IACR Transactions on Cryptographic Hardware and Embedded Systems
ISSN 2569-2925, Vol. 2021, No. 3, pp. 780–805. DOI:10.46586/tches.v2021.i3.780-805

Countermeasures against Static Power Attacks

– Comparing Exhaustive Logic Balancing and Other Protection Schemes in 28 nm CMOS –

Thorben Moos and Amir Moradi

Ruhr University Bochum, Horst Görtz Institute for IT Security, Bochum, Germany
firstname.lastname@rub.de

Abstract. In recent years it has been demonstrated convincingly that the standby power of a CMOS chip reveals information about the internally stored and processed data. Thus, for adversaries who seek to extract secrets from cryptographic devices via side-channel analysis, the static power has become an attractive quantity to obtain. Most works have focused on the destructive side of this subject by demonstrating attacks. In this work, we examine potential solutions to protect circuits from silently leaking sensitive information during idle times. We focus on countermeasures that can be implemented using any common digital standard cell library and do not consider solutions that require full-custom or analog design flow. In particular, we evaluate and compare a set of five distinct standard-cell-based hiding countermeasures, including both, randomization and equalization techniques. We then combine the hiding countermeasures with state-of-the-art hardware masking in order to amplify the noise level and achieve a high resistance against attacks. An important part of our contribution is the proposal and evaluation of the first ever standard-cell-based balancing scheme which achieves perfect data-independence on paper, i.e., in absence of intra-die process variations and aging effects. We call our new countermeasure Exhaustive Logic Balancing (ELB). While this scheme, applied to a threshold implementation, provides the highest level of resistance in our experiments, it may not be the most cost effective option due to the significant resource overhead associated. All evaluated countermeasures and combinations thereof are applied to a serialized hardware implementation of the PRESENT block cipher and realized as cryptographic co-processors on a 28 nm CMOS ASIC prototype. Our experimental results are obtained through real-silicon measurements of a fabricated die of the ASIC in a temperature-controlled environment using a source measure unit (SMU). We believe that our elaborate comparison serves as a useful guideline for hardware designers to find a proper tradeoff between security and cost for almost any application.

Keywords: Static Power · Side-Channel · SPSCA · Countermeasures · Shuffling · SDRL · QuadSeal · Exhaustive Logic Balancing · Threshold Implementation

1 Introduction

Complementary Metal-Oxide-Semiconductor (CMOS) technology is the predominant standard for integrated circuit (IC) fabrication since about 40 years now. One of the main reasons for its continued dominance is the conceptually guaranteed low idle power dissipation. In contrast to other logic families, CMOS gates do not dissipate any energy in stable states unless leakage currents occur. Leakage currents are defined as the undesired transfer of electrical energy across a boundary which is technically viewed as insulating. The main example relevant in this context is the flow of current across a transistor which is in the *off* state. While these leakage currents have been negligibly small in former generations of CMOS technology, the aggressive down-scaling of the physical feature size

Received: 2020-10-15 Accepted: 2020-12-15 Published: 2021-07-09

Table 1: Estimated leakage current of a 2-input `NOR` gate in a 22 nm CMOS technology for different input values [AO14].

A1	A2	Leakage Current [nA]
0	0	172.16
0	1	173.44
1	0	62.96
1	1	38.42

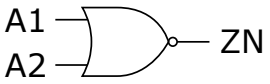
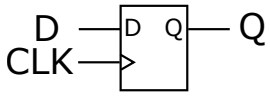


Table 2: Estimated leakage current of a `D-flip-flop` in a 22 nm CMOS technology for different input and output values [AO14].

D	CLK	Q	Leakage Current [nA]
0	0	1	421.79
0	1	0	446.39
0	1	1	370.71
1	0	0	376.11
1	0	1	441.54
1	1	0	437.42
1	1	1	386.61



throughout the past decades has led to a significant increase of their magnitude and therefore to a rise of the overall static power consumption of CMOS-based devices.

Nowadays, leakage currents are a crucial quantity to observe during the IC design flow. Modern Electronic Design Automation (EDA) tools devote a high effort towards reducing the overall leakage current of a circuit in order to keep devices suitable for battery-powered applications. Hereby, the leakage current conducted through a single standard cell in a stable state is modeled and characterized in a fairly simple manner, namely as a function of the logical signals applied. Indeed, the input values currently applied to a logic gate play a significant role in determining the leakage current conducted through the cell. The total magnitude of the current leaked by a circuit is then estimated as the sum of the individual leakage currents of all gates in the circuit. Design libraries which are used to estimate the timing, noise and power consumption during the synthesis and implementation stages of the IC design flow are typically characterized for one fixed set of temperature, supply voltage and process corner. All three of those parameters affect the leakage currents conducted by the cells globally. Yet, for one fixed set of conditions, the one local factor considered in the estimation of the current leaked is indeed the vector of logical input values applied to a logic cell. For each possible combination of inputs, one characterized magnitude of the current leaked is given by the design libraries. For memory cells like flip-flops the logical output value(s) and the clock input are also considered in the idle power estimation. For clarification, see the two exemplary leakage tables, one for a 2-input `NOR` gate in Table 1 and the other for a `D-type flip-flop` in Table 2, estimated for a 22 nm bulk CMOS process by the authors of [AO14]. It is obvious that any data value which is stored or currently processed by a standard-cell-based circuit has a direct impact on the total leakage current conducted. Therefore, it is no surprise that this quantity can be exploited via statistical analysis to learn details about the secret internals of cryptographic chips.

1.1 Related Work

The information leakage through the static power consumption of CMOS-based circuits has been identified in 2007 for the first time as a potential security threat for cryptographic hardware [GSST07]. It took until CHES 2014 before the first experimental analysis on the subject was made available in public literature [Mor14]. This work analyzed the data dependency of the leakage currents of different elements in the programmable fabric of modern FPGA devices manufactured in different nanometer-scaled technology generations. For the first time, the feasibility of such attacks was demonstrated in practice and the first implementations equipped with side-channel countermeasures had been evaluated against this new kind of analysis. In the following years, several practical case studies have been reported targeting both, programmable hardware [BCS+17] and dedicated ASIC chips [PSKM15, MMR17, KMM19, Moo19, MMR20, Moo20]. The general procedure of performing an attack based on the idle power consumption remained largely unchanged from the beginning. During the execution of the first (or last) round of a block cipher, the adversary halts the global clock signal of the device and therefore artificially creates an idle state that allows to measure the current flowing through the device without any ongoing computations for an extended period of time. Thus, the ability to halt or pause the clock signal of the device under test (DUT) is typically viewed as a requirement for this type of adversary. At CHES 2019 it was pointed out that sensitive information is often left behind by cryptographic co-processors after their operation, which allows the extraction of secret data even without any clock control abilities [Moo19]. The measurement setups used in previous practical case studies have mostly utilized an oscilloscope as the central measurement instrument for data acquisition [PSKM15, Mor14, MMR17, KMM19, Moo19, MMR20, Moo20], sometimes together with a differential probe with internal amplification [Mor14], sometimes together with custom DC amplifiers and low pass filters [MMR17, KMM19, Moo19, MMR20, Moo20]. To the best of our knowledge, the only work that used a commercial instrument dedicated for high-precision low-current measurements, namely a picoammeter, has been presented in [BCS+17]. Due to sensitive dependencies of the leakage currents on the supply voltage and the temperature, static power measurements typically require a little more care, setup-wise, to obtain the leakages in sufficient quality. In that regard, the experiments are often performed in temperature-controlled environments such as climate chambers [MMR17, KMM19, Moo19, MMR20, Moo20]. However, it was quickly discovered that the strong dependencies on environmental factors can be used in favor of the adversary to escalate the leakage of information [Moo19, MMR20]. While it appears to require a little more effort to build a setup and perform such experiments in practice (compared to dynamic power experiments), the implications can be significant once an adversary succeeds. In particular, even implementations that are typically less susceptible to passive SCA attacks or come with dedicated side-channel protections in place may be vulnerable to this kind of attack due to its different leakage mechanisms. This has been demonstrated especially with respect to dedicated logic styles [DGS+11, ABD+14], masking schemes [Mor14, MMR17, Moo19] and recently also unrolled implementations [Moo20].

With respect to dedicated countermeasures against static power attacks, the first obvious solution that comes to mind could be to build devices in such a manner that it is infeasible for an adversary to influence (esp. reduce the frequency or entirely halt) the clock signal of the circuit under analysis. Then, if the designer has also taken care that no sensitive intermediate values remain in the circuit while not currently computed upon, performing such attacks becomes virtually impossible. However, protecting the clock signal against exterior influences is easier said than done. Adversaries may employ invasive methods to stop the operation of a circuit part for some time and measure the current flowing. Or, even more importantly, depending on the functionality of a device it may be required to hold sensitive data in the circuit for an extended period of time without actively computing

on it. Thus, protecting the clock signal against adversarial access is often not sufficient. From a security designer's point of view it is generally undesirable for a circuit to silently leak information about the stored data, even in the absence of computation. Therefore, it is often preferable to apply dedicated protections against this kind of attack to the sensitive circuit parts. Different kinds of countermeasures have been proposed for this purpose over the years [NYH13, HMY13, ZZL13, ZZL14, JIA+15, PR16, YK17b, YK17a, YW18, FMM20]. In this work we analyze primarily the two standard-cell-based solutions introduced in [ZZL13, ZZL14] and [JIA+15].

1.2 Our Contribution

For the first time in literature we perform a practical analysis of dedicated countermeasures against static power side-channel attacks (SPSCA) on real hardware. We have developed a prototype chip in 28 nm CMOS technology containing 11 cryptographic coprocessors with different levels of SCA protection applied and analyze the effectiveness of the countermeasures by performing practical attacks on the fabricated chip inside a climate chamber. We use a source measure unit (SMU) as power supply and high precision current measurement instrument simultaneously. Compared to previous works on dedicated test chips [MMR17, KMM19, Moo19, MMR20, Moo20], the 28 nm node constitutes the most advanced CMOS technology generation. We also make a contribution in the area of SPSCA countermeasures by proposing the first ever standard-cell-based balancing scheme that provides perfect data independence under the assumption that multiple instances of the same standard cell on the same chip have the same exact leakage characteristics. Of course, in reality this assumption can not hold due to the existence of (intra-die) process variations and aging-related degradation effects [KMM19]. Yet, our scheme, which we call exhaustive logic balancing (ELB), is likely as close as one may get towards achieving a fully data-independent static power consumption. Hence, the evaluation of this scheme gives insight about the practical limits of balancing techniques in general. Like we do for most of the hiding-based SPSCA countermeasures in this work we combine ELB with provably secure hardware masking in order to amplify the noise and show that the resulting circuit provides a high level of resistance against attacks. However, considering the very significant resource overhead of this method, some of the other countermeasure we evaluate here may be preferable from a cost efficiency standpoint. In general, our results can be used as a guideline for hardware designers to find a tradeoff between security and cost when trying to protect circuits from leaking information through the static power.

