Gate Characterization Using Singular Value Decomposition: Foundations and Applications

Sheng Wei, Ani Nahapetian, *Member, IEEE*, Michael Nelson, Farinaz Koushanfar, *Member, IEEE*, and Miodrag Potkonjak, *Member, IEEE*

***Abstract*—Modern hardware security has a very broad scope ranging from digital rights management to the detection of ghost circuitry. These and many other security tasks are greatly hindered by process variation, which makes each integrated circuit (IC) unique, and device aging, which evolves the IC throughout its lifetime. We have developed a singular value decomposition (SVD)-based procedure for gate-level characterization (GLC) that calculates changes in properties, such as delay and switching power of each gate of an IC, accounting for process variation and device aging. We employ our SVD-based GLC approach for the development of two security applications: hardware metering and ghost circuitry (GC) detection. We present the first robust and low-cost hardware metering scheme, using an overlapping IC partitioning approach that enables rapid and scalable treatment. We also map the GC detection problem into an equivalent task of GLC consistency checking using the same overlapping partitioning. The effectiveness of the approaches is evaluated using the ISCAS85, ISCAS89, and ITC99 benchmarks. In hardware metering, we are able to obtain probabilities of coincidence in the magnitude of 10^{-8} or less, and we obtain zero false positives and zero false negatives in GC detection.**

***Index Terms*—Gate-level characteristics, ghost circuitry, hardware metering, process variation, singular value decomposition.**

I. Introduction

THE scope and challenges of modern security and, in particular, hardware and system security, are ever increasing. From the application point of view, in addition to the traditional tasks of ensuring privacy in data communication and storage, there are tasks such as digital rights management (DRM), reverse engineering (RE) and prevention of RE, trusted synthesis and manufacturing, and detection of ghost circuitry (GC) [2], [3].

From the system and hardware point of views, there are numerous factors that make enforcement of security policies difficult. They include the ever-increasing level of integration, the ever-decreasing ratio between the number of input/output pins and the number of transistors, and the continuously decreasing delays and other transistor characteristics. However, the dominant conceptual and practical challenges are posed by the unavoidable uncertainty induced during manufacturing and operation.

Specifically, the difficulty of numerous hardware security tasks is greatly complicated by intrinsic phenomena associated with deep submicron implementation technologies: process variation (PV) and device aging. PV makes each transistor and each gate have unique properties (e.g., effective channel length, threshold voltage, and delay) across integrated circuit (ICs) of the same design. Device aging results in many of these properties being altered over the IC's lifetime. Hence, PV makes ghost circuitry detection difficult, and device aging makes hardware metering more complex.

The foundation for our approach is gate-level characterization (GLC) using a set of nondestructive timing and/or power measurements. The measurements are treated as a set of linear equations solved using singular value decomposition (SVD) to precisely calculate the factors by which each gate of the circuit differs from its nominal value in terms of power and/or delay. The use of SVD is motivated by its strong numerical properties that ensure its accuracy in presence of measurement errors. We present two applications of our approach, namely hardware metering (HM) and ghost circuitry detection.

Hardware metering is a digital rights management technique that aims to protect the interests of designers against nonauthorized fabrication, dissemination, and use of their ICs. HM techniques can be classified into two groups: active and passive. Active techniques provide mechanisms for remote enabling, disabling, and control of ICs. They often induce nontrivial realization and operational overhead. Passive HM approaches provide only detections of DRM. However, they have no design, implementation, and operational overheads. Furthermore, while PV is essential for a majority of passive HM techniques, device aging is a strong and, until recently, unaddressed challenge. HM techniques often use PV extraction and manipulation for DRM.

Manuscript received April 15, 2011; revised October 06, 2011; accepted December 04, 2011. Date of publication December 23, 2011; date of current version March 08, 2012. This work was supported in part by the NSF under Award CNS-0958369, Award CNS-1059435, and Award CCF-0926127. This paper was presented in part at the 11th Information Hiding Conference (IH 2009). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ramesh Karri.

S. Wei and M. Potkonjak are with the Computer Science Department, University of California, Los Angeles, CA 90095 USA (e-mail: shengwei@cs.ucla.edu; miodrag@cs.ucla.edu).

A. Nahapetian is with the Computer Science Department, California State University Northridge (CSUN), Northridge, CA 91330 USA, and also with the Computer Science Department, University of California, Los Angeles, CA 90095 USA (e-mail: ani@csun.edu).

M. Nelson is with Computer Science Department, University of California, Los Angeles, CA 90095 USA and with Wilshire Associates, Santa Monica, CA 90401-1085 USA (e-mail: mike@thinkingpart.com).

F. Koushanfar is with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77251 USA (e-mail: farinaz@rice.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2011.2181500

However, device aging alters all gate properties, such as delay and energy consumption, that are used to form IC identification (ID). In order to eliminate this problem, we propose to use a gate characterization approach based on switching power that is not impacted by aging. In addition, in order to minimize the time required for measurements and to create an economically viable, passive HM approach, we find parts of the IC that are best suited for rapid ID extraction using GLC.

GC insertion is the malicious addition of hardware in the specification and/or implementation of an IC by an attacker intending to change circuit functionality. There are numerous GC insertion sources, including untrusted foundries, synthesis tools and libraries, testing and verification tools, and configuration scripts [4], [5]. The intentional hardware alteration of the design specification and IC implementation only affect the circuit's functionality in a few and rare specific circumstances and are hidden otherwise. GC is more difficult to detect than design bugs or manufacturing faults, since it is intentionally implanted to be unperceivable by the state-of-the-art debugging/testing methodologies and tools. In a GC insertion attack, the adversary adds one or more gates such that the functionality or the correctness of the design is altered. The gates can be added so that no measured path is altered between primary inputs and flip-flops (FFs), between FFs, and between FFs and primary outputs. However, leakage power is always altered. Even if the attacker gates the added circuitry, the gating requires an additional gate. Our goal here is to detect the insertion of GC, specifically added gates, in the face of low controllability and observability of gates. The GC detection approach is generic enough that it can easily be retargeted to other circuit components, such as interconnect, by considering more comprehensive timing and/or power models.

