PhaseSCA: Exploiting Phase-Modulated Emanations in Side Channels

Pierre Ayoub, Aurélien Hernandez, Romain Cayre, Aurélien Francillon, Clémentine Maurice

IACR Transactions on Cryptographic Hardware and Embedded Systems
ISSN XXXX-XXXX, Vol. 0, No. 0, pp. 1–28. DOI:XXXXXXXX

PhaseSCA: Exploiting Phase-Modulated Emanations in Side Channels

Pierre Ayoub[1], Aurélien Hernandez[1], Romain Cayre[1], Aurélien Francillon[1] and Clémentine Maurice[2]

[1] EURECOM, Sophia Antipolis, France first.last@eurecom.fr
[2] Univ. Lille, CNRS, Inria first.last@inria.fr

Abstract. In recent years, the limits of electromagnetic side-channel attacks have been significantly expanded. However, while there is a growing literature on increasing attack distance or performance, the discovery of new phenomenons about compromising electromagnetic emanations remains limited.
In this work, we identify a novel form of modulation produced by unintentional electromagnetic emanations: phase-modulated emanations. This observation allows us to extract a side-channel leakage that can be exploited to reveal secret cryptographic material. We introduce a technique allowing us to exploit this side-channel in order to perform a full AES key recovery, using cheap and common hardware equipment like a software-defined radio (SDR). Moreover, we demonstrate that the exploitation of this new phase leakage can be combined with traditional amplitude leakage to significantly increase attack performance. While investigating the underlying phenomenon causing this unintentional modulation, we identified several prior works that have approached similar exploitation – without being aware of each other. Creating a bridge between older and recent work, we unveil the relationship between digital jitter and signal phase shift in the context of side-channel attacks and fill the gap between prior works from various research fields.

Keywords: Side-channel attacks · Power/Electromagnetic analysis · Unintended modulation · Phase modulation · Angle modulation · Clock jitter

1 Introduction

Electromagnetic eavesdropping on computer applications is a well-known threat, studied for more than 70 years in the military world according to the TEMPEST specification from the NSA [Age72]. In the public domain, it has been known for at least four decades, popularized through the work of Van Eck, which eavesdropped video display through their electromagnetic radiation (EMR) as Van Eck Phreaking [VE85]. On the other hand, side-channel attacks are a threat to cryptosystems and have been increasingly studied in the last two decades since the initial research work released in the public domain [Koc96]. Electromagnetic side channels [QS01], a specific kind of side channels using signals recorded through near-field (NF) probes placed in the vicinity of a victim device, have received a lot of attention recently. The biggest advantage of EM side channels is that they can be performed at a distance up to half a meter [Mey12], or even more than 1 meter in specific conditions (*e.g.*, Screaming Channels [CPM+18]).

Unintended Phase Modulation Since their discovery, to our knowledge, electromagnetic side-channel attacks have always exploited an unintentional amplitude modulation from a target device. This unintentional modulation depends on the secret data that is processed

during a cryptographic operation, therefore leaking information that can allow an attacker to infer internal states of the cryptographic algorithm and indirectly recover the secret data. In this paper, we show that another type of unintentional modulation is present in some, and probably many, electromagnetic side-channel signals. We highlight the presence of a leakage modulated by a *Phase modulation (PM)*, a specific form of *angle modulation* caused by timing variations in a signal (so-called *jitter*). Using a radio receiver, we conduct a new side-channel attack and perform a full key recovery targeting the AES algorithm by taking advantage of this phase leakage. Such an attack can be performed easily with cheap radio equipment since measuring a phase can be done leveraging software-defined radio (SDR) costing from dozens to hundreds of dollars. In comparison, measuring the signal jitter directly requires high-end instruments, like oscilloscopes or FPGA, costing usually thousands of dollars.

In this paper, we follow two complementary approaches. First, we explore the feasibility of exploiting the phase information leakage on several SoCs. Second, we conduct a study of the leakage source and reproduce experimentally the phenomenon to analyze it in controlled conditions.

Research questions To the best of our knowledge, we are the first to find a side-channel information leakage on the phase of electromagnetic radiation (EMR). Our paper answers the following research questions:

RQ1 How to detect and exploit a data-dependent leakage in the phase of a signal?

RQ2 Is this phase leakage widespread in modern SoCs?

RQ3 What are the physical root causes resulting in a phase leakage in electromagnetic radiation?

Our first approach corresponds to research questions RQ1/2, while the second corresponds to RQ3.

Contributions In order to answer the previous questions, we made the following contributions:

C1 We propose a methodology to compute the phase shift of a signal and generate a trace that can be processed by a standard side-channel algorithm. In our evaluation, we found statistical correlations with the cryptographic input of an AES encryption and conducted successful side-channel attacks.

C2 Leveraging the state of the art on multi-channel attacks, we recombine attacks on amplitude and phase to increase the attack performance, proving that phase leakage is not redundant with amplitude but contains additional information that can be leveraged to facilitate exploitation.

C3 Using several popular SoCs (nRF51, nRF52, STM32, ATmega328), we identify leakage in the phase of signals, suggesting that the problem is widespread and not specific to a given implementation.

C4 Based on prior work and our own experiments, we show that a probable cause of this leakage is a coupling between the processor and an oscillator circuit, producing a jitter on the clock signal.

C5 By explaining the relationship between a signal jitter and a signal phase shift, we fill the gap between timing and electromagnetic side-channel attacks.

In order to ensure the full reproducibility of our work, our experiments and datasets are published as open-source software.[1]

[1] https://github.com/pierreay/phase_data.git

2 Background

2.1 Signal Jitter and Relation to Phase Shift

Ensuring that signal transitions always occur with precise and constant timing is nearly impossible within electronic circuits. This undesired phenomenon is called *jitter* in the electrical engineering field and applies to both analog and digital domains [HH15, p. 457]. Jitter measurement estimates the timing deviation of a given periodic signal relative to an ideal and expected one [BPRT19, SAHW90]. In digital circuits, clocking signals are typically affected by jitter effects, which are increasingly difficult to mitigate in more complex designs. This presents a significant engineering challenge as it directly affects the functional property of a design, potentially leading to data corruption or faults.

Clock jitter can be categorized based on its originating source [DDS, Han]. First of all, non-deterministic factors contribute significantly to the overall measured jitter. Those factors are intrinsic to electronic components, such as thermal and semi-conductor flickering noise. This is referred to as a random jitter, which follows a normal distribution. However, jitter may not be evenly distributed in some cases and could be statistically correlated to a predictable source. This non-random jitter is referred to as a deterministic jitter. Those predictable jitter sources in electronics are often caused by nearby disturbing components and coupling with close signal lines. For instance, a digital data line within an integrated circuit might impact the clock system in its vicinity. Thus, part of the observable jitter becomes correlated with data.

While clock jitter is naturally represented in the time domain as a period shift, it finds an equivalence in the frequency domain as phase noise [DDS, UMS, HT03]. Considering an ideal clock signal at a perfectly constant period, it will naturally exhibit a power spectrum perfectly contained within clock frequency. Short-term period shifts added to the temporal clock signal will translate as phase noise, with frequency components spreading around the nominal clock frequency as sides-lobes on a spectrum. Overall, the use of time or frequency domain representations mainly depends on the type of measuring instrument involved [Tekb] [JG, p. 376] – *i.e.*, a real-time oscilloscope or a spectrum analyzer. Introduced in Section 2.3, a quadrature sampling equipment with phase demodulation also constitutes a convenient way to observe jitter [Key].

2.2 Signal Representation

A signal can be defined as a real-valued function of time $s(t)$ in Equation 1:

$$s(t) = A\sin(2\pi ft + \varphi) \tag{1}$$

A is the amplitude, representing the magnitude of the variations of the periodic function in a single period. $f \in [0, +\infty]$ Hz is the frequency, representing the temporal rate of change of the instantaneous phase, *i.e.*, the time derivative of phase. $\varphi \in [-\pi, \pi]$ rad is the phase, representing the angle quantity representing the fraction of the cycle covered up. The difference between two phase values may be designated using *phase shift* when represented in the 1D time-domain or *phasor rotation* when being represented in the 2D complex plane. Because of trigonometric identities, the same signal can be defined as the sum of a *sine* and a *cosine* functions with both a phase of 0, as shown in Equation 2 [Lyo08, Put18]:

$$s(t) = I\cos(2\pi ft) + Q\sin(2\pi ft) \tag{2}$$

With $I \in \mathbb{R}$ and $Q \in \mathbb{R}$ representing the amplitudes of the two signals. Considering a signal recorded using in-phase and quadrature (I/Q) sampling, it will be sampled using the analytic representation. Therefore, our discrete signal is a complex-valued function $x(t)$ where each sample x_i is a complex number of the form $x_i = I + jQ$, with I the real

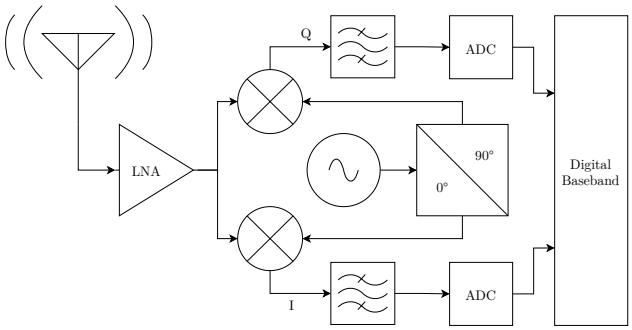


Figure 1: Radio receiver direct-conversion architecture.

part, Q the imaginary part and j the imaginary unit. With a discrete complex-valued signal, the amplitude can be computed as the quadratic sum of the I and Q samples, as shown in Equation 3:

$$A = \sqrt{I^2 + Q^2} \tag{3}$$

2.3 Radio Architecture and Relation with Phase Measure

Figure 1 shows a typical architecture of a direct-conversion radio receiver [Beh07]. The objective of a radio receiver is to sample a signal at baseband frequency (lower frequency) from a carrier frequency (higher frequency). A received signal, *via* an antenna or a probe, is amplified through a low-noise amplifier (LNA). It is then mixed with two in-quadrature signals, explained in Section 2.2, to perform the down-conversion, *i.e.*, lowering the frequency from the carrier to the baseband. In-quadrature corresponds to two signals with a phase shift of $\frac{\pi}{2}$ (or 90°), hence for sinusoidal functions, it corresponds to one *sine* and one *cosine* functions respectively. The resulting I and Q signals are then filtered and sampled by an analog-to-digital converter (ADC), which results in samples that are stored as complex numbers – where the I is the real part and the Q is the imaginary part.

