Security and Trust in Integrated Circuits

Student Name: Vasisht Duddu
Roll Number: 2015137

BTP report submitted in partial fulfillment of the requirements
for the Degree of B.Tech. in Electronics and Communication & Engineering
on $16^{th} November 2017$

BTP Track: Research

BTP Advisor
Dr Mohammad Hashmi
Dr Donghoon Chang

Indraprastha Institute of Information Technology
New Delhi

Student's Declaration

I hereby declare that the work presented in the report entitled **Security and Trust in Integrated Circuits** submitted by me for the partial fulfillment of the requirements for the degree of *Bachelor of Technology* in *Electronics and Communication & Engineering* at Indraprastha Institute of Information Technology, Delhi, is an authentic record of my work carried out under guidance of **Dr Mohammad Hashmi & Dr Donghoon Chang**. Due acknowledgements have been given in the report to all material used. This work has not been submitted anywhere else for the reward of any other degree.

..... **Place & Date: .....**

(Vasisht Duddu)

Certificate

This is to certify that the thesis titled "Security and Trust in Integrated Circuits" being submitted by Vasisht Duddu to the Indraprastha Institute of Information Technology Delhi, for the award of the Bachelor of Technology, is carried out by him under my supervision. The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

..... **Place & Date: .....**
(Dr. Mohammad Hashmi)

.....
(Dr. Donghoon Chang)

Abstract

The increasing globalization of Integrated Circuits(ICs) supply chain has reduced the control of the vendor over the design and fabrication process which increases the possible threats by adversaries to exploit vulnerabilities and compromise the hardware. The fabrication of ICs is a multi-step process spread across various parts of the world for commercial reasons. In such situation, hardware root of trust,i.e, the assumption that the hardware is secure against all possible attacks is violated. Design and study of trusted integrated circuit design and reliable circuits are therefore crucial, especially, when the circuits are being used for critical applications like military and critical national infrastructure. This work studies the threat model of Integrated circuits supply chain and study various attacks and then explores topics like Hardware Trojans and physically unclonable functions by implementing a counter based asynchronous trojan and an analog unclonable function.

Keywords: Hardware Security, Trust, Hardware Trojans, PUFs

Acknowledgments

I would like to express my sincere gratitude to my advisers Dr. Mohammad Hashmi and Dr. Donghoon Chang for providing excellent guidance and being supportive throughout the span of this thesis. They helped me in every step to learn, explore and conduct research. Without their patience, critiques and thoughtful insights this work would never have been completed. I am thankful for their time and efforts for guiding me in my thesis.

Contents

Chapter 1

Introduction

Due to globalization of semiconductor industry and distributed and multi-step nature of IC supply chain has introduced a large number of vulnerabilities which can be exploited by adversaries. Initially, the design house and the vendor had the complete control of all the processes of the IC supply chain. However, due to economic reasons, industries started to outsource different stages of IC production to smaller companies spread across the globe. This has reduced the control over the IC supply chain resulting in trust issues in the IC designs and manufacturing. While designing IC, cost, power, performance and reliability are the key metrics that are considered and security and trust and security of these circuits is an after thought.

A study found that the companies to which the tasks were outsourced to, were procuring the components further from untrusted sources in foreign countries and buying from online auctions and re-sellers. There have been instances where used components were being used.

Security engineers work on the software abstraction level and most of the cryptographic algorithms being executed on hardware assume that the hardware is resilient to all types of attacks and is secure(hardware root of trust).The system is compromised if hardware is not secure even though the software may be secure.

There have been many instances where the circuits used for military systems were found to have reliability and security issues. Having a separate secure supply chain is economically not feasible and hence the government has to outsource chip design to third party institutes. It is hence important to consider the security aspect of the hardware.

1.0.1 Threat Model

In this section, we explore the possible vulnerabilities and attacks vectors in IC supply chain. The IC supply chain is the series of steps that are required for design and fabrication of the integrated circuit. Figure 1.1 shows different steps of IC supply chain and their associated trust.

