Beware Your Standard Cells! On Their Role in Static Power Side-Channel Attacks

Jitendra Bhandari, Likhitha Mankali, Mohammed Nabeel, Ozgur Sinanoglu, *Senior Member, IEEE*, Ramesh Karri, *Fellow, IEEE*, and Johann Knechtel, *Member, IEEE*

***Abstract*—Static or leakage power, which is especially prominent in advanced technology nodes, enables so-called static power side-channel attacks (S-PSCA). While countermeasures exist, they often incur considerable overheads. Besides, hardware Trojans represent another threat. Although the interplay between static power, down-scaling of technology nodes, and the vulnerability to S-PSCA is already established, an important detail was not covered yet: the role of the components at the heart of this sensitive interplay, the standard cells. Here, we study this intricate relationship for two commercial 28nm and 65nm technologies, using a commercial-grade IC design setup, and under realistic PPA objectives. Specifically, we study how threshold-voltage (VT) tuning of standard cells impacts the resilience of representative AES and PRESENT cipher hardware, including versions with established countermeasures. Our proposed CAD framework enables a security-vs-PPA-aware design-space exploration. Contrary to the belief that high-performance designs are generally more vulnerable to S-PSCA, we find that timing constraints and the distribution of different VT cells are more pivotal factors. Furthermore, we discover that attackers can deploy highly effective and stealthy S-PSCA-based Trojans, all without any gate overheads or any timing violations.**

***Index Terms*—Hardware Security, CAD**

I. Introduction

As technology in general advances, there is significant innovation in microelectronics to meet arising demands. Modern technologies for integrated circuits (ICs) manufacturing, also referred to as technology nodes, offer a large range of so-called standard cells which are providing Boolean algebra as well as memories. Importantly, there are multiple versions of the same standard cells but with different profiles for power consumption, performance, and area (PPA). PPA requirements vary across different IC applications, hence such different profiles are are needed. High-end applications like GPUs prioritize performance and favor fast cells for timing closure, despite power overheads, due to their design complexity and long exploration runtimes. On the other hand, embedded/edge devices with power constraints prefer performance trade-offs to save power, supported by lower design complexity that eliminates the need for fast cells as a timing fix. Different PPA profiles are enabled by transistor-level tuning, including but not limited to tuning of the threshold voltage (VT).

Side-channel attacks represent a significant threat for the security of ICs, even if their underlying logic is cryptographically robust. Prior research has demonstrated that hardware can leak information through various side-channels, such as electromagnetic, timing, power, and others [1]–[3].

Power side-channel attacks (PSCA) in particular have been extensively studied by the community [4]. Such attacks can be conducted in two different ways, namely by either focusing on dynamic power consumption while the IC is running or by focusing on static power consumption while the IC is halted. The latter, static power side-channel attacks (S-PSCA), are especially concerning for modern technology nodes [5].

Hardware Trojans represent a substantial security risk. Trojans are malicious alterations to ICs and have two parts: a trigger and a payload. Under rare and carefully crafted conditions, the trigger activates the payload, which then executes malicious actions on parts of the IC. Such actions can cause system failures or leak sensitive information.

Hardware Trojans have attracted attention throughout decades [6]; this has only become more pronounced due to the outsourced supply chain of ICs. Since Trojan modifications can occur at any early point in the supply chain (i.e., design and manufacturing), detecting hardware Trojans before deployment in the field is a challenge for secure and trustworthy ICs. This holds true despite that outsourced assembly and test facilities (OSATs) verify the proper IC functionality, as the trigger condition is unknown and often based on specific, rare conditions, which are unlikely covered during regular testing.

Scope and Contributions: Here, we study the role of standard cells, in particular their VT level and timing constraints, on S-PSCA from both attack and defense perspectives.

For example, we propose a simple, side-channel-based Trojan that makes it easy for attackers to extract the secret key of crypto cores through S-PSCA. The Trojan is created by replacing certain register cells with functionally equivalent cells that have a low/ultra-low threshold-voltage profile (LVT/ULVT). While these cells switch faster, they leak higher currents, making them vulnerable to S-PSCA.[1] This Trojan is practical, stealthy, and zero cost.

It is important to emphasize that our work is not only focused on such Trojan design, but rather on the critical role of different VT cells for an IC's vulnerability to S-PSCA.

This work was supported in part by the NYU Center for Cybersecurity (CCS) and the NYUAD CCS.

J. Bhandari, L. Mankali, and R. Karri are with New York University, New York City, NY, 11201 USA. E-mail: {jb7410, lm4344, rkarri}@nyu.edu

J. Knechtel, M. Nabeel, and O. Sinanoglu are with New York University Abu Dhabi, UAE. E-mail: {johann, mtn2, ozgursin}@nyu.edu

[1] S-PSCA do not benefit per se from larger static power. As we show in this study, resilience (or lack thereof) is a more intricate interplay that is based on an IC's ratios for different VT cells, the resulting interspersion of power profiles, and the timing constraints.

Toward that end, we develop a security-focused design-space exploration framework using commercial CAD tools,

In short, our contributions are threefold as follows:

1) We propose a security-focused design-space exploration framework that utilizes commercial CAD tools.[2] This framework is essential for design-space exploration from both offense and defense perspectives.
2) We put forward a straightforward, highly effective concept for zero-gate Trojans. This has practical applications in facilitating S-PSCA.
3) We analyze the security-versus-PPA design-space across multiple technology nodes with all their VT cell options.

Section II provides an overview of fundamental concepts and inspirational aspects for this study, whereas Section III offers a review of related works. Our methodology is described in detail in Section IV, and Section V presents our experimental studies. Section VI offers our conclusions and perspectives.

II. Background

A. Power Side-Channel Attacks

Power side-channel attacks exploit variations in a device's power consumption to extract sensitive data like cryptographic keys. By analyzing this consumption and understanding the device's operations, attackers can infer the internal workings and associate actual power use with possible secret data profiles. This method is well-researched in cryptographic hardware implementation due to its potential security vulnerabilities [4].

There are various power analysis attacks, such as simple power analysis (SPA), differential power analysis (DPA), and correlation power analysis (CPA) [7]. SPA profiles power consumption during crypto operations, while DPA compares power consumption between similar operations to derive a secret key. CPA uses the Pearson correlation coefficient to relate predicted and actual power profiles.

Dynamic power side-channel attacks (D-PSCA) extract sensitive information by monitoring and analyzing power consumption during a device's operation. Static power side-channel attacks, S-PSCA, analyze power usage when a device is at a halt or "idle" state, thus removing the requirement of the device being active, only depending on the previously computed values. D-PSCA demands precise timing, whereas S-PSCA needs sophisticated equipment but no timing synchronization, just a halted clock. As static/leakage power becomes more significant with advanced technology nodes, the importance of S-PSCA grows [8], [9].

Designers counteract attacks by integrating masking [10], shuffling, and balancing [11], [12] into IC design. These measures complicate extraction of secret keys and reduce power consumption variations during cryptographic operations, making it harder to distinguish power traces. While studies attempt to minimize the leakage of information through the power side-channel, doing so incurs overhead. However, this may not always be practical, particularly when the cost of silicon per mm^2 is high. It is necessary to consider the trade-offs between mitigation and PPA overhead during design time.

TABLE I
Variation of static power of a DFF in two different commercial nodes, reported in nW. The power depends on both the VT cells as well as input and output data.

CLK	D	Q	28nm Node			65nm Node		
			LVT	RVT	HVT	LVT	RVT	HVT
0	0	0	101.6	8.1	0.9	78.1	23.0	16.3
0	0	1	171.3	12.8	1.2	118.4	26.1	15.5
0	1	0	161.6	11.9	1.1	106.9	29.5	20.6
0	1	1	135.0	9.9	1.1	113.8	28.3	18.6
1	0	0	104.8	8.2	0.9	86.6	24.1	16.7
1	0	1	117.1	9.1	0.9	136.2	29.2	17.5
1	1	0	142.0	10.5	1.0	88.4	25.9	18.1
1	1	1	109.4	8.3	0.9	103.8	26.6	17.8

B. Hardware Trojans

Hardware Trojans have been studied extensively in the research community to be aware of any unexpected behavior that can occur due to intentional, malicious modifications in the hardware. Such modifications can lead to various sorts of threats like denial of service (DoS), system failure, unwanted privileged access, et cetera [6], [13]. This sort of malicious modification can be realized by some untrusted entity at any stage of the design and manufacturing cycle, which makes it difficult to avoid with the current distributed ICs supply chain, where IPs from the design house travel all the way offshore for fabrication. Naturally, such threats can lead to some serious incidents in critical missions where the reliability of the underlying IC is of utmost importance.

Hardware Trojans can be difficult to detect and remove because they can be designed to activate only under specific conditions, such as a particular input signal or after a certain period of time. It has been shown that, to make some Trojans avoid conventional testing, it requires some rare trigger conditions to get activated [14]. Since such malicious modifications will be done on top of the baseline design, there are limitations for placement and routing; Trojans should require as few gates as possible and occupy as small an area as possible.

C. Leakage Power for Modern Technologies

Commercial technology nodes offer diverse standard cell versions for different PPA demands. These cells have unique physical properties, like area, power consumption, and propagation time, facilitating optimization for design constraints. The inclusion of low and ultra-low VT cells enhances performance, especially for time-constrained paths.

Although faster, these LVT/ULVT cells leak more static power than standard cells (Table I). For the 28nm node, the increase in leakage power when going from HVT to RVT as well as when going from RVT to LVT cells for a D-flip-flop (DFF) is a factor of around 10x, and in total (i.e., when going from HVT to LVT) it is a factor of around 120x. For the 65nm node, the increase is much less pronounced, with a factor of around 1.5x when going from HVT to RVT, but with a factor of still around 4x when going from RVT to the fastest LVT cell, and a corresponding factor of around 6x in total.

[2] Our intention is to make this framework publicly available, after omitting technology-specific configurations, as details of these libraries are confidential.

D. Design Automation and Security as New Objective

CAD tools have evolved to be more complex and adaptable, and are guided by designers to optimize based on past designs. These tools use heuristics and user constraints to generate an optimized netlist meeting the user's specifications. As these constraints traditionally only target PPA goals, and there are no clear standards for formulating other types of constraints, security concerns are not considered at this stage.

Accordingly, there are few if any commercial settings where one considers security first hand – this renders IC layouts susceptible to various threats that can be exploited in the field, like PSCA. Prior research work advocated considering security objectives during the design phase, e.g., to protect against Trojans and other threats [15]–[22]. However, as indicated, the important role of standard cells has not been studied thoroughly yet in context of PSCA.

III. Related Works

[23] showed, for the first time, the potential of S-PSCA as a security threat. [9] have conducted one of the first practical experiments for S-PSCA using FPGAs. [8] highlighted the importance of leakage power and its effect on the PSC especially for more advanced nodes. [24] experimentally studied the role of various measurement factors on the success of S-PSCA. [25] have shown the important effect of aging on smaller technology nodes, further compromising the security of modern devices under S-PSCA. [26] conducted a multivariate analysis on the S-PSC. [5] studied both the static and dynamic PSC for the 65nm node, where they have shown that S-PSC can undermine protection efforts even for this old node.

[12] studied various countermeasures against S-PSCA, e.g., balancing logic, which can come at considerable overheads. Their study is based on an 28nm IC. Furthermore, they have indicated the important role of different VT cells. [27] proposed standard cell delay-based dual-rail pre-charge logic (SC-DDPL) as specific countermeasures against S-PSC, where NAND gates are used to implement every other logic stage, thus increasing the symmetry in the design and the resulting power profiles. This approach has the shortcoming/limitation of being not compatible with commercial CAD tool optimization flows. [28] proposed a CAD framework aiming to reduce PSC vulnerabilities in a design by assigning related scores to different parts and iteratively optimizing the design. A limitation of that work is that it assumes timing slacks are available for any security-centric optimization – this is not realistic in real applications where designs are pushed to meet performance requirements. Furthermore, they do not conduct actual S-PSCA evaluations.

[29] studied Trojan-based PSCA and countermeasures. [30], [31] show the feasibility of Trojan-based PSCA on FPGAs. [30] proposed a masking scheme as protection which incurs overheads. [32] proposed a CAD framework for the insertion of PSC-based Trojans in the late design stages. [33] proposed design procedures to reduce overhead and increase stealth of Trojans against PSC-based detection. [34] proposed a methodology to implement zero-overhead Trojans. Although they have indicated on PSCA, they lack a clear attack evaluation. Furthermore, the consider only non-crypto circuitry for benchmarks, which is also not suitable in the context of PSCA.