We use leakage power as the side channel to detect GCs. Our idea is based on the fact that any inserted GCs cause systematic bias in the leakage power, even if they are inactive, or they are inserted wisely and do not impact the speed or functionality of the ICs. The main technical obstacle to GC detection is PV, as it has a significant impact on gate timing and power characteristics across ICs. The detection of ghost circuitry is accomplished using consistency analysis, where the intersection gates of large, overlapping parts of the IC are examined. Any inconsistency, namely the calculated properties of the gates in the intersection, serves as an indicator of GC, because of the fact that the bias caused by GCs in different segments impacts the GLC process differently. Furthermore, since the number of segments increases linearly as the size of the circuit, this partitioning also implies linear scalability of the GC detection approach.

We evaluate the proposed GLC, hardware metering, and GC detection approaches using a set of ISCAS85, ISCAS89, and ITC99 benchmarks. In GLC, our characterization error is smaller than the power measurement error. In hardware metering, we show that we are able to obtain probabilities of coincidence in the magnitude of 10^{-8} to 10^{-141}. Also, we obtain zero false positives and zero false negatives in the accuracy of GC detection under 1% measurement errors.

The remainder of the paper is organized as follows. We start by summarizing the related research and presenting the background of process variation, gate-level characterization, hardware metering, and ghost circuitry detection. We then introduce our SVD-based approach for gate-level characterization. Finally, we present the first device-aging-resilient hardware metering technique and our new approach to detect ghost circuitry. For both applications, we provide simulation results.

II. Related Work

In this section we briefly summarize existing literature along three related directions of research: 1) gate-level characterization; 2) hardware metering; and 3) ghost circuitry detection.

A. Gate-Level Characterization (GLC)

GLC has enabled a variety of hardware-based security applications, such as hardware metering [6], [7], hardware Trojan detection [8] and the creation of physically unclonable functions (PUFs) [9], [10]. The critical importance of GLC resulted in creation of a great variety of conceptually, statistically, and algorithmically different techniques, including: 1) direct measurements methods [11]; 2) field-programmable gate array (FPGA) reconfiguration-based approaches [12], [13]; 3) schemes that embed and observe dedicated IC structures and specialized circuity [14]; and 4) nondestructive universal techniques that employ global measurements and calculate scaling factors of each gate by solving a system of equations [15]–[18].

Among all the existing GLC approaches, the nondestructive techniques have recently drawn a great deal of attention in the hardware security research. The main strength of the nondestructive GLC approaches is their universal applicability, zero overhead, applicability to both power and delay measurements, and low cost. However, it was difficult to make the approaches scalable to large designs in the modern IC industry and, more importantly, they were not able to characterize significant percentages of gates in the circuit.

B. Hardware Metering

The impetus for the creation of hardware metering techniques was provided by the confluence of PV-based intrinsic IDs, hardware watermarking and more specifically fingerprinting (i.e., assignment or extraction of a unique watermark to each IC), world wide web (WWW) metering, and the horizontal IC manufacturing model. A decade ago, the first hardware metering technique was proposed [19]. The main idea of the first HM approach was to use fingerprinting as the postprocessing step to make each IC unique. With the emergence of PV, several active HM schemes have emerged that leverage physically unclonable functions (PUFs) [20], [21]. Somewhat earlier, a less secure active HM scheme was proposed that directly employs PV [22]. Alkabani *et al.* [23] proposed the first PV-based hardware metering scheme.

Our new HM technique is also passive and leverages PV through the use of GLC. However, it has at least two major innovations and advantages. First, we intentionally characterize only a small part of gates that are selected in such a way that very few measurements are required for accurate ID extraction. Second, the new technique is the first HM that is resilient against device aging.

C. Ghost Circuitry (GC) Detection

The initial techniques for GC detection assumed no process variation [24]. Therefore, conceptually clean and simple side channel-based techniques were very effective. They measured the overall power consumption of a small number of ICs for a given design. In order to ensure that the pertinent IC is GC free, the approach employed reverse engineering using destructive techniques. Then, the overall power profiles of the ICs used for reverse engineering were compared with other ICs of the same design. Power profiles contain data about the overall IC power consumption for different input vectors. The technique provides an adequate but slow and relatively expensive solution under the following assumptions: 1) no PV; 2) the availability of the IC for reverse engineering; 3) the IC either has or does not have GCs; and 4) no errors in measurements.

A number of subsequent papers exploited a significant but far from complete relationship between manufacturing testing and GC detection. For example, several early GC detection approaches tried to employ functional test techniques. For instance, researchers at Case Western University proposed the generation of test vectors that maximize the likelihood of GC detection, for GCs that consist of two-input gates that rarely switch [25]. Also, several automatic test pattern generation (ATPG) methods were employed within the divide-and-conquer paradigm [26]. Two types of GC detection techniques analyzed pertinent ICs in terms of their delay between flip-flops using either a deterministic or statistical way. UCLA [27], [28] and Rice [29] research groups advocate leakage current-based GC detection techniques.

A variety of GC detection techniques is presented and analyzed in a comprehensive survey that summarizes early GC detection efforts completely [2]. Probably the two most surprising assumptions in many GC detection papers are that the authors assume that both an IC with and without malicious circuitry are available and that they have identical PV characteristics for all gates. We employ a consistency-based divide-and-conquer paradigm that imposes no PV assumptions and that enables fast (linear time) scalability.

III. Preliminaries

In this section, we introduce the system models that we use in the SVD-based GLC approach, including process variation, measurement, and threat models.

A. Process Variation Model

Process variations (PVs) are due to the intense industrial CMOS feature scaling. With scaling of feature sizes, the physical limits of the devices are reached and uncertainty in the device size increases [30]. Variations in transistor feature sizes and, thus, in gate characteristics, e.g., delay or power, are inevitable. In present and pending technologies, the variation is large compared to the device dimensions. As a result, VLSI circuits exhibit a high variability in both delay and power consumption. In this work, process variation in gates is modeled as a multiplicative scaling factor.

For the evaluation of our approach, we select 45-nm technology and the variabilities in terms of effective channel length and threshold voltage (i.e., level of doping) as indicated in Asenov's paper [31]. Also, in order to capture the spatial correlations of gate-level properties (e.g., inter-chip, die-to-die, wafer-to-wafer, systematic, and random), we adopt two models by Cline *et al.* [32], namely principal component analysis (PCA) and quad-tree models.

