A Sensitivity Analysis of Power Signal Methods for Detecting Hardware Trojans under Real Process and Environmental Conditions

Reza Rad
reza2@umbc.edu
Univ. of Maryland, B.C.

Jim Plusquellic
jimp@ece.unm.edu
University of New Mexico

Mohammad Tehranipoor
tehrani@engr.uconn.edu
University of Connecticut

ABSTACT

Trust in reference to integrated circuits addresses the concern that the design and/or fabrication of the IC may be purposely altered by an adversary. The insertion of a hardware Trojan involves a deliberate and malicious change to an IC that adds or removes functionality or reduces its reliability. Trojans are designed to disable and/or destroy the IC at some future time or they may serve to leak confidential information covertly to the adversary. Trojans can be cleverly hidden by the adversary to make it extremely difficult for chip validation processes, such as manufacturing test, to accidentally discover them. This paper investigates the sensitivity of a power supply transient signal analysis method for detecting Trojans. In particular, we focus on determining the smallest detectable Trojan, i.e. the least number of gates a Trojan may have and still be detected, using a set of process simulation models that characterize a TSMC 0.18 um process. We also evaluate the sensitivity of our Trojan detection method in the presence of measurement noise and background switching activity.

1.0 Introduction

The globalization of the integrated circuit (IC) industry in combination with the dramatic increases in the complexity of ICs have raised new concerns regarding their trustworthiness [1][2]. The threat is the malicious modification to the function of an IC such as the inclusion of additional circuitry designed to enable an adversary to corrupt data or destroy, disable or remotely control the IC through a back door at a time of his or her choosing. A wide range of parametric-based and functional-based malicious modifications or Trojans are possible in out-sourced ICs [3].

Adversaries will employ a variety of methods to hide Trojans so that they are extremely difficult to detect through traditional manufacturing tests. For example, the inputs of an inserted Trojan will be selected so that its activation is statistically very unlikely. Hence, Trojans will be activated only under rare internal states, making conventional logic-based testing techniques ineffective for Trojan detection.

Trojan detection methods based on physical inspection and destructive reverse engineering are difficult and costly and, due to their destructive nature, cannot be applied to all chips. Moreover, such approaches applied to a subset of chips cannot guarantee that all chips are Trojan free because the adversary may insert Trojans into only a subset of the chips. If these Trojan-inserted chips are not selected for physical inspection, then the Trojan will be missed. These concerns drive the need for a new, non-destructive approach for Trojan detection that can be applied to all chips.

Parametric testing techniques such as those based on the analysis of power supply signals are better suited for Trojan detection because they can potentially detect a Trojan by only *partially activating* it[1]. Partial activation refers to the situation in which the applied test patterns cause switching activity on the inputs of a Trojan and/or within a subset of the Trojan's logic gates, but the output(s) of the Trojan do not change, and therefore the Trojan does not change the chip's functionality. Trojan detection through partial activation is possible because the presence of the Trojan unavoidably impacts the parametric behavior of the IC, e.g., by modifying the wire loads and other internal parameters of the IC such as power grid capacitance.

We must assume that the adversary is aware of the parametric anomalies introduced by a Trojan in, e.g., the static (I_{DDQ}) or transient (I_{DDT}) power supply signals, and that he/she will design the Trojan to minimize its visibility. Therefore, conventional approaches that analyze **global** I_{DDQ} and I_{DDT} signals will not have sufficient resolution to detect Trojans. A second major challenge to applying conventional I_{DDX} methods for detecting Trojans is dealing with process variation effects. Process variations are increasing significantly in advanced technology nodes, making it more difficult to differentiate between signal anomalies introduced by process variations and those introduced by Trojans.

Given the random statistical nature of process variations and other environmental noise sources, Trojan detection methods based on the analysis I_{DDX} signals need to be statistically based. Statistical methods require the definition of a threshold to account for noise and process variations effects. The threshold is used in the methods to distinguish between Trojan-free and Trojan-inserted ICs. In manufacturing test methods, data-driven techniques have been proposed as a means of deriving the threshold from hardware data. Unfortunately, this approach

1. Full activation is defined as a test scenario in which the statistically unlikely activation state is achieved, causing the Trojan to enter into its destructive or corruptive mode.

will not work for Trojan detection because, unlike random defects, the chip data used to define the threshold for Trojan detection is likely to be misleading as many or all of these chips may contain Trojans. Therefore, the statistical thresholds must be derived from 'golden' simulation models instead, and these models must be chosen such that they accurately characterize the Trojan-free chips across the inherent skew in the manufacturing process.

In [4] and [5], we describe a hardware Trojan detection method that addresses these issues. The method analyzes supply currents measured from multiple supply ports to deal with the small Trojan-signal-to-background-current ratios. Simple calibration circuits and procedures are used to reduce the adverse impact of process variation effects on Trojan detection resolution. A calibration technique is proposed that transforms the measured currents for each IC to match those produced from a golden, Trojan-free simulation model. This transformation process greatly amplifies Trojan signal anomalies.

In this paper, we build on the preliminary work described in [4] and [5] . In [4] , we developed the statistical analysis technique for detecting Trojans and applied it to a circuits containing 'large' Trojans. In [5] , we explored four different calibration methods to deal with the adverse effects of process and environmental variations on our statistical analysis procedure. In this paper, the focus is on determining the level of sensitivity of our power signal analysis technique to Trojans in the presence of realistic, process and environmental variations and under different measurement noise and background switching scenarios. To meet this goal, a novel approach is used to introduce Trojans that enables a systematic process for evaluating the true sensitivity of our technique. In particular, the following parameters are investigated in our experiments:

Trojan activity: When a test pattern is applied and power port signals measured, the level of Trojan activity plays a significant role on the ability to detect it. If some of the Trojan gates make transitions due to the applied test pattern, then the measured transients will be significantly affected and detection through statistical analysis becomes easier. When the applied test pattern does not cause any of the Trojan gates to switch, then the only observable effect of the Trojan anomaly will be due to the capacitive/resistive loading effects introduced by the Trojan. In our analyses, we consider both situations and show that our analysis and calibration method can detect Trojans with only one switching gate or with only a couple non-switching gates.

Measurement noise: One of the factors that can significantly limit the sensitivity of analysis methods based on transient signals is measurement noise. Therefore, we explore the effects of noise on our Trojan detection sensitivity.

