Information Theoretic and Security Analysis of a 65-nanometer DDSLL AES S-box

Mathieu Renauld*, Dina Kamel,
François-Xavier Standaert**, Denis Flandre.

UCL Crypto Group, Université catholique de Louvain.
Place du Levant 3, B-1348, Louvain-la-Neuve, Belgium.

Abstract. In a recent work from Eurocrypt 2011, Renauld et al. discussed the impact of the increased variability in nanoscale CMOS devices on their evaluation against side-channel attacks. In this paper, we complement this work by analyzing an implementation of the AES S-box, in the DDSLL dual-rail logic style, using the same 65-nanometer technology. For this purpose, we first compare the performance results of the static CMOS and dual-rail S-boxes. We show that full custom design allows to nicely mitigate the performance drawbacks that are usually reported for dual-rail circuits. Next, we evaluate the side-channel leakages of these S-boxes, using both simulations and actual measurements. We take advantage of state-of-the-art evaluation tools, and discuss the quantity and nature (e.g. linearity) of the physical information they provide. Our results show that the security improvement of the DDSLL S-box is typically in the range of one order of magnitude (in terms of "number of traces to recover the key"). They also confirm the importance of a profiled information theoretic analysis for the worst-case security evaluation of leaking devices. They finally raise the important question whether dual-rail logic styles remain a promising approach for reducing the side-channel information leakages in front of technology scaling, as hardware constraints such as balanced routing may become increasingly challenging to fulfill, as circuit sizes tend towards the nanometer scale.

1 Introduction

Side-channel attacks are an important concern for the security of cryptographic devices. Since their apparition in the late 1990s, a significant attention has been paid to the development of various solutions to prevent them, at different abstraction levels (e.g. hardware, algorithmic, protocol). In this paper, we are concerned with technological countermeasures, usually denoted as dynamic and differential logic (DDL). DDL aims to solve the side-channel issue directly at the circuit level. For this purpose, such logic styles typically ensure that the switching activity of an implementation is independent of the data that it manipulates. However, despite a constant switching activity, small data-dependent variations in the current traces can generally be observed, e.g. due to the unbalanced capacitances

* PhD student supported by the Walloon region SCEPTIC project.
** Associate researcher of the Belgian Fund for Scientific Research (FNRS-F.R.S.).

of the differential nodes and their interconnections. As a result, and similarly to other countermeasures against side-channel attacks, the design of DDL is mainly a tradeoff between performance and security. That is, logic styles such as SABL [26], WDDL [27], DyCML [1, 13], MCML [4] or MDPL [19] (to name a few), have been introduced as different attempts to best reach the security objectives of DDL while limiting their performance overheads.

Performance evaluation of integrated circuits is a relatively well understood topic. Figures of merit such as the area, the delay, the area-delay product or the power consumption of an implementation can be used, depending on the applications. Additionally, one also cares about the design facilities. In this respect, circuits that can be taped out from standard cell libraries (such as WDDL, MCML, MDPL) offer a flexibility advantage compared to full custom logic styles (such as SABL, DyCML). This flexibility naturally comes with a cost, as using full custom logic offers more freedom for the designer to limit the information leakages.

By contrast, evaluating the resistance against side-channel attacks is more challenging, and many different tools have been introduced for this purpose in the literature. For example, early works on DDL used specific criteria such as the normalized energy deviation (NED) [13, 26, 27]. Some other logic styles have been analyzed using dedicated attacks, e.g. based on the correlation coefficient [19, 21]. The main issue with such evaluation tools is that (at least in theory), they can possibly lead to a false sense of security, because they do not consider a worst-case scenario. The evaluation framework proposed at Eurocrypt 2009 was consequently introduced in order to relax this limitation [25]. It suggests to evaluate side-channel attacks in two steps. First, a (profiled) information theoretic analysis is performed, in order to quantify the physical leakages, independently of the adversary who exploits them. Second, a security analysis is performed, in order to measure the effectiveness of various (e.g. non-profiled) distinguishers. When designing countermeasures or logic styles, it is the information theoretic analysis that is most revealing, as it provides a (more) objective measure of their quality[1]. Unfortunately, while such an analysis has already been successfully applied in the case of different software implementations, no results have been published in the case of DDL, except evaluations based on simulated experiments, of which the practical relevance was left as an open problem [14].

Besides the development and evaluation of new countermeasures, the scaling of microelectronic technologies towards the nanometer scale also has a significant impact on the security of cryptographic devices. For example, it was recently observed that process variations imply changes in the typical models (based on the Hamming weight/distance) used in non-profiled side-channel attacks [12, 22]. As a result, distinguishers performing "on-the-fly" estimations of the leakage models have gained a particular interest. As discussed in [5], Schindler et al.'s stochastic

[1] Naturally, the objectiveness of this evaluation still depends on the accurate estimation of the leakage probability density functions. The expectation in [25] is that it is significantly simplified by using profiled attacks rather than non-profiled ones. Yet, as will be discussed Section 4.3, our analysis still relies on certain assumptions.

approach [23] is a very convenient tool for this purpose, when the actual leakage function can be approximated by a linear function of the target bits in the attack. By contrast, as discussed in [28], none of the non-profiled distinguishers used so far, including the ones based on Mutual Information Analysis [6], can perform successful key recoveries when this function becomes highly non-linear. Hence, it is an interesting problem to determine whether such highly non-linear leakage functions can be observed in practice, e.g. for dual-rail logic styles.

The present paper brings two contributions related to this state-of-the-art:

First, we report and analyze the performances of a full custom designed AES S-box, implemented in a 65-nanometer Dynamic and Differential Swing Limited Logic (DDSLL) [8]. We compare it with a static CMOS S-box (full custom designed as well) in the same 65-nanometer technology. In both cases, our evaluations are based on simulations *and* measurements of a test chip (which allows us to discuss the relevance of simulations). These experiments put forward an interesting tradeoff, as the DDSLL S-box shows similar area cost as the CMOS one (contrary to most previous similar logic styles), and reduced power consumption. We explain its limited power consumption by a limited swing and its limited area by the exploitation of trees in the DDL. Interestingly, and despite this clear focus on performance, we also show that the resulting implementation has reduced information leakages compared to the CMOS one.

