PROGRESS REPORT
Fall 2016-2017

Team Number ECE-46

<title of project>

Team Members

Name	Department	Email
Sunny Shah	ECE	
John Akibayo	ECE	
Derek Phillibert	ECE	
Jason Fung	ECE	

Team Advisor(s)

Name	Department/Company	Email
Professor Ionnis Savidis	ECE	

Group Leader's Signature : _____

Advisor's Signature : _____

1. Abstract

Hardware manufacturers are outsourcing their integrated circuits (ICs) fabrication work overseas because of their lower cost of manufacturing. Outsourcing the fabrication, however, poses a significant security risk for ICs used in all modern electronic devices. One of those security risks include the introduction of a Trojan onto the hardware. The Trojan insertion happens in numerous ways during the work flow, including through untrusted foundries, synthesis tools and libraries, testing and verification tools, and configuration scripts. Trojan attacks can compromise security and privacy of hardware users directly, or through interaction with pertinent systems and application software or data. Our design project's objective is to develop a system to detect this type of malicious circuitry by implementing side-channel cryptanalysis. By using noise modeling, the side-channel's information such as power, temperature, and timing can be used to identify the Trojan.

The design project will consist of three main objectives IC synthesis, Trojan injection and synthesis, and data modeling. To accomplish these tasks, industry standard tools and techniques will be used to create a work flow that is standard. For the ASIC design part of our project, Synopsys will be used for the data analysis and custom VLSI Cadence Virtuoso will be used. Our team's primary focus will be to take an IC and do a baseline side-channel analysis against that IC. Inject the Trojan into the IC and once again do a side-channel analysis. Once the data has been collected via Virtuoso, the data modeling can be done to see if our algorithm will be able to pick up the Trojan, and as to what level or resolution we can detect a Trojan. With conclusive data, we will be able to generate a sensory network to detect Trojan in real-time.

Table of Contents

List of Figures

List of Tables

2. Problem Description

A recent article was released regarding hardware Trojans on Dell servers. The computer company warned some of its server motherboards have been delivered to customers, and some of those motherboards may be carrying unwanted "Hardware Trojans" [15]. Due to the critical use of these servers, both the Pentagon and Homeland Security have started to spend millions on research to prevent such vulnerabilities from occurring on their servers.

Integrated circuits (ICs) have become the backbone of virtually all modern devices. Devices (such as the Dell server motherboards, medical devices, automotive, industrial control systems, power management, military devices, smartphones, computers, banking systems, and much more) rely on ICs for proper functionality. Often, circuit blocks in a single IC are designed by different parties, and manufactured by an external, possibly offshore, foundry. Packaging is also done by an additional company, and then supplied by an independent distributor. With multiple parties being involved in the creation of a single IC over time, that specific IC is exposed to several security issues, and opens itself up to a security vulnerability. The IC can be exposed to hardware Trojans that will cause several performance issues with the IC.

Trojans can exploit vulnerabilities in any IC by running a myriad of malicious modifications to prevent its proper use. While Trojans initially lay dormant when loaded onto the IC, Trojan will become highly disruptive to the system once activated. For example, they can lock the functionality of the IC, creating malfunctions in the circuitry, or even release classified information and functionality related to the circuits operations.

Modern IC design often involves intellectual property cores, also known as IP cores, being sent to 3rd party offshore manufacturing companies and foundries. With the possibility of these Trojans being introduced to the hardware, security issues and loss of intellectual property become a major risk for all those involved in the ICs creation.

By actively looking for Trojans on the circuit, these issues can be avoided. While there are current practices (mainly through physical inspection, functional testing and several built-in tests) in place, none of these actively search for Trojans after the fabricated chip has been finished. Analyzing different signals emitted by the IC will allow for the detection of a Trojan. After collecting data from the IC, a baseline for each expected state can be created. Any alteration to that baseline can then be suspected as a Trojan.

