A Thesis

entitled

Effects of Temporal Variations on Delay based Physical Unclonable Functions

by

Chayanika Roy Chaudhuri

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the

Master of Science Degree in

Electrical Engineering

_________________________________________
Dr. Mohammed Niamat, Committee Chair

_________________________________________
Dr. Mansoor Alam, Committee Member

_________________________________________
Dr. Junghwan Kim, Committee Member

_________________________________________
Dr. Amanda Bryant-Friedrich, Dean
College of Graduate Studies

The University of Toledo

December 2016
An Abstract of

Effects of Temporal Variations on Delay based Physical Unclonable Functions

by

Chayanika Roy Chaudhuri

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the
Master of Science Degree in
Electrical Engineering

The University of Toledo

December 2016

Counterfeiting of electronic goods is a growing concern with a huge impact on
the global economy and the security of critical infrastructure. One of the difficulties to
deal with counterfeiting stems from the fact that counterfeit goods can originate from
sources that are able to make copies in a way that it is hard to distinguish the copies from
their legitimate counterparts. Therefore, there is a pressing need to facilitate rapid growth
of hardware based security. Hardware based cryptographic protocols and authentication
of integrated circuits play an important role in the field of hardware security.
Conventional security methods rely heavily on digital storage of secret keys in non-
volatile memory which is vulnerable to side channel attacks and reverse engineering.
Physical Unclonable Functions (PUFs) leverage the inherent process variations during the
manufacturing process to create unique IDs and secret keys. For this reason, an untrusted
foundry cannot make a copy of the circuit since it is impossible to control the
manufacturing process variations.

With the downscaling of feature sizes in the integrated circuits, the variation of
the device parameters has become an area of concern. During the manufacturing process,
variations of process parameters are unavoidable. After manufacturing the circuit, the performance degrades over time owing to aging effects that trigger a drift of device parameters.

The delay of an Integrated Circuit is a function of the ambient temperature. While the effect somewhat depends on the semiconductor material as well, but in general, higher the operating temperature of a digital circuit, greater is the propagation delay through interconnects. Thus for all delay PUFs, it is critical to ensure the output is stable across a wide operational range. Compared to the measurements at room-temperature, the error rate of a typical PUF will increase by a small percentage when the temperature is varied over a wide operational range from -40 °C to +85 °C. Variation in supply voltage is another factor that is commonly taken into consideration. It is desirable that a design will exhibit similar behavior in the face of any environmental variation. In other words, PUFs should function reliably even in presence of voltage and temperature variations.

In this work, we study the effects of aging, temperature, and voltage variation on configurable Ring Oscillator (RO) PUF which is a delay based PUF. It is observed that the uniqueness of the RO-PUF is not affected to a great extent, where the inter-chip HD is found to be 45.9% which is closer to the ideal value of 50%. The intra-chip HD of PUF degrades with aging which makes the PUF responses unreliable. Also, the ROs are found to be sensitive to temperature and voltage variations. The bit flips do occur due to temperature and voltage variations but only when the frequency difference in the RO comparison pair is 1.5 MHz.
To my late father Kishalay Roy Chaudhuri, for his unconditional love, trust and encouragement; who believed in me and made me what I am today.
Acknowledgements

I would like to take this opportunity to express my gratitude to those who helped me with the various facets of research and the writing of this thesis. First and foremost, I must mention my advisor, Dr. Mohammed Niamat, for his precious guidance, support, and knowledge that helped me accomplish my research goals. My experience in the High Performance Computing Systems Lab has been instrumental in shaping my research and teamwork skills. I am thankful to my advisor for giving me much independence in my work and for trusting me when I made my own decisions concerning the direction of my thesis. I am also grateful to him for granting me ample time off to grieve a personal loss during the final semester of my Master’s program.

I would like to thank my committee members Dr. Mansoor Alam and Dr. Junghwan Kim for spending their valuable time to read my thesis and to provide me with their input. A special appreciation goes to Dr. Mansoor Alam and the Department of Electrical Engineering and Computer Science for considering me for financial support. Last but not least, I am appreciative of the professors who taught me in class, also, I thank my entire family, friends and colleagues for their continued encouragement, support, motivation, and help during my tenure as a graduate student at The University of Toledo.

All the FPGAs and test equipment used for this thesis was made available by the High Performance Computing Systems Laboratory at The University of Toledo.
# Table of Contents

Acknowledgements ........................................................................................................... vi

Table of Contents ............................................................................................................. vii

List of Tables ................................................................................................................... x

List of Figures .................................................................................................................. xi

List of Abbreviations ....................................................................................................... xiii

List of Notations ............................................................................................................... xiv

Chapter 1 ......................................................................................................................... 1

Introduction ....................................................................................................................... 1

1.1 Motivation .................................................................................................................... 1

1.2 Organization ................................................................................................................ 8

Chapter 2 ......................................................................................................................... 10

Background ....................................................................................................................... 10

2.1 Hardware Security ...................................................................................................... 10

2.2 PUF: Definition .......................................................................................................... 11

2.2.1 Challenge-Response Pair ...................................................................................... 13

2.3 Applications of PUFs ............................................................................................... 14

2.3.1 Signature Generation ............................................................................................ 15

2.3.2 Cryptography ........................................................................................................ 16

2.3.3 Random Number Generator ................................................................................. 16
2.4 Sources of Variations ................................................................. 16
2.5 Noise ......................................................................................... 19
2.6 Types of PUF ............................................................................. 20
  2.6.1 Delay based PUF ................................................................. 22
  2.6.2 Memory Based PUF ............................................................... 27
  2.6.3 PUFs Requiring Special Fabrication Process ....................... 29
2.7 Classification of PUFs Based on Security ................................. 31
2.8 Inter- and Intra- Hamming Distance ........................................ 33
2.9 PUF Metrics ............................................................................. 35
  2.9.1 Security Metric: Uniqueness ................................................ 35
  2.9.2 Security Metric: Randomness .............................................. 36
  2.9.3 Security Metric: Reliability .................................................. 38
2.10 Attacks against PUFs ............................................................... 38
2.11 Advantages and Disadvantages of PUFs ................................. 40
2.12 Field Programmable Gate Array (FPGA) ............................... 40
2.13 Types of Field Programmable Gate Arrays ............................. 41
2.14 Implementation of PUFs on FPGA .......................................... 41

Chapter 3 ....................................................................................... 43

Design and Implementation of c-ROPUF ....................................... 43
  3.1 Configurable ROPUF (c-ROPUF) .............................................. 43

Chapter 4 ....................................................................................... 50

Effects of Aging ............................................................................. 50
  4.1 Background ............................................................................ 50
List of Tables

4.1: Bit Flip occurrences on Spartan 3E FPGAs ..................................................58
4.2: Bit Flip Occurrences on FPGA for a period of 30 days.................................59
4.3: Average RO frequencies of the tested FPGA due to aging effects on Spartan 3E....59
4.4: Uniqueness of PUF due to aging.....................................................................63
4.5: Comparison of Average Uniqueness...............................................................64
5.1: Average RO frequencies of the tested FPGAs at varying supply voltages.............71
5.2: Percentage of Bit flips.....................................................................................71
List of Figures

1.1: Process Cycle of Counterfeit Parts.................................................................2
1.2: The life cycle of reconfigurable hardware...................................................4
2.1: On-chip Physical Unclonable PUF...............................................................12
2.2: Physical Unclonable Function.......................................................................13
2.3: Challenge Response of a PUF......................................................................14
2.4: Variability of ICs..........................................................................................19
2.5: Five-stage Ring Oscillator............................................................................23
2.6: Ring Oscillator PUF......................................................................................23
2.7: Arbiter PUF.................................................................................................26
2.8: Architecture of Tristate Buffer PUF..............................................................27
2.9: 6-T SRAM....................................................................................................28
2.10: Butterfly PUF.............................................................................................29
2.11: Coating PUF...............................................................................................30
2.12: Optical PUF................................................................................................31
2.13: PUF Uniqueness........................................................................................36
2.14: PUF Randomness.......................................................................................37
3.1: The proposed configurable ROPUF design in a single CLB.........................45
3.2: c-ROPUF design in a single CLB using hard macro procedure......................45
3.3: (a) Implementation of the proposed c-ROPUF for an FPGA region.............48
3.3: (b) Mapping the proposed design on the bottom right FPGA region.................49

4.1: Reasons for Performance Degradation due to Aging.............................51

4.2: Reliability of PUF..................................................................................54

4.3: Spartan 3E FPGA divided into six regions.................................................56

4.4 (a), (b), (c), (d), (e), (f): Frequency fluctuation in ROs due to aging............60

5.1: Experimental equipment: (a) Temperature control panel.........................70

5.1: (b) Two Basys-2 Spartan-3E FPGAs under test inside the temperature chamber.....70

5.2: Effects of temperature variations on FPGAs..........................................72
## List of Abbreviations

ASIC ..................... Application Specific Integrated Circuit
CLB ........................ Configurable Logic Blocks
CMOS ........................ Complementary Metal Oxide Semiconductor
CRP .......................... Challenge-Response Pair
FPGA ......................... Field Programmable Gate Array
HD .............................. Hamming Distance
HW .............................. Hamming Weight
IC .............................. Integrated Circuit
LUT .............................. Look-Up Table
MOSFET ........................ Metal Oxide Semiconductor Field Effect Transistor
MUX ............................. Multiplexer
NIST ........................... National Institute of Standards and Technology
PUF ............................. Physical Unclonable Function
RO .............................. Ring Oscillator
ROPUF ........................ Ring Oscillator Physical Unclonable Function
RTL ............................. Register Transfer Level
SRAM .......................... Static Random Access Memory
VHDL .......................... VHSIC Hardware Description Language
VLSI ............................ Very Large Scale Integration
List of Notations

\( \mu \) ...................... mobility
\( \sigma \) ....................... mobility temperature coefficient
\( f \) ........................... frequency
MHz ........................... Mega-Hertz
Chapter 1

Introduction

1.1 Motivation

There are numerous different computer systems in the world today. Some are general purpose computing systems, such as consumer desktops and laptops, however, there are many systems with very specific applications. These physical systems are beginning to become increasingly complex. Originally, the computational abilities of these systems were limited to a few hard coded operations. Nevertheless, these days many of such systems have a greater computing capacity, have capabilities that are more dynamic, and are connected to some sort of network, such as the Internet. These augmentations have allowed for a better remote interfacing and greater efficiency.

The use of electronic devices is ubiquitous since they exist in a wide range of applications that facilitate different sectors such as transportation, banking, health care, business etc. This wide range of applications process sensitive data which, if disclosed, may lead to loss of privacy and other serious implications. Moreover, software piracy, intellectual property (IP) theft, and counterfeited hardware are serious issues affecting the
electronic industry. 10% of all high-technology products sold globally are counterfeit [13]. Hence, with all the advances also comes the security issues.

