REALIZATION AND ANALYSIS OF HIGH PERFORMANCE PHYSICAL UNCLONABLE FUNCTIONS BASED ON RING OSCILLATORS

by

Giray Kömürcü

B.S., Microelectronics Engineering, Sabanci University, 2005

M.S., Electrical & Electronics Engineering, Boğaziçi University, 2008

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

Graduate Program in Electrical and Electronics Engineering
Boğaziçi University
2014

REALIZATION AND ANALYSIS OF HIGH PERFORMANCE PHYSICAL UNCLONABLE FUNCTIONS BASED ON RING OSCILLATORS

APPROVED BY:

Prof. Günhan Dündar . . . .
(Thesis Supervisor)

Assist. Prof. Ali Emre Pusane . . . .
(Thesis Co-supervisor)

Assoc. Prof. Sıddıka Berna Örs Yalçın . . . .

Assoc. Prof. Arda Yurdakul . . . .

Assist. Prof. İsmail Faik Başkaya . . . .

DATE OF APPROVAL: 29.05.2014

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my thesis supervisor, Prof. Günhan Dündar for giving me the pleasure of being his student during my M.Sc. and PhD studies. He has involved very closely to my thesis and given very valuable ideas. His vast knowledge, continuous support, and perfect guidance lead my research to a successful conclusion.

I am also very grateful to my co-advisor Assist. Prof. Ali Emre Pusane for being my supervisor during the PhD studies. His deep knowledge and point of view helped significantly to develop, bring to perfection, and conclude my thesis in every aspect. I felt his positive manner, energy, and very valuable support during the entire period of my PhD studies.

I would also like to thank Assoc. Prof. Arda Yurdakul, Assist. Prof. İ. Faik Başkaya, and Assoc. Prof. S. Berna Örs Yalçın for being a part of my thesis committee. Their ideas and advises definetely improved the quality of this thesis.

I would also like to express my gratitude to Dr. Aziz Ulvi Çalışkan who has been a lot more than an employer to me in the last decade. I have felt his support, encouragement, and positive manner during my academic and work life. Everything would be much harder without his continuous support.

My special thanks goes to Dr. Yaman Özelçi who has supported my M.Sc. and PhD studies with great patience and tolerance. I have learnt a lot from his knowledge and engineering skills. He has also helped on the most important step of this PhD study by proposing the topic of the thesis, which has eased my work significantly.

I owe really much to my colleagues and friends Sedat Soydan, Umut Güvenç, and Ülkühan Güler, who have helped creating a very friendly and cheerful working environment in the past nine years. They have also contributed to my PhD studies

with their remarkable ideas, assistances, and support.

I would like to state my sincere gratitude to my mom Hülya Kömürcü, who has taught the importance of education since the first day of primary school and helped me in every aspect for my success without showing a sign of tiredness at any time. I would not find the power to continue my academic life without her passionate support.

Finally, I would like to express my deepest gratitude and my love to my wife, Elif Sezgin Kömürcü and my daughter, Neva for their love, patience, understanding, and all the sacrifices they made during my thesis study. It would be impossible to complete this thesis without them. Their existence is the most meaningful thing in my life.

Above all, I dedicate this thesis to the memory of my beloved father, Feyzi Kömürcü, who has unfortunately passed away during my PhD studies. He has spent his whole life for the sake of his family and supported our education passionately. I owe my existence and success in my life to him.

ABSTRACT

REALIZATION AND ANALYSIS OF HIGH PERFORMANCE PHYSICAL UNCLONABLE FUNCTIONS BASED ON RING OSCILLATORS

Physical Unclonable Functions (PUFs) are powerful techniques that are proposed recently to address security related problems. They have a wide range of applications including cryptographic key generation and storage, authentication, identity generation, and intellectual property protection. PUFs offer new, low cost, and secure solutions in these areas with their ability to generate chip specific signatures on the fly. In the scope of the thesis study, quality metrics for the robustness and uniqueness properties of PUF circuits are derived. Confidence interval and confidence level concepts are adapted to PUF performance evaluation for the reliability of the results. Theoretical background of Ring Oscillators (ROs) is studied to analyze the effect of the number of stages and measurement time in RO-PUFs. Depending on the theoretical calculations, optimum number of stages and measurement time are determined and the theory is validated via experimental analysis. Then, ordering based RO-PUFs, which aim to maximize the robustness and entropy extraction, are discussed and dynamic programming is adapted for achieving lower complexity in the grouping step. Next, systematic analysis of the bit error probability in ordering based RO-PUFs is performed and area usage vs. robustness tradeoff is presented. Implementation and analysis of error correction codes to maintain the robustness in ordering based RO-PUFs are also discussed. In addition to these, two challenge-response pair enhancement methods for ordering based RO-PUFs are introduced. Finally, effects of aging on ordering based RO-PUFs and compensation mechanisms are presented.

ÖZET

HALKA OSİLATÖRÜ TABANLI VE YÜKSEK PERFORMANSLI FİZİKSEL KLONLANAMAZ FONKSİYON ANALİZİ VE GERÇEKLENMESİ

Fiziksel Klonlanamaz Fonksiyonlar (PUF) güvenlik uygulamalarında kullanılmak üzere tasarlanan güçlü tekniklerdir. Kullanım alanları arasında kriptografik anahtar üretimi ve depolaması, doğrulama, kimlik üretme ve fikri mülkiyet korunumu gibi uygulamalar bulunmaktadır. PUF'lar çalışma esnasında entegre devreye özgü imza üretme kabiliyetleriyle yeni, düşük maliyetli ve yüksek güvenlikli çözümler sunmaktadırlar. Tez çalışması kapsamında, PUF çıktılarının sağlamlık ve özgünlüklerini belirleyebilemek için kalite ölçütleri belirlenmiştir. Güven aralığı ve güven seviyesi kavramları PUF değerlendirme sürecine entegre edilerek sonuçların güvenilirliğinin sağlanması hedeflenmiştir. Halka osilatörlerin (RO) kuramsal temelleri çalışılarak, RO-PUF'lar için katman sayısı ve ölçüm zamanını en iyileme yöntemleri araştırılmıştır. Bu konuda kuramsal olarak belirlenen yöntemler deneysel verilerle de doğrulanmıştır. Sonrasında, RO-PUF çıktılarının sağlamlığını ve entropi kullanımını arttırmak için önerilen sıralama tabanlı RO-PUF sunulmuş, Dinamik Programlama gruplama adımına entegre edilerek algoritma karmaşıklığının azalması sağlanmıştır. Bunların yanında, sıralama tabanlı RO-PUF'larda hata analizi yapılmış ve alan kullanımı ile sağlamlık ilişkisi incelenmiştir. Olası hatalı çıktıları düzeltmek için hata düzeltme kodları uygulanmış ve analiz edilmiştir. Bunların yanında, sıralama tabanlı RO-PUF'lara yönelik iki adet sorgu-cevap çifti üretme metodu geliştirilmiş ve bu yöntemlerin güvenli bir şekilde kullanılabilmesi için üç adet güvenli kullanım senaryosu sunulmuştur. Son olarak, yaşlanmanın sıralama tabanlı RO-PUF'lar üzerindeki etkileri incelenmiş ve bu yaşlanma etkilerini düzeltme teknikleri geliştirilmiştir.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii

ABSTRACT v

ÖZET vi

LIST OF FIGURES x

LIST OF TABLES xvi

LIST OF SYMBOLS xviii

LIST OF ACRONYMS/ABBREVIATIONS xix

1. INTRODUCTION 1
    1.1. Motivation 1
    1.2. Contributions and Organization of the Thesis 2
2. PHYSICAL UNCLONABLE FUNCTIONS 4
    2.1. Definition and Applications 4
    2.2. PUF Properties 7
        2.2.1. Uniqueness 7
        2.2.2. Robustness 8
        2.2.3. Unclonability 9
        2.2.4. Unpredictability 9
    2.3. PUF Types 10
        2.3.1. Optical PUF 10
        2.3.2. Coating PUF 10
        2.3.3. Arbiter PUF 11
        2.3.4. Glitch PUF 14
        2.3.5. Memory Based PUF 16
            2.3.5.1. SRAM PUF 16
            2.3.5.2. Butterfly PUF 17
        2.3.6. RST PUF 18
        2.3.7. Ring Oscillator PUF 19
3. DERIVATION OF QUALITY METRICS 25

3.1. Derivation of Quality Metrics for Uniqueness 25
3.2. Derivation of Quality Metrics for Robustness 28
3.3. Confidence Interval and Confidence Level Concepts for PUF Evaluation 31
3.4. Implementation of Two Basic RO-PUF Structures 32
3.5. Analysis of Experimental Data 33
4. OPTIMIZING RO-PUF CIRCUITS 37
4.1. Effect of the Number of Stages 38
4.2. Effect of the Measurement Time 41
4.3. Experimental Analysis and Validation of the Theory 43
5. ORDERING BASED RO-PUF CIRCUITS 48
5.1. Maximizing Robustness and Entropy in RO-PUF Circuits 48
5.2. Adapting Dynamic Programming to RO-PUFs 53
5.2.1. Proof of Lower Complexity of DP Method over LISA by Contradiction 56
5.3. Experimental Analysis and Validation of the Theory 59
6. ERROR PROBABILITY IN ORDERING BASED RO-PUFs AND ANALYSIS OF ERROR CORRECTION CODES 69
6.1. Systematic Analysis of Bit Error Probability 69
6.1.1. Group Length Analysis 70
6.1.2. Symbol Error Probability and Validation 71
6.1.3. Symbol Error to Bit Error Conversion and Bit Error Probability 75
6.1.4. Worst Case Bit Errors per Problematic RO Pair 77
6.2. Area Usage vs. Robustness in Ordering Based RO-PUFs 79
6.3. Implementation and Analysis of Error Correction Codes 83
7. CRP GENERATION 92
7.1. CRP Concept in PUFs 92
7.1.1. CRP Properties 93
7.1.2. Importance of Large Number of CRPs 93
7.2. Enhanced CRP Set with the f_{thp} Selection Method 94
7.3. Enhanced CRP Set with RO Selection Method 98
7.4. Comparison of the f_{thp} Selection and RO Selection Methods 103

8. EFFECTS OF AGING AND COMPENSATION MECHANISMS 108
8.1. Aging Mechanisms and PUFs 108
8.2. Ordering Based RO-PUFs and Aging 110
8.3. Accelerated Aging Test and Analysis of Results 111
8.4. Effects of Aging on Ordering Based RO-PUFs and Compensation Mechanisms 116
9. CONCLUSION 121
APPENDIX A: LIST OF PUBLICATIONS 124
REFERENCES 126

LIST OF FIGURES

Figure 2.1. CRP supporting and non supporting PUFs. 5

Figure 2.2. PUF based authentication [1]. 6

Figure 2.3. Cryptographic key generation with PUFs [1]. 7

Figure 2.4. Optical PUF measurement setup and pattern from a silicon surface [2]. 10

Figure 2.5. Schematic cross-section of a coating PUF [3]. 11

Figure 2.6. Arbiter based PUF circuit. 12

Figure 2.7. Feed-forward arbiter based PUF. 12

Figure 2.8. Tristate buffer and truth table. 13

Figure 2.9. Delay unit in tristate buffers. 13

Figure 2.10. PUF architecture built with tristate buffers. 14

Figure 2.11. Input network [4]. 15

Figure 2.12. Output network [4]. 15

Figure 2.13. BPUF operation [5]. 18

Figure 2.14. LUT based RST-PUF circuit [6]. 19

Figure 2.15. RST PUF output generation [6]. 20

Figure 2.16. 5-stage RO schematic with enable input. 20

Figure 2.17. PUF output bit generation by conventional systems. 21

Figure 2.18. Effect of temperature variations on RO-PUF. 21

Figure 2.19. Configurable RO-PUF scheme [7]. 22

Figure 2.20. Location of ROs used for 1 bit generation. 24

Figure 2.21. Output generation mechanism proposed in [8]. 24

Figure 3.1. Number of measurements - confidence level relation with confidence interval of 0.01 and STD of 0.064. 32

Figure 3.2. Number of measurements - confidence level relation with confidence interval of 0.001 and STD of 0.064. 33

