Hardware Trojan Detection by Delay and Electromagnetic Measurements

X-T Ngo, I Exurville, S Bhasin, J-L Danger, S Guilley, Z Najm, Jean-Baptiste Rigaud, Bruno Robisson

Hardware Trojan Detection by Delay and Electromagnetic Measurements

X-T. Ngo[2], I. Exurville[1,3], S. Bhasin[2], J-L. Danger[2,4], S. Guilley[2,4], Z. Najm[2], J-B. Rigaud[3] and B. Robisson[1]

[1] CEA-Tech PACA, LSAS
Gardanne, France.
firstname.lastname@cea.fr

[2] TELECOM ParisTech
Paris, France.
firstname.lastname@enst.fr

[4] Secure-IC S.A.S.
Paris & Rennes, France.
firstname.lastname@secure-ic.com

[3] EMSE, LSAS
Gardanne, France.
lastname@emse.fr

***Abstract*—Hardware Trojans (HT) inserted in integrated circuits have received special attention of researchers. In this paper, we present firstly a novel HT detection technique based on path delays measurements. A delay model, which considers intra-die process variations, is established for a net. Secondly, we show how to detect HT using ElectroMagnetic (EM) measurements. We study the HT detection probability according to its size taking into account the inter-die process variations with a set of FPGA. The results show, for instance, that there is a probability greater than 95% with a false negative rate of 5% to detect a HT larger than 1.7% of the original circuit.**

I. Introduction

The trust and security of Integrated Circuits (IC) design and fabrication is critical for sensitive fields like finance, health, and governmental communications. Due to the complexity and the high cost of IC fabrication cycle, more and more firms outsource their production. This trend gives a possibility for an adversary to introduce malicious circuit, called Hardware Trojan horse (HT), in any IC. It can either perform a Denial Of Service (DOS), deteriorate circuit performance [8], or steal sensitive information. Therefore, the HTs are considered a real threat which has gained attention from researchers.

HT can be inserted at any point during the design or fabrication process from Register Transfer Level (RTL) to layout and circuit fabrication. For example in [11], authors show some techniques to insert malicious circuitry at RTL level. These HTs, which are activated with a specific pattern inputs, can leak secret key via RS232 channels. The HT, unlike a software trojan, cannot be removed once it is fabricated. So, it is better to proactively prevent the insertion of a HT: few methods have been proposed. One seminal work is known as "*private circuits II*" [9]. This paper describes a proof-of-concept, too costly to be implemented. A more reasonable option has been recently proposed in [5]: it uses two codes to encode the state and mix it with encoded randomness, which allows to prevent an easy triggering and has a detection capability.

Otherwise it is important to detect it before it becomes effective. Previous works classify detection methods into two wide categories: *destructive* and *non-destructive*. Invasive methods destroy the chip to reconstruct successfully the GDSII and the netlist of the chip using chemical products and optical observation apparati as Scanning Optical Microscopy (SOM), Scanning Electron Microscopy (SEM), etc. The main advantage of such invasive technique is its accuracy for all malicious insertions. However, the destructive nature and lengthy time needed to reconstruct the netlist of the chip are two significant drawbacks.

Non-destructive methods compare the physical characteristics or logical state of an IC with a genuine circuit also known as the "golden circuit" at testing time or run-time. An optical detection of changes in the last metal layer of metallization has been proposed in [4]. In [1], authors propose to add re-configurable DEsign-For-ENabling-SEcurity (DEFENSE) logic to the functional design. In test time, the first approach is using logic testing. It involves applying test patterns at the input and try to detect anomalous behaviors of ICs [3], [10]. But almost all test-time detection techniques are difficult to realize. They cannot ensure that HT will be activated because of the complexity of test patterns. Therefore, Side-Channel Analysis (SCA) can also be deployed to detect HTs. These methods observe and compare physical traits (power consumption, time delay, etc.) of an IC under test against a trusted IC [15], [16].

A promising approach for detecting HTs is to study delay measurements, because the HT does not need to be triggered and its mere presence can impact a part of the IC path delays. In [12], [18], path delays are used as a fingerprint for a trusted IC (Golden Model, or GM). The first purpose of our paper is to show how HTs can be detected despite having small size and being not logically connected with elements of critical data paths. A significant advantage of this method is the detection of IC alterations without activating the payload of the HT and by using a low cost mean. The importance of not only limiting the study to critical paths is shown. Moreover, the intra-die process variations are considered in these analyses.

