The impact of pulsed Electromagnetic Fault Injection on true random number generators

Maxime Madau* , Michel Agoyan* , Josep Balasch†, Miloš Grujić† , Patrick Haddad* , Philippe Maurine‡
Vladimir Rožić† , Dave Singelée† , Bohan Yang† , Ingrid Verbauwhede†

* STMicroelectronics, Rousset France
email: *firstname.name@st.com*
† COSIC, KU Leuven, Leuven Belgium
email: *firstname.lastname@esat.kuleuven.be*
‡ Laboratoire d'Informatique, de Robotique et de Microlectronique de Montpellier, Montpellier France
firstname.lastname@lirmm.fr

Abstract—**Random number generation is a key function of today's secure devices. Commonly used for key generation, random number streams are more and more frequently used as the anchor of trust of several countermeasures such as masking. True Random Number Generators (*TRNGs*) thus become a relevant entry point for attacks that aim at lowering the security of integrated systems. Within this context, this paper investigates the robustness of TRNGs based on Ring Oscillators (focusing on the delay chain TRNG) against pulsed electromagnetic fault injection. Indeed, weaknesses in generating random bits for masking scheme degenerate the Side Channel resistance. Finally by exploiting fault results on delay chain TRNG some general guidelines to harden them are derived.**

keywords: ***Pulse Electromagnetic fault Injection, True Random Number Generator, security guidelines***

I. Introduction

Random Number Generation (*RNG*) is a key embedded function of modern secure components which mainly relies, in the specific field of security, on hardware True Random Number Generators (*TRNG*). As random numbers are required to generate the secure cryptographic keys needed by authentication protocols, RNGs are commonly at the heart of countermeasures to protect against Side Channels Attacks (*SCA*) such as masking. Because of their crucial role in secure systems, TRNGs are thus a target of choice for attackers / evaluators.

TRNG relies on physical non deterministic phenomena such as thermal noise or metastability to provide unpredictable numbers. Among the former, some extract entropy from the internal jitter of free-running Ring Oscillators (*RO*) such as [1]. The Delay Chain True Random Number Generator (*DC-TRNG*) [2], [3] is one of these TRNGs.

DC-TRNG extracts random jitter by sampling the position in a delay chain of the output rising edge of free-running RO (i.e. the entropy source). To enhance the statistical properties of the obtained random number flow, two decimators are cascaded at the output of this entropy extractor.

One major threat against secure devices is Fault Injection (*FI*). Different tools can be employed to induce transient faults in Integrated Circuits (*IC*), each of them having its drawbacks and assets. Among them ElectroMagnetic Fault Injection (*EMFI*) offers a good trade-off in terms of spatial and temporal resolutions, ease of use (reduced preparation of targets) and costs.

This threat was first highlighted in 2002 in [4] in which the authors demonstrated they were able to corrupt the content of an embedded memory using an external EM field. Later an EM Pulse (*EMP*) induced by a lightning spark was used to inject faults into a CRT-RSA [5].

There are two types of EMFI platforms. On one hand, harmonic platforms deliver powerful continuous EM waves to disrupt ICs' behavior; especially the behavior of analog blocks. On the other hand, pulsed EMFI platforms deliver a short and strong variation of the magnetic field in the close vicinity of ICs to disrupt their computation. The common understanding about the effect of an EM pulse on ICs is that it creates a sudden current flow in the IC's power / ground networks. This parasitic current induces voltage drops and ground bounces, disrupting IC's operation. The efficiency of this type of EMFI platform in injecting exploitable faults (in symmetric cryptograpghy blocks) was first reported in [6] where faults were injected into an old microcontroller (designed with 350nm technology).

A first contribution of this paper is to evaluate the threat constituted by pulsed EMFI on a TRNG, namely the DC-TRNG. It is expected to be representative of many others TRNGs because of its typical architecture and of the widely exploited entropy source on which it relies. To the best of our knowledge this is the first paper reporting results about pulsed EMFI on a TRNG; the state-of-the-art EMFI technique to disrupt TRNGs being Harmonic EMFI: [7], [8] and [9]. A second contribution is to derive from the nature of observed faults some general guidelines to harden TRNGs against state-of-the-art pulsed EMFI.

