, Andrea Simonetti, and Alessandro Trifiletti. Analysis of data dependence of leakage current in CMOS cryptographic hardware. In Hai Zhou, Enrico Macii, Zhiyuan Yan, and Yehia Massoud, editors, *ACM Great Lakes Symposium on VLSI*, pages 78–83. ACM, 2007.

[8] Basel Halak, Julian P. Murphy, and Alex Yakovlev. Power balanced circuits for leakage-power-attacks resilient design. *IACR Cryptology ePrint Archive*, 2013:48, 2013.

[9] Dina Kamel, Cedric Hocquet, François-Xavier Standaert, Denis Flandre, and David Bol. Glitch-induced within-die variations of dynamic energy in voltage-scaled nano-CMOS circuits. In *European Solid-State Circuits Conference, ESSCIRC 2010*, pages 518–521. IEEE, 2010.

[10] Dina Kamel, François-Xavier Standaert, and Denis Flandre. Scaling trends of the AES s-box low power consumption in 130 and 65 nm CMOS technology nodes. In *ISCAS*, pages 1385–1388. IEEE, 2009.

[11] Lang Lin and Wayne Burleson. Leakage-based differential power analysis (LDPA) on sub-90nm CMOS cryptosystems. In *ISCAS*, pages 252–255. IEEE, 2008.

[12] Stefan Mangard. Hardware countermeasures against DPA ? a statistical analysis of their effectiveness. In Tatsuaki Okamoto, editor, *CT-RSA*, volume 2964 of *Lecture Notes in Computer Science*, pages 222–235. Springer, 2004.

[13] Stefan Mangard, Elisabeth Oswald, and François-Xavier Standaert. One for all - all for one: unifying standard differential power analysis attacks. *IET Information Security*, 5(2):100–110, 2011.

[14] Nele Mentens, Lejla Batina, Bart Preneel, and Ingrid Verbauwhede. A systematic evaluation of compact hardware implementations for the Rijndael S-Box. In Alfred Menezes, editor, *CT-RSA*, volume 3376 of *Lecture Notes in Computer Science*, pages 323–333. Springer, 2005.

[15] Silvio Micali and Leonid Reyzin. Physically observable cryptography (extended abstract). In Moni Naor, editor, *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, 2004, Proceedings*, volume 2951 of *Lecture Notes in Computer Science*, pages 278–296. Springer, 2004.

[16] Amir Moradi. Side-channel leakage through static power - should we care about in practice? -. In *CHES*, volume ?? of *Lecture Notes in Computer Science*, page ?? Springer, 2014, to appear.

[17] Amir Moradi, Oliver Mischke, Christof Paar, Yang Li, Kazuo Ohta, and Kazuo Sakiyama. On the power of fault sensitivity analysis and collision side-channel attacks in a combined setting. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, volume 6917 of *Lecture Notes in Computer Science*, pages 292–311. Springer, 2011.

[18] Emmanuel Prouff and Matthieu Rivain. Masking against side-channel attacks: A formal security proof. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, volume 7881 of *Lecture Notes in Computer Science*, pages 142–159. Springer, 2013.

[19] Mathieu Renauld, François-Xavier Standaert, Nicolas Veyrat-Charvillon, Dina Kamel, and Denis Flandre. A formal study of power variability issues and side-channel attacks for nanoscale devices.

In Kenneth G. Paterson, editor, *EUROCRYPT*, volume 6632 of *Lecture Notes in Computer Science*, pages 109–128. Springer, 2011.

[20] François-Xavier Standaert, Tal Malkin, and Moti Yung. A unified framework for the analysis of side-channel key recovery attacks. In Antoine Joux, editor, *EUROCRYPT*, volume 5479 of *Lecture Notes in Computer Science*, pages 443–461. Springer, 2009.

[21] François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard. The world is not enough: Another look on second-order DPA. In Masayuki Abe, editor, *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, volume 6477 of *Lecture Notes in Computer Science*, pages 112–129. Springer, 2010.

[22] Jasper G. J. van Woudenberg, Marc F. Witteman, and Bram Bakker. Improving differential power analysis by elastic alignment. In Aggelos Kiayias, editor, *Topics in Cryptology - CT-RSA 2011 - The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18, 2011. Proceedings*, volume 6558 of *Lecture Notes in Computer Science*, pages 104–119. Springer, 2011.

[23] Nicolas Veyrat-Charvillon, Marcel Medwed, Stéphanie Kerckhof, and François-Xavier Standaert. Shuffling against side-channel attacks: A comprehensive study with cautionary note. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 740–757. Springer, 2012.

[24] Nianhao Zhu, Yujie Zhou, and Hongming Liu. Counteracting leakage power analysis attack using random ring oscillators. *Sensor Network Security Technology and Privacy Communication System (SNS & PCS)*, 2013:74–77, 2013.