DETECTION OF MALICIOUS HARDWARE IN INTEGRATED CIRCUITS AND FIELD PROGRAMMABLE GATE ARRAYS

By

Trey Reece

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

December, 2009

Nashville, Tennessee

Approved:

Professor William H. Robinson

Professor Bharat L. Bhuva

Electrical Engineering

ABSTRACT

Detecting malicious modifications to a circuit is a daunting task, regardless of the medium. In a fabricated circuit, most methods of detecting malicious hardware (i.e., Trojans) rely on small changes in side-channel measurements, which can easily be disturbed by the presence of severe process variation. In a Field Programmable Gate Array (FPGA), the reprogrammability and design transparency reduces the initial difficulty of inserting a Trojan to a circuit. This thesis suggests solutions for both situations. The first method uses a signature generated from altering the supply voltage to the circuit in a controlled manner; this process leads to a change in shape of the transient current response of the integrated circuit (IC). Simulation results presented show significant differences between circuits with and without malicious hardware, despite large variations in individual transistor parameters. Second, a Trojan protection-by-design method is presented where a controllable ring-oscillator is inserted into a circuit in order to detect modifications that change the timing of its component gates. This design was able to win 2nd place in a competition held by the Polytechnic Institute of New York University.

ACKNOWLEDGEMENTS

I would like to thank my family and friends for supporting me through this research, as well as my adviser Dr. William H. Robinson, for his guidance and support throughout this research. I would also like to thank Dr. Bharat L. Bhuva, for his advice whenever I lost direction on areas of improvement. This work would not have been possible without help from all of you.

TABLE OF CONTENTS

Chapter

LIST OF FIGURES

Figure

LIST OF TABLES

Table

CHAPTER I

INTRODUCTION

As technology becomes increasingly advanced, possible threats to an advanced system also grow in complexity. When creating complicated circuits for secure applications, unless the chip is fabricated by a trusted source, the possibility exists for additional malicious hardware to be inserted covertly into the chip. This thesis details several methods of detecting such insertions in two different mediums. These two possible mediums for malicious modifications are Application Specific Integrated Circuits (ASICs) [1] and Field Programmable Gate Arrays (FPGAs) [2].

Application Specific Integrated Circuits are most likely the platform with the highest vulnerability to malicious modifications. This is due to the rising costs of constructing a fabrication plant, which have forced most companies to rely on third party foundries. In order to make up for this lack of security, the chips need be tested after receipt for possible intrusions. As the fabricated chips are generally of a very small feature size, it is both difficult and expensive to image the chips to determine their exact structure. Additionally, such an imaging process is generally destructive, and therefore cannot be performed on all of the fabricated chips. Other techniques often rely on known safe-chips for comparison, but usually only a simulated copy is available. The test method proposed in this thesis matches signatures from this simulated copy to signatures generated from copies under severe process variation. Simulations show that a signature generated from the change in an integrated circuit's transient current response can be used to identify the possibility of a malicious modification through increasing amounts of process variation. Since the trend for smaller feature size has also resulted in a great deal of process variation, this is a valuable result. This technique would be effective for identification of possible modifications in ASICs through strong variations. Even if additional tests were deemed necessary, this technique could be used to identify a 'safe' chip with which to conduct further tests.

FPGAs have different characteristics from ASICs [3], allowing for the use of other types of defenses. Since an FPGA is inherently designed for ease of use and flexibility for design, it is an excellent vehicle to implement defense mechanisms which require modifications to the original design. These tests are advantageous in that they allow for logical test vectors to measure the possibility of a modification, without the need to use additional side-channel measurements for specific timing or currents [4]. Unfortunately, such defensive structures are also present at the time of insertion, and therefore possible to design around. A method proposed by this thesis is the use of a circular set of NAND-gates forming a ring-oscillator. By spreading these gates around the chip, the manipulation of the hardware within the ring has a strong likelihood of disturbing the period of this oscillator. This design was submitted to a contest conducted by the Polytechnic Institute of New York University, and received second place in the overall competition. This oscillator could detect changes in the circuit from the additional hardware added by the other teams, and detected more modifications than any other team.

The organization of this thesis is as follows. Chapter II presents background concepts and related work from other researchers. Chapter III presents the side-channel detection method proposed for detecting modifications through hardware variation.

Chapter IV describes the use of a ring-oscillator as a defensive structure inserted into a circuit during the design stage. Finally, Chapter 5 concludes this thesis.

CHAPTER II

BACKGROUND CONCEPTS

As integrated circuit feature sizes are steadily decreasing size, the cost of fabricating these circuits is also increasing at a prodigious rate. In many cases, small to medium-sized companies cannot afford to construct their own fabrication plant, and instead must rely on outsourcing the fabrication of chips to a third party. However, without access to the services of a trusted foundry [5], this outsourcing gives third-party fabrication plants an opportunity to implant unwanted or even malicious hardware into the requested chips. Unfortunately, the features on these chips are of such a small size that determining whether they follow the requested designs is an expensive and often destructive process. One example of such is the use of a Focused Ion Beam [6]. The Focused Ion Beam works similar to an electron microscope, except that it fires ions instead of electrons. These ions then cause secondary particles from their collision which can be measured and used for imaging. Unfortunately, this interaction causes the material being viewed to be damaged during imaging. This damage, in addition to the fact that the chip would need to be broken apart into its separate layers, would effectively destroy the chip. Since it would be impractical to test all of the chips using a destructive test such as this, other methods of identifying possible modifications are needed.

Possible Risk of Trojans

Speculations about hidden hardware inside of fabricated Integrated Circuits have recently been given press and made public. In 2007, a Syrian air defense system had a critical failure at the exact same time as an Israeli Air Force attack [7], triggering suspicions of a built-in kill-switch used to disable the radars. Although there is a lack of concrete evidence of an inherent fail-mode, there has been a great deal of speculation as to the validity of these. However, the fact remains that with modern technology, this unlikely scenario is not as much of a fantasy as it may seem.

In 2008, Dr. Sam King and several others at the University of Illinois at Urbana Champaign [8] showed that by adding 1341 gates to the open source LEON3 processor [9], which natively has around 1.8 million gates, they were able to induce a hidden *shadow mode* of operation in which the processor would accept remote commands and allow the attacker complete high-level access. Because this was an insertion of about 0.08% of the total gates, it would be very difficult to detect. Additionally, the shadow mode was triggered via the receipt of a specially-crafted network packet that would never have been included in a standard test vector.

