Design-Based Fingerprinting Using Side-Channel Power Analysis For Protection Against IC Piracy

James Shey*†, Naghmeh Karimi†, Ryan Robucci†, Chintan Patel†
*Electrical and Computer Engineering Department, United States Naval Academy, Annapolis, MD, US.
shey@usna.edu
†Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, Baltimore, MD, US.
{jshey1, nkarimi, robucci, cpatel2}@umbc.edu

***Abstract*—Intellectual property (IP) and integrated circuit (IC) piracy are of increasing concern to IP/IC providers because of the globalization of IC design flow and supply chains. Such globalization is driven by the cost associated with the design, fabrication, and testing of integrated circuits and allows avenues for piracy. To protect the designs against IC piracy, we propose a fingerprinting scheme based on side-channel power analysis and machine learning methods. The proposed method distinguishes the ICs which realize a modified netlist, yet same functionality. Our method doesn't imply any hardware overhead. We specifically focus on the ability to detect minimal design variations, as quantified by the number of logic gates changed. Accuracy of the proposed scheme is greater than 96 percent, and typically 99 percent in detecting one or more gate-level netlist changes. Additionally, the effect of temperature has been investigated as part of this work. Results depict 95.4 percent accuracy in detecting the exact number of gate changes when data and classifier use the same temperature, while training with different temperatures results in 33.6 percent accuracy. This shows the effectiveness of building temperature-dependent classifiers from simulations at known operating temperatures.**

***Index Terms*—IP Piracy, Fingerprinting, Side-Channel Power Analysis, Machine Learning**

I. Introduction

Increasing the complexity of integrated circuits has raised design time and costs, and in turn has led to the globalization of the design and manufacturing process. Such globalization has increased IC counterfeiting and piracy rates [1]. In recent years, IC Piracy has become a major concern for government, military, and private sectors with increased cost and downtime of systems [2].

Applying traditional piracy avoidance methods such as IC fingerprinting and active metering schemes require additional hardware and/or impose delay and cost overhead [2–4]. In practice, modern low cost/power embedded systems have very small performance margins and adding additional complexity would either raise the price in terms of new hardware or make the device less responsive with additional software demands [5]. An adversary who has access to the gate-level design (either from a malicious insider or via reverse engineering) may only change a few gates while keeping the functionality intact, and introduce the IC as a new original circuit which can be sold under a new name. To address this problem, this paper proposes a technique using side-channel analysis to monitor the transient power consumption of an IC in order to detect if the IC has been altered during the manufacturing process 1(a).

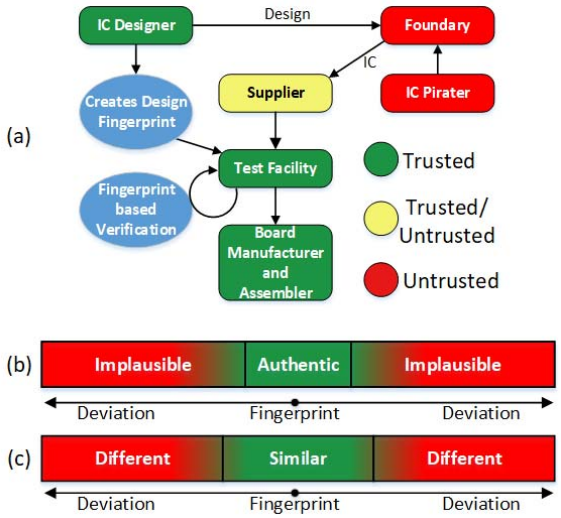


Figure 1: (a) Supply chain allowing a board manufacturer/assembler receiving components from a trusted/untrusted foundry via a trusted/untrusted supplier. Our method provides the design-based fingerprints that the trusted test facility will utilize to detect piracy. (b) Continuum showing likeliness measures of an IC compared to a given fingerprint used for authenticity certification. Green shows a match and red depicts a mismatch. (c) Continuum showing likeliness measures of an IC compared to a given fingerprint for "theft" detection.

In this paper, we present a new methodology for determining likeness of an IC to a precharacterized implementation. The designer generates a likeness measure that is fundamentally tied to the gate-level design and manufacturing process parameters. This likeness metric could be used for two purposes. One is to certify an IC as authentic (exact design and manufacturer) by enforcing a tight bound on the similarity, thereby rejecting any IC with design or manufacturing differences. The second application is to prevent theft, by catching "similar" designs sold under a different branding with a loose bound designed to catch a stolen and slightly modified design that keeps functionality intact.

