HAL
open science

IP Watermark Verification Based on Power Consumption Analysis

Cédric Marchand, Lilian Bossuet, Edward Jung

To cite this version:

Cédric Marchand, Lilian Bossuet, Edward Jung. IP Watermark Verification Based on Power Consumption Analysis. 27th IEEE International Systems-on-Chip Conference, SOCC 2014, Sep 2014, Las Vegas, United States. pp.330-335. ujm-01063085

HAL Id: ujm-01063085

https://ujm.hal.science/ujm-01063085v1

Submitted on 11 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

IP Watermark Verification Based on Power Consumption Analysis

Cédric Marchand, Lilian Bossuet
Laboratoire Hubert Curien, UMR CNRS 5516
University of Lyon
Saint-Etienne, France
{cedric.marchand,lilian.bossuet}@univ-st-etienne.fr

Edward Jung
School of Computing and Software Engineering
Southern Polytechnic State University
GA, USA
ejung@spsu.edu

***Abstract*—The increasing production costs of electronic devices and changes in the design methods of integrated circuits (ICs) has led to emerging threats in the microelectronics industry. Today, high value chips are the target of counterfeiting, theft and malicious hardware insertion (such as hardware trojans). Intellectual property (IP) protection has become a major concern and we propose to fight counterfeiting and theft by designing salutary hardware (salware). Instead of insert malicious effects inside an IP like a malware (*e.g.* a hardware trojan), a salware uses the same techniques, strategies and means for IP protection. One of the most studied salware is IP watermarking.**
Many works propose to target the finite state machine of digital IP to perform the watermarking. But, most of the time, the verification of the watermark is not clearly described. This conduces to a lack of credibility of these works. This paper proposes a watermark verification scheme using a correlation analysis based on the measurement of the IC power consumption. This article presents this process of verification and also discusses the selection of its parameters according to experimental results.

Keywords*—Hardware Security, salutary hardware, IP protection, IC counterfeiting, IP watermarking, side channel analysis, power consumption analysis.

I. Introduction

For several years, the microelectronic industry is facing the increase of costs of integrated circuits (ICs) production. This is due to the increasing complexity of systems and to the expensive technology refinement (e.g. the transition from 32 nm to 28 nm technology has been accompanied by a 40% increase in the manufacturing costs of wafers 300 mm in diameter and by a 30% increase in the manufacturing costs of 450 mm wafers). As a result, this industry has seen relocation of its production facilities and a sharp increase in the number of fabless companies (companies which do not produce ICs themselves). Time-to-market is increasingly tight and the increase of fabless companies has also led to a new method of ICs design, based on the reuse of intellectual property (IP) blocks. Thus, ICs manufactured today are produced with a high amount added value in a high competitive industry! All these changes has made electronic devices the target of counterfeiting, illegal cloning, theft and malicious hardware insertion (such as hardware trojans) [1].

The counterfeiting of ICs has become a major problem in recent years [2]. For example, the number of counterfeit electronic circuits seized by U.S. Customs between 2001 and 2011 has been multiplied by around 700 [3]. Between 2007 and 2010, U.S. Customs confiscated 5.6 million counterfeit electronic products [4]. Overall, counterfeiting is estimated to account for about 7% of the semiconductor market [5], which represents a loss of around US$ 10 billion per year for the lawful industry.

Design salutary hardwares (salwares) [6] is a way to protect IPs against these emerging threats. Salwares use the same techniques, strategies and means as malicious hardware (malwares) such as hardware trojans but instead of including some malicious effect inside an IC (secret leakage, malfunctions ...), salwares add protection to the chip. Exactly as hardware trojans, salwares are hardly detectable and difficult to remove. Examples of well-known salwares include physical unclonable functions (also known as PUF) for IC authentication, memory encryption, hardware metering, logic encryption, IC metering [7], remote activation [8] or IP watermarking [9].

For IP watermarking techniques, the verification of the watermark has two main objectives. The first one is the detection of illegal copies (illegal copies of an IP are also called clones). This case is detected when the creator of an IP find its watermark inside a product of another party who did not pay for the right to use the IP. In this case, the verification of the watermark can be used as proof in front of a court. The second objective of the watermark verification is to detect counterfeit IPs. This case is detected when an IPs without the mark is found among an set of ICs which contain the watermarked IP. The above arguments show that the verification method in watermarked IP research is as just as important as the watermark embedding method and the verification process must be precisely described.

