Wireless Security Techniques for Coordinated Manufacturing and On-line Hardware Trojan Detection

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

ABSTRACT

This paper addresses the hardware Trojan (HT) attacks that impose severe threats to the security and integrity of wireless networks and systems. We first develop HT attack models by embedding a single HT gate in the target design that triggers advanced malicious attacks. We place the one-gate HT trigger in such a way that it exhibits rare switching activities, consumes ultra-low leakage power, and hides from delay characterizations. Therefore, the HT attack models are capable of bypassing the widely used side channel-based HT detection schemes. Furthermore, based on the HT attack models, we investigate the potential on-line threat models during the system operation and develop an in-field trusted HT detection approach using physical unclonable functions (PUFs). We evaluate the effectiveness of the HT attack and defense models on a set of ISCAS'85, ISCAS'89, and ITC'99 benchmarks.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection—*Physical Security*

General Terms

Security

Keywords

Wireless security, hardware Trojan detection, process variation

1. INTRODUCTION

1.1 Motivation

Recently, wireless communication, computation, and sensing devices, such as mobile phones, laptops, and tablets, have been experiencing exceptionally explosive growth. For example, every second more than 30 cell phones are sold worldwide. In addition, emerging industrial sensor networks are both economically and strategically important. Furthermore, wireless security imposes a technically challenging set of objectives and requirements. For example, side channel and fault induction security attacks [18][19][24][32][41][45] are much more likely on cell phones and, in particular, on sensor nodes that may be deployed in unprotected or even hostile environments. Also, operational conditions and numerous design constraints such as low energy, low power, and low cost impose difficulties on security requirements.

As a consequence, wireless security has emerged as a premier research and development issue. Numerous important aspects have been addressed, such as key management schemes in mobile ad hoc networks [12] and distributed sensor networks [16], secure routing protocols [7] and localization algorithms [26] to prevent wireless sensor attacks, and privacy protection in RFID systems [34]. However, none of these important contributions address the detection of hardware Trojans (HTs) [22][40]. HTs are in a sense the most powerful way to complete compromise the security any wireless or other devices because they enable the attacker to bypass all application and system software defense mechanisms, to access any storage element, change access rights of any program, and abuse (e.g., induce high energy consumption) or destroy any piece of hardware.

1.2 New Types of Security Attacks

Our starting point is the observation that it is exceptionally easy to hide an arbitrary powerful and large HT inside even a small integrated circuit (IC). All what is required is to place the HT circuitry into power down mode that is enabled by rare input activation signals.

There are three main entities that can be measured on an IC: switching energy, leakage energy, and delay. The attacker may create HTs that either do not have impact or have exponentially low probability of impacting any of these three entities. Taking delay measurements as an example, an unresolved difficultly of timing measurements is the inability of individually sensitizing and characterizing each component using the test vectors. This is because of the existence of parallel routes that reconverge to a single point, which make it difficult to map the measured path delay to a specific path for the consideration of Trojans. As shown in a small example in Figure 1, even though we can measure the delay between input x and output y, we are unable to determine whether the measured delay is for path 1 or path 2. Furthermore, the presence of process variation [6][11][14] would further complicate the case, since the delay

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
WiSec'12, April 16–18, 2012, Tucson, Arizona, USA.
Copyright 2012 ACM 978-1-4503-1265-3/12/04 ...$10.00.

of path 1 may be smaller than that of path 2 on one chip but could be larger on another.

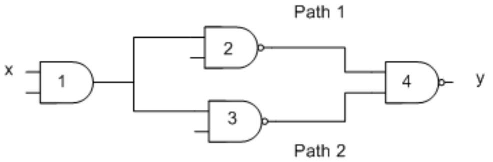

Figure 1: Example of reconvergent path that poses a potential risk for hidden HT attacks.

1.3 Contributions

Our technical contributions include the following.

1. We have invented techniques for creation of new types of exceptionally powerful HTs that are difficult to detect. The key idea is to use power (or clock) gating in such a way that in default mode the HT is placed in power down mode. The HT is activated by a single gate (e.g., an AND gate) that has an output rarely switched to value one (i.e., the activation condition) by anybody except the attacker. The gate is intentionally aged or implemented so to have very high threshold voltage that corresponds to ultra exponentially low leakage energy, which cannot be detected even by the most advanced state-of-the art energy measurement instruments. Therefore, it cannot be detected by any techniques that measures switching and/or energy. Finally, the HT is placed in such a way that it does not have any impact on delays between any two pairs of flip-flops. The HT is activated by an input vector sequence that is known by the attacker but otherwise has exponentially low probability of occurrence.

2. Coordinated manufacturing and in-field testing for HT detection. For the first time it is proposed that results from manufacturing time testing and in-field testing are combined and coordinated. Manufacturing testing provides golden standard measurements for scenarios where there is no HT or it is not activated. Hence, on-line in-field testing for activated HTs can be easily accomplished.

3. Employment of physical unclonable function (PUF) for secure testing. Man-in-the-middle attacks are among the most effective ways to compromise numerous security techniques. We use PUFs combined with random challenges to ensure that the attacker must report measurements that are actually done at the system-under-test at a specified moment.

4. Use of global position signals (GPS) signal to reduce communication and energy cost. To the best of our knowledge this is the first time that GPS is used for testing and HT detection purposes. It is also the first scheme that uses GPS signal to reduce the communications of random challenges.

5. Low energy yet comprehensive HT testing. Privileged information (e.g., passwords) is often data of the highest importance. A sophisticated HT may be enabled and disabled to further complicate their detection. Finally, any wireless security should induce very low energy expenditures. We simultaneously resolve these three requirements, by invoking HT testing only when gates that are associated with storing privileged information are activated and by finding low power HT test vectors.

2. RELATED WORK

In this section, we summarize the related work on hardware Trojan and its detection. We start with introducing the research efforts on process variation, which is considered as the dominant source of challenges for HT detection attempts. Then, we discuss the existing HT detection approaches with emphasis on the major differences in our contributions.

