GlitchSnipe: Toward Localized Voltage Fault Attacks

Fatemeh Khojasteh Dana[1], Saleh Khalaj Monfared[1], Hamed Okhravi[2] and Shahin Tajik[1]

[1] Worcester Polytechnic Institute, fdana@wpi.edu,skmonfared@wpi.edu,stajik@wpi.edu
[2] MIT Lincoln Laboratory, hamed.okhravi@ll.mit.edu

Abstract. Voltage glitching is one of the most prominent fault injection techniques due to its effectiveness and simplicity. Although it is generally regarded as a spatially global fault method, in which the injected glitch uniformly affects all circuits on the die, several studies have observed that specific locations may be affected more than others. To characterize this phenomenon, we draw inspiration from methods used in electromagnetic interference (EMI) analysis. In this paper, we demonstrate that voltage attacks can be modeled as the transfer of conducted electromagnetic energy through the power delivery network (PDN) to the chip's die. By analyzing voltage glitches in the frequency domain and modeling the PDN as a communication channel, we demonstrate that different frequency components of an injected glitch signal propagate through the network in distinct patterns. In this context, we further show that modulating the supply voltage with a single-frequency sinusoidal signal, rather than injecting a pulse-shaped glitch, enables an adversary to influence transistors in specific regions of the chip and thus induce localized faults. To validate these claims, we first propose a post-silicon profiling framework that identifies the frequency bands in which the system's PDN is most vulnerable and maps the spatial regions of the chip affected by each frequency component. To this end, we perform extensive profiling on several FPGAs using distributed time-to-digital converters (TDCs) to measure the impact of injected signals across a range of frequencies. As a proof-of-concept, we also demonstrate successful localized voltage attacks on simple FSMs and AES-128 implementations with various placements, to further show the sensitivity of chip locations to injected energy at different frequencies. Our results reveal that even minor changes in design placement can significantly affect a circuit's susceptibility to voltage-based fault attacks, either weakening or strengthening its resilience.

Keywords: Conducted Electromagnetic Propagation · Power Delivery Network · Time-to-Digital Converter · Voltage Fault Attacks · Voltage Glitching

1 Introduction

The security and integrity of microprocessors, system-on-chips (SoCs), and cryptographic hardware can be undermined by various physical attacks, including side-channel, fault injection, and probing attacks. Among these, fault injection attacks are potent due to their active nature. By deliberately altering parameters, such as the supply voltage, clock frequency, or temperature, an attacker can induce faulty behavior in the target device, potentially compromising its operation or revealing sensitive information.

DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited.

Voltage or power supply glitching is one of the most fundamental and practical methods of fault injection. Since its early documentation in the academic literature almost three decades ago [AK96, KK99], it has been deployed in many real-world scenarios to break the security of the most advanced chips [BJKS21, BKGS21, SMS23, KWJS25]. Despite its success, voltage glitching still requires considerable experimentation to be effective. One reason for such trial-and-error tuning is to get the timing of the faults right so they can cause sufficient timing violations to bypass a specific state at a particular point in time. Hence, considerable effort is devoted to determining the optimal shape and timing of the glitch to maximize the attack success rate [BFP19, LI23, MS19].

While the timing aspect of voltage fault attacks has been studied in the literature, their spatial impact across various locations on a chip has been less explored. In other words, voltage glitching is considered a globalized fault attack that affects multiple circuits simultaneously connected to the same power domain. Recent work [KGT20, YKS+23] has investigated voltage coupling effects within FPGAs to evaluate the effectiveness of both remote side-channel and fault attacks. These studies show that voltage fluctuations induced by an IP core (acting as either a victim or an aggressor) have a spatially non-uniform impact across the FPGA die. However, the underlying reasons why certain regions are more susceptible to such disturbances remain unclear. Moreover, it is unknown whether similar non-uniformity arises when disturbances are introduced by external sources, as in classical voltage-glitching attacks. Based on the observed results in the literature, it also appears that process variation [KGT18, PHT20, ATG+19, FHO21, SWO+24] and environmental conditions [FH21] could cause very different spatial fault impacts on samples even from the same family.

Therefore, we ask the following research questions: *(1) Why do various locations on the chip react differently to voltage fault attacks? (2) Can we exploit such location dependencies to perform non-invasive localized voltage fault attacks?*

Our Contribution. To better characterize inconsistencies in the physical response of different chip regions to voltage glitching, we draw on established concepts from electromagnetic interference (EMI) analysis. In this framework, the system's power delivery network (PDN), including the chip, package, and PCB, is modeled as a communication channel with a distinct frequency transfer function, while a voltage glitch is modeled as the transfer of electromagnetic energy from the power supply to the transistors on the die. Moreover, we argue that this communication channel, essentially a three-dimensional transmission line, is analogous to a wireless channel, as it exhibits a non-uniform spatial frequency response across the chip's PDN. Consequently, when the chip is subjected to a pulse-shaped glitch in the time domain, various frequency components of the signal propagate differently through the PDN, resulting in non-uniform fault effects across the die. To validate our claims, we shift our analysis from the time domain to the frequency domain. Instead of directly applying pulse-shaped glitches (which contain a wide range of frequencies), we decompose the glitch waveform and examine the system's response to voltage modulations at individual frequencies. In this context, we introduce a post-silicon profiling framework that (i) identifies the frequency bands to which the system's PDN is most susceptible, and (ii) maps the spatial regions of the chip influenced by each frequency component. For the spatial profiling, we deploy an array of voltage sensors implemented as time-to-digital converters (TDCs) across the FPGA to capture localized voltage variations under different modulation frequencies. Our results show that frequency-domain voltage glitching not only provides deeper insight into the spatial propagation of faults but also enables an adversary to induce localized faults by carefully selecting the modulation frequency. First, we demonstrate proof-of-concept attacks using finite-state machines as target circuits. Second, by mounting a differential fault analysis (DFA) attack on the AES-128 key with different state register placements, we show that the probability of

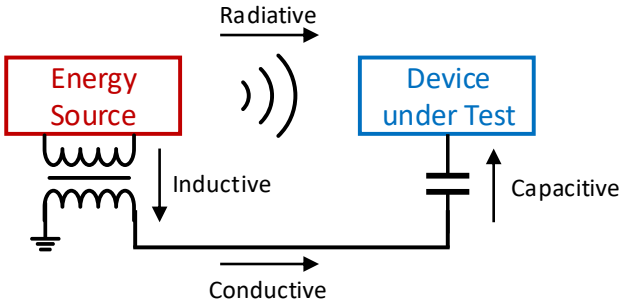


Figure 1: The four electromagnetic interference (EMI) coupling modes

key recovery strongly depends on a chip's spatial sensitivity to the frequency of injected energy. Additionally, this characterization improves the understanding of fault models and informs the placement and design of countermeasures. We further evaluate the impact of temperature and process variation by conducting experiments across multiple commercial and industrial AMD FPGAs from the same family.

2 Background

2.1 Glitching vs. Electromagnetic Interference (EMI)

Electromagnetic interference (EMI) refers to the disruption of an electronic device's normal operation due to unwanted electromagnetic disturbances from external or internal sources. Such disturbances can degrade or corrupt signals, leading to malfunctions, errors, or even complete system failure. In EMI analysis, these unwanted signals can couple into a chip's circuitry through conductive, inductive, capacitive, or radiated paths, as illustrated in Fig. 1. In this context, voltage glitching can be primarily modeled as conducted EMI, whereas electromagnetic (EM) fault injection is better represented by radiated/inductive EMI. Based on this model, voltage and EM fault attacks can be viewed as manifestations of the same underlying phenomenon (i.e., electromagnetic coupling) differing only in their coupling mechanisms. In both cases, the attacker perturbs the device by introducing glitches, spikes, or modulations into its PDN. In some attack scenarios, these two phenomena are even put hand in hand to cause a glitch in the hardware. For instance, Fokkens et al. [FXHH23] demonstrated that in an EM glitching attack (radiated EMI), the disturbance first couples into a resistive path caused by poor soldering of the chip's ground pin to the PCB, which in turn produces a voltage glitch (conducted EMI) on the die.

Such an equivalent EMI representation of voltage and EM glitching attacks helps explain the inconsistencies and reproducibility challenges associated with voltage glitching on different platforms. For example, ideally, a voltage glitch would spatially affect the chip uniformly; however, in reality, various circuits on the chip in different locations react differently to the same injected glitch [PHT19, ATG+19, PHT20]. To characterize these analog effects, the propagation path of a glitch can be modeled as a communication filter, or, equivalently, as a channel, enabling analysis of glitch behavior in the frequency domain.

2.2 PDN as a Communication Channel

The primary communication channel for the transfer of energy of a voltage glitch is the power delivery network (PDN) of the system [ZGJZ23]. The PDN is responsible for delivering a stable and low-noise voltage supply from the voltage regulator module (VRM) down to the individual transistors on the chip. It encompasses all conductive paths and

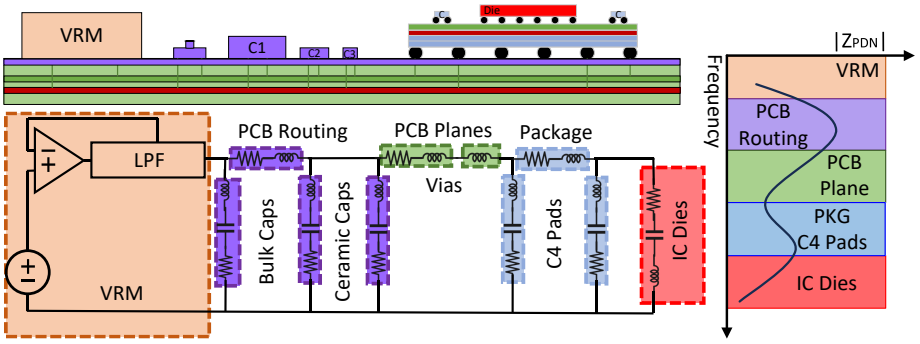


Figure 2: Equivalent RLC circuit model of the power delivery network of the PCB and chip and the contribution of different parts of the PDN to the impedance over frequency.

decoupling elements between the VRM and the active circuitry, ensuring that the required current is supplied at each operating frequency with minimal voltage fluctuation. A typical PDN includes both off-chip and on-chip components such as bulk capacitors, PCB routing, multilayer ceramic capacitors, PCB planes, vias, bonding wires, package bumps, and on-chip power grids, see Fig. 2. Each of these components contributes differently to the frequency-dependent impedance of the PDN [ZAB+18, XBZ+19].

At low frequencies, the PDN impedance is dominated by the voltage regulator and large off-chip capacitors. As frequency increases, parasitic inductance within capacitors and interconnects causes significant changes in impedance behavior [LFH10, MMST23]. Each capacitor exhibits a series resonance frequency, the point at which its parasitic inductance and capacitance form a resonant circuit. Below this frequency, the capacitor behaves as a short circuit, and above it, the inductive behavior dominates. Consequently, as frequency increases, larger capacitors (with higher inductance) cease to be effective, while smaller capacitors—due to their lower parasitic inductance continue to provide decoupling at higher frequencies. Beyond the resonance frequency of even the smallest ceramic capacitors, they effectively act as open circuits, and the interposer, package, and on-chip PDN structures dominate the impedance profile [ZAB+18], as shown in Figure 2.

