Golden Chip-Free Trojan Detection Leveraging Trojan Trigger's Side-Channel Fingerprinting

JIAJI HE, Tsinghua University
HAOCHENG MA, YANJIANG LIU, and YIQIANG ZHAO, Tianjin University

Hardware Trojans (HTs) have become a major threat for the integrated circuit industry and supply chain and have motivated numerous developments of HT detection schemes. Although the side-channel HT detection approach is among the most promising solutions, most of the previous methods require a trusted golden chip reference. Furthermore, detection accuracy is often influenced by environmental noise and process variations. In this article, a novel electromagnetic (EM) side-channel fingerprinting-based HT detection method is proposed. Different from previous methods, the proposed solution eliminates the requirement of a trusted golden fabricated chip. Rather, only the genuine RTL code is required to generate the EM signatures as references. A factor analysis method is utilized to extract the spectral features of the HT trigger's EM radiation, and then a *k*-means clustering method is applied for HT detection. Experimentation on two selected sets of Trust-Hub benchmarks has been performed on FPGA platforms, and the results show that the proposed framework can detect all dormant HTs with a high confidence level.

CCS Concepts: • **Security and privacy → Malicious design modifications**;

Additional Key Words and Phrases: Electromagnetic side channel, factor analysis, golden chip free, hardware Trojan detection, *k*-means clustering

ACM Reference format:
Jiaji He, Haocheng Ma, Yanjiang Liu, and Yiqiang Zhao. 2020. Golden Chip-Free Trojan Detection Leveraging Trojan Trigger's Side-Channel Fingerprinting. *ACM Trans. Embed. Comput. Syst.* 20, 1, Article 6 (December 2020), 18 pages.
https://doi.org/10.1145/3419105

1 INTRODUCTION

With an ever-growing need for cost reduction, globalization of the integrated circuit (IC) industry and supply chain, authenticity, and security of the ICs are exposed to several threats. Hardware Trojans (HTs) are malicious hardware modifications to ASICs, commercial-off-the-shelf (COTS) parts, microprocessors, digital signal processors, reconfigurable architectures, or IoTs [19, 37]. HTs have emerged as a major security concern for ICs that are employed in security-related situations,

This work was supported in part by the National Natural Science Foundation of China (grant 61832018) and the China Postdoctoral Science Foundation (grant 2019TQ0167).
Authors' addresses: J. He, Institute of Microelectronics, Tsinghua University, 30 Shuangqing Road, Haidian Qu, Beijing Shi, China; email: jiaji_he@mail.tsinghua.edu.cn; H. Ma, Y. Liu, and Y. Zhao, School of Microelectronics, Tianjin University, 92 Weijin Road, Nankai Qu, Tianjin Shi, China; emails: {hc_ma, yanjiang_liu, yq_zhao}@tju.edu.cn.

1539-9087/2020/12-ART6 $15.00
https://doi.org/10.1145/3419105

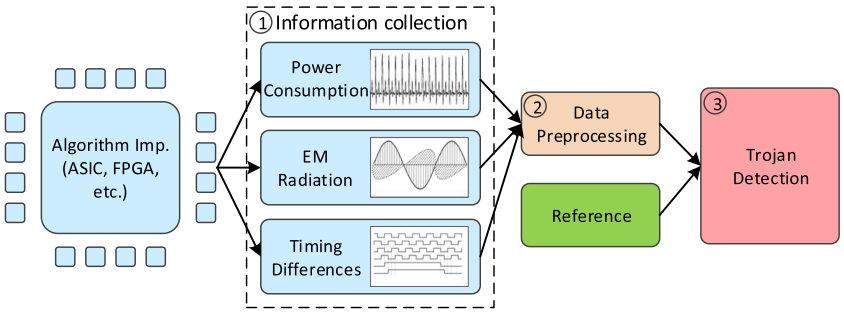

Fig. 1. A typical procedure of side-channel HT detection.

such as military, health care, aviation, communications, power management, and critical infrastructures.

After a decade of research efforts in this area, various HT detection and prevention methods have been proposed. These methods cover all stages of the IC lifecycle and are implemented in nearly all steps of the supply chain [28]. Compared with destructive methods, such as reverse engineering, side-channel-based methods are more flexible and easy to implement, where I/O ports and side-channel parameters are utilized to find abnormal behaviors. Although various HT detection approaches have been explored by many researchers, statistical side-channel analysis has been among the most heavily investigated. Side-channel analysis is based on the fact that any hardware modification that happens in a chip will impact side-channel information, such as power supply transient signals [32], leakage currents [1], time-window-based supply currents [29], path-delay analysis [2, 11], temperature [5], light [36], and electromagnetic (EM) radiation [3, 35].

For side-channel HT detection methods, there are several key steps, as shown in Figure 1. The first step is collecting side-channel information during the runtime execution of chips. The side-channel manifestations include but are not limited to power consumption, EM radiation, timing differences, and so forth. Note that it is difficult to use side-channel-based methods for HT detection if the HTs are tiny and their influences are covered by random noise. Therefore, this step is experiment dependent and based on the assumption that state-of-the-art measurement methods are precise enough for collecting the side-channel signal differences introduced by HTs. The second step is data preprocessing, including data integrating, denoising, and so forth. The purpose of this step is to raise the signal-to-noise ratio (SNR) and maintain the features of the original data. Process variations should be particularly considered because side-channel techniques suffer from detection accuracy issues in the context of process variations. The third step is HT detection, where various algorithms are leveraged. However, in the detection phase, most side-channel HT detection methods rely heavily on the existence of a reference, which is typically a trusted golden fabricated chip or other profiles alike, such as a golden layout. The absence of a reliable fabricated golden chip or golden layout invalidates practical applications of the detection approaches. Furthermore, many detection approaches require HTs to be activated, and works are trying to trigger specific HTs through particular stimuli, such as activation generation [41] and test generation [17]. However, to best ensure the security of the circuits, HTs should be detected before activation, but triggering an unknown HT is very difficult in real applications.

In this article, we propose a golden chip-free HT detection framework leveraging HT trigger's EM side-channel fingerprinting. We first build an EM side-channel radiation model (we refer to it as the EM model in the rest of the article) to generate the circuit's EM radiation using genuine RTL design. Optimization is made considering noise and variations to make the simulated EM model

match with real measurement in the frequency domain, and thus the EM model can be leveraged to evaluate the side-channel information of the circuit under test. Then the side-channel data is transformed into the frequency domain through Fast Fourier Transformation (FFT). Finally, to fully leverage the EM spectral features, a factor analysis algorithm is utilized to extract the EM signatures, then a *k*-means clustering algorithm is used for HT detection. Note that we do not require the HT to be activated in the detection phase, and we do not specify the stimuli for the circuits. The experiments are performed from two perspectives, where the EM modeling method is first assessed and then the HT detection is demonstrated. Two selected sets of Trust-Hub benchmarks, which are Advanced Encryption Standard (AES) and RSA benchmarks, are utilized for evaluation. Experiment results prove that the HT detection framework has excellent ability in finding various HT-infected circuits with a high confidence level.

Compared with previous works, this is one of the leading works in leveraging EM spectral features for HT detection with no constraints on the HTs' activation status, and further, the feasibility and capability of the model are also improved. In the previous model [15], only certain input vectors are simulated and the HTs are required to be activated for the effect of HT payloads to be observed. In addition, the previous method relies on the distinct differences of side-channel information introduced by activated HTs only. For the proposed model in this article, a large number of random input vectors are applied during the simulation and modeling process to generate the simulated traces. Leveraging the simulated traces, a comprehensive trusted EM side-channel bundle is obtained, and the corresponding details can be found in Section 3.1. The trusted bundle is more applicable as a golden reference for real HT detection because the dependence on input vectors is eliminated. Therefore, whichever input vectors are given to the chips under test, the trustworthiness of chips can be validated with only a few measured EM radiation traces utilizing the proposed model.