2 Countermeasures

In this section we introduce the hardware countermeasures which are implemented and practically evaluated throughout this work. Each countermeasure is applied to the serialized PRESENT-80 [BKL+07] block cipher implementation depicted in Figure 1. This area-optimized architecture has been proposed in [PMK+11].

2.1 High Threshold Voltage (HVT) [AE03]

Multi-Threshold Voltage CMOS (MTCMOS) is a popular technique available in most nanometer CMOS technology generations to reduce the leakage power of CMOS circuits while maintaining high performance. For this optimization strategy, standard cells exist in multiple versions with different threshold voltages. Cells with a lower threshold voltage (LVT) switch faster in response to their input signals and therefore are typically selected for gates in the critical path of a circuit. Cells with a higher threshold voltage (HVT) switch slower but consume a lower standby power [AE03]. In consequence, such cells are typically

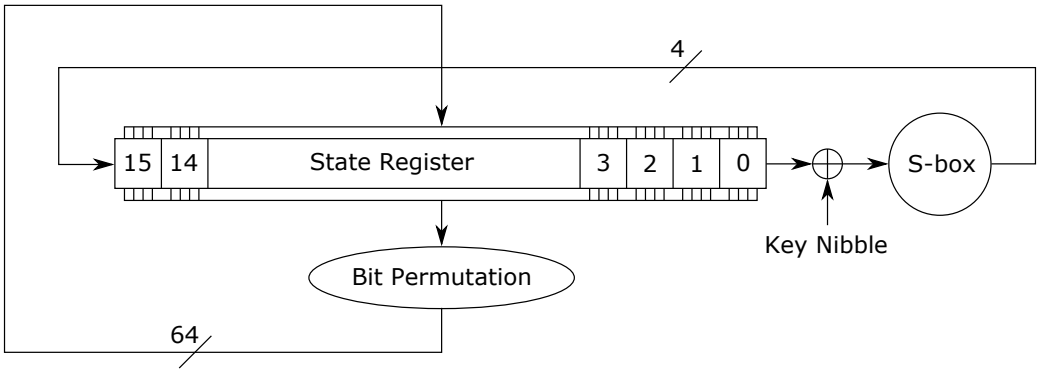


Figure 1: Architecture of the serialized PRESENT-80 hardware implementation. The key schedule is not shown.

selected for any path in a circuit where the timing constraints are not violated by the reduced performance of the cells. Although not explicitly proposed as a countermeasure against static power attacks yet, it is reasonable to assume that implementing a cryptographic primitive using only HVT cells with minimum drive strength will reduce the exploitable signal available to a static power adversary in relation to the noise level at the cost of a reduced performance of the circuit. In this work we verify whether this assumption holds by comparing two circuits derived from identical RTL code, one implemented for maximum performance and one implemented for minimum leakage current.

2.2 Random Start Index Shuffling (RSIS) [VMKS12]

Randomly changing the execution order of independent operations in a cryptographic algorithm is called shuffling and has been used as a side-channel countermeasure for many years. In modern symmetric block ciphers it is common to apply a non-linear substitution box (S-box) piecewise to the entire cipher state during the computation of each cipher round. These substitution boxes come in different sizes, but typical examples include 8-bit boxes like the AES S-box [DR98] and 4-bit boxes like the PRESENT S-box [BKL+07]. The substitution functions are applied to each byte or nibble of the state independently and the order of their execution, if executed sequentially, may be randomly reshuffled in each round or cipher iteration without affecting the outcome (when implemented correctly). In both examples, AES-128 and PRESENT-80, 16 consecutive S-box evaluations are performed in each cipher round whose order may be reshuffled. Essentially, there are two common methods to implement such a shuffling. Either a Random Permutation (RP) is chosen from all $16! \approx 2^{44.25}$ permutations or a Random Start Index (RSI) is chosen from 16 possible start indices [VMKS12]. First applications of both methods have focused on software implementations [HOM06, RPD09]. Later, the RSI method in particular has also been applied to hardware circuits [MMP11]. The idea behind both shuffling techniques is simple. When observing the execution of an unprotected cipher implementation, the adversary typically knows exactly at which point in time which part of the secret key is processed and, even more importantly, that in multiple executions of the same cipher the same key parts are processed at the same points in time. When shuffling is applied and it can safely be assumed that the adversary is unable to predict the permutation or the start index chosen then there are 16 possible positions where a certain targeted key part might be processed in a cipher iteration. For the most trivial side-channel attacks this translates to a reduction of the correlation between hypothesis and leakage recorded by a factor of about 16. The authors of [VMKS12] mention that this factor can be reduced to $\sqrt{16} = 4$ when

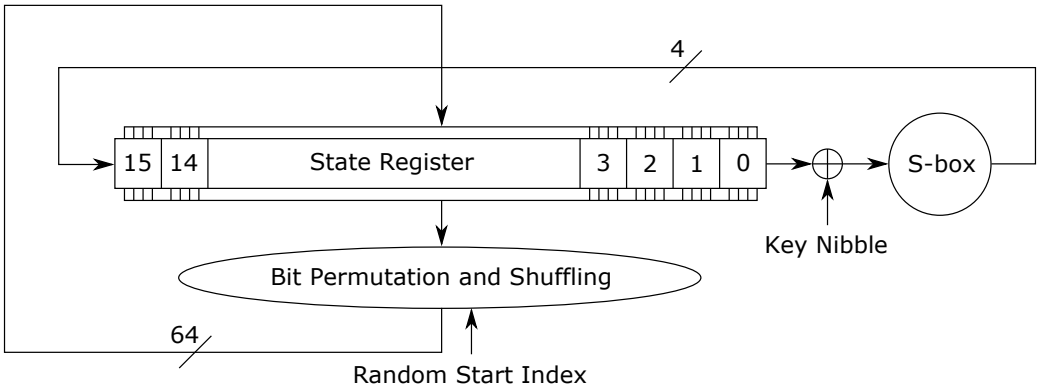
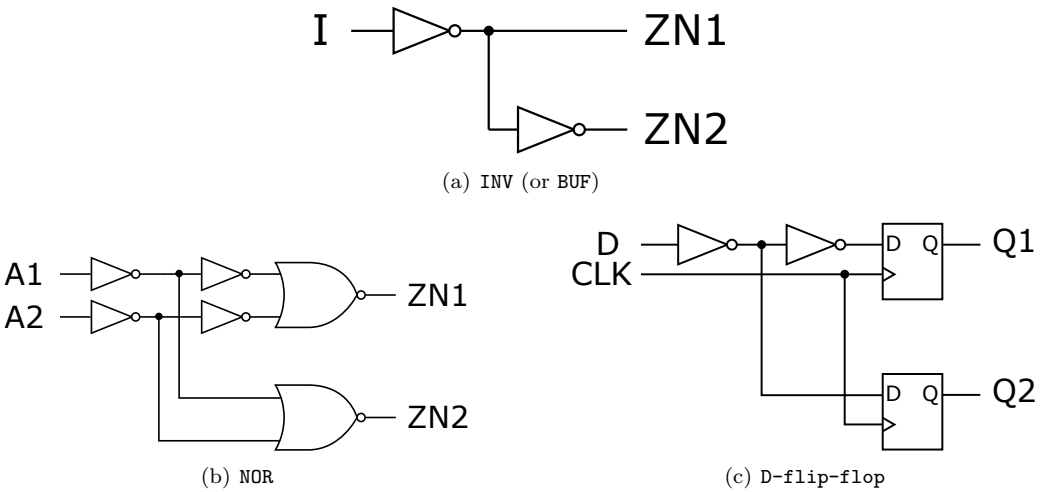


Figure 2: Architecture of the serialized PRESENT-80 implementation with Random Start Index Shuffling (RSIS). The key register, which also needs to be shuffled, is not shown.

so-called integrated DPA (or sliding window DPA) is used. The authors also voice some concerns regarding the perceived effectiveness of shuffling methods in general and Random Start Index (RSI) shuffling in particular when information about the chosen permutation or start index is leaked. However, this is less of a concern for hardware implementations where the randomness generation, reshuffling and cipher execution can be performed in parallel.
In this work we consider the RSI approach and apply it to a serialized PRESENT hardware implementation. A schematic of the result can be seen in Figure 2. The bit permutation operation, which previously could be realized purely through wire routing without any logic components, now receives the 4-bit random start index which determines by how many nibbles (0 to 15) the state register should be rotated. Clearly, this adds logic for the multiplexing, but since it is realized fully combinatorial, this change has no impact on the number of clock cycles required per encryption. Please note, that the same multiplexing logic is also required to rotate the current round key. While shuffling has primarily been proposed as a countermeasure against dynamic power or radiation side-channel attacks, it seems reasonable to expect that it also increases the difficulty of static power side-channel attacks. Especially when considering that the typical SPSCA adversary does not record a trace over time, but rather takes a single snapshot of the current state in the circuit. Thus, there are some qualitative differences between the impact of shuffling on the success probability of static power and dynamic power attacks which are discussed in Section 4.

2.3 Symmetric Dual-Rail Logic (SDRL) [ZZL13, ZZL14]

Symmetric Dual-Rail Logic (SDRL) has been proposed in [ZZL13, ZZL14] as the first standard-cell-based balancing technique dedicated to counteract static power attacks. The concept is very simple. In order to reduce the correlation between input vector applied and the leakage current of a certain cell, each standard cell is duplicated and the duplicated cell receives the inverted input vector. The general concept is illustrated for three exemplary cells in Figure 3. Please note, that outputs of gates which are not required can be left unconnected. Yet, a designer needs to make sure that the EDA tools do not remove gates whose output is not connected. From a high-level perspective, the inverter (or buffer) gate is perfectly balanced since each inverter receiving a logical '0' is accompanied by a second inverter receiving a logical '1'. Under the assumption that both inverters are instantiations of the same standard cell (including drive strength, threshold voltage, etc.) and that identical standard cells have identical leakage characteristics, the total leakage current should be indistinguishable regardless of which inverter receives the logical '0' and which



(a) `INV` (or `BUF`)

(b) `NOR`

(c) `D-flip-flop`

Figure 3: `INV` (or `BUF`), `NOR` and `D-flip-flop` in Symmetric Dual-Rail Logic (SDRL)

receives the logical '1'. Please note, that in contrast to dual-rail logic styles which are used as a countermeasure against dynamic power attacks, timing differences through imbalanced routing or other implementation specifics are not a concern here, since the timing of signals does not affect the static power consumption measured in a stable state. Hence, the inverter (or buffer) circuit should have a data-independent static power consumption. In reality, this is not entirely true, since two instances of the same CMOS standard cell never have exactly the same physical and electrical characteristics due to (intra-die) process variations [DGS+11, ABD+14] and aging-related degradation of transistors [KMM19]. Therefore, a small data-dependency of the leakage current typically remains. Regarding the other two gates in Figure 3, namely the `NOR` and the `D-flip-flop`, the situation is different. Here, the balancing is not perfect, even without considering process variations and aging mechanisms. However, a reduced dependency between input pattern and leakage current is achieved. Consider the SDRL `NOR` gate. The two cases (`A1=0`, `A2=0`) and (`A1=1`, `A2=1`) are indistinguishable when assuming identical leakage characteristics for multiple instance of identical gates, and so are (`A1=0`, `A2=1`) and (`A1=1`, `A2=0`). However, the two cases (`A1=0`, `A2=0`) and (`A1=0`, `A2=1`) are not indistinguishable from each other. Taking the numbers provided in Table 1 as an example, two `NOR` gates with inputs (`A1=0`, `A2=0`) and (`A1=1`, `A2=1`) have a leakage current of 172.16 nA + 38.42 nA = 210.58 nA, while two `NOR` gates with inputs (`A1=0`, `A2=1`) and (`A1=1`, `A2=0`) have a leakage current of 173.44 nA + 62.96 nA = 236.40 nA. In summary, the variation in the leakage current caused by different input vectors is decreased but not eliminated. A similar observation can be made for the `D-flip-flop`, since the value of `Q` also affects the leakage current. In order to investigate the effectiveness of this balancing technique to counteract static power analysis attacks in practice we have synthesized the serialized PRESENT implementation from Figure 1 exclusively with `INV`, `NOR` and `D-flip-flop` gates and replaced each cell with its SDRL counterpart before implementing the circuit on the chip. The results of its security analysis are presented in Section 4.