Measuring the amplitude of the received signal depends on the power of the electrical signal that is fed into the ADC – hence, mainly from the antenna and the LNA gain in dB. Measuring the phase of the received signal is relative to the phase of the local oscillator that is generating the two in-quadrature signals. In side channels, in order to build a dataset, we need to receive the "same signal" several times at different points in time – a form of time diversity. To have an absolute measure of the phase of the received signal with time diversity, the problem of phase coherency is equivalent to a multiple-input multiple-output (MIMO) system, where the same signal is measured at different points in space – spatial diversity. In MIMO systems, phase coherency is typically achieved by synchronizing the clocks – local oscillators – of all radio receivers with a reference signal.

2.4 Measurement Equipement

EM Side Channels Using an Oscilloscope An oscilloscope [HH15, p. 1158] is an equipment able to sample a large bandwidth of frequencies with a duration dependent on the sample rate and its buffer size. The sampling bandwidth ranges from 0 to $\frac{\text{sr}}{2}$ with sr being the sampling rate according to the Nyquist–Shannon sampling theorem. Sampling a large bandwidth is achieved by using an ADC with a very high sample rate – *e.g.*, several GHz, implying that the ADC is a costly circuit. In the side channels context, the acquisition of a trace is generally initiated using an accurate digital trigger. The oscilloscope generates a 1D real-valued vector corresponding to the sampled signal (using the real-valued representation defined in Section 2.2). This trace can then be used for a side-channel attack.

EM Side Channels Using a Radio Performing a side-channel attack using a radio equipment, such as a SDR, involves two standard steps:

- Acquire the signal through sampling and measurement with an EM probe, resulting in a complex-valued vector – the I/Q analytic representation as explained in Section 2.2.
- Compute the amplitude of the complex signal, resulting in a 1D real-valued vector, called a "trace" when represented in time-domain – see Equation 3. In telecommunication terms, this is analogous to performing amplitude demodulation, using each amplitude value as a symbol. This trace can then be used for a side-channel attack.

Comparison of Oscilloscope and Radio for EM Side Channels In contrast to a radio architecture exposed in Section 2.3, an oscilloscope does not perform any down-conversion process from an intermediate frequency to the baseband. As a result, to reach the same frequency, the local oscillator and mixer need to be replaced by a costly ADC. Compared to an oscilloscope, suited to acquire a signal on a very large band, a SDR can acquire a signal on a narrow band with a better sensitivity. Considering the accurate digital trigger, the absence of the down-conversion process, and the 1D real-valued sampled vector, the oscilloscope output signal can be directly used for a side-channel attack compared to the SDR output signal, which has to be pre-processed.

2.5 Side-Channel Attacks

Side-channel attacks are a class of attacks introduced by *Kocher et al.* against a security system using information originating from the interaction between the cryptosystem and its environment. Hence, it allows for an attacker to recover the secret key used during a cryptographic operation, *e.g.*, encrypting or signing data using a secret key. Side channels exploit a physical measurement to recover cryptographic material. For the first attacks, the duration [Koc96] or the power consumption [KJJ99] of the attacked cryptographic operation was used. Additional classes of side channels were discovered and exploited since [ZF05], *e.g.*, acoustic waves [GST17] or the optical photonic emanations [FH08].

Attack Overview From a high-level perspective, the side-channel attacks we used are conducted by collecting a high number of measurements – called "traces" – and computing correlations between predictions from a model and the actual measurements. A trace is a real-valued vector representing the measured physical quantity over time. The *model* predicts the leakage measurement of an intermediate value used in the cryptographic operation based on theoretical data input to the cryptosystem. The model can be chosen either theoretically, as in *non-profiled attacks* by using the Hamming Weight (HW) of the intermediate value for example, or estimated as in *profiled attacks*, by collecting a high number of training measurements to build a template [CRR02]. The Pearson Correlation Coefficient (PCC) (ρ or r) is often used as a distinguisher, the function used to compute the correlation. The PCC can also be used to find point of interests (POIs) [Mey12], the time samples where the leakage is located. Searching for the candidate key with the best correlation, the side-channel algorithm may lead to a full key recovery if the attack is successful.

AES and Side Channels AES [oSN01] is a block cipher, *i.e.*, processing blocks of data to encrypt plaintext into a ciphertext – or the opposite for decryption. AES defines several modes. In ECB-128 mode, each block of 128-bit input data is processed separately during 10 iterations, called "rounds". Each round repeats several operations, *e.g.*, XORing, S-Boxes, shifts. AES has been a target for side-channel attacks for two decades, *e.g.*, with timing attacks targeting the T-Table implementation [Ber05], Differential Power

Attack (DPA) targeting the first "AddRoundKey" operation [OGOP04], or Simple Power Attack (SPA) targeting the key scheduling algorithm [Man03].

Metrics Several metrics are used in the state of the art to evaluate a side-channel attack. The *Partial Guessing Entropy (PGE)* [PDY16] defines the rank (*i.e.*, the index) of the correct subkey among a list of all possible subkeys classified from the most probable to the least probable according to the side-channel output. For a single subkey, it hence estimates how many guesses are needed to find the correct subkey. The *Key Rank* [VCGS13] defines the rank (*i.e.*, the index) of the correct key among a list of all possible keys classified from the most probable to the least probable. Estimating how many guesses are needed to find the correct key is representative of the complexity of the key recovery. The key rank enumeration is a key brute force leveraging knowledge of the side-channel output.

Measurement Requirements From a side channel problem formalization by *Ouladj et al.* [OEMG+20], we identify two requirements on measurements: 1. The measured trace is a 1D real-valued vector. 2. For a given value of a given cryptographic input, the leakage output will be constant – the relation is deterministic. They will become challenges when analyzing the phase in the side-channel context, described in Section 5.1.

2.6 Multi-Channel Attacks

Agrawal et al. [ARR03] were the first to introduce the concept of *multi-channel attack*. At first, it designates the use of multiple channels in a single attack, *e.g.*, exploiting the power and EMR source simultaneously. Since then, it has been generalized to multiple sources of information which are combined to increase the performance of a single side-channel attack. *Yang et al.* [YZC+17] systematized multi-channel attacks, also called *fusion* or *combination* depending on the context, or using the Multi-Channel Fusion Attack (MCFA) acronym. [YZC+17] classified MCFA algorithms into 3 categories that depend on the level at which a chosen combination function is applied to merge the channels: 1. *Data-level*, where the combination function is used to merge the samples. 2. *Feature-level*, where the combination function is used to merge features extracted from the samples. 3. *Decision-level*, where the combination function is used to merge the final result of independent attacks on each channel. Several strategies exist, *e.g.*, combining several intermediate value targets in the cryptographic algorithm [MOW14], using multi-dimensional algorithms [GMGH19], or combining time samples using the product of the PCC [Mey12] – a form of time diversity.

3 Related Work

In this section, we expose foundational work in electromagnetic (EM) side channels as well as early hypotheses about unintended angle modulation by phase shift. While recent work has been done in exploiting signal jitter – timing variations – in side-channel attacks, none of them explored its impact on phase of electromagnetic radiation (EMR). Our motivation is to create a bridge through a side-channel perspective between jitter, a measure more common in electronics, and phase shift, a measure more common in radio electricity.

3.1 Foundational Work: Electromagnetic Side Channels

Electromagnetic radiation (EMR) has been identified to be a threat against cryptosystems by *Quisquater et al.* [QS01]. In this research, measuring EMR is depicted as a proxy for power consumption measurement without galvanic conduction. Numerous algorithms have been developed to exploit EMR measurement, the first ones being Simple EM Attack (SEMA) and Differential EM Attack (DEMA) [AARR02] which are analogous to

Simple Power Attack (SPA) and Differential Power Attack (DPA) [KJJ99] commonly used in power side-channels. EM side-channels have been systematically categorized by *Lavaud et. al* [LGG+21] depending on the emanation origin. While differing in architecture, oscilloscopes, and radios are common measuring instruments for EM side channels.

To the best of our knowledge, in the current state of the art about EM side-channel attacks using radios, all of them are performed using amplitude traces instead of phase traces (*e.g.*, [VP09, CPM+18, WWD20, GPPT22]). We explain this by two facts. First, amplitude traces have been found to leak strongly enough to exploit them empirically. Second, working with amplitude is easier in practice since it provides an absolute measure, compared to the phase, which is a relative and cyclic measure that requires synchronization or post-processing. Hence, in this paper, we analyze how to exploit the phase of the recorded signal from a side-channel perspective.

Early Hypothesis about Angle Modulation in EM Side Channels In the *NACSIM 5000: TEMPEST Fundamentals* standard from the *NSA* [Ros82], declassified in 2000, there is one mention of angle-modulated carrier in EM compromising emanations. However, this document (partially redacted) does not propose any root cause hypothesis or demonstration. In 2003, *Agrawal et al.* [AARR03] started to partially investigate the idea of angle modulation in an EM side-channel signal. This phase or frequency modulation would be caused by bad isolation between a data signal and a signal generation circuit. This work experimentally showed a frequency-modulated leakage dependent on one bit of data using a frequency-domain analysis through the Fourier Transform. However, it has not performed an end-to-end attack against a cryptosystem and does not assess the relation to phase modulation. In 2005, *Li et al.* [LMM05] emitted a similar idea, where a data line would be coupled to a Voltage-Controlled Oscillator (VCO) input voltage, resulting in angle modulation of the clock. This modulation would be visible as data-dependent timings in the time domain or frequency variations in the frequency domain. In 2011, *Kocher et al.* [KJJR11] mentioned that performing an angle demodulation before the analog-to-digital conversion may help to isolate the signal – without any additional explanation. Apart from these hypotheses, to our knowledge, no paper tried to asses side-channel leakage on the phase of a signal – *i.e.*, exploiting a leakage performing an unintended angle modulation.

3.2 Recent Work: Timing Side Channels Exploiting Jitter

Recent work has been done in timing side channels by measuring the signal jitter of clock signals. In 2021, *Gravellier et al.* [GDTM21] exploited signal jitter in delay lines to perform a remote power side-channel attack on AES. They explained that the jitter is induced by the coupling of voltage and temperature variations with power consumption. They measured the jitter using a software-based method reading registers of a delay locked loop (DLL). In 2023, *Schoos et al.* [SMTG23] published *JitSCA*, which exploits a signal jitter at the picoscale resolution of a clock signal to perform a power side-channel attack on AES. They measured the jitter using a time-to-digital converter (TDC) implemented using a delay-line in an FPGA, allowing them to have a higher time resolution than a typical oscilloscope. A follow-up blog post by Riscure [Wit] discussed *JitSCA*, and emitted the hypothesis of phase-locked loops (PLLs) being one of the root causes of signal jitter. This was also one of our hypotheses, to which we demonstrate it is a possible root cause but certainly not the only one.