The paper is organized as follows. Section 2 provides an overview of the DC-TRNG architecture and its operation before defining the threat model (attack path) considered in the rest of the paper. Section 3 successively describes

978-1-5386-8197-8/18/$31.00 ©2018 IEEE
DOI 10.1109/FDTC.2018.00015

the experimental setup and the obtained experimental results before discussing their meaning. Finally some guidelines to harden TRNGs are proposed in section 4 before concluding in section 5.

II. Case study

A. DC-TRNG architecture

The DC-TRNG was first proposed by Rožić *et al.* [2] in 2015. A refined stochastic model of it accounting the nonlinearity of the delay-chain was published in [3].

The entropy of the DC-TRNG comes from the timing jitter of free-running ring oscillators. The DC-TRNG takes advantage of the high-speed carry-chain primitives, which are designed to improve the operation speeds of additions and multiplications on Field Programmable Gate Array (*FPGA*). The carry-chain primitives are widely available on most FPGA devices from different vendors. They are cascaded together to form a delay-chain used for sampling the jittery signal with a high resolution. This architecture enables an efficient entropy extraction without the requirement of a high jitter accumulation time, which leads to a high-throughput TRNG.

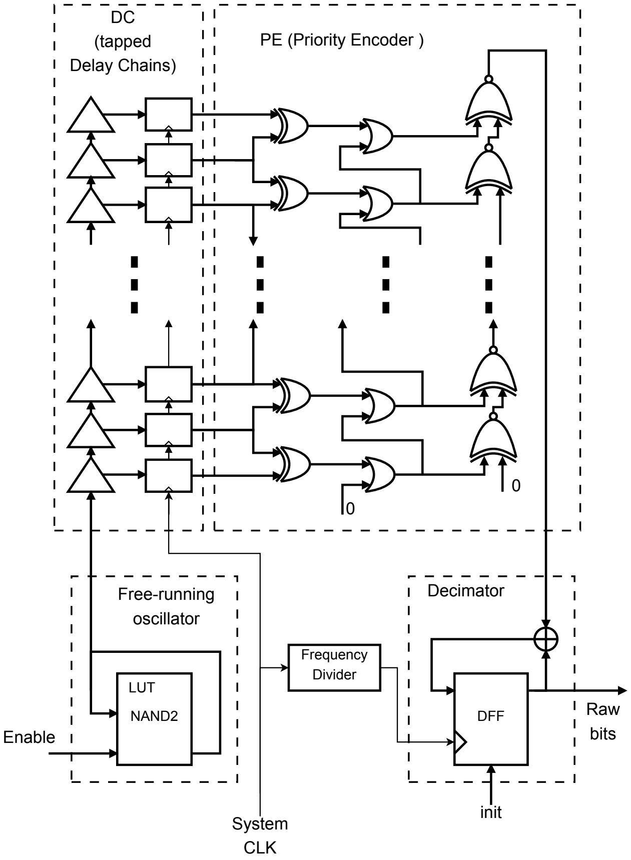

Fig. 1: Implementation of the digital noise source of DC-TRNG.

An implementation of the DC-TRNG is shown in Fig. 1. A tapped Delay Chain (*DC*), a Priority Encoder (*PE*) and a decimator comprise the digitization module. After a sufficient jitter accumulation period, the timing phase of the signal becomes unpredictable due to the accumulated Gaussian noise. This jittered signal coming from the RO propagates through the DC and is sampled by D flip-flops, each of which is attached to a delay element. Using the PE, the position of the signal edge in the tapped DC is encoded to extract a single raw bit. The PE filters the "bubbles" in the code that can be created due to the intrinsic architecture of the carry primitives or/and violating the timing conditions of the flip-flops. The least significant bit of the position is generated as the output of the PE. To mitigate the impact of the low frequency noise, the parity of the distance between two consecutive sample positions is calculated by a decimator to generate a raw random bit.