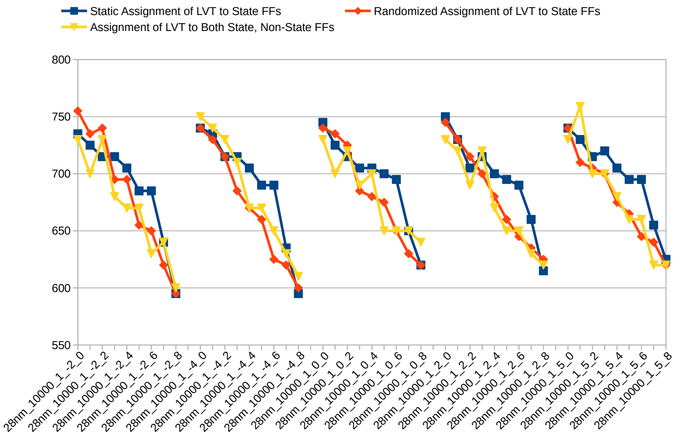

Fig. 1. Number of traces until key disclosure (NTD) for different VT settings, for a commercial 28nm node, with the baseline timing constraint set to 1ns.

IV. Methodology

A. Exploratory Study

To commence, we conducted an empirical investigation that helped us determine the general role of various VT cells for S-PSCA. This study is based on a commercial 28nm technology node (the same outlined in Table I), and it follows our security-aware design-space exploration framework. An overview, as well as more details for the framework, are provided in Sec. IV-C and IV-E, respectively.

We started with a baseline implementation of a regular AES core, with timing constraints set to 1ns, which would fulfill the most basic performance requirements while using only HVT cells. Accordingly, the design got optimized for area and power but not for performance; indeed, an inspection of the netlist confirms that no LVT cells are instantiated.

Next, we revise the netlist as follows, which we also refer to as scenario *Randomized Assignment of LVT to State FFs*. For the *state_reg* registers – the registers holding the state/intermediate texts that a classical PSCA is aiming at – we stepwise replace more and more HVT with LVT cells. Since registers are grouped by bytes in the hardware implementation (as well as for the modeling part in the S-PSCA), we replace HVT FFs with LVT counterparts in the range of 0–8 registers per byte. For any given number of registers to replace, for each byte, we randomly select the actual FFs within each byte, i.e., we select here in no particular order of bits.

As expected (from Table I), such substitutions impact the static power profile of the design. This directly impacts the prospects for S-PSCA, as demonstrated in Fig. 1: the number of traces required until disclosure (NTD) of the secret key (y-axis) decreases consistently with an increase of LVT registers employed in the registers (x-axis; usage of LVT registers increases, within each group of curves, from left to right).

Next, we explain the different scenarios in Fig. 1.[3]

1) *Randomized Assignment of LVT to State FFs:* As mentioned, this represents the baseline implementation where HVT registers are stepwise and randomly replaced with LVT counterparts. We find a consistent trend as in more LVT registers are employed, fewer traces are required until disclosure.
2) *Static Assignment of LVT to State FFs:* Here we replace FFs in a systematic way, namely in order of bits. For 3 HVT registers to be replaced by LVT counterparts, we replace bits 0, 1, and 2 for all the 16 state bytes. This scenario is important to understand the impact of bit-level position of LVT cells in the state bytes.
3) *Assignment of LVT to Both State, Non-State FFs:* Here we are modifying state registers as well as other, non-state registers. We replace the same number of cells for both state and non-state registers, and we follow the above strategies of static assignment for state registers versus randomized assignment across all other registers. This scenario is important to understand the role of LVT cells in state versus non-state registers, if any.

Across all three scenarios, we observe that neither the order of HVT versus LVT cells in state registers nor HVT versus LVT cells for state registers versus other/non-state registers impacts/undermines the general trend. From the one corner case of 0 LVT cells per byte to the other, 8 LVT cells per byte, the reduction of traces required until disclosure is around 20%. Overall, replacing HVT registers with LVT registers within state bytes undermines the IC's resilience against S-PSCA.

B. Scope and Threat Model

Different VT cells can significantly impact the S-PSC. Notably, the replacement/modification of HVT cells with LVT cells for a few selected gates, e.g., the crypto-core's state registers, induces a significant reduction in resilience. Naturally, this issue can go two ways: while a cautious designer could use it to evaluate and enhance the resilience of IC designs in a security-aware framework, a malicious entity in the design/manufacturing process could exploit it to create a PSC Trojan, given they have similar tools.

We assume classical threat models for PSCA and Trojans as follows. For PSCA, adversaries can obtain power measurements through physical access to the IC.[4] We further assume that attackers can observe – but not control – the externally accessible data – but not any internal data. Specifically, they can only observe the cipher-texts. For Trojans, the design can be compromised at any stage in the design and manufacturing cycle of ICs. Such Trojan represent a stealthy attack, as there is zero impact on area, no negative impact on timing, and only marginal impact on static power. See Sec. IV-D for more details on the Trojan design and threat modeling.

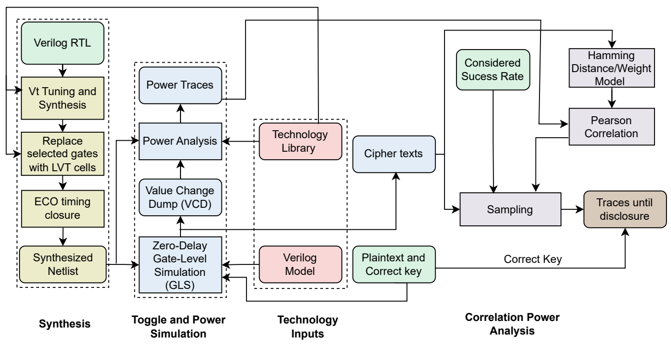

Fig. 2. The proposed security-aware design-space exploration framework.

Furthermore, we assume a threat model of collusion: this type of PSC-based Trojan assumes an attacker in the field, who is significantly supported by the reduced NTD. There are many cases where an attacker would benefit from such inherently more leaky power side-channel.[5]

From a defense viewpoint, the challenge lies in avoiding this increased leakage while adhering to strict PPA budgets. A straightforward avoidance of LVT cells is impractical for high-performance ICs with tight timing constraints. Thus, designers must carefully evaluate the susceptibility of the LVT cell usage scenario to PSCA.

C. Security-Aware Design-Space Exploration

1) Overview: Our security-aware design-space exploration framework Fig. 2 inputs the register-transfer level (RTL) of the design and different VT standard-cell libraries. It outputs the number of power traces until the secret key is disclosed. The more traces, the less susceptible the IC is to PSCA.

2) Research Questions: Our framework helps a designer check for possible vulnerabilities in their design against PSCA early on so that they can make necessary judgments to mitigate the vulnerability. However, doing so is more difficult than the attacker model of simply swapping selected cells to leak more information (Sec. IV-D). Because, as a defender, we have to look at various parts of the design and search for a relatively good solution with little impact on PPA. This challenge raises some questions in the mind of the designer:

1) How to make the design more resilient to power side-channel attacks without undermining PPA optimization?

[3] The labels along the x-axis of Fig. 1 are to be read as follows. First, there is the technology node (*28nm*), followed by the number of total traces available to the S-PSCA (*10000*), followed by the identifier of the secret key (*1*), followed by a variation of the baseline timing constraint in percentage (e.g., *-2*), and finally followed by the number of LVT registers employed per byte (e.g., *2*). Note that, for better readability, we have grouped all cases according to the varying timing constraints (ranging from -2% to +5%). Also note that, for a fair comparison, we employ the same random but fixed key across all cases, as well as the same random but fixed set of ciphertexts. More details for the S-PSCA experimentation are given in Sec. IV-E and V-A.

[4] Note that, in this paper, we do not conduct actual measurements for our experimental investigation. Still, as we consider commercial-grade and highly accurate technology libraries, our simulation-based study is accurate as well. Importantly, all findings can be considered conservative from the security point-of-view, i.e., as most powerful, best-case scenarios for adversaries, without any detrimental impacts from real-world measurement noises.

[5] First, attackers may have only limited access or could measure only over a limited number of traces before risking detection. Second, for the same number of traces collected, attackers would obtain a dataset with an inherently better signal-to-noise ratio which could help them, e.g., with a higher-order correlation test. Third, security protocols implemented at the system level, e.g., key update/replacement after so many encryption/decryption rounds, would possibly become easy to bypass for such an attacker.

2) Which part of the design plays a more dominant role when trying to reduce the leaking information through the power side-channel?
3) What if my security-aware design modifications lead to some violations of the previous optimization efforts performed by the tools?

These questions are all valid when it comes to security along with conventional PPA targets. This is not as "easy" to achieve as utilizing well-established heuristics like we do to meet our PPA targets – it rather requires a thorough understanding of different corners in the design space on both PPA as well as security, e.g., different usage profiles of VT cells on PPA as well as PSC information leakage.

To tackle these questions and resulting new challenges, we argue the following. Initially, the simplest solution to Questions 1 and 2 is to limit or disable the uses of LVT cells for state registers – the key finding from the exploratory study in Sec. IV-A would suggest this. However, doing so is not viable under aggressive timing constraints. Consequently, more sophisticated approaches are required, such as 1) optimizing timing closure in other design parts, 2) re-evaluating the role of LVT cells in state versus non-state registers, and 3) establishing design guidelines for optimal PPA and security trade-offs. These challenges necessitate comprehensive empirical studies using the proposed framework, which we present in Sec. V.

Related to the points just raised above, as well as to answer Question 3, we need to understand that any post-synthesis modification of VT cells can result in violations. More specifically, swapping faster (but more leaky) LVT cells with slower (but less leaky) RVT or HVT cells may well result in timing issues for the affected paths. Such timing violations would adversely affect the functionality of the design and cause erroneous behavior – they must be addressed also in a security-aware design-space exploration campaign, and we provide more details in Sec. IV-E.

D. Trojan Design

As already indicated, the proposed Trojan is a simple, yet highly effective, type of Trojan that aids S-PSCA in the field, by increasing the related information leakage. The Trojan is easy to realize on top of any design under attack, simply by replacing some instances of standard cells with other cells of the very same functionality but with low or, if available, even ultra-low VT profile, abbreviated by (U)LVT.

This type of Trojan is stealthy and practical as follows.

1) (U)LVT cells are faster than regular cells, implying that:
    a) An attacker can implement such Trojan (i.e., replace some RVT or HVT cells of interest with (U)LVT cells) right away, without any other changes needed in the design, and without risking any timing violations.
    b) Detecting such Trojan is impossible during functional testing.
    c) Detecting such Trojan is difficult even for parametric testing (which is costly and applied only selectively in the first place). For example, locally improved/faster timing, as induced by the Trojan, may well become masked by other slower cells that intersect along common timing paths.[6]
2) (U)LVT cells have the very same footprint as other cells, implying that:
    a) Such Trojan remains obscured for reverse engineering and simple optical inspection;
    b) Such Trojan is easy to integrate by any adversary within design and manufacturing entities.

We like to note that, if additional tests (i.e., beyond the official post-manufacturing testing) involving an actual S-PSCA setup would be done by designers, distributors, or end-users (DDEs), those parties would have some leverage for Trojan detection. However, even if DDEs do detect some significantly lower NTD than what they would have expected from the original design's specification, it will be difficult – if possible at all – to claim in an assertive manner that this is due to such S-PSCA-based Trojan introduced by the foundry. There are various considerations here as to why. First, since the concerned ICs are already handed over, liability would have been waived off from the foundry. Second, the foundry could make many arguments against that claim by DDEs, e.g., design files had been tampered by some malicious third party happened before reaching the foundry, post-silicon VT tuning was applied [35], or issues with materials, equipment, and/or exacerbated process variations occurred.[7] In any case, while the DDEs would be able to make an educated guess as to whether or not they should release the concerned ICs for some security-sensitive applications, there is still financial loss and other implications incurred on them, e.g., delays for any products that planned to use these ICs. Besides colluding with S-PSCA adversaries in the field, these implications may have been part of the foundry's malicious motivation.

