Malicious Circuitry Detection Using Thermal Conditioning

Sheng Wei, Saro Meguerdichian, and Miodrag Potkonjak, *Member, IEEE*

***Abstract*—Gate-level characterization (GLC) is the process of quantifying physical and manifestational properties for each gate of an integrated circuit (IC). It is a key step in many IC applications that target cryptography, security, digital rights management, low power, and yield optimization. However, GLC is a challenging task due to the size and structure of modern circuits and insufficient controllability of a subset of gates in the circuit. We have developed a new approach for GLC that employs thermal conditioning to calculate the scaling factors of all the gates by solving a system of linear equations using linear programming (LP). Therefore, the procedure captures the complete impact of process variation (PV). In order to resolve the correlations in the system of linear equations, we expose different gates to different temperatures and thus change their corresponding linear coefficients in the linear equations. We further improve the accuracy of GLC by applying statistical methods in the LP formulation as well as the post-processing steps. In order to enable non-destructive hardware Trojan horse (HTH) detection, we generalize our generic GLC procedure by manipulating the constraint of each linear equation. Furthermore, we ensure the scalability of the approaches for GLC and HTH detection using iterative IC segmentation. We evaluate our approach on a set of ISCAS and ITC benchmarks.**

***Index Terms*—Gate-level characterization (GLC), hardware Trojans, process variation.**

I. Introduction

As the scaling of high-performance integrated circuits (ICs) moves to deep-submicron (DSM) feature sizes, a higher degree of semiconductor integration provides ever increasing performance. However, simultaneously new challenges have been imposed on IC design and analysis. The consequences of deep-submicron technologies include exponentially increasing leakage energy, increased substrate noise, profound, and intrinsic process variation (PV), and increased susceptibility to environmental (e.g., thermal) and operational (e.g., supply voltage) variations. Among them, PV has emerged as the most limiting factor that essentially redefines IC synthesis and analysis flow.

Manuscript received September 29, 2010; revised April 26, 2011; accepted April 30, 2011. Date of publication May 23, 2011; date of current version August 17, 2011. This work was supported in part by the National Science Foundation under Award CNS-0958369, Award CNS-1059435, and Award CCF-0926127. An earlier version of this paper [1] was presented at the 47th Design Automation Conference (DAC '10). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ramesh Karri.

The authors are with the Computer Science Department, University of California, Los Angeles (UCLA), Los Angeles, CA 90095 USA (e-mail: shengwei@cs.ucla.edu; saro@cs.ucla.edu; miodrag@cs.ucla.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.2011.2157341

PV may be defined as the deviation of IC devices and overall metrics from nominal specifications. For example, it has been reported that the frequency of a chip designed in 90-nm technology can vary by up to 30% from its nominal design values [2]. For leakage current, the variations are much higher and may reach up to 20+X [2].

In particular, PV has a large impact on hardware and system security. Malicious attacks, such as hardware Trojan horses (HTHs) [3], are difficult to detect because otherwise easy to observe variations caused by added HTHs cannot be distinguished easily from those created by PV. Therefore, in order to detect HTHs, an accurate implicit or explicit characterization of the IC subject to PV is required.

Gate-level characterization (GLC) is the process of characterizing each gate of an IC in terms of its physical properties, such as gate width and effective channel length, or its manifestational properties, such as power and delay. Several research efforts [4]–[6] have proposed conceptually different non-destructive GLC techniques. However, none of them is capable of characterizing all gates due to insufficient diversity of linear equations that correspond to power or delay measurements. To the best of our knowledge, this is the first report of a technique that guarantees complete GLC.

We have developed a new approach for GLC that employs actuating side channels for IC conditioning. The crucial observation is that leakage energy increases rapidly and exponentially with temperature. Selective thermal conditioning exposes different subsets of gates to different temperatures and thus change their coefficients in the system of linear equations in addition to the changes produced by the different input vectors. It provides a simple and practical way to break the correlations in the system of equations and to increase the controllability over the gates. Hence, thermal conditioning enables conduction of accurate and complete GLC regardless of the structure and functionality of the design.

Our starting point for HTH detection is the following simple but essential observation. An intelligently placed HTH cannot be observed using delay analysis because it can be inserted so that it does not impact any critical path or path that becomes critical after its insertion. Also, the HTH can be embedded in such a way that it does not influence the overall switching energy because it may be activated only after a long period of time. However, any HTH always impacts leakage currents. Its leakage current impact can be very small for large circuits. Therefore, we need techniques that can greatly amplify this impact to enable detection. We have created three such techniques. The first is that we use a large number of measurements in order to detect a small systematic signal in much larger "random noise."

The second is that we select as input vectors ones that induce a majority of gates to have low leakage. Finally and most importantly, we impact the leakage currents of gates exponentially using thermal conditioning.

We detect HTHs by imposing additional constraints on the objective function and each of the GLC linear equations. Therefore, we use constraint manipulation that superimposes additional constraints onto the problem formulation in such a way that any produced solution has a specified set of features. New terms are added to the constraints of the linear programming (LP) formulation for GLC in such a way that their values indicate whether a HTH is added or not. Specifically, we add an additional variable that corresponds to HTH presence or absence. If a HTH is not present, the system of linear equations is consistent and the added variable will have zero or small value. If a HTH is present, each equation will be biased and LP solvers will assign nonzero value to the superimposed term.

Our major research contributions include the following:

- a thermal conditioning-based approach that ensures that scaling factors of all gates can be characterized accurately;
- the use of statistical methods, namely maximum-likelihood estimation (MLE), to improve the accuracy of characterization;
- a segmentation-based technique that ensures the scalability of the GLC approach by using iterative IC segmentation;
- a systematic method for HTH detection based on the thermally conditioned GLC that leverages a simple but effective constraint manipulation technique.

II. Related Work

In this section, we briefly review directly related GLC and HTH research. GLC techniques can be classified into four major groups: 1) direct measurements approaches; 2) schemes that employ FPGA reconfiguration; 3) approaches that create and observe special IC structures and specialized circuity; and 4) non-destructive techniques that conduct global measurements and deduce scaling factors of each gate by solving a system of equations.

Direct measurement techniques use atomic force microscopes (AFM), electric line measurements (ELM), and optical instruments to directly measure critical dimensions (e.g., effective channel length) [7]. They are very accurate, have a wide range of speeds (e.g., AFM-based techniques are much slower than ELM), and their application is often restricted to the measurements of critical dimensions.

