

# PoSyn: Secure Power Side-Channel Aware Synthesis

Amisha Srivastava\*, Samit S. Miftah\*, Hyunmin Kim†, Debjit Pal‡, Kanad Basu\*

\*ECE Department, University of Texas at Dallas, USA

†Technology Innovation Institute, UAE

‡ECE Department, University of Illinois Chicago, USA

**Abstract**—Power Side-Channel (PSC) attacks exploit power consumption patterns to extract sensitive information, posing risks to cryptographic operations crucial for secure systems. Traditional countermeasures, such as masking, face challenges like complex synthesis integration, high area overhead, and vulnerability to optimization removal during logic synthesis. To address these issues, we introduce PoSyn, a novel logic synthesis framework designed to enhance cryptographic hardware’s resistance against PSC attacks. Our approach focuses on the optimal bipartite mapping of vulnerable RTL components to standard cells from the technology library to minimize PSC leakage. By employing a cost function that integrates key characteristics from the RTL design and the standard cell library, we strategically modify the mapping criteria during the conversion of RTL designs into standard cell netlists without altering the design functionality. Furthermore, PoSyn is theoretically shown to minimize mutual information leakage, further reinforcing its security against PSC vulnerabilities. PoSyn is evaluated on a variety of cryptographic hardware, including AES, RSA, PRESENT, and post-quantum cryptography algorithms like Saber and CRYSTALS-Kyber across 65nm, 45nm, and 15nm nodes. Our experimental results demonstrate a significant reduction of success rates for Differential Power Analysis (DPA) and Correlation Power Analysis (CPA) attacks, as low as 3% and 6%, respectively. Furthermore, TVLA analysis confirms that the synthesized netlists exhibit negligible leakage. Moreover, compared to traditional countermeasures such as masking and shuffling, PoSyn achieves notably lower success rates, achieving a reduction by up to 72%, while simultaneously enhancing area efficiency by as much as 3.79×. These results highlight the effectiveness of PoSyn in securing cryptographic hardware with minimal impact on area and performance.

## I. INTRODUCTION

Power Side-Channel (PSC) attacks represent a formidable threat to cryptographic systems, leveraging observable physical phenomena such as dynamic power consumption to covertly extract sensitive information from encryption hardware. The advent of side-channel attacks has highlighted significant security vulnerabilities, where adversaries exploit measurable physical effects during cryptographic operations. Central to this concern is the ability of these attackers to use such phenomena to breach the security of crucial computational processes [1]. These attacks have a wide range of effects, impacting a diverse array of cryptographic algorithms, including but not limited to symmetric and asymmetric key algorithms, hash functions, and digital signature schemes [2]. As these cryptographic algorithms are fundamental to securing network protocols, enhancing their resilience against PSC

attacks is crucial for maintaining the trustworthiness of these systems and ensuring the confidentiality, integrity, and availability of sensitive information across various applications.

To address these challenges, a spectrum of countermeasures has been developed to safeguard sensitive data by obfuscating observable effects. Most existing approaches for mitigating PSC attacks are implemented at the post-silicon level [3], [4], [5], [6]. These include techniques such as real-time PSC attack detection systems that employ on-chip sensors for threat analysis and test vector leakage assessment [7], [8]. A major limitation of conducting PSC analysis at the post-silicon level is the difficulty in retrofitting security measures onto existing devices, which often results in expensive device redesigns. Therefore, the need to combat PSC attacks at the pre-silicon stage is paramount. Intervening early empowers designers to embed security features directly into the hardware design, ensuring robust defenses immediately. This early-stage intervention is important, as it offers a more amenable and cost-effective environment to modify hardware designs compared to rectifying security vulnerabilities in deployed systems.

At the Register Transfer Level (RTL), masking schemes with robust theoretical foundation have garnered significant attention as effective PSC countermeasures [9], [10], [11], [12]. Masking involves splitting sensitive information into random shares and processing these separately to dilute any exploitable patterns in power consumption [13]. It aims to diffuse sensitive information across multiple components, making it more challenging for attackers to reconstruct the original data from observable side-channel emissions. While the theoretical principles behind masking are sound, the practical implementation during the synthesis phase, which translates RTL designs into a netlist for manufacturing, introduces several challenges. The complexity added by integrating masking techniques often results in a significant increase in overhead where the final design can be multiple times larger than the original [14]. Moreover, practical challenges such as glitches or transition-based leakages compromise the critical assumption of independent leakage between shares, rendering the secure implementation of masking both challenging and time-consuming [15]. Furthermore, in the optimization phase of logic synthesis, which is focused on refining the design for efficiency and cost-effectiveness, tools may remove what they perceive as redundant logic. This can inadvertently weaken or completely strip away the protective measures introduced by masking, leaving the final netlist susceptible to attacks [16].

TABLE I: PoSyn vs. Existing Approaches

Approach	PSC Leakage Reduction	Area Overhead	Frequency Impact
Masking	✓ Significant Reduction	<b>High Overhead</b>	<b>Potential Degradation</b>
Conventional Synthesis	<b>Not Considered</b>	✓ Optimized for Area	✓ Optimized for Frequency
PoSyn	✓ Integrated Minimization	✓ Optimized for Area	✓ Meets Timing Constraints

Efforts to refine these masking schemes for enhanced efficiency and security are frequently marred by practical setbacks, including design inaccuracies and flaws [17], [18]. Such vulnerabilities lead to the requirement for a synthesis strategy that can reinforce defenses of the generated netlist against PSC attacks without the need to introduce extra operations in the RTL design. The traditional focus on performance and cost-efficiency during logic synthesis needs to be balanced with security, ensuring that protective measures are not compromised. Therefore, an effective logic synthesis strategy is required to obscure power consumption patterns, reinforcing the security of the generated netlist against PSC attacks without adding extra operations to the RTL design.

**In this paper, we propose, PoSyn, a side-channel-aware synthesis approach that strategically specifies the mapping criteria during the conversion of RTL designs into gate-level netlists.** PoSyn obscures power consumption patterns by generating a netlist that inherently resists power side-channel attacks while preserving the functional integrity of the original cryptographic design. To further illustrate the advantages of our proposed approach, Table I presents a comparative analysis of masking, conventional synthesis, and PoSyn. This comparison underscores the necessity of a synthesis framework that effectively balances security, area constraints, and performance. The major contributions of our paper are as follows:

- We, for the first time, introduce PoSyn, a novel logic synthesis framework that enhances cryptographic hardware resilience against PSC attacks by optimizing standard cell selection for vulnerable RTL components.
- We theoretically prove that PoSyn is secure, minimizing mutual information leakage and thereby, significantly enhancing its resilience against PSC vulnerabilities.
- We applied PoSyn to various cryptographic hardware, including AES, RSA, PRESENT, and post-quantum cryptography algorithms Saber and Kyber, and evaluated it on 65nm, 45nm, and 15nm libraries. PoSyn reduced DPA and CPA success rates to as low as 3% and 6%, achieving up to 72% improvement over conventional synthesis.
- Test Vector Leakage Assessment results confirm that PoSyn-generated netlists exhibit no detectable leakage across the cryptographic benchmarks and technology libraries, validating the security guarantees of PoSyn.
- Compared to existing masking and shuffling schemes, PoSyn reduces PSC attack success rates by up to 72%, while also achieving a 3.70 $\times$  reduction in area overhead, offering a significant improvement in both security and resource efficiency.

## II. RELATED WORKS

Power Side-Channel (PSC) attacks are of three types: (1) Simple Power Analysis (SPA), which observes power patterns

[19], (2) Differential Power Analysis (DPA), which uses statistical analysis over multiple operations [20], [21] and (3) Correlation Power Analysis (CPA), which correlates power consumption with intermediate values [22]. Both post-silicon and pre-silicon countermeasures have been explored to mitigate PSC attacks. Post-silicon techniques, such as test vector leakage assessment and structural information extraction [7], [8], [3], [6], detect vulnerabilities after fabrication but face limitations due to the need for costly device respins, as discussed in Section I. In contrast, pre-silicon methods address PSC vulnerabilities during the design phase. Balancing power consumption aims to maintain a constant profile regardless of operations but introduces additional logic requirements [23]. Shuffling disrupts power correlations by randomizing the order of operations, though this increases the complexity and results in high area overhead [24]. Masking involves adding random values to the data before and after processing, requiring extra hardware to manage the masked operations [13], [25]. Dual-rail precharge logic represents each bit with two wires, significantly increasing design complexity, area, and power consumption due to the doubled circuitry [26].

