Interaction between Fault Attack Countermeasures and the Resistance against Power Analysis Attacks

Francesco Regazzoni, Luca Breveglieri, Paolo Ienne, and Israel Koren

[1] Francesco Regazzoni, UCL Crypto Group, Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium and ALaRI - University of Lugano, CH-6904 Lugano, Switzerland
[2] Luca Breveglieri, DEI - Politecnico di Milano, 20133 Milano, Italy
[3] Paolo Ienne, École Polytechnique Fédérale de Lausanne (EPFL) School of Computer and Communication Sciences CH-1015 Lausanne, Switzerland
[4] Israel Koren, University of Massachusetts, Amherst, MA 01003, USA

1 Abstract

Most of the countermeasures against fault attacks on cryptographic systems that have been developed so far are based on the addition of information redundancy. While these countermeasures have been evaluated with respect to their cost (implementation overhead) and efficiency (fault coverage), little attention has been devoted to the question of the impact their use has on the effectiveness of other types of side-channel attacks, in particular power analysis attacks. This chapter presents an experimental study whose goal is to determine whether the added information redundancy can increase the vulnerability of a cryptographic circuit to power analysis attacks.

2 Introduction

In this chapter we discuss in a comprehensive way the interaction between countermeasures against fault injection attacks and the vulnerability to power analysis attacks, using AES as an example. We focus in particular on the non-linear transformation (S-box) within AES since it is the preferred attack point. Specifically, we concentrate on hardware implementations of AES to which error detection circuits have been added. Considered are the basic parity check, double parity, residue checks modulo 3 and 7, complementary parity and a Hamming error correcting code. For all the considered error detection or correction circuits, we analyze the effects that the redundant check bits may have on power analysis attacks using different metrics. These include an information theory based metric, the success rate of power analysis attacks based on correlation, and the effectiveness of the most common attacks based on difference of means and Hamming weights.

The effects that one specific countermeasure can have on the resistance to a different type of attack was studied in very few previous publications. Maingot *et al.* [5, 6] have analyzed the impact of four different error detection and correction schemes on power analysis resistance. Their study focused on a register storing the state of the AES encryption, which was enlarged to support the information redundancy necessary for each considered scheme. Using gate level simulations, they showed how the correlation between the value guessed by the adversary and the value of the register varies depending on the particular error detection code employed. They compared four different error detection codes in search for the best code for secure chips, and based on the correlation, concluded that a complementary parity scheme can improve the circuit robustness against power-based side channel attacks as well.

Transistor level simulations were performed by Regazzoni et al. [10, 11] to compare different error detection codes including parity codes and residue codes (e.g., mod 3 and 7) using a 180*nm* technology. As was done in [5], the authors focused on the output register of the S-box transformation in AES, and they have analyzed the impact that the considered codes may have on the resistance against power-based attacks and the role played by measurement noise. Furthermore, they discussed the question whether the knowledge of the particular error detection code used in the circuit affects the resistance against power-based side channel attacks and whether the redundancy helps the attacker even if he is unaware of its presence.

3 Considered error detection and correction circuits

Although we focus in this chapter on the *Rijndael* [3] block cipher (selected to be the Advanced Encryption Standard [8]), our conclusions are general and applicable to other block ciphers. We concentrate on the S-box step because the output of this non-linear transformation is where the difference between the correct key hypothesis and the wrong ones is highest, and thus it is the preferred attack point for an adversary [7]. Figure 1 depicts the basic configuration used in our experimental power analysis attacks. This configuration is a commonly used simplified implementation of one round of the AES cipher.

The plaintext is added (modulo 2) to the secret key and the result of this *xor* operation is used as input to the S-box. The output of the substitution step is stored into a register. In order to always have the same initial condition, a reset signal is applied to the register at the end of each write operation. Although a real implementation of the full algorithm would be somewhat different from this simplified diagram, our purpose is only to estimate the impact of error detection circuits (concatenated to the S-box) on the resistance to power analysis attacks. This approximation (shown in Figure 1) is accepted as sufficiently accurate for analyzing attacks on the most vulnerable portion of the cipher and is therefore, adequate for our needs.

Figure 2 shows, as an example, an S-box with a parity bit. In this figure, the added check bits are used to detect the presence of errors in two different instances: once at the input and then at the output of the S-box. When new data enters the S-box,

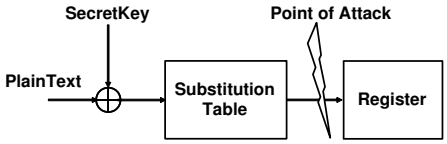

Fig. 1. Overview of the considered part of the AES Algorithm.

the check bits are separated from the data bits and an error detection is performed. If no error is detected, the data bits enter the S-box circuit. The S-box then produces the result of the non-linear transformation plus the corresponding check bits. At this point the second check is performed, again as described before. If no error is detected in both checks, the output of the S-box is forwarded to the next round transformation; otherwise, a faulty output composed of all zeros except the right most bit is generated to signal the error.

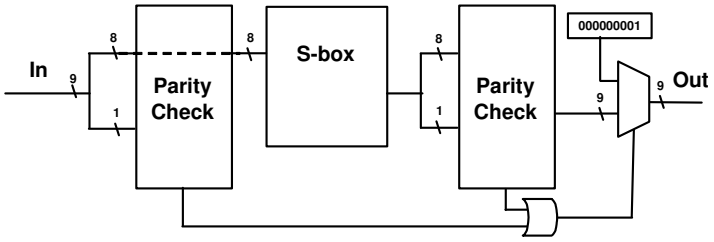

Fig. 2. Block diagram of the parity error detection scheme applied to an 8-bit S-box.

