Pinpointing the Side-Channel Leakage of Masked AES Hardware Implementations *

Stefan Mangard[1] and Kai Schramm[2]

[1] Institute for Applied Information Processing and Communciations (IAIK), Graz University of Technology, Inffeldgasse 16a, 8010 Graz, Austria
[2] Horst Görtz Institute for IT Security (HGI), Universitätsstr. 150, Ruhr University Bochum, 44780 Bochum, Germany
stefan.mangard@iaik.tugraz.at, schramm@crypto.ruhr-uni-bochum.de

Abstract. This article starts with a discussion of three different attacks on masked AES hardware implementations. This discussion leads to the conclusion that glitches in masked circuits pose the biggest threat to masked hardware implementations in practice. Motivated by this fact, we pinpointed which parts of masked AES S-boxes cause the glitches that lead to side-channel leakage. The analysis reveals that these glitches are caused by the switching characteristics of XOR gates in masked multipliers. Masked multipliers are basic building blocks of most recent proposals for masked AES S-boxes. We subsequently show that the side-channel leakage of the masked multipliers can be prevented by fulfilling timing constraints for $3 \cdot n$ XOR gates in each $GF(2^n)$ multiplier of an AES S-box. We also briefly present two approaches on how these timing constraints can be fulfilled in practice.

Keywords: AES, DPA, Glitches, Zero-Offset DPA, Zero-Input DPA, Masking, Delay Chains

1 Introduction

The Advanced Encryption Standard (AES) [13] is the most commonly used block cipher in modern applications. This is why there has been a significant effort during the last years to design implementations of this algorithm that are resistant against power analysis attacks [7].

One approach to secure implementations of AES against power analysis attacks is to mask the intermediate values that occur during the execution of the algorithm. Masking schemes for AES have been presented in [2], [22], [5], [11], [3], and [15]. The first two of these schemes have turned out to be susceptible to so-called zero-value attacks [5] and the second one is even susceptible to standard DPA attacks [1]. The third scheme is quite complex to implement and there are no published implementations of this approach so far. The last three schemes are provably secure against DPA attacks and the schemes can also be efficiently

* The work described in this paper has been supported in part by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

implemented in hardware. This is why these schemes are the most commonly used schemes to secure implementations of AES in hardware.

However, in 2005 several publications have shown that even provably secure masking schemes can be broken in practice, if they are implemented in standard CMOS. The reason for this is that in CMOS circuits a lot of unintended switching activities occur. These unintended switching activities are usually referred to as dynamic hazards or glitches. The effect of glitches on the side-channel resistance of masked circuits has first been analyzed in [8]. A similar analysis has also been presented in [19]. A technique to model the effect of glitches on the side-channel resistance of circuits has been published in [20]. The fact that glitches can indeed make circuits susceptible to DPA attacks in practice was finally shown in [9].

After the publication of these articles it was clear that considering the effect of glitches is crucial when implementing masking schemes in hardware. However, one important question has remained unanswered so far. The existing articles only show that implementations of masking schemes leak side-channel information. They do not pinpoint the exact gates or parts of the masked circuits that account for the leakage. In [9] for example, it has been shown that a CMOS implementation of [15] can be attacked because of glitches. However, it is not clear which gates within the masked S-box implementation actually account for this fact.

The current article answers this question by performing a close analysis of masked multipliers which are the basis of masking schemes such as [11], [15], and [3]. In fact, we show that the switching characteristics of the XOR gates in these multipliers account for the side-channel leakage. This insight and the fact how this insight can be used to develop DPA-resistant implementations of masking schemes constitute the main contribution of this article.

However, before we start our analysis of the masked multipliers, Sect. 2 first briefly recapitulates the different DPA attacks on masked AES hardware implementations that have been published recently. In particular, this section compares the attack presented in [9] with the zero-offset DPA attack presented in [23]. Both attacks are performed on a masked AES hardware implementation according to [15]. The comparison turns out that the first attack is significantly more effective. In fact, we are even able to show that a much simpler power model of the masked S-box leads to successful attacks as well.

Motivated by this fact Sect. 3 analyzes which parts of the AES S-box actually cause the side-channel leakage. As already pointed out, this analysis leads to the conclusion that the XOR gates within the masked multipliers of the AES S-box account for the leakage. This insight is used in Sect. 4 to present new approaches in order to securely implement masking schemes. Sect. 5 summarizes the most important results of this article and provides some conclusions.

2 Attacks on Masked AES Hardware Implementations

This section discusses results of three DPA attacks against a masked AES hardware implementation. The device under attack was an AES ASIC that is based on the masking scheme that has been proposed in [15]. The chip uses a 32-bit architecture and hence the computation of one AES round takes four clock cycles, and a complete AES encryption takes 40 clock cycles. All of our DPA attacks are based on a set of 1,000,000 power traces which we collected from the masked AES chip. The traces have been measured at 1 GS/s using a differential probe.