2.1 Process Variation

Process variation (PV) in IC manufacturing is the deviation of IC parameter values from nominal specifications, due to the nature of the manufacturing process [6][11][14] It is observed that PV is caused by the inability to precisely control the fabrication process at small-feature technologies [35]. For example, the lithographic lens aberrations result in systematic errors on transistor sizes, and dopant density fluctuations impose random variations on design parameters. Also, PV has impact on various levels of the IC properties, including wafer-level, die-level, and wafer-die interaction [37]. Consequently, PV is an unavoidable technological phenomenon of all deep submicron and nano IC realization technologies. The main PV ramification is that each device (e.g., gate, transistor, and interconnect) of the same design has different manifestational characteristics (e.g., delay or static power) on different integrated circuits (ICs). These device level characteristics have profound impact on the overall IC characteristics. For example, the operational speed may easily differ by more than 30% from nominal and leakage by factors of 20X [11].

Besides its direct impact on the physical or manifestational properties of ICs, PV is also considered as a major source of risk for hardware-based attacks, because the observable variations caused by the malicious hardware components can be easily attributed to the consequence of PV. It is difficult to identify the source of the variation and determine whether it is from the naturally existed PV or from malicious modifications to the design. Based on these thoughts, several research efforts have been made to characterize and quantify the impact of PV at the gate-level (i.e., gate-level characterization, or GLC) in a non-destructive way [4][5][42][44][43][47] . The scaling factors of the IC key parameters due to PV can be determined by measuring the properties of the entire circuit and solving a set of linear equations.

2.2 Hardware Trojan Detection

Hardware Trojan detection has become an active research area as the increasing trend of IC outsourcing conducted by the IC design companies to increase their revenue. Since DARPA issued its first call for the study of hardware systems security and, in particular, hardware Trojans in 2005, more than a hundred related security techniques have been proposed and evaluated.

Agrawal et al. [2] introduced the hardware Trojan problem and proposed the first HT detection approach using fingerprints generated from IC side channels. Thereafter, a large number of HT detection methods have been proposed, which can be classified into two categories. First, functional test-based HT detection simulates a set of test input vectors and verifies the correctness of the outputs. Researchers have proposed a variety of methods to generate the test input vectors with a goal of maximizing the probability of detection

[8][49]. Second, HT detection techniques using side channel-based analysis have been developed recently. These methods monitor the variations caused by HTs in representative IC properties, such as leakage power [3][42][44][46], switching power [2][9], delay [21][27], or a combination of the properties [23].

Despite the research efforts in various side channel-based HT detection methods, the current HT detection schemes did not consider the cases where the attacker is well aware of the detection techniques. The attacker may tend to minimize the variations in the well known side channels caused by HTs, or attribute them to process variation. We discuss and develop the advanced attack and defense strategies in this paper and showcase their applicability to secure wireless systems.

3. PRELIMINARIES

In this section, we first introduce the power and delay models that are used to quantify the gate-level manifestational properties for HT detection. Then, we discuss the IC aging model that we employ in the design and implementation of our approaches.

3.1 Power Models

Leakage power and switching power have been considered as two major side channels for the observations of HT behaviors. We refer to the leakage power model presented by Markovic et al. [29] for the creation of our leakage power-based HT model. Equation (1) shows the leakage power of a logic gate based on several physical level IC parameters, where W is gate width, L is gate length, V_{th} is threshold voltage, V_{dd} is supply voltage, n is subthreshold slope, μ is mobility, C_{ox} is oxide capacitance, D is clock period, ϕ_t is thermal voltage $\phi_t = kT/q$, and σ is drain induced barrier lowering (DIBL) factor.

$$P_{leakage} = 2 \cdot n \cdot \mu \cdot C_{ox} \cdot \frac{W}{L} \cdot (\frac{kT}{q})^2 \cdot D \cdot V_{dd} \cdot e^{\frac{\sigma \cdot V_{dd} - V_{th}}{n \cdot (kT/q)}} \quad (1)$$

We observe from Equation (1) that the leakage power of a logic gate depends on the threshold voltage (V_{th}) in a non-linear manner. In particular, if one can increase the V_{th}, either in the pre-silicon or post-silicon stage, the leakage power would decrease exponentially. This phenomenon provides attackers with a means of embedding ultra-low power components in the target circuit for HT attacks.

The gate-level switching energy model [29] is described by Equation (2), where α is the switching probability.

$$P_{switching} = \alpha \cdot C_{ox} \cdot W \cdot L \cdot V_{dd}^2 \quad (2)$$

Equation (2) indicates that the total switching power consumed by a logic gate during the IC operation is an accumulated value based on its switching activity. Therefore, one possible way of decreasing the switching power is to limit the number of switches, which can be achieved by controlling the input vectors of the circuit.

3.2 Delay Model

The delay of a single logic gate can be expressed as

$$d = gh + p \quad (3)$$

where g and h are logical effort and electrical effort, respectively; and p is parasitic delay [38]. In particular, we use the delay model in [29] that connects the gate delay to its sizing and operating voltages:

$$Delay = \frac{k_{tp} \cdot k_{fit} \cdot L^2}{2 \cdot n \cdot \mu \cdot \phi_t^2} \cdot \frac{V_{dd}}{(ln(e^{\frac{(1+\sigma)V_{dd} - V_{th}}{2 \cdot n \cdot \phi_t}} + 1))^2} \cdot \frac{\gamma_i \cdot W_i + W_{i+1}}{W_i} \quad (4)$$

where subscripts i and $i+1$ represent the the driver and load gates, respectively; γ is the ratio of gate parasitic to input capacitance; and k_{tp} and k_{fit} are fitting parameters.

3.3 IC Aging Model

Phenomena such as hot carrier injection (HCI) and in particular negative-bias temperature instability (NBTI) 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 [1]. For the discussion in this paper, we refer to the NBTI aging model presented by Chakravarthi et al. [13], as shown in the Equation (5):

$$\Delta V_{th} = A \cdot exp(\beta V_G) \cdot exp(-E_\alpha / kT) \cdot t^{0.25} \quad (5)$$

where V_G is the applied gate voltage; A and β are constants; E_α is the measured activation energy of the NBTI process; T is the temperature; and t is the stress time. The aging effect provides a method to increase the threshold voltage of a logic gate in the post-silicon stage regardless of the impact of PV. Considering the leakage power introduced by Equation (1), we consider aging as a convenient means for attackers to create ultra-low leakage HTs that are difficult to detect.