By considering the PDN as a linear small-signal network, its frequency-dependent impedance provides a compact way to characterize how supply disturbances and load-current transients propagate through the power-delivery path. In a simplified single-port voltage-divider model, the delivery path is represented by a series impedance ($Z_{\text{path}}(f)$), including board traces, vias, package parasitics, and interconnects, while the die-side network is represented by an input impedance ($Z_{\text{die}}(f)$), including on-die capacitance and local decoupling. Under these assumptions, the small-signal voltage transfer from the VRM output node to the die supply node can be written as

$$H_{\text{pdn}}(f) = \frac{V_{\text{die}}(f)}{V_{\text{vrm}}(f)} = \frac{Z_{\text{die}}(f)}{Z_{\text{path}}(f) + Z_{\text{die}}(f)}. \tag{1}$$

$$V_{die}(f) = H_{pdn}(f)\, V_{\text{vrm}}(f). \tag{2}$$

The ratio between the small-signal voltage at the die, $V_{\text{die}}(f)$, and the voltage generated by the VRM, $V_{\text{vrm}}(f)$, is expressed in Equation 1. By modeling the PDN as a series combination of the path impedance $Z_{\text{path}}(f)$ (including board traces, vias, and package parasitics as depicted in Fig. 2) and the die-side input impedance $Z_{\text{die}}(f)$ (on-die capacitance and local decoupling network) [SAF+99, HL15], $H_{\text{pdn}}(f)$ physically represents the voltage-divider relationship between the delivery path and the die load. Alternatively, Equation 2 quantifies how much of the VRM voltage is transferred to the die node as a function of

frequency. Thus, the frequency dependence of $H_{\text{pdn}}(f)$ reflects the interplay between the impedance of the delivery network and the die input impedance, which together determine the power integrity and noise transmission characteristics of the PDN.

The impedance $Z_{\text{pdn}}(f)$ fully defines the PDN's frequency-dependent transfer function $H_{pdn}(f)$, and consequently governs how the input excitation $V_{\text{in}}(f)$ is filtered into the observed on-die voltage $V_{die}(f)$. The PDN therefore behaves as a frequency-selective channel whose spectral behavior is determined by its impedance profile. It determines the frequency at which energy can travel from the power supply with the least attenuation to reach the transistors on the chip. In this model, a PDN can be considered as a 3D transmission line, in which each region of the die has a slightly different frequency-dependent impedance, leading to standing waves or resonant hot spots. When a disturbance couples into the PDN, these resonances determine where voltage transients are amplified or damped. This is why a global conducted or radiated EMI can disturb certain logic regions or memory cells, and thus, have localized effects.

2.3 Characterization PDN's Frequency Response

To characterize the frequency response of the PDN across different frequency bands, scattering (S) parameters are commonly employed [Bog10, Pup20]. These parameters are widely used in RF and microwave engineering, as well as in power and signal integrity analyses, to characterize signal reflection and transmission across various frequency bands. S-parameters are complex and frequency-dependent, representing the amplitude and phase of the traveling waves in the system. The number and arrangement of these parameters depend on the number of available electrical ports in the network. In other words, for instance, by considering the reflection coefficient S_{11}, one can measure at which frequency bands the energy of the injected signal is mostly "absorbed" by the system's PDN and potentially affect the circuits on a chip. Vector network analyzers (VNAs) are typically used to obtain these parameters. They include a signal source that excites the PDN by injecting sinusoidal waves of varying frequencies and receivers that measure the reflected and transmitted signals at each frequency, revealing the system's frequency response. For a reference system with characteristic impedance Z_0 (e.g., a VNA port, $Z_0 = 50\ \Omega$), the reflection coefficient $S_{11}(f)$ is related to the PDN impedance as

$$Z_{\text{pdn}}(f) = Z_0 \frac{1 + S_{11}(f)}{1 - S_{11}(f)}. \tag{3}$$

In this paper, we use S_{11} for PDN characterizations.

2.4 Time-to-Digital Converters

While S-parameters provide a global view of the PDN's frequency response, they do not specify the sensitivity of various chip locations on smaller scales. Therefore, we require local sensors to measure fault sensitivity across various frequency bands. Time-to-Digital Converters (TDCs) can be deployed for such sensing tasks. TDCs and its variants [GOT17, SGS23, JUP24] are one of the prominent voltage drop sensors used in many other domains, such as radiation detection [DMT25], laser fault injection detection [EMV+23, HSM25], electromagnetic fault injection detection [PMBC21], laser-assisted probing detection [KMMC+24], due to their ability to measure sub-nanosecond delays. A TDC is designed to measure the time difference between two signals, typically called the start and stop signals. They can be implemented on both FPGAs and ASICs [KMSST25, ABC+07, ZSZF13, SAL06]. The input of these sensors is a pulse signal that propagates through a chain of buffers and flip-flops, see Fig. 3. The output is the state of the buffers captured by the flip-flops. The resolution of a TDC depends on the minimum delay that the circuit can detect.

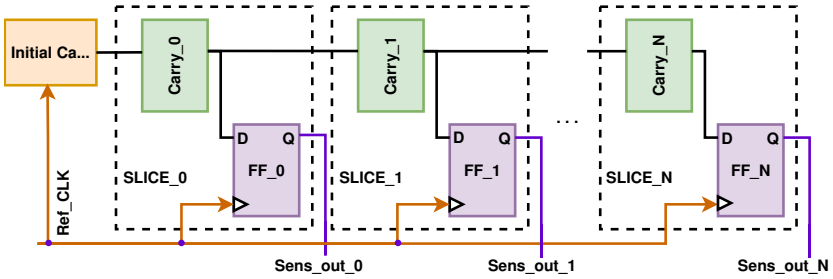


Figure 3: High-level implementation of a TDC-based fault detection sensor

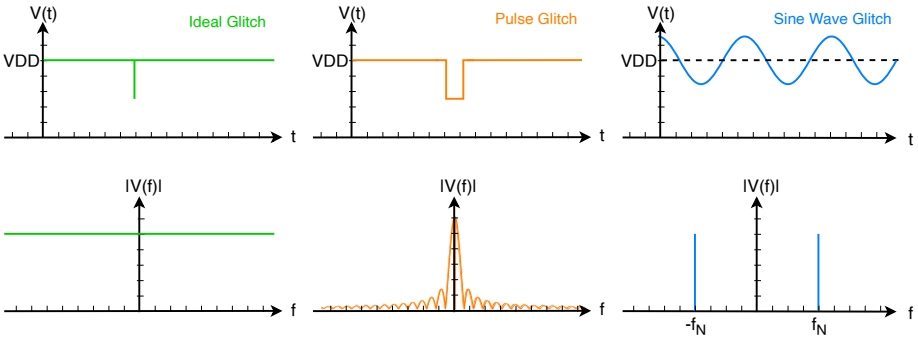


Figure 4: Glitching signals: ideal impulsed voltage glitch, realistic pulsed voltage glitch, and power rail modulation at a specific frequency (f_N) in the time and frequency domains.

2.5 Glitches in Frequency Domain

To measure the local sensitivity across various frequency bands using TDCs, we should unwrap the glitch signal in the frequency domain. An *ideal glitch* can be modeled as a negative or positive Dirac delta function in the time domain, which is a tall spike at a single point in time:

$$g_{\text{ideal}}(t) = k\,\delta(t). \tag{4}$$

Its Fourier transform is flat (constant magnitude across all frequencies):

$$G_{\text{ideal}}(f) = \mathcal{F}\{g_{\text{ideal}}(t)\}(f) = k\int_{-\infty}^{\infty}\delta(t)e^{-\mathrm{j}2\pi ft}dt = k. \tag{5}$$

Hence,

$$|G_{\text{ideal}}(f)| = k, \qquad \forall f, \tag{6}$$

which represents an *infinitely broadband* spectrum, meaning it contains equal contributions across all frequencies; see the green plots in Fig. 4.

However, a real-world and practical glitch, such as a rectangular pulse, has a finite, non-zero duration τ and a finite amplitude A in the time domain:

$$g_{\text{rect}}(t) = A \cdot \text{rect}\left(\frac{t}{\tau}\right). \tag{7}$$

The corresponding Fourier transform is the familiar rect–sinc pair as depicted in orange

plots in Fig. 4:

$$G_{\text{rect}}(f) = \mathcal{F}\{g_{\text{rect}}(t)\}(f) = A \int_{-\tau/2}^{\tau/2} e^{-\mathrm{j}2\pi f t}\, dt \tag{8}$$

$$= A\tau\, \text{sinc}\big(f\tau\big). \tag{9}$$

In any case, a glitch in time consists of several waves with different frequency components, making it challenging to analyze the impact of each frequency component separately. Therefore, one can instead modulate the supply voltage with single-frequency sine waves and sweep the frequency to observe the impact at various frequencies. Specifically, if a sinusoidal disturbance is superimposed on the main signal, the total waveform becomes

$$g_{\sin}(t) = A \sin(2\pi f_N t), \tag{10}$$

where A is the amplitude and f_N the glitch (disturbance) frequency.

In the frequency domain, using the Fourier transform identity, we have

$$\mathcal{F}\{\sin(2\pi f_N t)\}(f) = \frac{1}{2\mathrm{j}}\big[\delta(f - f_N) - \delta(f + f_N)\big]. \tag{11}$$

The spectrum consists of two impulses (delta spikes) located symmetrically at $\pm f_N$:

$$G_{\sin}(f) = \frac{A}{2\mathrm{j}}\big[\delta(f - f_N) - \delta(f + f_N)\big]. \tag{12}$$

Thus, an additive sinusoidal glitch introduces *discrete frequency components* at $\pm f_N$, unlike the broadband energy of an impulse. When superimposed on a baseband or digital signal, such sinusoidal artifacts can appear as narrowband interference (see blue plots in Fig. 4.)

3 Methodology

3.1 Approach

This section describes our methodology to systematically localize vulnerable regions on an FPGA and demonstrates targeted fault injection at those locations. To enable spatial sensing of voltage disturbances across the chip, we implement an array of TDC sensors distributed throughout the FPGA fabric. We calibrate each TDC under normal operating conditions at room temperature to create a baseline for comparison. By applying a standard voltage glitch (i.e., a negative pulse), we confirm that the TDCs register a nonuniform energy distribution on the FPGA PDN.

In **Phase 1** of profiling (see Fig. 5), we characterize the system's PDN using S-parameters (i.e., S_{11}) via a VNA. The S_{11} measurements allow us to identify frequency ranges where the chip absorbs more energy (corresponding to lower reflection) and, consequently, is most sensitive to disturbances on the voltage line. In **Phase 2**, we modulate the supply voltage of the FPGA with sinusoidal signals at several frequencies within the selected sensitive band (see Fig. 5) and record the output of every TDC. Based on the acquired results from TDCs at each frequency, we identify both sensitive and insensitive locations on the FPGA. In **Phase 3**, to determine if sensitive locations can be exploited for localized and surgical voltage fault injection attacks, we implement finite state machines (FSMs) (see Sect. 4.2.2) as attack targets in both sensitive (vulnerable) and insensitive (stable) locations, see Fig. 5. We also implement an AES core to demonstrate that the proposed framework can be used to extract the full key using DFA (see Sect. 4.2.3).