The first attack we discuss is the zero-offset DPA (ZODPA) as proposed in [23]. This attack requires that masks and masked data of the attacked device leak simultaneously and it uses squaring as a preprocessing step. Subsequently, we discuss a DPA attack based on a toggle-count power model of a masked S-box of our chip. This attack has been performed in the same way as it has been proposed in [9]. Finally, we present a simplification of this attack, which we refer to as zero-input DPA. This attack is based on the fact that the power consumption of our masked AES S-box implementation has a significant minimum for the case that the mask and the masked input are equal.

2.1 Zero-Offset DPA

Zero-offset DPA was originally proposed by Waddle *et al.* in [23] and it represents a special case of second-order DPA [10, 6, 14, 18]. This can be shown as follows. Let us assume the power consumption at time t_0 of the attacked device can be described as

$$P(t_0) = \epsilon \cdot (W(M) + W(Y)) + N \tag{1}$$

where $W(M)$ represents the Hamming weight of a random mask M, $W(Y)$ represents the Hamming weight of key-dependent data masked by M, ϵ is a constant of proportionality, and N represents additive Gaussian noise. When squaring this power signal, it can be observed that a zero-offset DPA is essentially equivalent to a second-order DPA. Both attacks rely on the term $W(M) \cdot W(Y)$.

$$P^2(t_0) = \epsilon^2 \cdot (W(M) + W(Y))^2 + 2 \cdot \epsilon \cdot (W(M) + W(Y)) \cdot N + N^2 \tag{2}$$

$$= \epsilon^2 \cdot \left(W^2(M) + 2 \cdot W(M) \cdot W(Y) + W^2(Y)\right) + 2 \cdot \epsilon \cdot (W(M) + W(Y)) \cdot N + N^2 \tag{3}$$

However, zero-offset DPA can only be used, if the mask and the masked data are processed simultaneously. While this scenario is unlikely to happen in masked software implementations, it commonly occurs in masked hardware implementations. In particular, it also occurs in our attacked AES ASIC and hence a zero-offset DPA should theoretically be possible. Consequently, we have squared our power traces and have computed the correlation coefficient between the squared traces and corresponding hypotheses. However, even with 1,000,000 measurements we have not been able to perform a successful zero-offset DPA.

2.2 Toggle-Count DPA

In conventional CMOS circuits, signal lines typically toggle several times during a clock cycle. In [8] it has been shown that the total number of signal toggles in

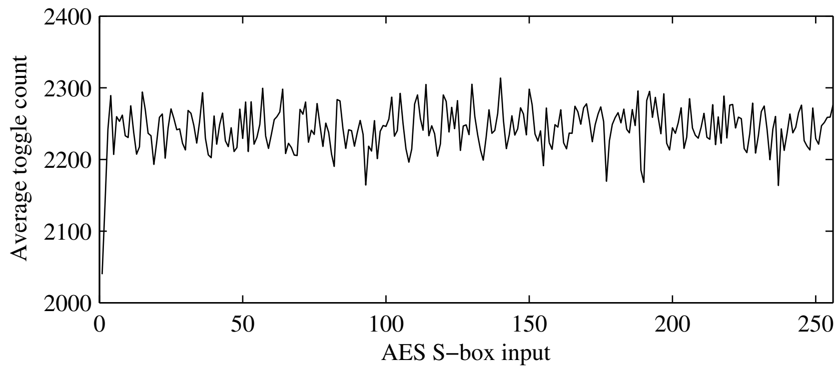

Fig. 1. Average number of toggles in our masked S-box circuit.

masked non-linear gates, *e.g.* in masked AND or masked OR gates, is correlated to the unmasked input and output signals. This fact has been exploited in a simulated DPA attack.

A similar approach has been pursued in [9] to break masked AES hardware implementations in practice. A back-annotated netlist of the attacked device has been used in order to derive a toggle-count model of masked AES S-boxes. Subsequently, these models were used in DPA attacks to reveal the secret key of an AES chip[3].

In order to confirm these results, we have performed these attacks on our masked ASIC implementation again. We have first simulated our chip to determine the average number of toggles that occur in our masked AES S-box for different data inputs. The power model of our S-box is shown in Fig. 1. In this figure, the number of toggles of our masked S-box are shown for all possible 256 S-box inputs. Please note that there occurs a distinct minimum for S-box input 0, *i.e.* the case when mask and masked data are equal.

We have used the power model shown in Fig. 1 to mount a DPA attack on our masked AES chip. We have correlated the measured power traces of our masked AES implementation with hypotheses based on the power model. In this attack, we have obtained a correlation coefficient of $r = 0.04$ for the correct key hypothesis using $1,000,000$ measurements. Approximately $15,000$ measurements were necessary to distinguish this correlation coefficient from the false correlation coefficients. The correlation coefficients for an attack based on $15,000$ measurements are shown in Fig. 2.