Un-desired switching activity: Un-desired switching activity of other components in the circuit will also play a significant limiting role in sensitivity. If the applied test patterns generate switching not just in the neighborhood of the Trojan but also in other parts of the chip, the measured power transients will be affected, reducing the sensitivity of the method.

The sensitivity analysis is carried out using simulations of a ISCAS '85 benchmark circuit [6] under a variety of adverse conditions, including those produced by process variations, environment noise and various levels of switching activity within and around the inserted gates that represent the Trojan. The simulation results demonstrate that our detection method can tolerate significant levels of noise and switching activity for small Trojans that are 'partially activated', defined as the situation in which a subset of the Trojan's logic gates switch under the test sequence, but it is less effective under conditions for cases in which only the inputs to the Trojan gates switch, i.e., the Trojan gates themselves do not switch.

The rest of the paper is organized as follows. A brief review of the published literature on the topic of Trojan detection is provided in Section 2.0. Section 3.0 reviews the Trojan detection method described in [4] and [5] and the calibration methods employed to reduce the process variation effects in our statistical analysis. Section 4.0 discusses the experiment setup for sensitivity analysis. Results of the sensitivity analysis are reported in Section 5.0. Conclusions are given in Section 6.0.

2.0 Background

The emergence of a globalized, horizontal semiconductor business model raises a set of concerns involving the security and trust of the information systems on which modern society is increasingly reliant for critical functionality. Hardware security and trust issues span a broad range including threats related to the malicious insertion of Trojan circuits designed, e.g., to act as a 'kill switch' to disable a chip, to integrated circuit (IC) piracy, to attacks designed to extract encryption keys and IP from a chip, and to malicious system disruption and diversion. Of these threats, the malicious insertion of hardware Trojans in ICs is a relatively new trust concern that must now be addressed in combination with other hardware security risks.

The following briefly summaries the approaches proposed by others in response to the need for Trojan detection methods. An analysis of the deficiencies of each of the proposed approaches makes it difficult to declare any one of these approaches as a solution to the problem. Although our strategy provides several unique advantages over other power signal analysis methods, it is not a complete solution for this problem, e.g., our method does not address the test stimulus issue. Therefore, the best solution is likely a combination of our signal analysis approach with features from other proposed methods as described below.

The authors of [7] were the first to address the hardware Trojan issue. They propose the use of side-channel signals, e.g., transient power supply currents, to identify Trojans in chips. Their method defines a "side-channel fingerprint" for each IC that is based on the analysis of a single global signal such as power, EM or current transients. Methods based on global signal measurements will not scale well to larger ICs. Also, in our own personal experience, it is necessary to collect signals such as power transients very close to the chip, e.g., by measuring these signals from the individual power ports during wafer probe, in order to obtain sufficient frequency resolution. Global signal measurements must be taken at a point in the power distribution network further removed from the chip, i.e.,

at a common connection point such as the power plane in the probe card. The mid- to high-frequency content of the transients are filtered out at these measurement points, which reduces the resolution of global signal analysis techniques.

The authors of [8] focus on the challenging issue of generating test patterns for Trojan detections and propose a method that first determines a set of target 'hard-to-observe' sites for a Trojan with q inputs and then uses ATPG (Automatic Test Pattern Generation) to generate patterns to activate the Trojan. Although this may be an effective strategy for Trojans with a small number of inputs, analysis complexity and test set size may make this type of approach impractical for larger Trojans.

A delay characterization method for IC authentication and Trojan detection is proposed in [9]. The authors propose an at-speed path delay measurement method for finding differences between path delays of Trojan-free circuits and those with Trojans. Path delay testing is a parametric strategy that may be very effective for detecting Trojans. The technique as proposed, however, requires precise characterization of silicon path delays at design time, which is becoming increasingly difficult because of mismatches between models and hardware in state-of-the-art technologies.

A circuit partitioning based method for detecting Trojans is described in [10]. The method is based on selecting a set of signals in specific regions of the circuit and generating input vectors that maximize the relative power consumption of the logic in that region. If a Trojan is present in a targeted region, then this strategy will increase the chances of detecting it. Although the method restricts logic switching to small regions, it analyzes a global signal for Trojan detection. Therefore, the method will be less sensitive in larger chips, particularly as leakage power increases as a fraction of total power in newer technologies.

A design modification strategy for improving Trojan detectability is proposed in [11] where the goal is to improve controllability and observability of hard-to-control or hard-to-observe nodes within the IC as a means of triggering the full activation of a Trojan. This strategy will be effective at improving the likelihood of activation, but only if the design modifications can be kept secret from the adversary. The controllability/observability analysis performed by the adversary after reverse engineering the layout will reveal the circuit modifications. This adversary is then free to connect the Trojan such that this type of activation strategy will be less effective.

A Trojan detection method based on a path delay fingerprint is proposed in [12] where the authors analyze path delays as the side channel signal. Principle component analysis is employed to analyze multiple path delays simultaneously to detect Trojan anomalies. For large chips, a large number of vectors may be needed to achieve adequate Trojan coverage, and therefore, it may be difficult to apply this type of strategy in practice.

In [13] authors exploed eight different RTL level attacks on FPGA implementation of an Alpha encryption module and demonstrated that digital systems can be vulnarable to such attacks. In [14] authors report their analysis on the performance of their path-delay based Trojan detection technique under process variations. Their results suggest that their delay characterization method can be effective in detection of Trojans in presense of variations in process parameters.

In [15] authors proposed a voltage inversion method for increasing the frequency of activations of Trojan gates and employ a method called sustained vector simulation to reduce the switching activity of the rest of the circuit. Reported results indicate that their method is effective in detecting small Trojans in benchmark circuits.

3.0 Trojan Detection using Power Supply Transient Signals (I_{DDT})

Our power supply transient analysis (I_{DDT}) technique analyzes *local* I_{DDT} measurements obtained from multiple individual *power ports* on the chip. The I_{DDT} signals are measured from each of the power ports as a test sequence is applied to the inputs of the core logic. The I_{DDT}s of neighboring power ports, e.g. PP_0 and PP_1 in Figure 1, are compared to identify anomalies introduced by the presence of a Trojan circuit. Unfortunately, the measured I_{DDT}s cannot be used directly in the detection method because of process and environmental noise effects. Signal calibration must first be applied to reduce these noise sources.