Figure 4.1. 5-stage RO schematic with delay components mentioned. 39

Figure 4.2. Frequency of ROs vs. Stage Number. 44

Figure 4.3. Accumulated Jitter vs. Stage Number. 45

Figure 4.4. Spatial Variation vs. Stage Number. 45

Figure 4.5. Accumulated Jitter and Spatial Variation. 46

Figure 4.6. Error Rate vs. Meas. Time for RO-PUF structure. 46

Figure 5.1. PUF output generation by ordering based systems. 50

Figure 5.2. LISA in pseudo code [9]. 51

Figure 5.3. Determining f_{thp} in pseudo code. 52

Figure 5.4. Frequency of ROs measured at 20^{o}C and 100^{o}C. 53

Figure 5.5. Frequency deviation of ROs. 54

Figure 5.6. DP sample execution for 12 elements. 56

Figure 5.7. Dynamic Programming in pseudo code. 57

Figure 5.8. Search for the largest group. 58

Figure 5.9. PUF output bit generation comparison. 61

Figure 5.10. Execution time of algorithms in Matlab. 62

Figure 5.11. Output generation sample execution. 65

Figure 5.12. Ordering circuit sample execution. 66

Figure 5.13. Output generation sample execution. 67

Figure 5.14. Output generation in pseudo code. 67

Figure 6.1. Number of groups per group length for different f_{thp} values. 71

Figure 6.2. Probability of two ROs to be placed in the same group. 73

Figure 6.3. Effect of ordering on symbol error [10]. 74

Figure 6.4. Symbol error rate. 75

Figure 6.5. Number of bit errors per group. 77

Figure 6.6. Bit error probability per generated bit. 78

Figure 6.7. Bit error probability per problematic RO pair. 79

Figure 6.8. Maximum error vs. group size. 80

Figure 6.9. Worst case symbol error probability vs. group size. 81

Figure 6.10. Bit error probability vs. f_{thp} chosen. 82

Figure 6.11. A system utilizing ECC. 84

Figure 6.12. Use of ECC in PUF based systems. 86

Figure 6.13. Area consumptions of BCH encoders with different data lengths and error correction capabilities. 87

Figure 6.14. Area consumptions of BCH decoders with different data lengths and error correction capabilities. 88

Figure 6.15. Area consumptions of BCH encoders with different error correction capabilities. 89

Figure 6.16. Area consumptions of BCH decoders with different error correction capabilities. 90

Figure 6.17. Execution times of BCH encoders with different data lengths and error correction capabilities. 90

Figure 6.18. Execution times of BCH decoders with different data lengths and error correction capabilities. 91

Figure 7.1. The relation between the f_{thpmin}, f_{thpmax}, and f_{thpdif} values. 95

Figure 7.2. DP sample execution for 12 elements with an f_{thp} value of 2.5 MHz. 96

Figure 7.3. Uniqueness Quality vs. f_{thpdif} Parameter. 99

Figure 7.4. Revised part of the DP algorithm in pseudo code. 100

Figure 7.5. DP algorithm sample execution for RO selection method. 101

Figure 7.6. Number of CRPs vs. additional ROs. 102

Figure 7.7. Uniqueness quality vs. additional ROs. 104

Figure 8.1. Effect of aging on conventional PUFs. 110

Figure 8.2. Effect of aging on ordering based RO-PUFs. 112

Figure 8.3. Accelerated aging test setup. 113

Figure 8.4. Mean frequency of ROs vs. accelerated aging test duration. 115

Figure 8.5. Distribution of ROs depending on frequency reduction due to aging.116

Figure 8.6. STD of frequency reduction due to aging. 117

Figure 8.7. Noise on RO frequency measurements vs. aging. 118

LIST OF TABLES

Table 3.1. Uniqueness and Robustness results of RO_PUF1 and RO_PUF2. . 35

Table 3.2. Relation between the number of circuits to be identified, PUF output length and security level. 36

Table 4.1. Notations and Meanings. 47

Table 5.1. Maximum Frequency Deviation of ROs due to Temperature Change. 60

Table 5.2. Area Reduction with DP. 62

Table 5.3. Uniqueness Results. 63

Table 5.4. Area Utilization of Frequency Detection Circuit. 64

Table 5.5. Area Utilization of RO-PUFs for Virtex5 Devices. 68

Table 5.6. Area Utilization of RO-PUFs for Spartan3 Devices. 68

Table 6.1. Output Generation Mapping. 76

Table 6.2. Number of Problematic RO Pairs. 81

Table 6.3. Simultaneous Symbol error probabilities for non-ideal f_{thp} values. . 83

Table 7.1. f_{thp} Selection Method Area Consumption vs. CRP count. 98

Table 7.2. Probability of Output Couples with HD Less than the Minimum HD Defined. 105

Table 7.3. Minimum HD Among 128 bit Outputs within 10,000 CRPs Based on the Number of ROs. 106

Table 7.4. Comparison of Ordering Based RO-PUFs in terms of Response Uniqueness. 107

Table 8.1. Equivalent IC Working Duration of Accelerated Aging Test. 114

Table 8.2. RO Frequency Reduction due to Aging vs. RO Location. 116

Table 8.3. Number of Problematic RO Pairs. 118

Table 8.4. Effect of Aging on f_{thp} Parameter and Corresponding Area Overhead.120

LIST OF SYMBOLS

f_{th}	Frequency Threshold
f_{thp}	Pre-determined Frequency Threshold

LIST OF ACRONYMS/ABBREVIATIONS

AAT	Accelerated Aging Test
AES	Advanced Encryption Standard
ASIC	Application Specific Integrated Circuit
BCH	Bose, Chaudhuri, and Hocquenghem
BPUF	Butterfly PUF
CMOS	Complementary Metal Oxide Semiconductor
CRP	Challenge-Response Pair
DP	Dynamic Programming
ECC	Error Correction Codes
FAR	False Acceptance Rate
FF	Flip-Flops
FPGA	Field Programmable Gate Array
FRR	False Rejection Rate
GVB	Gilbert-Varshamov Bound
HCI	Hot Carrier Injection
HD	Hamming Distance
Hi-Z	High Impedance
IC	Integrated Circuit
ID	Identity
IP	Intellectual Property
LUT	Look-Up Table
MV	Majority Voting
NBTI	Negative-Bias Temperature Instability
NMOS	N-Type Metal Oxide Semiconductor
NVM	Non-Volatile Memory
LISA	Longest Increasing Subsequence-Based Grouping Algorithm
NOC	Nominal Operating Conditions
PBTI	Positive-Bias Temperature Instability

PMOS	P-Type Metal Oxide Semiconductor
PS	Patience Sorting
PUF	Physical Unclonable Function
RFID	Radio Frequency Identification
RO	Ring Oscillator
RV	Random Variable
SRAM	Static Random Access Memory
STD	Standard Deviation
TDDB	Temperature-Dependent Dielectric Breakdown
UART	Universal Asynchronous Receiver/Transmitter
VT	Varying Temperature

1. INTRODUCTION

1.1. Motivation

Physical Unclonable Functions (PUFs) are recently proposed circuit primitives that are utilized mainly in security applications. PUFs provide the ability of creating chip specific signatures depending on the small mismatches present in the IC. Even though several different structures are developed and an important amount of work is presented in the literature, PUFs are still far from maturity and the subject requires significantly more work to be done.

One major gap in the literature pertains to the reliability of PUF circuits. Widely accepted and standardized evaluation methods of PUF circuits are not present in the literature. Therefore, performance of the structures are generally measured with basic formulations that do not maintain reliable evaluation results. 100% robust output generation is problematic for PUF circuits due to their noisy nature. Ordering based Ring Oscillator (RO) PUFs did not receive much attention in spite of their ability to generate reliable outputs and provide high entropy extraction even in Field Programmable Gate Array (FPGA) environment. Challenge-Response Pair (CRP) concept, which is very important for certain type of applications, such as authentication, has also not yet been defined for ordering based RO-PUF circuits. Attacks on PUF circuits and attack resistant designs are also productive research areas. Effects of aging on PUF circuits, an important topic to study the long term behavior of these circuits, is another area that drew little attention in the literature yet. In addition to the topics mentioned, PUFs require more work in many aspects for better performance and widespread usage. In the scope of the thesis, many of these issues are addressed and significant improvements in the design and evaluation of PUFs are achieved.

1.2. Contributions and Organization of the Thesis

The thesis is composed of 9 chapters. Organization and contributions of the thesis can be summarized as follows.

Chapter 2 focuses on the background of PUFs. Definition of PUF is given and applications utilizing these circuits are presented. Next, PUF properties, uniqueness, robustness, unclonability, and unpredictability are discussed. The chapter is concluded by presenting the standard PUF types proposed in the literature.

In Chapter 3, quality metrics for the uniqueness and robustness properties of PUF circuits are derived for a fair performance evaluation. Next, confidence interval and confidence level concepts are adapted to PUF evaluation in order to maintain the trustworthiness of the performance results. Then, two basic RO-PUF structures proposed in the literature are implemented in FPGA environment. Finally, outputs of both of the structures are collected using serial port and analyzed in Matlab environment according to the proposed quality metrics.

Theoretical foundations of ROs are discussed in Chapter 4. Based on these theoretical foundations, optimization techniques for the number of stages and measurement time of the RO-PUF structures are presented. Then, proposed techniques are validated using real implementation results collected from FPGAs. Based on the theoretical foundations and analysis results, optimum number of stages and measurement time are determined for best performing RO-PUF structures.

Maximizing the robustness and entropy extraction in RO-PUF structures are discussed in Chapter 5. Dynamic programming (DP) is adapted to ordering-based RO-PUF structures and its lower complexity compared to previous approaches is proved. Finally, the effectiveness of the proposed approach is validated via experimental analysis.

Chapter 6 focuses on error probability and correction in ordering based RO-

PUFs. First, a systematic analysis of bit error probability is presented. Then, the area consumption and robustness relation is discussed. Finally, Bose, Chaudhuri, and Hocquenghem (BCH) codes, which are multi-bit correcting type Error Correction Codes (ECC), are implemented due to their suitability for error correction in PUF circuits. Their area consumption and timing performances are analyzed for different output lengths and error correction capabilities.

CRP concept is presented in Chapter 7. CRP properties and importance of large number of CRPs are discussed in the first section. Two CRP enhancement methods, f_{thp} selection and RO selection, are proposed to overcome the drawbacks of small number of CRPs in RO-PUFs. With these methods, shortage of CRPs in RO-PUFs is eliminated with reasonable area overhead. f_{thp} selection and RO selection methods are compared in terms of CRP quality, area consumption, and timing efficiency.

Aging mechanisms and effects of aging on PUFs are discussed in Chapter 8. The results of an Accelerated Aging Test (AAT) applied in FPGA environment to determine the behavior change of ROs due to aging is presented. Effects of aging on ordering based RO-PUFs are analyzed and a compensation mechanism to maintain 100% reliability even after long years of operation is proposed. Finally, Chapter 9 concludes the thesis.

2. PHYSICAL UNCLONABLE FUNCTIONS

2.1. Definition and Applications

PUF is a relatively new concept introduced by Pappu *et al.* in 2001 [11,12], which has the unique capability of generating chip specific signatures during operation. Their unclonability is a result of uncontrollable components present in the manufacturing process, such as oxide thickness, threshold voltage, or doping concentrations. Since it is impossible to replicate these process variations in another die, generated signatures are unique and chip specific. Unclonability, uniqueness, robustness, and unpredictability are the main features that each PUF should provide. In addition to these, they should be easy to evaluate and hard to characterize [13].

PUFs provide efficient solutions to security related problems. IP protection, authentication, identity (ID) generation, and cryptographic key generation can be considered as the main application areas that PUF circuits provide powerful solutions [14]. The main advantages of PUFs over conventional techniques are their low cost, ease of integration, and resilience against physical and side channel attacks [4]. In addition to these, they eliminate the need for non-volatile memory and a secure channel to the device for ID or key storage, which improves the security of the system significantly [1]. PUFs also do not require any special manufacturing, programming, or testing steps and some PUF structures are also suitable for FPGA implementations as well.

