RAD-FS: Remote Timing and Power SCA Security in DVFS-Augmented *Ultra-Low-Power* Embedded Systems

DANIEL DOBKIN[**], Bar-Ilan University, Faculty of Engineering, Israel
NIMROD CEVER[*], Bar-Ilan University, Faculty of Engineering, Israel
ITAMAR LEVI[*], Bar-Ilan University, Faculty of Engineering, Israel

High-performance crypto-engines have become crucial components in modern System-On-Chip (SoC) architectures across platforms, from servers to edge-IoTs'. Alas, their secure operation faces a significant obstacle caused by information-leakage through various side-channels. Adversaries exploit statistical-analysis techniques on measured (e.g.,) power and timing signatures generated during (e.g.,) encryption, extracting secrets. Mathematical countermeasures against such attacks often impose substantial power-performance-area overheads. Adaptive Dynamic Voltage and Frequency Scaling (ADVFS) techniques provide power-efficiency by varying power consumption according to workload; these modulations are called power-states. Unintentionally, ADVFS introduces new inherent weaknesses exploitable by malicious actors: power-states *leaks* information in both power and timing side-channels, measurable in software and hardware. We introduce a method to increase side-channel resistance using integrated voltage regulators and DVFS: (1) Pushing known prior-art in the topic to ULP-regime (2) For the first time introducing a mechanism to aid in counteracting the inherent weakness of DVFS in SCA (3) Provide measurements performed on 40nm process ULP PLS15 test-chip down at 580 mV power-supply (4) Offering improved and parameterized resistance to *remote*-timing vulnerabilities inherent to DVFS. Various results and detailed analysis is presented, performance-cost and comparison to prior-art. Importantly, our solution is configurable in terms of security, maintaining degrees-of-freedom for power-optimization of DVFS.

CCS Concepts: • **Security and privacy** → **Hardware security implementation**; **Cryptanalysis and other attacks**; Embedded systems security; **Side-channel analysis and countermeasures**; **Cryptography**; **Web application security**; *Hardware-based security protocols*; • **Hardware** → *Hardware test*; *System on a chip*; Signal integrity and noise analysis.

ACM Reference Format:
Daniel Dobkin, Nimrod Cever, and Itamar Levi. 2024. RAD-FS: Remote Timing and Power SCA Security in DVFS-Augmented *Ultra-Low-Power* Embedded Systems. *ACM Trans. Embedd. Comput. Syst.* 1, 1, Article 1 (June 2024), 25 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Side-Channel Analysis (SCA) attacks [7, 20, 37] are a category of hardware security attacks, which extract or retrieve information from a system by utilizing non-standard channels as compared to conventional communication interfaces. These non-standard channels are *leaking* sensitive information manipulated by the digital electronic system or because of the physical implementation of theoretical cryptographic systems. Dynamic

[*] All authors contributed equally to this research.

Authors' Contact Information: Daniel Dobkin, Bar-Ilan University, Faculty of Engineering, Ramat-Gan, Israel, daniel.dobkin@live.biu.ac.il; Nimrod Cever, Bar-Ilan University, Faculty of Engineering, Ramat-Gan, Israel, Nimrod.cever@biu.ac.il; Itamar Levi, Bar-Ilan University, Faculty of Engineering, Ramat-Gan, Israel, itamar.levi@biu.ac.il.

ACM 1558-3465/2024/6-ART1
https://doi.org/XXXXXXX.XXXXXXX

voltage and frequency scaling (DVFS) [3, 22, 32, 44] is a widespread technique utilized to save power on a wide range of computing systems, from tiny Internet of Things (IoT) devices, embedded systems, laptops/desktops systems, high-performance server systems, etc, by reducing the frequency at which they operate. Vast literature and research exists on the (very successful) use of DVFS to improve the energy efficiency of computation systems. This is done by adapting the voltage/frequency to the workloads of the system jointly constrained with the general timing limitations. Various modern microprocessors (if not all) and embedded-systems are equipped with the DVFS functionality such as chips from NVIDIA, AMD, Intel [1, 27, 38]. In some devices one can find hundreds of voltage/frequency levels with high resolution, controllable through software such as MSI Afterburner [43] and different parts of the systems are adapting independent sub-systems, with their own *optimized* DVFS tactic, such as ARM big.LITTLE [19, 31].

Side-channel Analysis (SCA) attacks security is a critical requirement. Typical mechanisms to counteract SCA's in a mathematically rigorous way are very expensive. For example, by coding an internal value or variables in the computation to a redundant representation which is randomized and invertible, denoted by masking [12, 34]. On the less rigorous approach, more heuristic solutions exist utilizing other randomization mechanisms such as randomizing the time [40] or amplitude [23] signal domains. For time-variations, various Shuffling mechanisms were recently supported with theoretical models indicating quasi-linear security-levels [24]. For amplitude randomization mechanisms, a fine-grain leakage model supported by silicon measurement was recently provided showing also quasi-linear security levels [6]. The electronic-cost of both techniques is far lower than that of Masking, but each approach has its own limitations such as algorithmic-dependence or the need to embed specialized components, and more. Several other countermeasures, which rely on randomizing voltage / frequency are described in [17, 21, 36], but requires significant IC redesign, including (e.g.,) an all-digital clock modulation (ADCM), using its global modulator; On the contrary, the proposed approach is inherently embedded in such platforms and is software (SW) controlled, also pushing the limits towards ULV embedded systems.

ADVFS introduces inherent weaknesses exploitable by malicious actors. The power states (P-state) are easy to classify and categorise, as shown by a new class of software enabled attacks [41]. The hamming weight (HW) of calculated data causes variations in P-state that leak significantly. We offer a solution that co-exists with power optimization (Section 4) while increasing side channel resistance in both power and timing (illustrated in Fig. 15 and 25) by orders of magnitude in software and hardware. Our approach is optimized towards ULP embedded devices, sacrificing performance for null cost in terms of power, area and implementation effort (Tab. 2). Randomly assigning P-states from a group of frequencies instead of the single frequency approach as described in Fig. 3, **will drastically reduce requirements from algorithmic hiding schemes [23] and enable additional degrees of freedom to system designers**. It is important to note that the utilization of the proposed mechanism is orthogonal to any other multi-layer approach which in addition embeds more security measures (which is reasonable), such as by low-order Masking in software or any other software / hardware or architectural countermeasure. However, the cost of the proposed mechanism is tiny on all electronic-cost terms; meaning, it already provides several nice features for such ULP devices which cannot bear higher countermeasures cost and need remote-timing and power-SCA security.

1.1 Contribution

In this research we push forward three observations: (1) nowadays embedded-systems embed efficient DVFS mechanisms inherently which can be utilized to integrate security features without special design efforts and hardware intervention (2) such mechanisms can be very efficient, as supported by rigorous evaluation to counteract SCA, and (3) native DVFS mechanisms inherently induce other SCA channels (timing and power-state monitoring) which becomes data/workload-dependent, our proposed mechanisms can potentially aid in mitigation of these channels. We provide for the first-time significant advance on the analysis of all these aspects and we showcase several SOTA use cases on a very advanced platform, extending significantly the body of knowledge in such ULP regime. **(1):** Working with a SOTA ULP embedded device, where scarce platforms exist in the market going down to $500mV$ in operational chips **(2):** Providing security analysis over such devices for the first time to the best of our knowledge. **(3):** Proposing unique methodology and embedded mechanisms to provide SCA security utilizing the inherent DVFS features with ultra low-cost as compared to other solutions regarding area, latency, power, implementation effort overheads; denoted Randomized Aliasing Dynamic Frequency Scaling (RAD-FS). **(4):** Reporting analysis both for a Risc-V processor core and an NXP encryption accelerator embedded on the same device in 40 *nm* technology, in a comparative view. **(5):** For the first time we demonstrate a countermeasure to a new class of timing-attacks such as [41], which coexists with DVFS mechanisms (opposed to the naive solution of turning DVFS off), in addition to the inherent power-SCA attacks immunity. We show RAD-FS is very relevant for such network remote timing attacks mitigation. Results are demonstrated via. an ideal (optimal) oracle modeling the RAD-FS parameters. Our constructed oracle is very *generous* with how much control is given to the adversary.

The outline of this paper is as follows. In Section II, we present a background and literature review. In Section III we introduce the remote timing vulnerability inherent in DVFS. In Section IV, we present our novel solution with a detailed explanation and comparison to the current industry approach. Section V details the evaluated device and the measurement setup. Section VI details security analysis metrics. Section VII contains in-depth analysis of the results and an analytical approach to the performance cost. Section VIII discusses comparisons and challenges that adversaries face in a real life scenario. Section IX concludes the results of this work and proposes future prospects.

2 BACKGROUND

2.1 Side Channel Analysis

SCA attacks are powerful, repeatedly showing their efficiency in extracting sensitive information from a cryptographic system by analyzing unintentional side channels. For example, the timing of signals [41] or power consumption [26, 29], and electromagnetic (EM) radiation [2, 33] of a device. Generally, attacks are classified as model-based or profiling-based: model-based attack such as correlation power analysis (CPA) [8] utilizing Pearson's correlation coefficient ρ and a leakage model to compare side channel leakages to the model. Template attacks [14] utilize a template or a statistical model derived from the leakage of a specific internal value used to enhance the attack's effectiveness in an attack campaign by comparing it to the actual leakage. The costs of SCA countermeasures can be sublinear, linear [23, 24] or exponential [12, 34] and are very hard to embed with a strict energy budget. This is witnessed by the requirements of the NIST lightweight authenticated-encryption contest [39]; especially for low-power constrained or battery operated IoT devices that can not sacrifice energy budget as needed for rigorous security-level utilizing e.g., Masking.

2.2 Dynamic Voltage Frequency Scaling

The concept of Dynamic Voltage and Frequency Scaling (DVFS) is based on the understanding that the energy consumption of a component is directly proportional to its supply voltage, while the computation delay is inversely proportional to the square of its operating frequency (depicted in Fig. 3(a)). Stemming from strong-inversion/near-threshold current equations of transistors. By controlling the supply voltage or clock frequency, it is possible to achieve significant reductions in energy consumption. Frequency randomization is a technique used to introduce randomness or unpredictability in the occurrence or timing of events. It is often employed in various domains, including communication systems, network security, and data privacy, to mitigate potential attacks or reduce vulnerabilities. By randomizing the frequency of events, it becomes harder for adversaries to analyze patterns or launch coordinated attacks.