2.4 Quadruple Algorithmic Symmetrizing (QuadSeal) [JIA+15]

Quadruple Algorithmic Symmetrizing (QuadSeal) has been proposed as a countermeasure against both dynamic and static power analysis attacks in [JIA+15]. The goal of this method is to balance all Hamming weights and distances occurring in a cipher implementation and rotating the inputs to the balanced structures to account for remaining dependencies due

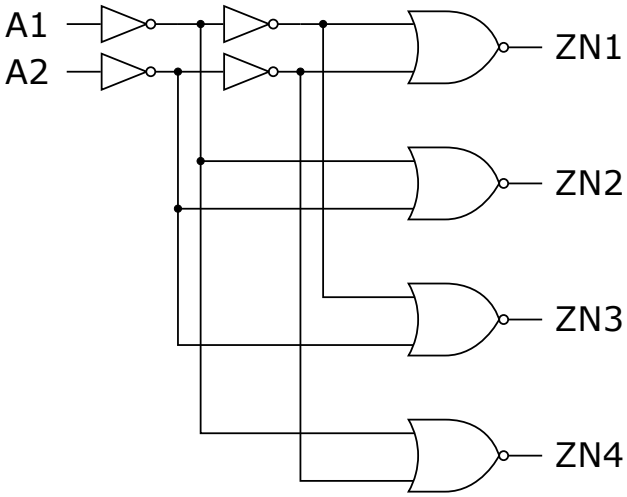


Figure 4: `NOR` gate with Exhaustive Logic Balancing (ELB).

to process variations, path imbalances and aging effects. In more detail, when applying this countermeasure to a cipher implementation, the unprotected circuit is quadrupled, while in three of the four circuits the S-box table is modified in the following way.

$$\mathrm{S}(\text{state\_nibble} \oplus \text{key\_nibble}) \tag{1}$$

$$\mathrm{S}^T(\text{state\_nibble} \oplus \overline{\text{key\_nibble}}) \tag{2}$$

$$\overline{\mathrm{S}}^T(\overline{\text{state\_nibble}} \oplus \text{key\_nibble}) \tag{3}$$

$$\overline{\mathrm{S}}(\overline{\text{state\_nibble}} \oplus \overline{\text{key\_nibble}}) \tag{4}$$

Then one of $4! = 24$ different permutations of inputs, keys and inverted inputs and keys is randomly selected (i.e., a 5-bit random number generator is sufficient). The full list of permutations is given in [JIA+15]. While the balancing of Hamming weights and Hamming distances is valuable to protect against attacks, the static power consumption of a combinatorial circuit like an S-box typically does not directly depend on the number of logical '1's in the inputs (see [KMM19] for an example of the input dependency of the leakage current exhibited by the PRESENT S-box in 65 nm CMOS). Thus, while this method is able to significantly reduce leakages from registers, it does not necessarily reduce the leakage from combinatorial S-boxes as well.

2.5 Exhaustive Logic Balancing (ELB) [this work]

In this paragraph we introduce Exhaustive Logic Balancing (ELB). ELB follows a similar concept as SDRL, but goes a step further. In particular, ELB makes sure that each gate is multiplied as often as the total number of different input vectors it may receive, and that in any stable state each of those input vectors is applied to one of the gates. An inverter or buffer gate with one input line can receive two different values, either logical '0' or logical '1'. Hence, the gate is duplicated and the gates can be connected as shown in Figure 3. A `NOR` gate with two input lines can receive four different input vectors and therefore needs to be quadrupled. In order to make sure that each input vector is received by one of the four `NOR` gates, the circuit has to be constructed as shown in Figure 4. Again, the output lines which are not required can be left unconnected, as long as it can be ensured that the IC design tools do not remove logic gates whose output is unconnected. Technically, any two-input logic gate can be quadrupled and implemented like this to

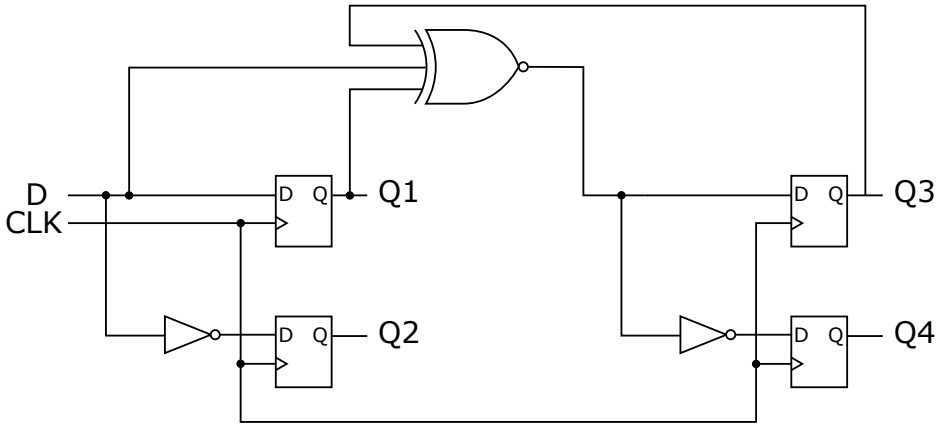


Figure 5: Logically balancing all D and Q values in the four `D-flip-flops` with a 3-input `XNOR`.

achieve a data-independent static power dissipation under the assumption that multiple instances of the same cell share exactly the same electrical characteristics. While this assumption is not precisely correct in reality, the characteristics of multiple instances of identical standard cells in close proximity on the same die should at least show a very similar electrical behavior. Unfortunately, the situation is more complex with respect to memory cells like flip-flops. At first sight it may appear that each flip-flop has only one data input and therefore simply needs to be duplicated, while giving the inverted input to the second flip-flop. However, as shown in Table 2, the leakage of a flip-flop also depends on the output value `Q`. Hence, the leakage depends on two data lines, which can have four different possible combinations. Therefore, each flip-flop needs to be quadrupled. Since `Q` is an output we can not apply the same technique as for the `NOR` gate above. Instead, we need to choose the input values for the four flip-flops as a function of their output values. One possible solution is shown in 5. Whenever applying a data value to input `D` and clocking once, each of the following four combinations is applied to one of the flip-flops: (`D=0`, `Q=0`), (`D=0`, `Q=1`), (`D=1`, `Q=0`), (`D=1`, `Q=1`). However, since the `XNOR` used for the logic function now causes its own data-dependent leakage current we need to replace it by a circuit with a balanced static power consumption. In this regard, we first express the `XNOR` function through only `NOR` and `INV` gates. The result is depicted in Figure 6. As a next step we replace those gates by their balanced version and apply some simple logic optimizations in order to reduce the number of balanced gates that have to be instantiated. The result can be seen in Figure 7. This final result achieves the optimal data independency that we are looking for. However, it is clear that the overhead to replace each flip-flop by this structure when trying to power balance a circuit is significant. The protection against static power attacks provided by this approach is analyzed experimentally in Section 4.

2.6 Threshold Implementation [NRR06]

Threshold Implementations (TIs) have been introduced in 2006 as the first hardware masking scheme that provides provable first-order security in the presence of glitches [NRR06]. Before the introduction of threshold implementations, masking schemes did not consider glitch resistance as a design objective and thus were commonly susceptible to a temporary recombination of the masks and masked values in combinatorial logic when the protected cryptographic primitive was realized as a hardware circuit. In consequence, early masking schemes could not easily guarantee practical first-order (or even higher-order) security in hardware. Since the introduction of TIs in 2006, the field of glitch-resistant masking has

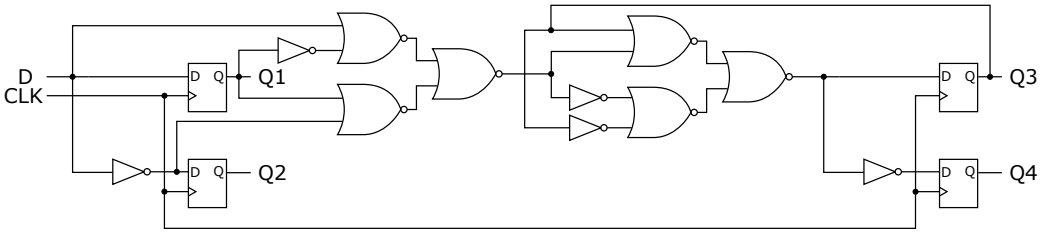


Figure 6: Logically balancing all D and Q values in the four `D-flip-flops` with only standard `NOR` and `INV` gates.

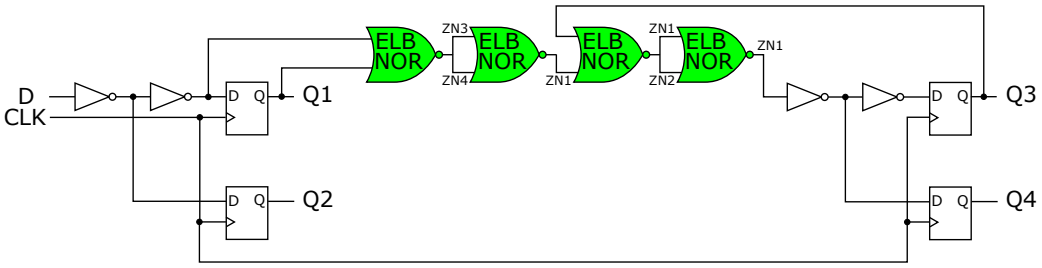


Figure 7: `D-flip-flop` with Exhaustive Logic Balancing (ELB). Each `ELB NOR` is an instantiation of Figure 4. The outputs to select (ZN1, ..., ZN4) are given.