PUFs can be divided into two types; CRP supporting and CRP non-supporting. CRP non-supporting systems generate outputs without using any input applied to the circuit. In these systems, a single bitstream is generated by each implementation of the structure. In CRP supporting systems, outputs (responses) are generated depending on the applied inputs (challenges), in addition to the intrinsic characteristics of the circuit. In this context, CRP supporting PUFs can be defined as a mathematical function that maps challenges C_i to responses R_i, which can be written as $R_i \Leftarrow$

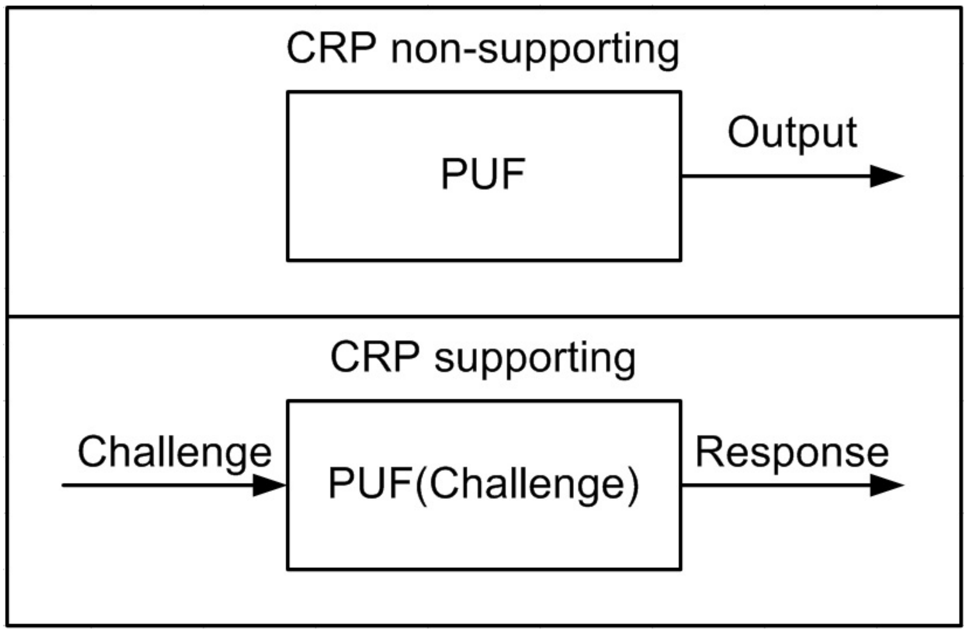


Figure 2.1. CRP supporting and non supporting PUFs.

PUF(C_i). CRP supporting and non-supporting PUFs are illustrated in Figure 2.1.

PUFs have been utilized in many security related applications after the first successful implementations were realized. Authentication via CRPs is an effective solution provided by PUF circuits. Authentication is performed in two phases when PUFs are employed. In the initialization phase, a number of CRPs are stored in a secure database. In the usage phase, a challenge that is already recorded in the database is sent to the device by the application that needs to verify the system. Then, the response of the device is compared with the response on the database. If the responses match, the device is considered to be authentic. In this system, each CRP should only be used once to prevent the success of replay attacks [2]. Authentication is illustrated in Figure 2.2.

By removing the noise present in the outputs, PUFs can be used as cryptographic key generators as well. Using the keys generated by the PUF circuit, all security applications utilizing standard symmetric and asymmetric cryptographic techniques can be employed [15, 16]. Generating the key using a PUF eliminates the need for storing the key on a battery-backed random access or non-volatile memory, decreasing the cost significantly. In addition to these, probability of theft of the key is significantly

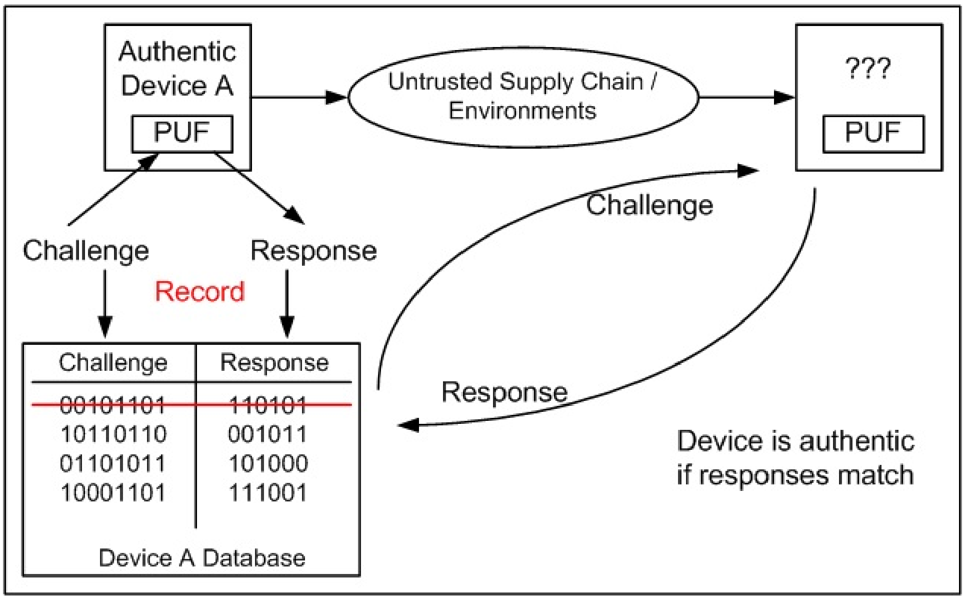


Figure 2.2. PUF based authentication [1].

reduced, since the key is deleted from the system immediately when used by the cryptographic algorithm. Generating the key on another device is also impossible due to the unclonability property of PUFs, which maintain the system security [17,18]. A sample scheme for key generation is illustrated in Figure 2.3.

Intellectual Property (IP) protection is another area that PUF circuits are employed. Since bitstreams are loaded to FPGAs every time the system is powered-up, intruders may copy the bitstream via using a logic analyzer or oscilloscope. With this method, IPs can be used in other systems without the information of the IP developer. Loading bitstreams to the FPGA in an encrypted form and decrypting the IP using the key that is already loaded to the device is a proper method against IP theft [17,19]. Generating the key on the fly using PUF is an efficient and secure way for such systems with the advantages discussed in the previous paragraph. As an alternative, public key authentication system for IP protection is proposed in [20]. In public key based IP protection systems, public key does not leave the device in any case; hence, increasing the system security.

PUF circuits are utilized for identification as well. In systems such as Radio Frequency Identification (RFID) ICs, IDs can be generated by PUFs, since they have

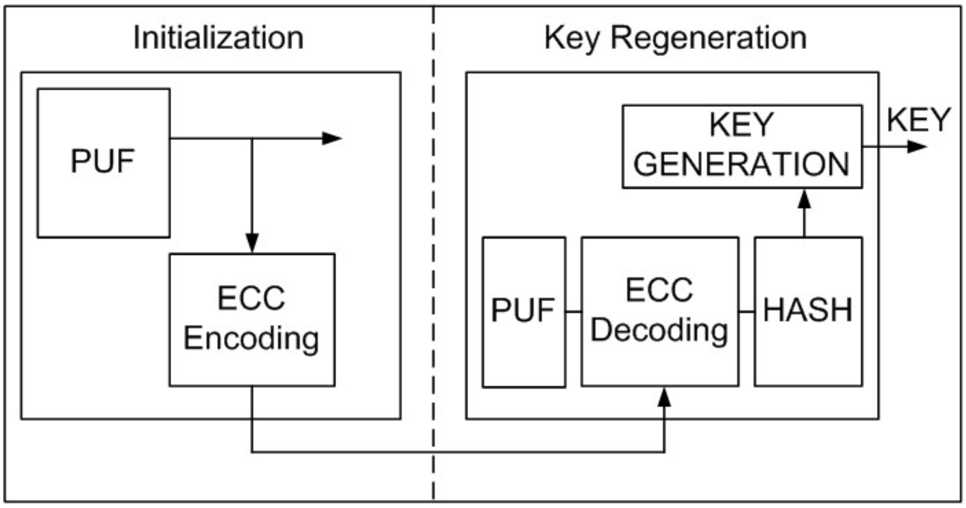


Figure 2.3. Cryptographic key generation with PUFs [1].

important advantages over traditional mechanisms. For instance, an attacker can easily copy the ID of one integrated circuit (IC) to another, which is not possible when the ID is generated by PUFs. In simple RFID tags, cloned tags are not distinguishable from authentic ones [21]. In addition to the areas that are mentioned so far, PUFs are used in different areas such as vehicle system security, seals for insurance applications [22], and error detection methods in finite state machines as well [23].

2.2. PUF Properties

2.2.1. Uniqueness

Uniqueness, which is also called inter-PUF variation, is the variation of the outputs generated by the same PUF structure on different IC instances. Ideally, outputs collected from different ICs should be statistically independent and uniformly distributed. In such a system, no correlation is present between different outputs. If the uniqueness property of a PUF is weak, different ICs may generate similar responses and identifying different circuits will not be possible. Highly unique outputs are very important for a system that utilizes PUF circuits to maintain security. The major quality metric for uniqueness is the inter-PUF Hamming distance (HD), which can be calculated as

$$U\_QM1 = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{HD(R_i, R_j)}{n}, \tag{2.1}$$

where k is the total number of outputs collected from different ICs, R_i is the response of ith circuit, HD is the Hamming distance function, and n is the total number of HD operations. The ideal value for U_QM1 is 0.5.

2.2.2. Robustness

Robustness, which is also called intra-PUF variation, is related to the number of bits that change value between different readings from the same IC. In an ideal system, intra-PUF variation should be zero, meaning that the outputs collected from the same circuit should be the same, independent of time and environmental conditions. However, since the outputs generated by PUF circuits depend on small mismatches present in the internal characteristics of the IC, environmental variations, such as temperature, supply voltage, and crosstalk may lead to generation of unreliable bits in the outputs collected consecutively [24]. The major quality metric for robustness is the inter-PUF HD, which can be calculated as

$$R\_QM1 = \frac{1}{x} \sum_{y=1}^{x} \frac{HD(R_i, R'_{i,y})}{n}, \tag{2.2}$$

where x is the total number of consecutive measurements from the same IC, R_i is the reference response of the IC, $R'_{i,y}$ is the yth output collected, and n is the total number of HD operations. The ideal value for R_QM1 is 0.

Since noisy data is not acceptable for applications such as cryptographic key generation, applying appropriate post-processing techniques such as ECC and extraction algorithms are required for noise-free output generation [25]. However, post-processing techniques decrease the area, time, and power efficiency of the system, and increase the complexity significantly. These overheads are also very dependent on the amount of unreliable bits at the output. Therefore, generating higher quality PUF responses is another aim of designers to minimize the post-processing cost. An alternative PUF structure that generates noise-free outputs without the need of ECC is presented within the scope of this work.

2.2.3. Unclonability

Unclonability is a fundamental behavior of PUFs that indicates the impossibility of building two identical PUF circuits that respond similarly to the same challenges. In addition to this, unclonability property indicates that it is very hard, time consuming, and practically impossible to build an accurate mathematical model of a PUF that will enable computation of the responses to the challenges, without using the PUF itself. The core of the unclonability property of PUFs is the uncontrollable process variations present in the system.

2.2.4. Unpredictability

Unpredictability is another key feature of PUF circuits. According to the unpredictability principle, even if the environmental conditions, structure, and layout of a PUF are known, responses to the challenges should be still unpredictable. In addition to this, the system should maintain that even if infinitely many CRPs are known, response to a new challenge should be still unpredictable. If the unpredictability property of the utilized PUF structure is low, system security will be threatened.

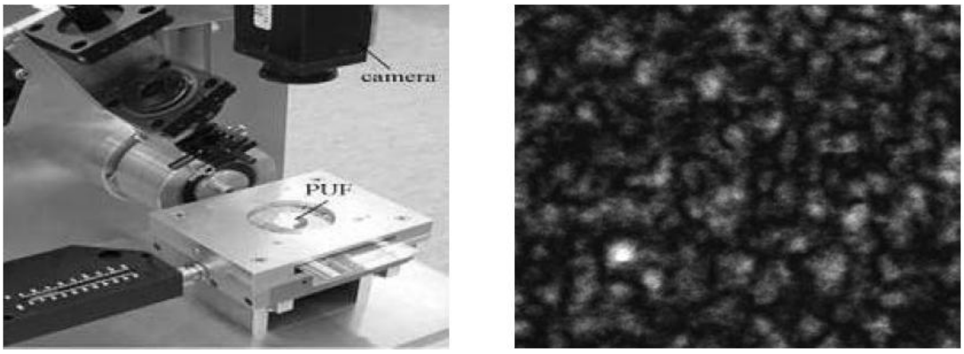


Figure 2.4. Optical PUF measurement setup and pattern from a silicon surface [2].

