Scalable Consistency-based Hardware Trojan Detection and Diagnosis

Sheng Wei Miodrag Potkonjak
Computer Science Department
University of California, Los Angeles (UCLA)
Los Angeles, CA 90095
{shengwei, miodrag}@cs.ucla.edu

***Abstract*—Hardware Trojans (HTs) have become a major concern in modern IC industry, especially with the fast growth in IC outsourcing. HT detection and diagnosis are challenging due to the huge number of gates in modern IC designs and the high cost of testing. We propose a scalable and efficient HT detection and diagnosis scheme based on segmentation and consistency analysis of the gate-level properties. Furthermore, we develop a HT masking approach that prevents the HTs from functioning using selective device aging. We evaluate our HT detection and diagnosis schemes on a set of ISCAS and ITC benchmarks.**

I. Introduction

Hardware trojans (HTs) [1] are malicious attacks on integrated circuits (ICs) that modify the functionality or impact the performance of the ICs. HTs are possibly embedded by attackers during the manufacturing process in the form of additional gates or resized gates compared to the design specification. As IC outsourcing has become more and more popular recently, HT detection and diagnosis have become a necessity for IC designers and users, because the ICs are exposed to anyone who is in charge of the manufacturing process for potential HT attacks.

A typical procedure of handling HT attacks would include three strategic steps: HT detection, HT diagnosis, and HT masking. HT detection is the process that determines whether any HTs exist in the circuit. If there are any, a HT diagnosis approach is required to locate the HTs in the IC in terms of their types, locations, and input pins. After that, a HT masking mechanism should be conducted to disable the malicious functionalities of the HTs.

Although many HT detection approaches have been proposed recently [2][3][4], the HT diagnosis and masking problems were seldom discussed and addressed. Also, in the existing HT detection approaches, scalability has become a major concern, especially with the fast development of submicron technologies. It is challenging for the detection procedure to determine whether there is any HT embedded among millions of gates in the circuit. The cost of testing and running time have become a major concern in conducting this type of detection. Also, even if one can detect the presence of HTs accurately, it will cost much more effort to determine their exact locations in the circuit. Furthermore, the masking of HTs after detection and diagnosis has a potential of damaging the circuit.

We develop a complete and scalable solution of HT detection, diagnosis and masking using an aging and consistency-based gate characterization scheme. In particular, we divide a large IC into several overlapping segments and analyze the gate-level properties in each segment. We detect the HTs in the case where the overlapping gates exhibit inconsistent characterized properties in different segments. After confirming the existence of HTs, we further develop a consistency-based HT diagnosis scheme to refine the scope of the HTs to a small segment. Next, we adopt an IC aging method to disable the HTs while keeping the regular gates in the circuit unaltered. Our main technical contributions in this paper include the following:

- a scalable and efficient HT detection method based on segmentation and consistency analysis;
- a HT diagnosis scheme to determine the locations of the HTs in the circuit; and
- a HT masking method using selective device aging.

II. Related Work

In this section, we briefly review the directly related work in HT research and the supporting techniques regarding gate-level characterization (GLC) and IC aging.

A. Hardware Trojan Detection

Agrawal et al. [2] proposed one of the first HT detection techniques in 2007. They construct fingerprints using side channels (e.g., power and temperature) of the circuit for a specific design and authenticate the IC instances based on the fingerprints. The technique is based on the assumptions that there is no process variation, ICs are available for reverse engineering, and there are no measurement errors in the side channels.

Several early HT detection approaches employed functional test techniques. Functional tests simulate the input vectors on the circuit and monitor the outputs to see whether they match the expected patterns. For example, Wolff et al. [3] proposed the generation of test vectors that maximize the likelihood of detecting rarely switching HT gates. Also, Banga et al. [4] proposed automatic test pattern generation (ATPG) techniques that employ the divide-and-conquer paradigm. Recently, HT detection methods using side channel-based analysis have been developed [5][6][7][8]. They characterize the target ICs for

978-1-4577-0460-4/11/$26.00 ©2011 IEEE

their manifestational properties, such as delay and power, in order to detect the embedded HTs.

Tehranipoor et al. [1] presented a comprehensive survey of HT detection. There are two most common conceptual mistakes in the existing HT detection approaches: (1) the authors assume that both an IC with and without HT are available, and (2) all the gates have the same PV properties. In this paper, we do not impose these two assumptions for HT attacks and detections. Furthermore, we employ segmentation-based gate characterization into the process of HT detection, which ensures the scalability of the approach.

B. Gate-level Characterization

Gate-level characterization (GLC) is the process of identifying the process variation in the manufactured IC [9]. Recently, there are two classes of GLC methods that have been proposed for IC synthesis and analysis. The first class conducts physical measurements of transistor parameters [10]. The second class employs nondestructive methods that measure the manifestational properties (e.g., delay, leakage power, and switching power) of the entire IC and characterize the gate-level properties. For example, some of these techniques use sophisticated mathematical techniques, such as single value decomposition and compressive sensing [11], while others rely more on statistical processing of data obtained from systems of linear or nonlinear equations [12][13][14].

C. IC Aging

IC lifetime is influenced by a variety of phenomena that have been studied by the material science and semiconductor community, such as time dependent dielectric breakdown (TDDB) [15], thermal cycling (TC) [16], and negative bias temperature instability (NBTI) [17]. These phenomena are causing significant alterations of both delay and leakage characteristics of a gate. For example, aging can increase delay by 10% and leakage energy by several times [18]. Currently, aging has been assumed as having a detrimental impact on IC performance. There have been a large number of efforts to develop techniques that accurately predict the level of aging and its impact on IC lifetime. Our novelty in this domain is that we use IC aging to disable the malicious attack conducted by the HTs.