3. Proposed Work and Deliverables

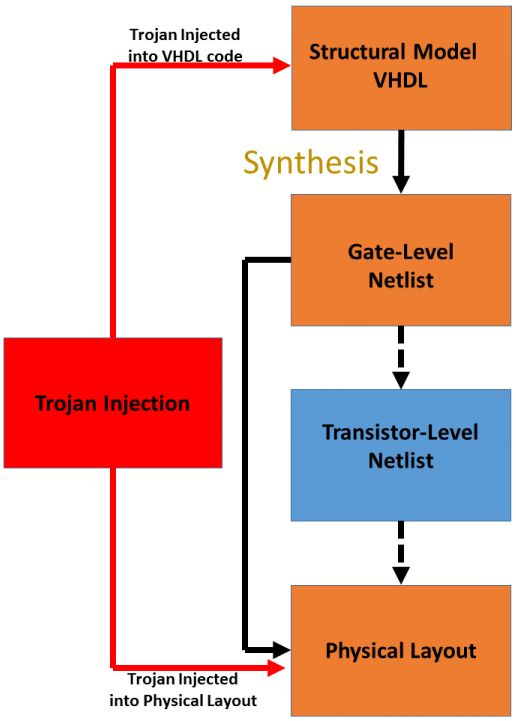


Fig 1. Block Diagram of Simulation

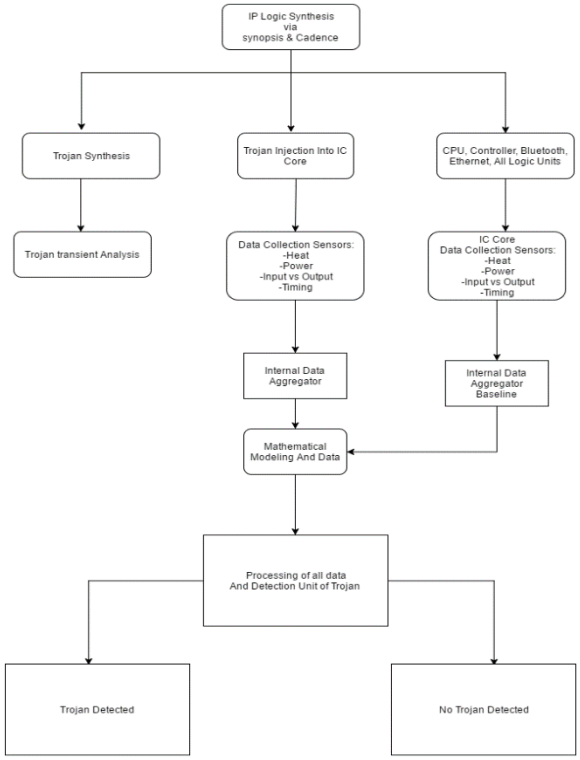


Fig 2. Workflow for Trojan Detection.

To address the issue of detecting a Trojan on an IC, the goal is to use simulation to develop a Trojan detecting algorithm. The process of synthesizing and simulating the circuit will be repeated for different ICs and Trojans. Fig. 1 shows our proposed workflow for a single simulation. Synthesis of an IC requires a process design kit, also known as a PDK. The PDK using in this project will be based on the Intel 180nm.

3.1. Circuit Synthesize via Synopsys

As seen in Fig. 2, the first task is to synthesize an IC with Synopsys. The first deliverable consists of a gate level netlist which will be imported into Cadence's Virtuoso program. An IC begins in hardware description language, coded in VHDL for this project. To generate a circuit diagram from the VHDL code, one must synthesize the VHDL code in Synopsys. Synopsys will read the VHDL code and provide a gate level netlist, which can be used by Virtuoso to generate an accurate schematic of the IC. Virtuoso will then be used to generate the physical layout.

3.2. Data Collection

Virtuoso will import the gate-level netlist and provide us with a circuit that can be simulated and tested, providing transistor-level layouts that will dictate the behavior of the IC. Once the IC is verified and proven to work properly, simulation and data collection will begin. Virtuoso allows for multiple types of simulations, including transient analysis and DC sweeps, opening the door for different tests to be run, as well as different sets of data to be collected. The results of the simulation pre-Trojan injection and post-Trojan injection will be collected with the use of an internal data aggregator. The data gathered from the IC relate to the power consumed, heat dissipation, the relationship between input and output voltages, and the relationship between input and output current. The data gathered will be necessary for determining if the IC does contain a Trojan, and will be the second deliverable for the team.

3.3. Developing Detection Algorithm

Based on the results in the internal data aggregator, a detection algorithm will be developed and tested. During this phase of the project the best placement of the probes during data collection will be decided based on the IP core. By using different Trojans and ICs, the team hopes to be able to report the resolution of the detection algorithm and the smallest Trojan that can be detected. The third deliverable will be what was discovered in relation to the resolution and size of a Trojan that was detected through the side-channels that we are using.

3.4. Trojan Design

The Trojans will need to be designed and coded by the team. The Trojans will consist of data monitoring devices and data sniffing devices. For example, an ammeter can be used as a data monitoring Trojan.

4. Operational Description

4.1 Synthesis

In creating an IC, logic-level code must first be created and synthesized. After creating VHD code to create the basis behind our logic level IC and Trojan, it is then necessary to synthesize the code in Synopsys to create a netlist. First, the IC will be coded and synthesized to create a schematic that can then be imported into Virtuoso. To become familiar with the software, the group has completed synthesis of simple logic gates.

4.1.1 IC Synthesis

For further analysis of different cores, the group will (and has begun) the synthesis of both a single-cycle and a multi-cycle CPU. A netlist will be created, showing the different logic gates needed for the fabrication of each CPU. While CMOS-level transistors can't be seen with this netlist, analysis can still be done through which logic gates are used, and the wiring done for the output.