2.3. PUF Types

2.3.1. Optical PUF

Optical PUF is the first structure developed in the name of physical one-way functions in 2001 [11, 12]. In this structure, bubble filled transparent epoxy is applied on top of the wafer and laser is shined on the sample to create a speckle pattern. Since this pattern is dependent on the material and thickness of the wafer, and the property and distribution of the epoxy, different ICs will have unique reflections that enable the generation of unique signatures. Even though a large number of CRPs can be generated via Optical PUFs by changing the wavelength, angle, or position of the laser, they are not very practical to use in the field, since measurement devices are quite complicated.

Reconfigurability is an advantage of optical PUFs, which enables changing the signature or key permanently when needed [26]. Reconfigurability is achieved via applying a high energy laser beam to the IC, which changes the optical properties of the epoxy. An optical PUF measurement setup and a pattern recorded from a silicon surface are presented in Figure 2.4.

2.3.2. Coating PUF

Coating type PUF was presented in 2006 [3]. In this structure, the IC is covered with a protective matrix coating doped with random dielectric particles. Then, a top

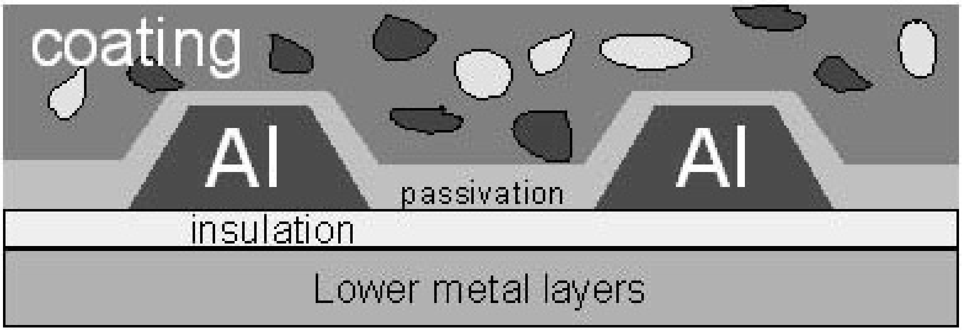


Figure 2.5. Schematic cross-section of a coating PUF [3].

metal layer is built on top of the IC to measure the local capacitance of the coating. Measured capacitance value is used to characterize the circuit and generate the unique signature of the device. Schematic cross-section of a coating PUF is illustrated in Figure 2.5. It is shown in [3] that 600 bits/mm^2 of signature can be generated using 200 capacitance sensors/mm^2. The main disadvantage of this method is its impracticality due to the requirement of additional processing steps that complicates the implementation and increases the cost.

2.3.3. Arbiter PUF

Varying timing behavior of elements on ICs are the basis for Arbiter PUF structures presented in [27, 28]. In arbiter PUFs, a number of delay elements that construct two parallel paths are connected serially and a rising signal is applied to both paths [29, 30]. At the end of these paths, an arbiter determines the signal that arrives at the other end faster and generates a one bit response as shown in Figure 2.6. In arbiter PUFs, multiplexers are used as delay elements to carry the input signals to the outputs. According to the value of the select signal that controls both of the multiplexers, one input passes through the first gate and the other passes through the second gate or vice versa. The challenge applied to the arbiter PUF determines the path that the signals will follow and the response will be one bit 0 or 1, based on the arrival ordering of the two input signals. Arbiter PUF generates $2n$ possible delay paths using n delay elements. In order to generate an m bit response, this structure can be duplicated m times or alternatively, m serial measurements can be taken by applying m different challenges.

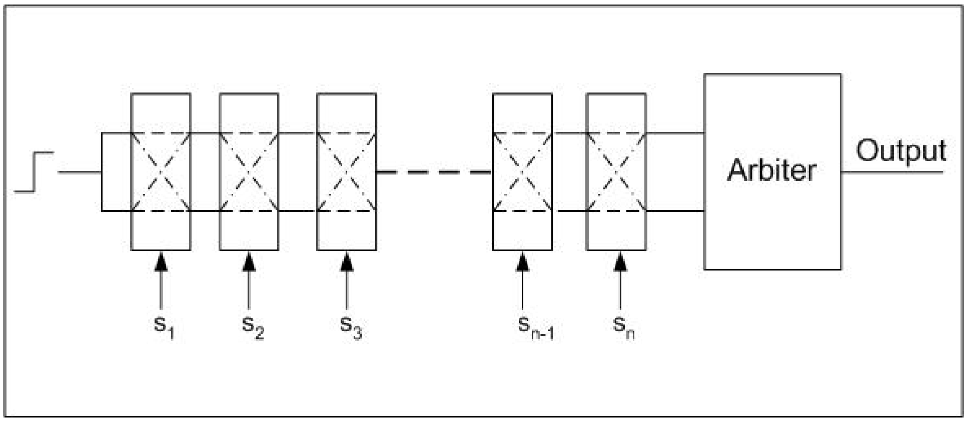


Figure 2.6. Arbiter based PUF circuit.

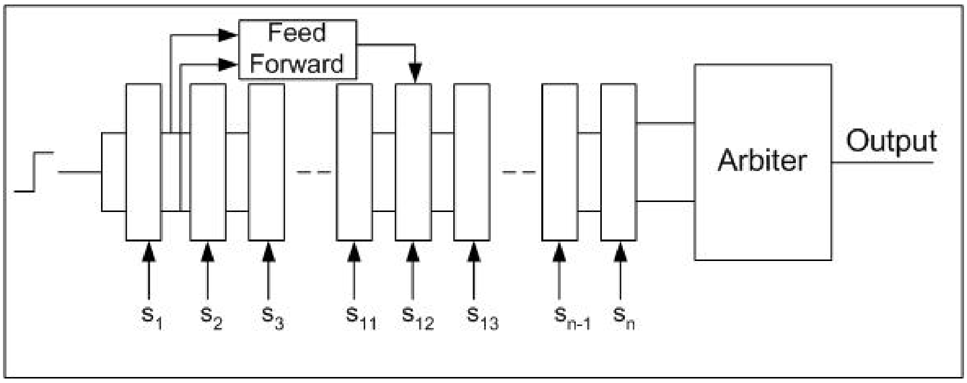
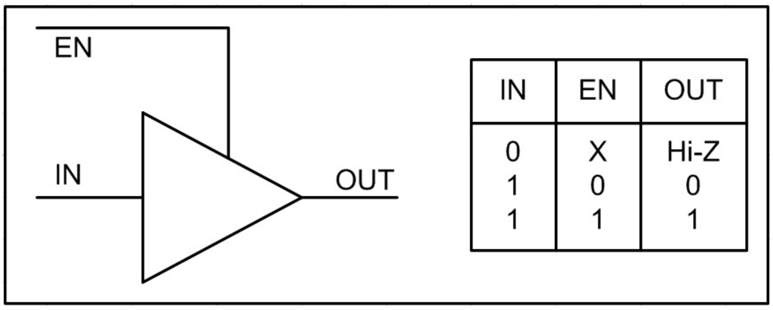


Figure 2.7. Feed-forward arbiter based PUF.

The main problem of arbiter based structures is their vulnerability against modeling attacks. The attacker may solve the behavior of a PUF after collecting a certain number of CRPs [31, 32]. To overcome this problem, feed forward arbiter structure was presented by Lim *et al.* [29]. With this scheme, non-linearity is added to the PUF to harden the modeling attacks as illustrated in Figure 2.7. According to the analysis presented, feed forward arbiter PUF structure has an inter-PUF variation of 38% and an intra-PUF variation of 4.5%. It is also reported that the maximum working frequency is 100 MHz and the power consumption of an 8 bit PUF is 137 μW with 0.18 μm TSMC process.

Tristate buffer based PUF is very similar to the arbiter PUF and it was proposed by Sunar *et al.* in 2008 [33]. In this structure, a certain number of switches are connected serially and each switch is controlled by one bit of the challenge in a similar fashion. The main difference from the multiplexer based structure is the type of the



IN	EN	OUT
0	X	Hi-Z
1	0	0
1	1	1

Figure 2.8. Tristate buffer and truth table.

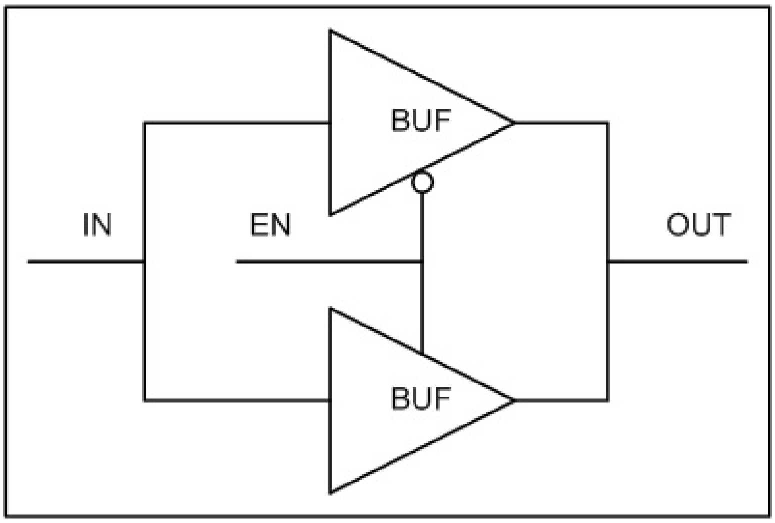


Figure 2.9. Delay unit in tristate buffers.

switch element. In this circuit, two tristate buffers are used to build the switch, rather than using multiplexers. The signals are routed through separate lines in the switch element, instead of interleaved routing within the multiplexer. Tristate buffers have three output states as 1, 0, and high impedance (Hi-Z), as shown in Figure 2.8.

Outputs of the two tristate buffers are connected to build a delay element with two possible paths. However, if both buffers become active at the same time, the circuit is under risk of being short circuited. To prevent such a condition, enable signals of two buffers should be complements of each other. In this case, the input signal will pass through either one of the buffers as illustrated in Figure 2.9.

In the next step, two parallel delay paths are constructed by cascading the delay units serially. Finally, the inputs of these delay lines are connected to each other and the outputs are fed to an arbiter. The arbiter is designed as a single Flip-Flop

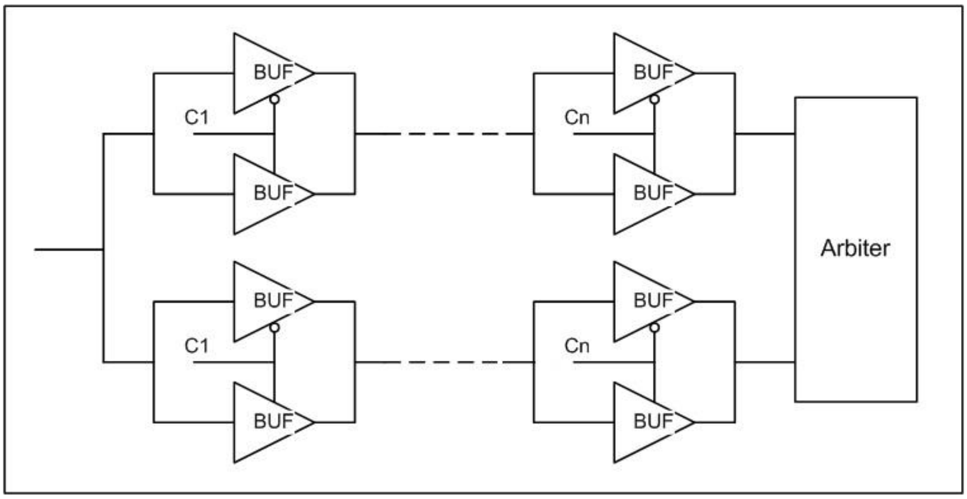


Figure 2.10. PUF architecture built with tristate buffers.

(FF), whose data and clock inputs are connected to the outputs of the delay lines. According to the delay difference of the paths, the output of the FF gets the value of 1 or 0. The schematic diagram of the proposed structure is presented in Figure 2.10. The advantage of a tristate buffer based PUF compared to the arbiter PUF proposed in [29] is its 20% lower power and area consumption. Reliability problem is not considered in this work.