The main contributions of this work are as follows:

- A trusted EM model is established utilizing the RTL simulation data rather than performing extra simulations, and the generated EM signatures serve as the reference in EM side-channel HT detection.
- The EM model is calibrated with the consideration of various interference to match with actual measurements utilizing a non-linear regression function in the frequency domain.
- No specific knowledge of the HT's activation is needed for detection. Utilizing state-of-the-art testing and data processing techniques, potential dormant HTs are detected.

The rest of the article is organized as follows. Section 2 discusses different side-channel HT detection methods and the relationship between HTs and EM radiation. The overall methodologies for golden chip-free EM modeling and model calibration are proposed in Section 3. The HT detection algorithms are introduced in Section 4. The effectiveness of our method is validated through experiments in Section 5. Finally, Section 6 presents the conclusions.

2 PRELIMINARIES

2.1 Attack Model

An HT can be inserted at many different stages of the IC lifecycle, including RTL, gate-level netlist, layout, and so forth. For the in-house designed chips, we assume that only the RTL design is trusted, and we have access to the trusted RTL code. We primarily focus on the detection of HTs that are triggered by sequential logic, because even if the HTs' payloads remain silent, the trigger parts will still monitor the working status of the circuits and emanate EM radiation. Hence, HT detection turns into the detection of abnormal or extra sequential logic through the EM features.

Furthermore, we assume that the state-of-the-art measurement methods are precise enough for collecting the side-channel signal differences introduced by HTs.

2.2 Side-Channel HT Detection Methods

Nearly all side-channel-based HT detection methods require a trusted golden chip or layout. It is of the high cost to ensure whether a chip has been tampered with through reverse engineering analysis [4]. Several side-channel-based golden chip-free HT detection methods have been proposed recently. In the work of Rad et al. [32], although a golden chip is not needed, an HT-free layout is required to serve as the trusted model. In the work of Narasimhan et al. [29], the proposed method avoids the need for a golden chip, but it requires a good understanding of the HT activation, which is not applicable in real applications. A method without a "golden model" is presented in the work of Liu et al. [24], where measurements and trusted simulation models are combined to generate a trusted region. However, the requirement of a precise model of the process variations in the time domain makes the technique difficult to implement. In the work of Lecomte et al. [21], an on-chip detection method by monitoring the static distribution of the supply voltage over the IC's surface is proposed; however, the introduction of an array of sensors results in extra detection overhead. A self-reference-based HT detection method is proposed by Xue and Ren [40], where a genuine chip is not required; however, the circuit's partitioning is limited by the number of power pads, which will influence the detection accuracy. In the work of Zhang et al. [42], the golden-free detection method exploiting the bit power consistency of processor is proposed, but this method requires numerous power traces that are very time consuming to collect. Although the work of He et al. [15] verifies the feasibility of simulated EM signals as a golden reference, this method requires constant measurement until the HTs get activated. In real applications, if the circuit's input changes the EM radiation will also change, so this method is not feasible for real-time detection.

Another important part of the detection is data preprocessing. Raising SNR and reducing the influence of process variations are the key goals in data preprocessing. Because process variations, environment noise, and measurement noise all cause some misalignment, researchers have proposed strategies to tackle this problem. In the work of Hou et al. [16], the intrinsic relationship between transient current I_{DDT} and quiescent current I_{DDQ} of different test vectors is exploited to eliminate the effects of the process variations. A new approach that minimizes the effects of process variations on delay via calibration using test structures is proposed by Cha and Gupta [8]. In the work of Chen et al. [9], a very detailed process variation modeling algorithm is discussed for the proposed gate profiling technique for HT detection. In the work of Pino et al. [31], the ring oscillators are utilized to measure the within-die (intra-die) process variations. The overall local frequency varies ±1.22% without HTs due to within-die variations on the whole FPGA, whereas it varies up to approximately 8.77% due to the insertion of HTs. In [26], the process variations' influence on ring oscillator–based physical unclonable function is studied. The results show an average die-to-die (inter-die) Hamming distance of 47.13%, and an average within-die Hamming distance of 0.86% at the normal operating condition. However, all of the current process variations research focuses on structure-assisted or contact-type HT detection methods such as timing, leakage power, and gate profiling. Unlike other side-channel parameters, there is little research on EM radiation, and there is even less research on process variations' influence on EM side-channel information.

As for HT detection algorithms, starting with the use of the Euclidean distance and Markov distance for HT detection, many research studies are trying to find the differences between different side-channel data or trying to extract features. One-class support vector machines are used in the work of Jap et al. [18], and they can detect very small HTs using EM side-channel data. Neural network algorithms are utilized in the work of Li et al. [22] and Wang et al. [39] for extracting the

features of the data. All of the neural network–based HT detection algorithms require a training or learning process; however, it is unrealistic to predicate what the HTs should be like or to mimic the variegated HT features. He et al. [13] proposed to train the neural network with simulated genuine and HT-infected benchmarks. However, the proposed method may not be able to detect potential unknown Trojans, and the influences of non-ideal factors are not properly handled. In the work of Kulkarni et al. [20], an adaptive real-time HT detection framework is proposed using machine learning algorithms, but the method requires feedback from the core information and data packet. The ideal goal is that HT detection algorithms should be precise and self-improving, and the algorithms should coordinate with the side-channel model to achieve the best detection results. However, previous HT detection algorithms rarely consider the characteristics and compositions of the side-channel data when selecting HT detection algorithms. Chen et al. [10] leveraged the clock tree embedded in the FPGA for HT detection, and the results validated that the proposed framework is capable of detecting always on and already triggered HTs. Even with the help of state-of-the-art HT detection algorithms, available measured data from HT-free and HT-infected circuits are required to train a back propagation neural network, thus enabling the ability to distinguish HT-infected FPGAs from HT-free ones. However, it will still be very hard for the framework proposed to detect potential HTs that are not included in the training phase.

2.3 EM Radiation of HT

EM radiation arises as a consequence of current flows within control, I/O, data processing, or other parts inside a chip. The currents correlate with logical changes performed inside the chip, and EM radiation contains abundant spatial information. Besides the currents inside the chip, EM radiation is also influenced by several other parameters, such as the coupling between different emissions, the design layout of the chip, and the position of the metering probe. However, for side-channel-based HT detection, it is the direct near-field EM radiation from intended currents that induce the signals that are captured by near-field EM probes. So once the chip is manufactured and the experimental environment is set, the EM side-channel radiation is mainly determined by the function of the chip.

The RTL code of design has been used for power consumption calculation utilizing existing software like PrimeTime. Concerning the simulation of an IC's EM radiation, a few works have put forward some ideas using Hamming distance, Hamming weight, or an improved Hamming distance model [27, 38]. In the same way, the RTL code can be used for estimating EM radiation. He et al. used an RTL-based model for EM radiation simulation, and the generated *L-traces* can be utilized for side-channel security analysis. No matter what the layout or packaging of the chip will be, the signatures of EM radiation are determined by the function (i.e., the EM spectral features are decided by the RTL code). Although other factors can influence the EM radiation captured by EM probes, these factors cannot alter the EM spectral features. He et al. [15] proposed an EM modeling method for matching real EM side-channel radiation using the RTL code, but the model only works when the HTs are activated and the authors neglected the variations of the EM radiation caused by different input vectors.

In real chips, current only flows when there are changes in logic states, and thus the EM radiation carries information about the currents and hence the events and relevant states inside the chips. Furthermore, when different input vectors are applied on the chip, the events and relevant states will vary accordingly. HTs are modifications to original circuits, and HTs usually consist of trigger parts and payload parts. The trigger parts are usually abnormal or extra sequential logic, which typically has strong relations with clock signals, finite state machines, or state nodes in the original circuits. The payload parts are responsible for conducting malicious functions. Again, our goal is to detect the HTs before they are triggered. Even if the HTs' trigger parts remain silent, they will

still influence the currents that flow within the circuit, and thus they will affect the EM radiation of the circuit. Further, the structural changes in the circuit, which are introduced by HTs, will cause variations in leakage currents, which will also alter the EM radiation.

