DPA-Resistance Without Routing Constraints?

– A cautionary note about MDPL security –

Benedikt Gierlichs

K.U. Leuven, ESAT/SCD-COSIC
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
benedikt.gierlichs@esat.kuleuven.be

Abstract. MDPL is a logic style claiming to provide resistance against Differential Side Channel Analysis on power consumption measurements. In this paper we show that the power consumption of a non-linear MDPL gate can be reliably exploited to determine signal values and hence secret data, if the random masks have a slight bias. We present an attack methodology and a case study on how to infer secret key bits of an MDPL secured AES-ASIC in practice by attacking a single MDPL AND gate in a VLSI circuit. Our attack is not based on frequently made assumptions on circuit "anomalies", but on the per definition unbalanced routing, realistic PRNG biases, and knowledge of the circuit layout.

Keywords: Differential Side Channel Analysis, DSCA, Masked Dual-rail Pre-charge Logic, MDPL, Gate-level masking, DRP

1 Introduction

Side Channel Analysis (SCA) is one of the most promising approaches to reveal secret data, such as cryptographic keys, from black-box secure cryptographic algorithms implemented in embedded devices. In this paper we focus on the power consumption side channel and hence on power analysis, which exploits the physical dependency of a device's power consumption and the data it is processing. Differential Side Channel Analysis (DSCA) exploits (small) differences in a set of measurements by means of statistics and is particularly well suited for the power analysis of block cipher implementations.

In the last decade, various attack methodologies have been put forward, such as Differential Power Analysis [13] and Correlation Power Analysis [6] as well as so called profiling attacks like the Stochastic Model [14] and Template Attacks [15]. As a consequence of the need for secure embedded devices such as smart cards, mobile phones, and PDAs research is also conducted in the field of DSCA prevention. Early countermeasures include algorithmic masking schemes [16, 17], noise generators [19], and random process interrupts [18]. All of them have in common that they do not address the issue of side channel leakage directly, but aim at obfuscating the observables. Most of these countermeasures have been proven to be either insecure or circumventable, *e.g.* with High-Order attacks or Digital Signal Processing.

In recent years, research and industry have started to approach the issue of side channel leakage right where it arises: at the gate level. There is a considerable body of research on gate level masking schemes, *e.g.* [2, 9, 10], which again aim at obfuscating the leakage and differential logic styles, which aim at reducing the leakage. Tiri and Verbauwhede introduced WDDL [20] where they use the concept of Fat Wires for the balanced routing of the complementary wire pairs. As a result, a WDDL circuit ideally has a constant power consumption and hence no side channel leakage. Popp and Mangard introduced MDPL [1] which applies both aforementioned concepts: it does not use special differential routing but instead randomizes the signals on the complementary wire pairs. As a result, the remaining leakage of an MDPL circuit is assumed to be randomized to the quality of the random numbers provided.
On the other hand, also attacks against these secured logic styles have been published. Most of them exploit circuit "anomalies" as for example glitches [11, 12] and the early propagation effect [8]. In [7] it has been shown that mask induced switching activity in the circuit can be exploited to circumvent single-rail gate level masking.

Masked Dual-rail Pre-charge Logic (MDPL) was published at CHES in 2005 [1]. It follows straight and simple design principles in order to provide DSCA resistance, which as the authors claim, can be achieved without routing constraints. In this work, we profoundly analyze the power consumption of and the security provided by non-linear MDPL logic gates, which are an important building block for MDPL secured circuits. By non-linear gate we denote any logic gate for which the distribution of the output bits [1], given uniformly distributed input bits, is not uniform.

We will show that MDPL provides enhanced security which will likely discourage amateur adversaries, but that it cannot withstand powerful expert attackers. Our attack does not require glitches or early propagation, but is based on the per definition unbalanced routing of MDPL circuits

[1] For differential logic styles this notion applies to one of the two complementary wires.

and assumes realistic (unknown) biases in the Pseudo Random Number Generator (PRNG) which supplies the masks.

We summarize the key properties of MDPL in Sect. 2 and introduce our notation and basic preliminaries in Sect. 3. The core of our contribution is Sect. 4, where we analyze non-linear MDPL gates in detail and present our attack methodology. In Sect. 5 we provide experimental results from our successful attack against a single AND gate in a MDPL secured VLSI circuit which can be generalized straight forward. As an approach to explain our results under the assumption that the PRNG implementation on the prototype chip is not (significantly) biased, we discuss the possibility that our attack unintentionally exploited circuit anomalies in Sect. 6. We conclude our work in Sect. 7.

2 MDPL

In our view, each of the letters MDP stands for a layer of security that enwraps the previous layer. At the core of this protective construction are standard CMOS gates, that are well known to be vulnerable to DSCA. In the next subsection, we summarize the MDPL design principles according to [1] and exemplify our view of the security layers using an MDPL AND gate.