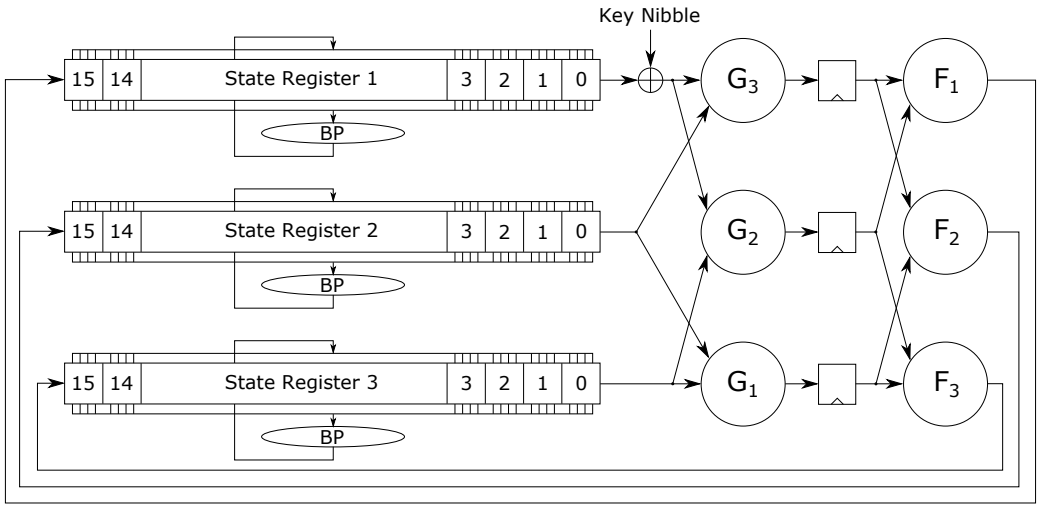
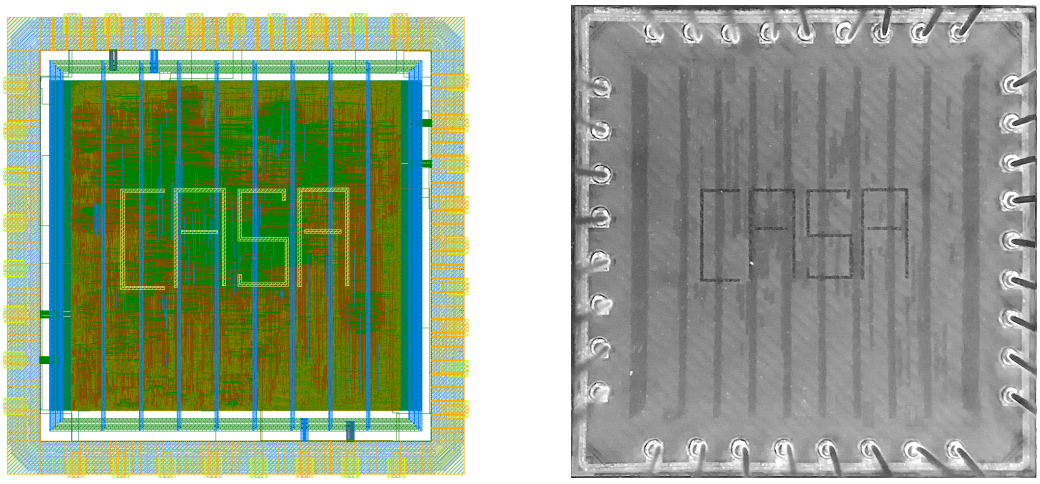


Figure 8: Architecture of the serialized PRESENT-80 threshold implementation. The key schedule is not shown.

grown significantly and many different schemes have been proposed and analyzed, including but not limited to [RBN+15, CRB+16, GMK16, GMK17, GM17, BDF+17, GM18, GIB18, FGP+18, MMSS19, CGLS20, CS20, SM21, CS21]. Yet, the plain and simple first-order threshold implementations are still one of the most popular SCA countermeasures today and arguably the easiest method to achieve provable first-order security in presence of glitches without requiring online randomness (although a recent work describes a $d+1$-masking scheme without the need for fresh randomness [SM21]).
Previous works have indicated that static power side-channel adversaries may potentially be able to exploit the higher-order leakages of threshold implementations (and other masking schemes) with a lower data complexity than attackers who observe the dynamic power consumption or radiation [MMR17, Moo19]. Yet, such higher-order leakages can be hidden effectively when the masking schemes are combined with proper hiding countermeasures. In that case even static power adversaries able to acquire measurements with low environmental and electronic noise influences are expected to require prohibitively large amounts of observations in order to extract sensitive information. Thus, in this work, we not only analyze the effectiveness of threshold implementations alone, but also in combination with the hiding countermeasures introduced earlier in this section and draw conclusions about the resulting protection levels.
A first-order threshold implementation of the serialized PRESENT architecture has been proposed in [PMK+11]. Since the PRESENT S-box has an algebraic degree of 3, its TI would normally require at least 4 shares to provide first-order security ($td+1$). However, a cheaper alternative than a 4-share TI can be achieved when decomposing the S-box S into two functions F and G with algebraic degree 2. In that case, the TI can be implemented with only 3 shares as long as a register stage separates the component functions of F and G in order to prevent glitch propagation between the combinatorial circuits. This type of 3-share first-order TI with a decomposed S-box has been introduced in [PMK+11] and is shown in Figure 8 applied to our serialized architecture. We use this implementation for our test circuits.
While most of the hiding countermeasures could be applied in a straightforward manner to the threshold implementation of PRESENT, this was not the case for QuadSeal. When attempting to combine the two countermeasures we encountered conceptual problems. The idea of QuadSeal is based on balancing Hamming weights and Hamming distances at the register stages to thwart side-channel leakage (dynamic and static). To achieve this, QuadSeal requires the implementation of four different (although related) substitution boxes. Thus, all four S-boxes need to be implemented as separate TIs. Additionally, in order to stick with the 3-share TI, each of them needs to be decomposed into two quadratic functions. This is possible for the four S-boxes required for QuadSeal PRESENT, but we have found no way of implementing the shared evaluation of their component functions in such a way that their collective outputs at each stage have a balanced Hamming weight and Hamming distance. In all evaluated cases either the intermediate register between the component functions, or their output was not properly balanced considering the whole quadrupled circuit. Therefore, we refrained from implementing a hybrid circuit that only realizes one of the two concepts properly.

3 Target and Setup

In the following we introduce the target device analyzed in this work and the measurement setup and procedure used to acquire the experimental results presented in Section 4.



(a) Layout (b) Photo

Figure 9: Layout and microscope photography of the 1380 µm × 1380 µm large 28 nm ASIC prototype.

3.1 Device Under Test (DUT)

The target for our practical analysis is a 28 nm CMOS ASIC prototype which we developed as a dedicated test chip for our investigation. The layout of the chip and a microscope photography of the manufactured and bonded die can be seen in Figure 9. The ASIC is 1380 µm × 1380 µm in size and has been designed to be operated at frequencies up to 100 MHz even under worst-case operating conditions. The chip requires an IO power supply of 1.8 V and a 0.9 V core power supply. The 981 µm × 981 µm large standard cell area in the center of the die contains 1 195 507 gate equivalents (GE) of logic. This includes 11 cryptographic co-processors based on the PRESENT block cipher which are practically analyzed for their static power side-channel security in Section 4. To be precise, each co-processor is based on the serialized PRESENT architecture described in Section 2 which is depicted in Figure 1 without masking applied and in Figure 8 as a threshold implementation. The 11 cipher cores differ from each other in the particular countermeasures that are employed to avoid key extraction via static power side-channel analysis. The levels of protection range from an unprotected circuit to a combination of exhaustive balancing and provably-secure masking. The full list of circuits evaluated in this work including their post-layout area consumption and an overhead comparison is given in Table 3.

The PRESENT core denoted by *High Performance (HP)* is a raw and unprotected implementation of the serialized cipher architecture shown in Figure 1, but optimized for maximum clock frequency. As already discussed in Section 2 such an optimization goal favors the use of low threshold voltage (LVT) cells in all timing critical paths. It is noteworthy that the HVT circuit, optimized for minimum leakage current, is smaller than the HP circuit, despite the fact that the slower high threshold voltage (HVT) cells are identically sized as the faster low threshold voltage (LVT) cells. The difference comes from the selection of standard cells with the lowest drive strength in the HVT circuit, which are generally smaller and consume less power than cells with a higher driving strength. Table 3 clearly shows that all other protected circuits come at an area overhead, which proves to be significant in some cases. Table 4 presents post-layout estimations of the critical path delay (or latency), maximum operating frequency and average power consumption (when operated at 100 MHz) for typical operating conditions (25 °C, 0.9 V) of all 11 circuits,

Table 3: Post-layout area consumption of the PRESENT co-processors.

PRESENT Core	Area [GE]	Overhead factor
High Performance (HP)	2 535.00	× 1.00
High Threshold Voltage (HVT)	2 406.67	× 0.95
Random Start Index Shuffling (RSIS)	2 613.00	× 1.03
Symmetric Dual-Rail Logic (SDRL)	10 789.33	× 4.26
Quadruple Algorithmic Symmetrizing (QuadSeal)	12 636.33	× 4.98
Exhaustive Logic Balancing (ELB)	20 207.00	× 7.97
Threshold Implementation + HP	7 233.33	× 2.85
Threshold Implementation + HVT	6 982.67	× 2.75
Threshold Implementation + RSIS	9 856.33	× 3.89
Threshold Implementation + SDRL	27 907.33	× 11.01
Threshold Implementation + ELB	58 442.33	× 23.05

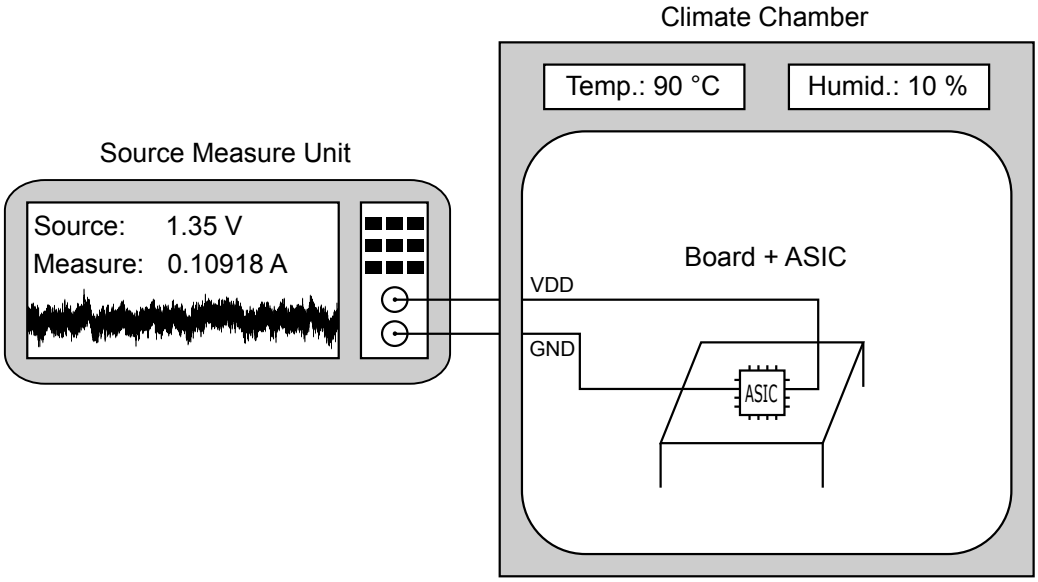
Table 4: Post-layout estimations of the critical path delay, maximum frequency and average power consumption at 100 MHz operation for all PRESENT co-processors.

