A Fully-Digital EM Pulse Detector

David El-Baze*, Jean-Baptiste Rigaud*, Philippe Maurine†
*Mines Saint-Étienne (email: firstname.name@emse.fr),
†LIRMM (email: philippe.maurine@lirmm.fr)

***Abstract*—ElectroMagnetic Pulse Injection (EMPI) has recently been demonstrated to be an efficient fault injection technique with many advantages especially when considering security issues of Systems on Chip (SoC) embedded on ball grid array packages, i.e. when adversaries do not have an easy access to the backside. EMPI must therefore be considered as a real threat against smartcards and SoC from now on. Among the usual countermeasures against fault attacks, one can identify the use of embedded sensors. If one can find voltage glitch or laser shot detectors in the literature, there is only one proposal which puts forward the idea of detecting ElectroMagnetic Pulse (EMP). However, this former sensor requires a fine tuning of some timing characteristics and, as a result, its use appears complex and even impractical within a SoC which are heterogeneous by nature and designed by worldwide teams. Within this context, this paper introduces and experimentally validates a new sensor allowing to detect EMP. Because the sensor is fully digital, it is low cost and above all fully compliant with the standard design flow of SoC.**

I. Introduction

Security is nowadays very largely ensured by smartcards that remain the main element of trust in many systems that manipulate sensitive data, such as mobile phone. However, there is a demand for secure and high performance SoC.

Those systems are complex systems of large size, heterogeneous by nature, which, unlike smartcards, are embedded in packages rendering the access to a side (the front-side or the back-side) difficult. This explains why EMPI [1], [2], [3] currently appears as a main threat against such systems [4]. Indeed, EMPI works perfectly front-side and back-side and does not require the removal of the package using chemical processes except if it is a metallic package. In the latter case, the removal is usually easy and does not require advanced equipments nor skills.

Among the usual countermeasures against physical attacks, one can identify on-chip sensors as a first barrier allowing to thwart Fault Attacks (FA) or at least to render their application much more difficult. Sensors allowing to monitor, directly or indirectly, on the fly the supply voltage [5], the temperature [6], to detect a physical intrusion [7] or the occurrence of light flashes are now widely deployed in modern smartcards. However, there is no proposal in the literature of a sensor allowing to detect the occurrence of short and powerful EMPs, except one in the paper [8] which demonstrates that a voltage glitch detector is partially efficient to detect EMPs. Furthermore, the proposed glitch detector, which monitors the timing slack before occurrence of a timing violation, is difficult to embed in a SoC produced in volume. Indeed, its use requires to finely tune the delay of some paths and to precisely know and control the timings, things that are really hard to achieve in complex SoC in which performances are controlled dynamically [9].

Within this context, this paper takes advantage of recent results [10], showing that EMPI does mainly induce faults by disrupting the switching process of D Flip-Flop (DFF), in order to define and experimentally validate a new sensor allowing to detect EMP. Because it is fully digital, this sensor consumes little power, occupies small area and is fully compliant with the standard cell design flow followed by Integrated Circuit (IC) designers to design complex SoCs while respecting the time to market constraint.

The rest of the paper is organized as follows. Section II is a review of the state of the Art about ElectroMagnetic (EM) injection with a focus on the effects of EMP on the behavior of IC. Section III introduces the proposed EM detector, explains its operation and details its characteristics. Section IV describes the test-chip and the EMP platform that have been considered to experimentally demonstrate the correctness of our proposal but also to estimate its performances and its limits. Section V describes the experimentations that have been performed but also details and discusses the obtained results. Finally, Section VI concludes the paper.

II. Related works : EM fault Injection and Sensors

This section gives the state of the Art on EMPI with a focus on the fault model and then discusses of embedded sensors to detect EMP.

A. State of the Art on EM Injection

EMPI, as a medium of attack, appears for the first time in literature in 2002 [1] in a paper stating that it is possible to read the memory contents by exploiting eddy currents. However, it was not until 2007 and [2] to see concrete results. They relate to fault attacks on a RSA mounted using a gas spark to generate EMP. Then in 2011, harmonic EM injection is demonstrated efficient to disrupt the behavior of an internal clock generator [11] but also that of a true random number generator [12]. Finally, in [4], [13], it has been proved that EMPI may induce faults during the course of an hardware Advanced Encryption Standard (AES) [14] but also in that of a soft implementation of this algorithm.