2.1 MDPL design principles

MDPL's main DCSA countermeasure is masking, while all other features aim at securing the masking as will be explained shortly. The atomic elements of MDPL logic are CMOS majority gates. A majority gate's output is "1" if the majority of its inputs are "1", otherwise its output is "0". In an MDPL circuit, all data values, e.g. a, are masked with the same mask m and physically present as the masked wire $a_m = a \oplus m$. The mask m must be refreshed every clock, *e.g.* by a PRNG. The mask update mechanism is integrated in the MDPL flip-flops (cf. [1]).

In MDPL, a majority gate always has three input signals, e.g. a_m, b_m, and m, and one output signal, e.g. q_m. Figure 1 depicts a majority gate and Fig. 2 its truth table. In order to prevent glitches, which are a serious concern for designers of hardware masking countermeasures [11, 12], all signals in the circuit are pre-charged. During the first half of every clock cycle, i.e. at the rising clock edge, every MDPL flip-flop starts a pre-charge wave that pre-charges the subsequent logic and wires to "0". During the second half of every clock cycle, i.e. at the falling clock edge, the logic

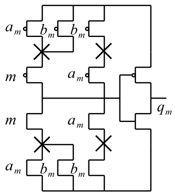

Fig. 1. Majority gate

a_m	0	0	1	1	0	0	1	1
b_m	0	0	0	0	1	1	1	1
m	0	1	0	1	0	1	0	1
q_m	0	0	0	1	0	1	1	1

Fig. 2. Majority gate's truth table

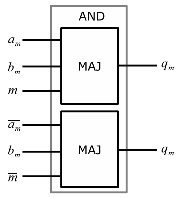

Fig. 3. MDPL AND gate

evaluation takes place and wires propagate masked data values. According to [1], neither a majority gate nor any gate built from majority gates produces glitches in a pre-charge circuit. This is said to be also true, when the inputs arrive with different delays. Suzuki and Saeki study the behavior of MDPL gates in such a scenario [8] and discover an early propagation effect.

The (pseudo-)random mask bit m has to be provided to *every* cell in the circuit at the beginning of each evaluation phase, hence one may expect a signal tree that is larger than the clock tree. Since also m is pre-charged to 0 during the pre-charge phase, its transitions during the evaluation phase are limited to $0 \rightarrow 1$ or $0 \rightarrow 0$. Due to the size of m's signal tree, one may expect that the Side channel leakage of a $0 \rightarrow 1$ transition is clearly distinguishable from that of a $0 \rightarrow 0$ transition. Hence, SPA/SEMA might be able to recover m's value for every clock cycle. Tiri and Schaumont apply a similar attack [7] exploiting mask induced switching activity on Random Switching Logic [21]. To render such attacks infeasible, MDPL implements the dual-rail principle. For every signal a (including m) also the complementary signal $\overline{a}$ is physically present in the circuit as masked wire $\overline{a}_m = \overline{a} \oplus m$ and every [2] MDPL gate actually contains two identical sets of logic that process complementary inputs and output q_m and $\overline{q}_m$. This way it is assured, that every pair of complementary wires and every MDPL gate switches exactly once per pre-charge and once per evaluation phase. Figure 3 depicts an MDPL AND gate taking into account the masking and the DRP principle.

[2] The XOR gate is a special case.

3 Notation and preliminaries

Let $\mathbf{A}, \mathbf{B}, \mathbf{M}$ be random variables on the (discrete) space $\mathcal{S} := \{0, 1\}$ with probability distributions $\mathbb{P}_{\mathbf{A}}, \mathbb{P}_{\mathbf{B}}$, and $\mathbb{P}_{\mathbf{M}}$. Let

$$\mathbb{P}_{\mathbf{A}} = \mathbb{P}_{\mathbf{B}} := \{0 : 0.5, 1 : 0.5\}, \mathbb{P}_{\mathbf{M}} = \{0 : \alpha, 1 : \overline{\alpha}\} \tag{1}$$

with $a + \overline{\alpha} = 1, 0 \leq \alpha \leq 1$ (where α denotes the bias of the distribution). We denote the conditional probability of $\mathbf{A}$ given $\mathbf{B}$ as $P(\mathbf{A}|\mathbf{B})$. It is defined as

$$P(\mathbf{A}|\mathbf{B}) = \frac{P(\mathbf{A} \cap \mathbf{B})}{P(\mathbf{B})} \tag{2}$$

where $P(\mathbf{A} \cap \mathbf{B})$ is the joint probability of $\mathbf{A}$ and $\mathbf{B}$. Often one has knowledge about conditional probabilities and would like to compute marginal probabilities. This can be done if and only if all conditional probabilities are known

$$P(\mathbf{A}) = \sum_b P(\mathbf{A}|\mathbf{B} = b) \cdot P(\mathbf{B} = b)\,. \tag{3}$$

In the following section we apply these concepts to non-linear MDPL gates.