4.2.2 Trojan Synthesis

In creating a logic-level Trojan, research must be done to find what it would consist of. Different Trojans must also be created to drive the data for this project. For proper analysis of IC detection, several different Trojan netlists will be created, each with different functionalities that will be used for detection. Detection should not be limited to a singular Trojan, so testing with different functionalities would be optimal, and prove detection on the IC.

4.2. Simulation

After synthesizing with Synopsys, the netlists created can then be imported into Virtuoso. If the code works as planned, the netlist will show the proper logic-level core. After being imported into Virtuoso, the CMOS-level transistor flow can be seen. With the transistor-level logic, the hardware can then be simulated.

4.2.1. Hardware Simulation

Too test the logic-level code created, simulations will be run with Virtuoso to verify the netlist created with Synopsys. After running proper simulations, the layout for the cores can be created. With the simulation, a baseline for expected results in the hardware simulation can be used for data analysis.

4.2.2. Trojan Simulation

Similarly, to hardware simulation, the netlists for the Trojans that were created can be imported into Virtuoso. The created Trojans can then be "injected" into the schematic of the hardware designed, and simulations can be run once again to see how it compares to the hardware simulation without the Trojan injected.

4.3 Data Analysis

After obtaining data on the ICs both with and without the Trojan injected, the output data can be compared to formulate some form of mathematical modeling. In formulation of a mathematical model for detection, research must be done to fully understand the calculations being done. After doing research in what would mathematically be considered a Trojan, it would then be necessary to incorporate the findings with what was discovered in the simulations.

4.4 Trojan Detection

After completing the data analysis and simulations, the final decision to be made would be to decide how a Trojan is detected. With Trojans varying in size and resolution, the main challenge would be to see how small a Trojan can be, as well as finding at what resolution can that Trojan be detected.

4.5 Physical Sensory Network

The Physical Sensory Network, will be a network of on-chip sensor that will actively be scanning the side-channels of the IC for any malicious activity. If the results from this side-channel analysis, design show that it is feasible to detect Trojans via the side-channels an optimal sensor network could be developed. Power side-channel

4.6 Trojan Activation Mechanisms:

When dealing with Trojans, it's important to understand their classifications. While Trojans are considered harmful, they can be classified based on three characteristics: their physical characterization, how they are activated, and the action the Trojans were designed to accomplish. As noted previously, once a Trojan activates, no matter how it is activated, the Trojan will typically do one of three things: transmit information, modify specification, or modify the designated function. Fig. 3 shows how a Trojan will be classified for this project. The proposed algorithm created for the IC hardware Trojan detection will search for any abnormal characteristic in the IC that will cause any of the action characteristics shown in Fig. 3.

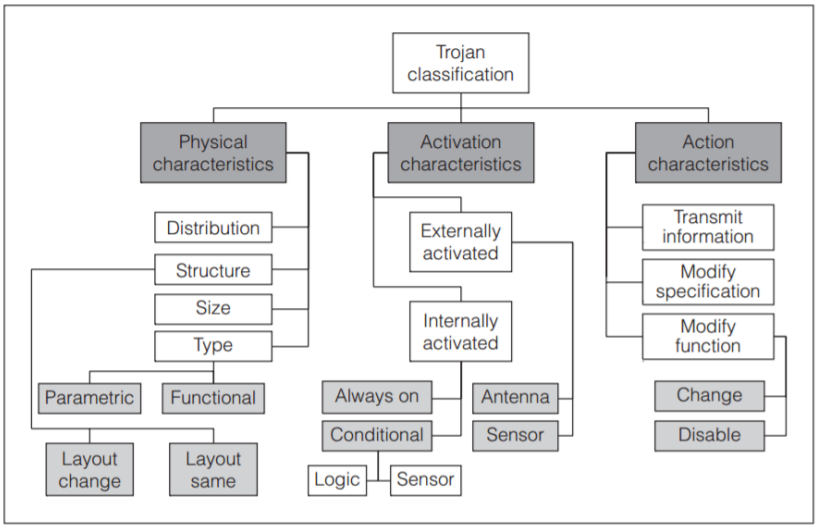


Fig. 3. Trojan Classifications

5. Completed Work

Thus far, the group has begun research on creation of an IC, and the steps it takes to simulate and create a fabricated IC. Prior to the experiment, there was a very basic level of knowledge on the fabrication, as well as the design methods used to create an IC with a specific function. However, there are several underlying layers in fabricating the IC. First, it was necessary to be familiar with Virtuoso, a program that is used to simulate the creation and use of custom integrated circuits. Since no one in the group had any prior experience, working with the program had started from the bottom, and a foundation of experience was built from the ground up. Next, the group began to work with Synopsys, a program that is used to create the software behind ICs, to its fullest ability. Synthesis of a logic-level IC was created, and Fig. 4 and Fig. 5 show the results obtained after synthesizing basic hardware in Synopsis. Finally, it was necessary to integrate the net list created by Synopsys and import it into Virtuoso.