[3] Note that the toggle-count model assumes that each signal toggle has an equal contribution to the power consumption. This condition is typically not met in real life. Nevertheless, the model is usually sufficient mount successful DPA attacks on masked implementations.

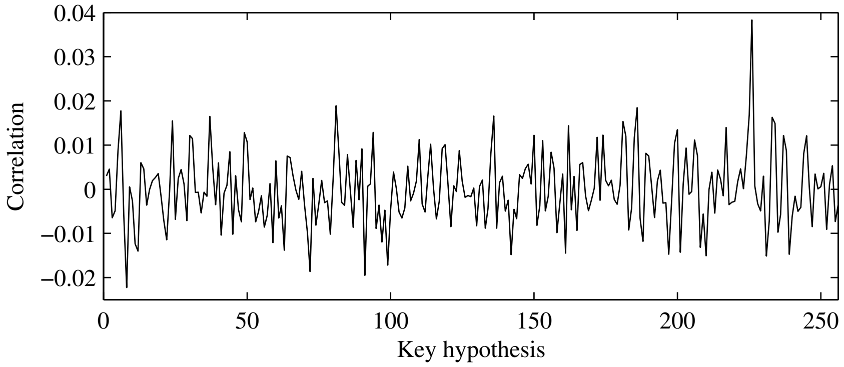

Fig. 2. Correlation coefficients of the toggle-count DPA against the masked AES ASIC with 15,000 measurements. The correct key hypothesis (225) is clearly distinguishable from all false key hypotheses.

2.3 Zero-Input DPA

As shown in Fig. 1, the simulated masked AES S-box has a significant power consumption minimum, if the S-box input $x = x_m \oplus m_x = 0$. This significant minimum suggests that it should also be possible to perform DPA attacks that just exploit this property. Hence, we have adapted our power model of the S-box to the following much simpler model $P(x)$.

$$\begin{aligned} P(x) &= 0 \qquad && if \quad x = 0 \\ &= 1 \qquad && if \quad x \neq 0 \end{aligned}$$

Using this generic zero-input power model we have repeated our attack based on the same set of power traces. We have obtained a correlation coefficient of $r = 0.022$ for the correct key hypothesis. About $30,000$ measurements were necessary to clearly distinguish this correlation coefficient from the ones of false key hypotheses. Fig. 3 shows the result of an attack based on $30,000$ measurements.

The number of measurements that are needed for a zero-input DPA is greater compared to the attack based on the more precise power model. However, the attack is still feasible and it is much more effective than a zero-offset DPA attack. The biggest advantage of the zero-input DPA over the two other attacks we have discussed, is that the zero-input DPA does not require detailed knowledge about the attacked device and it is still very effective. It exploits the fact that the power consumption of the masked S-box implementation has a significant minimum for the input value zero. In the following section, we analyze why implementations of masked S-boxes actually leak side-channel information and we pinpoint where the side-channel leakage is caused.

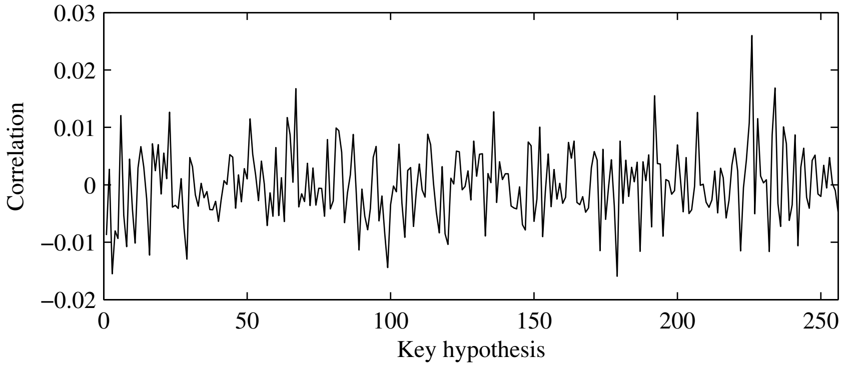

Fig. 3. Correlation coefficients of a zero-input DPA against the masked AES ASIC with 30,000 measurements. The correct key hypothesis (225) is clearly distinguishable from the false correlation coefficients.

3 Pinpointing the Side-Channel Leakage of Masked S-boxes

The masked AES S-box implementation we have attacked in the previous section is based on composite field arithmetic. In fact, most recent proposals for masked AES S-boxes (see [11], [15], and [3]) are based on this approach. Masked AES S-boxes of this kind essentially consist of an affine transformation, isomorphic mappings, adders and multipliers. All these elements except for the multipliers are linear and hence it is easy to mask them additively. An additive masking of a linear operation can be done by simply performing the operation separately for the masked data and the mask.