While this is an example of a back-door Trojan, there are also many other theorized possible types and payloads. A malicious modification could vary from a thinned wire or a transistor of reduced size, to complex assortment of combinational logic gates causing the circuit to act in a different manner under specific circumstances [10]. One of the few things similar to these designs is that they are intentionally difficult to notice. For this reason, a well-designed Trojan would not be activated while testing the circuit with the standard operability test vectors. It would instead be similar to the activation vector for the Trojan in the LEON3, in which a corrupted network packet acted as the trigger. As most modern integrated circuits have many possible inputs as well as a great deal of possible states, it is generally an impossible task to test every possible input at every possible state [11, 12]. Alternative methods must instead be developed for identifying this additional logic.

Ongoing Research Efforts

The goal of any proposed detection technique must be to correctly identify the presence of an unknown modification in a supplied circuit. This classification will take place at the same time as the standard logical testing phase that chips go through to verify functionality and identify good and bad chips (Figure 1a). In the process of forming this classification, there are the possibilities of both falsely trusted circuits, as well as falsely suspected circuits (Figure 1b). This classification would generally be determined by calculating a probability of trustworthiness for a given circuit, and then selecting a specific point to separate 'Trusted' and 'Suspected' circuits. The selection of this point would determine which is more important to the tester: minimization of falsely suspected chips, or minimization of falsely trusted chips.

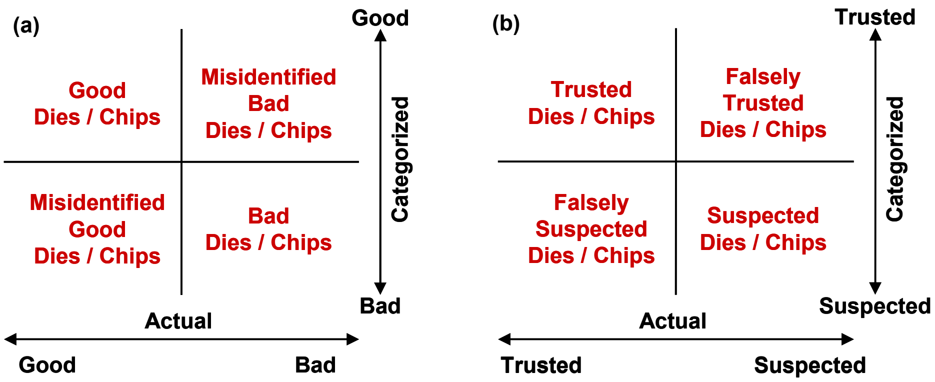

Figure 1: Testing to identify (a) functionality, as well as (b) possible chip modification

There have been several recent suggestions for detection schemes, which are typically classified into two groups: (1) timing analysis, in which the timing delay from the extra gates is expected to slow the response visibly [13, 14], and (2) electrical current analysis, where the extra current drawn by the inserted gates upon switching is measured on select inputs [15-17]. Both of these methods require access to a golden copy of the manufactured chip that is known to be without malicious modifications.

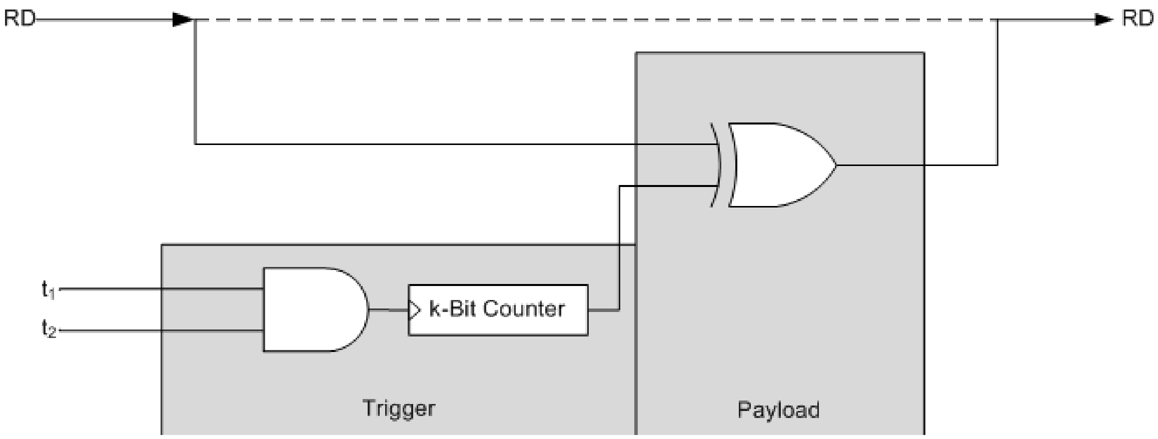

Figure 2: Schematic showing possible payload gate delay on timing [13]

Figure 2 shows a simple example of how a circuit's original timing could be affected by an inserted set of gates. The unknown trigger in the design is set to cause the payload gates to change the values of the logical line at some point in time. However, for these payload gates to be able to modify the output, they must add an additional gate delay to the logic path they modify. In order for this timing to be accurately compared, a golden copy must be used for comparison. However, identifying this golden copy can often be expensive or impractical, necessitating techniques either to detect suspicious trends without a golden copy, or to aid in the identification of a golden copy for use in further testing. Additionally, process variations in advanced technologies compound the difficulty in identification of such a golden copy.

Process variations are unavoidable side effects that impact individual transistor parameters due to advanced nanometer technologies. Although circuit design techniques have been developed to minimize the impact of such variations on overall circuit performance [18], the chip-level current drawn by an integrated circuit (IC) may vary significantly die-to-die, wafer-to-wafer, and lot-to-lot. Insertion-identification techniques that rely upon side-channel analysis [19] of circuit properties, such as chip-level electrical current or timing information, become much less effective in the presence of such process variations. These variations make it difficult to use the difference between the electrical parameters of two chips to determine the existence of additional hardware in one of the chips. Additionally, use of a poorly selected golden chip can incorrectly classify suspected chips or trusted chips due to process variations. Even determining a golden chip from a batch of unknowns ICs is a difficult and expensive process. Therefore, it is advantageous to investigate techniques to detect hardware insertion that do not require a manufactured golden copy.