## III. THREAT MODEL

Our threat model follows standard PSC attack assumptions [1]. The adversary can interact with the cryptographic hardware to encrypt arbitrary plaintexts and collect corresponding power traces. However, the adversary cannot alter the netlist’s internal architecture or access additional side-channel data (*e.g.*, timing or electromagnetic emissions). The attack is external, relying solely on observable power traces. The goal is to analyze multiple traces across different states of the algorithm in order to determine the secret cryptographic key. By leveraging Differential Power Analysis (DPA) and Correlation Power Analysis (CPA) [20], [22], the adversary applies statistical techniques to correlate power fluctuations with internal computations, gradually extracting sensitive cryptographic information from repeated observations.

## IV. POSYN

The proposed approach, PoSyn, is designed to preserve the functionality of the RTL design while significantly obfuscating the associated power consumption patterns that could be exploited for PSC attacks. Figure 1 shows an overview of PoSyn. It accepts the RTL design and the technology library as inputs and generates a netlist immune to PSC attacks. The approach employs two steps: (1) PSC-aware synthesis for vulnerable components, and (2) standard optimization synthesis for non-vulnerable components. Utilizing these targeted methods, PoSyn enhances the security of the synthesized hardware without compromising design performance.

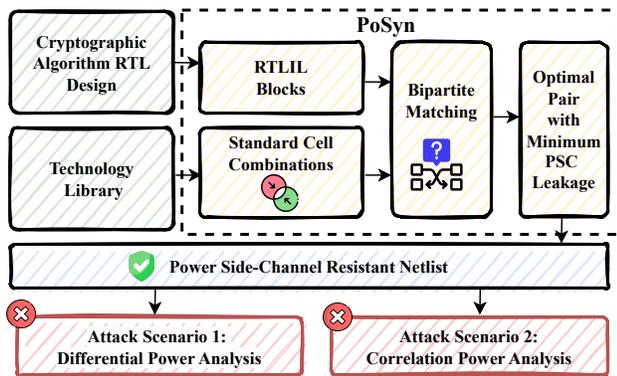


Fig. 1: Overview of the proposed side-channel aware synthesis framework (PoSyn) for synthesizing a netlist using only the RTL design and technology library as inputs.

### A. Vulnerable Component Identification in RTL

PoSyn initiates the process by identifying operations pivotal for encryption in the RTL design of the cryptographic algorithm [27], [28]. Our approach is partially user-defined, allowing designers to incorporate domain knowledge and specify criteria for marking components as vulnerable. This involves recognizing complex operations that are often repeated, like those found in *Sbox* lookup tables in AES. It also requires identifying sensitive variables, such as encryption keys, which are at a high risk of causing information leakage. We refine our analysis to discern the “leaky” modules and specific locations within these modules that are engaged in these critical operations and variables, thereby marking them as areas particularly vulnerable to PSC attacks. This strategy enables us to identify and subsequently fortify the design’s most critical areas against potential leakage. The following components from the RTL design are identified for the proposed side-channel aware synthesis method.

- **Sensitive variables:** The primary focus is on protecting sensitive variables, such as encryption keys, round keys, and critical intermediate states (e.g., *SBox* outputs in substitution-permutation ciphers). The exposure of these sensitive variables poses a dire risk, potentially compromising the security of encrypted data.
- **Encryption-specific operations:** Core encryption operations, which are susceptible to PSC attacks due to their repetitive nature and predictable power signatures, must be protected. Safeguarding these operations helps prevent the exploitation of their power profiles.
- **Leaky modules:** RTL modules prone to power leakage need to be identified using existing detection methods [27], [28]. These modules, which often handle sensitive variables or critical operations, contribute to power leakage due to bit-flipping and variable data patterns. We mitigate the risk of PSC attacks exploiting these power variations by securing these modules, such as the *SubBytes* module in AES.
- **Computationally intensive operations:** Even if not directly related to encryption, certain operations can still have a significant impact on the power profile of the

hardware. Variations in power consumption during these operations can unintentionally reveal details about underlying processes, leading to indirect data leakage. By safeguarding these operations, power consumption patterns are obscured, reducing the risk of attackers exploiting these variations.

- **High-fanout components:** Components with high fanout play a crucial role in shaping the overall power distribution of the circuit. Their strategic inclusion in our protection strategy comes from the understanding that a uniform power distribution complicates the execution of PSC attacks. Addressing these components helps in masking the overall power consumption patterns, thereby obfuscating any sensitive data from potential attackers.

The next step in PoSyn is preventing these identified components from being targeted by PSC attacks.

### B. Conversion to RTLIL Representation

Transitioning from RTL to RTLIL (Register Transfer Level Intermediate Language) is a crucial step in logic synthesis, facilitated by tools like Yosys [29]. RTLIL captures the design’s functionality and structural layout, detailing logic operations, data flows, and interconnections. This transformation breaks down RTL designs into fundamental elements such as cells, registers, and logic gates, enabling precise and direct mapping to physical components in the technology library cells while preserving design specifications. For instance, a simple half adder defined in RTL with *XOR* and *AND* operations for the sum and carry (Listing 1) is represented in RTLIL as *XOR* and *AND* cells within a module (Listing 2).

```

1 module half_adder (
2     input wire a, b,
3     output wire sum, carry);
4 assign sum = a ^ b; assign carry = a & b;
5 endmodule

```

Listing 1: RTL Code for half adder.

```

1 module \half_adder
2     wire width 1 input 0 \a, input 1 \b
3     wire width 1 output 2 \sum, output 3 \carry
4     cell $xor $xor_cell
5         parameter \A_SIGNED 0
6         parameter \B_SIGNED 0
7         parameter \Y_WIDTH 1
8         connect \A \a
9         connect \B \b
10        connect \Y \sum
11    end
12    cell $and $and_cell
13        parameter \A_SIGNED 0
14        parameter \B_SIGNED 0
15        parameter \Y_WIDTH 1
16        connect \A \a
17        connect \B \b
18        connect \Y \carry
19    end
20 end

```

Listing 2: RTLIL Code for half adder.

**Algorithm 1: Standard Cell Mapping (SCM)**


---

**Input:** *RTLIL\_Block*  
**Output:** *Optimal\_Cell\_Config*

- 1  $SC \leftarrow \text{Get\_Std\_Cell\_Lib}();$
- 2  $Func \leftarrow \text{Extract\_Func}(RTLIL\_Block);$
- 3  $Direct\_Matches \leftarrow \text{Find\_Direct}(Func, SC);$
- 4  $Indirect\_Comb \leftarrow [];$
- 5  $Comp\_Funcs \leftarrow \text{Decompose}(Func);$
- 6 **for**  $func \in Comp\_Funcs$  **do**
- 7      $Indirect\_Matches \leftarrow$   
        $\text{Explore\_Indirect}(func, SC);$
- 8      $Indirect\_Comb.append(Indirect\_Matches);$
- 9 **end**
- 10  $All\_Matches \leftarrow Direct\_Matches + Indirect\_Comb;$
- 11  $Optimal\_Config \leftarrow$   
      $\text{Simulated\_Annealing}(All\_Matches);$
- 12 **return**  $Optimal\_Config$

---

### C. Cell Selection from Technology Library

Once the RTL design is translated into RTLIL, the next step maps vulnerable components to standard cells from the technology library. In this section, we focus on generating all the possible cell combinations for RTLIL blocks of the vulnerable RTL components. Once the combinations are selected, we prioritize the combination, that minimizes PSC leakage, (further elaborated in Section IV-D). Algorithm 1 offers an overview of the approach for the selection of the combinations. Initially, the algorithm retrieves the available standard cells and extracts the specific functionality of the RTLIL block (lines 1-2). It then conducts a path exploration, searching for direct matches of the RTLIL cell functionality within the standard cell library (line 3). It also simultaneously decomposes the RTLIL cell functionality into simpler components to explore indirect mappings (lines 5-8). These potential solutions, encompassing both direct matches and indirect combinations, are then merged into a unified set (line 10). Finally, the algorithm employs simulated annealing to find an optimal solution from this set based on the defined criteria (line 11).

1) *Generating Combinations:* We employ a structured methodology to accurately translate the functionality of each RTLIL block by selecting standard cells from the given technology library. Our approach is twofold:

- **Direct Mapping:** Direct Mapping involves an immediate search for standard cells within the technology library that exactly match the functionality specified by the RTLIL block. For example, for an RTLIL block designed to execute a NAND operation, the step involves cataloging all NAND gate cells available in the standard cell library to ensure a broad spectrum of direct functional matches.
- **Indirect Mapping:** Indirect mapping decomposes the RTLIL block’s functionality into basic logical components, such as *AND*, *OR*, and *XOR* gates, and explores combinations of simpler standard cells to emulate the original functionality. This hierarchical decomposition and recombination process breaks down complex operations, like arithmetic or combinational logic, into man-

ageable elements for efficient mapping. By deconstructing these functionalities into their constituent parts, we increase the number of potential configurations. Each configuration represents a unique combination of simpler standard cells joined together to emulate the original, more complex function. This exploration spans various combinations of logical gates within the confines of the standard cell library, adhering to a well-defined threshold for combination exploration.