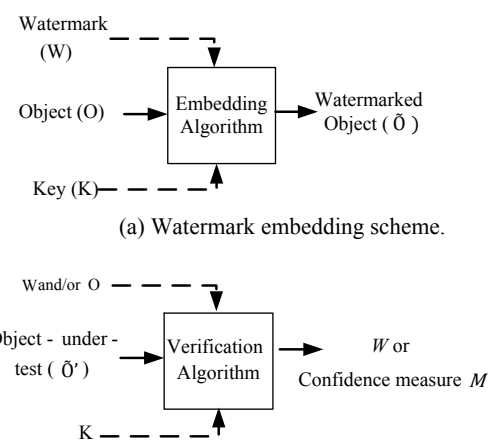
In this paper, we address the problem of verification of watermarked finite state machines (FSMs). The verification scheme of the IP watermark uses a correlation analysis based on the measurement of the power consumption of an IC. The power consumption is called the side channel (as it is used for cryptographic research [10]). In order to make this verification possible, a lightweight component which amplifies the side channel leakage is added to the IP. This component only highlight the state transition of the FSM by bringing non-linearity but does not interfere with the working FSM. In addition, it reduces the risk of collision between different IPs with the same FSM.

The rest of the paper is organized as follows: The next sec-

tion talks about IP watermarking and verification approaches. The Section III describes the correlation computation process used in the IP watermark verification. Section IV presents FSMs designed in order to perform IP watermark verification experiments and gather experimental results. Section V presents an analysis on different distinguishers used for the IP watermark verification. Finally, Section VI concludes the paper.

II. IP WATERMARKING

The concept of watermarking is well known for multimedia applications such as images, music or movies. The same approaches are used to insert a so called watermark inside digital IPs in order to provide a solution to protect the IP designer against counterfeiting and illegal copying. A survey on IP watermarking techniques is provided in [11] where requirements for general IP watermarking solutions are also presented. The usual method of watermarking is to insert the mark inside the FSM of an IP. In the related literature, the traditional way to do this is to add redundancy inside the FSM by adding new states and/or new transitions to the original FSM [12]. Thus, many FSM watermarking techniques are based on finding a solution to the graph partitioning [9] or to the graph coloring problem [13]. In [14], a new scheme of IP protection based on the extraction of specific FSM properties is proposed. Inserting the watermark inside the FSM of an IP has an advantage in that it makes it difficult to remove without damaging the functionality of the IP but it also makes it difficult to reforge a new valid watermark. In its most general form, the generic IP watermark scheme can be defined based on the embedding and detection (or verification) processes in Figure 1, similar to the model in [15].



(a) Watermark embedding scheme.

(b) Watermark detecting scheme.

Fig. 1. Generic IP watermarking scheme: an object (O) is transformed into a watermarked object (Õ) using a watermark (W) and/or a secret key (K) in the embedding process. In the detection process, an object-under-test (Õ') is verified, generally using the secret key (K), W, and/or O, depending on a watermark system [15]

For the verification of the inserted watermark, different approaches are also available. For example, it is possible to read the answer of the IC to a specific input sequence [16]. A side channel verification technique is proposed in [17] where specific inputs are also used to create a specific leakage. In [18], a side channel verification is performed using the power consumption of an IP as a physical hash function for the identification of the IP. These two last examples are very specific to cryptographic systems which are really different and possess very specific properties allowing the functions to be easily differentiated. The proposal of this article is based uniquely on the FSM of an IP. FSMs are the most important part of any digital synchronous IP. So, the proposed method is not application-dependent and can be adapted to any kind of digital systems which possess a FSM.

There are many reasons that it is very interesting to insert a watermark inside the FSM of an IP. First of all, the insertion can be performed automatically during the synthesis of the IP. In addition, it makes the watermark very difficult to remove because modifying complex FSM without damaging the intended system behavior is difficult. But, for this type of watermarking, the verification may be more difficult. Indeed, in many case, an access to state registers is needed and this access is not easy when the IP is embedded inside a complete integrated system. Thus, verification using measurement of the power consumption of an IC is the solution proposed in this article. This side channel contains a lot of information linked to the global switching activity of an IC (gates, registers, Inputs/Outputs...).

In order to prove that it is possible to access information about the FSM of a device via its power consumption, a new verification process is proposed in the following section.