3.3 Parallel Work: Side-Channel Exploiting Phase Modulation

Shortly before publishing our work, *Colin O'Flynn* [O'F24] published a paper which shares similarities to our results. Similarly to our work, *O'Flynn* work tries to fill the gap between *Agrawal et al.* [AARR03] hypothesis and *Schoos* [SMTG23] jitter exploitation. Moreover,

it also demonstrates the first side channel using an unintended phase modulation on an optical link and a digital bus (JTAG). Contrary to our work regarding the phase measurement, *O'Flynn* work uses a physical connection to the targeted clock signal. In our case, we do not need any conducted measurement through physical connection nor reference clock signal since our measurement is EM-based in the near-field (NF). Since we need fewer requirements and use only portable COTS hardware, we believe that our measurement method is simpler to use. In this paper, we propose several contributions complementary to *O'Flynn* work. First, we show how to exploit electromagnetic radiation (EMR) measurements using an SDR through a near-field (NF) probe to measure the phase shift and demonstrate a method to bypass the need for a reference signal. Second, we evaluated the side channel on several widespread SoCs, and we used a multi-channel attack to add evidence that angle modulation contains complementary information to amplitude modulation from the attacker's perspective. Finally, we experimentally demonstrate probable sources of data-dependent jitter and the relation between jitter and phase shift in the attacked SoCs.

Regarding previous work, our paper demonstrates that while jitter measurement is an indirect measure of power consumption in side channels, phase shift measurement is an indirect measure of jitter.

4 Threat Model

We consider an attacker who aims to obtain a secret processed by a target device (*.e.g*, a microcontroller) during a sensitive operation. In this paper, we aim to obtain the secret key processed by a cryptographic algorithm, however, the attack is generic and could impact other information types. The attacker will conduct a known-plaintext side-channel attack, recovering the key by correlating a set of measurements to a pre-computed model. The measurements are recorded using radio equipment when the cryptographic operation is occurring, in the target device vicinity (from millimeters to centimeters) using an EM probe. All our assumptions are standards for non-intrusive and passive side-channel attacks in the literature.

5 Methodology

Section 5.1 introduces our method to compute a trace usable in a side-channel attack from the phase of a signal to address RQ1. Section 5.2 introduces the method we used to recombine information for RQ1. Section 5.3 explains how we choose several SoCs to attack and the implications for RQ2. Finally, Section 5.4 introduces our reproduction methodology for RQ3.

5.1 Side-Channel Trace: From Complex-Valued Signal to Real-Valued Phase Trace

As seen in Section 3.1, side-channel algorithms need a 1D real-valued vector – called a trace – and a deterministic measure to work with. While the amplitude computation is absolute and can be used as-is, as seen in Section 2.2, the phase measure is relative, as presented in Section 2.3. Because the analytic signal recorded using a SDR is complex-valued, it is represented as a 2D real-valued vector. Thus, we cannot use the I/Q samples as-is for a side-channel attack. Because the phase computation is relative, it breaks the deterministic requirement for a measure, thus we cannot use the phase trace as-is for a side channel attack either. In this section, we explain how we post-process the measured signal to generate a phase trace that fulfills the two requirements presented in Section 2.5. Figure 2

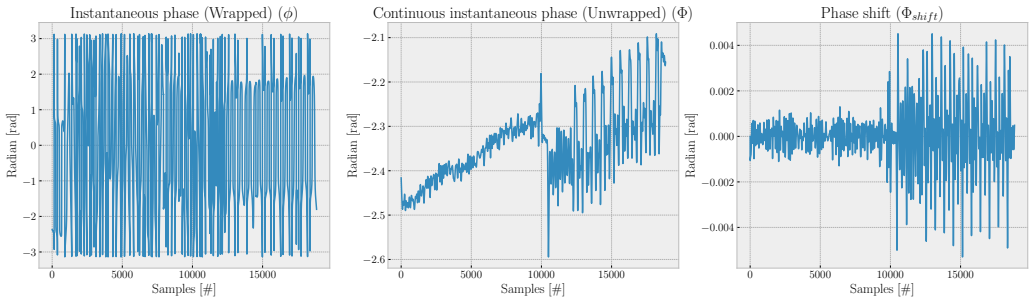


Figure 2: Phase trace (3 steps of computation).

illustrates 3 of the 4 steps that we present in this section to compute our trace. Such a phase trace is usable in a side-channel attack and allows us to answer half of RQ1 by testing for a data-dependent leakage.

Instantaneous Phase Analysis In the first step, we show how we can analyze the potential impact of system activity on the phase of our signal, exhibiting a possible data-dependent side-channel leakage. The recorded signal stored as complex numbers uses the analytic representation, as described in Section 2.2. We compute the instantaneous wrapped phase of our signal, the real-valued function $\phi \in [-\pi, \pi]$, by taking the argument of the complex-valued function $x(t)$ as shown in Equation 4:

$$\phi(t) = \arg(x(t)) = \arctan2\left(Q(t), I(t)\right) \tag{4}$$

The instantaneous wrapped phase represents the phase modulo 2π, *i.e.*, constrained to its principal value. The process of retrieving true continuous phase information is known as phase unwrapping [HMW+22], which leads to the continuous instantaneous phase (CIP) $\Phi(t)$ function, defined in Equation 5:

$$\Phi(t) = \phi(t) + k(t)2\pi, \tag{5}$$

with $k \in \{0, 1, 2, ...\}$ an integer multiple of 2π increased each time a 2π discontinuity is encountered in $\phi(t)$. Therefore, the $\Phi(t)$ is a cumulative function, not constrained to the 2π principal value. Considering the requirement of building a dataset to distinguish any side-channel leakage, we want to compare the CIP of multiple signals. Hence, we set our signal relative to 0 to have a common starting point, as defined in Equation 6:

$$\Phi_0(t) = \Phi(t) - \Phi(0) \tag{6}$$

Using the relative to zero CIP function $\Phi_0(t)$, we can test if the device under test is performing an unintended phase modulation through its EMR. Indeed, if two signals have a different phase shift at some point, they will diverge. If the difference in phase shift is due to the impact of two different system activities, side-channel information may be extracted from those signals.

Phase Shift Analysis The previous function Φ_0 is cumulative, which is not suitable for computing correlations and performing side-channel attacks. We must transform this function into one that will lead to a trace exploitable by side-channel algorithms. Instead of analyzing the CIP, we compute its first derivative, *i.e.*, the magnitude of changes in phase – the phase shift between two samples. Numerical differentiation of Φ_0 over time can be defined as $\Phi_{shift}(t) \in [-\pi, \pi]$ as Equation 7:

$$\Phi_{shift}(t) = \frac{d\Phi_0}{dt}(t) = \begin{cases} 0, & \text{if } t = 0 \\ \Phi_0(t) - \Phi_0(t-1), & \text{otherwise} \end{cases} \tag{7}$$

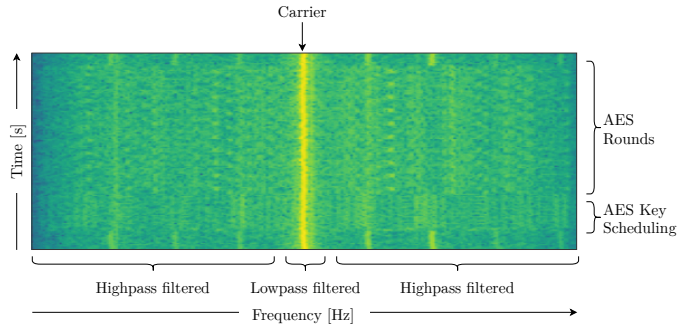


Figure 3: Waterfall illustrating filters isolating amplitude and phase shift leakage.

Φ_{shift} is now a function constrained to the 2π principal value, but without discontinuities and which value represents the variation to the previous sample. It hence resolves the problem of phase synchronization exposed in Section 2.3. After this transformation, the signal is now converted into a usable trace for a side-channel algorithm and can be used similarly to amplitude signals commonly exploited in state-of-the-art side-channel papers. This concludes our first part of C1.

5.2 Side-Channel Attack: Using our Phase Shift Trace

Leveraging our method in Section 5.1, we now have a phase shift trace that meets the requirements presented in Section 2.5. First, we explain how we performed our side-channel attack using only the phase shift, corresponding to the second contribution of C1. Next, we explain how we recombined amplitude and phase information into a single attack to increase the performance, corresponding to the contribution C2.

Mono-Channel Attack: Profiled and Non-Profiled Attacks on Phase Shift Our attacks target the step after the AES S-Box (`AddRoundKey` and `SubBytes`) inside the first round.

For our *non-profiled* attack, we used a standard correlation attack used by Camurati *et al.* named Correlation Radio Analysis (CRA) [CPM+18], which is a variation of Correlation Electromagnetic Analysis (CEMA) [Mey12] in the context of EM analysis [QS01], itself inspired from Correlation Power Analysis (CPA) [BCO04]. In this attack, we only collect a single dataset with random known plaintexts and a fixed unknown key. The side-channel output is the key that will maximize the correlation between this dataset and the theoretical model.

For our *profiled* attack, we used the Profiled Correlation Attack (PCA) used by Camurti *et al.* [CFS20]. First, in this attack, we build a profile using a similar device that can be instrumented before the attack, by collecting a training dataset with random known plaintexts and keys. Theoretically, the profile is created and then used across two different instances of the devices. In our laboratory setup, we used the same device for both – which does not change the validity of our results but increases performance for profiled attacks. Eventually, time diversity is possible by averaging traces where the encryption has been executed with the same input. Second, we collect an attack dataset using the target device, with random known plaintexts but a fixed unknown key. The side-channel output is the key that will maximize the correlation between the attack dataset and the pre-computed profile.

We additionally test the hypothesis that, for a given carrier frequency, the amplitude signal and the phase signal do not impact the same frequencies in the spectrum. For the amplitude signal, we assume that its impact is mainly located in the sidebands of the modulated carrier, hence, in higher frequencies around our center frequency. We make this

assumption since an amplitude-modulated signal is the algebraic sum of his carrier and its two sidebands signals, in which the power is mainly distributed [Fre16, p. 99]. For the phase signal, we assume that it is close to the carrier frequency, hence, in lower frequencies around our center frequency. We make this assumption since a phase-modulated signal induces small variations in the instantaneous frequency of the carrier, in which the power in mainly contained. By using an additional pre-processing step on the complex signal, before computing the phase trace, we apply high-pass and low-pass filters, illustrated in Figure 3. Observing the impact of the filters on the side-channel attack allows us to identify which frequencies are mainly impacted by the side-channel information.

