

Glitch Propagation through Flip-Flops Endangers Masking Schemes: Why Time Separation Is Required

Hasin Ishraq Reefat*, Mohammad Ebrahimabadi*, Sofiane Takarabt[‡], Sylvain Guilley[‡] and Naghmeh Karimi*.

*CSEE Department

[‡]Secure-IC S.A.S.

University of Maryland Baltimore County Technology & Strategy Division
Baltimore, US Paris, France

Abstract—Glitches are hardware-level hazards that are capable of compromising secure implementations. Even dominant protections against side-channel attacks must demonstrate immunity in the potential presence of glitches. In this paper, we study two hardware masking schemes rationales, namely Ishai-Shai-Wagner (ISW) and its Enhanced version (E-ISW), as well as Domain-Oriented Masking (DOM). While other glitch-aware masking schemes have been proposed, our focus is specifically on the differences between E-ISW and DOM. Those two styles rely respectively on *combinational* and on *sequential* separation of shares. It is known that sequential separation, realized through pipelining stages, does impact the latency of the hardware masking scheme. Additionally, in this paper, we show another drawback: pipelining does not provide full independence between manipulated shares. Indeed, we show that pipelining elements (DFFs in practice) can propagate upstream activity downstream. This results in first-order leakage in real-world systems, especially when parasitic effects are considered. In this respect, we show that DOM is leaking at first-order, and that this leakage increases with both the complexity of the netlist (in terms of number of DOM gadgets) and with the extent to which the operational environment can be worsened by an attacker (e.g., lowering the voltage to increase the leakage). These findings provide valuable insights for advancing secure hardware design.

I. INTRODUCTION

Side-Channel Analysis (SCA) attacks threaten any cryptographic devices. To protect against such attacks two main categories of countermeasures have been proposed in the literature: hiding and masking. The former mainly relies on equalizing the power consumption regardless of the computation being performed. Such schemes are mainly based on dual logic, and suffer from process variations as well as early propagation, e.g., in Wave Dynamic Differential Logic (WDDL) circuitries. In contrast, masking schemes rely on randomizing the computations to decorrelate the side-channel (e.g., power consumption) from the computation being performed thus concealing the secret key.

Masking is often advocated as a natural countermeasure against SCA attacks, as it is supported with formal security proofs. In particular, the complexity of an attack (measured by the number of traces required to extract the key) grows exponentially with the number of random shares [1]–[5]. However, for masking to deliver on its theoretical guarantees, certain assumptions must hold. Most importantly, the shares must remain independent. While this condition may be satisfied at the logical level during share generation and manipulation, such assumption may be violated at the physical level.

Indeed, prior work has shown that physical effects such as *crossstalk* and *IR drop* can undermine the independence assumption between shares [6], [7]. These issues are combinational in nature, violating the independence assumption at a specific point in time. Additionally, Sugawara *et al.* [8]

demonstrated first-order leakage that depends on static secret values. This is problematic, as such leakage should not occur in well-masked end-to-end circuits with complete masking (i.e., full randomization).

In this paper, we introduce another physical weakness that can jeopardize the security of masking schemes, namely the non-independence across pipelining stages (i.e., timing barriers or sequential recombination). In this respect, we contrast between E-ISW [9] and DOM [10], which are the current two contenders. Their respective pros and cons are summarized in Table I. In this table, the *critical clock period* refers to the combinational logic depth of the countermeasure, where the shorter the better, and *throughput* indicates the rate at which outputs are produced over time, while *latency* refers to the number of clock cycles required to obtain the result.

TABLE I: High-level comparison between E-ISW and DOM

Masking style	E-ISW	DOM
Critical clock	Bad	Good
Throughput	Bad	Good
Latency	Good	Bad
Shares separation	Combinational	Sequential

E-ISW implements combinational separation, whilst DOM implements sequential separation. In this article, we show that sequential separation actually is leaking more significantly than combinational separation.

Contributions: As an important contribution, in this paper we demonstrate the success of a first-order attack on DOM, even though DOM is generally considered secure against first-order attacks in the literature [11]. This vulnerability arises because the common assumption that DFFs completely stop glitches does not perfectly hold. Specifically, the high-level model considering that the DFF is opaque when the clock is not at a rising edge is true only in the absence of parasitics. In practice, with parasitics, any activity at the data input (D) of the DFF is propagated to the output (Q). Our characterization show that such transfer is more important when the clock signal (CLK) is low. Thereby even the supposed to be glitch resistant schemes (such as DOM) may leak in the simple probing model. Such findings show that the evaluation of glitch-resistant hardware-level schemes, such as E-ISW and DOM, must consider not only their security guarantees but also their impact on Power, Performance, and Area (PPA).

II. PRELIMINARY BACKGROUND ON MASKING SCHEMES

This section overviews masking schemes (including ISW, DOM, and E-ISW) used to implement the non-linear part of the encryption schemes (i.e., S-Box module). The masked AND gadgets of these schemes are shown in Fig. 1.