III. SIDE CHANNEL VERIFICATION: CORRELATION COMPUTATION PROCESS

In this part, let's assume that the owner of a device wants to test if a device under test (DUT) contains the owner's watermarked IP. In order to verify the DUT's authenticity, the owner provides one device manufactured in a trusted manner - a reference device ($RefD$). The $RefD$ does not contains anything else that the original watermarked IP. The verification method proposed uses the correlation between the power consumption of the $RefD$ and the DUT to determine the DUT's authenticity. So, a correlation calculation process must be defined in order to perform the verification.

To verify that a DUT contains the same watermarked FSM as the $RefD$, a number n_1 of power consumption traces are measured on the $RefD$ and grouped in a set called T_{RefD}. A large number n_2 of power traces are also measured on the DUT and grouped in the set called T_{DUT}. Then k traces are selected in T_{RefD} using a function which randomly selects k distinct elements uniformly inside a set X (noted $\mathfrak{U}_X(k)$). This function can be defined as follows:

$$\forall k \in [\![1;n]\!], \quad \mathfrak{U}_X(k) = \{e_1, ..., e_k\}, \text{ such that } \forall (i,j) \in [\![1;k]\!], i \neq j \Leftrightarrow e_i \neq e_j$$

The mean of selected traces is calculated and used as a unique reference to the correlation computation. This averaged trace is noted A_{RefD} and defined as follow:

$$A_{RefD} = mean(\mathfrak{U}_{T_{RefD}}(k))$$

The same operation is repeated to calculate a number m of k-averaged traces with the set T_{DUT}. The set noted A_{DUT}

contains these m k-averaged traces and is defined by:

$$A_{DUT,m} = \{mean(\mathfrak{U}_{T_{DUT}}(k))\}_m$$

When all k-averaged traces are calculated, the correlation between A_{RefD} and each element of $A_{DUT,m}$ (an element of this set is noted $A_{DUT,m}(i)$ with $i \in [\![1; m]\!]$) is computed using the Pearson coefficient defined by:

$$\rho(x, y) = \frac{\sum_{i=1}^{l}(x_i - \overline{x}) \cdot (y_i - \overline{y})}{\sqrt{\sum_{i=1}^{l}(x_i - \overline{x})^2 \cdot \sum_{i=1}^{l}(y_i - \overline{y})^2}}$$

Where x and y are two traces of length l and $\overline{x}$ is the mean of x.

The result of this process is a set of m correlation coefficients which is noted $\mathcal{C}_{RefD,DUT,m,k}$.

It is this set $\mathcal{C}_{RefD,DUT,m,k}$ which is analyzed in order to answer the question: Does the DUT contains the same watermarked FSM as the $RefD$? In order to automatically authenticate that the DUT contains the $RefD$, it is necessary to use distinguishers. A discussion about which distinguishers can be used is performed in Section V? Section V also discusses about the choice of the parameters n_1, n_2, k and m of the correlation computation process. Note that only one k-average trace (A_{RefD}) is used as reference in this computation process; this ensures that all variations between the m elements of the set $\mathcal{C}_{RefD,DUT,m,k}$ are due anloy to the DUT and not to the $RefD$. This correlation computation process is presented as a flow schematic in Figure 2. With this illustration, it can

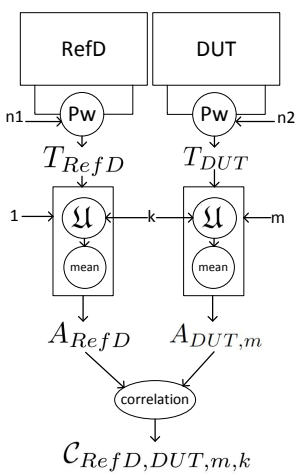


Fig. 2. Flow of the correlation computation process with one $RefD$, one DUT and the parameters n_1, n_2, k and m

be seen that this correlation process is a succession of three different functions leading to the final set $\mathcal{C}_{RefD,DUT,m,k}$:

1 A set of traces T_{device} ($device \in \{RefD, DUT\}$) is the result of a power signal acquisition (Pw) with the device and the number of measured traces (n) in parameter:

$$T_{device} = Pw(device, n)$$

2 A set of averaged traces $A_{device,m}$ is the result of a function mean which takes as one parameter a set of randomly chosen traces in the set T_{device} and m (representing the cardinal of the set $A_{device,m}$) as the other. For the random selection, the parameter is the number of averaged traces (k):

$$A_{device,m} = \{mean(\mathfrak{U}_{T_{device}}(k))\}_m$$

3 Finally, the set of correlation coefficients $\mathcal{C}_{RefD,DUT,m,k}$ for two devices ($RefD$ and DUT) is defined by:

$$\mathcal{C}_{RefD,DUT,m,k} = \{\rho(A_{RefD,1}, A_{DUT,m})\}$$

Note that when $m = 1$, the set $A_{device,1}$ is noted A_{device}.