PRESENT Core	Crit. Path [ps]	Freq. [MHz]	Dyn. Power [µW]	Stat. Power [µW]
HP	435.7851	2294.7	111.2826	35.9710
HVT	563.7664	1773.8	107.7309	1.7616
RSIS	597.3369	1674.1	103.4829	12.3302
SDRL	2064.9476	484.3	240.2608	4.1869
QuadSeal	785.0767	1273.8	463.3175	51.4306
ELB	1959.9415	510.2	673.4377	9.4815
TI + HP	358.3832	2790.3	277.4649	101.9850
TI + HVT	594.5498	1681.9	309.3094	3.7409
TI + RSIS	612.0424	1633.9	312.5164	54.4392
TI + SDRL	2510.5112	398.3	650.3030	7.7135
TI + ELB	2377.2272	420.7	1981.1074	17.3661

extracted using the Synopsys IC design flow. While all PRESENT cores require the same number of clock cycles (547) for one encryption (see [PMK+11]), the protected versions clearly show a reduced maximum frequency and average power consumption compared to the unprotected implementation. It is important to clarify, however, that the ASIC has been designed to operate at frequencies up to 100 MHz and even the slowest circuits in Table 4 achieve frequencies well above that threshold (at least for typical operating conditions). Thus, none of the circuits except the HP versions have been tightly constrained by their clock period and higher frequencies at the price of an increased area and energy consumption would definitely be possible. The comparably low static power consumption for SDRL and ELB circuits can be explained by the fact that they also consist of HVT cells with minimum drive strength exclusively. In part, this also causes their significantly higher latency compared to the other circuits.

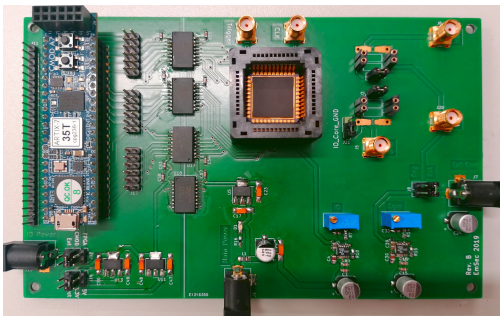
3.2 Measurement Setup

The measurement setup utilized in our practical experiments is depicted in Figure 10. On top, a schematic of the full setup used to acquire the current measurements is given. The measurement board containing the mounted chip is placed inside a climate chamber to precisely control the environmental temperature. In contrast to all previous works we have used a source measure unit (SMU) a.k.a. sourcemeter for the static power measurements.



(a) Setup

(b) Source Measure Unit (SMU)



(c) Custom Measurement Board

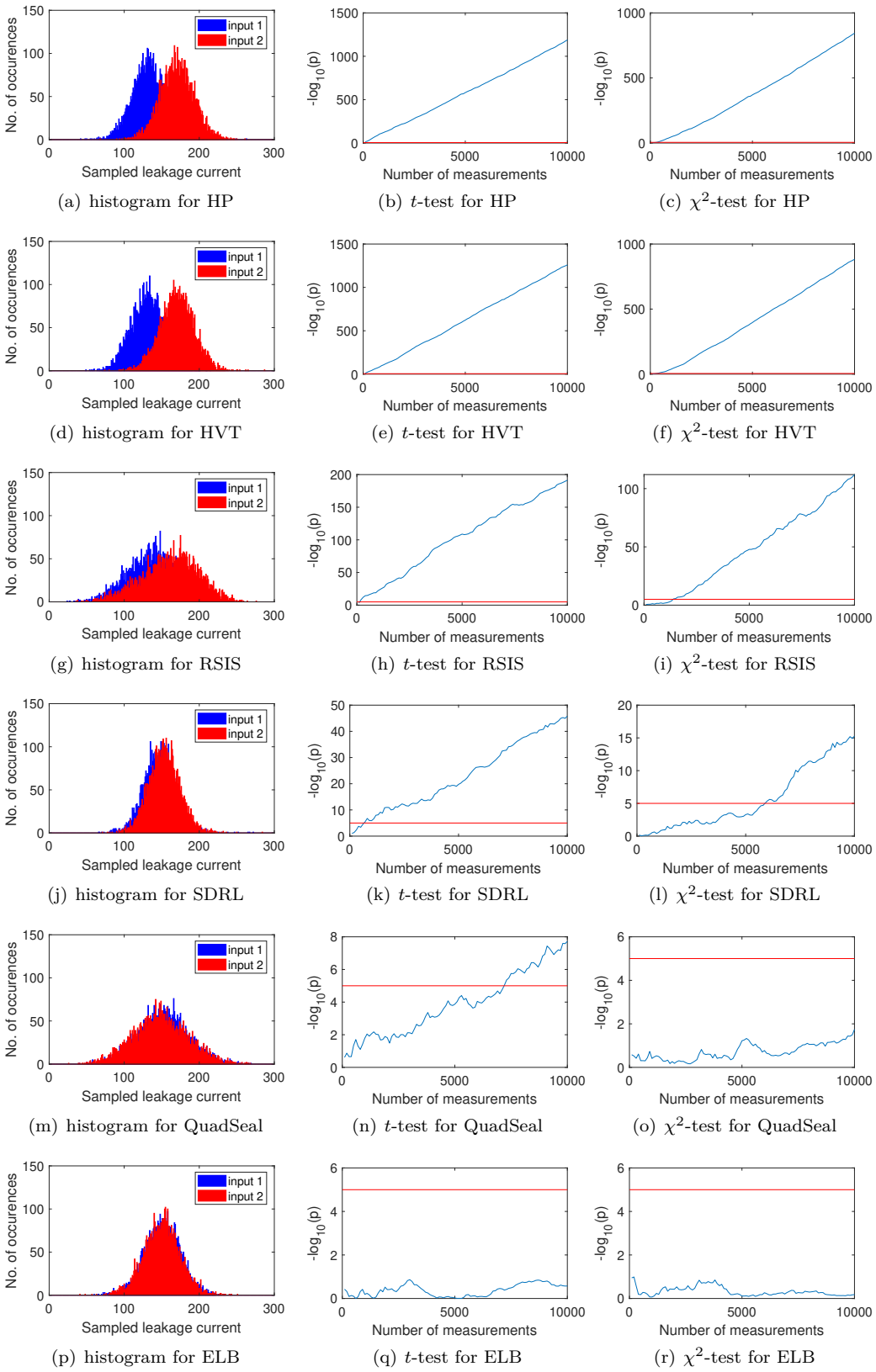
Figure 10: Measurement Setup used in the practical experiments.

A photography of the Keithley 2450 SourceMeter [Kei] is shown in Figure 10(b). This instrument has specifically been designed for characterizing nano-scale semiconductors and other small-geometry and low-power devices. In our experiments we have used it to simultaneously supply the core voltage to the chip and measure the leakage currents through the device. A photography of the custom measurement board can be seen in Figure 10(c). We have designed this PCB for evaluating our 28 nm test chip which can be seen in the middle of the board plugged into a PLCC44 socket. A Digilent Cmod A7 FPGA board [Dig] can be plugged into the 48 pin DIP socket on the left of the measurement board in order to function as an interface between the ASIC and the PC.
The procedure to acquire static power measurements using this setup works as follows. The FPGA board pauses the global clock signal of the ASIC during the first round of the PRESENT cipher operation and simultaneously generates a trigger signal to the SMU. The SMU waits for 20ms after receiving the positive trigger edge, takes a current measurement and saves it into the internal buffer. Afterwards, the SMU goes back to idle mode, waiting for the next trigger to arrive. The clock signal is continued and the PRESENT core completes its computation. Then a new encryption is initiated and the process repeats from the beginning. As soon as 100 measurements are collected, the internal buffer is read

out by the measurement script and the data is saved on the hard drive. Using this method, the data acquisition takes about 108.24 ms per measurement (the time required for fetching the buffer from the instrument and saving the traces to the hard drive is already included), which means that the acquisition of 1 000 000 traces takes about 30 hours and 4 minutes. This is significantly faster than many previous works [MMR17, KMM19, Moo19, Moo20].

4 Experimental Results

In this section we present our experimental analysis of the 11 different PRESENT co-processors realized on the 28 nm CMOS ASIC. As a first step we analyze the hiding countermeasures alone, under normal operating conditions. In this regard, we have placed the measurement board in the climate chamber set to a constant temperature of 20 °C and powered the ASIC by its nominal core voltage of 0.9 V. To compare the leakage exhibited by the PRESENT cores in this scenario we have collected measurements for two different fixed inputs (fixed-vs-fixed) in a randomly interleaved manner in order to perform a leakage assessment using the t-test [SM15] and the χ^2-test [MRSS18] respectively. The results are depicted in Figure 11. For a visual comparison between the different techniques we have also plotted the histograms for the two groups which have been used to extract the -$\log_{10}$(p) confidence values.
The first thing to notice is that the HVT circuit does not seem to hide the data dependency any better than the high performance (HP) implementation. The differences between the means of the leakage distributions and the test results look very similar in both cases. The other four protected implementations perform better and a gradual reduction of the test confidence and the number of traces to overcome the confidence threshold can be observed from top to bottom. The exhaustive logic balancing (ELB) achieves the best results in these experiments, since both statistical tests fail to reject the null hypothesis given a data set of 10 000 traces. Since multiple previous works have demonstrated that increasing the supply voltage and the temperature of the device under test significantly increases the leakage currents in relation to the measurement noise [Moo19, MMR20], we have repeated the same measurements as before with the temperature set to 90 °C and a core supply voltage of 1.35 V (50% over-voltage). Those results are shown in Figure 12. All results are improved by a significant margin in terms of confidence and number of traces to detect leakage. Therefore, we are able to confirm that the manipulation of operating conditions is a viable method to enhance the magnitude of the leakage currents and to improve the overall quality of the measurement results. Apart from the significantly increased distinguishability across the board, the most interesting observation is probably that the ELB circuit now also shows a significant amount of leakage. Hence, we can conclude that the variations of the physical and electrical characteristics between identical CMOS standard cells placed in close proximity to each other is certainly large enough to weaken the balancedness of the static power consumption sufficiently to detect a clear data dependency. In part this may be caused by the (uneven) aging-related degradation of the transistors which is immediately amplified when the power supply and temperature are as drastically increased as in our experiments. However, we have used a fresh sample of the ASIC for these experiments to avoid a prior manifestation of effects like described in [KMM19].
As a next step we now analyze the combined hiding and masking countermeasures, again under the leakage-enhancing operating conditions of 90 °C and 1.35 V. The results are depicted in Figure 13. Here, the t-test is performed at first, second and third order. It can be seen that no data dependency is reported with confidence for any of the first- or second-order tests. The χ^2-test is independent of statistical moments and, like the third-order t-test, reports leakage in four of the five experiments. Only for the combination of the threshold implementation with the exhaustive logic balancing the tests fail to reject