To realize the Trojan, we find that attackers would want to specifically replace the state register cells with such (U)LVT cells. For the actual Trojan implementation, i.e., the number and location of the state register that an attacker would want to replace, this depends on the overall design characteristics, especially on the timing paths. Our proposed framework for security-aware design-space exploration would also help the attacker tackle with these very questions.

E. Implementation Details

Recall the overview of our methodology as outlined in Fig. 2. Next, we provide some more implementation details, followed by the actual setup for experiments in Sec. V-A.

1) Simulation-Based Power Analysis: First, the cipher RTL is synthesized using the technology libraries of choice, considering all VT cell options. Then, using a testbench, the functionality of the gate-level, post-synthesis netlist is verified. After the user specifies a set of plain-texts and a key (or set

[6] An empirical study on this claim is left for future work. Besides timing, another relevant domain for parametric testing would naturally be static power. However, even there it remains to be seen whether the observed profile can be clearly identified as malicious. If only few more (U)LVT cells are introduced, among millions of gates in modern SoCs, their contribution to the overall static power profile will become negligible and hardly detectable.

[7] The latter argument is rather hypothetical though, as it would be detrimental for the foundry's reputation and business.

of keys), the testbench generates the corresponding set(s) of cipher-texts required for verification. During these gate-level simulations, a Value Change Dump (VCD) file is generated; it captures the switching activity of every gate/node within the netlist in a user-defined time resolution (e.g., 1 ps). Next, the VCD file is used for power simulation, along with the post-synthesis netlist and libraries of choice.

The power simulation tool calculates the power consumption of each cell by adding 1) static/leakage power, 2) internal power (from input-pin switching), and 3) switching power (from output-pin switching). The library power characterization stores data for leakage and internal power of the cells, and the tool uses the VCD's input/output state information to calculate the total leakage power of the design. Zero-delay simulations are used instead of full-timing simulations as our interest lies in capturing static power in a specific clock cycle, not average leakage power. Such static power value will be the same regardless if it is a full-timing simulation using a standard delay format (SDF) or a zero-delay simulation. We then sequentially obtain power traces while processing cipher-texts for the given secret key(s), focusing on the last-round operations [7]. This procedure is repeated separately for each scenario and technology setup considered in this work.

2) Correlation Power Analysis Attack, CPA: This attack uses the Pearson correlation coefficient (PCC) to measure the relationship between observed power consumption during cryptographic tasks and the secret data [7]. This involves comparing real power consumption and predicted power profiles for different key values across a range of observations. Then, the key is byte-wise inferred from the most promising candidates, i.e., those with the highest PCC values.

Note that predicted power profiles are derived from a power model of choice [7]. For S-PSCA, typically either the Hamming weight (HW) for the cipher-texts or the Hamming distance (HD) for the cipher-texts to the prior last-round operation are chosen, depending on the power profiles of the underlying technology nodes. For the nodes considered in our study, we observe a considerable dependency of the FFs' static power on both the input and output data (Table I); thus, the HD model seems more promising. Indeed, while we did run all experiments for both HW and HD models, and even investigated the prospects of technology-specific models that capture the exact static-power values, we note that the best results are achieved with the HD model.

In our detailed study on the role of VT cells, we conduct sampling campaigns for each scenario separately. We provide increasing numbers of random traces to the CPA and calculate the success rate over several CPA trials. We stop when we achieve a desired confidence level, such as a 90% success rate, and report the corresponding number of traces as NTD. To improve computational efficiency, we first conduct coarse sampling with fewer trials and a larger step size, which provides a reasonable starting point. We then conduct thorough sampling with more trials and smaller step-size, starting from the point identified in the coarse sampling.

3) Engineering Change Order (ECO) Modifications, Timing Closure: We use commercial synthesis tools to create baseline netlists for various scenarios. Then, through *Tcl* scripts, we make further ECO modifications on these netlists, specifically replacing gates of interest – the registers holding the intermediate cipher-texts – with different VT cells as outlined in Sec. IV-A and Sec. V. These ECO modifications may lead to violations, particularly timing violations. To handle this, we implement the following procedures (Algorithm 1).

Algorithm 1: ECO VT Tuning with Timing Closure

```
Input: List of failing paths $FT$; VT constraints, i.e., number of
        LVT/RVT/HVT cells allowed
Output: Report $file$ for further manual inspection
Function CorrectTiming($path$):
    $startPoint \leftarrow path.startpoint()$; // Start point of path
    $endPoint \leftarrow path.endpoint()$; // End point of the path
    $gates \leftarrow path.gates()$;    // List of gates in the path
    $gates \leftarrow$ SORT($gates$); // Sort the gates in decreasing
      order of delay
    foreach $i \in gates$ do
        $x \leftarrow$ GETALTERNATIVECELLS($i$);            // List of
          alternative cells available, considering
          driver strength and VT cells
        foreach $j \in x$ do
            if $j.delay() < i.delay()$ and VT constraints still met then
                REPLACE $i$ by $j$ ;  // Replace old one with
                  new cell
            else
                KEEP $i$ ;
    $STATUS \leftarrow$ REPORTTIMING($startPoint, endPoint$) ;
    return $STATUS$;                    // Return the status
$TEMP \leftarrow$ COPY($FT$);
$UnresolvedPath \leftarrow \emptyset$;
foreach $path \in TEMP$ do
    $STATUS \leftarrow CorrectTiming(path)$;
    if $STATUS == PASS$ then
        COMMIT;                    // Commit to the changes
    else
        $UnresolvedPath.append(path)$
$file \leftarrow \{UnresolvedPath\}$;     // List of paths for manual
  analysis
return $file$
```

Initially, we identify all failing paths and sort the gates on these paths in decreasing order of delay. Next, we explore gates with 1) greater drive strengths and/or 2) different VT cells to replace them. While Option 2) is based on the VT scenario under investigation and its constraints for the use of different VT cells, Option 1) is applied as needed. Next we re-evaluate the failing paths. If there are paths that fail to meet the timing, a manual analysis is necessary. Since many timing paths intersect, in some cases, revising other paths can help fix those violating paths. If failing paths remain, related timing constraints are infeasible, and the setup is revised.

V. Experimental Investigation

A. Setup

All experimentation (including the exploratory study in Sec. IV-A) are conducted on an *AMD EPYC 7542* server with 128 CPUS, 512KB caches, and 1TB RAM, operating with *Red Hat Enterprise Linux Server Release 7.9 (Maipo)*.

In our implementation of the CAD flow, we employ commercial tools as follows. We use *Synopsys VCS M-2017.03-SP1* for functional simulations at RTL and gate level, *Synopsys DC M-2016.12-SP2* for logic synthesis, and *Synopsys PrimeTime PX M-2017.06* for power simulations. For synthesis, we employ regular optimization settings. The proposed design-space exploration framework is implemented in *Tcl* scripts.

For AES, we leverage an in-house RTL working on 128-bit keys and 128-bit texts, using look-up tables for the S-Box. For PRESENT, we leverage an RTL working on 80-bit keys and 64-bit texts, obtained from [36]. The CPA code for AES is based on a *C++* release from [37], whereas the CPA code for PRESENT is implemented in *Python* in-house. For all CPA runs, we stepwise increase the number of available traces by 10. Furthermore, we employ coarse and thorough sampling over sets of available, for 64 and 640 trials, respectively. We report final results of t traces until key disclosure for a 90% success rate for thorough sampling. In other words, given t out of T traces, the CPA must succeed to infer all key bytes correctly for at least 576 out of 640 randomly selected subsets of t out of T traces. Note that T, the number of traces gathered in total, depends on the case study; for most studies, up to 10k traces are sufficient, whereas others require 50k, and some even more than 1M traces. For cases where CPA fails even for all T traces, we report PCC values for the best key candidates, averaged across all key bytes.

Finally, we employ two commercial libraries for a 65nm and a 28nm technology setup. For both cases, we consider their respective *TT* corners which are characterized for 25 degree Celsius and for 0.9V/1.0V for the 28nm/65nm node, respectively. We utilize VT cells as desired/appropriate for the different case studies.

B. Overview

We conduct an extensive set of case studies as follows.[8] Taking up the insights from the exploratory study in Sec. IV-A, Case Study 1 examines the role of HVT versus LVT cells for state registers in detail, i.e., across different timing constraints, technology nodes, and also considering the PPA impact. Case Study 2 examines the role of other, non-state registers with the same level of detail. Case Study 3 considers a wider range of VT cells, namely LVT, RVT, and HVT, for their security-versus-PPA design-space, all for the same level of detail. Case Study 4 explores whether the CPA benefits from technology-accurate power models over the commonly considered HD and HW models. Case Study 5 provides a comparison to *TrojanZero* [34], a prior work for zero-cost Trojans. Case Study 6 explores the security-versus-PPA design-space for the AES core protected with an S-PSCA countermeasure. Case Study 7 broadens our study to another crypto core, PRESENT. Case Study 8 covers the PRESENT core when protected with two different S-PSCA countermeasures.

C. Case Study 1: HVT versus LVT for State FFs

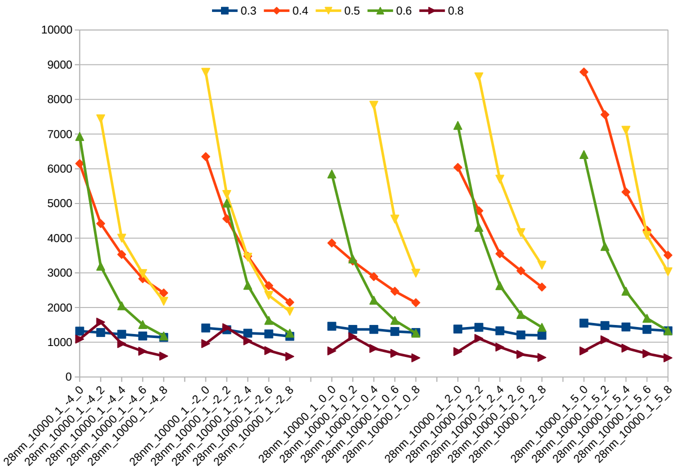

Fig. 3. NTD for AES core, in 28nm node. The legend refers to reference timing constraints in ns. For reading the naming scheme of different cases listed on the x-axis, please refer back to Sec. IV-A. Any missing data point indicates that CPA could not succeed with 10,000 traces.

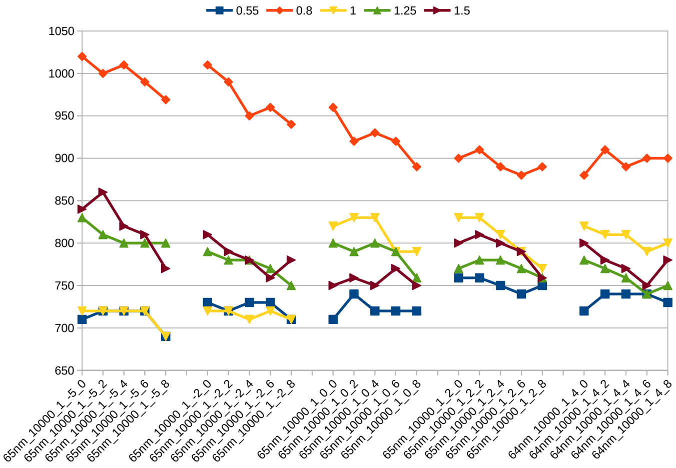

Fig. 4. NTD for AES core, in 65nm node. See also Fig. 3's caption.

First, we study the impact of different VT cells in general. As in the exploratory study (Sec. IV-A), we tune margins for timing constraints between -4% and +5%, while replacing varying numbers of HVT cells with LVT in the AES state registers. We consider different baseline timing constraints: {0.3, 0.4, 0.5, 0.6, 0.8}ns for 28nm vs. {0.55, 0.8, 1.0, 1.25, 1.5}ns for 65nm node. Some of the faster constraints are pushing the limits; timing closure can fail even when the majority of state registers are using LVT cells.

Security Analysis: The CPA results are provided in Figures 3 and 4, respectively, for the 28nm and the 65nm node.

The trend initially observed (Sec. IV-A) applies here as well: NTD reduces with an increase in the number of LVT cells in the state registers. However, for the 28nm node, the slope of the curves describing this correlation varies significantly; this is due the varied scales of timing constraints. This finding is interesting, as prior works anticipated that high-performance designs will be leakier in both terms of static power (which is correct) and information leakage (which is wrong).