4. HARDWARE TROJAN ARCHITECTURE AND PLACEMENT

4.1 Hardware Trojan Architecture

As the first step toward the wireless security analysis, we design a HT placement model that complicates the detection of embedded HTs in wireless systems. Our observation is that the existing side channel-based HT detection techniques check the exposure of HTs in terms of their manifestational characteristics, such as power and delay. Therefore, in the design of HT attack models, our goal is to minimize the possible variations caused by HTs in all aspects of their manifestational properties. Figure 2 shows the overall architecture of the designed HT model. We use only one single gate as the trigger of the malicious circuitry in order to minimize the observable variations in the original design. The one-gate HT trigger would activate the malicious circuitry only when a rare condition is satisfied, such as a specific combination of input signals. During the normal system operation when the activation condition is not satisfied, the embedded malicious circuitry is under the power off mode, in which it does not consume switching or leakage power nor observable through delay measurements. In this way, we are able to embed a malicious circuitry that is unobservable via all three most widely used manifestational properties, namely switching power, leakage power, and delay. Furthermore, the attacker would induce the wireless system to apply the rare activation condition only once during its life time and activate the security attack.

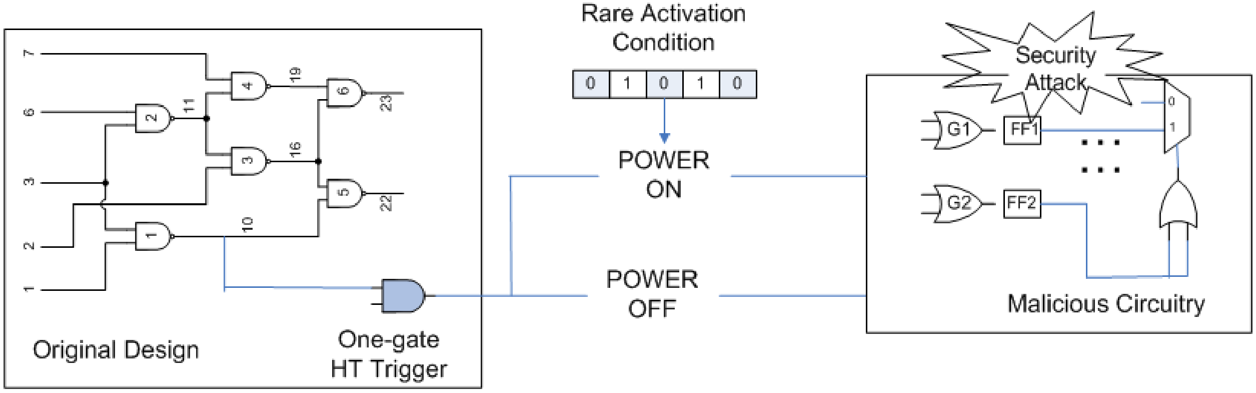

Figure 2: Overall architecture of the hardware Trojan attack.

4.2 Hardware Trojan Placement

The goal in HT placement, from the attacker's perspective, is to hide the HTs in the target IC in terms of the side channels that are commonly monitored in the HT detection schemes. We investigate three widely used IC properties, namely switching power, leakage power, and delay, and develop a HT placement strategy for each case that composes a challenging attack, especially when deployed in a wireless system that operates in an on-line environment. In particular, we have developed the following three HT placement models:

First, based on switching power, we place a HT at a rare switching location in the target design, so that it has an extremely low probability of switching during the normal IC operation. However, the HT can be switched by a certain set of input vector to activate the malicious circuitry.

Second, regarding leakage power, we create ultra-low leakage HTs by intentionally aging the HT to increase the threshold voltage and thus decrease the leakage power exponentially. We determine the input vectors for aging the HT gate using a Boolean satisfiability (SAT)-based approach.

Third, to complicate the timing-based HT detection, we identify the reconvergent paths in the target circuit and intentionally place the HT in one of them. Consequently, even though delay can still be measured, it is difficult for the HT detection method to determine which one of the reconvergent paths is being measured.

4.2.1 *Rare Switching HT Placement*

In the rare switching-based HT placement, our goal is to find the locations in the design where the gates have the least switching activities during the normal system operation. Meanwhile, those locations for HT placement must be switchable by a certain small set of input vectors, which can be used to activate the malicious circuitry.

In order to find the best location for low switching HTs, our intuition is that the switching activity of a specific gate depends on two factors: (1) the switching probability of its transitive fan-in gates; and (2) the correlation of the switching patterns of its transitive fan-in gates. Here we define correlation as the probability that two or more gates switch at the same time. Therefore, our idea is to create a HT gate and feed it with the outputs of the rare switching and highly correlated gates. We first conduct simulations on the target circuit using a set of random input vectors to find the least switching gate and add it to a candidate group. Next, we iteratively add to the candidate group one more gate, which is most correlated with the existing gates in the group and has the least switching activity. In particular, the one specific gate that we add in each iteration is determined approximately by the sibling gate of the existing gates in the candidate group that switches the most rarely. Furthermore, in each iteration, we initiate a SAT solving procedure (discussed in details in Section 4.2.2) and ensure that there exists at least one pair of input vectors that can switch the embedded HT and thus trigger the malicious circuitry.

4.2.2 *Low Leakage HT Creation*

We employ IC aging technique to create a HT gate that consumes ultra low leakage power. In particular, during or after the IC manufacturing process, we intentionally stress the HT gate so that its threshold voltage can be increased and, by following Equation (1), the leakage power would decrease exponentially. In particular, the method we use for aging a set of gates in the circuit is by setting the gates in the stress mode (i.e., signal 1). According to the aging model discussed in Section 3, there is a speed-up in V_{th} increase due to the stress. We use a SAT-based approach to select the input vectors that set the specific set of gates under stress.