3 GOLDEN CHIP-FREE EM MODELING

In this section, we discuss the overall framework, working procedures, and algorithms included in the modeling methodology. Specifically, multiple input vectors are applied in the modeling stage to get the simulated EM radiation traces under different input vectors. To match the simulated traces with real measurements, a non-linear regression algorithm is utilized to compensate for the simulated traces toward real measurement.

3.1 RTL EM Model Construction

Based on Hamming distance, the simulated traces are modeled by the factors that contribute most to the EM radiation, such as data transitions and drive capabilities, whereas factors that have low impacts for direct radiations, like coupling effect, are ignored. Through optimum seeking of factors, major parts of the radiation that are caused by signal transitions can be modeled. Note that for ASIC implementations, more parameters, including the drive capabilities, interconnect capacitance, and so forth, should be taken into consideration. In this procedure, there exist differences between the simulated trace and the measured traces. We address this problem by transforming traces from the time domain into the frequency domain and comparing particular frequency spots. However, under different input vectors, the data transitions and driving capabilities also vary subsequently. Therefore, a large number of random input vectors are utilized in the modeling process.

Taking the FPGA implementation for example, under the input vectors V_m, where m represents the number of input vectors, the initial and final states of the i_{th} register/LUT are denoted as P_i and Q_i, respectively, and t represents the moment of the transition as every clock rising/falling edge. According to the linear relationship proposed by Brier et al. [7], the fan-out number of the i_{th} register or LUT can be denoted as D_i, and then the simulated EM side-channel trace $R(t)$ can be modeled as Equation (1), where $\oplus$ denotes the exclusive OR operation. The simulation of the $R(t)$ is illustrated in Algorithm 1; please note that $R(t)$ is not real side-channel radiation. Instead, $R(t)$ represents the chip's side-channel behaviors [33], which can be utilized to provide EM signatures as references for HT detection. Following the algorithm, the RTL implementations will first be synthesized to extract static circuit information including the logic components and drive capabilities. Under different stimuli, the RTL code is then simulated to collect dynamic information such as the switching activities and logic states of internal signals. The switching activities within the circuit will cause changes in register states. During each clock cycle, the states of signals under evaluation and their drive capabilities are collected. With all of the information, the final output (i.e., $R(t)$) is calculated. All results from Equation (1) are added up along the time axis to get the simulated trace in the time domain with all fan-out numbers as their weights.

$$R(t) = \sum_{i=1}^{n} D_i \times (P_i \oplus Q_i)\Big|_{V_m} \tag{1}$$

The sequential HTs that are driven by the clock signal or its division are specifically considered. If the original circuit is contaminated by HTs, the numbers of the registers and LUTs and their logic states are changed, resulting in the changes of the values of i_{th} and n in Equation (1). HTs are also connected with the original circuit, and HTs change the fan-out numbers of some registers and LUTs, resulting in the changes of the value of D_i in Equation (1). Further, the inserted HTs alter the internal nodes of the original circuit, which leads to the alterations of the values, including

ALGORITHM 1: $R(t)$ simulation

Input:
RTL_{imp} ▷ Original circuit implementation.
V_m ▷ Input vectors.
t ▷ Moment of the transition.
Output: $R(t)$ ▷ Simulated EM side-channel trace.
$logiccomponents \leftarrow RTL_{imp}$;
$DriveCapabilities \leftarrow logiccomponents$;
for Each t **do**
    $List_{LogicState} \leftarrow RTL_{imp}|V_m(t)$;
end for
$P_i, Q_i\ D_i \leftarrow list_{LogicState}$; ▷ HD model.
$R(t) \leftarrow P_i, Q_i, D_i$.

initial states, final states, or both. Specifically, HTs cause changes in the values of P_i and Q_i in Equation (1).

Equation (1) is only dedicated for EM radiation simulation under input vectors V_m. Because the input vectors have a major influence on the EM radiation, a large number of different input vectors need to be applied in the simulation process to build a trusted EM radiation bundle for real applications eliminating the input vector dependencies. The goal is to detect the HTs with very few measured traces from chips under test whether the HTs are activated or not. The simulated EM radiation bundle $R(T)$ is shown in Equation (2), where T represents the total time points in the simulation. The EM reference bundle in the time domain is obtained through Algorithm 2. The $T * m$ matrix is transformed into the frequency domain and later used as the golden reference for HT detection.

$$R(T) = \begin{bmatrix} R(t_1) = \sum_{i=1}^{n} D_{i1} \times (P_{i1} \oplus Q_{i1})\Big|_{V_1} \\ R(t_2) = \sum_{i=1}^{n} D_{i2} \times (P_{i2} \oplus Q_{i2})\Big|_{V_2} \\ \vdots \\ R(t_m) = \sum_{i=1}^{n} D_{im} \times (P_{im} \oplus Q_{im})\Big|_{V_m} \end{bmatrix}_{T*m} \tag{2}$$

3.2 Golden Chip-Free EM Side-Channel Model

The principal basis of golden chip-free HT detection methodologies is to find the differences between the simulated trace and the measured traces from chips under test. However, due to the influence of process variations, measurement noise, and environmental noise, even chips without HTs behave slightly differently concerning EM side-channel. Furthermore, with the shrinking feature size of ICs, process variation influence keeps increasing on the circuit's power consumption and EM radiation in the time domain. Thus, in the time domain, if the influence introduced by an HT is masked by process variations, its detection will be interfered with. However, in the frequency domain, the EM spectrum is largely determined by the clock signal and finite state machines, both of which are insensitive to process variations.

Due to process variations and measurement noise, the actual EM side-channel radiation of a chip $E(t)$ is composed of E_{ori}, E_{pv}, and E_n, which are EM leakage of the original circuit, EM leakage due to process variations, and EM leakage due to measurement noise, respectively. If an HT is inserted into the original circuit, there will be an additional element E_T, which is the HT's EM

ALGORITHM 2: Getting the trusted EM reference bundle

Input:
$RTL_{original}$ ▷ Original circuit description.
V_m ▷ Circuit input vectors.
t ▷ Moment of the transition.
Output: $R(T)$ ▷ The trusted EM radiation bundle.

$RTL_{synthesized} \leftarrow RTL_{original}$;
for Each V_m **do**
  for Each t **do**
    $list_{registers}, list_{LUTs} \leftarrow RTL_{synthesized}$;
    $D_{im} \leftarrow t, V_m$;
    $P_i, Q_i \leftarrow list_{registers}(i), list_{LUTs}(i), t$;
  end for
  $R(t_m) \leftarrow D_{im}, P_{im}, Q_{im}$;
end for
$R(T) \leftarrow R(t_m)$.

side-channel information. The EM traces of an HT-infected chip are formulated as Equation (3). The impact of the noise can be eliminated by denoising, and then the FFT operation can be applied on Equation (3) to transform into the frequency domain.

$$\overrightarrow{E(t)} = \overrightarrow{E_{ori}} + \overrightarrow{E_{pv}} + \overrightarrow{E_n} + \overrightarrow{E_T} \tag{3}$$

Meanwhile, process variations will not change the frequency point distribution of the EM side-channel spectrum. Let $f(n\Delta t), n = 0, 1, \ldots, N-1$ denote the EM signal in the time domain. Its frequency spectrum processed by FFT is written as Equation (4), where Δt and Δf are the sampling interval in the time domain and frequency domain, respectively.

$$\mathcal{F}(k\Delta f) = \sum_{n=0}^{N-1} f(n\Delta t) e^{-i(2\pi k\Delta f)(n\Delta t)}, k = 0, 1, \ldots, N-1 \tag{4}$$

According to the process variation model [26, 31], $\Delta f(n\Delta t) \sim N(0, \sigma^2)$, which is the effect of the process variations in the time domain assumed normal. Hence, the corresponding EM spectrum processed by FFT is modified as Equation (5). As shown, spectrum modifications $\mathcal{F}(k\Delta f)$ caused by the process variations are scattered into k frequency points. Therefore, only very small Gauss noises are introduced into the frequency range for HT detection. Thus, the differences caused by process variations in the amplitude of each frequency point should be very small compared with the original amplitude.