FPGA GLC techniques iteratively create blocks of clock measurement circuitry and isolate a block under characterization to conduct delay characterization of gates and wires [8]. The third group of techniques populate chips with simple structures such as ring oscillators and delay lines that can be easily characterized in terms of gate delay through clock sweeping and counting techniques [9]. The main limitation of these two types of techniques is that they can be applied only to specific types of designs.

Finally, non-destructive GLC techniques can be divided into two classes. The first class does not impose any assumptions about spatial correlation of gate scaling factors [5], [10]–[12]. The second class uses spatial correlation, transformations, and techniques such as compressed sensing to reduce cardinality of the system of equations [4].

IBM researchers proposed one of the first techniques for HTH detection. There is a significant but far from complete relationship between manufacturing testing and HTH detection. Several early HTH detection approaches tried to employ functional test techniques. For example, Case Western researchers proposed generation of test vectors that maximize the likelihood of detecting HTHs that consist of 2-input gates that rarely switch [13]. Also, several automatic test pattern generation (ATPG) techniques were employed within the divide-and-conquer paradigm [14]. Two types of HTH detection techniques analyzed pertinent ICs in terms of their delay from one flip-flop to another using either deterministic [15] or statistical methods [16]. A number of HTH detection techniques advocate the use of switching power measurements [17], [18]. Researchers from UCLA [6] advocate leakage current-based HTH detection techniques. A comprehensive recent survey of HTH detection is presented in [3].

III. Preliminaries

In this section, we introduce the system models used by GLC and HTH detection approaches, including power and delay, process variation, and power measurements models.

A. Power and Delay Models

We adopt leakage power as the side channel for constructing the system of linear equations in GLC and HTH detection. In addition, The temperature characterization that is required in our thermal conditioning scheme is based on leakage power, switching power, and delay measurements. Equation (1) is the gate-level leakage power model [19], where L is effective channel length, V_{th} is threshold voltage, W is gate width, V_{dd} is supply voltage, n is substreshold slope, μ is mobility, C_{ox} is oxide capacitance, ϕ_t is thermal voltage $\phi_t = kT/q$, and σ is drain induced barrier lowering (DIBL) factor as follows:

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

The gate-level switching power model [19] is specified in (2), where the switching power is dependent on oxide capacitance C_{ox}, gate width W, effective channel length L, switching probability α, and supply voltage V_{dd} as follows:

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

Equation (3) shows the gate-level delay model [19], where C_L is load capacitance, k_{fit} is a model-fitting parameter, and k_{tp} is the delay-fitting parameter as follows:

$$Delay = \frac{k_{tp} \cdot C_L \cdot V_{dd}}{2 \cdot n \cdot \mu \cdot C_{ox} \cdot \frac{W}{L} \cdot \left(\frac{kT}{q}\right)^2} \cdot \frac{k_{fit}}{(ln(e^{((1+\sigma)V_{dd} - V_{th})/2 \cdot n \cdot (kT/q)} + 1))^2}. \tag{3}$$

B. Process Variation

We develop a GLC approach to characterize the impact of PV. The GLC method is generic and can characterize circuits of any arbitrary PV model. For the evaluation of our approach and the presentation of simulation results, we select 45-nm technology and the variabilities in terms of effective channel length and threshold voltage (level of doping) as indicated in Asenov's paper [20]. 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.* [21], namely principal component analysis (PCA) and quad-tree models.

There are two parameters that are directly impacted by PV: effective channel length (L) and threshold voltage (V_{th}) [20], [21]. For V_{th}, we adopt Gaussian distribution as proposed by Asenov *et al.* [20]. For L, we follow the quad-tree model proposed by Cline *et al.* [21] that considers the spatial correlations among gates. In the quad-tree model, the gate-level properties (e.g., effective channel length) subject to PV are distributed into multiple levels, with a different number of grids allocated on each level. The grids on each level are assigned variation values that follow a normal distribution. We calculate the total value of the target gate-level property as the sum of the variations on each level of the grids to which the corresponding gate belongs. We show the quad-tree model in (4), where Δs_{ij} is the quantitative variation of ith level and jth grid to which the gate belongs, and μ_i and σ_i are parameters of the normal distribution at level i, as follows:

$$\Delta s = \sum_i \Delta s_{ij}, \quad where \quad \Delta s_{ij} \sim N(\mu_i, \sigma_i). \tag{4}$$

C. Power Measurements

IDDQ and IDDT based tests for power measurements are widely used by the IC test community [22]. We note that all measurements are subject to errors, which have significant impact on GLC accuracy. For example, as discussed in Kocher's work [23], 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 [24] is capable of reducing the measurement errors to the range of 10^{-4}–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 further improve the accuracy greatly by applying more accurate power measuring instruments that are available and inexpensive. We formulate the measurement errors as follows:

$$\tilde{p} = (1 + e_s + e_r)p \tag{5}$$

where $\tilde{p}$ is the measured power, p is the actual power, e_s is the systematic error in the measurement of each gate, which is consistent over all gates, and e_r is the random error that is caused by random factors in the measurement. In selection of a random distribution for e_r, we consider two principles: 1) that the proposed distribution must be accurately addressed by linear programming; and 2) that combinations of this distribution with different parameters must be such that all interesting, popular, and relevant other distributions can be easily composed by the selected distribution and posed to a linear program that optimizes a piecewise linear objective function and can still be solved in polynomial time by LP solvers. Under these considerations, we select a triangular distribution where the probability density of error can be formulated as follows:

$$p(e_r) = \begin{cases} \frac{2(e_r - a)}{(b-a)(c-a)}, & \text{for } a \le e_r \le c \\ \frac{2(b - e_r)}{(b-a)(b-c)}, & \text{for } c \le e_r \le b \end{cases} \tag{6}$$

where a, b, and c are parameters for the triangular distribution, namely minimum, maximum and most likely errors, respectively. Furthermore, we set the parameters in such a way that the expected mean error over all gates is a constant error rate ϵ_e that we specify. In particular, we have the following equation for calculating the parameters of a triangular distribution

$$|e_s| + \int_a^b |e_r| p(e_r)\, de_r = \epsilon_e. \tag{7}$$

IV. Gate Level Characterization

The starting point for our HTH detection is GLC. Our goal in GLC is to characterize the scaling factor of each gate using a set of leakage power measurements of the entire circuit.