Furthermore, NTD for the two corner cases in the 28nm node (i.e., 0.3ns versus 0.8ns) are both in lower resilience ranges, whereas the middle ranges for timing (i.e., 0.4–0.6ns) are much more varied and running in much higher resilience ranges. For the 65nm node, while the different timing constraints generally induce less strong variations, it still holds true here that some "middle range" for timing (i.e., 0.8ns) induces the largest resilience, whereas faster and slower

[8] Unless specified otherwise, all case studies are on a regular AES core.

TABLE II
PPA RESULTS FOR AES CORE, IN 28NM NODE, FOR TIMING CONSTRAINT OF 0.3NS, WITH HVT('H') AND LVT('L') CELLS USED. 'TIME' REFERS TO % VARIATION OF THE TIMING CONSTRAINT. LEFT 'LVT' COLUMN REFERS TO NUMBER OF LVT CELLS IN EACH BYTE OF THE REGISTERS HOLDING AES STATE. 'LVT' AND 'HVT' REFER TO RATIO OF ALL GATES IN THE DESIGN IMPLEMENTED BY THESE VT CELLS. 'STATE FFS' AND 'NON-STATE FFS' (NS FFS) REFER TO THE COUNTS OF VT CELLS IN THESE REGISTERS. 'WNS' DENOTES WORST NEGATIVE SLACK.

Time	L	Area	WNS	Power	L	H	State FFs		NS FFs	
		(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
-4	0	15534	-0.05	23.89	50	50	0	128	318	206
	2	15534	-0.04	23.91	51	49	32	96	318	206
	4	15534	-0.04	23.94	51	49	64	64	318	206
	6	15534	-0.03	23.96	52	48	96	32	318	206
	8	15534	-0.01	23.98	52	48	128	0	318	206
-2	0	15599	-0.05	22.25	43	57	0	128	277	247
	2	15599	-0.03	22.27	44	56	32	96	277	247
	4	15599	-0.03	22.29	44	56	64	64	277	247
	6	15599	-0.01	22.32	45	55	96	32	277	247
	8	15599	-0.01	22.34	45	54	128	0	277	247
0	0	14867	-0.01	21.52	43	57	0	128	273	251
	2	14867	-0.01	21.54	44	56	32	96	273	251
	4	14867	-0.01	21.57	44	56	64	64	273	251
	6	14867	0	21.59	45	55	96	32	273	251
	8	14867	0	21.61	45	55	128	0	273	251
2	0	15038	0	20.84	41	59	0	128	261	263
	2	15038	0	20.86	41	59	32	96	261	263
	4	15038	0	20.89	42	58	64	64	261	263
	6	15038	0	20.91	42	58	96	32	261	263
	8	15038	0	20.93	43	57	128	0	261	263
5	0	14097	0	18.73	39	61	0	128	299	225
	2	14097	0	18.75	40	60	32	96	299	225
	4	14097	0	18.77	40	60	64	64	299	225
	6	14097	0	18.80	41	59	96	32	299	225
	8	14097	0	18.82	41	59	128	0	299	225

TABLE III
PPA RESULTS FOR AES CORE, IN 28NM NODE, FOR TIMING CONSTRAINT OF 0.5NS. SEE ALSO TABLE II'S CAPTION.

Time	L	Area	WNS	Power	L	H	State FFs		NS FFs	
		(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
-4	0	13448	0	11.08	9	91	0	128	25	499
	2	13435	0	11.08	9	91	32	96	25	499
	4	13424	0	11.07	10	90	64	64	25	499
	6	13408	0	11.06	10	90	96	32	25	499
	8	13401	0	11.06	11	89	128	0	25	499
-2	0	13386	0	10.61	8	92	0	128	12	512
	2	13375	0	10.61	8	92	32	96	12	512
	4	13362	0	10.60	9	91	64	64	12	512
	6	13352	0	10.60	9	91	96	32	12	512
	8	13339	0	10.59	10	90	128	0	12	512
0	0	13230	0	10.42	8	92	0	128	10	514
	2	13221	0	10.42	8	92	32	96	10	514
	4	13214	0	10.43	9	91	64	64	10	514
	6	13206	0	10.43	9	91	96	32	10	514
	8	13199	0	10.44	10	90	128	0	10	514
2	0	13178	0	10.16	8	92	0	128	25	499
	2	13170	0	10.17	8	92	32	96	25	499
	4	13162	0	10.17	9	91	64	64	25	499
	6	13157	0	10.18	9	91	96	32	25	499
	8	13153	0	10.19	10	90	128	0	25	499
5	0	13129	0	9.85	6	94	0	128	8	516
	2	13120	0	9.84	6	94	32	96	8	516
	4	13116	0	9.85	7	93	64	64	8	516
	6	13109	0	9.86	7	93	96	32	8	516
	8	13101	0	9.86	8	92	128	0	8	516

TABLE IV
PPA RESULTS FOR AES CORE, IN 65NM NODE, FOR TIMING CONSTRAINT OF 0.55NS. SEE ALSO TABLE II'S CAPTION.

Time	L	Area	WNS	Power	L	H	State FFs		NS FFs	
		(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
-5	0	67678	-0.03	38.17	86	14	0	128	389	135
	2	67678	-0.03	38.17	87	13	32	96	389	135
	4	67678	-0.01	38.17	87	13	64	64	389	135
	6	67678	-0.01	38.17	87	13	96	32	389	135
	8	67678	-0.01	38.18	88	12	128	0	389	135
-2	0	64324	0	34.77	84	16	0	128	381	143
	2	64324	0	34.77	84	16	32	96	381	143
	4	64324	0	34.77	85	15	64	64	381	143
	6	64324	0	34.77	85	15	96	32	381	143
	8	64324	0	34.78	86	14	128	0	381	143
0	0	66453	0	35.35	86	14	0	128	367	157
	2	66453	0	35.35	86	14	32	96	367	157
	4	66453	0	35.36	86	14	64	64	367	157
	6	66453	0	35.36	87	13	96	32	367	157
	8	66453	0	35.36	87	13	128	0	367	157
2	0	66464	0	34.50	82	18	0	128	367	157
	2	66464	0	34.50	83	17	32	96	367	157
	4	66464	0	34.50	83	17	64	64	367	157
	6	66464	0	34.50	84	16	96	32	367	157
	8	66464	0	34.51	84	16	128	0	367	157
4	0	65420	0	33.07	84	16	0	128	368	156
	2	65420	0	33.07	84	16	32	96	368	156
	4	65420	0	33.07	84	16	64	64	368	156
	6	65420	0	33.08	85	15	96	32	368	156
	8	65420	0	33.08	85	15	128	0	368	156

TABLE V
PPA RESULTS FOR AES CORES, IN 65NM NODE, FOR TIMING CONSTRAINT OF 0.8NS. SEE ALSO TABLE II'S CAPTION.

Time	L	Area	WNS	Power	L	H	State FFs		NS FFs	
		(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
-5	0	56163	0	19.77	27	73	0	128	263	261
	2	56163	0	19.77	28	72	32	96	263	261
	4	56163	0	19.78	28	72	64	64	263	261
	6	56163	0	19.78	29	71	96	32	263	261
	8	56163	0	19.78	29	71	128	0	263	261
-2	0	55415	0	19.10	25	75	0	128	232	292
	2	55415	0	19.11	25	75	32	96	232	292
	4	55415	0	19.11	26	74	64	64	232	292
	6	55415	0	19.11	26	74	96	32	232	292
	8	55415	0	19.11	27	73	128	0	232	292
0	0	54241	0	18.29	23	77	0	128	224	300
	2	54241	0	18.29	24	76	32	96	224	300
	4	54241	0	18.30	24	76	64	64	224	300
	6	54241	0	18.30	25	75	96	32	224	300
	8	54241	0	18.30	25	75	128	0	224	300
2	0	54228	0	17.59	22	78	0	128	221	303
	2	54228	0	17.59	23	77	32	96	221	303
	4	54228	0	17.60	23	77	64	64	221	303
	6	54228	0	17.60	24	76	96	32	221	303
	8	54228	0	17.60	24	76	128	0	221	303
4	0	54064	0	17.45	21	79	0	128	232	292
	2	54064	0	17.45	22	78	32	96	232	292
	4	54064	0	17.45	22	78	64	64	232	292
	6	54064	0	17.45	23	77	96	32	232	292
	8	54064	0	17.46	23	77	128	0	232	292

constraints induce lower resilience ranges.

Therefore, we can conclude that the S-PSCA information leakage is not solely determined by the number of LVT cells, but rather by both the number of LVT cells as well as the timing constraint. In fact, the latter will dictate the distribution or ratio of LVT to HVT cells, which is an important factor of resilience. After reviewing the CPA results along with the design properties of all these different scenarios, we argue the following for an explanation of this observation.

First, for CPA in general, a set of power traces that exhibits some consistent patterns for power consumption is easier to correlate with the correct key. Second, such consistent power patterns are more likely for designs with greater homogeneity, i.e., when the usage/ratio of different VT cells is either quite balanced or almost exclusively one-sided/singular. For example for the 28nm node, while we observe a balance in LVT and HVT cells for the overall design with the 0.3ns constraint, for 0.8ns there is a clear dominance of HVT cells over LVT cells. In contrast, for constraints of 0.4–0.6ns, LVT cells are utilized to a somewhat larger degree, yet far from balanced with HVT cells. Third, these trends are less pronounced for the 65nm node, which is expected from the less varied static power profiles (Table I). In short, a small number of LVT cells results in diverse power profiles, which challenges the CPA.

Attacker's Perspective for Trojans: It applies here as well that more LVT cells are preferable. While resilience values are converging for the maximum number of LVT cells across all timing constraints, some nuances do remain. The underlying differences arise from the imbalance in LVT/HVT cells usage for the overall design, not only the state registers; thus, an attacker might further reduce this by assigning more and more LVT cells to other gates as well, although larger-scale modifications might eventually be easier detected post-silicon. Also, the differences across timing constraints are smaller than the reduction in NTD achievable if the maximum of LVT cells is used. In short, the Trojan threat is practical across wide timing ranges and is powerful when using LVT cells for all 8 bits in state-register bytes.

PPA Analysis: For the 28nm node, results are given in Tables II–III and for 65nm node in Tables IV–V.

Despite our efforts (Algorithm 1), we encountered cases where timing was not met, which challenges the assumption in prior art [28] that sufficient slack is available for any VT optimization. For high-performance design settings (0.3ns for 28nm, Table II; 0.55ns for 65nm, Table IV), timing closure was difficult for the lower ends of timing tuning, although we were able to get close, with only -0.01–0.05 WNS remaining.

This indicates that these particular constraints are pushing the respective technology and synthesis limits – this is on purpose, to properly study this part of the design space as well. Naturally, there is an increase in total power for these aggressive constraints; see Tables III and V. As these aggressive and power-hungry constraints also yield lower resilience, this part of the design space is unfavorable for PPA-and-security co-optimization.

D. Case Study 2: State versus Non-State FFs

To examine the role of non-state registers in more detail, here we sweep the overall number of LVT instances for those registers, for different sets of LVT cells for state registers, and for varying timing constraints.

Security Analysis: The CPA results for thorough sampling are provided in Figure 5 for the 28nm node. CPA results for the 65nm node follow similar trends, albeit less pronounced; these are not reported separately here.

The general trend remains valid. That is, both the number of LVT cells in state registers and the timing constraints dominate the resilience. This is true again especially for the "middle range" timing constraints. The number of LVT cells in non-state registers has negligible impact.

Attacker's Perspective for Trojans: It holds true again that more LVT cells are preferable, but differences across timing constraints are more pronounced now for the attacker's best-case of all state FFs using LVT cells.

PPA Analysis: For the 28nm node, PPA results are provided in Tables VI–VII. Findings here are similar to those for Case Study 1; more specifically, timing closure becomes more challenging for the most aggressive setting, and power cost is considerable for those aggressive constraints. PPA results for the 65nm node are similar here as well to those for Case Study 1 and are, thus, not reported separately.

TABLE VI

PPA RESULTS FOR AES CORE, IN 28NM NODE, WITH 3 LVT CELLS USED IN EACH BYTE OF THE STATE. 'TIME' REFERS TO THE TIMING CONSTRAINT. 'NON-STATE FF SETTING' (NS FFS SETTING) REFERS TO THE TOTAL NUMBER OF LVT CELLS USED FOR NON-STATE FFS. OTHER COLUMNS ARE THE SAME AS BEFORE.