$$\mathcal{F}(k\Delta f) + \Delta\mathcal{F}(k\Delta f) = \sum_{n=0}^{N-1} [f(n\Delta t) + \Delta f(n\Delta t)] e^{-i(2\pi k\Delta f)(n\Delta t)} \tag{5}$$

Before we go into the frequency domain for HT detection, we need to address the influences caused by process variations. Unlike random noise, process variations' effect on side-channel measurement usually appears in a certain kind of pattern and can introduce much more obvious influences than noise. The golden chip-free EM side-channel model should be valid on different FPGAs, and thus the die-to-die process variations should be specifically considered. Assume on a wafer that there exist $L \times L$ different dies under test, which can be affected by process variations. Then the $L \times L$ matrix y denotes the EM signals with the presence of process variations collected from different dies. Discrete cosine transform is utilized [6] to build a relation between y and its

frequency-domain response Y using the 2D discrete cosine transform, denoted in Equation (6), where $u, v = 0, \ldots, L-1$. Note that our model solves the HT detection problem in the frequency domain. If FFT is applied on the right side of Equation (6), then the process variations' influences can only cause small mismatches in the low-energy part in the EM spectrum and cannot alter the frequency distribution of the spectrum.

$$Y(u,v) = \sum_{u=0}^{L-1} \sum_{v=0}^{L-1} y(u,v) \cdot cos\left[\frac{\pi}{L}\left(u+\frac{1}{2}\right)\right] \times cos\left[\frac{\pi}{L}\left(v+\frac{1}{2}\right)\right] \tag{6}$$

Theoretically, if FFT is applied on the simulated trace to get $\mathcal{F}(R(t))$, it will correlate well with $\mathcal{F}(E_{ori})$, and they will have many identical frequency points, as shown in Equation (7).

$$\begin{aligned} \mathcal{F}(E(t)) &= \mathcal{F}(E_{ori}) + \mathcal{F}(E_{pv}) + \mathcal{F}(E_T) \\ &= \mathcal{F}(E_{ori}) + \mathcal{F}(E_T) \\ &\cong \mathcal{F}(R(t)) + \mathcal{F}(E_T) \end{aligned} \tag{7}$$

3.3 HT EM Model Construction

The signal in the time domain is $R(t)$, and its corresponding expression in the frequency domain is $S(\omega)$. According to Fourier transform, we have Equation (8), where ω is its corresponding frequency.

$$S(\omega) = \int_{-\infty}^{+\infty} R(t) e^{-j\omega t} dt \tag{8}$$

The detailed composition of the $\mathcal{F}(E_{ori})$ signal captured by the probe includes the main clock and its harmonics, whose frequency can be denoted as $g_1, g_2, g_3 \cdots g_g$, respectively; some periodic signals generated by the circuits, whose frequency can be denoted as $f_1, f_2, f_3 \cdots f_f$, respectively; and other unintended signals, denoted as $U_1, U_2, U_3 \cdots U_u$, respectively. Assuming a sequential HT with signal transition frequency T_1 is inserted into the circuit, under the same circumstances and after FFT, the EM signals captured by the probe can be formulated as Equation (9), where A_{1i}, A_{2i}, A_{3i}, A_4, and A_5 denote the amplitude of each of the frequency components, respectively. Based on the heuristic observations, HT trigger parts and HT payload parts are taken into consideration in the model. $A_4S(jT_1)$ and $A_5S(jT_1)$ represent the contribution of the HT's trigger and payload, respectively. By decomposing the HTs into separate parts in our model, we can utilize the features of the HT trigger parts to detect HTs from the golden EM model. Before the HT is triggered, the trigger part $A_4S(jT_1)$ will always consume power and emit EM radiation. Note that due to the coupling effect of EM radiation, the insertion of HTs will potentially affect other circuit parts' EM radiation. The HT's payload part $A_5S(jT_1)$ will come in effect when the HT is activated.

$$\mathcal{F}(E(t)) = \sum_{i=1}^{f} A_{1i}S(jf_i) + \sum_{i=1}^{g} A_{2i}S(jg_i) + \sum_{i=1}^{o} A_{3i}S(jU_i) + A_4S(jT_1) + A_5S(jT_1) \tag{9}$$

3.4 HT Trigger EM Radiation Analysis

Based on the established HT EM model, the HT trigger parts and payload parts are separated. Before the payload parts are triggered, the trigger parts are always monitoring the working status of the circuits under certain input vectors. As discussed in Section 3.1, due to the dependency between input vectors and circuit status, different input vectors have a strong impact on the state of trigger parts and can further influence the EM radiation of the trigger parts. Once special status accumulates to the predetermined triggering conditions, the HT will be activated and execute malicious behaviors. However, the predetermined triggering conditions are not easy to reach, so the impact of the HT payload part $A_5S(jT_1)$ will be absent most of the time. Under these circumstances, the differences of the EM radiation between HT-free and HT-infected EM spectra are caused by the HT trigger part $A_4S(jT_1)$, and thus the differences can be analyzed for HT detection.

$$\begin{bmatrix} F(E_1(t)) \\ F(E_2(t)) \\ \cdots \\ F(E_n(t)) \end{bmatrix} = \begin{bmatrix} A_{11}^{(1)} & \cdots & A_{1f}^{(1)} & A_{21}^{(1)} & \cdots & A_{2g}^{(1)} & A_{31}^{(1)} & \cdots & A_{3u}^{(1)} \\ A_{11}^{(2)} & \cdots & A_{1f}^{(2)} & A_{21}^{(2)} & \cdots & A_{2g}^{(2)} & A_{31}^{(2)} & \cdots & A_{3u}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ A_{11}^{(n)} & \cdots & A_{1f}^{(n)} & A_{21}^{(n)} & \cdots & A_{2g}^{(n)} & A_{31}^{(n)} & \cdots & A_{3u}^{(n)} \end{bmatrix} \begin{bmatrix} S(jf_1) \\ \cdots \\ S(jf_f) \\ S(jg_1) \\ \cdots \\ S(jg_g) \\ S(jU_1) \\ \cdots \\ S(jU_u) \end{bmatrix} + \begin{bmatrix} A_4^{(1)}S(jT_1) \\ A_4^{(2)}S(jT_2) \\ \cdots \\ A_4^{(n)}S(jT_n) \end{bmatrix} \tag{10}$$

4 GOLDEN CHIP-FREE HT DETECTION

In this section, we discuss the algorithms used in HT detection. After the simulated EM traces and the measured EM traces from FPGA implementations are available, the EM spectra are analyzed to extract the features utilizing the factor analysis algorithm, then a k-means clustering algorithm is used to determine whether the chips under test are HT infected or not.

4.1 Factor Analysis–Based Feature Extraction

Factor analysis is an efficient statistical analysis method, which utilizes several common factors to construct enough information of original signals. The remaining specific factors after the decomposition represent the differences among signals, and thus factor analysis is extended to HT detection by comparing the differences of specific factors between the genuine circuit spectra and HT circuit spectra. According to the theory of factor analysis, for the dormant HTs, Equation (9) is rewritten as Equation (10) based on the factor model. $F(E_1(t)), F(E_2(t)) \cdots F(E_n(t))$ denote the n frequency spectra of simulated traces or measured traces, respectively. $S(jf_1) \cdots S(jf_f)$, $S(jg_1) \cdots S(jf_g)$, and $S(jU_1) \cdots S(jU_u)$ are the common factors of the traces. $A_4^{(1)}S(jT_1)$, $A_4^{(2)}S(jT_2) \cdots A_4^{(n)}S(jT_n)$ are the specific factors of $F(E_1(t)), F(E_2(t)) \cdots F(E_n(t))$.