2159-3477/18/$31.00 ©2018 IEEE
DOI 10.1109/ISVLSI.2018.00117

The first application (henceforth called authenticity certification in this paper) is important because in a non-vertical supply chain as shown in Fig. 1(a), there are multiple avenues for pirated ICs to enter the supply chain (through untrusted foundries or 3rd-party suppliers). Our method for supply verification allows a board manufacturer/assembler to verify the purchased ICs and distinguish them from pirated versions. In practice, in our method, the IC designer provides additional information to trusted entities to help them in distinguishing original designs from pirated versions using a tight tolerance on the fingerprint (See Fig. 1(b)).

The second application (henceforth called theft detection in this paper) is where a manufactured IC may not completely conform the GDSII file sent by original designer, i.e. a rouge element in the manufacturing process *alters the design such that the new device is functionally equivalent to the original one*, and may have reduced performance. This application targets the circuits that have same functionality, yet different gate-level implementations (in terms of a few gates) and hence a larger window for catching similar ICs as shown in Fig. 1(c).

The following major contributions are presented in this article:

1) A technique is proposed to determine if intellectual property (IP) manufactured within an IC can be verified by a design-based fingerprint provided by the IP designer. This fingerprint is created via analysis of the IC's power consumption, which using an SVM classifier can accurately determine if there is a change in a circuit from an original circuit.
2) Simulation-based evidence is presented showing that classification of the number of changed gates in a circuit from an original circuit can be determined, which does not require additional watermarking circuitry. This analysis includes considering the effect of temperature variations and transistor mismatch along varying sample rates on the performance of the classifier.
3) A temperature-dependent model is proposed for classifying changes in a circuit to mitigate real world temperature variations.

The remainder of paper is organized as follows. Section II covers existing techniques of detecting hardware changes. Section III covers the setup of the circuits analyzed, the simulation parameters used, and the setup of the classifier. The results are presented and discussed in Section IV. Finally, in Section V conclusions are drawn and future work is discussed.

II. Related Work

Prior work in the field of pirated ICs has concentrated on two avenues. The first is inclusion of and testing for fingerprinting, watermarking, etc. to protect IP [1, 3, 6]. The second is detection of hardware Trojans, additional elements added to a circuit that includes a trigger and payload to produce a desired effect [7].

To combat pirated devices, additional elements are added to an IC to help identify the IPs included in a design, because these elements are only understood by the designer it is difficult or impossible to remove them and therefore a pirated device may still contain these elements [4]. Common methods are watermarking, obfuscation, and fingerprinting. The term watermarking refers to embedding ownership information into an IC that can later be verified to show the company whose IP is included in the IC. Prior work in the area verifies the presence of a watermark finite state machine though power analysis [2]. The term obfuscation refers to the insertion of additional elements to the circuit such that, by using the correct key the device will operate correctly, but without the key the exact functionality can not be determined [6]. Recent work on obfuscation has extended the technique from combinational logic to include simple finite state machines [8]. Unlike these methods that require additional hardware, our method requires no additional hardware and can therefore be a cost savings. The term fingerprinting refers to ensuring that each IC has a unique identifier and each IC can be tracked though the supply chain its identifier [3]. Recent work in this area deals with leakage and switching power of the gates to create a device fingerprint [9]. Each device has a unique signature and therefore needs to be handled individually after manufacturing, whereas our method creates one signature for a design based on simulations and is therefore less costly to implement.

Altering the final circuit during the manufacturing process is very similar to Trojan insertion. Hardware Trojans are the elements that are maliciously inserted in circuits and mainly include trigger and payload parts. The trigger can be a specific input vector, or a sequence of input values that activate the Trojan circuit. When activated, the payload can result in circuit malfunction or data leakage [10, 11]. Trojan detection methods mainly rely on characterizing path delays [11, 12], and/or electromagnetic emissions[13]. However, in our threat scenario, the malicious change doesn't alter the functionally of a victim circuit, nor does it result in data leakage. Using path-delay based Trojan detection schemes to detect our victim circuits is costly because of the scalability of these methods with the exponentiation growth of the number of paths needed to be monitored. In practice, in our threat scenario, the adversary does not need (and even better not to) target critical or near-critical paths for gate changes. Thereby, path-delay based techniques encounter scalability issues as they need to monitor several paths. In addition, although our gate change can be considered as a Trojan but Trojan-detection schemes that rely on functionality change fail to detect our pirated ICs as in our threat scenario the functionality is not changed and so the even so-called Trojan is never triggered. On the other hand, we believe that our technique can be adapted for detecting Trojans as well. However, further investigation is needed to confirm this ability and therefore in this paper we focus on distinguishing the ICs which follow our threat model, and leave the Trojan detection problem as a future research.