B. Threat model

Side-Channel Attacks (*SCA*) exploit the statistical dependence between a physical information leaked from the device and intermediate values processed by the implementation in order to extract secret keys. One of the most deployed countermeasure against SCA is masking [10], [11]. The goal of masking is to eradicate the dependency between the physical leakage of the device and the intermediate values V it processes by splitting them into $d+1$ shares $(m_1, ..., m_{d+1})$ so that $V = m_1 \odot m_2 \odot ...m_{d+1}$, where $\odot$ denotes a group operation. Shares $m_1, ..., m_d$ are usually provided by a random number generator, and the share m_{d+1} is then computed as $m_{d+1} = V \odot m_1 \odot ...m_d$. From the above, it is clear that the security of the masking countermeasure directly relies on the randomness of the numbers provided by the embedded TRNGs.

In the context of this work, we explore security degradation of the 1st-order Boolean masking scheme [11] when supplied with non-ideal random numbers. This can occur in scenarios when random numbers are obtained from an attacked TRNG.

As a metric of security degradation we are using the minimum number of measurements needed for a univariate Correlation Power Analysis (*CPA*) attack to succeed. We opt to emulate a straightforward attack scenario on the AES Sbox output, and we use simulations based on the Hamming weight leakage model and additive Gaussian noise. The analysis results of the 1st-order Boolean masking scheme supplied with biased random numbers are shown in Fig.2. The curves in this figure illustrate the minimum number of measurements needed for successful CPA attack as a function of different noise levels σ and different levels of bias. In the case of Boolean masking, both types of biases (towards 0 and towards 1) have the same effect on security degradation. We observe that for low noise levels and a bias of 25%, the number of measurements needed for successful attack is only 225. On the other hand, for high noise levels and bias of only 5%, the attack can still be successfully mounted, but now requires approximately 650,000 measurements.

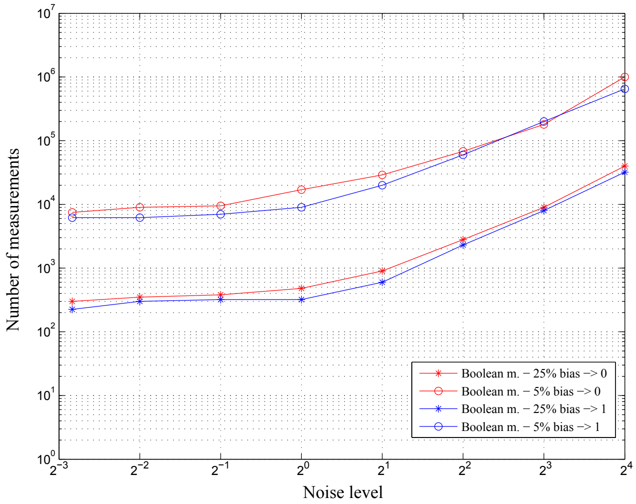

Fig. 2: Univariate *CPA* attacks against AES Sbox protected by 1st-order Boolean masking scheme.

In the case of the Inner Product (*IP*) masking scheme, the attack's success depends on the type of bias - the attack is more successful if the random numbers are biased towards 0. However, IP masking is more resilient than Boolean masking - the minimum number of measurements that enable the attack for low noise levels and a bias of 25% is 22,000. Furthermore, the attack is only successful for a bias level of at least 10%, when it requires approximately 1,250,000 measurements.

We observe that in the case of Boolean masking, both types of biases - towards 0 and towards 1, have the same effect on security degradation, i.e. the curves have very similar shapes. Furthermore, for low noise levels and a bias of 25% towards 1, the number of measurements needed for successful attack is only 225. On the other hand, for high noise levels and bias of only 5% towards 1, the attack can still be successfully mounted, but now requires approximately 650,000 measurements.

From the above analysis, two scenarios can be identified. Firstly, the attacker is able to fully control the random bits flow by disrupting directly the entropy source or its post-processing blocks (the *PE* and the *DC* chain in our case). In such a case, the attacker is thus able to craft a sequence of numbers passing the embedded statistical tests and to insert it once or several times in the random numbers flow. Secondly, the attacker is able to introduce a small bias of at least 10% to be able to successfully run a CPA.