III. Preliminaries

In this section, we introduce the preliminaries of our HT detection, diagnosis, and masking approaches, including process variation, gate-level characterization, and IC aging models.

A. Process Variation

Process variation (PV) during IC manufacturing causes IC key parameters to vary from their nominal design specifications. For example, PV may vary leakage power by up to 20X and frequency by 30% on a single wafer [19]. In particular, there are two physical level properties that are major sources of PV: threshold voltage and effective channel length. For example, the effective channel length of a manufactured gate can be expressed by Equation (1), where L_{nom} is the nominal design value of the effective channel length, and ΔL is the variation caused by PV.

$$L_{eff} = L_{nom} + \Delta L \tag{1}$$

Several models have been proposed to capture the impact of PV, which formulate ΔL as a random distribution or a combination of multiple distributions to represent the spatial correlations on a chip, as well as the inter-chip variations. Although the accuracy of the PV models have been verified with the experimental data, they are only effective from the perspective of statistical properties, i.e., when a large number of chips are presented. For IC applications based on specific IC instances, such as those in hardware security, the PV models are not appropriate because of their lack of control over the individual chips.

B. Gate-level Characterization

In the process of GLC [9], the PV is represented as a scaling factor towards the gate-level manifestational properties, such as delay and power. Then, a system of linear equations can be obtained by summing up the gate-level properties and measuring the total power and delay. Taking leakage power as an example, the system of linear equations can be formulated as follows:

$$\tilde{p_j} = e_{sj} + e_{rj} + \sum_{i=1}^{n} K_{ij}\ \alpha_i \tag{2}$$

where $\tilde{p_j}$ is the total leakage power of the entire IC when input vector j is applied; α_i is a variable that represents the PV scaling factor of gate i; K_{ij} is the nominal leakage power of gate i when input vector j is applied; and e_{sj} and e_{rj} represent the systematic and random measurement errors, respectively. Following Equation (2), we obtain a system of linear equations by applying different input vector j and measuring the leakage power of the entire circuit. Then, we solve the system of linear equations using a linear programming (LP) solver, with an objective function that minimizes the measurement errors, to obtain the α values.

C. IC Aging Model

IC aging causes the threshold voltage of the transistors to increase and, consequently, the speed of the circuit to decrease. In particular, the threshold voltage shift caused by NBTI is a function of stress time, temperature, and applied gate voltage, as shown in the following equation [17]:

$$\Delta V_{th} = A \cdot exp(\beta V_G) \cdot exp(-E_\alpha/kT) \cdot t^{0.25} \tag{3}$$

where t is the stress time; T is the temperature; V_G is the applied gate voltage; A, β and k are constants; and E_α is the measured activation energy of the NBTI process. We employ this aging model to quantify the threshold voltage increase of the gates that are in the stress mode.

IV. Consistency-Based Hardware Trojan Detection

Our goal in this section is to address the detection of HTs using consistency analysis given the results of GLC. Our idea is based on the fact that a circuit containing HT would cause systematic bias in the total leakage power consumption, no matter where the HT is, how it is constructed, and even whether it is activated or not. With our GLC process, since there are no variables in the system of equations (shown in Equation (2)) to represent the HT, the systematic bias in the total leakage power would create inconsistencies in the equations, and the bias would be reflected in the scaling factors of regular gates in the circuit. By observing the bias in the leakage power scaling factors, we are able to detect HTs embedded in the circuit.

There are two key challenges with the consistency-based HT detection approach. Firstly, we do not assume that we have a clean circuit that does not have any HTs. Therefore, it is difficult to observe the bias in the scaling factors caused by HTs, as there is no standard scaling factors to compare with. Secondly, since the number of gates in modern IC designs is up to the magnitude of millions, the size of the system of equations would easily exceed the computational limit of the LP solvers.

We address both challenges using segmentation. The segmentation of an IC is based on the divide-and-conquer paradigm, in which we divide a large IC into multiple small segments and characterize each of them using GLC. Segmentation can be implemented using input vector control, where we freeze the signals of a subset of inputs and vary the other. Consequently, only the gates controlled by the varying inputs would change their coefficients in the system of linear equations, while the other gates would have identical coefficients in all the equations. Therefore, we can represent all the frozen gates using a single variable in the system of linear equations. In this way, the size of the LP is greatly reduced, to the extent that can be handled by LP solvers.

Furthermore, there are overlapping gates across segments. This provides us with an opportunity to characterize a single (overlapping) gate in multiple sub-circuits (segments), and thus observe possible bias in scaling factors due to the presence of HT. For example, suppose there are two segments A and B with a overlapping gate X, we can characterize the scaling factors of X in both segment A and B, namely α_a and α_b. Our idea is that α_a and α_b will be consistent if there is no HT present in either segment A or segment B, as ensured by the accuracy of GLC in both segments. In the case where HT exists in either A or B, there exists inconsistencies in the segment that contains the HT, and the resulting scaling factor (α_a or α_b) will be biased to reflect the inconsistencies. In the case where HTs exist in both segments A and B, since the two segments are different in terms of their gates and overall leakage power, the systematic bias caused by the HTs will be different in the two segments, which will again result in different values for α_a and α_b. We use the average discrepancy (d_{avg}) in calculated scaling factors of overlapping segments as an indicator of whether a HT is present or not. d_{avg} is calculated as the average standard deviation of the scaling factors of the same gate in the overlapping segments.