On the other side, previous works on SCA-based detection have been limited to either simulations or power consumption measurements on some real circuit. In [13], authors present a practical evaluation of HT detection using SCA on FPGA. But the experiments were performed on a single FPGA so the

inter-die process variations were not taken into account. In our paper, we also study the HT detection method based on ElectroMagnetic (EM) measurements which provide a better spatial and temporal resolution than power measurements hence improving HT detection result. Moreover, we propose a metric to evaluate the impact of inter-die process variations in HT detection based on EM measurements according to HT size. Process variations are studied on 8 different Virtex 5 FPGA boards.

The rest of this paper is organized as follows. Section II describes the structure of created HTs and how they are inserted at layout level in FPGA. Section III presents the HT detection using delay analysis approach. Section IV shows the results of HT detection based on ElectroMagnetic (EM) measurements. Section V studies the impact of inter-die process variations on HT detection based on SCA. We finish with a small conclusion and some perspectives in section VI.

II. Hardware Trojan Insertion

A. Hardware Trojan Insertion Methodology

Our attack scenario is the following: attacker is an untrusted ASIC foundry. Upon reception of the tape-out database (GDS file), the founder inserts a HT before fabrication. In order to reduce the impact of HT on the genuine circuit, the HT must be inserted keeping the original placement and routing of target circuit. To imitate the HT insertion on ASICs, we need to keep the same placement and routing between the golden circuit and HT infected circuit on FPGAs. Hence the only difference between the two is the logic and the interconnect utilized by the HT. To insert a HT, without modifying the routing, we apply the following steps in the Xilinx framework:

1) Synthesize, translate, map, place & route the original circuit. In our case, it is an AES-128 block cipher.
2) Extract the Native Circuit Description (NCD) file which contain all the circuit, placement & routing information of original circuit (this is the golden model).
3) Open the NCD file using FPGA Editor tool and insert HT in unused LUTs and Slices of FPGA, manually or by a script.
4) Generate bit files for both original and infected circuits with FPGA Editor.

With this method, we can ensure that the placement and routing is the same in both golden and HT-infected circuit. It allows us to perform the proof-of-concept of HT attack at ASIC layout level (the FPGA fabric is seen as an ASIC).

B. Hardware Trojan Description

A HT is distinguished by its spread into the IC, its size and its behavior. For testing our method in different contexts, two types of HT were implemented: a **combinational** and a **sequential** one.

The trigger part of the combinational HT scans 32 SubBytes signals at the input of the SubBytes step. The trigger is activated when simultaneous occurrences at '1' of all the 32 SubBytes bits occur. Its payload consists in causing a DOS. This HT uses 0.19% of slices in the FPGA, whereas AES implementation covers 38.26% of the FPGA slices.

The sequential HT includes a 32-bit counter with a comparator which is incremented for each AES encryption. The HT becomes active when the counter is equal to a defined value. As for the combinational HT, the activation of the sequential HT involves a deny of service. For this HT, 0.36% of slices used in the FPGA were required.

In both cases, during our experiments, the two HTs were not activated. Indeed, we focus on detecting their presence into the FPGA before the payload is triggered. In addition, we insist that none of the two HT are on the critical path.

III. HT Detection Using Delay Analysis

A. Approach

Most digital ICs work according to the principle of synchronicity: they use a common clock signal to synchronize their internal operations. When a datum leaves a register bank on a clock rising edge, it progresses through the combinational logic, and goes into the next register bank to be sampled on the next clock rising edge. This is depicted in Fig. 1, where the square boxes represent registers and the cloud represents the combinational logic.