We set a threshold for exploring the various combinations to efficiently navigate the expansive search space while maintaining a balance between functional accuracy and area overhead. This threshold constrains the exploration to combinations that involve a limited number of standard cells, ensuring that area overhead and power leakage remain within acceptable bounds. This threshold is determined through an optimization process guided by simulated annealing, as detailed in Section IV-C2.

2) *Simulated Annealing to choose combinations:* In the optimization process for selecting standard cell combinations to accurately replicate RTLIL block functionalities, simulated annealing plays a crucial role in navigating the complex solution space [30], [31]. This method efficiently searches through a large set of potential combinations, identifying the ones that best fulfill the synthesis goals while managing complexity. By using a probabilistic technique, simulated annealing systematically refines the selection of standard cell combinations, ensuring that they match the desired functionality and adhere to constraints such as the number of cells used, which helps to control the circuit size and power consumption.

Simulated Annealing begins with an initial set of combinations, including both direct and indirect mappings of RTLIL blocks to corresponding standard cells. Through iterative exploration, simulated annealing generates “neighboring” solutions by slightly altering the current combination: adding, removing, or substituting cells. Each generated combination is evaluated against a threshold that limits the maximum allowable number of standard cells per combination. This threshold is either predefined based on area and power constraints or adaptively adjusted within the simulated annealing process to balance optimization efficiency with design feasibility. The criterion concerning the number of standard cells serves as a filter during this evaluation, disqualifying combinations that exceed the set limit. As the simulated annealing process progresses, the algorithm fine-tunes its exploration, increasingly favoring combinations that closely align with the optimization goals. It reduces the likelihood of accepting suboptimal combinations over time, thereby focusing the search on solutions that satisfy the criteria, including the limitation on the number of cells. The algorithm concludes with a selection of optimal combinations that replicate the intended RTLIL functionalities and also adhere to the design constraints imposed by area overhead and static power leakage. Following the selection of these combinations, we proceed to choose the optimal combination based on criteria that minimize the PSC leakage during synthesis. This has been elaborated subsequently in Section IV-D.

#### D. Optimal Bipartite Matching for PSC-Aware Mapping

Finally, with potentially multiple valid combinations for each RTLIL block, the process culminates in the selection of a single optimal mapping for each block. This selection is achieved by utilizing bipartite matching, which considers the entire set of RTLIL blocks and their corresponding valid combinations [32]. The goal is to find the best overall mapping that minimizes the circuit’s PSC leakage according to a set of predefined criteria while maintaining functional correctness. The following steps highlight the construction process of the Bipartite graph:

1. **Set Formation:** The bipartite graph is constructed with two distinct sets of vertices: one representing the RTLIL cells (Set 1) and the other representing the combinations of standard cells (Set 2).
2. **Edge Creation:** Edges are drawn between vertices from Set 1 to Set 2, where each edge represents a potential mapping from an RTLIL cell to a standard cell combination. Functional equivalence between the RTLIL cells and standard cell combinations is validated through graph isomorphism techniques to ensure feasibility of these connections.
3. **Cost function:** The cost function plays a pivotal role in determining the optimal mapping of RTLIL cells to standard cell combinations. This function is crafted from a mixture of criteria derived from the RTL design and standard cell library data, focusing on minimizing PSC leakage, described as follows:

1) *Factors Derived from Standard Cell Library:* From the standard cell library, we derive information crucial for mitigating PSC leakage:

- **Driving Strength (DS):** The driving strength of a cell influences the speed of signal transitions and peak current during switching. Higher driving strength can increase power signatures, making PSC attacks more effective, but it is necessary for high fanout cases to maintain integrity.
- **Capacitance (C):** The capacitance of a cell affects the power required for switching. Higher capacitance results in increased dynamic power consumption, thus raising the risk of power side-channel leakage. Reducing capacitance helps minimize power variations for cells involved in complex or frequent operations.

2) *Factors Influencing PSC Leakage from the RTL:* Power consumption leakage in digital circuits is influenced by multiple factors, including the nature of computations and the structural characteristics of the circuit components. These factors include:

- **Sensitive Variables (SV):** For RTLIL blocks containing sensitive variables, such as encryption keys, mapping these blocks to standard cells with lower driving strengths reduces PSC leakage risks by minimizing power variations. Lower driving strengths lead to slower transitions and reduced peak currents, thus diminishing detectable power signatures exploitable in attacks. However, it is essential to ensure that these driving strengths do not fall below a threshold that might compromise the circuit’s frequency or induce voltage drops causing functional issues. For example, in AES encryption, round keys are

mapped to cells with lower driving strengths, limiting power fluctuations while maintaining performance.

```

1 module SubBytes
2   wire width 8 input 0 data_in
3   wire width 8 output 1 data_out
4   memory 256 sbox
5   init sbox {
6     \63 \7c \77 \7b \f2 \6b \6f \c5 \30 \01
7     // Remaining S-box entries
8   }
9   assign data_out = sbox[data_in]
10  end
11 end

```

Listing 3: RTLIL Code with complex operation.

- **Intensive Operations (IO):** For RTLIL cells with complex and computationally intensive operations, the cost function favors mappings to cells with low capacitance and sufficient driving strength to balance power consumption and signal integrity. Lower capacitance minimizes dynamic power consumption, while higher driving strength ensures signal integrity during intricate operations, preventing delays that could increase static power consumption. For example, in AES the *SubBytes* step uses an *SBox* to perform non-linear byte substitution on the state matrix. The RTLIL code snippet in Listing 3 demonstrates the implementation of this *SubBytes* step, which leads to a peak in the power profile of the algorithm.
- **High Fanout Components (F):** For high-fanout components not directly involved in encryption, mapping to standard cells with higher driving strengths supports large loads efficiently, while introducing intentional variability in the power profile to obscure sensitive data processing. By selectively adjusting the power profile in these regions, PSC analysis can be misled, thereby reducing the risks of information leakage. Listing 4 provides an example of high-fanout operations in the AES *MixColumns*.

```

1 module MixColumns
2   wire width 8 input data_in0, data_in1,
3     data_in2, data_in3
4   wire width 8 output data_out0, data_out1,
5     data_out2, data_out3
6   assign data_out0 = data_in0 ^ data_in1;
7   assign data_out1 = data_in1 ^ data_in2;
8   assign data_out2 = data_in2 ^ data_in3;
9   assign data_out3 = data_in3 ^ data_in0;
10  end

```

Listing 4: RTLIL Code showcasing high fanout components.

3) *Derivation of the Cost Function:* To minimize PSC leakage, the cost function must account for relationships between these factors and their impact on power consumption.

The weighting factors  $\alpha, \beta, \gamma$  are introduced to appropriately scale the impact of each term based on its significance. These factors are determined through empirical methods such as grid search or optimization based on design-specific power and performance requirements. This flexibility allows the cost function to adapt to various design constraints, ensuring effective minimization of PSC leakage.

**Term #1: Sensitivity to Switching Activity and Driving Strength:** For an RTL component *A* with sensitive variables,

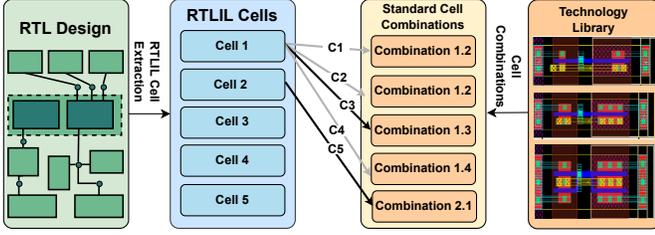


Fig. 2: Optimal Bipartite Matching: The grey lines from Set 1 (RTLIL cells) to Set 2 (standard cell combinations) indicate all possible mappings and the black lines indicate mappings with the lowest value of the cost function  $C$ . The optimal matching with the lowest  $C$  is selected for each vertex from Set 1.

the risk of leakage increases with higher switching activity. To mitigate this, we prioritize minimizing power variations by controlling the switching activity using driving strength  $DS$ . A higher switching activity can lead to more pronounced power fluctuations while  $DS$  influences the peak current during transitions. Therefore, we include a term inversely proportional to  $DS$  to prioritize higher driving strength cells, which can handle frequent transitions more effectively:

$$\text{Term \#1: } \frac{\alpha \cdot SV}{DS} \quad (1)$$

Here:

- $SV$  is a binary indicator (1 if the block contains sensitive variables, 0 otherwise).
- The term  $\frac{1}{DS}$  balances higher switching activity with the need for robust handling of frequent transitions.

### Term #2: Impact of Number of Operations and Capacitance:

For each RTL component  $A$ , the dynamic power consumption in its mapped standard cell  $S$  is directly influenced by the number of intensive operations performed and the capacitance  $C$ . As the number of operations increases, the switching activity  $SA$  also tends to increase because more signal transitions are likely to occur. Thus, the number of intensive operations can be considered a proxy for switching activity in the cost function.