We have implemented several versions of the non-linear function in the AES S-box, each with a different error detection or correction code. The following circuits are considered:

- **Reference version**. This circuit implements the non-linear transformation as described in the standard and is used as the reference version.
- **Single parity-based error code**. The single byte parity circuit implements the error detection scheme described by Bertoni et al. [1].
- **Double parity-based error codes**. This code computes two parities: one for the bits with even indices and one for the bits with odd indices.
- **Residue-based codes**. These are the residue codes that use the moduli 3 and 7.
- **Error codes based on complementary parity**. In this scheme, both the even and the odd byte parity bits are computed.
- **Hamming error correcting code**. We consider a (12,4) Hamming code described by the following parity matrix:

$$H = \begin{bmatrix} 1&1&1&0&0&0&1&1&1&0&0&0 \\ 1&0&0&1&1&0&1&1&0&1&0&0 \\ 0&1&0&1&0&1&1&0&0&0&1&0 \\ 0&0&1&0&1&1&0&1&0&0&0&1 \end{bmatrix}$$

4 Experimental Setup

Figure 3 depicts the complete setup we have used for our evaluation procedure, which is similar to the one presented by Regazzoni et al. in [9]. It is composed of a standard Electronic Design Automation (EDA) flow and includes a simulation environment for generating the power consumption traces which are used to provide a measure of the resistance against power analysis attacks. The input to the process is the Register Transfer Level (RTL) description of the S-box and one of the considered error detection/correction circuits. The output is a text file which stores the noise-free instantaneous current consumption of the circuit simulated at a very high resolution of both time and current.

In all our circuit implementations, the S-box module has been described using VHDL at the behavioral level. Because of this, it has been synthesized by the tool as a combinatorial circuit rather then a memory-based look-up table. It is therefore, not necessary to protect the address decoder against injected faults since this component is not present in the synthesized implementation of the substitution function. This approach does not constitute a limitation since it reflects a typical situation when designing a cryptographic unit, where the entire unit is specified using a hardware description language and then synthesized by an EDA tool with no specific implementation constraints imposed. In such cases, the S-box module is often realized as a combinatorial logic.

The VHDL description is synthesized using the ST-Microelectronics 90nm CMOS standard cell library [13] and *Synopsys Design Compiler* [14]. If the synthesis tool is set to minimize the circuit's area, it is possible that during the optimization phase of the synthesis process the redundant parts of the circuits (e.g., the circuit generating the complementary parity bit) will be removed. In order to prevent this from happening, we first synthesized each component of the circuit separately and then connected the individual components together forcing the tool to not further optimize the internal design of the individual components. The number of equivalent two-input NAND gates of each circuit is reported in Table 1.

The resulting circuit is then placed and routed using Cadence Design Systems *SoC Encounter* [2]. A parasitics file (in spef format) is produced, along with the Verilog netlist of the circuit and an sdf file for back annotation of the delays. The flow produces the spef and sdf files and the Verilog netlists of the entire design.

Post-place-and-route simulation is then performed using *ModelSim,* with the previously generated sdf files to verify the functionality of the circuit and to generate test vectors for transistor-level simulation that will be used to produce the simulated power traces. *Synposys Nanosim* is then executed to perform transistor-level simulation, using the spef file, the relative Verilog netlist, the SPICE models of the

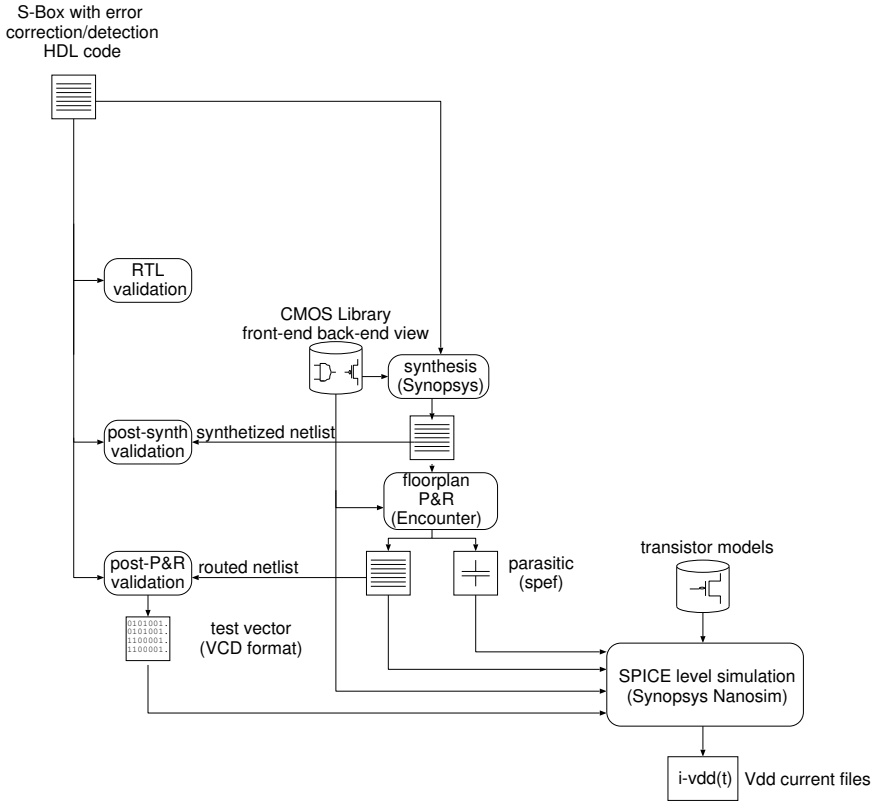

Fig. 3. The experimental setup.