The next task of the project is to begin synchronization of the two results. After synchronizing the net list from Synopsys and the created schematic from Virtuoso, a Trojan can then be inserted, and testing for detection can begin. As testing for detection occurs, it will also be necessary to make note of how that Trojan is detected, as well as the size of the detected Trojan. With that in mind, research on Trojan detection can (and has) begin. Side-channel analysis methods will be considered, and how they can be utilized in hardware detection of Trojans, and even possibly modify current methods to create a more efficient method.

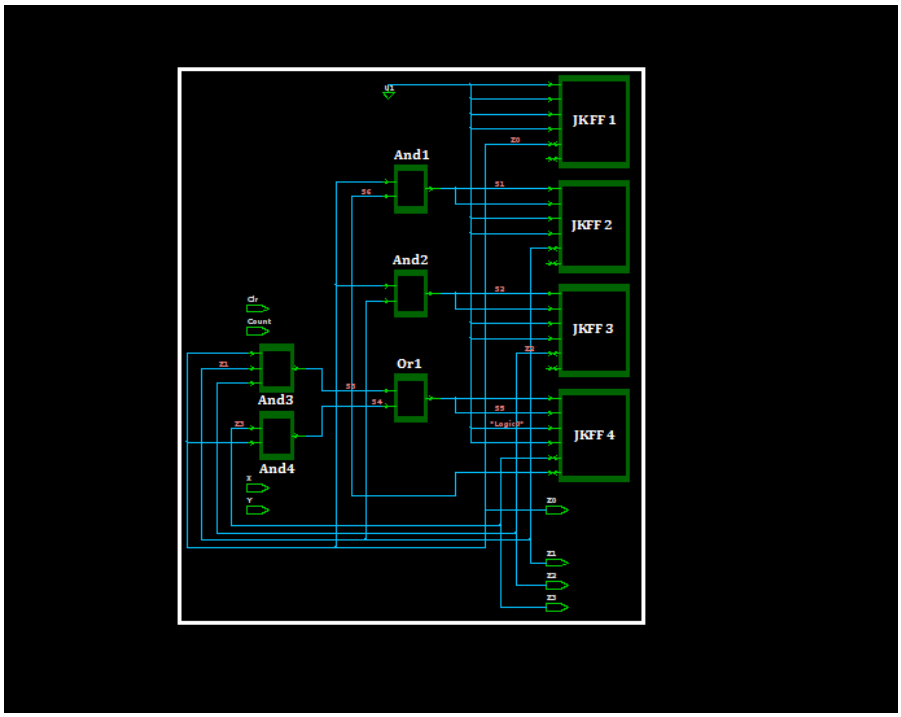


Fig.4. BCD Counter Synthesis

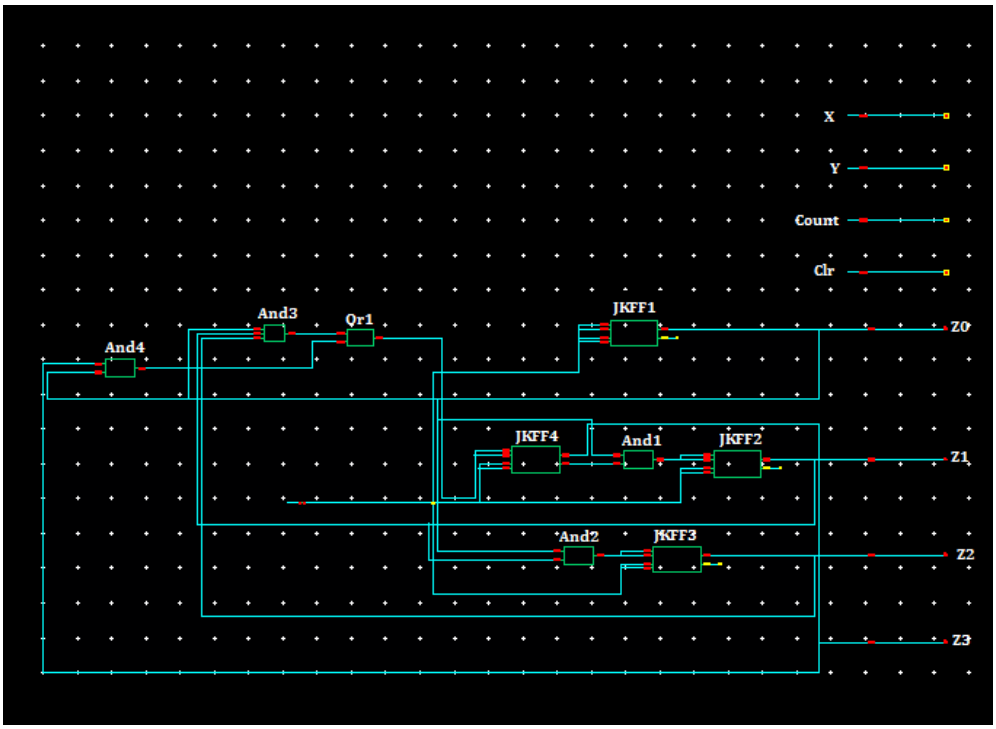
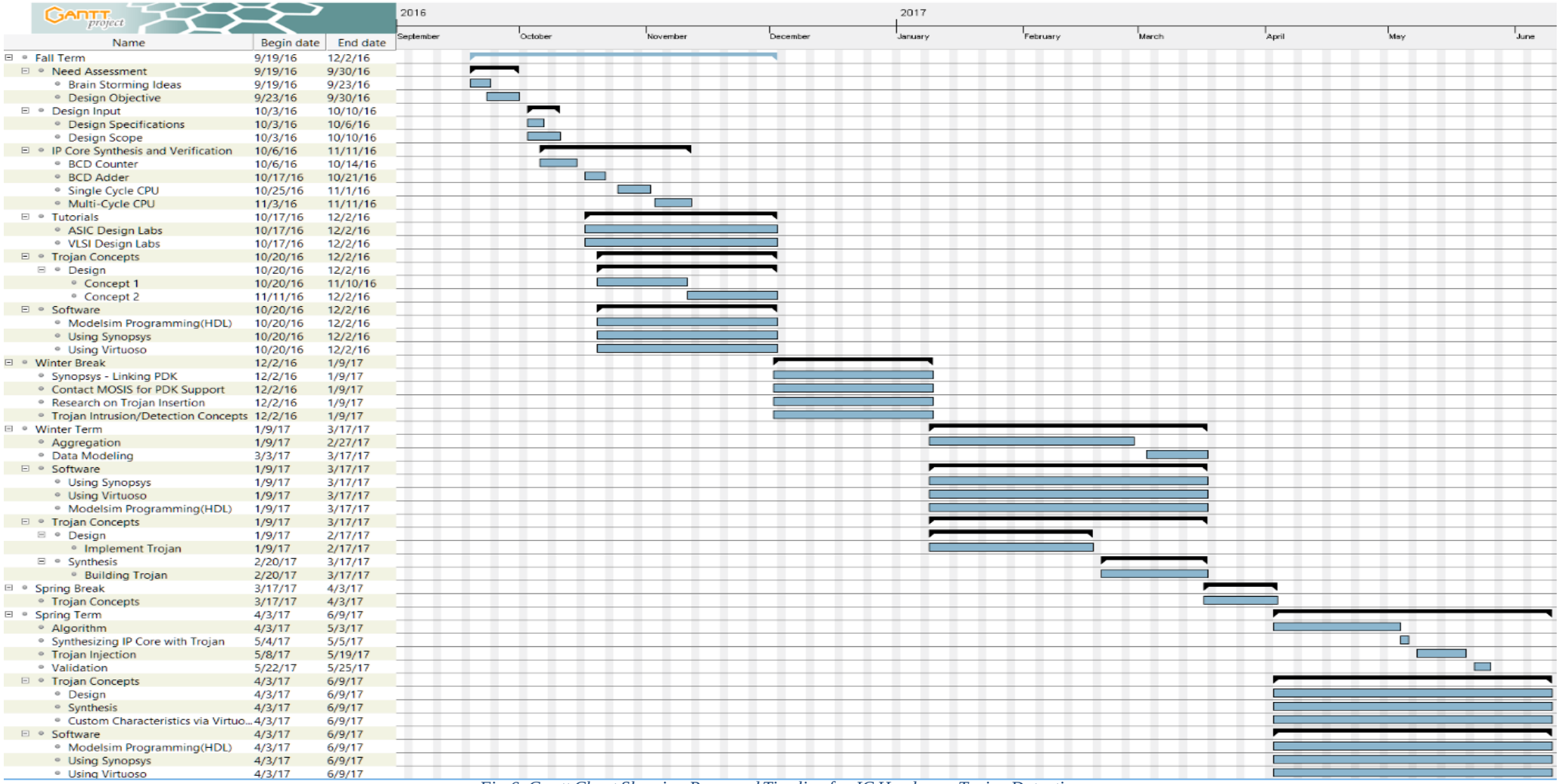


Fig.5. Gate Level BCD Counter in Virtuoso.