One of the ways suggested to overcome the vulnerability of arbiter PUFs against modeling attacks is to add input and output networks shown in Figures 2.11 and 2.12 to the system [4, 34]. The aim in the proposed method is to prevent the attacker from reaching the inputs and outputs of the arbiter PUF directly. Another countermeasure presented in the paper is to remove the PUF from the FPGA after the configuration is complete. This prevents the attacker from collecting enough number of CRPs from the structure. In addition to these, attack types are classified and system level countermeasures against modeling are also discussed with some mathematical background on the subject.

2.3.4. Glitch PUF

Glitch PUF, which is another PUF type that exploits the delay variation of circuits, was first presented by Suzuki *et al.* in 2009 [35]. In a glitch PUF, a certain

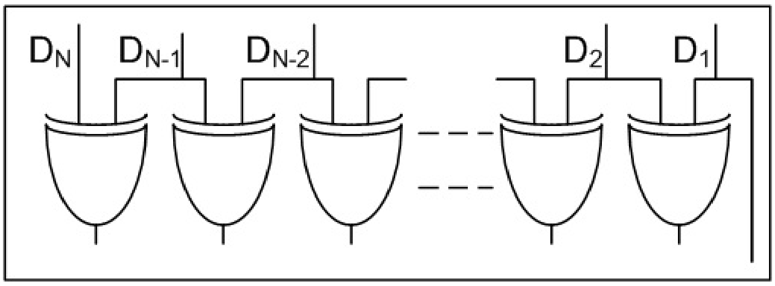


Figure 2.11. Input network [4].

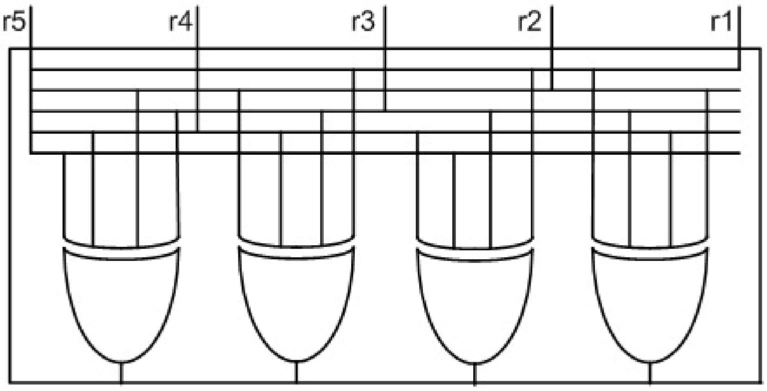


Figure 2.12. Output network [4].

set of input vectors is utilized as challenges and applied to a complex logic block known as glitch generator, and glitches at the output are sampled via registers. Since the delay of interconnects and gates will vary from die to die, different ICs will generate uniquely shaped glitches. In this structure, the logic block should be complex enough to generate enough entropy for valid response bits. For instance, Advanced Encryption Standard (AES) S-box is a proper block for glitch generation due to its complexity. Delay circuits that are used to convert the glitch shapes to output bitstream are also presented in the cited work. In addition to these, a technique based on standard delay format simulation is also presented to model the structure before manufacturing. According to the analysis presented, glitch PUF shows an average inter-PUF HD of 41.5%, and an average intra-PUF HD of 1.3% under Nominal Operating Conditions (NOC).

Two different versions of the glitch PUF are presented by Patel *et al.* in [36–38]. In the first method, glitch counts are used as unique signatures. In this way, either counter approach or one hot state shift register approach is used to count the glitches.

The second method presented is to sample the output of the glitch generator by one of the glitches within the specified block. In these works, a multiplier is used as the glitch generator and input patterns are analyzed to achieve reliable and unique outputs. Stability analysis and improvement methods are also discussed.

2.3.5. Memory Based PUF

2.3.5.1. SRAM PUF. Static Random Access Memory (SRAM) PUF is PUF type that utilizes SRAM in the circuit. Complementary Metal Oxide Semiconductor (CMOS) SRAM is a device with six transistors [39]. Four of the transistors are connected to form two cross-coupled inverters that will hold the stored value and two transistors are used as the load transistors to drive the value applied from outside to the cross coupled inverters. During write operation, the value stored in the SRAM may change, otherwise stable operation is maintained. However, any external signal is not applied to the inverters during power up. Therefore, the value of an SRAM cell will tend to be 0 or 1, depending on the minor voltage differences and mismatches between the two inverters. Since internal parasitics are mostly stable within the IC, SRAM outputs will be the same after consecutive power-ups with high probability. However, internal parasitics are different among ICs and initial condition of SRAM value will differ within different ICs. Based on these properties, SRAM can be used as a PUF [40].

The main advantage of an SRAM PUF is its convenient structure for FPGA implementations [19]. Most of the FPGAs that are in use today include built-in SRAM memory blocks that can be used to store data. However, some of the SRAMs in these products have initial conditions that prevent them from having random values during startup. These types of FPGAs do not allow SRAM PUF implementations.

One of the most important advantages of SRAM PUF is its ease of use since no evaluation circuitry is needed. Since SRAM bits get their value during power up immediately, only read operation is performed to get the output. In this sense, the required key or signature is generated in a short time compared to other PUF structures such as RO-PUF or Arbiter PUF.

Another work that focuses on SRAM PUF is presented in [41]. In this work, cross-coupled NOR structures are utilized as the memory element and Application Specific Integrated Circuits (ASICs) are manufactured rather than using FPGA environment. The main aim of the cited work is to minimize the power consumption.

Finally, a solution to the problem of FPGAs that have initialized FFs is presented in [42]. In the proposed method, initialization flow of FPGAs is modified by changing the bit file to remove the initialization step of FFs. Then, the values that have been settled in the FFs are read out to be used as PUF output. Next, the FFs are initialized again and normal operation flow is continued. According to the results that are presented in the paper, an important amount of post-processing is required to achieve unique signatures due to the present bias at FF output values.

2.3.5.2. Butterfly PUF . Cross-coupled structures are basic building blocks of many storage elements such as SRAMs and FFs. Due to positive feedback induced by the loop, the stored value is retained as long as the supply voltage is present. Butterfly PUF (BPUF) is a structure that behaves similarly to SRAM memories during the power-up phase [5]. BPUF cells are composed of two latches that can be forced to an unstable operating point during the PUF operation. When the latches are released, output of one of the latches can be used as PUF output. BPUF operation is illustrated in Figure 2.13.

The main advantage of BPUF over SRAM PUF is its suitability for all FPGAs [43]. Even though the structure is supported by every FPGA, it is not straight-forward to build BPUF due to the symmetric routing required. Automatic placement and routing cannot achieve proper layout for BPUF implementation. Therefore, manual routing is required. If symmetric routing cannot be achieved, the output of the BPUF cell will be the same on every FPGA and violate the uniqueness property.

According to the results presented in the paper, inter-PUF HD is around 50%, which is the ideal case for PUF structures and 6% of the output bits behave unreliably

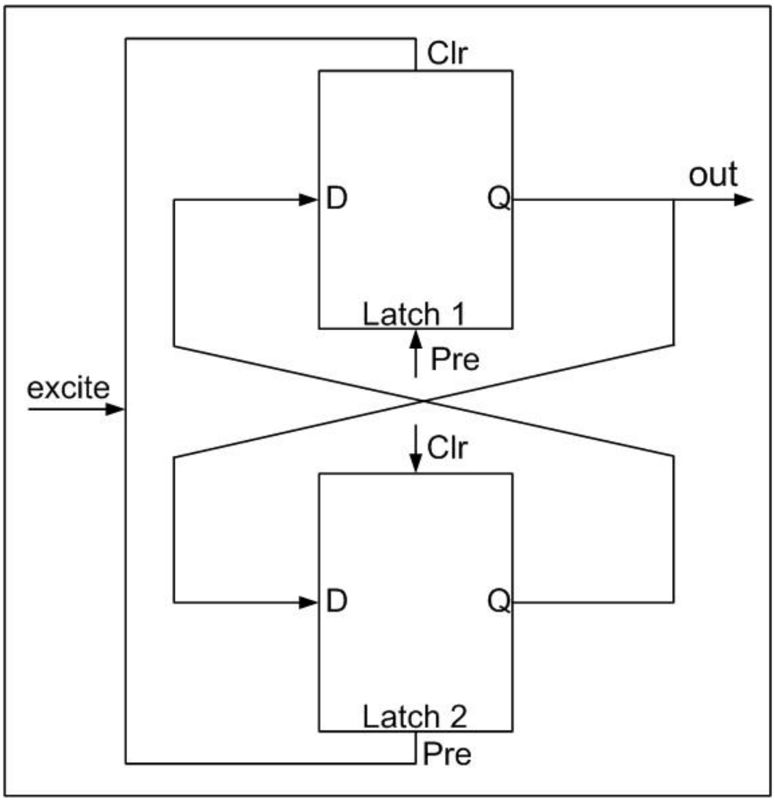


Figure 2.13. BPUF operation [5].

during the temperature change from 20^oC to 80^oC and -20^oC to 20^oC.

2.3.6. RST PUF

A novel PUF design called RST-PUF, which can be built on look-up table (LUT) based FPGAs, is proposed by Anderson *et al*. [6]. In this method, LUTs are configured as shift registers and two LUTs generate one bit PUF response, as illustrated in Figure 2.14. With the rising edge of the *clk* signal, output of each multiplexer switches and depending on the speed differences of the multiplexers, a glitch is generated at the output. This glitch is used to generate one bit PUF output, as shown in Figure 2.15. This structure is implemented just using VHDL coding. Manual place and route is not required, since the whole structure is within a slice and no external routing is required. According to the reliability analysis, 3.6% of the bits are unreliable under changing temperature conditions. This result is similar to many other PUF structures presented.

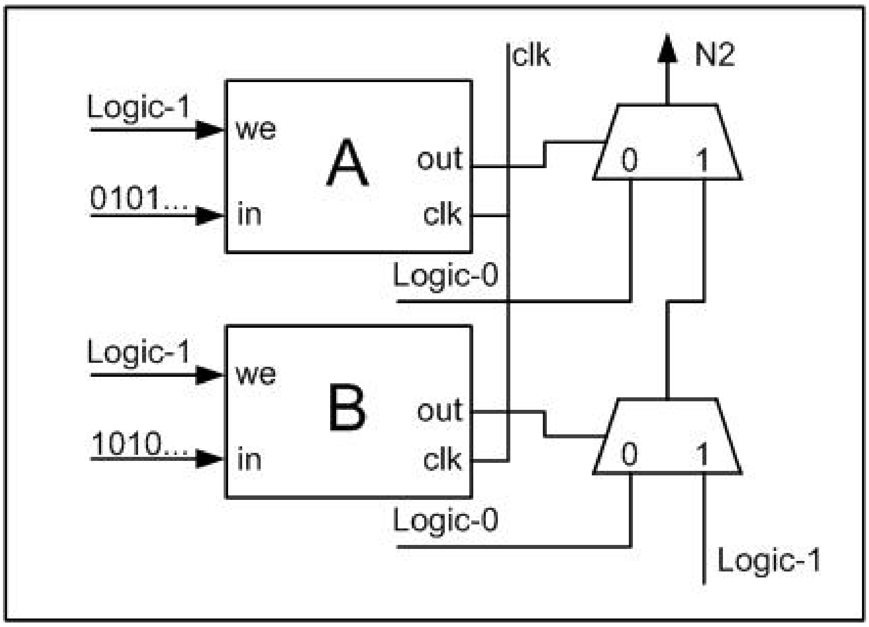


Figure 2.14. LUT based RST-PUF circuit [6].

2.3.7. Ring Oscillator PUF

RO type PUFs that utilize the delay differences of identical structures were first presented by Gassend *et al.* in 2002 [13]. In the cited work, a continuously oscillating variable delay circuit is built. In this circuit, the delay and the frequency of the oscillating loop changes according to the applied challenge. The frequencies of the variable delay circuit are used to generate the PUF outputs afterwards.