Counterfeit hardware is not just a concern because of the increasing cost burden they place on companies that design or supply ICs; they are also a potential threat to public safety. This is because the counterfeit components are often produced from poorly controlled processes or from discarded defective materials where the components are simply re-marked and sold in the market [1]. For instance, it was suspected that the failure of the Russian Phobos-Grunt spacecraft was due to counterfeit memory chips which were not adequately hardened against radiation [2]. Figure 1.1 shows the process cycle of the counterfeit parts.

Figure 1.1 Process cycle of Counterfeit parts [4]
In the recent years, computing systems designed using reconfigurable hardware like FPGAs are used in many sensitive applications. As reconfigurable hardware continues to become more potent and cheaper, it becomes all the more attractive to designers. On the flip side, it is also more susceptible to attackers, who attempt to take advantage of any security weakness.

Unfortunately, there are many potential ways to exploit a computing device which includes both hardware and software attacks. These attacks can steal confidential information, modify the system to perform devious activities. Attackers always target the weakest link and exploit the easiest flaws found anywhere in the life cycle. For that reason, when addressing security concerns, it is imperative to have a deep understanding of the lifecycle of a device from cradle to grave.

The life cycle entails the environment, development tools, and testing equipment etc. that will be used on the device. The lifetime is divided into three stages – manufacturing stage, application development stage and deployment stage shown in Figure 1.2.
Manufacturing Stage

Reconfigurable hardware is designed and manufactured by the FPGA vendors like Altera and Xilinx. New architectures are introduced and developed roughly every 12 to 18 months [91]. The details of each architecture are patented and/or are trade secrets, so as to have a competitive advantage. Most FPGA companies are fabless, so they rely on a third party foundry to manufacture the physical devices. After manufacturing, these devices are either sold directly to the system developers who build a final product, or to a third party partner company, e.g. Digilent, that constructs a development board with the FPGA and an array of peripherals. These boards are
bought by another entity, who customize the boards to fit their end product. The boards are usually targeted towards users in a specific industry such as medical, image processing, and high performance computing, so on and so forth.

➢ Application Development Stage

The second stage the life cycle is the application development stage. It is during this stage, the FPGA is integrated into the end-product, and is programmed to carry out its intended functionality. The designer uses either the FPGA chip and builds their own platform to meet their intended application, or uses a third party FPGA development board.

The process of application development requires the use of a variety of CAD tools typically from a variety of sources. These tools include Electronic System Design (ESL) tools which take a high level language, like C/C++, MATLAB, and translate it to a Register Transfer Level (RTL) hardware description language (HDL). The RTL is then synthesized into a logic netlist with the help of numerous logic synthesis tools from EDA companies like Cadence, Mentor Graphics, and Synopsys. In the end, physical synthesis tools transform the logic netlist into a bit-stream file which is used to program the FPGA.

These tools throughout the application design stage may use Intellectual Property (IP) cores. For instance, we may receive a core from one of the tool vendors or use some open source core [92] and each of these requires that you trust the maker of the core.
Deployment Stage

The last stage of lifetime of the reconfigurable hardware is when the chip is deployed into various systems. This clearly depends on the application because FPGAs have a wide range of applications which could be cars, planes, phones, satellites, wireless routers, weapons used in defense and so on. Also, the vulnerability of the device in the deployment stage is heavily dependent on its application. It involves the physical security of the device, in other words how easily accessible and protected is it from a physical handling, as well as the functionality of the device.

Owing to the enormous costs associated with fabricating chips most fabless companies send their design to the foundry. Accordingly, design, manufacturing, and testing of integrated circuits has moved across several companies residing in various countries. This makes it extremely difficult to secure the supply chain and opens up enormous possibilities for security threats. To sum it up, with the rapid growth of electronic devices used in different facets of day to day living, it is highly likely that security will remain a concern for many more years to come.

In the wake of these security issues, researchers are coming up with better techniques to counteract the security attacks which includes insertion of hardware Trojan, cloning, physical attacks, side channel attacks etc. Traditional security mechanism implements security measures, such as authentication, integrity, confidentiality, and non-repudiation, with the help of cryptographic primitives such as encryption/decryption algorithms, digital signature schemes, and authentication codes [3]. These cryptographic techniques help prevent passive physical attacks such as cloning, reverse engineering, counterfeiting, and the insertion of malicious components. The greatest inadequacy of
these types of security measures is that, they rely on the protection of the secret keys which are stored in non-volatile memory. The sensitive data stored in these non-volatile memories can be retrieved directly through invasive attacks [3]. Attackers aim to uncover this key in order to break an authentication protocol or to steal private data. For this reason, both generation as well as storage of the key needs to be robust against such invasive attacks. The security method to protect sensitive data should possess the following features:

- **Secure Key Generation:** A secure key that is random, unique and unpredictable which can be generated for every instantiation of the security primitive, for example, block cipher.
- **Secure Key Storage:** The secret key can be assigned to, stored and retrieved by instantiation without being revealed.
- **Secure Execution:** The instantiation of the primitive can execute the crypto algorithm without revealing any, whether complete or partial, information about the key and without an adversary being able to influence the internal execution.

Hence, each of the above assumptions require a layer called the physical security as the countermeasure. The physical security layer is implemented with the help of primitives and techniques including:

1) **True Random Number Generators (TRNGs):** TRNGs entails the distillation of random numbers truly physical sources of randomness which can be used to produce highly random keys from crypto applications.
2) Design Styles: These are the countermeasures that minimize and ideally eliminate certain physical side channels.

3) Physical Unclonable Functions: A novel technology called Physically Unclonable Function (PUF) provides a kind of device identification that is needed to solve the previously stated issue. A PUF is a security primitive that can be used to generate a response that is unique to a given device. PUFs are built by leveraging the small inconsistencies in the manufacturing process. Even with extreme control of fabrication process no two instances of a chip can be physically identical, hence, it is impossible to duplicate a PUF. In other words, PUFs are primitives that produce unpredictable, instance-specific bit-strings using implementation that provide physically secure key generation and storage.

PUFs can also be defined as a combination of a physical source of randomness or entropy, i.e. an IC component that exhibits within-die variations, and a measurement technique. This measurement technique can convert small analog signal difference to unique digital bit-strings. Also, these differences in the signal are introduced by within-die variations.

### 1.2 Organization

This thesis is organized as follows.

Chapter One: This chapter introduces the motivation of this research and organization of the thesis.
Chapter Two: This chapter provides a background about Hardware Security and PUFs and a brief discussion about the application of PUFs, different types of PUFs and PUF security metrics.

Chapter Three: Chapter three focuses on the design and implementation of a configurable Ring Oscillator PUF (c-ROPUF).

Chapter Four: Chapter four focuses on the effects of aging on c-ROPUF.

Chapter Five: Chapter five focuses on the effects of temperature and voltage variation on c-ROPUF.

Chapter Six: This chapter concludes the thesis and discusses the future work on this topic.
Chapter 2

Background

2.1 Hardware Security

Hardware oriented security and trust is an emerging field intended to provide secure environment for embodied systems. Hardware devices like computers and smartphones are used so very often to communicate sensitive information, such as financial data, in an untrusted environment which makes them vulnerable to attacks. Furthermore, computing platforms are being deployed in countless critical infrastructures, such as financial systems, smart grid, sensitive governmental organizations etc., where repercussions of a successful security attack could be serious. With a surge of tools that have the capability to crack hardware devices, academic as well as industrial researchers continue to unearth robust techniques to prevent hacking into the hardware systems. Consequently, these technologies have become more and more complex.

Due to economic reasons most of the modern ICs are manufactured in off-shore fabrication facilities [4]. Each party associated with the design and manufacture of an IC can be a potential adversary who is responsible for deliberate insertion of malicious modifications, also known as Hardware Trojans [5-8].
Hardware Trojan is a growing hardware security attack that poses real threats to the security of all types of electronics devices such as ASICs, FPGA, microprocessors, and embedded systems. Trojans can alter the functionality of a physical system that is embodied on a silicon chip intending to leak private data.

A Xilinx paper [9] discusses a number of threats to hardware design that are as follows:

- Reverse engineering involves a third party which inspects a design comprising of layout, components, and firmware with the goal to recreate it for future development.
- Cloning entails less of a desire to deeply understand a design; the attacker instead intends to copy and resell a design, without incurring the overhead of developmental costs.
- Overbuilding occurs when a subcontractor builds supplementary units than have been ordered for fabrication by the Original Equipment Manufacturer. The intent is to sell the units for themselves later on.
- Tampering occurs when an attacker attempts to gain unauthorized access to an electronic system, so as to alter its firmware.

2.2 PUF: Definition

A Physical Unclonable Function is a random probabilistic function that is unique for every instance of the die. PUFs derive their randomness from the uncontrolled random manufacturing process variations in the IC. PUFs are used for a variety of applications such as device authentication [10, 11] and secret key generation [12, 13]. It
is a hardware-based challenge-response function which maps its responses to its challenges exploiting complex statistical variation in the logic and interconnect in the Integrated Circuits (ICs).

![On-Chip Physical Unclonable Functions](image)

**Figure 2.1: On-chip Physical Unclonable PUF [3]**

PUFs are an alternative to storing random secret bits in volatile or non-volatile memory which are vulnerable to attacks [5]. Instead of storing these random secret bits, PUFs generate random bits every time they are evaluated. Furthermore, an efficient PUF generates a key that varies from one chip to another and reproduces a key from a chip each time the key is requested from that chip. Besides, a PUF should be robust against any attack that aims to reveal its key. PUFs provide protection against physical attacks and are suitable for implementing read-proof hardware which is very hard to read by an attacker.

Physical attacks are basically of two types: invasive and non-invasive. An invasive attack is defined as an attack where the attacker physically breaks into a device in order to modify its structure. A non-invasive physical attack is one where the attacker carries out measurements without modifying the structure of the device. An attack to a PUF whether invasive or non-invasive alters its behavior. This implies that a significantly different response will be generated when the same challenge is applied to a PUF. A
considerably different response is a response where the noise level, with respect to the enrollment measurement, is higher than the noise level of responses caused by means of environmental stresses. A detection method of these higher noise levels allows the device to take appropriate action when an attack is detected. In the event that the PUF responses are used to implement a secure key storage mechanism, these higher noise levels lead to a significantly different secret key being generated.

Unclonability of a PUF implies that they can be used for anti-counterfeiting purposes and secure key storage. When PUFs are used to assess the authenticity of a product, a physical property of the PUF is measured and subsequently translated into a bit string that is verified. When the PUF responses are used as secret keys, it is imperative that the PUF responses are dealt with within the device to keep them protected.

![Physical Unclonable Function](image_url)

**Figure 2.2: Physical Unclonable Function [83]**

### 2.2.1 Challenge-Response Pair

An input to a PUF is typically called the challenge and the generated output is typically called the response. An applied challenge and its response are generally called a
challenge-response pair or a CRP. These terms are from the field of security, where challenge-response authentication is a process that verifies an identity by requiring correct authentication information in response to a challenge.

Since the challenge-response relationship is determined by the structural disorder of a physical material (such as silicon in a chip), it is a physical function. It is unclonable because it is very difficult, if not impossible, to build an exactly identical copy of a PUF instance. Though no successful physical cloning has been reported thus far, modeling it mathematically has been shown [25].