An important outcome reported by these papers is relative to the type of faults produced by EMPI. Indeed, evidences that it produces timing faults are given. This means that EMPs reduce temporarily and locally the supply voltage within targeted ICs.

978-3-9815370-6-2/DATE16/

Thus, the propagation delays of CMOS gates increase and then one or several setup time violations are induced according to the strength of the EMP.

This is a intuitive explanation of how EMPI induces faults. However, in 2014, [10] has experimentally demonstrated that it is possible to generate faults within a circuit at rest (clocked stopped), using an enhanced EMPI platform. This means that EMPI is able to induce bitsets and bitresets. In 2015, Ordas et al, have finally introduced and experimentally demonstrated in [15] the correctness of a new fault model for EMPI. This model is called the sampling fault model.

According to this model, EMPI is able to modify significantly but temporarily the bias of any interconnect within an IC. Considering this, and the fact that the DFFs are the only CMOS gates that must meet some operating constraints, they assumed in [15] that DFFs are the main front door to inject faults within IC using EMPI. More precisely, the sampling fault model states that the EM susceptibility of DFFs is higher than any other type of CMOS gates and that its value is:

- really high during their switching, i.e. when a rising edge of the clock occurs. This means that an EMP of moderate amplitude is sufficient to induce a fault in an IC at each rising edge of the clock.
- lower when the clock signal is stable (at 0 or Vdd). This does not mean that no fault can be induced but solely that only bitsets or bitresets can be produced using a much more powerful EMP.

This model is summarized in Fig. 1. As shown, the EM susceptibility of an IC strongly depends on the clock signal. More precisely, its value is really high during a time windows centered around the rising edges of the clock (t_s) and much lower the rest of the time. This time window marked by a '(1)' in Fig. 1 corresponds to the time during which the input signal D of DFF must remain stable.

Indeed, for a correct operation of a DFF, the data D must be stable t_{setup} before and t_{hold} after the rising edge of the clock. These quantities depend on many technological and design parameters. Among them, one can identify the transition time of the clock, the supply voltage, the temperature, transistor widths, and all relevant technological parameters. For a typical 90 nm CMOS process, the duration of these windows could range within 250 ps to 1 ns. However, during the practice of EMPI, these time windows could be perceived much larger

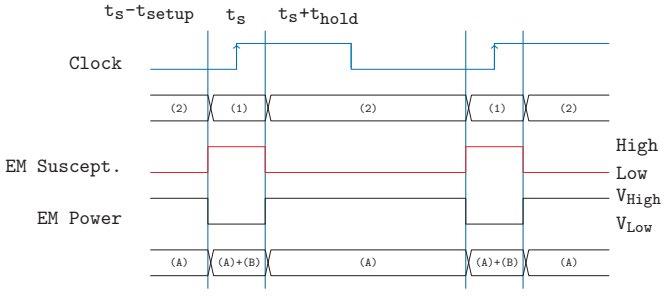

Fig. 1. The sampling fault model

because an EMP produces voltage drops and ground bounces characterized by longer durations. If a significant drop or bounce falls into one of these time windows a fault appears.

In Fig. 1 are also reported the effects of EMP, i.e. the type of faults that can be induced. During time windows marked by a '(2)', only bitsets and bitresets (marked by '(A)') can be obtained while during time windows marked by '(1)' bitsets, bitresets and sampling faults (markeds by '(B)') can be obtained.

B. Embedded Sensors to Detect Fault Attacks

Embedded sensors are important elements in the set of countermeasures deployed within IC, and especially smartcards, to thwart fault attacks. They allow detecting most but not all fault injection attempts. They are therefore not sufficient to fully stave off fault attacks but remain key elements of a complete defense strategy.