Finally, we examine how temperature and manufacturing process variations affect the fault-sensitivity pattern across various locations on the FPGA. To obtain these results, we

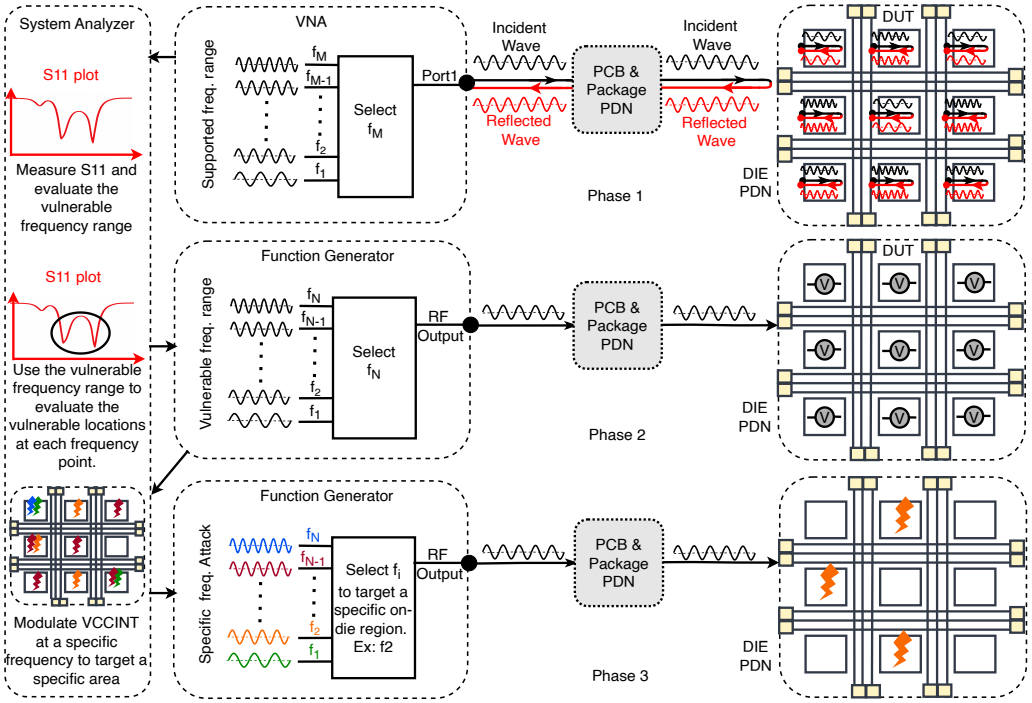


Figure 5: The concept of the GlitchSniping framework: **Phase 1:** profiling the system's PDN frequency response using the S_{11} parameter to identify the frequency band of interest. **Phase 2:** Profiling the sensitive locations by modulating the voltage and sweeping the frequency in the identified band, and reading out the TDCs outputs. **Phase 3:** Using the acquired knowledge from phases 1 and 2 to inject localized faults into an FSM or an AES.

repeat the frequency-sweep experiments (see Fig. 5) for different samples and at various temperatures. Before each experiment, we reestablish the TDC baselines for the new environment, then repeat the same power-modulation and TDC recording procedure.

3.2 Threat Model

While the primary focus of this paper is the physical characterization of voltage attacks, we can still consider a few scenarios in which both attackers and defenders can benefit from this characterization. Therefore, we consider two offensive and one defensive scenario for our threat model. The ultimate goal of the adversary in all offensive scenarios is to gain privileged access to a service, extract a cryptographic key via fault analysis, or mount a denial-of-service (DoS) attack. We also assume that the attack target is a configurable device, such as an FPGA, deployed on a PCB in a realistic system setting.

3.2.1 Scenario 1 (Offensive): Weakening Fault Resiliency of an IP Core

In this scenario, we consider a malicious system integrator capable of tampering with the placement and routing of an IP core during the design phase. The goal is to realize the IP on locations of a chip that are more sensitive to voltage variations. Hence, the attacker can make the IP core more susceptible to voltage fault attacks when the same chip is deployed in the field. In this case, we assume that in the profiling phase, the adversary has physical access to the system (i.e., a soldered FPGA on a PCB), allowing them to profile the device and measure its S-parameters to identify vulnerable frequency ranges. Moreover, they

can profile the fault-sensitive locations on the device by running their own sensors and modulating the core voltage with various frequencies.

3.2.2 Scenario 2 (Defensive): Strengthening Fault Resiliency of an IP Core

In this scenario, we consider a designer trying to make a design more resilient to malicious or natural faults (e.g., electromagnetic pulses from solar flares in space). In this case, the designer can profile the final product after packaging and soldering at various frequencies using S-parameters and TDCs. By identifying regions that consistently exhibit lower sensitivity to voltage perturbations, the designer can prevent critical logic from being placed in vulnerable locations, thereby reducing susceptibility to voltage-induced faults even in the presence of environmental, process, and board-level variations.

3.2.3 Scenario 3 (Offensive): Differential Fault Analysis Key Extraction

In the third scenario, we consider a classical fault adversary attempting to extract secret keys from a cryptographic IP core (e.g., AES) using differential fault analysis. The adversary is assumed to have physical access to the target device. This attacker model is consistent with widely accepted voltage-glitching threat models in the literature, in which the adversary can repeatedly reset and query the device but cannot invasively probe the silicon or modify the internal placement and routing of the victim logic. Unlike conventional voltage glitching attacks, which rely primarily on time-domain voltage pulses, the adversary can inject a signal at a specific frequency to affect a specific region of the die. We also assume that the attacker can profile the target using on-chip sensors, such as TDCs, and then use that information to carry out the attack. This is especially useful for the FPGA target, where the attacker can collect profiling information from a test bitstream and then perform the attack on the target design, aligning perfectly with template attack threat models.

4 Experimental Setup

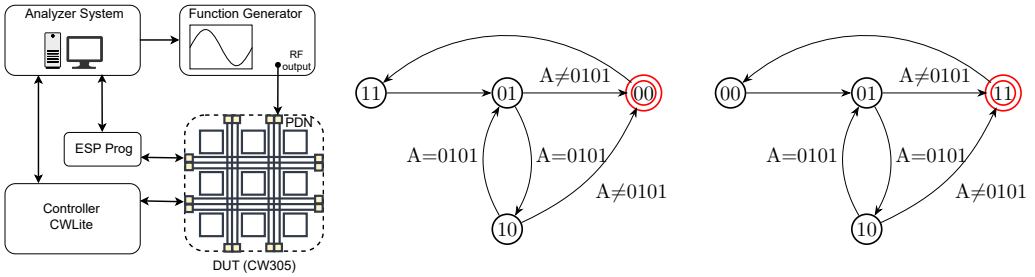
4.1 Measurement Setup

4.1.1 Devices Under Test

We deployed NewAE CW305 boards (NAE-CW305) [New25a] in two configurations for our experiments. The first configuration is equipped with a BGA socket option, which can accommodate any AMD/Xilinx Artix-7 device in the FT256/FTG256 package [Inc18]. This configuration was used to study the effects of temperature and manufacturing process variation on localized faults across various samples. In this case, we used industrial-grade and commercial-grade AMD/Xilinx Artix-7 FPGAs (XC7A35T-FTG256) fabricated with the 28 nm technology. The second configuration is equipped with a soldered AMD/Xilinx Artix-7 FPGA (XC7A100T-FTG256). This configuration was used for attacking an AES implementation. A key feature of the CW305 boards is their direct access to the FPGA's PDN. Our study specifically targeted the V_{CCINT} voltage rail, which supplies the FPGA's core logic and registers.

4.1.2 Frequency-based Voltage Modulation Setup

We used the Keysight N9310A RF Signal Generator [Key25], which operates over a frequency range of 9 kHz to 3.0 GHz and provides a configurable output amplitude from –127 dBm to +20 dBm. To ensure stable and reliable signal transmission, we employed shielded characterization cables (Mini-Circuits CBL-2FT-SMNM+) [MC25]. To control the target FPGA, we employed a NewAE CW-Lite board [New25b]. The CW-Lite establishes



(a) (b) (c)

Figure 6: Experimental setup for **Phase 3**: (a) schematic representation and (b), (c) corresponding two finite state machines (FSMs) used for fault injection experiments. Each FSM transitions normally between states 01 and 10; any bit flip forces a transition into the fault state. In (b), the fault state is 00, and in (c), the fault state is 11, both shown in red.

serial communication with the DUT, acting as a controller to send commands and receive responses from the target FPGA. A computer served as the analysis system, interfacing with the controller, DUT, and function generator. Communication between the analysis system and the DUT was established via UART serial communication using the ESP-Prog board [Sys25]. The overall experimental setup is shown in Fig. 6a. For data exchange with the FPGA, controlling, and measurement analysis, we used Python 3.12. Instrument communication was handled through the PyVISA library. The FPGA hardware designs were written in Verilog and synthesized using Vivado Design Suite.

4.1.3 *S*-parameter Characterization

To characterize the DUT in the frequency domain, we used a Mini-Circuits eVNA-63+ vector network analyzer (VNA) [Min22], which is a two-port, PC-controlled instrument covering 300 kHz to 6 GHz with output power levels from –50 dBm to +10 dBm. The DUT was connected to the VNA through coaxial cables, and calibration was performed using standard SOL (Short, Open, Load) routines. Measurement control and automation were carried out via the analyzer system using the SCPI-based API in Python.

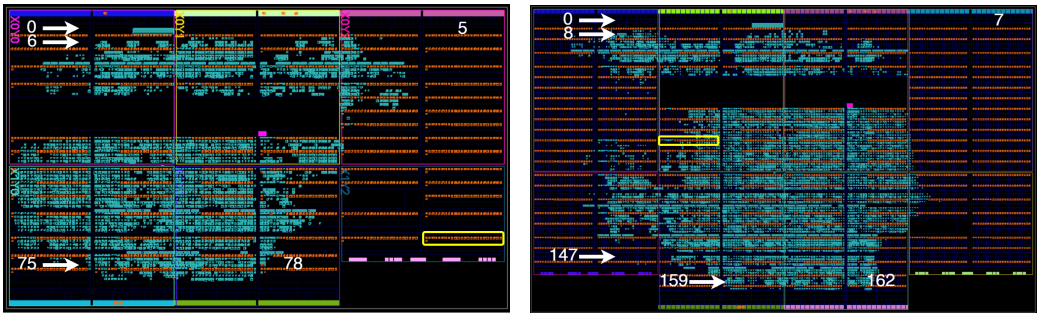
4.1.4 Thermal Chamber

For conducting temperature-controlled experiments, we used a TestEquity Model 107 temperature chamber [LLC09]. The TE-107 is a benchtop environmental chamber capable of providing precise thermal conditions over a wide range (–40°C to +130°C). It is equipped with digital controls for accurate programming and monitoring, allowing stable and repeatable operation during long experimental runs.