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 based on our requirements regarding the gates 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 [15] that deliver fast and accurate SAT solutions. 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. The gates that are not included in the objective file will be assumed as don't-care in the SAT solving process. In particular, the objectives in the SAT problem follow the following format:

$$obj_i = 0|1, i = 1...k \tag{6}$$

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

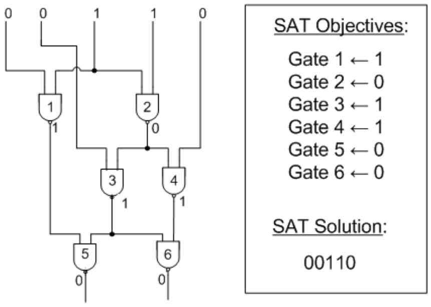

Figure 3: Example of SAT formulation for aging input vector selection. The output from the SAT solver provides the input vectors that satisfy the objectives.

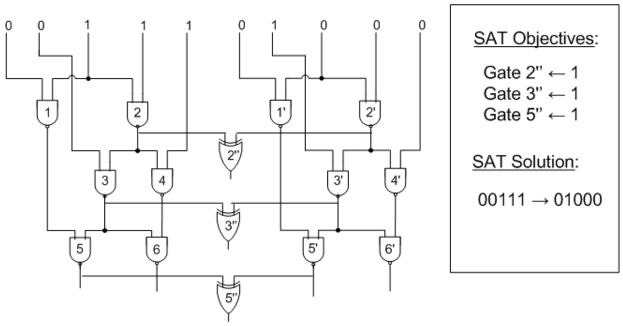

Figure 4: Example of SAT problem formulation for finding input vectors that switch a specific set of gates (e.g., gates 2, 3, and 5).

We demonstrate the SAT problem formulation using a small example in Figure 3. For the clarity of discussion, we consider only a small circuit with 6 NAND gates. In this example, we set gates 2, 5, and 6 to signal 0 and gates 1, 3, and 4 to signal 1, which we specify in the objective file. The SAT solver outputs input vector 00110 that satisfy the specified objectives.

Furthermore, we notice that a variation of the aforementioned SAT formulation method can be used to determine the input vectors that switch a gate in a specific way. In order to accomplish this, we first duplicate the target circuit and generate a "dual-circuit", where every gate in the original design has a duplicated counterpart in the dual section. As shown in Figure 4, each gate i now has a counterpart gate i' in the dual section of the circuit. Then, for each gate i that we want to determine switching input vector for, we add a two-input XOR gate i'' and feed it with the outputs of gates i and i'. The reason why we use a XOR gate is that its output can serve as an indicator of whether gate i switches or not. In other words, if we assume that the original section of the dual circuit represents the status as of clock cycle t, and the dual section represents that of clock cycle $t+1$, we claim that the output signal of the XOR gate i'' is 1 if and only if gate i switches from clock cycle t to $t+1$. Therefore, by specifying the input signals of the added XOR gates in Equation (6), we have found a method to determine the input vectors for a specific switching pattern.

Pseudocode 1 Backtracking algorithm for reconvergence identification.

```
Input: Netlist $Net$; Primary input $PI$; Primary output $PO$;
Ouput: A set of Paths $P$ between $PI$ and $PO$;
curNode ← PI;
push curNode into stack s;
repeat
    curNode ← PI;
    for each node in in curNode's inputs do
        if n is not visited then
            mark in as visited;
        end if
        if $in \neq PI$ then
            continue;
        else
            curNode ← in;
            if curNode is not a primary input in Net then
                push curNode to s;
                mark curNode as visited;
            else if curNode == PI then
                push curNode to s;
                mark curNode as visited;
                push all the nodes in s to P;
                curNode ← pop(s);
                Continue;
            else
                mark curNode as visited;
                curNode ← top(s);
            end if
        end if
    end for
    mark top(s) as unvisited;
    pop(s);
    curNode ← top(s);
until s is empty
return P;
```

4.2.3 Delay Testing and HT Placement

To prevent the embedded HT from being detected by timing-based approach, an attacker may consider placing the HT in one of the reconvergent paths, with which there are one or more other paths that are in parallel. In this case, it is difficult for the defender to determine which one of the parallel path has been measured and thus malicious modifications to the parallel paths can be well hidden from the delay monitoring.

Assuming that the netlist of the target circuit is a directed graph G, with each pin as an edge $e_i \in E$, and each gate (or input, output) as a node $n_i \in N$, we have the following definition for a reconvergence point on the circuit:

Definition 1. Reconvergence Point. A node $n_i \in N$ in netlist G is a reconvergence point if and only if the in-degree of n_i is larger than 1.

Pseudocode 1 shows the algorithm that we use to find reconvergence points in the target design. We conduct a depth-first search from the specific output PO toward the inputs. During this process, we keep pruning the edges using backtracking to trace all the possible paths toward a specific input PI. If there are more path in between PI and PO, we regard these paths as the possible locations where HT can embedded to bypass the timing-based HT detection.

4.3 Summary

The aforementioned three one-gate HT models greatly complicate the HT detection attempts. If the attacker intentionally places the HT at a location that combines all three types of attack models, an effective detection approach is hardly feasible unless the malicious circuitry that is triggered by the one-gate HT is activated. Therefore, by leveraging the one-gate HT models, an attacker may force the target of HT detection techniques to move from regular post-silicon testing to the system operation period after the IC is released. Therefore, the costs and difficulty level of conducting HT detection is greatly increased due to the one-gate HT models.

5. ON-LINE HARDWARE TROJAN ATTACK AND DEFENSE

During the system operation, the attacker must trigger and power up the malicious circuitry in order to activate the HT attack. Once the malicious circuitry is activated, one can easily detect the abnormality, since the malicious circuitry often contains a large number of gates as well as complicated structures in order to accomplish advanced security attacks, such as leaking confidential information or making the device malfunction. However, it is still possible for the attacker to manipulate the behavior of the wireless system to further bypass the on-line security checks after the activation of malicious circuitry. In this section, we discuss the possible mechanisms that an attacker may leverage to conduct on-line HT attacks and propose the corresponding defense methods.