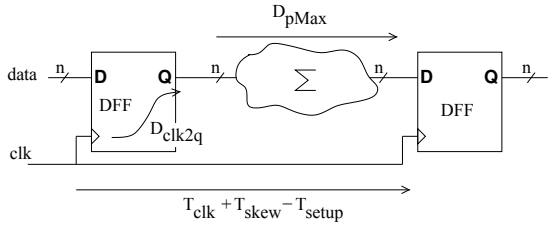

Figure 1. Internal architecture and timing of synchronous ICs

The clock period T_{clk} has to respect timing constraints, in particular a *setup condition* represented by equation (1):

$$T_{clk} > D_{clk2q} + D_{pMax} + T_{setup} - T_{skew} + T_{jitter} \quad . \qquad (1)$$

This ensures the datum is correctly stored in the register bank. This equation is related to the delay between the clock rising edge and the current update of a register's output D_{clk2q}, the value D_{pMax} of the maximum propagation time through the combinational logic. The T_{setup} associated to the set-up time which is the amount of time for which a D Flip-Flop input (DFF) must be stable before the clock's rising edge to ensure reliable operation. The T_{skew} represents the skew or slight phase difference that may exist between the clock signals at the inputs of two different registers and T_{jitter} is the clock jitter. A hold time T_{hold} expresses an equivalent constraint as the T_{setup}, but after the clock edge.

Chakraborty et al. implemented a key-enabling additional mode of operation [7] using logic testing and Abramovici et al. used a different approach with the design of a re-configurable platform [1] to inspect the system of a analog chip. In this paper, the proposed method did not call for an additional circuitry. Thus, there is no supplementary cost during the

manufactured step, and it remains easy and economical to realize. The only constraint is the access to the clock.

Our approach is based on a clock glitch, aiming at turning the comparison into an equality in Eqn. (1). The clock period is reduced gradually until the last signal transition of the datum becomes too close to the clock's rising edge. This triggers a set-up time violation which leads the DFF's output into a meta-stable state. In this way, the clock glitch establishes a good means to induce path delay violation in ICs [2].

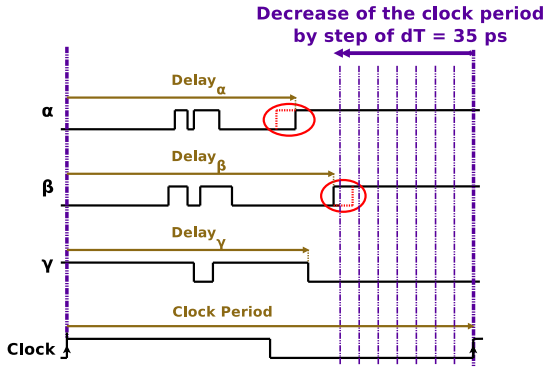

Figure 2. Principle of path measurement, for a given pair (P, K)

Our hypothesis is the following: the addition of HT in a design can have impact on the internal signals. Figure 2 illustrates the experimental procedure. Delays of α, β and γ are measured using an iterative decrease of the clock period (with a 35 ps step). These delays are performed with a GM as a reference and then with IC under test. The insertion of a HT will shift the timings, as circled in red in Fig. 2. The comparison of the critical path between the GM and the circuit returning from fabrication allows to detect a purported HT. In Fig. 2, the critical path is net β. Although the HT might not have modified the structure of this path, this path will be influenced (through the power-ground network) by the HT triggering logic, and thus will be slightly different. Of course, if the HT has gates inserted in the path β, then its length will be significantly longer.

B. Model and Results on One Chip

Our experiments were made with three different implementations of the AES: one clean and two infected with a HT. From these implementations, we generate three bit files which are inserted one after the other into the FPGA. We inject our clock glitch on the 10^{th} round of the AES. Thus, we obtained directly the faulted ciphertext. The experiment was repeated 10 times to lower measurement noise, for each $10,000$ random pairs {plaintext, key} for all three bit files. Internal signal waveforms depend on the data processed, thus the critical path varies when (P, K) varies. Therefore, a matrix is generated to record the delay for all values of the detected AES bits. A total of 51 decrease steps of 35 ps were performed.

From our data measurements, we extract the critical path of AES bits, estimated by the number decrements by clock steps needed to fault each considered bit.