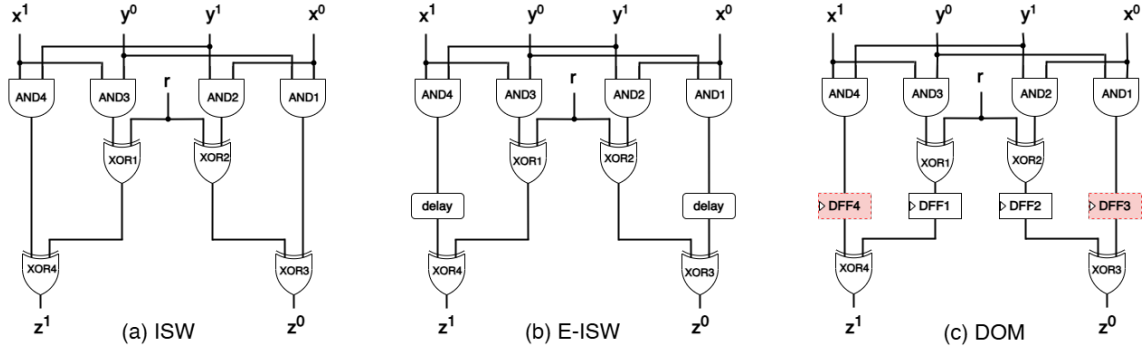


Fig. 1: ISW [1], E-ISW [9], and DOM [10] AND gadgets; DOM AND gadget can be implemented with 2 DFFs by removing the red DFFs and directly connecting the upper ANDs’ output to the bottom XORs

ISW: This protection was proposed by Ishai, Sahai, and Wagner [1] who initially referred to their scheme as “private circuits”. Later, as this seminal scheme gained traction and derivatives were proposed, the original scheme was simply referred to as “ISW” by the side-channel analysis community.

This scheme consists in computing on randomly shared data, and innovates in introducing the concept of *gadgets*. Since any vectorial Boolean function can be decomposed into polynomials [12, §2.2], the sufficient gadgets are addition and multiplication. In Boolean circuits, those operations are simply XOR and AND. In particular, in this architecture, the gadget implementing the AND gate requires a single bit of randomness, denoted as R . Assume that bit X is represented by a random sharing (X^0, X^1) such that $X = X^0 \oplus X^1$, and similarly bit Y is represented by a random sharing. The AND operation between X and Y , denoted as Z , is then computed according to the following equation:

$$\begin{cases} Z^0 = ((X^0 \cdot Y^1) \oplus R) \oplus (X^0 \cdot Y^0) \\ Z^1 = ((X^1 \cdot Y^0) \oplus R) \oplus (X^1 \cdot Y^1) \end{cases}$$

In these equations, the sequence of operations must respected the precedence of operators, including parentheses; therefore, the implementation must maintain the same gate order in the resulting netlist. It is worth noting that some optimizations of ISW proofs have been proposed, e.g., in [13]. Indeed, ISW assumes that its constitutive gates are synchronizing, which is not the case in practice. Thus, without surprise, ISW leaks in presence of glitches, as shown in [14]. Several glitch-resistant logic styles, such as DOM and Threshold Implementation (TI), have been proposed in literature to solve this problem. However, as we show in this paper even supposed-to-be first-order secure DOM is leaking.

DOM: Domain-Oriented Masking (DOM [10]) leverages pipelining barriers to prevent glitch propagation. In practice, D flip-flops (DFFs) are inserted only at sensitive places, namely where signals that depend on all input shares converge. The structure of DOM is organized into three layers [10, §3.2].

- Calculation (on shares);
- Resharing (adding fresh randomness within domains);
- Integration (coalescing shares to restore the same number of shares as in the input).

It is worth noting that in DOM, the addition of fresh randomness is referred to as “resharing” rather than “refreshing”. The latter term is instead used in the context of gadgets proved as Strong Non-Interfering (SNI), which are not required in DOM.

E-ISW: As mentioned earlier, ISW leaks in the presence of glitches. However, as shown in [9], ISW can be strengthened by inserting delays at strategic places, not to prevent glitches but rather to make them harmless from a security standpoint. In practice, these delays ensure that signals that could otherwise recombine in a manner violating the robust probing model (when the two paths together depend on all the shares) are safely separated. Such resulting logic style was referred to as Enhanced-ISW (E-ISW) in literature [9].

III. GLITCH PROPAGATION THROUGH DFF BARRIER

In this section, we analyze the behavior of a DFF design in the presence of glitches and assess the impact of the clock on masked circuits with respect to SCA leakage. We show that transitions occurring at the DFF’s input can still be observed – albeit with low amplitude – at the output, which is expected given the intrinsic imperfections of CMOS transistors.

A. DFF Implementation

We implemented a DFF using an industry standard library topology and simulated the circuit via Synopsys HSpice to study the propagation of glitches. The structure is shown in Fig. 2. The internal gating is controlled by complementary phases clk and n_clk generated locally. The model uses the library’s transistor-level devices and device ratios; parasitics internal to the cell are included by construction. The supply voltage was held at the nominal voltage of 1.1 V and the clock is a 50% duty-cycle rail-to-rail signal with nominal slew.