S-box type	Circuit Area (GE)
Reference	568
Parity	698
Complemented Parity	794
Double Parity	838
Residue Modulo 3	872
Residue Modulo 7	1013
Hamming Code	847

Table 1. Post Synthesis area of each error detection/correction circuit in Gate Equivalent (GE) area units.

technology cells and the transistor models. This simulation generates vector-based time-varying power profiles which are stored in a text format. This data corresponds to the simulated traces which are later used for mounting the power analysis attacks and the security evaluation.

5 Evaluation of the Effects on Power Analysis Resistance

We analyze in this section the possible impact on power analysis resistance of a fault detection circuitry added to a device to protect it against fault injection attacks, using state-of-the-art tools and methodologies.

We present below the results of several experiments focusing on the added structural redundancy, attacking the previously described configuration (depicted in Figure 1), where the result of the key addition is passed through the S-box.

In the first and second sets of experiments we explore how the different error detection/correction codes affect the two most common power analysis schemes, namely, the Differential Power Analysis (DPA) based on Kocher's Difference-of-Means and the DPA based on Pearson's correlation coefficients which uses the Hamming weight as model. These experiments were carried on using the noise-free traces generated by the SPICE level simulator.

In the first set of experiments we performed DPA attacks (based on the Kocher' Difference-of-Means) targeting all the output bits of the S-boxes. These experiments included attacks on the reference circuit (a straightforward implementation of the AES standard) and on all the error detection codes described in Section 3. In the second set of experiments we performed attacks which use as model the Hamming weight and as distinguisher Pearson's correlation coefficients, and we considered both situations where the adversary is aware or is unaware of the presence of the particular error detection code used.

The purpose of the third set of experiments has been to get a fair and objective comparison between the different error detection and correction circuits. To this end, we used a metric based on information theory to provide an attack-independent evaluation of each error detection/correction circuit. Based on this, we report how the number of information bits that are available to the attacker changes as a function of the measurement noise.

Finally, in the fourth set of experiments we analyze for each of the error detection/correction circuits, how the success rate of an adversary (exploiting the correlation coefficients and the Hamming weight) varies as a function of the model used during the attack. These experiments show whether the additional information available to the attacker can be beneficial even if the attacker is unaware of the presence of the specific error detection/correction code.

5.1 Evaluating the effects of the added check bits on Kocher's Difference-of-Means DPA

The goal of these experiments is twofold: determine whether the redundancy added to a cryptographic circuit affects the effectiveness of the classical Kocher's DPA, and find out whether the redundancy bits themselves are susceptible to this attack as are the data bits. We also want to explore whether the choice of a particular error detection code influences the results. We thus performed Kocher's DPA attack against various implementations of the AES S-box, with or without error detection.

For the reference circuit, we mounted Kocher's DPA attacks targeting, one by one, all the output bits of the S-box. In the attacks on implementations that include a fault detection circuit, we distinguished between two cases: in the first case we targeted only the 8 output bits of the S-box, mimicking the situation in which the attacker is unaware of the presence of the error detection circuit. In the second case, we included in our hypothesis the redundancy bits, assuming that the attacker knows about the specific error detection check bits that have been added to the S-box. All these attacks were performed directly on the noise-free power traces produced by the SPICE level simulator described in Section 4

Figures 4 and 5 show the results of a Kocher's DPA attack on one output bit of the AES S-box in the reference circuit and on the same bit of the S-box with an added error detection circuit based on complementary parity bits, respectively. The differential trace corresponding to the correct key is plotted in black, while all the others are in gray. As can be seen, during the computation of the output that corresponds to the initial part of the graph (approximately up to 1000 ps), the presence of the parity bits seems to make the attack more difficult. However, when the result is stored into the register (at about 1750 ps) the peak is, in both cases, of approximately the same height ($2.5 \cdot 10^{-4}$). The above situation repeats itself for all the other codes. Since the adversary typically targets the point that yields the highest probability to succeed with the DPA attack, and since this point is usually the time when the computed value is stored into the register, we can conclude that the presence of an error detection circuit does not substantially affect Kocher's DPA.

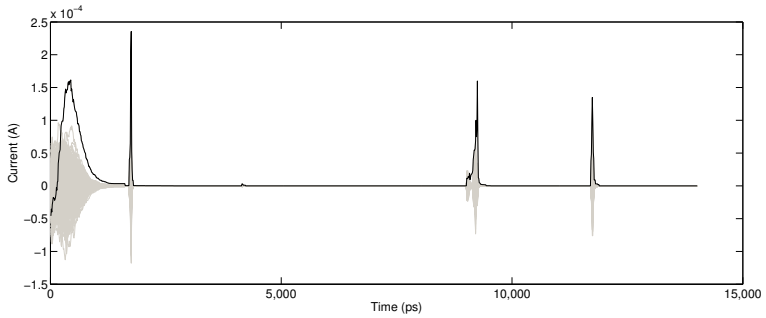

Fig. 4. Kocher's DPA attack on the reference implementation of the AES S-box.