3.1 Signal Calibration

Signal calibration is used to deal with process and environmental (PE) variation effects that occur in the chip's core logic, power grid and off-chip connections to the power ports. Calibration is carried out on each IC using a set of inserted calibration circuits. Each calibration circuit is a p-channel transistor whose gate is connected to the output of a scan flip-flop (FF). The source of the p-channel is connected to the power grid in metal 1 directly underneath the power port, e.g., PP_0 in Figure 1. One copy of this calibration circuit is inserted under each of the power ports on the chip.

A calibration test is carried out by configuring the calibration scan chain to deliver a step input to the gate of the calibration p-channel transistor. This can be implemented by scanning a 0 through the scan chain with all other elements initialized to 1. The step input enables the p-channel transistor and effectively creates a short between V_{DD} and GND. The step input response is then measured at each of the power ports. Example waveforms are shown for PP_0 and PP_8 in Figure 1.

In [5] , we investigated the effectiveness of four different DC and AC signal calibration techniques and found that an *AC sample* based calibration method is the most effective both from a cost and quality perspective. The AC sample method collects a single sample from the calibration response waveforms at a point in time immediately following the introduction of the step input. This sample is sufficient to capture the impedance characteristics of the power port. The data collected from each of the calibration tests is used to construct a linear transformation matrix, as given by *CM* in Equation 1, where the rows correspond to the calibration tests and the columns correspond to the power ports. Here, PPD_{xy} indicates the power port

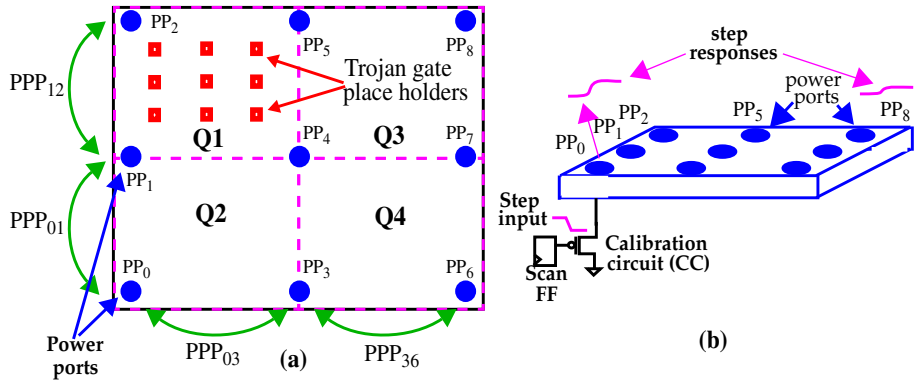

Figure 1. (a) Architecture of the simulation model, each of the quads (Q1-Q4) contains a copy of C499 benchmark, (b) calibration circuit (CC) and step

$$CM = \begin{bmatrix} \frac{PPD_{00}}{GD_0} & \frac{PPD_{01}}{GD_0} & \cdots & \frac{PPD_{08}}{GD_0} \\ \frac{PPD_{10}}{GD_1} & \frac{PPD_{11}}{GD_1} & \cdots & \frac{PPD_{18}}{GD_1} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{PPD_{80}}{GD_8} & \frac{PPD_{81}}{GD_8} & \cdots & \frac{PPD_{88}}{GD_8} \end{bmatrix} \begin{matrix} \text{cal. test 0} \\ \text{cal. test 1} \\ \cdots \\ \text{cal. test 8} \end{matrix} \quad \text{Eq.1. [5]}$$

data value (the sample) for calibration test x measured at power port y. The normalization factor given as the denominator, GD_x, is the sum of individual values along each row x. The matrix is used to transform the I_{DDT}s measured under Trojan tests to a 'golden' PE-variation-free model of the IC. This is accomplished by first computing a transformation matrix, X, from CM by taking its inverse, as given by Equation 2, where the elements of CM, i.e., PPD_{00}/GD_0, are represented as a_{00} in CM^{-1}. Once computed, X is subsequently used to calibrate the

$$X = CM^{-1}$$

$$\begin{bmatrix} x_{00} & x_{01} & \cdots & x_{08} \\ x_{10} & x_{11} & \cdots & x_{18} \\ \cdots & \cdots & \cdots & \cdots \\ x_{80} & x_{81} & \cdots & x_{88} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & \cdots & a_{08} \\ a_{10} & a_{11} & \cdots & a_{18} \\ \cdots & \cdots & \cdots & \cdots \\ a_{81} & a_{82} & \cdots & a_{88} \end{bmatrix}^{-1} \quad \text{Eq2. [5]}$$

path data measured under the Trojan tests, as given by Equation 3. The vector given by t_0 through t_8 corresponds to the nine data values, e.g., I_{DDT} waveform areas, collected under a Trojan test. The calibrated data is given by the column vector on the left, i.e., c_0 through c_8. The calibrated path data given by C_n in Equation 3 can be used directly in the prediction ellipse method described below. Reference [5] describes the details of this process.

3.2 Scatterplot Analysis for Trojan Detection

We employ a statistical analysis technique to detect transient signal anomalies introduced by Trojans. We apply the test sequences to each of the simulation models and measure the

$$C_n = T_n * X$$

$$\begin{bmatrix} c_0 \\ c_1 \\ \cdots \\ c_8 \end{bmatrix} = \begin{bmatrix} t_0 & t_1 & \cdots & t_8 \end{bmatrix} \times \begin{bmatrix} x_{00} & x_{01} & \cdots & x_{08} \\ x_{10} & x_{11} & \cdots & x_{18} \\ \cdots & \cdots & \cdots & \cdots \\ x_{80} & x_{81} & \cdots & x_{88} \end{bmatrix} \quad \text{Eq.3. [5]}$$

I_{DDT} areas produced on each of the nine power ports. The areas from the Trojan-free simulation models are first calibrated to reduce the adverse impact of process variations. The calibrated areas from pairs of neighboring power ports are then plotted in two-dimensional scatterplots. The mean and variation of the data points in each scatterplot are used to derive the statistical limits implemented as an enclosing ellipse. The region enclosed by the ellipse defines the space in which the data points from Trojan-free ICs are expected to fall. Data points that fall outside the limits are deemed to belong to an IC with a Trojan.