In regular RO-PUFs, the output depends on the frequencies of two identical ring oscillators. In this system, one bit response is generated by comparing two ROs. For instance, the output bit can be defined as 0 if RO_1 is faster than RO_2 and can be defined as 1 if RO_2 is faster than RO_1. Since a PUF structure is expected to generate a certain length bitstream, a number of identical ROs are implemented in the circuit and different pairs are selected for each output bit generation via multiplexers. The mechanism to compare the oscillation frequencies of ROs is to utilize counters that will count the number of transitions of the selected ROs in a certain time interval. Finally, a comparator is used to determine the counter with the higher value and produce one bit output accordingly. A five stage RO with enable input is illustrated in Figure 2.16. The reason for using an enable input at the RO is to suspend the ROs that are not

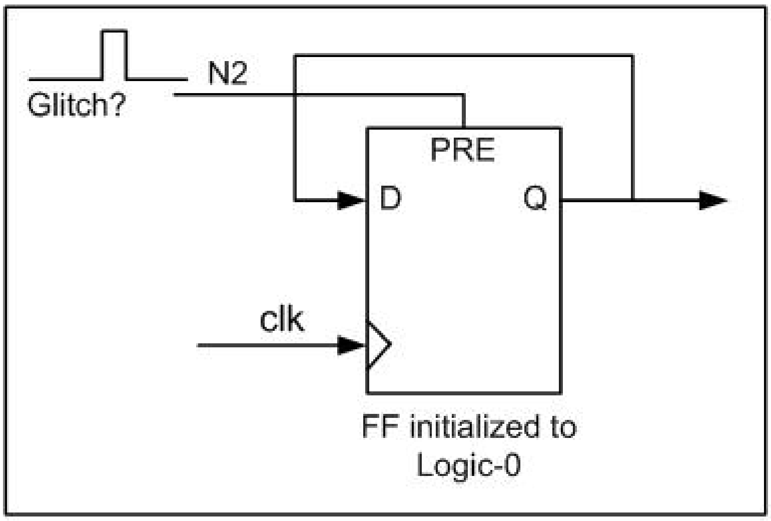


Figure 2.15. RST PUF output generation [6].

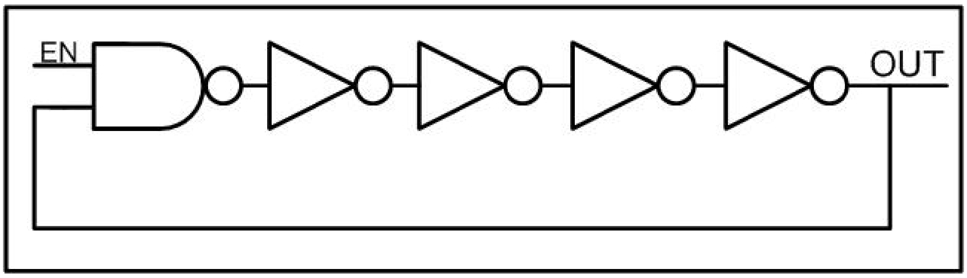


Figure 2.16. 5-stage RO schematic with enable input.

being measured to prevent inter-locking of ROs. The whole system is illustrated in Figure 2.17.

Similar to other silicon PUF types, RO-PUFs are vulnerable to environmental conditions as well. Temperature and supply voltage variations are the main sources of unreliable behavior of RO-PUFs. Due to environmental variations, frequency ordering of two ROs, whose frequencies are very near to each other, may flip as illustrated in Figure 2.18. In addition to these, due to the noise present in the system, two consecutive measurements may differ as well even under the same conditions. To overcome this problem, different structures are presented in the literature.

One of the proposed methods is to group more than two ROs and select two of them whose frequencies are far enough from each other as presented in [1]. In this design, eight ROs are grouped and two of these, which are better separable from each

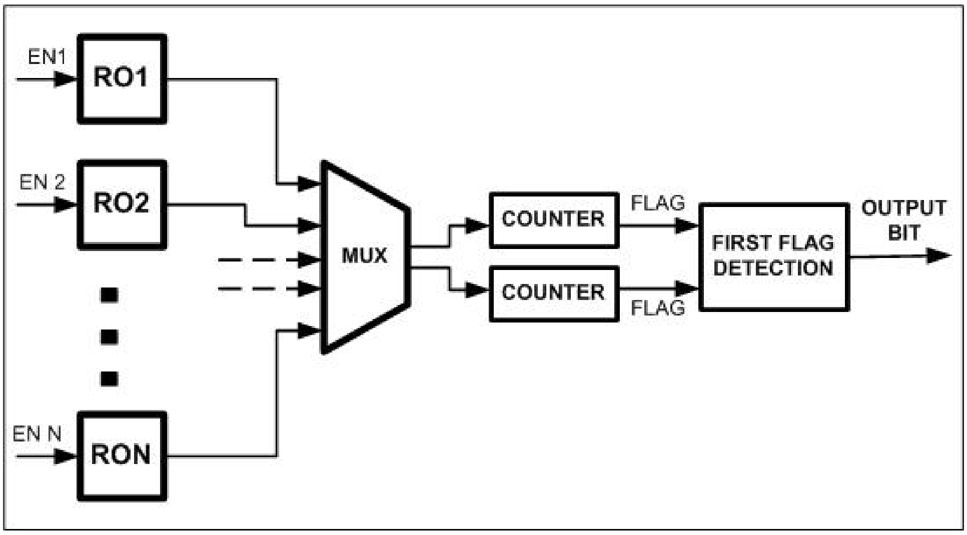


Figure 2.17. PUF output bit generation by conventional systems.

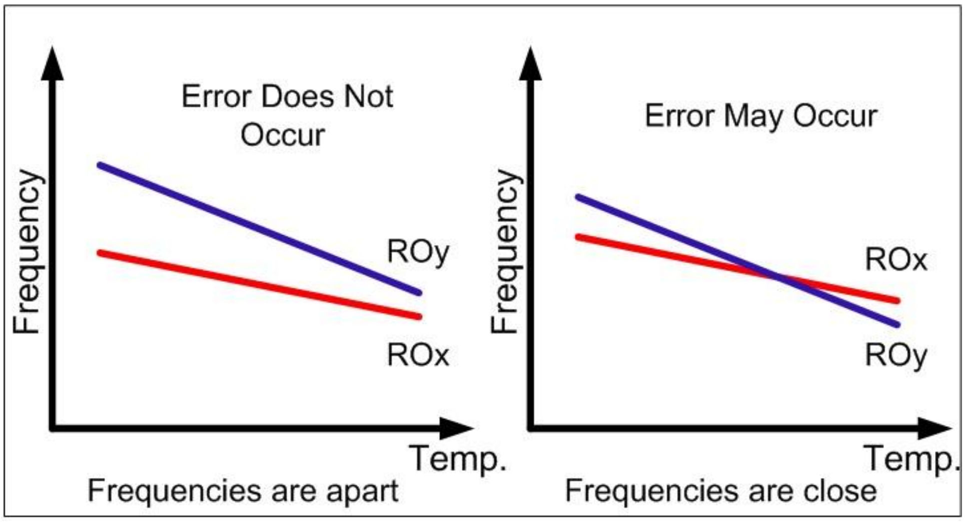


Figure 2.18. Effect of temperature variations on RO-PUF.

other are selected to generate the response bit. In this scheme $n/8$ bits are generated for n ROs. The main problem of this scheme is its high hardware cost, since $n/8$ bits are generated using n ROs. Via this approach, 46% inter-PUF HD and 0.48% intra-PUF HD is achieved. A similar approach utilizing the components within the slices is presented in the name of Configurable RO-PUF [7]. Using three select signals, eight different RO configurations are implemented, as shown in Figure 2.19. Then, the configuration resulting in the highest frequency difference between the two ROs of a pair is selected for generating reliable outputs. The same scheme is improved in [44] by generating 256 different configurations with eight select signals. Both of the methods offer higher reliability without area overhead.

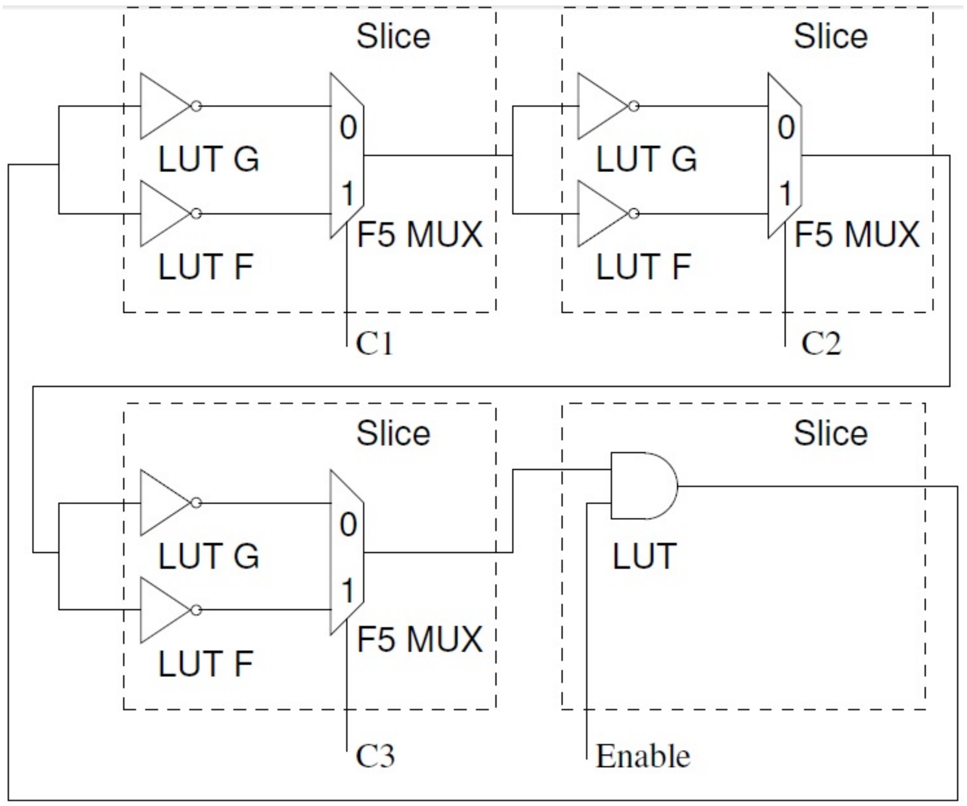


Figure 2.19. Configurable RO-PUF scheme [7].

A more complex, but hardware efficient solution to overcome the robustness problem of RO-PUFs, which is called Temperature-Aware Cooperative RO-PUF, was presented by Yin *et al.* in 2009 [10]. As discussed above, the faster RO within a pair may become the slower one because of the temperature or voltage variations. This will lead to generation of unreliable response bits; hence, resulting in the degradation of robustness. The proposed scheme depends on measuring the frequencies of RO couples just after manufacturing and defining rules for bit generation. These rules basically define the temperature ranges for each RO pair so that reliable bits are generated, and alternative pairs for the temperature ranges that may lead to erroneous output generation. For instance, for temperatures below t_1, RO_1 is faster than RO_2, for temperatures above t_2, RO_2 is faster than RO_1, for temperatures between t_1 and t_2, reliable bit generation is not possible with these two ROs and other ROs should be involved in the generation of this particular bit in that range. The main advantage of this method is its high robustness in unstable temperature conditions. The drawbacks are the need for temperature sensor, memory for storing the rules for bit generation, and long initialization time to generate the rules for each IC manufactured. In addition to these, the effect of voltage variation is not considered.

In [7,45,46], Maiti *et al.* proposed ways to improve the uniqueness and reliability of RO-PUFs. According to the cited works, frequencies of ROs at different regions of FPGA tend to differ more than the ones adjacent to each other. This behavior is a factor that decreases uniqueness, if the comparison is done between the ROs that are implemented at different regions. The proposed solution is to compare the ROs that are located next to each other. In addition to these, a mathematical model of regular RO-PUFs is built and a large scale characterization is done using 125 FPGAs. According to the analysis presented, RO-PUF shows an average inter-PUF HD of 47.31%, and an average intra-PUF HD of 0.86% under NOC.

Systematic variations of ICs are investigated and mathematical model of RO-PUFs is updated in [47]. In this work, it is shown that systematic variations are mainly caused by power supply variations and these can mask the manufacturing variability and decrease the uniqueness of RO-PUFs significantly. Under these circumstances inter-PUF HD of RO-PUF systems implemented on FPGAs may decrease down to 32% for certain cases, which is not acceptable for many PUF based applications.