A. Desiderata

We have two primary GLC desiderata:

- *Accuracy.* The results of characterization must be accurate, i.e., the difference between characterized scaling factors and the actual values is minimal. Since measurement errors are naturally present, we must filter out the error noise.
- *Number of characterized gates.* Our goal is to characterize all the gates on the target circuit so that we are able to detect HTH gates at any locations. In most cases, this objective is challenging due to the fact that there are a large portion of gates in the circuit that are of the same type and have low observability.

B. Overall Flow

We start our GLC approach by applying k different input vectors that are stored in flip-flops to the combinatorial logic and measure the total leakage power of the circuit for each of them. Next, we generate a system of k equations and formulate a linear program. The objective function is to minimize the sum of the absolute value ($l1$-norm) of measurement errors, as shown in (8), where m is the number of measurements, and e_i is the error of the ith measurement.

$$\min \quad \sum_{i=1}^{m} |e_i|. \tag{8}$$

The system of linear equations (constraints) has the following form:

$$K \cdot s = \tilde{p} + e \tag{9}$$

where $K \in R^{m \times n}$ is the matrix of coefficients for leakage power scaling factors that are impacted by gate types and their input vectors. m is the number of measurements, and n is the number of gates on the chip. s, $\tilde{p}$, and e are one-dimensional vectors representing the scaling factor of each gate, the measured power, and the unknown measurement error in each measurement, respectively. The format of (9) meets that of linear constraints in a LP. Note that we abstract the impact of PV on leakage power (or switching power, delay) using a scaling factor s_i for each gate i in the circuit. Our goal in GLC is to characterize the scaling factors of all gates by solving the LP. After obtaining the scaling factors, we can recover the manifestational properties due to PV from their nominal design values.

The flow of GLC and HTH detection includes three phases, namely pre-processing, GLC, and post-processing, as shown in Fig. 1. In the pre-processing phase, we first generate the IC instances that take into account the impact of PV. We combine the PV model for individual devices presented by Asenov's group [20] with the spatial correlation model proposed in [21]. Next, we generate a set of input vectors in the GLC measurement phase. The goal during input vector generation is to make the maximum number of gates, gate pairs, and gate triplets have all their possible input signal combinations, so that the possibility of linear dependencies in the system of linear equations is minimized. Also, we heat up the circuit selectively by switching certain gates on the circuit in such a way that all the remaining correlations in the system of linear equations can be resolved. After pre-processing, we begin the process of GLC, in which we apply the set of input vectors and measure the total leakage power for each of them. For each measurement value, we formulate a linear equation by summing up the leakage power of each gate and considering the measurement errors. For HTH detection, we further add a single HTH variable in each of the linear equations that represents the presence of HTHs. Next, we solve the system of linear equations using a LP solver and obtain results for each PV scaling factor as well as the value of HTH variable in the case of HTH detection.

We repeat the GLC procedure k times (in our simulations $k = 50$) and conduct post-processing using the obtained results. We apply maximum-likelihood estimation (MLE) that selects the most likely scaling factor for each gate as our eventual results of GLC. Finally, we employ statistical methods to validate our prediction results. For this purpose we use resampling, where 60% of the GLC results are used for the training set and 40% for the testing set. The entire GLC procedure terminates when the validated GLC accuracy is within a user predefined threshold value.

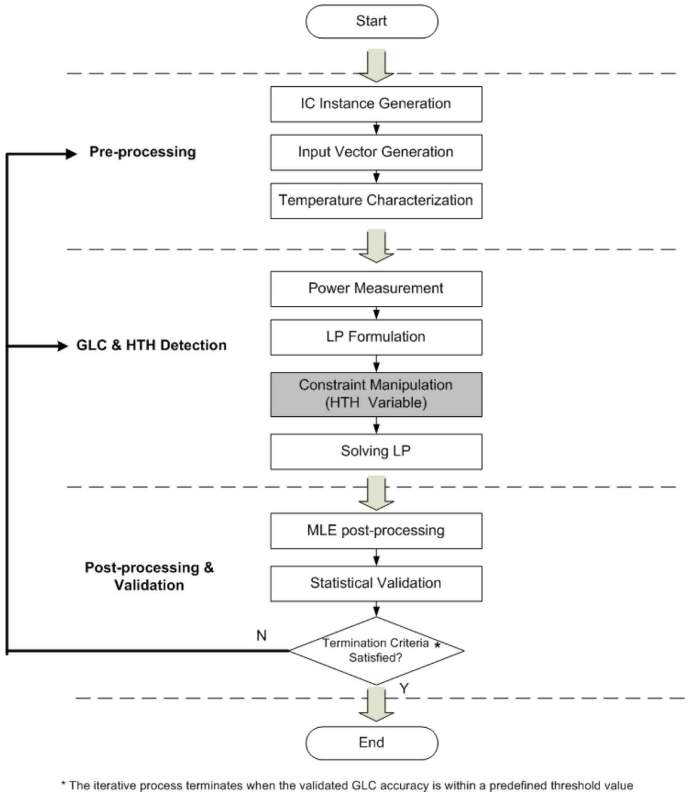


Fig. 1. Overall flow of our GLC and HTH detection scheme. We use a three-phase procedure (pre-processing, GLC, and post-processing). The shaded part is specific for the HTH detection procedure.

C. Measurements and Equations

1) Technical Issues: The structure of matrix K and thus the formulation of the LP are highly dependent on the choice of input vectors. In order to characterize the scaling factors accurately, one must minimize the dependencies amongst the variables in the system of equations. A simple way of achieving the goal is to create as many equations as possible. However, this technique has two strong negative ramifications:

- *Large-Size LP.* The number of gates is large in most of the benchmarks. Hence, the formulated LP may easily exceed the processing power of LP solvers.
- *Correlations.* Even if we are able to handle a large number of equations, ideally all the possible input vectors, we still cannot characterize the gates that are correlated in the system of linear equations. We define two types of correlations: 1) between gates that always have the same ratio of coefficients (collinearity correlation); and 2) between gates for which we are not able to obtain a sufficient number of independent equations because the number of variables is larger than the number of equations (insufficient controllability). Collinearity correlation usually occurs when multiple gates are of the same type and always have the same input states. The insufficient controllability correlation is a consequence of IC structure when a subpart of the circuit has many gates but few intermediate inputs that control them.