The IC supply chain can be broadly broken into the following:

Specification Phase: The designers decide on the high level description of the building blocks and define the system's characteristics including power consumption, area, delay and function-

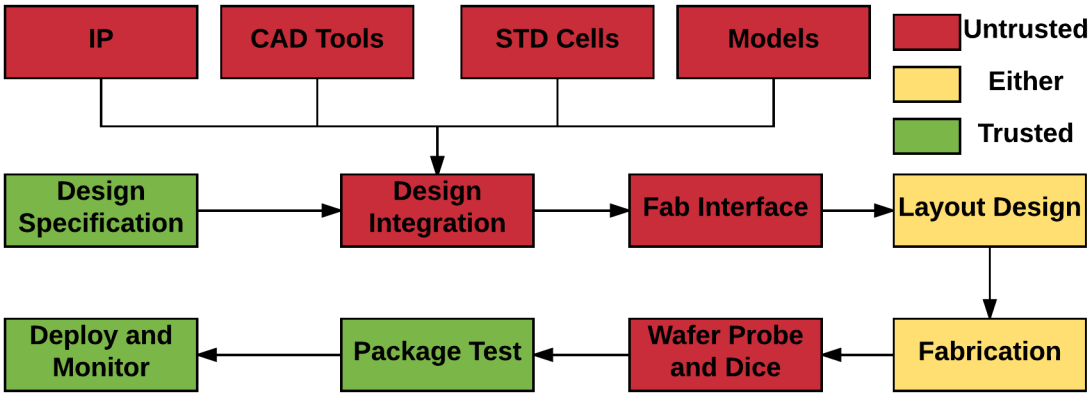

Figure 1.1: IC Supply Chain in Adversarial Settings[Chakroborty *et al*]

ality.

Design Phase: This is the most vulnerable phase where designers consider functional, logical, timing and physical constraints and they map the schematic to the hardware. This includes using multiple 3rd party IP cores, standard cells and EDA tools which might be vulnerable and contain trojans.

Fabrication Phase: Designers create the layout/mask and send it to the fab to create the chip. Attacker might make small modifications to the mask or change chemical composition to effect the current flow decreasing the reliability of the circuit.

Assembly: At this stage, different components are assembled on a PCB and every junction where two components are connected is a possible site for trojan insertion. Even if all individual chips are trustworthy, malicious assembly might create security issues in the circuit. The usage of unshielded wires to leak information via side channels.

Testing Phase: This is the phase for detection of any vulnerabilities introduced in the circuits which includes using trojan detection techniques and test vectors that have to kept secret to prevent attacker from effecting the testing phase. This phase is considered as trustworthy to detect any modification made by attacker to the circuit in the previous stages.

1.0.2 Threats

We now study various threats and attacks possible on integrated circuits by the adversaries at different stages of IC supply chain(Figure 1.2). Following are the most common threats on Integrated Circuits:

Hardware Trojan: An attacker either in the design house or foundry may add malicious circuitry or modify existing circuits to effect the reliability of circuit or run some malicious logic.

Intellectual Property(IP) Piracy or overbuilding: An IP user or rogue foundry may pirate the IP without the knowledge or consent of the designer leading to IP Theft. A malicious

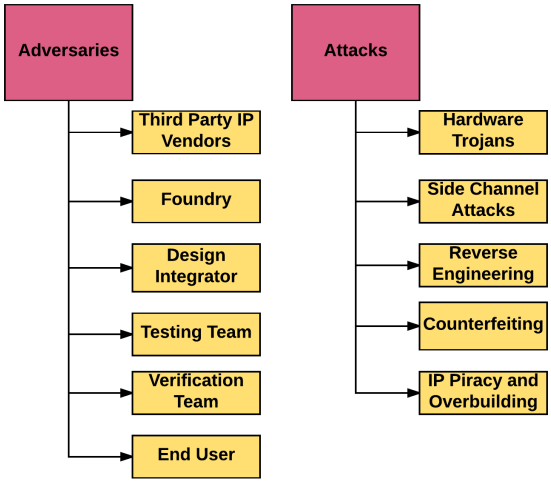

Figure 1.2: Threats to IC Supply Chain

foundry may build more than required number of ICs and sell the excess components to the grey market.