Time	NS FFs	Area	WNS	Power	L	H	State FFs		NS FFs	
	Setting	(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
0.3	0	14867	-0.05	21.55	44	56	48	80	273	251
	60	14867	-0.01	21.60	45	55	48	80	333	191
	120	14867	-0.01	21.63	46	54	48	80	393	131
0.5	0	13211	0	10.41	8	92	48	80	10	514
	60	13211	0	10.44	9	91	48	80	70	454
	120	13211	0	10.47	10	90	48	80	130	394
0.75	0	12336	0	6.09	1	99	48	80	0	524
	60	12336	0	6.10	2	98	48	80	60	464
	120	12336	0	6.12	3	97	48	80	120	404
1	0	12085	0	4.29	1	99	48	80	0	524
	60	12085	0	4.32	2	98	48	80	60	464
	120	12085	0	4.33	3	97	48	80	120	404

TABLE VII

PPA RESULTS FOR AES CORE, IN 28NM NODE, WITH 8 LVT CELLS USED IN EACH BYTE OF THE STATE. SEE ALSO TABLE VI'S CAPTION.

Time	NS FFs	Area	WNS	Power	L	H	State FFs		NS FFs	
	Setting	(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
0.3	0	14867	-0.01	21.61	45	55	128	0	273	251
	60	14867	0	21.65	46	54	128	0	333	191
	120	14867	0	21.69	47	53	128	0	393	131
0.5	0	13199	0	10.44	10	90	128	0	10	514
	60	13199	0	10.47	11	89	128	0	70	454
	120	13199	0	10.49	12	88	128	0	130	394
0.75	0	12336	0	6.12	2	98	128	0	0	524
	60	12336	0	6.13	4	96	128	0	60	464
	120	12336	0	6.16	5	95	128	0	120	404
1	0	12085	0	4.32	2	98	128	0	0	524
	60	12085	0	4.34	4	96	128	0	60	464
	120	12085	0	4.36	5	95	128	0	120	404

E. Case Study 3: LVT, RVT, and HVT

We study a practical scenario where all types of VT cells are used.[9] Otherwise, setting is similar to the experiments above.

Security Analysis: CPA results are provided in Figures 6 and 7 for the 28nm and 65nm nodes, respectively. The general trend remains valid here as well. The variation in resilience becomes more pronounced for different timing constraints. For the 28nm node, the trends are more pronounced when locally sweeping/revising the aggressive 0.35ns constraint and the medium 0.75ns constraint. For the 65nm node, note the lower resilience for negative timing variations for lower and upper ranges of baseline timing constraints.

Such trends help designers. For example for the 28nm node, securing high-performance design options is promising compared to using LVT, HVT cells. The improvement in

[9] However, we use RVT cells only for all other parts of the design. For state registers, we employ the same procedure of sweeping the number of HVT cells to be replaced by LVT cells. This is important to better understand the role of other parts of the design, which will be impacted to a larger degree by the more varied VT cell options than only the state registers.

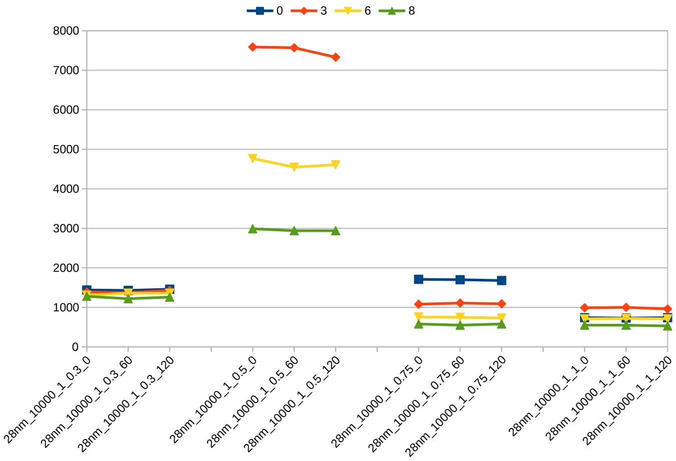

Fig. 5. NTD for AES core, in 28nm node. The legend refers to the number of LVT cells in state-register bytes. Cases listed on the x-axis differ here as follows in naming: the second-to-last element is timing constraints and the last element is the number of LVT cells in non-state registers. Missing data point indicates CPA fails after 10,000 traces.

TABLE VIII
PPA RESULTS FOR AES CORE, IN 28NM NODE, FOR TIMING CONSTRAINT OF 0.35NS, WITH HVT ('H'), RVT ('R'), AND LVT ('L') CELLS USED. SEE ALSO TABLE II'S CAPTION.

Time	L	Area	WNS	Power	L	R	H	State FFs			NS FFs		
		(μm^2)	(ns)	(mW)	(%)	(%)	(%)	L	R	H	L	R	H
-4	0	14529	-0.03	17.99	19	24	57	0	0	128	219	17	288
	2	14529	-0.03	18.01	20	24	57	32	0	96	219	17	288
	4	14529	-0.03	18.04	20	24	56	64	0	64	219	17	288
	6	14529	-0.03	18.06	21	24	56	96	0	32	219	17	288
	8	14529	-0.03	18.08	21	24	55	128	0	0	219	17	288
-2	0	13890	-0.02	17.03	18	26	55	0	0	128	209	26	289
	2	13890	-0.02	17.05	19	26	55	32	0	96	209	26	289
	4	13890	-0.02	17.07	19	26	54	64	0	64	209	26	289
	6	13890	-0.02	17.09	20	26	54	96	0	32	209	26	289
	8	13890	-0.02	17.11	20	26	53	128	0	0	209	26	289
0	0	14636	-0.01	16.79	13	24	63	0	0	128	179	48	297
	2	14636	-0.01	16.81	13	24	63	32	0	96	179	48	297
	4	14636	-0.01	16.83	14	24	62	64	0	64	179	48	297
	6	14636	-0.01	16.85	14	24	62	96	0	32	179	48	297
	8	14636	-0.01	16.87	15	24	61	128	0	0	179	48	297
2	0	14330	0	15.97	13	24	63	0	0	128	219	37	268
	2	14330	0	15.99	14	24	63	32	0	96	219	37	268
	4	14330	0	16.01	14	24	62	64	0	64	219	37	268
	6	14330	0	16.03	15	24	62	96	0	32	219	37	268
	8	14330	0	16.06	15	24	61	128	0	0	219	37	268
5	0	14560	0	15.65	10	24	65	0	0	128	111	75	338
	2	14558	0	15.67	11	24	65	32	0	96	111	75	338
	4	14558	0	15.69	11	24	64	64	0	64	111	75	338
	6	14558	0	15.71	12	24	64	96	0	32	111	75	338
	8	14558	0	15.73	12	24	63	128	0	0	111	75	338

resilience over the corresponding cases in Case Study 1 is around 3.5x (for 0.35ns timing constraint). For the 65nm node, for the most promising "mid range" of 0.8ns, the improvement over corresponding cases in Case Study 1 is around 2.8x. Timing constraints should be explored for resilience.

Since we did not, on purpose, use RVT cells for state FFs, these observations indicate that diverse options for all other gates play an important role. For example, for the 65nm node, there is a larger benefit now for positive timing variations. This is due to the flexibility of using RVT and HVT for relaxed timing. These observations do not contradict Case Study 2, but extend the related perspective: VT options for other gates can matter, but require more VT options.

TABLE IX
PPA RESULTS FOR AES CORE, IN 28NM NODE, FOR TIMING CONSTRAINT OF 0.5NS, WITH HVT ('H'), RVT ('R'), AND LVT ('L') CELLS USED. SEE ALSO TABLE II'S CAPTION.

Time	L	Area	WNS	Power	L	R	H	State FFs			NS FFs		
		(μm^2)	(ns)	(mW)	(%)	(%)	(%)	L	R	H	L	R	H
-4	0	13291	0	11.21	3	19	79	0	0	128	3	62	459
	2	13283	0	11.21	3	19	78	32	0	96	3	62	459
	4	13276	0	11.21	4	19	78	64	0	64	3	62	459
	6	13268	0	11.21	4	19	77	96	0	32	3	62	459
	8	13260	0	11.21	5	19	77	128	0	0	3	62	459
-2	0	13229	0	10.48	3	16	82	0	0	128	5	73	446
	2	13222	0	10.49	3	16	81	32	0	96	5	73	446
	4	13213	0	10.48	4	16	80	64	0	64	5	73	446
	6	13203	0	10.48	4	16	80	96	0	32	5	73	446
	8	13197	0	10.49	5	16	79	128	0	0	5	73	446
0	0	13155	0	9.99	2	14	84	0	0	128	3	68	453
	2	13155	0	10.01	3	14	83	32	0	96	3	68	453
	4	13155	0	10.03	3	14	83	64	0	64	3	68	453
	6	13155	0	10.04	4	14	82	96	0	32	3	68	453
	8	13155	0	10.06	4	14	82	128	0	0	3	68	453
2	0	13089	0	9.82	2	14	84	0	0	128	4	83	437
	2	13089	0	9.83	2	14	84	32	0	96	4	83	437
	4	13089	0	9.84	3	14	83	64	0	64	4	83	437
	6	13089	0	9.86	4	14	83	96	0	32	4	83	437
	8	13089	0	9.87	4	14	82	128	0	0	4	83	437
5	0	13082	0	9.60	1	12	87	0	0	128	0	32	492
	2	13082	0	9.62	2	12	86	32	0	96	0	32	492
	4	13082	0	9.62	3	12	86	64	0	64	0	32	492
	6	13082	0	9.63	3	12	85	96	0	32	0	32	492
	8	13082	0	9.64	4	12	85	128	0	0	0	32	492

TABLE X
PPA RESULTS FOR AES CORE, IN 65NM NODE, FOR TIMING CONSTRAINT OF 0.8NS, WITH HVT ('H'), RVT ('R'), AND LVT ('L') CELLS USED. SEE ALSO TABLE II'S CAPTION.

Time	L	Area	WNS	Power	L	R	H	State FFs			NS FFs		
		(μm^2)	(ns)	(mW)	(%)	(%)	(%)	L	R	H	L	R	H
-5	0	55109	0	20.05	21	30	50	0	0	128	257	24	243
	2	55109	0	20.05	21	30	49	32	0	96	257	24	243
	4	55109	0	20.05	22	30	49	64	0	64	257	24	243
	6	55109	0	20.05	22	30	48	96	0	32	257	24	243
	8	55109	0	20.06	23	30	48	128	0	0	257	24	243
-2	0	56525	0	19.96	14	29	57	0	0	128	229	9	286
	2	56525	0	19.96	15	29	56	32	0	96	229	9	286
	4	56525	0	19.96	15	29	56	64	0	64	229	9	286
	6	56525	0	19.97	16	29	55	96	0	32	229	9	286
	8	56525	0	19.97	16	29	55	128	0	0	229	9	286
0	0	56003	0	19.41	13	30	57	0	0	128	256	10	258
	2	56003	0	19.41	14	30	56	32	0	96	256	10	258
	4	56003	0	19.42	14	30	56	64	0	64	256	10	258
	6	56003	0	19.42	15	30	55	96	0	32	256	10	258
	8	56003	0	19.42	15	30	55	128	0	0	256	10	258
2	0	55514	0	18.39	11	30	58	0	0	128	221	3	300
	2	55514	0	18.39	12	30	58	32	0	96	221	3	300
	4	55514	0	18.39	12	30	57	64	0	64	221	3	300
	6	55514	0	18.39	13	30	57	96	0	32	221	3	300
	8	55514	0	18.40	13	30	56	128	0	0	221	3	300
4	0	54888	0	18.10	11	28	61	0	0	128	215	7	302
	2	54888	0	18.10	11	28	60	32	0	96	215	7	302
	4	54888	0	18.10	12	28	60	64	0	64	215	7	302
	6	54888	0	18.11	12	28	59	96	0	32	215	7	302
	8	54888	0	18.11	13	28	59	128	0	0	215	7	302

Attacker's Perspective for Trojans: It holds true again that more LVT cells in state registers are preferable, but the local variation of timing constraints also plays an important role now. This can also be explained by the more varied options for VT cells, especially with a more varied use of RVT and HVT cells for more relaxed constraints (Table X).