4 Attack methodology

We model the logic signals a, b, and m as the random variables $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{M}$ respectively. We model the output transition on wire q_m of a given MDPL gate as random variable $\mathbf{T}$ on the (discrete) space $\mathcal{T} := \{0 \rightarrow 1, 0 \rightarrow 0\}$. The transition $\overline{\mathbf{T}}$ on wire $\overline{q}_m$ is implicitly defined by being $\mathbf{T}$'s complement on an identical space $\overline{\mathcal{T}}$. The probability distribution $\mathbb{P}_{\mathbf{T}}$ is defined by the logic function of the gate. The probability to observe transition $\mathbf{T} = t \in \mathcal{T}$ on wire q_m given the specific (unmasked) input signals $\mathbf{A} = a, \mathbf{B} = b, \mathbf{M} = m$ is $\mathrm{P}(\mathbf{T} = t|a, b, m)$. We denote the observable output transition energy of a given MDPL gate as $E(\mathbf{T} = t)$. Note that this notion of energy includes the transition $\overline{T} = \overline{t}$ on the complementary wire.

Consider the MDPL security layers as explained in Sect. 2. Layer 1 (the masking) restricts us to only observe masked leakage, whereas a fresh mask is provided for every clock cycle. Layer 2 makes sure that no glitches can occur in the circuitry by pre-charging every wire and logic gate (not the flip-flops) to "0". Layer 3 limits the observable *difference* of output transition energies to $E(\mathbf{T} = 0 \rightarrow 1)$ - $E(\overline{\mathbf{T}} = 0 \rightarrow 1)$ which we expect to be much smaller than the difference between $E(0 \rightarrow 1)$

and $E(0 \rightarrow 0)$ for CMOS. However, as MDPL explicitly claims to be DPA resistant without differential routing constraints, we may assume that $E(\mathbf{T} = 0 \rightarrow 1) \neq E(\overline{\mathbf{T}} = 0 \rightarrow 1)$ for any given gate without loss of generality. Basically, layer 3 introduces a practical measurement problem.

What can we extract from this? As the circuit continuously applies a fresh mask for every clock cycle, we will focus our considerations on one clock cycle. If the PRNG is started correctly, $\mathbf{M} = m$ according to $\mathbb{P}_{\mathbf{M}}$. We denote $E(\mathbf{T} = 0 \rightarrow 1) = \delta$ and $E(\overline{\mathbf{T}} = 0 \rightarrow 1) = \gamma$ and may assume that $\delta \neq \gamma$. Further, we must also assume that the capacities of the wires that represent the signals q_m and $\overline{q}_m$ differ from gate to gate, thus the energy needed to charge them, and hence the *gate specific* δ and γ.

In the following subsection we present an attack methodology against non-linear MDPL gates based on a bias in the random masks. The detailed analysis will focus on a single gate to reveal its specific properties and weaknesses. Note that attacking several gates in parallel is rather difficult as the exploited gate specific δ and γ most likely differ from gate to gate. In particular, their difference might have a different algebraic sign. However, iterative attacks against several gates to sieve key candidates are straight-forward and will not be discussed in detail in this work.

4.1 Analysis of a single non-linear gate

We avail ourselves of an MDPL AND gate as representative of the class of non-linear MDPL logic gates. An MDPL AND gate as shown in Fig. 3 is composed of two majority gates which process complementary inputs and hence output complementary values. Figure 4 summarizes the relations between the input signals and input wires. Figure 5 shows the AND gate's truth table [3] and its output transition energies δ and γ as defined above. According to Eq. (3), the marginal probability distribution $\mathbb{P}_{\mathbf{T}}$ of the output transition $\mathbf{T}$ can be derived from the set of conditional distributions provided in Fig. 5. Applying Eq. (1) and (2) yields

$$\mathbb{P}_{\mathbf{T}} = \{0 \rightarrow 1 : 0.75 - 0.5\alpha, 0 \rightarrow 0 : 0.25 + 0.5\alpha\} .$$

Observations.

1. $\mathbb{P}_{\mathbf{T}}(0 \rightarrow 1)|_{\alpha=0.5} = 0.5$, for $\mathbb{P}_{\mathbf{M}}(0) = \mathbb{P}_{\mathbf{M}}(1) = 0.5$ the transitions in $\mathcal{T}$ are equally likely
2. $\mathbb{P}_{\mathbf{T}}(0 \rightarrow 1)|_{\alpha\neq0.5} \neq 0.5$, for $\mathbb{P}_{\mathbf{M}}(0) \neq \mathbb{P}_{\mathbf{M}}(1) \neq 0.5$ the transitions in $\mathcal{T}$ are *not* equally likely.

[3] We omit the obvious values of the complementary input signals for the sake of clearness wherever possible, but the reader needs to keep in mind that we discuss a differential logic style.