We capture this relationship by including a term that combines the number of operations  $IO$  and the capacitance  $C$ . Higher capacitance results in increased dynamic power consumption, especially when the number of operations (and thus switching activity) is high:

$$\text{Term \#2: } \beta \cdot IO \cdot C \quad (2)$$

Here:

- $IO$  represents the number of intensive operations performed by the block, which correlates with higher switching activity.
- $C$  is the capacitance of the cell, affecting the power consumed during switching.

This term ensures that blocks performing a higher number of operations, especially those with higher capacitance, are accounted for in the cost function to reduce power consumption variations.

**Term #3: Influence of Fanout and Driving Strength:** For an RTL component  $A$  with high fanout, its mapped standard cell  $S$  requires adequate driving strength to drive large loads effectively. While higher driving strength can increase power consumption, it is necessary to ensure signal integrity when handling large fanouts.

$$\text{Term \#3: } \gamma \cdot F \cdot DS \quad (3)$$

Here:

- $F$  represents the fanout or the number of loads driven by the signal.
- $DS$  is the driving strength of the cell.

The term reflects the need to balance adequate signal strength with efficient power management for high fanout scenarios.

**Complete Cost Function:** Combining the derived terms, the overall cost function becomes:

$$C(A, S) = \sum_i \left( \alpha \cdot \frac{SV}{DS_i} + \beta \cdot IO \cdot C_i + \gamma \cdot F \cdot DS_i \right) \quad (4)$$

Here,  $C(A, S)$  denotes the cost of mapping the RTL component  $A$  to the standard cell  $S$  and  $i$  indexes the standard cells (in Set 2).

Minimizing this cost function leads to selecting standard cell combinations for every RTLIL block that achieves an optimal balance between ensuring functional accuracy and minimizing PSC leakage from the design, as shown in Figure 2. The cost function equation is designed to be universally applicable across all pairs of vertices in Set 1 (RTLIL blocks) and Set 2 (standard cell combinations) for the mapping process. **With the bipartite graph constructed and the cost function defined, an optimal bipartite matching algorithm, the Hungarian algorithm, is employed to find the set of mappings that minimizes the total cost [33].** This process ensures that each RTLIL cell is mapped to the most suitable standard cell combination, taking into account the sensitive nature of the variables involved, the computational demands of the design, and the overall goal of achieving a power side-channel resistant mapping.

Finally, when all the RTLIL cells have been mapped to a combination of standard cells, we obtain the PSC-resistant netlist components. The rest of the RTLIL cells (which were not identified as prone to PSC attacks) are synthesized with the usual criteria in order to optimize area, power, and performance. In this manner, the generated netlist is not only resistant to PSC attacks but also fulfills the optimization needed for the placement and routing step.

## V. POST-SYNTHESIS VERIFICATION FOR FUNCTIONAL CORRECTNESS

Once the PSC-resistant netlist is generated, we check its functional correctness using a Logic Equivalence Checker (LEC) tool, between the RTL design and the post-synthesis netlist generated using PoSyn. The LEC tool, Synopsys Formality, is used in standard industry hardware design flow, post logic-synthesis, to ensure the functional correctness of the netlist [34]. The tool evaluates whether the logical functions of the post-synthesis netlist are identical to those of the original

RTL, thus ensuring that the synthesis process has not altered the intended functionality. By confirming both structural and functional equivalence, we ensure that the generated netlist is the true functional equivalent of the RTL design.

## VI. THEORETICAL ANALYSIS OF POSYN'S SECURITY GUARANTEE

As mentioned in Section IV-D, PoSyn employs the Hungarian algorithm to derive an optimal mapping  $M$  that minimizes the cost function  $C$ , representing power consumption leakage associated with cryptographic operations. By leveraging this cost-minimization approach, PoSyn effectively reduces the quantity of side-channel information that can be inferred from power traces, thus improving the system's resilience to PSC attacks.

**Mutual Information and Leakage Reduction:** Inheriting the concept of mutual information gain from information theory, we evaluate how much knowledge about a cryptographic system can be extracted from observed side-channel leakage. In this scenario, the mutual information gain  $I(K, L)$  measures the dependency between the cryptographic key  $K$  and the leakage  $L$ . It is defined as:

$$I(K, L) = H(K) - H(K | L), \quad (5)$$

where:

- $H(K)$  is the entropy of the key, quantifying its initial uncertainty.
- $H(K | L)$  is the conditional entropy of the key given the observed leakage, quantifying the remaining uncertainty about the key after observing  $L$ .

The goal of PoSyn is to reduce the correlation between power consumption and the cryptographic key. This is achieved by minimizing the cost function  $C(M)$ , defined as:

$$C(M) = \sum_{(A,S)} C(A, S) \quad (6)$$

where  $C(A, S)$  denotes the cost of mapping the RTL component  $A$  to the standard cell  $S$ , as specified in Equation 4.

Since power side-channel leakage  $L$  arises from variations in power consumption across different operations, it depends on the underlying physical implementation of the design. Specifically, the structural mapping of RTL components to standard cells affects switching activity, driving strength, and capacitance, all of which influence power consumption. Therefore, we model leakage as a function of the total mapping cost:

$$L = f(C(M)) \quad (7)$$

where  $f(C(M))$  captures the dependency between the design's structural mapping and its corresponding power leakage characteristics.

After minimizing the cost function  $C(M)$  directly impacts the mutual information  $I(K, L)$  by controlling design aspects linked to power leakage. Based on derivation of Equation 4:

- Term #1 minimizes fluctuations by balancing switching activity with driving strength, reducing power variations

caused by sensitive operations, and decreasing the correlation between power traces and key-dependent activity.

- Term #2 addresses high-capacitance cells and intensive operations, suppressing dynamic power signatures that could otherwise reveal patterns linked to  $K$ .
- Term #3 optimizes driving strength in high-fanout cases, stabilizing power and preventing leaks that could indicate specific operational patterns.

Together, these terms increase the conditional entropy  $H(K|L)$ , effectively reducing  $I(K, L)$  and making the power traces more independent of the cryptographic key.

**Impact on Entropy and Mutual Information:** The Hungarian algorithm guarantees that the mapping  $M$  is optimal for minimizing  $C(M)$ , directly impacting  $I(K, L)$  by reducing power-based leakage sources. As a result of the targeted reductions in  $C(M)$ , PoSyn increases  $H(K | L)$  such that:

$$H(K|L) \rightarrow H(K) \Rightarrow I(K, L) \rightarrow 0 \quad (8)$$

Consequently, as  $I(K, L) \rightarrow 0$ , the side-channel leakage conveys negligible information regarding the cryptographic key, significantly enhancing resilience against power-based side-channel attacks.

By reasoning that  $I(K, L) \rightarrow 0$ , we establish that the leakage  $L$  exhibits near-independence from the secret cryptographic key  $K$ . This reduction in mutual information serves as a theoretical guarantee for the enhanced security of PoSyn, affirming the Hungarian algorithm's efficacy in constructing a mapping resistant to PSC attacks.

## VII. RESULTS

### A. Experimental Setup

We use cryptographic RTL design benchmarks to perform our side-channel aware synthesis and validate the efficacy of our approach. We evaluate PoSyn across three technology libraries: 65nm, 45nm, and 15nm [35], [36], [37]. We also utilize the LEC tool, Synopsys Formality, to perform post-synthesis verification on the synthesized netlists [34]. First, simulation is performed to collect power measurements for both the PoSyn-generated netlists and the conventional netlists synthesized by Yosys (since it is a widely used open-source synthesis tool for conventional synthesis). Please note that any other synthesis tool can also be used for the same. For each cryptographic hardware implementation, we simulated the design with a wide range of input values while collecting corresponding power measurements. These values were recorded under identical operational conditions for both the side-channel resistant and conventional netlists, ensuring a fair basis for comparison. The tools utilized for this process were Synopsys VCS for the Value Change Dump (VCD) file generation and PrimeTime for the power measurements. Two sets of experiments are presented in this section: (1) DPA attack, and (2) CPA attack. In each experiment, 4000 power traces were captured and analyzed.

1) *Benchmarks*: We conduct an evaluation of PoSyn on four distinct implementations of the AES algorithm, namely, *AES\_Comp*, *AES\_TBL*, *AES\_PPRM1*, and *AES\_PPRM3* [38]. We also evaluate our method on the encryption algorithms RSA and PRESENT, with *RSA1024\_RAM* and *PRESENT* benchmarks, respectively [38], [39]. Furthermore, we also evaluate two lattice-based Post Quantum Cryptography (PQC) algorithms: Saber and CRYSTALS-Kyber [40], [41].