III. Proposed Method

To create a design-based fingerprint, a circuit is simulated with a set of input patterns and the current drawn from the circuit is monitored. The Fast Fourier Transform (FFT) of this time-domain signal along with the signals from the altered circuits (though functionally equivalent). Two classifiers are

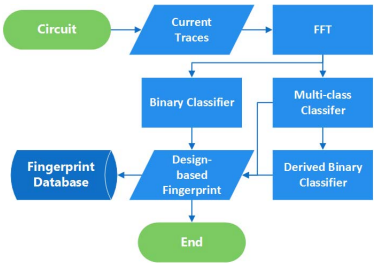


Figure 2: Flowchart demonstrating the generation of design-based fingerprints from current traces using binary and multi-class classifiers. A FFT is performed on current traces and the results passed to multiple classifiers, which combine to create the design-based fingerprint.

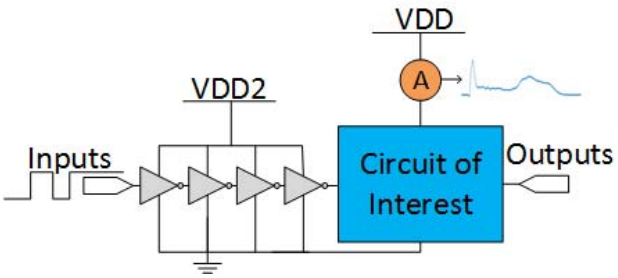


Figure 3: Simulation setup used in the experiments showing four chained inverters for input waveform smoothing and the location where the current is being measured.

created, a binary to identify the original circuit from altered circuits and a multi-class to identify the number of changes in a circuit. The second classifier can be dimensionally reduced to create a binary classifier. From these classifiers, a design based-fingerprint is created which allows a given IC to be analyzed in order to determine if the IC is a pirated one or not and in the case of piracy, how many gates have been changed. This sequence is shown in Fig. 2. This process can be applied to larger circuits via partitioning the circuits and considering one signature for each part using the above technique.

A. Circuit Setup

The circuits evaluated in this paper are selected from the ISCAS-85 benchmarks [14] and are detailed in Table I. The benchmarks represent a wide range of application areas and sizes. The original circuits for this work are an implementation of these standardized circuits. The experimental setup is shown in Fig. 3. The four chained inverters model a typical driver (e.g. limited slew rate). The current measured for this paper is that from the power supply that feeds the circuit and not the driver inverters. Altered circuits are created from the original circuit by a Python script that randomly selects either 1, 2, 5, 10 or 100 gates changes the selected gate from a NAND to an AND followed by an inverter, an AND to a NAND followed by an inverter, a NOR to an OR followed by an inverter, etc.

B. Circuit Simulation

To evaluate the effectiveness of the proposed design-based fingerprint verification, five different ISCAS'85 benchmarks

Table I: ISCAS'85 circuit descriptions.

Circuit	Number of Gates	Function
c432	215	27-channel interrupt controller
c499	245	32-Bit Single-Error-Correcting Circuit
c1355	589	32-Bit Single-Error-Correcting Circuit
c5315	2972	9-Bit ALU
c7552	4042	32-Bit Adder/Comparer

specified in Table I were simulated. Device-level simulations were performed at the schematic level for this paper. An in-house tool was used to generate the transistor-level model of the considered benchmarks. Using a 45-nm technology extracted from the open-source NANGATE library [15], transistor-level simulations were conducted using Synopsys' HSpice.

For each benchmark circuit (base circuit), we generated several altered circuits as follows. We changed one gate (selected randomly) of the base circuit to a functional equivalent gate(s) and repeated the process to generate 20 altered circuits. We also generated 20 circuits each with 2, 5, 10, and 100 gate changes compared to the original netlist of each benchmark circuit. To consider the effect of process variations and show the effectiveness of our method under the process variations, for each benchmark circuit, we conducted 620 Monte Carlo simulations of the base circuit and 31 simulations of each of the 20 altered counterparts using same randomly generated input sets to feed each circuit and its all counterparts.