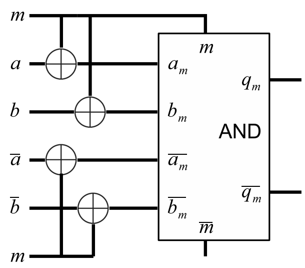

Fig. 4. AND gate

$\mathbf{A}_m$	$\mathbf{B}_m$	$\mathbf{M}$	bias	$\mathbf{A}$	$\mathbf{B}$	$\mathbf{T}$	$\overline{\mathbf{T}}$	E
0	0	0	α	0	0	$0 \to 0$	$0 \to 1$	γ
0	0	1	$1-\alpha$	1	1	$0 \to 0$	$0 \to 1$	γ
0	1	0	α	0	1	$0 \to 0$	$0 \to 1$	γ
0	1	1	$1-\alpha$	1	0	$0 \to 1$	$0 \to 0$	δ
1	0	0	α	1	0	$0 \to 0$	$0 \to 1$	γ
1	0	1	$1-\alpha$	0	1	$0 \to 1$	$0 \to 0$	δ
1	1	0	α	1	1	$0 \to 1$	$0 \to 0$	δ
1	1	1	$1-\alpha$	0	0	$0 \to 1$	$0 \to 0$	δ

Fig. 5. AND gate's truth table and transition energies

4.2 Attack against an AND gate given biased random masks

Suppose that **A** and **B** are intermediate results of a cryptographic computation carried out by a device implemented in MDPL. Suppose that **A** and **B** are independent but both depend on secret and known data [4]. We address our focus on the truth table in Fig. 5. We restrict the observable space of events to those, for which, based on the known data and a guess on the secret data, $\mathbf{A} = \mathbf{B}$ holds [5]. This space is marked with gray background color in Fig. 5.

Let $\Theta = E(\mathbf{T}|a = 0, b = 0) - E(\mathbf{T}|a = 1, b = 1)$. For a correct guess on the secret (at the relevant bits) and hence a correct guess on **A** and **B** we have, according to Eq. (1), (2), and (3):

$$E(\mathbf{T}|a = 0, b = 0) = \alpha\gamma + (1-\alpha)\delta,\ E(\mathbf{T}|a = 1, b = 1) = \alpha\delta + (1-\alpha)\gamma$$
$$\Rightarrow \Theta = 2\alpha\gamma - 2\alpha\delta + \delta - \gamma\,. \tag{4}$$

We observe that for $\alpha > 0.5$, Θ tends toward $\gamma - \delta$, while for $\alpha < 0.5$, Θ tends toward $\delta - \gamma$. For $\alpha = 0.5$ we have $\Theta = 0$. Note that the gate specific δ and γ as well as the bias α influence the algebraic sign of Θ.

For a wrong guess on the secret (at the relevant bits) such that the guess on **A** *and* **B** is wrong, we have:

$$E(\mathbf{T}|a = 0, b = 0) = \alpha\delta + (1-\alpha)\gamma,\ E(\mathbf{T}|a = 1, b = 1) = \alpha\gamma + (1-\alpha)\delta$$
$$\Rightarrow \Theta = 2\alpha\delta - 2\alpha\gamma + \gamma - \delta\,. \tag{5}$$

We observe that for $\alpha > 0.5$, Θ tends toward $\delta - \gamma$, while for $\alpha < 0.5$, Θ tends toward $\gamma - \delta$. For $\alpha = 0.5$ we have $\Theta = 0$. Note that for such a guess and $\alpha \neq 0.5$, Θ points exactly to the opposite direction.

[4] It is impossible to model the dependency further as it will be entirely defined by the specific cryptographic algorithm and its implementation.

[5] We assume that the cryptographic algorithm and its implementation are known.

For a wrong guess on the secret (at the relevant bits) such that the guess on *either* **A** *or* **B** is wrong, we have:

$$E(\mathbf{T}|a=0,b=0) = E(\mathbf{T}|a=1,b=1) = \alpha\gamma + (1-\alpha)\delta + (1-\alpha)\delta + \alpha\gamma \\ \Rightarrow \Theta = 0\,. \tag{6}$$

Note that for such a guess $\Theta = 0$ independently of α, if the the wrong guess is uniformly distributed over **A** and **B**, which we assume.

It follows from Eq. (4, 5, 6) that for any given bias $\alpha \neq 0.5$ the three values for Θ are different, if $\delta \neq \gamma$ which is very likely because MDPL does not demand differential routing. Thus, a guess that is wrong in *either* **A** *or* **B** is distinguishable without further knowledge on $\alpha, \delta,$ or γ. An adversary may exploit this property to reject all key hypotheses which lead to such a guess. Then she would run the same attack against a different AND gate for further sieving. Attacks that involve knowledge on $\alpha, \delta,$ and γ are not the scope of this work.

5 Experimental results

In this section we provide experimental results of our attacks against an MDPL protected AES-128 implementation. Our main focus is the so far not posed question whether the output transition energy difference Θ of a single gate in a VLSI chip is practically measurable outside the chip.

5.1 Experimental platform

The SCARD chip is an outcome of the "Side-Channel Analysis Resistant Design Flow - SCARD" project led by the European Commission. It implements an 8051 μC with AES-128 co-processor in CMOS and several secured logic styles, MDPL being one of them. It also implements a PRNG which supplies the masks for the masked logic styles. For a summary of the chip schematics we refer the reader to [3]. The architecture of the AES co-processor is discussed in detail in [4]. The AES implementation uses four parallel one-stage pipelined implementations of the AES SubBytes transformation, as described in [5].