III. Experimental Results and design conclusion

A. Experimentation

A DC-TRNG composed of ten carry chains and two decimation stages in cascade was implemented into a Xilinx Spartan-6 FPGA and operates at a clock frequency of $4.5MHz$. The entropy source was instantiated as one Lookup Table (*LUT*) and the delay chain is composed of CARRY4 primitives.

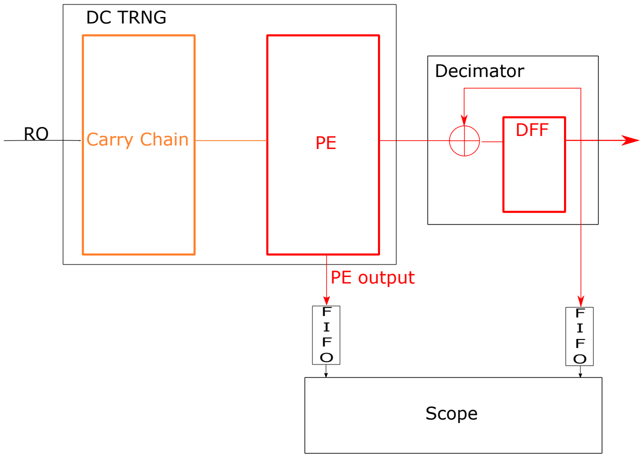

Fig. 3: FPGA design for experiment

The EMFI platform is composed of a pulse generator with a voltage amplitude in the range of $0V$ to $400V$ and a pulse width from $8ns$ to $100ns$. The pulse generator can shoot at a frequency of $2kHz$. The experimental campaigns were run with two different EM probes: an U-shape probe and a cylindrical probe with a flat end (Fig. 4). Probes are inductors made from a ferrite core with a various number of wire coil wound around. The research of the injection probe position leading to fault was done with the flat ended probe. Then in order to produce more localized faults (faults disrupting only one functional block of the DC-TRNG), the U-shape probe was used. Thus faults reported in the tables were obtained with the U-shape probe.

The DC-TRNG tests were designed to provide an insight into the internal behavior of its components (namely PE and DC) as well as computation involve in the random stream generation, i.e. decimator stage. Moreover across all the different tests we ran the PE, DC and RO were placed at the same position. As a first test, the whole design was considered, to monitor PE and Decimator's streams their output were sent directly to the scope alongside their sampling clocks (which was used as a trigger signal for pulse EM generation). Then to be able to separate real faults on TRNG to side effect on the design induced by EMFI, output of monitored signals were delayed by several clock cycles using FIFOs as depicted in Fig. 3. Therefore a fault occurring on the DC-TRNG would appear some clock cycles (equal to the length of the FIFO) after pulse generation as illustrated on Fig. 5 and Fig. 6. By contrast perturbation occurring on IO bonding wires or on our FIFO would appear either instantly or with a delay lower than the total length of the FIFO.

Detecting the occurrence of faults in the numbers stream provided by a TRNG is not straightforward. The way we proceeded is as follows: the DC-TRNG was launched one thousand times for each considered EMFI probe positions and injection time. Then the vectors of binary values collected at the output of the PE (synchronously with the clock) were

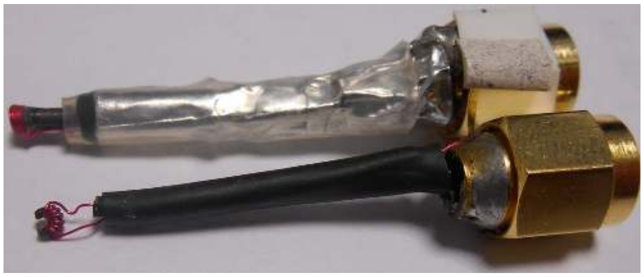

Fig. 4: EMFI Probes used for characterization

averaged so that to detect the occurrence of stuck-at faults. Since we used AC coupling, the expected behavior is an average trace close to 0, i.e. with an equal number of 1 and 0. Otherwise if a stuck-at 0 fault is systematically induced at a given position and at a given time, a negative extremum will appear (resp. a positive extremum will appear for a stuck-at 1 fault). This procedure is illustrated Fig. 5 in which the orange trace indicates the average vector. On this curves we can see that EMFI produces a systematic stuck at '0' fault at time samples 25,000.