For each of the simulations, the current traces were extracted. Simulations were carried out using the following process-variation parameters with a Gaussian distribution: transistor gate length L: $3\sigma = 10\%$; threshold voltage V_{TH}: $3\sigma = 30\%$, and gate-oxide thickness t_{OX}: $3\sigma = 3\%$. The process variation data reflects a 45-nm process in commercial use today [16]. The simulations were conducted assuming $45°\text{C}$ operating temperatures and $10\,\text{ps}$ temporal resolution and were controlled by a Python program.

C. Golden Waveform Generation

The use of a *golden waveform* and its effect on the accuracy of the classifier is investigated by comparing the results of the classifier with and without preprocessing by subtracting the *golden waveform* from the data. A *golden waveform* is generated by taking the average of the original circuit simulations, for this work, it is the average of the 620 simulations.

D. Classification Setup

In this paper, a support vector machine (SVM) is used for classification as it provided the highest accuracy when compared to other machine learning techniques. A similar scheme was used in prior work [17]. A basic binary SVM maps the training data into a higher dimensional feature space and then attempts to find a boundary-defining expression that supports useful separation of data while minimizing errors [18]. The binary classifier can be extended to a multi-class problem with such methods as an "one vs all" approach. This compares one class to the remaining data and tries to find

a boundary to maximize the separation between classes [19]. We use a binary classifier for both authentication certification and theft detection, however in the case of theft detection we can use a multi-class classifier to find an approximate number of gate changes. Additionally, the multi-class classifier is dimensionally reduced to create a derived binary classifier.

The SVM classifier was trained using two different preprocessing methods. The first was preprocessing by taking the time-domain waveform and windowing around the transients caused by input transitions. This is done to limit the amount of data processed and to eliminate noise contribution from expected quiescence periods after settling. The FFT was taken of the resultant windowed time-domain signal. The SVM was then trained using 5-fold cross validation to test the accuracy of the binary and multi-class classifications. The second preprocessing method is similar to the first, however, prior to windowing the time-domain data, the *golden waveform* is subtracted from the time-domain signal. The difference is then windowed and the remaining preprocessing steps are the same. We analyzed both time and frequency-domain classifiers, however the frequency-domain classifier resulted in 2% more accuracy and yielded less false negatives. Therefore we only present the frequency-domain data.

To confirm the accuracy of the classifier and to verify that the classifier is able to classify the particular number of gate changes, the classifier was fed with another completely separate set of data that included the same number of runs as the original, but had different gates changed from the original training data. The outcome of this classification supports determining if the classifier is overfit to particular data.

IV. Experimental Results and Analysis

A. Classification Results

Directly creating the classifier for the ISCAS'85 circuits shown in Table I, resulted in the accuracy shown in Table II. This is the accuracy of our authenticity certification and for all cases resulted in at least 83% accuracy. The binary classification was developed using a binary SVM classifier, while the derived binary was developed from a multi-class SVM classifier and then reduced to the binary outcome. A representative binary confusion matrix (for c432 circuit) is shown in Table III with the rows signifying actual class and the columns showing the inferred class. The accuracy of the multi-class SVM classifier is shown in IV and a representative multi-class confusion matrix (for c432 circuit) is shown in Table V. This demonstrates the ability to perform theft detection. In the multi-class case, direct classification yielded better performance for circuits with fewer gates.

Table II: Accuracy of direct binary vs. derived binary classification classifier using 10 ps temporal resolution.

	Direct Classification		Golden Waveform	
	Binary	Derived Binary	Binary	Derived Binary
c432	98.70%	93.00%	>99.9%	>99.9%
c499	98.30%	86.70%	>99.9%	>99.9%
c1355	>99.9%	>99.9%	>99.9%	>99.9%
c5315	83.4%	83.3%	92.5%	98.3%
c7552	96.1%	97.8%	96.9%	99.0%

Table III: Representative confusion matrix for the ISCAS'85 c432 circuit using 10 ps temporal resolution. Showing 551 unmodified circuits identified as unmodified and 113 altered circuits identified as unmodified.

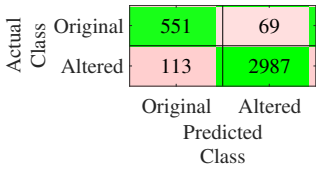


Table IV: Accuracy of direct classification vs. classification using a *golden waveform* with derived binary and multi-class classifier using using 10 ps temporal resolution.