Up to the best of our knowledge, there is only a publication [8] related to the use of a voltage glitch detector to detect EMP. If the results reported demonstrate convincingly the partial efficiency of this voltage glitch sensor to detect EMP, this proposal is not fully satisfactory. Indeed, the integration of this sensor within a smartcard appears difficult and even unpractical within a SoC. Let us explain why, starting by reminding its operation.

The principle of this detector consists in detecting the violation of a guarding delay prior to any timing violation. The clock signal is used as a reference to be able to draw comparisons between the guarding delay and the clock period (T_{CK}). In normal operation the guarding delay is set greater than the critical time (D_{pMax}), but smaller than T_{CK}. By doing so, if T_{CK} is decreased for the purpose of inducing a timing violation, it will have to be shorter than the guarding delay (what we call a guarding delay violation) before inducing a timing violation. Hence, if the detector is able to detect that, it will also be able to detect any fault injection attempt by clock glitches. Similarly, if the power supply voltage is decreased by an EMP, the guarding delay will be increased as well. As a consequence, the violation will raise and be detected.

The schematic of the detector depicted in Fig. 2-a fits with the above mechanisms. The guarding delay (denoted `delay`) is implemented with the circuit logic. It is used to obtain a delayed clock (denoted `Dclock`) from the clock signal (denoted `Clock`) where $\texttt{Dclock}(t) = \texttt{Clock}(t - delay)$. A DFF is used as a phase comparator between `Clock` (connected to its data input) and `Dclock` (connected to its clock input). The design is tuned in order to comply with the timing given in eq. 1.

$$D_{pMax} < delay < T_{CK} \tag{1}$$

In normal operation, as depicted in Fig. 2-b, the DFF's output (denoted `Alarm`) is high. Fig. 2-c illustrates the detection of a power supply glitch. As the power supply voltage is decreased the guarding delay is increased and goes larger than T_{CK}. Thus, a low level is latched by the DFF on the next rising edge of

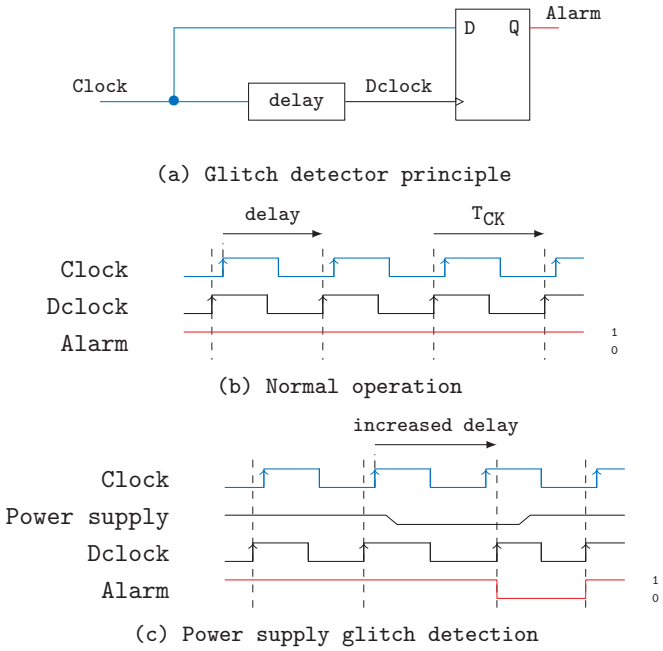

Fig. 2. Delay-based countermeasure principle

Dclock: the alarm is triggered at low level indicating a fault injection attempt.

The key drawback of this detector is the use of delay implemented with logic gates. Indeed, this delay must be tuned according to the IC timings. If this can be done using post silicon trimming on moderated size IC like smartcards, this is really difficult to do, and even impossible, on modern SoC designed in advanced technologies prone to large process variations, especially if Dynamic Voltage and Frequency Scaling (DVFS) technologies are implemented to control performances. Indeed, such solutions result in reducing at best the quantity $T_{CK} - D_{pMax}$, involved in eq. 1, to meet the performance while reducing at best the power consumption. Hence the difficulty of integrating this detector in high performance SoC.