B. Measurement Model

We employ IDDQ- and IDDT-based tests to measure the total leakage and switching power of the circuit [33], [34]. We note that all power measurements are subject to errors that can have significant impact on GLC accuracy. However, with modern measuring techniques and tools, the measurement errors can be controlled to a very small effect. For example, as discussed in Kocher's work in 1999 [35], well-equipped electronic labs have equipment that can digitally sample voltage differences at a rate of over 1 GHz with less than 1% error. More recently, there are accurate and inexpensive measuring instruments that are available in the market to minimize the measurement errors. For example, the power source measurement unit by National Instruments [36] is capable of reducing the measurement errors to the range of $10^{-4} \sim 10^{-5}$. In the simulation of GLC, we select the conservative estimate of 1% as the measurement error rate, in order to show that our GLC approach is accurate even under relatively large measurement errors. We model this value in our linear equations and have examined a uniform model for the measurement error in our work.

C. Threat Model

Since semiconductor manufacturing demands a large capital investment, the role of contract foundries has dramatically grown, increasing exposures to: 1) theft of masks; 2) attacks by insertion of malicious circuitry; and 3) unauthorized excess fabrication. The development of hardware security techniques is difficult due to reasons that include: 1) limited controllability and observability (50 000+ gates for each I/O pin in modern designs); 2) large size and complexity (the newest Intel processor has 2.06B transistors); 3) variety of components (e.g., clock tree, and finite state machine); 4) unavoidable design bugs; 5) possibility of attacks by nonphysically connected circuitry (e.g., using crosstalk and substrate noise); 6) many potential attack sources (e.g., hardware intellectual property (IP) providers, CAD tools, and foundries); 7) potentially sophisticated and well-funded attackers (foundries and foreign governments); and 8) manufacturing variability that makes each IC coming from the same design unique [37].

In this paper, we assume the attackers can embedded ghost circuitry, even as little as a single gate. This insertion can occur at various stages of the IC manufacturing process, including through CAD tools, through the use of outside IP, and at the foundry during the fabrication process [38]. In general, the attacker can carry out many different types of hardware attacks, including gate resizing, removing gates, and allowing crosstalk. However, in this paper, we consider ghost circuitry attacks that

obtain information from the IC, implying that at least one gate is inserted.

IV. Singular Value Decomposition for Gate-Level Characterization

A. Problem Formulation

We model the PV in power or delay behavior of gates by associating each gate with a scaling factor, α, which multiplies the nominal power and delay of the gate. Measurements of total power and path delay for various circuit inputs give rise to linear equations with the scaling factors as the unknowns. Each set of measurements produces a linear system $Ga = m + e$ where a is the vector of scaling factors, also referred to as the α-values, and related to gate size; $m+e$ is a vector of measured values; m is the measured value if there is no measurement error; e is the measurement error associated with each measured value and G is derived from the expected power and/or delay characteristics of the gates.

For N_g number of gates in the circuit and N_m number of measurements, G is $N_m \times N_g$, a is $N_g \times 1$, and m is $N_m \times 1$. More abstractly, one can imagine that the circuit's gate characteristics are split into two components represented by G and a. G represents the characteristics of gate classes, i.e., two-input NANDs power and delay characteristics for a given input vector, and it is inherent to the circuit design. This information is readily available, and in our experiments we have used the values provided by Yuan *et al.* [39] for leakage power. The vector a, which is a vector of α-values for all the gates in the circuit, represents the unknowns in the equation. In other words, a is the fingerprint for the circuit just as the α-value is the fingerprint for the individual gate. Due to PV, gate sizes are not exactly matched to the design specifications. The size of each gate in the circuit of each fabricated IC can have a variety of values. All circuits accordingly will have a large variety of sizes for most or all of their gates, and hence the extremely large combinations of possibilities of a result in a unique fingerprint for each circuit. Splitting each manufactured circuit into an invariant and into a variant component results in G, which is universal across all circuits of the same design for the same set of input vectors, and a, which represents the unique characteristics of the fabricated circuit.

A large set of measurements are taken for the total circuit. As we can only access the input and output pins of the circuit, all the measurements, represented by $m + e$, are made from a global circuit or path level and not at the individual gate level. Obviously, if we were able to measure these values at the gate level, we would easily be able to solve for each gate's α-value.

We do consider error in the formulation, as measurement error is possible when measuring total leakage power for the circuit and total delay along a path of the circuit from input to output pin. This error is represented by e, which is the error that may be introduced in the measurement for each input vector or pair of input vectors.

A singular value decomposition $G = U\Sigma V_T$ is used in the following way. G^+, the pseudo-inverse of G, gives a least squares solution to the system; a', an approximation of the scaling factors, gives the possibility of measurement errors being introduced. The procedure for fingerprinting circuits, i.e., determining the α-values as accurately as possible, is as follows: 1) Choose a set of circuit inputs. 2) Compute G and G^+. 3) Perform measurements on a circuit to produce $m + e$. 4) Compute the fingerprint $a' = G^+(m + e)$.

In this formulation, a' represents the fingerprint that we deciphered from the SVD. It does not necessarily match a, due to the measurement error and also due to gate correlations that hinder gate-level characterization.

In Sections IV-B and C, we provide not only the power models, but also a complete example that we solve to demonstrate more clearly the procedure followed to accomplish gate-level characterization.

TABLE I
Power Matrix for Example Circuit Given in Fig. 1
(Leakage Current in nA)

Input Vector	Gate 1	Gate 2	Gate 3	Gate 4
000	37.84	37.84	37.84	454.5
001	100.3	37.84	37.84	454.5
010	95.17	100.3	100.3	454.5
011	454.5	100.3	100.3	95.17
100	37.84	95.17	95.17	454.5
101	100.3	95.17	95.17	454.5
110	95.17	454.5	454.5	100.3
111	454.5	454.5	454.5	37.84

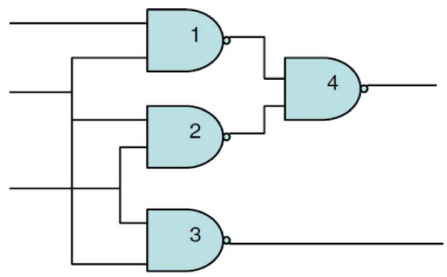

Fig. 1. Example circuit with NAND gates used to demonstrate SVD-based gate-level characterization.