	Direct Classification		Golden Waveform	
	Derived Binary	Multi-Class	Derived Binary	Multi-Class
c432	93.0%	93.8%	>99.9%	93.8%
c499	86.7%	74.0%	>99.9%	87.6%
c1355	>99.9%	95.8%	>99.9%	95.5%
c5315	83.3%	29.4%	98.3%	42.0%
c7552	97.8%	46.4%	99.0%	47.5%

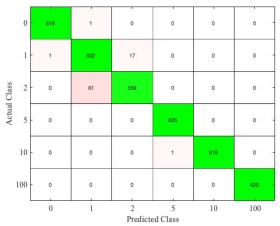
B. Golden Waveform Implementation

To create the *golden waveform* discussed in Section III-C 620 simulations of the original circuit (shown in Fig. 4 (a)) were averaged. The resultant *golden waveform* is shown in Fig. 4 (b). The *golden waveform* is visually differentiable in the time domain from the average of the waveforms for the simulations as shown in Fig. 5 (a) vs. Fig. 5 (b). The ability to differentiate the waveforms in the time-domain infers that classification is possible.

C. Golden Waveform Classification Results

Using the *golden waveform*, to classify the circuits resulted in the binary accuracy specified in Table II and multi-class accuracy specified in Table IV. A representative confusion matrix for data processed with the *golden waveform* for the binary case is shown in Table VI and multi-class confusion matrix shown in Table VII. In all cases the binary classifier performs better using the *golden waveform* and comparing binary vs. derived binary classifiers, the accuracy was higher in all cases for the derived binary. The multi-class classifier, also has accuracy gains for all but one circuit, which had a 0.3% decrease. Based on the binary and multi-class classification accuracy results, the use of the *golden waveform* outperforms

Table V: Confusion matrix for direct classification of ISCAS'85 c432 circuit using 10 ps temporal resolution. Each row represents the number of gate changes from the original circuit while each column is the outcome of the classifier with number of gate changes.



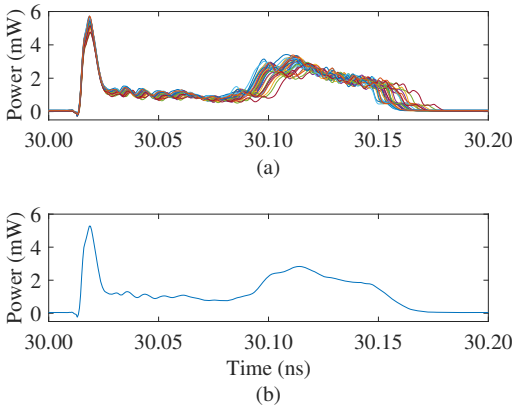


Figure 4: (a) The time-domain current traces showing 30 of the 620 Monte Carlo simulations demonstrating transistor mismatch and (b) the time-domain *golden waveform* for the ISCAS'85 c432 circuit.

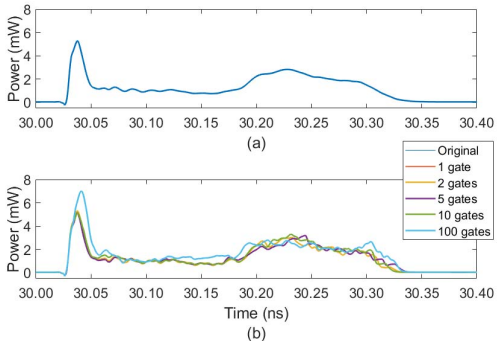
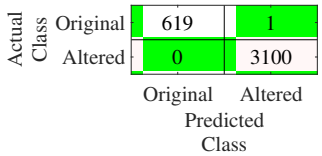


Figure 5: (a) Time-domain *golden waveform*, compared to (b) the average for each of the following: Original circuit, 1 gate modified, 2 gates modified, 5 gates modified, 10 gates modified, and 100 gates modified for the ISCAS'85 c432 circuit.

the direct classification. Using the *golden waveform* trained classifier, the binary accuracy can be maintained about 99% or 269 defects per million.

To verify that the classifier is performing correctly, a new set of simulations were run with the same number of runs as in the training scenario for the c432 circuit; twenty runs of thirty-one MC simulations each with a different number of changed gates (1, 2, 5, 10 and 100 gates). The results show the accuracy of 77.9% with the confusion matrix shown in Fig. VIII. This shows there is some over fitting to the data, but it

Table VI: Representative confusion matrix for *golden waveform* binary classification for the ISCAS'85 c1355 circuit using 10 ps temporal resolution.



maintains the ability to distinguish an altered circuit from an original circuit with 100% accuracy.