IV. Experiment on IP Watermark Verification

A. Designed FSMs

In order to perform experiments on watermarked IP verification using the correlation analysis process described in Section III, four IPs are designed with two different types of counters as FSM. The use of counters is the worst case for the proposed verification scheme. Indeed, a counter is extremely linear, cyclic and the amount of information leaked by the power consumption signal is limited. So, by addressing the problem of IP watermark verification with these worst case kinds of FSMs, the effectiveness and the credibility of the proposed verification method is also proven for systems with more complex FSMs.

The first FSM is an 8-bit binary-counter and the second is an 8-bit Gray-counter. A side channel leakage component is added to these two FSMs. This component contains a watermark key (K_w) and a substitution table of the Advance Encryption Standard (AES) [19] called SBox. Substitution tables are strongly non-linear functions which replace each input by a specific predetermined output. This kinds of functions are commonly used in cryptography. Note that in this study, the SBox implementation is done in memory and the space required is 2^8 bits.

In order to verify that it is possible to identify different FSMs, the two first IPs are created by using the same randomly chosen value K_{w1} for the key in the two FSMs. This defines the two first IPs (IP_A and IP_B). By distinguishing between these two FSMs, the watermark verification proposed in Section III is proven for 2 different FSMs. To prove that the watermark key (K_w) reduces the risk of collision between two different IPs with the exact same FSM, two other IPs are created using the 8-bit Gray-counter with two different values for the watermark key (K_{w2} and K_{w3}). This defines two new IPs (IP_C and IP_D). Thus, four IPs are designed in order to prove that it is possible to verify an IP watermark that has been embedded without any addition of edge or state. Figure 3 shows the schematics of the four designed IPs.

These four FSMs are implemented inside four Altera Cyclone 3 FPGAs to create four $RefD$ (IP_A, IP_B, IP_C and IP_D). The same IPs are implemented inside four other FPGA Cyclone 3 in order to create four $DUTs$ ($DUT_{\#1}$, $DUT_{\#2}$, $DUT_{\#3}$ and $DUT_{\#4}$). Note that similar results are obtained by using only one FPGA to perform all measurements. According to this, the use of different FPGAs shows that the proposed work is insensitive to the CMOS variation process.

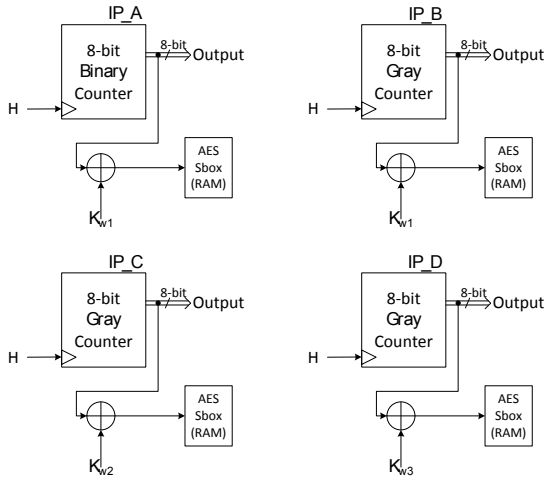


Fig. 3. Designed IPs schematics with an FSM and the side channel leakage component

In this experiment, all designed FSMs are strongly linear (8-bit counters) and the side channel leakage component helps the verification of an IP watermark because it brings some non linearity to the sequence of verification states. It is also important to note that it is not necessary to send specific input vectors to the designed IPs because they are all input independent. Furthermore, designed IPs are cyclic and it is possible to known exactly the periodicity of the designed FSM [14]. Thus, verification of watermarked FSMs is possible if the state sequence is long enough, *i.e.* the state sequence must be longer than the periodicity of the tested FSM. Nevertheless, the same input sequence is sent to the four IPs to ensure that the resulting state sequence has the same length in all tested IPs. In addition, the four FSMs are placed in the exact same state before starting any power consumption measurements.