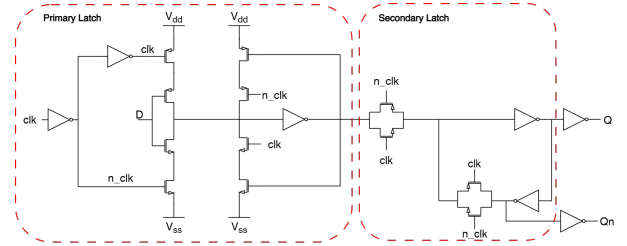


Fig. 2: Implemented DFF architecture

To excite glitches, we inject a narrow pulse on D by momentarily toggling it either $0 \rightarrow 1 \rightarrow 0$ (positive glitch) or $1 \rightarrow 0 \rightarrow 1$ (negative glitch). The pulse timing is swept in such a way that glitches occur (i) while primary latch is opaque, (ii) while it is transparent, and (iii) within the setup/hold aperture around the transfer edge.

B. Glitch Propagation Analysis Through DFF

Figure 3 exhibits four annotated time windows (Scenarios 1-4) that combine the pre-state of (D, Q) with the glitch polarity on D . To make the glitches visible, the bottom two traces zoom Q close to logic 0 and logic 1 levels which reveal millivolt-scale bumps/dips that are not obvious on the full-scale plot.

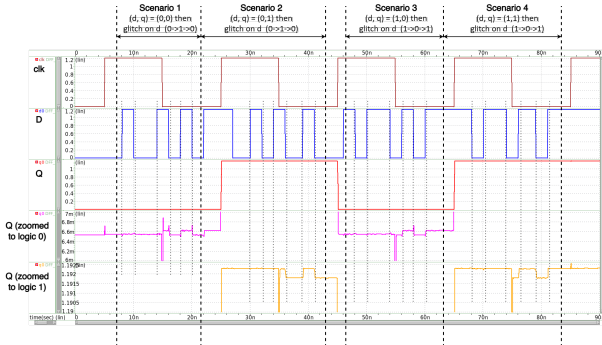


Fig. 3: Glitch propagation through DFF

Scenario 1 $(D, Q) = (0, 0)$ with a $0 \rightarrow 1 \rightarrow 0$ glitch on D : the glitch is almost filtered while $clk = 1$ (the primary latch is opaque) as the propagated glitch is barely visible which may appear in micro-volt range. However, while $clk = 0$ (the primary latch is transparent), small upward bumps are visible on the output Q (shown in Q (zoomed to logic 0) trace) while glitch is induced on input D . Additionally, an upward bump is visible right after a sharp dip when the glitch occurs during the clock falling edge (when the primary latch is opening, the secondary latch is closing).

Scenario 2 $(D, Q) = (0, 1)$ with a $0 \rightarrow 1 \rightarrow 0$ glitch: the stored logic 1 is preserved during $clk = 1$ (the primary latch is opaque) though the propagated glitches may appear in micro-volt range. However, the voltage level goes down slightly and small upward bumps are observed in Q (zoomed to logic 1) trace while $clk = 0$ (the primary latch is transparent). Unlike scenario 1, no sustained bump/dip is observed when the glitch occurs during the clock transition edge though there is a sharp dip exactly at the falling edge.

Scenario 3 $(D, Q) = (1, 0)$ with a $1 \rightarrow 0 \rightarrow 1$ glitch: the glitch is almost filtered while $clk = 1$ as previous scenarios. During $clk = 0$, the voltage level at Q goes up slightly and small downward dips are observed in terms of glitch propagation. For this case, no bump/dip is observed when the glitch coincides with the clock transition edge though a sharp dip is visible exactly at the falling edge.

Scenario 4 $(D, Q) = (1, 1)$ with a $1 \rightarrow 0 \rightarrow 1$ glitch: the latched logic 1 is maintained during $clk = 1$ as previous scenarios; during $clk = 0$, small downward dips are visible at Q . A downward dip is also observed right after a sharp dip when the glitch occurs during the clock falling edge (primary opening, secondary closing).

The takeaway point is that the DFF behaves as an effective glitch filter when the primary latch is opaque ($clk = 1$) as very less amount of glitch can be propagated but does not eliminate glitch propagation during $clk = 0$ (when primary latch is transparent).

Indeed the DFF consists of two main latches, primary and secondary. When clk is 1, the primary is closed (i.e.,

opaque), retaining the current stable data, and the secondary is opened (i.e., transparent). Here, all the transitions happening will be logically blocked by the primary. When clk is 0, the primary is opened, and the transitions are transmitted to the secondary (which is logically blocking). Thus, when clk is 0, the secondary is excited more than when clk is 1, which induces more leakage at the DFF output Q . These glitches are small in amplitude yet rich in dynamic/short-circuit energy and are precisely the kind of features that can reappear and be re-shaped by the first combinational stage after the register.

In our experiments, when the clock is low, propagation can be characterized by a transmission rate, equal to $(1.20 \text{ V})/(500 \mu\text{V}) = 1/2,400$. Although low, averaging over 2,400 traces yields the same leakage as an unmasked implementation. We also observe a pulse at the clock falling edge. Depending on DFF input D and output Q , it can produce a power spike of $\{2.6, 2.8, 3.0\}$ mV. Again, this minor variation, once averaged sufficiently, can reveal (D, Q) values, representing another leakage source.

In sum these observations show that DOM and other masking schemes that use the compression step (including some implementations of TI [15], [16]), fail to ensure security in the probing model.