The running times and coverage issues in the pertinent set of equations induce a need to reduce the size of the LP and to break the correlations. We address the first issue by preprocessing the input vectors in such a way that we maximize the number of unique coefficients in front of each variable. We address the second issue using correlation detection and thermal conditioning. We discuss correlation detection in the rest of this subsection. The thermal conditioning technique is covered in Section V.

2) Correlation Detection: Since the correlated variables in the system of linear equations cannot be solved by the LP solver,

we detect them so that we can either break the correlations or, in the worst case, combine those variables to reduce the size of the LP. We have developed two techniques to detect collinearity correlations. The first one is straightforward: we check the coefficients for all pairs of variables using exhaustive enumeration. If there exists a pair of gates for which the ratio of coefficients are identical over all the equations, the pair of gates is correlated.

The second technique employs our LP formulation itself. We add one more constraint in the LP formulation that sets one of the potentially correlated gates to a very large value; if correlations exist, the LP solution would show that several other variables become very small. Therefore, these gates are correlated with the gate whose value has been modified by the extra constraint.

For insufficient controllability correlations, the detection is not trivial, because the number of subparts of the circuit that can possibly have correlations is large. We solve this problem by manipulating the objective function. In particular, we change the objective function to maximize a single variable. If a subset of the gates in the circuit have insufficient controllability correlation, the other variables would become very small.

D. Improving the Objective Function

So far we have been using the $l1$-norm of measurement errors as the objective function in the LP. Although the $l1$-norm helps reduce large errors, it is not capable of leveraging the measurement error model. In order for the optimization process to follow exactly the measurement error distribution, we consider the likelihood function of the measurement error distribution in our objective function. Our goal is to find the solution of maximum likelihood. For the triangular distribution, the objective function is the following:

$$\max \quad l(e_r) = \sum_{i=1}^{m} \log(e_s + p(|e_r|)) \tag{10}$$

where $p(\cdot)$ is the probability density function of the triangular distribution. The new objective function is nonlinear and cannot be handled by the LP solver directly. Our solution is to create a piecewise linear function that approximates the nonlinear function. In particular, we find a subset of breakpoints on the nonlinear curve. By connecting these breakpoints using piecewise linear segments, we obtain a linear form of the objective function. Fig. 2 shows an example of the likelihood function and the piecewise linear approximation.

We treat finding the optimal breakpoints as an optimization problem. Specifically, we minimize the approximation error of the following form:

$$\min \quad d_N(a,b) = \int_a^b (l(e_r) - f(e_r))^2 de_r \tag{11}$$

where a and b are parameters of the triangular distribution, $l(e_r)$ is the nonlinear likelihood function, and $f(e_r)$ is the piecewise linear function determined by the breakpoints. In particular, we formulate $f(e_r)$ as the following expression:

$$f_i(e_r) = a_i e_r + b_i, \quad u_{i-1} \le e_r \le u_i \tag{12}$$

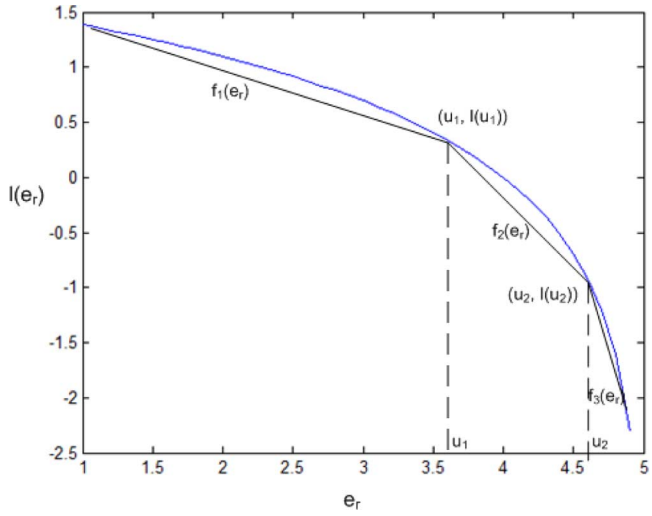


Fig. 2. Example of the likelihood function and piecewise linear approximation, where $l(e_r) = \log(5 - e_r)$, and two breakpoints u_1 and u_2 are being considered.

where $f_i(e_r)$ is the ith segment of the piecewise linear function. Suppose we have N breakpoints and thus $N+1$ linear pieces. $u_i (1 \le i \le N+1)$ are the breakpoints, with $u_0 = 0$ and $u_{N+1} = b$. a_i and b_i are parameters determined by u_{i-1} and u_i.

The problem of finding breakpoints in piecewise linear approximation can be solved provably optimally in polynomial time using dynamic programming [26]. The linearization is sufficient to obtain a piecewise linear representation of any arbitrary function. However, it requires the error function to be convex in order to guarantee the optimality of the results.

E. MLE Post-Processing

The results obtained from the LP are impacted by several factors including the precision of the LP solver and the accuracy of the power measurements. In order to obtain more accurate results, we post-process the result data using MLE. In particular, we repeat the GLC procedure k times and collect the results from the LP solver. Next, we apply goodness-of-fit tests on the data from each run, and estimate the statistical distribution of the scaling factors over different runs. According to the distribution that each scaling factor follows, we create its approximate density function, say $p(s_i)$, and set our estimated value of s_i to be the one that maximizes the following likelihood function:

$$\tilde{s_i} = argmax_{s_i} \log p(s_i). \tag{13}$$

We repeat the MLE post-processing over all the scaling factors and obtain the final GLC values.

V. Gate Level Characterization Using Thermal Conditioning

In this section, we discuss our thermal conditioning approach to GLC that resolves the correlation issues in the system of linear equations.

A. Technical Issues

As discussed in Section IV, there are two technical issues that we must resolve in GLC. First, if the target circuit is large, it requires a large number of measurements as well as a very large

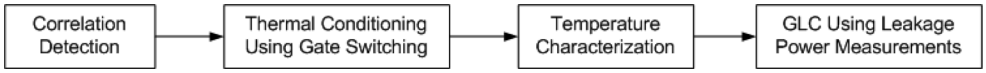


Fig. 3. Flow of thermal conditioning for GLC. We increase the temperatures of a subset of gates in the circuit to break the correlations in the system of linear equations.