As an example, Figure 2 shows the scatterplot of calibrated I_{DDT} areas for supply port pairing PP_1 (x-axis) and PP_2 (y-axis). The black circles represent the data points from the Trojan-free simulation models. A three sigma *prediction ellipse* is derived from these points and defines the Trojan-free space. The elliptical bound is computed from the eigen values of the Trojan-free covariance matrix and a three σ X^2 (chi-square) distribution statistic. The red data points are obtained from one of the Trojan-inserted models in ten different process models. In this case, all of the Trojan-inserted circuit data points are detected as outliers. Here, an outlier is defined as a data point that falls outside of the three σ ellipse.

The process of determining whether a given Trojan under a specific process model is detected is based on outlier analysis of scatterplot data. For each Trojan model, twelve scatterplots are analyzed, one for each adjacent power port pairing (PPPs). Several of these PPPs are labeled in Figure 1 as PPP_{01}, PPP_{12}, PPP_{03}, PPP_{36}. If any one of the twelve Trojan data point falls outside the prediction ellipse limits across the twelve scatterplots, the Trojan is identified as detected. The detection algo-

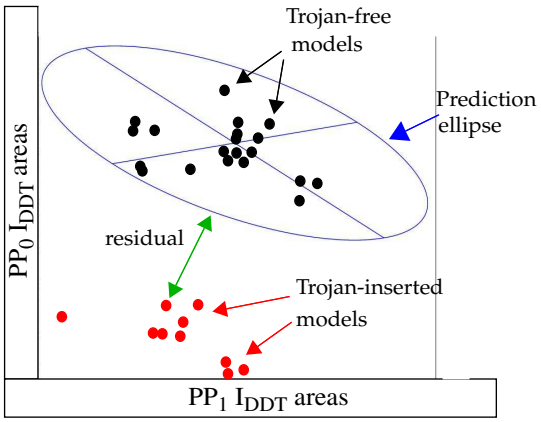

Figure 2. Example scatterplot for power port pairing PP0 and PP1 (referred to as PPP01)[5].

rithm evaluates the eleven remaining scatterplots (not shown) to determine the number of detections for each Trojan model.

The number of outliers identified for a specific Trojan test case can be considered as a measure of confidence in the detection decision, with higher numbers of outliers corresponding to a higher confidence. Another measure of confidence can be obtained from the maximum or average distance (across all scatterplots) of the Trojan data points from the surface of the 3σ ellipse. Here, again, larger distances correspond to a higher degree of confidence that the chip has a Trojan. This distance is called a *residual* and it is typically reported as a standardized quantity by dividing it by 3σ.

3.3 Statistical Data Characterization Issues

A 3σ limit is an industry standard for defining the bounds of statistical data, and for identifying outliers. The need to use other values to define the bounds is usually an indication that the data is drawn from a statistical distribution that is not normal, and other types of non-parametric statistical approaches may be required. In our experiments, we found that using a 3σ limit served well to bound the Trojan-free data points generated under the various process and noise models, and conclude that this type of noise is well-characterized as a normal distribution.

3.4 Robustness of the Technique to Sabotage by an Adversary

A major issue concerning Trojan detection techniques, particularly methods that introduce support circuitry, is related to their robustness to sabotage by an adversary. The adversary has the advantage of being able to modify the layout before fabrication and therefore can recognize, disable or subvert support circuity. Here, we consider the robustness of our multiply power port technique and the supporting calibration circuits (CCs) to sabotage.

The low resistance nature of the power grid makes it extremely sensitive to any type of design change. For example, modifying the connectivity of the metal interconnect in the power grid will produce major changes in the transient response observed at the power ports. Such attempts will be quickly recognized by comparing the simulation-generated current profiles produced from the calibration tests with those measured from the chips. For example, if the adversary attempts to add resistance between the power grid and a specific power port, in an attempt to 'distribute' the anomaly created by a Trojan across multiple ports, thereby reducing its observability, the calibration tests results will immediately reveal the high resistance connection, particularly if it occurs on every chip.

If the adversary attempts to disable the calibration circuits themselves or move them to other positions in the layout, this too will be immediately reflected as anomalies in the values in the calibration matrix. For example, disabling a calibration circuit will produce a row of zeros in the matrix, while moving the calibration circuits will produce anomalous shapes in the current distribution profile for a row. By positioning the CCs under the power ports, the largest current will almost always be produced in the power port directly above, with an approximate proportional decrease in the level of current in other power ports as a function of their distance from the enabled CC. Variation in probe card contact resistance will distort this relationship somewhat. In fact, the primary function of the CCs is to fix this type of distortion. However, contact resistance variations are expected to be random, so if a pattern of distortion is observed in the calibration matrices across multiple chips, then the probability that the CCs have been moved by an adversary increases significantly. In general, the two dimensional profiling carried out by the calibration procedure is very robust to tamper.

4.0 Experiment Setup

Figure 1 shows a top level view of the design used in the simulation experiments. It consists of four 'quads' labeled Q_1 through Q_4. A copy of the c499 ISCAS '85 benchmark circuit is inserted into each quad [6]. The layout of the C499 in Quad Q_1 is modified to include nine empty rectangles as shown in the figure. These rectangles are 'holes' in the layout and correspond to the size of a typical standard cell. Trojans are modeled by inserting a two input NAND gate into one or more of these rectangles, as explained below.

The design in Figure 1 was constructed using the technology rules for the TSMC 0.18 um process [16]. The power grid

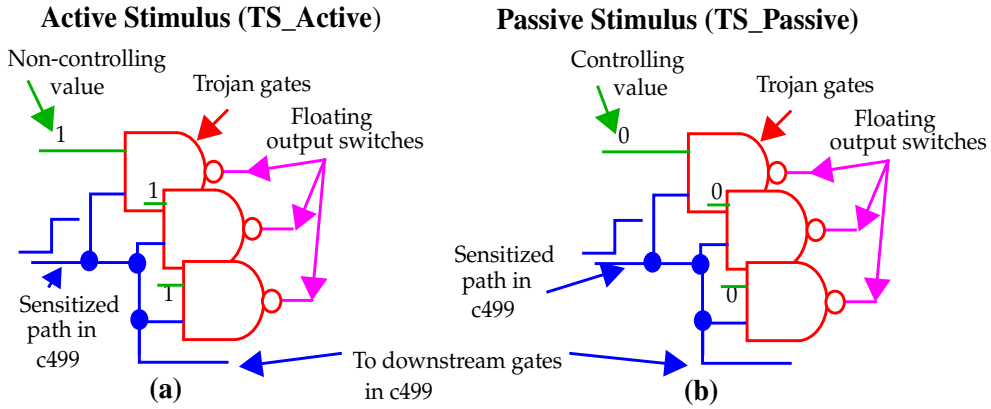

Figure 3. Trojan gate sensitization criteria under the two test sequences [4].

is routed in a standard mesh configuration over all six metal layers available in the process. Nine power ports, labeled PP_0 through PP_8, connect to the power grid in the top most metal layer. The ground grid is interleaved with the power grid (not shown) and is configured in a similar fashion.