Double-edged sword: While DVFS has obvious merits, as hinted above, it introduces security-flaws by its design. Workloads, manifested by different data manipulations, affect the electrical characteristics of an operation (voltage, latency etc.), leading to various sensitivities. As opposed to conventional SCA attacks that measure power or electromagnetic emissions and require a *close-contact* adversary, DVFS also allows for SW based attacks by various mechanisms: For instance, Hertzbleed [41] represents a novel category of side-channel attacks that exploit network timing and latency profiles in the spectrum. These vulnerabilities can be exploited regardless of the physical distance. Hertzbleed Has demonstrated extraction of cryptographic keys from remote servers which embed modern x86 CPU's in a scenario that was previously believed to be secure; *solely exploiting the inherent sensitivity DVFS offers.* Another example is [30] in which the voltage dependence offered by DVFS on an iPhone-13 affects the overall power of the device, and a video footage of a device's power-LED, radiating to a long distance can be used for SCA. Alternatively, the authors have shown that if the device is powered by a USB-hub, the hub's current draw can provide an attack entry. Noteworthy, in these reports the attacker makes use of an ultra low-resolution side-channel (as compared to high-resolution e.g., power-based SCA): The device's embedded power-sensor, or a low resolution web-camera. As an example, its resolution is clearly far from being as high as of a conventional 10 to 14-bit quantizer of an oscilloscope. In addition, it is typically a very slow sensor hence significant averaging and noise is incorporated in this physically measurable quantity. *Therefore, as will be discussed below it will be quite easy for our randomization mechanism to make these attacks hard.*

2.3 Ultra Low Power Regime and Design

In processors implemented with standard process technologies, the operating voltage-span ranges from V_{dd} overdrive of several hundreds of mV over the nominal voltage and down to still strong inversion / high near-threshold region of transistors, only several hundreds of mV under the nominal voltage. Implementing electronic systems that operate over a much larger voltage span, from over V_{dd} to sub-threshold voltages requires special (and not always standard) design-techniques such as embedding isolation-cells, level-shifters, and special cells libraries (high-V_{Th}) etc. The main challenge is that typically such devices are not fully characterized by the foundry in the entire voltage-span, making it hard to design robustly and perform full-digital design-flow with guarantees. Nevertheless, this possibility opens the question of DVFS efficiency in such extended ranges. Existing theoretical research have proven that further energy efficiency is possible with extended voltage-range, even ranging below $0.5 \cdot V_{dd}$. However, extending it to full-sub-threshold region is beneficial only for specific application classes. Therefore, in practice, with the emergence of Ultra-Low-Power (ULP) devices which sacrifice performance for longer (say) battery-life, we see unique ULP embedded platforms that are designed to work in 500-600 mV (generally mid to high Near-threshold operation). Such systems heavily utilize DVFS techniques. *Designing such systems is not an easy task and require high-expertise, but available devices exist on the market which we believe will provide game-breaking abilities for (e.g.,) IoTs.* One such unique platform is PLSense's PLS15 platform, used in this research.

Overall, the development of fully functional ULV standard cells has become crucial in meeting the growing demand for energy-efficient and low-power electronic systems.

2.4 Adversary and Threat Model

The adversary is modeled by a Probabilistic Polynomial Time (PPT) algorithm. Utilizing different assumptions and capabilities per scenario.

2.4.1 Power Analysis.

For Sub-sections 7(a-d) we assume the adversary has eavesdropper capabilities as well as SCA trace access (such as power leakages) as illustrated in Fig. 1. In our scenario the adversary has physical access to a device, with knowledge of the whereabouts of some power supply pin. The adversary requires no

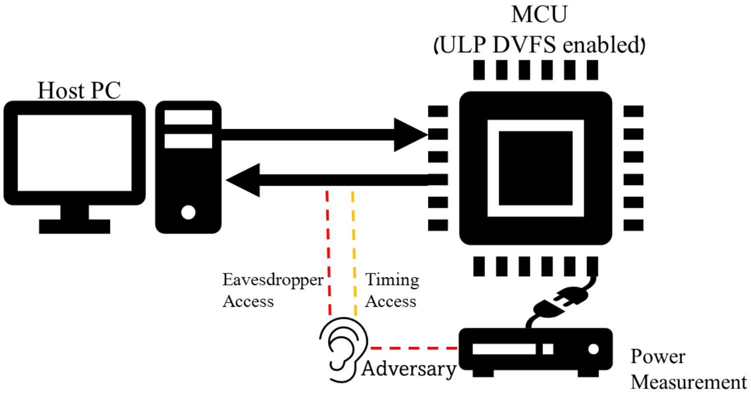

Fig. 1. Attack and adversary model where different eavesdropper and measurement capabilites are assumed depending on the scenario.

assumptions about the internals of the device or architectural knowledge, nor do we conceal any information regarding the DVFS mechanism / algorithm.

2.4.2 Time-Domain Attacks. For Sub-section 7(e) we assume the adversary has access only to the communication channel and measures response-time. The adversary assumes a DVFS module that introduces some dependency between average-HW values in the plaintext to power-state (e.g., for concrete attacks utilizing such a scenario [41]), which becomes apparent in time domain measurements, evaluated as time between a request from the host and the reply from the DVFS-enabled MCU. The goal set for the adversary is extracting secret information (values) from the time-domain distribution. Noteworthy, timing information is trivial to extract when countermeasures are not embedded [9, 10].

3 REMOTE TIMING ATTACKS

Remote timing attacks are devastatingly effective. Such attacks can extract secret information from a target device *remotely* [41], on both symmetric and asymmetric [42] encryption schemes. It is known that the execution time for various algorithms utilizing (e.g.,) modular exponentiation or multiplication [5, 11], depend linearly on the number of '1' bits in the key or state [28], Demanding for *Constant-Time* codes. However, even when some countermeasures are taken (e.g non-conditional branching, pipe-lining) they do not ensure that logical '1' and '0' are the same power wise, which may still be an issue with DVFS enabled systems and the associated remote timing attack as discussed here. In contrast, Hertzbleed [41, 42] focuses on the tendency of DVFS to change clock speeds based on manipulated data rather than making assumptions about some encryption scheme. In another scenario, adversaries may target the p-states themselves rather then assume some key-dependent timing dependency in some encryption scheme. Adversaries need only man-in-the-middle access to execute such attacks, i.e., extracting some random message sent by the client over the channel and measuring the time-to-response (TTR) from the host. In a variation of the exploit described by [4], we emulate a non-secure system, the frequency set by the DVFS in our PLS chip depends on some inputs of the computation sent (controlled) by the host for a clear illustration (measured from our platform).

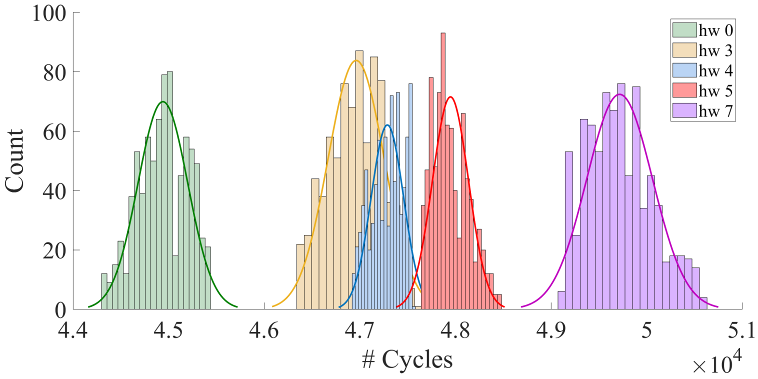

Fig. 2. Time distribution fitting for 100 encryption requests per data point

Fig. 2 shows the **power-state** leaking with Gaussian-proximate distributions, a clear vulnerability that could be utilized by adversaries.

4 THE PROPOSED APPROACH

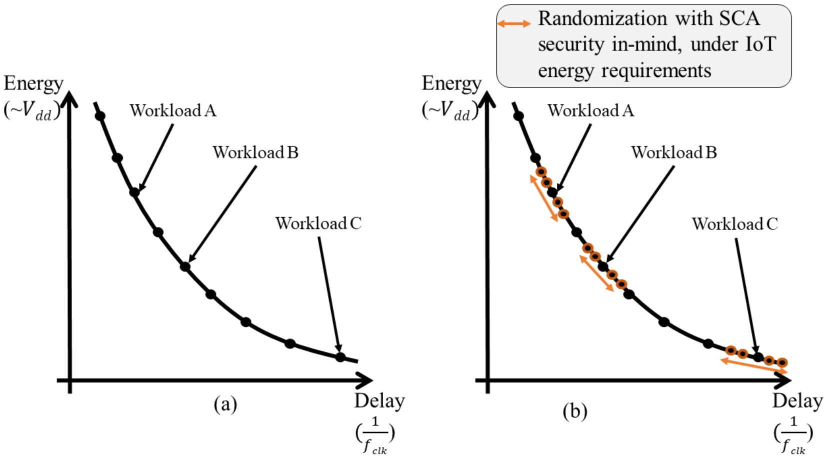

Fig. 3. DVFS conceptual implementation (a) insecure, single frequency per workload vs. (b) our proposed secure implementation, with multiple frequencies assigned per workload.

We introduce Randomized Aliasing Dynamic Frequency Scaling (RAD-FS), as a side-channel security mechanism. Our method suggests assigning a group of frequencies F to a workload from the generally available set of operation frequencies F' (i.e., F is a subset), instead of a single frequency f_{base} as illustrated in Fig. 3(b). F is constructed such that each $f_n \in F$ is within a bandwidth, BW, around f_{base}, allowing for power optimization in relation with workload. Denoting $||F||$ as the number of different $f_n \in F$.

For each $f_n, f_m \in F | n \neq m$ we achieve **aliased** distributions of the leakage induced by some internal value manipulation. Optimally, we aim for these disturbances to: (1) distribute uniformly to maximize the leakage entropy, and (2) overlap significantly to increase the noise-level for proximate time samples. As we increase $||F||$, adding more frequencies to our set F, we expect security-metrics e.g., the SNR, to show more distribution curves across different time samples, in correlation to various $f_n \in F$. These high leakage-correlation points-of-interest, POI's, reflect the now *shifted* value manipulation corresponding with f_n's. The overall noise increases due to cross-interference between different leakages stemming from different $f_n \in F$, as abstractly illustrated in Fig. 4. This should manifest in all uni-variate security metrics such as CPA, SNR, Template attacks and TVLA detection tests (We relate to other attack-settings in later sections).

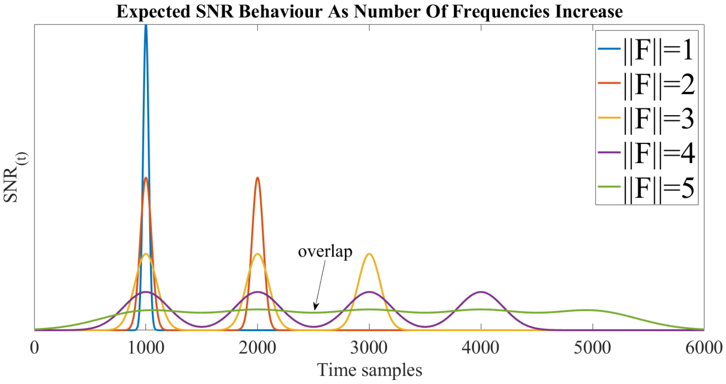

Fig. 4. Expectation for the number of peaks to increase along with group size $||F||$, while concurrently the magnitude (ρ, SNR etc.) of each peak decreases.

In practice, for every internal hypothesized calculation during e.g., an encryption, we find various POIs associated with some $f_n \in F$. In our evaluation environment (detailed below) implementing the AES cipher, it is possible to see 'ghost' peaks appearing with lesser amplitudes as $||F||$ goes up, shown in Fig. 5. The SNR value decreases, clearly owing to traces cross-interference of $f_n \neq f_m \in F$. Already, at this early stage, we can hint a significant order-of-magnitude improvement with only $||F|| = 5$.