LP that is difficult to solve. Second, we must break the correlations in the system of linear equations in order to characterize all the gates on the circuit. The only way to break these correlations is to find alternatives to supplement input vector variation for changing the coefficients of the scaling factor variables.

We solve both technical issues using thermal conditioning, where we heat up a subset of gates to change their coefficients in the system of linear equations. Our intuition is based on the fact that gate-level leakage power depends on the temperature of the gate [as shown in (1)] and that IC heat-up is much faster than the cooling process (as discussed in [25]). Therefore, thermal conditioning provides us with an additional way of controlling a subset of gates in the circuit and thus enables us to change the coefficients of the scaling factors in the system of linear equations. By using thermal conditioning, we can obtain sufficient number of equations regardless of the number of input vectors that can be applied. Furthermore, we are able to break the collinearity correlation, since we can obtain different coefficients even without changing the input vectors.

B. Flow of Thermal Conditioning

We show our flow of thermal conditioning in Fig. 3. We first conduct correlation detection using the techniques discussed in Section IV to determine the subset of gates that are either subject to collinearity correlation or insufficient controllability. Next, we perform thermal conditioning on the set of correlated gates by applying a set of input vectors that cause the gates to switch. The heat generated while switching increases the temperatures of the gates. In order to calculate the new coefficients of the scaling factors in the system of linear equations, we select a subset of gates on the circuit as the representative gates, which can provide us with the temperature profile of the entire circuit (as shown in Algorithm 1). We characterize the new temperatures of the subset of gates by measuring leakage power, switching power, and delay and solving a system of nonlinear equations following the power and delay models discussed in Section III. We utilize the characterized temperatures of the gates as representative temperatures and determine the temperature of each gate in the circuit under the consideration of heat transfer. Finally, we apply the new coefficients to GLC using leakage power measurements and characterize the scaling factor of each gate.

Algorithm 1 Temperature Characterization: we conduct leakage power, switching power, and delay measurements on the pertinent IC and formulate three nonlinear equations with variables T, L, and V_{th}. We calculate T by solving the three nonlinear equations.

Input: Target circuit for characterization.

Output: Temperature T_i of each gate i.

1: **repeat**

2: Select a set I of input vectors;

3: Determine (via correlation detection) S_{leakage}, the set of gates that do not have correlations with other gates in the system of linear equations in leakage power GLC;

4: Determine (via simulation) $S_{switching}$, the set of gates that switch with more than $k\%$ probability when I is applied;

5: Determine (via simulation) S_{delay}, the set of gates that are on critical path with more than $k\%$ probability when I is applied;

6: $S_{\text{rep}} = S_{\text{leakage}} \cap S_{switching} \cap S_{\text{delay}}$;

7: **until** $S_{\text{rep}} \neq \emptyset$;

8: Conduct leakage power, switching power, and delay characterization using GLC;

9: **for** each gate i in S_{rep} **do**

10: Formulate leakage power equation using **(1)**;

11: Formulate switching power equation using **(2)**;

12: Formulate delay equation using **(3)**;

13: Solve the three equations for T, L, and V_{th} using non-linear programming;

14: **end for**

15: Calculate the new temperatures T_i for all gates in the circuit using S_{rep};

16: **return** T_i;

1) Thermal Conditioning Using Gate Switching: We conduct input vector control to increase by different amounts the temperatures of the subset of gates that are subject to correlations in the system of linear equations. In particular, we select a set of input vectors in such a way that they can switch the set of correlated gates identified by correlation detection in different ways. The heat generated during switching increases the temperatures of the gates and thus change their coefficients in the system of linear equations. The key observation is that gate switching is very fast (on the order of nanoseconds), while the cooling process is much slower (on the order of seconds) [25]. Therefore, we can increase the temperatures of the subset of gates rapidly and assume that the new temperatures we obtain stay constant for seconds until we completely characterize all gates in GLC.

2) Temperature Characterization Using Leakage Power, Switching Power, and Delay Measurements: In order to calculate the new coefficients in the system of linear equations after thermal conditioning, we must determine the temperature profile of all the gates in the circuit. There are three variables in the gate-level properties, as shown in (1)–(3), temperature (T), effective channel length (L), and threshold voltage (V_{th}). From these three equations, we are able to solve for temperature T. The formulation of these three equations requires that we obtain gate-level leakage power, switching power, and delay. Our approach for temperature characterization is nondestructive and does not require complicated thermal models or thermal management tools. We first select a subset of gates for which

we can characterize all the three properties using GLC, i.e., the gates that are on critical path, switch often, and do not have correlations in the system of linear equations with other gates in terms of leakage power. Next, we conduct gate-level leakage power, switching power, and delay characterization of the selected gates using the GLC method (described in Section IV). After obtaining their gate level leakage power, switching power, and delay, we formulate three nonlinear equations according to (1)–(3) for each gate and solve for the variables T, L, and V_{th}. Finally, we use the temperatures of the selected gates as the representative temperatures and determine the temperature profile of the entire circuit, under the consideration that the gates that are physically close to each other on the circuit have similar temperatures due to heat transfer. We show the pseudocode for selecting the representative gates and calculating the temperatures in Algorithm 1.

3) Gate Level Leakage Power Characterization: After obtaining the new temperatures of the gates, we follow (1) to set the new coefficients in the system of linear equations. In this way, we are able to create various independent linear equations that break both types of correlations.

C. Scalability

Due to limitations of the LP solver, the aforementioned thermal conditioning based GLC scheme only works for a circuit with hundreds of gates. However, in the modern IC industry, most IC designs have many more gates, up to the magnitude of millions. Our main idea to ensure scalability is to apply a divide-and-conquer approach to the target circuit. In particular, we develop a segmentation-based technique that decomposes the large circuit into small subparts, each of which can be easily solved by the LP solver.

There are three ways that we have considered to segment the circuit: 1) varying a subset of the inputs and freezing the rest; 2) increasing the temperatures of a certain part of the circuit; or 3) using extra circuitry (e.g., multiplexers) to select a small part of the circuit each time. In each of these methods, the key idea is to freeze a large part of the circuit either accurately or approximately, so that we can represent it as a constant in the LP. Next, we treat a small varying part of the circuit as a separate circuit, which can be easily characterized by solving a small system of linear equations.

