Let's make it Noisy: A Simulation Methodology for adding Intrinsic Physical Noise to Cryptographic Designs

Kashif Nawaz, Léopold Van Brandt, François-Xavier Standaert and Denis Flandre
ICTEAM institute, Université catholique de Louvain, Belgium

***Abstract*—Noise in digital circuits for the sake of performance has always been minimized in typical designs. However, for cryptographic applications, increased noise could be beneficial. It can be used effectively to reduce the mathematical SNR (signal-to-noise ratio) further and make it more difficult for the adversary to gather useful information from the side channel leakage data. In this paper, we introduce a methodology to exploit the intrinsic physical noise (i.e. flicker and thermal noise) at the circuit level and use the obtained values in a relevant cryptographic context. Our simulations show that the calculated cryptographic noise values are in close agreement with the noise levels extracted from noisy distributions using transient noise analysis. Consequently, this noise is shown to increase with the number of transistors or the supply voltage.**

I. Introduction

Side channel attacks, such as differential power analysis (DPA) [1], exploit the leakage signal from a cryptographic device to guess the secret keys. Logic styles such as differential dual rail, have been proposed to reduce the signal value, but do not scale well with technology [2] e.g, moving from 65nm to 28nm nodes. Existing countermeasures against side-channel analysis such as shuffling (adding noise in time domain) and masking (or algorithmic noise, adding noise in the amplitude domain) work well only if the SNR has been sufficiently reduced. A possible option to reduce the SNR even further would be to increase the intrinsic physical noise coming from the transistors themselves. In this paper, we explore the design of noisy CMOS implementations, and propose a methodology to exploit the *intrinsic* physical MOSFET noise allowing designers to derive an insight of the impact through gate-level simulations. More specifically, we investigate a methodology to answer the following questions, i.e. can the MOSFET noise be quantified from a cryptographic perspective, how does it scale up with the number of transistors, and how does it behave with low voltage implementations?

This paper is divided into 5 sections. We first review the state-of-the-art existing for this work, then in Section III we discuss our methodology to introduce noise sources, their simulation cost and budget. In Section IV we discuss the results of our methodology. In section V, we present a statistical analysis of our traces and finally wrap up in Section VI with conclusion and perspectives.

II. State-of-the-art: A Review

A quick state-of-art metric for quantifying the cryptographic leakage from a side channel (a leaky implementation) arises from the classical univariate metric, the Signal-to-Noise ratio. In this paper, we use Mangard's SNR defined in [3] as:

$$\mathrm{SNR} = \frac{\hat{\mathsf{var}}_x(\hat{\mathsf{E}}_i(L_x^i))}{\hat{\mathsf{E}}_x(\hat{\mathsf{var}}_i(L_x^i))}, \tag{1}$$

where $\hat{\mathsf{E}}$ (resp. $\hat{\mathsf{var}}$) denotes the sample mean (resp. variance) operator and L the leakage. In our following simulations, this SNR will be computed for noise-based traces of the current consumption of the digital gates at the supply rail as a function of time, denoted as $I_{DD}(t^*)$ and would include the noise coming from physical *intrinsic* MOSFET noise sources. Using eqn (1), the signal is the "useful" part that is obtained by the adversary as a measure of the information leakage. The lower the signal value, the lower is the "perceived" side-channel leakage. The maximum signal, as a metric to quantify the leakage, in case of noiseless simulations [2], has been used by the authors to show the scaling trends of the signal with respect to technology scaling from 65nm bulk to 28nm FDSOI for standard CMOS and dual rail differential logic styles. As reported, dual rail logic styles lose their advantage with technological scaling and standard CMOS continues to be the design of choice with respect to technological scaling. For CMOS, with lowering of the supply voltage V_{DD}, in 28nm, the signal value reduces further compared to 65nm technology, which is highly desirable for the implementation of cryptographic algorithms. These comparisons justify the choice of standard CMOS implementations in scaled down technologies over dual-rail styles and provides the necessary motivation for the design of noisy CMOS implementations.