Multi-Channel Attack: Combining Amplitude and Phase in a Single Attack Leveraging multi-channel attacks presented in Section 2.6, we performed recombination at the decision level. We chose this level because it is best suited to be implemented with our Profiled Correlation attack described above. If it improves the results, even without being the best recombination method, it is sufficient to support our hypothesis that phase and amplitude leakages may be complementary. Our technique is inspired by *Meynard* [Mey12], which uses the product as a combination function applied to correlation coefficients of different time samples. We first perform two individual attacks, one using the amplitude and one using the phase, which results in $\rho_A(sk, g)$ and $\rho_\Phi(sk, g)$, respectively. They correspond to the value of our distinguisher, the Pearson Correlation Coefficient (PCC), for all subkeys sk – 16 in AES-128 – and possible guess g – 256 values using $p \oplus k$ as leakage variable. In our case, we use the sum as a combination function applied on PCCs of different "channels", *i.e.*, the amplitude and the phase, defined in Equation 8:

$$\rho_{\text{multi}}(sk, g) = \rho_A(sk, g) + \rho_\Phi(sk, g) \tag{8}$$

We then use ρ_{multi} as our new distinguisher to perform the final decision. Using this recombination method, we test our hypothesis that both components can be used simultaneously to increase performance over using only one.

5.3 Generalization: Attacking multiple SoCs using Phase Shift

Leveraging our attack from Section 5.2, we are now able to conduct an attack on an SoC for both amplitudes, similarly to prior work and phase, using our new method. We want to know if the presence of this unintended angle modulation – leakage on the phase – was common across several SoCs. To test this assumption, we select the most popular SoCs used for microcontrollers or IoT applications, detailed in Section 6.1. For each selected SoC, we record datasets using a center frequency similar to the fundamental frequency of the system clock, accurately measured using a spectrum analyzer. Some of these SoCs exhibit interesting differences regarding the separation between power domains and the generation of clock signals. Evaluating if leakage is present or not, regarding the microarchitecture of the SoC, provides an important insight into the root cause, detailed in Section 5.4. We also want to know if the observation of an unintentional amplitude modulation leaked from an SoC systematically results in the observation of an angle modulation leaked – or if these two phenomena are independent and can be found separately. The evaluation and answer to those questions constitute our contribution to C3.

5.4 Reproduction: Inducing Jitter and Phase Shifts in a Controlled Environment

Motivations As seen in Section 2.1, jitter on digital lines can be generated due to coupling with other components in their vicinity. Previous research introduced in Sections 3.2 and 3.3 already exploited this effect within integrated circuits to extract secret data from

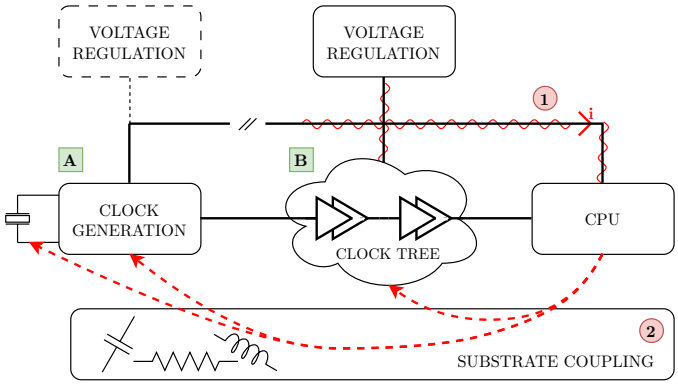


Figure 4: Hypothetical jitter source from processor activity.

a processor. The influence of the temperature and supply voltage on the delays of clock transmission lines, such as clock buffers and delay lines, is often designated as a root cause hypothesis. However, we suggest that there might be other causes that correlate processor activity with jitter. As such, this emerging type of timing side-channel can potentially be generalized to a broader range of integrated circuits, from low-end microcontrollers to highly integrated System-on-Chips. Aside from showing the exploitability using phase measurements on the EM field, our research focused on filling the gap between the exploitability of the phenomenon and its root causes. To do so, we empirically observed the implication of various hardware components with the observable jitter in order to answer to RQ3.

Figure 4 introduces a coarse view of a microcontroller, highlighting its clock circuit, processor, and shared power supply. It depicts an overview of the potential sources of jitter caused by processor activity, summarizing knowledge from the literature. We denote two types of sensitive elements prone to generate jitter. First is the clock generation circuit [A], responsible for generating the various required clocks to the digital circuit. It generally employs an oscillator circuit for base clock generation and a PLL for further clock synthesis. Both oscillators [HP00, Moh11] and PLL [BOFM02, HP, Li18] are sensitive to voltage level variations, resulting in additional jitter on their respective outputs. The security implications of this phenomenon were previously introduced by *Agrawal et al.* [AARR03]. Second is the clock-tree [B], which is responsible for delivering clock signals to various digital components. It employs a set of buffers to balance the clock timing across the digital circuit. As such, a clock tree is similar to delay lines, as exploited by *Gravellier et al.* [GDTM21], *Schoos et al.* [SMTG23], and is affected in the same way by supply voltage fluctuations [MSY13].

The processor can affect sensitive components through a main (1) or parasitic (2) coupling. The main coupling refers to voltage variations on the power supply line due to the variable consumption caused by the processor activity. This is due to the load-regulation capability of voltage regulators [Lee], a phenomenon usually exploited to perform power side channels. In many integrated circuit designs, the supply lines of clock generation circuits and digital logic are decoupled and provided by different power supplies. However, coupling might happen over the silicon substrate (2) due to parasitic capacitive and inductive effects.

Jitter Source Study We developed a strategy to empirically confirm our previous hypothesis. The core idea is to measure the jitter of a clock signal from a chosen microcontroller while setting the latter in various configurations and states. The microcontroller's clock line can come from either a direct internal system clock or a derivative of the latter, such as a clock line from a synchronous bus (*e.g.*, SPI, I2S, JTAG). Using custom firmware, the

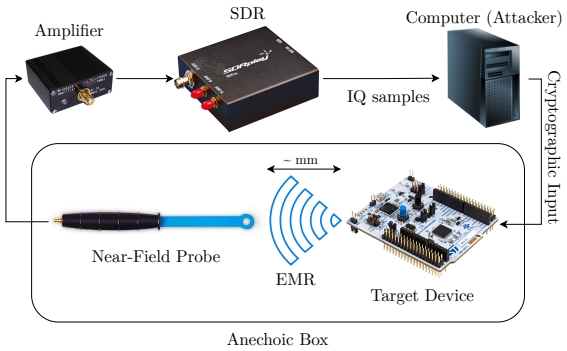


Figure 5: Hardware experimental setup for side-channel evaluation.

Table 1: Multiple target devices and SoCs evaluated in the side-channel analysis.

Ref.	SoC	CPU	Board	Clock generator	Clock freq. [MHz]
[STMb]	STM32L1	ARM Cortex-M3	NUCLEO-L152RE	RC + PLL	32
[Sem21]	nRF52832	ARM Cortex-M4	PCA10040	PLL	64
[Sem13]	nRF51422	ARM Cortex-M0	PCA10028	RC	16
[Mic]	ATmega328	megaAVR	Arduino Nano	RC	16
[Pi]	RP2040	ARM Cortex-M0	Raspberry Pi Pico	PLL	125

microcontroller executes different sets of instructions, each expected to exhibit different consumption profiles. The microcontroller's internal clock circuit should include various sensitive circuits, such as oscillators and clock synthesizers, which can be selectively enabled. Hence, we can determine if: 1) The observed microcontroller jitter is correlated to the processor activity. 2) Some internal clock generation circuits have a significant impact on the observable jitter. For this purpose, we use an external hardware circuit that induces a controllable current flow on a microcontroller's GPIO. This current flow increases the overall chip consumption independently from the processor. Additionally, the phase shift EM field is also measured along with jitter measurement on the clock line, using the methodology introduced in Section 5.1. With this setup, we can determine if: 1) The clock jitter is correlated not only to the processor activity but also to the overall chip consumption. 2) The clock jitter directly translates to a phase shift observable on the EM field.

6 Experimental Setup

In this section, we present two hardware experimental setups. The first setup presented in Section 6.1 is used to evaluate the side-channel attack on the phase trace for several SoCs, used for research questions RQ1/2. The second setup presented in Section 6.2 is used to study the source phenomenon of the unintended angle modulation, used for research questions RQ3.

6.1 Side-Channel Attack

Figure 5 illustrates our experimental setup used in Sections 5.2 and 5.3.

Hardware The radio receiver in use is an SDRPlay RSPdx [SDR24], allowing to record a signal down to 1 kHz, hence capturing "low frequency" clock signal of several MHz. Moreover, it allows to record 10 MHz of bandwidth at once, enabling the capture of a large

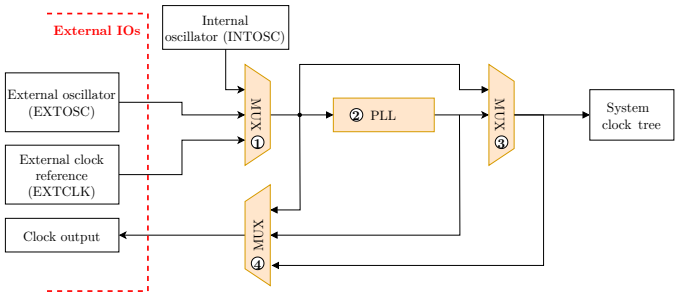


Figure 6: Simplified view of the STM32F103RB internal clock system.

band of EMR where we apply different filters in post-processing. The input of our SDR is the measured unintentional EMR of the target device using the TekBox TBPS01 [Tek24] near-field (NF) probe placed at a few millimeters of the target, without galvanic contact. For some SoCs that exhibit a weak leakage, the TekBox TBWA2 [Teka] amplifier was used to amplify the measured EMR. The SDR is connected to a standard desktop computer through USB 3.0, sending raw I/Q during the recording. For reproducibility purposes, we isolated our measurement setup from RF environmental noise inside an anechoic box. The evaluated target devices are presented in Table 1. All devices have been chosen to be representative of modern SoCs, with different processor and clock generation circuit combinations.

Software The computer is running a program controlling the target device using a serial interface. The target device is running a custom C firmware, embedding a TinyAES [Kok] software implementation, and a hardware implementation for some SoCs (when available). In our evaluation, we focus on the software AES since its leakage is stronger than the hardware AES and the focus of the paper is not to assess the difficulty of attacking hardware-based implementation.

6.2 Jitter and Phase Shift Reproduction

Target Microcontroller We use the STM32F103RB microcontroller [STMa] from STMicroelectronics, which is mounted on the NUCLEO-F103RB development board [STMc]. We chose this microcontroller because it offers sufficient flexibility in terms of clock configuration to apply the methodology described in Section 5.4. As depicted in Figure 6, the STM32F1 allows switching between an internal or externally provided oscillator or clock reference ①. To provide a system clock higher than the reference one, the microcontroller uses an optionally ③ selectable PLL, whose output frequency can be tuned by the user ②. We leveraged this feature to compare the jitter effects of different sensitive clock circuits under different configurations. More importantly, the STM32 family can route an internal clock line to an external GPIO ④. This hardware feature permits direct measurements on the internal system clock (*i.e.*, the clock fed to the digital logic), as well as the oscillator and PLL output clock signals. We configure the STM32F103RB with a custom firmware with TinyAES implementation (like in Section 6.1). Additionally, we added functions to reconfigure the internal clock system on-demand at runtime. We furthermore implemented the following processor states that are expected to induce different power consumption profiles:

SLEEP The processor is completely powered off by clock-gating, expecting the lowest overall consumption.