Reverse Engineering: The attacker traces back the IC to some abstraction level in the IC supply chain. The attacker may use Reverse engineering to reach a desired abstraction level and then modify the IP or change it.

Side Channel Analysis: Circuits while executing some code or performing cryptographic operations radiates EM waves or information from other channels. An attacker can extract information from these side channels and process them to get secret information like cryptographic keys.

Counterfeit: Attacker forges original component y reverse engineering the IC to certain abstraction level and rebuild these circuits using other components. Systems using these circuits might effect the reliability and security.

1.0.3 Attacker Motives

In this section we enumerate the possible attacker motives to attack and compromise ICs. A few common motives for the attacker could be:

- Leak information
- Steal Data
- Counterfeit/Duplicate circuit
- Reduce the reliability of the system
- Denial of Service

- Maintain and monitor system activities

1.0.4 Attacker Knowledge

Attack on integrated circuits requires a lot of knowledge of the tools, circuit design and the intricacies of the chip. The attacker will be able to perform an attack successfully if he has the required knowledge.

The attacker may have complete Knowledge of the system using which he may reverse engineer the circuit, counterfeit the circuit design, leak secret information like cryptographic keys and insert trojan to maintain backdoor access to the system.

In case the attacker has partial knowledge he is required to alleviate knowledge using side channel and Reverse Engineering and then perform other attacks like insert trojans for future access and control of system.

Chapter 2

Hardware Trojans

Trustworthy circuits exhibit the functionality it was designed for, no more no less. They conceal information about the computation being performed from side channels and their design should be transparent only to designers and opaque to others, such that attacker should not know anything about the design and internals. On the other hand, untrusted ICs fail to deliver certain required functionality effecting the reliability and/or have hardware trojans inside chip/systems effecting the security of the system.

Hardware trojans are any malicious addition or modification to a circuit or system. There are two main characteristics of hardware trojans:

Malicious Goals: This includes change or control functionality, leak sensitive information and reduce circuit reliability

Intentional addition or modification

HTs are very hard to detect as the opaqueness of circuit internals reduces observability, preventing use of some common types of detection methods. Secondly, the technology scaling in semiconductor makes it difficult to differentiate between variation and hardware trojan behavior. Thirdly, there is a large uncharacteristic space for the possible Trojan insertion making it difficult to detect the trojans.

2.0.1 Trojan Taxonomy

Many taxonamies have been proposed to classify the trojans. But we will consider the classification given by Banga et al[8] and will design trojans based on the classification on trigger and payload.

Te trojans can be classified based on the following classification:

IC Supply Chain: This is based on where the HT is inserted in the supply chain which has been discussed in Chapter 1.

Abstraction level: This classification is based on where the HT is embedded in the design