The situation is slightly different when the adversary attacks one of the redundant check bits. When the target bit (used as selection function) generates two sets that contain approximately the same number of elements (as in the case of the parity bit, the complementary parity, the Hamming code and the dual parity), the value of the peak for the attacks on a check bit has approximately the same height as that for an attack on an information bit. This shows that an attack on the parity bit is just as efficient as an attack on any other bit. In contrast, when the selection bit does not

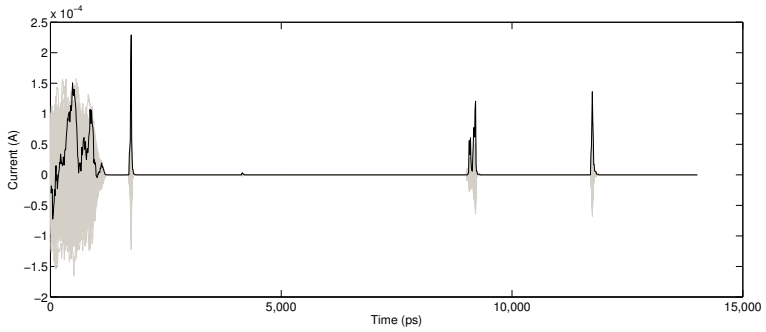

Fig. 5. Kocher's DPA attack on the implementation of the AES S-box with added complementary parity.

split the set evenly (as in the case of residue modulo 3 or 7), the strength of Kocher's DPA attack is significantly reduced and it may even fail sometimes.

In summary, the experiments described above show that Kocher's DPA attacks based on the difference of means and performed on noise-free traces are not substantially affected by the presence of an error detection circuit. Additionally, very often the check bits can also be exploited by the attacker. Consequently, as a general rule, when adding countermeasures against power analysis to an implementation that includes an error detection circuit, the check bits should be protected as well.

5.2 Evaluating the effects of check bits on Pearson's correlation-based DPA

The second set of experiments focused on Pearson's correlation coefficients DPA that uses Hamming weight with the goal of finding out whether one of the circuits is easier to attack than the others. Another objective is exploring whether the knowledge about the presence of an error detection circuit can be exploited by the attacker. We performed a series of correlation-based DPA attacks against all the considered implementations of the AES S-box. It is important to notice that since we reset the circuit after each computation, the Hamming weight and the Hamming distance models would yield exactly the same results.

In the case of AES, the attacker usually hypothesizes all the 8 output bits of the S-box. However, when the S-box is extended to provide error detection or correction, the number of output bits is higher than 8. Since the correlation value changes depending on the number of bits included in the hypothesis, it is worthwhile to explore the effects of the added check bits in two cases: when the adversary hypothesizes only the 8 output bits of the AES S-box (i.e., the attacker is unaware of the presence of an error detection circuit) and when the adversary hypothesizes all the output bits including the check bits (i.e., the attacker is aware of the particular error detection code used).

As in the previous series of experiments, we performed our evaluation on the noise-free traces produced by the SPICE level simulator and we considered the cir-

cuit depicted in Figure 1. Furthermore, as in the case of Kocher's DPA, since the traces obtained by simulation are noise-free and the size of the S-box is 8 bits, we can fully characterize the device by simulating all the 256 input plaintexts for each of the 256 possible keys.

The first series of attacks was performed including in the attack hypothesis all the bits of the target register. This corresponds to the situation where the adversary is aware of the particular error code used. Then, we included in the attack hypothesis only the 8 output bits of the S-box, i.e., we assumed that the adversary is unaware of the presence of an error detection circuit.

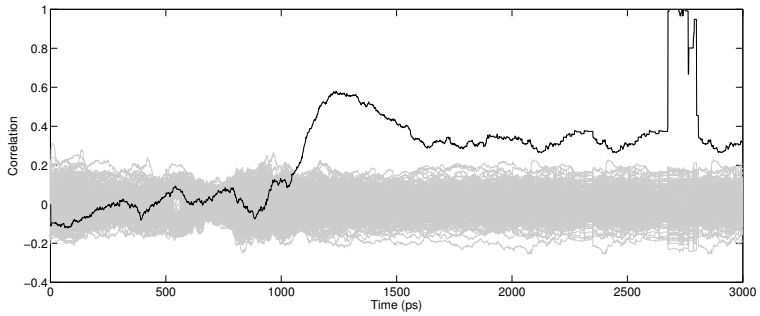

Fig. 6. Pearson's correlation coefficients DPA attack on the reference implementation of the AES S-box.

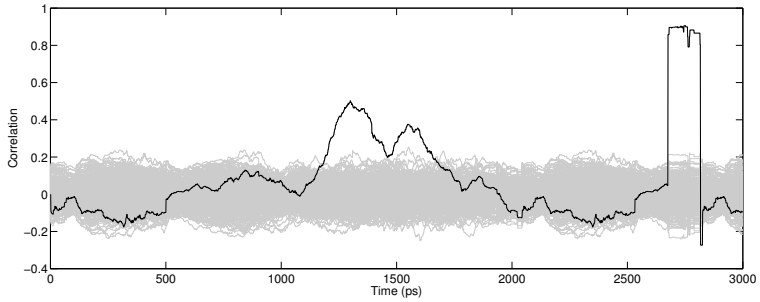

Fig. 7. Pearson's correlation coefficients DPA attack on the AES S-box with added double parity with the attacker being unaware of the error detection circuit.