AES The processor is enabled and continuously performs AES computation.

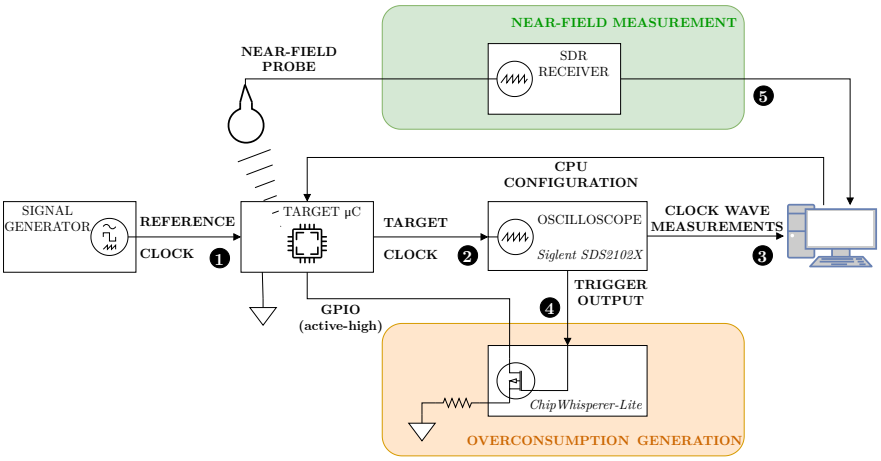


Figure 7: Hardware setup for jitter source study.

STALL An intermediate step between SLEEP and AES where the active processor executes an infinite while loop. This way, the processor only performs a branching construction without involving data memory access or arithmetic computation.

Measurement Hardware Figure 7 illustrates our complete testbed for jitter reproduction, implementing the methodology from Section 5.4. It mainly comprises a Siglent SDS2102X real-time oscilloscope [Sig] and the STM32F103RB target microcontroller. As seen in Figure 6, some STM32F1 clock configurations require an external reference that we provide using the built-in signal generator function from the oscilloscope ①. Independently from other clock configurations or processor states, the microcontroller is configured to constantly output its internal CPU system clock on an external pin for further measurement. This target clock signal is connected to the input of the oscilloscope ②. The latter is configured to periodically trigger its acquisition on any clock-rising edge, acquiring as many traces as possible at a sampling rate of 2 GSa/s. The oscilloscope is interfaced with a computer through the standardized SCPI protocol, providing continuous clock measurement traces ③. However, the communication latency does not allow retrieving enough waveforms to perform statistical computation on the jitter in a reasonable amount of time. As a workaround, we configured the oscilloscope to perform measurement and statistics of the clock period locally at the highest possible rate. This way, we collect the statistical results on the host computer only after a complete round of measurements.

To generate overconsumption independently from the processor activity, we use the glitch circuit of a ChipWhisperer-Lite [New], originally designed to conduct hardware fault attacks ④. It embeds an FPGA for fast trigger acquisition and a MOSFET to perform short circuits. The ChipWhisperer trigger input is connected to the trigger output of the oscilloscope. The MOSFET drain is connected to a General-Purpose Input/Output (GPIO) from the target, which is configured to provide a continuous active-high output with an internal pull-up. On every oscilloscope acquisition trigger, a rapid short circuit of configurable length is generated on the target. This way, the oscilloscope, and the glitching circuit are synchronous, allowing us to observe the jitter effect of overconsumption on each clock trace acquisition. We reused the same setup described in Section 6.1 to observe the relation between the measured jitter on the clock and phase shifts in the EM field. Using the SDRPlay RSPdx, we tuned it to the configured system clock frequency, with an EM probe placed in the vicinity of the microcontroller ⑤.

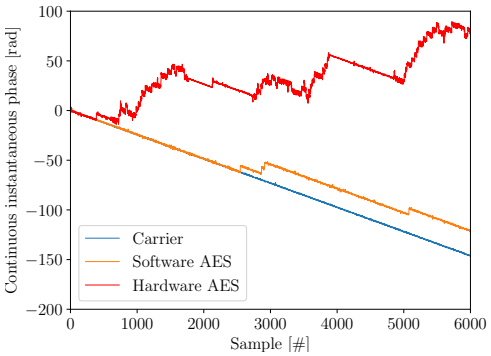


Figure 8: AES influence on the continuous instantaneous phase (CIP) of signals.

7 Evaluation

This section details the results of our methodology presented in Section 5, using the experimental setup outlined in Section 6. In particular, Section 7.1 provides a preliminary answer to RQ1 and RQ3. Section 7.2 formally answers to RQ1/2. Section 7.3 answers to RQ3.

7.1 Identifying Unintended Phase Modulation on a Target SoC

The first step before performing a side-channel attack is to identify if the target device is emanating EMR that depends on the system activity. Using our setup from Section 6.1 and leveraging our 1^{st} method exposed in Section 5.1, we performed what we call an "Instantaneous phase analysis".

Instantaneous Phase Analysis Figure 8 shows three relative to zero CIP functions ($\Phi_0(t)$), where only the beginning of recorded signals are shown for visualization purposes. The blue signal represents the CIP of the signal at a system clock harmonic without any system activity. This signal follows a constant trend across time, representing a null phase shift. The two other orange and red signals represent the CIP of the recorded signal when there is an additional system activity, respectively, software and hardware AES. We can see that those signals have variations that are not present in the first one, representing phase shifts. From this analysis, we conclude that both types of AES have an influence on the phase of the recorded signal. It raises the question of data-dependent phase modulation, implying the presence of an information leakage that may allow an attacker to compute correlations with the processed data. To answer this question, leveraging our 2^{nd} method exposed in Section 5.1, we performed what we called a "Phase shift analysis".

Phase Shift Analysis Using $\Phi_{shift}(t)$ from Section 5.1 allows us to analyze the phase shift on the recorded signal generated by the system activity, more precisely, the execution of the software AES. In Figure 9, we show the resulting traces for the amplitude and the phase shift in both the time domain and frequency domain. This amplitude signal (left) is the one exploited by conventional EM side-channel attacks from the state of the art. This phase signal (right) is clearly generated by the AES execution, as we can identify the different steps of the algorithm. The two traces appear very similar, suggesting that the data-dependent leakage, *i.e.*, a side-channel phase modulation, is present in our recorded signal. To the best of our knowledge, it has never been exploited in the state of the art prior to this paper, and this is our preliminary contribution C1.

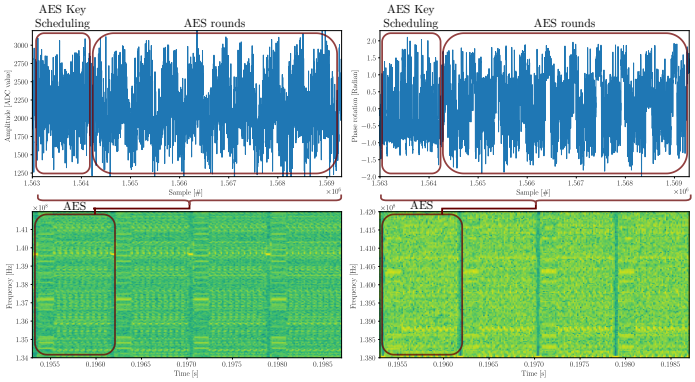


Figure 9: AES trace in amplitude (left) and phase shift (right).

Table 2: Summary of results for evaluated SoCs in the side-channel analysis. Legends: ✅ successful exploitation, ❌ unsuccessful exploitation

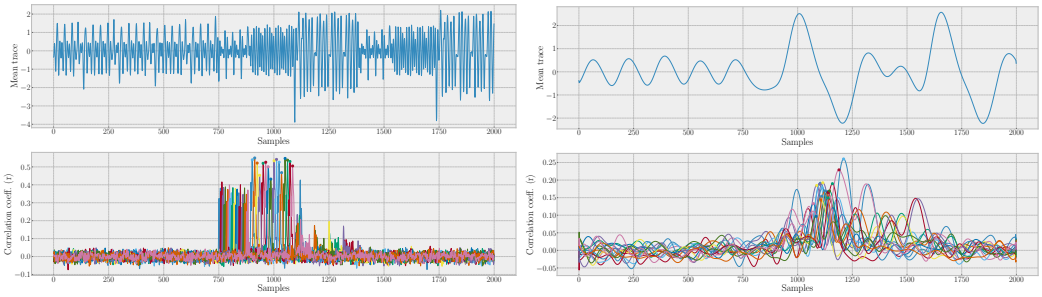
SoC	Identify AM / PM (Section 7.1)	Key recovery AM / PM (Section 7.2)
STM32L1	✅ / ✅	✅ / ✅
nRF52832	✅ / ✅	✅ / ✅
nRF51422	✅ / ✅	✅ / ✅
ATmega328	✅ / ✅	✅ / ✅
RP2040	❌ / ❌	❌ / ❌

Results on Different SoCs We evaluated the presented analysis on the SoCs introduced in Section 6.1. The results are presented in the "Identify AM / PM" column of Table 2. Except for the RP2040, every tested SoC exhibits a leakage in the phase shift trace similar to Figure 9. In other words, every positively tested SoC suffers from an unintended angle modulation, which is our preliminary contribution C3. Concerning the RP2040, while the spectrum is affected by the CPU activity, we did not find any exploitable side-channel signal. This absence of exploitable signals could be linked to various factors, like a dynamic frequency scaling that could add distortion to the leakage or better isolation of the power domains reducing the leakage. The experimental results suggest that when a leakage is found on the amplitude, it is also found on the phase shift. This correlates with the results we will discuss in Section 7.3, indicating that both amplitude and phase shift are a proxy measurement for power consumption.

7.2 Side-Channel Attack using Phase Shift

Based on the methodology presented in Section 5.2 and on the setup presented in Section 6.1, we conducted a side-channel attack using phase shift. First, we collected training datasets of 4000 traces and attack datasets of 1000 traces for the nRF52 and the STM32L1 SoCs. Training datasets were used to create profiles of the leakage with different software filters.

Signal Filtering Figure 10 is one example that shows the PCC (ρ) between the data inputs and the measured traces in the training dataset for each sample, in order to find the point of interests (POIs). It demonstrates that the leakage of the phase trace is significant (> 0.5) on several time samples. While the signals were recorded using a bandwidth of 10MHz around the carrier, filtering the signal as described in Figure 3 with a low-pass filter of 50kHz leads to a profile still usable to perform a full key recovery. This experiment shows that the signal of interest in the phase shift leakage is a low-frequency signal, *i.e.*, the main frequency components are close to the carrier frequency in the frequency domain. On the



(a) Using a low-pass filtered signal at 1MHz. (b) Using a low-pass filtered signal at 50kHz.