We focus on the first approach that applies input vector control (IVC) in order to freeze a large part of the circuit. We also adopt increasing the temperatures of the gates as a complementary approach in the case where a large number of gates are correlated in the system of linear equations and difficult to separate. Our IVC scheme for segmentation is based on the controllability list [27]. A controllability list for a gate is a input vector that sets that specific gate to either logic state 0 or 1. For our purpose, we obtain two controllability lists for each gate in the circuit, one for logic state 0 and one for logic state 1. The algorithm for determining a controllability list is a heuristic algorithm as discussed in [27]. We utilize the controllability lists to determine the input vectors to segment the circuit. Specifically, we select input vectors in such a way that a large amount of gates stay in low leakage power, and thus we are able to separate the remaining small part out of the circuit. Our segmentation-based approach is iterative; we treat the characterized results of known segments as inputs to the characterization of remaining segments. In this way, we ensure that the whole process converges and we can characterize all the gates in the circuit.

VI. HTH Detection Using GLC

A hardware Trojan horse (HTH) [3] is a malicious modification of an IC. The alteration may impact the functionality of the circuitry, change the original characteristics (e.g., propagation delay or leakage power), or even leak confidential information. HTHs may pose a significant threat to IC security. Due to the increasing trend in IC outsourcing, today's design and manufacturing is a global business. However, it is difficult to address this issue because in the current IC manufacturing model, foundries have complete access to the hardware specification and may conduct malicious modifications. Therefore, HTH detection after manufacturing is of high necessity and has become a main concern in the IC industry. A typical type of HTH attack is to add additional gates to the circuit [3].

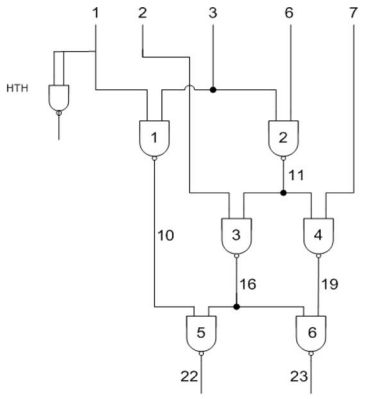
HTH detection is much more challenging than IC testing because attackers tend to hide the HTHs from common detection techniques. For example, attackers may embed a very small HTH that is activated only when a rare activation condition is satisfied. This kind of HTH would render the traditional functional testing method ineffective in detecting the HTHs, because it is extremely difficult for the test vectors to activate and capture the embedded HTHs.

Our starting observation is that regardless of the HTH type, switching activity, or placement strategy, the added gates always increase the total leakage energy. However, just observing the leakage energy is not sufficient due to the presence of PV. The key insight is that the extra HTH gates introduce a systematic bias in the total leakage power and, therefore, enable detection of any HTH by using systematic measurements.

Our idea is to introduce an extra component in the power measurement equations that captures the systematic bias caused by HTHs. Since we do not have any information about possible HTHs, such as their types, locations, or input signals, we abstract all HTHs into a single variable called HTH variable. In the process of HTH detection, we add this HTH variable to each of the linear equations regardless of whether or not any HTHs exist, which we do not actually know before the HTH detection procedure. We keep other parts of the linear equations unchanged.

Note that the most difficult case for HTH detection using leakage energy is when only one extra gate is added in the circuit, because it causes the least bias in leakage power and can be best hidden under PV or other sources of errors. Therefore, hereafter we only discuss and demonstrate the case where a single HTH gate is added by attackers.

In order to better illustrate our HTH detection technique, we show a HTH detection example in Fig. 4 on ISCAS benchmark circuit C17. We assume that the attacker may have added an extra HTH gate in the circuit. We use a NAND gate in this example as the HTH gate because NAND gates have the lowest leakage power among all the gates, and thus this is the hardest case for HTH detection. Our linear program to detect the HTH includes eight input vectors. We also add an extra variable z in each linear equation of leakage power measurement. In this



INPUT	Leakage (nA)
00	37.84
01	100.3
10	95.17
11	454.5

No HTHs → z = 0

HTHs exist → z = 256.4

$$
\begin{aligned}
&454.50\,x1 + 95.17\,x2 + 454.50\,x3 + 454.50\,x4 + 37.84\,x5 + 37.84\,x6 + z = 1846.11 + e1\\
&100.30\,x1 + 454.50\,x2 + 95.17\,x3 + 37.84\,x4 + 454.50\,x5 + 454.50\,x6 + z = 1821.31 + e2\\
&37.84\,x1 + 100.30\,x2 + 454.50\,x3 + 454.50\,x4 + 95.17\,x5 + 37.84\,x6 + z = 1358.40 + e3\\
&95.17\,x1 + 37.84\,x2 + 454.50\,x3 + 100.30\,x4 + 95.17\,x5 + 100.30\,x6 + z = 1176.99 + e4\\
&100.30\,x1 + 454.50\,x2 + 37.84\,x3 + 95.17\,x4 + 454.50\,x5 + 454.50\,x6 + z = 1722.24 + e5\\
&454.50\,x1 + 454.50\,x2 + 95.17\,x3 + 95.17\,x4 + 100.30\,x5 + 454.50\,x6 + z = 1866.29 + e6\\
&37.84\,x1 + 100.30\,x2 + 454.50\,x3 + 454.50\,x4 + 95.17\,x5 + 37.84\,x6 + z = 1305.09 + e7\\
&100.30\,x1 + 454.50\,x2 + 95.17\,x3 + 95.17\,x4 + 454.50\,x5 + 454.50\,x6 + z = 1797.66 + e8
\end{aligned}
$$

Fig. 4. Example of GLC-based HTH detection scheme on benchmark C17. The coefficients (nominal leakage power values) [28] are shown in the table. We add one extra HTH variable z to the system of measurement equations as the indicator of HTHs. e_i $(i = 1, 2, \ldots, 8)$ represents the systematic and random errors in leakage power measurement. The solution of z is zero when no HTHs are present and it is a large value (256.4) in the case where HTHs exist.

