PUF based FPGAs for hardware security and trust

Muslim Mustapa

University of Toledo

Follow this and additional works at: http://utdr.utoledo.edu/theses-dissertations

Recommended Citation
Mustapa, Muslim, "PUF based FPGAs for hardware security and trust" (2015). Theses and Dissertations. 1945.
http://utdr.utoledo.edu/theses-dissertations/1945

This Dissertation is brought to you for free and open access by The University of Toledo Digital Repository. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of The University of Toledo Digital Repository. For more information, please see the repository's About page.
A Dissertation

entitled

PUF based FPGAs for Hardware Security and Trust

by

Muslim Mustapa

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

Doctor of Philosophy Degree in Electrical Engineering

______________________________
Dr. Mohammed Niamat, Committee Chair

______________________________
Dr. Mansoor Alam, Committee Member

______________________________
Dr. Jackson Carvalho, Committee Member

______________________________
Dr. Junghwan Kim, Committee Member

______________________________
Dr. Weiqing Sun, Committee Member

______________________________
Dr. Patricia R. Komuniecki, Dean
College of Graduate Studies

The University of Toledo

August 2015
An Abstract of

PUF based FPGAs for Hardware Security and Trust

by

Muslim Mustapa

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the Doctor of Philosophy Degree in Electrical Engineering

The University of Toledo

August 2015

Hardware security threats have become a major issue in the technology sector and cyberspace. In 2011, more than 1300 counterfeit incidents were reported from around the world to the Electronic Resellers Association International (ERAi). The incidents reported in 2011 were more than double compared to the incidents reported in 2010 and 2008. The federal contract report states that counterfeiting of electronic parts has threatened the operability and reliability of the US weapons system. Electronic parts counterfeiting has become a very big business perpetrated by corrupt operators.

Just like ASIC semiconductors, reconfigurable hardware is also prone to hardware security threats. The most commonly used reconfigurable hardware is the Field Programmable Gate Array (FPGA). Demand for FPGAs has increased as can be seen by the growth in FPGA companies such as Xilinx and Altera. Despite the increased demand and use of FPGAs in the market, there is a great concern that security is not currently a part of the FPGA hardware and software to the fullest extent. Design theft, and hardware tampering threats on FPGAs can be dealt using Ring Oscillator Physical Unclonable Function (ROPUF). A ROPUF takes advantage of the process variation on a silicon chip
to generate a unique ID for the purpose of authentication. A ROPUF can be implemented on an FPGA chip to produce a unique ID for each FPGA chip. An adversary that tries to tamper with the ROPUF inadvertently changes the properties of the process variation in the silicon chip; thus any tampering attempt can be detected.

In this research, ROPUF based hardware security for FPGAs is presented. A total of 50 Xilinx FPGAs are used in our investigation. Performance in terms of uniqueness and reliability is evaluated. The effects of temperature variation, voltage variation, and aging on these parameters are also studied. Our work shows that lower number of stages used in the Ring Oscillator (RO) offers better security feature. The lower number of stages used in ROs yield higher Challenge and Response Pairs (CRPs). The higher number of CRPs contributes to higher security. In addition, we have introduced a technique called Random Patch Mixer (RPM) to minimize the systematic variations effect on the frequency generated from ROPUFs on FPGA. The results obtained by using RPM technique are shown to be better than other techniques that have been proposed before. The responses generated from ROPUF after applying the RPM technique passed most of the NIST statistical test for randomness. Finally, we show how the ROPUF can be used for the security of a Smart Grid. The security of ROPUF system is also tested using support vector machine (SVM). The SVM is trained using a large data set of challenges to predict the response sets. Results obtained show that the SVM fails to predict ROPUF responses based on the challenges, thus enhancing the security offered by the proposed authentication system.
Acknowledgements

My wonderful Ph.D. journey will never be a success without great people around me. Firstly, I would like to thank my advisor, Dr. Mohammed Niamat for his guidance, advice and support throughout my time here at University of Toledo. Secondly, my thanks go to my Ph.D. committee members who have spent their time to review my research work and share their thoughts with me. I cannot forget to thank all my colleagues for the fun time that we have had together. I also would like to use this opportunity to thank my sponsor, the Malaysian Government, which has given me the opportunity to further my Ph.D. studies. Finally, and the most important one for me, I would like to thank my wife, children, parents, and my family for their continuous support, understanding, and sacrifices throughout this Ph.D. journey.
# Table of Contents

Abstract .......................................................................................................................... iii

Acknowledgements .......................................................................................................... v

Table of Contents ............................................................................................................. vi

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

List of Figures ................................................................................................................... xii

List of Abbreviations ...................................................................................................... xv

1 Introduction ..................................................................................................................... 1
   1.1 Motivation ............................................................................................................... 2
   1.2 Research Objectives .............................................................................................. 3

2 Research Background .................................................................................................... 5
   2.1 Process Variations ................................................................................................. 5
   2.2 Physical Unclonable Functions (PUFs) .................................................................. 6
       2.2.1 Arbiter Physical Unclonable Function (APUF) .............................................. 6
       2.2.2 Butterfly Physical Unclonable Function (BPUF) ............................................. 7
       2.2.3 Ring Oscillator Physical Unclonable Function (ROPUF) ............................... 7
       2.2.4 PUF Implementation on FPGA ................................................................. 8
   2.3 Challenge and Response on ROPUF ................................................................. 9
5.3.1 Systematic Variations Effect on Frequency Distribution .................44
5.3.2 Systematic Variations Minimization by RPM Technique ...............46
5.3.3 NIST Statistical Test for Randomness ...........................................50
5.4 Summary ............................................................................................51

6 Temperature, Voltage, and Aging Effects on ROPUFs Function ..........52
6.1 Introduction ........................................................................................52
6.2 Background ........................................................................................53
6.2.1 Ring Oscillator PUF Response .........................................................54
6.2.2 Number of Stages in Ring Oscillator ..............................................54
6.3 Experimental Setup ............................................................................54
6.4 Results and Analysis .........................................................................56
6.5 Summary .............................................................................................66

7 A Comparative Study of Ring Oscillator PUFs on Different FPGA Families .....67
7.1 Introduction ........................................................................................67
7.2 Related Work .......................................................................................69
7.3 Background ........................................................................................70
  7.3.1 Ring Oscillator PUF Response .........................................................70
  7.3.2 Number of Stages in Ring Oscillator ..............................................70
  7.3.3 ROPUF Parameters ......................................................................70
7.4 Experimental Setup ............................................................................73
7.5 Results and Analysis .........................................................................74
  7.5.1 ROPUF Uniqueness ......................................................................75
  7.5.2 ROPUF Uniformity .......................................................................78
7.5.3 ROPUF Bit Aliasing .................................................................79
7.5.4 ROPUF Reliability .................................................................79
7.5.2 ROPUF Diverseness .................................................................84
7.5 Summary ..................................................................................84

8 ROPUF Application: Hardware-Oriented Security-Based Authentication for
Advanced Metering Infrastructure ..........................................................85
8.1 Introduction .................................................................................85
8.2 Related Work ..............................................................................88
8.3 Hardware-Oriented Security-Based Authentication for AMI .................90
  8.3.1 ROPUF Design .....................................................................96
  8.3.2 Authentication ....................................................................98
8.4 Proof of Concept .......................................................................101
8.4 Summary ..................................................................................106

9 Conclusions ..................................................................................107
9.1 Summary and Conclusions ..........................................................107
9.2 Contributions and Results ............................................................108
9.3 Future Works ............................................................................110

References .....................................................................................112
List of Tables

2.1 Comparison of ROPUF, APUF, and BPUF ................................................................. 9
3.1 Standard deviation for all RO stages ........................................................................... 15
3.2 Number of slice and CLB used on FPGA for single RO ............................................. 18
4.1 Uniqueness, bit-aliasing, diverseness, and uniformity for 3, 5, and 7-stage ROs .31
4.2 SD for Chip 1, Chip 2, and Chip 3 ............................................................................... 32
4.3 Reliability for Chip 3 ................................................................................................... 33
4.4 Number of comparison pairs generated on Chip 1, Chip 2, and Chip 3 ............... 36
5.1 Hamming Distance (HD) for FPGA chip 1 before RPM is applied ..................... 49
5.2 Hamming Distance (HD) for FPGA chip 1 after RPM is applied ........................... 49
5.3 NIST statistical test for randomness results ................................................................. 51
6.1 Bit flip occurrences on Spartan 3E ............................................................................. 57
6.2 Bit flip occurrences due to aging on Spartan 3E ........................................................ 59
6.3 Percentage of bit flip occurrences ................................................................................ 65
6.4 Number of comparison pairs according to threshold frequency ......................... 66
7.1 ROPUF’s parameters comparison ............................................................................. 75
7.2 ROPUF’s reliability due to changes in temperature, voltage, and aging ............... 80
8.1 Comparison of different schemes based on Smart Grid requirements .................. 89
8.2 Number of possible CRPs .......................................................................................... 98
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.3</td>
<td>Authentication time for each level</td>
<td>102</td>
</tr>
<tr>
<td>8.4</td>
<td>Data storage size for each authentication level</td>
<td>102</td>
</tr>
<tr>
<td>8.5</td>
<td>Data storage size needed based on number of devices on the AMI</td>
<td>103</td>
</tr>
</tbody>
</table>
List of Figures

2-1 Arbiter Physical Unclonable Function (APUF) .................................................................6
2-2 Butterfly Physical Unclonable Function (BPUF) ...............................................................7
2-3 Ring Oscillator Physical Unclonable Function (ROPUF) ..................................................8
2-4 ROPUF circuit ....................................................................................................................9
3-1 ROs mapped used three slices ..........................................................................................13
3-2 Test circuit diagram ........................................................................................................13
3-3 Frequency pattern for a 3-stage ring oscillator ...............................................................14
3-4 Frequencies (MHz) for 3, 5, 7, 9, and 11 stage ring oscillators ......................................18
4-1 5-stage RO .......................................................................................................................22
4-2 3-stage RO .......................................................................................................................23
4-3 7-stage RO .......................................................................................................................23
4-4 Xilinx Spartan 2 CLBs layout ..........................................................................................27
4-5 FST circuit diagram .........................................................................................................28
4-6 Difference between comparison pairs and CRPs ............................................................34
5-1 Frequencies across columns 1-3 of the CLBs .................................................................41
5-2 Location of ROs on Spartan 2 FPGA ...............................................................................42
5-3 3-stage RO frequencies across Spartan 3E .......................................................................44
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-4</td>
<td>3-stage RO frequencies across Spartan 3E FPGA after RPM technique has been applied</td>
</tr>
<tr>
<td>6-1</td>
<td>ROs numbering system based on spatial location</td>
</tr>
<tr>
<td>6-2</td>
<td>The relationship between the RO frequency distance and the probability of a PUF output flip</td>
</tr>
<tr>
<td>6-3</td>
<td>Frequency changes in ROs due to the aging effect on Spartan 3E</td>
</tr>
<tr>
<td>6-4</td>
<td>Frequency changes with respect to the temperature variations on Spartan 3E</td>
</tr>
<tr>
<td>6-5</td>
<td>Frequency changes with respect to the voltage variations on Spartan 3E</td>
</tr>
<tr>
<td>6-6</td>
<td>Temperature chamber</td>
</tr>
<tr>
<td>7-1</td>
<td>RO frequencies versus location on Spartan 3E</td>
</tr>
<tr>
<td>7-2</td>
<td>RO frequencies versus location on Artix-7</td>
</tr>
<tr>
<td>7-3</td>
<td>Spartan 3E</td>
</tr>
<tr>
<td>7-4</td>
<td>Artix-7</td>
</tr>
<tr>
<td>7-5</td>
<td>RO frequency changes with respect to aging</td>
</tr>
<tr>
<td>8-1</td>
<td>AMI in Smart Grid</td>
</tr>
<tr>
<td>8-2</td>
<td>ROPUF connected to a smart meter</td>
</tr>
<tr>
<td>8-3</td>
<td>Smart meter to utility company authentication</td>
</tr>
<tr>
<td>8-4</td>
<td>Smart meter to utility company fail authentication</td>
</tr>
<tr>
<td>8-5</td>
<td>Data concentrator to utility company authentication</td>
</tr>
<tr>
<td>8-6</td>
<td>Utility company to smart meter authentication</td>
</tr>
<tr>
<td>8-7</td>
<td>Utility company to smart meter fail authentication</td>
</tr>
<tr>
<td>8-8</td>
<td>ROPUF logic blocks</td>
</tr>
<tr>
<td>8-9</td>
<td>Parity Bits PBi generator</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>8-10</td>
<td>Parity bits from 128 response bits</td>
</tr>
<tr>
<td>8-11</td>
<td>ROPUFs registration with utility company</td>
</tr>
<tr>
<td>8-12</td>
<td>Parity bits and corresponding ROs</td>
</tr>
<tr>
<td>8-13</td>
<td>SVM prediction accuracy for ROPUF</td>
</tr>
<tr>
<td>8-14</td>
<td>Bit flip probability vs. frequency difference (MHz)</td>
</tr>
</tbody>
</table>
List of Abbreviations

AMI............................Advanced Metering Infrastructure
APUF..........................Arbiter Physical Unclonable Function
BPUF..........................Butterfly Physical Unclonable Function
CLB............................Configurable Logic Blocks
CPBP.........................Challenges and Parity Bit Pairs
FDT............................Frequency Difference Thresholds
FPGA..........................Field Programmable Gate Array
FST.............................Full Scan Technique
HD...............................Hamming Distance
HW...............................Hamming Weight
MUX.............................Multiplexer
PUF.............................Physical Unclonable Function
RO...............................Ring Oscillator
ROPUF........................Ring Oscillator Physical Unclonable Function
RPM.............................Random Patch Mixer
SD...............................Standard Deviation
SVM.............................Support Vector Machine
Chapter 1

Introduction

As a society, a lot of trust is placed on the hardware we use on a daily basis. For example, communication regularly occurs on sophisticated digital phones or computers. These same devices are also capable of monitoring our bank accounts and buying and selling goods electronically. For these reasons, it is vital that the security of hardware devices continues to improve in order to ensure the secure transfer of information across untrusted networks. By using hardware based authentication, a digital system can verify that a user is in fact who he or she claims to be through the use of unique secret keys. This secret key can be stored in the memory, or generated specifically when it needs to be used. The first option is not used because memory is vulnerable to inexpensive attacks [5][6]. The second option is more appealing because it is both simple to implement and difficult to attack.

Physically Unclonable Functions (PUFs) are one way of generating secret keys on the spot, without relying on memory. PUFs exploit process variations, which are unintentionally introduced during the manufacturing process of integrated circuits. The process variation in turn causes small amounts of additional delays within the circuit. By
using this additional delay effectively, secure bits can be generated. Silicon PUFs (SPUFs) are PUFs that are specifically designed to take advantage of the silicon manufacturing process. The SPUFs are designed to exploit the process variation and circuit delays to create unique challenge-response patterns [6]. There are two kinds of SPUF; Arbiter PUFs and Ring Oscillator PUFs (ROPUFs). An Arbiter PUF is constructed from multiplexers and an arbiter. ROPUFs are constructed from delay loops (ring oscillators) and counters.

Arbiter PUF circuits need to be symmetric in order to ensure that the routing lengths are the same [6]. ROPUFs on the other hand do not need to be symmetric. For this reason, ROPUFs are the preferred solution when working with FPGAs [7]. There are many techniques used by researchers in order to improve the reliability and uniqueness of ROPUFs [6][8][9][10]. Reliability means that the secret key generated by the ROPUF will be the same despite any change in operating conditions [8]. Uniqueness, on the other hand, refers to how each and every FPGA is able to generate a unique secret key [8].

1.1 Motivation

A ROPUF takes advantage of the process variation on the silicon chip to generate a unique ID for authentication. A ROPUF can be implemented on any VLSI chip including a FPGA to produce a unique ID for each FPGA chip. An adversary who tries to tamper a ROPUF will change the properties of the process variation in the silicon chip; thus any tampering effort will fail [3]. A ROPUF cannot be modeled because the process variation on a silicon chip is random. Until now, there was no technology that could measure the process variation with high accuracy [3].
1.2 Research Objectives

ROPUF research areas can be divided into four main categories: fabrication variation extraction [11], secret selection [6][9][10][11][22][25][26], error correction, and tests for security and reliability. Fabrication variation extraction is the study on the physical behavior of the silicon chip. This is the most fundamental research area in ROPUF that interacts directly with the process variation. The uniqueness and reliability parameters of the ROPUF are studied thoroughly in this work to take full advantage of the process variation [11][12]. Secret selection is the study of the algorithm to select the comparison pairs that are known as challenge and response pairs. The randomness parameter of the ROPUF is also studied in this research. Error correction is studied by using an algorithm that corrects any flipped bits. This is important especially for ROPUF implementation as a cryptography technique, where zero bit flipped occurrence is expected [6]. Finally, tests for security and reliability research look into the diffuseness, bit-aliasing, and probability of misidentification parameters of ROPUF.