In [14], the delay model is refined as the sum of a static part (d_S) and a random process variation part (d_R). In our study, we define the delay D_{GM} of a net N_a of the GM by Eq. (2), where d_{PV} is the arbitrary delay induced by the intra-die process variations and d_{M_r} is the random metastability, environmental and factor noises (noted r):

$$D_{GM}(N_a, r_1) = d_{S_a} + d_{PV_a} + d_{M_{r_1}} \ . \quad (2)$$

We assume that adding a HT in a device changes at least the delay of one net. If the HT is close to a net N_a (or directly connected to a net N_a), d_{HT_a} is defined as the random delay added by the HT to the net N_a in an infected IC. Thus, the delay equation for the net N_a is:

$$D_{HT}(N_a, r_2) = d_{S_a} + d_{PV_a} + d_{M_{r_2}} + d_{HT_a} \ . \quad (3)$$

To deal with sources of uncertainty represented by d_{M_r}, we performed our experiments 10 times with the GM bit files. $\Delta\mathcal{D}^{10}_{\mathcal{GM}}(N_a)$ is the mean delay for the delay differences $\Delta D_{GM}(N_a, r)$ of all 10 experiments performed on the same board for different experimental conditions.

Compared to the GM, the infected IC has a different layout. Thus, if there is a HT, the delay difference $\Delta D(N_a, r)$ is described in Eqn. (4):

$$\Delta D(N_a, r) = |\Delta\mathcal{D}^{10}(N_a) - D_{HT}(N_a, r)| = |\Delta d_{M_r} - d_{HT_b}| \ . \quad (4)$$

We determine the difference $|\Delta\mathcal{D}^{10}(N_a) - D(N_a)|$, for different bit files with and without HT. As illustrated on Fig. 3, for the sake of clarity, four differences were plotted: two for an uninfected AES in light green (Clean_1 and Clean_2) and two for each HT in dark red (HT_{comb} and HT_{seq}).

The results are illustrated for two representative pairs (P, K), namely n°13 and 47. The trend is the same for all 50 tested pairs (P, K). This technique detects the two implemented HTs. The X-axis is the bit number (in range $[\![1, 128]\!]$) and the Y-axis is the delay difference calculated (refer to Eqn. (4)).

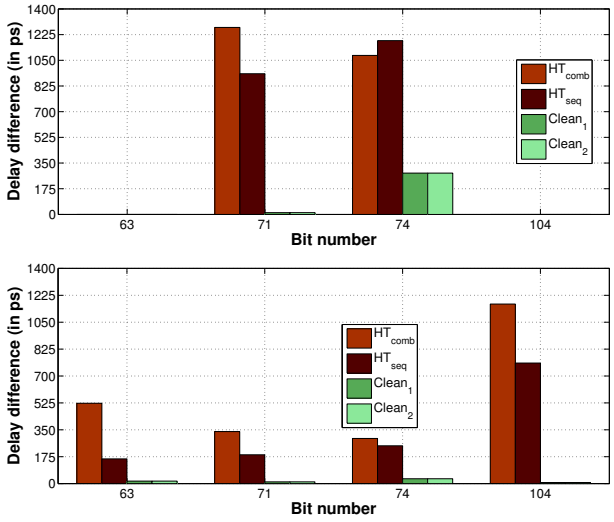

Figure 3. Impact of two HTs on delays, for (P, K) n°13 (up) and n°47 (down)

One can highlight on the importance to study not only the critical path but all the data path delays. Each implemented

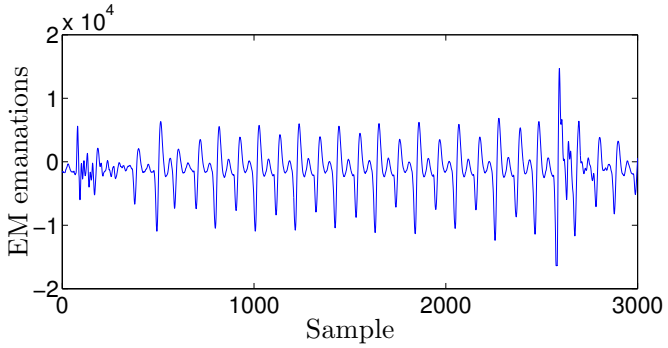

Figure 4. EM measurement of a single AES-128 encryption

wire can be considered as a HT sensor. Even if no logical connection exists between the design and the HT, both share the same power grid inside the FPGA. These electric connections make the HT detection easier.