Figure 10: Correlation coefficients (ρ) for POIs on phase shift for the nRF52.

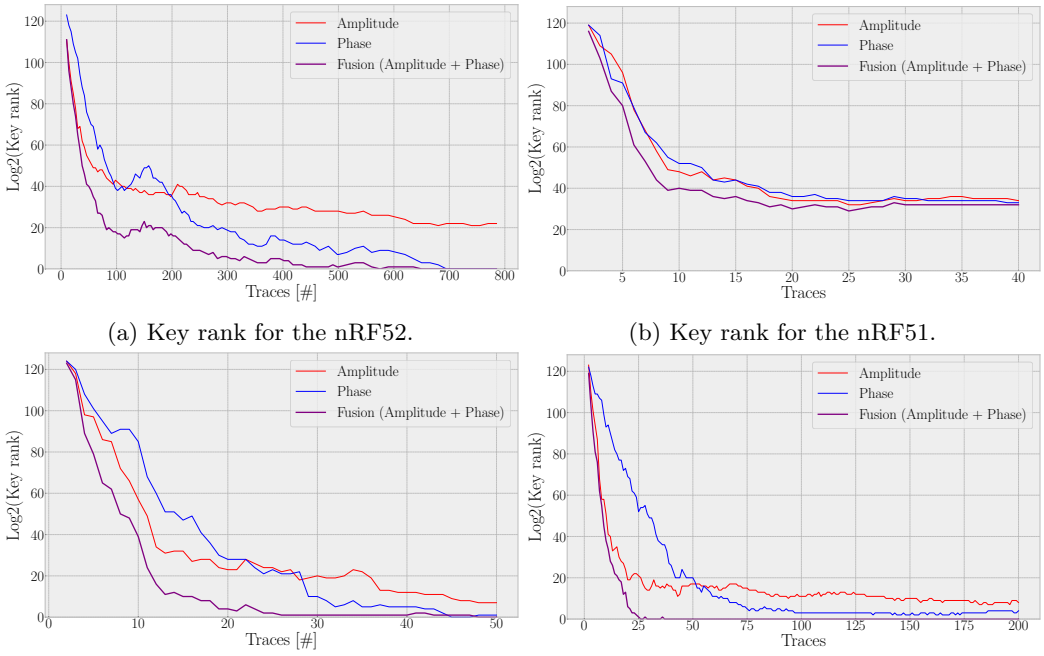
contrary, we observe an opposite phenomenon for the amplitude: a high-pass filter improves the profiles and the performances, while a low-pass filter degrades them. This experiment shows that the signal of interest in the amplitude leakage is a high-frequency signal, *i.e.*, contained in the wide sidebands of the carrier in the frequency domain. Applying a filter matching our signal of interest allows us to improve the attack efficiency by diminishing the noise added by frequency bands out of interest.

Phase Performance and Fusion with Amplitude Figure 11 and Figure 12 show the key rank over the number of traces for profiled and non-profiled attacks, respectively. We evaluate our attacks for the nRF52, nRF51, the ATmega328, and the STM32L1, listed under the "Key Recovery AM / PM" column of Table 2. The figure combines mono-channel attacks independently performed on both amplitude and phase, as well as the multi-channel attack combining amplitude and phase. First, our attacks show that using only the phase is often better than using the amplitude, *e.g.*, from an order of magnitude of 22 for nRF52 using 700 traces with profiled attack or 40 for the nRF51 using 1000 traces with non-profiled attack. Second, we observe that recombining amplitude and phase systematically perform better than attacking using them independently, *e.g.*, from an order of magnitude of 20 for the nRF52 using 100 traces with the profiled attack or using 150 traces with the non-profiled attack. This significant performance increase using a recombination method implies that, even if the radiated information originates from the same phenomenon on the emitter side, the measured information is complementary when exploiting the two components simultaneously from the receiver side.

As a conclusion to C1, we can see that our method of computing a phase trace allows the exploitation of phase in a side channel attack. Concerning C2, we observe that the phase shift trace can lead to better performance than amplitude and that combining the two seems to systematically improve attack performance. It is noteworthy that we did not perform more physical measurements than state-of-the-art attacks on amplitude: we exploit the signal using an alternative approach, leading to a significant increase in performance. Finally, regarding C3, all tested SoCs except the RP2040 – which do not seem to leak, whether considering amplitude or phase – had strong evidence for the presence of the phase leakage, which we experimentally demonstrated for all of them.

7.3 Jitter and Phase Shift Reproduction

Based on the methodology presented in Section 5.4 and setup presented in Section 6.2, we conducted experiments under various microcontroller conditions. For each experiment, we collected the mean, standard deviation, minimum, and maximum values of clock period measurements computed over 10,000 clock traces collection.



(a) Key rank for the nRF52.

(b) Key rank for the nRF51.

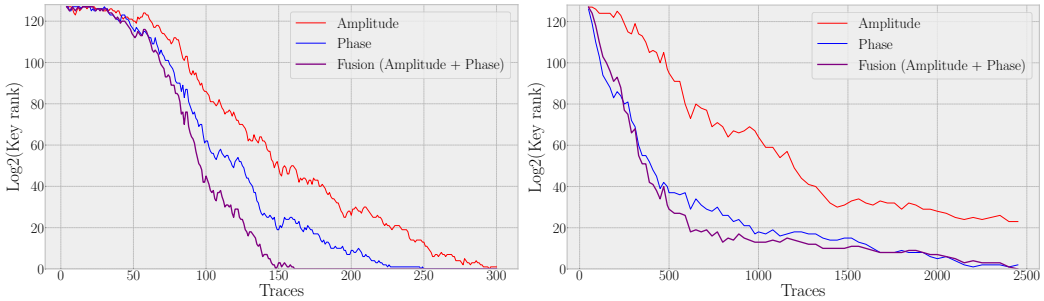
(c) Key rank for the STM32L1.

(d) Key rank for the ATmega328.

Figure 11: Performance over number of traces for profiled attacks. The smaller, the better.

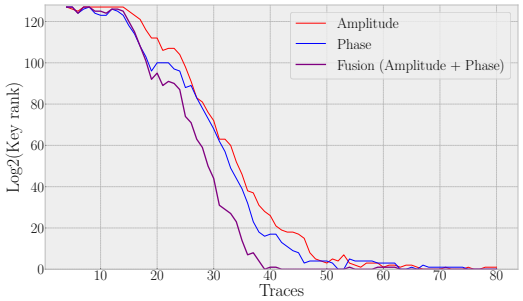
Jitter *vs.* Consumption We compared jitter measurements under different power consumption caused by both processor activity and overconsumption induced through a GPIO connected to the glitching circuit of the ChipWhisperer. Results are shown in Figure 13a. The mean value is represented at the center of each bar plot, where standard deviation, minimum, and maximum values are spread along. The processor's system clock is set at 64 MHz (15.625 ns period), highlighted by the central red line in the figure. Deviation from the nominal period denotes higher jitter values. Results show that the sole processor activity (blue lines) impacts the clock jitter, denoted by a substantial increase of standard deviation between the *sleep* and *stall/aes* states. We, however, observe minor differences between the two *stall* and *aes* states. Independently from the processor activity, occurring overconsumption appear to affect the jitter deviation significantly. Interestingly, we denote a mean value shifting around the nominal period in those cases. This can be explained by the fact that our observations of the clock signal occur during overconsumption events, systematically shifting the period in a specific direction. Regarding the direction of the shifts, we assume that, the rising and falling edges of overconsumption events cause transient current flows in both directions, which conversely results in positive or negative shifts of the internal supply voltage.

Jitter *vs.* Clock Configuration We compared the jitter measurements under different microcontroller clock configurations in the goal of determnining the clock circuit that potentially contribute to the most data-dependent jitter effect. Specifically, we examined the effects of using the internal oscillator (INTOSC) versus an external clock source (EXTCLK), both with and without the PLL enabled. Disabling the PLL limits the CPU system clock frequency to the frequency of the oscillator (8 MHz). Thus, we set the system clock to 8 MHz in all experiments to ensure the correctness of the comparison. Results are shown in Figure 13b. We observe that the oscillators (both internal and external) without the use of the PLL contribute to most of the jitter effect when the processor switches from

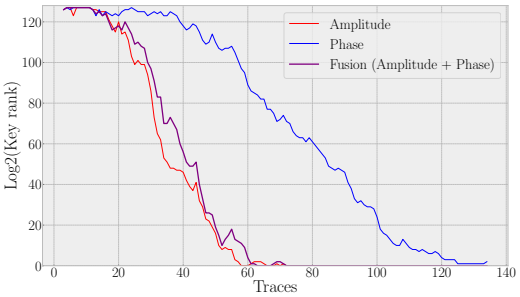


(a) Key rank for the nRF52.

(b) Key rank for the nRF51.



(c) Key rank for the STM32L1.



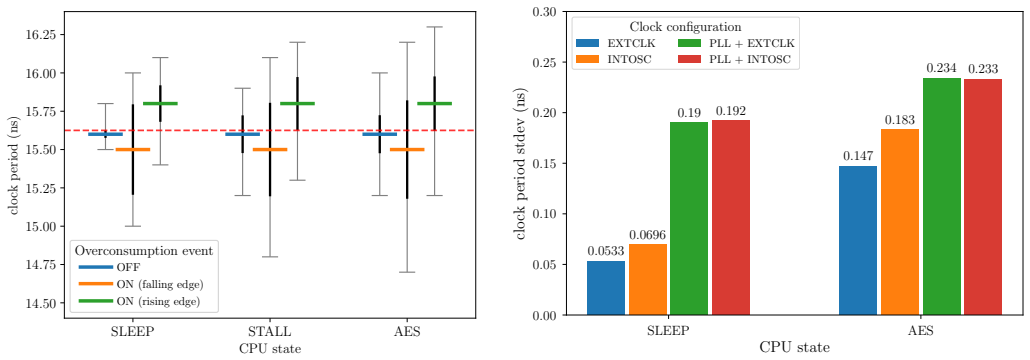
(d) Key rank for the ATmega328

Figure 12: Performance over number of traces for non-profiled attacks. The smaller, the better.

idle to *aes*, with a phase deviation increase of an order of magnitude of 0.1 ns. With the use of the PLL, we also denote a smaller deviation increase of around 0.04 ns from *idle* to *aes*. However, it seems that the sole PLL use naturally adds to most of the clock jitter without the implication of any processor consumption. Overall, we observe that the various internal components contribute to both the natural jitter and the data-dependant one. However, with only the external clock source (EXTCLK), the significant jitter increase from *idle* to *aes* shows that clock circuits are not only at stake, with the hypothesis that the clock tree and paths is potentially another significant source of jitter. We keep detailed investigation of this effect for future work.