Fig. 6 shows the FFT of several scenarios. In Fig. 6(a) we show the log-scale power of the single-sided spectrum when a single frequency is set ($||F|| = 1$). In Fig. 6(b) we construct F with the same frequencies used in Fig. 6(a) applying RAD-FS ($||F|| = 7$). The energy per frequency is significantly reduced as expected (note the logarithmic scale) and is distributed quasi-uniformly among $f_n \in F$. This implies that any filtering attempt will either eliminate information, induce overlaps in the time domain, and alternatively increase the noise-floor.

We regard such approach as a ultra low cost signal hiding countermeasure, as analysed in sections below. We also conjecture it may/should be combined with algorithmic countermeasures such as hiding / masking with low orders d for increased efficiency [23], depending on the security-level required. As the results show below, this approach alone, is found to be ultra efficient in mitigation of both remote-timing and power-SCA per its' cost.

5 THE EVALUATED DEVICE, TESTING MODIFICATIONS AND MEASUREMENT SETUP

5.1 The Evaluated Device

Our tests were performed on the PLS15, an advanced chip made by PLSense. The PLS15 is an ultra-low power MCU with multiple analog and digital interfaces and other capabilities like ML inference engine, crypto-cores, Risc-V processor and other ULP features making it a very interesting candidate for our experiment due to desirable features for IoT. The PLS15 is manufactured on the 40nm TSMC process. Usually, this process node has nominal

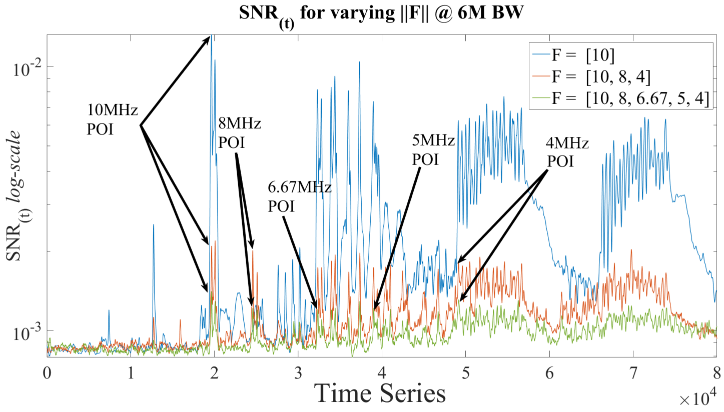

Fig. 5. Displaying with arrows when peak SNR values appear regarding to chosen $f_n \in F$, per case with different sized $\|F\|$. The Δ between the POI and the closest secondary peaks decreases, making it harder to choose a time sample for further analysis (i.e., template attacks).

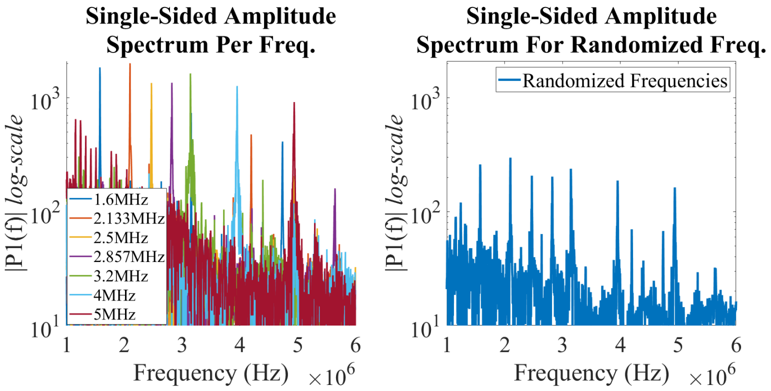

Fig. 6. FFT of different scenarios: (a) no randomization, single frequency $f_n \in F'$, $\|F\| = 1$ (b) Randomized scenario, $f_n \in F$, $\|F\| = 7$.

working voltage of 1.1V. The reason that the 40nm Low-Power (LP) process was chosen is owing to device-leakage current and dynamic power consumption savings of up to 51% compared to its 65nm counterpart. By utilizing mixed threshold-voltage (V_t) transistors in a single cell, supported by unique Adaptive Dynamic Voltage Control in the PLS15, the chip allows reduction of the operating voltage, bulk biasing, sensitivity to process variations, and more, to achieve a sub-threshold operating voltage of 0.45V-0.6V according to the workload conditions. Relevant blocks in the chip are a Risc-V Core, NXP AES Accelerator (unprotected), DMA controller and Adaptive-DVFS (ADVFS) Logic. Though some other devices exist on the market incorporating DVFS and ULP process towards IoTs, we didn't come across competitors reaching such deep near-threshold voltages in such a complex SoC. We have evaluated the voltage-frequency map of the PLS15 device as illustrated in Fig, 7 showing 19 discrete possible frequencies.

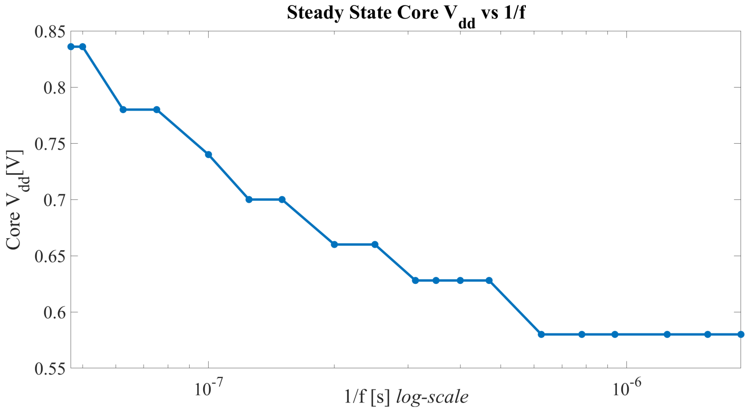

Fig. 7. Core voltage per 1/freq. available on our target device, PLS15

5.2 Testing Modifications

The PLS15 chip has an ADVFS controller module that affects the chip's core voltage. The ADVFS controls the input voltage to the core by utilizing an external Op-Amp in a negative feedback loop, connected to the power supply pin. We modified the PLS15 test kit board such that the ADVFS is semi-enabled; The ADVFS operation is in a mode where it operates based on SW commands only, and not according to workload. This gives us full control over data-dependency and P-states, allowing us to reduce algorithmic noise. In one scenarios, it allows us to play-out a scenario where the adversary has some form of access to the device DVFS. For power measurement, we added a jumper in serial between the external core voltage regulator and the core IO pin for induction-based current sensing, utilizing the Tektronix-CT1 current probe, connected through an amplifier to a Picoscope Oscilloscope as illustrated in Fig. 8.

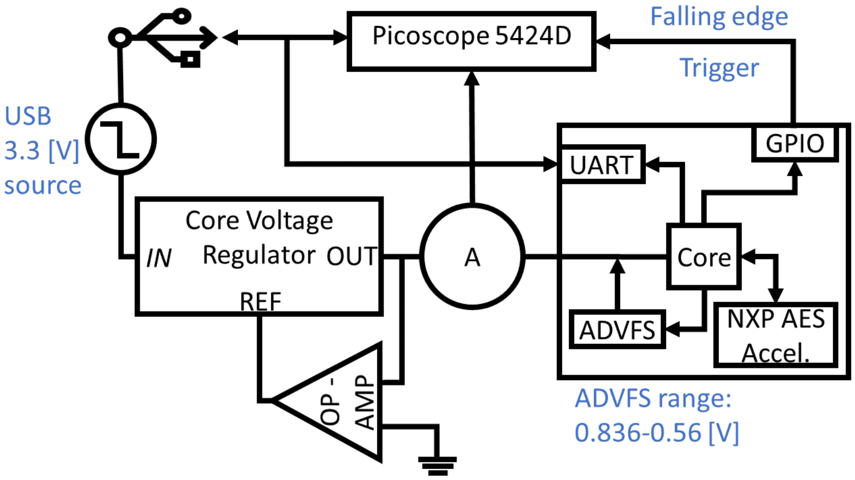

Fig. 8. Connection scheme of the core power supply and voltage regulation for the PLS15, with our current probe location.

5.3 Measurement Setup

Our test bench is composed of a PC, Picoscope 5424D, Rohde & Schwartz Signal Amplifier, Tektronix CT1 and PLS15 target connected as shown in Fig.9. The measurement setups is shown in Fig. 10.

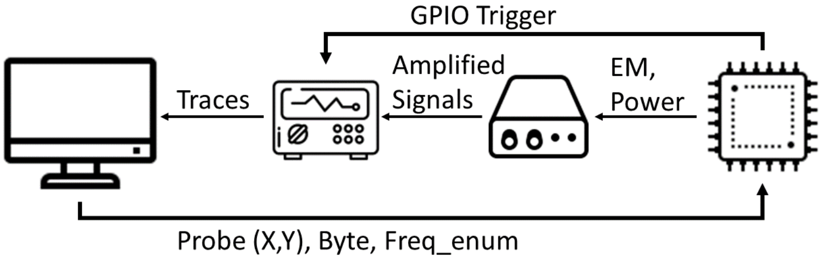

Fig. 9. Illustration of the test bench we assembled comprising a PC running a python script, Picoscope which records measurements of power and EM traces, Signal amplifier, Riscure Probing Station, PLS15 Chip target. Icons from [18]

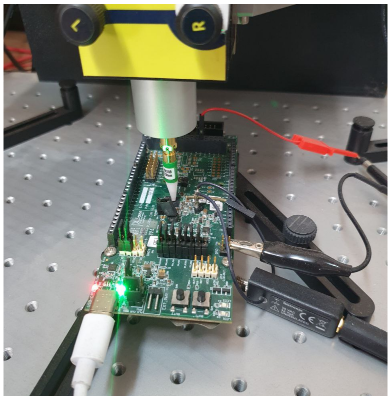

Fig. 10. Measurement setup with the PLS15 evaluation-board, the CT-1 current probe connected to a power jumper.

Our experiment method pseudo-code is described in Fig.26 of Appendix 10. For the basic SCA evaluation we wanted to eliminate algorithmic noise and to show the adversary best-case scenario. Hence, we evaluate the leakage stemming from one byte manipulation leakage at a time (e.g., Byte 7 in the aforementioned figure). We also control the operating frequency driven by the regulator through our python shell triggering the device, and we have added NOP cycles inline Assembly to our C implemented code loaded to the device, to reduce noise after a triggering event.