4.1 Simulation Process Models

Ten different layouts of the design were constructed. In the Trojan-free version, all nine of the rectangles in Q1 are empty. In the first Trojan layout, a standard cell gate is inserted into one of the rectangles and is connected to nodes in the surrounding neighborhood. This process is repeated for each of the remaining (eight) Trojan layouts, with one additional standard cell gate added to an empty rectangle in each successive layout. The Trojan models are referred to as T_1 (one gate version) through T_9 (nine gate version). The position of the rectangles are kept constant in all ten layouts to minimize the differences.

A set of twenty simulation models are extracted from the Trojan-free layout using published process parameters for the TSMC 0.18 um process. Fifteen of the simulation models are used to define the statistical limits and derive the 3σ ellipses for the Trojan-free case as described in Section 3.2. The remaining five are used as control samples to evaluate the false alarm rate of our Trojan detection method. False alarms occur when a Trojan is detected in one of the Trojan-free simulation models that are designated as control. For each of the nine Trojan designs, ten simulation models are extracted using the same process models as those used to create the first ten Trojan-free, non-control models.

4.2 Test Sequences

Two sets of test sequences are used as the stimulus to evaluate Trojan sensitivity. Both sets of sequences propagate signals along the same paths in the Q_1 copy of the c499 shown in Figure 1. The difference in the stimuli is related to the off-path inputs of the inserted Trojan gates.

Figure 3 shows the two scenarios, labeled (a) and (b), using an NAND gate to represent the Trojan. One of the inputs of the NAND is connected to a sensitized path in the c499. The other input is held constant at one of two values. In (a), a non-controlling value is placed on the off-path input. Therefore, when the test sequence is applied to the PIs of the c499, the propagating signal causes the NAND gate's output to switch (and consume power). We refer to this test sequence as **TS_Active**. In (b), a controlling value is placed on the off-path input preventing the NAND gate from switching. Therefore, only the capacitive loading of the on-path input can affect the power consumption. We refer to this test sequence as **TS_Passive**.

Note that the Trojan's output is not connected in our experiments. For an actual Trojan instance, this would not be the case. Therefore, our Trojan models minimize the impact of the Trojan on power consumption, better suiting the objective of our sensitivity analysis. The configurations shown in Figure 2 are replicated for Trojan models that incorporate more than one gate, i.e., T_2 through T_9.

4.3 Measurement and Background Switching Noise

Our main goal is to determine the sensitivity of our Trojan detection method to Trojan size. However, in order to better model the conditions that exist in an actual environment, we include two additional parameters in the simulations; noise and background switching activity.

In our experiments, we introduce additive, white Gaussian noise (AWGN) at three levels including 10 dB, 20 dB and 30 dB signal-to-noise-ratios (SNRs), onto the waveforms generated from the power ports under both the calibration and Trojan applied test sequences. Although there are several ways of reducing the level of noise in hardware measurements, e.g., averaging the measurements over repeated cycles and filtering the transients to remove the out-of-band noise, no technique is ideal and therefore, a portion of the noise remains. We expect that after applying such techniques, a SNR level of 30 dB can be achieved in actual hardware measurements. The analysis using 20 dB and 10 dB SNR represent extreme cases and are included for completeness.

A second important source of noise that is difficult to control is that produced by the switching activity of other components in the circuit. The application of a test pattern sequence will generate a series of transitions along paths in the circuit. The paths that propagate transitions are called 'sensitized paths'. Automatic test pattern generation (ATPG) can be used to control the number of paths that are sensitized but complete control, e.g., producing test patterns that sensitize only a single

path at a time, is extremely difficult or impossible. Therefore, it is inevitable that the applied test patterns will sensitize paths that are not connected to the Trojan inputs, and consequently, these paths will contribute to the background switching noise. High levels of background switching noise will 'wash out' the Trojan anomaly in the measured power port transients, making it more difficult to detect it. In our experiments, we introduce background switching noise by applying three variants of the two test sequences described above. Each of the variants generates switching activity along other paths in the chip in addition to the targeted path.

4.4 Expected Impact of Using a Larger Circuit Model

The simulation model used in this work in small in comparison to commercial designs, and therefore, this raises concerns about the applicability of this technique to larger designs. It should be noted, however, that the power port data captures transient activity primarily from regions (quads) that they are topologically close to. In other words, the individual power ports create a virtual partitioning of the power grid such that the measured transients are primarily those that are generated locally. Therefore, we believe that simulating a larger circuit with, for example, hundreds of quads would lead us to draw the same general conclusions as reported in this paper.

5.0 Simulation Results

As indicated in Section 4.0, we use fifteen of the twenty Trojan-free models to define the three σ prediction ellipses for each of the twelve scatterplots. The remaining five Trojan-free models are used as control samples. Each of the nine Trojan layouts are extracted under ten different process models (for a total of ninety models) to represent our Trojan-inserted test chips. The data used in the scatterplot analysis is first calibrated to remove process and environmental variations.

The detection results are reported in two ways. The first method reports the *number of detections* (or outliers) produced under each Trojan model, i.e., the number of times the data point for a Trojan model falls outside the ellipse. Given that there are twelve scatter plots analyzed per Trojan model, the maximum number of detections is bound by twelve. The second method reports the *maximum residual* produced across the twelve scatterplots[1]. As defined in Section 3.2, a residual is the distance of the Trojan data point from the surface of the 3σ ellipse. The maximum residual among the twelve scatter plots for a Trojan reflects the level of the signal anomaly introduced by the Trojan. Therefore, higher values for either the number of detections or the maximum residual metrics reflect a higher 'degree of confidence' in the detection decision.