III. Transient Noise Analysis: A simulation methodology

A. Target Designs

Conventional noise analyses in circuit designs mostly use the ac or the harmonic based approaches. However, in digital cryptographic applications, where each point in a transient run is potentially a source of information leakage (from an adversary perspective), it makes it worth analyzing the effect of noise on *each* time sample. From a cryptographic perspective, this is a univariate analysis compared to a bivariate analysis (where multiple time-samples are used). In the scope of this work, we focus on the univariate aspect only. The time sample with the highest value of signal (or SNR) is chosen as the point-of-interest (POI). Using the Transient noise simulations in Eldo software (provided by Mentor Graphics)

[4], we provide a methodology to introduce the *intrinsic* physical noise sources (i.e. noise coming from the transistors themselves and *not* externally) and calculate the resulting "cryptographic"[1] signal and noise (units of A^2) (as defined in eqn. 1), then compare them to the values obtained from ac simulations and variance calculations.

Our results are based on transient noise simulations in an Eldo environment using a 28nm FDSOI PDK (process design-kit) provided by an industrial foundry. The sizing of the transistors is kept minimum to maximize the noise produced (flicker noise especially). We use a simple 2-bit XOR, a 4-bit PRESENT Sbox and an 8-bit AES S-box, all custom designed using Cadence Virtuoso software, to show the scaling trends of the signal and noise w.r.t the supply voltage and the number of transistors.

B. Simulation settings

All the 3 designs are simulated with the Transient Noise analysis built in Eldo (called by the *.noisetran* command) upto 100 transient noise runs. The noise sources correspond to the physical flicker and thermal noises intrinsic to the MOS transistors. They are generated by Eldo in the frequency bandwidth specified by the input parameters of the transient noise analysis. In our simulations, we chose f_{min} = 1/T and f_{max} = 1/2*dt, where dt is the minimum time step being used by the simulator or specified by the user. The input data signals to the circuits are a recurring 0 to an arbitrary input for 4 transitions (for the 2-input XOR), 16 and 256 transitions (for the 4-bit PRESENT and 8-bit AES S-boxes repectively) at a clock frequency of 10 MHz. All simulations are done at 298K, TT corner and for a V_{DD} range from 0.5V to 1V.

C. Simulation cost and Budget

In this section, we investigate the cost of our methodology and quantify the total budget, both in terms of CPU runtime and number of runs required to obtain convergent metrics which estimate the "crypto" signal and noise. The noise transient simulations are indeed well known to be time-and memory-intensive [5]. Basically, our simulation budget can be stated as

$$N_{traces} \cdot \frac{T}{\mathrm{d}t}, \tag{2}$$

where N_{traces} is the number of traces, i.e. noise realizations, T is the simulation duration for one trace, and $\mathrm{d}t$ is the time step. These parameters correspond to `NBRUN`, `TSTOP` − `TSTART` and `HMAX` of Eldo `NOISETRAN` command [4], respectively. The number of samples for each trace is given by

$$N_s = \frac{T}{\mathrm{d}t}. \tag{3}$$

Since the *.noisetran* analysis specifies a number of input parameters, it is of importance to analyze the effect of each of these on our calculated values of signal and noise.

1 We first analyze the impact of the f_{max} parameter specified in the *.noisetran* analysis on the CPU runtime, as shown in figure 1. Choosing a larger f_{max} also means increasing very significantly the CPU runtime, as shown in figure 1(bottom). We then propose to choose an optimum value of the f_{max}, which trades-off CPU simulation time and convergent signal/noise values. As the fmax parameter is primarily defined by the sampling time, keeping in mind the Nyquist sampling criteria, we see that oversampling may lead to higher obtained noise values, however, at the expense of increased CPU runtime. Moreover, oversampling would simply allow the "cryptographic" adversary to average over a larger number of time samples, thus effectively reducing the noise by averaging and hence, increasing the SNR, which is contrary to our requirements. Hence, an optimum value of fmax has to be chosen which minimizes the simulation time.