This research focusses on process variation extractions, secret selection of challenge-response pairs, and tests for reliability and security. For the process variation extraction, the relationship between different numbers of stages used in RO with the ROPUF’s reliability and uniqueness is studied. The objective of this study is to improve the ROPUF’s uniqueness and reliability parameters by manipulating the structure of ROs. For the challenge-response secret selection, the systematic variation effect on the ROPUF is studied. The objective in this study is to develop an algorithm that can dismiss the systematic variation effect on ROPUF. For the tests and security reliability, we have
conducted a study on the weaknesses of the ROPUFs. The objective in this study is to develop an algorithm which will enhance the security and reliability of the ROPUF.

This dissertation is organized as follows:

Chapter One: This chapter briefly introduces the motivation and objective of this research.

Chapter Two: This chapter gives background information about PUF, RO and ROPUF.

Chapter Three: This chapter discusses the relationship between the number of stages used in an RO and the uniqueness of the frequency of the RO.

Chapter Four: This chapter discusses the relationship between the number of stages used in a ROPUF and the number of challenge-response pairs on CRPs on a FPGA.

Chapter Five: This chapter discusses the RPM technique developed to dismiss systematic variation effect on a ROPUF.

Chapter Six: This chapter discusses the temperature, voltage, and aging effects in ROPUF.

Chapter Seven: This chapter discusses a comparative study of ring oscillator PUFs on different FPGA families.

Chapter Eight: This chapter discusses the hardware-oriented security-based authentication for advanced metering infrastructure.

Chapter Nine: This chapter forms the conclusion of this dissertation and also discusses future work.
Chapter 2

Research Background

2.1 Process Variations

The reduced feature sizes in silicon chip devices make it hard to attain uniformity in manufacturing. This results in variation in the transistor gate length and oxide thickness that introduces propagation delays in the silicon chip [13]. This variability in the manufacturing is known as process variation. Process variations are random and cannot be controlled. Process variations can be divided into two types namely intra-die variations and inter-die variations. Intra-die variations are the variations within a single die and inter-variations are variations from chip to chip.

Intra-die variations can be categorized into two types: systematic (process shift) and stochastic (process spread) variations [13]. Systematic variations are created by reticle stepper alignment errors, mask errors from inaccuracies in the process model, and lithographic off-axis focusing errors. The sources of stochastic variations are: wafer unevenness, non-uniformity in resist thickness, and vibrations during lithography.
2.2 Physical Unclonable Functions (PUFs)

A PUF is a chip level structure that deliberately exploits random process manufacturing variations to establish the chip’s identity. There are three common types of delay PUFs that can be used to extract the delay introduced by the process variations: Arbiter PUF (APUF), Butterfly PUF (BPUF), and ROPUF.

2.2.1 Arbiter Physical Unclonable Function (APUF)

An APUF is composed of two identically configured delay paths that are stimulated by an activating signal as shown in Figure 2-1. The difference in the propagation delays of the signals in the two delay paths is measured by an edge triggered flip-flop known as an arbiter. There are two main components used in an APUF: switches, and the arbiter [14]. Various response bits can be generated by configuring different delay paths.

![Figure 2-1: Arbiter Physical Unclonable Function (APUF).](image-url)
2.2.2 Butterfly Physical Unclonable Function (BPUF)

A BPUF consists of two cross coupled latches as shown in Figure 2-2. The BPUF exploits the random assignment of a stable state from an unstable state that is forcefully imposed by holding one latch in preset while holding the other in clear mode by an excitation signal. The final state is determined by the random delay mismatch in the pair of feedback paths and the excitation signal paths due to process variations [15].

![Figure 2-2: Butterfly Physical Unclonable Function (BPUF).](image)

2.2.3 Ring Oscillator Physical Unclonable Function (ROPUF)

A ROPUF is composed of an odd series of inverters. The RO frequency is generated from the inverted signal that travels through the RO loop as shown in Figure 2-3. The presence of process variations inside logic gates and wires causes an uneven delay across the chip.
A pair of ROs could produce two different frequencies because of the presence of process variations.

2.2.4 PUF Implementation on FPGA

Researchers have compared the implementation of APUF, BPUF and ROPUF on FPGAs [16]. For APUF and BPUF, the implementation on FPGA is tedious because both designs need to be symmetric as shown in Table 2.1. It is almost impossible to get symmetric design on an FPGA because the design needs to be mapped using a fixed routing. ROPUF design does not need symmetric design which makes it the best candidate for PUF on FPGAs. ROPUF implementation on FPGAs require identical instantiation as shown in Figure 2-3.
Table 2.1: Comparison of ROPUF, APUF, and BPUF [16].

<table>
<thead>
<tr>
<th>ROPUF</th>
<th>APUF</th>
<th>BPUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does not require symmetric routing in a building block.</td>
<td>Requires symmetric routing in a building block.</td>
<td>Requires symmetric routing in a building block.</td>
</tr>
<tr>
<td>Building blocks require identical instantiation.</td>
<td>Identical instantiation of building blocks may not be necessary.</td>
<td>Identical instantiation of building blocks may not be necessary.</td>
</tr>
</tbody>
</table>

2.3 Challenge and Response on ROPUF

The response bit from an ROPUF can be generated by comparing the output signals of two ROs. More response bits can be generated by comparing additional RO pairs. The RO pair selection of RO pairs is determined by the challenge. For example, RO1 and RO2 pair generates one response bit and RO3 and RO4 pair generates another response bit. All ROs are connected to two MUXs, as shown in Figure 2-4. The challenge bits for the ROPUF circuit shown in Figure 2-4 are applied at the input of each MUX.

![Figure 2-4: ROPUF circuit.](image-url)
The challenge selects one RO to each MUX. The selected RO from each MUX will be fed into the counter to measure the number of cycles generated from each RO for a certain period of time. After both counters have measured the number of cycles from each RO, the comparator will compare the number of cycles generated from each RO. Finally, the response bit is generated. The logic used is: if the number of cycles measured from the first counter is larger than the number of cycles measured from the second counter, then the response bit is ‘1’; otherwise, it is ‘0’ or vice versa.

2.4 RO Delay

Equation 2-1 shows the RO delay is comprised of three components [11]. The parameter $d_{avg}$ is the delay component that comes from the routing and is the same for all ROs. The parameter $d_{PVa}$ is the delay component that comes from the process variations and is expected to be different for different ROs. The parameter $d_{NOISEa}$ comes from the noise factor and is a dynamic component that changes over time. When the delay between two ROs are compared ($d_a$ and $d_b$), the $d_{avg}$ cancels each other. Thus, the delay difference between two ROs comes from process variations and noise delay components.

\[
d_a = d_{avg} + d_{PVa} + d_{NOISEa}
\]  

\[
d_b = d_{avg} + d_{PVb} + d_{NOISEb}
\]

\[
d_a - d_b = (d_{PVa} - d_{PVb}) + (d_{NOISEa} + d_{NOISEb}) = \Delta d_{PV} + \Delta d_{NOISE}
\]
Chapter 3

Frequency Uniqueness in ROPUF on FPGA

Hardware security in Field Programmable Gate Arrays (FPGAs) that use PUF rely on the ability to produce a large number of unique frequencies. This chapter explores the frequency uniqueness as it relates to the number of stages used to build an RO.

3.1 Introduction

There are many techniques used by researchers in order to improve the reliability and uniqueness of ROPUFs [6][8][9][10]. Reliability means that the secret key generated by the ROPUF will be the same despite changing operating conditions [8]. Uniqueness on the other hand refers to how each and every FPGA is able to generate a unique secret key [8].

The measure of how much each ring oscillator frequency varies from the next is called frequency uniqueness. By increasing the frequency uniqueness of a system, it is possible to increase the security of that system. Until now, to the best of our knowledge, there has not been any research on how the number of stages in a ring oscillator PUF affects the frequency uniqueness. This chapter addresses this issue.
3.2 Experimental Setup

This section explains the procedure used to determine frequency uniqueness for varying stages of ring oscillators. For this purpose, three FPGA development boards are used. Each board has a single Xilinx Spartan 2 XC2S100 TQ144 FPGA. The three boards generate a total of 60 ring oscillators for each stage. Initially, data was obtained at room temperature. Various configurations of ring oscillators are tested, including rings oscillators with 3, 5, 7, 9, and 11 stages. For each of these stages, the frequency produced by each ring oscillator is measured and recorded.

The first step in designing the experiment is to create a hard macro for a single ring oscillator. This ensures that the routing lengths for each ring oscillator are identical. It is important that the routing lengths are the same so that there is no additional delay in any single ring oscillator. The total delay for each ring oscillator is as shown in Equation 2-1.

For the FPGAs used in this experiment, one Configurable Logic Block (CLB) is composed of two slices, as shown in Figure 3-1. For the hard macros used in this experiment, each slice contains only one inverter. So an N stage ring oscillator will use N inverters, N slices and N/2 CLBs. This macro is horizontally placed 20 times on the Spartan 2 FPGA as shown in Figure 3-2. Each of these ring oscillators is connected to a 1-to-20 demultiplexer which acts as an enable signal for each of the ring oscillators. The purpose of this is to ensure that neighboring ring oscillators do not cause extra noise while they are not in use. Enabling all oscillators at the same time will also produce extra heat that could affect the frequencies being generated [13].

The output of the ring oscillators is fed to a 20-to-1 multiplexer with the same select lines as the demultiplexer shown in Figure 3-2. The output of the multiplexer can
be sampled and measured. Measurements for this experiment are done using an Agilent 16801A Logic Analyzer. By using a logic analyzer, the entire waveform produced by the ring oscillators is observed and counted. The counting feature of the logic analyzer is particularly useful because the patterns produced by the ring oscillators are not uniform, as shown in Figure 3-3. So, the frequencies reported are actually the average frequencies.

Figure 3-1: ROs mapped use three slices.

Figure 3-2: Test circuit diagram.
3.3 Results and Analysis

In this section, the results of the experiments described in the previous section are presented and analyzed. This section is divided into two parts relating to the number of stages used in the ring oscillator and the reasons why some ring oscillators vary in frequency more than others. The first part focuses on the security of the Physically Unclonable Functions on an FPGA as it relates to the number of stages used. The second part focuses on the reasons why 3-stage ring oscillators have a higher variation in terms of frequency as compared to ring oscillators with more stages.

3.3.1 Impact of Number of Stages on PUF

Figure 3-4 displays the average frequency (MHz) produced by the ring oscillators. Each line represents the results based on the number of stages that were used while implementing the ring oscillators on the Spartan 2 FPGA. For each of the different stage ring oscillators, the frequency produced is nearly constant, except for the 3-stage ring oscillators. The 3-stage ring oscillators have a much greater frequency variation compared with others.

Figure 3-3: Frequency pattern for a 3-stage ring oscillator.
While ring oscillators with more than 3 stages may vary by a few MHz, 3-stage ring oscillators have been shown to vary from 120 MHz to nearly 200 MHz. The frequencies produced by 5, 7, 9, and 11 stage ring oscillators remain almost constant in comparison. Their frequencies are centered around the 115 MHz, 80 MHz, 65 MHz and 55 MHz marks, respectively. As the number of stages increases, it appears that the frequencies become more consistent. The value of the average frequency generated by the ring oscillator also decreases as the number of stages increase. As more stages are added, more delays are introduced in the circuit.

Table 3.1 shows the standard deviation of the frequencies produced from the ring oscillators when configured with different number of stages. As the table shows, the standard deviation is very low for all ring oscillators with more than 3 stages. A low standard deviation indicates that the ring oscillators may be more susceptible to bit flipping caused by noise, and therefore, erroneous results. For this reason, these ring oscillators are not suitable for ROPUF applications. However, the larger standard deviation in the 3-stage ring oscillator makes it more appropriate for ROPUF applications.

Table 3.1: Standard deviation for all RO stages.

<table>
<thead>
<tr>
<th>Number of stages</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 stages</td>
<td>11.3</td>
</tr>
<tr>
<td>5 stages</td>
<td>1.6</td>
</tr>
<tr>
<td>7 stages</td>
<td>0.74</td>
</tr>
<tr>
<td>9 stages</td>
<td>1</td>
</tr>
<tr>
<td>11 stages</td>
<td>0.62</td>
</tr>
</tbody>
</table>

In Table 3.1, the highest standard deviation for multistage ring oscillators occurs when there are 3 stages. This indicates that a 3-stage ring oscillator will have the highest
frequency uniqueness of any of the multistage ring oscillators measured. Due to the high frequency uniqueness, this model will be useful in generating secret keys, since the likelihood of flipping the bits is low.

By assuming that there are $N$ ring oscillators that produce unique frequencies, the circuit will produce $\log_2(N!)$ bits of entropy [6]. If 60 unique ring oscillators existed on a device, 272 security bits could reliably be produced. If 100 unique ring oscillators existed, 525 security bits could reliably be produced. This is only possible when each frequency is sufficiently unique from the others. As the frequency uniqueness is reduced, so is the possible number of security bits generated.

As the frequencies of two ring oscillators approach each other, the possibility for comparison errors increase due to noise. At one instance in time, the first frequency may be faster; however, at another instance, the second might be slightly faster. This will result in a flipped security bits, making the secured message completely unreadable by the receiving party. To reduce the possibility of this happening, it is important that there be a minimum difference between all ring oscillator frequencies that will be used. Ring oscillators that are not sufficiently unique should not be used. To generate a large number of security bits, it is important to maximize the number of unique frequencies produced by the ring oscillators.

### 3.3.2 Explanation of High Frequency Variation in a 3-stage RO

This section discusses the reasons why 3-stage ring oscillators have the highest variations among the ring oscillators tested. In [18], some of the basic ideas of why lower stage oscillators have higher process variation are discussed. One of the reasons this
occurs is that increasing the number of stages used by a ring oscillator also increases the
correlation coefficient between the actual delays and the theoretical delays. As more
stages are added, the delay is less dependent on process variation. This means that the
frequencies generated by a ring oscillator will converge to a central frequency as the
number of stages increases.

The authors of [13] have shown that there are two types of variation within-die,
systematic and stochastic. Systematic variation is caused by lithographic off-axis focusing
errors, reticle stepper alignment errors, and mask errors due to inaccuracies in the process
model. Stochastic variation is caused by non-uniformity in resists thickness, vibrations
during lithography, and wafer unevenness. The researchers in [11] suggest that systematic
variations are the primary cause of process variation, and thus, the largest influence on
frequency uniqueness. Certain patterns are enforced in the die via systematic variation that
reduces frequency uniqueness as the ring oscillators increases in stages.

There is a direct relationship between the amount of space consumed on an FPGA
and the number of stages in a ring oscillator. Table 3.2 shows that as the number of stages
increases, so do the number of slices and CLBs used, and therefore, space consumed on
the FPGA also increases. As this area increases, the delay from one ring oscillator will
begin to correspond more with the delay from other ring oscillators [18]. In effective
ROPUF applications, this correspondence should be minimized. By minimizing the
correspondence in delay, the ROPUF will effectively be able to produce more secure bits,
and thus, increase the security of the application.
Table 3.2: Number of slice and CLB used on FPGA for single RO.

<table>
<thead>
<tr>
<th>Ring oscillator</th>
<th>Slice used</th>
<th>CLB used</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 stages</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>5 stages</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>7 stages</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>9 stages</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>11 stages</td>
<td>11</td>
<td>6</td>
</tr>
</tbody>
</table>

Figure 3-4: Frequencies (MHz) for 3, 5, 7, 9, and 11 stage ring oscillators.
(a) FPGA Board 1.
Figure 3-4: Frequencies (MHz) for 3, 5, 7, 9, and 11 stage ring oscillators.