Second, and for the first time, we apply the information theoretic analysis of [25] to a DDL circuit. As the estimation of its leakage distributions turns out to be relatively simple (as the logic style is not masked), this allows us to provide a fair evaluation of its information leakage reduction. In addition to a fully profiled analysis using templates [3], we also pay attention to the linear nature of the leakages. We confirm that the stochastic approach from [23] is a very interesting tool for analyzing this linearity. We also suggest to use it as an informal criteria, reflecting the easiness of performing a successful non-profiled attack. These results allow us to put forward a variety of leakage samples where, depending on the cases: (1) a simple model based on the Hamming weight allows efficient key recoveries, (2) only the on-the-fly stochastic approach, or single-bit DPA attacks using well-chosen bits, are successful, (3) none of the non-profiled attacks attempted was successful (under our measurement constraints). They confirm the importance of profiled information theoretic evaluations if all the available information is to be exploited, in a worst-case security evaluation.

Finally, our results raise important open questions related to the impact of technology scaling on DDL. Informally speaking, the expectation for such logic styles is that they allow reducing the information leakage and to make basic assumptions (such as Hamming weight/distance models) invalid. In this respect, the prevailing intuition for advanced technologies may suggest that this clear advantage over CMOS vanishes as the technologies are shrinking, for two main concurrent reasons. On the one hand, CMOS circuits become harder to attack with side-channel analysis, as discussed in [22]. On the other hand, the constraints of DDL (e.g. the need of properly balanced capacitances) could become

more difficult to fulfill in advanced technologies, because of variability. Nevertheless, our results highlight that well-designed DDL could remain an interesting alternative to CMOS for securing cryptographic hardware, in order to both reduce the information leakage and to increase its non-linearity.

Note that, because of place constraints, the variability issues are not discussed in this paper. However, we mention that the observations made for the static CMOS S-box in [22] essentially hold for the DDSLL one as well.

2 Previous works

This section briefly surveys results related to DDL and side-channel attacks.

The first logic style purposed to prevent side-channel attacks was SABL. It is a full custom logic style, in which the problem of unbalanced intrinsic differential output capacitances is addressed by adding a transistor to each gate, in order to discharge all internal nodes independent of the data. Compared to CMOS, the area of a Kasumi S-box SABL implementation is increased by a factor of 1.8, and its energy per cycle by a factor of 2, in a 0.18μm 1.8V technology in [26].

WDDL was introduced shortly after SABL and aims to emulate the behavior of SABL gates using static CMOS standard cells. An implementation of a WDDL AES coprocessor, in a 0.18μm 1.8V CMOS technology, was proposed in [9]. It costs a 3 times increase in area, a 3.8 times decrease in throughput and a 3.7 times increase of the power consumption at 50 MHz, compared to its static CMOS counterpart. WDDL is also expected to provide less security margins, as it inherits from certain weaknesses of the CMOS library it is based on [14].

DyCML is a full custom, low-swing and self-timed current mode logic. It was introduced independently of power analysis concerns and constitutes an interesting alternative to SABL. SPICE simulation results using a 0.13μm 1.2V CMOS partially depleted SOI technology suggest that DyCML and SABL have similar NED, while the first one shows slightly better performance (e.g. a reduced power consumption) [13]. By combining complex functions into a differential pull-down network (DPDN), such logic styles can also implement cryptographic functionalities with limited circuit size. However, the design of these DPDN may contain unbalanced intrinsic capacitances, hence causing increased information leakage.

MCML is a CMOS current mode logic. It can be seen as a standard cell counterpart to DyCML and has been the focus of significant attention with respect to side-channel attacks [21]. As for WDDL, its main limitation is a significantly increased power consumption and an area increase by a factor around 2.

Finally, MDPL is masked and dual-rail logic style. It was introduced in order to get rid of the need of balanced capacitances in DDL. Experiments performed on a prototype chip showed that this logic style is affected by an "early propagation" effect [18]. It constitutes a good illustration of the difficulty to predict all types of leaking events that can occur in electronic circuits.

Summarizing, and as already mentioned, these previous works propose different tradeoffs between security and efficiency, and generally show large overheads when compared to CMOS. As a result, the next section first tackles the question whether it is possible to design a (full custom) DDL for which the performances better compare to CMOS. For this purpose, we investigate the implementation of a DDSLL AES S-box, that combines a reduced swing (hence, power consumption) and the combination of complex (here, 4-bit) functionalities into DPDN.

3 Performance analysis

The DDSLL logic proposed in [8] is a low-power, dynamic & differential, self-timed, low-swing logic. Figure 1 shows the basic structure of a generic DDSLL gate. It mainly consists of a DPDN to realize the function of the gate, a dynamic current source made of transistors M_1, M_2, a feedback circuit made of transistors M_3, M_5, a precharge circuit featuring transistors M_6, M_7, M_{10} and M_{11}, a latch realized by transistors M_{12}, M_{13} and finally a self-timing buffer which is a simple inverter (transistors M_{14}, M_{15}). The DDSLL logic operates as a typical DDL, with two phases: precharge and evaluation. During precharge, the clock signal Clk_i is low charging the output nodes (out and $\overline{\mathsf{out}}$) to VDD via transistors M_{10} and M_{11}. Meanwhile, node S discharges to GND as transistor M_7 turns on, which subsequently switches off transistor M_3. At the same time, node ENO charges to VDD via transistor M_6, which in turn switches on transistor M_2. However, there is no DC current path from VDD to GND, as transistor M_1 is switched off. Next, during evaluation, the clock signal Clk_i is high, turning on transistor M_1, thus creating a current path from VDD to GND, through the DPDN. Simultaneously, the transistors of the precharge circuit are turned off (M_6, M_7, M_{10} and M_{11}), allowing the DPDN to evaluate. As a result, one of the output nodes will discharge, turning on one of the feedback transistors (M_4, M_5), which in turn charges node S to VDD. Hence, transistor M_3 turns on and starts discharging node ENO to GND. This will cause the dynamic current source to cut off the current supply of the DPDN, thereby limiting the voltage swing of the output node. Also, as node ENO discharges to GND, the output clock signal Clk_{i+1} charges to VDD, via transistor M_{14} of the self-timing buffer circuit, indicating the termination of the evaluation phase of the current block.