Conversely, this also means that an attacker may benefit as

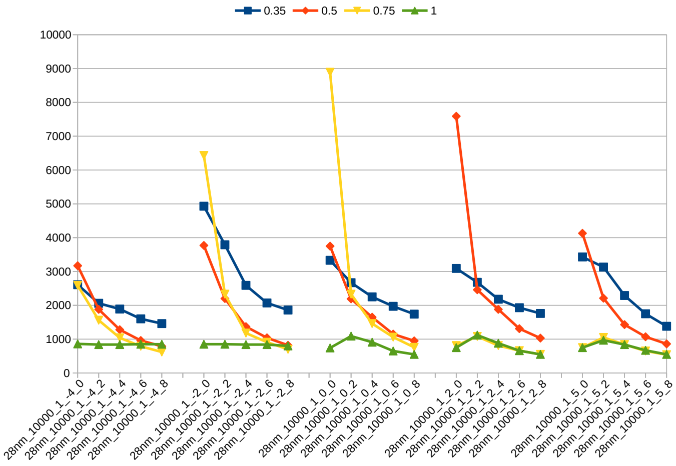

Fig. 6. NTD for AES core, in 28nm node, using LVT, RVT, and HVT cells. For reading the naming scheme of different cases listed on the x-axis, please refer back to Sec. IV-A.

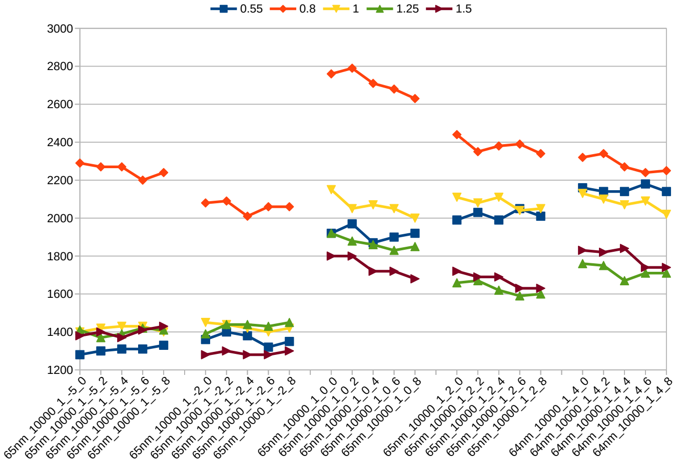

Fig. 7. NTD for AES core, in 65nm node, using LVT, RVT, HVT cells. See also Fig. 6's caption.

well from employing more LVT cells in other parts of the design. We deduct this from the fact that, for negative timing variations, we had to employ more LVT cells in other parts as well (Table X) in order to meet timing (Algorithm 1) – this had a considerable detrimental effect on resilience, and an attacker could achieve a similar outcome by adding more LVT cells in other design parts. This can be interesting when trying to avoid malicious modifications directly of the sensitive parts of a design such as the AES state registers, as these registers might be particularly vetted by runtime detection against Trojans.

PPA Analysis: For the 28nm node, PPA results for time constraints of 0.35ns and 0.5ns are reported in Table VIII and Table IX, respectively. Note that the constraints considered vary somewhat at both lower and upper ends from those considered in Case Study 1; this is on purpose to allow for better utilization of RVT cells as appropriate. For the 65nm node, PPA results are provided in Table X. Overall, the findings are similar to those in Case Study 1. In short, most aggressive timing constraints do not offer promising trade-offs for PPA cost (especially not for power) versus security, whereas more relaxed constraints are promising.

F. Case Study 4: Technology-Accurate Power Model for CPA

Here we study whether a technology-accurate (TA) power model is more effective than the commonly considered HD model for the CPA. Such TA model simply describes the actual data-dependent power values, instead of simplifying/abstracting the power consumption into '0' and '1' for the corresponding data patterns.

We have conducted this case study on the (so far) most challenging scenario for the CPA, namely the baseline AES design for 0.5ns timing constraint and -2 timing offset. The related CPA results are shown in Table XI.

TABLE XI
CPA RESULTS FOR AES CORE, FOR COMPARISON OF HD AND TA POWER MODEL, IN 28NM NODE, OVER 50K TRACES IN TOTAL.

LVT Cells	NTD for HD Model	NTD for TA Model
0	17300	15100
2	7050	48200
4	4550	N/A
6	3050	4950
8	2150	2500

We note that the TA power model is (1) only slightly more efficient for the case of 0 LVT cells, (2) somewhat close for the cases of the majority of the 8 bits being implemented in LVT cells, but (3) much worse for cases of 2 and 4 LVT cells. This maintains our argument that the role of LVT versus HVT cells is critical even in the presence of more advanced PSCA efforts. Next, we explain these results in more detail.

Observation (1) is because the actual power values can indeed be beneficial for correlation analysis. Observation (2) is due to the fact that the static power of LVT dominates that of HVT cells by orders of magnitudes. Thus, in such scenarios with the majority of cells being LVT, the role of the HVT cells in minority is masked and does contribute little to the more complicated noise profiles discussed next. Most important is to note that Observation (3) shows that the benefit of the TA power model does not apply anymore when there is a mix of LVT and HVT cells employed within the bytes of the state registers. This is because of a well-known limitation for the CPA setup, which is explained next.

Instead of exploring all possible keys, e.g., 2^{128} for AES with 128 bits keys – which is computationally intractable – attack frameworks do decompose the problem at the byte level, i.e., into 16 separate search spaces for sub-keys of size 2^8. Doing so is well feasible and also successful in general. However, an important implication here is the following: for any sub-key / byte currently under attack, all 15 remaining sub-keys still contribute to the power observed in the traces. This is by construction: attackers can only observe the total power trace, not traces for individual bytes or even bits. Thus, there are inherent noise profiles to deal with.

Now, for the commonly considered HD model, there are no differences in power values for LVT versus HVT cells, as the model's sole focus is on differences in '0' versus '1' for input and output data at FF level. This is essential when having to deal with the above explained noise profiles for all other 15 bytes. For these bytes, even if the attackers know the specific LVT versus HVT cell assignment, they cannot know (at that point in time when attacking the remaining specific byte) about their input and output data as they do not jointly explore the corresponding sub-key search space for those remaining 15 bytes (due to the computational intractability). Thus, given the inherent limitations of CPA, for scenarios of mixed LVT and RVT cells, the VT-agnostic HD model is more suitable, as the related noise profiles are more uniform across the board, thereby avoiding mis-correlations which can well occur for the TA model (due to varying assignments of LVT and RVT cells in the remaining sub-keys not under consideration at any given point in time when attacking some other specific sub-key).

Finally note that the we required 50k traces here, unlike 10k traces for the previous experiments. With larger datasets, the absolute number of outliers would increase,[10] and, thus, NTD reported for the HD model are also somewhat higher here.

G. Case Study 5: Comparison to TrojanZero

Recall that *TrojanZero* [34] was proposed as a methodology to implement zero-overhead Trojans. Although the authors of that work have indicated on PSCA, they lack a clear attack evaluation. Here we provide a comparison of *TrojanZero* to our regular Trojan design as follows. First, we implement the idea of *TrojanZero*'s strategy for reclaiming area. Second, since *TrojanZero* requires an actual Trojan for its implementation, we devise an alternative for our regular Trojan as follows. Constrained by the amount of reclaimed area and the corresponding budget in dynamic power, we add as many additional LVT FFs as possible to the AES core. We connect these LVT FFs in parallel to the state FFs, and we do so evenly across the bytes encoding of the state FFs.

TABLE XII
CPA RESULTS FOR AES CORE, FOR COMPARISON WITH *TrojanZero*, IN 28NM NODE.

Scenario	NTD
Baseline	18800
Our *TrojanZero*	15400
Our Trojan (2 LVT Cells)	10200
Our Trojan (4 LVT Cells)	7830
Our Trojan (6 LVT Cells)	4550
Our Trojan (8 LVT Cells)	2990
Our Trojan (2 LVT Cells) + *TrojanZero*	9900
Our Trojan (4 LVT Cells) + *TrojanZero*	7400
Our Trojan (6 LVT Cells) + *TrojanZero*	4300
Our Trojan (8 LVT Cells) + *TrojanZero*	2600

The related CPA results are shown in Table XII. While *TrojanZero*'s methodology allows us to implement a working S-PSCA-based Trojan, both stand-alone or on top of our regular Trojan design, our regular Trojan design by itself is more effective for undermining the resilience. This holds true even for only few changes in VT cells for ours. Besides, the PPA costs for both approaches are comparable, i.e., area and timing costs are zero, whereas power costs are marginal.

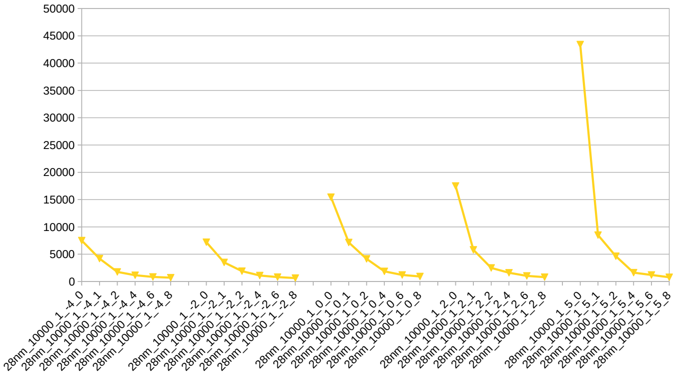

Fig. 8. NTD for AES core, in 28nm node, with the ELB countermeasure [12] in place. For reading the naming scheme of different cases listed on the x-axis, please refer back to Sec. IV-A.

H. Case Study 6: AES with Countermeasure

To further demonstrate the effectiveness of our method in case there are some countermeasures in place, we have exemplarily implemented the Exhaustive Logic Balancing (ELB) scheme proposed in [12]. We implemented this by replacing all state registers with the ELB circuitry as outlined in [12, Fig. 7]. On top of ELB, we again replace HVT cells with LVT cells following our method, just this time across the exhaustively balanced state registers. Without loss of generality, we consider a timing constraint of 0.5ns here.

Security Analysis: The CPA results for thorough sampling for the 28nm node are provided in Figure 8.

As expected, the overall resilience is much higher with the ELB countermeasure in place, reaching up to 43,400 NTD. Second, the ELB countermeasure itself is sensitive to timing variations, as can be seen from the considerable variations in traces across all cases with 0 LVT cells. This observation also suggests some further analysis of the design space and corresponding re-design; such tuning of the ELB countermeasure is left for future work. Third, the overall trends are the same as before: an increase in LVT cells reduces the resilience across all timing settings.

Attacker's Perspective for Trojans: This is the first time the effect of S-PSCA-based Trojan design on the ELB countermeasure is studied. Our insights for the role of LVT cells are still applicable here, yet with even more pronounced variations: already a single LVT cell can bring significant gains for attackers. This shortcoming of ELB can be explained by its construction: the key idea of ELB is to balance/equalize the static power of an FF across all possible data patterns, by replacing it with a set of FFs that operate in parallel on differential data patterns at once. By imposing LVT cells for these FFs, however, this very idea is undermined. The fact that we can obtain such a notable reduction in resilience, effectively nullifying the protection offered by ELB, all without any overheads, renders this Trojan attack a real threat.

PPA Analysis: For the 28nm node, PPA results for the most interesting / most resilient constraint of 0.5ns are reported in Table XIII. Due to ELB by itself, the required area is almost 2× of the baseline design (Table III). This is in agreement with the results reported in [12].

[10] Outliers are to be understood as traces that, while corresponding to the correct key, remain difficult to correlate against, simply by going against the overall correlation trends. Since PCC is based on linear correlation only, such outliers can arise relatively easily.

TABLE XIII
PPA RESULTS FOR AES CORE, WITH ELB COUNTERMEASURE, IN 28NM NODE, FOR TIMING CONSTRAINT OF 0.5NS. SEE ALSO TABLE II'S CAPTION.