(b) FPGA Board 2.
(c) FPGA Board 3.
3.4 Summary

The number of stages of a ring oscillator plays a critical role in generating secure bits on a FPGA. By choosing the correct number of stages while designing a ring oscillator, the number of unique frequencies can be maximized. As the number of unique frequencies increases, the number of frequency comparisons also increases; thus, creating more secure bits, which could be used in a secret key.
Chapter 4

Relationship between Number of Stages in ROPUF and CRP Generation

4.1 Introduction

Physical Unclonable Function (PUF) is commonly used to prevent hackers from stealing information from semiconductor chips. PUFs utilize the process variations on the chip to create an irreversible function that generates unique response bits for each challenge. A good response bit can be generated by comparing two Ring Oscillators (RO) frequencies that have a significant amount of difference. An insignificant amount of frequency difference can cause bit flip in the response bit generated. A higher threshold for the frequency difference is preferred to dismiss the bit flip occurrence. As the frequency difference threshold (FDT) increases, the number of challenge and response pairs (CRP) is reduced. In this chapter, it is shown that the higher Standard Deviation (SD) of RO frequencies can compensate the higher FDT. The Full Scan Technique (FST) is used on different number of RO stages to determine the number of stages that have the highest SD for RO frequencies. The experimental results show that the SD of RO frequencies increase as the number of stages decrease. It is also shown that by reducing the number of stages, good Inter-Hamming Distance (HD), Hamming Weight (HW), and percentage of bit flip occurrences can still be obtained.
Despite the promising solution offered by ROPUF, there are still challenges that need to be overcome for ROPUF to become a practical solution. Making the ROPUF response better in uniqueness and increasing its reliability are among the challenges. Uniqueness refers to the ability of similar ROPUF circuits to generate unique responses on different chips. Reliability refers to the generation of the same response under various environmental conditions, such as temperature and humidity.

This chapter focuses on the process variation extraction for a ROPUF on a FPGA. Three different RO stages are tested and compared in terms of the SD, HW, and inter-HD. The three different RO stages are tested using our new proposed Full Scan Technique (FST), which records frequencies from all CLBs available on the FPGA.

### 4.2 Background

RO frequency is generated from the inverted signal that travels through the RO loop, as shown in Figure 4-1. The presence of process variation inside logic gates and wires causes an uneven delay across the chip. As a result, a pair of ROs will produce two different frequencies: \( f_a \) and \( f_b \). The frequencies are compared to see if \( f_a \) is greater than \( f_b \). If \( f_a \) is greater than \( f_b \), response bit 1 is generated; otherwise, the response is 0 as shown in Equation 4-1.

\[
\text{Response bit} = \begin{cases} 
1 & \text{if } f_a > f_b \\ 
0 & \text{otherwise}
\end{cases}
\]

(4-1)

**Figure 4-1:** 5-stage RO.
4.2.1 Number of Stages of the RO

In this experiment, there are three different number of stages used. Figure 4-1 shows the 5-stage RO where each component in the RO counts as one stage. The 5-stage RO consists of one NAND gate and 4 inverter gates. The NAND gate is used to control the switching of the RO. The RO is activated (starts to produce an oscillation) when the input is set to high. Figure 4-2 shows the 3-stage RO. The 3-stage RO consists of one NAND gate, one buffer gate and two inverter gates. The reason for using a 3-stage RO is explained in section 4.4. One buffer gate is used instead of inverter gate because the inverting components need to be odd in number in order to produce an oscillation. The buffer gate is added to increase the total delay in the RO; therefore, reducing the RO frequency. Finally, Figure 4-3 shows the 7-stage RO. The 7-stage RO consist of one NAND gate and 6 inverter gates.

![3-stage RO](image)

Figure 4-2: 3-stage RO.

![7-stage RO](image)

Figure 4-3: 7-stage RO.
4.2.2 RO Parameters

There are number of parameters proposed to measure PUF performance, such as uniformity, reliability, steadiness, uniqueness, diverseness, bit-aliasing, and probability of misidentification [12][13][14][15][16]. In this research, 4 existing parameters and one newly proposed parameter are used. The 4 existing parameters are chosen based on the suitability of measuring the performance of the different number of stages used in the ROPUF. The 4 parameters are uniqueness, reliability, uniformity and bit-aliasing. One new parameter proposed in this research is diverseness. Uniqueness represents the ability of a PUF to uniquely differentiate a particular chip among a group of chips of the same type [12]. Uniqueness can be measured by calculating the inter-chip HD, as shown in Equation 4-2. In this equation, \( m \) is the number of chips used, \( u \) and \( v \) are the two chips being compared, and \( n \) is the number of response bits generated. \( R_u \) and \( R_v \) are the response bits from the same challenge \( C \) for chip \( u \) and \( v \). HD is the hamming distance between response bits generated from chip \( u \) and \( v \). A good uniqueness value is around 50%. This means that at least 50% of the responses generated from chip \( u \) and \( v \) differ from each other (responses obtained by giving the same challenge to chip \( u \) and \( v \)).

\[
Uniqueness = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(R_u,R_v)}{n} \times 100\% \tag{4-2}
\]

Reliability refers to how efficient a PUF is in reproducing the response bits. Reliability can be measured by using Equation 4-3 and 4-4. \( R_i \) is the response from chip \( i \) at normal operating condition (at room temperature). \( R_{i,t} \) is \( t \)-th sample of \( R_i \)’s response from chip \( i \) at different operating conditions such as different temperature setting. A good
reliability value is 100%. As can be seen in Equation 4-4, if the HD intra (comparison of response under normal operating conditions and different operating conditions) is low or zero, then the reliability is around 100%.

\[
Intra - chip\ HD = \frac{1}{k} \sum_{l=1}^{k} \frac{HD(R_sR_{s,l})}{n} \times 100\% \tag{4-3}
\]

\[
Reliability = 100\% - HD\ Intra \tag{4-4}
\]

Uniformity estimates how uniform the ratio of ‘0’ s and ‘1’ s is in the response bits of a PUF. Uniformity can be measured by calculating the intra-chip Hamming Weight HW, as shown in Equation 4-5 where \( r_{s,l} \) is the \( l \)-th binary bit. A good value for uniformity is around 50%, which means the response from RO is well distributed between ‘0’ s and ‘1’ s.

\[
Uniformity = \frac{1}{n} \sum_{l=1}^{n} r_{s,l} \times 100\% \tag{4-5}
\]

Bit-aliasing estimates the uniformity of ‘1’ s and ‘0’ s in each bit in the responses across a group of chips of the same type. Bit-aliasing can be measured by calculating the inter-chip HW, as shown in Equation 4-6. A good value for bit-aliasing is 50%, which means each bit in the responses across a group of chip is well distributed between ‘0’ s and ‘1’ s. Uniformity and bit-aliasing are the parameters that can measure the randomness features in the responses generated.
Bit – aliasing = \frac{1}{m}\sum_{i=1}^{m} r_{s,t} \times 100\% \tag{4-6}

Finally, the new parameter, diverseness, is used to measure the range of frequencies in the different number of stages used in a ROPUF. Diverseness can be measured by calculating the standard deviation SD of the frequencies from each stage used in an ROPUF as shown in Equation 4-7, 4-8, and 4-9. h is the number of ROs used on a chip. \( f_{i,j} \) is the individual frequency for each RO. \( f_{i,j,q} \) is the \( q \)-th frequency sample of the \( j \)-th RO in the \( i \)-th chip. \( f_{avg} \) is the average frequency on a chip.

\[
Diverseness = \sqrt{\frac{1}{h-1}\sum_{j=1}^{h}(f_{i,j} - f_{avg})^2} \tag{4-7}
\]

\[
f_{i,j} = \frac{1}{q}\sum_{q=1}^{q} f_{i,j,q} \tag{4-8}
\]

\[
f_{avg} = \frac{1}{h}\sum_{j=1}^{h} f_{i,j} \tag{4-9}
\]

4.3 Experiment Setup

In this experiment, three Xilinx Spartan 2 XSA-100 boards are used. There are 600 CLBs on each chip, as shown in Figure 4-4 [19]. Each CLB contains two slices and each slice contains two Lookup Tables (LUTs). One stage in the RO occupies one LUT. One CLB is used for the 3-stage RO and two CLBs are used for the 5-stage and the 7-
stage RO. Six hundred 3-stage ROs and 300 5-stage and 7-stage ROs are mapped on each chip.

Figure 4-4: Xilinx Spartan 2 CLBs layout.

The FPGA area is divided into two parts, left and right (three hundred CLBs on each part). The experiment is run two times for each chip and RO stage. The first run occupied the right area with ROs and the left area with other circuits needed, such as MUX and counters. The blue boxes in Figure 4-4 show the occupied CLBs. As can be seen on the right side of Figure 4-4, 300 ROs occupy half of the FPGA. The other half of the FPGA is partially occupied by other logic used in FST. In the second run, the left area is switched for other logic and right area for ROs. For each RO, the frequency is recorded 10 times. Overall, 18,000 frequencies for 3-stage ROs and 9,000 frequencies for each 5-stage and 7-stage ROs are recorded.

Figure 4-5 shows the logic blocks for the FST test circuit. The challenge generator produces the inputs to MUX that activate one RO at a time. Each RO is activated for 0.4
ms, and there is a 0.1 ms time period before the next RO is activated. This reduces the noise in the form of heat that is generated from the adjacent CLB [20]. The RO is activated from the top and moves down to the bottom of each column of the CLBs. A 0.2 ms gap between the RO and counter activation allows the signal to stabilize before the measurement is started. The timing controller regulates all time intervals involved, such as the time interval for each RO being activated and the time interval for the counter to measure each RO.

Figure 4-5: FST circuit diagram.

Frequency is computed using Equation 4-10, where \( x \) is the cycle counts from each RO and \( y \) is the cycle counts for the 50 MHz reference clock. The preset value for \( y \) is set to be 7,000 cycles. That means the RO cycles are measured within a 0.14 ms
period. The accuracy of the measurement is 0.007 MHz/cycle which is adequate to record
the differences between frequencies generated from the ROs.

\[ f = x \times \frac{50}{y} \text{ MHz} \]  
\hfill (4-10)

4.4 Results and Analysis

Response bits from 4, 5, and 7-stage ROs are generated to calculate diverseness, uniformity, uniqueness, bit-aliasing, and reliability. The response bits are generated using the chain-like neighbor coding method where neighboring ROs are compared [6]. The first response bit generated from the comparison of RO1 is mapped in row 1 and column 1 of the CLB with RO2 mapped in row 2 and column 1 of the CLB. Equation 4-1 is used for comparison.

Table 4.1 shows the diverseness, uniformity, uniqueness, and bit-aliasing for 4, 5, and 7-stage ROs. The diverseness of frequencies for the 3-stage RO is the highest compared to 5 and 7-stage. The results in Table 4.1 clearly show that as the number of stages used in ROs is reduced, the diverseness of the RO frequencies increases. However, there is a limitation on the usability of ROs with low number of stages because each FPGA chip has a maximum operating frequency. The Spartan 2 FPGA family has the maximum operating frequency of 200 MHz [19]. The lowest number of RO stages that can be used on Spartan 2 FPGA is 4 because the average frequency generated from the 3-stage RO on Spartan 2 is 182.77 MHz. The average frequency generated from the 3-stage RO on Spartan 2 is 220 MHz, which exceeds the maximum operating frequency for Spartan 2. If n RO is produces a frequency beyond the operating frequency of the FPGA,
the frequency from the RO cannot be measured correctly. Frequencies generated from the RO for all stages are verified using the Agilent Logic Analyzer, where the RO output is connected directly to the output pin of the FPGA board [17].

A high diverseness of RO frequencies is good for ROPUF because it indicates that there are high amounts of frequency variations. High frequency variations are desirable for generating a high number of good CRPs which are discussed later in this section. For authentication in ROPUF applications, the challenge cannot be reused because this reduces the security level of ROPUF as the response bits traverse the open domain for verification and is susceptible to adversary attack [6]. This means that in order to make the ROPUF effective, ample number of CRPs is needed.

A good RO should exhibit high diverseness for RO frequencies and should also have good uniformity and uniqueness. As mentioned in Section 4.2, a good uniformity and uniqueness average is 50%. For the uniformity average, the 5-stage ROs have the highest value, and the 7-stage ROs have the lowest. However, the difference in uniformity between the two is only 0.87%. The average uniformity results for all stages used can be considered good as the values are close to 50%. High uniformity value means the secret bits generated are uniformly distributed between 1s and 0s which is a desired randomness characteristic.
Table 4.1: Uniqueness, bit-aliasing, diverseness, and uniformity for 3, 5, and 7-stage ROs.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Uniqueness (%)</th>
<th>Bit-aliasing (%)</th>
<th>Diverseness (MHz)</th>
<th>Uniformity (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>40.178</td>
<td>47.0228</td>
<td>1.9469</td>
<td>47.0228</td>
</tr>
<tr>
<td>5</td>
<td>34.5596</td>
<td>47.7146</td>
<td>1.2375</td>
<td>47.7146</td>
</tr>
<tr>
<td>7</td>
<td>40.5797</td>
<td>46.1539</td>
<td>0.736</td>
<td>46.1538</td>
</tr>
</tbody>
</table>

Table 4.1 shows the 3-stage and 7-stage ROs have better uniqueness than the 5-stage ROs for inter-chip measurements. Average uniqueness is obtained by comparing the responses generated from all three FPGA chips. The higher the differences between responses from each chip, the higher the value of uniqueness. It is important to make sure that the uniqueness is high because this indicates that the ROPUF could generate unique response from mass number of FPGA chips under the same challenge.

For bit-aliasing, the 5-stage RO has the highest percentage at 47.71%, and the 7-stage has the lowest percentage at 46.15%. Nevertheless, all stages have good bit-aliasing percentages that are close to 50%. Table 4.2 shows the diverseness frequencies for 4, 5 and 7-stage ROs for each FPGA chip used. The 3-stage ROs have the highest SD of RO frequencies value for all three chips used compared to the other stages. These results are consistent with the previous results presented in [17] where it was shown that as the number of stages used in an RO is reduced, the diverseness of RO frequencies obtained is higher. All three different FPGA chips showed the same pattern. It is found that the diverseness of RO frequencies increases as the number of stages in ROs is reduced.
Table 4.2: SD for Chip 1, Chip 2, and Chip 3.

<table>
<thead>
<tr>
<th>Stage</th>
<th>CHIP 1</th>
<th>CHIP 2</th>
<th>CHIP 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-Stage</td>
<td>2.117440</td>
<td>2.586673</td>
<td>1.136851</td>
</tr>
<tr>
<td>5-Stage</td>
<td>0.938292</td>
<td>1.878756</td>
<td>0.895488</td>
</tr>
<tr>
<td>7-Stage</td>
<td>0.828066</td>
<td>0.821414</td>
<td>0.558649</td>
</tr>
</tbody>
</table>

We also study the effect of the number of stages on a ROPUF based on the percentage of bit flip occurrences. To calculate the percentage of bit flip occurrences, responses are recorded at different environmental conditions. In this experiment, responses from 3, 5 and 7-stage ROs are generated at four different temperature settings, as shown in Table 4.3. The experiment is conducted in a temperature controlled test chamber. The frequencies from each RO are recorded 10 times at each temperature setting. The responses are generated by comparing the average RO frequencies obtained. The bit generation equation used is shown in Equation 4-1.

All responses obtained at various temperature settings are compared with the responses generated at room temperature. The results obtained are shown in Table 4.3. The lowest reliability is 97.32% at 0°C for 3-stage ROs. In this case 8 bits flipped out of 299 bits. The highest reliability is found to be 99.33% at 20°C for 4 and 7-stage ROs. In this case 4 bits flipped, out of 599 bits for the 3-stage ROs, and 2 bits flipped out of 299 bits for the 5-stage ROs. From Table 4.3, it can be observed that reducing the number of stages in ROs has no direct relationship with the percentage of bit flip occurrences.
Table 4.3: Reliability for Chip 3.

<table>
<thead>
<tr>
<th>ROs stage</th>
<th>0°C</th>
<th>20°C</th>
<th>45°C</th>
<th>70°C</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>98.1636</td>
<td>99.3322</td>
<td>98.9983</td>
<td>98.9983</td>
</tr>
<tr>
<td>5</td>
<td>97.3244</td>
<td>98.9967</td>
<td>98.9967</td>
<td>97.9933</td>
</tr>
<tr>
<td>7</td>
<td>98.3278</td>
<td>99.3311</td>
<td>98.9967</td>
<td>98.6622</td>
</tr>
</tbody>
</table>

We also investigate the relationship between the number of stages used in ROs and CRP generation. To do this, all possible comparison pairs need to be generated. It should be noted that there is a differences between a challenge and comparison pairs. A challenge is the selection of the comparison pairs to form a response bitstream. One challenge can consist of many comparison pairs depending on the design of the challenge and the length of the response. For example, a challenge that produces 128 bits response might have 128 comparison pairs.

Figure 4-6 shows the list of possible challenge formations. The first three response bits are generated from Pair 1, Pair 2, and Pair 3. Assume that comparison result for Pair 1, Pair 2, and Pair 3 are 1,0, and 1, then response bits are 101. The challenges for this response are the combination of the MUX inputs for Pair 1, Pair 2, and Pair 3 that are 0000 0001 0010. The number of possible challenges can be measure by n!/(n-r)!r! where n is the number of available comparison pairs and r is the number of response bits. As the number of available comparison pairs increases, the number of possible challenges will also increase.
The easiest way to generate all possible comparison pairs is by selecting a sort algorithm that has $O(n^2)$ complexity. As mentioned earlier, the comparison pairs generated need to be good. This means each comparison pair needs to pass a certain Frequency Difference Threshold (FDT). To determine the FDT, the frequency differences at all bit flip occurrences on all FPGA chips are checked. The maximum frequency difference that causes the bit flip is set as FDT. It is observed that the majority of bit flips occur when the frequency difference between ROs is 1 MHz and below. The maximum frequency difference where bit flips can occur is 3.5 MHz which is also the FDT.