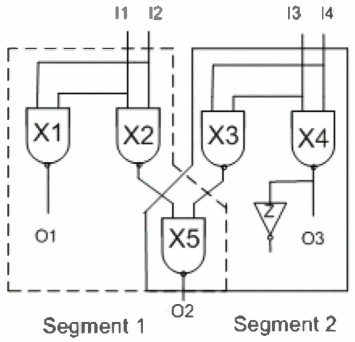
We illustrate our segmentation-based HT detection scheme using an example shown in Fig. 1. For the sake of brevity and clarity, the circuit has only five NAND gates (named $X1$ to $X5$) as shown in Fig. 1(a). We adopt normalized values as shown in Fig. 1(b) for their nominal leakage power. Our goal is to determine whether there is any HT embedded in the circuit. We first partition the circuit into two segments, as shown in Fig. 1(a). We obtain Segment 1 (gates $X1$, $X2$, and $X5$) by freezing inputs 3 and 4 and by applying different input vectors to inputs 1 and 2. Similarly, we obtain Segment 2 (gates $X3$, $X4$, and $X5$) by freezing inputs 1 and 2.

Next, we conduct GLC for each individual segment. In particular, we apply four input vectors to each segment that provide four sets of nominal leakage values for gates $X1$, $X2$, and $X5$ in Segment 1 and gates $X3$, $X4$, and $X5$ in Segment 2. For HT detection, we show three cases where HT exists or does not exist in Segment 1 and Segment 2. We assume that we do not know whether the circuit has HT in advance, and we form the system of linear equations to conduct GLC for each segment as shown in Fig. 1(c).

In case 1 (where HT is present in neither Segment 1 nor Segment 2), the values of overlapping gate $X5$ in the two segments are identical. In case 2 (where a single HT gate is present in Segment 1), the two calculated values of $X5$ have a 30.8% discrepancy. Finally, in case 3 (where HT is present in both Segment 1 and Segment 2), the values of $X5$ have a 3.7% discrepancy. These results indicate that the discrepancy between overlapping gates in multiple segments can serve as an indicator for the systematic bias in leakage power caused by embedded HTs. Therefore, we check the GLC results of overlapping gates between pairs of segments in the circuit. As long as the segments can cover all the gates in the circuit, our approach can detect any HTs embedded in the circuit. Furthermore, the use of segmentation ensures the scalability of GLC, since the number of gates being characterized in each system of linear equations is drastically reduced.

V. Consistency-based Hardware Trojan Diagnosis

The goal in HT diagnosis is to determine the locations of the HTs in the circuit if any exist, so that one can either remove or mask the HTs from the circuit. We design a scalable HT diagnosis scheme based on our consistency-based HT detection method. We have observed that one can detect the existence of HTs using two segments with overlapping gates. However, the HT detection results do not indicate which segment the HTs may be embedded in, and thus it is difficult for the HT masking process to handle the HTs. In order to diagnose the HTs, we introduce a third segment with the same set or subset of overlapping gates and use it as an arbiter for HT diagnosis. Fig. 2 shows an example of the consistency-based HT diagnosis. We find one more segment (Segment 3) compared to the example in Fig. 1. The three segments have an



(a) Segmentation

Gates X1 – X5

Input	Nominal Leakage
00	1
01	3
10	4
11	10

HT Gate Z

Input	Nominal Leakage
0	2
1	5

(b) Leakage power lookup table

Case 1: HT-free vs. HT-free

Input Vectors I1 I2 I3 I4	System of Equations (Segment 1)	Input Vectors I1 I2 I3 I4	System of Equations (Segment 2)
0000	$1 \alpha_1 + 1 \alpha_2 + 10 \alpha_5 = 15.3$	0000	$1 \alpha_3 + 1 \alpha_4 + 10 \alpha_5 = 15.4$
0100	$4 \alpha_1 + 3 \alpha_2 + 10 \alpha_5 = 21$	0001	$4 \alpha_3 + 3 \alpha_4 + 10 \alpha_5 = 21.35$
1000	$3 \alpha_1 + 4 \alpha_2 + 10 \alpha_5 = 21.1$	0010	$3 \alpha_3 + 4 \alpha_4 + 10 \alpha_5 = 21.45$
1100	$10 \alpha_1 + 10 \alpha_2 + 3 \alpha_5 = 26.9$	0011	$10 \alpha_3 + 10 \alpha_4 + 4 \alpha_5 = 29.2$
Results	$\alpha_1 = 1.1;\ \alpha_2 = 1.2;\ \alpha_5 = 1.3$	Results	$\alpha_3 = 1.15;\ \alpha_4 = 1.25;\ \alpha_5 = 1.3$