4.2 Hardware Implementation

4.2.1 TDC Implementations

We implemented 79 64-bit time-to-digital converter (TDC) sensors, across the fabric of the XC7A35T FPGA (see Fig. 7a) and 163 TDC sensors across the fabric of the XC7A100T FPGA (see Fig. 7b). As shown in Fig. 7, the TDCs are highlighted in orange. Some regions of the FPGA lack the necessary components to implement a TDC and, therefore, cannot be directly monitored. The sensors were arranged in a 6-row by 16-column grid on the XC7A35T and an 8-row by 23-column grid on the XC7A100T, covering a significant portion of the device. A 250 MHz clock drove the TDCs, while the main system clock of



(a) (b)

Figure 7: Physical layout of the chip, highlighting the positions of the TDCs in orange: (a) XC7A35T and (b) XC7A100T. The yellow box in both (a) and (b) indicates the region corresponding to TDC_{70}.

the chip operated at 100 MHz. The TDC results were read through UART communication provided by the ESP-Prog board [Sys25]. We calibrated it so that about half the bits are ones and half are zeros.

4.2.2 Fault Evaluation FSMs

To evaluate the fault, we designed two finite state machines (FSMs), illustrated in Fig. 6b and Fig. 6c. Each FSM includes a final (fault) state, which cannot be reached under normal operating conditions. However, when a fault occurs, such as a bit flip in either the state register or a transition signal, the FSM enters the fault state. Once in the fault state, a counter is triggered to record the event by incrementing the fault count. After logging the fault, the FSM returns to the initial state to repeat the process. During normal operation, the FSM continuously alternates between states 01 and 10. The FSMs are operated with a 400 MHz clock.

4.2.3 AES Implementation

For differential fault analysis (DFA) experiments, we implemented a basic 1-round/cycle 128-bit AES core [New26], consisting of 10 rounds, on the ChipWhisperer CW305 (XC7A100T) board. The design operates at a 100 MHz clock.

5 Results

This section presents the results of our approach, as detailed in Sect. 3. We begin by performing voltage glitch fault injections and verifying that the TDCs can detect the nonuniform transfer of energy on the chip. Afterward, we focus on frequency-dependent effects. We characterize the PDN of the FPGA board using S_{11} parameter. Moreover, we modulate the FPGA's core voltage over a wide range of frequencies while we record the TDC outputs. Finally, we evaluated how temperature and manufacturing process variations affect the spatial vulnerability patterns.

5.1 Pulsed Voltage Glitching

We performed voltage glitching on the FPGA's V_{CCINT} voltage rail and monitored the outputs of the TDC sensors over time. We connected the CWLite's GLITCH SMA output to the CW305's X3 SMA connector. Using a Python script, we controlled the CWLite

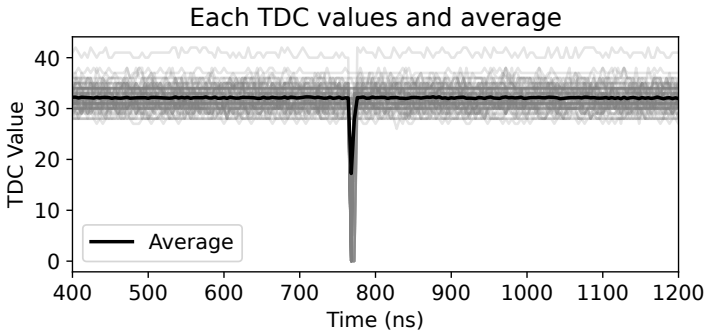
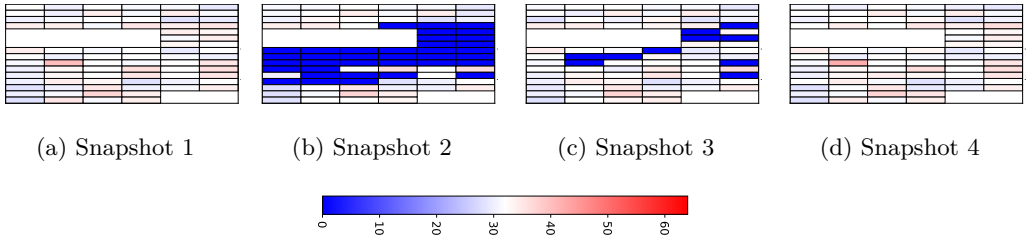


Figure 8: Impact of voltage glitching on TDCs. Each TDC line is in gray, and the average line is in black.



(a) Snapshot 1 (b) Snapshot 2 (c) Snapshot 3 (d) Snapshot 4

Figure 9: TDC heatmaps captured during voltage glitch fault injection, shown across four snapshots. Each snapshot is taken one TDC clock cycle apart (4ns).

to generate and inject voltage glitches into the CW305 board. Our automated script (1) programs the FPGA with the target bitstream and calibrates the TDCs, (2) arms the glitch hardware at X3, (3) triggers a controlled glitch pulse, (4) reads back on-chip TDC sensor data to identify voltage variations, and (5) logs and saves the TDC outputs for analysis. In this voltage glitch setup, the system was configured to inject a short, precisely timed disturbance on the target's power rail. The glitch was applied only once for each trigger event. The glitch was configured to repeat for a single clock cycle.

After measuring all TDCs over time and calculating their average, we observed that the average TDC value reveals the voltage glitch occurring inside the chip. Fig. 8 shows that the average of all TDCs can reveal the glitch event inside the chip. We recorded each TDC's output for 10,000 TDC clock cycles (250 MHz). The chip's clock was set to 100 MHz. For clarity, the figure shows only the subset of cycles from 100 to 300. By examining each TDC individually, we observed that the response varied for each sensor. Some TDCs showed clear changes during the fault injection, while other TDCs were not affected. We performed this experiment on one of our industrial-grade FPGA samples. Note that the arrival time of glitch to various TDCs differ depending on the geometry and impedance of the die's PDN; however, we are neglecting these small time differences in this work by aligning the traces based on the maximum TDC value drops in Fig. 8.

Fig. 9 shows a heatmap of all TDCs across four snapshots in time. Snapshots 2 and 3 match the time of the glitch injection. These two snapshots show that some TDCs are more sensitive and vulnerable than others. Snapshots 1 and 4 represent the states of the chip before and after the voltage glitch, respectively, demonstrating that all TDC sensors were in calibration mode. This is already an indication that the impact of voltage glitching is nonuniform across the die. To understand the relationship between the sensitive locations and energy of the glitch in different frequencies, we should consider every frequency separately, as described in Sect. 2.5. But first, we should reduce our search space to specific

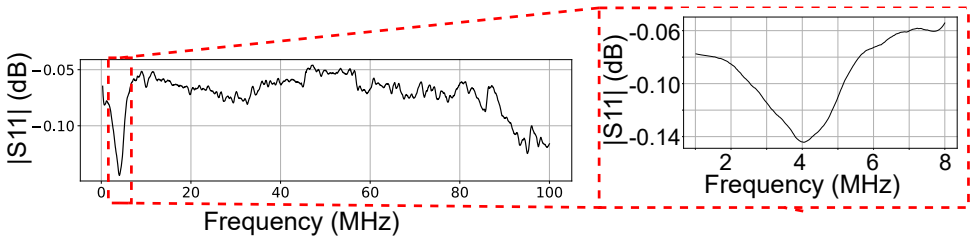


Figure 10: The amplitude of S11 parameter in the range of 3 kHz to 100 MHz. A resonance frequency at can be observed between 1 MHz to 8 MHz centered at 4 MHz.

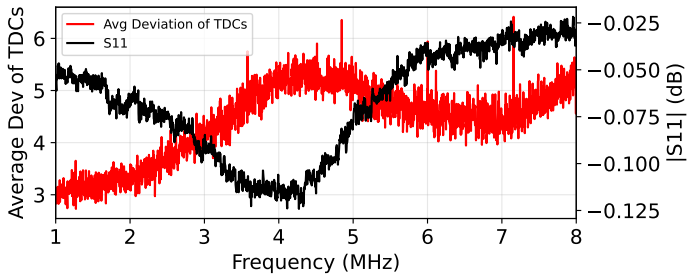


Figure 11: The correlation between the S_{11} parameter and the average deviation of TDC measurements relative to their calibration point.

frequency bands, which are more sensitive to voltage variations.

5.2 Finding Vulnerable Frequency Band using S-parameters

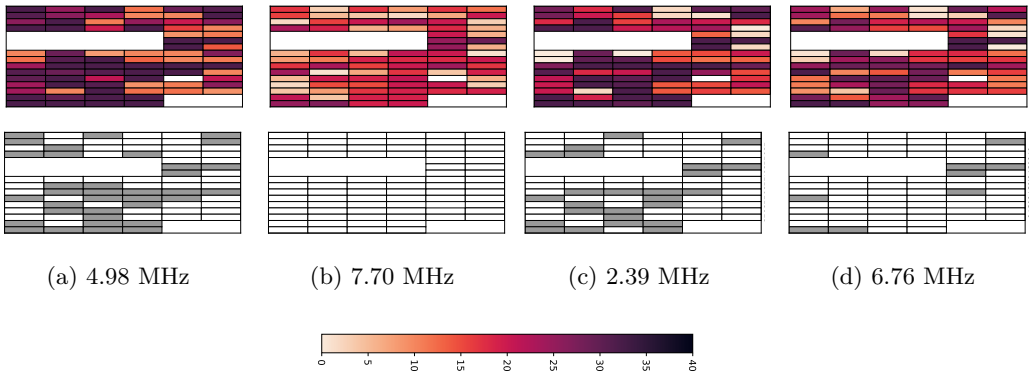
To inject faults by modulating the V_{CCINT}, we first needed to identify the frequency range that makes the device most vulnerable. For this purpose, we measured the S_{11} parameter from 3 kHz to 100 MHz using 5000 frequency points at a power level of 0 dBm. To measure S_{11}, we calibrated the vector network analyzer, connected the board's SMA connector X3, and recorded the S-parameter response. Measurements were taken with the chip powered without any bitstream loaded. As shown in Fig. 10, there is a resonance frequency between 1 MHz and 8 MHz, where the system's PDN absorbs more energy, indicating higher sensitivity and vulnerability in this frequency region.

5.3 Voltage Modulation Fault Injection

After measuring the S_{11} parameter and identifying the vulnerable frequency range, we modulated the V_{CCINT} within that range. We configured the FPGA with the bitstream containing the TDCs. After TDCs' calibration at room temperature, we connected the output of the RF function generator to the CW305's X3 SMA port. The RF function generator applies controlled voltage modulation to the FPGA's power rail across the desired frequency band. We set the amplitude to +20 dBm. We performed voltage modulation on the FPGA's V_{CCINT} in the frequency range from 1 MHz to 8 MHz with 2200 points. By observing the average deviation of TDC values from their calibration point, it is evident that there is a correlation between the TDCs' highest deviation and the resonance frequency band obtained by the S_{11} parameter, see Fig. 11. This confirms that in the resonance frequency range, the system absorbs more energy compared to other frequency bands, and thus, the TDCs are also more affected.

Table 1: TDCs with maximum and minimum deviation across the frequency band 1-8 MHz, showing that TDC_{46} dominates in most frequency ranges, and TDC_{39} is the most stable location.