Using the above fault detection procedure, the impact of EMFI was analyzed at the output of both PE and Decimator. Tables II indicates the pulse parameters required to obtained certain types of faults at a specific position and at any time instants. Being able to draw such tables is a direct illustration that an attacker can fully, but temporarily, control the output of TRNGs. This partially sustains the threat model introduced in the preceding section.

Fault types	Amplitude	Pulse width	Delay
stuck at 0:	171V	12.6ns	1.12ns
stuck at 1:	179V	12.6ns	1.12ns

TABLE I: Decimator fault result

Fault types	Amplitude	Pulse width
stuck at 0:	292V	6.45ns
stuck at 1:	274V	6ns
stuck at 00:	356V	7.7ns
stuck at 01:	356V	7.4ns
stuck at 10:	314V	11ns
stuck at 11:	350V	11ns

TABLE II: PE Fault result

Table II gives the required settings of the EM pulse to obtain different types of faults at the PE's output. However, at that point, for these settings, it is impossible to state if faults occur in the PE or in the carry-chain. The content of the carry-chain was thus monitored in a last experiment. This last experiment focus on the effect of pulsed EMFI on the DC and PE, thence their stream were output to a serial port. Since this test enable to get DC and PE values, PE's outputs were re-computed off-line using the obtained DC values to decide (by comparison) if a fault occurs in the DC or in the PE.

This last experiment demonstrated that EMFI was able to induce stuck at faults in the PE for some probe positions without any effect on the carry chain, but also to disrupt the carry-chain's content for other positions. However, the fault induced in the carry-chain were uncontrollable because of the asynchronous operation with regards to the master clock of the free-running RO. Indeed, instead of having a vector like ...0000011111... that corresponds to the correct operation of the DC-TRNG, we obtained vectors like ...01...10..11...0..10110...1111 but with the position of the first '1' changing at each run.

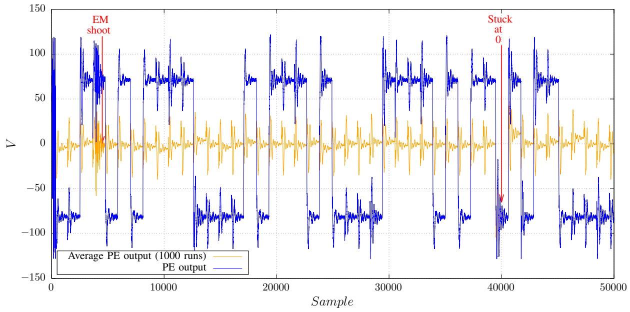

Fig. 5: The blue curves is the output of one run of the PE and the orange curves is the mean value taken on 1000 curves (on normal behavior 0 is expected). Beware the scope is in AC coupling explaining the expected means of 0 for normal behavior.

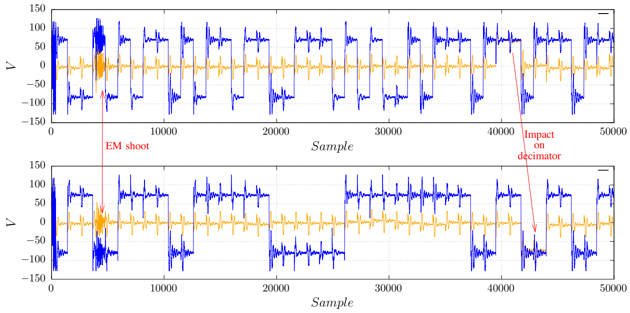

Fig. 6: On the top graph the blue curve is the output of one run of the PE and the orange curve is the mean value taken on 1000 curves (on normal behavior 0 is expected). For the bottom graph the color code is the same but it is the decimator output. Beware the scope is in AC coupling explaining the expected means of 0 for normal behavior.