CHAPTER III

SIGNATURE-BASED DETECTION THROUGH VOLTAGE STEPPING

While other methods of detection have generally required the presence of a golden copy for comparisons, this chip is often unavailable. In the presence of strong process variations, identifying a safe chip from among many unknowns is a difficult task. In order to counteract the problems introduced by process variation, the method proposed in this thesis uses signatures generated by inherent characteristics of the circuit. These characteristics will remain constant, regardless of the process variation introduced. As this signature is based on the circuit structure, it is possible to generate a test signature from the Process Development Kit (PDK) which was used to generate the layout for fabrication. This signature can then be used to classify suspicious chips, or even to identify a safe chip for further comparative testing.

Comparison Through Feature Extraction

While process variations impact the entire circuit, the effect from a well-hidden Trojan will be strongly localized to only the nodes which have been compromised. Therefore when detecting a Trojan, it is important to segment the circuit as much as possible so as to reduce the noise from process variation. This can be done in a number of ways, such as carefully selecting the input vectors so as to only activate parts of the circuit [20], or by observing power from only a section of the circuit so as to selectively remove the other sections from the response [21]. However, the solution presented in this thesis is to focus on small time windows in the electrical current response to isolate specific logic depths of execution. This can be done by locating large, easily identifiable features in the transient current response (such as maxima or minima), and shifting the responses such that the points coincide.

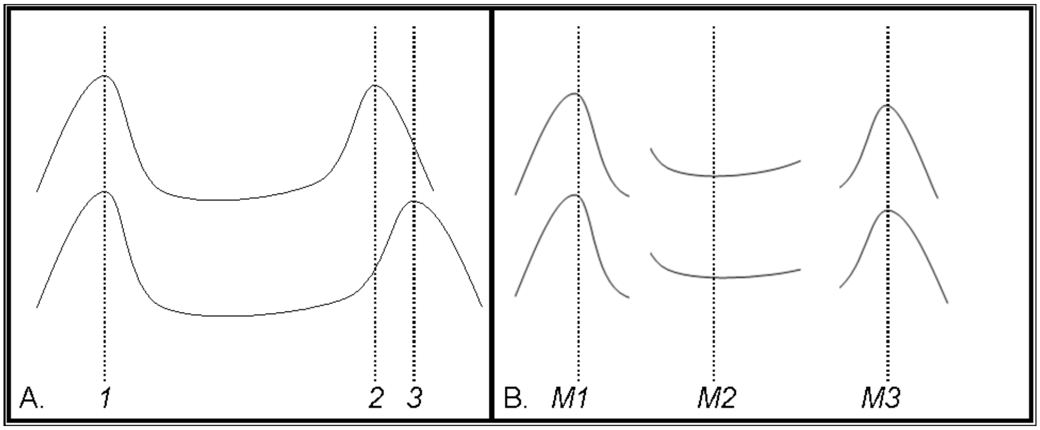

Figure 3: (A) Absolute comparison between two simple waveforms, and (B) piecewise comparison

The advantage of this method is that a large delay introduced early in the execution path will not have a repeated effect on the difference measured between the two circuits. In Figure 3 part ‘A’, despite the initial maxima being lined up at Point ‘1’, delay causes the maxima at Points ‘2’ and ‘3’ to be slightly offset, despite their similar shape. In observing this circuit, it is likely that at Points ‘2’ and ‘3’, the circuits are at the same stage in their execution paths, as it is unlikely that process variation would be able to change significantly the shape of a large feature such as this maxima. The act of separating them so as to align both Points ‘2’ and ‘3’ into the feature ‘M3’ allows for a significant reduction of the noise.

There are two issues to note with this approach. First, the most accurate results will be closest to the point at which the features are arranged, and as such, differences should be weighted accordingly. In most cases, a Gaussian distribution would be a logical selection, in which parameters such as standard deviation should be set according to the distance to the next feature. Second, the selected features must be strong, easy to identify points. In Figure 3, part 'B', Feature 'M2' is relatively ambiguous, and could shift a great deal due to process variation, while Features 'M1' and 'M3' are definitive and easy to identify. In order for Feature 'M2' to be used reliably, additional measures must be put in place to limit possible shifts in the timing.

Another benefit to this type of analysis is that when a suspicious point is identified in the current response, the timing can be compared to that of the known features to estimate a general logic execution depth. In combination with other segmentation techniques, it is possible to identify not only the existence of a possible Trojan, but the probable location as well.

Voltage Stepping to Compare Features in Integrated Circuits

The concept behind a signature-based method of detecting a malicious insertion is that for a given circuit, the signature can be generated from the Process Development Kit (PDK) provided by the manufacturer. Upon receiving fabricated ICs from a foundry, the signature measurements are compared to the simulated signature in order to determine 'trusted' and 'not-trusted' ICs. Whether this is used as a final determination as to the trustworthiness of various ICs, or merely the identification of a golden copy, it can be done at the same time as the standard functional test vectors are taken, and without damaging the chips.

A subsequent benefit of using features to compare waveforms is that it becomes possible to compare a circuit's response at one source voltage with the same circuit's response at a different voltage. Because the circuit's capacitance causes an exponential relationship between the current and the voltage, plotting the response on a log-log plot fits all of the waveforms to the same size and shape (See Figure 4).

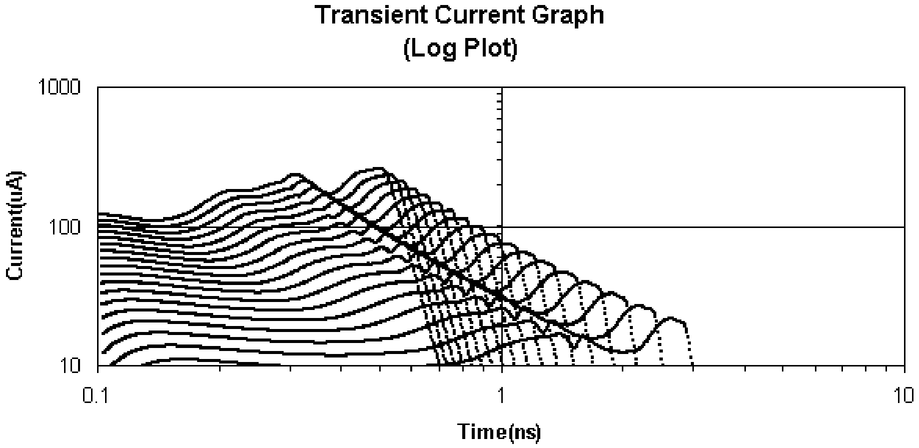

Figure 4: Transient responses at multiple values on a log-log plot

As long as the voltage exceeds the minimal operational threshold, the responses can be closely superimposed with a simple shift. An example of these responses is shown in Figure 5.