Time	L	Area (μm^2)	WNS (ns)	Power (mW)	L (%)	H (%)	State FFs		NS FFs	
							L	H	L	H
-4	0	25265	0	47.85	6	94	0	128	25	499
	2	25265	0	47.87	7	93	32	96	25	499
	4	25265	0	47.86	8	92	64	64	25	499
	6	25265	0	47.87	9	91	96	32	25	499
	8	25265	0	47.86	10	90	128	0	25	499
-2	0	25268	0	47.73	5	95	0	128	12	512
	2	25268	0	47.74	7	93	32	96	12	512
	4	25268	0	47.73	8	92	64	64	12	512
	6	25268	0	47.75	9	91	96	32	12	512
	8	25268	0	47.76	10	90	128	0	12	512
0	0	25125	0	46.15	5	95	0	128	10	514
	2	25125	0	46.16	7	93	32	96	10	514
	4	25125	0	46.17	8	92	64	64	10	514
	6	25125	0	46.15	9	91	96	32	10	514
	8	25125	0	46.16	10	90	128	0	10	514
2	0	24879	0	44.25	5	95	0	128	25	499
	2	24879	0	44.26	6	94	32	96	25	499
	4	24879	0	44.26	7	93	64	64	25	499
	6	24879	0	44.25	8	92	96	32	25	499
	8	24879	0	44.25	9	91	128	0	25	499
5	0	24772	0	43.78	4	96	0	128	8	516
	2	24772	0	43.79	5	95	32	96	8	516
	4	24772	0	43.81	7	93	64	64	8	516
	6	24772	0	43.80	8	92	96	32	8	516
	8	24772	0	43.79	9	91	128	0	8	516

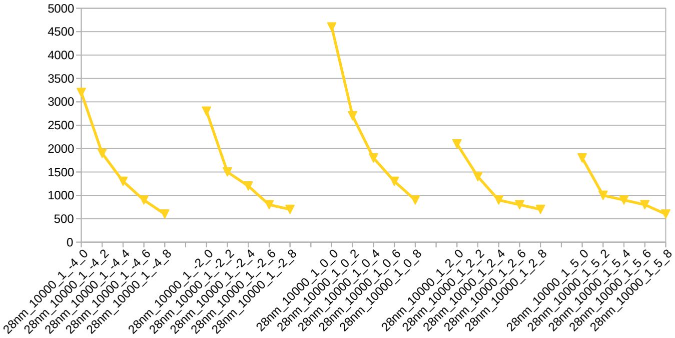

Fig. 9. NTD for PRESENT core, in 28nm node. For reading the naming scheme of different cases listed on the x-axis, please refer back to Sec. IV-A.

I. Case Study 7: PRESENT Crypto Core

We implement another crypto core, PRESENT, and showcase our method's effectiveness on that as well. PRESENT [36] is a commonly considered, lightweight crypto core. Without loss of generality, we consider a timing constraint of 0.3ns here, as that provides a competitive hardware implementation with fast operation for the 28nm node.

Security Analysis: The CPA results for thorough sampling for the 28nm node are shown in Figure 9. The observed trend remains consistent: the number of traces reduces with an increase in the number of LVT cells in the state registers.

Attacker's Perspective for Trojans: Also for the PRESENT crypto core, we observe the same significant impact as before: with an increase in LVT cells in the state registers, NTD reduces, thus allowing related Trojans to significantly comprise the core's security.

PPA Analysis: Similar to our observations for the AES core, we succeeded to implement the changes in the state registers with almost no overhead; Table XIV shows the corresponding results. Most importantly, the WNS is not affected.

TABLE XIV
PPA RESULTS FOR PRESENT CORE, IN 28NM NODE, FOR TIMING CONSTRAINT OF 0.3NS. SEE ALSO TABLE II'S CAPTION.

Time	L	Area (μm^2)	WNS (ns)	Power (mW)	L (%)	H (%)	State FFs		NS FFs	
							L	H	L	H
-4	0	789	0	3.23	8	92	0	64	3	86
	2	792	0	3.23	9	91	16	48	3	86
	4	789	0	3.23	9	91	32	32	3	86
	6	790	0	3.24	10	90	48	16	3	86
	8	791	0	3.25	11	89	64	0	3	86
-2	0	768	0	3.11	8	92	0	64	3	86
	2	768	0	3.11	8	92	16	48	3	86
	4	768	0	3.12	8	92	32	32	3	86
	6	768	0	3.13	9	91	48	16	3	86
	8	768	0	3.13	9	91	64	0	3	86
0	0	771	0	3.04	7	93	0	64	1	88
	2	771	0	3.04	7	93	16	48	1	88
	4	773	0	3.04	7	93	32	32	1	88
	6	771	0	3.05	8	92	48	16	1	88
	8	775	0	3.06	8	92	64	0	1	88
2	0	775	0	3.01	6	94	0	64	1	88
	2	775	0	3.02	6	94	16	48	1	88
	4	778	0	3.02	7	93	32	32	1	88
	6	776	0	3.03	7	93	48	16	1	88
	8	777	0	3.04	8	92	64	0	1	88
5	0	792	0	2.88	3	97	0	64	0	89
	2	793	0	2.88	3	97	16	48	0	89
	4	792	0	2.88	4	96	32	32	0	89
	6	792	0	2.89	5	95	48	16	0	89
	8	795	0	2.91	6	94	64	0	0	89

J. Case Study 8: PRESENT with Countermeasures

We implement the ELB countermeasure for PRESENT as outlined in Case Study 6. Furthermore, we implement Threshold Implementation (TI) masking as outlined in [12, Fig. 8], which is based on [38, Profile 2 (Data Sharing)].[11] Note that, given the considerable modifications in logic introduced by this TI scheme, we first conduct functional verification to ensure correctness of the implementation. Further note that we implement both countermeasures separately as well as in combination, for the latter with ELB applied on top of TI.

For design-space exploration, we again replace HVT cells with LVT cells, this time for the masked and/or balanced state registers. Without loss of generality, we consider a timing constraint of 0.5ns here.

Security Analysis: The CPA results for the 28nm node are provided in Table XV. Note that we only report on the adversarial mode for our method here, i.e., on targeted reduction of NTD, not on the general design-space exploration. Also note that, for a fair comparison across all scenarios, we further report the PCC values for best key candidates.

While the ELB countermeasures shows similar lack of resilience as in Case Study 6, the TI countermeasure remains resilient, even across 1M traces. Note that the finding of TI contributing the most for combined countermeasures aligns with [12]. Also note that in [38], the reference for the actual TI implementation, it was indicated as well that this TI implementation should remain resilient, with the caveat that this statement was made in the context of D-PSCA in [38].

[11] This TI scheme is tailored to PRESENT; thus, we cannot apply this countermeasure for AES as well.

TABLE XV
CPA results for PRESENT core, with TI and ELB countermeasures, in 28nm node, over 1M traces in total.

PRESENT Countermeasure	NTD	PCC
ELB	35000	0.330
ELB + Ours (Adversarial Mode)	1100	0.400
TI	N/A	0.039
TI + Ours (Adversarial Mode)	N/A	0.033
TI + ELB	N/A	0.029
TI + ELB + Ours (Adversarial Mode)	N/A	0.013

TABLE XVI
PPA results for PRESENT core, with TI and ELB countermeasures, in 28nm node, for timing constraint of 0.5ns. See also Table II's caption.

Time	L	Area	WNS	Power	L	H	State FFs		NS FFs	
		(μm^2)	(ns)	(mW)	(%)	(%)	L	H	L	H
-4	0	8495	0	15.30	35	65	0	192	247	35
	2	8495	0	15.33	36	64	48	144	247	35
	4	8495	0	15.34	36	64	96	96	247	35
	6	8495	0	15.34	37	63	144	48	247	35
	8	8495	0	15.35	37	63	192	0	247	35
-2	0	8321	0	15.21	33	67	0	192	253	29
	2	8321	0	15.21	33	67	48	144	253	29
	4	8321	0	15.21	34	66	96	96	253	29
	6	8321	0	15.22	34	66	144	48	253	29
	8	8321	0	15.23	35	65	192	0	253	29
0	0	8187	0	15.18	29	71	0	192	239	43
	2	8187	0	15.18	29	71	48	144	239	43
	4	8187	0	15.19	30	70	96	96	239	43
	6	8187	0	15.19	30	70	144	48	239	43
	8	8187	0	15.20	31	69	192	0	239	43
2	0	8078	0	15.16	29	71	0	192	232	50
	2	8078	0	15.16	29	71	48	144	232	50
	4	8078	0	15.16	30	70	96	96	232	50
	6	8078	0	15.16	30	70	144	48	232	50
	8	8078	0	15.17	31	69	192	0	232	50
5	0	7952	0	15.14	26	74	0	192	224	58
	2	7952	0	15.14	26	74	48	144	224	58
	4	7952	0	15.14	27	73	96	96	224	58
	6	7952	0	15.15	27	73	144	48	224	58
	8	7952	0	15.16	28	72	192	0	224	58

Attacker's Perspective for Trojans: As in Case Study 6, S-PSCA-based Trojans remain a practical threat for the ELB countermeasure applied for the PRESENT core. However, for the TI countermeasure, this does not hold true anymore. The fact that TI remains resilient can be explained by its construction and working principles: the key idea of TI is to incorporate randomized masking for all the state registers' operation. Thus, any notable increase in static power (induced by more LVT cells) for the masked state registers cannot represent the original and sensitive computation anymore, but only the randomly masked and protected computation. In fact, the stronger emphasis on randomly masked data leads to lower PCC values, i.e., such Trojans are even counterproductive.[12]

PPA Analysis: The results are shown in Table XVI. Most important to note are the considerable overheads for the combined countermeasure: the required area is $\approx 11\times$ that of the baseline design (Table XIV). This is in agreement with the results reported in [12]. Thus, whether this countermeasure can be applied in practice depends on the margins/budget for the IC to protect.

K. Lessons Learned

Our experiments offer a range of novel and important insights into S-PSCA. From our study findings, we postulate the following guidelines for security-aware design:

1) Prior art assumed that high-performance designs are generally vulnerable to S-PSCA, but we found that timing constraints as well as the distribution and ratio of different threshold-voltage cells play a much more important role. Thus, the security-versus-PPA design-space should be explored thoroughly.
2) Related to 1), considering the significant variability throughout the design space, which is dictated by timing constraints and different VT cells, it is advisable to thoroughly explore PPA-security trade-offs. Such efforts are supported by our security-aware CAD framework.
3) Limit LVT cells in critical parts like crypto-core state registers. A general and strict rule to not allow LVT cells in such registers may well complicate timing closure, even if those parts are only small. For this very reason, a CAD framework like ours is essential.
4) Use LVT cells in other parts where necessary for managing timing closure. Doing so will generally not compromise resilience, although this depends also on the technology node to some degree.
5) Use all available types of VT cells. This promotes a diverse application of VT cells by CAD tools, leading to a variety of power profiles that are difficult to compromise.
6) Aim for moderate and/or slightly aggressive timing constraints; avoid lax timing. Fortunately, this aligns with standard objectives for PPA optimization.

Concerning the offense perspective, an attacker can implement highly effective PSC-based Trojan by revising only few gates toward LVT cells. Such Trojans are stealthy as they are using zero additional gates and do not undermine timing paths if designed properly. From an attacker's perspective, we postulate that one should use as many LVT cells in the sensitive gates as possible; the more LVT cells are used for sensitive gates, the lower the resilience to S-PSCA. While this holds true across wide ranges of timing constraints, it can be limited by varied usage of different VT cells in other parts of the design, older technology nodes, and countermeasures (if any are in place, given their considerable cost).

VI. Conclusions

Here, we studied the role of different threshold-voltage cells on the resilience of representative AES and PRESENT crypto-cores (implemented for commercial 28nm and 65nm nodes) against static power side-channel attacks. Toward that end, we first developed a security-aware design-space exploration framework using commercial CAD tools and an open-source CPA tool. On the one hand, this framework can be used by designers and security experts to trade-off between design cost and security and, most importantly, for IC security closure

[12] With these insights, and following the discussion in [38], attackers might resort to additionally employing LVT cells for the round-key registers. As such approach would require an entirely different CPA framework for security analysis, this is left for future work. In any case, taking up a classical "game of cat and mouse," the TI scheme can be extended to round-key registers as well [38], albeit with area overheads being incurred twice then.

before tape-out. On the other hand, this framework can also guide adversaries for zero-cost Trojans that render ICs much more vulnerable to static power side-channel attacks. Second, based on an extensive and thorough set of case studies, we provide important insights for both attackers and defenders.