B. Power Model

Equation (1) is the gate-level leakage power model [40], where L is effective channel length, $V_{\rm th}$ is threshold voltage, W is gate width, V_{dd} is supply voltage, n is substreshold slope, μ is mobility, $C_{\rm ox}$ is oxide capacitance, ϕ_t is thermal voltage $\phi_t = kT/q$, and σ is drain induced barrier lowering (DIBL) factor

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

The total leakage power consumed by a circuit is the sum of the leakage power of its gates [39]. For a particular circuit input i and a measurement ML_i of total leakage power with input i, we have the equation, $\sum_g GL_{gi}\alpha_g = ML$, where GL_{gi} is the expected leakage current for gate g when the global input is i. Each equation contributes a row to G and an entry to m in the overall system, $Ga = m + e$. Table I shows a matrix G computed from the example circuit in Fig. 1, using input-dependent leakage values from Yuan *et al.* [39], shown in Table II.

The gate-level switching power model [40] is described by (2), where the switching power is dependent on gate width W,

TABLE II
INPUT-DEPENDENT LEAKAGE CURRENT FOR TWO-INPUT NAND GATE

Input Vector	Leakage Current
00	37.84 nA
01	95.17 nA
10	100.3 nA
11	454.5 nA

gate length L, and supply voltage V_{dd}. p is the switching probability that indicates how often the gate switches, and $C_{\rm ox}$ is the oxide capacitance

$$P_{\rm switching} = p \cdot C_{\rm ox} \cdot W \cdot L \cdot V_{dd}^2. \tag{2}$$

The reason why we use the switching power model, in addition to leakage power, is that the switching power does not depend on $V_{\rm th}$, which is a major component that is impacted by the gate aging effect [41]. This enables us to design a stable GLC approach independent of the time when the IC is characterized and eliminates the need of characterizing gate-level physical properties, such as $V_{\rm th}$. Similar with the formulations in the leakage power and the delay model, we can formulate a linear equation in terms of the nominal switching power values and the scaling factor α

$$\sum_g GS_{gi} \cdot \alpha_g = MS_i \tag{3}$$

where GS_{gi} is the switching power of gate i when input vector pair i is applied to the target circuit; α_g is the scaling factor of gate g; and MS_i is the measured total switching power of the target circuit. Note that when a certain input vector pair i is applied, there is only a part of the gates in the circuit that switch, and consequently, only a part of the gates will have a nonzero GS_{gi} and appear in the linear equation. However, by changing the input vector pair i and obtaining different equations, we can cover more gates in the circuit.

C. Computing Scaling Factors

The equations generated from leakage and/or delay measurements are combined into the system $Ga = m + e$. Recall that N_g is the number of gates in the circuit, and N_m is the number of measurements. A singular value decomposition of G has the form $G = U\Sigma V^T$, where V is $N_g \times N_g$ and orthogonal, U is $N_m \times N_m$ and orthogonal, and Σ is $N_m \times N_g$ and diagonal; the entries on its diagonal are the singular values. The rank of G is equal to the number of nonzero singular values; by convention, we assume that the nonzero singular values are in the left-most columns of Σ.

The pseudo-inverse of G is $G^+ = V\Sigma^+U^T$, where Σ^+ is derived from Σ by replacing each nonzero singular value σ with its inverse $1/\sigma$. Performing the multiplication $G^+(m+e)$ gives our fingerprint a', the vector in the column space of G for which the norm of $Ga' - (m+e)$ is minimized. The fingerprint vector a' has the following properties: 1) If G has rank N_g, then a' is an approximation of a. 2) If G has rank $<N_g$, then a' is an approximation of the portion of a which is not annihilated by G.

TABLE III
α-VALUE FINGERPRINT OBTAINED FOR EXAMPLE CIRCUIT

a	m	a'
1.015	537.4	1.015
1.103	600.7	1.025
0.9473	723.6	1.025
0.9271	755.0	0.9271
	654.9	
	718.2	
	1121	
	1428	

Table III shows an example for the circuit in Fig. 1. The measurement vector m is computed from this a and the power matrix in Table I and the resulting fingerprint vector a'. Because this matrix is not full rank, some α-values are inaccurate, even though we did not add any measurement error.

As shown in Fig. 1, gates 2 and 3 are both two-input NANDs, and they both have the same input vector in all possible measurements, as they both have by design the same input vectors. As a result, it is not possible to separately characterize gates 2 and 3 since their G matrix entries will be the same for all inputs vectors. The best that is achievable is to characterize the sum of their α-values, which in this case is 2.050, and it has been properly characterized. This demonstrates how even without measurement error, it is possible to improperly fingerprint a circuit in some cases.

The task of determining the input vectors that are applied for taking the measurements is not as straightforward as it seems. First, due to the prohibitive size of the input vector domain, an exhaustive search can only be applied to the smallest of circuits. Second, certain input vectors will maximize the solution quality, while others may be redundant or even obfuscate the true value. For large circuits, a set of input vectors must be chosen that maximizes the rank of G. We have used the following heuristics in our work in this paper: 1) start with an empty G; 2) choose a random input vector and compute its matrix row; 3) if the row is independent of the existing rows of G, add it to G (increasing G's rank); and 4) repeat from Step 2).

Since we do not know the maximum possible rank in advance, this process must be repeated until some arbitrary stopping condition is met, such as some number of failed choices in a row. For numerical robustness, N_m should be larger than $\mathrm{rank}(G)$, and more random inputs can be added afterward to accomplish this condition.

V. IC METERING USING GLC

IC metering is the process of detecting and preventing a foundry from producing more chips than specified in the contract. There have been two types of metering approaches, namely extrinsic metering and intrinsic metering. Extrinsic metering inserts extra components into the IC design, either with a new hardware component or with a programmable module, which can generate unique IDs or fingerprints for the manufactured ICs. The unique IDs can be used to distinguish the ICs from each other, and thus differentiate unauthorized IC copies from authorized ones. Intrinsic IC metering generates unique IDs without having to modify the IC design. Instead, it characterizes the gate-level characteristics of an IC and uses

them to uniquely represent the chip. This approach leverages the presence of process variation, which naturally exists in the IC manufacturing process and which results in all ICs being unique and different in their nominal design properties.

The existing approaches for IC intrinsic metering can detect IP violations in IC manufacturing. However, according to the power and delay model, the existing approaches that use leakage power or delay as the unique IDs are highly dependent on the threshold voltage $V_{\rm th}$. On one hand, $V_{\rm th}$ is subject to PV, which ensures the uniqueness of the IDs among all the chips from the same design, but on the other hand, the value of $V_{\rm th}$ is subject to the negative bias temperature instability (NBTI) aging effect [42], and hence increases as the IC ages. Aging would change the unique ID of the IC over time, which not only violates the purpose of IC identification, but also increases the probability of coincidence between chip IDs.