Side Observations During near-field experiments, we configured the ChipWhisperer to produce a glitch at fixed intervals. On the phase demodulated output obtained from the near-field measure, we observed periodic phase shifts of significant amplitude appearing at the same fixed interval. When converted to a timing delta using the formula $\phi = 2\pi f \cdot \Delta t$, the measured phase shift corresponds to the jitter observed on the clock line with small incertitude. As an example, when measuring a jitter of 275 ps with the oscilloscope, we measured a phase shift of 0.125 rad, leading to an incertitude of around 0.014 rad or 11%. It highlights the direct relationship between on-chip power consumption and the phase shift observed in the near-field measurements.

During our experiments, we closely observed the clock signal behavior and jitter measurement during overconsumption events. From that observation, we concluded that the period shift seems proportional to the short-circuit load used and, thus, the current intensity. Interestingly, we also noticed that those clock period shifts only occur during the transient effect of a short-circuit, *i.e.*, the rising and falling slopes of the GPIO voltage. Hence, the jitter effect appears to be a derivative function of the chip consumption. This is consistent with the behavior of a voltage regulator, whose voltage output sensibly reacts to a varying load over time.



(a) Clock period under various CPU states and overconsumption events.

(b) Clock period deviation for various clock configurations.

Figure 13: STM32F1 internal clock signal measurement under various CPU states, clock configurations, and overconsumption events.

8 Discussion

Standard Side-Channel Countermeasures Masking [Sha79, PR13] mix the secret with internal random values to mask intermediate values, while hiding [LH20] involves controlling execution time and power consumption to appear random or constant. External shielding [WIX+21] of the SoCs to prevent the measure of its EMR will not be effective against a motivated attacker, which could simply remove the shield.

Jitter-Specific Countermeasures Suppose that a designer inserts a re-synchronizer circuit in the clock tree to suppress the data-dependent jitter to output a clean clock signal. First, with our attack, which measures EMR, we would still be able to measure the jitter (*i.e.*, the phase shift) of the clock signal that is not cleaned because it will radiate in the near-field (NF) before reaching the re-synchronizer circuit. Second, this assumes that the inserted element itself will not be coupled with the data by its power supply. For example, inserting a PLL to suppress the jitter will add another data-dependent jitter if its VCO is coupled to the data *via* the power supply. This circular problem may be solved with proper and perfect isolation between data processing and any clock generation circuitry. *Yu et al.* [YK18] evaluated countermeasures at the voltage regulator level. By masking the data-dependent leakage by performing voltage and frequency scaling, they reduced the correlation coefficient up to only 80% for a DPA. Therefore, this seems to be a hard problem due to phenomenons like, but not limited to, substrate coupling [Par09], for which we do not have proper countermeasures.

Side Channel using Phase Shift While our work focuses on a TinyAES software implementation, our observations lead us to consider that the phase shift leakage is directly related to power consumption. Additional observations also allow us to note a significant impact of a hardware implementation of AES on the phase shift, suggesting a promising direction for developing new exploitation techniques targeting hardware accelerators, which are more and more common in recent microcontrollers. Therefore, our experiments strongly suggest that the problem is not confined to a specific algorithm or implementation. Future work may focus on demonstrating that more targets of power side-channel attacks are also vulnerable to phase shift analysis, replicating our results on a larger set of SoCs would demonstrate the widespread nature of this leakage. Let us note that our approach to compute phase shift traces allows us to manipulate them similarly to standard amplitude

traces, which allows applying pre-processing from prior work (*e.g.*, normalization, alignment using cross-correlation, averaging to reduce noise).

Experimental Reproduction Setup Limitations We acknowledge some limitations regarding the proposed experimental setup to reproduce the phase shift. Due to the data throughput limitations of the oscilloscope, we had to rely solely on its built-in statistics computation. This limited us to the mean and standard deviation information for a given measurement set, which does not allow us to get quartiles and outliers. A more flexible setup may allow us to compute better statistics when characterizing the phenomenon.

Additional Jitter Hypothesis It is known from the Electromagnetic Compatibility (EMC) literature that capacitive coupling between two digital lines (a culprit and a victim) can induce propagation delay on the victim, dependent on the culprit activity [SR92, NP08]. In our case, this crosstalk-induced delay is another hypothesis for the jitter source (and hence, the phase shift). Moreover, part of the data-dependent jitter might also be due to impedance variation inside the target device, as exploited in previous work [MMT23, KFKH23, MMST23]. However, further research needs to be conducted to establish such a relation between our observations and this phenomenon.

9 Conclusion

In this paper, we demonstrated for the first time an EM side-channel attack exploiting unintentional phase modulation instead of amplitude modulation. We were able to conduct this attack without a galvanic connection to the target by measuring the signal using an SDR and a near-field probe. Our attack allows us to successfully recover full AES keys only using the phase of the signal, highlighting the relevance of the signal phase for conducting side-channel attacks. Moreover, we applied a fusion attack technique, allowing us to combine both amplitude and phase information to significantly improve the attack performance. This approach outperforms amplitude-only attacks without additional hardware, resulting in an improvement by an order of magnitude of 40 in the best scenario. Indeed, with our approach, fewer traces are required to conduct an attack or attacks are more accurate for a given number of traces. Moreover, by evaluating the leakage on five popular off-the-shelf SoCs, we identified that four out of five suffer from unintended phase modulation and were able to exploit this unintended modulation to conduct a successful key recovery. Finally, we investigated the root causes of this phenomenon by designing an experimental setup to reproduce the physical problem inducing the phase leakage. We highlighted the relationship between data-dependent jitter on clock signals and EM phase leakage caused by variations in the power consumption correlated to the processor activity. Based on our experiments and a comprehensive literature review, we filled the gap between the early hypothesis about this leakage introduced twenty years ago and the recent timing-based jitter side-channel attacks, paving the way for a better understanding of the phenomenon and the development of new offensive techniques.

Acknowledgements

This work was supported in part by the French National Agency for Research (ANR) grant 18-CE39-0019 (MobiS5) and ANR-22-PECY-0009 (France 2030 project PEPR REV) and in part by the European Union under grant agreement no. 101070008 (ORSHIN project). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

References

[AARR02] Dakshi Agrawal, Bruce Archambeault, Josyula R. Rao, and Pankaj Rohatgi. The EM Side-Channel(s). In *Revised Papers from the 4th International Workshop on Cryptographic Hardware and Embedded Systems*, CHES '02, page 29–45, Berlin, Heidelberg, 2002. Springer-Verlag.

[AARR03] Dakshi Agrawal, Bruce Archambeault, Josyula Rao, and Pankaj Rohatgi. The EM Side-Channel(s): Attacks and Assessment Methodologies. Technical report, IBM, 2003.

[Age72] National Security Agency. TEMPEST: A Signal Problem. Technical report, NSA, 1972.

[ARR03] Dakshi Agrawal, Josyula R. Rao, and Pankaj Rohatgi. Multi-Channel Attacks. In *Workshop on Cryptographic Hardware and Embedded Systems*, 2003.

[BCO04] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004*, pages 16–29, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[Beh07] Arya Behzad. *Wireless LAN Radios*. Wiley, May 2007.

[Ber05] Daniel J. Bernstein. Cache-timing attacks on AES. 2005.

[BOFM02] N. Barton, D. Ozis, T. Fiez, and K. Mayaram. The Effect of Supply and Substrate Noise on Jitter in Ring Oscillators. In *Proceedings of the IEEE 2002 Custom Integrated Circuits Conference (Cat. No.02CH37285)*, pages 505–508, 2002.

[BPRT19] E. Balestrieri, F. Picariello, S. Rapuano, and I. Tudosa. Review on Jitter Terminology and Definitions. *Measurement*, 145:264–273, 2019.

[CFS20] Giovanni Camurati, Aurélien Francillon, and François-Xavier Standaert. Understanding Screaming Channels: From a Detailed Analysis to Improved Attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems (CHES 2020)*, 2020(3):358–401, June 2020.

[CPM+18] Giovanni Camurati, Sebastian Poeplau, Marius Muench, Tom Hayes, and Aurélien Francillon. Screaming Channels: When Electromagnetic Side Channels Meet Radio Transceivers. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, page 163–177, New York, NY, USA, 2018. Association for Computing Machinery.

[CRR02] Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi. Template Attacks. In *Workshop on Cryptographic Hardware and Embedded Systems*, 2002.

[DDS] Nicola Da Dalt and Ali Sheikholeslami. *Understanding Jitter and Phase Noise: A Circuits and Systems Perspective*. Cambridge University Press, 1 edition.

[FH08] Julie Ferrigno and Martin Hlavác. When AES Blinks: Introducing Optical Side Channel. *IET Inf. Secur.*, 2(3):94–98, 2008.

[Fre16] Louis E. (Jr.) Frenzel. *Principles of Electronic Communication Systems*. 4th edition, 2016.

[GDTM21] Joseph Gravellier, Jean-Max Dutertre, Yannick Teglia, and Philippe Loubet Moundi. SideLine: How Delay-Lines (May) Leak Secrets from Your SoC. In Shivam Bhasin and Fabrizio De Santis, editors, *Constructive Side-Channel Analysis and Secure Design*, pages 3–30, Cham, 2021. Springer International Publishing.

[GMGH19] Christophe Genevey-Metat, Benoît Gérard, and Annelie Heuser. Combining Sources of Side-Channel Information. In *C&ESAR*, 2019.

[GPPT22] Daniel Genkin, Lev Pachmanov, Itamar Pipman, and Eran Tromer. Stealing Keys from PCs Using a Radio: Cheap Electromagnetic Attacks on Windowed Exponentiation. In *Cryptographic Hardware and Embedded Systems – CHES 2015*, page 207–228, Berlin, Heidelberg, 2022. Springer-Verlag.

[GST17] Daniel Genkin, Adi Shamir, and Eran Tromer. Acoustic Cryptanalysis. *J. Cryptology*, 30:392–443, 2017.

[Han] Johnnie Hancock. Jitter—Understanding It, Measuring It, Eliminating It Part 1: Jitter Fundamentals.

[HH15] Paul Horowitz and Winfield Hill. *The Art of Electronics*. Cambridge University Press, March 2015.

[HMW+22] Wangwang Huang, Xuesong Mei, Yage Wang, Zhengjie Fan, Cheng Chen, and Gedong Jiang. Two-Dimensional Phase Unwrapping by a High-Resolution Deep Learning Network. *Measurement*, 200:111566, 2022.

[HP] P. Heydari and M. Pedram. Jitter-Induced Power/Ground Noise in CMOS PLLs: a Design Perspective. In *Proceedings 2001 IEEE International Conference on Computer Design: VLSI in Computers and Processors. ICCD 2001*, pages 209–213. IEEE Comput. Soc.

[HP00] P. Heydari and M. Pedram. Analysis of Jitter due to Power-supply Noise in Phase-Locked Loops. In *Proceedings of the IEEE 2000 Custom Integrated Circuits Conference (Cat. No.00CH37044)*, pages 443–446, 2000.