Case 2: HT-free vs. HT-present

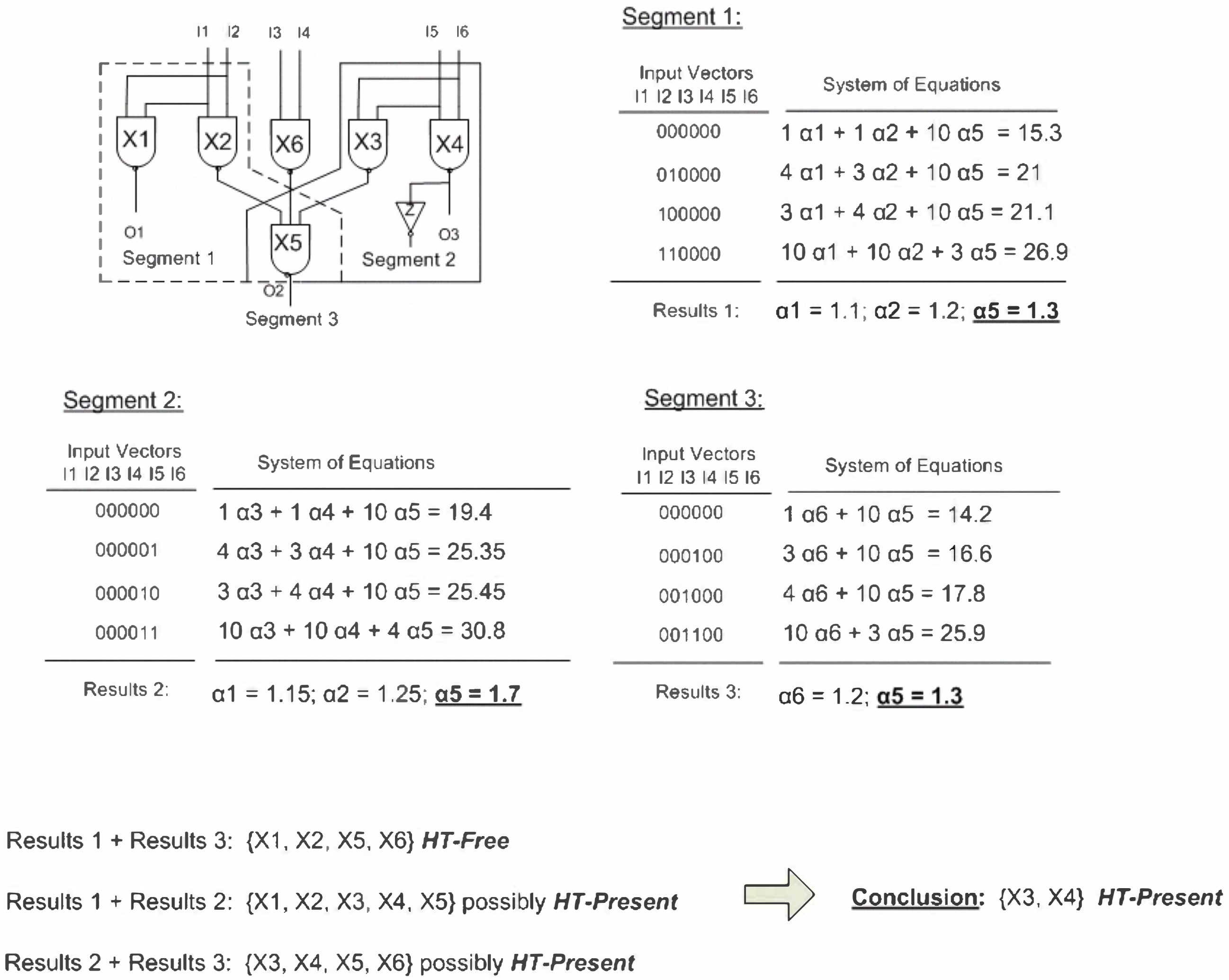
Input Vectors I1 I2 I3 I4	System of Equations (Segment 1)	Input Vectors I1 I2 I3 I4	System of Equations (Segment 2)
0000	$1 \alpha_1 + 1 \alpha_2 + 10 \alpha_5 = 15.3$	0000	$1 \alpha_3 + 1 \alpha_4 + 10 \alpha_5 = 19.4$
0100	$4 \alpha_1 + 3 \alpha_2 + 10 \alpha_5 = 21$	0001	$4 \alpha_3 + 3 \alpha_4 + 10 \alpha_5 = 25.35$
1000	$3 \alpha_1 + 4 \alpha_2 + 10 \alpha_5 = 21.1$	0010	$3 \alpha_3 + 4 \alpha_4 + 10 \alpha_5 = 25.45$
1100	$10 \alpha_1 + 10 \alpha_2 + 3 \alpha_5 = 26.9$	0011	$10 \alpha_3 + 10 \alpha_4 + 4 \alpha_5 = 30.8$
Results	$\alpha_1 = 1.1;\ \alpha_2 = 1.2;\ \alpha_5 = 1.3$	Results	$\alpha_3 = 1.15;\ \alpha_4 = 1.25;\ \alpha_5 = 1.7$

Case 3: HT-present vs. HT-present

Input Vectors I1 I2 I3 I4	System of Equations (Segment 1)	Input Vectors I1 I2 I3 I4	System of Equations (Segment 2)
0000	$1 \alpha_1 + 1 \alpha_2 + 10 \alpha_5 = 18.8$	0000	$1 \alpha_3 + 1 \alpha_4 + 10 \alpha_5 = 19.4$
0100	$4 \alpha_1 + 3 \alpha_2 + 10 \alpha_5 = 24.5$	0001	$4 \alpha_3 + 3 \alpha_4 + 10 \alpha_5 = 25.35$
1000	$3 \alpha_1 + 4 \alpha_2 + 10 \alpha_5 = 24.6$	0010	$3 \alpha_3 + 4 \alpha_4 + 10 \alpha_5 = 25.45$
1100	$10 \alpha_1 + 10 \alpha_2 + 3 \alpha_5 = 28.3$	0011	$10 \alpha_3 + 10 \alpha_4 + 4 \alpha_5 = 30.8$
Results	$\alpha_1 = 1.12;\ \alpha_2 = 1.21;\ \alpha_5 = 1.64$	Results	$\alpha_3 = 1.15;\ \alpha_4 = 1.25;\ \alpha_5 = 1.7$

(c) Formulation of systems of linear equations for HT detection

Fig. 1. Example of the segmentation-based HT detection approach: (a) shows that a circuit with five gates is segmented into two segments, and gate $X5$ is the overlapping gate of the two segments; (b) shows the nominal leakage power values for all the gates in the circuit; and (c) demonstrates the formulation of systems of linear equations and their solutions in three cases regarding whether a HT is present in each segment. The discrepancy of the solutions of overlapping gates ($X5$) in the two segments is an indicator of whether any HT exists or not.