In hardware, masked linear operations are usually implemented by two completely separate circuits. One circuit performs the linear operation for the masked data and one circuit performs the linear operation for the corresponding mask. There is no shared signal line between these two circuits. Therefore, the power consumption P_1 of the first circuit exclusively depends on the masked data and the power consumption P_2 of the second circuit exclusively depends on the mask. According the definition of additive masking [2], the masked data and the mask are pairwise statistically independent from the corresponding unmasked data. Hence, P_1 and P_2 are also pairwise independent from the unmasked data.

In practice this means that an attacker who does not know the mask can not perform a successful first-order DPA attack on the power consumption of either of these two circuits. An attacker can only formulate hypotheses about unmasked intermediate values of the performed cryptographic algorithm. In this article, we denote the set of all unmasked intermediate values of the attacked algorithm as $\mathcal{H}$. Our previous argumentation hence formally means that $\rho(H, P_1)$ and $\rho(H, P_2)$ are both 0 for all $H \epsilon \mathcal{H}$. This also implies that the total power consumption is uncorrelated to all intermediate values, *i.e.* $\rho(H, P_1 + P_2) =$

$0 \ \forall H \epsilon \mathcal{H}$. Throughout this article, we use the common assumption that the total power consumption of a circuit is the sum of the power consumption of its components. Using this assumption, it is clear that the linear elements of a masked S-box do not account for the side-channel leakage we have observed in the toggle-count and zero-input DPA attacks presented in Sect. 2. As the power traces are not pre-processed in these attacks, the side-channel leakage can only be caused by the non-linear elements, *i.e.* the multipliers which combine masks and masked data.

In general, there exist several approaches to mask a multiplier. However, there is also one very common approach. Fig. 4 shows the architecture of a masked $GF(2^n)$ multiplier according this common approach. The multiplier takes two masked inputs a_m and b_m that are masked with m_a and m_b, respectively. The output q_m is the product of the corresponding unmasked values a and b masked with m_q.

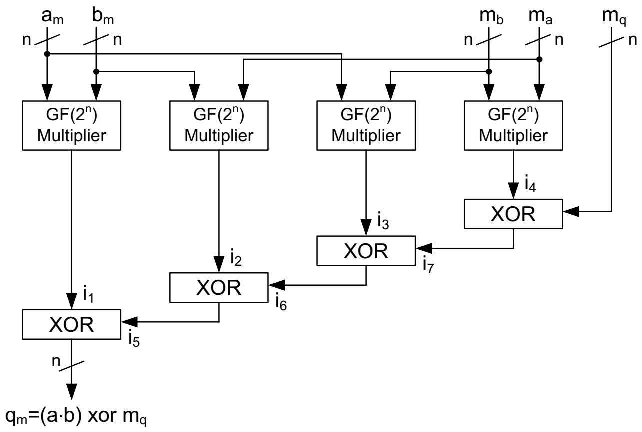

Fig. 4. Common architecture of a masked multiplier.

The masked multiplier consists of four unmasked multipliers that calculate the intermediate values $i_1 \ldots i_4$. These intermediate values are then summed by $4 \cdot n$ XOR gates. A masked multiplier of this kind has been used as a masked AND gate ($n = 1$) in [21]. Furthermore, this architecture is also used in the masked S-boxes presented in [11], [3], and [15]. This is why we now analyze this architecture more closely. We start our analysis by first looking at a masked AND gate ($n = 1$). Subsequently, we look at multipliers in $GF(2^2)$ and $GF(2^4)$. Finally, we look at the side-channel leakage of masked S-boxes as a whole that contain several such masked multipliers.

3.1 Masked AND Gate

Masked AND gates that are based on the architecture shown in Fig. 4 have already previously been analyzed in [8] and [20]. These analyses have revealed that such gates indeed leak side-channel information. However, in neither of these publications the source of the leakage has been pinpointed exactly. Both publications essentially state that there occurs leakage due to timing properties. Yet, these properties are not analyzed further. In the current article, we pinpoint the exact cause of the side-channel leakage.

For this purpose we have implemented a masked AND gate based on the architecture shown in Fig. 4. We have then simulated the back-annotated netlist of this gate for all possible input transitions. There are five input signals and hence there are 2^{10} possible input transitions[4]. For each of these 2^{10} cases we have counted the number of transitions that occur on each signal line in the design. We denote the these numbers of transitions with $T(a_m)$, $T(b_m)$, $T(m_a)$, $T(m_b)$, $T(m_q)$, $T(q_m)$, and $T(i_1)\dots T(i_7)$.