6. Work Schedule / Proposed Timeline



Name	Begin date	End date
Fall Term	9/19/16	12/2/16
Need Assessment	9/19/16	9/30/16
Brain Storming Ideas	9/19/16	9/23/16
Design Objective	9/23/16	9/30/16
Design Input	10/3/16	10/10/16
Design Specifications	10/3/16	10/6/16
Design Scope	10/3/16	10/10/16
IP Core Synthesis and Verification	10/6/16	11/11/16
BCD Counter	10/6/16	10/14/16
BCD Adder	10/17/16	10/21/16
Single Cycle CPU	10/25/16	11/1/16
Multi-Cycle CPU	11/3/16	11/11/16
Tutorials	10/17/16	12/2/16
ASIC Design Labs	10/17/16	12/2/16
VLSI Design Labs	10/17/16	12/2/16
Trojan Concepts	10/20/16	12/2/16
Design	10/20/16	12/2/16
Concept 1	10/20/16	11/10/16
Concept 2	11/11/16	12/2/16
Software	10/20/16	12/2/16
Modelsim Programming(HDL)	10/20/16	12/2/16
Using Synopsys	10/20/16	12/2/16
Using Virtuoso	10/20/16	12/2/16
Winter Break	12/2/16	1/9/17
Synopsys - Linking PDK	12/2/16	1/9/17
Contact MOSIS for PDK Support	12/2/16	1/9/17
Research on Trojan Insertion	12/2/16	1/9/17
Trojan Intrusion/Detection Concepts	12/2/16	1/9/17
Winter Term	1/9/17	3/17/17
Aggregation	1/9/17	2/27/17
Data Modeling	3/3/17	3/17/17
Software	1/9/17	3/17/17
Using Synopsys	1/9/17	3/17/17
Using Virtuoso	1/9/17	3/17/17
Modelsim Programming(HDL)	1/9/17	3/17/17
Trojan Concepts	1/9/17	3/17/17
Design	1/9/17	2/17/17
Implement Trojan	1/9/17	2/17/17
Synthesis	2/20/17	3/17/17
Building Trojan	2/20/17	3/17/17
Spring Break	3/17/17	4/3/17
Trojan Concepts	3/17/17	4/3/17
Spring Term	4/3/17	6/9/17
Algorithm	4/3/17	5/3/17
Synthesizing IP Core with Trojan	5/4/17	5/5/17
Trojan Injection	5/8/17	5/19/17
Validation	5/22/17	5/25/17
Trojan Concepts	4/3/17	6/9/17
Design	4/3/17	6/9/17
Synthesis	4/3/17	6/9/17
Custom Characteristics via Virtuo...	4/3/17	6/9/17
Software	4/3/17	6/9/17
Modelsim Programming(HDL)	4/3/17	6/9/17
Using Synopsys	4/3/17	6/9/17
Using Virtuoso	4/3/17	6/9/17

Fig.6. Gantt Chart Showing Proposed Timeline for IC Hardware Trojan Detection.

The Gantt chart, as seen in Fig. 6, shows the proposed plan and timeline for this project. Covering the span of nine months, the timeline splits the proposed work into three trimesters, which will span the Fall, Winter, and Spring Quarters at Drexel University, breaking down into three main objectives. The first objective is to get familiar with the software tools provided. The second objective is to design the IP core and Trojan integration. The third and final objective is to model the data. During the fall term, the group put together a need assessment, design Objective, Design Input. Which allows us to obtain scope of the project. Also, The IC Synthesis and Verifications, and Trojan concepts were developed. During the fall quarter, it was imperative to become familiar with the tools that would allow for the creation and synthesis of an IC circuit. Over time, the group will synthesize a circuit with Synopsys, as well as create circuits and layouts with Virtuoso. In the following winter quarter, the focus will be put on importing the synthesized netlist into Virtuoso. By importing the netlist, simulations can be run, and then data can be gathered to create a baseline for the data collected. With the collected data, algorithms will be created for the detection of a Trojan to be tested. Spring Term, will involve the testing and modification of the algorithm created to see as to what resolution of a Trojan will be detected. Then Final prototyping of the solutions.

7. Industrial Budget

Software Budget			
Item	**Quantity**	**Price**	**Total**
Virtuoso	4	$15,000	$60,000
Synopsys	4	$38,000	$152,000
ModelSim	4	$3000	$12,000
MATLAB	4	$150	$600
IBM 180nm PDK PDK Silicon Fa. (50 Units)	1	~$7,200	~$7,200
Total			$419,800

Table 4. Budget for All Software Used in IC Trojan Detection.

Industrial Budget			
Item	**Quantity**	**Price**	**Total**
Computer Engineer Salary	4	$120,000	$480,000
Laptop	4	$900	$3,600
Total			$483,600

Table 3. Budget for All Industrial Components for IC Trojan Detection.

Total	$903,400

8. Out-Of-Pocket Cost:

Due to this design project being done through simulation, no external hardware needed to be purchased. Drexel also provided all software used, so there was no out-of-pocket cost.

9. Societal, Environmental or Ethical Impacts

There is currently no industry standard for detecting Trojans within an IC in real-time. The plan is to develop a versatile algorithm that can be applied across multiple ICs that will allow for real-time Trojan detection. The simulation data will provide conclusive data on how companies can detect foreign entities with their ICs. Companies can protect their ICs to prevent any third party from re-creating their ICs and stealing their intellectual property.