To ensure the stealthiness of HTs, an intelligent adversary chooses the rare switching or occurrence as the trigger conditions, and thus the HTs are extremely difficult to activate under the excitation of random vectors. Due to the stealthy nature of the HTs, it will be very hard to directly distinguish the differences introduced by HTs through common factors between genuine circuit spectra and HT circuit spectra. However, except for common factors, there are still differences introduced by the trigger parts of HTs that can be utilized for HT detection. The specific factors of the HT trigger parts $A_4^{(1)}S(jT_1)$, $A_4^{(2)}S(jT_2) \cdots A_4^{(n)}S(jT_n)$ make it possible to identify the additional EM radiation of HTs—for instance, the HT detection turns into the detection of abnormal or extra sequential logic's EM radiation.

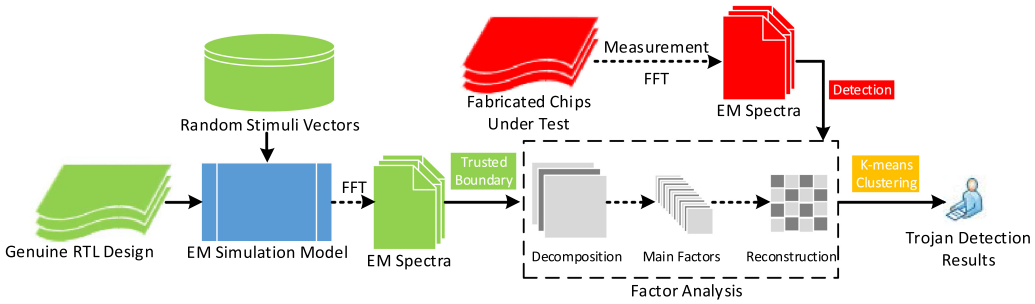

Fig. 2. The detection framework.

4.2 HT-Based *k*-Means Clustering

As mentioned, the HT problem is formulated as a two-class classification problem, and the clustering algorithm is introduced to classify chips under test as genuine or HT-infected ones. The *k*-means clustering algorithm is an unsupervised algorithm that is widely applied to various fields, such as data mining, pattern recognition, and decision support [23]. *k*-Means clustering method provides good fault tolerance because it uses the mean of samples as the centroid of each class and achieves high classification accuracy because it evaluates the clustering quality iteratively according to the cost function. The iteration is only finished until the cost function reaches the minimum value. In addition, it does not require the knowledge of HTs' implementations, and thus *k*-means clustering is suitable for identifying the existence of HTs. As we only have access to the trusted RTL code and the EM model, the *k*-means clustering algorithm is utilized to detect chips under test from a trusted golden reference. Any chip under test that has different spectral features is considered to be HT infected. The data is divided into different clusters iteratively using the cost function. The cost function f_c is expressed in Equation (11). This iteration process is repeated until the intra-cluster distance between data points of the same cluster is minimal and inter-cluster distance between data points of the same cluster is maximized. Where μ_j is the centroid of a cluster C_j, C_j is the jth sample of a given dataset $\chi = \{A_4^{(i)}S(jT), A_4^{(2)}S(jT) \cdots A_n^{(i)}S(jT)\}$, and k is the number of clusters.

$$f_c = \sum_{j=1}^{k} \sum_{\forall A_4^{(i)} S(jT) \in C_j} \left\| A_4^{(i)} S(jT) - \mu_j \right\|^2 \tag{11}$$

To evaluate the detection results of *k*-means clustering analysis, there are four values [12]: true-negative value (TN), false-positive value (FP), true-positive value (TP), and false-negative value (FN). TN shows the number of genuine spectra identified to be genuine spectra, FP shows the number of genuine spectra identified to be HT spectra mistakenly, TP shows the number of HT spectra identified to be HT spectra, and FN shows the number of HT spectra identified to be genuine spectra mistakenly. Further, there are a few more values to evaluate the detection and classification results: the true positive rate, the true negative rate, the precision, and the accuracy. The true-positive rate is defined by TP/(TP+FN), and the true-negative rate is defined by TN/(TN+FP). The precision, P, is defined by P = TP/(FP+TP). The accuracy, A, is defined by A = (TP+TN)/(TP+FP+TN+FN).

5 EXPERIMENTATION

In this section, the proposed framework is evaluated by detecting potential HTs in fabricated chips without a golden reference chip. The framework is demonstrated in Figure 2. The experimentation

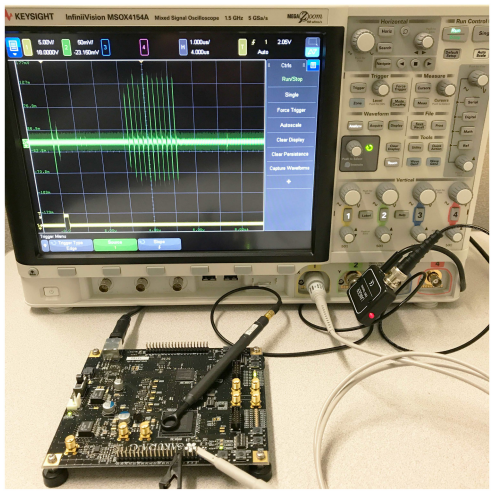

Fig. 3. Experiment setup.

is carried out in the following three steps. First, the performance of the EM radiation model is assessed in the frequency domain and calibrations are made. Then the factor analysis algorithm is utilized to extract signatures of the trusted EM reference bundle. Finally, the *k*-means clustering is utilized for HT detection with measured traces from the implementations of benchmarks under test.

5.1 Experimental Setup

The experiment platform is a SAKURA-G FPGA board specifically designed for research and development on hardware security. Two Spartan FPGAs, the controller FPGA (Device: XC6SLX9-2CSG225C), and the main FPGA (Device: XC6SLX75-2CSG484C) are integrated on the board. Although both FPGAs are built on a proven 45-nm technology node, the proposed method can be applied for HT detection on other FPGA boards with different technology nodes. The input operands are provided by the controller FPGA to the main FPGA. The main FPGA runs operations and will not be affected by other parts on the board. With the consideration that EM radiation is easily affected by differences in place and route, the Pblock [25] technique is applied to constrain the circuits into certain regions of the FPGA. Further, the EM probe is utilized to cover the whole Pblock part of the circuit's configurations, and the position of the EM probe is fixed during the whole experimentation to acquire EM radiations. After acquiring EM radiation by the probe, the signals are amplified using a pre-amplifier up to +30-dB magnification. Then the signals are captured and transferred to the host computer for further analysis. The experiment setup is shown in Figure 3. In the experimentation, we evaluated two selected sets of the cryptography benchmarks from Trust-Hub [34]. The benchmarks are a selected implementation of a 128-bit version of the AES block cipher and a basic 128-bit version of the RSA block cipher. There is a wide variety of implementations of HTs that attack the encryption circuits. The platform we utilized for data processing and results calculation has 24 Intel Xeon CPUs (X5690 @ 347 GHz) and 128 GB of RAM, and the OS is Red Hat Enterprise Linux Server release 5.6.

5.2 EM Model Assessment and Calibration

To validate the feasibility and consistency of the EM model with actual EM radiation, simulation traces are compared with actually measured traces in the frequency domain. The AES benchmark

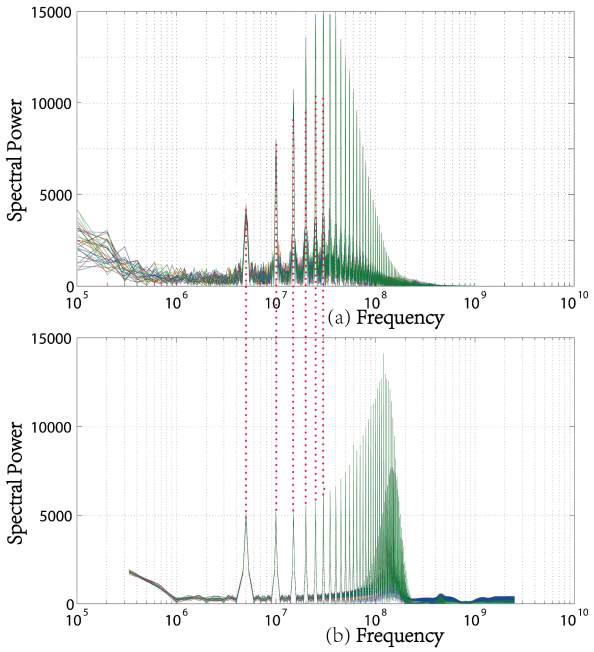

Fig. 4. EM model verification in the frequency domain. (a) EM spectra of measured traces. (b) EM spectra of simulated traces.

used in the experiment runs on 5-MHz frequency. Specifically, 50 simulated traces of the HT-free AES circuit from the EM model are randomly picked and transformed into the frequency domain. The HT-free AES circuit is configured on FPGA, then 50 measured traces are collected from the circuit's EM radiation and transformed into the frequency domain. As demonstrated in Figure 4, Figure 4(a) is the EM spectra of actual measured EM traces, whereas Figure 4(b) is the EM spectra of simulated EM traces. From results in the figures, there are several identical frequency points (greater than 82%) in both actual and simulated spectra, and the points are aligned on the red dashed line across the subfigures. The EM model matches with real implementations very well in the low-frequency band. However, there are two frequency bands where two spectra have few mismatches.