(a) histogram for HP (b) t-test for HP (c) χ^2-test for HP

(d) histogram for HVT (e) t-test for HVT (f) χ^2-test for HVT

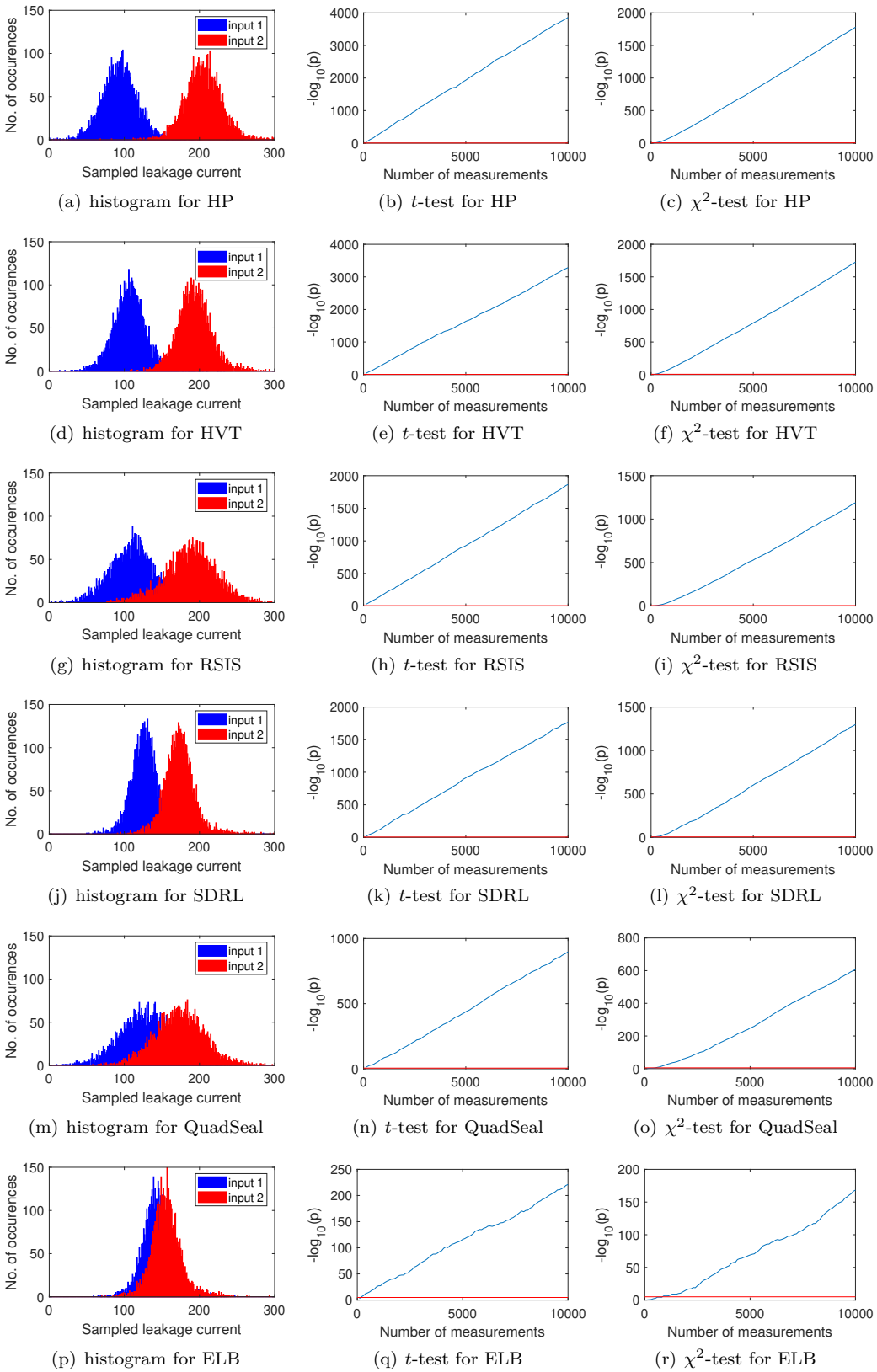
(g) histogram for RSIS (h) t-test for RSIS (i) χ^2-test for RSIS

(j) histogram for SDRL (k) t-test for SDRL (l) χ^2-test for SDRL

(m) histogram for QuadSeal (n) t-test for QuadSeal (o) χ^2-test for QuadSeal

(p) histogram for ELB (q) t-test for ELB (r) χ^2-test for ELB

Figure 11: Leakage assessment of the (unmasked) hiding countermeasures at 20 °C, 0.9 V.



(a) histogram for HP (b) t-test for HP (c) χ^2-test for HP

(d) histogram for HVT (e) t-test for HVT (f) χ^2-test for HVT

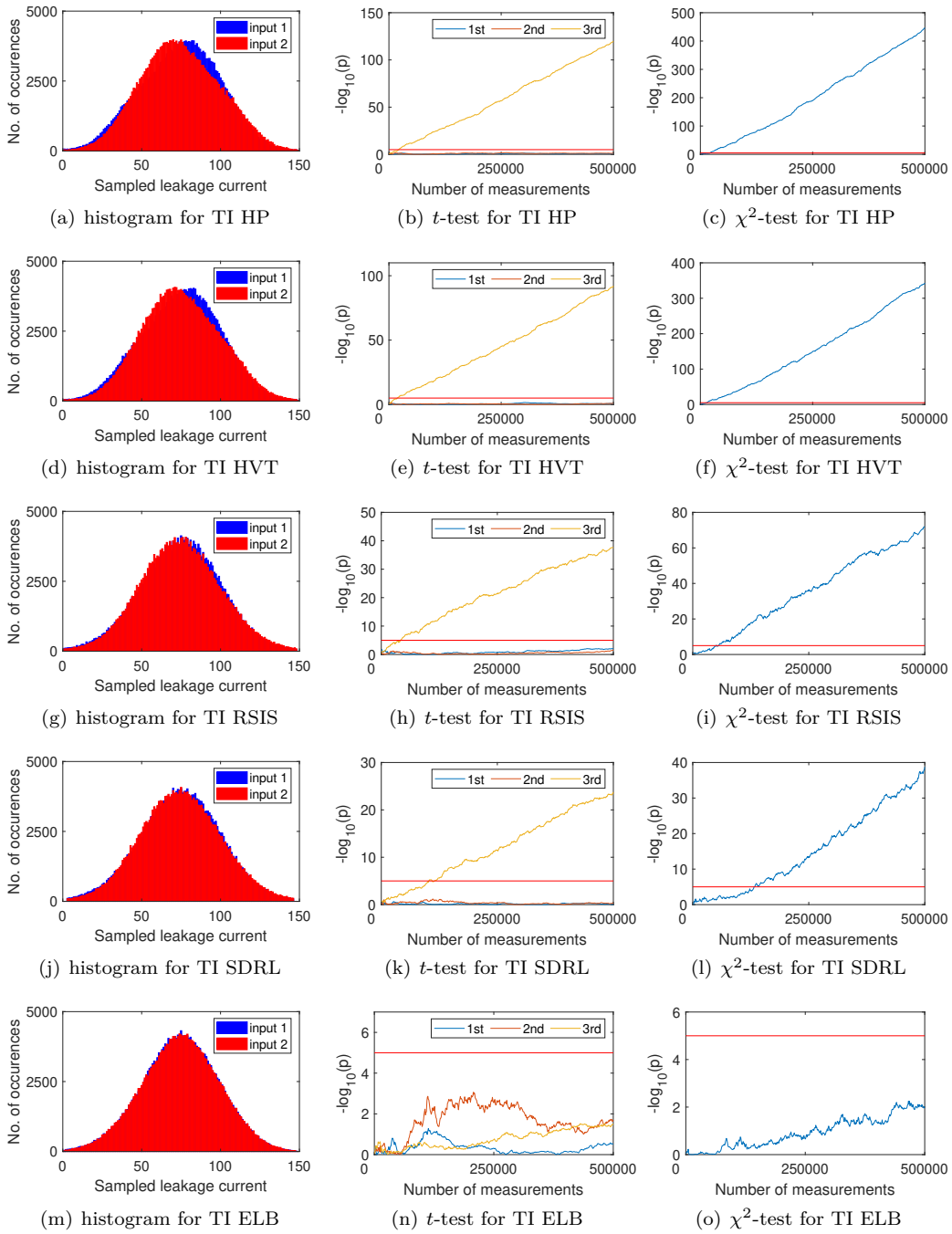
(g) histogram for RSIS (h) t-test for RSIS (i) χ^2-test for RSIS

(j) histogram for SDRL (k) t-test for SDRL (l) χ^2-test for SDRL

(m) histogram for QuadSeal (n) t-test for QuadSeal (o) χ^2-test for QuadSeal

(p) histogram for ELB (q) t-test for ELB (r) χ^2-test for ELB

Figure 12: Leakage assessment of the (unmasked) hiding countermeasures at 90 °C, 1.35 V (50% over-voltage).