Figures 4 and 5 show the number of detections results for the ninety Trojan experiments and the five Trojan-free control chips under test sequences TS_Active and TS_Passive, respectively. Each figure shows the results produced under a noise-free analysis in addition to the results produced using the three noise models. The x-axis lists the ten process models under which the Trojan-inserted circuits are extracted and the five processes used to create the Trojan-free control chip models. The labels *PR1* through *PR10* identify the processes associated with each of the Trojan-inserted models while the labels *CT1* to *CT5* identify the Trojan-free control models. Each cluster contains nine bars, one for each of the nine Trojans, labeled *TR1* to *TR9* along the y-axis. The height of the bar indicates the number of detections. Larger bars indicate more detections and a corresponding higher confidence in the detection decision.

As discussed in Section 4.0, test sequence TS_Active assigns non-controlling values to the off-path inputs of the Trojan gate(s). This enables the Trojan gate(s) to switch as the on-path input (the input connected to the sensitized path in the c499) toggles. Therefore, the number of detections under TS_Active is expected to be larger than the number under TS_Passive. This trend is clearly visible by comparing the height of the bars across plots in Figures 4 and 5.

A second important trend that is clear in the results is that the number of detections increases as the number of Trojan gates increases. For example, the histogram in upper left corner of Figure 4 labeled "*Noise Free*" shows the number of detections for TR1 (one-gate Trojan) is three in many cases, while number of detections for TR2 trends toward four. This pattern continues across TR3 to TR9 and is observable in Figure 5 as well.

A third important trend observable in Figures 4 and 5 is the sensitivity in presence of noise. The four charts in each of these figures show the number of detections under "Noise Free", "30 dB SNR", "20 dB SNR" and "10 dB SNR" conditions, with SNR indicating signal-to-noise ratio. Based on these results, it is clear that increasing the level of noise reduces the number of detections, and the corresponding sensitivity of the method as we would expect. However, in the results for the TS_Active (Figure 4), Trojans of four gates or more are easily detected even under sever noise conditions (10 dB). On the other hand, in Figure 5 where the stimulus is passive, detection sensitivity is significantly reduced with noise. Here, only larger Trojans (TR7,TR8,TR9) are detected and these detections only occur in two of the process models under 10 dB SNR conditions. Trojans of four or more gates can be detected in some of the process models for the 20 dB SNR, but overall, the number of detections is significantly lower compared to TS_Active case.

Figure 6 shows the average number of detections across ten processes for the original active stimulus (TS1) and for three variants (TS2, TS3, TS4), each of which add random background switching activity. The simulations are performed under each of the noise models. Therefore, each data point in the chart represents the average number of detections across ten process corners for a specific noise level (Noise Free, 30 dB, 20 dB and 10 dB SNR), a specific Trojan (TR1 to TR9) and a specific stimulus (TS1, TS2, TS3 or TS4). The presence of background switching activity increases the smallest detectable Trojan to four gates (TR4) for TS3 and to six gates (TR6) for TS2 and TS4. Overall, the number of detections is smaller in the presence of background switching activity (TS2, TS3

1. The residuals are actually *standardized* residuals, as described in Section 3.2.

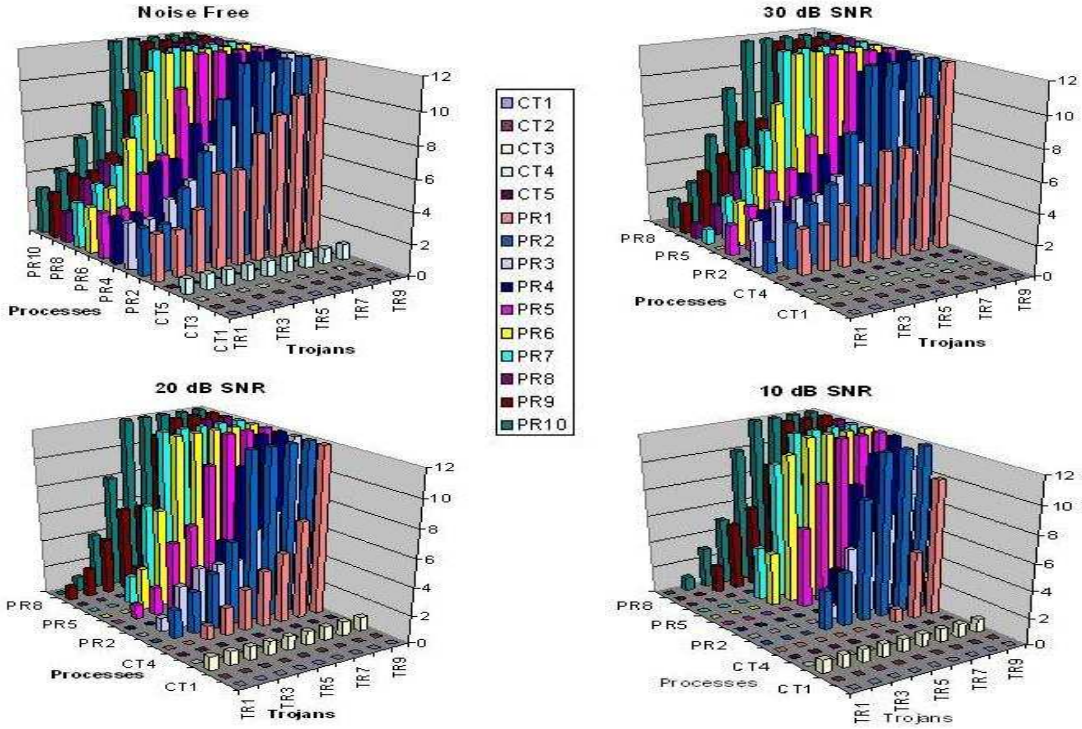

Figure 4. Number of Detections under TS_Active **stimulus for noise-free and three different noise levels.**

and TS4) compared to TS1 where no background switching exists.