Besides, as the data path depends on the data processed, it is relevant to use several (P, K) pairs, to detect different sets of bits. Each one brings information. As illustrated on Fig. 3, the more (P, K) pairs are studied, the more bits will be sampled, the more evidence about HT presence is collected. Furthermore, the false positive rate is decreased.

C. First Conclusion

In this first part, we have shown that delays are affected, sometimes by as much as 1 ns, if a HT is inserted. This method requires an access to the internal clock of the design. In cases where the internal clock access cannot be reachable, Side Channels Analysis can be another solution to detect HTs. An EM measurement approach is described in Sec. IV.

IV. HT Detection Using Electromagnetic Measurement

In this section, we present the result of HT detection by EM measurement. The genuine AES and the infected AES with combinational HT presented in II-B are used to evaluate this method. These two designs are implemented on FPGA Virtex 5 ($LX30$). The experiment is performed using the setup described in Appendix B. The circuit is clocked with a frequency of 24 MHz. EM traces are acquired for random plaintexts, where each trace is averaged 1000 times by oscilloscope to minimize the measurement noise. A single EM trace is shown in Fig. 4. We noticed that the SNR seems good thanks to averaging done by oscilloscope. All the ten rounds of encryption can be distinctively seen in this trace.

The figure 5 shows 3 traces (two traces for the genuine AES design and one traces for infected AES with combinational HT design) for the same plaintext. The traces in black and blue are for genuine AES design taken at different moments with the same plaintext. That means we implement genuine AES design on the FPGA Virtex5, we acquire the first trace with plaintext P1. Then we turn off the setup, after we re-implement genuine AES on the same FPGA and we acquire the second trace with the same plaintext P1. It allow us to evaluate the measurement noise created by setup installation. Regarding Fig. 5, these two traces are nearly the same by averaging $1,000$ times with oscilloscope. Therefore setup noise is removed. The third trace is the one of infected AES design which is acquired with the same plaintext P1 as the two genuine AES traces. We noticed that the trace of infected AES (in red) is different comparing with the genuine AES traces at some samples. This difference comes from HT insertion. Therefore, the HT can be easily detected by comparing directly the genuine AES traces and infected AES traces with the same plaintext. Notice that the plaintext is fixed but unknown and that the HT is never activated during the experiments.

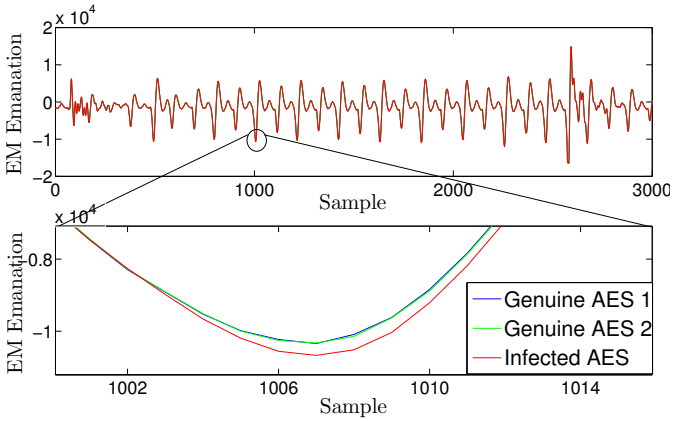

Figure 5. Hardware trojan detection using averaged EM traces

V. Impact of Inter-Die Process Variations on HT Detection

In the previous sections III and IV, HT insertion can be easily detected by using delay/EM analyses. But in these experiments, the genuine and infected circuits are programmed in the same FPGA. Therefore it is logical that the difference of delay or EM measurement between these two designs is visible which makes the detection of the inserted HT easy. In a real scenario, genuine and infected circuits are two distinct physical circuits. In this case, the comparison of delay or EM measurement between genuine and infected design is more difficult because of inter-die **Process Variations** (PV). Indeed two circuits fabricated with the same process, and in the same wafer, have slightly different physical and electrical behaviors. Therefore the result of HT detection using Side-Channel Analysis varies. In this section, we study the impact of PV on the HT detection method using EM measurement. More precisely, we want to evaluate the probability of HT detection according to its size taking into account the process variations of 65 nm technology.