Figure 4-6: Difference between comparison pairs and CRPs.
The pseudocode of the algorithm used to generate the various comparison pairs is shown below. The input to the algorithm are all RO frequencies generated at room temperature. The algorithm compares the frequency difference between one RO with the rest of the ROs available based on O(n^2) complexity. If the frequency difference passes the FDT, then those ROs are selected as the comparison pair.

**Comparison Pair Generation in pseudocode**

**Input:**
1) 600 frequencies for 3-stage ROs and 300 frequencies for 5 and 7-stage ROs represented as RO frequencies(i).
2) n is equal to the number of ROs.

**Output:**
1) The list of all possible ROs comparison pairs that passed the FDT represented as ROs comparison pair(k,i).

**Algorithm**
1. i <- 0, j <- 0, k <- 1
2. for i = 1 to n-1
3.     for j = i + 1 to n
4.         frequency difference = absolute (ROs frequencies(i)-RO frequencies(j))
5.     if frequency difference > FDT
6.         comparison pair(k,1) = i
7.         comparison pair(k,2) = j
8.     k++
9. end if
10. end for
11. End for

Table 4.4 shows the results obtained for comparison pair generation. It is observed that the highest number of comparison pairs are generated from 3-stage ROs on FPGA chip 2 at FDT equal to 2 MHz. The lowest number of comparison pairs are generated from 7-stage ROs on FPGA chip 1 and 3 at FDT value equal to 3 and 3.5 MHz. In general, Table 4.4 shows that the number of comparison pairs generated are higher when the number of stages used in RO is reduced. As the FDT increases the number of comparison pairs are reduced.
As mentioned earlier, the FDT used to filter all the bit flip occurrences is 3.5MHz. In Table 4.4, it is observed that the 7-stage RO is adversely affected by the higher value of FDT. The comparison pairs generated from 7-stage ROs are 302 on chip 2, and only 1 on chips 1 and 3. This shows that 7-stage ROs cannot be used in ROPUF as the lower number of comparison pairs generated diminishes the ROPUF application. For 5-stage ROs, the comparison pairs generated at FDT 3.5 MHz are very low for chips 1 and 3 (381 and 150). The 3-stage ROs have the highest number of comparison pairs that can be generated at FDT equal to 3.5 MHz.

### 4.5 Summary

This experiment was run on the Xilinx Spartan 2 FPGA chip that uses 180 nm semiconductor process technology. Therefore, conclusions are based on the Xilinx Spartan 2 FPGA and cannot be generalized on different FPGA technology. For the Spartan 2 FPGA chips, it can be concluded that the diverseness of RO frequencies

<table>
<thead>
<tr>
<th>FPGA Chip</th>
<th>ROs stage</th>
<th>FDT (MHz)</th>
<th>Number of Comparison Pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>2</td>
<td>2.5</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>45757</td>
<td>28746</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5955</td>
<td>2749</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1221</td>
<td>167</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>102800</td>
<td>95161</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>22287</td>
<td>18422</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1171</td>
<td>525</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>37932</td>
<td>23095</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>3811</td>
<td>2790</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>154</td>
<td>44</td>
</tr>
</tbody>
</table>
increases as the number of stages is reduced. The lowest number of stages that can be used in an RO is dependent on the operating frequency of the FPGA chip. For Spartan 2 FPGA, the maximum operating frequency is 200 MHz. Therefore, the lowest number of RO stages that can be used is 4 as the frequency produced from a 3-stage RO exceeds the maximum operating frequency. This chapter shows that the lower number of stages used in an RO does not compromise the uniqueness, uniformity, bit-aliasing, and reliability of the ROs. The relationship between the number of stages used in the ROs and CRPs is also established experimentally. It is found that more comparison pairs are generated when lower number of stages is used.
Chapter 5

A Novel RPM Technique for Minimize Systematic Variations

Because PUFs rely highly on process variations, the response bits generated are governed by the systematic process variations which reduce the randomness in the response bits. In this chapter, we describe a novel Random Patch Mixer (RPM) technique to minimize the systematic variation effects on the response bits. The RPM technique is applied on data obtained from FPGA chips. It is shown that the RPM technique successfully nullifies the systematic variation effect on the response bits generated by the ROPUF on FPGA. It also demonstrates that the responses generated after application of the RPM Technique pass the NIST statistical test for randomness [21].

5.1 Introduction

The ROPUF produces a stream of ‘1’s and ‘0’s based on the process variation on a silicon chip. The process variation is a random process that occurs during silicon chip fabrication and is caused by inaccuracy in the fabrication process. This inaccuracy produces a small delay that is not visible in the functional operation of the circuit on a silicon chip. The ROPUF magnifies this small delay through the frequency generation from a Ring Oscillator (RO). The difference in the frequencies generated by ROs is used
to generate a random binary bit stream, which is used for authentication or producing a cryptographic encryption and decryption key [6]. The random binary bit stream is known as the response. Each response is generated by a given challenge from the user. A challenge is a binary bit stream that selects the RO pairs for comparison. Each challenge produces a unique response.

The response generated from the ROPUF needs to be truly random for it to be used for authentication or developing the cryptographic key. One way to verify true randomness is by applying the NIST statistical test for randomness. Generating true random response is one of the main challenges in a ROPUF. By default, a ROPUF does not produce a true random response because the process variation is not completely random [8]. In silicon chips, there are two types of variations, systematic and stochastic [13]. Systematic variation is caused by process and equipment non-uniformity, dissimilar interactions between circuit layout and chemical mechanical polishing process, and the gradient of thermal annealing [13][27][28]. A stochastic variation is caused by the random component that accounts for the difference between the observed data and the model estimates. These include atomic-level stochastic phenomena, such as random dopant profiles, any unidentified patterns, and measurement errors [3][28][29]. It seems that systematic variation is more prevalent than stochastic variation in a ROPUF [8].

A systematic variation has a direct link with the true randomness in the response generated from a ROPUF. It has been shown that the immediate output from a ROPUF fails the NIST statistical test for randomness [29]. Therefore, the responses generated from an ROPUF are not truly random [29]. Amongst the several techniques used to deal with systematic variation is the regression based distiller [29]. The regression based
distiller is based on the polynomial regression and is applied before the secret selection step. The regression based distiller has high computational cost when implemented on the hardware.

In this chapter, a new Random Patch Mixer (RPM) technique used to cancel the systematic variation effect on an FPGA chip is developed.

5.2 Systematic Variations and Bit Flip Minimization

5.2.1 The Effect of Systematic Variations

Shown in Figure 5-1 systematic variations cause neighboring frequencies to be correlated to each other. The graph shows a repeating pattern. Response bits are generated by comparing the neighboring ROs, RO-n and RO-n+1. The response is 1 if RO-n is greater than RO-n+1, otherwise the response is 0. For RO-1 until RO-20, the response A is 00110000100100100101, and for RO-21 until RO-40, the response B is 01010100110110100101. RO-1 to RO-20 are mapped on the first column of the CLBs and RO-21 to RO-40 are mapped on the second column of the CLBs, as shown in Figure 5-2. The hamming distance for the two responses is 5, which is very low. This implies that response A and B, are 75% similar to each other. This reduces the randomness of the responses.

5.2.2 Regression Based Distiller

One way to normalize the systematic variation effect on frequency distribution is to apply the regression-based distiller technique as proposed in [13]. This technique uses polynomial regression to capture the systematic variation. The regression-based distiller
technique eliminates the systematic variation effect for polynomial regression of order 2 and above. The regression-based distiller has been tested on several challenge selection techniques such as S-sequence, T-sequence, 1-out-of-8 coding and neighbor coding. The responses generated from all 4 techniques are evaluated using the NIST randomness test, but none of the responses fully pass the tests. Nonetheless the response from S-sequence, T-sequence and 1-out-of-8 coding pass most of the tests. Only the neighbor coding technique failed the entire test.

Figure 5-1: Frequencies across columns 1-3 of CLBs.
The regression-based distiller also incurs more computational cost. It can be seen that majority of the challenge selection techniques require a polynomial regression of order 2 or above to pass the NIST randomness test.

5.2.3 RPM Technique

The RPM technique is based on the uniform random number generated from the pseudorandom number generator. There are three steps involved in this technique:

a) Generation of N uniform random numbers that range from 0 to 1 (N is equal to number of ROs) as shown in Equation 5-1.

b) Normalization of the random numbers generated to the maximum value of RO frequency difference from the average RO frequency on an FPGA chip as given by Equation 5-2, 5-3, and 5-4. The normalized random numbers will be the *Patch* of the RO frequencies.
c) Addition of Patch to the RO frequencies. We use Equation 5-5 to determine how the Patch can be added to the RO frequencies.

\[
PRN x = x_i, x_{i+n}, \ldots x_n \{i = 1,2,3..n\} \ 0 < x < 1 \tag{5-1}
\]

\[
f_{avg} = \frac{1}{n} \sum_{i=1}^{n} f_i \tag{5-2}
\]

\[
a = \max\{f_i - f_{avg}, f_{i+1} - f_{avg}, \ldots f_n - f_{avg}\} \tag{5-3}
\]

\[
x^\wedge = a\{x_i, x_{i+1}, \ldots, x_n\} \tag{5-4}
\]

\[
f'_i = \begin{cases} 
  x^\wedge_i + f_i, & \text{if } f_i < f_{avg} \\
  x^\wedge_i - f_i, & \text{otherwise}
\end{cases} \tag{5-5}
\]

The RPM technique is designed to be simple and easy to implement on the ROPUF circuit. The technique utilizes the normalized pseudorandom number (Patch) to improve the frequency distribution randomness. The Patch used is different for each of the FPGA chips used. The Patch is stored in the memory as part of the ROPUF circuit. A concern with ROPUF security is whether or not an adversary could predict the responses if he knows Patch of a certain ROPOF. In a later section, the responses generated after the RPM technique has been applied, will be shown to have no correlation with the Patch; hence, the responses cannot be predicted from a Patch.

5.3 Results and Discussion

In this section, the RPM technique is applied to the frequencies obtained from 3-stage ROs on 29 Spartan 3E FPGA chips. The Spartan 3E chip has 240 CLBs and each
CLB has 4 slices with two LUTs on each slice. Each stage occupies one LUT. Thus, a 3-stage RO can be fitted in one CLB. A total of 240 3-stage ROs are mapped on each FPGA chip.

5.3.1 Systematic Variation Effect on Frequency Distribution

Figure 5-3 shows the frequency distributions across Spartan 3E FPGA 1, 2, and 3 for 3-stage ROs. In Figure 5-3 (a), the ROs on the left side region of the FPGA chip 1 produce lower frequencies, and the ROs in the center region produce higher frequencies. It can be observed that the frequency of each RO is close to the neighboring ROs’ frequency. A similar trend can be observed on all three FPGA chips shown in Figure 5-3.
Figure 5-3: 3-stage RO frequencies across Spartan 3E.
(b) Board 2
(c) Board 3
It is observed that for the 29 Spartan 3E FPGA chips, high frequency ROs are grouped mostly in the center region and low frequency ROs are distributed around the high frequency ROs’ region. Figure 5-3 shows the effect of systematic variation on the ROs frequency distribution.

5.3.2 Systematic Variations Minimization by RPM Technique

Figure 5-4 shows the ROs frequency distribution after the RPM technique is applied to FPGA chips 1, 2, and 3. It can be observed that the RPM technique efficiently increases the frequency distribution randomness and minimizes the systematic variation effect on the frequency distribution across FPGA chips 1, 2, and 3. There are no more low and high frequency regions, as can be observed in Figure 5-3.

Figure 5-4: 3-stage RO frequencies across Spartan 3E FPGA after RPM technique has been applied.

(a) Chip 1
Figure 5-4: 3-stage RO frequencies across Spartan 3E FPGA after RPM technique has been applied.
(b) Chip 2
(c) Chip 3
Responses from FPGA Spartan 3E chip 1 before and after the RPM technique is applied are compared to verify that the RPM technique successfully minimizes the systematic variation effect on the responses generated. A response is generated using neighbor coding. A total of 65 response bits are generated from each FPGA chip, as shown in Table 5.1 and 5.2. The response bits are generated from the center part of the Spartan 3E FPGA where the systematic variation is visibly present. The correlation for responses before and after the RPM technique is applied is compared by measuring the hamming distance (HD) percentage between each CLB column (comparisons are made for CLB columns 7 until 11). For this comparison, an acceptable HD percentage is 50% or higher, which means the responses generated between the CLBs columns have around 50% dissimilarity. Table 5.1 shows the response bits generated from each CLBs column (table on the left) and the HD between neighboring CLBs column (table on the right) before the RPM technique is applied. The results in Table 5.1 show that the HD percentages for CLBs column 7-8 and 8-9 are low which represents the effect of the systematic variation.

Table 5.2 shows the response bits generated from each CLBs column (table on the left) and the HD between neighboring CLBs column (table on the right) after the RPM technique is applied. The results in Table 5.2 show how the RPM technique successfully minimized the systematic variation effect on the response generated. The HD percentage for CLBs column 7-8 is increased from 15.38% to 46.15% and for CLBs column 7-8 is increased from 30.77% to 53.85%
## Table 5.1: Hamming Distance (HD) for FPGA chip 1 before RPM is applied.

<table>
<thead>
<tr>
<th>CLBs Column</th>
<th>Hamming Distance HD between CLBs Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 8 9 10 11</td>
<td>7-8 8-9 9-10 10-11</td>
</tr>
<tr>
<td>0 0 0 0 0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>1 1 1 1 1</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>0 0 0 0 1</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>1 0 0 1 0</td>
<td>1 0 1 1</td>
</tr>
<tr>
<td>1 1 1 0 1</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>0 0 1 0 0</td>
<td>0 1 1 1</td>
</tr>
<tr>
<td>1 1 0 1 0</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>0 0 0 0 1</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>1 1 1 1 0</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>HD Percentage %</td>
<td>15.38 30.77 46.15 61.54</td>
</tr>
</tbody>
</table>

## Table 5.2: Hamming Distance (HD) for FPGA chip 1 after RPM is applied.

<table>
<thead>
<tr>
<th>CLBs Column</th>
<th>Hamming Distance HD between CLBs Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 8 9 10 11</td>
<td>7-8 8-9 9-10 10-11</td>
</tr>
<tr>
<td>1 1 1 1 0</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>1 0 0 1 0</td>
<td>1 0 1 1</td>
</tr>
<tr>
<td>0 0 0 1 1</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>1 1 0 0 0</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>0 0 1 0 1</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>0 1 0 1 1</td>
<td>0 1 0 1</td>
</tr>
<tr>
<td>1 0 1 0 1</td>
<td>1 1 0 1</td>
</tr>
<tr>
<td>0 0 1 0 1</td>
<td>0 1 1 1</td>
</tr>
<tr>
<td>0 1 0 1 0</td>
<td>0 1 1 1</td>
</tr>
<tr>
<td>1 0 0 1 1</td>
<td>1 0 1 1</td>
</tr>
<tr>
<td>0 1 1 0 0</td>
<td>1 0 1 1</td>
</tr>
<tr>
<td>HD Percentage %</td>
<td>46.15 53.85 76.92 53.85</td>
</tr>
</tbody>
</table>
The responses from Patch values for each FPGA chip used are generated to measure the correlations between a Patch and the responses generated from the ROs frequency after applying the RPM technique. A total of 239 bits are generated from each Patch and ROs’ frequency. This is done to ensure that no correlation exists between the Patch and response generated. If there is a correlation between the Patch and response generated, the level of ROPUF security is compromised. HD percentage is used to measure the correlation between the response generated from the Patch and the ROs’ frequency. The average HD obtained for 29 FPGA chips is 49.97%. Therefore, there is no correlation between the Patch and the response.

5.3.3 NIST Statistical Test for Randomness

The responses generated from a ROPUF need to be truly random to ensure a good security level for the ROPUF. The NIST Statistical Test for Randomness can be used to measure the randomness feature inside the response generated from a ROPUF. In this research, we tested responses generated by using neighbor coding, and an 8-to-1 selection technique for 29 FPGA chips.

A total of 240 ROs are used to generate the responses. Results obtained for the NIST statistical test for randomness are shown in Table 5.3. The responses generated after applying the RPM technique by using the Neighbor Coding selection passed the entire test except for ‘runs’ test. The responses generated from the 8-to-1 selection passed all the tests.
Table 5.3: NIST statistical test for randomness results.