Figure 7 shows the results using passive stimulus. The results from the active stimulus with no background switching

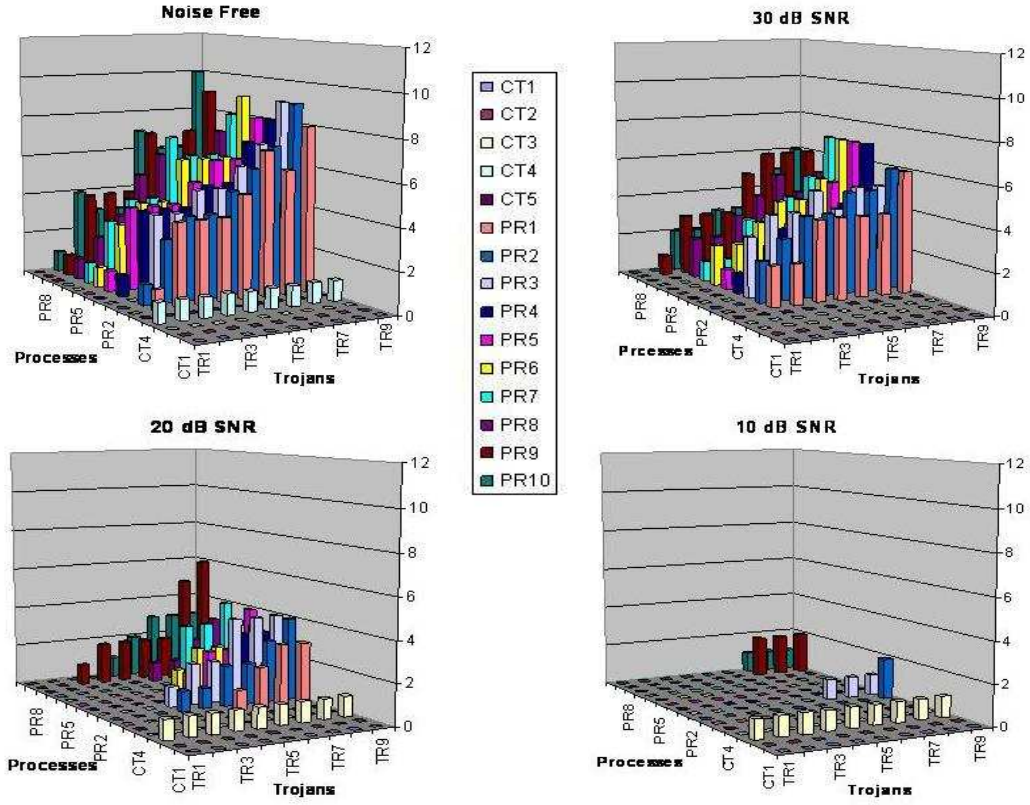

Figure 5. Number of Detections under TS_Passive **stimulus for noise-free and three different noise levels.**

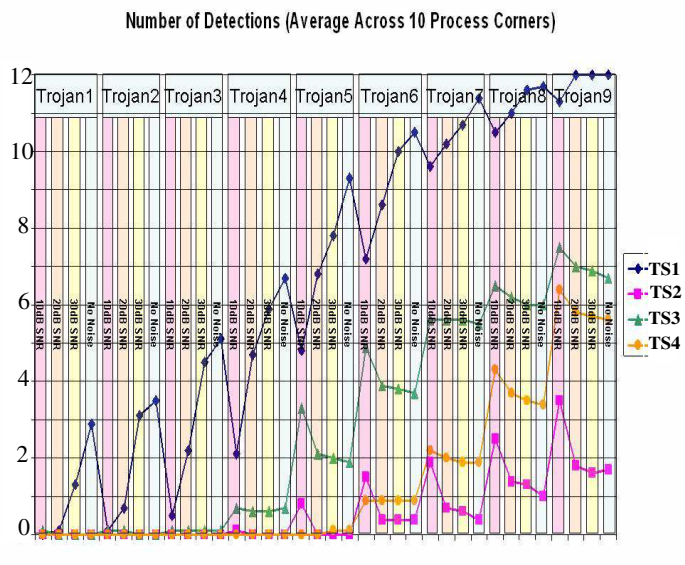

Figure 6. Number of Detections for TS1, the original Active stimulus, and for the three variants TS2, TS3 and TS4 that add random background switching activity to the circuit.

(TS1_Active) are included as a reference. Similar to the previous results, it can be observed that the average number of detections using the passive stimulus is smaller than that using the active stimulus. In this case, the stimuli that generate background switching activity (TS2, TS3 and TS4) are unable to detect any of the Trojans.

Figures 8 and 9 show the maximum residual results for Trojans (TR1 to TR9) under the noise and stimuli models for active and passive cases, respectively. The information obtained from these charts is similar to what obtained from Figures 6 and 7. However, in the new figures, a positive maximum residual indicates that the Trojan is detected and the

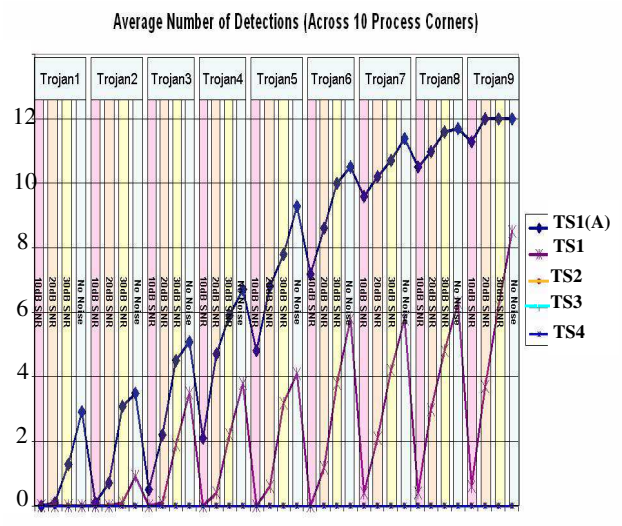

Figure 7. Number of Detections for TS1_Active, TS1_Passive (the original Passive stimulus), and the variants TS2, TS3 and TS4 that add random undesired switching activity to the circuit.