6 SECURITY ANALYSIS METRICS

6.1 Low Complexity Adversary - Estimators

As discussed above we evaluate (as a starting step) Mangard's SNR [25] & Brier's CPA [8] correlation as defined by:

$$\mathrm{SNR(t)} = \frac{\mathrm{Var}_{x_i,k}(\mathrm{E}[l^t_{x_i,k}])}{\mathrm{E}_{x_i,k}(\mathrm{Var}_i[l^t_{x_i,k}]} \tag{1}$$

$$\rho_{\mathrm{l}^{\mathrm{t}}_{\mathrm{x_i,k}},\mathrm{h}^{\mathrm{t}}_{\mathrm{x_i,k^*}}}(\mathrm{t}) = \frac{\mathrm{Cov}(l^t_{x_i,k^*}, h^t_{x_i,k^*})}{\sigma_{l^t_{x_i,k}} \sigma_{h^t_{x_i,k^*}}} \tag{2}$$

where, Var and E are the variance and expected estimators, $l^t_{x_i,k}$ is the leakage trace l in point in time samples t, taken from a cryptographic operation processing key k and plaintext (e.g.,) byte x_i.
In accordance with the correlation CPA distinguisher, we enumerate all possible (sub-) keys hypothesis k^* so as to generate the leakage hypothesis h. Then the k^* which maximizes the correlation is estimated to be the correct key. We then compare:

$$SNR_{t=POI} = max_t(|SNR|) \tag{3}$$

$$Corr_{t=POI} = max_t(|Corr|) \tag{4}$$

From now referred to as Point Of Interest (POI) for the SNR & Corr (CPA's ρ) accordingly. Both estimators were computed over $0.5{\cdot}10^6$ to $10{\cdot}10^6$ traces (as needed) or queries. Our results include both a Risc-V implementation of tiny-AES 128-bit code, verified against NIST [15], and an NXP crypthash hardware (HW) accelerator, embedded in the PLS15 SoC, running 128-bit AES as well. For illustration, Fig. 11 shows the mean power-trace of 10K leakage traces of the fast HW accelerator to the left and the slow SW implementation to the right. As further example Fig. 12 shows side-by-side the SNR's of the NXP AES accelerator and the SW AES on the Risc-V processor in a comparative view, to the left and right, respectively. Note the very high SNR value achieved and the ultra fast operation of the accelerator owing to the fact that it is *seated* in its own power domain in a tailored IP block and not as part of a *sea-of-gates* as is typically the case for processors-cores.

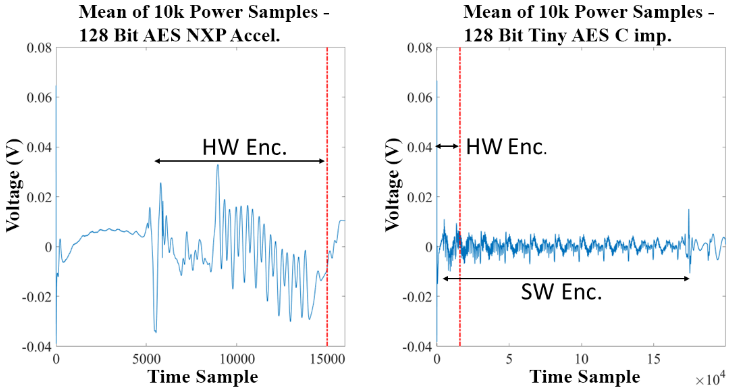

Fig. 11. Mean of 10k traces 20MHz @ core frequency. NXP crypthash accelerator running 128-bit AES (left), 128-bit tiny AES C implementation (right). The vertical red dashed line marks the end of the HW AES accelerator relatively to the SW one.

6.2 Detection Test - TVLA

As a detection test we apply TVLA verified against [35] to traces measured on our system, in order to show the difference in populations and the shift of data to higher statistical momentum. The compared populations are constant plaintext & a varying plaintext, this experiment set is run several times under different conditions.

6.3 High complexity adversary - Templates

Template attacks [13] are performed in two consequent (or interleaved) phases of *profiling* and *attack*. It is assumed that the adversary got hold of one device for which he can program (or control) the secret key and therefore

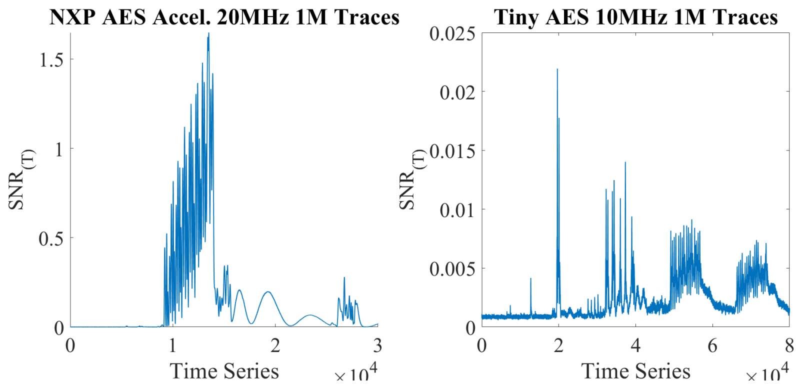

Fig. 12. SNR of the NXP AES accelerator **to the left** and the software Risc-V AES **to the right** under evaluation.

profile the leakage, and another target device from which he tries to extract information on the underlying key. We built a probability density function (PDF), for an internal manipulation, represented by function F, $y = \mathrm{f}(x_i, k)$. A set of $\mathcal{L}_p$ profiling traces of size N_p was used in order to estimate distributions, denoted as $\hat{M_y}$. Specifically, $\mathrm{f}\,(l \mid y) = \mathcal{N}\left(\widehat{\mu}_{l|y}, \widehat{\sigma}_{l|y}\right)$. For the attack phase, we utilize $\mathcal{L}_{att}$ of size N_{att} traces. The secret key k^* which maximizing the univariate Maximum Likelihood (denoted by LH) is chosen: $k^* = \underset{k}{\operatorname{argmax}}\ \mathrm{LH}(k)$, *i.e.* $k^* = \underset{k}{\operatorname{argmax}} \prod_{j=1}^{N_{att}} \left(\mathrm{f}\left(l_i \mid y_i\right)\right)$. As standard, owing to practical computational reasons and numerical errors, the log-likelihood (LLH) was used [16] $k^* = \underset{k}{\operatorname{argmax}}\ \mathrm{LLH}(k) = \underset{k}{\operatorname{argmax}} \sum_{j=1}^{N_{\mathrm{att}}} \log\left(\mathrm{f}\left(l_i \mid y_i\right)\right)$.

6.4 Timing Attacks

We aim to show the relevance of our technique to timing attacks that rely on the vulnerability introduced by the very same DVFS mechanism we rely on. As an example, [41] relies on the time variance induced by DVFS, which can be manipulated by an adversary to gleam secret information via the time channel. Our technique should make it harder for an adversary to see the different distributions and to separate information about secret computation from time measurement.

7 MEASUREMENTS RESULTS AND ANALYSIS

7.1 Sterile Analysis - Ideal View

First, we aimed to establish that our devised method works in a sterile clean scenario, i.e., by gradually increasing the % of traces taken with altered core frequency (i.e., randomized). This is analogous to uneven weights in a distribution function. For example, 10% altered frequencies means we operate 90% of the times with say f_{base} = 20MHz and 10% of the time with some other frequency in the set say $f_n \in F$:

$$P(f_{base} = 20MHz) = 0.9, P(f_n \in F \setminus f_{base}) = 0.1$$

Fig. 27 and Fig. 28 of Appendix 10 show a proximate linear decrease in estimator value as the distribution nears uniform weights. Secondly, we observe differences with different BW's upon which we will rigorously discuss below. At first glance these results are easy to dismiss since the reduction is negligible, as discussed below, several degrees of freedom are available for design to enhance these results.

Setting a base line for the viability of remote timing attacks, we performed a scenario similar to [41], where plaintext HW affects the p-state, Fig. 22(a) shows easily distinguishable distributions in the time domain, we aim to introduce randomness to this process using the same scheme, proving it's double effectiveness. I.e., in Fig. 22(b), and Fig. 22(c) we gradually increase $||F||$ showing how such timing distributions cannot be distinguished anymore.

7.2 Our Optimization parameters for RAD-FS

Under system restrictions (i.e., the discrete DVFS values described in Fig.7) we grouped frequencies within F' to isolate the parameters of BW and $||F||$ as classified in Table 1. In addition, we grouped the frequencies from F' to isolate the effects of f_{min} per given BW as partitioned in Table 3 of Appendix 10. Figures 13 and 14 show the resulting maximum SNR and correlation values over time, respectively, for the Risc-V SW implementation of Tiny-AES.

Table 1. Classification of frequency map of F to different BW with varying $||F||$s

$\|\|F\|\|$\BW	9.33[MHz]	8.3[MHz]	6[MHz]	4.57[MHz]	1.9[MHz]
$\|\|F\|\| = 2$	[16 6.67]	[13.3 5]	[10 4]	[6.67 2.1]	[4 2.1]
$\|\|F\|\| = 3$	[16 10 6.67]	[13.3 8 5]	[10 8 4]	[6.67 5 2.1]	[4 3.2 2.1]
$\|\|F\|\| = 4$	[16 13.3 10 6.67]	[13.3 10 8 5]	[10 8 6.67 4]	[6.67 5 3.2 2.1]	[4 3.2 2.5 2.1]
$\|\|F\|\| = 5$	[16 13.3 10 8 6.67]	[13.3 10 8 6.67 5]	[10 8 6.67 5 4]	[6.67 5 4 3.2 2.1]	[4 3.2 2.857 2.5 2.1]

$||F||$ **size:** Randomizing chosen frequencies from F. described in Table 1 in a uniform distribution. A reduction by 2 orders of magnitude is observable just from increasing $||F||$ to 5 frequencies (increasing $||F||$ even further is clearly possible (as outlined earlier and depends on the system under evaluation).

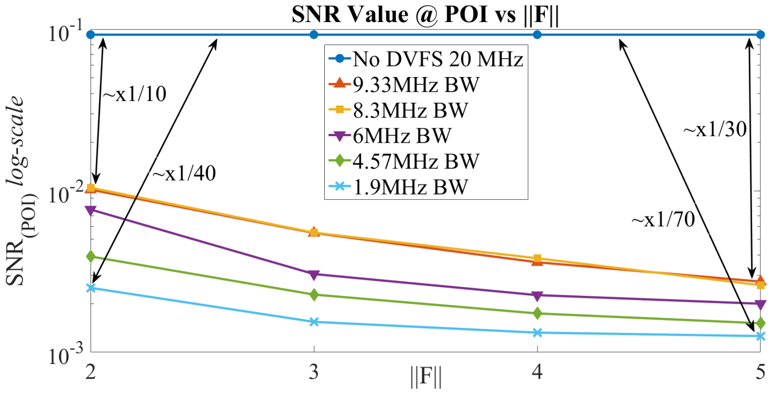

Fig. 13. SNR@POI values vs. size of group F, for different BW's.