The S-box we considered in this work is taken from Mentens et al. [17]. The resulting DDSLL architecture is designed in such a way that complex functions, like the inversion in $GF(2^4)$, can be implemented with 4 DPDN, corresponding to the 4 output bits of this inversion. Such a design choice has clear advantages in terms of area cost, but potentially implies more side-channel leakage. In order to limit this leakage, we applied the methodology described in [15] for the implementation of the DPDN. It essentially exploits binary decision diagrams for choosing the representation of the DPDN that minimizes the power dependencies caused by variations of the number of internal capacitances that are charged/discharged in each cycle. The logic style additionally allows resources sharing (the dynamic current source, parts of the precharge circuit, the feedback

circuit and the self-timing buffer of functions that evaluate at the same time). As illustrated in Figure 1, the internal clock of the DDSLL logic is driven from gate to gate, without any buffer added, and the block responsible of this derivation is considered as part of the logic in our different (simulated and measured) experiments. Hence, all the VDD nodes in the figure are included in these experiments. Note that the balanced routing, that is important for dual-rail logic styles to reduce side-channel leakages, has been hand-made as part of the full-custom S-box design. We checked the capacitors of the differential routes after extraction from the layout and, in the worst cases, found differences of 0.8 fF, corresponding to roughly 10% of imbalance. Eventually, the complete S-box used in the following accounts for 1275 transistors, with a logic depth of 13 gates.

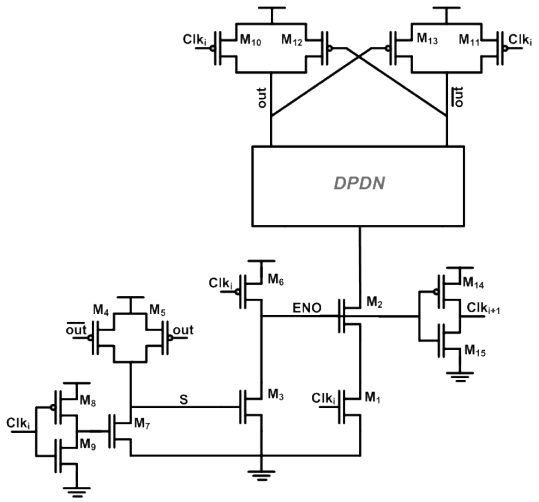

Fig. 1. Schematic of a generic DDSLL gate.

Our test chip includes two versions of the AES S-box: the DDSLL one and a static CMOS one, used for reference and based on the design described in [22]. The chip was fabricated using a low-power 65-nanometer technology. All measurements were done at ambient temperature and using a nominal supply voltage of 1.2 V. None of the S-boxes uses flip flops and both are fed with buffered inputs. The DDSLL S-box additionally has a buffered clock. Each S-box is furnished by its own power supply, which is different than the input buffer power supply. Both S-boxes are full custom designed, with a target of minimizing the area. The measurement setup is standard and monitored the voltage variations over a resistor included in the S-box supply circuit, using a differential probe.

The performances of these S-boxes, obtained from actual measurements of the prototype chip, are summarized in Table 1. One can notice that the area of the DDSLL S-box is only 1.125 times that of the static CMOS S-box. This can be explained by two factors. First, the use of DPDN to implement complex (e.g.

4-bit) functions allows reduced sizes compared to the gate level approaches used in previous SABL or WDDL designs. Next, the logic style allows sharing the dynamic current sources and self-timing buffers of functions evaluating at the same time. In addition, the average power consumption of the DDSLL S-box at 100 kHz is 36 % less than that of the static CMOS. This results from the low-swing logic (combined with the previously mentioned low area). These interesting figures come at the cost of a 2.6× increase in delay, which can be tolerated for low-cost applications (e.g. running at 100 KHz is reasonable for RFID).

Table 1. Comparison between the DDSLL S-box and the static CMOS S-box.

S-box:	Static CMOS	DDSLL
Area	1000 μm^2	1125 μm^2
Avg. power @ 100kHz	128 nW	82 nW
Delay	3 ns	8 ns

4 Side-channel attacks

In this section, we aim to compare the DDSLL logic style with static CMOS implementations, from a side-channel attacks point of view. For this purpose, we will analyze both simulated traces and actual measurements, obtained from the test chip described in the previous section. Following [25], we will first consider a (worst-case) information theoretic analysis and next perform a security analysis, in order to evaluate how efficiently a non-profiled distinguisher can take advantage of the information leakages. We start by introducting our notations, metrics and tools, together with an informal investigation of our leakage traces.

4.1 Notations, metrics & tools

Notations. We use capital letters for random variables, lower cases for samples and sans serif fonts for functions. Let a power trace l be the output of a leakage function L. In our experiments, the leakage function output corresponds to the power consumption of the AES S-boxes we investigate and essentially depends on two input arguments: x and n. These sample values correspond to the discrete random variable X, representing the S-box input, and the continuous random variable N, representing the measurement noise. As a result, the continuous random variable corresponding to the leakage is noted $L(.,.)$, with the parameters written as capital letters if they are variable or as lower cases if they are fixed. For instance, $L(X,N)$ is the variable corresponding to a random input X with a random noise vector N, and $l(x,n)$ is a specific leakage for a fixed input x. Finally, the leakage variable at a specific time sample t is noted L_t. Our experiments considered three types of traces. First, simulated traces obtained from ELDO, to which we added a Gaussian noise, are denoted as $L^1(X,N) = L^{\text{simu}}(X) + N$.