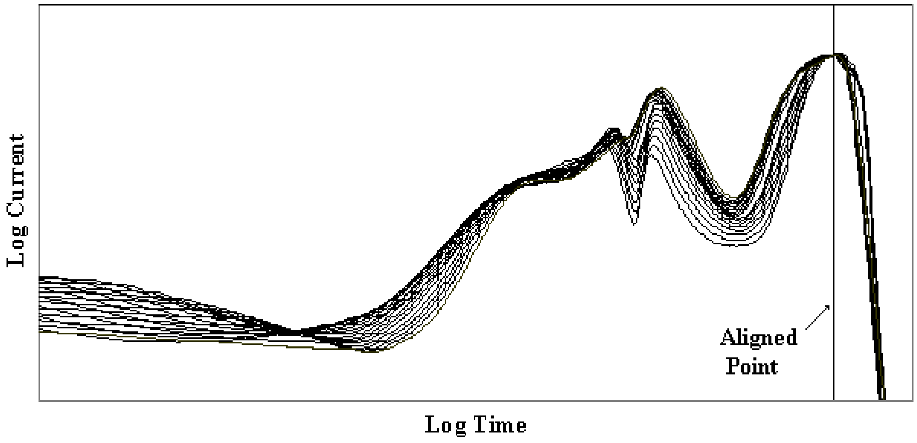

Figure 5: Aligned transient current responses as a function of time and supply voltage on a log-log plot.

Upon comparing the current responses at differing voltages, each individual feature will show specific trends as the voltage changes. These trends are inherent to the circuit, and very difficult, if not impossible, to control at design. These differences between the transient-current waveforms are the basis of the signature used to compare circuits.

The signature is calculated using the differences in measured current around easily identifiable features, namely maxima and minima in the transient response, illustrated by the aligned point in Figure 5. By aligning at these points, it is reasonable to infer that the active areas at that time location are generally equivalent. In this way, voltage step signatures can be generated at each feature. Figure 6 shows an example of a signature generated for a single feature. For increased performance, a large variety of signatures would be generated for testing, using a wide range of inputs and features.

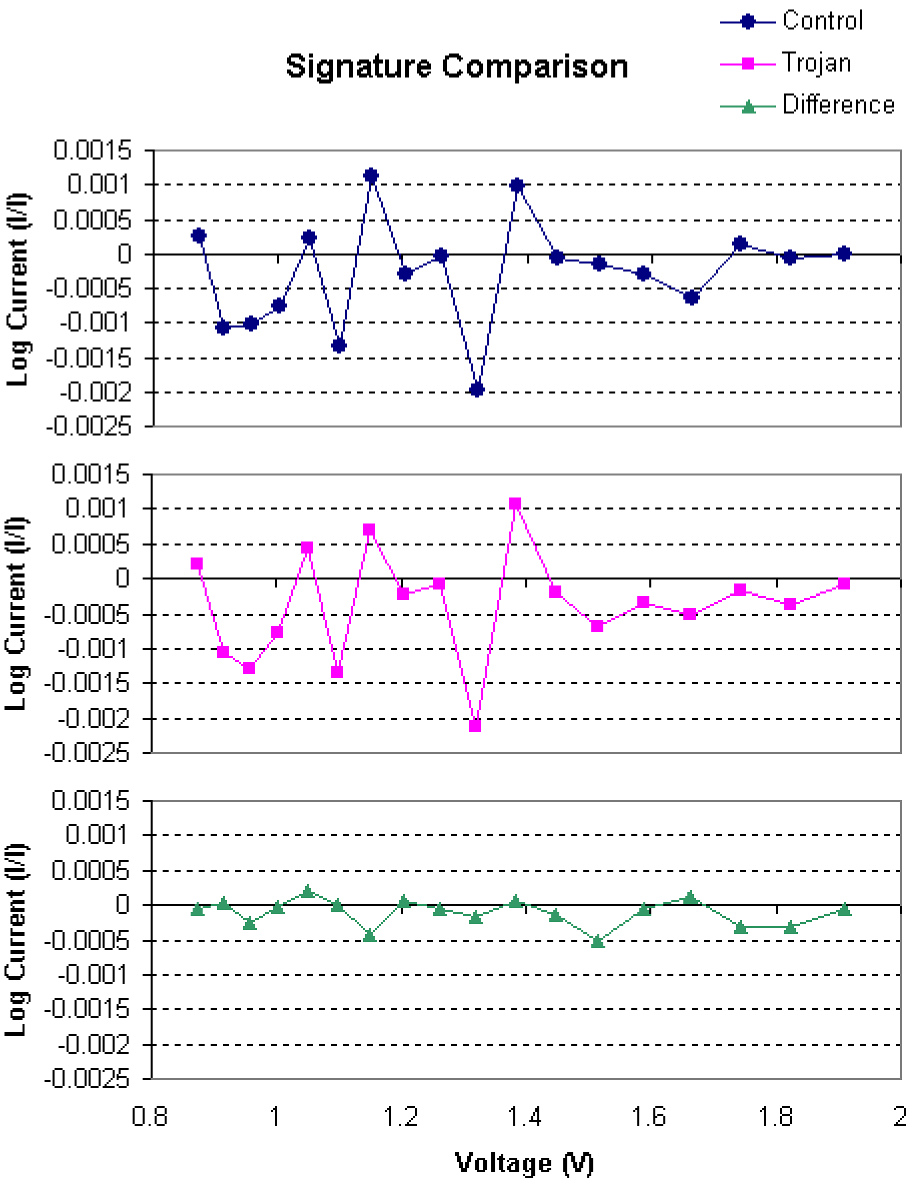

Figure 6: NAND Chain - Comparison showing a signature of an unmodified circuit derived via simulation (top), a signature of Trojan circuit with process variations (middle), and the difference between the signatures (bottom).

As a proof-of-concept test, this method was applied to a short chain of NAND gates, wherein which an additional NAND gate was connected to various nodes in the chain (Figure 7). As the NAND gate was disabled with a ground on its second input, this Trojan gate was forced to a non-controlling state, and remained unchanged despite the voltage of the compromised node. The only effect this gate could have on the remainder of the circuit was as an additional capacitance loading the connected node.