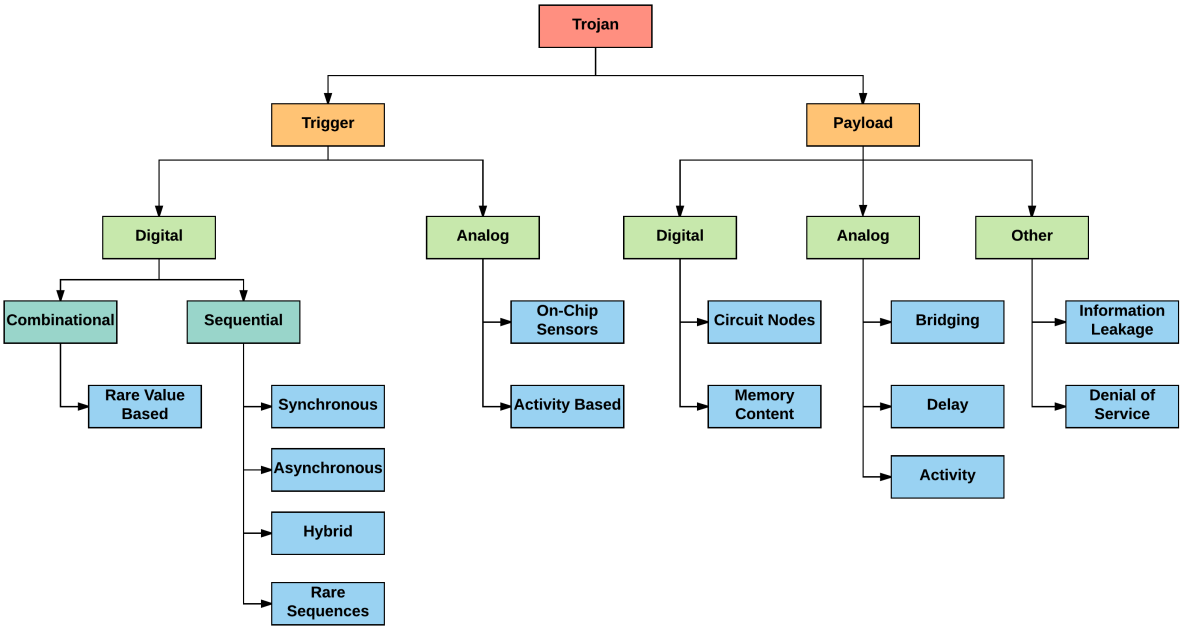

Figure 2.1: Trojan Classification based on Trigger and Payload[Karri *et al.*]

phase abstraction. Following are the possible phases and corresponding trojans that can be inserted:

- Architectural or RTL Level: Attacker can make changes in the Verilog or VHDL code to modify the functionality.
- Functional level: Attacker can make modification to gate level schematics or sum equation or product function.
- Logic synthesis/gate level: Attacker can include a kill switch to disable a circuit using a control signal and modifying the gate to include a control signal.
- Circuit or Transistor level: This includes changing the threshold voltage or transistor to effect delay or power consumption
- Physical design or Layout Level: Dimensions or location of components can be changed to include HT.

Activation or Trigger Mechanism: This classification is on how the trojan will be activated and can be further broken into combinational trojan where the activation depends on certain rare condition in circuit and sequential trojan where activation depends on sequence of rare values in circuit. It is more difficult to trigger sequential as a particular sequence of rare values are required and test vectors may not cover them during the testing phase.

The trojans can also be classified as synchronous which uses clock to count cycles(time bombs) and trigger at some instant or asynchronous trojan which uses output of some other logic to

count the cycles.

Effects or Payload: This is on the basis of the result or payload of the trojan after being activated which includes DoS, changing memory content, effecting the parameters of circuit like performance, power and noise margin.

Location: This is based on where on-chip is HT located. The trojan may be located in the processor(change execution and functionality), memory(alter memory contents), IO devices(control/modify/modify data communication), power supply units(change power/current to reduce the reliability) and clock grid(change freq. to cause fault or failure)

Type: Trojans can be further classified into parametric or functional where parametric trojans aim to reduce the reliability of the system to increase the chance of performance failure. This includes thinning wires, weakening logic gates, modifying the power supply or changing the transistor size and properties. On the other hand, functional trojans includes addition/deletion of gates to effect the system's functionality.

2.0.2 Detection Methods

The goal hardware trojan detection methods is to determine if there is a HT or not and if the IC can be trusted. If trojan is found this implies that the IC is untrusted but this does not imply design team is untrusted as the trojan may have been from 3rd party IP or other vendors. However, if no trojan is detected, this does not imply that the IC is trusted or design team is trusted. If a method does not detect a trojan it does not mean that the IC does not have a trojan as the trojan may be designed to circumvent the detection method.