Figure 2.3: Challenge Response of a PUF [46]

2.3 Applications of PUFs

A few applications for PUFs with a strong focus on security have been proposed over the years. It is worth mentioning that the suitability of a PUF for a particular application depends on the application requirements.
2.3.1 Signature Generation

A group of individual PUF structures can be used to generate a unique signature for authentication of a device. In other words, PUF can be used to identify physical objects the same way as biometrics is used to identify people. Since the signature has been generated dynamically from the physical properties of the device in which it is embedded, hence it is tamper resistant and cannot be duplicated. This tamper resistant signature is used in a variety of intellectual property (IP) protection schemes.

For instance, in case of a Ring Oscillator PUF, to derive (N-1) bit digital signature of PUF, consisting of N ROs, the frequencies of oscillators are compared. The output bit after the comparison is assigned “0” or “1” depending on which oscillator is faster. The selection of oscillators for comparison of the frequency is controlled by MUXs based on the input challenge applied to the circuit.

The idea for IP protection is that the unique ID of each device is enrolled in a database by the manufacturer before it is released in the market. So even if a third party illegitimately obtains the full design of the hardware, they will not be able to produce a device that can be authenticated, because the signature is generated by processes that cannot be precisely controlled during manufacture [15]. Some examples of applying PUFs’ ability to generate unique signatures to enhance the security of RFID authentication are provided in [16, 17, 18]. Another application of PUF is used in the symmetric-key authentication protocol proposed in [19]. Lastly, PUFs are applied for authentication of mobile sensor network nodes in [20]. Hence, the idea is to prevent attacks when physical access to a node is possible.
2.3.2 Cryptography

Another application of PUF is the generation of secrets for cryptography. For example, a mobile phone whose firmware must be decrypted on each startup is a cryptographic application. The cryptographic key must somehow be stored securely. Nonvolatile memory or volatile memory are vulnerable to physical attacks or side channel attacks. Now, physically disassembling such a circuit will destroy its delay characteristics and consequently change its output, therefore PUFs can reduce these vulnerabilities to a great extent. More so, the advantage of a PUF is that these secrets do not have to be stored anywhere on the hardware since they are generated dynamically.

2.3.3 Random Number Generator

With some modifications in the design, a PUF can function as a cryptographically secure random number generator. There are a number of ways that True random number generators have been made by exploiting the meta-stability of a D-Flip flop [22], Ring Oscillators [23], and SRAM PUFs [21]. Similarly, deterministic random bit generators (DRBG) can be created as shown in [24]. DRBGs uses a deterministic algorithm in order to create pseudo-random numbers and subsequently seed it with the random signature generated by a PUF. Hence, the numbers that are generated are not predictable if the seed remains secret.

2.4 Sources of Variations

As silicon PUFs exploit random variations this section discusses different sources of variations in IC. These random variations in a circuit affect parameters such as
threshold voltage, leakage current, delay, etc. which changes the timing and logical behavior of a circuit. These random variations or physical imperfections establish themselves as changes in the electrical characteristics of the component. Various reasons for variations are:

- Manufacturing process: Variations in process parameters like oxide thickness, doping concentration, diffusion depth, etc. results in manufacturing process variation. They are caused due to non-uniformity in silicon wafer and variabilities in diffusion of dopants.
- Operating conditions: The output of a circuit varies due to environmental factors such as temperature variation and operating voltage fluctuation.
- Aging: Aging leads to permanent degradation of the characteristics of transistors with long-drawn-out usage. Aging leads to slower operation of circuits, irregular-timing characteristics, increase in power consumption, and functional failures.

There are two components of a process variation:

- Inter-die process variations – Inter-die variations are the variations that arise between chips on the same wafer or different wafers.
- Intra-die process variations – Intra-die variations are the variations that arise between different devices and interconnects that reside within the same chip. In other words, within-die variations are the geometrical and chemical imperfections that exists in the nanometer sized components in the chip.
Process variation is also divided into two types: temporal variation and spatial variation. On-chip spatial variation is introduced during the manufacturing process of the chip which can either be random or systematic. Stochastic or random variation is caused due to random concentration of dopants in transistors and variation in gate oxide thickness [26]. These are uncontrolled variations that is intrinsic to the silicon material. On the contrary, systematic variation is caused by errors in the fabrication process for example reticle stepper alignment errors.

A PUF exploits the spatial random variation to generate its Challenge-Response Pairs. This uncontrolled random variation produces random responses in a PUF. Systematic spatial variation caused as a result of imperfections in the fabrication process decreases the randomness of a PUF. On the contrary, temporal variation is triggered during the operating phase of a chip. Temporal variation can be further classified as reversible or irreversible temporal variation. Temporal reversible variability is caused by thermal effect and temperature variation makes PUF responses noisy and unreliable. Temporal irreversible variability, in other words, aging also affects the reliability of a PUF. Figure 2.4 shows the different types of variabilities in the Integrated Circuits.
2.5 Noise

All electronic device will have some kind of noise. The source of noise varies from one device to the other. In [98], different types of noise produced due to different sources are explained in details which are as follows:

- Noise due to manufacturing process - The variation in silicon during the design and manufacturing may cause noise in an electronic device. This noise is explicit to an integrated circuit. After the device is manufactured, it includes the information of the noise produced due to variations. A Physical Unclonable Function must be able to exploit the manufacturing noise which in turn helps in identifying the chip properties.
• Local Noise – A noise that emerges during the circuit operation is known as local noise. This kind of noise is observed as a result of the random thermal motion of the charge carriers. Even though the local noise may help in generation of random numbers but it is not desirable in PUF circuits. This noise has to be compensated in order to lower the intra-chip variation.

• Global Noise – This noise is attributable to the environmental conditions like variation in temperature and supply voltage during the circuit operation. High environmental noises can make the PUF responses unreliable and lead to high intra-chip variation. This makes it difficult to perform circuit identification. Hence, it is essential that a PUF design is not sensitive to the environmental noise.

2.6 Types of PUF

A variety of PUF designs have been documented over a period of time. It is noted that a new PUF design has appeared each year since its invention [11]. A literature survey was conducted for this thesis to get an idea of different types of PUFs available in the literature. In 2001, Pappu et al. introduced the concept of a physical one-way function which led to the idea of a PUF [27]. Subsequently, several other PUF techniques have been proposed. Additionally, Maes et al.’s work presents a comprehensive study on different types of PUFs that has been proposed so far [28].

PUFs can be broadly classified into two types depending on the physical material used to construct it during the fabrication process: a) non-silicon PUFs b) silicon PUFs.
Silicon PUFs are fabricated using existing ASIC process. They are based on uncontrollable random process variations and thus generate unique signatures. Non-silicon PUFs derive their key from variations occurring in the physical device rather than from integrated circuit. Likewise, PUFs can also be classified in two categories: electronic PUFs and non-electronic PUFs.

- **Electronic PUF:** In this type of PUFs, the behavior of the challenge-response pair is determined based on the electronic properties such as threshold voltage of a transistor, delay of a gate etc.

  Some examples of Electronic PUFs are:
  
  a) Arbiter PUF, b) Ring Oscillator PUF, c) Butterfly PUF, d) SRAM PUF, e) Coating PUF, f) Bi-stable PUF, g) Flip-Flop PUF, h) Glitch PUF etc.

- **Non-electronic PUF:** In non-electronic PUFs, the challenge-response behavior is determined based on the non-electronic properties.

  Some examples of Non-electronic PUFs are:

  a) Optical PUF, b) RF PUF, c) Magnetic PUF, d) CD PUF, e) Acoustical PUF, f) Paper PUF

In this work, we focus only on the silicon PUF designs which are designed and fabricated just like any other CMOS circuit. Silicon PUFs allow easy integration with existing silicon based security system designs. Non-silicon based PUF implementations have also been proposed in the literature, e.g. Optical PUF, RF PUF [29], but their practicality is limited because of restricted manufacturing flow and difficulties in integration of non-silicon PUFs with standard CMOS design.
2.6.1 Delay based PUF

In the recent years, several delay based PUF implementations have been proposed in the literature. A typical PUF generates random bits by amplifying the difference in some electrical characteristic of two identically designed circuits. PUFs either exploit FPGA configuration memory [30] or components such as SRAM cells [31, 32] or add a circuit to extract the process variation. Delay PUF is based on the delay variation of logics and interconnects. The principle mechanism of a delay based PUF is to compare a pair of structurally identical circuit elements and compare the delay mismatch introduced by random manufacturing variation to derive a secret response. Some examples of Delay based PUFs are ROPUF, Arbiter PUF, Tristate Buffer PUF etc.

2.6.1.1 Ring Oscillator PUF

Ring Oscillator (RO) PUF is one of the earliest classes of delay-based PUFs first proposed by Suh and Devadas [3]. Due to the limitations of routing, PUF designs based on symmetry are deemed less suitable. In case of a ROPUF, an absolute symmetry is not necessary, which is one of the reasons for its popularity. A RO is nothing but a loop of invertors having odd number of stages where the output of the last inverter is fed as an input to the first. The circuit oscillates with a frequency which can be determined from the delay of each inverter stage. Moreover, the inverters are implemented with the help of MOSFETs, for which the gate capacitance has to be charged for source-drain current to flow. This introduces a delay in the output of the inverter which changes along with the input after a finite amount of time.
Consider the example of a RO as shown in the figure 2.5. All five inverter gates are assumed to have the same delay and interconnect lines are supposed to have a negligible delay. However, in actual instantiation of a RO, these assumptions are not valid on account of the uncontrollable manufacturing delay. Hence, even if the ROs are produced on the same manufacturing line, the output frequency generated is slightly different.

Ring Oscillators are widely used in ICs to generate clocks or characterize performance. Each RO has a unique frequency even if many oscillators are fabricated from the same mask.

Figure 2.6: Ring Oscillator PUF [3]
Also, ROs have been widely used for creating sensors to measure voltage and temperature effects on different platforms [34] because RO is sensitive to process and environmental variations. PUF capitalizes on this sensitivity to variation of a RO to generate key required for authentication. ROPUF derives its secret response from variations in propagation delay of identical ROs and each inverter adds to the delay resulting into a square wave. ROPUF is considered more reliable under a wide range of temperatures. ROPUFs is widely accepted because they are less complex and easier to fabricate.

ROPUFs use a different approach toward measuring small random delay variations caused by manufacturing variations. The output of the digital delay line is inverted and fed back to its input, creating an asynchronous oscillating loop. Due to random process variation, the exact frequency will also be partially random and device dependent. The frequency measurements are done by either an edge detector, which detects rising edges in a periodic oscillation, or a counter which counts the number of edges over a period of time. The value of the counter contains all the details of the desired measure and is considered the PUF response.