(a) histogram for TI HP (b) t-test for TI HP (c) χ^2-test for TI HP

(d) histogram for TI HVT (e) t-test for TI HVT (f) χ^2-test for TI HVT

(g) histogram for TI RSIS (h) t-test for TI RSIS (i) χ^2-test for TI RSIS

(j) histogram for TI SDRL (k) t-test for TI SDRL (l) χ^2-test for TI SDRL

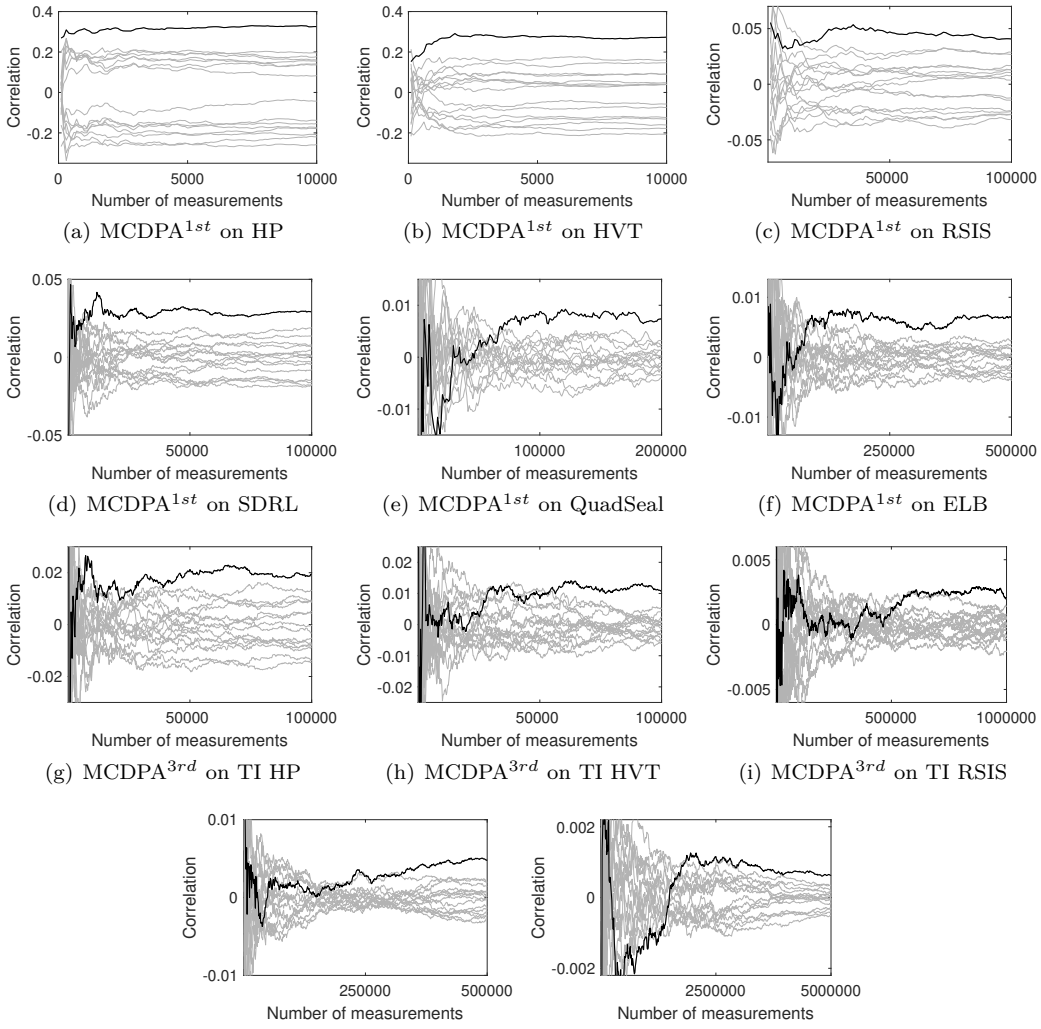
(m) histogram for TI ELB (n) t-test for TI ELB (o) χ^2-test for TI ELB

Figure 13: Leakage assessment of the combined masking and hiding countermeasures at 90 °C, 1.35 V (50% over-voltage).

the null hypothesis given 500 000 traces. Our leakage assessment results already give us a decent idea about the level of protection each of the (combined) countermeasures is able to provide. However, the presence of detectable leakage alone does not necessarily prove the insecurity of a device. Leakages about the inputs and outputs, independent of the secret internal key, are flagged by leakage detection methods but do not necessarily undermine the security of a device. Hence, we have also attempted to perform key recovery attacks on all 11 different PRESENT co-processors. In order to provide a fair comparison we chose to perform Moments-Correlating DPA (MCDPA) attacks on the targets as this collision-based method does not depend on the choice of a suitable leakage model [MS16]. This type of attack has already been applied to static power measurements in [MMR17]. We have also attempted classical CPA attacks with explicit leakage models [BCO04], but learned that this approach leads to a distortion of the comparison. In fact, for some of the circuits the Hamming weight of the S-box output is the optimal model, for others its a single S-box output bit (LSB, MSB, ...) or a combination of multiple bits. None of the specific models we tested worked well on all circuits. Please note, that transitional models like the Hamming distance between consecutively processed values are not a promising candidate here since the static leakage does not naturally capture transitions. Only the small difference in the leakage of a flip-flop cell between (`D=1, Q=0`) and (`D=1, Q=1`) (or analogous combinations) could cause a correlation between the Hamming distance of values and the measured leakage current. The independence of a leakage model featured by Moments-Correlating DPA is indeed a crucial property for a fair comparison of the vulnerability of the circuits. The attack succeeds in all experiments independent of an explicit model and allows a comparison of the data complexity of the attacks. In fact, whenever the leakage of an intermediate value does not closely resemble one of the classical leakage models like the Hamming weight or distance, but rather a more complex leakage function (common for protected implementations) it is plausible that MCDPA is able to extract more information than classical CPA. This expectation is backed up by our observation that no individual CPA in our tests could outperform the MCDPA with respect to attacks on the TI variants. Our MCDPA results for all circuits are shown in Figure 14. Please note the differences in the number of traces utilized for each of the attacks. In order to enable an easier comparison between the different results we have assembled Table 5, which not only lists the number of traces required for a successful recovery of a sub-key difference and the resulting correlation coefficient, but also puts the data complexity for an attack in relation to the area of the circuit. The only discrepancy between the leakage detection results and the key recovery attacks is that the shuffled variants, in relation to the other implementations, show leakage early and strong in a detection scenario, but are still relatively hard to exploit. Due to the nature of shuffling, a stronger attack could probably be performed when recording the leakage after each clock cycle of a cipher round and thereby building a leakage trace over time (similar to a dynamic power measurement). In that case, integrated DPA attacks could reduce the data complexity for a key recovery (see Section 2). Technically, with unrestricted control over the clock signal (the strongest attacker model in this context), the adversary would be capable of single-stepping through the whole encryption operation and measuring the leaked current after each clock cycle. However, we do not consider such an analysis here in order to keep all attacks identical. Tailoring each attack to the countermeasure under analysis would greatly complicate the comparison.

5 Conclusions and Future Work

The standby power of CMOS chips silently leaks information to potential adversaries. Several practical case studies have demonstrated this concerning fact throughout the last couple of years. Common side-channel countermeasures used to thwart dynamic leakage



(a) MCDPA^{1st} on HP (b) MCDPA^{1st} on HVT (c) MCDPA^{1st} on RSIS

(d) MCDPA^{1st} on SDRL (e) MCDPA^{1st} on QuadSeal (f) MCDPA^{1st} on ELB

(g) MCDPA^{3rd} on TI HP (h) MCDPA^{3rd} on TI HVT (i) MCDPA^{3rd} on TI RSIS

(j) MCDPA^{3rd} on TI SDRL (k) MCDPA^{3rd} on TI ELB

Figure 14: MCDPA attacks on all countermeasures at 90 °C, 1.35 V (50% over-voltage). MCDPA^{1st} = first-order MCDPA; MCDPA^{3rd} = third-order MCDPA.

Table 5: Data complexities and correlation coefficients for all MCDPA attacks. Data complexities given as absolute values (DC MCDPA) and per gate equivalents (DC / GE).

PRESENT Core	Area [GE]	DC MCDPA	DC / GE	Correlation Coefficient
HP	2 535.00	< 100	< 0.039	0.3258
HVT	2 406.67	200	0.083	0.2734
RSIS	2 613.00	15 000	5.741	0.04069
SDRL	10 789.33	8 800	0.816	0.02907
QuadSeal	12 636.33	67 000	5.302	0.007471
ELB	20 207.00	120 000	5.939	0.006618
TI + HP	7 233.33	23 600	3.263	0.01913
TI + HVT	6 982.67	53 000	7.590	0.01070
TI + RSIS	9 856.33	596 000	**60.469**	0.002144
TI + SDRL	27 907.33	320 000	11.467	0.004860
TI + ELB	58 442.33	**2 930 000**	50.135	**0.0006170**

attacks have shown to be of limited effectiveness against this threat. Thus, specialized countermeasures based on the principles and characteristics of the static power consumption of CMOS devices need to be developed and tested. Practical experiments are especially vital in this process as simulation results often do not sufficiently model all mechanisms that play into the vulnerability of a device. In this work we tried to make a first step in that direction by implementing and evaluating a set of countermeasures consisting of both, previously proposed techniques from the literature and novel ideas, on a 28 nanometer CMOS chip. Our experiments have partially been performed under extreme environmental conditions (90 °C and 50% over-voltage) to figuratively squeeze the information out of our target device. The result of that analysis is that none of the tested countermeasures could withstand attacks with 3 000 000 traces and more than the half of the countermeasure-protected circuits allow extraction of sub-keys with less than 100 000 traces. The strongest protection was achieved by a combination of exhaustive balancing and provably secure hardware masking. However, this combined countermeasure increases the circuit size by a factor of 23, the critical path by a factor of 4, the energy consumption by a factor of 14 and was still susceptible to attacks. This result also speaks to the limits of balancing techniques in general, since even exhaustively balanced circuits are not sufficiently balanced to avoid key extraction. Purely algorithmic approaches, like a combination of masking and shuffling achieve a better cost efficiency, but exhibit a much higher leakage in a detection scenario which may become problematic for device certification. In summary, it seems that existing countermeasures, even rather expensive ones, can only increase the data complexity of static power attacks to a certain extent. The quest for better solutions has to continue.

Future Work. From our point of view, masking schemes which avoid univariate leakage altogether could potentially provide a high level of resistance against SPSCA adversaries. However, that is conceptually difficult to realize since univariate leakage with respect to static power adversaries is much more inclusive than univariate leakage with respect to dynamic power adversaries. A static power adversary can virtually see the cumulative leakage of any gate in a circuit in a single snapshot and not only the leakage of gates that switch simultaneously. Yet, thinking about approaches in this direction may be worthwhile.

Acknowledgments

The work described in this paper has been supported in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972 and through the project 271752544 "NaSCA: Nano-Scale Side-Channel Analysis".

References

[ABD+14] Massimo Alioto, Simone Bongiovanni, Milena Djukanovic, Giuseppe Scotti, and Alessandro Trifiletti. Effectiveness of leakage power analysis attacks on dpa-resistant logic styles under process variations. *IEEE Trans. Circuits Syst. I Regul. Pap.*, 61-I(2):429–442, 2014.

[AE03] Mohab Anis and Mohamed Elmasry. *Multi-Threshold CMOS Digital Circuits: Managing Leakage Power*. Springer, 2003.

[AO14] Zia Abbas and Mauro Olivieri. Impact of technology scaling on leakage power in nano-scale bulk CMOS digital standard cells. *Microelectronics Journal*, 45(2):179–195, 2014.

[BCO04] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. 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 16–29. Springer, 2004.

[BCS+17] Davide Bellizia, Danilo Cellucci, Valerio Di Stefano, Giuseppe Scotti, and Alessandro Trifiletti. Novel measurements setup for attacks exploiting static power using DC pico-ammeter. In *2017 European Conference on Circuit Theory and Design, ECCTD 2017, Catania, Italy, September 4-6, 2017*, pages 1–4. IEEE, 2017.

[BDF+17] Gilles Barthe, François Dupressoir, Sebastian Faust, Benjamin Grégoire, François-Xavier Standaert, and Pierre-Yves Strub. Parallel implementations of masking schemes and the bounded moment leakage model. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part I*, volume 10210 of *Lecture Notes in Computer Science*, pages 535–566, 2017.

[BKL+07] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 2007.

[CGLS20] Gaëtan Cassiers, Benjamin Grégoire, Itamar Levi, and François-Xavier Standaert. Hardware private circuits: From trivial composition to full verification. *IACR Cryptol. ePrint Arch.*, 2020:185, 2020.

[CRB+16] Thomas De Cnudde, Oscar Reparaz, Begül Bilgin, Svetla Nikova, Ventzislav Nikov, and Vincent Rijmen. Masking AES with d+1 shares in hardware. In Benedikt Gierlichs and Axel Y. Poschmann, editors, *Cryptographic Hardware and Embedded Systems - CHES 2016 - 18th International Conference, Santa Barbara, CA, USA, August 17-19, 2016, Proceedings*, volume 9813 of *Lecture Notes in Computer Science*, pages 194–212. Springer, 2016.