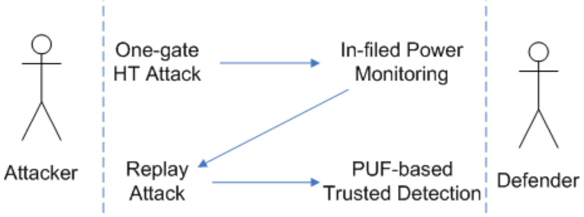

Figure 5: On-line attack and defense model.

5.1 Attack and Defense Models

Figure 5 demonstrates the attack and defense models after the HT-embedded wireless system enters the operation mode. The attacker would activate the malicious circuitry by inducing the system to run an application that satisfies the rare activation condition. Then, as a method of defense, the defender would sample and monitor the power profile of the remote wireless system and observe the variations that may be caused by the activation of the malicious circuitry. However, it is possible for the attacker to conduct a more advanced attack, namely replay attack, that tricks the online monitoring scheme with outdated power profiles that do not reflect any variations caused by the malicious circuitry. In order to resolve the replay attack, we develop a PUF-based trusted HT detection technique that authenticates each sample of power profile with specific time and location information and, therefore, any attempts to report replayed power profiles would be detected.

Pseudocode 2 In-field power monitoring for detecting one-gate HT attack during system operation.

```
Designer implements a test trigger into the design that monitors the activity of the privileged area for security attacks;
Attacker embeds the HT gate and the malicious circuitry in the wireless system;
The wireless system passes post-silicon test, since the malicious circuitry powered off;
The wireless system starts operating;
Defender collects power profile during the initialization period as the baseline profile;
The wireless system operates normally for a period of time $t$;
Attacker triggers the one-gate HT and activates the malicious circuitry;
The test trigger activates the power meter to measure the power profile and reports it to defender;
Defender observes abnormal variation in power profile caused by the activated malicious circuitry;
Defender terminates the operation of the wireless system that is under HT attack;
```

5.2 On-line HT Detection by In-field Power Measurements

During the operation mode of the wireless system when the malicious circuitry can be possibly activated, we employ in-field power metering techniques [20][28][36] to keep track of the power profile. The micro power meter that is integrated into the wireless system is capable of measuring the real-time power profiles and reporting to the remote administrator for further assessment. In order to reduce the cost of conducting such power measurements, we employ a test trigger gate to monitor the activity of the privileged area in the design. The test trigger is activated and the power meter starts measuring the power profile only when the privileged area is suspected to be attacked. When this situation occurs, the power profile data is sent to the administrator for further analysis to confirm the existence of HT attacks. Pseudocode 2 describes the detailed procedure of in-field power measurements for HT detection. The power meter in the wireless system first collects a set of power samples at the beginning of the system operation, which can serve as a baseline for the normal power profile. Once the test trigger gate is activated, the administrator would be able to collect instant power profiles from the power meter and determine whether there is any HT attack being conducted. We consider the variation of power profile as an indicator of HT attack, since the activated malicious circuitry would consume a relatively large amount of power and cause a surge in the leakage power profile compared to the baseline.

5.3 On-line Replay Attack

We note that the straightforward HT detection technique via in-field power profiling can still be bypassed by the attacker. For example, it is possible that the attacker conducts replay attack [39], in which a set of normal leakage power profiles are pre-recorded and reported to the monitoring system constantly. Pseudocode 3 illustrates a typical case of replay attack, which results in the failure of detection. Note that the attacker may start or terminate the replay procedure at any time, or vary the power profile considering environmental factors and the workload on the wireless system to generate more trustworthy power reports.

Pseudocode 3 On-line replay attack that bypasses the in-field power sampling approach.

```
Attacker embeds the HT gate and the malicious circuitry in the wireless system;
The wireless system passes post-silicon test as the malicious circuitry is powered off;
Defender enables the on-line in-field power profiling process;
The wireless system starts operating;
The wireless system operates normally for a period of time $t$;
Attacker records the power profiles $f_t$ within the time period $t$;
The wireless system starts responding with power profiles $f_t$ anytime when there is a profiling request;
Attacker triggers the one-gate HT and activates the malicious circuitry;
Defender observes normal power profile $f_t$ constantly;
The wireless system is compromised by the activated malicious circuitry;
```

5.4 Trusted HT Detection Using Physical Unclonable Functions

Considering the possible on-line replay attack, we develop a trusted HT detection approach based on the use of physically unclonable functions (PUFs) [17][30] . A PUF is a specially designed circuitry in which the prediction of output signals from known inputs is computationally infeasible, unless one has access to the netlist of the circuitry and conduct simulations. Figure 6 shows a sample PUF design, where the complexity of determining the output vectors grows exponentially as the increase of the number of levels in the design. Since there is a huge difference between the simulation time (e.g., in the magnitude of nanoseconds) and the prediction time (e.g., in the magnitude of seconds) for obtaining the output signals, PUFs can be used as a security key for identity authentications in many applications [10][31][33][48].

However, a direct use of PUF with randomly generated challenge bits cannot resolve the replay attack, since we must ensure that the collected power profile in the monitoring process are those generated from the specific sensor at the specific time frame. This requires us to associate each sample with both time and location information and take into consideration of the (time, location, power) triplets at the checking time. For the time stamp, we leverage the secure navigation signals that can be received synchronously from integrated GPS systems [25] at both the remote wireless system and the local administration site. For the identification of sensors, we leverage the fact that each PUF exhibits different delay characteristics due to process variation. Consequently, the output signals are different for the PUFs on different sensors, since they are highly dependent on the accumulated delay at each level of the design. Figure 7 shows our design of the PUF system for resolving the replay attack toward the remote wireless system.

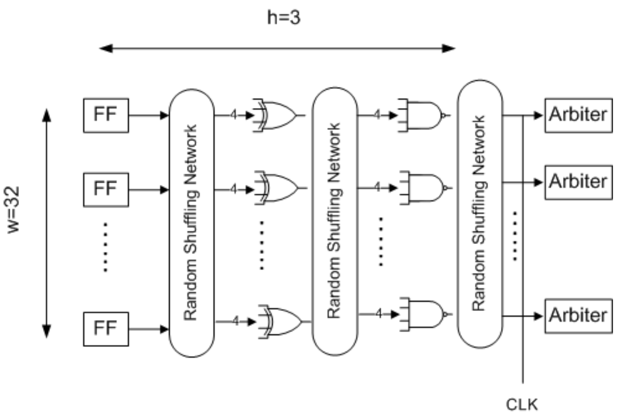

Figure 6: PUF architecture.