A ROPUF is comprised of $n$ identically laid-out ROs, RO1 to ROn. A pair of ROs out of $n$ ROs, are selected with the help of multiplexers where the PUF challenge is the select bits of the multiplexers. The frequencies of the ROs, $f_a$ and $f_b$ ($a \neq b$), are measured by means of a pair of counters. On account of process variation, the frequencies $f_a$ and $f_b$ have a tendency to differ randomly from each other. Using a simple comparison method, a response bit $r_{ab}$ is produced by the quantization of two real-valued quantities, $f_a$ and $f_b$, that is presented as follows [33]:

\[ r_{ab} = \text{quantize}(f_a, f_b) \]
Response bits, \( r_{ab} = \begin{cases} 1 & \text{if } f_a > f_b \\ 0 & \text{otherwise} \end{cases} \)  \hspace{1cm} (2.1)

Measurement of RO frequencies is a part of the sample measurement. In the identity mapping phase, the frequency difference between the two pairs, \( f_a \) and \( f_b \), is calculated. The sign and magnitude of the quantity represented by \( \Delta f = (f_a - f_b) \) differ from one chip to another which is attributable to the process variation.

2.6.1.2 Arbiter PUFs

Arbiter PUF was first introduced in 2004 [60] which is based on the variation in the propagation of identical delay lines. Arbiter PUF architecture consists of switches and an arbiter. Arbiter PUF lets the rising-edge signal to travel through two paths with different delays. At the end of the delay paths is the arbiter that decides which delay path is the winner of the race. One path is faster due to manufacturing process variations which reaches the flip flop first to provide a one-bit response. The arbiter at the end of a delay path is simply a D Flip-Flop with one signal attached to the clock pin and the other to the data pin. Switch based PUF needs a larger challenge/response space than ROPUF. Further, there are meta-stability issues with the flip flop. This is why ROPUF is preferred over switch based PUFs.
2.6.1.3 Tristate Buffer

This type of PUF is same as Arbiter PUF where the switching elements are replaced with a tristate buffer which selects the delay paths. However, unlike the Arbiter PUF, the two paths to D-FF and the clock of the flip flop are independent of each other. The enable input of the buffer acts as the challenge. Each buffer has slightly different propagation delay on account of the process variation. So, different challenges produce different propagation delays between two delay paths which in turn generates different response bits at the output. Tristate buffers takes two inputs and produces one output and they have three states of operation: logic ‘0’, logic ‘1’and high impedance. When the enable pin of the buffer is set, the input is mirrored to the output, else output reaches a high impedance state. A one-bit response is generated depending on the faster path. Also, Tristate Buffer PUFs are claimed to be more power (i.e. 18%) and area (i.e. 23%) efficient when compared to the arbiter PUF [35].
2.6.2 Memory Based PUF

Memory-based PUFs work on the concept of unpredictable start up states of the feedback based on CMOS memory structures to generate unique response bits [36]. On power up, these structures settle to one of their stable states and provide a signature when an array of these is used. The response is limited to the internal logic and is not accessible to the outer logic for high security. These are more susceptible to environmental noise. Some examples of Memory based PUFs are SRAM PUF, Butterfly PUFs etc.

2.6.2.1 SRAM PUF

A Static Random Access Memory (SRAM) is a volatile memory technology which loses its contents shortly after power-down. The SRAM cell uses two cross-coupled inverters to achieve the memory effect. The inverter circuit has two stable states
i.e. it is bi-stable. The switching speed of the transistors of the inverters determines the state after power-up. Theoretically, the transistors of both inverters are built equally so as to achieve an unpredictable initial state. However, in practice the SRAM cells show a bias because of process variations. So the unpredictable initial state of the biased SRAM cell is used as PUF value. A 6-T SRAM cell consists of two cross coupled inverters and two access transistors.

![6-T SRAM Diagram](image)

**Figure 2.9: 6-T SRAM**

### 2.6.2.2 Butterfly PUF

Butterfly PUFs are designed for FPGAs. It consists of two latches with output of one connected to other. To obtain a response, both of the latches are forced into unstable state. After sometime the unstable condition is removed and the latches are let to settle down, thereby generating a one-bit response.
2.6.3 PUFs Requiring Special Fabrication Process

This type of PUF is based on the measurements taken from a special layer of deposited material. The nature of elements and the measurement style establish the difference between them.

2.6.3.1 Coating PUF

Coating PUFs are categorized as Non Silicon or Non-electronic PUF. A coating PUF consists of alumino phosphate coated over the IC, which is doped with dielectric particles like TiO$_2$, SrTiO$_3$, BaTiO$_3$ [37]. Dielectric particles are random in size and shape with a relative dielectric constant different from the dielectric constant of the coating material. Furthermore, there is an array of metal sensors between the passivation layer and the coating. If the dielectric particles are smaller than the distance between the
sensor parts, then only a satisfactory randomness is achieved. Figure 2.10 shows the structure of a Coating PUF.

The challenge is a voltage of certain frequency and amplitude applied to the sensors at a certain point in the sensor array. The sensor plates act as capacitors with a random capacitance value due to the presence of the dielectric particles and coating material. The values obtained from the capacitors are subsequently converted into secret keys.

![Figure 2.11: Coating PUF](image)

### 2.6.3.1 Optical PUF

The Optical PUF was proposed by Pappu et.al which was one of the first PUFs with its original name as “Physical One-Way Functions (POWFs)”. As shown in Figure 2.11, Optical PUF consists of a circular polarizer blocking the light rays that reflect directly from the top of the PUF. The challenge for an Optical PUF consist of the angle of
incidence, focal distance or wavelength of the laser beam, a mask pattern blocking a part of the laser light, or any other variation in the display pattern. The output of such a PUF is a speckle pattern that consists of several randomly distributed bright and dark patches. The speckle pattern results from multiple scattering of laser light in a disordered optical medium [38, 39, 40]. Also, bit-string with high entropy can be obtained from the speckle pattern with the help of image analysis.

Figure 2.12: Optical PUF [97]

2.7 Classification of PUFs Based on Security

PUFs can be classified based on different aspects, the ease to evaluate is one of the aspects that shows the ease and efficiency with which a secret response can be extracted from the PUF. Physically Unclonable Functions are classified into three major types depending on the applications and the security features they provide [41]:

31
- **Strong PUFs [3, 38]:** Due to a large CRPs space, strong PUFs can be directly authenticated without any cryptographic hardware. Strong PUFs have the following properties:
  1) Strong PUFs support a large CRPs space which makes it harder for an adversary to compute all the CRPs within a given time.
  2) Strong PUFs generate responses that are stable against environmental conditions.
  3) Even if a polynomial sized sample of CRPs is given, an adversary cannot predict the response to a new and randomly applied challenge.

- **Controlled PUFs [42]:** Controlled PUFs are primarily used to prevent modeling attacks [41]. Controlled PUFs apply a wrapper logic built around a strong PUF in order to prevent the challenges being applied directly to the strong PUF and also to inhibit the direct access to responses obtained from the PUF.

- **Weak PUFs [43]:** Weak PUFs are called weak because the number of responses are linearly related to the number of components that subject to process variation. Weak PUFs have the following properties:
  1) Generates small number of CRPs.
  2) The responses generated by a weak PUF are stable across numerous readings as well as environmental variations.
  3) Responses are unpredictable and strongly depend on the manufacturing process variation.
2.8 Inter- and Intra- Hamming Distance

The fundamental application of PUFs lies in its identification purposes. For a set of instantiations of a particular PUF construction, inter- and intra-distances are calculated as follows:

- For an applied challenge, the inter-distance between two different PUF instantiations is the distance between the two responses generated from this challenge applied once to both the PUFs.
- For an applied challenge, the intra-distance between two evaluations on one single PUF instantiation is the distance between the two responses generated from this challenge applied twice to one PUF.

It is imperative that both inter- and intra-distance are measured on a pair of responses resulting from the same challenge. The distance measure which is used can vary depending on the nature of the response. In most cases Hamming Distance is used when the response is a bit-string. The value of both inter- and intra-distance are dependent on the challenge applied and the PUFs involved. The inter- and intra-distance characteristics are evaluated with the help of histograms showing the occurrence of both inter-and intra-distances, observed over a number of different challenges and a number of different pairs of PUFs. In many cases, the histograms for both inter and intra-distances can be estimated by a Gaussian distribution and are summarized with the help of their means, represented by $\mu_{\text{inter}}$ and $\mu_{\text{intra}}$ respectively, and their standard deviations, represented by $\sigma_{\text{inter}}$ and $\sigma_{\text{intra}}$ respectively.
Inter-chip Hamming Distance (HD\textsubscript{inter})

Inter-chip Hamming Distance is a metric that measures the uniqueness of a PUF. HD\textsubscript{inter} is used to assess the difference in the responses of two PUF instances evaluated with the same challenge. The ideal value of HD\textsubscript{inter} is 50%. Average or Mean values for different PUFs can vary significantly from 20% to 50%, and greatly depends on whether bias effects are present.

The HD\textsubscript{inter} can be evaluated as shown in the following example:

\[\begin{array}{c}
1010010110 \quad \text{(bit-string of chip}_0\text{ during enrollment under conditions } \alpha_1) \\
1100011101 \quad \text{(bit-string of chip}_1\text{ during enrollment under conditions } \alpha_2) \\
\hline
0110010111 \quad = 5/10 = 50\%
\end{array}\]

Intra-Chip Hamming Distance (HD\textsubscript{intra})

Intra-Chip Hamming Distance is a metric that measures the resilience of a PUF to fluctuating environmental conditions. In other words, HD\textsubscript{intra} is used to measure the difference in the response of a particular PUF when the same challenge is applied. There are two phases of generating the bit-strings: Enrollment phase and Regeneration phase. Enrollment is the process of generating the bit-string for the first time; typically done at 25°C and at normal supply voltage. Whereas, Regeneration is the process of reproducing the bit-strings; typically done at different temperatures like 0°C, 25°C, 85°C and supply voltages ranging from -5% to 5%.

The HD\textsubscript{intra} can be evaluated as shown in the following example:
HD_{intra} quantifies the number of differences that occur in the bit string during one or more succeeding regenerations; where the number of differences is the Hamming Distance between the bit strings. HD_{intra} expresses the average noise in the responses and indicates Reproducibility. The ideal value for Intra-Chip Hamming Distance is 0%.

2.9 PUF Metrics

A good PUF design is the one where changing one bit in an applied challenge should alter nearly half of the responses. PUF response is considered to be a random variable with a characteristic probability distribution [3]. Response analysis is composed of three components or parameters, namely: Uniqueness, Randomness, and Reproducibility.

2.9.1 Security Metric: Uniqueness

Uniqueness, in other words inter-die randomness, is a measure of how the response bits are not related across dies. It is a measure of how different a PUF instantiation on a die is from another such instantiation on another die built using the same design files and technology. High uniqueness can be achieved by reducing any
systematic bias in the design that can lead to a predictable bit response across chips. Various studies in literature have showed that a high degree of uniqueness across dies is achievable for most PUF implementations [44, 45, 46, 47, 48, 49, 50, 51, 3, 52].