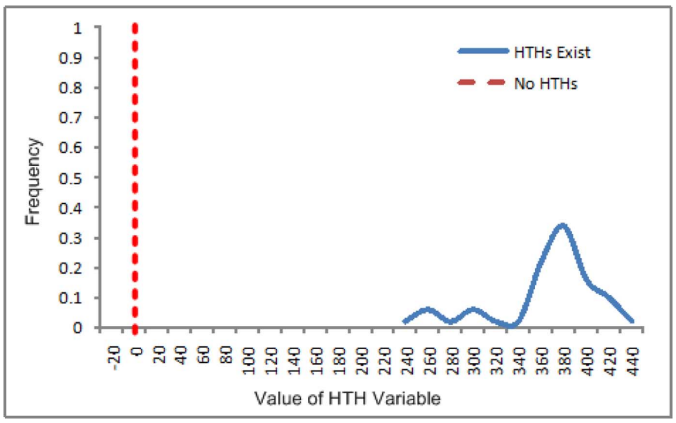


Fig. 5. Probability density function of HTH variable for 500 runs of HTH detection on benchmark C17. For all the 500 runs, the value of the HTH variable is 0 in the case where there are no HTHs, and it is a large value between 220 and 440 when HTHs are present.

example, after solving the system of linear equations using a LP solver, we find that the value for HTH variable z is 256.4; this indicates that the target circuit contains malicious circuitry. For the consideration of possible false positives in HTH detection, we further test our approach on an instance of C17 circuit without any HTHs. In that case, the obtained value for variable z is zero, which indicates that there is no systematic bias in the leakage power compared to its normal value, and thus there are no HTHs. The accuracy of our HTH detection technique is ensured by our GLC approach using thermal conditioning, which provides accurate characterization results for all the gate-level scaling factors when there is no malicious circuitry.

Furthermore, in order to evaluate the reliability of our HTH detection scheme, we repeat our simulation 500 times and plot the probability density function of the HTH variable in Fig. 5. In all 500 runs, the HTH variable is 0 when no HTHs exist, and it is a large value between 220 and 440 when HTHs are present. The results show a large gap in the values of the HTH variable between the two cases. Therefore, the HTH variable serves as a reliable indicator of HTH presence.

VII. Simulation Results

We evaluate our thermally conditioned GLC approach and HTH detection scheme on a set of ISCAS and ITC benchmarks. We use leakage power to characterize the scaling factors. We use the triangular distribution with mean value 1% as our measurement error model. We use as our LP solver lp_solve 5.5.

When evaluating the simulation results, we are guided by the objectives discussed in Section IV. The accuracy of characterization is evaluated using the relative characterization error that is calculated using the following formula:

$$Error_i = \frac{|s_{calc_i} - s_{real_i}|}{s_{real_i}} \tag{14}$$

where $Error_i$ is the relative characterization error, and s_{calc_i} and s_{real_i} are the calculated scaling factor of gate i and its real value, respectively. The resulting error over all gates in the circuit is calculated as the average of all the $Error_i$:

$$Error_{avg} = \frac{1}{n}\sum_{i=1}^{n} Error_i \tag{15}$$

where n is the number of gates in the circuit, and $Error_{avg}$ is the average result error for GLC. In the rest of this section, we use $Error_{avg}$ to evaluate the accuracy of GLC.

We evaluate our HTH detection approach on the ISCAS and ITC benchmarks. For each benchmark, we simulate two cases: one where the HTHs do not exist, and one where a single HTH gate (NAND gate) is embedded at random locations on the target

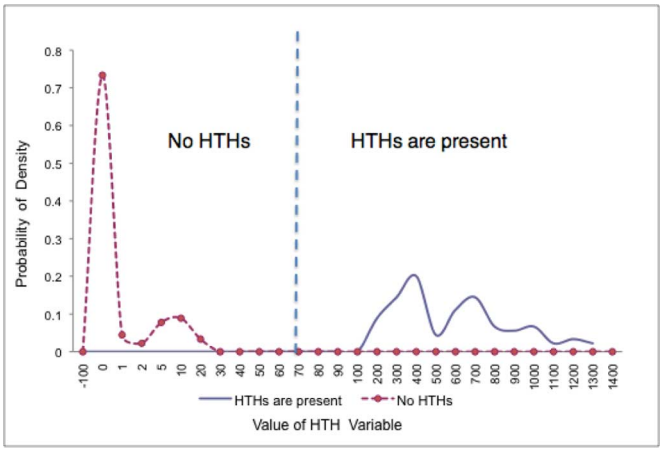


Fig. 6. PDF of the HTH variable in HTH detection, integrated with all the ISCAS and ITC benchmarks in Table I. In the case when no HTHs exist, the HTH variable has a small value from 0 to 11.2. When a single HTH gate is present, the HTH variable ranges from 151 to 1214. There is a large enough gap between the two cases to enable us to draw a decision line at around 70 to distinguish the two cases.

TABLE I
HTH DETECTION AND DIAGNOSIS ON ISCAS AND ITC BENCHMARKS

Design	Gates	GLC Error (%)	With HTHs	No HTHs
C17	6	0.0057	240 ∼ 303	0
C432	160	0.11	582 ∼ 603	0 ∼ 8.9
C499	202	0.26	174 ∼ 254	0 ∼ 6.3
C880	383	0.34	151 ∼ 231	0 ∼ 10.2
C1355	546	0.40	298 ∼ 666	0 ∼ 4.5
C1908	880	0.98	600 ∼ 1022	0 ∼ 2.5
C2670	1193	0.75	567 ∼ 1182	0 ∼ 1.9
C3540	1669	1.72	232 ∼ 881	0 ∼ 0.2
C5315	2307	0.52	223 ∼ 1214	0
C6288	2416	0.13	342 ∼ 912	0 ∼ 0.4
C7552	3512	0.39	492 ∼ 838	0 ∼ 1.3
S526	214	0.33	195 ∼ 315	0 ∼ 11.2
S832	292	0.73	214 ∼ 355	0 ∼ 10.8
S38417	22179	0.29	380 ∼ 619	0 ∼ 1.0
S38584	19253	0.20	381 ∼ 1198	0 ∼ 6.6
b17	27852	0.60	586 ∼ 1210	5.8 ∼ 9.7
b18	94249	0.89	392 ∼ 1025	0 ∼ 4.6
b19	231266	0.91	892 ∼ 1130	0 ∼ 7.8

circuit. We repeat the leakage power measurements for all the benchmarks 50 times. The results are shown in Table I. Also, we plot the probability density function (pdf) of the HTH variable in Fig. 6. We observe a large gap between the two cases in terms of the probability distribution of the HTH variable. This enables us to draw a decision line between the two situations, with which we achieve zero false positives and zero false negatives in HTH detection.