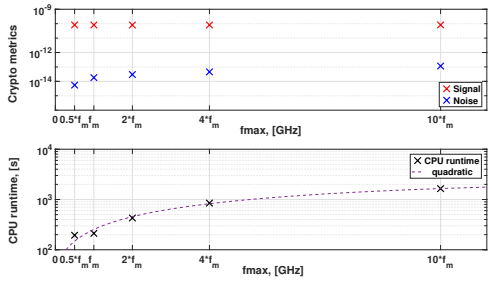

Figure 1: Impact of f_{max} parameter on the "Cryptographic" *Signal* and *Noise* and the CPU runtime for an XOR gate at V_{DD}=0.9V and f_m = 12.5 GHz

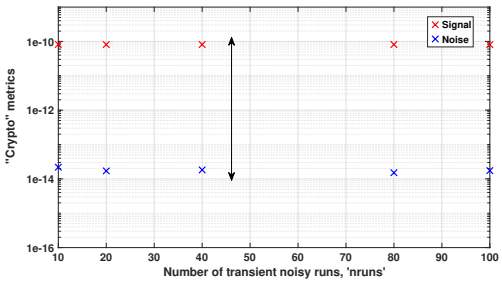

Figure 2: Impact of *nruns* parameter on the "Cryptographic" *Signal* and *Noise* for an XOR gate at V_{DD}=0.9V

2 The simulation time also depends on the number of transient runs, *nruns* as shown by eqn 2. The greater this value, the longer the simulation time; we also define a tolerance value η which is defined as

$$\eta = \frac{\sigma}{\mu} \tag{4}$$

where σ is the standard deviation across the observations and μ is the mean for the observed values. For the purposes of our simulations, we choose an η tolerance value of $\lessapprox 15\%$. We observe in figure 2 that even with an increasing number of runs, the values of "crypto" signal and noise calculated remain well within our tolerance levels. This justifies the usage of lower number of transient noisy runs to minimize the simulation run-time.

[1] We use the term cryptographic and crypto interchangeably, they both denote the one and same thing

IV. Results of the Transient Noise Analysis

In this section, we now discuss the results of our ongoing work. We aim to show our present results for a 2-input XOR, a 4-bit Sbox using the PRESENT cryptographic implementation and an 8-bit AES Sbox implementation. These include a total of 12, 684 and 1884 transistors respectively.

A. Effect on the Cryptographic Signal and Noise

Thanks to the 40 transient noise runs, we are now able to calculate the maximum signal and the maximum noise, (as per equation 1). Figures 3 and 4 show the scaling of the maximum signal and noise for a range of V_{DD}, i.e, from 0.5V to 1V. We can make the following observations from the above 2 plots,

1 By increasing the supply voltage, V_{DD}, the value of the maximum signal and noise increase. This can be explained by the fact that as the V_{DD} increases, the power consumption, P_{dyn} increases (hence the increase in I_{ON}) which increases the *Signal* value. The increase in the "crypto" *noise* could be explained by the increase in the thermal noise which increases with the increase of the I_{ON} current value. Reciprocally, signal and noise decrease with V_{DD}. We can observe the signal decrease is faster than the noise decrease which is of interest for our purpose.

2 For a particular, V_{DD}, the signal (and the noise) increase with the increase in design complexity, i.e as the number of transistors increases for the given circuit (e.g. moving from a 2-bit XOR to a 4-bit PRESENT Sbox to an 8-bit AES Sbox)

The increase in the signal can be modeled by the following relation

$$\bar{S}^{circuit}_{V_{DD}} = S^{XOR}_{V_{DD}} N_T^{\beta} \tag{5}$$

where $\bar{S}^{circuit}_{V_{DD}}$ is the signal for the target cricuit, $S^{XOR}_{V_{DD}}$ is the signal produced by a 2-bit XOR for the same supply voltage, V_{DD}, N_T is the ratio of the increase in the number of transistors w.r.t a 2-bit XOR and β is a technology factor which varies $0.4 < \beta < 1.5$ for most circuits and depends on the value of the supply voltage, V_{DD}.