![Figure 2.13: PUF Uniqueness](image)

As shown in equation 2.2, uniqueness can be calculated using the inter-chip Hamming Distance (HD). In the equation, \( u \) and \( v \) are the two chips that are being compared, \( m \) is the number of chips used, and \( n \) is the number of response bits generated. \( R_u \) is the response bits generated from chip \( u \) after applying challenge \( C \). \( R_v \) is the response bits generated from chip \( v \) after applying the same challenge \( C \). HD is distance between response bits generated from chip \( v \) and chip \( u \). The ideal value for Uniqueness is 50%.

\[
Uniqueness = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(R_u, R_v)}{n}
\]  

(2.2)

2.9.2 Security Metric: Randomness

Randomness within a die is a measure of the unpredictability of the response. This implies that despite the prior knowledge of a large number of challenge-response pairs
(CRPs), response for a new challenge is unpredictable. It also indicates that even with prior knowledge of all other response bits, every bit in the response is unpredictable. There are a variety of test suites that aim to find non-randomness, in the form of a pattern, in a sequence of bits under test [21, 29, 49, 58]. In other words, a failure to find patterns in any set of tests can suggest randomness from a statistical point of view. A bias (percentage of 1’s and percentage of 0’s) in the response of nominally equivalent PUFs is a result of a systematic bias in the variations that are exploited to generate the response bits. An analysis of the potential systematic biases in the design may provide a better understanding for designing a PUF with high randomness. The randomness of PUF response is analyzed by running the National Institute for Science and Technology (NIST) statistical tests [53] on the responses.

A higher value of randomness can be attained by using circuits with regular layout with small sized devices. Regularity of the layout minimizes the impact of intra-die systematic variation [55] and a small sizing maximizes the impact of random variations [54].

![Figure 2.14: PUF Randomness](image)

Figure 2.14: PUF Randomness
2.9.3 Security Metric: Reproducibility

Reproducibility is the consistency with which a PUF generates its response across environmental variations, ambient noise, and aging [91]. Although most response bits generated by a PUF demonstrate high reliability, some do not have a strong bias towards resolving to either of the two states i.e. a ‘1’ or a ‘0’. High reliability is not only desirable but also critical in applications such as encryption key generation. A conventional method to enhance reproducibility is to use error correction codes (ECC) in order to correct the raw response from the PUF [56]. However, these ECC blocks have high VLSI overheads, which increases with the increase of error correction capability. Moreover, a careful implementation of these error correction techniques is required in order to avoid introduction of any security vulnerability through leakage of information about the response bits [56]. Researchers have also proposed some error reduction techniques with overall lower overheads and/or higher security [57, 58].

\[
Intra - chip \text{ } HD = \frac{1}{k} \sum_{t=1}^{k} \frac{HD(R_s, R_s^t)}{n} \times 100%
\]  
\[\text{Reproducibility} = 100\% - HD_{intra}\]

2.10 Attacks against PUFs

The following are a set of possible attacks that can be attempted against a PUF [33]:

- Active attack – An active attack can be physically invasive or non-invasive. In an invasive attack, a chip is delayered by means of advanced devices, for instance,
focused ion beam or by making use of specialized chemicals. In theory, these kind of attacks tend to change the internal configuration of a chip permanently and consequently destroy the PUF. Accordingly, the PUF secret is not revealed making it tamper-resistant. However, in practice, no such attack has been carried out on a PUF hence this property cannot be confirmed. Conversely, non-invasive attacks do not bring about a permanent change in the PUF structure, but they make the PUF unstable by disrupting the operating condition of a PUF.

- **Passive attack** - A PUF may be attacked passively through side-channel information like power consumption or electromagnetic radiation emitted from a chip containing a PUF.

- **Replay attack** - An attacker can make copies of challenge-response pairs during an authentication process and use it subsequently to fake the original PUF. This sort of attack can be avoided by using a particular CRP only once. However, this approach requires a large number of CRPs to be generated from a PUF so as to authenticate a chip as many times before it runs out of CRPs. However, producing a large number of CRPs is a challenge especially with limited resources.

- **Cloning attack** - An adversary attempts to build a physical copy of the original PUF with duplicate challenge-response behavior. However, no successful cloning attacks on PUF has been reported so far in the literature. The main reason is that one has to manufacture a sizable number of chips using identical layout mask, and then one has to compute the complex delay characteristics of each one of them in order to find out any close matching. Owing to the constraints of time and money
topped off with no control over the manufacturing variation, it is extremely difficult to successfully clone a PUF.

- Modeling attack - Constructing a model of a PUF mathematically is another security threat. A successful modeling attack can steal sensitive information or get access to restricted resources. In [25] Ruhmair et. al. reports a successful modeling attack on delay based PUFs. Generating truly random PUF responses is critical to thwart a modeling attack.

2.11 Advantages and Disadvantages of PUFs

The advantages of PUFs are cost reduction and security. The biggest disadvantage is noisy output. PUF produces unstable response which is sensitive to environmental factors. Also noise results in random errors.

2.12 Field Programmable Gate Array (FPGA)

Field Programmable Gate Arrays (FPGA) are silicon chips that can be reprogrammed. FPGAs consists of embedded SRAMs, high speed I/Os, logic blocks etc. FPGAs comprises of logic elements and interconnects that connect the logic elements. The basic elements of a FPGA are Configurable Logic Block (CLB), Configurable Input Output Block and programmable interconnects. Configurable Logic Block is the basic logic cell which is RAM or PLD based. CLBs are surrounded by a set of programmable configurable I/Os blocks.
2.13 Types of Field Programmable Gate Arrays

There are four types of FPGA – SRAM based, Anti fuse based, EPROM based and EEPROM based.

- **SRAM based FPGA**: SRAM based FPGAs can be configured again and again. A non-volatile memory programs the device whenever it is powered up.

- **Anti-Fuse based FPGA**: This type of FPGA can only be programmed once. Anti-fuse is a device that can conduct current only when “burned”.

- **EPROM based FPGA**: EPROM is a non-volatile memory cell. EPROM based FPGAs function in a similar way as EPROM devices. EPROM devices are erasable using ultra violet light. EPROM device consists of cells that are configured off board in a device programmer.

- **EEPROM based FPGA**: EEPROM is a non-volatile memory cell but can be erased electrically. The device can be configured and reconfigured off board using a device programmer.

2.14 Implementation of PUFs on FPGA

As far as PUFs are concerned, actual device implementation and testing is either done on an Application-Specific Integrated Circuit (ASIC) or a Field-Programmable Gate Array (FPGA). FPGAs are invaluable to PUF research for several reasons:
• FPGAs are re-configurable.

• Experiments can be performed swiftly and simultaneously with the whole test bench loaded onto the same hardware as the PUFs.

• Even small variations to routing and component placement can be verified quickly.

• A host of dedicated hardware components with nominal delay characteristics are already present on-chip for testing.

• FPGA PUFs are more constrained when compared to their ASIC counterparts.

• Placement of circuitry on-chip is constrained to discrete locations and usable locations may not be distributed evenly across a die.

• It is hard to transfer a design between FPGA technologies due to the large variation in available resources.
Chapter 3

Design and Implementation of c-ROPUF

In this chapter, we introduce a configurable ROPUF design consisting of configurable ROs that can fit into one Configurable Logic Block (CLB) in Xilinx-Spartan-3E FPGA. A technique that selects 1-RO pair among k-RO pairs which has the maximum frequency difference is proposed in [3]. This technique is known as 1-out-of-k technique that improves the reliability of a PUF [59]. However, the 1-out-of-k scheme sacrifices k times more area when it is implemented on a FPGA. Hence, to overcome this drawback [60] introduced a configurable RO design with 8 possible configurations which occupies the same area but with higher quality. However, it is not mentioned if this method can generate more output bits, also, whether it can produce stronger secret key. Furthermore, the performance of the design as proposed in [60] was not validated with respect to uniqueness, reliability, reproducibility, etc.

In this section, we discuss a configurable ROPUF that can generate more response bits while using the same area on Xilinx Spartan 3E FPGAs.

3.1 Configurable ROPUF (c-ROPUF)

The configurable ROPUF architecture is designed with the help of Look Up Tables (LUTs) and dedicated multiplexers on a single CLB that is controlled by Programmable
XOR gates. Each CLB consists of a local routing that connects the dedicated multiplexers with fixed routing delay. The individual RO loop delays are constructed with the help of CLB local routing which greatly reduces the size of the design. This size restriction ensures that any difference in the delay between the identical ROs only depends on the manufacturing process variations. In order to accomplish an identical mapping, ROs are defined as a hard macro and duplicated to ensure that all routings to other logic resources remains identical. A CLB in a Xilinx Spartan 3 device consists of four slices, where each slice is comprised of two Look-Up Tables (LUTs), two flip-flops, a multiplexer (F5 MUX), and some other resources.

The proposed design generates sixteen response bits using two CLBs instead of generating a single response bit from two identically instantiated ROs in two CLBs. Although, some response bits will always be the same, a neighbor-like-chain algorithm is adopted to generate response bits in order to avoid data dependency that may exist among the response bits. Figure 3.1, presents the internal CLB routings of the c-ROPUF that connects LUTs to FiMUXs with fixed delay [60].
Figure 3.1: The proposed configurable ROPUF design in a single CLB.

Figure 3.2: c-ROPUF design in a single CLB using hard macro procedure
Figure 3.1 and 3.2 above presents the c-ROPUF design in a single CLB using hard macro procedure. As observed in Figure 3.1, slice1 and slice 2 consists of LUTs that are configured as PXOR gates which act as controlled inverters in order to control active RO oscillations. The RO oscillations are controlled by connecting one input of these LUTs to logic ‘1’ and feeding back the output of the active RO to the other input. A pair of the PXORs is connected to its corresponding F5MUX in the same slice. The FiMUX at the bottom of the slice is programmed as F6MUX in order to connect two F5MUXs that are located at the bottom slices. The output of F6MUX at the bottom is connected to all the four LUTs in the slices, slice 3 & slice 4.

These LUTs are configured as PXOR gates which act as buffers when connected to logic ‘0’ so as to bypass the inverted RO-bit from the bottom F6MUX throughout the RO loop. The top two F5MUXs are connected to the top F6MUX located at slices 3. The output of c-ROPUF is driven from top F6MUX and fed back to PXORs at the bottom of two slices to control the active RO's oscillations. Four selection lines S1, S2, S3 and S4 are used to select a pair of RO (from two CLBs). Hence, 16 unique RO frequencies can be obtained from one pair of the configurable ROs under the assumption that each pair of RO is compared with the same configuration.

After mapping the design on a single CLB, the design is instantiated on the entire area of each tested FPGA. To implement the design on the entire area of each FPGA, every chip was divided into six equal regions with 40 CLBs per region. The basic structure of our proposed design for a FPGA region (40 CLBs) is shown in Figure 3.2. As shown in the same figure, the 1 to 40 decoder connects 40 CLBs to a VDD signal (logic 1) for activating a RO from one CLB among 40 CLBs for a short time. Each CLB is
The system is comprised of 16 ring oscillator loops. Since the ROPUF design is sensitive to heat, deactivating ROs for a short time is necessary to avoid the impact of induced heating by these ROs. This prevents the occurrence of noisy RO frequencies on account of the change in the delay around RO loops [62]. To achieve that, a period of 0.1ms is allowed before the next RO is activated [62]. This period is controlled by the timing controller in Figure 3.2 that synchronizes a 10-bit challenge generator, a 16-bit up-counter that counts the active RO and a reference counter that counts the FPGA clock cycles during this period.