Trojan detection methods can be broadly classified as:

Destructive: This method uses one sample or set of sample ICs and reverse engineer each IC to reveal inner working and structure and find malicious modifications. It is not a practical approach as it requires tools, equipment, knowledge and time. Also, a sample may not have a trojan however, me other IC may have the trojan.

Non-Destructive:

Non-Destructive detection techniques can be broken as:

- Test Time Techniques

  These techniques are used for trojan detection after the circuits have been designed and fabricated and during the testing phase by a trusted verification team.

  Logic Test based HT detection This includes running different test vectors and monitoring the circuit's output behavior. If HT is triggered, its malicious behaviour will be observed which can be detected. However, for big circuits the input size is huge and full coverage test is impractical. Random test will fail as HT are triggered by rare test vectors which may not be selected in random test.

Side Channel Analysis Based HT Detection:

This method monitors the side channel information during execution at test time. The presence of HT on chip will show on some physical parameters and can be observed through certain side channels. However, this method may give high false positives.

- Run Time Techniques

  These techniques monitor the execution at real time and observe malicious behavior. The stop execution once HT is detected to protect system. It is a good complementary approach to test time approach which may not be 100% effective. It however takes extra resources and more performance overhead.

2.0.3 Metrics

Various metrics are used to analyze the trustworthiness of an integrated circuit based on false positives, time required to detect trojan and probability of a detection method to find the trojan. Following are the commonly used metrics compute efficiency of various defences:

Probability of Detection: It is defined as ratio of the number of trojans detected by the technique to the total number of trojans in the design.

Prob. of false alarm: It is defined as the the ratio of number of trojan free designs that are incorrectly classified as trojan to the number of trojan free design.

Time required to detect trojans: It is reported in terms of number of applied test vector in foundry. In 3rd party IPs they are reported in terms of required clock cycles.

2.0.4 HT Prevention

Hardware Trojans can be prevented by employing trusted circuit design techniques. These techniques can be broken down into pre-synthesis techniques and post-synthesis techniques.

Pre-Synthesis Techniques:

The aim is to ensure IPs are trusted before using them. This is done by changing the problem into a formal verification problem and perform property/model/equivalence checking. For sequential components, one can use FSM equivalence check using product machines.

Post-Synthesis Techniques

These techniques removes the dead spaces in the IC to prevent direct HT insertion which limits the insertion of big and sophisticated HTs. The empty spaces in the Integrated Circuits are potential locations for inserting HTs. One method to remove the dead space is by adding dummy logic and circuits.

Circuit obfuscation is another method used to make reverse engineering and side channel analysis by EM radiation harder to understand and analyze.

Interface protection can be used to prevent HTs that depend on a trigger signal(IO pins, internal module interface, scan chain,power/clock) to activate. The idea is to monitor the interface and whenever there is a suspicious communication we can verify and intercept to prevent the trojan from being triggered.

Another method is to establish hardware root of trust and chain of trust for IP and EDA tools to ensure that there is no hardware trojan present in the third party software and IPs.

Salmani *et al.*[17] studied the probability of getting 1 and 0 as output in a combinational circuit and based on the probability, they reduced the rare event occurrence by introducing the scan flip flop to make reduce the probability of 1 and 0 at output and make the rare events less rare preventing the triggering of trojan.

Li *et al.*[18] designed a path delay based technique. HT cannot be detected, if we cannot measure internal delay due to opaque nature of circuit to attacker. Add a shadow register which has the same clock as system but with different phase can result in storing the same value in the shadow register as in the output register of the circuit. But when the phase of the clock changes the we will get a difference in the result which can help us to increase the visibility into the circuit. This is used to find the path delay and hence help towards HT detection.

Banga *et al.*[19] reverse the logical gates' power supply and ground which enabled rare events turning into a frequently occurring events which can further facilitate HT detection.

Gu *et al.*[20] proposed a method to change system specification deliberately. If a HT is inserted, it will change system's performance dramatically which can be observed.