For our experiments we obtained two sets of power measurements. The power samples $W(t)$ represent the voltage drop over a 50Ω resistor inserted in the dedicated core VDD supply. We measure during the first round of AES-128 encryption of random uniformly chosen plaintexts X with a constant key K. The sets are

1. $N_1 = 100\,000$ traces, sampled at 2GS/s, PRNG bias $\alpha = 1$
2. $N_2 = 200\,000$ traces, sampled at 2GS/s, PRNG bias $\alpha =$ unknown

5.2 MDPL vs. unMaskedDPL

We begin our experimental analysis with the comparison of the results of two runs of the same "standard" [6] attack against the MDPL AES implementation. For the first attack, we use measurement set 1 with a bias $\alpha = 1$. For the second attack, we use 100 000 measurements from set 2 for which the PRNG has been setup and started correctly with an unknown bias α. The point of this comparison is to show that MDPL is vulnerable to not specifically crafted power attacks, if the masking is completely disabled. Further, it shows that MDPL is resistant against the same attack if masking is active. Finally, it verifies that we initialized and activated the PRNG correctly.

The attack we perform is Correlation Power Analysis (CPA) [6]. It estimates the correlation coefficient

$$\rho_{WH} = \frac{N\sum W_i H_i - \sum W_i \sum H_i}{\sqrt{N\sum W_i^2 - (\sum W_i)^2}\sqrt{N\sum H_i^2 - (\sum H_i)^2}} \tag{7}$$

between a vector of observations (W_i) and a vector of predictions (H_i). The summations are taken over the N samples and the correlation coefficient has to be estimated for each time slice within the observations $W_i(t)$ separately. For a detailed discussion we refer to [6].

Since the actual storage element in an MDPL flip-flop is a standard CMOS flip-flop which is not pre-charged to "0", we expect the energy dissipation of a flip-flop to depend on whether the value to store changes or not. Therefore, the predictions are based on the Hamming Distance Model and aim at the simultaneous transitions of four 8-bit registers from their previous value $R_i \in \{0,1\}^{32}$ to their new value $D_i \in \{0,1\}^{32}$. It is $H_i = \mathrm{HW}(R_i \oplus D_i)$, where $\mathrm{HW}(\cdot)$ is the well known Hamming weight function. Whether an attack on eight key bytes (four for D and four for R) in parallel is practical or not is beyond the scope of this paper. The goal of this experiment simply is to show the protective effect of the masking.

Figure 6 shows the correlation trace derived from attack 1 for the correct key in the upper plot. The peak at the time index of about 23 000 is not large but significant and seems to allow key recovery to a certain extend [7]. The lower plot in Fig. 6 shows the correlation trace derived from

[6] With standard attack we denote an attack that is not specifically crafted for the properties of MDPL.

[7] We verified that 2^8 wrong subkeys were rejected in favor of the correct subkey.

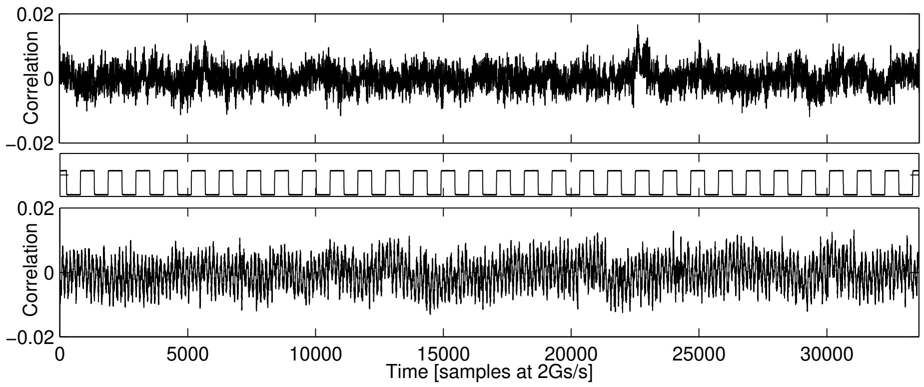

Fig. 6. CPA on MDPL with bias $\alpha = 1$ (upper plot) and activated masking (lower plot); clock signal (middle)

attack 2 for the correct key. As one can see, no visible peaks appear at the time index of about 23 000 or elsewhere, which indicates no reliable key recovery [8]. Obviously, the masking provides security against a correlation attack in the given setting.

5.3 Results of our attack against a single AND gate

In this section we provide the result of our attack methodology against the SCARD chip's AES-128 implementation in MDPL. We provide results from a DPA attack based on measurement set 2 against a single AND gate as proof of concept. Our intention is to experimentally verify that leakage occurs and can be exploited as concluded in Sect. 4.2.

The AES Sbox is implemented in combinational logic using composite field representation. Figure 7 shows the relevant part of the Sbox architecture. The conversion of elements of $GF(2^8)$ to $GF((2^4)^2)$ is given by the function map, shown in Fig. 8.