In order to analyze which signal lines account for the side-channel leakage of the gate, we have calculated the correlation between these numbers on the one hand and the unmasked values a, b and q on the other hand. Due to the masking $T(a_m)$, $T(b_m)$, $T(m_a)$, $T(m_b)$, and $T(m_q)$ do not leak side-channel information. Furthermore, it turns out that also $\rho(T(i_j), a) = 0$, $\rho(T(i_j), b) = 0$ and $\rho(T(i_j), q) = 0$ for $j = 1 \dots 4$. This result is actually not surprising. The four multipliers (the four AND gates in case of $n = 1$) never take a masked value and a corresponding mask as input. For example, there is no multiplier that takes a_m and m_a as input. Each pair of inputs of the multipliers is not only pairwise independent of a, b and q, but it is completely statistically independent of these values. Therefore, also the power consumption of the multipliers and their outputs are independent of a, b and q. The side-channel leakage can only be caused by the XOR gates.

At first sight this might seem counter-intuitive because the number of transitions that occur at the output of an XOR gate intuitively correspond to the sum of transitions that occur at the inputs of the gate. Each input transition should lead to one output transition. The number of input transitions does not leak side-channel information and hence also the number of output transitions should not. Unfortunately, this reasoning is wrong in practice.

It is true that an XOR gate usually switches its output each time an input signal switches. However, the gate does not switch its output, if both input signals switch simultaneously or within a short period of time. In this case, the input transitions are "absorbed" by the XOR gate and not propagated further. Exactly this effect accounts for the side-channel leakage of the masked AND gate. Our simulations have shown that the number of absorbed transitions is indeed correlated to a, b and q. This means that the arrival times of the input signals at the XOR gates depend on the unmasked values. It is the joint distribution of the arrival times of the signals $i_1 \dots i_4$ that causes the side-channel leakage of

[4] In our simulation all input signals are set at the same time.

the gate. The arrival times are different for different unmasked values and hence a different number of transitions is absorbed. This in turn leads to a different power consumption.

It is important to point out that it is exclusively this effect that accounts for the side-channel leakage of the masked AND gate. If each XOR gate would switch its output as often as its inputs switch, the gate would be secure. This is a consequence of the fact that $T(i_1) \ldots T(i_4)$ are uncorrelated to a, b and q.

3.2 Masked Multipliers for $GF(2^2)$ and $GF(2^4)$

In order to confirm the insights gained from the analysis of the masked AND gate, we have also implemented masked multipliers for $GF(2^2)$ and $GF(2^4)$. Multipliers of this kind are used in the masked AES S-boxes of [11], [3], and [15]. As in the case of the masked AND gates, we have performed different simulations based on back-annotated netlists of these multipliers.

First, we have confirmed that $T(i_1) \ldots T(i_4)$ are indeed independent of a, b and q. This analysis was actually just done for sake of completeness. From a theoretical point of view it is clear that the power consumption of the four multipliers shown in Fig. 4 is independent of the unmasked values. As already pointed out before, the inputs of each multiplier are completely statistically independent from the unmasked values. This fact is independent of the bit width of the multipliers.

In the second step, we have again analyzed the switching characteristics of the XOR gates. Our simulations have confirmed that the number of absorbed transitions depends on the unmasked values a, b and q—exactly as in the case of the masked AND gate. The side-channel leakage of all masked multipliers that are based on the architecture shown in Fig. 4 is obviously caused by the same effect.

However, unfortunately it is not possible to make a general statement on how much information such masked multipliers leak. The fact how many transitions are absorbed by the XOR gates depends on many implementation details. The arrival times of the signals at the XOR gates strongly depend on the placement and routing of the circuit. Of course also the used CMOS library has a strong impact. The library affects the timing of the input signals and it also determines how big the delay between two input transitions of an XOR gate has to be in order propagate.

Based on our experiments, we can make one general statement. We have implemented several masked multipliers and we have also placed and routed them several times. In all cases, we have observed side-channel leakage. In order to prevent that the XOR gates absorb transitions, it is therefore necessary to explicitly take care of this issue during the design process (see Sect. 4).

3.3 Masked AES S-boxes

Masked AES S-boxes as they are presented in [11], [15], [3] contain several masked multipliers. We now analyze two concrete implementations of masked

AES S-boxes in order to check how the side-channel leakage of the multipliers affects the other components of the S-boxes. We first analyze an implementation of the AES S-box proposed in [15] and then we look at an implementation of [11].

Masked S-box of Oswald *et al.* The first step of our analysis was to generate a back-annotated netlist of the masked AES S-box described in [15]. Subsequently, we have simulated this netlist for $200,000$ randomly selected input transitions. During these simulations, we have counted the number of transitions that occur on each of the internal signal lines of the S-box. Based on these numbers it was possible to determine which signal lines cause the most side-channel leakage.