Chakraborty *et al.*[21] used circuit obfuscation techniques to make it much harder to understand circuit internals and hence increase the complexity of trojan insertion.

Love *et al.*[22] proposed a technique that uses proof carrying hardware to build IP and use them as building block to develop trustworthy circuits.

Dunbar *et al.*[23] build a trustworthy sequential circuits by adding control signals to each basic building block (flip flops).

Chapter 3

Design and Analysis of Trojan

The trojan design mechanism has two aspects: trigger and output payload. The trojan should be triggered under rare events or conditions and should not be easily detected by testing.

In this work, we work on a Trojan design based on Asynchronous Counter. The design uses a 128 bit up down counter where the payload is to modify the output bit. The design schematic is given below.

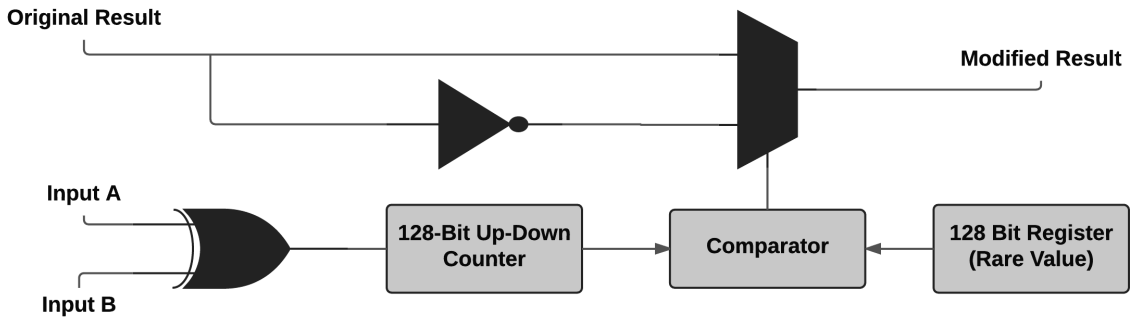

Figure 3.1: Trojan Design

Verilog Code for Trojan

```
module trojan(
input original_result,
input a,
input b,
input clk,
input reset,
output modified_result
);
wire mux_in, up_down, select_line;
wire [127:0]count;
not not1(mux_in,original_result);
xor xor1(up_down,a,b);

up_down_counter counter1(count,up_down,clk,reset);
comparator comparator1(select_line,count);
mux2_1 mux1(modified_result,original_result,mux_in,select_line);

endmodule
```

Verilog Code for 128-bit Comparator

```
module comparator (
input wire [127:0] a,
output reg equal
);
always @* begin
if (a==128'd8) begin
equal = 1;
end
else begin
equal = 0;
end
end

endmodule
```

Verilog Code for 128-bit Counter

```
module up_down_counter (
out ,
up_down ,
clk ,
reset
);

output [127:0] out;
input up_down, clk, reset;

reg [127:0] out;

always @(posedge clk)
if (reset) begin
out ¡= 128'b0 ;
end else if (up_down) begin
out ¡= out + 1;
end else begin
out ¡= out - 1;
end

endmodule
```

Verilog Code for 2 by 1 Multiplexer

```
module mux2_1(q, d1, d2, select);
        input d1, d2, select ;
        output q;
        wire q;
        assign q = (select) ? d2 : d1;
endmodule
```