In future work, we would seek to obtain access to commercial libraries for even more advanced nodes and to advanced attacks, e.g., using machine learning-based models. We will release our scripts later on, after modularizing procedures along with outsourcing of technology settings, as we are legally not allowed to share details for commercial libraries.

References

[1] K. Gandolfi *et al.*, “Electromagnetic analysis: Concrete results,” in *International workshop on Cryptograph. Hardw. and Embedded Syst.* Springer, 2001, pp. 251–261.

[2] P. Kocher *et al.*, “Differential power analysis,” in *Annual International Cryptology Conf.* Springer, 1999, pp. 388–397.

[3] P. C. Kocher, “Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems,” in *Annual International Cryptology Conf.* Springer, 1996, pp. 104–113.

[4] M. Randolph *et al.*, “Power side-channel attack analysis: A review of 20 years of study for the layman,” *Cryptography*, vol. 4, no. 2, 2020.

[5] S. M. Del Pozo *et al.*, “Side-channel attacks from static power: When should we care?” in *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2015, pp. 145–150.

[6] K. Yang *et al.*, “A2: Analog malicious hardware,” in *2016 IEEE Symposium on Security and Privacy (SP)*, 2016, pp. 18–37.

[7] E. Brier *et al.*, “Correlation power analysis with a leakage model,” in *Cryptographic Hardware and Embedded Systems - CHES 2004*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 16–29.

[8] M. Alioto *et al.*, “Leakage power analysis attacks: A novel class of attacks to nanometer cryptographic circuits,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 2, pp. 355–367, 2010.

[9] A. Moradi, “Side-channel leakage through static power,” in *Cryptographic Hardware and Embedded Systems – CHES 2014*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 562–579.

[10] T. Popp *et al.*, “Masked dual-rail pre-charge logic: Dpa-resistance without routing constraints,” in *Cryptographic Hardware and Embedded Systems – CHES 2005*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 172–186.

[11] N. Veyrat-Charvillon *et al.*, “Shuffling against side-channel attacks: A comprehensive study with cautionary note,” in *Advances in Cryptology – ASIACRYPT 2012*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 740–757.

[12] T. Moos *et al.*, “Countermeasures against static power attacks: comparing exhaustive logic balancing and other protection schemes in 28 nm cmos ,” *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2021, no. 3, p. 780805, Jul. 2021.

[13] S. Skorobogatov *et al.*, “Breakthrough silicon scanning discovers backdoor in military chip,” in *Cryptographic Hardware and Embedded Systems – CHES 2012*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 23–40.

[14] R. S. Chakraborty *et al.*, “Mero: A statistical approach for hardware trojan detection,” in *Cryptographic Hardware and Embedded Systems - CHES 2009*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 396–410.

[15] J. Knechtel *et al.*, “Benchmarking security closure of physical layouts: Ispd 2022 contest,” in *Proceedings of the 2022 International Symposium on Physical Design*, ser. ISPD ’22. New York, NY, USA: Association for Computing Machinery, 2022, p. 221228.

[16] M. Eslami *et al.*, “Benchmarking advanced security closure of physical layouts: Ispd 2023 contest,” in *Proceedings of the 2023 International Symposium on Physical Design*, 2023, pp. 256–264.

[17] F. Wang *et al.*, “Security closure of ic layouts against hardware trojans,” in *Proceedings of the 2023 International Symposium on Physical Design*, 2023, p. 229237.

[18] J. Knechtel *et al.*, “Towards secure composition of integrated circuits and electronic systems: On the role of EDA,” in *DATE*, 2020.

[19] J. Knechtel *et al.*, “Security closure of physical layouts,” in *ICCAD*, 2021.

[20] J. Lienig *et al.*, “Toward security closure in the face of reliability effects,” in *ICCAD*, 2021.

[21] J. Bhandari *et al.*, “Defending integrated circuit layouts,” Cryptology ePrint Archive, Paper 2023/205, 2023, https://eprint.iacr.org/2023/205.

[22] G. Guo *et al.*, “Assurer: A ppa-friendly security closure framework for physical design,” in *Proceedings of the 28th Asia and South Pacific Design Automation Conference*, ser. ASPDAC ’23. New York, NY, USA: Association for Computing Machinery, 2023, p. 504509.

[23] J. Giorgetti *et al.*, “Analysis of data dependence of leakage current in cmos cryptographic hardware,” in *Proceedings of the 17th ACM Great Lakes Symposium on VLSI*, ser. GLSVLSI ’07. New York, NY, USA: Association for Computing Machinery, 2007, p. 7883.

[24] T. Moos *et al.*, “Static power side-channel analysisan investigation of measurement factors,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 2, pp. 376–389, 2020.

[25] N. Karimi *et al.*, “Exploring the effect of device aging on static power analysis attacks,” *TCHES*, vol. 2019, p. 233256, May 2019.

[26] M. Djukanovic *et al.*, “Multivariate analysis exploiting static power on nanoscale cmos circuits for cryptographic applications,” in *Progress in Cryptology - AFRICACRYPT 2017*. Cham: Springer International Publishing, 2017, pp. 79–94.

[27] D. Bellizia *et al.*, “Sc-ddpl: A novel standard-cell based approach for counteracting power analysis attacks in the presence of unbalanced routing,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 7, pp. 2317–2330, 2020.

[28] P. Slpsk *et al.*, “Karna: A gate-sizing based security aware eda flow for improved power side-channel attack protection,” in *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2019.

[29] J. a. o. Zhang, “On trojan side channel design and identification,” in *2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2014, pp. 278–285.

[30] W. Meng *et al.*, “An implementation of trojan side-channel with a masking scheme,” in *2017 13th International Conference on Computational Intelligence and Security (CIS)*, 2017, pp. 566–569.

[31] L. Lin *et al.*, “Trojan side-channels: Lightweight hardware trojans through side-channel engineering,” in *Cryptographic Hardware and Embedded Systems - CHES 2009*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 382–395.

[32] T. Perez *et al.*, “Side-channel trojan insertion - a practical foundry-side attack via eco,” in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2021, pp. 1–5.

[33] X. Wang *et al.*, “Sequential hardware trojan: Side-channel aware design and placement,” in *2011 IEEE 29th International Conference on Computer Design (ICCD)*, 2011, pp. 297–300.

[34] I. H. Abbassi *et al.*, “Trojanzero: Switching activity-aware design of undetectable hardware trojans with zero power and area footprint,” in *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2019, pp. 914–919.

[35] J. Tschanz *et al.*, “Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage,” *IEEE Journal of Solid-State Circuits*, vol. 37, no. 11, pp. 1396–1402, 2002.

[36] A. Bogdanov *et al.*, “Present: An ultra-lightweight block cipher,” in *Cryptographic Hardware and Embedded Systems - CHES 2007*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 450–466.

[37] J. Knechtel *et al.*, “Design-time exploration of voltage switching against power analysis attacks in 14 nm finfet technology,” *Integr. VLSI J.*, vol. 85, no. C, p. 2734, jul 2022. [Online]. Available: https://doi.org/10.1016/j.vlsi.2022.02.006

[38] A. Poschmann *et al.*, “Side-channel resistant crypto for less than 2,300 ge,” *J. Cryptology*, vol. 24, pp. 322–345, 2011.

Jitendra Bhandari received the B.Tech degree in electrical engineering with a minor in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India in 2020. He is currently pursuing Ph.D. degree with the NYU Centre for Cybersecurity. His research interests include hardware security, Side Channel Analysis, FPGA design, and wireless security.

Likhitha Mankali is a Ph.D. candidate at the Department of Electrical and Computer Engineering at Tandon School of Engineering, New York University, NY, USA. She is also a Global Ph.D. fellow with New York University Abu Dhabi, UAE. Her research interests include Hardware Security and using Machine learning to enhance and quantify the security of IP protection techniques. She has won third place in the CSAW-Logic Locking Conquest 2021 and HeLLO CTF competition 2022.

Mohammed Nabeel holds a Bachelors degree in Electrical and Electronics Engineering from the National Institute of Technology Calicut in India. With over ten years of experience in the chip design industry, he is currently pursuing Ph.D. degree in the Department of Computer Science and Engineering at the Tandon School of Engineering, New York University, NY, USA. His research interests include hardware security, side-channel analysis, and the design of hardware accelerators for various cryptographic schemes.

Ozgur Sinanoglu is a professor of electrical and computer engineering at New York University Abu Dhabi where he is also the Associate Dean of Engineering currently. He earned his B.S. degrees, one in Electrical and Electronics Engineering and one in Computer Engineering, both from Bogazici University, Turkey in 1999. He obtained his MS and PhD in Computer Science and Engineering from University of California San Diego in 2001 and 2004, respectively. He has industry experience at Texas Instruments, IBM and Qualcomm, and has been with NYU Abu Dhabi since 2010. During his PhD, he won the IBM PhD fellowship award twice. He is also the recipient of the best paper awards at IEEE VLSI Test Symposium 2011 and ACM Conference on Computer and Communication Security 2013. Sinanoglu received the inaugural NYUAD Distinguished Research Award in 2021. Also in 2021, he was inducted into the Mohammed bin Rashid Academy of Science in the UAE.

Prof. Sinanoglus research interests include design-for-test, design-for-security and design-for-trust for VLSI circuits, where he has around than 250 conference and journal papers, and 20 issued and pending US Patents. Sinanoglu has given more than a dozen tutorials on hardware security and trust in leading chip design automation conferences. He has served as track/topic chair or technical program committee member in about 15 conferences, and as (guest) associate editor for various journals including IEEE TIFS, IEEE TCAD, IEEE TETC, IEEE ESL, ACM JETC.

Prof. Sinanoglu is the director of the Center for Cybersecurity as well as the Design-for-Excellence Lab at NYU Abu Dhabi. His recent research in hardware security and trust is being funded by US National Science Foundation, US Department of Defense, Semiconductor Research Corporation, Intel Corp, and Mubadala Technology.

Ramesh Karri (Fellow, IEEE) received the B.E. degree in electrical and computer engineering from Andhra University, Visakhapatnam, India, in 1985, and the Ph.D. degree in computer science and engineering from the University of California at San Diego, La Jolla, CA, USA, in 1993.

He is a Professor of Electrical and Computer Engineering at New York University, New York, NY, USA. He co-founded and co-directs the NYU Center for Cyber Security. He founded the Embedded Systems Challenge, the annual red team blue team event. He co-founded Trust-Hub. His research and education activities in hardware cybersecurity include trustworthy integrated circuits (ICs); processors and cyber-physical systems; security-aware computer-aided design, test, verification, and validation; nano meets security; hardware security competitions, benchmarks, and metrics; biochip security; and manufacturing security. He has published over 300 articles in leading journals and conferences.

Dr. Karri is a Fellow of the IEEE for leadership and contributions to Trustworthy Hardware. His work on hardware cybersecurity received the best paper nominations at ICCD 2015 and DFTS 2015 and awards at ACM TODAES 2018, ITC 2014, CCS 2013, DFTS 2013, and VLSI Design 2012. He received the Humboldt Fellowship and the NSF CAREER Award. He serve(d)s on the Editorial Board of IEEE and ACM Transactions, including IEEE Transactions on Information Forensics and Security, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, TODAES, ESL, D&T, and JETC. He served on the Executive Committee of IEEE/ACM DAC leading the SecurityDAC initiative from 2014 to 2017. He served as the program/general chair for conferences and serves on program committees. He is Editor-in-Chief of ACM Journal on Emerging Technologies in Computing Systems (JETC). He was an IEEE Computer Society Distinguished Visitor from 2013 to 2015.

Johann Knechtel received the M.Sc. degree in Information Systems Engineering (Dipl.-Ing.) and the Ph.D. degree in Computer Engineering (Dr.-Ing., summa cum laude) from TU Dresden, Germany, in 2010 and 2014. He is a Research Scientist with New York University Abu Dhabi, United Arab Emirates. From 2015 to 2016, he was a Postdoctoral Researcher with the Masdar Institute of Science and Technology, Abu Dhabi; from 2010 to 2014, he was a Ph.D. Scholar with the DFG Graduate School “Nano- and Biotechnologies for Packaging of Electronic Systems” hosted at TU Dresden; in 2012, he was a Research Assistant with the Chinese University of Hong Kong; and in 2010, he was a Visiting Research Student with the University of Michigan at Ann Arbor, MI, USA. His research interests cover VLSI physical design automation, with particular focus on emerging technologies and hardware security. He has (co-) authored around 50 publications.