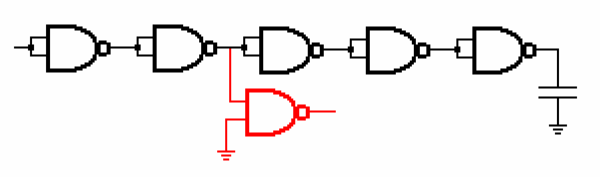

Figure 7: NAND chain with an extra inactive gate, with the Virginia Tech 250-nm cell library [22, 23]

After generating a test signature using the simulated schematic in Cadence Spectre, process variation was introduced to both Trojan and clean circuits by varying the threshold voltage, transistor width, and transistor length [24] by a Gaussian distribution with increasingly larger standard deviations. The cell library used in these simulations was the Virginia Tech 250-nm Cell Library [22, 23]. After comparing the signatures generated by these varied circuits with the initial signature, the difference due to the Trojan gate remained relatively constant regardless of the amount of induced process variation. Additionally, the depth of the Trojan could be accurately estimated by identifying the features at which this difference was most pronounced. A basic labeling of these features is shown in Figure 8 for the NAND chain transient. Figure 9 shows the differences as the Trojan NAND gate is shifted to progressively later nodes.

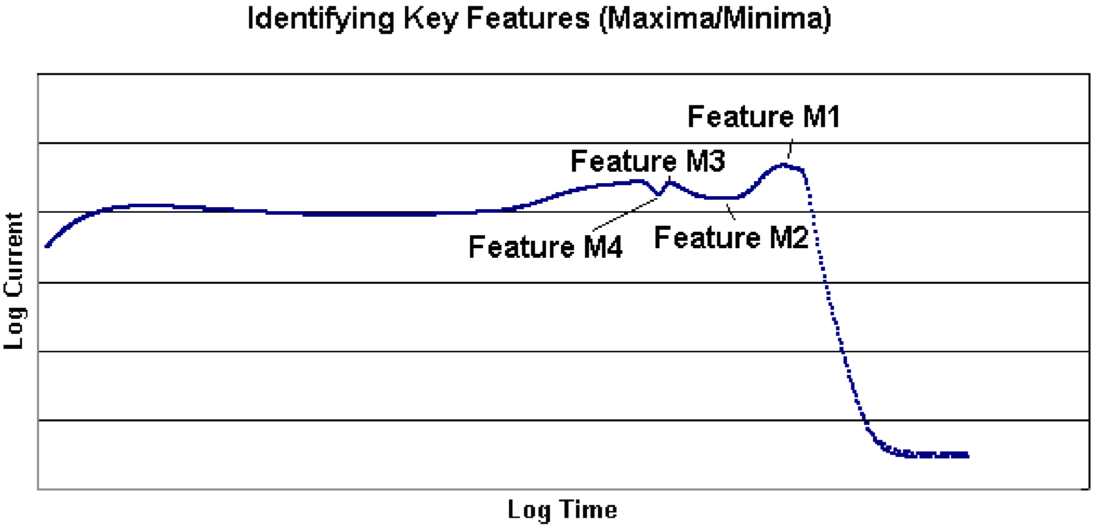

Figure 8: First four features in the transient response of the NAND gate chain

In Figure 9, as the Trojans move farther towards the end of the circuit, the differences are more pronounced in features later in the response. For example, when the Trojan is inserted to the 4th node, the response from Feature M1 dominates, whereas when the Trojan is in the 3rd node, the responses from Features M2 and M3 dominate, with a small change seen at Feature M1. This allows us to form a rough estimate as to the insertion depth of a Trojan based on the depth of the differences observed in the features.

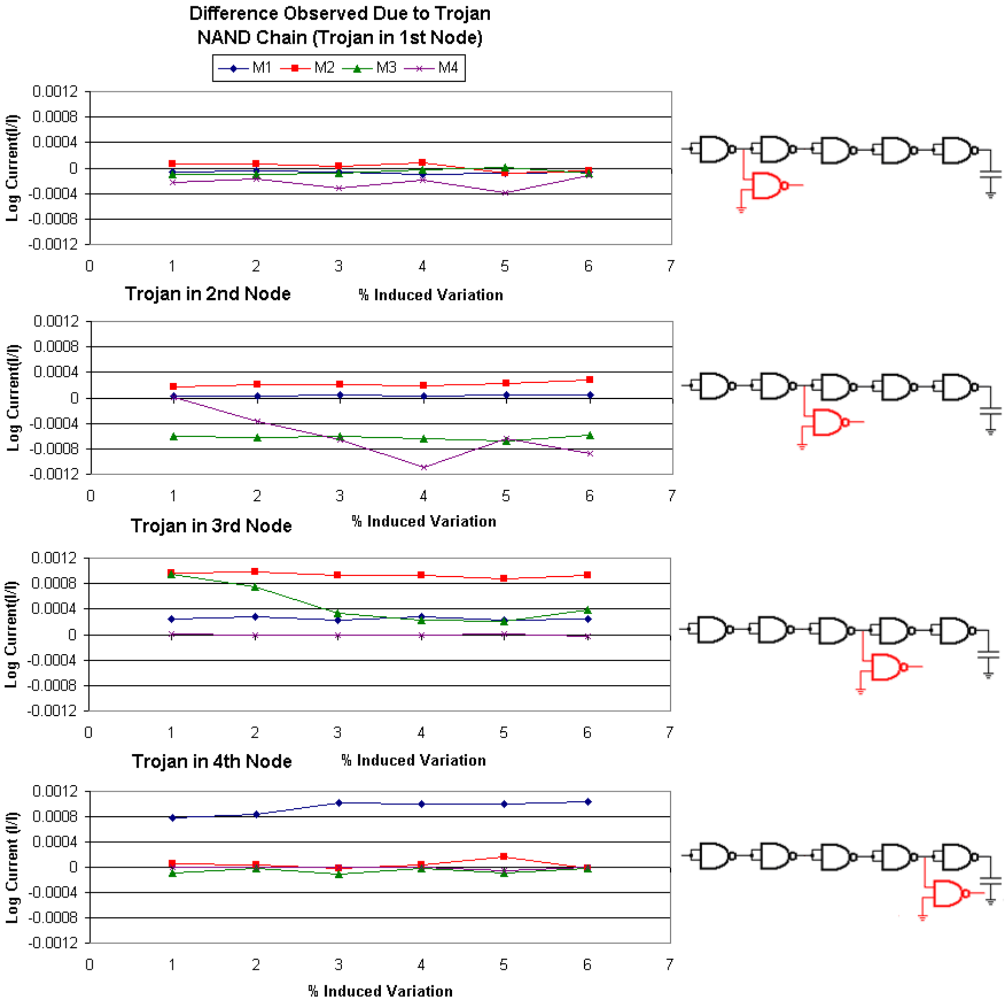

Figure 9: Results at different features varying by Trojan location

As a continued experiment, this method was applied to the 74181 benchmark [25], a standard 4-bit Arithmetic Logic Unit (ALU)/Function Generator. The procedure was tested by once again inserting a Trojan NAND gate into a random node in the circuit. The resulting signatures that were formed by circuits containing the Trojan consistently differed from the unmodified circuit's signature. The results from the first four features of this circuit's transient current response are shown in Figure 10.