As illustrated in Figure 3.2 (a), S1 to S4 selections are used to select an RO from 16 ROs in the CLB. Logic “1” is provided to select a RO in the chosen CLB. Thus, the selected RO is activated and connected to the up-counter through a 40x1 encoder to count the active RO oscillations for 0.1ms. Thus, in order to compare the frequency of two ROs, both of them must be identically configured. This will guarantee that any difference in the generated frequency is only because of manufacturing process variations and not dynamic routings. In the design, each region (40-CLBs) includes 40*16= 640 ROs.
Figure 3.3: (a) Implementation of the proposed c-ROPUF for an FPGA region;
The total possible generated frequencies are $16 \times 640 = 3840$ per FPGA region. Figure 3.3 (b) shows an instance of our proposed design mapped on the bottom right FPGA region (40 CLBs). Neighboring chain algorithm with adjacent CLB technique is used to generate response bits in order to remove dependency and reduce systematic variation effects on ROPUF response. In addition, we are proposing a new algorithm to nullify systemic variation effects which is discussed in the following section.
Chapter 4

Effects of Aging

A PUF is a one-way probabilistic function based on the random manufacturing process variation. This manufacturing process variation is spatial in nature and is expected to remain static post fabrication of a chip. Apart from spatial variation i.e. process variation, ICs are affected by temporal variation. After manufacturing an IC, the performance degrades due to temporal variation that causes a drift of device parameters. Temporal variations can be reversible like change in the ambient temperature and supply voltage. However, temporal variation caused due to aging is irreversible. Aging and environmental variations, such as temperature and voltage variation, may lead to performance degradation and lower reproducibility in ICs [63-65, 66, 67].

4.1 Background

The most important reason for any performance and reproducibility degradation is the run-time aging effect caused by hot carrier injection (HCI), negative bias temperature instability (NBTI), electro-migration (EM), and oxide breakdown [68]. Moreover, Negative Bias Temperature Instability (NBTI) [69] and Hot Carrier Injection (HCI) [70] are considered to be the most important drift-related aging effects in current technologies.
Furthermore, threshold voltage (VT), channel length, timing, etc. are critical parameters that have an impact on the performance of a chip.

HCl is a function of switching activity, where shifting the threshold voltage of the transistors any further accelerates degradation over a period of time [68]. The extent of degradation due to NBTI and HCl is dependent on factors like temperature, supply voltage, threshold voltage, logic function, technology and geometry [70, 71, 72]. Also, the components of the circuitry slowly undergo structural degradation with constant usage of ICs which results in hard faults. These faults cannot be corrected and the chip becomes unreliable [73]. Additionally, aging permanently alters many crucial parameters the PUF uses to generate its key, so it is a concern for PUF reproducibility.

Figure 4.1: Reasons for Performance Degradation due to Aging
In other words, there are two main reasons of degradation which are: a) oxide wear-out and b) interconnect failure [73].

1) Oxide wear-out – Oxide wear-out is the phenomena where the gate oxide of a transistor wears out due to the following reasons:

a) Hot Carrier Injection (HCI) – The carriers with a high energy collide with the gate oxide layer and remain trapped there. Hence the oxide layer is damaged which results in modification of the transistor characteristics. Moreover, high switching rate of a circuit as well as excess supply voltage augment this effect.

b) Negative Bias Temperature Instability (NBTI) - The applied electric field across the gate oxide causes dangling bonds to develop at the interface of the channel and the oxide layer. This affects a transistor by increasing the threshold voltage, consequently, making switching difficult. NBTI is boosted by high temperature and high supply voltage.

c) Temperature-Dependent Dielectric Breakdown (TDDB) - Due to the voltage applied across the gate oxide, conduction starts through the gate oxide with the help of the trapped charges. This results in a gradual break-down of the oxide layer. A high operating voltage as well as higher temperature accelerates Temperature-Dependent Dielectric Breakdown.

2) Interconnect failure- Apart from the transistors, interconnecting wires are also prone to aging. Electro-migration causes failure in the interconnects. As a result of high current flow, the metal atoms in the wires shift, resulting in faulty connections. This phenomenon is enhanced by high temperature.
Though a preventive solution is required in case a PUF is affected by aging, it is just as important to study how aging actually affects the functionality of a PUF. We present a study of the aging effect on a PUF based on on-chip aging experiments.

In this work, we simulate aging that is [68] caused by HCI. Due to accumulation of electrical charges within the dielectric layer caused by HCI, the threshold voltage needed to turn on the transistor increases. The increased threshold voltage results in increased transistor switching time, consequently slowing down the transistor speed [15].

### 4.1.1 Reproducibility of PUF due to Temporal Variation

HCI heavily depends on the RO frequency. As discussed in 2.5.1.1, each ring oscillator oscillates with a slightly different frequency on account of manufacturing variation. A fixed sequence of oscillator pairs is selected so as to generate a fixed number of bits. The output bits or the response bits from the same sequence of oscillator pair comparisons will differ from chip to chip. Provided that oscillators are identically laid out, the frequency differences are determined by manufacturing variation and an output bit is equally likely to be “1” or “0” if random variations dominate.

As a result of different degradation rates, the RO with higher frequency may get slower than the other in a pair. This condition is responsible for a bit flip which makes the PUF unreliable in case the same pair is selected by the applied challenge and their relative positions remain unchanged. So as to increase the robustness of a PUF under environmental variations and the device’s aging, one may choose only such pairs which have a large-base frequency difference. Figure 4.1 demonstrates how the reproducibility of an ROPUF is affected by environmental variations and aging. In an ROPUF, one can compare the RO frequency of a given pair selected by a particular challenge.
Suppose a pair has two ROs, RO1 and RO2, selected by two multiplexers, MUX1 and MUX2, respectively. The pair will always generate a reproducible bit if and only if the frequency of the RO1 is always higher than the frequency of RO2, irrespective of environmental variations and device aging. This pair is called as a stable or good pair. Figure 4.1(a) and Figure 4.1(b) illustrates how a bit can flip beyond a certain operating condition or after a certain amount of aging of the device. In the event that the inverters of RO1 experience more logic ‘0’ than the inverters of RO2, the bit flip could occur well before the target chip lifetime. Target chip life time is the time we expect to use the device containing the PUF before it fails to operate.

RO1 is faster than RO2, before considering any environmental variations and aging. However, after a device reaches a particular age or experiences a different operating condition, the RO1 could become slower than the RO2 and flip the bit. This point is called as the crossover point. The potential crossover in frequencies makes it harder to get reproducible bits in an ROPUF under different operating conditions over a long period of time. The rapid VT degradation due to aging will compel some bits in the PUF output to flip. So, the components in an ROPUF should be designed in a way that it is less sensitive to aging and so the PUF can generate reproducible keys. The rate of
increasing or decreasing frequency, i.e. the slope of the straight lines as shown in Figure 4.1, is crucial.

4.1.2 Response of a Ring Oscillator PUF

RO frequency is typically generated from a series of inverters in the RO loop. The presence of process variation inside the chip causes uneven delays across the chip. Hence a pair of ROs mapped at two different chip locations produces two different frequencies: \( f_a \) and \( f_b \). Frequencies \( f_a \) and \( f_b \) are compared to see which one has the higher frequency. If \( f_a \) is greater than \( f_b \), a response bit 1 is generated; otherwise the response is 0.

4.1.3 Configurable ROPUF

In the configurable RO PUF design, each FPGA chip is divided into six equal regions containing 40 CLBs in each region, with a total of 240 CLBs per FPGA chip. Each CLB is comprised of 16 ring oscillator loops and each region (40 CLBs) consists of \( 40 \times 16 = 640 \) ROs. The total possible generated frequencies are \( 16 \times 640 = 3840 \) per FPGA region. The frequency is computed using \( (x \times \left(\frac{50}{y}\right)) \), where \( x \) is the cycle count for each RO, \( y \) is the cycle count for a 50 MHz reference clock. The value of \( y \) is preset at 10000 cycles which implies that the RO cycles are measured within a period of 0.1 ms.
4.2 Previous Work on Aging

Previous work that focused on solving the error correction, mainly concentrated on the errors introduced by the reversible temporal variations in ICs. This type of variation is expeditious and can be observed over a relatively shorter period of time.

Since aging is time dependent and often takes a long period of time to present noticeable errors, accelerated aging test has been one of the ways to estimate aging. Stott et al. conducted accelerated aging tests to estimate how FPGA components such as LUTs, routing wires etc. degrade with time [74]. In this work, we run accelerated aging tests on an FPGA-based ROPUF to observe how the PUF is responds to aging.

Kirpatrik et al. proposed software-based techniques to thwart the effects of aging on PUFs [75]. Lim et al. briefly mention aging in their work on Arbiter PUFs [76] where
they performed a one-month-long aging test on the Arbiter PUF under normal condition. Guajardo et al. also discussed how robust the SRAM PUF is against aging [77].

4.3 Methodology and Experimental Set up

In this work, we present accelerated aging experimental results along with the effects of temperature and voltage performed under varying environmental conditions on 5 Spartan 3E FPGAs. This aging experiment is conducted for 30 days. The number of ROs mapped on each region of the 90 nm Spartan 3E is 640. As there are six regions in each FPGA so six bit-stream files are obtained. Diligent-Adept software is used to download the bit stream file on each FPGA region. After the FPGA region is configured, it is connected to a 16-bit data bus of 16801 Agilent Logic Analyzer. The Logic-Analyzer can accurately measure data samples and observe the non-harmonic frequencies of the individual ROs. Responses are generated with the help of Chain-like Neighbor Coding where RO (n) is compared with RO (n+1). Thus, there are 639 response bits generated from 640 ROs. Responses from all FPGAs are recorded once every day. To begin with, these responses are compared to the responses generated at normal setting to determine the bit flip occurrences. The percentage of bit flip occurrences for Spartan 3E FPGA is also calculated for the 30-day aging period. The average frequencies for all six regions of the FPGA are calculated along with the standard deviation. Subsequently, the inter-chip Hamming Distance (HD) and intra-chip HD are calculated. Inter-chip HD is the measure of uniqueness of the PUF, whereas, intra-chip HD evaluates the reproducibility of the PUF under varying environmental conditions.
4.4 Results and Analysis

Responses are recorded once every day for the 30-day period. The bit flip occurrences are measured by comparing these responses to the responses generated at normal setting. The total number of bit flip occurrences for a 30-day aging experiment on 5 Spartan 3E FPGAs are shown in Table 5.1. The bit flip occurrences for each region of the FPGA are tabulated and then the average bit flips per FPGA chip is calculated. It is observed that the number of bit flips are higher when the difference in frequency between the RO pairs is low and vice versa. One of the region in FPGA 1 has the lowest number of bit flips i.e. 15, which is because the RO comparison pairs have high frequency difference. On the other hand, one of the regions on the same FPGA has a higher number of bit flips i.e. 27. This is because many RO comparison pairs have frequency difference of less than 1.0 MHz, in some instances it was as low as 0.02 MHz. Hence, it is imperative that the RO comparison pairs with higher frequency difference should be selected in order to prevent the bit flip occurrences.