This explains why a new EMP detector is introduced in this paper. Its main advantage is to be fully digital and therefore easily integrable in modern SoC independently of their size. Because it is fully digital, no post silicon solutions are needed to use it and its area and power consumption are low.

III. A Fully Digital EMP Detector

A. Half-Detector (HD)

The proposed detector is made of two HDs shown Fig. 3. One may observe that a HD features mainly DFFs which are, according to the EMP susceptibility model introduced in [10], the most susceptible gates, especially when their clock input toggles. DFF_1 's output (Q_1), which is initialized respectively at 1, switches at each rising edge of the clock. In a complementary way, DFF_2 's output (Q_2), which is initialized respectively at 0, switches at each falling edges of the clock. Thus Q_1 and Q_2 takes systematically opposite values at each rising clock edge (see Fig. 3 and Fig. 4) except when an EMP occurs as shown on Fig. 5 . This design choice was done to maximize the amount of time per clock cycle during which at least a DFF is switching. This allows maximizing the time during which the EM susceptibility of the detector is high. DFF_1 and DFF_2 are therefore the sensitive part of the detector.

DFF_3 generates the alarm by xoring Q_1 and Q_2. In normal condition, the output of DFF_3 remains stable at '1'. When an EMP induces a fault during the switching of DFF_1 or DFF_2, i.e. either during a rising edge (see Fig. 5) or a falling edge of the clock, the output of DFF_3 switches to '0'; an alarm is raised. Note that the fault can also be directly injected in DFF_3 and so be detected. Therefore, DFF_3 is also a sensitive part of the half detector.

B. Full detector

The full detector features two HDs. This choice was done in order to cover all possible transitions of the DFF : (rising clock edge, Q rising), (rising clock edge, Q falling), (falling clock edge, Q falling), (falling clock edge, Q rising). To do that, the DFFs of the two half detectors are initialized in an opposite manner as shown in Fig. 6 showing the design of a full detector.

The latter finally features five DFFs, 6 inverters, 2 Xor gates and 1 And2 gate (for a total of 34 nand eq.) which generate signal which will be sampled.

IV. EMP detector design, EMPI platform and Experimental Protocols

In order to demonstrate the efficiency of the proposed detector as well as to confirm the correctness of the proposed EMP fault model introduced in [10], it was decided to perform

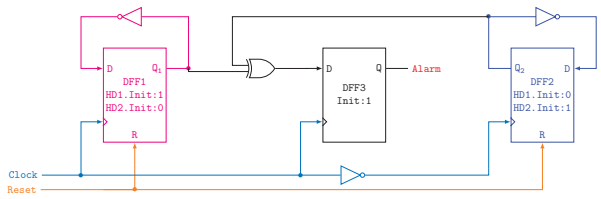

Fig. 3. Half-Detector Schematic

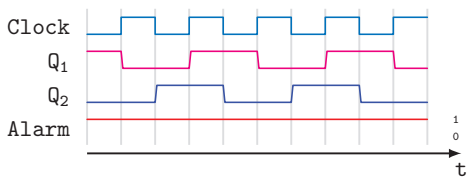

Fig. 4. The correct behavior of the Half-Detector

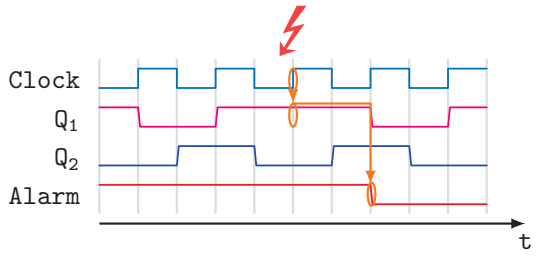

Fig. 5. An EMP induces a fault during a rising edge of the clock and is detected by the half detector

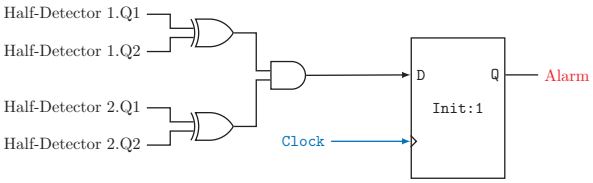

Fig. 6. Full-Detector Schematic

some tests on an Field-Programmable Gate Array (FPGA) prototype.