B. Experimental results

The correlation computation process described in Section III is used for the four $RefDs$ with the four $DUTs$. For the IP_X ($X \in \{A, B, C, D\}$), the T_{IP_X} is created by measuring 400 power consumption traces. $10,000$ power traces are measured using the $DUT_{\#y}$ with $y \in \{1, 2, 3, 4\}$ in order to create the set $T_{DUT_{\#y}}$.

Then, using $k = 50$ and $m = 20$, A_{IP_X} and the set $A_{DUT_{\#y},20}$ are computed and defined by:

$$A_{IP_X} = mean(\mathfrak{U}_{T_{IP_X}}(50))$$

and

$$A_{DUT_{\#y},20} = \{mean(\mathfrak{U}_{T_{DUT_{\#y}}}(50))\}_{20}$$

Finally, $\mathcal{C}_{X,y,50,20}$ is created by calculating the correlation between A_{IP_X} and each element of $A_{DUT_{\#y}}$. In Figure 4, for $X \in \{A, B, C, D\}$ the sub-figure titled IP_X shows sets: $\mathcal{C}_{X,1,50,20}$, $\mathcal{C}_{X,2,50,20}$, $\mathcal{C}_{X,3,50,20}$ and $\mathcal{C}_{X,4,50,20}$.

To detect which DUT contains the same watermarked IP as each $RefD$, two distinguishers can be considered: the higher mean of the correlation and the lower variance of the correlation.

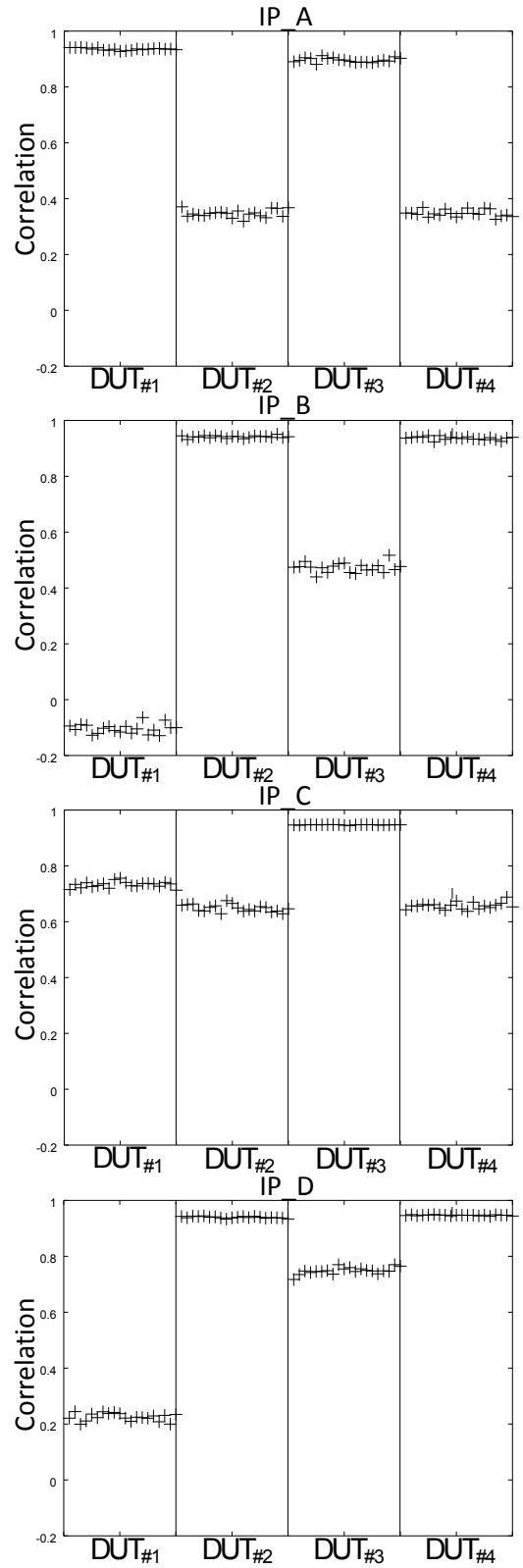


Fig. 4. Verification of watermarked IP (IP_A, IP_B, IP_C and IP_D) using the correlation computation process with $k = 50$ and $m = 20$.

If the higher mean of the correlation is considered as a distinguisher, according to Figure 4, it is clear that $DUT_{\#3}$ contains IP_C but even if it is not impossible, it is not

sure to determine which DUTs contains which IPs for IP_A, IP_B and IP_D. At the contrary, if the lower variance of the correlation is considered, it is clear that $DUT_{\#1}$ contains IP_A, $DUT_{\#2}$ contains IP_B, $DUT_{\#3}$ contains IP_C and $DUT_{\#4}$ contains IP_D. The next section will give a comprehensive analysis of the two distinguishers.