Segment 1:

Input Vectors I1 I2 I3 I4 I5 I6	System of Equations
000000	$1\alpha 1 + 1\alpha 2 + 10\alpha 5 = 15.3$
010000	$4\alpha 1 + 3\alpha 2 + 10\alpha 5 = 21$
100000	$3\alpha 1 + 4\alpha 2 + 10\alpha 5 = 21.1$
110000	$10\alpha 1 + 10\alpha 2 + 3\alpha 5 = 26.9$
Results 1:	$\alpha 1 = 1.1$; $\alpha 2 = 1.2$; **$\alpha 5 = 1.3$**

Segment 2:

Input Vectors I1 I2 I3 I4 I5 I6	System of Equations
000000	$1\alpha 3 + 1\alpha 4 + 10\alpha 5 = 19.4$
000001	$4\alpha 3 + 3\alpha 4 + 10\alpha 5 = 25.35$
000010	$3\alpha 3 + 4\alpha 4 + 10\alpha 5 = 25.45$
000011	$10\alpha 3 + 10\alpha 4 + 4\alpha 5 = 30.8$
Results 2:	$\alpha 1 = 1.15$; $\alpha 2 = 1.25$; **$\alpha 5 = 1.7$**

Segment 3:

Input Vectors I1 I2 I3 I4 I5 I6	System of Equations
000000	$1\alpha 6 + 10\alpha 5 = 14.2$
000100	$3\alpha 6 + 10\alpha 5 = 16.6$
001000	$4\alpha 6 + 10\alpha 5 = 17.8$
001100	$10\alpha 6 + 3\alpha 5 = 25.9$
Results 3:	$\alpha 6 = 1.2$; **$\alpha 5 = 1.3$**

Results 1 + Results 3: {X1, X2, X5, X6} ***HT-Free***

Results 1 + Results 2: {X1, X2, X3, X4, X5} possibly ***HT-Present***

Results 2 + Results 3: {X3, X4, X5, X6} possibly ***HT-Present***

Conclusion: {X3, X4} ***HT-Present***

Fig. 2. Example of consistency-based HT diagnosis. We demonstrate the gate characterization in three segments with overlapping gates. The consistency in Segment 1 and Segment 3 exposes the possible HTs in Segment 2.

overlapping gate $X5$. We vary the controlling inputs of each segment and characterize the scaling factor of all the gates. In the case where the HT is embedded in Segment 2, we have the scaling factor of $X5$ consistent in Segment 1 and Segment 3 (e.g., $\alpha 5 = 1.3$), while that in Segment 2 has a different value (e.g., $\alpha 5 = 1.7$). Then, we analyze each combination of the pair of segments following the rule that an inconsistency in the scaling factor of the overlapping gate indicates possible HTs in either of the segments, while a consistent result ensures that both of the involved segments are HT-free. For example, as shown in Fig. 2, we conclude that the HTs are present in Segment 2 (i.e., gates X3 and X4).

Pseudocode 1 describes the detailed procedure of the consistency-based HT diagnosis. In each round of the diagnosis, we first characterize three segments with at least one overlapping gate. Then, we compare the scaling factor values of the overlapping gate obtained from the three segments. The one that has a large difference compared to the other two values is in the segment that is possibly HT-present. In the case where all three scaling factor values have large difference compared to the others, we conclude that multiple HTs are embedded in at least two segments and find more segments that cover the overlapping gates to further diagnose the HTs.

VI. Aging-based Hardware Trojan Masking

After identifying the locations of the HTs, we must find a way to either remove them from the circuit or disable them so that they may not conduct malicious attacks to the target IC. Since physical methods to remove a particular component from an IC are very expensive to apply and have a potential of damaging the normal parts of the IC, we design a non-destructive approach to disable the functionalities of the HTs. Our key idea is to utilize proactive IC aging that increases only the threshold voltage of the HTs while maintaining the other gates on the circuit unaffected. Consequently, when the threshold voltages increase to an extend that saturates the performance (speed) of the HTs, they are assumed nonfunctional in terms of the malicious attacks. Even for the high leakage energy attack, in

Pseudocode 1 Consistency-based HT diagnosis.

```
Input: Target circuit with embedded HTs;
Ouput: Segment set $Seg$, which contains all the segments
    that are HT-present;
Detect the existence of HTs;
Search for $S$, the three-segment set that covers all the
    gates in the circuit;
for each $S_i$ in $S$ do
    for $j = 1 \rightarrow 3$ do
        Characterize Segment $S_{ij}$ and obtain scaling factor
            $\alpha_j$ for the overlapping gate;
    end for
    $d_1 = min\{|\alpha_1 - \alpha_2|, |\alpha_1 - \alpha_3|\}$;
    $d_2 = min\{|\alpha_2 - \alpha_1|, |\alpha_2 - \alpha_3|\}$;
    $d_3 = min\{|\alpha_3 - \alpha_1|, |\alpha_3 - \alpha_2|\}$;
    $h = argmax\{d_1, d_2, d_3\}$;
    Insert $S_{ih}$ into $Seg$;
end for
return $Seg$;
```

which an attacker attempts to leak a large amount of energy during IC operation, once the threshold voltage increases to a value close to the supply voltage, the additional leakage energy caused by the HTs becomes negligible.