2) *Metrics for Comparison*: To evaluate the effectiveness of PoSyn in mitigating PSC leakage, we consider the following two key metrics:

- **Success Rate**: The effectiveness of DPA and CPA attacks depends on their ability to accurately extract secret keys through power measurement analysis. In DPA, this involves generating key hypotheses, testing possible values, and refining them based on observed power consumption patterns. In CPA, each key guess is evaluated using a predictive model that estimates power consumption, typically leveraging the Hamming weight of data due to its strong correlation with power usage.

The success rate of these attacks serves as a crucial metric, offering a quantitative measure of their efficacy against cryptographic algorithms. Defined as the proportion of successful key extractions to the total attempts made, this metric is represented by Equation 9.

$$\text{Success Rate} = \frac{\text{No. of Successful Key Recoveries}}{\text{Total No. of Attack Attempts}} \quad (9)$$

Here, “Total No. of Attack Attempts” refers to the total number of attempts or iterations an attacker undertakes to successfully extract the secret key or significant portions of it from the cryptographic algorithm. In each iteration, the attack is conducted using a new set of power traces, and if the extracted candidate key matches the true key, that iteration is considered a success. This metric provides a statistical measure of how reliably an attack can recover the key across multiple independent attempts. A lower success rate indicates that the attacker needs many attempts to extract the key, while a higher success rate suggests that the key can be reliably recovered with fewer iterations.

- **Test Vector Leakage Assessment**: Test Vector Leakage Assessment (TVLA) is a statistical method used to evaluate the presence of power side-channel leakage in cryptographic implementations [42]. This metric provides a quantitative measure of how distinguishable power traces are when subjected to different input conditions, offering insights into the effectiveness of countermeasures against PSC attacks.

TVLA is conducted by comparing power traces obtained using fixed versus random input vectors. Welch’s t-test is applied to each time sample to calculate t-values, which indicate the level of statistical disparity between the two sets. If the cryptographic implementation exhibits no leakage, the distribution of t-values should be approximately normal under the null hypothesis [43].

Consequently, a standard threshold of  $\pm 4.5$  corresponds to an extremely low probability (on the order of 0.00001%) of observing such a value purely by chance [44]. This means that in the absence of leakage, the probability of a t-value exceeding  $\pm 4.5$  is about one in ten million. Any t-value surpassing this threshold is considered a strong indicator of exploitable leakage [45]. A lower proportion of t-values exceeding  $\pm 4.5$  suggests greater resilience against PSC attacks, whereas higher occurrences indicate statistical evidence of leakage.

Next, we proceed to calculate the success rates for the netlists synthesized by PoSyn (side-channel resistant) and the netlists generated by Yosys (conventional synthesis). A lower success rate indicates that the attack required numerous attempts to successfully extract the key. Following this, we present the TVLA results for all benchmarks across the three technology libraries. These results provide a statistical validation of power leakage, complementing the success rate analysis by quantifying how distinguishable the power traces are under different input conditions.

## B. Success Rate Results

Table II summarizes the success rates of DPA and CPA attacks for the various cryptographic benchmarks across three technology libraries: UMC 65nm, Nangate 45nm, and Nangate 15nm. For each benchmark, the table presents the DPA and CPA success rates for the side-channel resistant netlists generated by PoSyn, along with the percentage change in the synthesized netlist compared to the conventional netlist. It can be observed that across all three libraries, DPA has a lower success rate compared to CPA. In our evaluation of PoSyn for cryptographic cores, the success rate is influenced by the limited number of power measurements available for analysis. This constraint reflects real-world scenarios where attackers may face technical or access-related limitations. In DPA, having a limited number of power measurements makes it challenging to conduct the statistical analysis required to accurately distinguish the key-dependent variations from noise.

Moreover, Figures 3 and 4 illustrate the DPA and CPA success rates for PoSyn-generated side-channel resistant netlists and conventional netlists across all benchmarks for the three synthesis libraries. Each graph corresponds to a specific synthesis library, with the x-axis representing the benchmarks, while the y-axis shows the success rates of the attacks for both types of netlists.

1) **Results: UMC 65nm Library**: The UMC 65nm library is designed for advanced ASIC development utilizing UMC’s 65 nm Low-K Standard Performance SHVT process and offers 216 cell types with multiple drive strengths to support high-density IC designs [35].

The results shown for the 65nm library in Table II demonstrate that the side-channel resistant netlists achieve significantly lower success rates for both DPA and CPA attacks, with modest changes in the netlist structure, highlighting the effectiveness of PoSyn in enhancing security while maintaining design efficiency. Figures 3a and 4a showcase a comparison of success rates between the side-channel resistant netlist

TABLE II: Benchmark evaluation results on success rates of DPA and CPA for the encryption algorithms along with % change in netlist in terms of standard cells for three technology libraries.

Benchmark	65 nm			45 nm			15 nm		
	Success Rate		Change in Netlist	Success Rate		Change in Netlist	Success Rate		Change in Netlist
	Differential Power Analysis	Correlation Power Analysis		Differential Power Analysis	Correlation Power Analysis		Differential Power Analysis	Correlation Power Analysis	
AES_Comp	0.05	0.08	19%	0.04	0.08	21%	0.03	0.06	25%
AES_TBL	0.10	0.13	15%	0.09	0.16	17%	0.08	0.12	20%
AES_PPRM1	0.15	0.20	21%	0.12	0.18	24%	0.12	0.17	30%
AES_PPRM3	0.12	0.18	20%	0.10	0.18	22%	0.10	0.15	28%
RSA1024_RAM	0.16	0.21	25%	0.07	0.13	28%	0.12	0.18	35%
PRESENT	0.10	0.15	10%	0.08	0.10	12%	0.08	0.12	18%
SABER	0.09	0.12	27%	0.13	0.22	29%	0.08	0.10	33%
KYBER	0.14	0.18	22%	0.11	0.19	20%	0.10	0.15	30%

synthesized by our framework and the conventional netlist synthesized by Yosys. It can be observed that the success rate values are significantly lower for the side-channel resistant netlists generated by PoSyn across all benchmarks. For the *AES\_Comp* benchmark, the success rate for a DPA attack on the side-channel resistant netlist is remarkably low at 5%, compared to a much higher 65% success rate for the conventional netlist, achieving a maximum reduction in DPA success rate of up to 60% with only a 19% change in the netlist, which reflects the percentage of cells that differ from the conventional netlist, indicating the proportion of cell types that were either added or replaced in the transformed netlist.

Moreover, as seen in Figure 4a, when shifting to CPA, the success rate for the *AES\_Comp* benchmark slightly increases to 8% for the side-channel resistant netlist, yet it remains significantly lower than the 75% success rate observed for the conventional netlist. This increase in the success rate from DPA to CPA reflects the greater precision of CPA in exploiting correlations within power consumption data, yet it also highlights the effectiveness of the method in keeping the success rate low. The trend of increased success rates for CPA attacks in comparison to DPA attacks persists across all benchmarks. Nonetheless, the side-channel resistant netlists consistently exhibit substantially lower success rates than their conventional counterparts for both types of attacks. Even for benchmarks involving asymmetric cryptography algorithms such as *RSA\_1024* and PQC schemes like *SABER* and *KYBER*, the side-channel resistant netlists showcase a lower success rate. For instance, *RSA\_1024* benchmark shows a success rate of 16% for DPA and 21% for CPA, which are significantly lower than the 55% and 65% rates for the conventional netlist, respectively. Moreover, with a significantly high number of power measurements, the success rate of the conventional netlist without any countermeasures can go up to 100%. Lastly, it can be observed that PoSyn incurs at most 27% change in the netlist cells.

2) **Results: Nangate 45nm Open Cell Library:** The Nangate 45nm Open Cell Library, designed for educational and research use, offers a versatile set of 134 types of standard cells each with different driving strengths, that are crafted to align with the demands of the 45nm process [36].