Figures 6 and 7 show the results of a Pearson's correlation coefficients DPA attack on the output of the AES S-box in the reference circuit and on the output of the S-box with an added error detection circuit based on double parity, respectively, when the presence of the error detection code is unknown to the attacker. The figures show the time period during which the outputs of the S-box and, when present, the check bits, are computed (approximately up to 2500 ps), and the time interval in

which the results and the check bits are stored into the register (approximately from 2500 ps to 3000 ps).

As can be seen, this is the best situation for the attacker and this is confirmed by the high value of the correlation coefficients for the time intervals which correspond to the register write operations. In fact, in the attack mounted when the presence of the error correction code was known to the attacker, all the circuits showed a correlation value approximately equal to 1. Additionally, when the presence of the error detection code was unknown to the attacker, for all the considered circuits the correlation coefficients were slightly decreased, but still very high (never below 0.81), thus sufficient to easily extract the secret key.

We can also see that during the computation, from the beginning of the trace to approximately 2500 ps, the difference between the correlation coefficients of the wrong key guesses and those of the correct key guess decreases, mainly because of the reduction in the correlation coefficient of the correct key guess. However, this does not represent a significant problem for the adversary since the easiest attack point is still the register store operation, and there the correlation, as shown before, is not significantly affected by the presence of the error detection/correction circuit.

These results show that when the adversary has noise-free traces or, alternatively, a sufficient number of traces to completely filter out the noise, the correlation-based DPA which uses the Hamming weight model is always successful, independently of the particular error detection/correction code used and of the attacker's knowledge about the specific circuit implemented.

5.3 Evaluating the effects of the check bits using information theory

As previously shown, when the power consumption traces are noise-free or include very little noise, the presence of any one of the considered error detection/correction circuits does not significantly impact the resistance against the two most common power analysis attacks.

In order to obtain a more fair comparison among the different error detection/correction schemes and to quantify the effect that the redundancy may have on the resistance to power analysis attack, it is necessary to analyze the behavior of each circuit in the presence of noise, independently of the particular attack hypothesis and scenario.

To this end, we use the metric based on information theory proposed by Standaert et al. [12], which was developed to allow the evaluation of side-channel information leakage.

Intuitively, this information theoretic metric measures the resistance against the strongest possible type of side-channel attack and allows an evaluation that is independent of a particular attack scenario. Practically, the metric measures how much uncertainty about the secret key remains after the attacker took advantage of the given information leakage.

More formally, let K be a random variable representing the key that the adversary wants to recover in the side-channel attack. Let X be a random variable representing the known plaintext entering the target operations (in our case the S-box with

or without an error detection/correction circuit) and let L be a random variable representing the power consumption traces generated by a computation with input X and key K. L is obtained by adding a certain amount of normally distributed random noise R to the power trace T produced by a SPICE level simulation, i.e., $L = T + R$. The conditional entropy H between the key K and its corresponding leakage L is defined as follows:

$$\mathrm{H}[K|L] = -\sum_k \Pr[k] \cdot \sum_x \Pr[x] \int \Pr[l|k,x] \cdot \log_2 \Pr[k|l,x] \; dl$$

where $\Pr[k]$ is the probability of the key k, $\Pr[x]$ is the probability of the plaintext x, $\Pr[l|k,x]$ is the probability of the leakage l given the key-plaintext pair (k,x), and $\Pr[k|l,x]$ is the probability of the key k given the leakage-plaintext pair (l,x). The probability density function of L is assumed to be approximated by the normal distribution $\mathcal{N}(\mu_{k,x}, \sigma^2)$, where $\mu_{k,x}$ is the noise-free leakage measured during the computation of the (k,x) pair and σ is the standard deviation of the noise. As a result, the conditional entropy of K given L, can be rewritten as:

$$H[K|L] = -\sum_k Pr[k] \cdot \sum_x Pr[x] \cdot \int_{-\infty}^{\infty} \mathcal{N}_l(\mu_{k,x}, \sigma^2) \cdot log_2 \frac{\mathcal{N}_l(\mu_{k,x}, \sigma^2)}{\sum_{k^*} \mathcal{N}_l(\mu_{k^*,x}, \sigma^2)} dl.$$

There are different factors that influence this conditional entropy. The first one is obviously the simulated power trace. The second one is the standard deviation of the noise in the leakage. The size of the power traces is also important: simulated traces are typically composed of several thousands of samples. Hence, directly applying multivariate statistics on these large dimensionality variables is hardly tractable.

In order to reduce the dimensionality of the power traces, some compression techniques such as the Principal Component Analysis (PCA) [4] and integration over the full trace, were proposed in the past. For the experiment carried out in this work, we used the latter, namely we first integrated the noise-free trace to reduce its dimensionality and then we evaluated the entropy. Therefore, the mutual information is extracted from a trace that was first compressed to one single sample.

Given the conditional entropy one can calculate the so called mutual information [12] (which intuitively quantifies what the adversary knows about the secret key K assuming that he has the knowledge of the leakage L) as follows:

$$\mathrm{I}[K,L] = \mathrm{H}[K] - \mathrm{H}[K|L]$$

where $\mathrm{H}[K]$ is the entropy. Since all key values are equi-probable, $\mathrm{H}[K]$ is equal to n which is the number of bits of the key K.