In order to eliminate the aging impact in IC metering, we use switching power as the side channel for IC identification, since switching power is not subject to aging effect according to (2). The problem we face in switching-power-based IC metering is that only a subset of the gates can be characterized in terms of switching power, because only the gates that switch consume switching power. However, we note that the probability of coincidence (i.e., ID collision) that two different chips end up having an identical ID is extremely low by using the combination of switching power of even a small number of gates in the circuit. In particular, we can quantify the probability of coincidence using a similar calculation with the birthday paradox problem [43], where the probability of coincidence that two different ICs have the identical ID decreases exponentially as the number of properties that forms the ID. Therefore, for IC metering purpose, not all the gates are required to construct a unique ID for a specific chip that has a low probability of coincidence with other chips. We can characterize only a small subset of gates for their switching power values to ensure that there are no ID collisions in sufficiently large amount of chips of the same design.

VI. Consistency-Based Ghost Circuitry Detection Using Segmented GLC

Based on the characterized gate-level leakage power and switching power, we are able to detect whether there is any malicious circuitry embedded in the target circuit. The idea is to capture the possible systematic bias caused by the malicious circuitry in the leakage power consumption. There are two technical issues that we must address in order to achieve accurate GC detection results. Firstly, we must be able to handle the case where the GC is ultra small (e.g., as small as one single gate) compared to the large number of gates (e.g., in the magnitude of millions) in modern IC designs in terms of leakage or switching power consumptions. In this case, the variation can be easily hidden under the process variation or indistinguishable from the characterization errors in the GLC process. Secondly, due to the existence of process variation, we cannot assume there is a clean version of the IC that is free of GC attacks for each single chip and, therefore, there is no standard to compare against to conclude the existence of GC.

We employ a segmentation-based GC detection approach that resolves both issues without introducing additional overhead into the process. Fig. 2 shows the flow of our GC detection scheme. First of all, the GC detection approach leverages a divide-and-conquer paradigm that partitions the target IC into several small segments. Each segment serves as the basic unit for GC detection. Consequently, the size of the problem can be greatly reduced, which also improves the accuracy of detection, since the relative systematic bias (i.e., the leakage power increase caused by GC compared to the total leakage power consumption) is larger in the segmented subcircuits. Fig. 3(a) shows an example of circuit segmentation via input vector control. Segment 1 (i.e., gates X1-X4) and Segment 2 (i.e., gates X4-X7 and GC gate Z) are controlled solely by inputs (I1, I2) and inputs (I3, I4), respectively. Therefore, if we vary inputs (I1, I2) and freeze (I3, I4) during input vector creation, only the leakage power of Segment 1 would vary. The leakage power of Segment 2 would stay constant, which can be represented by a single variable and cancelled out from the equations. Similarly, Segment 2 can be separated and characterized by freezing inputs (I1, I2) and varying inputs (I3, I4). After segmentation, the number of gates being characterized in the SVD-based GLC process is greatly reduced compared to all the gates in the circuit, e.g., by 42.9% in both segments, as shown in Fig. 3(a). The GC detection process can cover all possible locations in the target IC by conducting GC detection iteratively in all the segments.

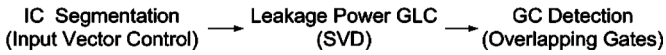

Fig. 2. Flow of segmentation and SVD-based GC detection.

Next, in each segmented subcircuit, we conduct SVD-based gate-level characterization to determine the leakage power scaling factor of each gate, following the procedure introduced in Section IV. In the segment that is GC free, the GLC process is normal and would provide us with accurate scaling factor values that reflect the process variations. However, in the segment with GC embedded, the systematic bias caused by the GC would increase the total leakage power consumption and, consequently, the characterized scaling factors would be inconsistent with the ones caused purely by PV. Furthermore, since we do not assume the availability of a GC-free circuit, we identify at least one representative gate (R gate) in each segment such that the R gate belongs to at least two segments (i.e., the R gate is an overlapping gate between at least two segments). Then, we repeatedly conduct GLC in all the segments that contain the R gate and compare the characterization results. The scaling factors of the R gate in different segments can serve as a built-in indicator of whether GC exists or not, in the sense that an inconsistency of the characterized values in different segments would indicate the presence of GC in at least one of the segments. Note that in the case where GC exists in multiple overlapping segments, since the number of gates, gate types, and thus the leakage power consumptions are different in different segments, we would still obtain inconsistent R gate scaling factors that reflect the different relative systematic bias of leakage power in the segments. Fig. 3(b) shows an example of consistency-based GC detection in the two overlapping segments. We first characterize all the scaling factors in both

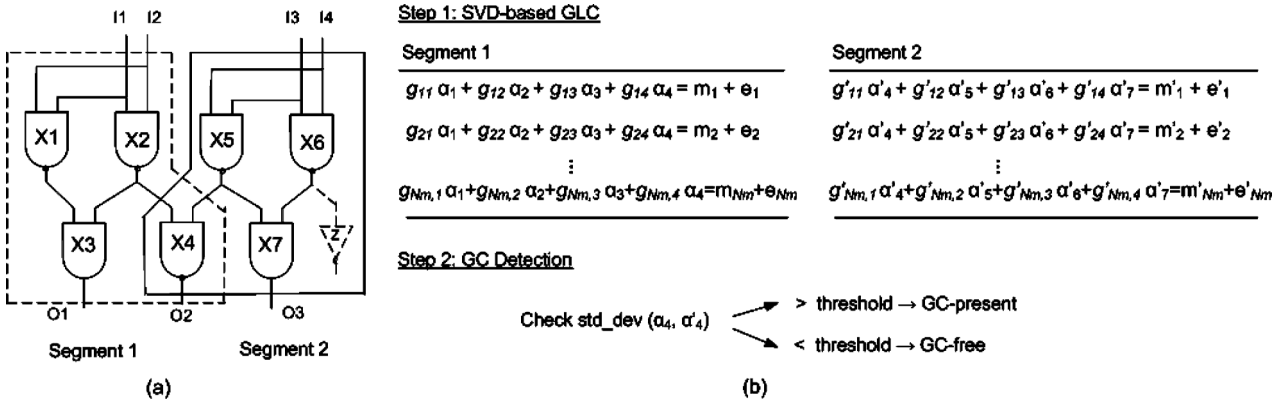

Fig. 3. Example of GC detection using segmentation and SVD-based GLC. (a) Segmentation of a circuit with seven regular gates (X1-X7) and one GC gate Z via input vector control (gate X4 is the R gate that belongs to both segments). (b) GLC and GC detection steps, where N_m is the number of measurements, g_{ij} and g'_{ij} are the nominal power values in Segment 1 and Segment 2, respectively, $(i = 1 \ldots N_m, j = 1 \ldots 4)$, α_k and α'_k are the leakage power scaling factors $(k = 1 \ldots 7)$, m_i and m'_i are the measured power values $(i = 1 \ldots N_m)$, and e_i and e'_i are the measurement errors $(i = 1 \ldots N_m)$. (a) Segmentation. (b) GC detection.