C. Glitch Propagation Through XOR Gate After DFF

To check the glitch propagation through XOR gate after the DFF, we fed the DFF output Q into a 2-input XOR from the same library to see how a post-register gate reshapes the narrow glitches propagated through the DFF. The other XOR input (consider it as B) was fixed to a constant: first to logic 0 and then to logic 1. The stimulus on D and the clocking are identical to the DFF study. The plots shown in Fig. 4 are annotated with the same four windows (Scenarios 1-4). For both Q and xor_out , rail-zoomed traces expose millivolt-scale bumps/dips that are otherwise invisible.

Case $B = 0$ (Fig. 4a). Functionally $A \oplus 0 = A$, therefore xor_out follows Q . In the measurement, if we focus on the bottom two traces (XOR output zoomed to 1 and 0), we can see that the propagated glitches are present yet less pronounced than at the DFF output Q .

Case $B = 1$ (Fig. 4b). Functionally $A \oplus 1 = \bar{A}$, therefore xor_out is the inverted version of the polarity of the DFF output Q . Consistent with the vector-dependent conduction paths inside the XOR, scenarios where xor_out is at logic 1 (Scenarios 1 and 3) show very little visible propagation, whereas scenarios where xor_out is at logic 0 (Scenarios 2 and 4) show clear, time-aligned excursions that closely match those at the DFF output Q .

For both constants on the second XOR input ($B = 0$ or $B = 1$), glitches that coincide during clock falling edge (primary latch opening, secondary latch closing) propagated through the XOR gate and the propagated glitch is stronger when the XOR output is at logic 0 than when it is at 1.

The takeaway point is that a post-register XOR does not filter the propagated glitches; rather, it forwards them with small input vector-dependent changes in delay and amplitude. Although the excursions are less pronounced when the other input is 0, they are significant when the other input is 1.

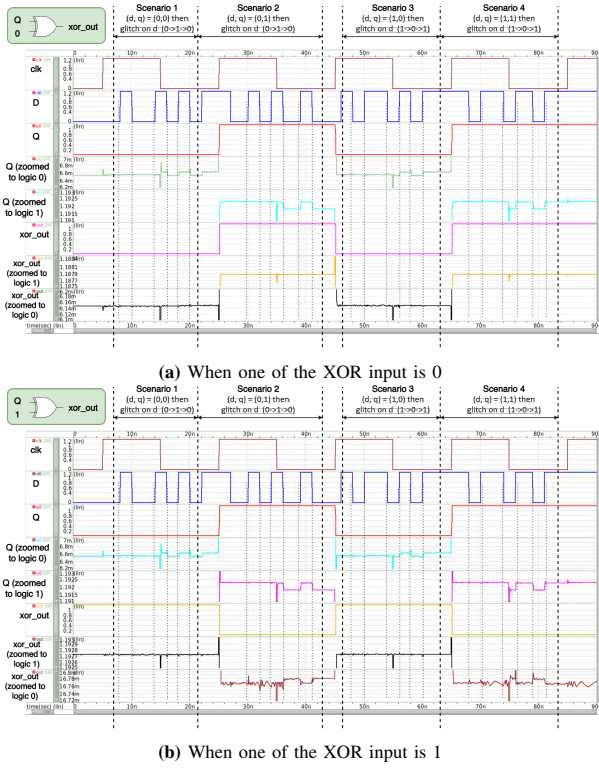


Fig. 4: Glitch propagation through DFF and XOR - Used for integration step.

IV. LEAKAGE ANALYSIS METRICS: WHY USING NICV

We first discuss the relevant choice for leakage detection metric. The International Standard ISO/IEC 17825 [17] presents two estimators in this respect:

- Student T-test, an independent two-sample t-test, for equal sample sizes and variance, applied to two groups (1, 2) [18], and
- Normalized Inter-Class Variance (NICV [19]).

T-test is used to compare the average of two samples. The computation consists of: $t = (\bar{X}_1 - \bar{X}_2) / (s_p \sqrt{2/n})$, where $\bar{X}_i, i \in \{1, 2\}$ is the average of either class, and where $s_p = \sqrt{\frac{s_{X_1}^2 + s_{X_2}^2}{2}}$. Here s_p is the pooled standard deviation for the same number of samples $n = n_1 = n_2$ in each class, and s_{X_i} is the unbiased estimator of the population variance.

Now, achieving a precise estimation of a statistical parameter (in this case, the difference of means) requires a sufficiently large sample size and reaching the 4.5 threshold (commonly adopted in the SCA context, [17, §8.4]) is not always feasible, even when the two samples are known to have different means. Besides, the T-test consists of analyzing whether a measured bias (measured by the difference $\bar{X}_1 - \bar{X}_2$) is significant relative to some variability (measured by s_p). The purpose of the test is to determine whether the bias is significant, i.e., whether a high T-test value is not merely due to estimation error. In this context, it is assumed that the variability arises from noise, and the T-test computes a ratio of inter-class variance to intra-class variance. As in our SPICE simulation there is no noise, only the inter-class variance matters.

In this respect, NICV is more suitable. Indeed, NICV consists of the normalized inter-class variance. Specifically, by the law of total variance, the variance can be decomposed

into inter- and intra-class components. Therefore, NICV, being the ratio between inter-class variance and total variance, is a unitless quantity with values in the interval $[0, 1]$.