Max Deviation		**Min Deviation**	
TDC ID	**Count**	**TDC ID**	**Count**
TDC_{46}	1891	TDC_{39}	1514
TDC_{75}	1697	TDC_{29}	1355
TDC_{28}	1691	TDC_{66}	1332



(a) 4.98 MHz (b) 7.70 MHz (c) 2.39 MHz (d) 6.76 MHz

Figure 12: Behavior of the TDCs at four different frequency points: (a) a frequency with many active TDC deviations, (b) a frequency with fewer deviations, and (c–d) two additional representative frequency points. The top row displays heatmaps of the deviation of TDC values from their calibration point, while the bottom row highlights TDCs with deviations exceeding 30 (shown in gray), highlighting the most fault-sensitive locations. While there might be some overlaps in some locations, there are also sensitive locations that are unique at different frequencies.

While the average deviation of TDCs shows a strong correlation with the S_{11} magnitude, individual TDCs at different locations display varying sensitivities and behaviors across different frequency points. Fig. 12 shows how the sensitivity map of TDCs varies at different frequency points. It can be observed that every frequency causes a unique sensitivity pattern on the FPGA die. We also calculated the maximum and minimum deviations of each TDC relative to its calibrated values under normal conditions, across frequencies. We then identify the TDCs that deviate the most over the frequency band. Table 1 shows that in most frequencies, TDC_{46} had the maximum deviation, and TDC_{39} had the minimum deviation. This result indicates that the location of TDC_{46} is more vulnerable, and the location of TDC_{39} is less vulnerable across most frequencies.

To validate our hypothesis, we placed the FSMs described in Sect. 4.2.2 at the locations of TDC_{46} and at the location of TDC_{39}. At each location, we placed two FSM variants to check both types of bit flip directions, from 0 to 1 and from 1 to 0, for the final state. For this reason, in one case the final state was set to "00" and in the other case to "11". Fig. 13 shows the results of our fault injection attempts by modulating the voltage between 1 and 8 MHz. It can be observed that while the FSMs placed at the location of TDC_{46} experience faults at some frequency points, the FSMs at the location of TDC_{39} show no faults at all. The number of faults for FSM at the location of TDC_{46} transitioning from zero to one is smaller than those from one to zero. These results confirm that if the frequency profile of a device is known, it is possible to glitch specific locations on the chip by modulating the voltage at the right frequency.

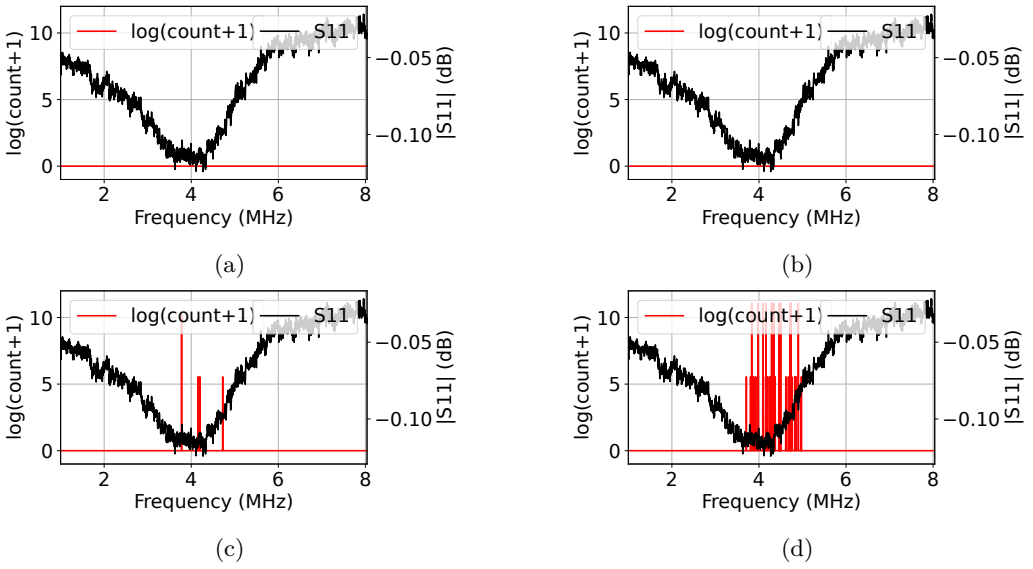


Figure 13: Results of Fault Injection on the State Machine at (a) TDC_{39} – Final State 00 (b) TDC_{39} – Final State 11 (c) TDC_{46} – Final State 00 (d) TDC_{46} – Final State 11. We used the logarithm of the fault counts as the number of faults at some frequency points are very high while at others it was very low.

Table 2: Max and Min Deviation Counts and Percentages for Different Temperatures.

	Max Deviation		Min Deviation	
Temperature	**TDC ID**	**Count**	**TDC ID**	**Count**
0°C	TDC_5	1428	TDC_2	778
Room	TDC_{10}	2164	TDC_0	993
85°C	TDC_{72}	1361	TDC_{67}	1447

5.4 Localized Voltage Glitching on AES Subkeys

Since the XC7A100T FPGA (used for the AES implementation) has more resources than the XC7A35T (used in the previous experiments), we instantiated a larger number of sensors (163 TDCs) to extend the spatial coverage of sensing across the device. We then measured the TDCs to identify the TDC instances with the largest deviations, representing the most sensitive locations. From this analysis, the six highest-deviation TDCs were TDC_{60}, TDC_{109}, TDC_{52}, TDC_{96}, TDC_{74}, and TDC_{106}. We used this information to constrain four target AES state bytes to be placed in these vulnerable regions.

We followed the well-known DFA procedure, targeting the timing window between eighth and ninth MixColumns, primarily with single fault occurrence [DLV03]. In our setup, faults were injected by modulating the FPGA core supply voltage (V_{CCINT}) using an external function generator. This process produced faulty ciphertexts, which were then used in our DFA to recover the AES key [Aru25]. A single byte fault in this case propagates through the last AES MixColumns in the 9th round and produces faulty ciphertexts in 4 bytes.

We mapped the first four AES state bytes (each corresponding to the recovery of four subkeys denoted by $Subkey_1$ to $Subkey_4$) onto selected high-deviation locations and evaluated the faulty behavior of the AES core. From the faulty ciphertexts, we can consequently recover groups of 4-byte keys. The recovered AES subkeys correspond to the following byte index groups in our DFA experiments:

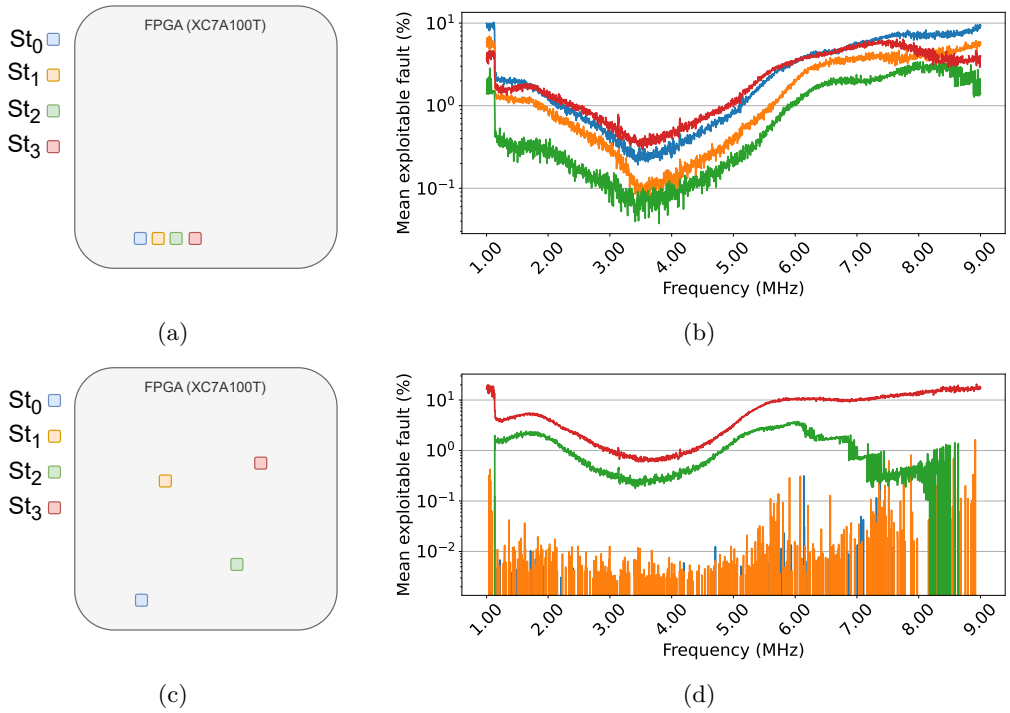


Figure 14: Comparison of AES fault behavior under one-location (top) and four-location (bottom) state-byte placements. (a) Scenario 1: the four target state bytes are colocated at the TDC_{60} region. (b) Scenario 1 results: the subkey-recovery fault probability varies with frequency and differs across the four bytes. (c) Scenario 2: the four target state bytes are distributed across four different TDC regions (TDC_{106}, TDC_{74}, TDC_{96}, and TDC_{52}). (d) Scenario 2 results: the byte placed at the highest-deviation location exhibits the highest fault probability, followed by the remaining bytes according to the relative deviation of their corresponding locations.

1) $Subkey_1$: key bytes $\{0, 7, 10, 13\}$ 2) $Subkey_2$: key bytes $\{1, 4, 11, 14\}$ 3) $Subkey_3$: key bytes $\{2, 5, 8, 15\}$ 4) $Subkey_4$: key bytes $\{3, 6, 9, 12\}$

As described earlier, our DFA targets the last occurrence of MixColumns [DLV03] with the ideal goal of injecting a fault that results in a single-byte difference in the target window [TMA11]. We provide two distinct scenarios in our experiments:

Scenario 1: The first four state bytes (St_0 to St_3) located at the single most vulnerable location. Hence, we placed the St_0 to St_3 state bytes in the region corresponding to TDC_{60} (the location of maximum deviation with regard to TDCs), arranging them in four adjacent columns within the same row.

Scenario 2: The first four state bytes are distributed across multiple high-deviation locations. In the second scenario, we distributed the four AES state bytes across four high-deviation locations: TDC_{52}, TDC_{96}, TDC_{74}, and TDC_{106}. Note that we did not use the two most extreme sensitive locations (TDC_{60} and TDC_{109}) since their deviation was so large that faults occurred almost exclusively in these two state bytes, making it impractical to induce isolated single-byte faults for the remaining two bytes across the frequency band.