The first mismatch happens around 50 MHz in the actual spectra, where the amplitudes of frequency points are higher than those of simulated spectra. The difference is caused by the on-board external 48-MHz crystal oscillator that offers the clock signal for FPGA. However, as mentioned earlier, in the proposed EM model, only the signals that drive the circuit are taken into consideration. Hence, when experiments are carried out on the FPGA, measured traces are heavily affected by the external crystal oscillator. The mismatches introduced by the external oscillator need to be calibrated before real applications. A non-linear regression algorithm is utilized to compensate for the simulated data toward the measured data. The relationship $f : \overrightarrow{E_{sim}} \mapsto \overrightarrow{E_{mea}}$ of the measured data between the simulated data is established. More specifically, in the field of non-linear regression, the radial basis function network can establish the best-fit relationship among the variables with any given accuracy with enough neurons [30]. The frequency range selected in the calibration stage is 500 KHz $\sim$ 50 MHz. After calibration, the overall correlation coefficient is greater than 98%, and the distribution of coefficient is illustrated in Figure 5. Note that all simulated EM spectra are calibrated before using them as the golden reference.

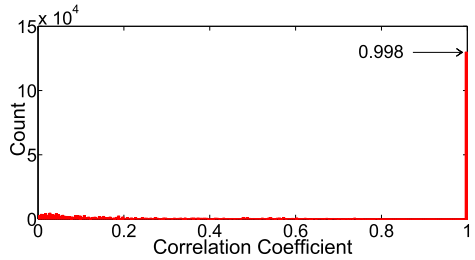

Fig. 5. Correlation coefficient after calibration.

Table 1. HT Detection Evaluation

Benchmarks	AES						RSA				
	Origin	T200	T1700	T1900	T2100	Overall	Origin	T100	T200	T300	Overall
Gate count	17,572	17,594	19,703	18,498	17,847	N/A	7,172	7,581	7,449	7,894	N/A
HT percentage (%)	0	0.12	12.13	5.27	1.56	N/A	0	5.71	2.47	10.06	N/A
Precision (%)	N/A	93.37	93.55	95.30	95.33	98.59	N/A	71.18	78.23	71.18	89.51
Accuracy (%)	N/A	79.97	80.91	94.12	94.44	82.52	N/A	65.67	77.67	65.67	65.17

N/A, data not applicable.

The second mismatch is in the high-frequency band over 120 MHz, where the amplitudes of simulated frequency points are higher. The reason is that the proposed EM simulation model calculates all logic state changes, as mentioned in Section 3.1, within the circuits; however, some of the logic state changes in the high-frequency band will not induce currents that contribute to EM radiation in actual chips. However, the preceding mismatch frequency does not influence the accuracy of HT detection, because the frequency points are concentrated in a high-frequency band, which exceeds our frequency of interest (500 KHz ∼ 50 MHz).

Overall, the actual spectra have more obvious variations than the simulated spectra, which are caused by noise and variations in real experiments. Therefore, the data preprocessing processes, such as denoising, are utilized to improve the performance of the proposed model. When measuring traces through experiments, the traces are averaged using the oscilloscope to eliminate most of the random noise. After the data is stored, further denoising is performed to achieve both noise reduction and data feature preservation, such as transients and abrupt changes. In this work, a wavelet transform is utilized to reduce noise and raise the SNR [15]. In addition, the normalization of the simulated and measured data is performed for the *k*-means clustering algorithm.

5.3 HT Detection

In this section, the details utilizing the *k*-means clustering algorithm for HT detection are demonstrated. The golden reference in the experiment is the EM spectra obtained through simulation after the calibration process. The chips under test in the experiment include the FPGA implementations of the genuine AES, genuine RSA, AES-HT benchmarks, and RSA-HT benchmarks. A detailed description of the benchmark circuits is demonstrated in Table 1. The genuine AES and RSA are the same circuits used in the EM model for generating the golden reference. Four AES and three RSA representative HT-infected benchmarks are chosen from Trust-Hub that perform malicious functionality to compromise the integrity of the circuit. For the AES benchmarks, the genuine AES represents the HT-free AES circuit, AES-T200 represents data-leak HTs through capacitance, AES-T1700 represents data-leak HTs through an antenna, AES-T1900 represents denial-of-service-type

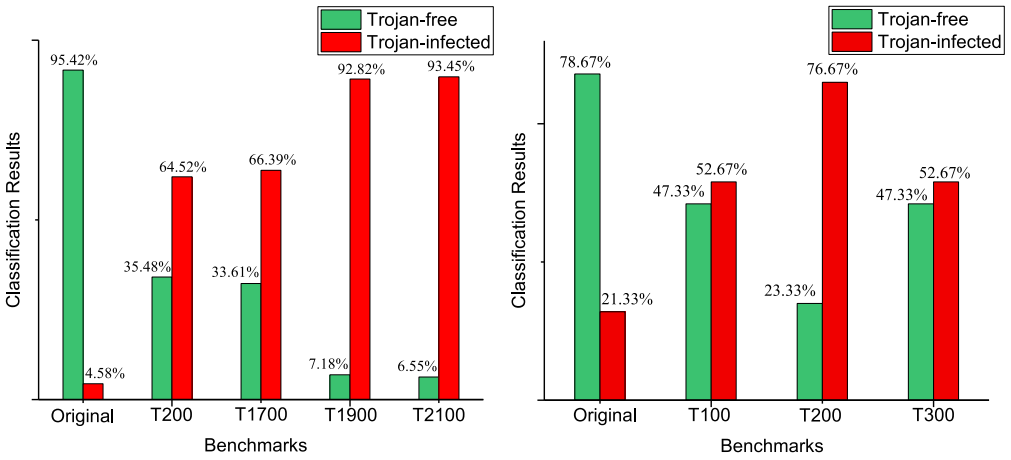

Fig. 6. HT detection results on different benchmarks. (a) AES benchmark. (b) RSA benchmark.

HTs, and AES-T2100 represents combinational and sequential data-leak HTs. For the RSA benchmarks, the genuine RSA represents the HT-free RSA circuit, RSA-T100 represents the data-leak HTs through dedicated output, RSA-T200 represents the denial-of-service-type HTs, and RSA-T300 represents the data-leak HTs through the data bus. Most of the HTs are relevant to clock signals of the HT-free circuit. For the HT-infected benchmarks, all HTs have sequential triggers.

Before the HT detection using the *k*-means clustering algorithm, the EM model is first evaluated from the HT detection perspective, in which the HT detection framework should be able to recognize the HT-free circuits. The golden reference EM spectra and measured original circuit spectra are compared. The results are demonstrated as the first stripe in Figure 6(a) and (b). Used in the evaluation phase are 1,000 measured spectra under random input vectors. It is clearly shown in the results that the majority of the measured spectra are classified into the HT-free class.