<table>
<thead>
<tr>
<th>Statistical test</th>
<th>NIST input parameters</th>
<th>Neighbor Coding</th>
<th>8-to-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td></td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>BlockFrequency</td>
<td>128</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>CumulativeSums</td>
<td></td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Runs</td>
<td></td>
<td>Failed</td>
<td>100</td>
</tr>
<tr>
<td>LongestRun</td>
<td></td>
<td>60</td>
<td>100</td>
</tr>
<tr>
<td>FFT</td>
<td></td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>NonOverlappingTemplate</td>
<td>9</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>OverlappingTemplate</td>
<td>9</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>ApproximateEntropy</td>
<td>4</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Serial</td>
<td>16</td>
<td>80</td>
<td>100</td>
</tr>
<tr>
<td>LinearComplexity</td>
<td>500</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

The results obtained are better than the regression-based distiller technique results [3]. None of the polynomial distiller order makes meaningful improvement in the randomness for the neighbor coding selection. For the 8-to-1 selection, the responses passed all of the tests only when the 4th order distiller is applied.

5.4 Summary

As a ROPUF utilizes the process variations to generate the secured response bits, vulnerability still exists. The systematic variations dominated the overall process variations; therefore it is important to nullify the systematic variation effect and increase the true randomness on the response generation in a ROPUF. To address this issue, we propose our RPM technique which gives better results in terms of the response randomness generated from the ROPUF.
Chapter 6

Temperature, Voltage, and Aging Effects on ROPUFs Function

6.1 Introduction

Silicon physical unclonable functions (PUFs) take advantage of the random process variations inherent in silicon chips. The random process variations are unique for each chip and cannot be modeled. This randomness and uniqueness characteristics of silicon chips have been exploited by researchers in designing PUFs for hardware security. As PUFs are highly reliant on process variations in the chip, it is desirable that they should be resilient towards other temporal changes. The changes may occur due to exposure to temporal variabilities which can be observed in the frequency of the Ring Oscillator. The temporal variabilities can be divided into reversible and irreversible variabilities. The reversible variability causes temporary changes to the circuit’s behavior inside the silicon chip and can be caused by the environmental variations such as voltage and temperature. However, overexposure to high voltage and temperature may lead to irreversible variability [30]. Irreversible variability can also be caused by silicon chip aging. There are three types of aging effects, the first one is the hot carrier injection or HCI, the second is the trap charge in the dielectric due to bias temperature instability, and the third is the oxide breakdown due to electrically active defects known as traps. These traps occur within the dielectric. In this study, we simulate the first type of aging that is
caused by HCI. The HCI causes the electrical charges to build up within the dielectric layer thereby increasing the threshold voltage needed to turn the transistor on. The increased threshold voltage results in increased transistor switching time, thus slowing the transistor speed [31].

In this chapter, we present accelerated aging experimental results along with temperature and voltage effects done under normal environmental conditions on 9 Spartan 3E FPGAs. ROs having 3, 5, and 7-stages are mapped on Spartan 3E FPGAs. Voltage and temperature variation experiments are performed separately (3 and 5-stage ROs).

6.2 Background

We briefly describe work done in the past to study the effect of temperature, voltage, and aging on ROs. Accelerated aging experiment for 5-stage ROs mapped on 90 nm FPGAs is presented in [32]. In this study, it is observed that aging causes ROPUF responses to be unreliable. Simulated aging on ROs using HSPICE is presented in [33]. It is observed that 4% of the ROPUF bits are prone to instability due to aging. Temperature and voltage effects on ROs have been analyzed in [11]. This study concludes that ROPUF reliability reduces due to voltage and temperature variations. Whereas prior work is focused on studying the effect of temporal changes for fixed stage ROPUFs, we analyze the effect of temporal changes on ROPUFs having different stages.
6.2.1 Ring Oscillator PUF response

RO frequency is typically generated from a series of inverters comprising 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.

6.2.2 Number of Stages in Ring Oscillator

In this experiment, ROs having three different stages are used. The 5-stage RO consists of one NAND gate and 4 inverter gates as shown in Figure 4-1. The NAND gate is used to control the on and off switching of the RO. The RO is activated (starts to produce an oscillation) when the input is set to high. The 3-stage RO consists of one NAND gate, one buffer, and two inverter gates. The buffer is used instead of the inverter to obtain an odd number of inversions. The buffer gate is added to increase the total delay in the RO in order to reduce the RO frequency. The 7-stage RO consists of one NAND gate and 6 inverters.

6.3 Experimental Setup

The experimental circuitry is shown in Figure 4-5. The challenge generator is used to produce the inputs to the MUX which activates one RO at a time. ROs are activated, one at a time, from the top to the bottom of each column of the FPGA. Each RO is
activated for 0.4 ms. There is a 0.1 ms delay before the next RO is activated; this is to reduce the noise in the form of heat that can be generated from the adjacent CLB [20]. A 0.2 ms delay gap is given between the RO and the counter activation for the signal to be stabilized before the measurement starts. The timing controller controls all time intervals involved, such as the time interval for the RO activation and the time interval for the counter to measure each RO.

The frequency is computed using $x \times (50/y)$, where $x$ is the cycle counts from each RO and $y$ is the cycle counts for the 50 MHz reference clock. The preset value for $y$ is set to 10000 cycles implying that the RO cycles are measured within a 0.2 ms period. The accuracy of the measurement is 0.005 MHz/cycle which is good enough to measure the differences between frequencies generated from the ROs.

For the accelerated aging experiment on Spartan 3E, each RO is activated every 64 ms. Each activation turns on the RO for a time period of 0.4 ms. Thus, each RO is activated 1.3 million times a day. This aging experiment is conducted for 30 days. The number of ROs mapped on Spartan 3E is 120. ROs are numbered according to the location they are mapped on the Spartan 3E as shown in Figure 6-1. Responses are generated by using a neighbor chain scheme where RO(n) is compared with RO(n+1). In total, there are 119 response bits generated from 120 ROs.
6.4 Results and Analysis

Table 6.1 shows the total number of bit flip occurrences for Spartan 3E FPGAs for the 30 day aging period. Responses from all FPGAs are recorded once every day. Thus 30 responses are recorded for 30 days from each FPGA. These responses are compared to the responses generated at normal setting to measure the bit flip occurrences. The total bit flip occurrences for 3, 5, and 7 stage ROs are found to be 192, 250 and 267, respectively (three FPGAs are used for each RO stage). FPGA 3 has the lowest number of bit flip occurrences of 18. This is because many of the RO comparison pairs in FPGA 3 have a high frequency difference. On the other hand, FPGA 9 has the highest number of bit flip occurrences since many of the RO comparison pairs have a low frequency difference. Figure 6-2 (a) shows the example of the bit flip occurrence when the difference between RO comparison pair is small (below 1 MHz) [6]. The frequency...
generated from the blue RO tends to reduce faster when compared to the green RO’s frequency when the temperature is increased, therefore bit flip occurs. Figure 6-2 (b) shows how the bit flip occurrence can be prevented by selecting an RO comparison pair that has higher frequency difference. It is important to note that most of the bit flips occur at the same bit locations which have lower frequency difference in the RO pairs.

Table 6.1: Bit flip occurrences on Spartan 3E.

<table>
<thead>
<tr>
<th></th>
<th>3-stage ROs</th>
<th>5-stage ROs</th>
<th>7-stage ROs</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-stage ROs</td>
<td>FPGA</td>
<td>FPGA</td>
<td>FPGA</td>
</tr>
<tr>
<td>1</td>
<td>88</td>
<td>86</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>75</td>
<td>78</td>
<td>97</td>
</tr>
<tr>
<td>3</td>
<td>82</td>
<td>82</td>
<td>103</td>
</tr>
<tr>
<td>Total</td>
<td>192</td>
<td>250</td>
<td>267</td>
</tr>
</tbody>
</table>

Figure 6-2: The relationship between the RO frequency distance and the probability of a PUF output flip.

Table 6.2 shows the bit flip occurrences for 3, 5, and 7-stage ROs with respect to frequency differences in RO pairs. For 5 and 7-stage ROs, most of the bit flips occur when the frequency difference in RO comparison pairs is lower than 0.4 MHz. Few bit flips occur when the frequency difference lies between 0.3 and 0.7 MHz. For the 3-stage ROs, maximum bit flips occur when the frequency difference is lower than 0.3 MHz.
There are some bit flips at the higher frequency range. These results suggest that the 3-stage ROs are more susceptible to noise compared to 5 and 7-stage ROs. For the maximum frequency difference (1.0-1.2 MHz), the number of bit flips in the 3-stage ROs is 8 which still can be considered as low since 3-stage ROs have a standard deviation of 1.9 MHz compared to 1.2 and 0.7 MHz for 5 and 7-stage ROs, respectively [34]. High standard deviation in ROs implies that the range between the minimum and maximum frequency used in the ROPUF is high. Therefore, many RO comparison pairs that have high frequency differences are generated.

Figure 6-3 (a), (b), and (c) show the frequency changes due to aging effects for 10 different ROs for 3, 5, and 7-stages. It can be seen that the 3, 5, and 7-stage ROs have minimal frequency fluctuations for the 30 day aging period. Some frequencies overlap (e.g. RO1 and RO2, RO4 and RO9 in Figure 6-3 (b)). This illustrates how bit flips can occur. It can also be seen from Figure 6-3 that there is no significant difference in frequency fluctuations, as a result of aging, when different number of stages are used in the ROPUF.
Table 6.2: Bit flip occurrences due to aging on Spartan 3E.

<table>
<thead>
<tr>
<th>RO Pairs’ Frequency Differences Ranges (MHz)</th>
<th>Bit Flip Occurrences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3-stage</td>
</tr>
<tr>
<td>0-0.09</td>
<td>46</td>
</tr>
<tr>
<td>0.1-0.19</td>
<td>42</td>
</tr>
<tr>
<td>0.2-0.29</td>
<td>46</td>
</tr>
<tr>
<td>0.3-0.39</td>
<td>11</td>
</tr>
<tr>
<td>0.4-0.49</td>
<td>12</td>
</tr>
<tr>
<td>0.5-0.59</td>
<td>7</td>
</tr>
<tr>
<td>0.6-0.69</td>
<td>5</td>
</tr>
<tr>
<td>0.7-0.79</td>
<td>12</td>
</tr>
<tr>
<td>0.8-0.89</td>
<td>2</td>
</tr>
<tr>
<td>0.9-0.99</td>
<td>1</td>
</tr>
<tr>
<td>1.0-1.2</td>
<td>8</td>
</tr>
</tbody>
</table>
Figure 6-3: Frequency changes in ROs due to the aging effect on Spartan 3E.
(a) 3-stage ROs
(b) 5-stage ROs
(c) 7-stage ROs
Figure 6-4 (a) and (b) show the frequency changes with respect to the temperature variations for 40 ROs that are mapped at different spatial locations on the same FPGA chip for 3 and 5-stage ROs. Responses are generated at three different environment temperatures, namely, room temperature, 45°C and 70°C. Different temperatures are generated using temperature chamber as shown in Figure 6-6. It is observed that the ROs are sensitive to the temperature variations. As the environment temperature increases, both 3 and 5-stage RO frequencies decrease uniformly. Similar patterns are observed for all RO frequencies at each of the three different environment temperatures which suggests that the effect of the temperature variations are uniformly distributed on the FPGA chip.

Figure 6-5 (a) and (b) show the frequency changes with respect to the voltage variations for 40 ROs that are mapped on different locations (as shown in Figure 6-1) in the same FPGA chip for 3 and 5-stage ROs. Responses are generated for three different internal core supply voltages ($V_{CCINT}$), namely, 1.2V (normal), 1.3V and 1.4V. It is observed that both 3 and 5-stage RO frequencies are sensitive to the voltage variations. The average frequency increment is 20 MHz for 1.3V and 50 MHz for 1.4V when compared to the normal 1.2V $V_{CCINT}$. Although the frequency has high increment with respect to the higher $V_{CCINT}$, the RO frequency for 3 and 5-stage ROs follow the same pattern which implies the voltage variation effect is uniformly distributed throughout the FPGA chip. The bit flips do occur due to temperature and voltage variations but only when the frequency difference in the RO comparison pair is lower than 1.5 MHz.
Figure 6-4: Frequency changes with respect to the temperature variations on Spartan 3E.
(a) 3-stage ROs
(b) 5-stage ROs
Figure 6-5: Frequency changes with respect to the voltage variations on Spartan 3E.
(a) 3-stage ROs
(b) 5-stage ROs
Table 6.3 shows the percentages of bit flip occurrences due to temperature, voltage variations, and aging on 9 Spartan 3E FPGAs. For temperature variations, the responses from ROPUF are generated at three different settings: room temperature, 45°C, and 70°C. For voltage variations, the responses are generated using three different internal core supply voltages: 1.2V (normal), 1.3V, and 1.4V.

Responses generated at different temperature and voltage settings are compared with the responses generated at normal settings to measure the percentage of bit flip
occurrences. As the number of stages increases, the percentage bit flip occurrences also increase (except at 70°C). Voltage variations seem to be causing the most bit flip occurrences followed by temperature and aging. The maximum bit flip percentages for 3, 5, and 7-stage ROPUFs are 2.8%, 5.6%, and 8.4%, respectively. Based on our experimental results, we conclude that bit flips occur only when the frequency difference in the RO comparison pair is lower than 1.5 MHz.

Table 6.3: Percentage of bit flip occurrences.

<table>
<thead>
<tr>
<th>RO number of stages</th>
<th>Temperature</th>
<th>Voltage</th>
<th>Aging</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>45°C 70°C</td>
<td>1.3V 1.4V</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2.24 1.96</td>
<td>2.24 2.80</td>
<td>1.79</td>
</tr>
<tr>
<td>5</td>
<td>2.52 3.92</td>
<td>5.88 5.60</td>
<td>2.33</td>
</tr>
<tr>
<td>7</td>
<td>4.20 3.08</td>
<td>7.28 8.40</td>
<td>2.46</td>
</tr>
</tbody>
</table>

The results presented in this chapter suggest that the temporal variabilities can affect the ROPUF functionality only if the frequency difference between RO comparison pair is low. We propose that a high threshold be used to select the RO comparison pairs in ROPUF to prevent the effect of temporal variabilities. Table 6.4 shows the number of RO comparison pairs generated based on different frequency thresholds. The comparison pairs are generated using select sort algorithm that has $O(n^2)$ complexity [34].

The higher number of RO comparison pairs are also required for better security [34]. The results in Table 6.4 suggest that the 3-stage RO has better security feature as it has the highest number of RO comparison pairs compared to the 5 and 7-stage ROs.
Table 6.4: Number of comparison pairs according to threshold frequency.

<table>
<thead>
<tr>
<th>ROs stage</th>
<th>Threshold Frequency (MHz)</th>
<th>Number of RO comparison pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
<td>4671</td>
</tr>
<tr>
<td>3</td>
<td>2.5</td>
<td>4158</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3682</td>
</tr>
<tr>
<td>3</td>
<td>3.5</td>
<td>3152</td>
</tr>
<tr>
<td>5</td>
<td>4043</td>
<td>3446</td>
</tr>
<tr>
<td>7</td>
<td>3128</td>
<td>2344</td>
</tr>
<tr>
<td>7</td>
<td>3.5</td>
<td>2860</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>2336</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>3128</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>1701</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>1191</td>
</tr>
</tbody>
</table>

6.5 Summary

In this chapter, we study the effect of accelerated aging, voltage, and temperature variations for different number of stages used in a ROPUF. Our experimental results show that RO frequencies are sensitive to aging, voltage, and temperature regardless of the number of RO stages used in a ROPUF. The percentage of bit flips is observed to increase as the number of stages increase. Most bit flips occur when the frequency difference between RO comparison pairs is low. We suggest that only RO comparison pairs that have high frequency differences be used in a ROPUF in order to reduce temporal variabilities. Our work shows that the 3-stage ROPUF has the lowest percentage of bit flip occurrences and is more secure.
Chapter 7

A Comparative Study of Ring Oscillator PUFs on Different FPGA Families

7.1 Introduction

ROPUF utilizes ring oscillators (ROs) to exploit the process variation inside a silicon chip to generate a unique ID. A typical ROPUF comprises of ring oscillators, multiplexers (MUXs), counters, and a comparator. A ROPUF can generate a binary bit stream (response) from a given input bit stream (challenge). A ROPUF can generate multiple sets of responses from different sets of challenges. A challenge that produces a response is known as a challenge-response pair (CRP).