Utilizing this library, PoSyn continues to demonstrate significant improvements in mitigating side-channel attacks across

various cryptographic benchmarks, as seen in Table II. The success rates for DPA and CPA on the side-channel resistant netlists are consistently lower than those for the conventional netlists, as seen in Figures 3b and 4b. **It can be observed that the *AES\_Comp* benchmark shows a DPA success rate of only 4% for the side-channel resistant netlist, compared to a much higher rate of 70% for the conventional netlist, resulting in a maximum reduction in DPA success rate of up to 66% with PoSyn. Similarly, the success rate for CPA on the *AES\_Comp* benchmark for the side-channel resistant netlist is 8%, which is still substantially lower than the 78% observed for the conventional netlist, achieving a maximum reduction in CPA success rate of up to 70%. This trend is consistent across all benchmarks. Additionally, the side-channel resistant netlists consistently exhibit a significant percentage change in netlist structure, with changes ranging from 12% to 29%, depending on the benchmark. Notably, benchmarks involving asymmetric cryptography algorithms such as *RSA\_1024* and post-quantum cryptography schemes like *SABER* and *KYBER* also demonstrate substantially lower success rates on the side-channel resistant netlists.**

3) **Results: NanGate 15nm Open Cell Library:** The NanGate 15nm Open Cell Library, developed with Silvaco and based on NCSU's FreePDK15, offers 76 digital cell types with varying drive strengths, supporting the demands of modern 15nm process technologies [37]. By utilizing 15nm technology, PoSyn has enhanced the resistance of cryptographic netlists to PSC attacks. The reduced transistor size minimizes power consumption and signal leakage, making DPA and CPA attacks more challenging. Table II reflects the efficacy of our side-channel resistant designs: the success rates for DPA and CPA are considerably lower than those of conventional netlists. **For example, in the *AES\_Comp* benchmark, the side-channel resistant netlist by PoSyn achieves a maximum reduction in success rates of 65% for DPA and 66% for CPA compared to the conventional netlist.** This notable reduction in success rates persists across various cryptographic algorithms, demonstrating our method's robustness in mitigating risks of key extraction. Other algorithms like *RSA\_1024* and post-quantum cryptography algorithms such as *SABER* and *KYBER* also showcase significant reductions in success rates, underscoring the benefits of our security enhancements at the 15nm scale. This has been illustrated in Figures 3c and

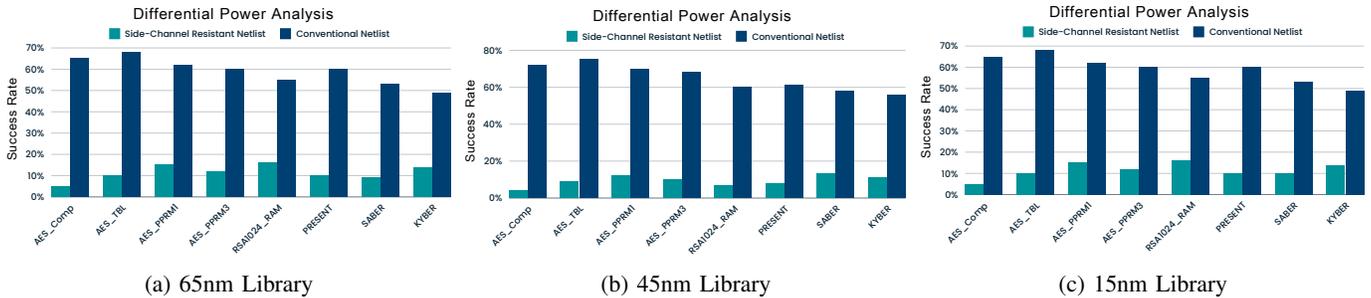


Fig. 3: DPA results for Side-Channel Resistant vs Conventional Netlist: Graphs showcase the success rates for the attack across all benchmarks for the three synthesis libraries.

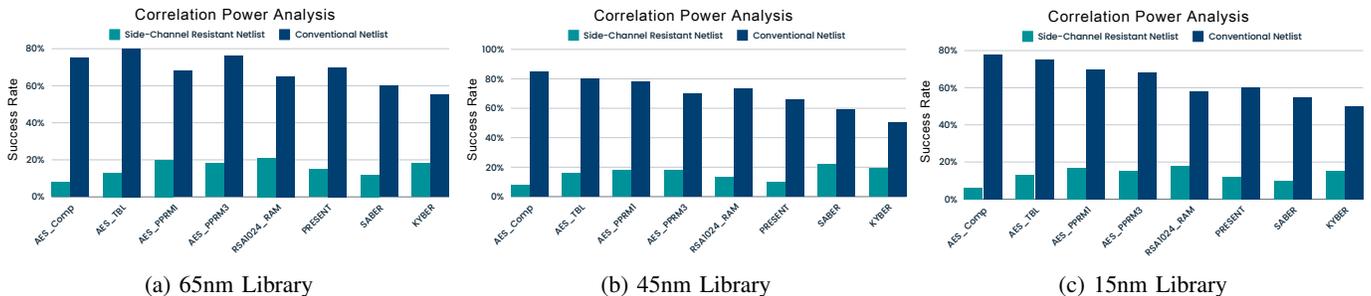


Fig. 4: CPA results for Side-Channel Resistant vs Conventional Netlist: Graphs showcase the success rates for the attack across all benchmarks for the three synthesis libraries.

4c showing Posyn’s effectiveness in significantly lowering the success rates of the attacks.

**Summary:** Older nodes are inherently more vulnerable due to higher power consumption, making them more susceptible to side-channel attacks. PoSyn mitigates these vulnerabilities effectively, achieving consistently low success rates with modest netlist changes across all benchmarks.

### C. Test Vector Leakage Assessment

Figure 5 illustrates the results of the Test Vector Leakage Assessment (TVLA) for multiple cryptographic benchmarks synthesized using conventional and PoSyn methodologies across the three technology libraries. The x-axis represents various benchmarks, grouped by technology node, while the y-axis shows the maximum absolute t-value observed for each benchmark-library combination.

It can be observed that the conventional netlists exhibit widespread leakage, with maximum absolute t-values consistently exceeding the standard threshold across all benchmarks and technology libraries, confirming their vulnerability to DPA and CPA attacks. In contrast, the PoSyn-generated netlists exhibit t-values below the standard threshold for all tested benchmarks and libraries, demonstrating significantly reduced PSC leakage.

These results indicate that PoSyn’s synthesis-level modifications successfully prevent data-dependent power variations, making cryptographic hardware more resilient against PSC

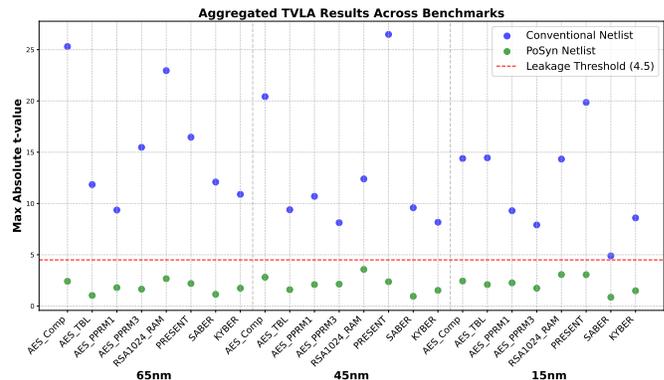


Fig. 5: Maximum absolute t-values from TVLA for cryptographic benchmarks across 65nm, 45nm, and 15nm nodes.

attacks. The effectiveness of PoSyn is consistent across multiple cryptographic algorithms including AES, RSA, PRESENT, SABER, and KYBER, as well as across different technology nodes, underscoring its robustness as a scalable and technology-independent technique.

### D. Timing and Memory Overheads

PoSyn’s standard cell mapping prioritizes PSC resistance over traditional metrics such as power, area, and performance while ensuring that timing constraints are met. To accommodate signal propagation delays introduced by security-driven transformations, The PoSyn generated netlist operates at a lower clock frequency while ensuring that paths meet setup and hold timing requirements within an extended clock period.

This approach maintains timing integrity while addressing performance overhead concerns. However, the reduction in clock frequency may decrease throughput, lowering the number of bits encrypted per second compared to the original design. This security-focused strategy enhances PSC resistance but introduces a trade-off, requiring users to balance security and performance.

Timing overhead, quantified as the percentage increase in critical path delay compared to the conventional netlist, is measured using OpenSTA. This reflects the reduction in maximum operating frequency due to PoSyn’s security-driven modifications. Additionally, while PoSyn does not explicitly optimize for memory efficiency, we assess its Memory Overhead to quantify the computational costs of storing transformed netlists. Here, Memory Overhead refers to the additional memory usage incurred by the synthesis tool when processing the PoSyn-modified design, ensuring that resource demands remain practical for larger circuits.

Table III summarizes both timing and memory overheads for benchmarks across the three standard cell libraries. Notably, PoSyn’s overhead remains minimal compared to existing masking or shuffling-based countermeasures, making it more efficient for protecting cryptographic cores.

**Summary:** PoSyn satisfies all the timing constraints without negative slack or timing violations, preserving the design’s timing integrity. The memory overhead introduced by PoSyn is minimal across all evaluated libraries, ensuring efficient resource utilization while enhancing PSC resistance.

TABLE III: Timing and Memory Overheads Across 15nm, 45nm, and 65nm Libraries.