In the HT detection phase, 1,000 traces are measured of the HT-infected benchmarks each under random input vectors. Then the traces are transformed into the frequency domain for HT detection. For the AES benchmarks, more than 90% of the AES-T1900 and AES-T2100 spectra are classified into HT-infected class, meaning that the framework performs perfectly in finding these HT-infected circuits. For the AES-T200 and AES-T1700, more than 60% of the spectra are classified into the HT-infected class, and thus the results still provide a very high degree of confidence in finding HT-infected circuits. The HT detection results on RSA benchmarks are similar. For the RSA-T200 benchmark, more than 75% of the spectra are classified into the HT-infected class, whereas RSA-T100 and RSA-T300 have more than 50% of the spectra classified into the HT-infected class. Even though only 52.67% of the HTs are detected in RSA-T100 and RSA-T300 benchmarks, compared with the original circuit, we can still tell that the circuit is highly suspicious for infection with HTs. Note again that the HT detection results are achieved with no constraints on the HTs under random input vectors. Although the HT detection results have a relatively high false-negative rate, the framework performs well in the detection of HT-free original circuits with a high true-negative rate. In addition, it is clearly illustrated that there is a gap between the percentage of the detection results between the original benchmark and HT-infected benchmarks. From the overall results discussed earlier, we can conclude that the golden chip-free HT detection framework can distinguish whether the chips under test are the HT-infected circuit or not.

Table 1 also summarizes the specific detection results of various benchmarks. As shown, for the AES benchmarks, the precision of detection reaches greater than 93%, proving that the

framework has excellent ability in finding HT-infected circuits correctly. Meanwhile, the accuracy of the detection for overall types of HTs reaches 82.52%. Compared with the results in the work of He et al. [13], the detection accuracy is slightly lower by 7.49%; however, the decrease in detection accuracy is acceptable considering the different natures of these approaches. Please note that the original AES circuit's recognition rate is far higher than the results in the work of He et al. [13], where the recognition rate is improved from only 80% to greater than 95.42%. In addition, the detection results on AES-T1900 and AES-T2100 in this work are better than the results of He et al. [13]. Note that the detection results are also validated on RSA benchmarks. As for the relationship between HT detection accuracy and HT area, there are no accordant connections. For AES-T1700, the HT area is larger than AES-T200; however, the HT detection results are similar. The reason for this observation is that the AES-T1700's payload part is larger than the payload part of AES-T200. For AES-T1900 and AES-T2100, the HT trigger parts introduced distinct spectra features. The same conclusion can be applied to the RSA-T200 benchmark, where the HT trigger part has more distinct features than that of RSA-T100 and RSA-T300. The overall detection results are affirmative, as the detection results are achieved with no specific knowledge of the HTs, and the input vectors are random. The limitation of the proposed HT detection framework lies in whether the HT detection algorithm can distinguish the extra side-channel signatures introduced by HTs.

6 CONCLUSION AND FUTURE WORK

In this article, we proposed a simulation model utilizing the HT-free RTL code at the behavior level to generate a trusted EM side-channel reference bundle. Then the simulated bundle was used to compare with actual EM signals for model calibration and then HT detection leveraging a *k*-means clustering algorithm. The proposed HT detection framework gets rid of the requirements for a fabricated golden chip. The experimental results on FPGAs demonstrate that the HT detection framework can distinguish a genuine circuit from HT-infected circuits with high credibility even without knowledge of the potential HTs under random input vectors. Even if the Pblock technique is utilized to constrain the circuits into certain regions, there still exist minor differences of the configurations on FPGA, so for more precise localized EM measurements, this factor needs to be particularly considered. Alternatively, the proposed golden chip-free HT detection framework provides a very promising manner of HT detection. For future work, we plan to combine test generation methods with the framework to improve HT detection accuracy further and to validate the detection framework on combinational or mixed-type HTs. As our HT detection method happens in the frequency domain, the HT detection may be influenced by clock variance across multiple ICs and more experiments on different ICs will be performed.

REFERENCES

[1] J. Aarestad, D. Acharyya, R. Rad, and J. Plusquellic. 2010. Detecting Trojans through leakage current analysis using multiple supply pad I_{DDQ}s. *IEEE Transactions on Information Forensics and Security* 5, 4 (Dec. 2010), 893–904. DOI: https://doi.org/10.1109/TIFS.2010.2061228

[2] Atieh Amelian and Shahram Etemadi Borujeni. 2018. A side-channel analysis for hardware Trojan detection based on path delay measurement. *Journal of Circuits Systems and Computers* 27, 9 (Aug. 2018), 1850138. DOI: https://doi.org/10.1142/S0218126618501384

[3] J. Balasch, B. Gierlichs, and I. Verbauwhede. 2015. Electromagnetic circuit fingerprints for hardware Trojan detection. In *Proceedings of the 2015 IEEE International Symposium on Electromagnetic Compatibility (EMC'15)*. 246–251. DOI: https://doi.org/10.1109/ISEMC.2015.7256167

[4] C. Bao, D. Forte, and A. Srivastava. 2014. On application of one-class SVM to reverse engineering-based hardware Trojan detection. In *Proceedings of the 15th International Symposium on Quality Electronic Design*. 47–54. DOI: https://doi.org/10.1109/ISQED.2014.6783305

[5] C. Bao, D. Forte, and A. Srivastava. 2015. Temperature tracking: Toward robust run-time detection of hardware Trojans. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34, 10 (Oct. 2015), 1577–1585. DOI: https://doi.org/10.1109/TCAD.2015.2424929

[6] Barry R. Masters, Rafael C. Gonzalez, and Richard Woods. 2009. Book Review: *Digital Image Processing*, Third Edition. *Journal of Biomedical Optics* 14, 2 (2009), 029901. DOI: https://doi.org/10.1117/1.3115362

[7] Eric Brier, Christophe Clavier, and Francis Olivier. 2004. Correlation power analysis with a leakage model. In *Proceedings of the International Workshop on Cryptographic Hardware and Embedded Systems*. 16–29.

[8] B. Cha and S. K. Gupta. 2012. Efficient Trojan detection via calibration of process variations. In *Proceedings of the 2012 IEEE 21st Asian Test Symposium*. 355–361. DOI: https://doi.org/10.1109/ATS.2012.64

[9] X. Chen, L. Wang, Y. Wang, Y. Liu, and H. Yang. 2017. A general framework for hardware Trojan detection in digital circuits by statistical learning algorithms. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 36, 10 (Oct. 2017), 1633–1646. DOI: https://doi.org/10.1109/TCAD.2016.2638442

[10] Z. Chen, S. Guo, J. Wang, Y. Li, and Z. Lu. 2019. Toward FPGA security in IoT: A new detection technique for hardware Trojans. *IEEE Internet of Things Journal* 6, 4 (2019), 7061–7068.

[11] F. N. Esirci and A. A. Bayrakci. 2017. Hardware Trojan detection based on correlated path delays in defiance of variations with spatial correlations. In *Proceedings of the Design, Automation, and Test in Europe Conference and Exhibition (DATE'17)* 163–168. DOI: https://doi.org/10.23919/DATE.2017.7926976

[12] K. Hasegawa, M. Yanagisawa, and N. Togawa. 2017. Hardware Trojans classification for gate-level netlists using multi-layer neural networks. In *Proceedings of the 2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS'17)*. 227–232. DOI: https://doi.org/10.1109/IOLTS.2017.8046227

[13] Jiaji He, Yanjiang Liu, Yidong Yuan, Kai Hu, Xianzhao Xia, and Yiqiang Zhao. 2018. Golden chip free Trojan detection leveraging electromagnetic side channel fingerprinting. *IEICE Electronics Express* 16, 2 (2018), 20181065.

[14] Jiaji He, Haocheng Ma, Xiaolong Guo, Yiqiang Zhao, and Yier Jin. 2020. Design for EM side-channel security through quantitative assessment of RTL implementations. In *Proceedings of the 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC'20)*. IEEE, Los Alamitos, CA, 62–67.