If NICV detects leaks, that is $\text{NICV} \neq 0$, then it can be concluded that there is a first-order leakage. The first-order leakage is the most dreadful one, as:

- it is the easiest to exploit, in terms of attack complexity in number of traces (there is no noise amplification despite the masking scheme);
- no sophisticated attack is required: simple setups, such as correlation power analysis (CPA [20]), well-known and extensively described in classical manuals [21], [22].

V. LEAKAGE EXPLOITATION IN DOM

Building on our observation that the DFF is not fully opaque (Recall Sec. III), in this section we show that this leakage can be exploited in the DOM architecture, which is otherwise expected to provide first-order security.

Setup: We implemented the masked AND gadget for ISW, DOM (different architectures including 2 DFFs and 4 DFFs), and E-ISW [9] as shown in Fig. 1 at the transistor level using a 65 nm commercial technology. Transistor-level simulations were performed in Synopsys HSpice under typical conditions: temperature of 25°C, $V_{dd} = 1.1$ V.

To capture the power traces, each applied trace consists of two parts: an initial input, which intentionally sets the circuit to a known state, and a final input, during which the cryptographic circuit transitions from the initial state to the final value. The recorded power corresponding to this transition is used for analysis. Power samples are collected at a rate of 1 TSample/s. For the AND gadget simulations, both the initial and final inputs are exhaustively applied.

A. Leakage Comparison in Nominal Voltage

Figure 5 compares the three implementations of masked AND gate: ISW, E-ISW, and DOM (with 2 DFFs and with 4 DFFs) under identical acquisition conditions. Row (a) overlays average power traces, row (b) shows the class-wise mean power for four unmasked input classes $(x, y) = (0, 0), (0, 1), (1, 0), (1, 1)$, and row (c) depicts the NICV computed over those classes at nominal voltage supply.

For the ISW's AND gadget (Fig. 1(a)), although the class-wise mean power is small, the NICV exhibits distinct early peaks confined to a short interval. The class-wise means separate slightly in the same region. This behavior represents the glitch propagation and unbalanced routing, which enables information to leak even when shares are logically separated. For the E-ISW AND, the class-wise mean waveforms essentially overlap, and the NICV remains at the noise floor across the window. We observe no visible leakage within our measurement resolution at nominal voltage.

With DOM based on two registers, the mean power becomes more structured (activity concentrated around processing phases). The class-wise means nearly overlap, yet the NICV reveals a narrow, localized peak aligned with the principal switching event, indicating temporal alignment at register boundaries concentrates data-dependent current, making small leakage measurable even though the average behavior appears similar across classes. Furthermore, for DOM with 4 DFFs,

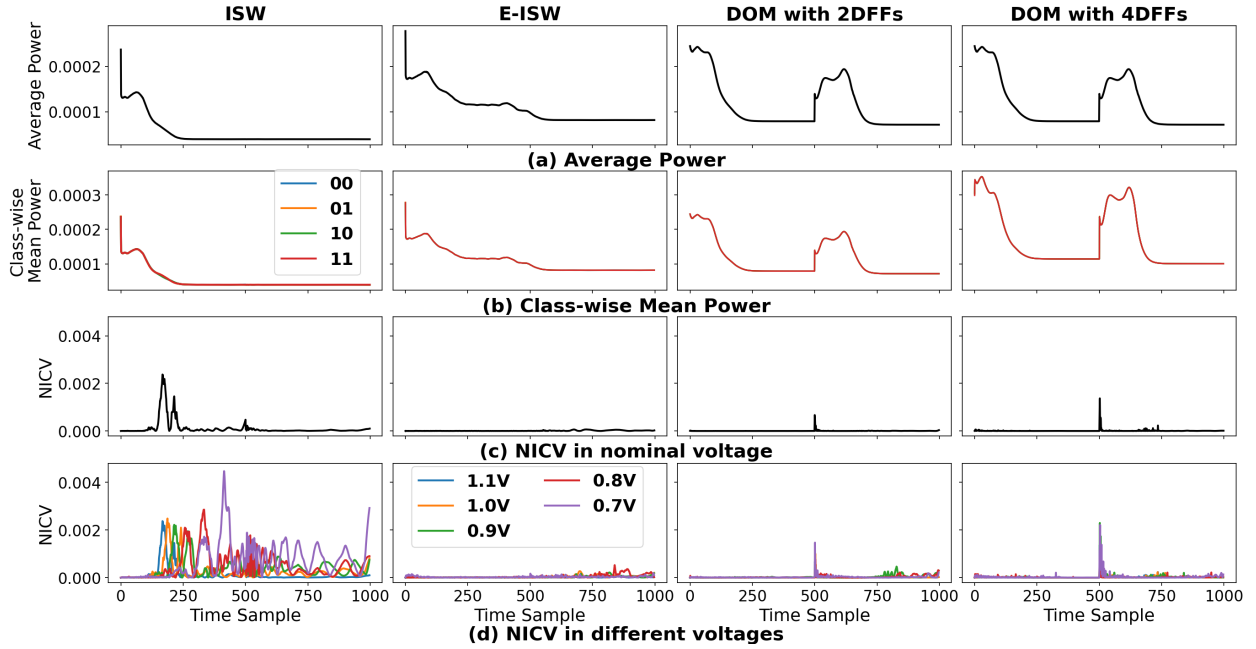


Fig. 5: (a) Average Power, (b) Class-wise Mean Power, (c) NICV plots in nominal voltage and (d) NICV plots in different voltages for ISW, E-ISW, DOM with 2DFFs and 4DFFs

the NICV peak becomes sharper and larger than with 2 DFFs, which indicates that additional registers increase the amplitude of a leakage spike at clock phases.