Earlier studies have shown that ROPUF can be implemented on FPGAs [3][4]. The fact that ROPUF circuits do not need to be symmetric compared to other types of PUFs, such as Arbiter PUF (APUF) and Butterfly PUF (BPUF) that require a stringent symmetric circuit makes the ROPUFs attractive. The only requirement in ROPUF circuit is that the ROs need to be identical, and this can be achieved by creating a hard macro for the RO and instantiating it as many times as needed. If two ROs are identical then the difference in the frequencies generated is due to process variation.

FPGA security is a concern among the FPGA manufacturers. FPGAs are prone to several security issues such as IP protection, cloning, side channel attack and tampering. Xilinx, for example, has introduced Device DNA as the additional security feature in
some of its FPGA chips [45]. ROPUF can be used as an additional security feature in FPGAs. Any tampering attempt by hackers will change the unique parameters of the process variation [3]. ROPUF can be applied as a secret bit generator (which is known as response in PUF applications) where it can generate $n$ bits of response for authentication purpose. Besides that, response bits generated from ROPUF can be applied as a cryptography key to encode and decode secure information [6].

Despite the promising solution offered by ROPUF, there are still challenges that need to be overcome for ROPUF to become a practical solution. Making the ROPUF response better in uniqueness and increasing in reliability are among the challenges. Uniqueness refers to the ability of similar ROPUF circuits to generate unique responses on different chips. Reliability refers to the generation of same response under various environmental conditions such as temperature and voltage. ROPUF’s reliability can also be affected by silicon aging.

Current FPGA families are fabricated using the latest silicon technology which provides smaller transistor size. Smaller transistors size gives better performance on the FPGA chip in terms of speed and power consumption, but in terms of the performance of ROPUF implementation on FPGA, it still needs to be studied [6]. As the silicon technology shrinks, the process variation parameters will also change [13]. In this work, we analyze ROPUF parameters on two different Xilinx FPGA families that use different silicon technologies; 28 nm technology (Artix 7) and 90 nm technology (Spartan 3E). The work focuses on:

1) ROPUF’s comparison on two FPGA families that use different silicon technologies: We compare ROPUF’s responses from two different
FPGA families in terms of five parameters; uniqueness, reliability, uniformity, bit aliasing, and diverseness.

2) Temperature, voltage, and aging effects: For reliability, we compare the responses generated at different temperature and voltage settings. We also compare the responses generated through an accelerated aging experiment.

7.2 Related Work

Some work has been done in the past to study ROPUF performance on FPGA. Large scale characterization of ROPUF on Spartan 3E (90 nm silicon technology) FPGAs has been done in [11]. They show that the average inter-die hamming distance (HD) for ROPUF is 47.31% and the average intra-die HD is 0.86% at normal operating condition. The hamming weight (HW) for ROPUF responses is shown to lie between 46% and 56%. In [6], implementation of ROPUF on Virtex 4 (90 nm silicon technology) FPGAs is presented. It is shown that the inter-chip HD is 46.15%. The accelerated aging experiment on 5-stage ROs mapped on Spartan 3E FPGAs is presented in [46]. It is observed that aging causes ROPUF responses to be unreliable. Simulated aging on ROs using HSPICE is shown in [47]. It is observed that 4% of the ROPUF bits are prone to instability due to aging. The experiment on temperature and voltage effects on ROs is presented in [11]. It is shown that ROPUF reliability reduces due to voltage and temperature variations.
7.3 Background

7.3.1 Ring Oscillator PUF response

RO frequency is typically generated from a series of inverters comprising 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.

7.3.2 Number of Stages in Ring Oscillator

In this experiment, 5-stage ROs are used. The 5-stage RO consists of one NAND gate and 4 inverter gates as shown in Figure 4-1. The NAND gate is used to control the on and off switching of the RO. The RO is activated (starts to produce an oscillation) when the input is set to high.

7.3.3 ROPUF parameters

For PUF implementations, different researchers have used different parameters in the past [11][34][48]. In this work, we use five of the most common parameters. These parameters are uniqueness, reliability, uniformity, bit-aliasing and diverseness. The uniqueness can be measured by comparing the Hamming Distance (HD) between responses from different FPGA chips in the same family. The equation used to measure the uniqueness is shown in Equation 7-1:
Uniqueness = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(R_u,R_v)}{n} \times 100\%  \quad (7-1)

where, \( m \) is the number of chips used, \( u \) and \( v \) are the two chips being compared, and \( n \) is the number of responses generated. \( R_u \) and \( R_v \) are the response from the same challenge \( C \) for chips \( u \) and \( v \). \( HD \) is the hamming distance between the responses generated from chips \( u \) and \( v \). The higher uniqueness percentage represents the better uniqueness in the response generated from ROPUF. But considering the large number of response bits, a good uniqueness percentage should be around 50\%. This means that at least 50\% of the responses generated from chip \( u \) and \( v \) differ from each other (responses obtained by given the same challenge to chip \( u \) and \( v \)).

The reliability can be measured by comparing the response from the same FPGA chip that is generated under different environmental conditions such as temperature and voltage. The equation used to measure the reliability is shown in Equations 7-2 and 7-3. \( R_s \) is the response from chip \( i \) at normal operating condition (at room temperature and normal operating voltage). \( R_{s,t} \) is \( t \)-th sample of \( R \)'s response from chip \( i \) at a different operating condition such as different temperature and voltage settings. A good reliability value is 100\%. As can be seen in Equation 7-3, if the HD intra is low or zero, then the reliability will be around 100\%.

\[
HD\,Intra = \frac{1}{k} \sum_{t=1}^{k} \frac{HD(R_s,R_{s,t})}{n} \times 100\% \quad (7-2)
\]

\[
Reliability = 100\% - HD\,Intra \quad (7-3)
\]
The uniformity and bit-aliasing parameters can be measured by using Hamming Weights (HWs) as shown in Equation 7-4 and 7-5 where \( r_{s,l} \) is the \( l \)-th binary bit. The HW of the response from an FPGA chip represents the uniformity and the HW of the responses from different FPGA chips represents the bit-aliasing. The HW for bit aliasing is measured across the same bit location in responses from different FPGA chips. A good value for uniformity and bit aliasing is around 50%, which means the response from RO is well distributed between ‘0’ s and ‘1’ s.

\[
Uniformity = \frac{1}{n} \sum_{l=1}^{n} r_{s,l} \times 100\% \quad (7-4)
\]

\[
Bit \text{- aliasing} = \frac{1}{m} \sum_{l=1}^{m} r_{s,l} \times 100\% \quad (7-5)
\]

The diverseness of the frequency can be measured by using standard deviation as shown in Equation 7-6, 7-7 and 7-8. Diverseness represents the range of the frequency generated from the ROs. A ROPUF’s diverseness that has a value which is close to 0 shows that the ROs’ frequencies tend to be very close to the ROs’ mean frequency. A high diverseness shows that the frequencies are spread out over a wider range of values. The advantages of having higher diverseness have been discussed in detail [34]. In equation 7-6, \( h \) is the number of ROs, \( f_{i,j} \) is frequency for each RO, \( f_{i,j,q} \) is the \( q \)-th frequency sample of the \( j \)-th RO in the \( i \)-th chip. \( f_{avg} \) is the average frequency of the ROs on an FPGA chip.

\[
Diverseness = \sqrt{\frac{1}{h-1} \sum_{j=1}^{h} (f_{i,j} - f_{avg})^2} \quad (7-6)
\]
\[ f_{i,j} = \frac{1}{q} \sum_{q=1}^{q} f_{i,j,q} \]  

(7-7)

\[ f_{avg} = \frac{1}{h} \sum_{j=1}^{h} f_{i,j} \]  

(7-8)

7.4 Experimental Setup

In this work, ROPUF performance on 29 Xilinx Spartan 3E and 20 Xilinx Artix-7 FPGA chips is analyzed. Test circuitry that runs completely on the FPGA chip has been developed. The ROs’ frequencies are recorded using Agilent 16801A logic analyzer. The architecture of the design is shown in Figure 4-5. The challenge generator is used to produce the inputs to the MUX which activates one RO at a time. ROs are activated, one at a time, from the top to the bottom of each column of the FPGA. Each RO is activated for 0.4 ms. There is a 0.1 ms delay before the next RO is activated; this is to reduce the noise in the form of heat that can be generated from the adjacent CLB [20]. A 0.2 ms delay gap is given between the RO and the counter activation for the signal to be stabilized before the measurement starts. The timing controller controls all time intervals involved, such as the time interval for the RO activation and the time interval for the counter to measure each RO.

The frequency is computed using Equation 7-9 where \( x \) is the cycle counts from each RO and \( y \) is the cycle counts for the 50 MHz reference clock. The preset value for \( y \) is set to 10000 cycles implying that the RO cycles are measured within a 0.2 ms period.
The accuracy of the measurement is 0.005 MHz/cycle which is good enough to measure the differences between frequencies generated from ROs.

\[ frequency = x \times \frac{50}{y} \]  

(7-9)

For the accelerated aging experiment on Spartan 3E and Artix-7 FPGAs, each RO is activated every 64 ms and 107.5 ms respectively. Each activation turns the RO on for a time period of 0.4 ms. Therefore, each RO is activated 1.3 million times a day for Spartan 3E and 0.8 million times a day for Artix-7. This aging experiment is conducted for 30 days. The number of ROs mapped on Spartan 3E and Artix-7 is 120 and 171 respectively. ROs are numbered according to the location they are mapped as shown in Figure 6-1. Responses are generated by using a chain-like neighbor coding where RO(n) is compared with RO(n+1). In total, there are 119 response bits generated from 120 ROs for Spartan 3E and 170 response bits are generated from 171 ROs for Artix-7.

### 7.5 Results and Analysis

ROs are mapped on all the CLBs available on the Spartan 3E FPGAs, and on half of the CLBs available on Artix-7 to record the frequencies. Responses are generated from the frequencies recorded. Chain-like neighbor coding technique is used to select the RO comparison pair [3]. Equation 7-10 is used to generate the response. Table 7.1 shows the uniqueness, reliability, uniformity, bit aliasing and diverseness results for both FPGA families used in this experiment.
\[ \text{Response bit} = \begin{cases} 1 & \text{if } f_a > f_b \\ 0 & \text{otherwise} \end{cases} \] (7-10)

Table 7.1: ROPUF’s parameters comparison.

<table>
<thead>
<tr>
<th></th>
<th>Spartan 3E (90nm)</th>
<th>Artix-7 (28nm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniqueness (%)</td>
<td>39.79</td>
<td>45.15</td>
</tr>
<tr>
<td>Uniformity (%)</td>
<td>51.25</td>
<td>50.17</td>
</tr>
<tr>
<td>Bit aliasing (%)</td>
<td>50.54</td>
<td>50.17</td>
</tr>
<tr>
<td>Reliability (%)</td>
<td>96.34</td>
<td>97.28</td>
</tr>
<tr>
<td>Diverseness</td>
<td>2.09</td>
<td>3.88</td>
</tr>
</tbody>
</table>

7.5.1 ROPUF Uniqueness

ROPUF responses on Artix-7 has the highest uniqueness percentage (45.16%) compared to Spartan 3E (39.79%). Each response from Artix-7 and Spartan 3E contains 780 bits and 239 bits respectively. The Artix-7 used in this experiment has 101,440 logic cells compared to the Spartan 3E that has 2160 cells. Thus Spartan 3E has limited resources compared to Artix-7. The maximum number of ROs that can be mapped on Spartan 3E is 240. Artix-7 uniqueness is shown to be closer to the ideal uniqueness value of 50%. Spartan 3E uniqueness seems to be a little bit far from the ideal uniqueness value.

Figure 7-1 and Figure 7-2 show the planar view of graphs for RO frequencies versus RO locations for Spartan 3E and Artix-7. These graphs are plotted to better understand the uniqueness difference in ROPUF responses between these two FPGA families. Figure 7-1 shows RO frequencies from three Spartan 3E FPGAs. The dark blue blocks are the inaccessible area on the FPGA. It can be observed that ROs with high
frequency are mostly distributed in the red circle. The same observation can be made for all 29 Spartan 3E FPGAs where most of the ROs with high frequencies are located in the middle of the FPGA. Figure 7-2 shows RO frequencies for the three Artix-7 FPGAs. In Figure 7-2, (a) and (b), it can be observed that most of the ROs with high frequency are located in the top part of the FPGAs. But this same observation is not found in the 20 Artix-7 FPGAs.

![Figure 7-1: RO frequencies versus location on Spartan 3E.](image)

(a) FPGA 1  
(b) FPGA 2  
(c) FPGA 3
Figure 7-2: RO frequencies versus location on Artix-7.
(a) FPGA 1
(b) FPGA 2
(c) FPGA 3
The high frequency distribution at the same FPGA location in Spartan 3E FPGAs used in this experiment demonstrates the effect of systematic variation. There are two types of process variations: systematic and stochastic variation [13]. The systematic variation is usually caused by the mask, lithographic, and reticle stepper errors. Systematic variation has high correlation on all ICs that are manufactured on the same line. The stochastic variation is caused by the vibrations during lithography, wafer unevenness and non-uniformity in resist thickness. Stochastic variation possesses more random characteristics, which are different for each chip [13]. Stochastic variation’s effect can be observed when there is a random high and low RO frequency distribution. On the contrary, systematic variation’s effect can be observed when there is a certain pattern of high or low RO frequency distribution in a group of FPGAs. ROPUF response uniqueness decreases due to the systematic variation on FPGAs. Response bits generated from ROs located in an area that is affected by systematic variation tends to be the same [49].

7.5.2 ROPUF Uniformity

As far as uniformity is concerned, the Spartan 3E and Artix-7 chips both have good uniformity percentages (51.25% and 50.17%) which represent a good balance between ‘1’ and ‘0’ bits in the responses. The uniformity result shows that the ROPUF carry the randomness feature in the response within the FPGA chip regardless of the change in the silicon technologies used. This result also shows that the systematic variation effect that is observed in Spartan 3E FPGAs does not affect the ROPUF’s uniformity.
7.5.3 ROPUF Bit Aliasing

The bit aliasing percentages for Spartan 3E (50.54%) and Artix-7 (50.17%) chips are close to the ideal value of 50%. These results represent that there is a balance in the bits ‘1’ and ‘0’ composition across the same bit location in the responses from different FPGAs. The ROPUF responses are observed to carry the randomness feature between the FPGAs in the same family regardless of the change in the silicon technologies used.

7.5.4 ROPUF Reliability

The ROPUF’s responses are generated at different temperature and voltage settings to measure the reliability. Temperature variation experiment is done using a temperature chamber. Four different temperature settings are used: 0°C, 20°C, 45°C, and 70°C. For voltage variations, three different internal core supply voltages (V_{CCINT}) are used for Spartan 3E: 1.2V (normal), 1.3V, and 1.4V. For Artix-7, two different V_{CCINT} are used: 1.0V (normal) and 1.2V. The responses that are generated at the different temperature and voltage settings are compared with the responses that are generated at room temperature. The 30 day accelerated aging experiment is also implemented to extend the ROPUF’s reliability study on FPGAs that are fabricated using different silicon technologies.

The average ROPUF’s reliability for Spartan 3E and Artix-7 is 96.34% and 97.28%, respectively. Table 7.2 shows the individual ROPUF’s reliability due to temperature, voltage, and aging effects on ROPUF. It can be seen that Artix-7 reliability is higher than Spartan 3E for temperature, voltage, and aging. The lowest ROPUF’s reliability for both Spartan 3E and Artix-7 is caused by the voltage variations that are
94.26% and 94.82% respectively. The ROPUF’s reliability on the temperature and voltage effects for both Spartan 3E and Artix-7 is fairly high.

Table 7.2: ROPUF’s reliability due to change in temperature, voltage, and aging.

<table>
<thead>
<tr>
<th>Reliability (%)</th>
<th>Spartan 3E</th>
<th>Artix-7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature</td>
<td>97.10</td>
<td>98.16</td>
</tr>
<tr>
<td>Voltage</td>
<td>94.26</td>
<td>94.82</td>
</tr>
<tr>
<td>Aging</td>
<td>97.67</td>
<td>98.86</td>
</tr>
</tbody>
</table>

Figures 7-3 and 7-4 show the RO frequencies with respect to the temperature and voltage variations for Spartan 3E and Artix-7, respectively. Figure 7-3 (a) and Figure 7-4 (a) show how the RO frequencies on Spartan 3E and Artix-7 decrease when the environment temperature increases. It can be observed that the frequency for each RO is decreasing uniformly with respect to the temperature changes. This observation suggests that the temperature changes affect the RO frequency uniformly regardless of its location.