A similar phenomenon is observable in the correlation graph, albeit the scale of reduction is smaller. Note that the reduction in both SNR and correlation is inversely proportional to the data-/time-/computation-complexity of an attack. *Therefore, two orders of magnitude are quite significant, leading to a noteworthy security level, attack complexity, etc.*

BW: By reordering the data to look at BW influence on the SNR, we show the decrease of $SNR_{t=POI}$. As demonstrated in Fig. 15, decreasing the BW increases the overlaps (i.e., aliasing) of leakages between sets of traces from different f_n's in the time domain. A smaller BW implies interference in the frequency domain, thus

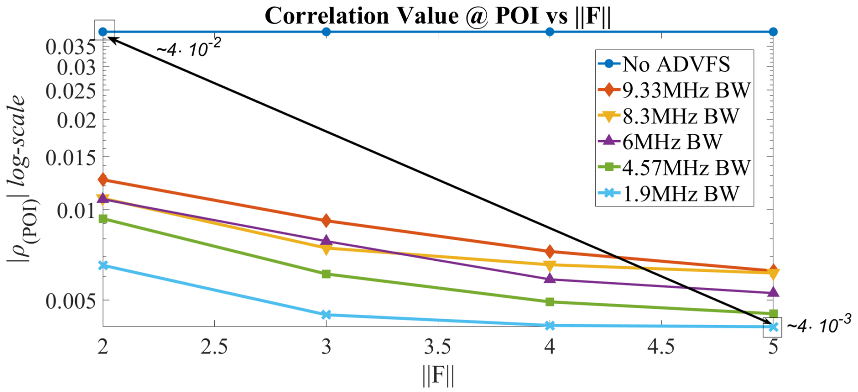

Fig. 14. Correlation@POI values vs. size of group F, for different BW's.

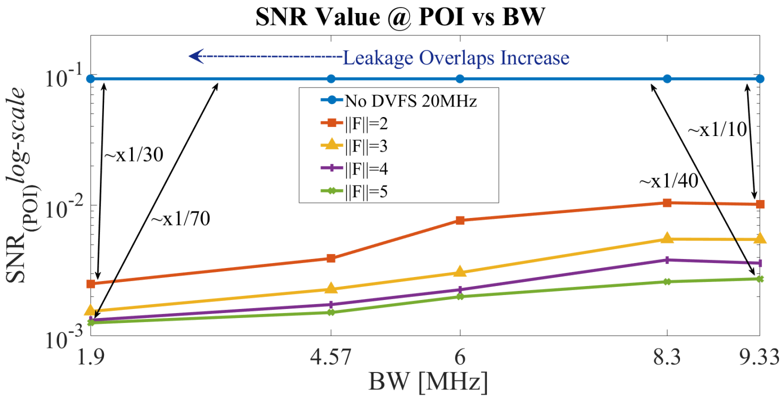

Fig. 15. SNR@POI values vs Bandwidth, a decrease by a factor of one to two magnitudes is achieved carefully selecting the BW to generate aliasing in the time-domain.

the expected value of uni-variate analysis mixes up different time samples or internal computations. The more the leakages overlap the larger the effect is on the estimator (e.g, SNR).

f_{min}**:** To isolate the effect of f_{min}, we grouped 2 frequencies, keeping the BW constant to the best of our abilities under system limitations (listed in Table 3. Comparing Fig.16 to Fig. 15,13 we can see that although f_{min} has an effect, it is considerably weaker than the effect of $\|F\|$ and BW.

Generally, the experiment set highlights that both the Corr. and the SNR estimators performed quite similarly. However, results (security gains) were slightly poorer with CPA since with correlation the signal is not scaled to the noise compared to the SNR.

7.3 Concrete Security Evaluation - Detection Test

In Fig. 17 We performed T-Test on $0.5{\cdot}10^6$ to $10{\cdot}10^6$ traces to evaluate the security of the proposed mechanisms for different $\|F\|$. Fig. 19 and Fig. 18 show the maximum absolute values over time of the T-Test detection vs. the number of collected traces, for different $\|F\|$ for the NXP HW accelerator and the Risc-V SW implementation, respectively. It is evident from Fig. 18 that using RAD-FS shifts the data to higher statistical moments, observing $\|F\| = 1$ we see no leakage in the 2nd moment, wherein $\|F\| = 2$ and above the T value becomes significant. This

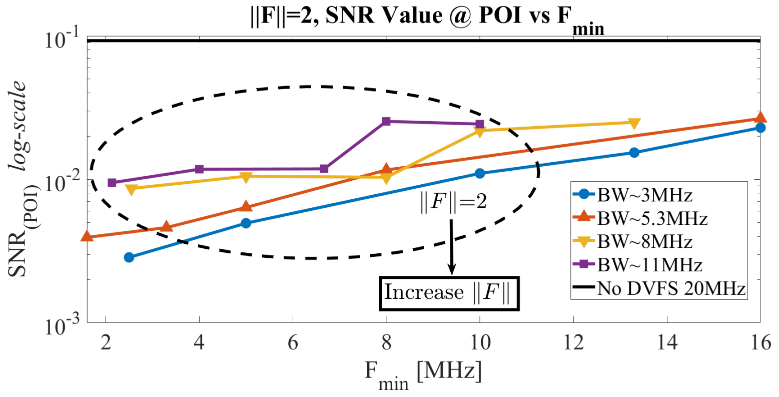

Fig. 16. SNR@POI values vs f_{min} for different BW.

requires higher computational efforts from a potential adversary for a successful attack. It is important to note that information evidenced by a detection test does not practically imply an attack is known or easy, we show below that with the best uni-variate template attack success-rate, the protection level provided by RAD-FS is quite remarkable. On any account, even with the T-Test, adversary-complexity increases by orders-of-magnitudes. Fig 21 shows the number of traces required to disclose a secret N_{td} with $\|F\| = 2$ is 10x times more, going to as high as 80x more with $\|F\| = 5$. in modern DVFS systems with high frequency resolution [43] this is easily improved upon, into exponential gains.

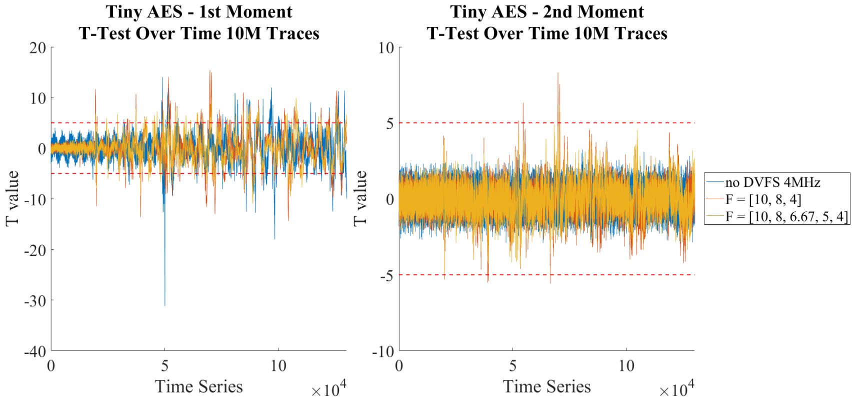

Fig. 17. **Risc-V SW:** example T-Test detection test based on TVLA methodology over time for 10M traces and for different $\|F\|, BW = 6MHz$.

7.4 Gaussian-Template Based Attack

In this subsection our goal was to show how hard it is to make use of the information leakage measured by TVLA. As discussed in Subsection 6.1, our RAD-FS approach reduces attack-based metrics by orders of magnitudes. Thus, our goal was to perform model-less (profiled) evaluation utilizing templates. First, to reduce computational effort we have found POI's using SNR. Then, we profiled leakages in a subset of time samples using a Gaussian-Template

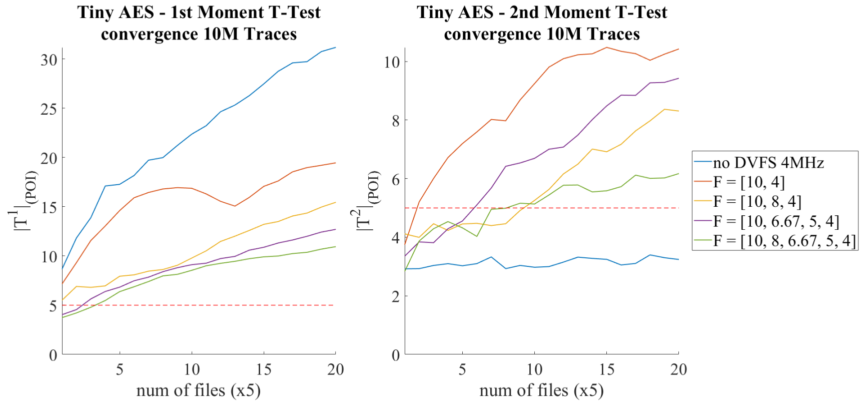

Fig. 18. **Risc-V SW Tiny-AES:** Maximum absolute values over time of a T-Test detection test based on TVLA methodology vs. the number of collected traces and for different $||F||$ for a given BW(6MHz), $V_{min} = 0.66[V]$.

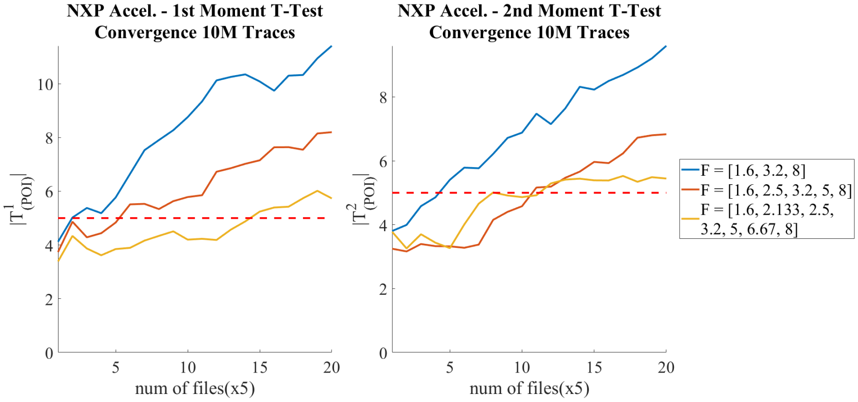

Fig. 19. **NXP AES ACCEL.:** Maximum absolute values over time of a T-Test detection test based on TVLA methodology vs. the number of collected traces and for different $||F||$ for a given BW (5.4MHz), $V_{min} = 0.58[V]$.

model. As shown in Fig. 20 for a 6-1.9MHz BW the attack's success-rate (SR) drops rapidly both with increased $||F||$ and reduced BW. As shown in Fig. 21, higher $||F||$ requires exponentially more traces for a successful extraction of key values, while the BW inversely affects the exponential growth constant. Whereas $5 \cdot 10^5$ traces are requires to achieve a meaningful attack (SR> 0.5) with $||F|| = 1$ with $||F||$ as little as 7 and BW of 4.57M more than $7 \cdot 10^6$ traces are needed, and with $||F||$ as little as 5 (or 7) and BW of 1.9M more than $8 \cdot 10^6$ (or $\sim 40 \cdot 10^6$) traces are needed. Clearly, a minor change of $||F||$ to (say) 8 and considering the exponential increase, can make attack data complexity ultra-high.