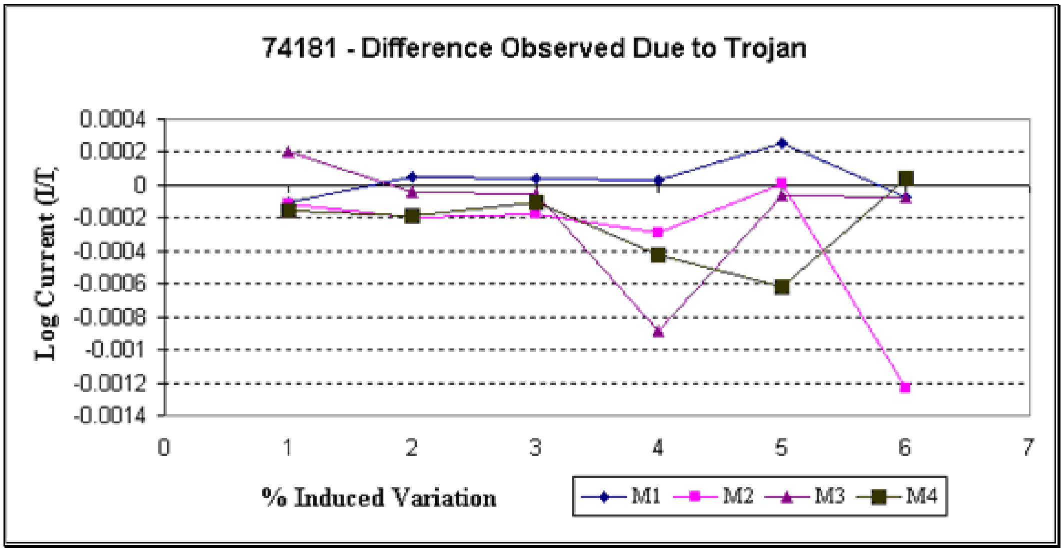

Figure 10: 74181 Benchmark results – Inserted gate causes net negative change in signature comparison. Amount varied as variations increased, but reliable differences were seen on features M3 and M4.

In this result, a significant signature difference is seen compared to what was expected due to merely process variation. Some of the features show very large differences as higher levels of variation are reached (M2, M3), and others show relatively constant differences (M4). However, these results can still be used to identify the presence of a modification. Because this is an ALU circuit, unlike the NAND chain, there are multiple input vectors available for testing. Upon testing several input vectors, if the above differences persisted through several different tests, then it would strongly suggest the presence of a modification. Likewise, if none of the tests from other input vectors resulted in a difference like the one shown above, then the change would be attributed to process variation. Additionally, the features showing the most prominent and reliable differences allow for a rough estimate as to the execution depth of the modification. A repeated difference at a specific feature could even assist in the identification of the rough location of the modification on the layout.

CHAPTER IV

TROJAN DETECTION IN FIELD PROGRAMMABLE GATE ARRAYS

Another method of protecting a circuit against malicious insertions is the act of building safeguards into the design before shipping the circuit to a fabrication plant. These safeguards need to be able to detect if a circuit has been disturbed from the original designs. In general, these protections are designed to identify a change in timing caused by the insertion of extra gates, be it from capacitance or gate delay. A good example of this is the 'shadow register' [26] structure in Figure 11.

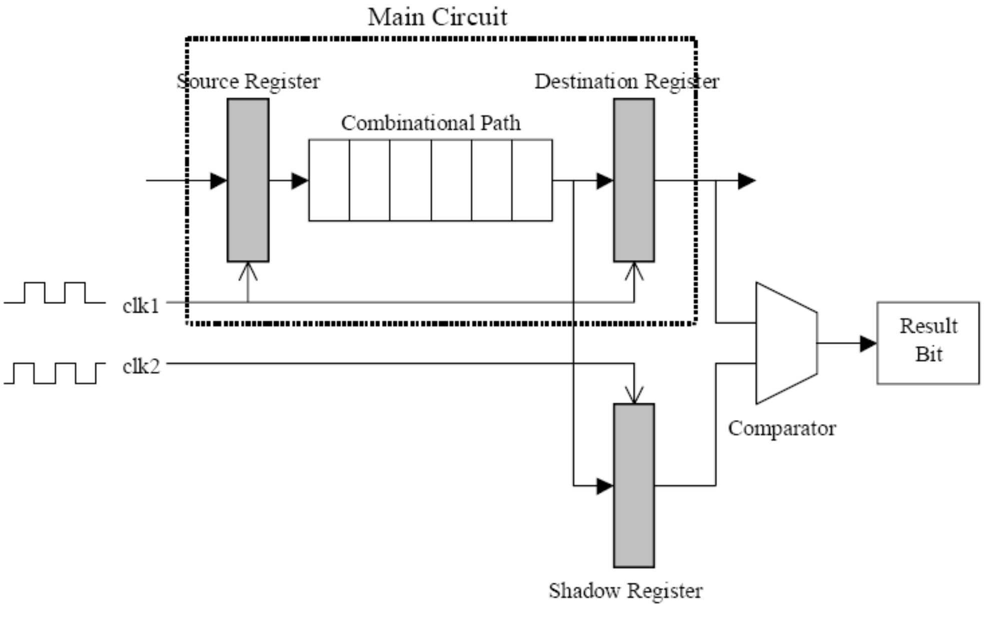

Figure 11: Implementation of shadow registers

This structure functions by inserting additional registers into the circuit which are then attached to a separate clock signal than the standard registers. When testing these circuits, a tunable clock is attached to identify the expected logic at specific time points, since the clock can be increased to capture interim values before the circuit has settled. If a value changes at a later time than expected, it strongly suggests the possibility of a Trojan insertion on that logical path.