[HT03] David Howe and T Tasset. Clock Jitter Estimation based on PM Noise Measurements. In *Proceedings of the 2003 IEEE International Frequency Control Symposium and PDA Exhibition*. 2003 Joint Mtg. IEEE Intl. Freq. Cont. Symp. and EFTF Conf , Tampa, FL, 2003-01-01 2003.

[JG] Howard W. Johnson and Martin Graham. *High-Speed Digital Design*.

[Key] Keysight. Measuring Phase Noise with a Real Time Sampling Oscilloscope. https://docs.keysight.com/kkbopen/measuring-phase-noise-with-a-real-time-sampling-oscilloscope-584447063.html.

[KFKH23] Shugo Kaji, Daisuke Fujimoto, Masahiro Kinugawa, and Yuichi Hayashi. Echo tempest: Em information leakage induced by iemi for electronic devices. *IEEE Transactions on Electromagnetic Compatibility*, 65(3):655–666, June 2023.

[KJJ99] Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In Michael Wiener, editor, *Advances in Cryptology — CRYPTO' 99*, pages 388–397, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[KJJR11] Paul C. Kocher, Joshua Jaffe, Benjamin Jun, and Pankaj Rohatgi. Introduction to Differential Power Analysis. *Journal of Cryptographic Engineering*, 1:5–27, 2011.

[Koc96] Paul C. Kocher. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '96, page 104–113, Berlin, Heidelberg, 1996. Springer-Verlag.

[Kok] Kokke. TinyAES: Small portable AES128/192/256 in C. https://github.com/kokke/tiny-AES-c.

[Lee] Bang S Lee. Understanding the Terms and Definitions of LDO Voltage Regulators.

[LGG+21] Corentin Lavaud, Robin Gerzaguet, Matthieu Gautier, Olivier Berder, Erwan Nogues, and Stéphane Molton. Whispering Devices: A Survey on How Side-Channels Lead to Compromised Information. *Journal of Hardware and Systems Security*, 5:143 – 168, 2021.

[LH20] JongHyeok Lee and Dong-Guk Han. Security Analysis on Dummy Based Side-Channel Countermeasures—Case Study: AES with Dummy and Shuffling. *Applied Soft Computing*, 93:106352, 2020.

[Li18] Fanlong Li. Impact of Power-Supply Noise on Phase Noise Performance of RF DACs. Technical report, Texas Instruments, 2018.

[LMM05] Huiyun Li, A. Theodore Markettos, and Simon Moore. Security Evaluation Against Electromagnetic Analysis at Design Time. In *Proceedings of the 7th International Conference on Cryptographic Hardware and Embedded Systems*, CHES'05, page 280–292. Springer-Verlag, 2005.

[Lyo08] Richard Lyons. Quadrature Signals: Complex, But Not Complicated. https://dspguru.com/files/QuadSignals.pdf, 2008.

[Man03] Stefan Mangard. A Simple Power-Analysis (SPA) Attack on Implementations of the AES Key Expansion. In Pil Joong Lee and Chae Hoon Lim, editors, *Information Security and Cryptology — ICISC 2002*, pages 343–358, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[Mey12] Olivier Meynard. *Characterization and Use of the EM Radiation to Enhance Side Channel Attacks*. Theses, Télécom ParisTech, January 2012.

[Mic] Microchip. *megaAVR Data Sheet: ATmega48A / PA / 88A / PA / 168A / PA / 328 / P*.

[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, Issue 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*, CCS '23, page 1675–1689. Association for Computing Machinery, 2023.

[Moh11] Habeeb Ur Rahman Mohammed. Supply Noise Effect on Oscillator Phase Noise. Technical report, Texas Instruments, 2011.

[MOW14] Luke Mather, Elisabeth Oswald, and Carolyn Whitnall. Multi-Target DPA Attacks: Pushing DPA Beyond the Limits of a Desktop Computer. In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology – ASIACRYPT 2014*, pages 243–261, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[MSY13] Jim Monthie, Vineet Sreekumar, and Ranjit Yashwante. Impact of Power Supply Noise on Clock Jitter in High-Speed DDR Memory Interfaces. In *2013 26th International Conference on VLSI Design and 2013 12th International Conference on Embedded Systems*, pages 262–266, 2013.

[New] NewAE. CW1173 ChipWhisperer-Lite - NewAE Hardware Product Documentation. https://rtfm.newae.com/Capture/ChipWhisperer-Lite/.

[NP08] Shahin Nazarian and Massoud Pedram. Crosstalk-affected delay analysis in nanometer technologies. *International Journal of Electronics*, 95(9):903–937, 2008.

[OEMG+20] Maamar Ouladj, Nadia El Mrabet, Sylvain Guilley, Philippe Guillot, and Gilles Millerioux. On The Power of Template Attacks in Highly Multivariate Context. *Journal of Cryptographic Engineering*, 10, 11 2020.

[O'F24] Colin O'Flynn. Phase Modulation Side Channels: Jittery JTAG for On-Chip Voltage Measurements. *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, 2024(4):382–424, Sep. 2024.

[OGOP04] S.B. Ors, F. Gurkaynak, E. Oswald, and B. Preneel. Power-Analysis Attack on an ASIC AES Implementation. In *International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004.*, volume 2, pages 546–552 Vol.2, 2004.

[oSN01] National Institute of Standards and Technology (NIST). *Advanced Encryption Standard (AES)*, November 2001.

[Par09] Raj S. Parihar. Substrate Coupling Noise: Modeling and Mitigation Techniques. Technical report, University of Rochester, 2009.

[PDY16] Hoda Pahlevanzadeh, Jaya Dofe, and Qiaoyan Yu. Assessing CPA Resistance of AES with Different Fault Tolerance Mechanisms. In *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 661–666, 2016.

[Pi] Raspberry Pi. *RP2040 Datasheet, A Microcontroller by Raspberry Pi.*

[PR13] Emmanuel Prouff and Matthieu Rivain. Masking Against Side-Channel Attacks: A Formal Security Proof. In *Advances in Cryptology - EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 142–159. Springer, 2013.

[Put18] Sadasivan Puthusserypady. Complex Signals. http://bme.elektro.dtu.dk/31610/notes/complex.signals.pdf, 2018.

[QS01] Jean-Jacques Quisquater and David Samyde. ElectroMagnetic Analysis (EMA): Measures and Counter-Measures for Smart Cards. In *Proceedings of the International Conference on Research in Smart Cards: Smart Card Programming and Security*, E-SMART '01, page 200–210, Berlin, Heidelberg, 2001. Springer-Verlag.

[Ros82] Howard E. Rosenblum. NACSIM 5000: Tempest Fundamentals. Technical report, National Security Agency (NSA), 1982.

[SAHW90] Donald B. Sullivan, David W. Allan, David A. Howe, and Fred L. Walls. Characterization of Clocks and Oscillators. Technical report, March 1990.

[SDR24] SDRplay. *RSPdx: Multi-Antenna Port 14-bit SDR*, 2024.

[Sem13] Nordic Semiconductor. *nRF51422 Product Specification*, 2013.

[Sem21] Nordic Semiconductor. *nRF52832 Product Specification*, 2021.

[Sha79] Adi Shamir. How to Share a Secret. *Commun. ACM*, 22(11):612–613, nov 1979.

[Sig] Siglent. SDS 2000X-Plus Datasheet. https://siglentna.com/wp-content/uploads/dlm_uploads/2021/03/SDS2000X-Plus_Datasheet_-DS0102XP_E01B.pdf.

[SMTG23] Kai Schoos, Sergej Meschkov, Mehdi B. Tahoori, and Dennis R. E. Gnad. JitSCA: Jitter-based Side-Channel Analysis in Picoscale Resolution. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2023(3):294–320, Jun. 2023.

[SR92] E. Sicard and A. Rubio. Analysis of crosstalk interference in cmos integrated circuits. *IEEE Transactions on Electromagnetic Compatibility*, 34(2):124–129, 1992.

[STMa] STMicroelectronics. *Datasheet - STM32F103xC, STM32F103xD, STM32F103xE.*

[STMb] STMicroelectronics. *Datasheet - STM32L151xE STM32L152xE.*

[STMc] STMicroelectronics. *STM32 Nucleo-64 Development Board with STM32F103RB MCU.*

[Teka] TekBox. *TBWA2: Wideband RF Amplifiers.*

[Tekb] Tektronix. Jitter Analysis: A Brief Guide to Jitter. https://anlage.umd.edu/Microwave

[Tek24] TekBox. Tbps01 probe, 2024.

[UMS] Un-Ku Moon, K. Mayaram, and J.T. Stonick. Spectral Analysis of Time-Domain Phase Jitter Measurements. 49(5):321–327.

[VCGS13] Nicolas Veyrat-Charvillon, Benoît Gérard, and François-Xavier Standaert. Security Evaluations Beyond Computing Power. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology – EUROCRYPT 2013*, pages 126–141, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[VE85] Wim Van Eck. Electromagnetic Radiation from Video Display Units: An Eavesdropping Risk? *North-Holland Computers & Security*, pages 269–286, 1985.

[VP09] Martin Vuagnoux and Sylvain Pasini. Compromising Electromagnetic Emanations of Wired and Wireless Keyboards. In *Proceedings of the 18th Conference on USENIX Security Symposium*, SSYM'09, page 1–16. USENIX Association, 2009.

[Wit] Marc Witteman. Security Highlight: You May Be Leaking Secrets if You Don't Keep Your Pace.

[WIX+21] Meizhi Wang, Vishnuvardhan V. Iyer, Shanshan Xie, Ge Li, Sanu K. Mathew, Raghavan Kumar, Michael Orshansky, Ali E. Yilmaz, and Jaydeep P. Kulkarni. Physical Design Strategies for Mitigating Fine-Grained Electromagnetic Side-Channel Attacks. In *2021 IEEE Custom Integrated Circuits Conference (CICC)*, pages 1–2, 2021.

[WWD20] Ruize Wang, Huanyu Wang, and Elena Dubrova. Far Field EM Side-Channel Attack on AES Using Deep Learning. *Proceedings of the 4th ACM Workshop on Attacks and Solutions in Hardware Security*, Nov 2020.

[YK18] Weize Yu and Selçuk Köse. Exploiting Voltage Regulators to Enhance Various Power Attack Countermeasures. *IEEE Transactions on Emerging Topics in Computing*, 6(2):244–257, 2018.

[YZC+17] Wei Yang, Yongbin Zhou, Yuchen Cao, Hailong Zhang, Qian Zhang, and Huan Wang. Multi-Channel Fusion Attacks. *Trans. Info. For. Sec.*, 12(8):1757–1771, aug 2017.

[ZF05] YongBin Zhou and DengGuo Feng. Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing, 2005.