Figure 8 depicts the value of the mutual information for each considered error detection/correction circuit as a function of the standard deviation of the noise. Intuitively, the higher the number of bits available to the attacker, the lower the resistance against the power analysis. In the left part of the graph, where the noise level is under a certain threshold, all the circuits have an information leakage that is as high as 8. This means that the attack is not affected by the presence of the particular circuit

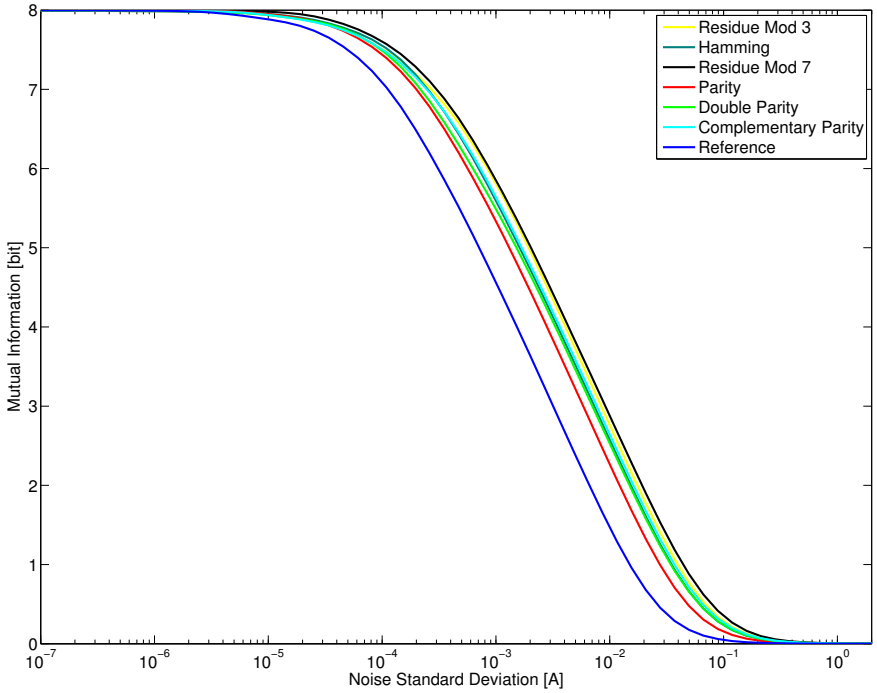

Fig. 8. Mutual information as a function of the noise's standard deviation for each of the considered error detection/correction codes.

used, since it will be successful in any case. This confirms the results presented in Sections 5.1 and 5.2, which have shown that when the noise is completely absent, the effectiveness of both Kocher's and Pearson's DPAs was not affected by the presence of error detection/correction circuits. Figure 8 also depicts the dual situation, which corresponds to the case when the noise level is higher than a certain threshold: here too, the attack is not affected by the particular circuit used. However, in this case the mutual information is always 0, and the adversary will not be able to retrieve the secret key in any case.

When the standard deviation of the noise is in the middle interval, it is possible to quantify the different effect that each of the error detection circuits may have on the strongest possible power analysis attack. The Reference S-box (the one without any error detection code), is characterized by the smallest number of bits leaked, followed by the parity scheme. The two worst circuits are the ones implementing the residue codes modulo 3 or 7. The graph thus confirms the intuition that, except for the case of the Hamming error correction code, there is a direct relation between the number of check bits and the amount of information leakage. It is important to emphasize

that the ranking obtained using the information theory metric shows the number of bits available to the strongest possible attacker: it is possible that in a specific attack scenario in which the adversary is not able to exploit all the information present in the power trace, the ranking of the circuits will be different. Still, this metric is the most objective one since it does not depend on the specific attack hypothesis. In the next section we show how the information stored in the power traces may be exploited by an attacker who uses the correlation-based DPA and the Hamming weight model.

5.4 Evaluating the effect of check bits on the Success rate

The goal of the last set of experiments was to evaluate how the success rate of an attacker who uses the correlation-based DPA varies for each different circuit as a function of the power model used. We also discuss in this section whether one of the considered circuits is easier to attack than the others, when the same attack is used.

Intuitively, the better the power model is, the easier it is for an adversary to recover a key. One important result which can be obtained from these experiments is whether the additional information leakage generated by the error detection/correction circuit can be exploited by the attacker even if he/she is unaware of its existence.

To this end, we performed a set of correlation-based DPA attacks against all the considered implementations of the AES S-box, using different attack hypotheses and increasing at each run the number of traces. The attacks were performed using all the 256 possible keys and randomly selecting the input plaintext. For the first set of experiments, the correlation between the Hamming weight of the 8-bit S-box output and the power traces was calculated for all the considered circuits. The underlying assumption was that the adversary is unaware of the presence of the error detection/correction circuit and can therefore construct only an approximated power model. In the second set of experiments we computed the correlation between the power traces and the Hamming weight of the full output of the S-box (including the check bits), thus assuming that the attacker is aware of the particular redundancy added to the S-box circuit.

Using the results of all the attacks that were mounted, we can compute the first order success rate [12], defined below. Given an adversary who attacks a secret key K and generates n key guesses $g = [g_1, g_2, \ldots, g_n]$ that are sorted according to the attack result (correlation-based DPA in our case), we define a function f that returns 1 if the correct key is g_1 and 0 otherwise. The first order success rate of the attack against the secret key K is defined as:

$$\mathrm{Succ}^{K}_{\mathrm{attack}} = \Pr[f = 1]$$

We concentrate on one of the intermediate situations discussed in Section 5.3, where the noise is present but is not sufficiently high to completely overshadow the signal and cause the attack to fail. To simulate the noise conditions, we added white noise to the traces generated by the transistor level simulator.