The 4×4-bit multiplier in Fig. 7 contains 16 AND gates. One of them will be the target of our attack. It computes the intermediate result: $\mathbf{A} \wedge \mathbf{B} = a_{l0} \wedge a_{h0} = (a_4 \oplus a_5 \oplus a_6) \wedge (a_0 \oplus a_4 \oplus a_5 \oplus a_6)$.

The attack is based on the methodology introduced in Sect. 4.2 and standard DPA [13]. Our attack requires $\mathbf{A} = \mathbf{B}$ where a is the 8bit word $x_i \oplus k$ entering one of the Sboxes. We partition our measurements $W(t)$

[8] Again we also tried the same 2^8 wrong key hypotheses, this time the results were fuzzy.

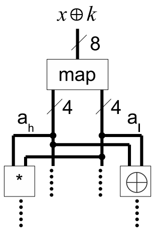

Fig. 7. Relevant part of the Sbox architecture

$$
\begin{array}{l}
a_h x + a_l = \mathrm{map}(a) \\
\text{with } (a_h, a_l \in GF(2^4), a \in GF(2^8)) \\
\hline
a_A = a_1 \oplus a_7, a_B = a_5 \oplus a_7, a_C = a_4 \oplus a_6 \\
a_{l0} = a_C \oplus a_0 \oplus a_5, a_{l1} = a_1 \oplus a_2 \\
a_{l2} = a_A, a_{l3} = a_2 \oplus a_4 \\
a_{h0} = a_C \oplus a_5, a_{h1} = a_A \oplus a_C \\
a_{h2} = a_B \oplus a_2 \oplus a_3, a_{h3} = a_B \\
\hline
\end{array}
$$

Fig. 8. Function map

into three sets p_0, p_1, and p_2. These are consequently filled with measurements w_i according to: $p_0 := \{w_i | \mathbf{A} = \mathbf{B} = 0\}$, $p_1 := \{w_i | \mathbf{A} = \mathbf{B} = 1\}$ while all other measurements in $p_2 := \{w_i | \mathbf{A} \neq \mathbf{B}\}$ (usually 50% of the set) are discarded. Then we compute the means m_0 of p_0 and m_1 of p_1 and finally the DPA bias $\Theta = m_0 - m_1$. The upper plot in Fig. 9 shows the DPA bias signal for the set of keys which are correct in the relevant bits and the lower plot for the set of keys with an error in the least significant bit. The presence of so called "ghost peaks" is explained by the fact that

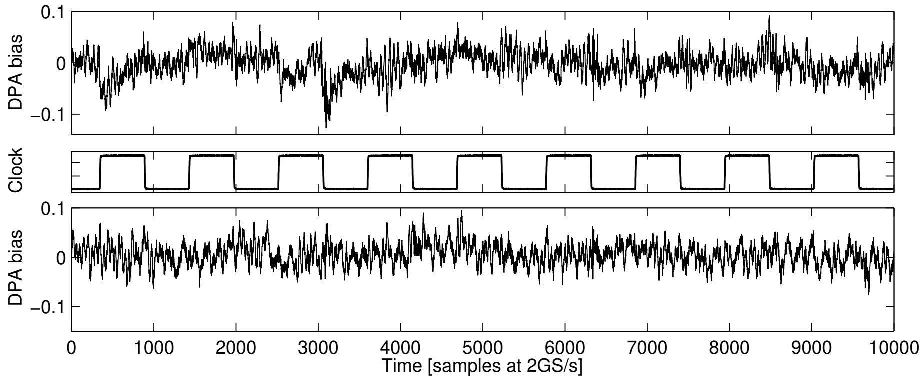

Fig. 9. DPA bias AND gate, correct key set (upper) and 1-bit error key set (lower); clock signal (middle)

we attack an intermediate result before any non-linear function has been computed and by the fact that the attack focuses on only a few bits in a pipelined VLSI circuit. However, the peak for the set of "correct" keys is clearly distinguishable from the other one. Finally, Fig. 10 shows the value of Θ at the peak's time index for each key hypothesis in the upper plot. As expected, the plot shows a "digital" pattern which divides the key space in three parts. The lower plot shows the number of bits in **A** and **B** which have been guessed correctly, for each key hypothesis. The similarity in the patterns is obvious. All key guesses which are incorrect in one bit lead to a small (absolute) value Θ and can be rejected. Rejection of more key candidates is not possible because we do not assume knowledge on $\alpha, \delta,$ and γ. Hence, the adversary cannot predict, whether the set of "correct" keys must lead to a maximum or minimum DPA peak (cf. Sect. 4.2). However, she may attack a different AND gate for further sieving.