segments using the SVD-based GLC approach. Then, we check the scaling factor of the R gate (i.e., gate X4). If the standard deviation of the scaling factors in all the overlapping segments is larger than the predefined threshold, we can conclude that a GC exists in at least one of the segments being examined.

VII. Simulation Results

A. Switching-Power-Based GLC

We simulate our switching-power-based GLC on ISCAS85, ISCAS89 and ITC99 benchmarks. We generate the IC instances in the simulations following Asenov's PV model [31] and Cline's quad-tree model [32] for spatial correlations. For each benchmark circuit, we apply a set of input vectors (e.g., 1024 input vectors in our simulation) per segment and measure the total switching power for each of them to formulate the system of linear equations. After GLC, we evaluate the accuracy of characterization using the relative characterization error as follows:

$$E_i = \frac{|\alpha_{calc_i} - \alpha_{\mathrm{real_i}}|}{\alpha_{real_i}} \tag{4}$$

where E_i is the relative characterization error of gate i, and $\alpha_{\mathrm{calc_i}}$ and $\alpha_{\mathrm{real_i}}$ are the calculated scaling factor of gate i and its real value, respectively. Then we use the average relative error to evaluate the GLC accuracy

$$E_{\mathrm{avg}} = \frac{1}{n}\sum_{i=1}^{n} E_i \tag{5}$$

where n is the number of gates in the circuit.

Table IV shows our simulation results where all GLC errors are around or below 1%. For each benchmark circuit, there is a part of the gates that cannot be addressed due to their rare switching or correlations with other gates in the circuit. However, as we discussed in Sections V and VI, for the purposes of GC detection and hardware metering, there is no need to characterize all the gates in the circuit. By characterizing only a subset of the gates, we can greatly save the running time and enable the scalability of our approach. In Sections VII-B and C, we demonstrate our simulation results of the two security applications based on the switching power gate-level characterization.

TABLE IV
Accuracy of Switching-Power-Based GLC

Benchmark	Gates	Characterized Gates	GLC Error (%)
C432	160	152	0.33
C499	202	162	0.18
C880	383	369	1.01
C1355	546	500	0.91
C1908	880	355	0.09
C2670	1193	598	0.13
C3540	1669	878	0.29
C5315	2307	1334	0.07
S526	72	37	0.57
S820	290	145	0.50
S823	226	80	0.69
S38417	22179	13176	0.45
S38584	19253	12861	0.36
b17	27852	13703	0.73
b18	94249	50611	0.56
b19	231266	81637	0.89

B. Aging-Independent IC Metering

We evaluate our switching-power-based IC metering approach by analyzing the resulting probability of coincidence that two chips have identical IDs. As discussed in Section V, due to the large numbers of gates in the benchmark circuits, it is not required that all the gates are characterized in order to generate IDs that provide low probability of coincidence. Therefore, in our simulation, we select a small segment of gates from each circuit and use the combination of their switching power scaling factors as the ID. The method we use to segment the circuit is discussed in [44] and [45], in which we freeze a subset of the inputs and vary the others to obtain a segment of the circuit. Table V shows the details of the segment we use for each benchmark circuit as well as the results of probability of coincidence analysis. By characterizing less than 100 gates for each circuit, we are able to obtain probabilities of coincidence in the magnitude of 10^{-8} or less. These results indicate that

TABLE V
PROBABILITY OF COINCIDENCE IN AGING-INDEPENDENT IC METERING

Benchmark	# Gates	# Selected Inputs	# Selected Gates	Prob. Coincidence
C432	160	10	19	3.6E-12
C499	202	10	22	5.7E-14
C880	383	10	40	8.3E-25
C1355	546	10	43	1.3E-26
C1908	880	10	21	2.3E-13
C2670	1193	30	27	5.6E-17
C3540	1669	8	47	5.0E-29
C5315	2307	8	26	2.2E-16
S526	72	30	13	1.5E-8
S820	290	15	42	5.2E-26
S832	226	15	44	3.2E-27
S38417	22179	15	56	2.4E-59
S38584	19253	15	18	1.5E-11
b17	27852	18	63	3.3E-93
b18	94249	18	72	5.8E-129
b19	231266	18	80	1.4E-141

TABLE VI
GC DETECTION RESULTS USING CONSISTENCY-BASED GLC: VALUES IN "GC-FREE" AND "GC-PRESENT" COLUMNS REPRESENT STANDARD DEVIATION OF R GATE SCALING FACTORS IN DIFFERENT SEGMENTS

Benchmark	Number of Gates	GC-Free	GC-Present
C432	160	2.3E-3	8.8E-2
C499	202	1.4E-2	0.20
C880	383	8.7E-3	7.3E-2
C1355	546	1.1E-2	0.27
C1908	880	6.0E-3	0.23
C2670	1193	2.9E-3	0.13
C3540	1669	9.7E-3	0.12
C5315	2307	1.3E-2	0.12
S526	72	1.7E-3	1.30
S820	290	2.2E-2	0.39
S823	226	5.6E-4	2.72
S38417	22179	2.3E-2	0.29
S38584	19253	2.8E-3	0.24
b17	27852	7.6E-3	0.59
b18	94249	3.7E-3	0.40
b19	231266	5.9E-3	0.38

there are no identical IDs for at least 100 million chips that are metered using our approach. Also, with less than 100 gates for characterization, the size of the problem is well controlled, which ensures the scalability of our approach.