Another technique to improve reliability is presented in [8]. In this work, the resources of the FPGA are used in a way to decrease the effects of spatial gradients like doping concentration. In order to achieve this, four neighboring ROs are arranged in a square, occupying the same LUT position in their individual configurable logic blocks, as shown in Figure 2.20. Then, the oscillation counts of the crosswise ROs are added and output bit is generated according to the comparison of additions, as shown in Figure 2.21. To improve the reliability performance of the proposed structure, a threshold value is determined and the measurement is repeated as long as the difference between the comparator input values is below the threshold value. According to the analysis presented, increasing the threshold value improves the robustness performance of RO-PUFs. However, since more measurements are required, timing performance of the structure is degraded.

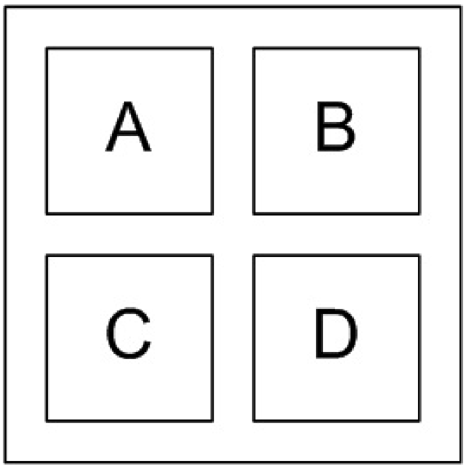


Figure 2.20. Location of ROs used for 1 bit generation.

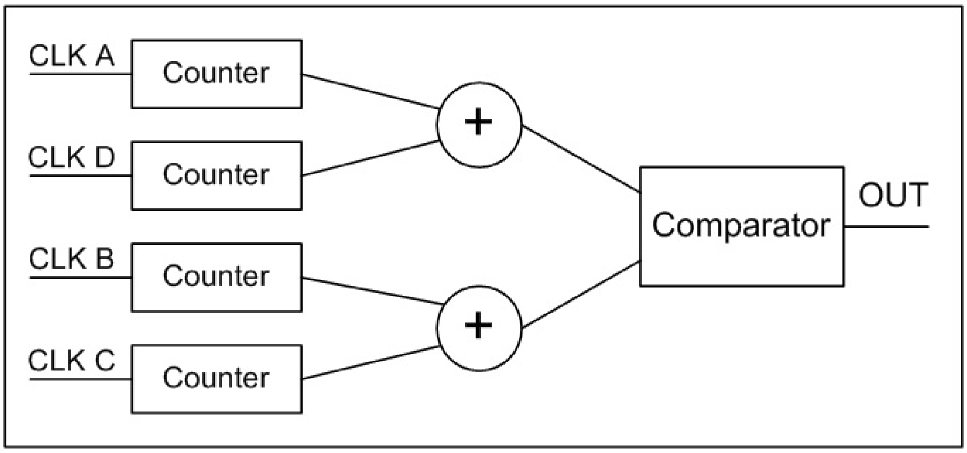


Figure 2.21. Output generation mechanism proposed in [8].

PUF types summarized in this section do not involve all structures proposed in the literature. Only the structures that have drawn significant attention are presented. Reconfigurable PUF [26], Controlled PUF [48], Public PUF [16], Time Division Multiplexed PUF [49], Sense Amplifier Based PUF [50, 51], Current-Based PUF [52], Bistable Ring PUF [53], Sub-Threshold PUF [54], Power Distribution System PUF [55], and Cellular Nonlinear Networks PUF [56] are the other PUF types proposed in the literature. In this work, we focus on RO-PUFs, since they are the most convenient type for FPGA implementation and work more reliably under changing environmental conditions [10, 45].

3. DERIVATION OF QUALITY METRICS

Even though quite a number of different PUF structures and their analysis results are presented in the literature, robustness and uniqueness properties are generally not evaluated in detail. This prevents comparing PUF circuits using a fair and standard performance criterion and choosing the best fitting structure for a specific application. In this section, new quality metrics for uniqueness and robustness are derived and the confidence interval concept is adapted to ensure the trustworthiness of the results. In addition to these, two basic RO-PUF structures are implemented and their performance evaluation is presented according to the new set of quality metrics proposed.

3.1. Derivation of Quality Metrics for Uniqueness

As explained briefly in Chapter 2, uniqueness is the inter-chip variation of PUFs. In the ideal case, PUF outputs generated on different ICs should be uniformly distributed and statistically independent. The most common metric for uniqueness, U_QM1, is the average HDs of outputs collected from different ICs. It has an ideal value of 0.5 and can be calculated as

$$U\_QM1 = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{HD(R_i, R_j)}{n}. \tag{3.1}$$

Even though the quality metric U_QM1 gives information about the performance of the system, it does not guarantee uniform distribution, since non-uniform outputs may have 50% HD. Due to this weakness of the U_QM1, two PUF structures with different uniqueness performances may be evaluated as the same if U_QM1 is used as the only performance parameter.

At this point, we propose defining another quality metric to evaluate the uniqueness of PUFs, depending on the fact that HDs of a uniformly distributed set of outputs will have a Gaussian distribution. This second quality metric, U_QM2, should check how close to Gaussian the distribution of HDs is. U_QM2 can be calculated via correlating the HD distribution of PUF outputs with the ideal Gaussian distribution. A better quality design will exhibit higher correlation with the ideal Gaussian distribution in this sense. U_QM2 can be defined as

$$U\_QM2 = Corr(DIS\_HD, G(Mn(HD\_PUF), \sigma)), \tag{3.2}$$

where $Corr$ is the correlation function, G is the function that generates data with the specified mean and standard deviation (STD) according to the Gaussian distribution, DIS_HD is the HD distribution of the collected PUF outputs, and $Mn(HD\_PUF)$ and σ are the mean and STD of HDs of the collected data, respectively. The closer the result is to 1, the closer the distribution is to Gaussian; hence, the circuit is better performing in terms of uniqueness.

In [18], Gilbert-Varshamov Bound (GVB) is utilized to determine the security of PUF outputs against exhaustive search attacks via calculating the minimum HD between two outputs within a set of outputs. In a similar manner, GVB can be used to determine the uniqueness of the structures as well. After collecting a certain number of outputs, the minimum distance, dHm, among them is determined in terms of bit count. Next, using dHm and PUF output length, N, R' is calculated as

$$R' \leq 1 - H2(\underline{d}_{Hm}) = 1 + \underline{d}_{Hm} \log_2(\underline{d}_{Hm}) + (1 - \underline{d}_{Hm}) log_2(1 - \underline{d}_{Hm}), \tag{3.3}$$

where $\underline{d}_{Hm}$ is calculated as

$$\underline{d}_{Hm} = \frac{d_{Hm}}{N}. \tag{3.4}$$

Then, using the number of measurements, M, and PUF output length, N, the ideal R is calculated via

$$R = \frac{log_2 M}{N}. \tag{3.5}$$

Proportion of R' to the ideal R can serve as a quality metric, U_QM3, for uniqueness and can be calculated as

$$U\_QM3 = \frac{R}{R'}. \tag{3.6}$$

If the outputs are highly unique, minimum HD will be compatible with the GVB and U_QM3 will converge to unity. Otherwise, the minimum HD will be worse than the bound states and U_QM3 will be smaller than 1.

The main advantage of U_QM3 is its capability of comparing dissimilar sets of PUF outputs. For instance, a set of 1,000 outputs with 100 bit length can be compared with another set of 500 outputs with 128 bit length. The set with the higher U_QM3 value performs better than the other in terms of uniqueness.

In addition to these, GVB and U_QM3 can be used to determine the number of circuits that can be identified with a previously determined security level and a certain length of PUF output. Similarly, the minimum required output length can be determined, for a previously set security level and the number of ICs to be identified. Here, the security level is $\underline{d}_{Hm}$, which is the ratio of the minimum HD between any two PUF outputs within a certain number of outputs to the output length and it is determined by the user.

For this purpose, U_QM3 is calculated, and a security level is set as $\underline{d}_{Hm}$. Then, R' is calculated again with the new $\underline{d}_{Hm}$ and the result is multiplied by U_QM3 to determine R. Finally, using Equation 3.5, the designer can either set the number of PUF bits, N, and calculate the maximum number of circuits that can be identified, M, or set the number of circuits and calculate the minimum length of PUF output.

3.2. Derivation of Quality Metrics for Robustness

Robustness is the intra-die variation that should be ideally zero for best performing PUF circuits, as discussed in Chapter 2. However, due to environmental variations and internal characteristics of PUF circuits, some bits of the output may flip within multiple measurements. The most common metric for robustness is R_QM1, which is the average HDs of measurements collected from a single IC and can be calculated as

$$R\_QM1 = \frac{1}{x}\sum_{y=1}^{x}\frac{HD(R_i, R'_{i,y})}{n}. \tag{3.7}$$

The ideal value of R_QM1 is 0, which represents noise-free data.

Some of the application areas that utilize PUF circuits require 100% robust outputs. For this purpose, ECCs can be used to generate noise-free data at every measurement, even under changing environmental conditions.

Since the complexity and cost of ECC depend on the maximum number of erroneous bits they can recover, R_QM1, which is based on an average, does not give enough information for the design of such systems. Therefore, we propose using the maximum error rate within a certain number of measurements as the second quality metric, R_QM2, which can be calculated as

$$R\_QM2 = \max \frac{HD(R_i, R'_{i,y})}{n} \quad (1 \leq y \leq number\ of\ measurements). \tag{3.8}$$

The ideal value of R_QM2 is 0, which indicates noise-free outputs. Higher values of R_QM2 indicates the need of more and more complex ECC for the generation of 100% robust outputs.

Another set of data presented in [35] is the distribution of errors on the output bits. This data is used to mask the most erroneous bits and improve the robustness performances of PUF structures. This approach may be helpful in practice, if it is convenient to detect the most problematic bits in each circuit and eliminate them for future use. Thus, error reduction rate with masking a certain number of bits may serve as another quality metric, R_QM3, for robustness in PUF circuits. This can be calculated as

$$R\_QM3 = \frac{R\_QM1 - R'\_QM1}{R\_QM1}, \tag{3.9}$$

where R'_QM1 represents the mean error rate after masking the most erroneous n bits. n can be selected depending on the robustness performance and cost requirements of the system. As n increases, the robustness performance of the structure increases. However, since more bits will be eliminated, area efficiency of the system will be degraded.

A common method to improve the robustness of PUF circuits is majority voting (MV) [4, 35]. Each bit of output is generated for a number of times and the result is determined via MV. This method increases the robustness performance of the structure especially under NOC. Thus, mean error rate after MV, R_QM4, can serve as an important quality metric as well. This is calculated as

$$R\_QM4 = \frac{1}{x}\sum_{y=1}^{x}\frac{HD(R_i, R'_{MVi,y})}{n}, \tag{3.10}$$

where the $R'_{MVi,y}$ represents the output after MV for n times.

Stable bit count, the bits that generate the same output at each measurement, is also an important parameter. If stable bits are selected and used, the need for an error correction mechanism is eliminated. Therefore, stable bit count can serve as an another quality metric, R_QM5, for robustness, which can be calculated as

$$\begin{gathered} R\_QM5 = \sum_{y=1}^{N} r_y \\ r_y = \begin{cases} 1, & R_{i,y} = R_{1,y}, \forall \text{i} \\ 0, & \text{otherwise} \end{cases}, \end{gathered} \tag{3.11}$$

where N is the number of bits in the output and $R_{i,y}$ is the yth bit of the ith response.

Finally, since PUF outputs are very vulnerable to changes in the environment, PUF structures should be evaluated according to the metrics presented under both NOC and varying temperature and/or supply voltage for a proper performance evaluation.

3.3. Confidence Interval and Confidence Level Concepts for PUF Evaluation

In the previous two sections, quality metrics that will be used to evaluate the uniqueness and robustness performance of PUF circuits are presented. However, the number of measurements to be taken for a reliable evaluation is still questionable. Therefore, in order to determine the trustworthiness of the results, confidence interval and confidence level concepts are adapted to PUF performance evaluation. Confidence interval is an estimation of values and indicates your measurements precision. Confidence level indicates the certainty of your estimation and expressed as a percentage. For instance, frequency of an RO can be expressed as 100 MHz with a confidence interval of 1 MHz and confidence level of 99%. This means that the frequency of the RO will be in the interval of 99-101 MHz with 99% certainty.