Independent workers will also be able to protect their original ideas to ensure that it is not taken by bigger companies. The goal is to protect circuit developers everywhere, by preventing Trojans from unethically stealing IPs.

Since the bulk of the project is simulation currently, there is no direct major environmental impacts. However, the process of producing an IC impacts the environment greatly. A facility producing 6 inch wafers, another term for an IC, will use roughly 5 million gallons of water a day. Since production plants are usually located in dry areas of the country, such as New Mexico, Arizona, and California, the production of the IC drains an already scarce natural resource in those areas. Breaking or tampering with ICs due to Trojans causes another issue, as those resources are then wasted, and cannot be replaced.

Ethical Impacts

Technology is rapidly growing every day. Thus, we trust these technologies with more personal information. Protecting that information becomes crucial in maintaining our daily lives. The Internet of Things (IoT) connect more devices every day and predicts that our world will have 24 billion IoTs devices by 2020. [15] A survey consisting of 5,000 enterprises was conducted by AT&T's Cybersecurity Insights Report. It showed that 85% of enterprises are in the process of or intend to deploy IoT devices however, only 10% feel like they could secure those devices against hackers.

With this growth comes many benefits. For example, smart cars will drive themselves, homes will have artificial intelligence, and clothes will have embedded systems. While these advances are great, they introduce more entry points for hackers and cybercriminals. Currently, we see these entry points only in software, but as we begin to see more IoT devices, we will begin to see more hardware attacks. In 2010, for example, Dell released some PowerEdge R410 server motherboards with computer malware on the board itself. The malware was found on the embedded server management firmware. Dell had to send Dell engineers to remove the malware off the infected IC. The companies supplying these ICs are liable for any kind of malware found

on their intellectual property and, therefore, a real-time Trojan detection algorithm will have a positive ethical impact on the consumer and the company.

Our goal is to look at hardware Trojans in the manufacturing process of ICs. We hope to not only protect the personal information of the user, but also the intellectual property itself. With the creation of our real-time Trojan detection, we hope to protect the IP of anyone who looks to create ICs.

10. Summary/Conclusions

We have determined the scope of the project and how much time needed to be allocated for each stage. In our first stage, the fall quarter of 2016, the group needed to be familiar with the software used for the simulation of the integrated circuit. The ASIC design software, Synopsys, and the VLSI design software, Virtuoso, was decided to be the software to be used for the simulation of the IC. Synopsys allows us to create netlist code from synthesis. Using HDL code, which all of us have experience in, Synopsys can create a netlist for synthesis, and the output created can be imported into Virtuoso.

First, everyone needed to set up accounts on Xunil and learn the software on there. Second, we synthesized a core with Synopsys, which allowed the netlist to be developed. Third, we began learning Virtuoso; this will help us verify the netlist which we generated from Synopsys. Lastly, we are in the process of determining how to import the netlist into Virtuoso. Once we can import the netlist into Virtuoso, we will have both programs working with one another. This will allow us to begin our next phase in term 2, which is to insert a Trojan into the IC and test for detection. We will find the resolution of the Trojan that can be detected which will help us determine how small the Trojans can be before it cannot be detected.

11. References

[1]C. Florian, "Report: The Most Vulnerable Operating Systems and Applications in 2012", *GFI Blog*, 2013. [Online]. Available: http://www.gfi.com/blog/report-the-most-vulnerable-operating-systems-and-applications-in-2012/. [Accessed: 17- Nov- 2016].

[2]C. Florian, "The Most Vulnerable Operating Systems and Applications in 2011", *GFI Blog*, 2012. [Online]. Available: http://www.gfi.com/blog/the-most-vulnerable-operating-systems-and-applications-in-2011/. [Accessed: 17- Nov- 2016].

[3]C. Florian, "Report: Most vulnerable operating systems and applications in 2013", *GFI Blog*, 2014. [Online]. Available: http://www.gfi.com/blog/report-most-vulnerable-operating-systems-and-applications-in-2013/. [Accessed: 17- Nov- 2016].

[4]C. Florian, "Most vulnerable operating systems and applications in 2014", *GFI Blog*, 2015. [Online]. Available: http://www.gfi.com/blog/most-vulnerable-operating-systems-and-applications-in-2014/. [Accessed: 17- Nov- 2016].

[5]J. Vijayan, "Security researchers create undetectable hardware trojans", *Computerworld*, 2013. [Online]. Available: http://www.computerworld.com/article/2485155/computer-processors/security-researchers-create-undetectable-hardware-trojans.html. [Accessed: 17- Nov- 2016].

[6]A. Meola, "How the Internet of Things will affect security & privacy", *Business Insider*, 2016. [Online]. Available: http://www.businessinsider.com/internet-of-things-security-privacy-2016-8. [Accessed: 17- Nov- 2016].