C. GC Detection

In our simulation of GC detection, we add a single NAND gate sized to half of its nominal size and place it at the signal which most often is 0 so that induced additional leakage is minimal. Note that even an inverter has larger leakage power than NAND gate (e.g., the minimal leakage power for inverter is 100.3 nA, while that for NAND is 37.84 nA) [39]. We use this test case because it is the most difficult case for GC detection. As for the location of the GC, we use a random location on the circuit in our simulation in order to test the reliability and coverage of our approach.

The simulation results for consistency-based GC detection are given in Table VI. In each of the benchmark circuits, we first select segments that overlap with one or more gates and that cover all the gates in the circuit. Then, we conduct GLC for each segment in terms of leakage power to characterize the scaling factors of the R gates. Finally, we calculate the standard deviation of the scaling factors for the R gates in all the segments. For each benchmark, we conduct simulations for both cases where the circuit is GC free and where there is GC embedded. The "GC-Free" and "GC-Present" columns in the table show the corresponding standard deviations in the two cases. As we can observe from the results, there is a more than 15X gap between the two cases, across all the benchmark circuits. Therefore, by creating a decision line (threshold value) within this large gap, we can obtain zero false positives and zero false negatives in GC detection.

VIII. CONCLUSION

Process variation and device aging are intrinsic to modern and pending silicon implementation technologies and are major impediments to a wide spectrum of hardware and system security tasks such as hardware metering and ghost circuitry detection. In order to address these two security tasks in the presence of process variation and device aging, we have developed a new gate-level characterization procedure that employs singular value decomposition. Based on the GLC, we have developed the first robust hardware metering scheme that enables rapid and low cost enforcement of digital rights enforcement. Also, we accomplished ghost circuitry detection by analyzing the consistency of the characterized scaling factors in overlapping segments. Our statistical analysis indicates that any inconsistency larger than a very low threshold is a reliable sign of the presence of ghost circuitry. The effectiveness of the techniques is evaluated using a set of ISCAS85, ISCAS89, and ITC99 benchmarks.

REFERENCES

[1] M. Nelson, A. Nahapetian, F. Koushanfar, and M. Potkonjak, "SVD-based ghost circuitry detection," *Inform. Hiding (IH)*, pp. 221–234, 2009.
[2] M. Tehranipoor and F. Koushanfar, "A survey of hardware Trojan taxonomy and detection," *IEEE Design Test Comput.*, vol. 27, no. 1, pp. 10–25, Jan. 2010.
[3] M. Potkonjak, S. Meguerdichian, A. Nahapetian, and S. Wei, "Differential public, physically unclonable functions: Architecture and applications," in *Proc. Design Automation Conf. (DAC)*, 2011, pp. 242–247.
[4] S. King, J. Tucek, A. Cozzie, C. Grier, W. Jiang, and Y. Zhou, "Designing and implementing malicious hardware," in *Proc. Usenix Workshop Large-Scale Exploits and Emergent Threats (LEET)*, 2008, pp. 8–8.
[5] C. Sturton, M. Hicks, D. Wagner, and S. King, "Defeating UCI: Building stealthy and malicious hardware," in *Proc. IEEE Security Privacy (SP)*, 2011, pp. 64–77.
[6] S. Wei, A. Nahapetian, and M. Potkonjak, "Robust passive hardware metering," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2011.
[7] S. Wei, F. Koushanfar, and M. Potkonjak, "Integrated circuit digital rights management techniques using physical level characterization," in *Proc. ACM Workshop Digital Rights Management (DRM)*, 2011, pp. 3–14.
[8] S. Wei and M. Potkonjak, "Scalable consistency-based hardware Trojan detection and diagnosis," in *Proc. Int. Conf. Network System Security (NSS)*, 2011, pp. 176–183.
[9] S. Meguerdichian and M. Potkonjak, "Matched public PUF: Ultra low energy security platform," in *Proc. Int. Symp. Low Power Electronics Design (ISLPED)*, 2011, pp. 45–50.
[10] J. B. Wendt and M. Potkonjak, "Nanotechnology-based trusted remote sensing," *IEEE Sensors*, vol. 3, no. Mar., pp. 1213–1216, 2011.
[11] S. Smith, A. Tsiamis, M. McCallum, A. Hourd, J. Stevenson, A. Walton, R. Dixson, R. Allen, J. Potzick, M. Cresswell, and N. Orji, "Comparison of measurement techniques for linewidth metrology on advanced photomasks," *IEEE Trans. Semiconduct. Manuf.*, vol. 22, no. 1, pp. 221–234, Jan. 2009.