The increase in the noise can be modeled as

$$\bar{N}^{circuit}_{V_{DD}} = N^{XOR}_{V_{DD}} N_T^{\alpha} \tag{6}$$

where $\bar{N}^{circuit}_{V_{DD}}$ is the noise for the target circuit, $N^{XOR}_{V_{DD}}$ is the noise calculated for a 2-bit XOR at the same V_{DD}, N_T is the ratio of the number of transistors w.r.t a 2-bit XOR and α is a parameter which scales with the supply voltage and is $\lessapprox 2$ for most circuits and depends on the value of the supply voltage, V_{DD}. Consequently, we observe that noise increases faster with the number of transistors than the signal. This could be related to the fact that the intrinsic MOSFET noise sources are not correlated and hence add on I_{DD}, whereas the signal is more proportional to the number of circuit branches connected to V_{DD}.

Since the calculated "cryptographic" noise is essentially a mean of the variance across different inputs for *nruns* number of traces, we should be able to relate this noise to the histogram of the measured current. We explore this in the next section.

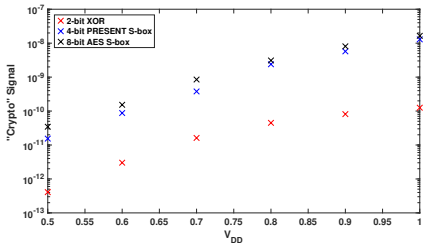

Figure 3: Scaling of "Cryptographic" *Signal* as a function of V_{DD} for different circuits

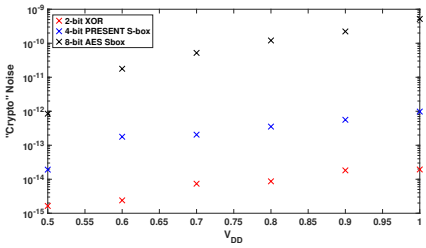

Figure 4: Scaling of "Cryptographic" *Noise* as a function of the number of transistors, N_T for different V_{DD}

V. Statistical properties of the supply current noise of the XOR gate

In this section, we study the *first-order statistics* of the noise present in the supply current for one of the above mentioned circuits of interest, i.e. the XOR gate.

Fig. 5 contains noise traces for the set of parameters $N_{traces} = 40$, T = 800ns and dt = 40ps for a total budget of $\sim 8 \times 10^5$ for a supply V_{DD} = 0.5V

A. Statistical characterization of the static region

The static region indicated in Fig. 5 is useful to get insight on the noise behaviour within the circuit and how it affects the supply current. Input voltages of the gate are fixed, and so are all the averages of the branch currents and node voltages within the circuit (since there are noise fluctuations). Hence, the supply current noise is treated as a *wide-sense stationary*

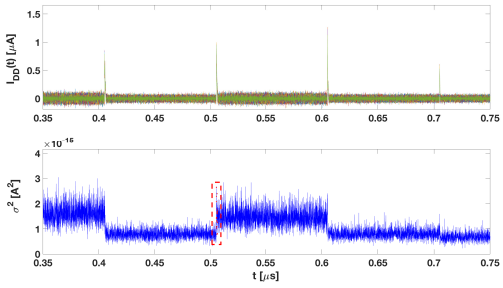

Figure 5: Zoom on the dynamic and static regions (top) and associated variance plot based on all the 40 traces (bottom). Noise in the static region is shown to follow a stationary Gaussian distribution.

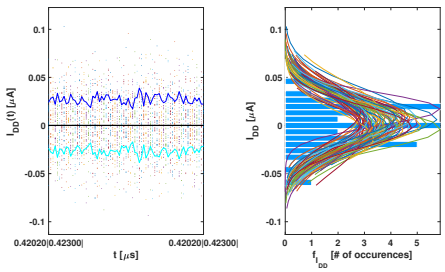

Figure 6: Histogram construction for time t = 0.4202μs - 0.4230μs within the static region in Fig. 5 (left). Extracted Gaussian distribution is also shown (right).