V. Analysis

In this section, an analysis of the two previously mentioned distinguishers (*e.g.* higher mean and lower variance of the correlation) is done and the choice of parameters $n1$, $n2$, k and m of the correlation computation process is discussed.

A. Distinguishers analysis

For this study, the mean of a set $\mathcal{C}_{X,y,k,m}$ is noted $\overline{\mathcal{C}_{X,y,k,m}}$ and the variance is noted $v(\mathcal{C}_{X,y,k,m})$. In order to analyze the mean of the correlation as a distinguisher, let's define a so called confidence distance that gives a percentage of confidence about a performed verification process (Figure 1). This value is defined for one IP_X ($X \in \{A, B, C, D\}$) by:

$$\Delta_{mean}(X) = 100 \times \left[1 - \frac{max_2(\{\overline{\mathcal{C}_{X,y,k,m}}, y \in \{1,2,3,4\}\})}{max(\{\overline{\mathcal{C}_{X,y,k,m}}, y \in \{1,2,3,4\}\})}\right]$$

where max_2 is the function which gives the second highest value in a set E. Table I shows the values of the mean of the different set $\mathcal{C}_{X,y,k,m}$ with $X \in \{A, B, C, D\}$ and $y \in \{1, 2, 3, 4\}$ and the mean confidence distance Δ_{mean} for each row.

TABLE I. Means of the different sets of correlation coefficients

	$DUT_{\#1}$	$DUT_{\#2}$	$DUT_{\#3}$	$DUT_{\#4}$	Δ_{mean}
IP_A	**0.936**	0.347	0.896	0.347	4%
IP_B	-0.104	**0.941**	0.473	0.936	0.52%
IP_C	0.733	0.648	**0.947**	0.657	22.6%
IP_D	0.225	0.940	0.748	**0.947**	0.78%

In the same way, the confidence distance of the variance of the correlation is defined to analyze the variance as a distinguisher. Let's define Δ_v by:

$$\Delta_v(X) = 100 \times \left[1 - \frac{min(\{v(\mathcal{C}_{X,y,k,m}), y \in \{1,2,3,4\}\})}{min_2(\{v(\mathcal{C}_{X,y,k,m}), y \in \{1,2,3,4\}\})}\right]$$

where min_2 is the function which returns the second smallest value in a set E. Table I shows the values of the variance of the different set $\mathcal{C}_{X,y,k,m}$ with $X \in \{A, B, C, D\}$ and $y \in \{1, 2, 3, 4\}$ and the variance confidence distance Δ_v for each row.

TABLE II. Variance of the different sets of correlation coefficients

	$DUT_{\#1}$	$DUT_{\#2}$	$DUT_{\#3}$	$DUT_{\#4}$	Δ_v
IP_A	$\mathbf{1.612e^{-5}}$	$1.831e^{-4}$	$6.443e^{-5}$	$1.477e^{-4}$	75%
IP_B	$2.925e^{-4}$	$\mathbf{1.928e^{-5}}$	$3.008e^{-4}$	$3.502e^{-5}$	44.9%
IP_C	$1.18e^{-4}$	$1.66e^{-4}$	$\mathbf{9.90e^{-7}}$	$1.47e^{-4}$	99.2%
IP_D	$1.91e^{-4}$	$1.04e^{-5}$	$1.53e^{-4}$	$\mathbf{3.04e^{-6}}$	70.66%

For both distinguishers (higher mean and lower variance of the correlation), the higher the confidence distance is, the better the distinguisher is. So, the right column of Table I and Table II show that the variance of computed correlation coefficients is a better distinguisher than the means of the correlation coefficients. Indeed, the confidence distance of the variance varies from 44.9% to 99.2% while the confidence distance of the mean varies from 0.52% to 22.6%.