7.5 Timing attacks & P-States

In this Sub-section we show RAD-FS is very relevant for such network timing attacks mitigation. Results are demonstrated via. an ideal (optimal) oracle modeling the RAD-FS parameters. Our constructed oracle is very *generous* with how much control is given to the adversary: Prior-art discusses that bits-states affects DVFS

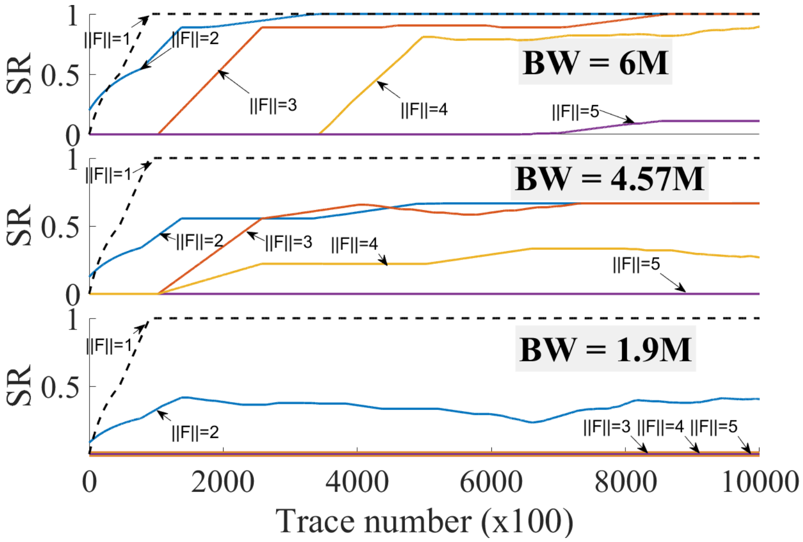

Fig. 20. **Risc-V SW Tiny-AES:** the success rate of a Gaussian template attack with varying $||F||$. Black dashed line - no DVFS 20MHz, Blue - $||F||$ = 2, Orange - $||F||$ = 3, Yellow - $||F||$ = 4, Purple - $||F||$ = 5.

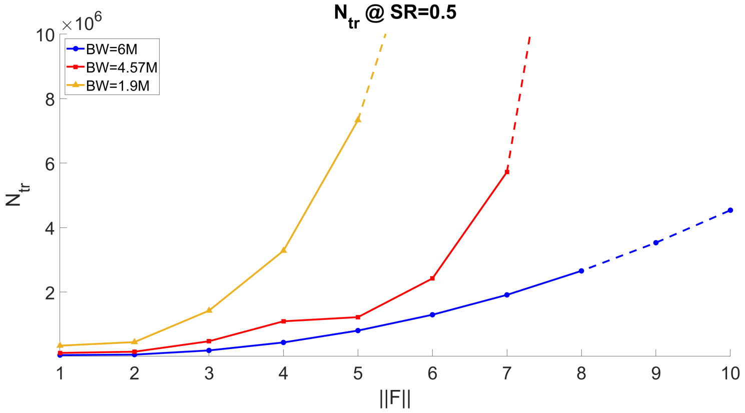

Fig. 21. **Risc-V SW Tiny-AES:** Number of traces required to achieve $SR = 0.5$ vs $||F||$ across different BW, dashed line denotes curve extrapolation

algorithms selections. I.e., different combinations of 'on' and 'off' bits (i.e., hamming weight), in essence generate different workloads and allows adversaries to manipulate the DVFS power-control (and therefore also the timing) of a function call or code. All is needed is an eavesdrop adversary on the protocol (remote) in a known plaintext scenario. This is a manifestation of the mechanism relied upon in [41]-attack, the adversary manipulates the Hamming-Weight (HW) of the plaintext, & the resulting time-to-encryption is measured. As our defense mechanism, we emulate a firmware based solution, that soft over-rides the frequency input from the OS and randomises it with our proposed RAD-FS in mind: I.e., the OS requests that the DVFS switches to some f_{OS} that matches a specific HW plaintext, serving as the *generous* (i.e., most sensitive) SCA Oracle while the firmware chooses a random frequency f_{RAD-FS} that has some relation to f_{OS}. This technique makes the vulnerability harder to exploit by introducing uniformly distributed noise (ideally), making the measured computation time (side-channel) harder to analyse. In the first experiment, the adversary measures the time to perform a sequence of steps: UART communication, change_clock_freq and 1000 encryptions. Fig. 22 shows that when RAD-FS is introduced at the firmware level, increasing $\|F\|$ size causes the distributions that are easily separable with $\|F\| = 1$, to alias unto one, and make it increasingly difficult for the adversary to discern them from one another. In Fig. 22(a), (b) and (c) we gradually increase $\|F\|$ showing that even in this hard scenario of 1000 encryptions distributions start to completely overlap. Expanding upon this, in the next scenario we measured the clock cycles required to perform a set number of encryptions (100), while changing the clock and performing UART communications. We compared changing the clock in 3 different intervals; 100, 50 and 10, meaning as we reduce the interval the adversary is weaker and the scenario is more realistic. 10 encryptions is the same power-state is clearly a reasonable scenario. Several interesting phenomena are visible in Fig. 23,Fig. 24,Fig. 25: **(1)** in Fig. 23 even for an interval of 10 (many clock frequency changes), the distributions are easily distinguishable with $\|F\| = 1$; Demonstrating that without RAD-FS, the DVFS mechanism introduces a strong timing bias. **(2)** introducing RAD-FS shows a significant improvement for any interval. **(3)** going from Fig. 23 to Fig. 24 we can clearly see an increasing aliasing of the distributions & confirming our intended use case. **(4)** in Fig. 25 we achieve almost uniform distributions, with full aliasing, hampering the adversaries attempt to gain very significant amount of information via the side channel. Clearly, in this respect a uniform distribution (maximum entropy) is the best one can hope for.

7.6 Impact on Performance

The RAD-FS ideal scenario performance-security wise is achieved with the highest possible mean frequency (f_{base}) for the encryption engine (as the computation is intensive and mapped to high workload) concurrently with a large $\|F\|$ within a tight BW. Considering IoT applications, the main parameters for optimisation are area and power. For some notations, we denote by **(1)** OH - the performance overhead, i.e., the increase in computation time. **(2)** m - # encryptions done in a given f_n. **(3)** T_{switch} - the time to switch between $f_n, f_m \in F|n \neq m$ for $\|F\| = k$ a total of $\binom{k}{2}$ options. **(4)** f_i - a frequency in F. That is, the total encryption time can be written by $\frac{1}{f_i} \cdot \#_{cycles}$, considering #cycles clock-cycles.

$$\mathrm{T_{av}} = \frac{E[T_{switch_i}] + m \cdot E[\frac{1}{f_i} \cdot \#_{cycles}]}{m} \quad (5)$$

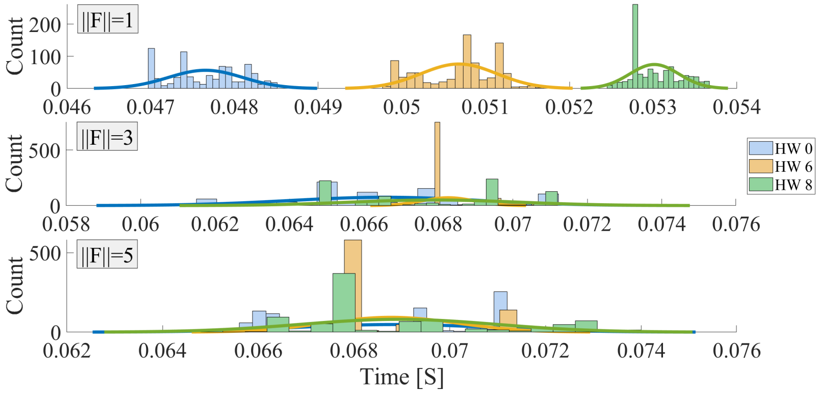

Fig. 22. Timing measured over 1K encryptions, performed with the same HW/freq./voltage. For a worst-case analysis, all plaintext bytes are the same for minimal noise. Measured with 10-100k traces per HW for increased $\|F\|$.

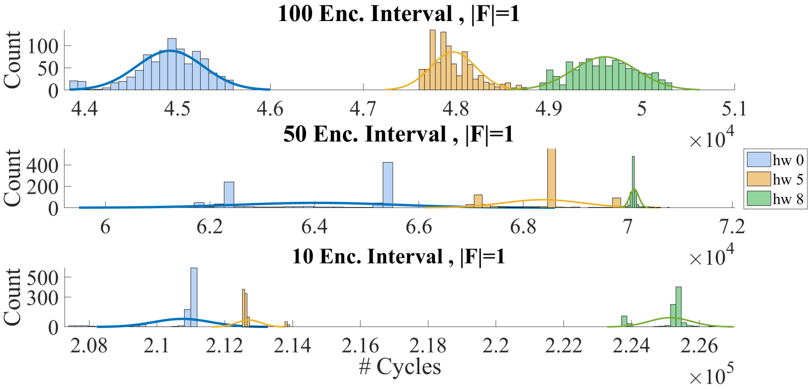

Fig. 23. Clock cycles measured over 100 encryptions, frequency changed with varying intervals. $||F|| = 1$.

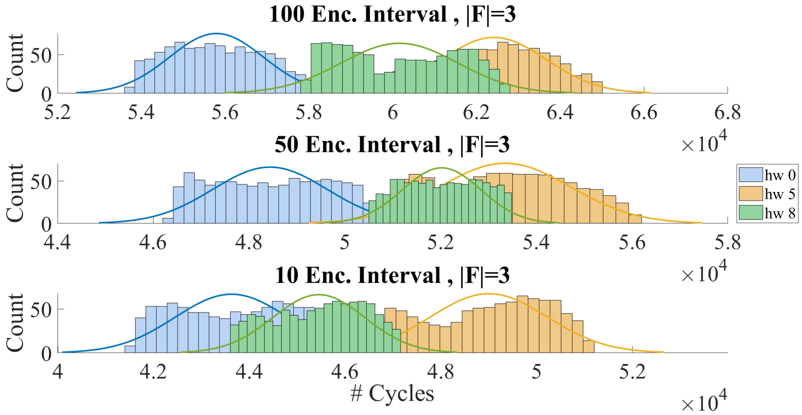

Fig. 24. Clock cycles measured over 100 encryptions, frequency changed with varying intervals. $||F|| = 3$.