At nominal voltage, E-ISW AND shows no measurable leakage, while ISW leaks early and DOM leaks at register-aligned instants, with stronger leakage for 4 DFFs due to additional capture events increasing data-correlated switching at clock edges. In the case of the ISW, the leakage induced by glitches is quite significant. This leakage is maximal when it reveals exactly the secret value. When considering the case where a glitch arrives on the X^0 input, the dynamic power consumption c at the XOR gate calculating Z^0 can be modeled as the transition created by the signal X^0 .

$$Z^0 = (X^0 \cdot Y^1 \oplus R) \oplus (X^0 \cdot Y^0)$$

$$\tilde{Z}^0 = (\tilde{X}^0 \cdot Y^1 \oplus R) \oplus (\tilde{X}^0 \cdot Y^0)$$

$$c = Z^0 \oplus \tilde{Z}^0 = Y^1 \oplus Y^0 = Y$$

The same holds symmetrically for Z^1 and X . To maximize the observable leakage induced by glitches, we set up the same scenario for DOM to evaluate the leakage through DFF. The result is illustrated in Figure 9.

B. Effect of Power Supply Variation on the Leakage

In Fig. 5, row (d) displays the NICV across supply voltages (1.1 V \rightarrow 0.7 V) in 0.1 V steps for the targeted masked implementations of AND gadget. As shown, in ISW, multiple NICV lobes grow and shift slightly, widening the exploitable window as supply voltage decreases. E-ISW remains flat near nominal, but at 0.8 V a distinct, narrow NICV peak emerges, indicating that aggressive undervolting defeats its robustness by re-introducing potential glitch windows. For DOM with 2DFFs and 4DFFs, the dominant NICV feature is a single spike tightly aligned with the register capture event which strengthens as the supply voltage decreases, with the largest

amplitude at 0.7 V. The fully pipelined version (with 4 DFFs) consistently shows the most pronounced spike.

The takeaway point is that undervolting amplifies leakage relative to the noise. In ISW, multiple leakage lobes appear, and DOM spikes intensify markedly, while E-ISW shows no significant leakage. This is because, in E-ISW, decreasing the voltage does not compromise separation: both shares remain isolated in two distinct time samples.

C. Effect of Cascading DOM AND Gadgets on the Leakage

To further verify that the upstream glitch can propagate and appear as downstream glitch in the deeper logic, as shown in Fig. 6, we implemented an 8-input AND gadget using a two-stage cascading structure with each 4-input AND gadget fully pipelined (4 DFFs version shown in Fig. 1(c); inside each 4-input AND gadget is not shown here). In this experiment, inputs are applied over 16 possible classes.

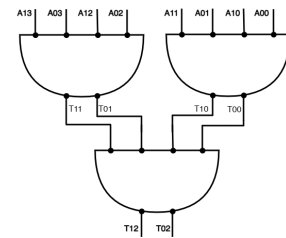


Fig. 6: An 8-input DOM AND gadget using a cascaded implementation

Figure 7 shows the class-wise mean power and NICV over those classes. The class-wise mean power traces largely overlap but still there are some localized mismatches among the classes which are clearly visible in the NICV plot. It is visible that there are some early small NICV peaks (at \approx sample 600) that correspond to localized switching in the first stage. A dominant NICV spike, at \approx sample 2000, is aligned with negative edge of the clock. Additional small peaks later in the window reflect further glitch availability on second stage.