The only difference that can be noticed in the RO frequency changes due to the temperature effect between Spartan 3E and Artix-7 is the frequency decrement quantity. The frequency of the ROs on Spartan 3E reduces 2 to 3 MHz on an average at 45°C and 5 to 6 MHz at 70°C. For Artix-7, the RO frequency reduces 1 to 0.5 MHz at 45°C and 1.5 to 1 MHz at 70°C.

Figure 7-4 (b) and Figure 7-4 (b) show the RO frequency changes due to the voltage variation for Spartan 3E and Artix-7, respectively. The RO’s frequency can be seen as increasing uniformly as the $V_{CCINT}$ is increased. This observation suggests that the voltage changes affect the RO’s frequency uniformly despite the RO’s location. It can be noticed from these figures that the RO’s frequency is sensitive towards the $V_{CCINT}$.
changes. The RO’s frequency for Spartan 3E increases 20 MHz at 1.3V and 40 MHz at 1.4V. The RO’s frequency on Artix-7 increases by 90 MHz at 1.2V. The RO frequency change due to the voltage variation is significantly higher than the changes due to the temperature variations. This observation suggests that the ROPUF’s reliability is lowest for both Spartan 3E and Artix-7 due to the voltage effect.

(a)

(b)

Figure 7-3: Spartan 3E
(a) RO frequency changes with respect to temperature variations.
(b) RO frequency changes with respect to voltage variations.
Figure 7-4: Artix-7
(a) RO frequency changes with respect to temperature variations.
(b) RO frequency changes with respect to voltage variations.

Figure 7-5 shows the aging effect on 10 RO frequencies on Spartan 3E and Artix-7. For Spartan 3E, the frequencies from 10 ROs are observed to have normal fluctuations. No increasing or decreasing frequency pattern is observed. However, for Artix-7, the
frequencies from 10 ROs are observed to have similar decreasing pattern. In average, the RO frequencies on Artix-7 are reduced by 0.5 MHz at the end of the aging experiment. This observation suggests that the aging affects the frequency of the ROs uniformly regardless of their spatial location.

Figure 7-5: RO frequency changes with respect to aging.
(a) Spartan 3E
(b) Artix-7
7.5.4 ROPUF Diverseness

The Artix-7 has the highest ROPUF diverseness of 3.89. Thus represents the high gap between the maximum and minimum frequencies. High diverseness is a good feature for ROPUF as it increases the number of CRPs which can be generated [34]. The Spartan 3E diverseness is found to be slightly lower than Artix-7 namely, 2.09. It is observed that the diverseness increases when advanced silicon technologies are used. This is due to the reduced transistor size which increases the RO’s frequency. Therefore, slight changes in the process variations are amplified by the higher RO’s frequency.

7.6 Summary

In this work, we have implemented ROPUFs on 20 Artix-7 FPGA chips and 29 Spartan 3E chips which cover a wide range of silicon technologies. We have recorded and analyzed thousands of RO frequencies from each chip. We conclude that only diverseness parameter changes with respect to the silicon technologies used, and the uniqueness parameter improves as the FPGA chip density increases.
Chapter 8

ROPUF Application: Hardware-Oriented Security-Based Authentication for Advanced Metering Infrastructure

8.1 Introduction

A smart grid is often described as the merging of a traditional power grid with an advanced communication technology to increase the power network’s delivery efficiency. The smart grid is capable of two-way electricity and information. The two-way communication in the smart grid allows many parties in the network to exchange information. For example, the power provider can receive information on the customer’s power usage, and the customer can receive the recent pricing information from the power provider. The information exchanges facilitate the power provider to estimate and control the power generation more efficiently. The customer can utilize the pricing information to optimize their electricity usage. Many other benefits can be obtained through the smart grid’s implementations that are not mentioned here [35].

Though the smart grid has been utilized in some places in the world, there is a growing concern on its security. The smart grid’s security objectives can be grouped into three categories: availability, integrity, and confidentiality [35]. Availability ensures timely and reliable access to and use of information in all components of the smart grid. Integrity guards the information from being modified or destructed to ensure information nonrepudiation and authenticity. Confidentiality preserves the authorized restrictions on
information access in order to protect personal privacy and proprietary information. Authentication is one of the important security features that contributes to the integrity and confidentiality security objectives.

In a smart grid, the authentication scheme has to be different from the one used by internet technology [35]. This is due to the different threats that exist in the smart grid network compared to threats that exist in internet technology. The possible threats of a smart grid network include attacks targeting data integrity and operation disruption [35][36]. These types of attacks can be managed by having secured authentication. There are some basic requirements in smart grid authentication protocol such as high efficiency and tolerance to faults and attacks [35]. We discuss our design based on these requirements later in section 8.4.

We use a Physical Unclonable Function (PUF) as the authentication scheme for the Advanced Metering Infrastructure (AMI) in the smart grid. There are many types of PUFs: this work uses silicon PUF, specifically PUF on a FPGA. During fabrication of the silicon PUF, minor irregularities occur. The minor irregularities cause slight differences in the electrical delay in the silicon chip. The differences are not noticeable in the functionality of the chip, but PUF exploits the minor irregularities to generate a number of binary IDs that are unique for each chip. Delay-based PUF uses ring oscillators (ROs) to extract the minor irregularities and make them visible through different frequencies (will be discussed in Section 8.3). A Ring Oscillator Physical Unclonable Function (ROPUF) is highly secure; it cannot be modeled since the minor irregularities that occur during the fabrication process are random for each chip. Confidential information is not stored on its circuit.
In this work, we discuss our proposed hardware oriented security based authentication on AMI using ROPUF on FPGAs. Our contributions are as follows:

1. We introduce an authentication scheme using ROPUF. The authentication scheme is limited to the network between the utility company and smart meter which we refer to as AMI. The intention is to set a design boundary so that the proposed scheme is developed to meet the authentication requirements within that boundary.

2. The proposed authentication scheme focuses on the current enhancement of the existing AMI. No major changes in the protocol are needed to implement the scheme. Our scheme can be combined with the existing protocol.

3. We have proved that our ROPUF is tolerant to attack since it cannot be modeled. A linear support vector machine (SVM) is used to test the ROPUF, and the results show that the SVM fail to model the ROPUF.

4. We have also proved that the proposed ROPUF meet the efficiency and tolerance requirements through experiments conducted for the proof of concept.
8.2 Related Work

In this chapter, we limit the scope of authentication to communication in AMI. There are many authentication schemes that have been proposed. We divide the proposed schemes into two groups. The first group comprises of schemes proposed in terms of algorithms on the existing resources. In [37], a lightweight message authentication scheme that uses a shared session key established using Diffie-Hellman exchange protocol is presented. In [38], an authentication scheme that is based on Merkle hash tree scheme which is used to construct a tree based on a one-way cryptographic hash function is described. In [39], smart grid key management (SGKM) based on enhanced identity based cryptography (EIBC) is suggested. These schemes are based on non-volatile memory technologies that are vulnerable to invasive/spoofing attacks.

The second group for the authentication scheme is based on hardware-oriented security. In [40], an interoperable device identification in a smart grid based on a trusted platform module (TPM) is proposed. This technology is defined by a trusted computing group implementing consistently behaving computer systems as a technology. Trusted computing technology provides methods for reliably checking a system’s integrity and identifying anomalous or unwanted characteristics. In [41] an authentication and key management scheme for advanced metering infrastructures using PUF is proposed.

In this chapter, our proposed scheme for authentication in AMI fits in the second group which uses PUF for authentication. The advantages offered by our scheme compared to [40] and [41] are discussed in section 8.5. We list the requirements of the smart grid security environment and discuss each authentication scheme that belongs to the second group according to the requirements as shown in Table 8.1 [36].
Table 8.1: Comparison of different schemes based on Smart Grid requirements

<table>
<thead>
<tr>
<th>Smart Grid Security Applications Requirements</th>
<th>Trusted Computing Technology [40]</th>
<th>PUF Scheme [41]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) High performance in terms of latency and jitter in message exchange. (efficiency)</td>
<td>Message length is not mentioned. Authentication process takes 11 steps to complete.</td>
<td>Message length is not mentioned. Average authentication process takes 3 steps to complete.</td>
</tr>
<tr>
<td>2) Timeliness: computation and communications subsystems must meet real-time requirements of applications (efficiency).</td>
<td>The scheme takes 982.91 ms to complete the authentication process.</td>
<td>The exact information is not stated. The only information mentioned about time is 2.4 ms for PUF execution and 0.2 ms (average) for the SHA-1 on 32 bit PUF response using the PC.</td>
</tr>
<tr>
<td>3) Comprehensive security design, as schemes are likely targets for sophisticated cyber-attacks (tolerance to attacks).</td>
<td>Highly sophisticated TPM which provides unique identity for each module and strong cryptography co-processor. TPM has secure storage using a unique asymmetric storage root key (SRK) of which the private part never leaves the TPM.</td>
<td>Scheme based on the unclonable features derived from the process variation on the silicon chip.</td>
</tr>
<tr>
<td>4) Adaptable and evolvable designs because components typically have a lifetime of 15 or more years once deployed (tolerance to faults).</td>
<td>This scheme is adaptable since it is added to the existing devices on a smart grid. It is not evolvable as the design is made on application specific IC (ASIC).</td>
<td>This scheme is both adaptable and evolvable. The scheme is independent and can be added to any smart meter. It is also evolvable because it uses FPGA; hence, it can be reprogrammed.</td>
</tr>
</tbody>
</table>
The latency to complete the authentication process is 11 steps for the trusted computing scheme and 3 steps for the PUF scheme. The trusted computing scheme takes 982.91 ms to complete the authentication process. The timing information for the authentication using PUF scheme is not sufficient to summarize the total time. The latency implies the availability of the scheme to operate in real time. Both technologies have very good features that support tolerance to attacks: the security scheme is embedded in the hardware. The last requirement discussed in Table 8.1 regards the adaptable and evolvable designs due to the limitation on the components’ life span. The trusted computing scheme is adaptable because the module can be replaced over time, but is not evolvable as it uses ASIC in the design. The PUF scheme is both adaptable and evolvable because it is designed to be a stand-alone unit that can integrate with any smart meter and is also evolvable through the configurability features that are offered by FPGA.

8.3 Hardware-Oriented Security-Based Authentication for AMI

Based on the AMI, the utility companies monitor their customers’ usage through the smart meter. All data from smart meter is sent to the utility companies through a number of smart meters (hopping network) and data concentrators. This means that all data sent to and from the utility companies goes through a number of hops before it reaches the destination as shown in Figure 8-1. For the authentication process, we propose the utility company to be the center point for the data concentrators and smart meters authentication. There are three good reasons for the utility company to be the center point. The first one is that utility companies need to monitor and update their
customers regularly. The second reason is that all critical control messages, such as switching off certain users’ appliances, can only come from the utility companies; and the third reason is the utility companies have bigger and more secure storage (secured from site channel attacks).

All devices (data concentrators and smart meters) involved in the communication from the utility company to the smart meter need to have the ROPUFs chip as shown in Figure 8-2. The utility company does not need to have the ROPUF chip as it is the trusted authority that controls and monitors the network. The ROPUF uses the existing network protocol. In this chapter we present the proposed short authentication protocol that needs to be added in addition to the existing protocol. Our focus is to provide a practical solution that can be applied to the existing technology at low cost.

The first step in the implementation is to recognize all devices present in the AMI. The utility company scans and records all the challenge and parity bits pairs (CPBPs) from each ROPUF chip. The scanned ROPUF chips are connected to each device in the AMI. Through this method an utility company can keep track of the devices that are present in the AMI.
When the smart meter needs to send data to the utility company, it needs to authenticate itself. The smart meter first requests that the utility company send data as shown in Figure 8-3. Then utility company sends a challenge $C_i$ to the smart meter. The smart meter then uses the $C_i$ received to produce the authentication code in the form of Parity Bits (PBi) and send it back to the utility company. The utility company verifies the
response PBi; if it is correct then permission is granted, but if it is wrong, two more chances are given until the smart meter can send a correct response as shown in Figure 8-4. The worst case scenario is that the smart meter fails to send a correct response, in which case the utility company sends a broadcast signal (BLOCK(broadcast)) so that all devices in the AMI drop any packet received from that particular smart meter, and no data can be sent through the smart meter. This action automatically isolates the smart meter from the AMI. To solve this problem, the utility company needs to verify with the customer whether it is a technical error or an adversary attack.

The next potential adversary attack is against the data concentrator. The data concentrator acts as the forwarding device that completes the data path from the smart meter to the utility company. Identity impersonation and data jamming are potential
attacks on the utility company. To make sure that all data concentrators in the network are real, the authentications need to be done regularly. For the smart grid authentication, we propose an authentication every 15 minutes, as suggested in [36][41]. For the data concentrator authentication, the utility company first sends a verification request signal with the challenge (VER(C_i)) to the data concentrator as shown in Figure 8-5. The data concentrator uses the C_i sent by the utility company to produce the PB_i and send it to utility company. The utility company verifies the PB_i, and if the PB_i matches, then an acknowledge (ACK) is sent and the data concentrator can operate as usual. If the PB_i does not match then two chances are given for the data concentrator to send a correct PB. If the data concentrator still fails to produce a correct PB, the utility company drops that data concentrator from the network by sending a broadcast signal to all devices on the network in order to isolate the particular data concentrator. The steps taken are the same as shown in Figure 8-4.

![Figure 8-5: Data concentrator to utility company authentication.](image)

The most critical attack possible is the utility company impersonation. The utility company holds the main authority over all devices in the network. An adversary can get control of the customer’s smart meter if they can impersonate the utility company. The ANSI C12.18 standard defines six security levels of access that indicate different privileges, L0 to L5, with L5 being the highest privilege. The security level L0 requires
no password [41]. To differentiate the security levels we are using five different lengths of PBs. The five different length of the PBs are 64, 128, 256, 512 and 1024 bits. Higher security privilege access requires a higher length of PBs. Authentication occurs when the utility company sends a request to the smart meter for a specific level of access permission (REQ(level)) as shown in Figure 8-6. Then the utility company sends a challenge and a hamming code parity bits pair (CPBPi-length(level)) to the smart meter. The smart meter verifies the PBi-length sent from the utility company by generating its own PBi-length from the given Ci-length. The utility company has two more chances if the smart meter fails to verify the CRPi-length(level) as shown in Figure 8-7. If the adversary tries to impersonate the utility company, the utility company detects the attack when receiving the first NACK from the smart meter.

Figure 8-6: Utility company to smart meter authentication.

Figure 8-7: Utility company to smart meter fail authentication.
8.3.1 ROPUF Design

In our ROPUF design we use 3-stage ROs. The benefits of using 3-stage ROs have been discussed in another chapter [17]. For the smart grid authentication application we propose to use ROPUF that can generate up to 2048 bits responses. Our ROPUF design uses 120 ROs. If the smart meter in the smart grid application is required to authenticate itself every 15 minutes for the smart meter to update the usage information, this means that it needs to authenticate 1,752,000 times or 50 years’ time span. Our ROPUF design is capable of handling this type of requirement. Each authentication uses new response bits which makes it harder for the adversary to launch an attack.

Figure 8-8 shows the logic blocks for the ROPUF circuit. Each RO activates for 0.4 ms and there is a 0.1 ms gap before the next RO is activated; this is to reduce the heat noise that generates from the adjacent CLB [17]. A 0.2 ms gap between the RO and counter activation allows the signal to be stabilized before the measurement starts. The timing controller controls all time intervals involved, such as the time interval for each RO’s activation and the time interval for the counter to measure each RO. The counter 1 measures the number of cycles generated from the RO’s frequency selected through multiplexer 1 (Mux 1) and Counter 2 measures the number of cycles generated from RO’s frequency selected through Mux 2. Then the comparator compares the number of cycles recorded by Counter 1 and 2 to generate one response bit. The generated response is stored in the register.
To reduce the bit flip occurrences we measured the response generation under various environmental factors such as temperature and voltage variations in order to study the bit flip occurrences [34]. From the study, we found the importance of using the ROs comparison pairs that have high differences to avoid the bit flip occurrences. Figure 6-2 (a) shows the example of ROPUF output flip occurrence when the difference between two selected ROs is small. Figure 6-2 (b) shows how the output flip can be prevented by selecting two ROs that have a higher frequency difference. From the data obtained on specific FPGA chips (S3E100), we found the best threshold to be 5 MHz. The threshold measures the number of possible ROs comparison pairs that have higher frequency differences than 5 MHz. The number of ROs comparison pairs and CRPs that pass the 5MHz threshold for 5 different FPGA chips (S3E100) are shown in Table 8.2 (we use
\[
(n!/(n-r)!(r!)) \text{ equation). The number of possible CRPs generated is abundant enough to support the frequent authentication requirement, and the security is guaranteed as CRPs will not be reused.}
\]

Table 8.2: Number of possible CRPs.