VIII. CONCLUSION

We have developed a gate level characterization approach that employs thermal conditioning. By utilizing the additional degrees of freedom in organizing measurements, we are able to break all the correlations in all tested circuits and to characterize the scaling factors for all the gates in the circuits. We have additionally improved the accuracy of the characterization by applying statistical methods. The GLC procedure is the starting point for creation of a HTH detection approach that also employs constraint manipulation techniques. The simulation results on several ISCAS and ITC benchmarks show that we are able to characterize all the gates with an average error less than the measurement error, and that HTHs can be detected accurately by our approach. By using iterative segmentation of the pertinent circuit through inputs freezing, we have accomplished linear scalability of the GLC and HTH techniques.

REFERENCES

[1] S. Wei, S. Meguerdichian, and M. Potkonjak, "Gate-level characterization: Foundations and hardware security applications," in *Proc. DAC*, 2010, pp. 222–227.
[2] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Proc. DAC*, 2003, pp. 338–342.
[3] 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.-Feb. 2010.
[4] F. Koushanfar, P. Boufounos, and D. Shamsi, "Post-silicon timing characterization by compressed sensing," in *Proc. ICCAD*, 2008, pp. 185–189.
[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 *Proc. DAC*, 2008, pp. 606–609.
[6] M. Potkonjak, A. Nahapetian, M. Nelson, and T. Massey, "Hardware Trojan horse detection using gate-level characterization," in *Proc. DAC*, 2009, pp. 688–693.
[7] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos, "Modeling within-die spatial correlation effects for process-design co-optimization," in *Proc. ISQED*, 2005, pp. 516–521.
[8] M. Brown, C. Bazeghi, M. Guthaus, and J. Renau, "Measuring and modeling variability using low-cost FPGAs," in *Proc. FPGA*, 2009, pp. 286–286.
[9] N. Drego, A. Chandrakasan, and D. Boning, "A test-structure to efficiently study threshold-voltage variation in large MOSFET arrays," in *Proc. ISQED*, 2007, pp. 281–286.
[10] W. Zhang, X. Li, and R. Rutenbar, "Bayesian virtual probe: Minimizing variation characterization cost for nanoscale IC technologies via Bayesian inference," in *Proc. DAC*, 2010, pp. 262–267.
[11] S. Wei and M. Potkonjak, "Scalable segmentation-based malicious circuitry detection and diagnosis," in *Proc. ICCAD*, 2010, pp. 483–486.
[12] S. Wei and M. Potkonjak, "Scalable hardware Trojan diagnosis," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 2011, to be published.
[13] F. Wolff, C. Papachristou, S. Bhunia, and R. Chakraborty, "Towards Trojan-free trusted ICs: Problem analysis and detection scheme," in *Proc. DATE*, 2008, pp. 1362–1365.
[14] M. Banga and M. Hsiao, "A region based approach for the identification of hardware of Trojans," in *Proc. HOST*, 2008, pp. 40–47.
[15] J. Li and J. Lach, "At-speed delay characterization for IC authentication and Trojan horse detection," in *Proc. HOST*, 2008, pp. 8–14.
[16] Y. Jin and Y. Makris, "Hardware Trojan detection using path delay fingerprint," in *Proc. HOST*, 2008, pp. 51–57.
[17] X. Wang, H. Salmani, M. Tehranipoor, and J. Plusquellic, "Hardware Trojan detection and isolation using current integration and localized current analysis," in *Proc. DFT*, 2008, pp. 87–95.
[18] J. Aarestad, D. Acharyya, R. Rad, and J. Plusquellic, "Detecting Trojans through leakage current analysis using multiple supply pad IDDQS," *IEEE Tran. Inf. Forensics Security*, vol. 5, no. 4, pp. 893–904, Dec. 2010.
[19] D. Markovic, C. Wang, L. Alarcon, T. Liu, and J. Rabaey, "Ultralow-power design in near-threshold region," *Proc. IEEE*, vol. 98, no. 2, pp. 237–252, Feb. 2010.
[20] A. Asenov, "Random dopant induced threshold voltage lowering and fluctuations in sub-0.1 MOSFETs: A 3-D atomistic simulation study," *IEEE Trans. Electron Devices*, vol. 45, no. 12, pp. 2505–2513, 1998.
[21] B. Cline, K. Chopra, D. Blaauw, and Y. Cao, "Analysis and modeling of CD variation for statistical static timing," in *Proc. ICCAD*, 2006, pp. 60–66.
[22] R. Rajsuman, "Iddq testing for CMOS VLSI," *Proc. IEEE*, vol. 88, no. 4, pp. 544–566, Apr. 2000.
[23] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. CRYPTO*, 1999, pp. 388–397.

[24] [Online]. Available: http://sine.ni.com/nips/cds/view/p/lang/en/nid/204239
[25] P. Michaud and Y. Sazeides, "ATMI: Analytical model of temperature in microprocessors," in *Proc. MoBS*, 2007, pp. 1–10.
[26] R. Bellman, "On the approximation of curves by line segments using dynamic programming," *Commun. ACM*, vol. 4, no. 6, pp. 284–284, 1961.
[27] R. Rao, F. Liu, J. Burns, and R. Brown, "A heuristic to determine low leakage sleep state vectors for CMOS combinational circuits," in *Proc. ICCAD*, 2003, pp. 689–692.
[28] L. Yuan and G. Qu, "A combined gate replacement and input vector control approach for leakage current reduction," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 14, no. 2, pp. 173–182, Feb. 2006.

Sheng Wei is currently pursuing the Ph.D. degree in computer science at the University of California, Los Angeles.

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

Saro Meguerdichian is currently pursuing the Ph.D. degree in computer science at the University of California, Los Angeles.

His research interests include system and physical security, customization, combined statistical modeling and optimization, and low-power medical devices and sensor systems.

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 the Computer Science Department at the University of California, Los Angeles. He created 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.