Second, real measurements are denoted as $L^2(X,N) = L^{\text{meas}}(X,N)$. Finally, we also considered a hybrid context, in which ELDO simulations are replaced by the average measurement traces $\overline{L^{\text{meas}}}(X) = \hat{\mathbf{E}}[L^{\text{meas}}(X,N)]$, with $\hat{\mathbf{E}}$ the sample mean operator. This final context is denoted as $L^3(X,N) = \overline{L^{\text{meas}}}(X) + N$.

Information theoretic metric. In order to evaluate the leakage of the CMOS and DDSLL S-boxes, we start by estimating the perceived information:

$$\text{PI}(X;L) = \text{H}[X] - \sum_{x \in \mathcal{X}} \Pr[x] \sum_{l \in \mathcal{L}} \hat{\Pr}_{\texttt{chip}}[l|x] \log_2 \hat{\Pr}_{\texttt{model}}[x|l]. \tag{1}$$

This metric essentially captures how accurately the leakage model used by an adversary (denoted as $\hat{\Pr}_{\texttt{model}}[x|l]$) can predict the actual leakage distribution of a target chip (denoted as $\hat{\Pr}_{\texttt{chip}}[l|x]$). The perceived information has been introduced in [22] in order to capture the fact that in certain contexts (e.g. when inter chip variability is significant), the adversary's model and actual chip's leakage distribution can strongly differ, possibly resulting in a negative perceived information. If the adversary's model is perfect (i.e. exactly corresponds to the chip's one), then the perceived information is equal to the mutual information metric from [25] and accurately captures the worst-case information leakage.

Tools. As a matter of fact, estimating the perceived information essentially requires to perform a good estimation of the leakage distributions. The better this estimation, the more accurate the evaluations. For this purpose, the following section will consider two types of estimation tools: the (Gaussian) template attacks introduced in [3] and the stochastic approach proposed in [23]. The template attacks are useful to estimate the worst-case scenario, with the most powerful adversary in the information theoretic sense. The stochastic approach with a linear model allows to evaluate the linearity of the leakage function.

Template attacks work as follows. First, during a profiling phase, the adversary builds 256 templates, corresponding to the 256 possible input values of the AES S-box. Each of these templates is a Gaussian distribution $\mathcal{N}(l|\hat{\mu}_{x,N}, \hat{\sigma}^2_{x,N})$, defined by two parameters: a sample mean $\hat{\mu}_{x,N}$ and a sample variance $\hat{\sigma}^2_{x,N}$. Profiling just means that the adversary (or evaluator) estimates these parameters for each S-box input[2]. Next, during the online phase, these templates are used to select the candidate input that has maximum likelihood:

$$\tilde{x} = \underset{x^*}{\text{argmax}}\ \hat{\Pr}_{\texttt{model}}[x^*|l]. \tag{2}$$

The stochastic approach works in a slightly different fashion than template attacks. During profiling, the adversary chooses a basis $[g_0(x), g_1(x), ..., g_N(x)]$. This basis is usually made of monomials in the input and/or output bits of the target operation in the attack (e.g. the S-box in our case). Then, he performs a regression in order to find the model $\hat{\mathsf{L}}_t = \sum_i \ \beta_{i,t} \cdot g_i(x)$ that best matches the

[2] Template attacks can be directly generalized to a multivariate setting, by replacing the means and variances by mean vectors and covariance matrices.

measured leakages. The output of this model can be used as a replacement of the sample means $\hat{\mu}_{x,N}$ in template attacks. Intuitively, the stochastic approach offers a tradeoff between the precision and robustness of the model. For example, a linear model obtained from a 9-element basis (corresponding to the S-box output bits and a constant term) can be estimated rapidly (i.e. its profiling is cheap), but only provides a rough approximation of the leakage function if it is highly non-linear. Next, increasing the degree of the stochastic model allows refining the approximation at the cost of a more expensive profiling. Eventually, a stochastic model of degree 8 with 256 coefficients is equivalent to a template. A convenient feature of the stochastic approach is that it can be used to evaluate the non-linearity of a leakage function, by comparing the information leakage obtained from a linear basis model to the one obtained from exhaustive templates.

4.2 Leakage traces

As a first (informal) step in our analysis, we observe the power traces in Figures 2 and 3. For the CMOS S-box, they represent the 256 transitions between 0 and an arbitrary S-box input. For the DDSLL logic style, they correspond to the evaluation phase of the S-box computation, for the same 256 possible inputs. The figures also display the standard deviations computed over the different inputs. Note that, for readability purposes, the scaling of the Y-axis is not the same in the different figures (i.e. we zoomed on the relevant parts). One can observe that the simulated and measured traces have significantly different shapes, in particular for the DDSLL case. For example, the simulated DDSLL traces are perfectly aligned at the beginning of the clock cycle and then begin to misalign. By contrast, this misalignment is much smaller in actual measurements, where the variance between the curves rather comes from an amplitude difference. The most likely reason to explain these differences is that our simulation environment does not include the specificities of the measurement setup and the filtering it implies. Modeling this setup and including it in our simulations is an interesting scope for further research. However, we also note that, despite these visual differences, the standard deviations for the static CMOS and DDSLL differ by one order of magnitude, both in measurements and simulations.

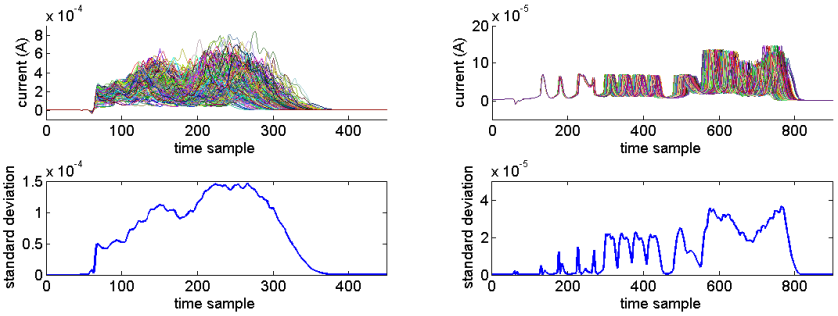

Fig. 2. Up: simulated power traces for the static CMOS (left) and DDSLL (right) S-boxes. Down: std. deviation over the inputs for different time samples.