We placed the first four state bytes (St_0 to St_3) as follows: St_0: TDC_{106}, St_1: TDC_{74}, St_2: TDC_{96}, and St_3: TDC_{52}. The attack procedure begins by configuring and activating the RF fault-injection source over a predefined frequency range (1–9 MHz). For each

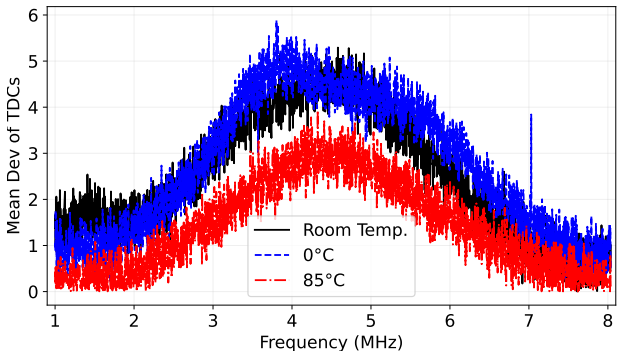
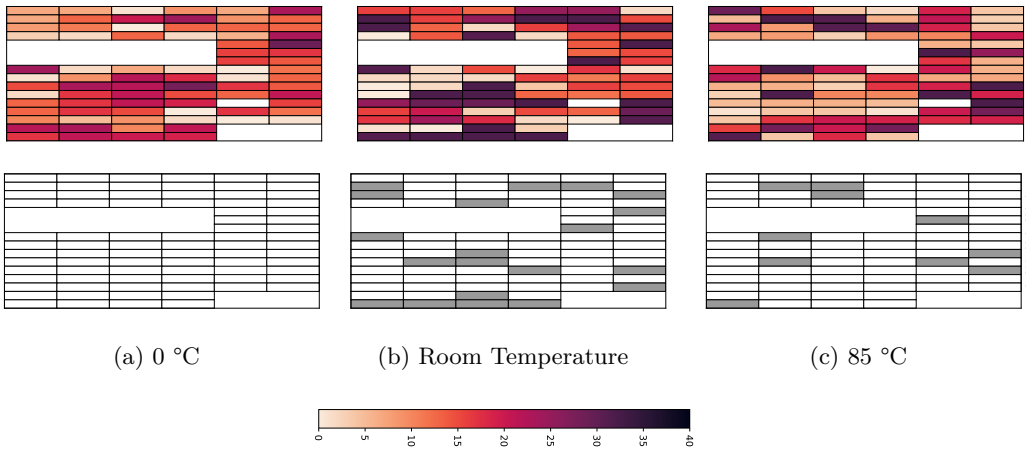


Figure 15: The average deviation of TDCs under V_{CCINT} modulation from 1 MHz to 8 MHz at three different temperatures (0°C, room temperature, and 85°C)



(a) 0 °C (b) Room Temperature (c) 85 °C

Figure 16: Behavior of the TDCs at 6.76 MHz under three different temperature conditions: (a) 0 °C, (b) room temperature, and (c) 85 °C. The top row displays heatmaps of absolute deviation values, while the bottom row highlights TDCs with deviations exceeding 30 (shown in gray), highlighting the most fault-sensitive locations. It can be observed that the die becomes less sensitive at lower 0°C.

frequency point, the target device is repeatedly triggered to execute AES under a known-plaintext setting. Since precise timing control over the injected faults is not available, we compensate by performing a large number of repeated encryptions. This is realistic as AES is often used for bulk encryption. Specifically, we use three nested loops: 400 plaintexts and 200 repeated traces per plaintext over 2000 frequency points. This results in a total of 160 million traces per scenario, with a full acquisition time of approximately 16.5 hours.

Because the fault injection timing is random, the injections statistically overlap with the critical window between the final two MixColumns operations. For single-fault events, roughly one in ten injections is expected to fall within this window and produce exploitable faulty ciphertexts. During the experiment, we record plaintexts, faulty ciphertexts, and corresponding injection frequencies, enabling both DFA-based key recovery and frequency-dependent analysis.

Following the DFA post-processing procedure in [TMA11], faulty ciphertexts are filtered based on the number of affected bytes. Due to the lack of precise timing control in our fault injection setup, the majority of faulty ciphertexts (approximately 90%) do not satisfy the required fault model, i.e., faults occurring between the final two MixColumns operations.

In particular, only ciphertexts exhibiting exactly four faulty bytes are considered valid. This criterion stems from the diffusion property of the MixColumns transformation, whereby a single-byte fault propagates to exactly four bytes. Consequently, ciphertexts with more or fewer than four faulty bytes are discarded, as they do not conform to the assumed DFA model [TMA11].

In the first scenario, 47.2 million out of 160 million traces were faulty, among which 2.3 million were exploitable. In the second scenario, where the four AES state bytes were spatially separated, 35.7 million traces were faulty, of which 1.5 million were exploitable. From a practical attack perspective, the first scenario required only 6 million traces (approximately the first 34 minutes for measurement) to obtain ciphertexts needed for DFA (for extracting 128-bit full key), whereas the second scenario required 17.2 million traces (approximately 1.5 hours) to reach the same condition.

For the first scenario, where targeted state registers are placed in close proximity to each other, the results show that the successful fault probabilities vary across the four subkeys over the frequency band, even though the bytes are placed in nearly the same vulnerable region. This means their behavior differs across frequency points and locations. Fig. 14b illustrates the rate of exploitable single-byte faults on the target window (among all faulty ciphertexts) that are observed for each subkey:

$$R_{\text{Subkey}}(f) = \frac{N_{\text{Subkey}}(f)}{N_{\text{all}}(f)} \tag{13}$$

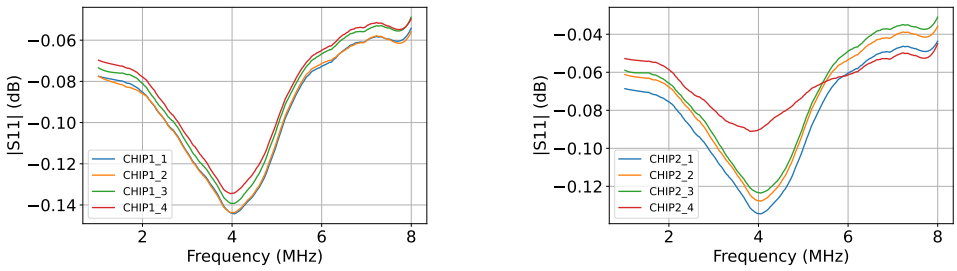
$N_{\text{Subkey}}(f)$ denotes the number of exploitable single-byte faulty ciphertexts corresponding to a given subkey at frequency f, and $N_{\text{all}}(f)$ represents the total number of faulty ciphertexts observed at the same frequency. Hence, $R_{\text{Subkey}}(f)$ captures the fraction of faults that are exploitable for that subkey at frequency f.

For the second scenario, where state registers are placed in distant regions, Fig. 14d shows the corresponding results. In this scenario, the observed fault rate shows an overall trend of $R_{subkey_4} > R_{subkey_3} > R_{subkey_2} > R_{subkey_1}$ in the selected frequency band. This trend is consistent with the fact that each byte experiences a different fault exposure level due to its placement, and the locations with higher TDC deviation tend to induce faults more frequently across the swept frequency range.

Overall, our results demonstrate successful full key recovery by leveraging spatial-frequency information inferred from our TDC-enabled analysis (described in Section 5.3) for FPGA-based AES implementations via DFA. Comparison of the two scenarios (shown in Fig. 14) suggests that by guiding the placement of target state bytes into high-sensitivity regions, the probability of inducing exploitable faults increases, thereby reducing the number of required traces and the overall attack time. Note that when all states are placed in nearly the same location, their fault probabilities follow a similar trend and remain close to one another, which agrees with our previous results on FSMs. In contrast, when the state bytes are placed in four distinct locations, the fault probabilities exhibit different trends.

5.5 Effect of Temperature

To examine the effect of temperature on the chip's behavior, we placed the CW305 board inside the thermal chamber, see Sect. 4.1.4. All temperature experiments were conducted on a commercial-grade FPGA. The experiments were performed at three different temperatures: 0°C (minimum recommended operating condition), room temperature, and 85°C (maximum recommended operating condition). At each temperature, the TDCs were first recalibrated and then used for measurements. The results are shown in Fig. 15. The results indicate that at 85°C, all TDCs respond more slowly than at room temperature, leading to lower average measurements. Conversely, the TDCs operate faster at 0°C relative to room temperature. The smaller variation between 0°C and room temperature,



(a) (b)

Figure 17: S_{11} measurements of all eight FPGAs. Each FPGA shows slightly different S_{11} values across frequencies, and clear differences can be seen between the (a) industrial-grade and (b) commercial-grade ICs. The S_{11} parameter of sample 4 (red) in commercial-grade FPGAs shows a larger deviation because it had been aged during our temperature experiments.

compared to that between room temperature and 85°C, is due to the smaller temperature difference between the two lower conditions.

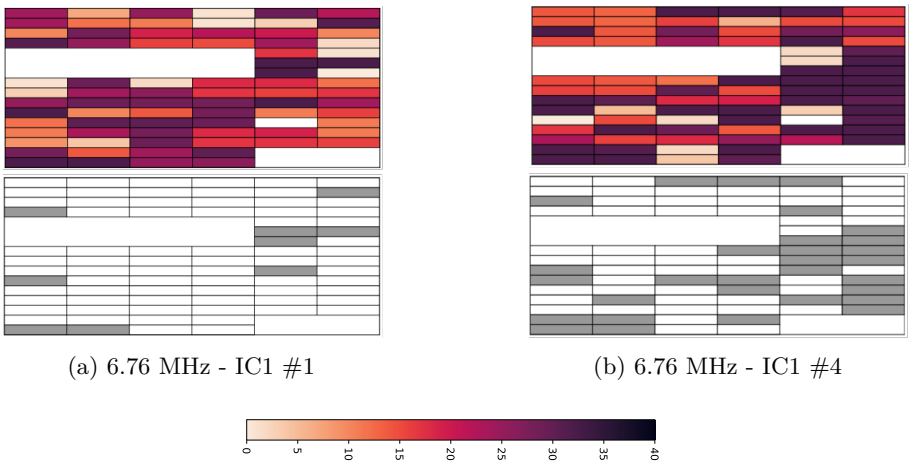
Fig. 16 shows how the sensitivity map of TDCs varies at different temperatures. It can be observed that temperature variations affect the fault sensitivity pattern on the FPGA die. Table 2 demonstrates that temperature variations can change the sensitive locations across the chip. As shown, for the commercial-grade FPGA, TDC_{10} exhibited the largest deviation at room temperature, accounting for about 98% of cases. However, at 0°C and 85°C, the maximum deviation occurred in TDC_5 (65%) and TDC_{72} (62%), respectively. Similarly, the TDCs with the minimum deviations also changed with temperature. At room temperature, TDC_0 showed the lowest deviation, while at 0°C and 85°C, TDC_2 and TDC_{67} exhibited the smallest deviations. Based on the temperature results, spatial profiles are device-specific but also depend on operating conditions. Even on the same chip, changes in temperature can shift the sensitive frequency bands and affected regions. Therefore, profiling should ideally be performed under conditions that closely reflect the actual attack/deployment setting.

5.6 Effect of Manufacturing Process Variation

Finally, we studied the effect of process variation across different FPGA samples. As mentioned earlier, the existing socket on the CW305 board enabled us to experiment with multiple FPGAs without modifying the PCB. This setup isolates the impact of manufacturing process variation on the FPGA samples only. We tested eight different FPGAs, four commercial-grade and four industrial-grade. In the first step, we measured the S_{11} parameter for all eight FPGAs. As shown in Fig. 17, each FPGA reveals different S_{11} signatures between 1 MHz and 8 MHz, and noticeable differences can be observed between the commercial-grade and industrial-grade chips.

We examined the behavior of the TDCs across all eight chips by modulating V_{CCINT} over 2200 frequency points from 1 MHz to 8 MHz. For each FPGA, the TDCs were recalibrated at room temperature before modulating the V_{CCINT}.