A. SAT-based IC aging

One of the challenges in aging-based HT masking is how to ensure that only the HT gates are stressed by the applied input vectors, while the regular gates on the circuit should not be constantly under stress. We address this issue by defining and solving a Satisfiability (SAT) problem that searches for input vectors to set only the HT gates in the stress mode (i.e., signal 1) and the normal gates unstressed (i.e., signal 0).

SAT is a problem that determines if a set of variables can be assigned to satisfy a boolean formula. In the IC domain, if the netlist of a circuit is known, the signal of each gate can be expressed as a boolean formula with a set of primary input signals as the variables. Therefore, the input vector selection problem that aims to set a specific gate or a set of gates to specific signals can be naturally converted to a SAT problem. By solving the SAT problem, we can provably find the desirable input vectors for aging based on our requirements regarding the gate signals.

SAT has been proved as one of the first known examples of NP-Complete problems. Recently, there have been many SAT solvers developed in the SAT community [20] that deliver fast and accurate SAT solutions. In this paper, we do not discuss the details of SAT solving. Instead, we mainly focus on how we use SAT solving techniques to address the problem of input vector selection for aging the HT gates.

In our SAT problem formulation, we use an objective file to specify the signals of a subset of gates that we are obtaining input vectors for. In particular, the objectives in the SAT problem follow the following format:

$$obj_i = 0|1, i = 1...k \tag{4}$$

where obj_i is corresponding to a gate ID in the circuit netlist, and k is the number of gates that we expect to specify signal 0 or 1 for. If the SAT problem is satisfiable, the output from the SAT solver is a list of input vectors that satisfies the objectives.

We demonstrate the SAT problem formulation for HT masking using a small example in Fig. 3. For the clarity of discussion, we consider only a small circuit with four AND gates and two NAND gates. In this example, gate 6 is the HT embedded by an attacker. In the SAT objective file, we set the HT gate (gate 6) to signal 1 and all the regular gates (gates 1-5) to signal 0. The SAT solver outputs the input vector 00111 that satisfies the specified objectives. Then, we apply the input vector 00111 to the circuit, which constantly pushes the HT gate in the stress mode.

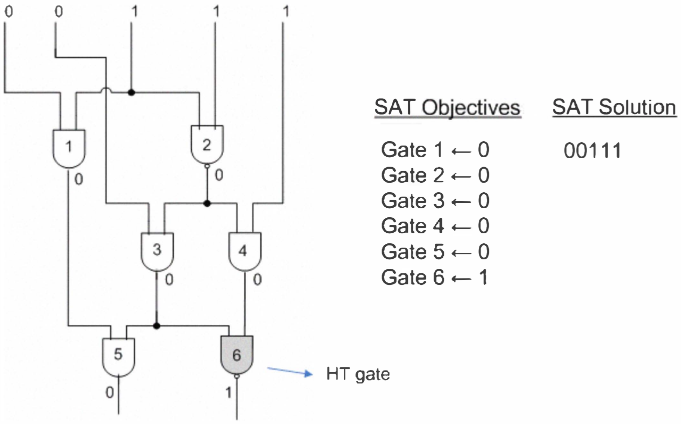


Fig. 3. Example of SAT formulation for HT masking. The SAT objectives are formulated to set the HT gate (gate 6) to signal 1 and the regular gates (gates 1-5) to signal 0. The output from the SAT solver provides the input vectors that satisfy the objectives.

B. Input Vector Selection

The major issue in applying SAT-based approach for aging input vector selection is that the SAT problem is often unsatisfiable, under the strict SAT objectives that all HT gates are set to signal 1 and all regular gates are set to signal 0. This is due to the internal structure of the circuit (netlist) and the correlations between the gates. In this case, we must design an input vector selection scheme on top of the SAT problem formulation, which ensures that all the HTs are disabled and the impact on the regular gates is minimized. We achieve this goal by applying three technical approaches in three different phases, namely weight assignment, LP-based input vector selection, and adaptive body bias compensation.

1) Weight-based Iterative SAT Solving: In the case where the strict SAT formulation is not satisfiable, we first divide the gates into two groups based on their importance in terms of the performance and energy consumption. One group, called CP-gates, includes all the gates that are on the critical path (e.g., the path that has the longest delay and determines the delay of the circuit). The other group, called non-CP gates, includes all the gates that are off the critical path. The CP gates are more important because of the fact that they determine the delay of the IC directly. The non-CP gates are relatively less important because their delay increases would not impact the delay of the IC as long as they do not become CP gates. Based on this intuition, we first relax the objectives of setting non-CP gates in the case where the strict SAT formulation is not satisfiable.

Within each group, either the CP-gates or non-CP gates, we further sort the gates based on their importance. Here the importance of a gate is quantified by the value of delay increase caused by aging. For example, if the delay of a gate would increase by a high value due to aging, we should avoid aging it. In order to implement this key idea, we assign a weight factor to each gate using the following formula, which can be determined via simulation:

$$Weight_j = \frac{1}{m}\sum_{i=1}^{m} d_{ij}(t) \tag{5}$$

where m is the number of input vectors we apply to the target circuit, and $d_{ij}(t)$ is the delay increase of gate j under input vector i for a time period of t. In our simulation to determine the weight factor for each gate, we randomly select m input vectors, characterize the delay increase of the gate using GLC, and calculate the weight factor.