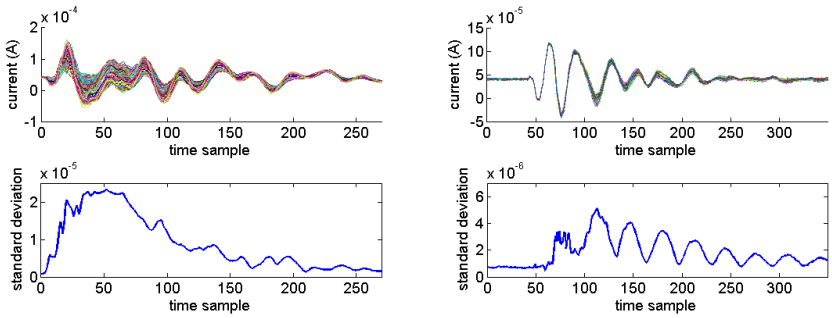

Fig. 3. Up: avg. measured power traces for the static CMOS (left) and DDSLL (right) S-boxes. Down: std. deviation over the inputs for different time samples.

4.3 Information theoretic analysis

The second step in our analysis is to apply the information theoretic metric to the DDSLL and the CMOS logic styles, for simulations and real measurements, using the tools from the previous section (i.e. template attacks, stochastic approach). As mentioned in the introduction (Footnote 1), this information theoretic analysis still relies on a number of hypotheses that we now detail:

H1. The measurement noise is assumed to be normally distributed.
H2. Only 256 transitions are considered for the CMOS S-box (among the 256^2 possible ones). This restriction was mainly motivated by practical measurement constraints and is not expected to strongly affect the comparison between the logic styles[3]. In practice, when performing attacks in Section 4.4, we considered a pre-charge to zero before each S-box computation.
H3. We focused our analysis on the leakage samples of the evaluation phase for the DDSLL S-box, because they exhibited the largest information leakages.
H4. All our analyzes are univariate, i.e. they consider leakage samples one by one. This choice was motivated by the goal to compare the amount and nature of the leakage function observed for the different samples.

In order to estimate the perceived information, we use Equation 1 in which the three probability distributions are computed as follows. First, $\Pr[x]$ is the prior on the input value, i.e. the probability of each input hypothesis before taking into account the side-channel information. We considered a uniform prior $\Pr[x] = 1/256$. Next, $\hat{\Pr}_{\mathtt{chip}}[l|x]$ is the estimated conditional probability of observing a leakage l given an input x. We considered two cases: either real measurements with simulated noise in which case we have $\hat{\Pr}_{\mathtt{chip}}[l|x]$ computed from the leakage probability density functions, or real measurements with real noise, in which case this distribution is sampled from the actual chip, i.e. we use $\sum_x \Pr[x] \sum_l \hat{\Pr}_{\mathtt{chip}}[l|x] = \sum_{(x,l)=1}^{q} \frac{1}{q}$, with q the number of traces measured.

[3] The standard deviation curves of Figures 2, 3, obtained from the transitions between a fixed value x and 256 byte values, looked essentially the same for different x.

Finally, $\hat{\Pr}_{\mathtt{model}}[x|l]$ is the conditional probability of the input given the leakages, derived using the adversary's model. This probability is computed from the template distributions estimated during the profiling. Both for the CMOS and the DDSLL S-box, our estimations are based on sets of 100 measurements for each of the input events x, both for the profiling and the attack phases.

We started by investigating the informativeness of the different time samples in the traces in Figure 4. Note again the different Y axes of the CMOS and DDSLL plots. As expected, one can see that the information extracted with templates is always higher than the one obtained from stochastic models using a linear basis (i.e. the S-box output bits). This is natural, as the templates capture all the information available, in an extensive manner, while the stochastic approach provides a "simpler to estimate" approximation [7, 24]. Nevertheless, one can also observe that some of the samples are very accurately predicted by a linear leakage model. Interestingly, the template-based curve is also reasonably predicted by the standard deviation curves in Figure 3, confirming the analysis of standard DPA in [16]. As a result, these information curves can be directly used for selecting the points of interest for univariate attacks.

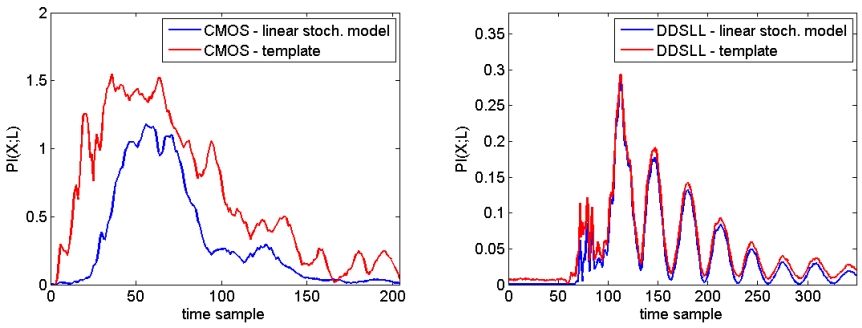

Fig. 4. Perceived information in function of the time samples in the actual measurement traces with real noise for the CMOS (left) and DDSLL (right) S-boxes.

Next, we investigated the information theoretic metric for different noise levels. As an illustration, Figure 5 shows the perceived information for the two logic styles, computed from simulations (left) and actual measurements (right). In both cases, we selected the time sample that maximized the perceived information. For the real measurement curves, we used the hybrid scenario (i.e. $L^3(X,N) = \overline{L^{\mathrm{meas}}}(X) + N$) as well as the real noise values (i.e. $L^2(X,N) = L^{\mathrm{meas}}(X,N)$), represented by the dots on the figure). This experiment allows a number of useful observations that we now detail:

1. The Gaussian noise hypothesis is reasonably accurate, as the dots in Figure 5 are remarkably close to their corresponding (simulated) curves.
2. More importantly, the curves clearly illustrate the information leakage reduction obtained by implementing the AES S-box in the DDSLL logic style (rather than in CMOS), in front of a worst-case template adversary.