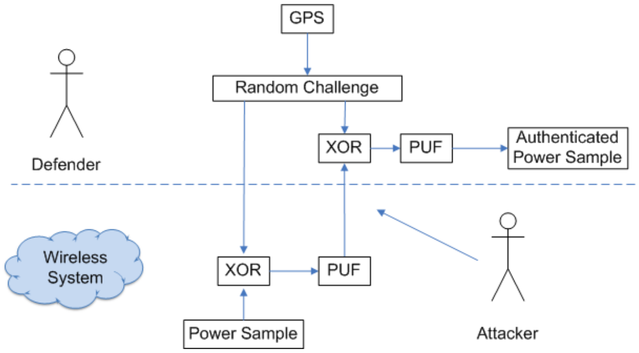

Figure 7: PUF-based trusted HT detection.

6. EXPERIMENTAL RESULTS

We evaluate our HT attack and detection models on a set of ISCAS'85, ISCAS'89, and ITC'99 benchmarks. We model the process variations of the designs following the Gaussian distribution presented in [6] and the quad-tree model presented in [14].

We first examine the effectiveness of the one-gate HT placement by observing the resulting switching activities, leakage power reductions, as well as the number of the delay-unobservable gates due to reconvergent paths. Then, we evaluate the PUF-based trusted HT detection scheme by checking the randomness of the output bits.

6.1 Rare Switching-based HT

Table 1 shows our simulation results regarding rare switching-based HT placement. We select 5 groups of fan-in gates for the one-gate HT (e.g., an AND gate), with up to 10 gates in each group. Our algorithm ensures that the groups of gates, which serve as the fan-in gates of the one-gate HT, would result in rare switching activities of the HT gate. Meanwhile, we ensure that the HT can be switched by a certain set of input vectors to activate the malicious circuitry, which is proved by the solution of the SAT problem. We simulate the switching probability of the HT gate using 5,000 pairs of randomly generated input vectors. The results show that we obtain less than 0.50% switching probabilities of the HT gate in all the benchmark circuits, which is considered very low and difficult to observe by the switching power-based detection. Therefore, our results indicate that the attacker can leverage the rare but non-zero switching activities to activate the HT during the operation of the wireless system, since the one-gate HT is difficult to detect before the actual activation of the malicious circuitry.

6.2 Low Leakage-based HT

Table 2 shows the simulation results of leakage power reduction by intentionally aging the HT gate. For each benchmark circuit, we select three input vectors that would stress the HT gate as well as a minimum number of other gates in the circuit using the SAT-based approach. Then, we apply each input vector to the circuit for a certain amount of time so that the threshold voltages of the stressed gates can be increased by 10% due to aging. We simulate the leakage power reduction of both the HT gate and the entire circuit, by following the leakage power model (i.e., Equation (1)). The results indicate that the selected input vectors can reduce the leakage power of the HT gate by more than 80% in all the tested circuits, while the leakage power reduction of the entire circuit is much less (below 35%). This enables the ultra-low leakage HT to easily hide under measurement errors or process variations.

6.3 Delay Testing and Placement

Figure 8 shows our simulation results on ISCAS'85, ISCAS'89, and ITC'99 benchmarks regarding the gates that cannot be characterized by using delay as the side channel due to reconvergences. The only possibility to conduct delay characterization is that there is no reconvergence from a specific input to a specific output in the design [1]. We observe that there is a large number of the gates (at least 40%) that are subject to reconvergences and thus are uncharacterizable using non-destructive delay measurements, leaving a large portion of the circuit under the risk of HT insertion. An attacker may easily search in the circuit for reconvergent paths using Pseudocode 1 and embed the one-gate HT in one of the reconvergent paths to bypass security checks.

6.4 PUF-based On-line In-field Detection

In order to evaluate the PUF-based in-field HT detection method, we simulate the implemented PUF design using random challenge bits and observe for the randomness of the output signals. Our idea is that if the output signals are random (i.e., in the optimal case, with 50% probability being 1 and 50% being 0), the prediction attempt within any reasonable amount of time will fail, under the consideration that the complexity of prediction grows exponentially with the number of output pins. Figure 9 shows our simulation results, where we observe probabilities of signal 1 for all the output pins close to the optimal probability (50%).

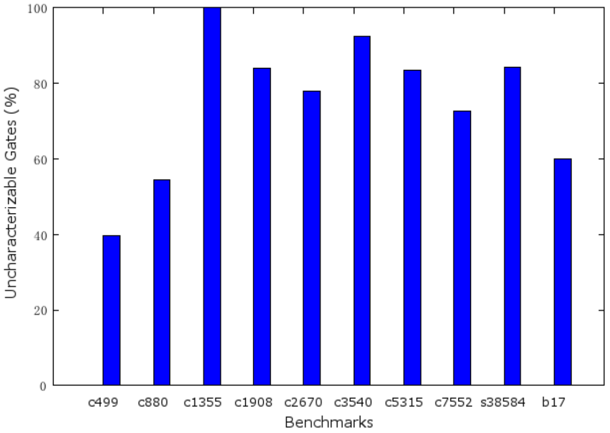

Figure 8: Simulation results regarding delay-uncharacterizable gates due to reconvergences. The high percentage of uncharacterizable gates in each benchmark circuit indicates that there is a large number of candidate locations for embedding the one-gate HT that is difficult to detect using delay-based approaches.

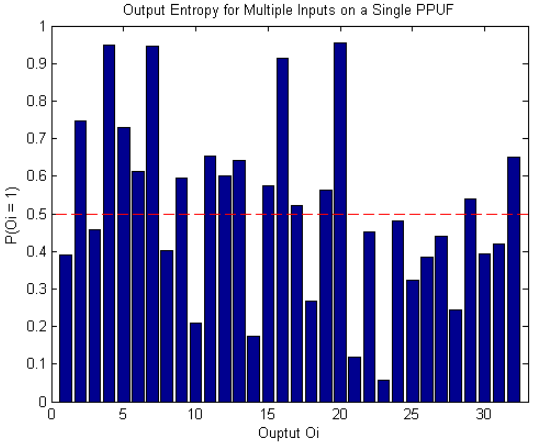

Figure 9: The probability of output bit O_i being 1 in the PUF (w=32, h=3) following the architecture in Figure 6. The probabilities of all output pins are close to the baseline probability (0.5) for a completely random prediction.