Table 4.1: Bit Flip occurrences on Spartan 3E FPGAs

<table>
<thead>
<tr>
<th>FPGA Regions</th>
<th>FPGA 1</th>
<th>FPGA 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top Left</td>
<td>15</td>
<td>19</td>
</tr>
<tr>
<td>Top Middle</td>
<td>27</td>
<td>31</td>
</tr>
<tr>
<td>Top Right</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>Bottom Left</td>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td>Bottom Middle</td>
<td>23</td>
<td>13</td>
</tr>
<tr>
<td>Bottom Right</td>
<td>20</td>
<td>38</td>
</tr>
<tr>
<td>Average Bit Flips per FPGA chip</td>
<td>19</td>
<td>21</td>
</tr>
</tbody>
</table>

The same phenomena are observed for another Spartan 3E FPGA. One of the region of FPGA 2 has a low number bit flip occurrence i.e. 19, where most bit flips occur at places where the RO comparison pairs have a frequency difference of 0.1 MHz. For
FPGA 2, most bit flips occur when the RO comparison pairs have a frequency difference of 0.1 - 0.5 MHz.

Table 4.2: Bit Flip Occurrences on FPGA for a period of 30 days

<table>
<thead>
<tr>
<th>FPGA Regions</th>
<th>5th day</th>
<th>10th day</th>
<th>15th day</th>
<th>20th day</th>
<th>25th day</th>
<th>30th Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top</td>
<td>Left</td>
<td>13</td>
<td>20</td>
<td>12</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Middle</td>
<td>32</td>
<td>24</td>
<td>25</td>
<td>27</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>Right</td>
<td>29</td>
<td>14</td>
<td>36</td>
<td>25</td>
<td>15</td>
</tr>
<tr>
<td>Bottom</td>
<td>Left</td>
<td>12</td>
<td>17</td>
<td>18</td>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Middle</td>
<td>13</td>
<td>17</td>
<td>24</td>
<td>23</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>Right</td>
<td>14</td>
<td>20</td>
<td>21</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>Average Bit Flips</td>
<td>18</td>
<td>18</td>
<td>22</td>
<td>21</td>
<td>19</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 4.3 shows the average regional frequency for $k$th region (six regions per FPGA) in an $i$th FPGA. In the table, the average regional frequency of the tested FPGA, attributable to the aging effect, is calculated for every five days during the 30-day period. In order to do that 150 data sheets from all the five FPGAs are collected for analysis. To accurately estimate the average frequency for each Ring Oscillator, 10 data samples from each data sheet was considered. It is observed that there are no extremes in the average RO frequencies which implies that FPGAs do not exhibit aging effects.

Table 4.3: Average RO frequencies of the tested FPGA due to aging effects on Spartan 3E

<table>
<thead>
<tr>
<th>FPGA Reg.</th>
<th>1st Day</th>
<th>5th Day</th>
<th>10th Day</th>
<th>15th Day</th>
<th>20th Day</th>
<th>25th Day</th>
<th>30th Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top</td>
<td>Left</td>
<td>220.66</td>
<td>220.93</td>
<td>220.80</td>
<td>220.84</td>
<td>222.43</td>
<td>221.24</td>
</tr>
<tr>
<td></td>
<td>Middle</td>
<td>217.32</td>
<td>227.34</td>
<td>217.52</td>
<td>217.42</td>
<td>219.02</td>
<td>217.85</td>
</tr>
<tr>
<td></td>
<td>Right</td>
<td>233.07</td>
<td>227.34</td>
<td>233.15</td>
<td>233.17</td>
<td>235.11</td>
<td>233.43</td>
</tr>
<tr>
<td>Bottom</td>
<td>Left</td>
<td>227.11</td>
<td>220.64</td>
<td>220.68</td>
<td>220.69</td>
<td>222.21</td>
<td>220.92</td>
</tr>
<tr>
<td></td>
<td>Middle</td>
<td>217.62</td>
<td>217.62</td>
<td>217.57</td>
<td>217.51</td>
<td>219.16</td>
<td>217.83</td>
</tr>
<tr>
<td></td>
<td>Right</td>
<td>218.05</td>
<td>218.25</td>
<td>218.30</td>
<td>218.20</td>
<td>219.83</td>
<td>218.52</td>
</tr>
<tr>
<td>Avg. Reg. frequency</td>
<td>222.30</td>
<td>222.02</td>
<td>221.33</td>
<td>221.30</td>
<td>222.96</td>
<td>221.63</td>
<td>224.63</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
</tbody>
</table>

Figure 5.1 (a), (b), (c), (d), (e) and (f) show the change in frequency due to aging effects for 16 ROs for all the six regions of the configurable ROPUF. It is observed that the frequency fluctuation is not significantly high for the duration of 30 days. Although, the frequency fluctuation for most regions is less than 20 MHz for all the regions. There is also an overlap of frequencies which explains the occurrence of bit flips. All in all, frequency fluctuation observed across all the FPGA chips is minimal.

Figure 4.4 (a): Top Left - Frequency fluctuation in ROs due to aging.
Figure 4.4 (b): Top Middle - Frequency fluctuation in ROs due to aging

Figure 4.4 (c) Top Right - Frequency fluctuation in ROs due to aging
Figure 4.4 (d): Bottom Left - Frequency fluctuation in ROs due to aging

Figure 4.4 (e): Bottom Middle - Frequency fluctuation in ROs due to aging
Inter-chip Hamming Distance (HD) of PUF: In the aging experiment, 639 response bits were generated from 640 ROs with the help of Chain-like Neighbor Coding. The inter-chip HD is calculated using equation 2.2 for both pre-aging as well as post-aging effects. The inter-chip HD is also a measure of the uniqueness of the PUF which is shown in Table 4.4. The average Uniqueness is calculated by comparing the responses generated by all five FPGAs.

Table 4.4: Average Uniqueness of PUF due to aging

<table>
<thead>
<tr>
<th></th>
<th>Average</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-aging</td>
<td>46.85%</td>
<td>35.90%</td>
<td>56.25%</td>
</tr>
<tr>
<td>Post-aging</td>
<td>45.9%</td>
<td>36.56%</td>
<td>59.30%</td>
</tr>
</tbody>
</table>

Table 4.5 shows the comparison of average uniqueness of the c-ROPUF after aging when compared with other PUF designs in the literature before aging.
Table 4.5: Comparison of the average uniqueness

<table>
<thead>
<tr>
<th>Comparison</th>
<th>[95, 96]</th>
<th>[3]</th>
<th>[22]</th>
<th>c-ROPUF Post-Aging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniqueness %</td>
<td>45.5%, 47.31%</td>
<td>45.4%</td>
<td>45%</td>
<td>45.9%</td>
</tr>
</tbody>
</table>

Higher the difference between responses from each chip, higher is the value of uniqueness. It is imperative that the uniqueness is high because it indicates that the ROPUF could generate unique response from more number of FPGA chips under the same challenge.

**Intra-chip Hamming Distance (HD) of PUF:** Intra-Chip Hamming Distance measures the resilience of a PUF to varying environmental conditions such as temperature, voltage and aging. In this the intra-chip HD is calculated for evaluating the impact of aging on the PUF. HD\(_{\text{intra}}\) shows the average noise in the responses and indicates the Reproducibility of the PUF responses. The ideal value for Intra-Chip Hamming Distance is 0%.

The HD\(_{\text{intra}}\) for all the six regions for each tested FPGA was calculated using the equation (2.3). Most regions of FPGA 1 had an HD\(_{\text{intra}}\) value between 2.34% – 4.21% where 2.34% is closer to 0%. The reproducibility of the PUF was evaluated with the help of equation 2.4. Considering the above mentioned values of intra-chip HD and plugging these in equation 2.4, we get the reproducibility value to be in the range of 95.79% - 97.66% post-aging as opposed to pre-aging reproducibility value of 99.7%. Hence, it is concluded that the intra-chip HD goes down with aging which in turn effects the reproducibility of the PUF.
4.5 Summary

In this chapter, we studied the effect of aging on PUFs based on an accelerated aging testing on a configurable ROPUF. Our results show that RO frequencies are sensitive to aging. The experimental results based on a group of five 90-nm FPGAs show that inter-chip HD of PUF remains unaffected by aging. The results also show that the intra-chip HD of PUF degrades with aging making the PUF responses unreliable. The bit flip occurrences for each region of the FPGA are calculated. It is observed that most bit flips occur when the frequency difference between RO comparison pairs is low.
Chapter 5

Effects of Temperature and Voltage Variation

Temperature variation is another concern for PUF reliability since operating temperature affects the delay of a device by changing its threshold voltage and mobility [68]. As temperature increases, the threshold voltage decreases which in turn increases the drain saturation current. Furthermore, increase in the temperature decreases the mobility of the MOSFET, which in turn decreases the drain saturation current. This mobility degradation results in decrease of the delay of the device [65]. However, unlike aging, the effects can be reversed by returning the device to its nominal operating temperature.

5.1 Background

Instance-specific response of a PUF is affected by:

- Fixed within-die variation: This is the kind of variation that occurs within the chip as discussed in section 3.2.
- Gradual changes: These type of change is a slow change that occurs due to wear out.
- Environmental conditions: temperature and supply voltage variation affects the response of a PUF.
Environmental variations temporarily alter the process variations around the individual RO loop which results in noisy frequency being generated by the individual ROs. In CMOS circuits, any temperature variation changes the delay. These variations in the delay may compromise the ability of the PUF to generate the same response for an applied challenge under different temperature conditions. Reproducibility of a delay based PUF is highly dependent on consistency of the delay, which in turn depends on the drain saturation current of the transistor [78, 79]. The equation for drain saturation current of a MOSFET is as follows [80]:

\[
I_{DS} = \frac{1}{2} \mu(T) C_{OX} \frac{W_{eff}}{L_{eff}} (V_{GS} - V_t(T))^\alpha
\]  

(5.1)

Where, \(I_{DS}\), \(W_{eff}\), \(L_{eff}\), \(\alpha\), \(C_{OX}\), \(V_{GS}\), \(V_t\), \(T\) and \(\mu\) are saturation current, effective channel width, effective channel length, velocity saturation index, gate capacitance, gate to source voltage, threshold voltage, operating temperature and mobility respectively.

The mobility and threshold voltage are dependent on the temperature [80] as shown in equation (5.1) and (5.2):

\[
\mu(T) = \mu(T_0) \left(\frac{T}{T_0}\right)^{-k}
\]  

(5.2)

\[
V_t(T) = V_t(T_0) - \sigma(T - T_0)
\]  

(5.3)

where, \(T_0\) is the reference temperature, \(k\) is the mobility exponent with values ranging from 1.2 to 2 [81] and \(\sigma\) is mobility temperature coefficient of threshold voltage with values in the range of 0.5mV/K to 3 mV/K [82].