To conclude, all these experiments validates a fault model stating that pulse EM can stick one bit at a specific value. This fault model applies on both the PE or the decimator stages and has a bit accuracy. Yet in the particular case of the PE we were able to stuck-at a specific value up to two bits using one injection as illustrated in Fig. 6 with the example of stucking two bits at 1. Since the fault impacts two bits of PE's output we can see its direct effect on the decimator stage.

B. Design guidelines

These experiments highlight the fact that entropy source is not the only entry point to induce bias in the random numbers flow delivered by a TRNG. Still today's strategy to harden TRNGs mainly consists in monitoring the statistical properties of the entropy sources. Nonetheless, when dealing with pulsed EMFI, exploitable faults seems to be more easily induced in digital post-processing stages than in the entropy source. Therefore it seems mandatory to also protect these processing blocks.

The above results show that with a single EM injection an attacker is only able to create a signle bit stuck-at (at best two consecutive bits) at a fixed and controllable value. This means that controlling the whole bit flow requires the use of EMFI platform with a repetition rate at least equal to the throughput of the targeted TRNGs. Thus, high throughput is a desirable feature for TRNG to counter pulsed EMFI. Indeed modern EMFI platforms are limited to only a few kHz.

Yet controlling the whole random bit flow is not the only threat. Indeed, as pointed out in Section 2, a bias level of at least 10% is necessary to lead a successful CPA against a masked implementation. Indeed, controlling the whole random stream is not required to be able to lead successful CPA against a masked implementation. Instead, in the case of CPA the bias introduce by pulsed EMFI on the TRNG should also be monitored. As pointed out in Section 2, a bias level of at least 10% is required to lead a successful CPA against IP masking scheme while a bias level of 5% is required in the case of Boolean masking. To measure the criticity of modern pulse EMFI platforms against masking scheme relying on RO based TRNGs we computed the bias induced for different values of DCTRNG's operating frequency and different induced fault types. Moreover we consider our EMFI platform performing at full speed, i.e. $500\mu s$. From the results reported in Fig. 7 it is clear that unless a DC-TRNG running with a frequency lower than $170kHz$, pulsed EMFI does not constitute a major threat (as a reminder during the experiment part the DC-TRNG was running at $4.5MHz$). However, there is room to increase the repetition rate of EMFI platforms and adding EMFI detectors [12], [13], [14] to protect TRNGs should not be viewed as a luxury solution in the future.

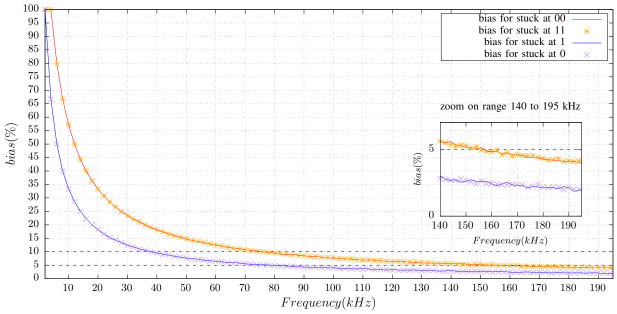

Fig. 7: bias on the TRNG random output agains functionning frequency

IV. Conclusion

Security characterization of a TRNG under pulse EMFI has been presented in this paper. The TRNG used as a case of study is the DC-TRNG which relies on free-running oscillators as entropy source. This analysis has given insights about the effects of EM pulses against this family of devices and hence clues on how to mitigate them. A fault model has been derived from experimental results showing that EM pulse can stick one random bit in controllable manner and that exploitable faults can be induced the digital part of TRNG rather than in the entropy source in which only uncontrollable faults are induced.

Regarding this fault model, two scenarios were considered. One in which the attacker has a full control of the random numbers flow and one in which the adversary can induce a sufficient bias to successfully perform a CPA on masked AES implementations. Fortunately, none of these scenarios is actually possible using modern EMFI platforms. However, there is room for increasing the repetition rate of injections of such platforms and thus in near future protecting the TRNGs in their whole and not solely the entropy source must be envisaged.