Library	Benchmark	Overhead	
		Timing Overhead	Memory Overhead
15nm	AES_Comp	12%	10%
	AES_TBL	7%	5%
	AES_PPRM1	17%	13%
	AES_PPRM3	14%	7%
	RSA1024_RAM	22%	15%
	PRESENT	6%	4%
	SABER	16%	12%
	KYBER	13%	11%
45nm	AES_Comp	10%	8%
	AES_TBL	6%	5%
	AES_PPRM1	14%	12%
	AES_PPRM3	12%	8%
	RSA1024_RAM	18%	14%
	PRESENT	5%	4%
	SABER	13%	10%
	KYBER	10%	9%
65nm	AES_Comp	8%	7%
	AES_TBL	5%	4%
	AES_PPRM1	12%	10%
	AES_PPRM3	10%	7%
	RSA1024_RAM	16%	13%
	PRESENT	4%	3%
	SABER	11%	9%
	KYBER	9%	8%

### E. Comparison with Existing Countermeasures

Among widely adopted pre-silicon PSC countermeasures, masking and shuffling are the most common techniques. This section compares PoSyn with these methods, examining their effectiveness against PSC attacks.

1) *Masking Schemes:* Masked logic reduces data-dependent power consumption by dividing sensitive data into multiple independent shares [46]. Masking techniques often incur significant area and performance overhead due to the additional circuitry required. In contrast, PoSyn strategically modifies the synthesis process to enhance PSC resistance without extensive circuit modifications. To evaluate this, we implemented AES with first-order masking, using a masked *Sbox* to obscure the relationship between the cryptographic key and power consumption [47]. As shown in Table V, first-order masking reduces DPA and CPA success rates to 15% and 18%, respectively. PoSyn alone achieves a lower success rate of 3% for DPA and 6% for CPA. In contrast, a hybrid approach combining masking with PoSyn increases DPA and CPA success rates to 5% and 8%. These findings demonstrate PoSyn’s effectiveness in generating PSC-resistant netlists compared to masking or a hybrid approach.

Furthermore, we also evaluate the area overhead of various first-order masking schemes for AES using the 45nm library. This comparison highlights the silicon area utilization of different approaches. The area is quantified in gate equivalences (GE), normalized to the area of a 2-input NAND gate, which serves as the base unit in the given standard cell library and typically defines the technology-dependent unit area in contemporary CMOS technologies. Table IV illustrates the comparison results and consists of three columns: the first column indicates the approach evaluated, and the second column specifies a measure of the silicon area consumed by each scheme in kilo gate equivalences (kGE). Lastly, the third column quantifies the extent of area improvement PoSyn offers compared to the approach. The results are based on synthesized netlists for each scheme, revealing that PoSyn demonstrates the least area overhead of 4.51kGE and offers a significant improvement over other existing methods. This underscores the effectiveness of our proposed synthesis methodology, PoSyn, in overcoming the major limitation posed by masking.

TABLE IV: Comparison of implementation cost of existing masking schemes for AES.

Approach	Area in kGE	Improvement in Area
PoSyn	4.51	-
Changing of Guards [48]	13.66	3.02 ×
d + 1 Share Masking [49]	6.68	1.48 ×
Threshold Implementation [50]	8.12	1.80 ×
4-Share AES [51]	7.60	1.68 ×
2-Share AES [52]	7.71	1.70 ×
3-Share AES [53]	17.10	3.79 ×

Additionally, our results indicate that PoSyn achieves comparable area overheads of approximately 4.79 kGE and 5.22 kGE for 65nm and 15nm technology libraries, respectively.

2) *Shuffling Techniques:* Shuffling is another common PSC countermeasure that mitigates vulnerabilities by randomizing

TABLE V: Comparison with Masking and Shuffling Countermeasures on AES.

Success Rate	PoSSyn	First Order Masking	Shuffling
DPA	3%	15%	9%
CPA	6%	18%	11%

the order of operations, thereby reducing the correlation between power traces and sensitive data [54], [55]. To evaluate its effectiveness, we used power traces from the open-source ASCAD database, which includes traces from a shuffled AES implementation by ANSSI [56]. The results in Table V show that while shuffling offers some level of protection, PoSyn’s synthesis-level enhancements further reduce the success rates of DPA and CPA attacks, highlighting its added effectiveness. Due to limited access to open-source RTL implementations for shuffled AES, we were unable to measure the area overhead for the synthesis process.

**Summary:** PoSyn outperforms traditional countermeasures such as Masking and Shuffling, achieving significantly lower success rates for DPA and CPA attacks. Additionally, PoSyn incurs minimal area overhead, making it an efficient and robust solution for cryptographic hardware in high-security environments demanding strong protection against PSC attacks.

## VIII. CONCLUSION

This paper introduces PoSyn, a novel side-channel aware synthesis framework that enhances cryptographic hardware resistance against PSC attacks. PoSyn defines mapping criteria for synthesizing RTL designs to standard cell netlists, strategically integrating characteristics from both RTL designs and technology libraries. This approach preserves functional integrity while fortifying designs against PSC attacks. Additionally, PoSyn is theoretically proven to minimize mutual information leakage, further reinforcing its security against PSC vulnerabilities. The framework’s effectiveness has been validated across benchmarks, including AES, RSA, PRESENT, and post-quantum algorithms like Saber and CRYSTALS-Kyber. Tested on 65nm, 45nm, and 15nm nodes, PoSyn consistently reduces DPA and CPA success rates, achieving 3% and 6%, respectively, for 15nm nodes. Furthermore, TVLA analysis confirms that the PoSyn-synthesized netlists consistently maintain t-values well within the  $\pm 4.5$  threshold, thereby ensuring negligible side-channel leakage. PoSyn mitigates PSC vulnerabilities with minimal trade-offs: timing overheads up to 22% and memory overheads up to 15%. Compared to masking and shuffling techniques, PoSyn reduces CPA success rates and improves area efficiency by up to  $3.79\times$ . These results highlight PoSyn’s robustness, offering an effective solution for securing cryptographic devices in high-security applications such as network protocols and systems.

## IX. ACKNOWLEDGMENT

This research is partially supported by Technology Innovation Institute (TII), Abu Dhabi, UAE.

## REFERENCES

- [1] M. Randolph *et al.*, “Power side-channel attack analysis: A review of 20 years of study for the layman,” *Cryptography*, 2020.
- [2] N. P. Smart, “Physical side-channel attacks on cryptographic systems,” *Software Focus*, vol. 1, no. 2, pp. 6–13, 2000.
- [3] S. A. Huss, M. Stöttinger, and M. Zohner, “Amasive: an adaptable and modular autonomous side-channel vulnerability evaluation framework,” in *Number Theory and Cryptography: Papers in Honor of Johannes Buchmann on the Occasion of His 60th Birthday*. Springer, 2013, pp. 151–165.
- [4] X. Wang, S. Narasimhan, A. Krishna, and S. Bhunia, “Scare: Side-channel analysis based reverse engineering for post-silicon validation,” in *2012 25th International Conference on VLSI Design*. IEEE, 2012, pp. 304–309.
- [5] D. D. Hwang, K. Tiri, A. Hodjat, B.-C. Lai, S. Yang, P. Schaumont, and I. Verbauwhede, “Aes-based security coprocessor ic in 0.18-*muhboxm* cmos with resistance to differential power analysis side-channel attacks,” *IEEE Journal of Solid-State Circuits*, vol. 41, no. 4, pp. 781–792, 2006.
- [6] J.-M. Schmidt and C. H. Kim, “A probing attack on aes,” in *Information Security Applications: 9th International Workshop, WISA 2008, Jeju Island, Korea, September 23-25, 2008, Revised Selected Papers 9*. Springer, 2009, pp. 256–265.
- [7] G. Becker, J. Cooper, E. DeMulder, G. Goodwill, J. Jaffe, G. Kenworthy, T. Kouzminov, A. Leiserson, M. Marson, P. Rohatgi *et al.*, “Test vector leakage assessment (tvla) methodology in practice,” in *International Cryptographic Module Conference*, vol. 1001. sn, 2013, p. 13.
- [8] N. Gattu *et al.*, “Power side channel attack analysis and detection,” in *Proceedings of the 39th International Conference on Computer-Aided Design*, 2020.
- [9] M.-L. Akkar and C. Giraud, “An implementation of des and aes, secure against some attacks,” in *Cryptographic Hardware and Embedded Systems—CHES 2001: Third International Workshop Paris, France, May 14–16, 2001 Proceedings 3*. Springer, 2001, pp. 309–318.
- [10] J. Blömer *et al.*, “Provably secure masking of aes,” in *International workshop on selected areas in cryptography*. Springer, 2004.
- [11] J. D. Golić and C. Tymen, “Multiplicative masking and power analysis of aes,” in *Cryptographic Hardware and Embedded Systems—CHES 2002: 4th International Workshop Redwood Shores, CA, USA, August 13–15, 2002 Revised Papers 4*. Springer, 2003, pp. 198–212.
- [12] E. Oswald, S. Mangard, N. Pramstaller, and V. Rijmen, “A side-channel analysis resistant description of the aes s-box,” in *Fast Software Encryption: 12th International Workshop, FSE 2005, Paris, France, February 21–23, 2005, Revised Selected Papers 12*. Springer, 2005, pp. 413–423.
- [13] T. S. Messerges, “Securing the aes finalists against power analysis attacks,” in *International Workshop on Fast Software Encryption*. Springer, 2000, pp. 150–164.
- [14] S. Belaïd, V. Grosso, and F.-X. Standaert, “Masking and leakage-resilient primitives: One, the other (s) or both?” *Cryptography and Communications*, vol. 7, pp. 163–184, 2015.
- [15] J. Balasch, B. Gierlichs, V. Grosso, O. Reparaz, and F.-X. Standaert, “On the cost of lazy engineering for masked software implementations,” in *Smart Card Research and Advanced Applications: 13th International Conference, CARDIS 2014, Paris, France, November 5-7, 2014. Revised Selected Papers 13*. Springer, 2015, pp. 64–81.
- [16] M. Tempelmeier and G. Sigl, “Maskver: a tool helping designers detect flawed masking implementations,” in *2016 1st IEEE International Verification and Security Workshop (IVSW)*. IEEE, 2016, pp. 1–6.
- [17] T. Moos, A. Moradi, T. Schneider, and F.-X. Standaert, “Glitch-resistant masking revisited: Or why proofs in the robust probing model are needed,” *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 256–292, 2019.
- [18] D. Knichel, A. Moradi, N. Müller, and P. Sasdrich, “Automated generation of masked hardware,” *Cryptology ePrint Archive*, 2021.
- [19] S. Mangard, “A simple power-analysis (spa) attack on implementations of the aes key expansion,” in *Information Security and Cryptology—ICISC 2002: 5th International Conference Seoul, Korea, November 28–29, 2002 Revised Papers 5*. Springer, 2003, pp. 343–358.
- [20] P. Kocher, J. Jaffe, and B. Jun, “Differential power analysis,” in *Advances in Cryptology—CRYPTO’99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19*. Springer, 1999, pp. 388–397.
- [21] P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, “Introduction to differential power analysis,” *Journal of Cryptographic Engineering*, vol. 1, pp. 5–27, 2011.