D. Varying Temporal Resolution Results

For this work, a 10 ps temporal resolution was used. Examining the frequency spectrum of the *golden waveform* shown in Fig. 6a, a different temporal resolution may yield the same accuracy, but at a lower temporal resolution needing less computation. To evaluate the effect of varying the temporal resolution, simulations were conducted with a temporal resolution of 1 ps and 100 ps and the resultant frequency spectrums are shown in Fig. 6a and Fig. 6c respectively. The accuracy of the resultant sampling rates for the multi-class classifier are 93.8%, 95.5%, and 93.5% for 1 ps, 10 ps, and 100 ps temporal resolution respectively. For all cases accuracy is >99.9% for the binary classifier and >93% for multi-class classifier. The higher accuracy at 10 ps when compared to 100 ps for the multi-class classifier can be explained by the extra dimensions from the 100 ps classifier. Additional preprocessing steps to limit the dimensionality of the classifier can be performed. As it stands the accuracy for all cases is >99% for binary classification and >93% for multi-class classification.

E. Varying Temperature Results

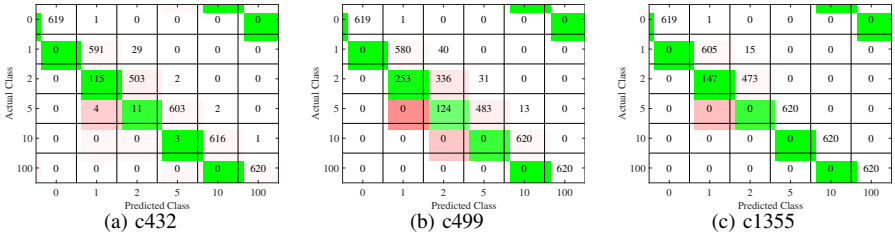
The baseline temperature used in this paper is 45°C. To test the effects of temperature on the classification accuracy, simulations were conducted at 25°C and 70°C to cover the range of many commercial ICs. The results of these simulations were used by the classifier trained with 45°C data and resulted in 33.6% and 28.1% accuracy, respectively for the multi-class classifier. When the same temperature data was used to train the classifier as the data being classified the accuracy improved to 95.4% and 92.1% for 25°C and 70°C, respectively. This means that the classifier is temperature dependent, and therefore the classifier must be created from data at the same temperature that subsequent testing and classification is going to be performed at. For a lab environment this implies that the temperature of the IC needs to be held within a temperature band to accurately classify the results. In an uncontrolled environment, the temperature should be recorded and a classifier created for that temperature.

V. Conclusions and Future Work

The experimental results confirmed that by deploying a design-based fingerprint and preprocessing using the *golden waveform*, our binary classifier has an almost 100% accuracy rate for authenticity certification. Going to the multi-class classifier for theft detection, we can predict the number of gates changed with an accuracy of 77.9% for a data set with unknown gates that are not part of the training data. In this research, we also considered the impact of sampling rate and showed that it is robust (>90% accuracy) for temporal resolutions between 1 ps and 100 ps.

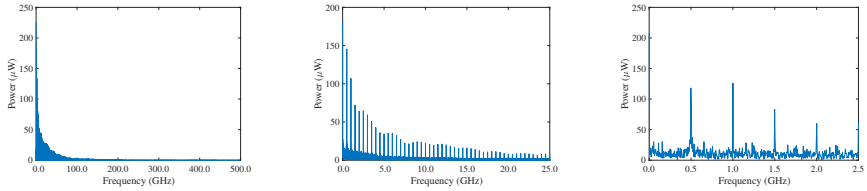
Future directions will include creating a temperature-independent classifier and test the ability to detect Trojans. Additionally, the effects of aging on a circuit may change

Table VII: Confusion matrices for the *golden waveform* classification of ISCAS'85 circuits using 10 ps temporal resolution.



(a) c432 (b) c499 (c) c1355

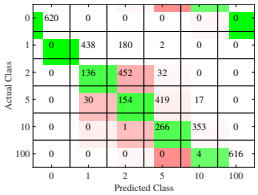
(d) c5315 (e) c7752



(a) 1 ps temporal resolution (b) 10 ps temporal resolution (c) 100 ps temporal resolution

Figure 6: Frequency-domain of the *golden waveform* for the ISCAS'85 c432 circuit with different temporal resolutions.