A. HT Detection Feasibility with Inter-Die Process Variations

In order to evaluate the process variation noise, a set of 8 FPGA Virtex 5 LX30 with the technology of 65 nm are used. The EM detection method is performed in these FPGA. Using the test board FF324 Virtex 5 described in Appendix B, we can easily change the test FPGA while keeping other setup parts (probe position, board test, etc) intact. It ensures that we evaluate only the impact of process variations.

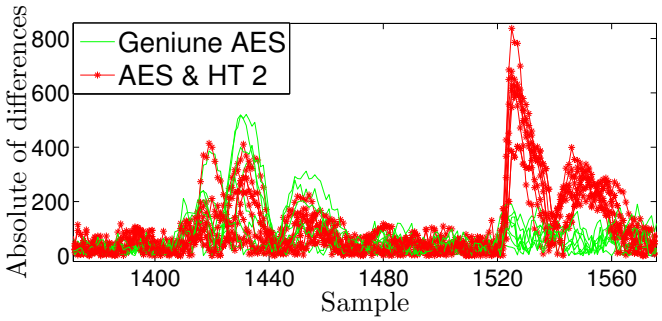

Figure 6. Impact of Process Variations on EM measurements

In order to evaluate the impact of HT size on the HT detection probability, we implemented 3 varieties of combinational HT described in Sec. II-B on the Virtex 5 FPGA. The description of these 3 different varieties is the following:

- **HT 1**: is activated when $2^5 = 32$ SubBytes input signals are at '1'. It occupies 0.5% of original AES.
- **HT 2**: is activated when $2^6 = 64$ SubBytes input signals are at '1'. It occupies 1.0% of original AES.
- **HT 3**: is activated when $2^7 = 128$ SubBytes input signals are at '1'. It occupies 1.7% of original AES.

In total, four different designs (genuine AES, infected AES with HT 1, HT 2 and HT 3) are implemented on 8 FPGA Virtex 5. For each implementation we acquired one trace (averaged 1,000 times) for the same plaintext. So finally, we acquired 32 mean traces for both genuine and infected AES.

In Fig. 6 we plotted the difference $Dg_j = |G_j - \underset{8}{\mathbb{E}}(G)|$ of all golden circuit inserted in 8 FPGA in green, and $Dt_{s,j} = |T_{s,j} - \underset{8}{\mathbb{E}}(G)|$ of all HT s infected circuit in red. Our notations are defined below:

- G_j is the EM trace of jth ($j \in \{1, 2, \ldots, 8\}$) golden circuit,
- $T_{s,j}$ is the EM trace of jth HT s ($s \in \{1, 2, 3\}$) infected circuit, and
- $\underset{8}{\mathbb{E}}(G)$ is the EM trace mean over all 8 golden circuits.

We can notice the static differences between green (or gray) curves that are due to process variations. We also notice that the difference of EM measurement in red (or black) for AES with HT 2 (1%) is bigger than the fluctuation of process variations for certain samples thus allow the possibility to detect this HT. This clearly shows that the insertion of HT is detectable if we choose specific points of interest.

B. False Negative Rate of HT Detection

As mentioned before, the reference to build a SCA detection method is mainly biased by process variations (PV), thus deteriorating HT detection efficiency. The process variation effect is modeled by a *random noise* with a Gaussian distribution [6]. In particular, some FPGAs will emit more and some less. The impact of the HT is a *deterministic* shift of the EM emanations, with more or less impact on different fabricated ICs. Therefore the HT contribution to the side-channel (e.g., the EM field) can be modeled by an activity offset on a net used by the HT. This is illustrated on figure 7.

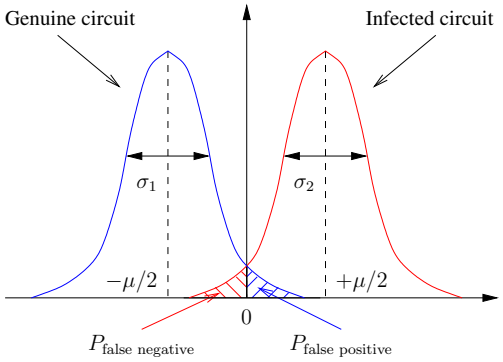

Figure 7. EM field probability density functions for a genuine and an infected circuit