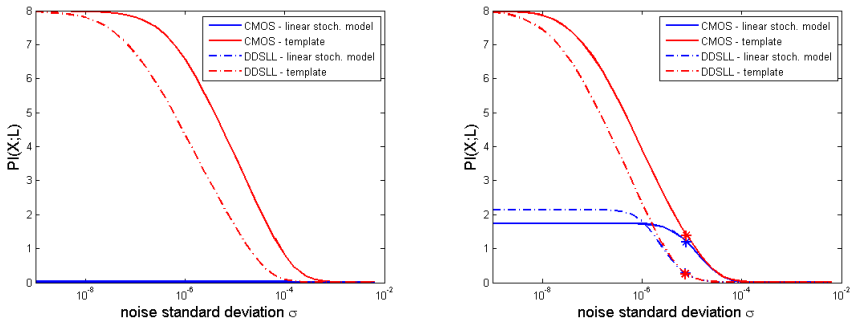

Fig. 5. Perceived information in function of the noise, for the simulations (left), and real measurements (right). The PI with real noise is marked with stars (*).

3. Both simulations and actual measurements show reduced information leakage. However, there is a noticeable reduction of the gap between logic styles when moving from simulations towards measurements. This confirms that, while ELDO (or Spice) simulations are a useful first step in order to analyze physical security issues, the quantified data that their analysis provides does not offer any formal guarantee for the security of the final test chip. This difference is essentially due to the previously mentioned difference between time shifts and amplitude shifts in the traces corresponding to different inputs.
4. In the context of simulated DDSLL traces, the linear stochastic models do not allow to extract any information (i.e. the leakage function corresponding to the selected time sample cannot be approximated accurately enough with a linear combination of the S-box output bits in this simulated case).
5. Finally, the perceived information can be higher for the DDSLL S-box than for the CMOS one, when a linear stochastic model is used in the estimations. It happens, e.g. in the right part of Figure 5, when the (simulated) noise standard deviation is extremely small. This (seemingly paradoxical) observation is explained by the impossibility to predict the actual leakages precisely using an (incomplete) linear basis. Note that this observation would vanish by extending the basis (as it is not observed with templates) and that it does not happen for the actual noise values observed in our measurements.

Combined with the performance analysis in Section 3, these results lead to contrasted conclusions. On the one hand, they confirm that DDL can be an efficient solution for improving security against side-channel attacks. Clearly, the focus of the investigated DDSLL is more on performance than on perfectly data-independent power consumption. But even in this case, experimental data shows that the security improvement over CMOS remains noticeable in a 65-nanometer technology. On the other hand, the information leakage reduction is also limited and not sufficient to provide security when used as a stand alone solution (i.e. without additional countermeasures). As a result, an interesting question is to determine whether the information leakage of the CMOS and DDSLL S-boxes are similarly easy to exploit with non-profiled side-channel attacks. We tackle this problem (i.e. the security analysis of our test chips) in the next section.

4.4 Security analysis

The previous experiments considered the worst-case information theoretic analysis of two prototype S-boxes. In this section, we detail the second part of the evaluation framework in [25], namely the security analysis. For this purpose, we investigated the success rates of various profiled and non-profiled attacks.

As far as profiled attacks are concerned, we carried out the template attacks described in [3], as a worst-case (univariate) scenario. Next, and as a non-profiled complement, we started by applying a variant of the stochastic approach described in [11], in which the adversary builds a leakage model on-the-fly, using a linear basis. The underlying assumption of this variant is that a correct key hypothesis should give rise to the most accurate model, given that the base vectors used to build it reasonably match the actual leakages. In addition, we also performed a number of previously introduced attacks, namely the Correlation Power Analysis (CPA) using a Hamming weight leakage model introduced by Brier et al. in 2004 [2] and Kocher et al.'s single-bit Differential Power Analysis (DPA) [10]. However, we note that these attacks are redundant to some extent and that most intuitions can be extracted from the on-the-fly stochastic attacks[4]. We now detail a few meaningful observations from our experiments.

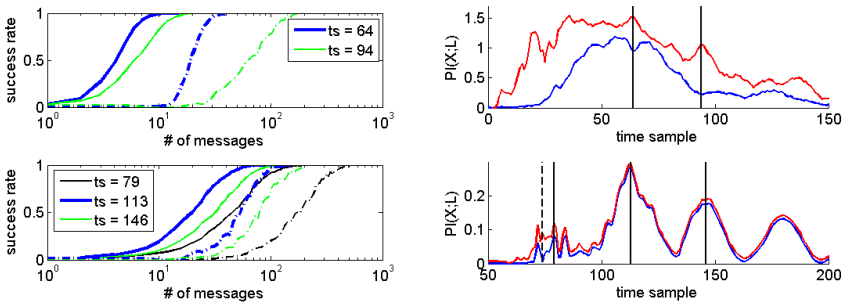

Fig. 6. Left: success rates for the template attacks (straight lines) and on-the-fly stochastic attacks (dotted lines) against the CMOS (top) and DDSLL (bottom) S-box. Right: perceived information for the selected samples (vertical lines).

First, Figure 6 shows the success rates corresponding to the template and on-the-fly stochastic attacks, estimated using the CMOS and DDSLL measurements, for different time samples, including the most informative ones. The template curves provide a worst-case estimate of the number of measurements needed

[4] Summarizing, when provided with a single-element basis that corresponds to the target bit of a single-bit DPA, or the model of a CPA, the on-the-fly stochastic approach is essentially similar to these CPA/DPA. Following a reasoning similar to to [16], one could show that minor differences in the success rates of these distinguishers are due to statistical artifacts. By contrast, when provided with a larger (e.g. 9-element linear) basis, it allows improved resistance against incorrect assumptions, at the cost of a more expensive estimation, reflected in a slightly higher data complexity.

to recover the key. They exhibit a security increase of approximately one order of magnitude, between the CMOS and DDSLL S-boxes. As expected, the number of texts required to perform a successful template attack is nicely correlated with the perceived information computed with profiled templates. Interestingly, the perceived information computed with profiled and linear stochastic models also provides an accurate prediction of the non-profiled and linear stochastic attacks for certain samples, although we have no formal guarantee in this case. These results underline that both for the CMOS and DDSLL S-boxes, there are samples in the traces that have sufficiently strong linear dependencies in the S-box output bits for being easily exploited with non-profiled attacks.