B. Parameter choice

The first two parameters to choose are n_1 which is the size of the set T_{RefD} and n_2 which is the size of the set T_{DUT}. In order to choose these parameters, it is necessary to take into account the two other parameters k and m by respecting the following expressions:

$$n_1 \geq k \tag{1}$$

$$n_2 \geq k \times m \tag{2}$$

Expression 1 is required to compute the set A_{RefD} from the set of traces T_{RefD}. Expression 2 is required to compute the set A_{DUT} from the set T_{DUT}. Nevertheless, because traces in T_{DUT} are randomly chosen, it is important to choose n_2 high enough to minimize the probability of selecting one trace of T_{DUT} more than one time. The probability for one trace $t_i, i \in [\![1, n_2]\!]$ of the set T_{DUT} to appear in one selection of k traces is:

$$P(t_i) = \frac{k}{n_2}$$

Then, because n_2 depends upon k and m, it is possible to find a number $\alpha \in \mathbb{R}, \alpha \geq 1$ (Expression 2) such that:

$$n_2 = \alpha k m$$

By substituting this for n_2 by this in the previous expression, the probability to select the trace t_i in one selection becomes:

$$P(t_i) = \frac{1}{\alpha m}$$

Now, let's call ζ the following event:

For m selections, the trace t_i is selected more than one time.

The probability of ζ can be defined by:

$$P(\zeta) = 1 - \sum_{l=0}^{1} \binom{m}{l} P(t_i)(1 - P(t_i))$$

So, for m selections in T_{DUT}, the probability of ζ is:

$$P(\zeta) = 1 - (1 + \frac{m-1}{\alpha m})(1 - \frac{1}{\alpha m})^{m-1}$$

Looking at this, the first thing to note is that this probability does not depend on the parameter k. Next, let's note the last expression is a function ($f_\alpha(m)$). This function has two very interesting properties:

$$P1 : \forall m \in \mathbb{N}, \lim_{\alpha \to +\infty} f_\alpha(m) = 0.$$

$$P2 : \forall \alpha \geq 1, \lim_{m \to +\infty} f_\alpha(m) = 1 - (\frac{\alpha+1}{\alpha}) e^{\frac{-1}{\alpha}}.$$

The property $P1$ shows that m does not act on the probability $P(\zeta)$. Indeed, $P(\zeta)$ is only defined by the number α (property $P1$ and $P2$). So, in order to choose all parameters of the

correlation computation process described in Section III, the most important thing to choose is this probability $P(\zeta)$, because it allows to define all parameters. Once the probability is selected and thus, the number α too, the parameter m can be chosen to be as close as required of the limit of $f_\alpha(m)$.

For example, let's take $\alpha = 10$. Figure 5 shows the function $f_{10}(m)$ with its limit and a zone of available m to approach the limit at 5%. In this picture, it can be seen that $f_\alpha(m)$ reaches its limit quickly, so m does not have to be very large. Indeed, with $\alpha = 10$, approaching the limit at 5% means to select $m \geq 17$. Note that this characteristic is true for all value of $\alpha \geq 1$. In the experiment, Section IV, $\alpha = 10$ and $m = 20$, so the probability of the event ζ is fixed to: $P(\zeta) = 0.0045$.

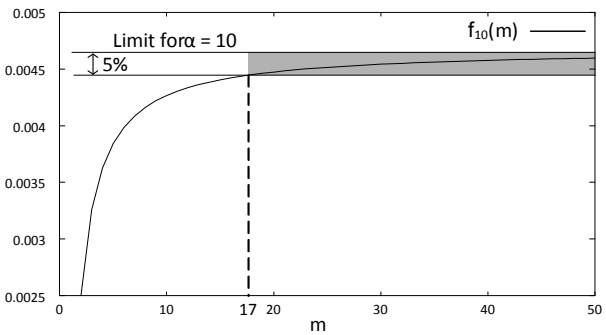


Fig. 5. $f_\alpha(m)$ for $\alpha = 10$, with the limit and 5% area for m.

In addition, values for k and m have not had a significant impact on the effectiveness of the proposed verification process which is characterized by the confidence distance presented in the Section V. Nevertheless, the parameter m has an impact on the computation time of the correlation, so its choice is a tradeoff between being very close to the selected probability and the time required to compute the correlation coefficient. For the last parameter (k), its choice determines the time of the power consumption acquisition. Indeed, because it has no impact on the probability of the event ζ, this parameter only impacts the time required for measurement. In section IV, the parameter k is set to 50. Thus, knowing α, m and k, the number of measured power traces on the DUT is: $n_2 = \alpha k m = 10000$.

VI. Conclusion

Designing salware is a way to fight against emerging threats brought on by the increasing cost of IC manufacturing (counterfeiting, theft, ...). A famous type of salware is IP watermarking. In this paper, a new watermark verification process using a correlation analysis based on the measurement of the power consumption of an IC is described.