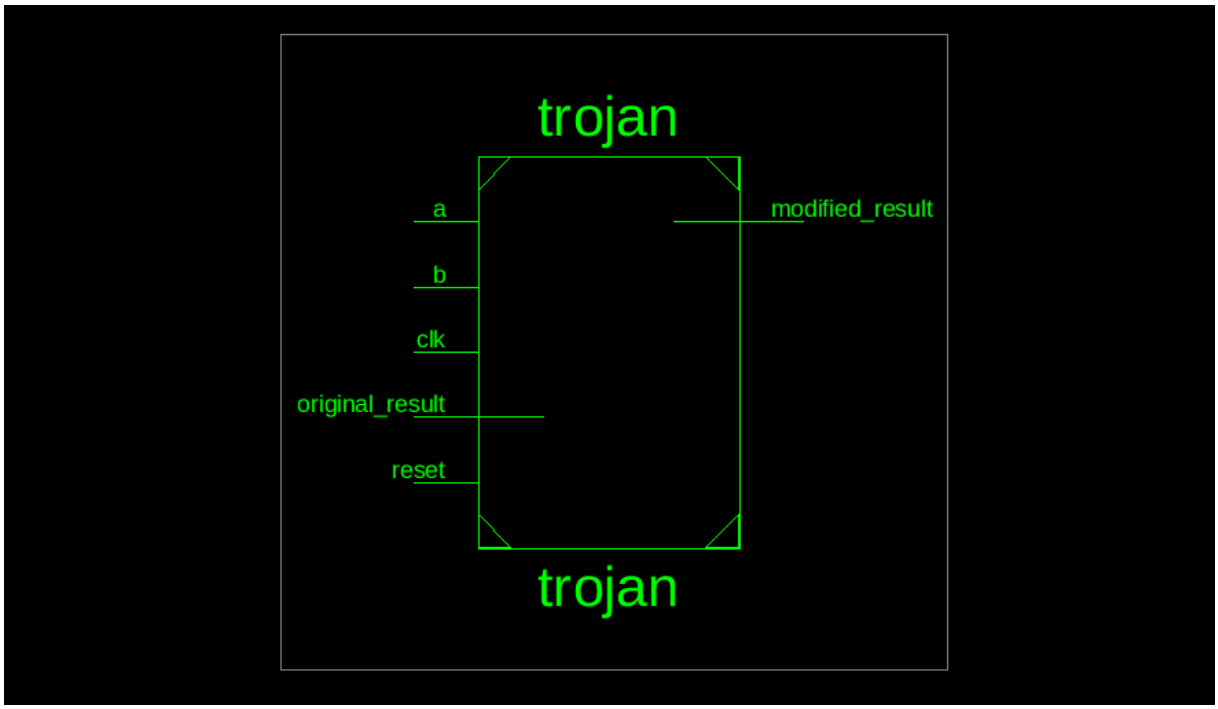

Figure 3.2: RTL Schematic

Chapter 4

Conclusion

This study explored different possible threats to Integrated Circuits and we specifically studied hardware trojan design and implementation. Hardware trojans present a very real threat and requires constant proactive approach to address this issue. Most defences are ad-hoc or can be used for particular attack case and there is no silver bullet solution to protect hardware from such attacks.

The major challenge is to develop detection mechanisms for sequential and analog trojans as they are triggered by a wide possible condition and compare them with hardware trust benchmark by quantifying and identifying trustworthy circuits. Most mechanisms rely on presence of golden ICs or trojan free ICs which may not always be available and other techniques to design trustworthy circuits without relying on golden ICs has to be explored.

As part of future work, I would explore and design an analog physically unclonable function(PUF) as a hardware security primitive to improve the security and authenticate circuits. The study would explore various PUF architectures and study the recent trend in PUF design and implement an analog PUF using cadence(virtuoso)/spice and perform simulations.

Bibliography

[1] Ramesh Karri and Jeyavijayan Rajendran, Kurt Rosenfeld, Mohammad Tehranipoor, Trustworthy Hardware: Identifying And Classifying Hardware Trojans, IEEE Design & Test of Computers, 2010

[2] Masoud Rostami, Farinaz Koushanfar, and Ramesh Karri, A Primer on Hardware Security: Models, Methods, and Metrics, Vol. 102, No. 8, August 2014 — Proceedings of the IEEE

[3] Swarup Bhunia, Michael S. Hsiao, Mainak Banga, and Seetharam Narasimhan, Hardware Trojan Attacks: Threat Analysis and Countermeasures, Vol. 102, No. 8, August 2014 — Proceedings of the IEEE