This figure illustrates the activity distribution at a specific sample time and for a set of reference devices. The blue Gaussian curve is the activity distribution without any HT. And the red Gaussian curve is the distribution through the same set of circuits which contains the infected AES (with HT). It is merely an offset which depends on the HT size, placement and position relative to the probe in case of EM acquisitions. Using this model, the probability of detecting a HT can be calculated. Precisely, we can estimate the false positive and false negative probability of HT detection as a function of HT size for a set of ICs.

$$P_{\text{false negative}} = P_{\text{false positive}} = \frac{1}{2} - \frac{1}{2} \cdot \text{erf}\left(\frac{\mu}{2\sigma\sqrt{2}}\right) , \quad (5)$$

where erf is error function and $\sigma_1 \approx \sigma_2 = \sigma$.

Returning to our specific case study, we apply the false negative rate equation on the absolute of differences. More precisely, we calculate the false negative rate on the sum of the local maxima of the absolute differences Dg_j and $Dt_{s,j}$. Indeed, we found in Fig. 6 that the difference between genuine and infected AES traces are mainly located at the trace peaks. Therefore the local maxima are point of interest and HT detection is logically based on these point. By summing these local maxima, we can increase the HT detection probability. The computation of the false negative rate is the following:

- Calculate Dg_j and $Dt_{s,j}$.
- Find the local maxima of Dg_j and $Dt_{s,j}$.
- Sum the local maxima of each Dg_j and $Dt_{s,j}$.
- Compute the false negative rate on the sum of local maxima.

The false negative rates of HT occupying 0.5%, 1% and 1.7% of the AES area are respectively 26%, 17% and 5%. We can notice that the detection probability increases when the HT size increases. This is logical because the bigger the HT the more it contributes to the IC activity (hence larger EM emanations). We can also notice that for a HT 1 of size 0.5%, there is a high false negative rate of 26% because of process variations. For a HT 3 of size of 1.7%, the false negative rate decreases to 5% which is quite acceptable. In the state of the art, there are some obfuscation techniques used to force an attacker to increase the size of his HT hence improving the detection probability using this approach. So the HT can be detected even with the presence of process variations.

VI. Conclusion and Perspectives

In this paper, we studied the detection of HT implanted on FPGAs, without changing the placement and routing of the original circuit. This represents a proof of concept study for ASIC circuit where HT are added by untrusted chip manufacturers before the final fabrication of the chip. First of all, a novel HT detection approach using delay analysis us presented. This method is based on decreasing gradually the clock signal to create clock glitches. Then the IC critical paths delays for several bits are estimated with the faulted outputs and number of decreased clock steps. The HT detection relies on the defined delay model for a net. Second, an HT detection method using EM analysis is also introduced. It provides a better spatial and temporal resolution than power measurements. In these two methods, HT can be detected by comparing directly the delay/EM measurement between genuine and infected design using AES as target circuit.

Next, we tested HT of different sizes to estimate the detection probability as a function of its size taking into account the inter-die process variations. 8 different FPGA of the same reference (Xilinx *LX*30) are used to study the impact of inter-die process variations on this detection probability. We also introduce the metric to detect HT by exploiting EM techniques. This metric sums the local maxima of the absolute difference of the EM captured traces. It predicts the probability of HT detection with determined false positive and false negative rates. The results show that, using this metric, there is a probability greater than 95% with a false negative rate of 5% to detect a HT larger than 1.7% of the original circuit area.

Further extension of this work can include a more precise evaluation of impact of process variations on detection probability using both delay and EM measurements. This precision can be achieved by conducting the same experiments on n FPGAs, where $n \gg 8$.

Acknowledgments

This project has been funded by the French Government (BPI-OSEO), under grant FUI #14 HOMERE (Hardware trOjans : Menaces et robustEsse des ciRcuits intEgrés) and was partially supported by a DGA-MRIS scholarship.

References

[1] Miron Abramovici and Paul Bradley. Integrated circuit security: new threats and solutions. In *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies*, 2009.

[2] Michel Agoyan, Jean-Max Dutertre, David Naccache, Bruno Robisson, and Assia Tria. When clocks fail: On critical paths and clock faults. In *Smart Card Research and Advanced Application*, 2010.

[3] Mainak Banga and Michael S Hsiao. Odette: A non-scan design-for-test methodology for trojan detection in ics. In *Hardware-Oriented Security and Trust (HOST), 2011 IEEE International Symposium on*, 2011.