<table>
<thead>
<tr>
<th>ROPUF</th>
<th>Comparison pairs</th>
<th>128 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1863</td>
<td>$1.18 \times 10^{285}$</td>
</tr>
<tr>
<td>2</td>
<td>3942</td>
<td>$5.758 \times 10^{243}$</td>
</tr>
<tr>
<td>3</td>
<td>5880</td>
<td>$1.947 \times 10^{266}$</td>
</tr>
<tr>
<td>4</td>
<td>4504</td>
<td>$1.919 \times 10^{251}$</td>
</tr>
<tr>
<td>5</td>
<td>3381</td>
<td>$1.18 \times 10^{235}$</td>
</tr>
</tbody>
</table>

8.3.2 Authentication

For the authentication in the AMI, the Ri generated from the ROPUF is not used because the adversary could model the ROPUF based on the Ci and Ri that are sent through the network. To enhance security, the hamming code parity bits (PBs) are sent out for authentication as shown in Figure 8-9. Hamming code is a linear error correcting code that generalizes the hamming $(7,4)$ code. A block of data that has a length of k bits is assigned with $n-k$ parity bits. The length of the message after adding the parity bits is n bits. The block length is represented as $n=2r-1$, and the message length is represented as $k=2r-r-1$ where r is the length of the parity bits and $r \geq 0$. The block of data represented by $m$ and the data with parity bits represented by $x$ are given by: $x=mG$, where $G$ is the generating matrix.
PB\textsubscript{i} is used in the authentication by generating 4 parity bits for every 8 bits of R\textsubscript{i} as shown in Figure 8-10. For 128 bits R\textsubscript{i}, 64 bits of PB\textsubscript{i} are generated. Based on the equation mentioned before, 4 hamming code parity bits are able to cover 16 bits of data for the error detection and correction. But in our ROPUF design, we use hamming code as an authentication code generator. We maximize the length of authentication code to increase the security level by generating 4 hamming code parity bits for every 8 bits of data. There are three advantages of using hamming code parity bits as the authentication code. First, hamming code is a one-way function. The second advantage is that there is no way to model the ROPUF using the Ci and PB\textsubscript{i} (discussed in Section 8.4). Third, the authentication code has a shorter length compared to R\textsubscript{i} yet produces better security.

Figure 8-11 shows how each ROPUF chip is registered with the utility company. The utility company has a database that stores all possible challenge for each ROPUF.
The combinations of challenge for each ROPUF are different as discussed in the previous section because only ROs comparison pairs that pass the frequency difference threshold are selected. Challenge for all ROPUFs should be provided by the manufacturer. Utility company sends one C_i at a time to each ROPUF and records the PB_i generated from each particular ROPUF in the database. The ROPUF registration process starts with the challenge for 128 bits response and continues until 2048 bits response for different level of security access.

![Image of ROPUF registration process]

**Figure 8-11:** ROPUFs registration with utility company.

ROPUFs that have been registered can be used by the smart meters and data concentrators in the AMI. The ROPUF does not need any additional storage from the devices. It just needs to be connected serially to the device as shown in Figure 8-2 via serial connection, and the firmware on the devices needs to be updated to support the additional protocol proposed to generate the authentication key.
8.4 Proof of Concept

In this section, we discuss the proposed ROPUF design for smart grid authentication based on three smart grid requirements mentioned in Section 8.1. To prove the concept, we implement our ROPUF design on Spartan 3E FPGAs. The PC acts as an utility company that stores all the CRPs, and the smart meters and data concentrators are implemented on FPGAs. This set-up simulates the protocol and the effectiveness of our scheme. Table 8.3 shows the time taken to transfer the PBi and Ci through the USB connection. For the first authentication level, the total authentication time taken is 65.364 ms via the USB connection. This time is within the range of achieving real-time communication. The first authentication level is used the most as it involves the information passing between the utility company and smart meter. The second level authentication takes 130.73 ms, followed by third (261.46 ms), fourth (522.91 ms) and fifth (1045.83 ms) level authentications. In terms of high efficiency, our system meets the smart grid requirement in which the authentication can be achieved in real-time.

Another factor to consider is cost of storage. Extra data storage is needed to store the challenge and PB from ROPUFs. Table 8.4 shows the data storage needed to store all challenge and parity bits pairs (CPBPs) for 50 years. For the first authentication level, we assume that authentication would take place every 15 minutes. In this case, one ROPUF needs 35136 CPBPs in one year. The size of data storage needed to store the CPBPs for one year is 8 MB. If the life span of the AMI in smart grid is expanded to 50 years, then the data storage size needed to store CPBPs for one ROPUF would be 408 MB. For other authentication levels, we assume that a 5 times a day usage will require 1830 CPBPs in one year for one ROPUF. For a 50-year life span, the second authentication level needs
46 MB data storage, followed by the third, fourth and fifth level authentications (93 MB, 186 MB and 371 MB, respectively).

Table 8.5 shows the total data storage needed to store all the CPBPs according to the number of devices (data concentrator and smart meter) involved in the AMI. Storing complete CPBPs (all authentication levels) for one device takes 1.1 GB of data storage. If the AMI has 2000 devices, 2207 GB of data storage is needed. Two TB of data storage cost around $140 currently. Thus, the proposed authentication scheme using ROPUF does not incur high cost to the utility company and is cost effective.

Table 8.3: Authentication time for each level.

<table>
<thead>
<tr>
<th>Authentication level</th>
<th>R (bits)</th>
<th>PBi (bits)</th>
<th>Ci (bits)</th>
<th>PBi Generation</th>
<th>PBi Transfer</th>
<th>Ci Transfer</th>
<th>Total Authentication</th>
</tr>
</thead>
<tbody>
<tr>
<td>First L1</td>
<td>128</td>
<td>64</td>
<td>1792</td>
<td>64</td>
<td>0.047</td>
<td>1.317</td>
<td>65.364</td>
</tr>
<tr>
<td>Second L2</td>
<td>256</td>
<td>128</td>
<td>3584</td>
<td>128</td>
<td>0.094</td>
<td>2.634</td>
<td>130.728</td>
</tr>
<tr>
<td>Third L3</td>
<td>512</td>
<td>256</td>
<td>7168</td>
<td>256</td>
<td>0.188</td>
<td>5.268</td>
<td>261.457</td>
</tr>
<tr>
<td>Fourth L4</td>
<td>1024</td>
<td>512</td>
<td>14336</td>
<td>512</td>
<td>0.376</td>
<td>10.537</td>
<td>522.913</td>
</tr>
<tr>
<td>Fifth L5</td>
<td>2048</td>
<td>1024</td>
<td>28672</td>
<td>1024</td>
<td>0.753</td>
<td>21.074</td>
<td>1045.827</td>
</tr>
</tbody>
</table>

Table 8.4: Data storage size for each authentication level.

<table>
<thead>
<tr>
<th>CPBP authentication level</th>
<th>Year(s)</th>
<th>Data size (megabytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>1</td>
<td>8.152</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>81.516</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>163.031</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>244.547</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>326.062</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>407.578</td>
</tr>
<tr>
<td>Second</td>
<td>50</td>
<td>46.4</td>
</tr>
<tr>
<td>Third</td>
<td>50</td>
<td>92.8</td>
</tr>
<tr>
<td>Fourth</td>
<td>50</td>
<td>185.6</td>
</tr>
<tr>
<td>Fifth</td>
<td>50</td>
<td>371.2</td>
</tr>
</tbody>
</table>
Table 8.5: Data storage size needed based on number of devices on the AMI.

<table>
<thead>
<tr>
<th>Number of devices</th>
<th>Data Size (gigabytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.103578</td>
</tr>
<tr>
<td>100</td>
<td>110.3578</td>
</tr>
<tr>
<td>200</td>
<td>220.7156</td>
</tr>
<tr>
<td>300</td>
<td>331.0734</td>
</tr>
<tr>
<td>400</td>
<td>441.4312</td>
</tr>
<tr>
<td>500</td>
<td>551.789</td>
</tr>
<tr>
<td>600</td>
<td>662.1468</td>
</tr>
<tr>
<td>700</td>
<td>772.5046</td>
</tr>
<tr>
<td>800</td>
<td>882.8624</td>
</tr>
<tr>
<td>900</td>
<td>993.2202</td>
</tr>
<tr>
<td>1000</td>
<td>1103.578</td>
</tr>
<tr>
<td>2000</td>
<td>2207.156</td>
</tr>
</tbody>
</table>

To test our ROPUF security level, we use a support vector machine (SVM) to model the ROPUF based on the Ci and PBi. In this model we assume that the adversary has knowledge of the encryption code used in the network and also of the hamming code to generate the PBi for every 8 bits of the response. We use SVM because the Ci and PBi can be classified as ‘1’ and ‘0’. A SVM classifies data by finding the best hyper-plane that separates one class from another class. The best hyper plane has the largest margin between the two classes.

First, we train the SVM classifier with a group of data with the correct classifier. The data X and classifier Y are fed to the SVM train function to train the classifier. The Gaussian Radial Basis Function Kernel with a scaling factor of sigma equal to one is used for the classifier training. The data set consists of the RO pairs used to generate the parity bit as shown in Figure 8-12. As an example, response bit b_0, b_1, b_3, b_4, and b_6 are used for the generation of the first parity bit. RO_1 and RO_2 are used to generate b_1, RO_3 and RO_4
are used to generate b2, and so on as shown in Figure 8-12. The parity bit p0 is the classifier Y, and the data X consists of RO1 to RO10. One authentication key has 64 parity bits comprises of p0 until p63.

![Figure 8-12: Parity bits and corresponding ROs.](image)

Figure 8-13 shows the accuracy of the prediction results. As the number of data used to train the SVM classifier increases, the accuracy also increases to a point, but then gradually decreases as the number of data used is further increased. The best accuracy obtained is 60.9%, showing that the SVM cannot model the ROPUF by using the Ci and PBi. This test proves that the proposed ROPUF design for the AMI is secure from the adversary’s attack. Another advantage of using ROPUF as the authentication system is that no clues useful for the adversary to crack the ROPUF are stored on the devices. Even if the adversary is able to model one of the ROPUFs in the AMI, it will take more time to break another ROPUF because the only way to break the ROPUF (if there is a way of doing it) is by gathering the challenge and PB pairs and creating a model.

The security of the database that stores all challenge and PBi for the devices on the AMI is also important. However, the database is less vulnerable to adversary attack
since the other devices have no access to the utility company. The only communication the utility company has with the devices in the network is sending and receiving information. But, if the adversary is able to hack the utility company, or some of the utility company employees breach trust, then the threat is unavoidable. However, if that type of attack occurs, our ROPUF authentication system can be fixed. The ROPUF can be reprogrammed, and different sets of ROs can be used, rendering the previous challenge and PB invalid. In terms of the tolerance to attack requirements, the ROPUFs design requires endless effort from the adversary in order to model the ROPUFs. The vulnerability exists at the utility company database, and we assume that the utility company has to have a good firewall to protect the system as a whole.

![Graph](image.png)

**Figure 8-13:** SVM prediction accuracy for ROPUF.

Regarding the tolerance to faults, the authentication system is designed to tolerate 10% of discrepancies in the PB. Additionally, the comparison pairs used to generate the response in the ROPUFs have at least 5 MHz difference. This ensures that the Ri generated from the ROPUFs will not get flipped when exposed to anomaly voltage and temperature conditions. Figure 8-14 shows that the bit flip probability trend reduces when the frequency difference increases. We find that the bit flip occurs most when the maximum frequency difference is 1 MHz [34]. To guarantee the ROPUF is able to deal
with the worst case scenario, though, we recommend using the largest threshold frequency possible.

![Figure 8-14: Bit flip probability vs. frequency difference (MHz)](image)

8.5 Summary

In this work, we propose a new scheme for authentication of the Advanced Metering Infrastructure in a smart grid. The novel authentication scheme using ROPUF offers high security with less overhead compared to previous proposed schemes [40][41]. In terms of latency, our proposed scheme takes, at most, four steps for authentication. The complete authentication times for the most used security level L1 and L2 are 65.4 ms and 130.7 ms, respectively. These times satisfy the availability requirements in a smart grid. The authentication keys sent through the network do not provide any clues that allow the adversary to model the ROPUF as proved in the SVM trained data results. This system is designed as a stand-alone unit so that it can work in addition to the existing protocol currently used in the industry. The system is also designed to tolerate the fault occurrence in the system, such as using high RO comparison pairs only, and tolerating 10% discrepancies in the PB. The reconfigurability feature offered by the FPGA makes the ROPUF evolvable as it can be reprogrammed at any time.
Chapter 9

Conclusions

9.1 Summary and Conclusions

The importance of hardware security and trust is increasing as the industry supply chain has become more complex and also more vulnerable to adversary attacks. An article in the IEEE Spectrum entitled “The Hidden Dangers of Chop-Shop Electronics” describes how clever counterfeiters sell old components as new, thus threatening both military and commercial systems [43]. On August 17, Boeing warned the U.S. Navy that an ice-detection module in the P-8A Poseidon (new reconnaissance aircraft) contained a reworked part that should not have been put on the airplane originally and should have been replaced immediately. The company that supplies the ice-detection module has blamed the part, a Xilinx field-programmable gate array (FPGA), for the failure of the ice-detection module during a test flight. However, retracing that FPGA’s path led not to Xilinx but to a Chinese company called, “A Access Electronics”. It apparently had turned a quick profit by selling used Xilinx parts as new. This incident is one of the examples of
how the vulnerability in the hardware security and trust has become a security threat. There are two points that can be highlighted from this true incident. The first is how widely the programmable chip or FPGA is used in the industry. The second is the vulnerability that exists in the industry’s supply chain that could lead to serious safety and security issues. Therefore, it is important to increase the security of FPGAs and other custom designed chips. Some techniques have been proposed in the past to enhance the security of FPGAs. In this work, we have proposed a ring oscillator based technique which extracts the process variation effects from the FPGA and converts it to a unique ID. A ROPUF can be used as an authentication technique for an FPGA to verify its trustworthiness. Apart from that, ROPUFs can also be used as a cryptography technique to encrypt and decrypt information.

9.2 Contributions and Results

Major contributions made in this research are listed below:

- Three different FPGA families which are fabricated using different silicon technologies are used to explore the ROPUF. The ROs are studied and compared based on five parameters; uniqueness, reliability, uniformity, bit-aliasing, and diverseness.
  - The temperature variations, voltage variations, and accelerated aging experiments are done to measure the reliability.
  - The FPGA fabricated using the latest technology shows better performance based on the five parameters used.
Different numbers of stages used in ROs are explored. The experimental results obtained suggest that a lower number of stages used in a ROPUF on FPGA contributes to better performance regardless of the silicon technologies used.

Different FPGAAs may have different minimum number of stages that can be used in ROs due to the limitation on the FPGA components’ maximum operating frequency. The minimum number of RO stages that can be used in Spartan 2 and Spartan 3E is 3, and for Artix-7 is 5-stage.

Based on the experimental results, we conclude that a ROPUF is applicable regardless of different silicon fabrication technologies used to produce an FPGA.

- The systematic variation effect on ROPUF’s security reliability has also been studied in this work.

  - The experimental results showed that systematic variation does affect the ROPUF responses’ randomness and uniqueness parameter.

  - The RPM technique is developed to overcome this effect. The results obtained by using the RPM technique are shown to be better than other techniques that have been proposed before.

  - The responses generated from ROPUFs after applying the RPM technique passed most of the NIST statistical test for randomness.
• The ROPUF is applied as the hardware-oriented security-based authentication for advanced metering infrastructure (AMI). The authentication system is developed based on ROPUF and is targeted for AMI.
  o ROPUF is used to generate the unique ID for the devices involved in the AMI for the authentication.
  o This system is designed to fit in the current AMI system with low cost implementation.
  o Details of the implementation cost are shown in this work as the proof of concept.
  o The security of ROPUF system used is also tested using support vector machine (SVM). The SVM is trained using a large data set and challenges are fed into the SVM to predict the response sets.
  o Results obtained show that SVM failed to predict ROPUF responses based on the challenges, thus lending credence to the security offered by the proposed authentication system.

9.3 Future Works

The research work done in this dissertation can be further extended by performing the following:

• Implementation of the ROPUF authentication scheme for the AMI network using a simulation software such as NS-2 to analyze its performance.
• An efficient error correction circuit can be developed to improve the ROPUF security.
• A ROPUF scheme can be developed for cryptography
• A hardware Trojan detection technique can be designed using the RO.
References


[38] Li Hongwei, Lu Rongxing, Liang Zhou, Bo Yang, and Xuemin Shen, "An Efficient Merkle-Tree-Based Authentication Scheme for Smart Grid," *Systems Journal, IEEE*, vol.8, no.2, pp.655,663, June 2014.