A. Mapping of the EMP detector on FPGA

To easily integrate several EMP detectors in a testchip, a *hard macro* [16] was designed (a standard cell based circuit could have been designed in case of integration in an Application Specific Integrated Circuit (ASIC)). This choice was done in order to be able to control the placement of the EMP detectors in our testchip while being fully compliant with the HDL flow of Xilinx. The testchip considered during the experimental campaigns features an hardware AES [14], an RS232 communication block and a finite state machine 37 EMP detectors were also mapped into this testchip. The placement of this sensor was done geometrically so that to cover the whole IC surface; no special placement policy was therefore followed. All the EMP detectors were organized in a mesh allowing to generate a global alarm signal. The mesh of EMP detectors occupies only 5 % of the FPGA. As a result, the instrumented design is 13% larger than the unprotected one.

This testchip was mapped into an Spartan3E-1000 from Xilinx and works at 100Mhz. Fig. 7 shows the device that has been used. One may observed that the package was chemically removed in order to process to some tests with the EM injectors close to the IC surface (front-side) but also at a height greater or equal to the thickness of the package. Even if EMPI does not need the package removal, injections are more accurate when we remove it.

B. EMPI Platform

To access the EMP susceptibility of the testchip as well as the efficiency of the EMP detectors, the EMPI platform shown Fig. 8 was used to scan the FPGA's surface or package surface depending at which height was positioned the EM injector. When the EMP injector was positioned above the

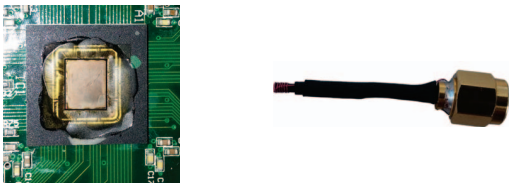

Fig. 7. Decapsulated FPGA and injector used during experimental campaigns

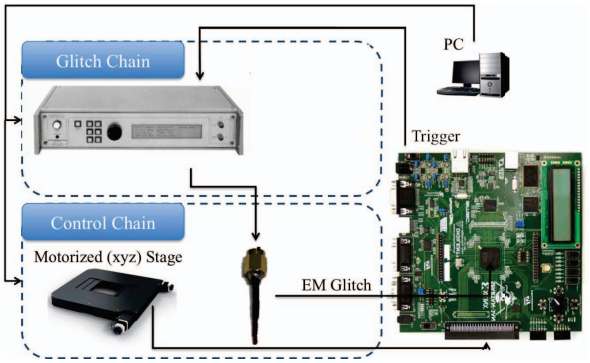

Fig. 8. EMPI platform used during experimental campaigns

package the scanned area was $17 \times 17\,\mathrm{mm}^2$, while when it was positioned at the contact of the FPGA surface the scanned area was $7 \times 4.9\,\mathrm{mm}^2$. The EMP injector we used is shown Fig. 7. It is really similar to that used in [10]. As shown the EMPI platform features an voltage pulse generator (400V), a motorized stage with an accuracy of $5\mu\mathrm{m}$ and a PC that controls all the equipments and the Device Under Test (DUT).

C. Scanning process

Most experiments consisted in near field scans. For a given positioning (X, Y) of the EM injector, its starts by the programming of the FPGA. During this step the output of the voltage pulse generator is disabled. Then the settings (the amplitude, the delay, the pulse width, ...) of the pulse generator are set and its output enabled. At that stage, the FPGA waits for a key, a plaintext to be ciphered and finally a command allowing the AES to start the ciphering. A trigger signal is also generated on an IO pin of the FPGA before the ciphering. This signal is used as an external trigger for the voltage pulse generator. The ciphering finished, the ciphertext is sent back to the user via the RS232; the state of the alarm signal is also collected. Finally, the process can be repeated as many times as wanted at the same position or not, with the same settings or not. It should be noticed that a maximum response time is set to deal with EMPI leading to a no response of the IC.