Figures 9 and 10 show the success rate of the correlation-based DPA as a function of the number of traces for a fixed value of the noise standard deviation (equal to

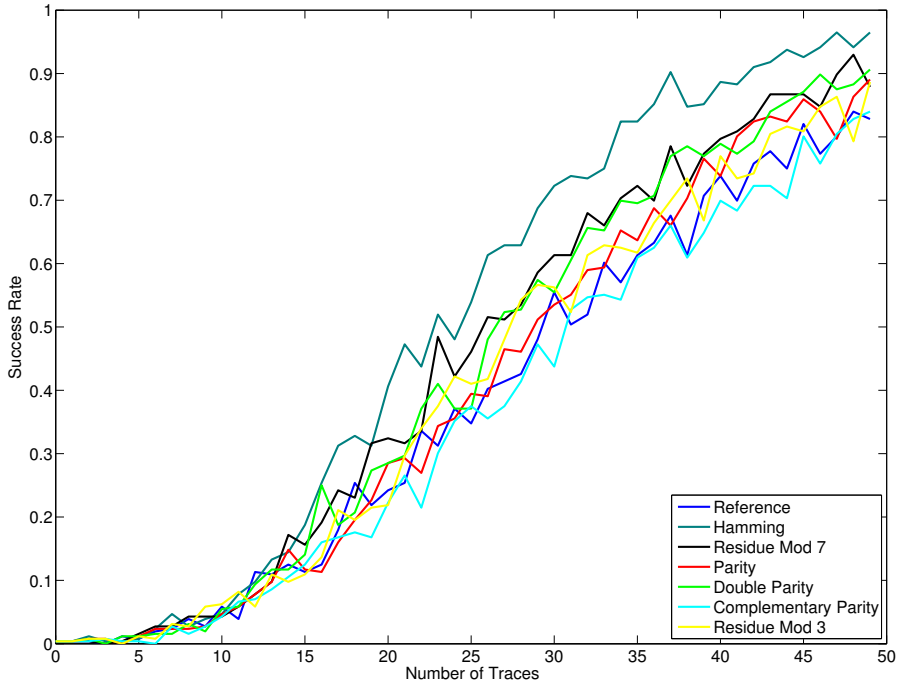

Fig. 9. Success rate of the correlation-based DPA attack vs the number of traces using attack hypothesis of the full size of the output register (the adversary is aware of the error detection/correction code used).

$5 * 10^{-}4$) using two different attack hypotheses. Figure 9 shows the success rate of the correlation-based DPA when the attacker is aware of the presence of the code, thus targeting all the bits of the output register. Figure 10 shows the success rate of the DPA when the attack hypothesis is based on the Hamming weight of only the 8 output bits of the S-box, i.e., the attacker is unaware of the code used. In both figures, the faster the curve approached 1, the easier it is for the attacker to recover the secret key.

As can be seen from Figure 9, the error detection/correction circuit that yields the worst resistance against the correlation-based DPA attacks is the one that uses the Hamming correcting code. The only code that seems to be slightly more resistant than the reference S-Box is the one based on complementary parity. We can therefore state that when the adversary mounting a correlation-based DPA attack is aware of the particular error correction code used, the added redundancy significantly helps the adversary. Note that the ranking of the circuits in Figure 9 is in agreement with that reported in Section 5.3. The information theory metric was computed using all

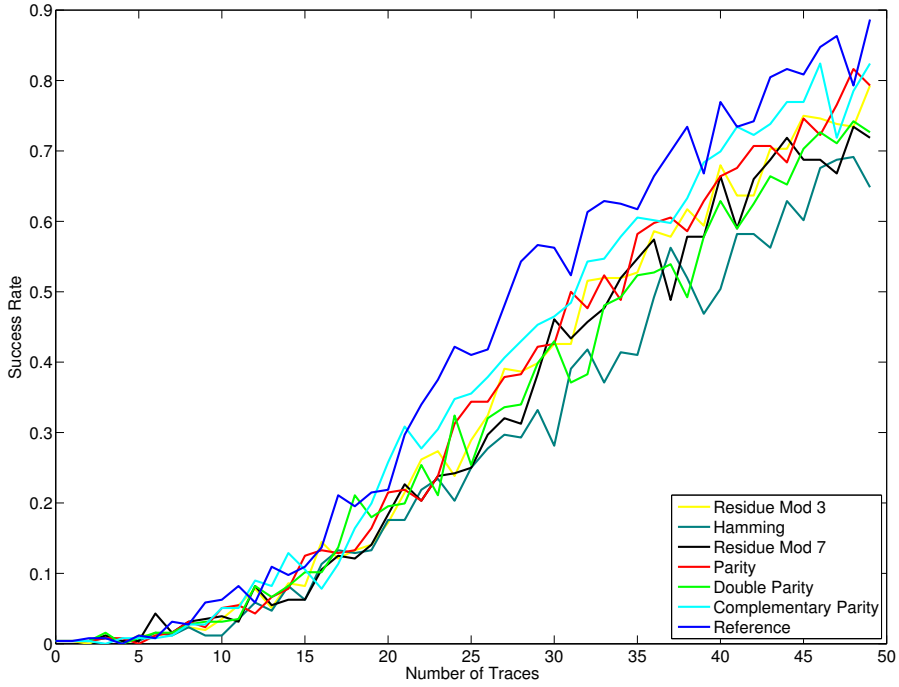

Fig. 10. Success rate of the correlation-based DPA attack vs the number of traces using attack hypothesis of 8 bits only (the adversary is unaware of the error detection/correction code used).

the points of the trace and assumes a strong adversary, thus it incorporates also information which can not be exploited by the attack considered in this case.