7. CONCLUSION

We have developed three one-gate hardware Trojan attack models that can bypass the widely used side channel-based HT detection schemes and pose threats on the security of wireless systems. In particular, the HT models leverage a single HT trigger that exhibits rare switching activities, consumes ultra-low leakage power, and hides from delay characterizations due to reconvergent paths. We showed that the proposed one-gate HT models are capable of compromising

[1] Note that the delay characterization is applicable to certain cases of reconvergences, where all paths except the path being tested can be fixed to a certain signal value using SAT. This is out of the scope of this paper.

Table 1: Switching probabilities of the embedded HT gate, which are evaluated using simulations with 5000 pairs of random input vectors. For each benchmark, we obtain 5 candidate groups and up to 10 gates in each group that can serve as the transitive fan-in's of the HT gate. Furthermore, the HT gate has been proven by SAT to be switchable by at least one pair of input vectors.

Benchmark	# Gates	# Inputs	# Outputs	Switching Probability of the HT Gate (%)
C499	202	41	32	0.04
C880	383	60	26	0.12
C1355	546	41	32	0.32
C1908	880	33	25	0.20
C2670	1193	233	140	0.04
C3540	1669	50	22	0.12
C5315	2307	178	123	0.32
C6288	2416	32	32	0.04
C7552	3512	207	108	0.28

Table 2: Leakage power reduction of the HT gate and the entire circuit.

Benchmark	# Gates	# Inputs	# Outputs	Leakage Power Reduction of the HT Gate (%)	Leakage Power Reduction of the Entire Circuit (%)
C499	202	41	32	92.2	11.6
C880	383	60	26	95.6	17.2
C1355	546	41	32	97.6	22.9
C1908	880	33	25	93.4	13.5
C2670	1193	233	140	89.7	20.6
C3540	1669	50	22	97.9	32.8
C5315	2307	178	123	98.0	8.8
C6288	2416	32	32	80.3	12.8
C7552	3512	207	108	89.7	21.2

the detection attempts before the activation of the malicious circuitry, forcing effective HT detection to move from post-silicon testing to on-line system operation. Furthermore, we investigated the attack and defense models during the system operation where the malicious circuitry may be triggered by the attacker. We introduced an on-line replay attack model that may be conducted by an attacker, and we developed a PUF-based trusted detection approach to resolve the attack. Simulation results on a set of ISCAS'85, ISCAS'89, and ITC'99 benchmarks verified the effectiveness of the HT attack and detection methods.

8. ACKNOWLEDGEMENTS

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

9. REFERENCES

[1] M. Agarwal, B.C. Paul, M. Zhang, and S. Mitra. Circuit failure prediction and its application to transistor aging. In *VLSI Test Symposium (VTS)*, pages 277–286, 2007.

[2] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar. Trojan detection using IC fingerprinting. In *IEEE Symposium on Security and Privacy (SP)*, pages 296–310, 2007.

[3] Y. Alkabani and F. Koushanfar. Consistency-based characterization for IC Trojan detection. In *International Conference on Computer-Aided Design (ICCAD)*, pages 123–127, 2009.

[4] Y. Alkabani, F. Koushanfar, N. Kiyavash, and M. Potkonjak. Trusted integrated circuits: A nondestructive hidden characteristics extraction approach. In *Information Hiding (IH)*, pages 102–117, 2008.

[5] Y. Alkabani, T. Massey, F. Koushanfar, and M. Potkonjak. Input vector control for post-silicon leakage current minimization in the presence of manufacturing variability. In *Design Automation Conference (DAC)*, pages 606–609, 2008.

[6] A. Asenov. Random dopant induced threshold voltage lowering and fluctuations in sub-0.1 μm MOSFET's: A 3-D "atomistic" simulation study. *IEEE Transactions on Electron Devices*, 45(12):2505–2513, 1998.

[7] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens. An on-demand secure routing protocol resilient to byzantine failures. In *ACM workshop on Wireless security (WiSe)*, pages 21–30, 2002.

[8] M. Banga and M.S. Hsiao. A region based approach for the identification of hardware Trojans. In *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST)*, pages 40–47, 2008.

[9] M. Banga and M.S. Hsiao. VITAMIN: Voltage inversion technique to ascertain malicious insertions in ICs. In *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST)*, pages 104–107, 2009.

[10] N. Beckmann and M. Potkonjak. Hardware-based public-key cryptography with public physically unclonable functions. In *Information Hiding (IH)*, pages 206–220, 2009.

[11] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De. Parameter variations and

impact on circuits and microarchitecture. In *Design Automation Conference (DAC)*, pages 338–342, 2003.

[12] S. Capkun, L. Buttyan, and J.-P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(1):52–64, 2003.

[13] S. Chakravarthi, A. Krishnan, V. Reddy, C.F. Machala, and S. Krishnan. A comprehensive framework for predictive modeling of negative bias temperature instability. In *International Reliability Physics Symposium (IRPS)*, pages 273– 282, 2004.

[14] B. Cline, K. Chopra, D. Blaauw, and Y. Cao. Analysis and modeling of CD variation for statistical static timing. In *International Conference on Computer-Aided Design (ICCAD)*, pages 60–66, 2006.

[15] N. Een and N. Sorensson. An extensible SAT-solver. In *International Conferences on Theory and Applications of Satisfiability Testing (SAT)*, pages 333–336, 2004.

[16] L. Eschenauer and V. Gligor. A key-management scheme for distributed sensor networks. In *ACM conference on Computer and communications security (CCS)*, pages 41–47, 2002.

[17] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas. Silicon physical random functions. In *ACM Conference on Computer and Communications Security (CCS)*, pages 148–160, 2002.