[7]C. Sechen, "Design Vision", *Utdallas.edu*, 2015. [Online]. Available: http://www.utdallas.edu/~hxh025000/index_files/design_vision.htm. [Accessed: 18- Nov- 2016].

[8]R. Reese, "Synopsys", *www.my.ece.msstate.edu*, 2003. [Online]. Available: http://my.ece.msstate.edu/faculty/reese/EE8273/lectures/synopsys/synopsys.pdf. [Accessed: 17- Nov- 2016].

[9]G. Becker, F. Regazzoni, C. Paar and W. Burleson, "Stealthy Dopant-Level Hardware Trojans", *www.sharps.org*. [Online]. Available: http://sharps.org/wp-content/uploads/BECKER-CHES.pdf. [Accessed: 17- Nov- 2016].

[10]T. Chen, "Training Course of Design Compiler", *www.ee.ncu.edu.tw*, 2011. [Online]. Available: http://www.ee.ncu.edu.tw/~jfli/vlsi21/lecture/dc.pdf. [Accessed: 17- Nov- 2016].

[11]Y. Gwon, H. Kung, D. Vlah, K. Huang and Y. Tsai, "Statistical Screening for IC Trojan Detection", 2012. [Online]. Available: http://www.eecs.harvard.edu/~htk/publication/2012-iscas-gwon-kung-vlah-huang-tsai.pdf. [Accessed: 17- Nov- 2016].

[12]A. Iqbal, "Understanding Integrated Circuit Security Threats", *www.sdm.mit.edu*, 2011. [Online]. Available: https://sdm.mit.edu/news/news_articles/webinar_021014/iqbal_021014.pdf. [Accessed: 17- Nov- 2016].

[13]D. Lockhart, "RTL-to-Gates Synthesis using Synopsys Design Compiler", *www.csl.cornell.edu*, 2016. [Online]. Available: http://www.csl.cornell.edu/courses/ece5745/handouts/ece5745-tut2-dc.pdf. [Accessed: 17- Nov- 2016].

[14]"Computer-Aided Design of ASICs Concept to Silicon", *www.eng.auburn.edu*, 2011. [Online]. Available: http://www.eng.auburn.edu/~agrawvd/COURSE/E7950_Fall11/TALKS/ASIC%20CAD%20Seminar%202011.pdf. [Accessed: 17- Nov- 2016].

[15]"Dell warns of hardware trojan", *Homeland Security News Wire*, 2010. [Online]. Available: http://www.homelandsecuritynewswire.com/dell-warns-hardware-trojan. [Accessed: 17- Nov- 2016].

[16]M. Tehranipoor and F. Koushanfar, "A Survey of Hardware Trojan Taxonomy and Detection", *www.trust-hub.org*, 2010. [Online]. Available: https://www.trust-hub.org/publications/P1.pdf. [Accessed: 17- Nov- 2016].

Appendix A: Design Constraints Summary

Team Number: ECE-46
Project Title: **IC Level Trojan Detection**

Summary of the Design Aspects:

The IC Level Hardware Trojan detection, will be able a system to detect foreign circuitry on any type of IC, through certain side-channels of data (power, heat, and timing). The design part of the project will consist of synthesizing a IC using ASIC design and creating a Trojan to be synthesized with it. The software pages used will be Cadence Virtuoso for VLSI and Synopsis for ASIC. In parallel to this an algorithm or a mathematical system will be developed in order to detect the Trojan using the data collected via the side-channels.

Design Constraints:

Economic: The group aims to design a system that will help companies and private individuals protect their Intellectual Property. This will help keep the large amount of wealth that the IP might bring to the company secure while in development.

Manufacturability: Since this project is part of a larger research goal, the end goal of the research project itself will be to manufacture or produce either a piece of software or sensor system, which, if this project is a success, will be feasible.

Sustainability: Since the project is heavily software oriented, and the need for this sort of detection system is needed this is sustainable.

Environmental: The group aims to protect the corruption of ICs. As ICs are better protected, Trojans will have a harder time corrupting cores, therefore preventing the materials used to create the ICs from being wasted. As mentioned previously, a number natural resources, such as water, silicon, and many others, are used to create ICs. By preventing Trojan corruption, we also prevent the waste of the Earth's natural resources.

Ethical, health, and safety: Knowing Trojans can leak intellectual property, it was important to understand the ethics behind stolen intellectual property. Whether it be big business, or even an independent engineer, the IC Hardware Trojan detection system aims to protect the hardware both group's intellectual property.

Standards and Regulations

[1]"AMBA Open Specifications - ARM", *Arm.com*, 2016. [Online]. Available: https://www.arm.com/products/amba-open-specifications.php. [Accessed: 18- Nov- 2016].

Appendix B: Resumes