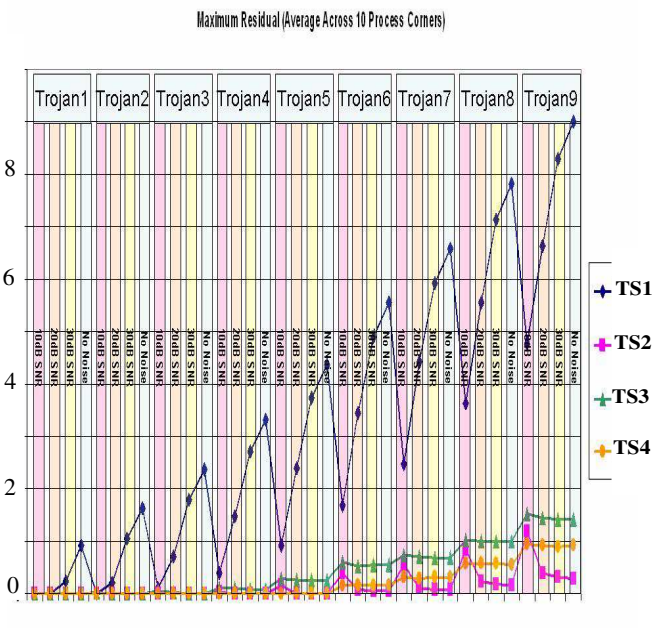

Figure 8. Maximum Residual Results for Active Stimuli

higher the value of the residual the higher the confidence. Similar to Figures 6 and 7, it can be concluded that stimuli that generate switching activity in some of the Trojan gates will have a high chance of revealing the Trojan even if these stimuli generate switching activity in some other parts of the circuit. On the other hand, if the stimulus does not generate switching in the Trojan gates, the method can only detect the Trojan if it causes switching in close proximity to the Trojan and not in other sections of the circuit.

6.0 Conclusions

The objective of this research is to determine the sensitivity

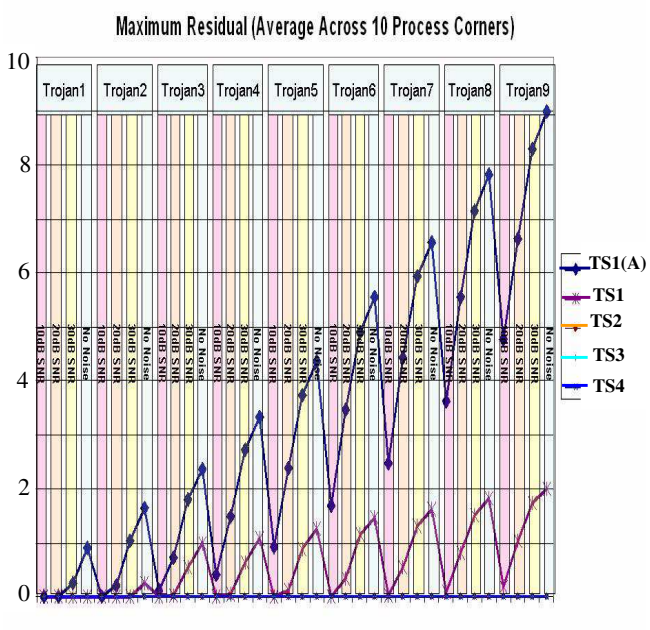

Figure 9. Maximum Residuals for Passive stimuli

of our power supply transient signal method to Trojans under a variety of adverse conditions. The results indicate that under noise free conditions, we can potentially detect Trojans as small as a single gate if that gate switches in response to our test sequence. This number increases to two gates for Trojans that do not switch but are connected to a sensitized path (see Figures 4 and 5). When noise and background switching activity are considered, sensitivity varies from one gate for 30 dB SNR to four gates for 10 dB SNR when the stimulus generates switching in Trojan gates and from three gates to seven gates when the stimulus is not generating switching in Trojan gates. In cases where the applied stimulus generates switching in the Trojan gates and also in other random parts of the chip, sensitivity depends on the type and amount of background switching generated. However, our results demonstrate that Trojans with more than five switching gates are detectable under the three random stimuli cases simulated.

7.0 Acknowledgements

The work of Reza Rad and Jim Plusquellic was supported in part by NSF grant CNS-0716559. The work of Mohammad Tehranipoor was supported in part by NSF grant CNS-0716535.

8.0 References

[1] Defense Science Task Force Report, "High Performance Microchip Supply", http://www.acq.osd.mil/dsb/reports/2005-02-HPMS_Report_Final.pdf

[2] DARPA, "Trust In Integrated Circuits", http://www.darpa.mil/mto/solicitations/baa07-24/index.html

[3] X. Wang, M. Tehranipoor and J. Plusquellic, "Detecting Malicious Inclusions in Secure Hardware: Challenges and Solutions," in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), pp. 15-22, June 2008.

[4] R. Rad, J. Plusquellic, M. Tehranipoor, "Sensitivity Analysis to Hardware Trojans using Power Supply Transient Signals", International Workshop on Hardware-Oriented Security and Trust, 2008, pp. 3-7.

[5] Reza M. Rad, Xiaoxiao Wang, Mohammad Tehranipoor, Jim Plusquellic, "Power Supply Signal Calibration Techniques for Improving Detection Resolution to Hardware Trojans", International Conference on Computer-Aided Design (ICCAD), 2008, pp. 632-639.

[6] ISCAS'85 Benchmarks Circuits: http://www.eecs.umich.edu/~jhayes/iscas.restore/benchmark.html

[7] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, B. Sunar, "Trojan Detection using IC Fingerprinting", Symposium on Security and Privacy, 2007, pp. 296 - 310.

[8] F. Wolff, C. Papachristou, S. Bhunia, and R. Chakraborty, "Towards Trojan-Free Trusted ICs: Problem Analysis and Detection Scheme", Design, Automation and Test in Europe, 2008, pp. 1362-1365.

[9] J. Lie, J. Lach "At-Speed Delay Characterization for IC Authentication and Trojan Horse Detection," in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), pp. 8-14, June 2008.

[10] M. Banga, M. Hsiao, "A Region Based Approach for the Identification of Hardware Trojans," in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), pp. 43-50, June 2008.

[11] R.S. Chakraborty, S. Paul and S. Bhunia "On-demand Transparency for Improving Hardware Trojan Detectability," in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), pp. 51-53, June 2008.

[12] Y. Jin, Y. Markis "Hardware Trojan Detection using Path Delay Fingerprint," in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), pp. 8-14, June 2008.

[13] Y. Jin, N. Kupp, and Y. Makris "Experiences in Hardware Trojan Design and Implementation," in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), July 2009.

[14] D. Rai and J. Lach "Performance of Delay-Based Trojan Detection Techniques under Parameter Variations", in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), July 2009.

[15] M. Banga and M. S. Hsiao "VITAMIN: Voltage Inversion Technique to Ascertain Malicious Insertions in ICs", in Proc. of IEEE Int. Workshop on Hardware-Oriented Security and Trust (HOST), July 2009.

[16] TSMC 0.18 um Technology Files:http://www.mosis.com/Technical/Testdata/tsmc-018-prm.html