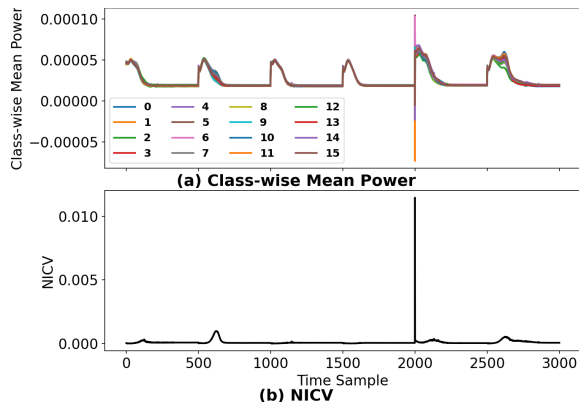


Fig. 7: (a) Class-wise Mean power and (b) NICV for Fig. 6 circuit

D. Reasons Behind the Leakage in Transition Edges of Clock

As we have observed leakages during both the falling and rising edges of the clock, we traced the signal from downstream to upstream components in the DOM AND gadget (Fig. 1(c)) to identify the cause. Fig. 8 shows the primary inputs X^0 and Y^1 , which are unaffected by the circuit’s electrical state, and the output of the $AND2$ gate T^{01} , where $T^{01} = X^0 \cdot Y^1$. Upon close inspection, two distinct variations are clearly visible at samples 1500 and 4500, corresponding to the falling and rising edges of the clock. Thus, we infer that T^{01} is affected by clock activity, and interestingly, this impact differs depending on whether the value is logical 0 or logical 1. Since the output of these gates depends on a single share, the effect also depends on that share. Additionally, the DFF triggers the XOR gate behind it, which in turn depends on the second share ($X^0 \cdot Y^0$) stored in the other DFF. Hence, the influence of the falling and rising edges depends on both shares of the secret value Y , explaining the leakage detected by the NICV. This leakage can be observed in almost any gate of the DOM circuits. We also note that we intentionally supplied independent V_{dd} to each component, so this leakage is not due to combined leakage through a single supply voltage, as is typically the case in real circuits. Naturally, in practical scenarios, leakage would be even more pronounced as all gates manipulating different shares are connected to the same V_{dd} .

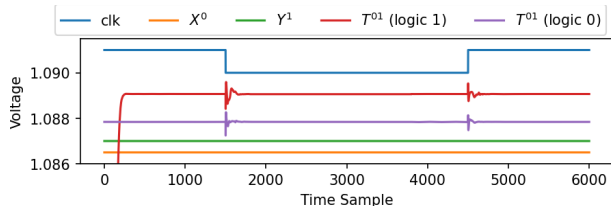


Fig. 8: Input X^0 , Y^1 and $AND2$ gate output T^{01}

E. Effect of Glitch Propagation Induced in the Middle of Clock

As shown in Sec. III, glitch propagation is stronger during $clk = 0$ than $clk = 1$. To observe this in the DOM AND gadget, we induced glitches through a primary input after the DOM DFFs sampled the inputs. Fig. 9 shows the clock, mean trace of input X^0 (where glitches were applied), and the NICV of the XOR gate ($XOR3$ in Fig. 1(c)) producing output Z^0 . Data was loaded at time 0, with glitches applied around sample 6000 ($clk = 1$) and 9000 ($clk = 0$). The NICV plot clearly shows that leakage at sample 9000 is stronger than at 6000, confirming that glitches propagate through the logic and DFFs

to the final XOR, consistent with the observations in Sec. III where post-register gates preserve and slightly reshape spikes.

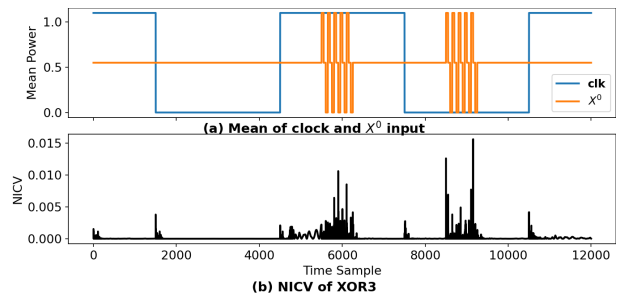


Fig. 9: NICV of final XOR (generating Z^0) in DOM AND gadget when glitch is induced on X^0

VI. DISCUSSION

Transposition to FPGA as a target: This article presents figures for ASIC. In FPGA targets, glitches remain a concern, but the circuit structure differs: combinational logic is implemented with Look-Up Tables (LUTs), and DFFs are built-in components. Since power is dominated by LUTs, other leakage mechanisms (e.g., coupling through V_{dd}) are more prominent than glitch propagation through DFFs. We leave this for future.

Time-sharing Masking (TSM): We underline that time separation prevents two shares from being recombined by glitches. This concept was recently applied in a masking implementation called TSM [23].

Higher-order masking: Our finding (glitch propagation through DFFs) can also undermine the security of higher order masking. Indeed, such recombination can reduce the security level from any order $d > 1$ to order 1. The amount of leakage might be small though as glitches do not occur for all data combinations and glitch amplitude is lower than that of nominal signals.

VII. CONCLUSION AND PERSPECTIVES

In practice, resisting against shares recombination by glitches requires specific care. Accordingly, state-of-the-art masking leverages separation. This paper compared two modes of separation: *combinational* vs *sequential*, and showed that sequential separation, in the context of real-world technology, does not fully compartmentalize shares. Indeed, parasitic capacitances allow glitches to traverse a DFF even when triggering clock signal is steady. Those parasitic effects cannot be ignored in critical applications where maximum security is required, as the probing model is used as the baseline for formal proofs.

In practice, the leakage through DFF is amplified if 1) the attacker can manipulate the Process-Voltage-Temperature (PVT) to exacerbate the effect of parasitics over functional signal propagation; and 2) the design has deep depth (e.g., AND-tree). Even though the leakage might be small in value, it is still dangerous as it is a first order leakage, thereby defeating masking even at higher order. Thus, protections, along the lines of TSM, or leveraging a double-pipelining, could be envisioned when security level must be improved.

ACKNOWLEDGEMENTS

This work was supported in part by the National Science Foundation CAREER Award (NSF CNS-1943224).