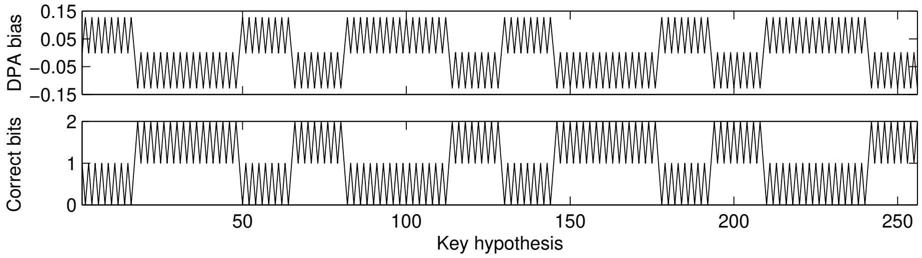

Fig. 10. DPA bias at peak position (upper) and number of correct bits (lower) for all key hypotheses

According to our analysis of the MDPL AND gate in Sect. 4.2, this attack result indicates that the PRNG in the VLSI circuit implementation is biased. In order to verify this conclusion we simulated a gate-level netlist of the PRNG implementation using the same seed data. The statistical analysis of one million output bits of this simulated netlist however showed no *distinct* bias. The sample contained 50.01% zeros, which means $\alpha = 0.5001$. It is unclear whether such a marginal bias can enable our attack methodology or not.

Summarizing we can say that the success of our attack and the specific values we obtained for Θ, namely one value very close to "0" while the two others differ only in their signs, indicate an exploitable bias in the output bits of the PRNG implementation. On the other hand, one might believe the marginal bias in the output bits of the netlist simulation to be negligible, which would then contradict this conclusion. Finally, one must also consider the possibility that the implementation of the PRNG on the prototype chip does not exactly behave like the simulated netlist.

In the next section we identify approaches to explain the success of our attack assuming that the output bits of the PRNG implementation are not or not significantly biased.

6 Investigation

Assuming that the PRNG implementation on the prototype chip is not biased, we need to look for another explanation of our results. In the following, we sketch an approach which will be subject of our future research.

A possible explanation for our observation and the successful attack is the early propagation effect. In short, this effect describes the fact that certain gates possibly evaluate at a data dependent instant. MDPL gates have been studied with regard to this effect in [8]. The authors concluded that a timing delay of the input signals can yield an early propagation effect. Given that an adversary uses the right criterion to partition the power traces, that the attacked gate is actually vulnerable to the effect, and that the timing difference is large enough to be detected by her measurement setup, she can exploit the effect. In this case, a power analysis attack against a DSCA "resistant" circuit turns into an attack which is similar to power analysis attacks against non-constant time implementations, as for example naïve "square-and-multiply".

We assume that the early propagation effect becomes apparent in time and data dependent histograms of the power measurements. Therefore, we generated data dependent histograms of the sets $p_0 := \{w_i | \mathbf{A} = \mathbf{B} = 0\}$ and $p_1 := \{w_i | \mathbf{A} = \mathbf{B} = 1\}$ for a correct guess on $\mathbf{A}$ and $\mathbf{B}$ at the time instant of the DPA peak and neighboring samples. Yet, first inspections did not lead to a clear conclusion. A thorough investigation will be subject of our future work.

However, the observations gave rise to another possible explanation for the success of our attack. Studying the switching behavior of a majority gate in detail, we discovered a potential problem. There are "internal nodes" in the pull up and pull down networks, which can not always be fully charged respectively discharged, depending on the input signals' values and delays. These internal nodes are marked with a cross in Fig. 1. This fact induces what we will call a memory effect. Possibly, there exist combinations of delays in the input signals, for which a (small) bias in the distribution of the random masks leads to a data dependent memory effect. In that case, the side channel leakage of an AND gate would be increased. Note that such delay combinations need not necessarily lead to early propagation. We will investigate the memory effect and the requirements for the input signals' delays in the near future.

Summarizing we have to say that it remains unclear whether the success of our attack is based on a bias in the output bits of the PRNG implementation or not. To clarify this uncertainty will be subject of our future research.

7 Conclusion

We developed a model for the output transition energies of non-linear MDPL gates. We have shown that the transition energies depend on the bias α in the source of the randomness and that they can be reliably exploited to derive signal values. The requirements for our attack methodology are slight and realistic (unknown) PRNG biases, which have been assumed in attacks against masking schemes before, and knowledge of the circuit layout, which should be assumed by default to model a powerful adversary. We have empirically verified our theoretic approach with practical measurement results. We showed that MDPL is vulnerable to our attack methodology in practical cases where the randomness is not perfect. Further on, we have identified approaches to explain the success of our attack assuming not or not significantly biased random masks. A more detailed investigation about the exact cause is subject of our future research.

Acknowledgements

The author would like to thank George Danezis and Kazuo Sakiyama for inspiring discussions.