As a structure often used as a substitute for an Integrated Circuit, Field Programmable Gate Arrays (FPGA) can be used as a platform to easily test prospective Trojan defenses. The basic structure of an FPGA is comprised of a large number of freely programmable cells, which are generally used as an alternative to an integrated circuit for applications requiring only small numbers of devices. This structure also allows for a greater ease of use, and generally more flexibility than a comparable IC would. Unfortunately, attaching defensive components to an FPGA is not as simple a task as it may seem, because this ease of use applies to the possible attacker as well. If a defensive structure is easy to implement, it is also much easier to work around. This makes it much more important that a defensive structure hide its presence in a design so that the attackers do not realize its purpose or mode of operation.

Implementation of a Ring Oscillator as a Defensive Structure

The simple structure proposed for detection is that of a concealed oscillating ring of gates spread out among key parts in a circuit. The output of this oscillator would be directly dependent upon the period of the oscillator formed by the logical ring. Any sizeable changes to the design within the area of the oscillator would likely perturb the locations of the oscillating gates, and cause a change in the frequency, and thus the output. In an ideal situation, several of these oscillators would be placed around the design, with all of their outputs observed for variations.

Since a standard ring-oscillator constructed from inverters would directly inform the attackers as to the defensive structure's purpose, an alternate design for this structure is suggested. By instead constructing the oscillator out of three NAND gates, it both conceals the function of these gates, as well as gives the oscillator a level of control (Figure 12). Except in the situation where all of the inputs are logical '1's, the output of the NAND gates follows standard combinational logic. However, in the case of all inputs being high, the NAND gates enter an unstable state, and will alternate values, forming an oscillating ring.

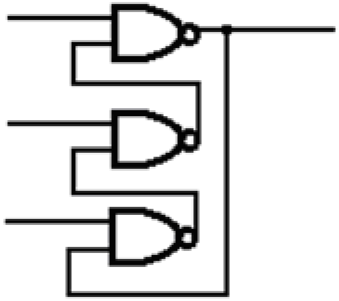

Figure 12: Three NAND gates form a ring oscillator when inputs are high.

Embedded Systems Challenge

The ring-oscillator as a detection technique was implemented in the design competition held by the Polytechnic Institute of New York University, dubbed the Embedded Systems Challenge [27]. This competition was carried out with the purpose of exploring the effectiveness of defensive structures, through implementation on a Digilent Spartan 3E FPGA development board, shown in Figure 13 [28]. Participants were instructed to modify the Verilog code for the encryption design ‘Beta’, so as to detect unwanted modifications.

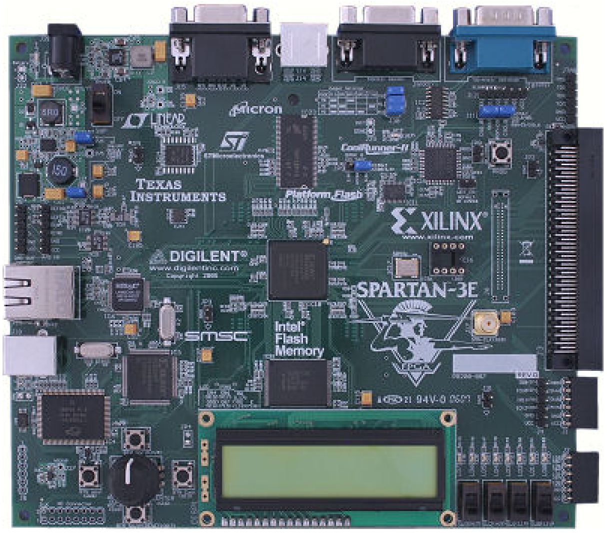

Figure 13: Spartan-3E Xilinx Development FPGA Board

The basic structure of the ‘Beta’ design was that of encryption device making use of the low-resource Trivium algorithm. The original design contained 451 gates, with a serial UART module, interpreter module, Trivium encryption module, and a JTAG TAP control module. It communicated via an RS-232 serial port, and would accept either ‘0’, ‘1’ characters representing data, or ‘k’ and ‘r’ characters, for replacing the key and for resetting the algorithm respectively.

Unfortunately, the tests were limited to logical tests and the use of side-channel measurements was disallowed. This restriction made the implementation of complex structures such as shadow registers much more difficult, since they would require additional circuitry to implement a tunable clock within the design. The additional circuitry would also help the attackers identify the purpose of the registers, and reduce the likelihood of falling for the trap.

The structure of this competition was set up in two phases: a defensive phase where teams insert detection structures into a supplied Verilog encryption design, and an attack phase, where the teams then attempt to insert Trojans into the protected designs. The teams were scored in this tournament by giving points for successful defenses, as well as successful attacks. Were an attack to fail, the team would also lose a point as a penalty.

Upon implementing the oscillating NAND ring to this design, it was important to conceal the purpose of the ring from the attackers, lest they design around it. For this purpose, several NAND latches were attached around the design in seemingly random locations to draw attention away from the oscillating gates. Finally, the inputs were attached to these gates such that upon a specific keystroke, the design would enter the unstable 'debug mode' where it would output a result based on the oscillating output. In order to test effectiveness of this structure, only a single ring of NAND gates were added to this design. In showing that 3 NAND gates are able to police an area 100 times larger than the added gates, it supports as a proof-of-concept for applying this method to other designs, and with larger and greater numbers of oscillating rings.

The results of the competition proved positive for this detection method, as it found more inserted Trojans than any other defense structure. Even without attacking designs from other teams, it was still able to gain enough points through purely defending to win 2nd place overall. Only one Trojan was able to slip by undetected (Table 1), and it occurred because the attacking team was able to identify the oscillator, its purpose, and specifically design so as to not disturb it significantly. In some cases the teams mistakenly thought that the debug mode was broken due to the unstable output.

Table 1 – Results of Oscillator Outputs for Embedded Systems Challenge

	Percent 1's	Average string length of 1's	Average string length of 0's	Trojan detected?
Control	45% +/- 3%	1.85 +/- 0.1	2.2 +/- 0.1	N/A
Yale	41.46	1.750	2.469	Yes
Case Western	43.74	1.813	2.332	Yes
UMD	43.71	1.810	2.330	Yes
NYU-JV	46.90	1.898	2.149	No

Table 1 displays the results of the oscillating output for each team's Trojan, as well as the original unmodified values with expected deviation. The control group represents the testing performed on an FPGA without insertion of any Trojans, as a representation of how the circuit should ideally act. After multiple tests of 1,000 bits, the expected values and a cutoff range were measured. The other groups values were each calculated through single tests of 10,000 bits. Although there were 5 attacking groups, one group's Trojan effectively broke the circuit, and was disqualified from the calculations. Of the attacking groups, only the New York University – JV team was able to insert a Trojan undetected. As is visible from the values, the Trojan inserted by this team did not disturb the oscillator to a large enough degree to suspect the circuit. This was mainly because this team correctly identified the ring-oscillator structure, and carefully inserted a Trojan as far away as possible from the 3 oscillating NAND gates.