[18] M. Hicks, M. Finnicum, S. King, M. Martin, and J. Smith. Overcoming an untrusted computing base: Detecting and removing malicious hardware automatically. In *IEEE Symposium on Security and Privacy (SP)*, pages 159–172, 2010.

[19] T. Huffmire, B. Brotherton, G. Wang, T. Sherwood, R. Kastner, T. Levin, T. Nguyen, and C. Irvine. Moats and drawbridges: An isolation primitive for reconfigurable hardware based systems. In *IEEE Symposium on Security and Privacy (SP)*, pages 281–295, 2007.

[20] X. Jiang, P. Dutta, D. Culler, and I. Stoica. Micro power meter for energy monitoring of wireless sensor networks at scale. In *International Conference on Information Processing in Sensor Networks (IPSN)*, pages 186–195, 2007.

[21] Y. Jin and Y. Makris. Hardware Trojan detection using path delay fingerprint. In *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST)*, pages 51–57, 2008.

[22] R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor. Trustworthy hardware: Identifying and classifying hardware Trojans. *IEEE Computer Magazine*, 43(10):39–46, 2010.

[23] F. Koushanfar and A. Mirhoseini. A unified framework for multimodal submodular integrated circuits Trojan detection. *IEEE Transactions on Information Forensics and Security*, 6(1):162–174, 2011.

[24] F. Koushanfar and M. Potkonjak. CAD-based security, cryptography, and digital rights management. In *Design Automation Conference (DAC)*, pages 268 –269, 2007.

[25] M. Kuhn. An asymmetric security mechanism for navigation signals. In *Information Hiding Workshop (IH)*, pages 239–252, 2004.

[26] L. Lazos and R. Poovendran. SeRLoc: secure range-independent localization for wireless sensor networks. In *ACM workshop on Wireless security (WiSe)*, pages 21–30, 2004.

[27] J. Li and J. Lach. At-speed delay characterization for IC authentication and Trojan horse detection. In *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST)*, pages 8–14, 2008.

[28] M. Malinowski, M. Moskwa, M. Feldmeier, M. Laibowitz, and J. Paradiso. CargoNet: a low-cost micropower sensor node exploiting quasi-passive wakeup for adaptive asychronous monitoring of exceptional events. In *International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 145–159, 2007.

[29] D. Markovic, C.C. Wang, L.P. Alarcon, Tsung-Te Liu, and J.M. Rabaey. Ultralow-power design in near-threshold region. *Proceedings of the IEEE*, 98(2):237–252, 2010.

[30] S. Meguerdichian and M. Potkonjak. Device aging-based physically unclonable functions. In *Design Automation Conference (DAC)*, pages 288–289, 2011.

[31] S. Meguerdichian and M. Potkonjak. Matched public PUF: Ultra low energy security platform. In *International Symposium on Low Power Electronics and Design (ISLPED)*, pages 45–50, 2011.

[32] M. Potkonjak. Synthesis of trustable ICs using untrusted CAD tools. In *Design Automation Conference (DAC)*, pages 633–634, 2010.

[33] M. Potkonjak, S. Meguerdichian, and J.L. Wong. Trusted sensors and remote sensing. In *IEEE Sensors*, pages 1104–1107, 2010.

[34] A. Sadeghi, I. Visconti, and C. Wachsmann. Anonymizer-enabled security and privacy for RFID. In *International Conference on Cryptology and Network Security (CANS)*, pages 134–153, 2009.

[35] S.R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas. VARIUS: A model of process variation and resulting timing errors for microarchitects. *IEEE Transactions on Semiconductor Manufacturing*, 21(1):3–13, 2008.

[36] T. Stathopoulos, D. Mclntire, and W.J. Kaiser. The energy endoscope: Real-time detailed energy accounting for wireless sensor nodes. In *International Conference on Information Processing in Sensor Networks (IPSN)*, pages 383–394, 2008.

[37] B.E. Stine, D.S. Boning, and J.E. Chung. Analysis and decomposition of spatial variation in integrated circuit processes and devices. *IEEE Transactions on Semiconductor Manufacturing,*, 10(1):24–41, 1997.

[38] I. Sutherland, B. Sproull, and D. Harris. *Logical effort: designing fast CMOS circuits*. Morgan Kaufmann, 1999.

[39] P. Syverson. A taxonomy of replay attacks. In *Computer Security Foundations Workshop (CSFW)*, pages 187–191, 1994.

[40] M. Tehranipoor and F. Koushanfar. A survey of hardware Trojan taxonomy and detection. *IEEE Design Test of Computers*, 27(1):10–25, 2010.

[41] A. Waksman and S. Sethumadhavan. Silencing hardware backdoors. In *IEEE Symposium on Security and Privacy (SP)*, pages 49–63, 2011.

[42] S. Wei, S. Meguerdichian, and M. Potkonjak. Gate-level characterization: Foundations and hardware security applications. In *Design Automation Conference (DAC)*, pages 222–227, 2010.

[43] S. Wei, S. Meguerdichian, and M. Potkonjak. Malicious circuitry detection using thermal conditioning. *IEEE Transactions on Information Forensics and Security*, 6(3):1136–1145, 2011.

[44] S. Wei and M. Potkonjak. Scalable segmentation-based malicious circuitry detection and diagnosis. In *International Conference on Computer-Aided Design (ICCAD)*, pages 483–486, 2010.

[45] S. Wei and M. Potkonjak. Integrated circuit security techniques using variable supply voltage. In *Design Automation Conference (DAC)*, pages 248–253, 2011.

[46] S. Wei and M. Potkonjak. Scalable consistency-based hardware Trojan detection and diagnosis. In *International Conference on Network and System Security (NSS)*, pages 176–183, 2011.

[47] S. Wei and M. Potkonjak. Scalable hardware Trojan diagnosis. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2011.

[48] J.B. Wendt and M. Potkonjak. Nanotechnology-based trusted remote sensing. In *IEEE Sensors*, pages 1213–1216, 2011.

[49] F. Wolff, C. Papachristou, S. Bhunia, and R.S. Chakraborty. Towards Trojan-free trusted ICs: Problem analysis and detection scheme. In *Design, Automation and Test in Europe (DATE)*, pages 1362–1365, 2008.