Generally, considering the specifications of commercial devices (and the PLS15 characteristics itself), the switching time and the enc.-time are of the same scale, $E_i[T_{switch}] \approx E_i[\frac{1}{f_i} \cdot E\#_{cycles}]$, denoted by E_T. Clearly, taking $\lim_{m\to\infty}$, $\mathrm{T}_{\mathrm{av}} = E_T$ with 1 over m convergence. For example, taking the m=10 case:

$$\mathrm{L}^{\mathrm{OH}}\% = \frac{T_{av}}{T_{f_{base}}} = \frac{1.1 \cdot E_T}{T_{f_{base}}} \tag{6}$$

Under our chip’s limitation, for example, we can cal culate the overhead using 2 frequencies inside a 1.33MHz bandwidth with the following parameters: $F = \{21.33MHz, 20MHz\}, ||F|| = 2, BW = 1.33MHz$ we achieve:

$$\mathrm{L}^{\mathrm{OH}}\% = \frac{1.1 \cdot \frac{1}{k} \cdot \sum_{i=0}^{k-1} \frac{1}{f_i}}{T_{f_{base}}} = 1.136 = 13.6\% \tag{7}$$

For this configuration our RAD-FS mechanism endures a 13.6% latency overhead, which is rather efficient compared to prior art as discussed below. Considering the energy overhead, due to the quadratic dependency of the

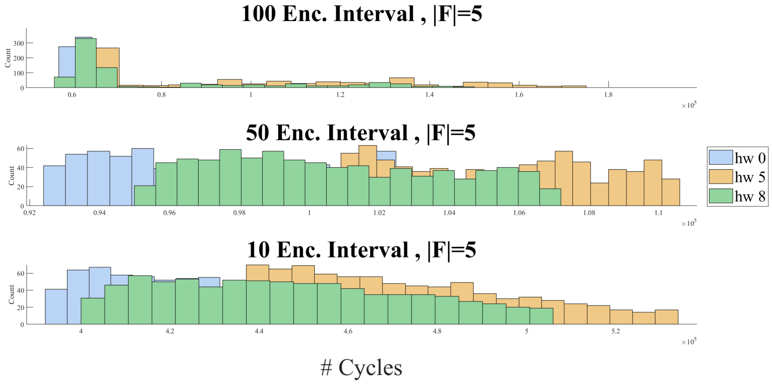

Fig. 25. Clock cycles measured over 100 encryptions, frequency changed with varying intervals. $\|F\| = 5$.

operation frequency, F, on the operating voltage, V_{dd}. We can approximate the voltage OH by $\Delta\mathrm{V} \sim \sqrt{\frac{1}{\Delta\mathrm{T}}} = 0.938$, concluding in:

$$\mathrm{P}^{\mathrm{OH}}\% = \frac{\Delta\mathrm{P}}{\mathrm{P}} = \frac{\Delta\mathrm{f}\cdot\mathrm{C}_{\mathrm{eff}}\cdot\Delta\mathrm{Vdd}^2}{\mathrm{f}\cdot\mathrm{C}_{\mathrm{eff}}\cdot\mathrm{Vdd}^2} = \frac{1.136\cdot 0.938^2}{1} = 99\% \tag{8}$$

Under our chip's limited options, and taking the above estimations, we estimate the P^{OH} to be improved by 1%. Actual commercial devices numbers may even be better with more granular freq./voltage steps.

Table 2. *SW implementation, **HW analog implementation, ***HW analog & custom memory implementation

Source	$Area^{OH}$	L^{OH}	P^{OH}	$Effort^{OH}$
[36]	+6.6%	+17.4%	-3.5%	**
[17]	+20%	0	+24%	**
[21]	+10%	+0.07%	+8%	***
Our Work	0	+13.6%	-1%	*

For final comparison, we considered implementation effort, which roughly estimates resources (dedicated design, engineering, verification etc.) needed to implement different solutions and the flexibility offered by different approaches; SW implementation which requires some SW code, HW Analog implementation which requires additional silicone area and dedicated HW design, HW analog design with dedicated memory design that requires custom memory cells in addition to analog design.

As shown in Table 2, with L and P denoting latency and performance, respectively, our solution aims at ULVT MCU's and IoT chips, where area and power are the main optimization concerns. Due to using a block that pre-exists within such devices, we present no area overhead, with low implementation effort, requiring only a SW implementation. Using a SW solution, we sacrifice latency within an acceptable margin even when comparing to SOTA. This can be further improved upon by implementing RAD-FS in assembly and not in high level C code.

7.7 Results Summary

To summarize our results: **(1)** we start by generally demonstrating that with a simple distinguisher our concept has merit as shown by Fig. 27 and Fig. 28. The more uniform the frequency distribution, the less information is leaked **(2)** We continue to deep dive into the different parameters such a technique would utilize, using various estimators and SCA methodologies. From Fig. 14 we conclude that CPA is weak in our scenario and focus on TVLA & SNR. **(3)** We isolated the different variables, $||F||$, BW & f_{min}, into different sets of measurements to determine which is the most prudent. Comparing results from Fig. 15 and Fig. 13 to Fig. 16, from the difference in the slope and distribution of the measurements, we can conclude that the best scenario requires as many frequencies as possible, in as small a BW as possible to generate *overlaps* in the frequency domain, which in turn are harder to filter out in the time domain. **(4)** Using TVLA (Figures 18, 19), our objective was to show two things. First, security increases; that is to say, the two compared populations (fixed and randomized) are harder and harder to distinguish to a meaningful extent, as evident by the higher number of traces needed to converge and the final convergence value. Secondly the shift of information to higher statistical moments, seen in Fig. 18 the only measurement showing no information in the second moment is the NO RAD-FS measurement set, thus proving the computational complexity for an adversary introduced by RAD-FS. **(5)** Gaussian-Template based attacks - we chose a strong(er) attack model to show that even in a scenario where the adversary is knowledgeable can profile a device and take large amount of such profiling traces, a successful attack is still not trivial (regardless of the univariate TVLA results). Fig. 21 shows that $N_{tr} \sim \exp\{BW \cdot ||F||\}$. **(6) Identifying the strength of RAD-FS in protecting against remote timing-based attacks**, we show clear aliasing in the time domain in a scenario where the adversary has a way to manipulate the DVFS mechanism either directly (via SW) or via an oracle (as done in [41]). By randomizing the frequency, we reduce the direct relation between workload and frequency, making it harder for to discern information about the encryption from the run time. This is *important as several industry standard encryption schemes, both symmetric and asymmetric public key encryptions are vulnerable to such attacks.*

8 ASSUMPTIONS AND REAL-LIFE

We will now outline the features of our protection mechanism by identifying the essential requirements for a successful attack:

(1) Perfect synchronization and measurement triggering - this is how the analysis is done in this paper/analysis. In real-life things are much harder while will induce some overhead from the adversary.

(2) Pre-knowledge on *Trace-Length* - when we randomize frequencies using RAD-FS, the adversary clearly does not know the number of samples needed to be captured, as it depends on the randomized frequency. Therefore, the best scenario is to take some fixed number of samples which will imply mixtures of frequencies appearing in the captured leakage even if f_n is only randomized once per several encryptions.

(3) Isolating the relevant leakages is a lot harder in a parallel computation scenario. running on multiple cores will drastically increase the algorithmic noise and will impede the adversary's ability to filter out leakages not correlating with the hypothesis data manipulation ($\oplus, sbox$ etc.).

In addition, it is important to emphasize that real-life applications with more conventional DVFS mechanisms are different: **(a)** DVFS resolution is very high - even hundreds of power/frequency- states [1, 27, 38, 43]. That is, countermeasure security parameters such as $||F||,\ BW,\ f_{min}$ can be significantly optimized.

(b) Synchronization in large embedded-SoCs featuring miniaturized is fairly complex, and any pre-processing and trigger estimation will dramatically increase the noise.

(c) Modern ADVFS solutions optimize each core individually depending on the workload, adding a plethora of protection flexibility. **(d)** As discussed in page 8, filtering attempts may eliminate information and induce overlaps thus increasing the noise.

9 CONCLUSIONS AND FUTURE RESEARCH

In this paper we demonstrate RAD-FS, a new security technique that is scalable, software-based, and easy to implement, that applies to most if not all modern microchips. Improving protection against DPA and timing SCA attacks in orders of magnitudes. Discussing several different estimators under DPA, we show the radical effect achieved by RAD-FS on the analysed estimator and its' convergence. We show the different effects of our parameters, and conclude that the ideal scenario revolves around as large $||F||$ as possible within the smallest BW achievable considering system limitations, as to increase aliasing. Moreover, it substitutes the naive solution against timing attacks (shutting off the ADVFS controller) (e.g., Hertzbleed) and enables an SCA-secure-chip coexisting with DVFS optimization. We discussed a countermeasure for the inherent weakness in power/freq.-data dependency.

In future work, we would like to test adding "fuzziness" to the RAD-FS process though uniformly distributed amount of "NOP" asm commands before/within the enc.-operation, to touch upon multi-DVFS-model security analysis, and combine this ultra low-cost solution with additional layers of security, and proceed with in depth electromagnetic SCA analysis with the proposed mechanism.

ACKNOWLEDGEMENTS

We sincerely thank the *PLSense* team (today acquired by NXP), namely, Uzi Zangi, Arnon Kaminsky, Shahar Dinar, Shaked Matzner and Tzach Hadas: for their invaluable support and guidance while working with their development environment and beta (development and testing) boards and setups, such assistance and dedication towards innovative science is not trivial for such a small start-up.

DECLARATIONS

Itamar Levi and Daniel Dobkin were partially funded by the Israel Innovation Authority (IIA), Bio-Chip Consortium grant file No. 75696, Israel, and Israel Science Foundation (ISF) grant 2569/21.

10 APPENDIXES

Table 3. Classification of frequency map of F' to different BWs with a varying F_{min}

~const BW\\F_{min}	F_{min}	~2F_{min}	~3F_{min}	~4F_{min}	~5F_{min}	~6F_{min}	~13F_{min}
BW~3MHz	[5, 2.5]	[8, 5]	[13.3, 10]	[16, 13.3]		[20, 16]	
BW~5.3MHz	[6.67, 1.28]	[8, 2.857]	[8,3.3]	[10, 5]			[21.3, 16]
BW~8MHz	[10, 2.5]	[13.3, 5]	[16, 8]	[20, 10]	[21.33, 13.3]		
BW~11MHz	[13.3, 2.13]	[16, 4]	[16, 6.67]	[20, 8]	[21.33, 10]		

+

REFERENCES

[1] Acun, B., Chandrasekar, K., and Kale, L. V. Fine-grained energy efficiency using per-core dvfs with an adaptive runtime system. In *2019 Tenth International Green and Sustainable Computing Conference (IGSC)* (2019), pp. 1–8.

[2] Agrawal, D., Archambeault, B., Rao, J. R., and Rohatgi, P. The em side—channel (s). In *International workshop on cryptographic hardware and embedded systems* (2002), Springer, pp. 29–45.

[3] Bao, W., Hong, C., Chunduri, S., Krishnamoorthy, S., Pouchet, L.-N., Rastello, F., and Sadayappan, P. Static and dynamic frequency scaling on multicore cpus. *ACM Transactions on Architecture and Code Optimization (TACO) 13*, 4 (2016), 1–26.

[4] Bernstein, D. J. Cache-timing attacks on aes.