A logfile is also generated. It contains all the settings of each EMPI (X,Y, pulse amplitude, ...), the plaintext, the key, the received ciphertext and the state of the alarm. This logfile is then exploited to interpret the results and draw the cartographies.

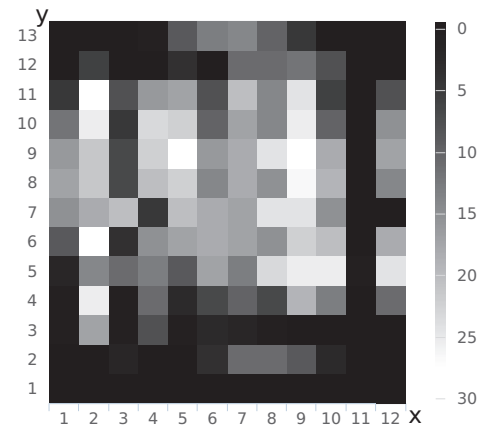

Fig. 9. Number of EMPI detected by the mesh of EMP detectors

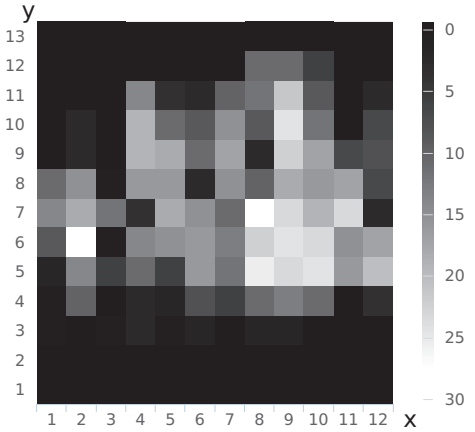

Fig. 10. Number of EMPI that have lead to a faulty response

V. Experimental Results

A. Efficiency of the EMP detector

To demonstrate that the detector is able to efficiently detect EMPs, a first experiment was conducted. During this test, the FPGA was supplied with its nominal voltage: 1.2 V. It has consisted in performing injections at $12 \times 13 = 156$ positions (at the contact of the IC surface) with pulse amplitudes in $\{-300\,\mathrm{V}, ..., -80\,\mathrm{V}, 80\,\mathrm{V}, ..., +300\,\mathrm{V}\}$. The scan area was therefore the die, and at each coordinate 44 EMPI were performed.

Fig. 9 gives the number of EMPI detected by the mesh of EMP detector at each coordinate whereas Fig. 10 gives at each coordinate the number of EMPI that have lead to a fault.

As shown most, but not all, the EMPI leading to a faulted cipher-text have been detected by the mesh of EMP detector. There are also many positions at which EMPI are detected even if there is no fault induced in the IC. These first results validate the proposed schematic but also confirm the correctness of the EMP fault model introduced in [10].

To definitively validate the proposed CounterMeasure (CM), it is necessary to demonstrate that its EMP susceptibility is higher than that of the design. For that, at each coordinate, we measured $V_{S_{AES}}$ defined as the minimum (in absolute value) value of the pulse allowing to induce a fault. Similarly, $V_{S_{CM}}$, defined as the minimum (in absolute value) value of the pulse allowing to trigger the mesh of EMP detector, was measured at each coordinate. White squares in these cartographies correspond to coordinate at which the EMPI is easily detected while black squares correspond to positions at which EMPI induce a fault before triggering the mesh of EMP detectors. As aforementioned, in most case EMPI trigger first the alarm.

B. Detailed analysis of the induced faults

If the last cartographies demonstrate that the propounded EMP detector is able to detect EMPI, no distinction between the different responses of the FPGA was considered to draw them. In practice, different situations (or events) were observed according to the settings of the EMPI and especially the EM injector's position:

Event AF:
: It corresponds to an EMPI triggering the alarm at a lower pulse voltage than the one required to induce a fault; this is a favorable case.

Event CF:
: It corresponds to an EMPI triggering the alarm at a higher pulse voltage than the one required to induce a fault; this is an unfavorable case.