Pseudocode 2 shows our iterative SAT solving algorithm for determining a set of satisfiable SAT objectives. We begin with the objectives that set all the signals of regular gates to 0 and HT gates to 1. If the resulting SAT problem is not satisfiable, we keep removing the gate that has the lowest weight from the SAT objectives until a satisfiable SAT problem is obtained.

Pseudocode 2 Iterative input vector selection.

```
Input: Netlist of the target circuit; delay model;
Ouput: Input vector set $IV$ for aging
Detect and diagnose the HT gate set $HT$;
for each $obj_i$ in SAT $S$ do
    if $obj_i \in HT$ then
        $obj_i$ = 1;
    else
        $obj_i$ = 0;
    end if
end for
$IV = sat\_solve(S)$;
while $IV == \emptyset$ do
    Remove $Obj_k$ with the lowest weight from SAT $S$;
    $IV = sat\_solve(S)$;
end while
return $IV$
```

2) Linear Programming: The iterative SAT solving process provide us with a set of input vectors that can be used in the IC aging process to stress the HT gates in order to disable the attacks. Although we specified in the strict SAT objectives that the regular gates should be set in the unstressed mode, the resulting relaxed SAT objectives would cause some of the regular gates to be aged by the applied input vectors. Consequently, if we only apply a single input vector during the entire HT masking process, the HT can be disabled but at the price that some regular gates are constantly aged, which greatly impacts the overall performance of the IC and has a potential of causing a subset of the regular gates to malfunction.

The speed degradation issue require us to use multiple aging input vectors alternately, in order not to age the same set of regular gates during the HT masking process. Our idea is to select a subset of input vectors from the entire input vector set provided by the SAT solving process and use them to keep the timing constraint of each gate satisfied. We achieve this goal by setting an additional set of timing constraints in a linear program, where the objective is to maximize the total delay increase of the HT gates with the following timing constraints for each regular gate:

$$D_j = \sum_{i=1}^{m} \eta_j d_{ij} \leq Th_j \tag{6}$$

where m is the number of candidate input vectors obtained from the SAT solving process; n is the number of regular gates; $\eta_j (j = 1...m)$ is the percentage of time to apply each candidate input vector; $d_{ij} (i = 1...m, j = 1...n)$ is the expected delay increase of regular gate j when input vector i is applied; and Th_j is the threshold for the delay increase of gate j. Among all the parameters, m is provided by the aforementioned SAT solving process; d_{ij} can be obtained by the gate characterization process; and η_j's are the variables we are characterizing in the LP;

3) Delay Compensation Using Adaptive Body Bias: As discussed in the previous subsections, the aging-based HT masking approach causes the delays of the regular gates to increase, because the SAT problem that strictly sets all regular gates to 0 and HT gates to 1 is often unsatisfiable. Be sides introducing timing constraints in the LP formulation, we further use adaptive body bias (ABB) to compensate for the delay degradation. ABB has been proposed as an effective approach to compensate for the PV impact on performance and power consumption. It provides the ability to manipulate transistor threshold voltage through the body effect and thus enables either a forward or a reverse body effect to change threshold voltage [21]. Here we use ABB to manipulate the threshold voltage of regular gates that are increased by aging, so that we can compensate for the degradation in delay.

VII. Simulation Results

A. Consistency-based HT detection

We show the simulation results on ISCAS and ITC benchmarks for consistency-based HT detection in Table I. For each benchmark, we simulate two cases where HTs are present (i.e., HT-present) and there are no HTs in the circuit (i.e., HT-free). The threat model we consider is the additional gate attack, where the attacker embeds one or more small sized gate (e.g. a NAND gate) into the circuit. The metric we use for identifying HTs is the average discrepancy (d_{avg}) of the scaling factors that is defined in Section IV. We select pairs

TABLE I
HT DETECTION RESULTS USING CONSISTENCY-BASED GLC: THE VALUES IN THE "HT-FREE" AND "HT-PRESENT" COLUMNS REPRESENT THE AVERAGE DISCREPANCY OF THE OVERLAPPING GATES IN TERMS OF THEIR SCALING FACTORS.

Benchmark	Number of Gates	HT-Free	HT-Present
C432	160	0.0018	0.088
C499	202	0.0062	0.20
C880	383	0.0058	0.073
C1355	546	0.0039	0.27
C1908	880	0.0021	0.23
C2670	1193	0.0014	0.13
C3540	1669	0.015	0.12
C5315	2307	0.0062	0.12
S526	72	0.0013	1.30
S38584	19253	0.0047	0.24
b19	231266	0.0059	0.38

of segments that have overlapping gates and can cover all the gates in the circuit, conduct GLC of each of the segment, and calculate the d_{avg} value over all pairs. We can observe from Table I that for the sizing-based HT attack, there are large gaps (more than 15X) in terms of d_{avg} between the HT-free case and the HT-present case. This enables us to draw a decision line between the d_{avg} values in the two cases and use it to determine whether HTs exist or not. In this way, we obtain zero false positives and zero false negatives in HT detection.

B. *Consistency-based HT Diagnosis*

We evaluate the consistency-based HT diagnosis approach on a set of ISCAS benchmarks, as shown in Fig. 4. For each benchmark, we show the scaling factors of the overlapping gates in three segments, where a single HT is embedded in one of the segments (e.g., Segment 3). We observe from the results that the two values of scaling factors from the HT-free segments are consistent with each other, and that in the HT-present segment is either a very high value or a very low value apart from the two consistent values. These results enable us to conclude that the HT is embedded in Segment 3 with zero false positives and zero false negatives.