[12] M. Brown, C. Bazeghi, M. Guthaus, and J. Renau, "Measuring and modeling variability using low-cost FPGAs," in *Proc. Int. Symp. Field-Programmable Gate Arrays (FPGA)*, 2009, pp. 286–286.
[13] J. Wong, P. Sedcole, and P. Cheung, "Self-measurement of combinatorial circuit delays in FPGAs," *ACM Trans. Reconfigurable Technology Systems*, vol. 2, no. 2, pp. 1–22, 2009.
[14] A. Keshavarzi, G. Schrom, S. Tang, S. Ma, K. Bowman, S. Tyagi, K. Zhang, T. Linton, N. Hakim, S. Ouvall, J. Brews, and V. De, "Measurements and modeling of intrinsic fluctuations in mosfet threshold voltage," in *Proc. Int. Symp. Low Power Electronics Design (ISLPED)*, 2005, pp. 6–29.
[15] 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 *Proc. Design Automation Conf. (DAC)*, 2008, pp. 606–609.
[16] F. Koushanfar, P. Boufounos, and D. Shamsi, "Post-silicon timing characterization by compressed sensing," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2008, pp. 185–189.
[17] S. Wei, S. Meguerdichian, and M. Potkonjak, "Gate-level characterization: Foundations and hardware security applications," in *Proc. Design Automation Conf. (DAC)*, 2010, pp. 222–227.
[18] S. Wei, S. Meguerdichian, and M. Potkonjak, "Malicious circuitry detection using thermal conditioning," *IEEE Trans. Inform. Forensics Security*, vol. 6, no. 3, pp. 1136–1145, Jun. 2011.
[19] F. Koushanfar, G. Qu, and M. Potkonjak, "Intellectual property metering," in *Inform. Hiding (IH)*, 2001, pp. 81–95.
[20] Y. Alkabani and F. Koushanfar, "Active hardware metering for intellectual property protection and security," in *Proc. USENIX Security Symp.*, 2007, pp. 1–16.
[21] Y. Alkabani, F. Koushanfar, and M. Potkonjak, "Remote activation of ICs for piracy prevention and digital right management," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2007, pp. 674–677.
[22] M. Khan and S. Tragoudas, "A method for hardware metering," in *Proc. Int. Conf. Communications (ICCOM)*, 2005, pp. 4–4.
[23] Y. Alkabani, F. Koushanfar, N. Kiyavash, and M. Potkonjak, "Trusted integrated circuits: A nondestructive hidden characteristics extraction approach," in *Inform. Hiding (IH)*, 2008, pp. 102–117.
[24] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in *Proc. IEEE Symp. Security Privacy*, 2007, pp. 296–310.
[25] F. Wolff, C. Papachristou, S. Bhunia, and R. Chakraborty, "Towards Trojan-free trusted ICs: Problem analysis and detection scheme," in *Proc. Design, Automation, and Test in Europe (DATE)*, 2008, pp. 1362–1365.
[26] M. Banga and M. Hsiao, "A region based approach for the identification of hardware Trojans," in *Proc. Int. Symp. Hardware-Oriented Security and Trust (HOST)*, 2008, pp. 40–47.
[27] M. Potkonjak, A. Nahapetian, M. Nelson, and T. Massey, "Hardware Trojan horse detection using gate-level characterization," in *Proc. Design Automation Conf. (DAC)*, 2009, pp. 688–693.
[28] S. Wei and M. Potkonjak, "Integrated circuit security techniques using variable supply voltage," in *Proc. Design Automation Conf. (DAC)*, 2011, pp. 248–253.
[29] Y. Alkabani and F. Koushanfar, "Consistency-based characterization for IC Trojan detection," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2009, pp. 123–127.
[30] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Proc. Design Automation Conf. (DAC)*, 2003, pp. 338–342.
[31] A. Asenov, "Random dopant induced threshold voltage lowering and fluctuations in sub-0.1 μm MOSFETs: A 3-D atomistic simulation study," *IEEE Trans. Electron. Devices*, vol. 45, no. 12, pp. 2505–2513, Dec. 1998.
[32] B. Cline, K. Chopra, D. Blaauw, and Y. Cao, "Analysis and modeling of CD variation for statistical static timing," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2006, pp. 60–66.
[33] S. Sabade and D. Walker, "IDDX-based test methods: A survey," *ACM Trans. Design Automation Electronic Systems*, vol. 9, no. 2, pp. 159–198, 2004.
[34] R. Rajsuman, "Iddq testing for CMOS VLSI," *Proc. IEEE*, vol. 88, no. 4, pp. 544–566, Apr. 2000.
[35] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. Int. Cryptology Conf. (CRYPTO)*, 1999, pp. 388–397.
[36] N. I. Corporation, Pxi-4130 Power SMU 2011 [Online]. Available: http://sine.ni.com/nips/cds/view/p/lang/en/nid/204239
[37] A. Srivastava, D. Sylvester, and D. Blaauw, "Statistical analysis and optimization for VLSI: Timing and power," in *Series on Integrated Circuits and Systems*. New York: Springer, 2005.
[38] L. Lin, W. Burleson, and C. Parr, "Moles: Malicious off-chip leakage enabled by side-channels," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2009, pp. 117122–117122.
[39] L. Yuan and G. Qu, "A combined gate replacement and input vector control approach for leakage current reduction," *IEEE Trans. VLSI Syst.*, vol. 14, no. 2, pp. 173–182, Feb. 2006.
[40] D. Markovic, C. Wang, L. Alarcon, T. Liu, and J. Rabaey, "Ultra low-power design in near-threshold region," *Proc. IEEE*, vol. 98, no. 2, pp. 237–252, 2010.
[41] M. Agarwal, B. Paul, M. Zhang, and S. Mitra, "Circuit failure prediction and its application to transistor aging," in *VLSI Test Symp. (VTS)*, 2007, pp. 277–286.
[42] C. Young, R. Choi, J. Sim, B. Lee, P. Zeitzoff, Y. Zhao, K. Matthews, G. Brown, and G. Bersuker, "Interfacial layer dependence of HFSIxOy gate stacks on VT instability and charge trapping using ultra-short pulse in characterization," in *Proc. Int. Reliability Physics Symp. (IRPS)*, 2005, pp. 75–79.
[43] P. Flajolet, D. Gardy, and L. Thimonier, "Birthday paradox, coupon collectors, caching algorithms and self-organizing search," *Discrete Appl. Math.*, vol. 39, no. 3, pp. 207–229, 1992.
[44] S. Wei and M. Potkonjak, "Scalable segmentation-based malicious circuitry detection and diagnosis," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, 2010, pp. 483–486.
[45] S. Wei and M. Potkonjak, "Scalable hardware Trojan diagnosis," *IEEE Trans. Very Large Scale Integration (VLSI) Syst.*, to be published.

Sheng Wei is working toward the Ph.D. degree in computer science from the University of California, Los Angeles.

His research interests include computer-aided design of VLSI circuits, hardware security, and wireless network.

Ani Nahapetian (S'03–M'07) received the B.S. degree in computer science and engineering, and the M.S. and Ph.D degrees in computer science all from the University of California, Los Angeles (UCLA).

She is an Assistant Professor of computer science at the California State University, Northridge, and an Assistant Adjunct Professor with the Computer Science Department, UCLA. Her research interests include hardware-based system security, remote health monitoring systems, and algorithm design for embedded systems.

Michael Nelson received the M.S. degree in computer science from University of California, Los Angeles, in 2009.

He currently works in financial industry.

Farinaz Koushanfar (S'99–M'06) received the M.A. degree in statistics and Ph.D. degree in electrical engineering and computer science both from University of California, Berkeley.

She is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Rice University, Houston, TX, where she directs the Texas Instruments DSP Leadership University Program. Her research interests include adaptive and low power embedded systems design, hardware security, and design intellectual property protection.

Miodrag Potkonjak (M'02) received the Ph.D. degree in electrical engineering and computer science from University of California, Berkeley, in 1991.

He is a Professor with Computer Science Department, University of California, Los Angeles. He created the first watermarking, fingerprinting, and metering techniques for integrated circuits, as well as first remote trusted sensing and trusted synthesis approaches, compilation using untrusted tools, and public physical unclonable functions.