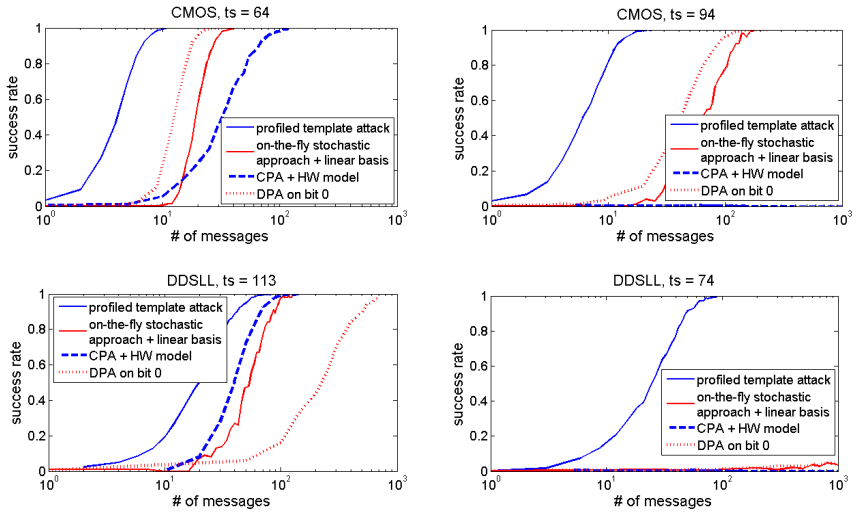

Fig. 7. Non-profiled attacks against the CMOS (up) and DDSLL (bottom) test chips, using the most informative samples (left) and less informative ones (right).

Next, the left parts of Figure 7 show the results of DPA and CPA attacks using the most informative samples in our traces. They confirm that these attacks, when based on a good assumption (e.g. single-bit DPA for the CMOS S-box, CPA with Hamming weight leakage model for the DDSLL one) can slightly outperform the on-the-fly stochastic approach (and naturally remain bounded by the worst-case template curves). Yet, as discussed in [5], the increase in data complexity of the on-the-fly stochastic approach with linear basis is very limited. Interestingly, we could verify that the S-box output bits having high weight in the stochastic models are the ones for which a single-bit DPA succeeds. In other words, all the intuition provided by a single-bit DPA could be extracted from a stochastic attack in this case. As illustrated in the upper right part of Figure 7, there also exist time samples for which a correlation attack using a Hamming weight leakage model does not succeed in recovering the key, while the non-profiled stochastic attack using a linear basis does. This confirms the

previous observation in [22] that the latter distinguisher is a tool of choice for dealing with leakage in recent technologies (including possible variability issues). Finally, we could spot leakage samples, e.g. in the lower right part of Figure 7, for which only profiled side-channel attacks allow successful key recoveries (i.e. for which even the best single-bit DPA could not reach a high success rate).

Regarding the discussion about non-linear leakage functions in [28], these experiments again lead to contrasted conclusions. First, let us mention that with "non-linear" leakage samples, we simply denote the ones that could not be exploited with a linear leakage model. As a matter of fact, many leakage samples in the traces, including the most informative ones, were sufficiently linear. On the other hand, we could also spot a few pathologic samples (e.g. ts = 74 for the DDSLL S-box) for which these attacks do not succeed. As a result, and as far as the experiments in this work are concerned, one can conclude that the non-linearity issue is not a limitation for practical attacks, in which the existence of a few linear leakage samples is sufficient to perform a successful key recovery. But the existence of non-linear leakage samples also confirms that only a profiled information theoretic evaluation is theoretically able to evaluate all the information leaked by an implementation (i.e. in a worst-case scenario extended to the multivariate setting, by getting rid of H4 in Section 4.3).

5 Discussion and open questions

This paper brings two main contributions related to the implementation of the AES S-box in protected logic styles, using a 65-nanometer technology.

First, we show that optimizing such logic styles with performances in mind can strongly mitigate the overheads of DDL compared to CMOS. Arguably, our investigated DDSLL S-box is based on full custom design, which remains an important drawback. But it illustrates that for some primitives, it could be an acceptable solution, both in terms of area cost and power consumption.

Second, we put forward the information leakage reduction of such an S-box, when evaluated in front of various side-channel distinguishers. In the case of a worst-case template attack, it typically corresponds to one order of magnitude, in terms of number of measurements to recover the key. Hence, practically secure implementations would clearly require to combine the investigated logic style with other countermeasures, like masking. Yet, it remains that relying on DDL gives additional means for the designers to control the information leakage of an implementation. As a consequence, it is an interesting open problem to determine whether the gap between CMOS and such modified designs further reduces in smaller technologies. That is, do the traditional advantages of DDL vanish as the circuit sizes are shrinking, because of hardware constraints that become harder to fulfill? In this respect, a very interesting project would be to systematically compare technology nodes (e.g. 130nm, 90nm, 65nm, 45nm and 22nm) in terms of their respective resistance against side-channel attacks.

Eventually, one disappointing point of our DDSLL test chip is the strongly linear nature of its leakages (that goes against the expectations of ELDO/Spice simulations). Although stochastic models are very useful to discuss such questions, the precise understanding of this linearity remains difficult. Hence, an important question for further research is to determine if the use of DPDN in our designs (allowing improved efficiency) is not the main cause of this limitation. In other words, do other DDL such as SABL, WDDL or MCML provide more non-linear power consumption traces? More generally, is it possible to develop a logic style with non-linearity as a design guideline? In addition to the reduced information leakages, this would then provide a clear gain over CMOS devices, in view of the difficulty to exploit such leakages with non-profiled attacks [28].