Event Idem:
: It corresponds to an EMPI triggering the alarm at the same pulse voltage than the one required to induce a fault; this is a quite favorable case.

Considering these three events, we defined the Detection Rate (DR) of an EMP detector as follows:

$$DR = \frac{\mathrm{Card}(AF) + \mathrm{Card}(Idem)}{\mathrm{Card}(AF) + \mathrm{Card}(CF) + \mathrm{Card}(Idem)}. \quad (2)$$

With such a definition, we were able to quantify the efficiency of our mesh of EMP detector from the experimental results described above: its detection rate DR is equal to 86%, i.e. it is able to detect $86\,\%$ of the EMPI produced with the EMP injector at the contact of the IC's surface. This is a quite significant results. One may wonder if setting the supply voltage Vdd to its corner values (1.1 V and 1.2 V) changes the DR. Experiments were done again with $Vdd = 1.1\,\mathrm{V}$ and $Vdd = 1.3\,\mathrm{V}$. The DR was found equal to $85\,\%$ in both cases demonstrating that the EMP detector operates correctly on the whole supply voltage range on which the IC manufacturer warrants its IC.

C. Overpackage Detection Rate

If the proposed EMP detector has been demonstrated efficient in detecting EMPI produced in the close vicinity of the IC's surface, one may wonder if it is the case for EMPI produced over the package. Full package scans were launched, following the same experimental protocol than for the scan of

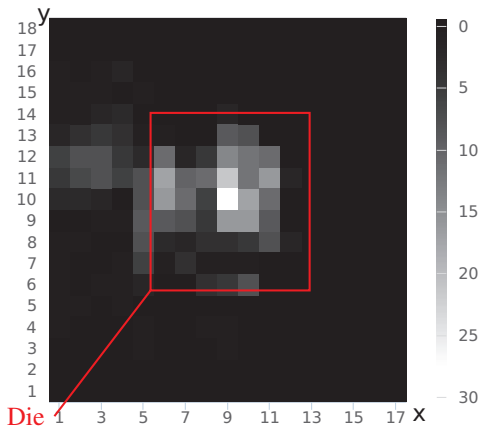

Fig. 11. Number of detections by firing position

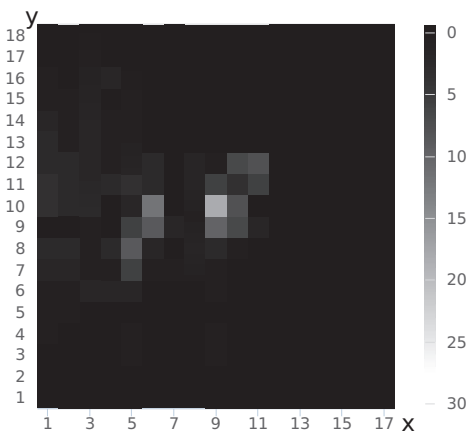

Fig. 12. Number of faulted ciphers by firing position

the die, but this time with the probe at the surface of the package.

Fig. 11 gives the number of EMPI among 44 detected by the mesh of detectors at each coordinate. Fig. 12 gives the number of EMPI that have induced a faulty behavior. As shown, the mesh of detectors is still efficient. It is even able to detect EMPI produced far from the die. During this experiment, the Detection Rate was measured : $DR = 92\,\%$.

VI. Conclusion

In this paper, we have introduced and experimentally validated a fully digital EMP detector. The fact it is fully digital can be perceived surprising if we think about the inherent analogue nature of EMPI. However, it has been rendered possible by the EMP's fault model established by S. Ordas in [10]. The latter states that D-type Flip Flop are the most sensitive gates in digital designs.

Because the proposed EMP detector is fully digital it can easily be deployed in complex systems on chip as in smartcards so that to form a more or less complex mesh of detectors. In addition, because of its digital nature, its area and its power consumption are really low.

Next steps will be to study the optimal placement of EMP detectors with respect to power ground networks expected to convey EMP disturbances. In addition, because of its structure, one may wonder if the proposed detector is able to detect laser shots or Body Bias Injection attempts [17].