As expected, all the linear operations that are performed at the beginning of the S-box do not leak any information. The transitions that occur on the corresponding signal lines are independent of the unmasked S-box input. The first leakage within the S-box occurs in the first masked multiplier. The XOR gates of this multiplier absorb a different number of transitions for different data inputs. The number of transitions that occur on the output signal of the masked multiplier is therefore correlated to the unmasked version of the S-box input.

The fact that the switching activity of this signal is correlated to the unmasked S-box input has severe consequences for all components that use this signal as input. The switching activity of all these components typically also becomes correlated to the unmasked S-box input[5]. This holds true for linear and non-linear components. Therefore, the leakage that is caused by the first masked multiplier spreads out like an avalanche through the remaining S-box.

This leakage is additionally amplified by the leakage of all other masked multipliers in the S-box. In fact, the leakage continuously grows on its way through the S-box. In case of our S-box implementation of [15] this leads to the power consumption characteristic we have already shown in Fig. 1. A different amount of transitions occurs for every unmasked S-box input. A significant minimum for the number of transitions occurs for the case that the input value is 0. In this case, the masked S-box input and the corresponding mask are equal. The arrival times of the signals in the masked multipliers are more uniform in this case than in all other cases. Therefore, more transitions are absorbed by the XOR gates and also less transitions propagate through the components that are connected to the multipliers.

Masked S-box of Morioka and Akishita We have also analyzed the masked AES S-box proposed by Morioka and Akishita in [11]. The architecture of this S-box is based on the unmasked S-box proposed by Satoh *et al.* in [17]. As in the case of the masked S-box by Oswald *et al.* [15] we have first generated a

[5] There are of course also gates that do not propagate the leakage. For example, the output signal of a NAND gate that is connected to a leaking signal on input one and to 0 on input two does not leak any information. However, there are typically sufficient gates connected to a leaking signal that at least some of the gates propagate the leakage.

back-annotated netlist of the design. Subsequently, we have simulated $200,000$ random input transitions and we have counted the number of transitions for each signal line. Again, we have noticed that the total number of transitions in the masked S-box circuit is clearly correlated to the unmasked S-box input. As a matter of fact, we were able to successfully mount a simulated zero-input attack on this masked S-box. The attack only required a few thousand simulated power traces, *i.e.* simulations of transition counts. This result also confirms our aforementioned claim that a precise power model of a masked S-box implemented in CMOS is not always necessary to successfully perform a DPA attack.

In order to investigate why the number of toggles has a minimum, if the mask and the masked input are equal, we have evaluated transition count data of various S-box subcircuits. We have then performed zero-input attacks against these subcircuits. Exactly as in the case of the masked S-box by Oswald *et al.* we have found out that glitches are absorbed in XOR gates of a masked finite field multiplier. Our analysis has confirmed that the number of absorbed transitions is again correlated to the unmasked S-box input and that there is a significant power consumption minimum for input 0. The masked S-box of Morioka and Akishita is highly symmetric with regard to the signal paths of the mask and the masked input. This symmetry seems to be the main reason why transitions are absorbed by the XOR gates, if the mask and the masked input are equal.

In general, it is difficult to make a general statement on whether all masked S-boxes have a significant minimum of the power consumption for the case that the input is 0. Many implementation details influence the exact switching characteristic of an S-box. However, based on our observations we assume that most masked S-boxes are vulnerable to zero-input attacks.

4 Countermeasures

In the previous section, we have analyzed the side-channel leakage of masked multipliers that are based on the architecture shown in Fig. 4. It has turned out that the XOR gates summing the outputs of the four unmasked multipliers of this architecture, account for the side-channel leakage. These XOR gates absorb transitions and the number of absorbed transitions is correlated to the unmasked operands of the masked multiplier.

In Sect. 3, we have already pointed out that it is exclusively this absorbtion that causes the side-channel leakage. A masked multiplier is secure against DPA attacks, if no transitions are absorbed by the XOR gates. This means that the number of transitions at the output of an XOR gate needs to be equal to the total number of transitions occurring at the inputs. A masked multiplier that implements XOR gates in this way is secure. The transitions of the signal lines $i_1 \dots i_4$ are uncorrelated to a, b and q. If the XOR gates propagate these transitions to the output q_m without any absorbtion, the whole multiplier is secure.

In a masked $GF(2^n)$ multiplier, there are $4 \cdot n$ XOR gates that sum the signals $i_1 \dots i_4$ and m_q. When looking at Fig. 4, it is clear that the n XOR gates that sum i_4 and m_q, are actually not critical. The input signals of these gates depend on mask values only and hence the absorbed number of transitions of these gates cannot depend on a, b or q. As a consequence, there are actually only $3 \cdot n$ XOR gates in a masked multiplier that must not absorb any transitions. These are the gates summing i_1, i_2, i_3 and i_7. Preventing an absorbtion at these gates means that the inputs of these gates must not arrive simultaneously or within the propagation delay of the XOR gate. This is the timing constraint that needs to be fulfilled by the input signals.