This work was supported in part by the IAP Programme P6/26 BCRYPT of the Belgian State (Belgian Science Policy), by FWO projects G.0475.05 and G.0450.04, by the European Commission FP6 MIRG project SESOC, number MIRG-CT-2004-516568, and by the K.U. Leuven-BOF.
The information in this document reflects only the author's views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

References

1. T. Popp, S. Mangard: Masked Dual-Rail Pre-charge Logic. In: J. R. Rao and B. Sunar (eds.): Cryptographic Hardware and Embedded Systems – CHES 2005, LNCS 3659, pp. 172–186, Springer, 2005
2. W. Fischer, B. Gammel: Masking at Gate Level in the Presence of Glitches. In: J. R. Rao and B. Sunar (eds.): Cryptographic Hardware and Embedded Systems – CHES 2005, LNCS 3659, pp. 187–200, Springer, 2005
3. M. Aigner, S. Mangard, F. Menichelli, R. Menicocci, M. Olivieri, T. Popp, G. Scotti, A. Trifiletti: Side channel analysis resistant design flow. IEEE International Symposium on Circuits and Systems, ISCAS 2006, 4 pages
4. S. Mangard, M. Aigner, S. Dominikus: A highly regular and scalable AES hardware architecture. IEEE Transactions on Computers, Vol. 52, Issue 4, pp. 483–491, 2003
5. J. Wolkerstorfer, E. Oswald, M. Lamberger: An ASIC Implementation of the AES SBoxes. In B. Preneel (ed.): Topics in Cryptology - CT-RSA 2002, LNCS 2271, pp. 67–78, Springer, 2002
6. E. Brier, C. Clavier, F. Olivier: Correlation Power Analysis with a Leakage Model. In: M. Joye and J.-J. Quisquater (eds.): Cryptographic Hardware and Embedded Systems — CHES 2004, LNCS 3156, pp. 16–29, Springer, 2004
7. K. Tiri, P. Schaumont: Changing the odds against Masked Logic. Selected Areas of Cryptography (SAC) 2006, LNCS, Springer, to appear
8. D. Suzuki, M. Saeki: Security Evaluation of DPA Countermeasures Using Dual-Rail Pre-charge Logic Style. In: L. Goubin and M. Matsui (eds.): Cryptographic Hardware and Embedded Systems – CHES 2006, LNCS 4249, pp. 255–269, Springer, 2006
9. Y. Ishai, A. Sahai, D. Wagner: Private circuits: securing hardware against probing attacks. In Proceedings of CRYPTO 2003, LNCS 2729, pp. 463–481, Springer, 2003
10. E. Trichina, T. Korkishko, K.-H. Lee: Small size, low power, side channel immune AES coprocessor design and synthesis results. In: H. Dobbertin and V. Rijmen and A. Sowa (eds.): Advanced Encryption Standard - AES4, LNCS 3373, pp. 113–127, Springer, 2005
11. S. Mangard, T. Popp, B. Gammel: Side-Channel leakage of masked CMOS gates. In: A. Menezes (ed.): Topics in Cryptology - CT-RSA 2005, LNCS 3376, pp. 351–365, Springer, 2005
12. S. Mangard, N. Pramstaller, E. Oswald: Successfully attacking masked AES hardware implementations. In: J. R. Rao and B. Sunar (eds.): Cryptographic Hardware and Embedded Systems – CHES 2005, LNCS 3659, pp. 157-171, Springer, 2005
13. P.C. Kocher, J. Jaffe, B. Jun: Differential Power Analysis. In: M. Wiener (ed.): Advances in Cryptology – CRYPTO '99, LNCS 1666, pp. 388–397, Springer, 1999
14. W. Schindler, K. Lemke, C. Paar: A Stochastic Model for Differential Side Channel Cryptanalysis. In: J.R. Rao, B. Sunar (eds.): Cryptographic Hardware and Embedded Systems – CHES 2005, LNCS 3659, pp. 30–46, Springer, 2005
15. S. Chari, J.R. Rao, P. Rohatgi: Template Attacks. In: B.S. Kaliski Jr., Ç.K. Koç, C. Paar (eds.): Cryptographic Hardware and Embedded Systems – CHES 2002, LNCS 2523, pp. 13–28, Spinger, 2003
16. M. Akkar, C. Giraud: An implementation of DES and AES, secure against some attacks, In: Ç. Koç, D. Naccache, C. Paar, (Eds.): Cryptographic Hardware and Embedded Systems – CHES 2001, LNCS 2162, pp. 309–318, Springer, 2001
17. E. Oswald, S. Mangard, N. Pramstaller, V. Rijmen: A Side-Channel Analysis Resistant Description of the AES S-box. In: H. Gilbert and H. Handschuh (eds.): Fast Software Encryption 2005, LNCS 3557, pp. 413–423, Springer, 2005

18. C. Clavier, J.S. Coron, N. Dabbous: Differential Power Analysis in the Presence of Hardware Countermeasures. In Ç. Koç, D. Naccache, C. Paar, (Eds.): Cryptographic Hardware and Embedded Systems – CHES 2000, LNCS 1965, pp. 253–263, Springer, 2000
19. T.S. Messerges, E.A. Dabbish, R.H. Sloan: Examining Smart-Card Security under the Threat of Power Analysis Attacks. In: Proc. USENIX Workshop on Smartcard Technology, pp. 151–161, 1999
20. K. Tiri, I. Verbauwhede: A Digital Design Flow for Secure Integrated Circuits, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 25(7), pp. 1197-1208, 2006
21. D. Suzuki, M. Saeki, T. Ichikawa: Random Switching Logic: A Countermeasure against DPA based on Transition Probability. Cryptology ePrint Archive, Report 2004/346, 2004