In a MOSFET, if the temperature increases, then the threshold voltage of the transistor decreases, which in turn increases the gate overdrive, \(V_{GS} - V_t(T)\), consequently the drain saturation current increases. On the contrary, mobility of the
carriers increases which results in a decrease of saturation current. However, the decrease in mobility is only under nominal operating conditions. Hence, gate delay increases with increase in temperature.

Thus, ROPUF suffers from instability of the generated response bits also called as bit flips. This in turn decreases the ability of ROPUF to generate the same response bits at varying operating conditions (ROPUF reproducibility). Furthermore, a higher bit flips percentage reduces the percentage of CRPs (RO frequency pairs) that can be used to generate unique responses (Section 4.1.1); this negatively impacts the overall uniqueness of ROPUF. The performance of ROPUF in terms of the uniqueness and reproducibility of the generated response bits is negatively affected by the environmental variation.

Similar to temperature variation and aging, power supply noise also affects the delay of the device, hence the reliability of an IC. However, unlike aging the impact of voltage variation is temporary, i.e., when Vdd is back to its nominal value (1.2V), the PUF operates with a higher reproducibility.

5.2 Methodology and Experimental Set up

The design is first implemented in a single CLB comprising 16 different ROs with fixed internal routings. Fixed CLB routings excludes any possible noise from the dynamic routing due to the external routings used and warrants that the generated frequencies depend upon the random manufacturing process variations of the individual ROs.

The design is implemented on five Spartan 3E FPGAs (90nm), where each FPGA consists of 240 CLBs. Each FPGA is divided into six equal regions consisting of 40 CLBs and 640 ROs per region. By means of Xilinx ISE CAD Tools, a hard macro
procedure is performed on a single CLB, which is then duplicated on the entire region (40 CLBs). Consequently, six bit-stream files, each file pertaining to one region, is obtained. These files ensure faster mapping of the design and make certain that all ROs are identically mapped on each FPGA region via internal CLBs routings with a fixed delay. Data samples are collected from all the six regions of each chip.

As seen in Figure 5.1, a 1000 series Test-Equity chamber is used along with Agilent 18601 Logic Analyzer to collect sample frequencies at varying temperatures and DC voltage supply. As shown in Figure 5.1, the temperature chamber comprises of racks where FPGAs are placed. The environmental temperature for FPGAs under test is controlled by a temperature control unit. This unit has the ability to change the environmental temperature inside the chamber. FPGAs are placed in the chamber and the temperature is adjusted to a desired value. As shown in the figure, two FPGAs are placed in the temperature chamber and data samples are collected from the entire area of the FPGA (240 CLBs) using the Logic Analyzer. Digilent-Adept software is used to download the bit stream files from each FPGA region. After the FPGA region is configured, it is connected to a 16-bit data bus of 16801 Agilent Logic Analyzer. The Logic-Analyzer can precisely measure data samples and detect the non-harmonic frequencies of the ROs.

Temperature variation experiment is conducted at different temperatures using a temperature chamber. Responses are generated at room temperature (21°C), 25°C, 50°C, 70°C. For the voltage variation experiment, a DC power supply is used to generate responses for four different supply voltages: 1.11V, 1.20V (normal), 1.32V and 1.46V. Subsequently, average frequency for each region is calculated.
Figure 5.1: Experimental equipment: (a) Temperature control panel (b) Two Basys-2 Spartan-3E FPGAs under test inside the temperature chamber

5.3 Results and Analysis

As shown in Figure 5.1, temperature and voltage variation experiments were conducted with the help of a test chamber and DC power supply respectively. Responses are generated at room temperature (21°C), 25°C, 50°C, 70°C for temperature variation experiment and voltage variation experiment was carried out at 1.1V, 1.20V (normal), 1.32V and 1.46V.

Table 5.1 presents the average regional frequency for each $k^{th}$ region (six regions per FPGA) in an $i^{th}$ FPGA at specific supply voltages. In the Table, average FPGA regional frequency at four different supply voltages is calculated.
Table 5.1: Average RO frequencies of the tested FPGAs at varying supply voltages

<table>
<thead>
<tr>
<th>FPGA Regions</th>
<th>Applied Voltage (Volts)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.11 V</td>
</tr>
<tr>
<td>Top</td>
<td>Left</td>
</tr>
<tr>
<td></td>
<td>Middle</td>
</tr>
<tr>
<td></td>
<td>Right</td>
</tr>
<tr>
<td>Bottom</td>
<td>Left</td>
</tr>
<tr>
<td></td>
<td>Middle</td>
</tr>
<tr>
<td></td>
<td>Right</td>
</tr>
<tr>
<td>Avg. Reg. Frequency</td>
<td>228.36</td>
</tr>
</tbody>
</table>

As shown in Table 5.1, the frequency changes with respect to the voltage variations. The average frequency increment is 23 MHz for 1.32V and 46.6 MHz for 1.4V when compared to the nominal supply voltage i.e. 1.2V.

What is more, environmental temperature variations and noisy oscillators can cause the response bits to flip. As shown in Table 5.2, the percentage of bit flips is calculated for voltage and temperature variation. The bit flips occur due to temperature and voltage variations but only when the RO comparison pair has a frequency difference lower than 1.5 MHz for temperature variation. Furthermore, the result shows that the most bit flip occurs when the maximum frequency difference is 1 MHz in case of voltage variation.

Table 5.2: Percentage of Bit flips

<table>
<thead>
<tr>
<th>FPGA 1</th>
<th>Percentage of Bit Flips (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Voltage Variation</td>
</tr>
<tr>
<td>FPGA Region</td>
<td>Bottom</td>
</tr>
<tr>
<td>Bottom</td>
<td>Left</td>
</tr>
</tbody>
</table>
For a certain challenge, reproducibility can be evaluated by calculating Hamming Distances (HDs). The ideal bit flips percentage for response bits that are generated from the same chip should be 0%. The percentage of bit flips is calculated as follows:

\[ Bit\_flip = 1 - \frac{1}{n} \sum_{i=1}^{n} HD(r_i, r_{i,ref}) \times 100\% \]  

(5.1)

Figure 5.2 shows the effect of ambient temperature variations on two FPGAs. It is observed that for every FPGA, the average regional frequencies decrease with an increase in temperature and vice versa.

Figure 5.2: Effects of temperature variations on FPGAs
5.3 Summary

In this chapter, the RO frequencies are recorded for six different regions of each FPGA for varying temperature such as room temperature, $25^0\text{C}$, $50^0\text{C}$ and $70^0\text{C}$. The regional RO frequencies are also recorded for the same regions of each FPGA for varying voltage including 1.2V (normal), 1.1V, 1.32V, 1.46V. In the temperature variation experiment, it is observed that the average RO regional frequencies increase with decrease in temperature and vice versa. Conversely, the average RO regional frequencies of the FPGA chip under test increase with an upsurge of voltage supply and vice versa.
Chapter 6

Conclusion and Future Work

6.1 Conclusion

With an increasing demand and usage of computing devices, security challenges like protection of user data and preservation of user privacy are becoming consequential. Research on Hardware Security is highly interdisciplinary by nature and researchers from the various fields such as physics, mathematics, statistics, cryptography, circuit design theory, and coding theory have made significant contributions in this area. An on-chip PUF can solve the hardware security challenges in an efficacious way.

A PUF is a die-specific random probabilistic function that is unique for every instance of the die. PUFs derive their randomness from the uncontrolled random manufacturing process variations in the IC. PUFs are increasingly used for applications such as device identification or authentication [10, 11] and secret key generation [12, 13]. It is a hardware-based challenge-response function which maps its responses to its challenges exploiting complex statistical variation in the logic and interconnect in the Integrated Circuits (ICs). PUFs are an alternative to storing random secret bits in volatile or non-volatile memory which are vulnerable to attacks [14]. Instead of storing these random secret bits, PUFs generate random bits every time they are evaluated. Furthermore, an efficient PUF generates a key that varies from one chip to another and
reproduces a key from a chip each time the key is requested from that chip. Moreover, a PUF should be robust against any attack that aims to reveal its key.

In this work, we study the effects of temporal variations such as aging, temperature, and voltage variation on configurable Ring Oscillator (RO) PUF. While a PUF is built upon spatial variation of ICs, temporal variations negatively affect the reproducibility of the PUF responses. Once the applied variation is removed, varying environmental temperature and supply voltage that bring about changes in ICs that cease to exist. In contrast, aging causes irreversible changes in circuit components resulting in permanent shifts in the circuit parameters and behavior.

In a month long aging experiment, no significant fluctuation in frequency is noted. Moreover, it is observed that the uniqueness of the RO-PUF is not affected to a great extent, where the inter-chip HD is found to be closer to the ideal value of 50%. However, the intra-chip HD is affected negatively making the response bits unreliable. In the experiments conducted for temperature and voltage variation, the ROs are found to be sensitive to temperature and voltage variations. It was noticed that the bit flips occur as a result of temperature and voltage variations but only when the RO comparison pair has a frequency difference of 1.5 MHz. On the whole, we observe that even though uniqueness and randomness are relatively easy to achieve in majority of well-designed PUFs, achieving high reproducibility of the response is a challenge.

### 6.2 Contribution

In this work, a detailed analysis of the aging effect on delay based PUF based on an on-chip 30-day aging experiment is presented. The experimental data is based on a
delay PUF which is a configurable ring oscillator PUF implemented on commercially available Spartan 3E FPGAs. The contributions in this work are as follows:

- A month long on-chip accelerated aging testing is performed to observe the effect of aging on PUFs. The analysis is done by evaluating the effects of aging on the PUF metrics like uniqueness and reliability.
- A detailed study of the effect of aging on the functionality of a PUF is presented. This demonstrates how the frequencies of the ROs in a configurable ROPUF change with aging and how that change ultimately influences the PUF functionality.
- An experiment was conducted to evaluate the effect of temperature variations and voltage variations in order to determine how any variation in the ambient temperature and supply voltage impacts the reproducibility of a PUF.

### 6.3 Future Work

As a part of future work, the following could be considered:

- A variety of techniques can be explored to prevent the aging effect on PUF.
- The security risks on PUFs due to aging can be studied in detail. New techniques can also be devised to thwart those risks.
- The same aging experiment can be performed on different types of PUFs and then the results can be used to establish which type is more robust against any reversible or irreversible temporal variations.
- The temperature variation experiment can also be conducted under cryogenic conditions.
- Designing and implementing an Aging-resistant PUF on sensors.
References


[65] I.M. Filanovsky and A. Allam, “Mutual Compensation of Mobility and Threshold Voltage Temperature Effects with Applications in CMOS Circuits,” IEEE


[83] [http://rijndael.ece.vt.edu/puf/main.html](http://rijndael.ece.vt.edu/puf/main.html)