REFERENCES

- [1] Y. Ishai, A. Sahai, and D. Wagner, "Private Circuits: Securing Hardware against Probing Attacks," in *CRYPTO*, vol. 2729. Springer, 2003, pp. 463–481.
- [2] A. Duc, S. Faust, and F. Standaert, "Making Masking Security Proofs Concrete - Or How to Evaluate the Security of Any Leaking Device," in *Advances in Cryptology - EUROCRYPT*, 2015, pp. 401–429.
- [3] M. T. H. Anik, H. I. Reefat, W. Cheng, J.-L. Danger, S. Guilley, and N. Karimi, "Multi-modal pre-silicon evaluation of hardware masking styles," *Journal of Electronic Testing, Theory and Applications (JETTA)*, vol. 40, no. 6, pp. 723–740, 2024.
- [4] J. Bahrami, M. Ebrahimabadi, J.-L. Danger, S. Guilley, and N. Karimi, "Leakage power analysis in different s-box masking protection schemes," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2022, pp. 1263–1268.
- [5] M. T. H. Anik, J.-L. Danger, S. Guilley, and N. Karimi, "On the resiliency of protected masked s-boxes against template attack in the presence of temperature and aging misalignments," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems (TVLSI)*, vol. 32, no. 5, pp. 911–924, 2024.
- [6] T. D. Cnudde, B. Bilgin, B. Gierlichs, V. Nikov, S. Nikova, and V. Rijmen, "Does Coupling Affect the Security of Masked Implementations?" in *Constructive Side-Channel Analysis and Secure Design (COSADE) 2017*, pp. 1–18.
- [7] J.-L. Danger, S. Guilley, A. Heuser, A. Legay, and M. Tang, "Physical Security Versus Masking Schemes," in *Cyber-Physical Systems Security*, 2018, pp. 269–284.
- [8] T. Sugawara, D. Suzuki, M. Saeki, M. Shiozaki, and T. Fujino, "On measurable side-channel leaks inside ASIC design primitives," *J. Cryptographic Engineering*, vol. 4, no. 1, pp. 59–73, 2014.
- [9] S. Takarabt, J. Bahrami, M. Ebrahimabadi, S. Guilley, and N. Karimi, "Securing ISW Masking Scheme Against Glitches," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2024, pp. 1–2.
- [10] H. Groß, S. Mangard, and T. Korak, "Domain-Oriented Masking: Compact Masked Hardware Implementations with Arbitrary Protection Order," in *Workshop on Theory of Implementation Security (TIS@CCS)*, 2016, p. 3.
- [11] A. Saxena, M. Kaur, H. Pahuja, and V. A. Chhabra, "Design and performance analysis of cmos based d flip-flop using low power techniques," *Int'l Journal of Research in Electronics and Computer Engineering*, vol. 4, no. 5, 2017.
- [12] C. Carlet, "Vectorial Boolean Functions for Cryptography," in *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*. Cambridge University Press, Y. Crama and P. Hammer eds, 2010, pp. 398–469.
- [13] B. Wang, J. Zhang, Y. Yu, and W. Wang, "Tighter Security Notions for a Modular Approach to Private Circuits," in *EUROCRYPT*, vol. 15608, 2025, pp. 124–152.
- [14] D. B. Roy, S. Bhasin, S. Guilley, J.-L. Danger, and D. Mukhopadhyay, "From theory to practice of private circuit: A cautionary note," in *Int'l Conf. on Computer Design (ICCD)*, 2015, pp. 296–303.
- [15] S. Nikova, C. Rechberger, and V. Rijmen, "Threshold Implementations Against Side-Channel Attacks and Glitches," in *ICICS*, vol. 4307. Springer, 2006, pp. 529–545.
- [16] H. I. Reefat, H. Pourmehrani, W. Cheng, C. Carlet, A. Daif, C. Tavernier, S. Guilley, and N. Karimi, "Cbm-ti: Code-based masking against glitches by hybridization with threshold implementation," in *VLSI Test Symposium (VTS)*, 2025, pp. 1–11.
- [17] ISO/IEC JTC 1/SC 27/WG 3, "ISO/IEC 17825:2024. Information technology – Security techniques – Testing methods for the mitigation of non-invasive attack classes against cryptographic modules," p. 38, 2024.
- [18] F. Standaert, "How (Not) to Use Welch's T-Test in Side-Channel Security Evaluations," in *Smart Card Research and Advanced Application Conference (CARDIS)*, 2018, pp. 65–79.
- [19] S. Bhasin, J.-L. Danger, S. Guilley, and Z. Najm, "Side-channel Leakage and Trace Compression Using Normalized Inter-class Variance," in *Hardware and Architectural Support for Security and Privacy (HASP)*, 2014, pp. 7:1–7:9.
- [20] É. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Cryptographic Hardware and Embedded Systems (CHES)*, 2004, pp. 16–29.
- [21] S. Mangard, E. Oswald, and T. Popp, *Power analysis attacks - revealing the secrets of smart cards*. Springer, 2007.
- [22] M. Ouladj and S. Guilley, *Side-Channel Analysis of Embedded Systems - An Efficient Algorithmic Approach*. Springer, 2021.
- [23] D. K. S. Veeraraghavan, S. Dhooghe, J. Balasch, B. Gierlichs, and I. Verbauwhede, "Higher-Order Time Sharing Masking," *IACR Trans. on Cryptographic Hardware and Embedded Systems (CHES)*, vol. 2025, no. 2, pp. 235–267, 2025.