[CS20] Gaëtan Cassiers and François-Xavier Standaert. Trivially and efficiently composing masked gadgets with probe isolating non-interference. *IEEE Trans. Inf. Forensics Secur.*, 15:2542–2555, 2020.

[CS21] Gaëtan Cassiers and François-Xavier Standaert. Provably secure hardware masking in the transition- and glitch-robust probing model: Better safe than sorry. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2021(2):136–158, 2021.

[DGS+11] Milena Djukanovic, Luca Giancane, Giuseppe Scotti, Alessandro Trifiletti, and Massimo Alioto. Leakage power analysis attacks: Effectiveness on DPA resistant logic styles under process variations. In *International Symposium on Circuits and Systems (ISCAS 2011), May 15-19 2011, Rio de Janeiro, Brazil*, pages 2043–2046. IEEE, 2011.

[Dig] Digilent. Cmod a7 reference manual. https://reference.digilentinc.com/reference/programmable-logic/cmod-a7/reference-manual. Accessed: 15.01.2021.

[DR98] Joan Daemen and Vincent Rijmen. The block cipher rijndael. In Jean-Jacques Quisquater and Bruce Schneier, editors, *Smart Card Research and Applications, This International Conference, CARDIS '98, Louvain-la-Neuve, Belgium, September 14-16, 1998, Proceedings*, volume 1820 of *Lecture Notes in Computer Science*, pages 277–284. Springer, 1998.

[FGP+18] Sebastian Faust, Vincent Grosso, Santos Merino Del Pozo, Clara Paglialonga, and François-Xavier Standaert. Composable masking schemes in the presence of physical defaults & the robust probing model. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(3):89–120, 2018.

[FMM20] Bijan Fadaeinia, Thorben Moos, and Amir Moradi. BSPL: balanced static power logic. *IACR Cryptol. ePrint Arch.*, 2020:558, 2020.

[GIB18] Hannes Groß, Rinat Iusupov, and Roderick Bloem. Generic low-latency masking in hardware. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(2):1–21, 2018.

[GM17] Hannes Groß and Stefan Mangard. Reconciling d+1 masking in hardware and software. In Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, volume 10529 of *Lecture Notes in Computer Science*, pages 115–136. Springer, 2017.

[GM18] Hannes Groß and Stefan Mangard. A unified masking approach. *J. Cryptographic Engineering*, 8(2):109–124, 2018.

[GMK16] Hannes Groß, Stefan Mangard, and Thomas Korak. Domain-oriented masking: Compact masked hardware implementations with arbitrary protection order. In Begül Bilgin, Svetla Nikova, and Vincent Rijmen, editors, *Proceedings of the ACM Workshop on Theory of Implementation Security, TISCCS 2016 Vienna, Austria, October, 2016*, page 3. ACM, 2016.

[GMK17] Hannes Groß, Stefan Mangard, and Thomas Korak. An efficient side-channel protected AES implementation with arbitrary protection order. In Helena Handschuh, editor, *Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14-17, 2017, Proceedings*, volume 10159 of *Lecture Notes in Computer Science*, pages 95–112. Springer, 2017.

[GSST07] Jacopo Giorgetti, Giuseppe Scotti, Andrea Simonetti, and Alessandro Trifiletti. Analysis of data dependence of leakage current in cmos cryptographic hardware. In *Proceedings of the 17th ACM Great Lakes Symposium on VLSI*, GLSVLSI '07, page 78–83, New York, NY, USA, 2007. Association for Computing Machinery.

[HMY13] Basel Halak, Julian P. Murphy, and Alexandre Yakovlev. Power balanced circuits for leakage-power-attacks resilient design. *SAI*, pages 1178–1183, July 2013.

[HOM06] Christoph Herbst, Elisabeth Oswald, and Stefan Mangard. An AES smart card implementation resistant to power analysis attacks. In Jianying Zhou, Moti Yung, and Feng Bao, editors, *Applied Cryptography and Network Security, 4th International Conference, ACNS 2006, Singapore, June 6-9, 2006, Proceedings*, volume 3989 of *Lecture Notes in Computer Science*, pages 239–252, 2006.

[JIA+15] Darshana Jayasinghe, Aleksandar Ignjatovic, Jude Angelo Ambrose, Roshan G. Ragel, and Sri Parameswaran. Quadseal: Quadruple algorithmic symmetrizing countermeasure against power based side-channel attacks. In *CASES*, pages 21–30, 2015.

[Kei] Tektronix Keithley. 2450 sourcemeter smu instrument datasheet. https://de.tek.com/datasheet/smu-2400-graphical-sourcemeter/model-2450-touchscreen-source-measure-unit-smu-instrument-. Accessed: 15.01.2021.

[KMM19] Naghmeh Karimi, Thorben Moos, and Amir Moradi. Exploring the effect of device aging on static power analysis attacks. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2019(3):233–256, 2019.

[MMP11] Amir Moradi, Oliver Mischke, and Christof Paar. Practical evaluation of DPA countermeasures on reconfigurable hardware. In *HOST 2011, Proceedings of the 2011 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), 5-6 June 2011, San Diego, California, USA*, pages 154–160. IEEE Computer Society, 2011.

[MMR17] Thorben Moos, Amir Moradi, and Bastian Richter. Static power side-channel analysis of a threshold implementation prototype chip. In David Atienza and Giorgio Di Natale, editors, *Design, Automation & Test in Europe Conference & Exhibition, DATE 2017, Lausanne, Switzerland, March 27-31, 2017*, pages 1324–1329. IEEE, 2017.

[MMR20] Thorben Moos, Amir Moradi, and Bastian Richter. Static power side-channel analysis - an investigation of measurement factors. *IEEE Trans. VLSI Syst.*, 28(2):376–389, 2020.

[MMSS19] Thorben Moos, Amir Moradi, Tobias Schneider, and François-Xavier Standaert. Glitch-resistant masking revisited or why proofs in the robust probing model are needed. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2019(2):256–292, 2019.

[Moo19] Thorben Moos. Static power SCA of sub-100 nm CMOS asics and the insecurity of masking schemes in low-noise environments. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2019(3):202–232, 2019.

[Moo20] Thorben Moos. Unrolled cryptography on silicon A physical security analysis. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2020(4):416–442, 2020.

[Mor14] Amir Moradi. Side-channel leakage through static power - should we care about in practice? In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September 23-26, 2014. Proceedings*, volume 8731 of *Lecture Notes in Computer Science*, pages 562–579. Springer, 2014.

[MRSS18] Amir Moradi, Bastian Richter, Tobias Schneider, and François-Xavier Standaert. Leakage detection with the x2-test. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(1):209–237, 2018.

[MS16] Amir Moradi and François-Xavier Standaert. Moments-correlating DPA. In Begül Bilgin, Svetla Nikova, and Vincent Rijmen, editors, *Proceedings of the ACM Workshop on Theory of Implementation Security, TIS@CCS 2016 Vienna, Austria, October, 2016*, pages 5–15. ACM, 2016.

[NRR06] Svetla Nikova, Christian Rechberger, and Vincent Rijmen. Threshold implementations against side-channel attacks and glitches. In Peng Ning, Sihan Qing, and Ninghui Li, editors, *Information and Communications Security, 8th International Conference, ICICS 2006, Raleigh, NC, USA, December 4-7, 2006, Proceedings*, volume 4307 of *Lecture Notes in Computer Science*, pages 529–545. Springer, 2006.

[NYH13] Nianhao Zhu, Yujie Zhou, and Hongming Liu. Counteracting leakage power analysis attack using random ring oscillators. In *Conference on Sensor Network Security Technology and Privacy Communication System*, pages 74–77, 2013.

[PMK+11] Axel Poschmann, Amir Moradi, Khoongming Khoo, Chu-Wee Lim, Huaxiong Wang, and San Ling. Side-channel resistant crypto for less than 2, 300 GE. *J. Cryptol.*, 24(2):322–345, 2011.

[PR16] C. Padmini and J. V. R. Ravindra. Calpan: Countermeasure against leakage power analysis attack by normalized ddpl. In *ICCPCT*, pages 1–7, 2016.

[PSKM15] Santos Merino Del Pozo, François-Xavier Standaert, Dina Kamel, and Amir Moradi. Side-channel attacks from static power: when should we care? In Wolfgang Nebel and David Atienza, editors, *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition, DATE 2015, Grenoble, France, March 9-13, 2015*, pages 145–150. ACM, 2015.

[RBN+15] Oscar Reparaz, Begül Bilgin, Svetla Nikova, Benedikt Gierlichs, and Ingrid Verbauwhede. Consolidating masking schemes. In Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 764–783. Springer, 2015.

[RPD09] Matthieu Rivain, Emmanuel Prouff, and Julien Doget. Higher-order masking and shuffling for software implementations of block ciphers. In Christophe Clavier and Kris Gaj, editors, *Cryptographic Hardware and Embedded Systems - CHES 2009, 11th International Workshop, Lausanne, Switzerland, September*

6-9, 2009, Proceedings, volume 5747 of *Lecture Notes in Computer Science*, pages 171–188. Springer, 2009.

[SM15] Tobias Schneider and Amir Moradi. Leakage assessment methodology - A clear roadmap for side-channel evaluations. In Tim Güneysu and Helena Handschuh, editors, *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings*, volume 9293 of *Lecture Notes in Computer Science*, pages 495–513. Springer, 2015.

[SM21] Aein Rezaei Shahmirzadi and Amir Moradi. Re-consolidating first-order masking schemes nullifying fresh randomness. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2021(1):305–342, 2021.

[VMKS12] Nicolas Veyrat-Charvillon, Marcel Medwed, Stéphanie Kerckhof, and François-Xavier Standaert. Shuffling against side-channel attacks: A comprehensive study with cautionary note. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 740–757. Springer, 2012.

[YK17a] Weize Yu and Selçuk Köse. False key-controlled aggressive voltage scaling: A countermeasure against LPA attacks. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 36(12):2149–2153, 2017.

[YK17b] Weize Yu and Selçuk Köse. Security-adaptive voltage conversion as a lightweight countermeasure against LPA attacks. *IEEE Trans. on VLSI*, 25(7):2183–2187, 2017.

[YW18] Weize Yu and Yiming Wen. Leakage power analysis (LPA) attack in breakdown mode and countermeasure. In *SOCC*, pages 102–105, 2018.

[ZZL13] Nian-Hao Zhu, Yu-Jie Zhou, and Hong-Ming Liu. Employing symmetric dual-rail logic to thwart LPA attack. *IEEE Embed. Syst. Lett.*, 5(4):61–64, 2013.

[ZZL14] Nian-hao Zhu, Yu-jie Zhou, and Hong-ming Liu. A standard cell-based leakage power analysis attack countermeasure using symmetric dual-rail logic. *Journal of Shanghai Jiaotong University (Science)*, 19(2):169–172, 2014.