In general, timing constraints are quite challenging to fulfill in practice. However, there exist two approaches that can be used to reach this goal. The first approach is to insert delay elements into the paths of the input signals of the XOR gate. A similar approach has actually already been used in [12] to reduce the power consumption of an unmasked AES S-box. In case of a masked multiplier, delay elements need be inserted into the lines i_1, i_2 and i_3 in such a way that the timing constraints for the XOR gates are fulfilled. We have successfully implemented a secure $GF(2)$ multiplier based on this approach. Simulations of this multiplier have confirmed that the transitions of all signal lines in the design are indeed independent of a, b and q.

However, it is important to point out that it is not always possible to efficiently fulfill the timing constraints of the XOR gates by inserting delay elements. For our masked multiplier we have assumed that all masked input signals arrive at the same time. However, the arrival times of the operands at a masked multiplier can vary significantly, if the multiplier is not connected to flip flops directly. If the multiplier is part of a long combinational path, the approach of inserting delay elements is usually not the best one to fulfill the timing constraints.

An alternative to inserting delay elements is to use enable signals in the circuit. The basic idea of this approach is to generate enable signals by a dedicated circuit that enable the inputs of the critical XOR gates just at the right time. Enable signals of this kind have for example also been used in [19] to control the switching activity of masked gates. Of course, the generation of enable signals requires a certain effort and it increases the design costs.

However, building secure masked circuits is always associated with costs. The proposal for secure masked gates presented in [4] is also associated with timing constraints that need to be fulfilled when building a masked circuit. One approach for secure masked circuits without timing constraints has been presented in [16]. However, this approach requires a pre-charging phase and hence the throughput of such implementations is halved compared to standard CMOS circuits.

5 Conclusions

In the first part of this article, we have presented results of three different DPA attacks on a masked AES ASIC implementation. One of these attacks was a simplification of the attack presented in [9]. Comparing this attack with zero-offset DPA has turned out that glitches are indeed the biggest problem of masked hardware implementations of AES. Motivated by this fact, we have pinpointed which parts of masked AES S-boxes cause glitches that lead to side-channel leakage. Our analysis has turned out that the glitches are caused by switching characteristics of XOR gates in masked multipliers.

We have subsequently shown that the side-channel leakage can be prevented by fulfilling timing constraints for $3 \cdot n$ XOR gates in each $GF(2^n)$ multiplier of an AES S-box. In practice, these timing constraints can essentially be fulfilled by two approaches: the insertion of delay elements and the usage of enable signals.

6 Acknowledgements

The authors would like to thank Elisabeth Oswald, Takashi Wanatabe, and Takashi Endo for the very helpful discussions.

References

1. Mehdi-Laurent Akkar, Régis Bevan, and Louis Goubin. Two Power Analysis Attacks against One-Mask Methods. In Bimal K. Roy and Willi Meier, editors, *Fast Software Encryption, 11th International Workshop, FSE 2004, Delhi, India, February 5-7, 2004, Revised Papers*, volume 3017 of *Lecture Notes in Computer Science*, pages 332–347. Springer, 2004.
2. Mehdi-Laurent Akkar and Christophe Giraud. An Implementation of DES and AES, Secure against Some Attacks. In Çetin Kaya Koç, David Naccache, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2001, Third International Workshop, Paris, France, May 14-16, 2001, Proceedings*, volume 2162 of *Lecture Notes in Computer Science*, pages 309–318. Springer, 2001.
3. Johannes Blömer, Jorge Guajardo, and Volker Krummel. Provably Secure Masking of AES. In Helena Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography, 11th International Workshop, SAC 2004, Waterloo, Canada, August 9-10, 2004, Revised Selected Papers*, volume 3357 of *Lecture Notes in Computer Science*, pages 69–83. Springer, 2005.
4. Wieland Fischer and Berndt M. Gammel. Masking at Gate Level in the Presence of Glitches. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, Scotland, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 187–200. Springer, 2005.
5. Jovan D. Golić and Christophe Tymen. Multiplicative Masking and Power Analysis of AES. In Burton S. Kaliski Jr., Çetin Kaya Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, volume 2535 of *Lecture Notes in Computer Science*, pages 198–212. Springer, 2003.
6. Marc Joye, Pascal Paillier, and Berry Schoenmakers. On Second-Order Differential Power Analysis. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 293–308. Springer, 2005.
7. Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In Michael Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual*

International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.