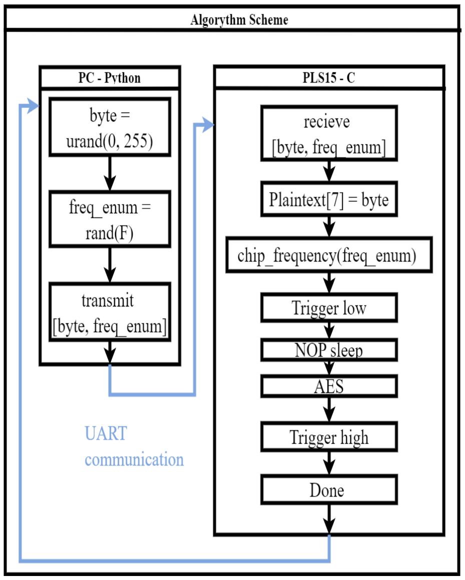

Fig. 26. Experiment pseudo-code Flowchart

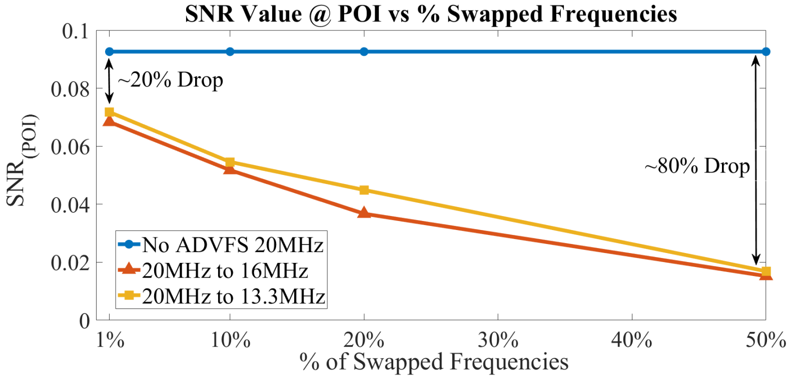

Fig. 27. Swapping from default 20MHz to a different core frequency for a percentage of the measured queries led to a reduction in the SNR POI value

[5] Bos, J. W., Halderman, J. A., Heninger, N., Moore, J., Naehrig, M., and Wustrow, E. Elliptic curve cryptography in practice. In *Financial Cryptography and Data Security: 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers 18* (2014), Springer, pp. 157–175.

[6] Breuer, R., Standaert, F.-X., and Levi, I. Fully-digital randomization based side-channel security—toward ultra-low cost-per-security. *IEEE Access 10* (2022), 68440–68449.

[7] Brier, E., Clavier, C., and Olivier, F. Correlation power analysis with a leakage model. In *International workshop on cryptographic hardware and embedded systems* (2004), Springer, pp. 16–29.

[8] Brier, E., Clavier, C., and Olivier, F. Correlation power analysis with a leakage model. In *Cryptographic Hardware and Embedded Systems-CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings 6* (2004), Springer, pp. 16–29.

[9] Brumley, B. B., and Tuveri, N. Remote timing attacks are still practical. In *European Symposium on Research in Computer Security* (2011), Springer, pp. 355–371.

[10] Brumley, D., and Boneh, D. Remote timing attacks are practical. *Computer Networks 48*, 5 (2005), 701–716.

[11] Carts, D. A. A review of the diffie-hellman algorithm and its use in secure internet protocols. *SANS institute 751* (2001), 1–7.

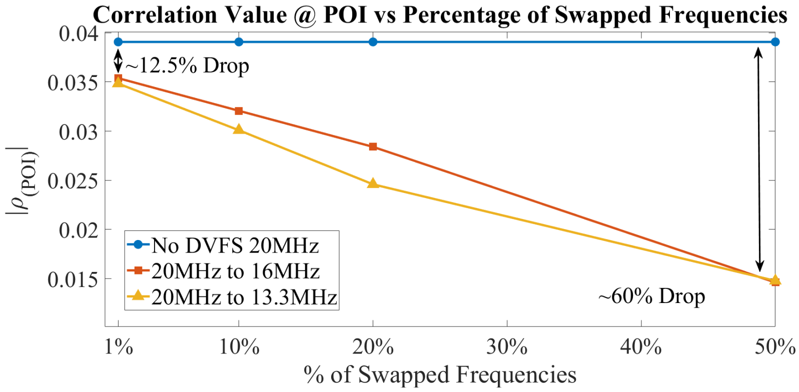

Fig. 28. Swapping from default 20MHz to a different core frequency for a percentage of the measured queries led to a reduction in the Corr POI value

[12] Cassiers, G., Grégoire, B., Levi, I., and Standaert, F.-X. Hardware private circuits: From trivial composition to full verification. *IEEE Transactions on Computers* (2020).

[13] Chari, S., Rao, J. R., and Rohatgi, P. Template attacks. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2002), Springer, pp. 13–28.

[14] Chari, S., Rao, J. R., and Rohatgi, P. Template attacks. In *Cryptographic Hardware and Embedded Systems-CHES 2002: 4th International Workshop Redwood Shores, CA, USA, August 13–15, 2002 Revised Papers 4* (2003), Springer, pp. 13–28.

[15] Dworkin, M. Recommendation for block cipher modes of operation. methods and techniques. Tech. rep., National Inst of Standards and Technology Gaithersburg MD Computer security Div, 2001.

[16] Fei, Y., Ding, A. A., Lao, J., and Zhang, L. A statistics-based fundamental model for side-channel attack analysis. *Cryptology ePrint Archive* (2014).

[17] Ghosh, A., Das, D., and Sen, S. Physical time-varying transfer functions as generic low-overhead power-sca countermeasure. *arXiv preprint arXiv:2003.07440* (2020).

[18] Icons8. Scientific icons, 2023. https://icons8.com/ [Accessed: (1.06.2023].

[19] Inc, A. Arm big. little, 2022. https://www.arm.com/technologies/big-little [Accessed: (10.07.2023].

[20] Kocher, P., Jaffe, J., Jun, B., and Rohatgi, P. Introduction to differential power analysis. *Journal of Cryptographic Engineering 1*, 1 (2011), 5–27.

[21] Kumar, R., Liu, X., Suresh, V., Krishnamurthy, H. K., Satpathy, S., Anders, M. A., Kaul, H., Ravichandran, K., De, V., and Mathew, S. K. A time-/frequency-domain side-channel attack resistant aes-128 and rsa-4k crypto-processor in 14-nm cmos. *IEEE Journal of Solid-State Circuits 56*, 4 (2021), 1141–1151.

[22] Le Sueur, E., and Heiser, G. Dynamic voltage and frequency scaling: The laws of diminishing returns. In *Proceedings of the 2010 international conference on Power aware computing and systems* (2010), pp. 1–8.

[23] Levi, I., Bellizia, D., Bol, D., and Standaert, F.-X. Ask less, get more: Side-channel signal hiding, revisited. *IEEE Transactions on Circuits and Systems I: Regular Papers 67*, 12 (2020), 4904–4917.

[24] Levi, I., Bellizia, D., and Standaert, F.-X. Beyond algorithmic noise or how to shuffle parallel implementations? *International Journal of Circuit Theory and Applications 48*, 5 (2020), 674–695.

[25] Mangard, S. Hardware countermeasures against dpa–a statistical analysis of their effectiveness. In *Topics in Cryptology–CT-RSA 2004: The Cryptographers' Track at the RSA Conference 2004, San Francisco, CA, USA, February 23-27, 2004, Proceedings* (2004), Springer, pp. 222–235.

[26] Mangard, S., Oswald, E., and Popp, T. *Power analysis attacks: Revealing the secrets of smart cards*, vol. 31. Springer Science & Business Media, 2008.

[27] Mei, X., Wang, Q., and Chu, X. A survey and measurement study of gpu dvfs on energy conservation. *Digital Communications and Networks 3*, 2 (2017), 89–100.

[28] Menezes, A. J., Van Oorschot, P. C., and Vanstone, S. A. *Handbook of applied cryptography*. CRC press, 2018.

[29] Messerges, T. S., Dabbish, E. A., and Sloan, R. H. Investigations of power analysis attacks on smartcards. *Smartcard 99* (1999), 151–161.

[30] Nassi, B., Iluz, E., Cohen, O., Vayner, O., Nassi, D., Zadov, B., and Elovici, Y. Video-based cryptanalysis: Extracting cryptographic keys from video footage of a device's power led. *Cryptology ePrint Archive* (2023).

[31] Padoin, E. L., Pilla, L. L., Castro, M., Boito, F. Z., Alexandre Navaux, P. O., and Méhaut, J.-F. Performance/energy trade-off in scientific computing: the case of arm big. little and intel sandy bridge. *IET Computers & Digital Techniques 9*, 1 (2015), 27–35.

[32] Pering, T., Burd, T., and Brodersen, R. The simulation and evaluation of dynamic voltage scaling algorithms. In *Proceedings of the 1998 international symposium on Low power electronics and design* (1998), pp. 76–81.

[33] Quisquater, J.-J., and Samyde, D. Electromagnetic analysis (ema): Measures and counter-measures for smart cards. In *International Conference on Research in Smart Cards* (2001), Springer, pp. 200–210.

[34] Salomon, D., and Levi, I. Masksimd-lib: on the performance gap of a generic c optimized assembly and wide vector extensions for masked software with an ascon-p test case. *Journal of Cryptographic Engineering* (2023), 1–18.

[35] Schneider, T., and Moradi, A. Leakage assessment methodology: A clear roadmap for side-channel evaluations. In *Cryptographic Hardware and Embedded Systems–CHES 2015: 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings 17* (2015), Springer, pp. 495–513.

[36] Singh, A., Kar, M., Mathew, S. K., Rajan, A., De, V., and Mukhopadhyay, S. Improved power/em side-channel attack resistance of 128-bit aes engines with random fast voltage dithering. *IEEE Journal of Solid-State Circuits 54*, 2 (2019), 569–583.

[37] Standaert, F.-X. Introduction to side-channel attacks. In *Secure integrated circuits and systems*. Springer, 2010, pp. 27–42.

[38] Su, B., Gu, J., Shen, L., Huang, W., Greathouse, J. L., and Wang, Z. Ppep: Online performance, power, and energy prediction framework and dvfs space exploration. In *2014 47th Annual IEEE/ACM International Symposium on Microarchitecture* (2014), pp. 445–457.

[39] Turan, M. S., McKay, K., Chang, D., Bassham, L. E., Kang, J., Waller, N. D., Kelsey, J. M., and Hong, D. Status report on the final round of the nist lightweight cryptography standardization process.

[40] Veyrat-Charvillon, N., Medwed, M., Kerckhof, S., and Standaert, F.-X. Shuffling against side-channel attacks: A comprehensive study with cautionary note. In *International Conference on the Theory and Application of Cryptology and Information Security* (2012), Springer, pp. 740–757.

[41] Wang, Y., Paccagnella, R., He, E. T., Shacham, H., Fletcher, C. W., and Kohlbrenner, D. Hertzbleed: Turning power {Side-Channel} attacks into remote timing attacks on x86. In *31st USENIX Security Symposium (USENIX Security 22)* (2022), pp. 679–697.

[42] Wang, Y., Paccagnella, R., Wandke, A., Gang, Z., Garrett-Grossman, G., Fletcher, C. W., Kohlbrenner, D., and Shacham, H. Dvfs frequently leaks secrets: Hertzbleed attacks beyond sike, cryptography, and cpu-only data. In *2023 IEEE Symposium on Security and Privacy (SP)* (2023), pp. 2306–2320.

[43] Xinxin Mei, Ling Sing Yung, K. Z. X. C. Gpu dvfs, 2013. https://dl.acm.org/doi/pdf/10.1145/2525526.2525852 [Accessed: (10.07.2023].

[44] Zhai, B., Blaauw, D., Sylvester, D., and Flautner, K. Theoretical and practical limits of dynamic voltage scaling. In *Proceedings of the 41st Annual Design Automation Conference* (2004), pp. 868–873.