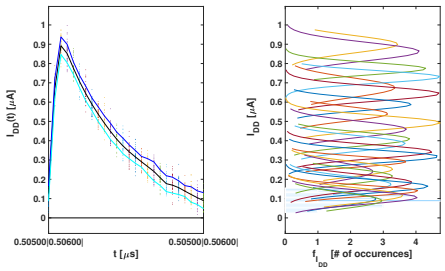

Figure 7: Histogram construction for time t = 0.505μs - 0.506μs within the dynamic region marked red in Fig. 5(below). Extracted Gaussian distribution mixture is also shown(right).

stochastic process, for which a complete definition can be found in [6]. Especially, the probability density function (pdf) does not depend on t:

$$f\left(I_{DD}\left(t\right)\right) = f\left(I_{DD}\right), \tag{7}$$

and the variance is also independent of t:

$$\sigma\left(t\right) = \sigma. \tag{8}$$

As a consequence, every sample $I_{DD}\left(t^*\right)$ at some time t^* of every trace is understood as a realization of one single random variable I_{DD}.

B. Challenges regarding the dynamic region

In order to accurately capture the dynamic region behaviour, we plot the time-varying histograms of the very *narrow* dynamic region enclosed by a red rectangle in figure 5. The supply current noise now is a *nonstationary* stochastic process. Its distribution is explicitly time-dependent:

$$f\left(I_{DD}\left(t\right)\right) = f\left(I_{DD}, t\right), \tag{9}$$

as well as the variance $\sigma\left(t\right)$. Since each point in the dynamic region is non-stationary, we observe the *time-varying* histograms for each sample to extract the mean and the variance and show the *envelope* of the $\pm 1\sigma$ over the mean trace as shown in figure 7

Fortunately, our main goal is not the histograms themselves but the extracted variances, for which rough estimations are sufficient. In this case, this is achieved by performing a nonlinear fitting of the granular histogram. Using the extracted σ values from the distributions above and comparing them with the mathematical "cryptographic" noise values calculated by eqn (1), we obtain a good matching between the data, thus validating the fact that the noise present in the mathematical calculations can indeed be traced back to the results of the transient noise analyses including the MOSFET noise sources. This is of importance to further expand, optimize and exploit the methodology of our on-going work.

VI. Conclusion and Open Questions

In this case study, we have for the first time, to the best of our knowledge, proposed a methodology to analyze and simulate the addition of *intrinsic* MOSFET physical noise on cryptographic implementations using transient noise simulations to compute the first-order univariate security metrics. This can be used to discuss how the physical noise sources from the MOSFETs can be used for effectively deriving the cryptographic SNR value, especially at design stages of a cryptographic implementation. Longer transient runs (higher fmax), which add more noise, however come at the expense of increased simulation run times (high CPU time). The complexity would further increase with the number of gates in case of a full cryptographic implementation such as the AES or the PRESENT block cipher. At this stage our results, while suggesting that the use of intrinsic physical noise in MOSFETs to add more cryptographic noise is an effective method (since such noise sources are predicted to increase significantly with further technology scaling and voltage), its simulation and extension to correlated noise sources especially for larger circuits needs to be optimized and remains an open research question.

Acknowledgments. This work has been funded in parts by the ARC Project NANOSEC. François-Xavier Standaert is a research associate of the Belgian Fund for Scientific Research.

References

[1] P. C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, 1999, pp. 388–397. [Online]. Available: http://dx.doi.org/10.1007/3-540-48405-1_25

[2] K. Nawaz, D. Kamel, F.-X. Standaert, and D. Flandre, "Scaling trends for dual-rail logic styles against side-channel attacks: A case-study," in *Constructive Side-Channel Analysis and Secure Design - 8th International Workshop, COSADE 2017, Paris, France, April 13-14, 2017, Revised Selected Papers*, 2017, pp. 19–33.

[3] S. Mangard, "Hardware countermeasures against DPA ? A statistical analysis of their effectiveness," in *Topics in Cryptology - CT-RSA 2004, The Cryptographers' Track at the RSA Conference 2004, San Francisco, CA, USA, February 23-27, 2004, Proceedings*, 2004, pp. 222–235. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24660-2_18

[4] Mentor Graphics Corporation, "Eldo User's Manual, Release AMS 2008.2," 2008.

[5] A. Demir and A. Sangiovanni-Vincentelli, "Time-domain non-monte carlo noise simulation," in *Analysis and Simulation of Noise in Nonlinear Electronic Circuits and Systems*. Springer, 1998, pp. 113–161.

[6] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, ser. McGraw-Hill Series in Electrical Engineering. McGraw-Hill, 1991.