[4] Mohammad Tehranipoor, Farinaz Koushanfar, A Survey of Hardware Trojan Taxonomy and Detection, IEEE Design & Test of Computers

[5] Seetharam Narasimhan, Xinmu Wang, and Swarup Bhunia, Wen Yueh and Saibal Mukhopadhyay, Improving IC Security Against Trojan Attacks Through Integration of Security Monitors

[6] Miodrag Potkonjak, Ani Nahapetian, Michael Nelson, Tammara Massey, Hardware Trojan Horse Detection Using Gate-Level Characterization, DAC09, July 26-31, 2009, San Francisco, California, USA

[7] Rajat Subhra Chakraborty, Seetharam Narasimhan and Swarup Bhunia, Hardware Trojan: Threats and Emerging Solutions, IEEE 2009

[8] Mainak Banga and Michael S. Hsiao, Trusted RTL: Trojan Detection Methodology in Pre-Silicon Designs, IEEE 2010

[9] Huafeng Liu, Hongwei Luo, Liwei Wang, Design of Hardware Trojan Horse Based on Counter, IEEE 2011

[10] Jie Zhang and Qiang Xu, On Hardware Trojan Design and Implementation at Register-Transfer Level, 2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)

[11] Yier Jin and Yiorgos Makris, Hardware Trojans in Wireless Cryptographic ICs, IEEE Design & Test of Computers

[12] Swarup Bhunia, Michael S. Hsiao, Virginia Tech, Jim Plusquellic, Mohammad Tehranipoor, Miron Abramovici, Dakshi Agrawal, Paul Bradley, Protection Against Hardware Trojan Attacks:

Towards a Comprehensive Solution, IEEE Design & Test 2012

[13] K. XIAO, D. FORTE, Y. JIN, R. KARRI, S. BHUNIA and M. TEHRANIPOOR, Hardware Trojans: Lessons Learned after One Decade of Research, ACM Transactions on Design Automation of Electronic Systems, 2016

[14] Samantha Pham, Jennifer L. Dworak, and Theodore W. Manikas, An Analysis of Differences between Trojans inserted at RTL and at Manufacturing with Implications for their Detectability

[15] Dakshi Agrawal Selcuk Baktr, Deniz Karakoyunlu, Pankaj Rohatgi, Berk Sunar, Trojan Detection using IC Fingerprinting

[16] Nicole Fern, Shrikant Kulkarni and Kwang-Ting (Tim) Cheng, Hardware Trojans Hidden in RTL Dont Cares - Automated Insertion and Prevention Methodologies, INTERNATIONAL TEST CONFERENCE, 2015

[17] H. Salmani, M. Tehranipoor, J. Plusquellic, New Design for Imroving hardware trojan detection and reducing Trojan activation time, HOST 2009

[18] J.Li and J. Lach, At speed delay characterization for IC authentication and hardware trojan Horse detection, HOST 2008

[19] M. Banga, M. Hsiao, VITAMIN: voltage inversion technique to ascertain malicious insertions in ICs, HOST 2009

[20] J. Gu, G. Qu and Q. Zhou, Information Hiding for trusted system design, DAC 2009

[21] RS Chakraborty and S. Bhunia, Security against hardware trojan through a novel application of design obfuscation, ICCADD 2009

[22] E. Love, Y. Jin, Y. Makris, Enhancing security via provably trustworthy hardware intellectual property, HOST 2011

[23] C. Dunbar, G. Qu, Designing trusted embedded system fro finite state machine, TECS 2014

[24] Nicole Fern, Shrikant Kulkarni, Kwang-Ting TIm Cheng, Hardware Trojans Hidden in RTL Don't Cares- Automated Insertion and Prevention Methodologies, International Test Conference, 2015

[25] Jie Zhang, Qiang Xu, On Hardware Trojan Design and Implementation at Register-Transfer Level