References

[1] D. Samyde, S. Skorobogatov, R. Anderson, and J.-J. Quisquater, "On a new way to read data from memory," in *Security in Storage Workshop, 2002. Proceedings. First International IEEE*. IEEE, 2002, pp. 65–69.

[2] J.-M. Schmidt and M. Hutter, *Optical and em fault-attacks on crt-based rsa: Concrete results*. na, 2007.

[3] 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-*, 2012, published: Cryptology ePrint Archive, Report 2012/123 http://eprint.iacr.org/.

[4] P. Maurine, "Techniques for em fault injection: equipments and experimental results," in *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2012 Workshop on*. IEEE, 2012, pp. 3–4.

[5] A. G. Yanci, S. Pickles, and T. Arslan, "Detecting voltage glitch attacks on secure devices," in *Bio-inspired Learning and Intelligent Systems for Security, 2008. BLISS'08. ECSIS Symposium on*. IEEE, 2008, pp. 75–80.

[6] J. J. L. Franco, E. Boemo, E. Castillo, and L. Parrilla, "Ring oscillators as thermal sensors in fpgas: Experiments in low voltage," in *Programmable Logic Conference (SPL), 2010 VI Southern*. IEEE, 2010, pp. 133–137.

[7] J. Unsworth and M. Mapson, "Electro-active cradle circuits for the detection of access or penetration," Oct. 4 1994, uS Patent 5,353,350.

[8] L. Zussa, A. Dehbaoui, K. Tobich, J.-M. Dutertre, P. Maurine, L. Guillaume-Sage, J. Clediere, and A. Tria, "Efficiency of a glitch detector against electromagnetic fault injection," in *Proceedings of the conference on Design, Automation & Test in Europe*. European Design and Automation Association, 2014, p. 203.

[9] S. Yang, W. Wolf, N. Vijaykrishnan, D. N. Serpanos, and Y. Xie, "Power attack resistant cryptosystem design: a dynamic voltage and frequency switching approach," in *Proceedings of the conference on Design, Automation and Test in Europe-Volume 3*. IEEE Computer Society, 2005, pp. 64–69.

[10] S. Ordas, L. Guillaume-Sage, K. Tobich, J.-M. Dutertre, and P. Maurine, "Evidence of a larger EM-induced fault model," in *Smart Card Research and Advanced Application Conference (CARDIS)*, Paris, France, Nov. 2014. [Online]. Available: http://hal-emse.ccsd.cnrs.fr/emse-01099037

[11] F. Poucheret, K. Tobich, M. Lisart, L. Chusseau, B. Robisson, and P. Maurine, "Local and direct em injection of power into cmos integrated circuits," in *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2011 Workshop on*. IEEE, 2011, pp. 100–104.

[12] P. Bayon, L. Bossuet, and A. Aubert, "Random number generation: a potential target of electromagnetic emanation analysis?" in *Cryptographic Architectures Embedded in Reconfigurable Devices, Cryptarchi 2011*, 2011.

[13] A. Dehbaoui, J.-M. Dutertre, B. Robisson, and A. Tria, "Electromagnetic transient faults injection on a hardware and a software implementations of AES," in *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2012 Workshop on*. IEEE, 2012, pp. 7–15.

[14] "FIPS 197, advanced encryption standard (aes)," http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf, accessed: 2015-03-30.

[15] S. Ordas, L. Guillaume-Sage, and P. Maurine, "Em injection: Fault model and locality," in *Fault Diagnosis and Tolerance in Cryptography (FDTC 2015)*, Saint Malo, France, 2015.

[16] "Xilinx. fpga editor guide." http://www.xilinx.com/support/swmanuals/21i/download/fpedit.pdf, accessed: 2015-03-30.

[17] K. Tobich, P. Maurine, P.-Y. Liardet, M. Lisart, and T. Ordas, "Voltage spikes on the substrate to obtain timing faults," in *Digital System Design (DSD), 2013 Euromicro Conference on*. IEEE, 2013, pp. 483–486.