8. Stefan Mangard, Thomas Popp, and Berndt M. Gammel. Side-Channel Leakage of Masked CMOS Gates. In Alfred Menezes, editor, *Topics in Cryptology - CT-RSA 2005, The Cryptographers' Track at the RSA Conference 2005, San Francisco, CA, USA, February 14-18, 2005, Proceedings*, volume 3376 of *Lecture Notes in Computer Science*, pages 351–365. Springer, 2005.
9. Stefan Mangard, Norbert Pramstaller, and Elisabeth Oswald. Successfully Attacking Masked AES Hardware Implementations. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, Scotland, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 157–171. Springer, 2005.
10. Thomas S. Messerges. Using Second-Order Power Analysis to Attack DPA Resistant Software. In Çetin Kaya Koç and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2000, Second International Workshop, Worcester, MA, USA, August 17-18, 2000, Proceedings*, volume 1965 of *Lecture Notes in Computer Science*, pages 238–251. Springer, 2000.
11. Sumio Morioka and Toru Akishita. A DPA-resistant Compact AES S-Box Circuit using Additive Mask. In *Computer Security Composium (CSS), October 16, 2004, Proceedings*, pages 679–684, September 2004. (in Japanese only).
12. Sumio Morioka and Akashi Satoh. An Optimized S-Box Circuit Architecture for Low Power AES Design. In Burton S. Kaliski Jr., Çetin Kaya Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, volume 2535 of *Lecture Notes in Computer Science*, pages 172–186. Springer, 2003.
13. National Institute of Standards and Technology (NIST). FIPS-197: Advanced Encryption Standard, November 2001. Available online at `http://www.itl.nist.gov/fipspubs/`.
14. Elisabeth Oswald, Stefan Mangard, Christoph Herbst, and Stefan Tillich. Practical Second-Order DPA Attacks for Masked Smart Card Implementations of Block Ciphers. In David Pointcheval, editor, *Topics in Cryptology - CT-RSA 2006, The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13-17, 2006, Proceedings*, volume 3860 of *Lecture Notes in Computer Science*, pages 192–207. Springer, 2006.
15. Elisabeth Oswald, Stefan Mangard, Norbert Pramstaller, and Vincent Rijmen. A Side-Channel Analysis Resistant Description of the AES S-box. In Henri Gilbert and Helena Handschuh, editors, *Fast Software Encryption, 12th International Workshop, FSE 2005, Paris, France, February 21-23, 2005, Proceedings*, volume 3557 of *Lecture Notes in Computer Science*, pages 413–423. Springer, 2005.
16. Thomas Popp and Stefan Mangard. Masked Dual-Rail Pre-Charge Logic: DPA-Resistance without Routing Constraints. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, Scotland, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 172–186. Springer, 2005.
17. Akashi Satoh, Sumio Morioka, Kohji Takano, and Seiji Munetoh. A Compact Rijndael Hardware Architecture with S-Box Optimization. In Colin Boyd, editor, *Advances in Cryptology - ASIACRYPT 2001, 7th International Conference on the*

Theory and Application of Cryptology and Information Security, Gold Coast, Australia, December 9-13, 2001, Proceedings, volume 2248 of *Lecture Notes in Computer Science*, pages 239–254. Springer, 2001.

18. Kai Schramm and Christof Paar. Higher Order Masking of the AES. In David Pointcheval, editor, *Topics in Cryptology - CT-RSA 2006, The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13-17, 2006, Proceedings*, volume 3860 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2006.
19. Daisuke Suzuki, Minoru Saeki, and Tetsuya Ichikawa. Random Switching Logic: A Countermeasure against DPA based on Transition Probability. Cryptology ePrint Archive (`http://eprint.iacr.org/`), Report 2004/346, 2004.
20. Daisuke Suzuki, Minoru Saeki, and Tetsuya Ichikawa. DPA Leakage Models for CMOS Logic Circuits. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 366–382. Springer, 2005.
21. Elena Trichina, Tymur Korkishko, and Kyung-Hee Lee. Small Size, Low Power, Side Channel-Immune AES Coprocessor: Design and Synthesis Results. In Hans Dobbertin, Vincent Rijmen, and Aleksandra Sowa, editors, *Advanced Encryption Standard - AES, 4th International Conference, AES 2004, Bonn, Germany, May 10-12, 2004, Revised Selected and Invited Papers*, volume 3373 of *Lecture Notes in Computer Science*, pages 113–127. Springer, 2005.
22. Elena Trichina, Domenico De Seta, and Lucia Germani. Simplified Adaptive Multiplicative Masking for AES. In Burton S. Kaliski Jr., Çetin Kaya Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, volume 2535 of *Lecture Notes in Computer Science*, pages 187–197. Springer, 2003.
23. Jason Waddle and David Wagner. Towards Efficient Second-Order Power Analysis. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems – CHES 2004, 6th International Workshop, Cambridge, MA, USA, August 11-13, 2004, Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 2004.