References

1. M. Allam and M. Elmasry. Dynamic current mode logic: a new low-power high-performance logic style. *Journal of Solid State Circuits*, 36:550–558, March 2001.
2. E. Brier, C. Clavier, and F. Olivier. Correlation power analysis with a leakage model. In M. Joye and J.-J. Quisquater, editors, *CHES*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.
3. S. Chari, J. R. Rao, and P. Rohatgi. Template attacks. In B. S. K. Jr., Çetin Kaya Koç, and C. Paar, editors, *CHES*, volume 2523 of *Lecture Notes in Computer Science*, pages 13–28. Springer, 2002.
4. Z. T. Deniz and Y. Leblebici. Low-power current mode logic for improved dpa-resistance in embedded systems. In *ISCAS (2)*, pages 1059–1062. IEEE, 2005.
5. J. Doget, E. Prouff, M. Rivain, and F.-X. Standaert. Univariate side channel attacks and leakage modeling. In *Journal of Cryptographic Engineering*, to appear.
6. B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel. Mutual information analysis. In E. Oswald and P. Rohatgi, editors, *CHES*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, 2008.
7. B. Gierlichs, K. Lemke-Rust, and C. Paar. Templates vs. stochastic methods. In L. Goubin and M. Matsui, editors, *CHES*, volume 4249 of *Lecture Notes in Computer Science*, pages 15–29. Springer, 2006.
8. I. Hassoune, F. Macé, D. Flandre, and J.-D. Legat. Dynamic differential self-timed logic for robust and low-power security ics. *Integration*, 40(3):355–364, 2007.
9. D. D. Hwang, K. Tiri, A. Hodjat, B.-C. Lai, S. Yang, P. Schaumont, and I. Verbauwhede. Aes-based security coprocessor ic in 0.18um cmos with resistance to differential power analysis side-channel attacks. *IEEE Journal of Solid-State Circuits*, 41(4): pages 781–792, April 2006.
10. P. C. Kocher, J. Jaffe, and B. Jun. Differential power analysis. In M. J. Wiener, editor, *CRYPTO*, volume 1666 of *LNCS*, pages 388–397. Springer, 1999.
11. K. Lemke-Rust. Models and algorithms for physical cryptanalysis. PhD dissertation, University of Bochum, January 2007.
12. L. Lin and W. P. Burleson. Analysis and mitigation of process variation impacts on power-attack tolerance. In *DAC*, pages 238–243. ACM, 2009.
13. F. Macé, F.-X. Standaert, I. Hassoune, and J.-D. Legat. A dynamic current mode logic to counteract power analysis attacks. In *DCIS, pp 186-191, 2004.*
14. F. Macé, F.-X. Standaert, and J.-J. Quisquater. Information theoretic evaluation of side-channel resistant logic styles. In P. Paillier and I. Verbauwhede, editors, *CHES*, volume 4727 of *LNCS*, pages 427–442. Springer, 2007.

15. F. Macé, F.-X. Standaert, J.-J. Quisquater, and J.-D. Legat. A design methodology for secured ics using dynamic current mode logic. In *PATMOS*, volume 3728 of *LNCS*, pages 550–560. Springer, 2005.
16. S. Mangard, E. Oswald, and F.-X. Standaert. One for all - all for one: Unifying standard dpa attacks. to appear in IEE Information Security, 2011.
17. N. Mentens, L. Batina, B. Preneel, and I. Verbauwhede. A systematic evaluation of compact hardware implementations for the rijndael s-box. In *CT-RSA*, volume 3376 of *Lecture Notes in Computer Science*, pages 323–333. Springer, 2005.
18. T. Popp, M. Kirschbaum, T. Zefferer, and S. Mangard. Evaluation of the masked logic style mdpl on a prototype chip. in *CHES 2007* [19], pp 81-94.
19. T. Popp and S. Mangard. Masked dual-rail pre-charge logic: Dpa-resistance without routing constraints. In Rao and Sunar [20], pages 172–186.
20. J. R. Rao and B. Sunar, editors. *Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *LNCS*. Springer, 2005.
21. F. Regazzoni, T. Eisenbarth, A. Poschmann, J. Großschädl, F. K. Gürkaynak, M. Macchetti, Z. T. Deniz, L. Pozzi, C. Paar, Y. Leblebici, and P. Ienne. Evaluating resistance of mcml technology to power analysis attacks using a simulation-based methodology. *Transactions on Computational Science*, 4:230–243, 2009.
22. M. Renauld, F.-X. Standaert, N. Veyrat-Charvillon, D. Kamel, and D. Flandre. A formal study of power variability issues and side-channel attacks for nanoscale devices. in the proceedings of Eurocrypt 2011, Lecture Notes in Computer Science, vol 6632, pp 109-128, Tallinn, Estonia, May 2011.
23. W. Schindler, K. Lemke, and C. Paar. A stochastic model for differential side channel cryptanalysis. In Rao and Sunar [20], pages 30–46.
24. F.-X. Standaert, F. Koeune, and W. Schindler. How to compare profiled side-channel attacks? In M. Abdalla, D. Pointcheval, P.-A. Fouque, and D. Vergnaud, editors, *ACNS*, volume 5536 of *LNCS*, pages 485–498, 2009.
25. F.-X. Standaert, T. Malkin, and M. Yung. A unified framework for the analysis of side-channel key recovery attacks. In A. Joux, editor, *EUROCRYPT*, volume 5479 of *Lecture Notes in Computer Science*, pages 443–461. Springer, 2009.
26. K. Tiri and I. Verbauwhede. A dynamic and differential cmos logic with signal indipendent power consumption to withstand differential power on smart cards. In *Proceedings of the 28th European Solid-State Circuits Conference (ESSCIRC'02)*, pages 403 – 406, Florence, Italy, 2002.
27. K. Tiri and I. Verbauwhede. A logic level design methodology for a secure dpa resistant asic or fpga implementation. In *DATE*, pages 246–251, 2004.
28. N. Veyrat-Charvillon and F.-X. Standaert. Generic side-channel distinguishers: Improvements and limitations. to appear in the proceedings of CRYPTO 2011.