[4] Shivam Bhasin, Jean-Luc Danger, Sylvain Guilley, Xuan Thuy Ngo, and Laurent Sauvage. Hardware Trojan Horses in Cryptographic IP Cores. In Wieland Fischer and Jörn-Marc Schmidt, editors, *FDTC*, pages 15–29. IEEE, 2013.

[5] Shivam Bhasin, and Jean-Luc Danger Xuan Thuy Ngo, Sylvain Guilley, and Zakaria Najm. Encoding the State of Integrated Circuits: A Proactive and Reactive Protection against Hardware Trojans Horses. In *Proceedings of the 9th Workshop on Embedded Systems Security*, WESS '14, New York, NY, USA, October 17 2014. ACM. New Dehli, India. DOI: 10.1145/2668322.2668329.

[6] Keith A Bowman, Steven G Duvall, and James D Meindl. Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration. *Solid-State Circuits, IEEE Journal of*, 2002.

[7] Rajat Subhra Chakraborty, Somnath Paul, and Swarup Bhunia. On-demand transparency for improving hardware trojan detectability. 2008.

[8] U.S. Department Of Defense. Defense science board task force on high performance microchip supply.

[9] Yuval Ishai, Manoj Prabhakaran, Amit Sahai, and David Wagner. Private Circuits II: Keeping Secrets in Tamperable Circuits. In *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 308–327. Springer, May 28 – June 1 2006. St. Petersburg, Russia.

[10] Susmit Jha. Randomization based probabilistic approach to detect trojan circuits. In *High Assurance Systems Engineering Symposium, 2008. HASE 2008. 11th IEEE*, 2008.

[11] Yier Jin, Nathan Kupp, and Yiorgos Makris. Experiences in hardware trojan design and implementation. In *Hardware-Oriented Security and Trust, 2009. HOST'09. IEEE International Workshop on*, 2009.

[12] Yier Jin and Yiorgos Makris. Hardware trojan detection using path delay fingerprint. In *HOST 2008. IEEE International Workshop on*, 2008.

[13] Sebastian Kutzner, Axel Y Poschmann, and Marc Stöttinger. Hardware trojan design and detection: a practical evaluation. In *Proceedings of the Workshop on Embedded Systems Security*, 2013.

[14] Sergey Morozov, Abhranil Maiti, and Patrick Schaumont. An analysis of delay based puf implementations on fpga. 2010.

[15] Miodrag Potkonjak, Ani Nahapetian, Michael Nelson, and Tammara Massey. Hardware trojan horse detection using gate-level characterization. In *Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE*, 2009.

[16] Reza Rad, Jim Plusquellic, and Mohammad Tehranipoor. Sensitivity analysis to hardware trojans using power supply transient signals. In *Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on*, 2008.

[17] Xilinx. Synthesis Options (XST). http://www.xilinx.com/support/documentation/sw_manuals/xilinx11/pp_db_xst_synthesis_options.htm.

[18] Xuehui Zhang, Kan Xiao, and Mohammad Tehranipoor. Path-delay fingerprinting for identification of recovered ICs. In *DFT 2012, IEEE International Symposium on*, 2012.

Appendix

To evaluate the HT detection using delay and EM analysis, an AES 128 bits is used as the target circuit. This AES needs 10 rounds for one cipher computation. The proof-of-concept is done on FPGAs from Xilinx [17].

A. Delay Measurement

Delay measurement platform is composed of:

- Xilinx Spartan 3AN based FPGA board. The test chip's nominal clock period is 10 ns, and its core nominal voltage is 1.2 V.
- Xilinx Virtex V based FPGA board, used as an external clock.

B. Electromagnetic Measurement

EM measurement platform is composed of:

- FF324 Virtex 5 experimental board with a ZIF socket that allows to change the device under test (DUT).
- Xilinx FPGA Virtex 5 (LX30) fabricated in 65 nm technology node.
- Langer RFU-5-2 probe that captures the global EM activity of the chip.
- 30 dB Langer EMV power amplifier to amplify EM signal coming from the probe.
- Agilent 54853A infiniium DSO configured at 5 GS/s.
- Agilent E3631A stabilized power supply for the test board.