- [22] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Cryptographic Hardware and Embedded Systems-CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings 6*. Springer, 2004, pp. 16–29.
- [23] X. Fang, P. Luo, Y. Fei, and M. Leeser, "Balance power leakage to fight against side-channel analysis at gate level in fpgas," in *2015 IEEE 26th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*. IEEE, 2015, pp. 154–155.
- [24] N. Veyrat-Charvillon, M. Medwed, S. Kerckhof, and F.-X. Standaert, "Shuffling against side-channel attacks: A comprehensive study with cautionary note," in *Advances in Cryptology-ASIACRYPT 2012: 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings 18*. Springer, 2012, pp. 740–757.
- [25] I. Damgård, Y. Ishai, and M. Krøigaard, "Perfectly secure multiparty computation and the computational overhead of cryptography," in *Annual international conference on the theory and applications of cryptographic techniques*. Springer, 2010, pp. 445–465.
- [26] M. Morrison and N. Ranganathan, "Synthesis of dual-rail adiabatic logic for low power security applications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 7, pp. 975–988, 2014.
- [27] M. He *et al.*, "Rtl-psc: Automated power side-channel leakage assessment at register-transfer level," in *IEEE VTS*, 2019.
- [28] A. Srivastava, S. Das, N. Choudhury, R. Psiakis, P. H. Silva, D. Pal, and K. Basu, "Scar: Power side-channel analysis at rtl level," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, pp. 1–14, 2024.
- [29] C. Wolf, "Yosys open synthesis suite," 2016.
- [30] D. Bertsimas and J. Tsitsiklis, "Simulated annealing," *Statistical science*, vol. 8, no. 1, pp. 10–15, 1993.
- [31] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [32] R. M. Karp, U. V. Vazirani, and V. V. Vazirani, "An optimal algorithm for on-line bipartite matching," in *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, 1990, pp. 352–358.
- [33] K. Fukuda and T. Matsui, "Finding all minimum-cost perfect matchings in bipartite graphs," *Networks*, vol. 22, no. 5, pp. 461–468, 1992.
- [34] S. Formality, "Equivalence checking using," 2010.
- [35] United Microelectronics Corporation (UMC), "55/65/90nm technologies," [https://www.umc.com/en/Product/technologies/Detail/55\\_65\\_90nm](https://www.umc.com/en/Product/technologies/Detail/55_65_90nm), accessed: 2024-04-29.
- [36] T. O. Project, "Openroad-flow-scripts: Nangate45 platform," <https://github.com/The-OpenROAD-Project/OpenROAD-flow-scripts/tree/master/flow/platforms/nangate45>, 2024, accessed: 2024-04-29.
- [37] M. Martins, J. M. Matos, R. P. Ribas, A. Reis, G. Schlinker, L. Rech, and J. Michelsen, "Open cell library in 15nm freepdk technology," in *Proceedings of the 2015 Symposium on International Symposium on Physical Design*, 2015, pp. 171–178.
- [38] "Verilog designs," <http://www.aoki.ecei.tohoku.ac.jp/crypto/web/cores.html>, 2024.
- [39] T. De Cnudde *et al.*, "Higher-order glitch resistant implementation of the present s-box," in *BalkanCryptSec*. Springer, 2015.
- [40] M. Imran *et al.*, "Design space exploration of saber in 65nm asic," in *Proceedings of the 5th Workshop on Attacks and Solutions in Hardware Security*, 2021.
- [41] F. Yaman *et al.*, "A hardware accelerator for polynomial multiplication operation of crystals-kyber pqc scheme," in *2021 IEEE DATE*.
- [42] A. A. Ding, L. Zhang, F. Durvaux, F.-X. Standaert, and Y. Fei, "Towards sound and optimal leakage detection procedure," in *Smart Card Research and Advanced Applications: 16th International Conference, CARDIS 2017, Lugano, Switzerland, November 13–15, 2017, Revised Selected Papers*. Springer, 2018, pp. 105–122.
- [43] A. A. Ding, C. Chen, and T. Eisenbarth, "Simpler, faster, and more robust t-test based leakage detection," in *Constructive Side-Channel Analysis and Secure Design: 7th International Workshop, COSADE 2016, Graz, Austria, April 14-15, 2016, Revised Selected Papers 7*. Springer, 2016, pp. 163–183.
- [44] T. Schneider and A. Moradi, "Leakage assessment methodology: Extended version," *Journal of Cryptographic Engineering*, vol. 6, pp. 85–99, 2016.
- [45] E. Ferruffino, L. Beckwith, A. Abdulgadir, and J.-P. Kaps, "Fobos 3: An open-source platform for side-channel analysis and benchmarking," in *Proceedings of the 2023 Workshop on Attacks and Solutions in Hardware Security*, 2023, pp. 5–14.
- [46] "A very compact "perfectly masked" s-box for aes," <https://faculty.nps.edu/drcanrig/pub/acns2008corr.pdf>.
- [47] A. R. Shahmirzadi, D. Božilov, and A. Moradi, "New first-order secure aes performance records," *Cryptology ePrint Archive*, 2021.
- [48] A. Askeland, S. Dhooghe, S. Nikova, V. Rijmen, and Z. Zhang, "Guarding the first order: The rise of aes maskings," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2022, pp. 103–122.
- [49] T. De Cnudde, O. Reparaz, B. Bilgin, S. Nikova, V. Nikov, and V. Rijmen, "Masking aes with shares in hardware," in *International Conference on Cryptographic Hardware and Embedded Systems*. Springer, 2016, pp. 194–212.
- [50] B. Bilgin, B. Gierlichs, S. Nikova, V. Nikov, and V. Rijmen, "Trade-offs for threshold implementations illustrated on aes," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 7, pp. 1188–1200, 2015.
- [51] F. Wegener and A. Moradi, "A first-order sca resistant aes without fresh randomness," in *Constructive Side-Channel Analysis and Secure Design: 9th International Workshop, COSADE 2018, Singapore, April 23–24, 2018, Proceedings 9*. Springer, 2018, pp. 245–262.
- [52] A. R. Shahmirzadi and A. Moradi, "Re-consolidating first-order masking schemes: Nullifying fresh randomness," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 305–342, 2021.
- [53] T. Sugawara, "3-share threshold implementation of aes s-box without fresh randomness," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 123–145, 2019.
- [54] A. Yahya, A. M. Abdalla, H. Arabnia, and K. Daimi, "An aes-based encryption algorithm with shuffling," in *Security and Management*, 2009, pp. 113–116.
- [55] Y. Wang and Y. Ha, "An area-efficient shuffling scheme for aes implementation on fpga," in *2013 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2013, pp. 2577–2580.
- [56] "Ascad: Anssi sca database," <https://github.com/ANSSI-FR/ASCAD>.