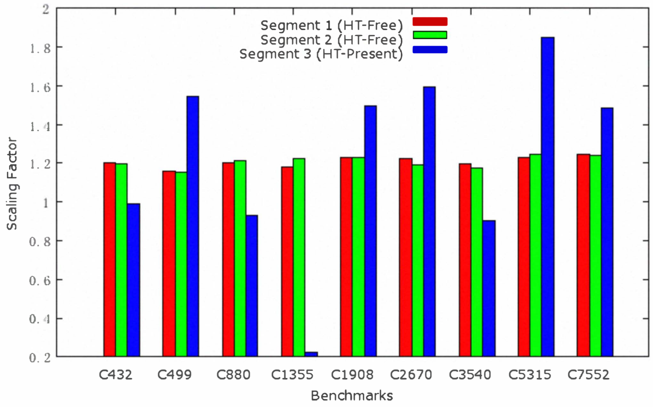


Fig. 4. Simulation results for consistency-based HT diagnosis.

VIII. Conclusion

We developed a complete solution of HT detection, diagnosis, and masking. We employed segmentation and consistency-based gate characterization to determine the existence of HTs and their locations. Next, we select input vectors to age the HTs embedded in the circuit and disable their functionalities. Our simulation results on a set of ISCAS and ITC benchmarks indicate that the proposed approach is scalable and capable of detecting and diagnosing HTs accurately.

Acknowledgment

This work was supported in part by the NSF under awards CNS-0958369, CNS-1059435, and CCF-0926127.

References

[1] M. Tehranipoor, F. Koushanfar, A Survey of Hardware Trojan Taxonomy and Detection, IEEE Design and Test of Computers, Vol. 27, No. 1, 2010, pp. 10-25.

[2] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, B. Sunar, Trojan Detection Using IC Fingerprinting, SP 2007, pp. 296-310.

[3] F. Wolff, C. Papachristou, S. Bhunia, R. Chakraborty, Towards Trojan-free Trusted ICs: Problem Analysis and Detection Scheme, DATE 2008. pp. 1362-1365.

[4] M. Banga, M. Hsiao, A Region Based Approach for the Identification of Hardware of Trojans, HOST 2008, pp. 40-47.

[5] J. Li, J. Lach, At-speed Delay Characterization for IC Authentication and Trojan Horse Detection, HOST 2008, pp. 8-14.

[6] Y. Jin, Y. Makris, Hardware Trojan Detection Using Path Delay Fingerprint, HOST 2008, pp. 51-57.

[7] S. Wei, M. Potkonjak, Scalable Segmentation-Based Malicious Circuitry Detection and Diagnosis, ICCAD 2010, pp. 483-486.

[8] S. Wei, M. Potkonjak, Scalable Hardware Trojan Diagnosis, IEEE Transactions on VLSI Systems, 2011.

[9] S. Wei, S. Meguerdichian, M. Potkonjak. Gate-level Characterization: Foundations and Hardware Security Applications, DAC 2010, pp. 222-227.

[10] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, C. Spanos, Modeling Within-Die Spatial Correlation Effects for Process-Design Co-Optimization, ISQED 2005, pp.516-521.

[11] M. Nelson, A. Nahapetian, F. Koushanfar, M. Potkonjak, SVD-Based Ghost Circuitry Detection, Information Hiding 2009, pp. 221-234.

[12] S. Wei, S. Meguerdichian, M. Potkonjak, Malicious Circuitry Detection Using Thermal Conditioning, IEEE Transactions on Information Forensics and Security, 2011.

[13] S. Wei, M. Potkonjak, Integrated Circuit Security Techniques Using Variable Supply Voltage, DAC 2011, pp. 248-253.

[14] S. Wei, A. Nahapetian, M. Potkonjak, Robust Passive Hardware Metering, to appear, ICCAD 2011.

[15] J. Stathis, Physical and Predictive Models of Ultrathin Oxide Reliability in CMOS Devices and Circuits, IEEE Transactions on Device and Materials Reliability, Vol. 1, No. 1, 2001, pp. 43-59.

[16] J. Pang, D. Chong, T. Low, Thermal Cycling Analysis of Flip-chip Solder Joint Reliability IEEE Transactions on Components and Packaging Technologies. Vol. 24, No. 4, 2001, pp. 705- 712.

[17] S. Chakravarthi, A. Krishnan, V. Reddy, C. Machala, S. Krishnan, A Comprehensive Framework for Predictive Modeling of Negative Bias Temperature Instability, Reliability Physics Symposium Proceedings, 2004, pp. 273-282.

[18] M. Agarwal, B. Paul, M. Zhang, S. Mitra. Circuit Failure Prediction and Its Application to Transistor Aging, VTS 2007, pp.277-286.

[19] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, V. De. Parameter Variations and Impact on Circuits and Microarchitecture. DAC 2003, pp. 338-342.

[20] N. Een, N. Sorensson, An Extensible SAT-solver, SAT 2003, pp. 333-336.

[21] T. Chen, S. Naffziger, Comparison of Adaptive Body Bias (ABB) and Adaptive Supply Voltage (ASV) for Improving Delay and Leakage Under the Presence of Process Variation, IEEE Transactions on VLSI Systems, Vol. 11, No. 5, 2003, pp. 888-899.