[15] J. He, Y. Zhao, X. Guo, and Y. Jin. 2017. Hardware Trojan detection through chip-free electromagnetic side-channel statistical analysis. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25, 10 (Oct. 2017), 2939–2948. DOI: https://doi.org/10.1109/TVLSI.2017.2727985

[16] B. Hou, C. He, L. Wang, Y. En, and S. Xie. 2014. Hardware Trojan detection via current measurement: A method immune to process variation effects. In *Proceedings of the International Conference on Reliability, Maintainability, and Safety (ICRMS'14)*. 1039–1042. DOI: https://doi.org/10.1109/ICRMS.2014.7107361

[17] Y. Huang, S. Bhunia, and P. Mishra. 2018. Scalable test generation for Trojan detection using side channel analysis. *IEEE Transactions on Information Forensics and Security* 13, 11 (Nov. 2018), 2746–2760. DOI: https://doi.org/10.1109/TIFS.2018.2833059

[18] D. Jap, Wei He, and S. Bhasin. 2016. Supervised and unsupervised machine learning for side-channel based Trojan detection. In *Proceedings of the 2016 IEEE 27th International Conference on Application-Specific Systems, Architectures, and Processors (ASAP'16)*. 17–24. DOI: https://doi.org/10.1109/ASAP.2016.7760768

[19] Yier Jin. 2015. Introduction to hardware security. *Electronics* 4, 4 (2015), 763–784.

[20] A. Kulkarni, Y. Pino, and T. Mohsenin. 2016. Adaptive real-time Trojan detection framework through machine learning. In *Proceedings of the 2016 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'16)*. IEEE, Los Alamitos, CA, 120–123. DOI: https://doi.org/10.1109/HST.2016.7495568

[21] M. Lecomte, J. Fournier, and P. Maurine. 2017. An on-chip technique to detect hardware Trojans and assist counterfeit identification. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25, 12 (Dec. 2017), 3317–3330. DOI: https://doi.org/10.1109/TVLSI.2016.2627525

[22] Jun Li, Lin Ni, Jihua Chen, and E. Zhou. 2016. A novel hardware Trojan detection based on BP neural network. In *Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC'16)*. 2790–2794. DOI: https://doi.org/10.1109/CompComm.2016.7925206

[23] Hongfu Liu, Junjie Wu, Tongliang Liu, Dacheng Tao, and Yun Fu. 2017. Spectral ensemble clustering via weighted *k*-means: Theoretical and practical evidence. *IEEE Transactions on Knowledge & Data Engineering* 29, 5 (2017), 1129–1143.

[24] Yu Liu, Ke Huang, and Yiorgos Makris. 2014. Hardware Trojan detection through golden chip-free statistical side-channel fingerprinting. In *Proceedings of the 51st Annual Design Automation Conference (DAC'14)*. Article 155, 6 pages.

[25] Patrick Lysaght, Brandon Blodget, Jeff Mason, Jay Young, and Brendan Bridgford. 2006. Enhanced architectures, design methodologies and CAD tools for dynamic reconfiguration of Xilinx FPGAs. In *Proceedings of the 2006 International Conference on Field Programmable Logic and Applications*. IEEE, Los Alamitos, CA, 1–6.

[26] Abhranil Maiti, Jeff Casarona, Luke McHale, and Patrick Schaumont. 2010. A large scale characterization of RO-PUF. In *Proceedings of the 2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'10)*. IEEE, Los Alamitos, CA, 94–99.

[27] F. Menichelli, R. Menicocci, M. Olivieri, and A. Trifiletti. 2008. High-level side-channel attack modeling and simulation for security-critical systems on chips. *IEEE Transactions on Dependable and Secure Computing* 5, 3 (July 2008), 164–176. DOI: https://doi.org/10.1109/TDSC.2007.70234

[28] S. Moein, J. Subramnian, T. A. Gulliver, F. Gebali, and M. W. El-Kharashi. 2015. Classification of hardware Trojan detection techniques. In *Proceedings of the 2015 10th International Conference on Computer Engineering Systems (ICCES'15)*. 357–362. DOI: https://doi.org/10.1109/ICCES.2015.7393075

[29] S. Narasimhan, X. Wang, D. Du, R. S. Chakraborty, and S. Bhunia. 2011. TeSR: A robust Temporal Self-Referencing approach for hardware Trojan detection. In *Proceedings of the IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'11)*. Los Alamitos, CA, 71–74. DOI: https://doi.org/10.1109/HST.2011.5954999

[30] J. Park and I. W. Sandberg. 2014. Universal approximation using radial-basis-function networks. *Neural Computation* 3, 2 (2014), 246–257.

[31] Youngok Pino, Vinayaka Jyothi, and Matthew French. 2014. Intra-die process variation aware anomaly detection in FPGAs. In *Proceedings of the 2014 International Test Conference*. IEEE, Los Alamitos, CA, 1–6.

[32] R. Rad, J. Plusquellic, and M. Tehranipoor. 2008. Sensitivity analysis to hardware Trojans using power supply transient signals. In *Proceedings of the 2008 IEEE International Workshop on Hardware-Oriented Security and Trust (HOST'08)*. Los Alamitos, CA, 3–7.

[33] Siddika Berna Ors, Frank Grkaynak, Elisabeth Oswald, and Bart Preneel. 2004. Power-analysis attack on an ASIC AES implementation. In *Proceedings of the International Conference on Information Technology: Coding and Computing*. 546.

[34] Bicky Shakya, Tony He, Hassan Salmani, Domenic Forte, Swarup Bhunia, and Mark Tehranipoor. 2017. Benchmarking of hardware Trojans and maliciously affected circuits. *Journal of Hardware and Systems Security* 1, 1 (March 2017), 85–102. DOI: https://doi.org/10.1007/s41635-017-0001-6

[35] O. Söll, T. Korak, M. Muehlberghuber, and M. Hutter. 2014. EM-based detection of hardware Trojans on FPGAs. In *Proceedings of the IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'14)*. Los Alamitos, CA, 84–87. DOI: https://doi.org/10.1109/HST.2014.6855574

[36] F. Stellari, P. Song, A. J. Weger, J. Culp, A. Herbert, and D. Pfeiffer. 2014. Verification of untrusted chips using trusted layout and emission measurements. In *Proceedings of the IEEE International Symposium on Hardware-Oriented Security and Trust (HOST'14)*. Los Alamitos, CA, 19–24. DOI: https://doi.org/10.1109/HST.2014.6855562

[37] M. Tehranipoor and F. Koushanfar. 2010. A survey of hardware Trojan taxonomy and detection. *IEEE Design & Test of Computers* 27, 1 (Jan. 2010), 10–25. DOI: https://doi.org/10.1109/MDT.2010.7

[38] K. Tiri and I. Verbauwhecle. 2005. Simulation models for side-channel information leaks. In *Proceedings of the 2005 42nd Design Automation Conference*. 228–233.

[39] S. Wang, X. Dong, K. Sun, Q. Cui, D. Li, and C. He. 2016. Hardware Trojan detection based on ELM neural network. In *Proceedings of the 2016 1st IEEE International Conference on Computer Communication and the Internet (ICCCI'16)*. 400–403. DOI: https://doi.org/10.1109/CCI.2016.7778952

[40] H. Xue and S. Ren. 2018. Self-reference-based hardware Trojan detection. *IEEE Transactions on Semiconductor Manufacturing* 31, 1 (Feb. 2018), 2–11. DOI: https://doi.org/10.1109/TSM.2017.2763088

[41] M. Yoshimura, T. Bouyashiki, and T. Hosokawa. 2017. A hardware Trojan circuit detection method using activation sequence generations. In *Proceedings of the 2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC'17)*. 221–222. DOI: https://doi.org/10.1109/PRDC.2017.40

[42] Yang Zhang, Houde Quan, Xiongwei Li, and Kaiyan Chen. 2018. Golden-free processor hardware Trojan detection using bit power consistency analysis. *Journal of Electronic Testing* 34, 3 (2018), 305–312.

Received April 2020; revised August 2020; accepted August 2020