Acknowledgments

This work is supported in part by Horizon 2020 research and innovation programme under grant agreement No 644052 HECTOR and Cathedral ERC Advanced Grant 695305.

References

[1] B. Sunar, W. J. Martin, and D. R. Stinson, "A provably secure true random number generator with built-in tolerance to active attacks," *IEEE Transactions on Computers*, vol. 56, 2007.

[2] V. Rozic, B. Yang, W. Dehaene, and I. Verbauwhede, "Highly efficient entropy extraction for true random number generators on fpgas," *Design Automation Conference DAC'15*, 06 2015.

[3] M. Grujic, V. Rozic, B. Yang, and I. Verbauwhede, "A closer look at the delay-chain based trng," *IEEE International Symposium on Circuits and System ISCAS2018*, pp. 1–5, 05 2018.

[4] J. Quisquater and D. Samyde, "Eddy current for magnetic analysis with active sensor," in *Proceedings of ESmart 2002*. Eurosmart, 2002, p. pp 185194.

[5] J.-M. Schmidt and M. Hutter, "Optical and em fault-attacks on crt-based rsa: Concrete results," in *Austrochip 2007, 15th Austrian Workhop on Microelectronics, 11 October 2007, Graz, Austria, Proceedings*, J. W. Karl C. Posch, Ed. Verlag der Technischen Universität Graz, 2007, pp. 61 – 67.

[6] A. Dehbaoui, J.-M. Dutertre, B. Robisson, P. Orsatelli, P. Maurine, and A. Tria, "Injection of transient faults using electromagnetic pulses - practical results on a cryptographic system-," *IACR Cryptology ePrint Archive*, vol. 2012, p. 123, 2012.

[7] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, and P. Maurine, "Contactless electromagnetic active attack on ring oscillator based true random number generator," in *COSADE*, 2012, pp. 151–166.

[8] A. T. Markettos and S. W. Moore, "The frequency injection attack on ring-oscillator-based true random number generators," in *Cryptographic Hardware and Embedded Systems - CHES 2009*, C. Clavier and K. Gaj, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 317–331.

[9] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, and P. Maurine, "Contactless electromagnetic active attack on ring oscillator based true random number generator," in *Constructive Side-Channel Analysis and Secure Design*, W. Schindler and S. A. Huss, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 151–166.

[10] S. Chari, C. S. Jutla, J. R. Rao, and P. Rohatgi, “Towards Sound Approaches to Counteract Power-Analysis Attacks,” in *Advances in Cryptology - CRYPTO ’99*, ser. Lecture Notes in Computer Science, M. J. Wiener, Ed., vol. 1666. Springer, 1999, pp. 398–412.

[11] L. Goubin and J. Patarin, “DES and Differential Power Analysis (The ”Duplication” Method),” in *Cryptographic Hardware and Embedded Systems - CHES’99*, ser. Lecture Notes in Computer Science, Çetin Kaya Koç and C. Paar, Eds., vol. 1717. Springer, 1999, pp. 158–172.

[12] D. El-Baze, J. Rigaud, and P. Maurine, “An embedded digital sensor against EM and BB fault injection,” in *2016 Workshop on Fault Diagnosis and Tolerance in Cryptography, FDTC 2016, Santa Barbara, CA, USA, August 16, 2016*, 2016, pp. 78–86.

[13] L. Zussa, A. Dehbaoui, K. Tobich, J.-M. DUTERTRE, P. Maurine, L. Guillaume-Sage, J. Clédière, and A. Tria, “Efficiency of a glitch detector against electromagnetic fault injection,” in *DATE: Design, Automation and Test in Europe*, ser. Design, Automation and Test in Europe Conference and Exhibition (DATE), 2014. Dresden, Germany: IEEE, Mar. 2014, pp. 1–6. [Online]. Available: https://hal-lirmm.ccsd.cnrs.fr/lirmm-01096047

[14] N. Miura, Z. Najm, W. He, S. Bhasin, X. T. Ngo, M. Nagata, and J.-L. Danger, “Pll to the rescue: a novel em fault countermeasure,” pp. 1–6, 06 2016.