In contrast, as shown in Figure 10, when the presence of the particular error detection/correction circuit is not known, the success rate of the reference S-box is higher than all the others. This however, does not mean that the presence of the check bits can never help the attacker when he is unaware of them. The implementations that include check bits still generate a large amount of information leakage as indicated by the information theory metric (and shown in Section 5.3), and it is possible that different technological libraries or more sophisticated attacks can show a significant improvement even when the presence of the code is unknown.

6 Conclusions

We have presented in this chapter an evaluation of the effect that an error detection circuit may have on the resistance to power analysis attacks on hardware imple-

mentations of cryptographic S-boxes. The evaluation was carried out using the ST-Microelectronics 90nm CMOS technology library and specific synthesis and place and route options to prevent the tool from removing the redundancy that is present due to the added error checking circuitry.

Our results show that the presence of the error detection/correction circuit increases the amount of information available to the attacker. We also show that, depending on the particular attack hypothesis, the adversary may take advantage of this additional information.

It is important however to mention that our conclusions regarding the impact of the different error detection and correction codes on the vulnerability to power analysis attacks may be different for other design environments. As is the typical case with hardware designs, even the same high-level description of a module may lead to a quite different VLSI circuit if the design options and technology library are changed.

Nevertheless, when incorporating fault detection or correction circuits into hardware implementations of cryptographic algorithms, it is of crucial importance to be aware of the possible effects that the added redundancy may have on the robustness against power analysis attacks.

The experiments presented in this chapter provide an example of how to perform an evaluation of the vulnerability of the designed circuit to power analysis attacks, prior to manufacturing.

References

1. Bertoni, G., Breveglieri, L., Koren, I., Maistri, P., Piuri, V.: Error analysis and detection procedures for a hardware implementation of the advanced encryption standard. IEEE Transactions on Computers **52**(4), 492–505 (2003)
2. Cadence Corporation: SoC encounter. `http://www.cadence.com/products/di/soc_encounter/pages/default.aspx`
3. Daemen, J., Rijmen, V.: AES proposal: Rijndael (1999)
4. Macé, F., Standaert, F.X., Quisquater, J.J.: Information theoretic evaluation of side-channel resistant logic styles. In: Paillier, P., Verbauwhede, I. (eds.) Cryptographic Hardware and Embedded Systems – CHES 2007, *Lecture Notes in Computer Science*, vol. 4727, pp. 427–442. Springer (2007)
5. Maingot, V., Leveugle, R.: Error detection code efficiency for secure chips. In: 13th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2006), pp. 561–564. IEEE Press (2006)
6. Maingot, V., Leveugle, R.: On the use of error correcting and detecting codes in secured circuits. In: 2007 Ph.D. Research in Microelectronics and Electronics Conference (PRIME 2007), pp. 245–248. IEEE Press (2007)
7. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer (2007)
8. National Institute of Standards and Technology (NIST): Advanced encryption standard (AES). Federal Information Processing Standards Publication 197, Gaithersburg, MD, USA (2001)

9. Regazzoni, F., Cevrero, A., Standaert, F.X., Badel, S., Kluter, T., Brisk, P., Leblebici, Y., Ienne, P.: A design flow and evaluation framework for DPA-resistant instruction set extensions. In: Clavier, C., Gaj, K. (eds.) Cryptographic Hardware and Embedded Systems – CHES 2009, *Lecture Notes in Computer Science*, vol. 5747, pp. 205–219. Springer (2009)
10. Regazzoni, F., Eisenbarth, T., Breveglieri, L., Ienne, P., Koren, I.: Can knowledge regarding the presence of countermeasures against fault attacks simplify power attacks on cryptographic devices? In: Bolchini, C., Kim, Y.B., Gizopoulos, D., Tehranipoor, M. (eds.) 23rd IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT 2008), pp. 202–210. IEEE Computer Society (2008)
11. Regazzoni, F., Eisenbarth, T., Großschädl, J., Breveglieri, L., Ienne, P., Koren, I., Paar, C.: Power attacks resistance of cryptographic S-boxes with added error detection procedures. In: Bolchini, C., Kim, Y.B., Salsano, A., Touba, N.A. (eds.) 22nd IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT 2007), pp. 508–516. IEEE Computer Society (2007)
12. Standaert, F.X., Malkin, T., Yung, M.: A unified framework for the analysis of side-channel key recovery attacks. In: Joux, A. (ed.) Advances in Cryptology – EUROCRYPT 2009, *Lecture Notes in Computer Science*, vol. 5479, pp. 443–461. Springer (2009)
13. STMicroelectronics: Clock 90 GPLVT 1.2V 2.2 Standard Cell Library User Manual and Databook (2006)
14. Synopsis Corporation: Design compiler ultra. `http://www.synopsys.com/Tools/Implementation/RTLSynthesis/Pages/DCUltra.aspx`