Fig. 17 illustrates the variations in the S_{11} parameters of identical FPGAs from the same family, indicating that manufacturing process variations directly affect the impedance characteristics of otherwise identical samples, and consequently, their fault sensitivity. Similarly, Fig. 18 further shows how the TDC sensitivity maps differ between two samples of the same FPGA family. These observations confirm that manufacturing process variations influence the spatial pattern of fault sensitivity across FPGA dies.



(a) 6.76 MHz - IC1 #1 (b) 6.76 MHz - IC1 #4

Figure 18: Behavior of the TDCs at 6.76 MHz for two different industrial-grade ICs. The top row displays heatmaps of absolute deviation values, while the bottom row highlights TDCs with deviations exceeding 30 (shown in gray), highlighting the most fault-sensitive locations.

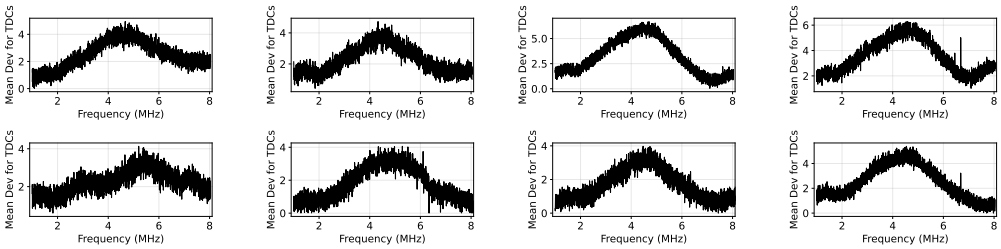


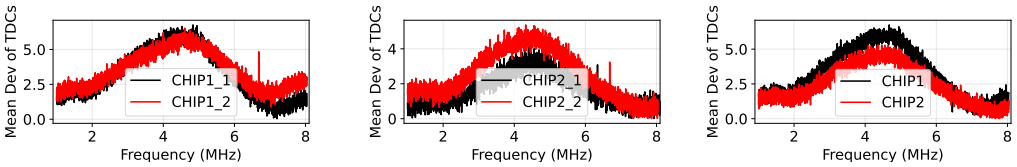
Figure 19: Mean of the deviation of TDC values from their calibration point under V_{CCINT} modulation (1–8 MHz) for eight FPGAs: Four commercial-grade (top four graphs) and four industrial-grade (bottom four graphs).

The results in Fig. 19 also show the mean of the TDCs across frequency, demonstrating that process variation can significantly influence the TDC mean at different frequencies. Table 3 shows that the fault locations vary across different FPGAs. As seen in the table, the industrial-grade FPGAs (IC1) and the commercial-grade FPGAs (IC2) exhibit different TDCs with maximum and minimum deviations. This observation suggests that the vulnerable and resilient regions on the chip are also device-dependent and influenced by process variations. The comparison of the TDC mean values for two industrial-grade ICs, two commercial-grade ICs, and one industrial-grade and commercial-grade IC is shown in Fig. 20a, Fig. 20b, and Fig. 20c, respectively. The impact of process variation was relatively small for industrial-grade ICs, whereas it was more noticeable for commercial-grade ICs.

6 Discussion

6.1 Comparison with Conventional Timed Voltage-Glitch Attacks

The proposed sinusoidal-modulation approach differs from conventional precisely timed voltage-glitch attacks in that it injects sine waves over several clock cycles at specific frequencies rather than a short pulse at a specific clock cycle. In our current setup, the modulation is applied while the target repeatedly executes the operation, so faults occur



(a) (b) (c)

Figure 20: Comparison of the TDC mean deviation values for (a) two industrial-grade ICs, (b) two commercial-grade ICs, and (c) one industrial-grade IC and one commercial-grade IC under V_{CCINT} modulation (1–8 MHz).

Table 3: Max and Min Deviation for Each IC

IC1					IC2				
	Max Deviation		Min Deviation			Max Deviation		Min Deviation	
#	TDC ID	Count	TDC ID	Count	#	TDC ID	Count	TDC ID	Count
1	TDC_{46}	1891	TDC_{39}	1514	1	TDC_{76}	1889	TDC_{21}	1751
2	TDC_{51}	1970	TDC_{48}	1493	2	TDC_{68}	1742	TDC_{16}	1652
3	TDC_{29}	2040	TDC_{55}	1507	3	TDC_{59}	1948	TDC_{22}	1775
4	TDC_{27}	2082	TDC_{68}	1573	4	TDC_{10}	2164	TDC_{0}	993

when the sinusoidal disturbance overlaps with a sensitive computation window. However, the approach can be made time-controlled by gating or enabling the RF signal only during a selected trigger window, or by using a burst-mode RF source synchronized with the target clock or trigger signal. In this case, the attacker could combine spatial selectivity from the modulation frequency with temporal selectivity from the trigger timing, similar to conventional voltage-glitch attacks.

6.2 Applicability Beyond FPGA Targets

Our profiling method is most directly applicable to FPGAs, as distributed TDC sensors can be implemented on the reconfigurable fabric. For non-FPGA targets, such as microcontrollers or ASICs, the attacker can still perform the first profiling phase by measuring the S11 response of the chip PDN and identifying vulnerable frequency bands. However, without distributed on-chip sensors, the second phase must rely on indirect observations, such as fault rates, faulty output patterns, or side-channel leakage, while sweeping the modulation frequency. In this case, the attacker may not obtain a detailed TDC-based spatial map, but can still learn which frequencies are more likely to affect specific computations or registers. Therefore, the general approach extends beyond FPGAs, although the quality of spatial localization depends on the availability of on-chip sensors or other observable feedback.

6.3 Effect of Spatial Sensitivity on Fault-Based Key Recovery

Spatial profiling remains practically useful for DFA even when the attacker does not directly control the AES placement or know the exact locations of the AES state registers. In this case, the profile does not necessarily provide a deterministic way to target a specific register, but it can still guide the attacker toward frequency ranges more likely to produce localized exploitable faults. This can provide indirect information about where sensitive AES state registers are likely located and which frequencies are more effective for disturbing them. Therefore, even without placement control, spatial profiling can reduce the search space, explain why certain frequencies or placements yield more useful DFA faults than others, and help predict the attack's relative difficulty.

6.4 Potential Countermeasures

Several countermeasures can be considered based on the observed spatial and frequency-dependent fault sensitivity. Critical logic should preferably be placed in regions with lower sensitivity during profiling, rather than in regions strongly affected by specific modulation frequencies. In addition, improving PDN decoupling can reduce resonance effects and attenuate the injected energy before it reaches the die.

6.5 Relationship with Impedance SCA

It can be observed that there is a duality between the proposed frequency-based voltage fault attacks and impedance side-channels, in that both excite the PDN with sinusoidal waves. The frequency points identified in this work for fault injection differ from the frequency bands typically reported for the leakage through static impedance side-channel analysis [MMST23, MMT23, KMFT24]. This difference stems from the distinct physical goals of the two classes of experiments. Fault injection aims to deliver RF energy into the device PDN so that the energy is *absorbed* by the PDN and internal circuitry, perturbing supply voltages or timing and thereby producing faults or timing violations. By contrast, impedance side-channel techniques exploit the *reflected* or radiated signatures that vary with device state. Hence, they rely on frequency bands where state-dependent reflection is prominent. Impedance-based SCA commonly probes GHz bands where reflections and scattering encode state-dependent information that can be measured externally.

7 Conclusion

In this work, we revisited the conventional assumption that voltage glitching produces spatially uniform faults across a die. By leveraging concepts from EMI analysis, we modeled voltage fault injection as the transfer of conducted electromagnetic energy through the PDN. Our frequency-domain characterization revealed that the PDN behaves as a non-uniform communication channel, in which different frequency components of an injected signal propagate along distinct paths and affect specific regions of the chip. Through extensive post-silicon profiling on FPGAs equipped with distributed TDCs, we demonstrated that the susceptibility of on-chip circuits to voltage faults strongly depends on both the injected signal frequency spectrum and the design's physical placement. The proposed framework enables systematic identification of PDN frequency vulnerabilities and mapping of spatial fault sensitivity across the die. We also showed that an adversary could potentially induce localized faults by carefully selecting the frequency of the injected signal. Proof-of-concept experiments on finite-state machines and AES implementations confirm the feasibility of such targeted attacks. Beyond this, our frequency-domain characterization contributes to a more accurate modeling of fault behavior and offers valuable guidance for the placement and design of countermeasures. Finally, experiments conducted across multiple commercial and industrial FPGAs reveal that temperature and process variations significantly influence the system's susceptibility to voltage-based fault attacks.

Acknowledgment

This work is supported in part by the NSF Grant CNS-2338069 and in part by the Under Secretary of Defense for Research and Engineering under Air Force Contract No. FA8702-15-D-0001 or FA8702-25-D-B002. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Under Secretary of Defense for Research and Engineering.

References

[ABC+07] Alberto Aloisio, Paolo Branchini, Roberta Cicalese, Raffaele Giordano, Vincenzo Izzo, and Salvatore Loffredo. FPGA Implementation of a High-Resolution Time-to-Digital Converter. In *2007 IEEE Nuclear Science Symposium Conference Record*, volume 1, pages 504–507. IEEE, 2007.

[AK96] Ross Anderson and Markus Kuhn. Tamper Resistance – A Cautionary Note. In *Proceedings of the Second USENIX Workshop on Electronic Commerce*, volume 2, pages 1–11, 1996.

[Aru25] Arusson. DFA-AES: Efficient C Implementation of Differential Fault Analysis on AES (Round 8 or 9). https://github.com/arusson/dfa-aes, 2025. GitHub repository, last updated 2025-01-04, accessed 2026-01-30.

[ATG+19] Md Mahbub Alam, Shahin Tajik, Fatemeh Ganji, Mark Tehranipoor, and Domenic Forte. RAM-Jam: Remote Temperature and Voltage Fault Attack on FPGAs Using Memory Collisions. In *2019 Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC)*, pages 48–55. IEEE, 2019.

[BFP19] Claudio Bozzato, Riccardo Focardi, and Francesco Palmarini. Shaping the Glitch: Optimizing Voltage Fault Injection Attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 199–224, 2019.

[BJKS21] Robert Buhren, Hans-Niklas Jacob, Thilo Krachenfels, and Jean-Pierre Seifert. One Glitch to Rule Them All: Fault Injection Attacks Against AMD's Secure Encrypted Virtualization. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 2875–2889, 2021.

[BKGS21] Otto Bittner, Thilo Krachenfels, Andreas Galauner, and Jean-Pierre Seifert. The Forgotten Threat of Voltage Glitching: A Case Study on NVIDIA Tegra X2 SoCs. In *2021 Workshop on Fault Detection and Tolerance in Cryptography (FDTC)*, pages 86–97. IEEE, 2021.

[Bog10] Eric Bogatin. *Signal and Power Integrity–Simplified*. Pearson Education, 2010.

[DLV03] Pierre Dusart, Gilles Letourneux, and Olivier Vivolo. Differential Fault Analysis on AES. In *International Conference on Applied Cryptography and Network Security*, pages 293–306. Springer, 2003.

[DMT25] Fatemeh Khojasteh Dana, Saleh Khalaj Monfared, and Shahin Tajik. Logical Maneuvers: Detecting and Mitigating Adversarial Hardware Faults in Space. *Workshop on the Security of Space and Satellite Systems*, 2025.