Considering the size of the design, as well as the size of the modifications, this design could have easily been improved with the implementation of additional oscillating rings. These additional rings would have certainly made avoiding the defensive structures much more difficult. However, this structure worked very well in this competition to show that the concept of a ring-oscillator defense structure is effective.

CHAPTER V

CONCLUSION

The techniques described in this thesis are promising methods of introducing additional security to a relatively unsecured area of interest. Both the post-fabrication signature-based test and the pre-fabrication defense-on-design structure show possible methods of defending a circuit against an intrusion. Because this is a relatively young threat, there is still a great deal of work left to be done in researching detection techniques. This is even more important given that no malicious hardware has as of yet been officially identified in the wild. For defending against such intrusions, only speculation as to their possible structure is available. For the next step of testing this detection technique, several 74181 benchmark circuits are planned for fabrication, with some containing an extra insertion to a node. In addition to this testing, further extensive simulations with both larger circuits, as well as larger amounts of introduced process variation are currently being performed.

REFERENCES

[1] M. J. S. Smith, *Application specific integrated circuits*, Reading, Mass.: Addison-Wesley, 1997.

[2] J. Rose, A. El Gamal, and A. Sangiovanni-Vincentelli, "Architecture of field-programmable gate arrays," *Proceedings of the IEEE*, vol. 81, pp. 1013-29, 1993.

[3] I. Kuon and J. Rose, "Measuring the gap between FPGAs and ASICs," in *14th International Symposium on Field Programmable Gate Arrays*, Monterey, CA, USA, 2006, pp. 21 - 30.

[4] K. Tiri and I. Verbauwhede, "Simulation models for side-channel information leaks," in *42nd Annual Design Automation Conference*, Anaheim, CA, 2005, pp. 228 - 33.

[5] Trusted Access Program Office (TAPO), http://www.nsa.gov/business/tapo.cfm

[6] H. Satoh, M. Owari, Y. Nihei, "Three-dimensional analysis of a microstructure by submicron secondary ion mass spectrometry", *Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures*, Volume 7, Issue 4, July 1989, pp. 609 – 617.

[7] J. Markoff, “Old trick threatens the newest weapons” http://www.nytimes.com/2009/10/27/science/27trojan.html.

[8] S. T. King, J. Tucek, A. Cozzie, C. Grier, W. Jiang, and Y. Zhou, "Designing and implementing malicious hardware," in *1st Usenix Workshop on Large-Scale Exploits and Emergent Threats (LEET 2008),* San Francisco, CA, 2008.

[9] Aeroflex Gaisler: http://www.gaisler.com

[10] X. Wang, M. Tehranipoor, and J. Plusquellic, "Detecting malicious inclusions in secure hardware: Challenges and solutions," in *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST 2008)*, 2008, pp. 15-19.

[11] S. Edbom and E. Larsson, "An integrated technique for test vector selection and test scheduling under test time constraint," *13th Asian Test Symposium*, 2004, pp. 254-257.

[12] S. Y. Lee, K. K. Saluja, "Efficient test vectors for ISCAS sequential benchmark circuits," *IEEE International Symposium on Circuits and Systems*, 1993, vol.3, pp.1511-1514.

[13] Y. Jin and Y. Makris, "Hardware trojan detection using path delay fingerprint," *in IEEE International Workshop on Hardware-Oriented Security and Trust (HOST 2008)*, 2008, pp. 51-57.

[14] J. Li and J. Lach, "At-speed delay characterization for IC authentication and trojan horse detection," in *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST 2008)*, 2008, pp. 8-14.

[15] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in *IEEE Symposium on Security and Privacy (SP '07)*, 2007, pp. 296-310.

[16] X. Wang, H. Salmani, M. Tehranipoor, and J. Plusquellic, "Hardware trojan detection and isolation using current integration and localized current analysis," in *IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems (DFTVS '08)*, 2008, pp. 87-95.

[17] R. Rad, J. Plusquellic, and M. Tehranipoor, "Sensitivity analysis to hardware trojans using power supply transient signals," in *IEEE International Workshop on Hardware-Oriented Security and Trust (HOST 2008)*, 2008, pp. 3-7.

[18] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture" in *ACM/IEEE Design Automation Conference*, Anaheim, CA, 2003 pp. 338 - 342.

[19] R. M. Rad, X. Wang, M. Tehranipoor, J. Plusquellic, "Power supply signal calibration techniques for improving detection resolution to hardware Trojans," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD 2008)*, 2008, pp.632-639.

[20] M. Banga and M. S. Hsiao, "A novel sustained vector technique for the detection of hardware trojans," in *22nd International Conference on VLSI Design*, New Delhi, India, 2009, pp. 327-332.

[21] M. Banga, "Partition based approaches for the isolation and detection of embedded trojans in ICs," in Electrical and Computer Engineering. Master of Science Blacksburg, VA: Virginia Polytechnic Institute and State University, 2008, p. 65.

[22] J. B. Sulistyo, J. Perry, and D. S. Ha, "Developing standard cells for TSMC 0.25um technology under MOSIS DEEP rules", Department of Electrical and Computer Engineering, Virginia Tech, Technical Report VISC-2003-01, November 2003.

[23] J. B. Sulistyo and D. S. Ha, "A new characterization method for delay and power dissipation of standard library cells", *VLSI Design* 15 (3), pp. 667-678, 2002.

[24] A. Asenov, A. R. Brown, J. H. Davies, S. Kaya, and G. Slavcheva, "Simulation of intrinsic parameter fluctuations in decananometer and nanometer-scale MOSFETs," *IEEE Transactions on Electron Devices*, vol. 50, pp. 1837-1852, 2003.

[25] Benchmark Catalog, J.P. Hayes: http://www.eecs.umich.edu/~jhayes/iscas.restore/74181.html

[26] D. Rai and J. Lach, "Performance of delay-based Trojan detection techniques under parameter variations," *IEEE International Workshop on Hardware-Oriented Security and Trust*, June 2009.

[27] Polytechnic Institute of New York University, Embedded Systems Challenge: http://isis.poly.edu/csaw/embedded

[28] Digilent Inc. : http://www.digilentinc.com