Experimental results are presented and prove that it is possible to clearly identify different FSMs with the same watermark key (Kw) and the same FSM with a different watermark key too. Thus, our method is robust against some kinds of collisions. In addition, the verification scheme is insensitive to the CMOS process variation. Finally, a discussion about the choice of the correlation computation process shows that all parameters can be chosen by the selection of the probability of the event ζ described in the previous section. The advantage is that the selection of parameters does not significantly impact the effectiveness of the verification process.

Acknowledgment

The work presented in this paper was carried out using the framework of the SALWARE project number ANR-13-JS03-0003 supported by the French "Agence Nationale de la Recherche" and by the French "Fondation de Recherche pour l'Aéronautique et l'Espace".

References

[1] S. Narasimhan, R. S. Chakraborty, and S. Chakraborty, "Hardware ip protection during evaluation using embedded sequential trojan," *IEEE Design & Test of Computers*, vol. 29, no. 3, pp. 70–79, 2012.

[2] K. Huang, J. M. Carulli, and Y. Makris, "Counterfeit electronics: A rising threat in the semiconductor manufacturing industry," in *ITC*. IEEE Computer Society, 2013, pp. 1–4.

[3] C. Gorman, "Counterfeit chips on the rise," *Spectrum, IEEE*, vol. 49, no. 6, pp. 16–17, 2012.

[4] AGMA, "Alliance for gray markets and counterfeit adatement, http://www.agmaglobal.org."

[5] M. Pecht and S. Tiku, "Bogus: electronic manufacturing and consumers confront a rising tide of counterfeit electronics," *Spectrum, IEEE*, vol. 43, no. 5, pp. 37–46, 2006.

[6] L. Bossuet, D. Hely *et al.*, "Salware: Salutary hardware to design trusted ic." in *Workshop on Trustworthy Manufacturing and Utilization of Secure Devices, TRUDEVICE 2013*, 2013.

[7] R. Maes, D. Schellekens, P. Tuyls, and I. Verbauwhede, "Analysis and design of active ic metering schemes," in *HOST*, M. Tehranipoor and J. Plusquellic, Eds. IEEE Computer Society, 2009, pp. 74–81.

[8] Y. Alkabani, F. Koushanfar, and M. Potkonjak, "Remote activation of ics for piracy prevention and digital right management," in *ICCAD*, G. G. E. Gielen, Ed. IEEE, 2007, pp. 674–677.

[9] G. Wolfe, J. L. Wong, and M. Potkonjak, "Watermarking graph partitioning solutions," in *DAC*. ACM, 2001, pp. 486–489.

[10] P. C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *CRYPTO*, ser. Lecture Notes in Computer Science, M. J. Wiener, Ed., vol. 1666. Springer, 1999, pp. 388–397.

[11] A. T. Abdel-Hamid, S. Tahar, and E. M. Aboulhamid, "A survey on ip watermarking techniques," *Design Autom. for Emb. Sys.*, vol. 9, no. 3, pp. 211–227, 2004.

[12] I. Torunoglu and E. Charbon, "Watermarking-based copyright protection of sequential functions," *Solid-State Circuits, IEEE Journal of*, vol. 35, no. 3, pp. 434–440, 2000.

[13] G. Qu and M. Potkonjak, "Analysis of watermarking techniques for graph coloring problem," in *ICCAD*, 1998, pp. 190–193.

[14] E. Jung, C.-C. Hung, M. Yang, and S. Choi, "An locking and unlocking primitive function of fsm-modeled sequential systems based on extracting logical property," *Int. Journal of Information (INFORMATION)*, vol. 16, no. 8, 2012.

[15] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding-a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[16] B. L. Gal and L. Bossuet, "Automatic low-cost ip watermarking technique based on output mark insertions," *Design Autom. for Emb. Sys.*, vol. 16, no. 2, pp. 71–92, 2012.

[17] G. T. Becker, M. Kasper, A. Moradi, and C. Paar, "Side-channel based watermarks for integrated circuits," in *HOST*, J. Plusquellic and K. Mai, Eds. IEEE Computer Society, 2010, pp. 30–35.

[18] S. Kerckhof, F. Durvaux, F.-X. Standaert, and B. Gérard, "Intellectual property protection for fpga designs with soft physical hash functions: First experimental results," in *HOST*. IEEE, 2013, pp. 7–12.

[19] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*, ser. Information Security and Cryptography. Springer, 2002.