[EMV+23] Mohammad Ebrahimabadi, Suhee Sanjana Mehjabin, Raphael Viera, Sylvain Guilley, Jean-Luc Danger, Jean-Max Dutertre, and Naghmeh Karimi. DELFINES: Detecting Laser Fault Injection Attacks via Digital Sensors. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(3):774–787, 2023.

[FH21] Daisuke Fujimoto and Yuichi Hayashi. Investigation of the Effect of Temperature on Fault Injection Using Intentional Electromagnetic Interference. In *2021 Asia-Pacific International Symposium on Electromagnetic Compatibility (APEMC)*, pages 1–3. IEEE, 2021.

[FHO21] Christian Fibich, Martin Horauer, and Roman Obermaisser. Device and Temperature Dependency of Systematic Fault Injection Results in Artix-7 and iCE40 FPGAs. In *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1600–1605. IEEE, 2021.

[FXHH23] Tanner Fokkens, Shengxuan Xia, Aaron Harmon, and Chulsoon Hwang. Coupling Path Analysis for Smart Speaker Intentional Electromagnetic Interference Attacks. In *2023 IEEE Symposium on Electromagnetic Compatibility & Signal/Power Integrity (EMC+ SIPI)*, pages 488–494. IEEE, 2023.

[GOT17] Dennis RE Gnad, Fabian Oboril, and Mehdi B Tahoori. Voltage Drop-Based Fault Attacks on FPGAs Using Valid Bitstreams. In *International Conference on Field Programmable Logic and Applications (FPL)*, pages 1–7. IEEE, 2017.

[HL15] Huanyu He and James Jian-Qiang Lu. Modeling and Analysis of PDN Impedance and Switching Noise in TSV-Based 3-D Integration. *IEEE Transactions on Electron Devices*, 62(4):1241–1247, 2015.

[HSM25] Shungo Hayashi, Junichi Sakamoto, and Tsutomu Matsumoto. Design Methodology of Digital Sensors for Detecting Laser Fault Injection Attacks in FPGAs. *Journal of Cryptographic Engineering*, 15(2):12, 2025.

[Inc18] NewAE Technology Inc. CW305 Artix FPGA Target Datasheet. https://media.newae.com/datasheets/NAE-CW305_datasheet.pdf, 2018. Online; accessed 2025-09-22.

[JUP24] Darshana Jayasinghe, Brian Udugama, and Sri Parameswaran. 1LUTSensor: Detecting FPGA Voltage Fluctuations Using LookUp Tables. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2024(1):51–86, 2024.

[Key25] Keysight Technologies. N9310A RF Signal Generator. https://www.keysight.com/us/en/product/N9310A/n9310a-rf-signal-generator.html, 2025. [Online; accessed 2025-09-22].

[KGT18] Jonas Krautter, Dennis RE Gnad, and Mehdi B Tahoori. FPGAhammer: Remote Voltage Fault Attacks on Shared FPGAs, Suitable for DFA on AES. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 44–68, 2018.

[KGT20] Jonas Krautter, Dennis Gnad, and Mehdi Tahoori. CPAmap: On the Complexity of Secure FPGA Virtualization, Multi-Tenancy, and Physical Design. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 121–146, 2020.

[KK99] Oliver Kömmerling and Markus G Kuhn. Design Principles for Tamper-Resistant Smartcard Processors. *Smartcard*, 99:9–20, 1999.

[KMFT24] Saleh Khalaj Monfared, Domenic Forte, and Shahin Tajik. RandOhm: Mitigating Impedance Side-Channel Attacks Using Randomized Circuit Configurations. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pages 1–9, 2024.

[KMMC+24] Saleh Khalaj Monfared, Kyle Mitard, Andrew Cannon, Domenic Forte, and Shahin Tajik. LaserEscape: Detecting and Mitigating Optical Probing Attacks. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pages 1–10, 2024.

[KMSST25] Saleh Khalaj Monfared, Maryam Saadat Safa, and Shahin Tajik. ChipletQuake: On-Die Digital Impedance Sensing for Chiplet and Interposer Verification. *Sensors*, 25(15):4861, 2025.

[KWJS25] Niclas Kühnapfel, Christian Werling, Hans Niklas Jacob, and Jean-Pierre Seifert. Three Glitches to Rule One Car: Fault Injection Attacks on a Connected EV. In *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*, pages 1235–1249, 2025.

[LFH10] Sungkeun Lim, Jiwei Fan, and Alex Q Huang. Transient-Voltage-Clamp Circuit Design Based on Constant Load Line Impedance for Voltage Regulator Module. *IEEE Transactions on Industrial Electronics*, 57(12):4085–4094, 2010.

[LI23] Stanislav Lyakhov and Vincent Immler. Analysis of Arbitrary Waveform Generation for Voltage Glitches. In *2023 Workshop on Fault Detection and Tolerance in Cryptography (FDTC)*, pages 36–47. IEEE, 2023.

[LLC09] TestEquity LLC. Model 107 Temperature Chamber Operation and Service Manual. https://assets.testequity.com/te1/Documents/pdf/testequity/TE-107_Operation_Service_Manual_Current.pdf, 2009. accessed 2025-10-02.

[MC25] Mini-Circuits. Mini-Circuits Datasheets. https://www.mouser.com/datasheet/2/1030/CBL_2FT_SMNM_2b-2303455.pdf, 2025. [Online; accessed 2025-09-22].

[Min22] Mini-Circuits. eVNA-63+ Vector Network Analyzer Datasheet. https://www.minicircuits.com/pdfs/EVNA-63+.pdf, 2022. Accessed: 2025-10-02.

[MMST23] Tahoura Mosavirik, Saleh Khalaj Monfared, Maryam Saadat Safa, and Shahin Tajik. Silicon Echoes: Non-Invasive Trojan and Tamper Detection Using Frequency-Selective Impedance Analysis. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2023(4):238–261, 2023.

[MMT23] Saleh Khalaj Monfared, Tahoura Mosavirik, and Shahin Tajik. LeakyOhm: Secret Bits Extraction Using Impedance Analysis. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1675–1689, 2023.

[MS19] Dina Mahmoud and Mirjana Stojilović. Timing Violation Induced Faults in Multi-Tenant FPGAs. In *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1745–1750. IEEE, 2019.

[New25a] NewAE. CW305 Artix FPGA Target. https://rtfm.newae.com/Targets/CW305ArtixFPGA, 2025. [Online; accessed 2025-09-22].

[New25b] NewAE. NewAE Hardware Product. https://rtfm.newae.com/Capture/ChipWhisperer-Lite, 2025. [Online; accessed 2025-09-22].

[New26] NewAE Technology Inc. ChipWhisperer AES Source Code. https://github.com/newaetech/chipwhisperer/tree/develop/firmware/fpgas/cryptosrc/aes_googlevault, 2026. GitHub repository directory, accessed 2026-01-14.

[PHT19] George Provelengios, Daniel Holcomb, and Russell Tessier. Characterizing Power Distribution Attacks in Multi-User FPGA Environments. In *International Conference on Field Programmable Logic and Applications (FPL)*, pages 194–201. IEEE, 2019.

[PHT20] George Provelengios, Daniel Holcomb, and Russell Tessier. Power Distribution Attacks in Multitenant FPGAs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 28(12):2685–2698, 2020.

[PMBC21] Michael Paquette, Brian Marquis, Rachel Bainbridge, and Joe Chapman. Visualizing Electromagnetic Fault Injection with Timing Sensors. In *2021 IEEE Physical Assurance and Inspection of Electronics (PAINE)*, pages 1–8. IEEE, 2021.

[Pup20] Peter J Pupalaikis. *S-parameters for Signal Integrity*. Cambridge University Press, 2020.

[SAF+99] Larry D Smith, Raymond E Anderson, Douglas W Forehand, Thomas J Pelc, and Tanmoy Roy. Power Distribution System Design Methodology and Capacitor Selection for Modern CMOS Technology. *IEEE Transactions on Advanced Packaging*, 22(3):284–291, 1999.

[SAL06] Jian Song, Qi An, and Shubin Liu. A High-Resolution Time-to-Digital Converter Implemented in Field-Programmable-Gate-Arrays. *IEEE Transactions on Nuclear Science*, 53(1):236–241, 2006.

[SGS23] David Spielmann, Ognjen Glamočanin, and Mirjana Stojilović. RDS: FPGA Routing Delay Sensors for Effective Remote Power Analysis Attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 543–567, 2023.

[SMS23] Xhani Marvin Saß, Richard Mitev, and Ahmad-Reza Sadeghi. Oops..! I Glitched It Again! How to Multi-Glitch the Glitching-Protections on ARM TrustZone-M. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 6239–6256, 2023.

[SWO+24] Marc Schink, Alexander Wagner, Felix Oberhansl, Stefan Köckeis, Emanuele Strieder, Sven Freud, and Dominik Klein. Unlock the Door to my Secrets, but Don't Forget to Glitch: A Comprehensive Analysis of Flash Erase Suppression Attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2024(2):88–129, 2024.

[Sys25] Espressif Systems. Introduction to the ESP-Prog Board. https://docs.espressif.com/projects/esp-iot-solution/en/latest/hw-reference/ESP-Prog_guide.html, 2025. Accessed: 2025-10-05.

[TMA11] Michael Tunstall, Debdeep Mukhopadhyay, and Subidh Ali. Differential Fault Analysis of the Advanced Encryption Standard Using a Single Fault. In *IFIP International Workshop on Information Security Theory and Practices*, pages 224–233. Springer, 2011.

[XBZ+19] Jun Xu, Siqi Bai, Biyao Zhao, Kartheek Nalla, Mike Sapozhnikov, James L Drewniak, Chulsoon Hwang, and Jun Fan. A Novel System-Level Power Integrity Transient Analysis Methodology Using Simplified CPM Model, Physics-Based Equivalent Circuit PDN Model and Small Signal VRM Model. In *2019 IEEE International Symposium on Electromagnetic Compatibility, Signal & Power Integrity (EMC+ SIPI)*, pages 205–210. IEEE, 2019.

[YKS+23] Yuan Yao, Pantea Kiaei, Richa Singh, Shahin Tajik, and Patrick Schaumont. Programmable RO (PRO): A Multipurpose Countermeasure Against Side-Channel and Fault Injection Attack. In *Security of FPGA-Accelerated Cloud Computing Environments*, pages 297–325. Springer, 2023.

[ZAB+18] Shuze Zhao, Ibrahim Ahmed, Vaughn Betz, Ashraf Lotfi, and Olivier Trescases. Frequency-Domain Power Delivery Network Self-Characterization in FPGAs for Improved System Reliability. *IEEE Transactions on Industrial Electronics*, 65(11):8915–8924, 2018.

[ZGJZ23] Huifeng Zhu, Xiaolong Guo, Yier Jin, and Xuan Zhang. PowerScout: Security-Oriented Power Delivery Network Modeling for Side-Channel Vulnerability Analysis. *IEEE Transactions on Emerging Topics in Computing*, 12(2):532–545, 2023.

[ZSZF13] Kenneth M Zick, Meeta Srivastav, Wei Zhang, and Matthew French. Sensing Nanosecond-Scale Voltage Attacks and Natural Transients in FPGAs. In *Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, pages 101–104, 2013.