Table VIII: Confusion matrix showing the results of cross validation data conducted with the classifier for the c432 circuit.



its design-based fingerprint and needs to be investigated. The impact of noise on the circuit, as well as layout-level routing and power-grid effects and variations need to be accounted for and included in the models to more accurately represent real world conditions in the simulations. We expect these variations to have minimal effects on the outcome of this technique, since they can be accounted for with additional simulation parameters. Future work will expand on the these premises while maintaining a high accuracy.

References

[1] R. D. Newbould, J. D. Carothers, J. J. Rodriguez, and W. T. Holman, "A hierarchy of physical design watermarking schemes for intellectual property protection of IC designs," in *IEEE Int. Symp. on Circuits and Syst.*, 2002, pp. IV–862–IV–865.

[2] C. Marchand, L. Bossuet, and E. Jung, "IP watermark verification based on power consumption analysis," in *IEEE Int. System-on-Chip Conf. (SOCC)*, Sept 2014, pp. 330–335.

[3] B. Liu, Y. Jin, and G. Qu, "Hardware design and verification techniques for supply chain risk mitigation," in *Int. Conf/ Comput.-Aided Des. and Comput. Graphics (CAD/Graphics)*, Aug 2015, pp. 238–239.

[4] M. Ni and Z. Gao, "Watermarking system for IC design IP protection," in *Int. Conf. on Commun., Circuits and Syst.*, vol. 2, June 2004, pp. 1186–1190.

[5] K. Ly, W. Sun, and Y. Jin, "Emerging challenges in cyber-physical systems: A balance of performance, correctness, and security," in *IEEE Conf. Comput. Commun. Workshops (INFOCOM)*, April 2016, pp. 498–502.

[6] G. Qu, "Publicly detectable watermarking for intellectual property authentication in VLSI design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst*, vol. 21, no. 11, pp. 1363–1368, Nov 2002.

[7] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in *IEEE Symp. Security Privacy*, May 2007, pp. 296–310.

[8] Q. Yu, J. Dofe, and Z. Zhang, "Exploiting hardware obfuscation methods to prevent and detect hardware trojans," in *IEEE Int. Midwest Symp. on Circuits and Syst. (MWSCAS)*, Aug 2017, pp. 819–822.

[9] S. Wei, A. Nahapetian, and M. Potkonjak, "Robust passive hardware metering," in *IEEE/ACM Int. Conf. on Comput.-Aided Des. (ICCAD)*, Nov 2011, pp. 802–809.

[10] M. Tehranipoor and F. Koushanfar, "A survey of hardware trojan taxonomy and detection," *IEEE Des. Test of Comput.*, vol. 27, no. 1, pp. 10–25, Jan 2010.

[11] Y. Jin and Y. Makris, "Hardware trojan detection using path delay fingerprint," in *IEEE Int. Workshop on Hardware-Oriented Security and Trust*, June 2008, pp. 51–57.

[12] R. Vaikuntapu, L. Bhargava, and V. Sahula, "Golden IC free methodology for hardware trojan detection using symmetric path delays," in *Int. Symp. on VLSI Design and Test (VDAT)*, May 2016, pp. 1–2.

[13] J. He, Y. Zhao, X. Guo, and Y. Jin, "Hardware trojan detection through chip-free electromagnetic side-channel statistical analysis," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 10, pp. 2939–2948, Oct 2017.

[14] F. Brglez and H. Fujiwara, "A neutral netlist of 10 combinational benchmark circuits and a targeted translator in FORTRAN," in *IEEE Int. Symp. on Circuits and Syst. (ISCAS)*, June 1985.

[15] NanGate FreePDK45 Open Cell Library. [Online]. Available: http://www.nangate.com/?page_id=2325

[16] N. Karimi and K. Chakrabarty, "Detection, diagnosis, and recovery from clock-domain crossing failures in multiclock SoCs." *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 9, pp. 1395 – 1408, 2013.

[17] D. Chang, S. Ozev, O. Sinanoglu, and R. Karri, "Approximating the age of RF/analog circuits through re-characterization and statistical estimation," in *Design, Automation and Test in Europe Conf. (DATE)*, March 2014, pp. 1–4.

[18] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 42, no. 2, pp. 513–529, April 2012.

[19] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar 2002.