In this method, the number of measurements that has to be taken is determined according to the previously set confidence level and interval parameters, and using the STD of the outputs. This is calculated via the Chebyshev inequality as

$$Q(A) = 1 - 2\phi(\frac{c\sqrt{n}}{\sigma}), \tag{3.12}$$

where c is the confidence interval, n is the number of measurements, σ is the STD of the outputs, ϕ is the normal cumulative distribution function, and Q is the confidence level [57].

As the number of measurements increases, confidence level increases and/or confidence interval diminishes. For instance, 1,000 measurements can assure 99.9% confidence within 0.1% confidence interval, whereas 25 measurements can only provide 95% confidence level within 2% confidence interval when the STD of the HDs is 0.064, which is calculated based on the measurements taken from the FPGA environment.

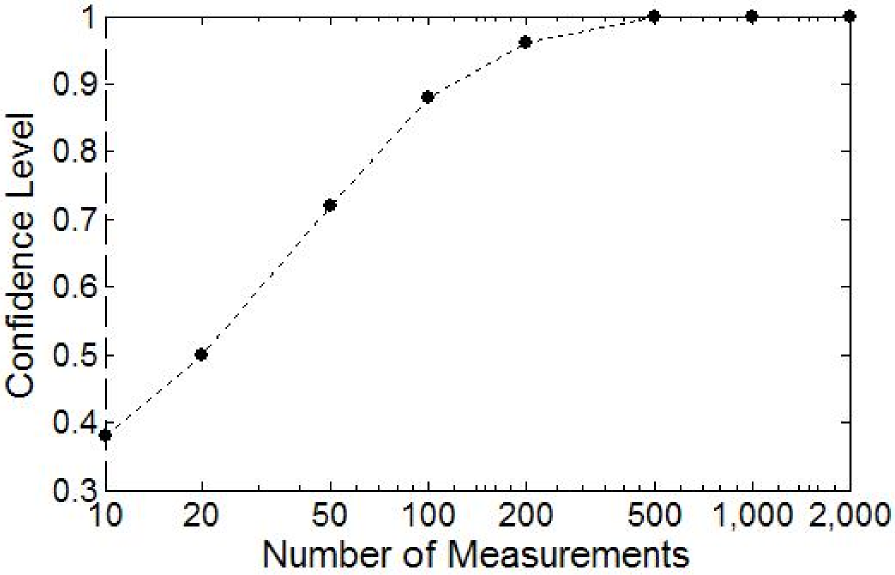


Figure 3.1. Number of measurements - confidence level relation with confidence interval of 0.01 and STD of 0.064.

The relationship between the number of measurements, confidence interval, and confidence level is presented in Figures 3.1 and 3.2. As can be seen from the figures, confidence level of the results increases as the number of measurements increases. In addition to this, higher confidence level is assured with fewer measurements in Figure 3.1, since confidence interval is ten times larger than the one in Figure 3.2.

3.4. Implementation of Two Basic RO-PUF Structures

Two RO-PUF structures presented in [48] are implemented on FPGA and their performances are evaluated according to the new set of quality metrics. Both designs employ a ring oscillator composed of five inverting stages and an enable input as illustrated in Figure 2.16. To maintain equal wire loads and minimize the systematic variation, the RO is built as a hard macro. One bit of PUF output is generated by comparing the oscillation frequencies of two ROs. Frequency comparison is done via using two counters, which count the number of oscillations of the two ROs within a certain time interval. In the first structure, $n+1$ ROs are implemented and each two ROs that are placed next to each other are compared to generate one bit output. This structure generates n bit outputs using $n+1$ ROs. In the second structure, $2n$ ROs

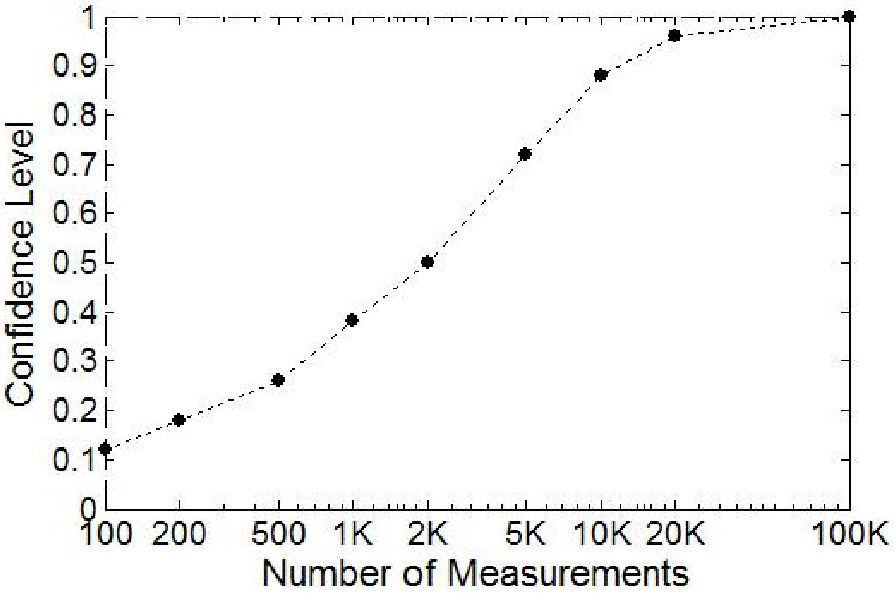


Figure 3.2. Number of measurements - confidence level relation with confidence interval of 0.001 and STD of 0.064.

are used to generate n bit outputs. In this structure, each RO is used only once and adjent ROs are used to generate the output bits, similar to the first structure.

3.5. Analysis of Experimental Data

In the system we have set up, two RO-PUF structures are implemented on a Xilinx 3S5000 FPGA and outputs are collected via Matlab. The number of outputs that will be collected is determined using the confidence interval approach. For uniqueness, 25 measurements are used to achieve 95% confidence within a 2% confidence interval. Uniqueness measurements are done by mapping the PUF structure to different parts of the FPGA, since we did not have enough number of ICs for PUF implementation. For robustness, 1,000 outputs are used providing 99.9% confidence within a 0.1% confidence interval. Robustness is measured under NOC and Varying Temperature (VT). For VT, 1,000 measurements are taken each at the operating temperatures 0, 20, 40, 60, 80, and 100 C^o.

Uniqueness and robustness results of the two PUF structures are presented in Table 3.1. For uniqueness, RO_PUF2 seems to be performing better than RO_PUF1 in terms of all quality metrics proposed. This result is meaningful since each RO is used only once in RO_PUF2, whereas each RO, except the first and the last one, is used twice in RO_PUF1, which decreases the entropy of the system; hence, the uniqueness.

For robustness, according to the R_QM1 under NOC, error rates of RO_PUF1 and RO_PUF2 are 0.8% and 1.3%, respectively. If the measurements are done under VT, error rates according to the R_QM1 are almost tripled for both of the implementations. R_QM2 states that the maximum error rate is 3.9% for both of the structures. According to R_QM3, masking the most erroneous three bits reduces the errors significantly for both of the structures. The effect of MV is presented as R_QM4. Robustness increases significantly if the MV is applied under NOC. However, its effect diminishes if the temperature varies in the system. Finally, according to R_QM5, 85-88% of the bits are stable for the two structures even under VT.

As can be seen from the results presented, both RO-PUF schemes behave as PUF in terms of robustness and uniqueness. However, their performances differ according to some of the quality metrics proposed. Depending on the performance requirements of the application in terms of uniqueness, robustness, and time and area consumption, best fitting architecture can be utilized in the system.

The number of circuits that can be identified with a certain security level using 128 bits PUF output is presented in Table 3.2 by using U_QM3. In addition to this, the required number of PUF bits is calculated with a certain security level and the number of ICs to be identified is also presented. Since RO_PUF2 has a better uniqueness, it enables identifying more ICs than RO_PUF1 with the same PUF output length and security level. For instance, at a security level of 0.2, RO_PUF1 identifies roughly 900,000 circuits, whereas RO_PUF2 identifies 5,800,000 circuits. Similarly, fewer PUF bits are enough to identify 10,000,000 chips by using RO_PUF2. However, the hardware cost of RO_PUF2 for the same output length is almost twice the cost of RO_PUF1. Under these circumstances, it can be concluded that the proposed quality metrics for uniqueness and robustness help the system designer to choose the optimum structure for the specific application.

Table 3.1. Uniqueness and Robustness results of RO_PUF1 and RO_PUF2.

Uniqueness Analysis	**Time per bit μs**	**# of Meas.**	**Conf. Int.**	**Conf. Level**
RO_PUF1	81.92	25	2	96
RO_PUF2	81.92	25	2	96.6
Metrics	**U_QM1**	**U_UM2**	**U_QM3**	
RO_PUF1	49.05	0.92	0.558	
RO_PUF2	49.55	0.94	0.631	
Robustness Analysis	**Time per bit μs**	**# of Meas.**	**Conf. Int.**	**Conf. Level**
RO_PUF1	81.92	1,000	0.1	99.9
RO_PUF2	81,92	1,000	0.1	99.9
Metrics	**R_QM1 under NOC**	**R_QM1 under VT**	**R_QM2**	**R_QM3**
RO_PUF1	0.89	2.63	3.9	1.4
RO_PUF2	1.31	3.65	3.9	2.4
Metrics	**R_QM4 under NOC**	**R_QM4 under VT**	**R_QM5 under NOC**	**R_QM5 under VT**
RO_PUF1	0.77	2.55	92.18	85.15
RO_PUF2	1.17	3.62	92.96	88.06

Table 3.2. Relation between the number of circuits to be identified, PUF output length and security level.

U_QM3	**PUF length**	**Security Level (min. HD)**	**Max. IC to identify**
RO_PUF1	128	0.2	912,938
RO_PUF2	128	0,2	5,809,570
RO_PUF1	128	0.3	357
RO_PUF2	128	0,3	776
U_QM3	**# of IC to identify**	**Security Level (min. HD)**	**Required PUF length**
RO_PUF1	10,000,000	0.2	149
RO_PUF2	10,000,000	0,2	132
RO_PUF1	10,000,000	0.3	351
RO_PUF2	10,000,000	0,3	310

4. OPTIMIZING RO-PUF CIRCUITS

Although RO-PUFs are better performing than the other PUF structures in terms of robustness, 100% error-free output generation is still very hard to achieve, even without speed and resource usage considerations [10]. There are two types of variations among RO structures that should be considered for a well-performing RO-PUF design. The first is the variation of the oscillation frequency within each RO at different time instances, which is called jitter. The second is the variation of oscillation frequency between identical ROs at different locations on the same IC and is called spatial variation. Due to the jitter phenomenon and environmental variations in ROs, some output bits generated by the circuit differ from measurement to measurement.

Jitter can be classified as short-term jitter and long-term jitter, as explained in the literature [58–61]. Short-term jitter is the instantaneous changes on oscillations that are observed from period to period. Long-term jitter is the jitter over a time period and is also called accumulated jitter. Since PUF operation requires measurements over a time period, accumulated jitter is subject to analysis in this work and the term jitter will refer to the accumulated jitter throughout the thesis. In order to generate outputs with minimum error using optimum resources and limited time, RO characterization in terms of accumulated jitter and spatial variation is required as the first step. In [60] and [62], spatial variation is presented based on measurements collected from FPGAs. The effects of the number of stages and supply voltage on both jitter and spatial variation are presented in [63] based on experimental data. Even though ROs are very well studied in the literature, the effects of the number of stages and measurement time on accumulated jitter and spatial variation have not been studied in the context of PUF performance.

In this chapter, we provide a design methodology for all types of RO-PUFs, which guarantee maximizing their performance in terms of robustness, area, and measurement time. For this purpose, the effect of the number of RO stages on frequency, accumulated jitter, and spatial variation is analyzed theoretically and verified experimentally. Next, the effect of the measurement time on accumulated jitter and spatial variation is analyzed and a design methodology for RO-PUFs is devised based on the foundations developed. Experimental validation is also presented using the results of previously built basic RO-PUF structures. The notation used in this chapter is presented in Table 4.1.

4.1. Effect of the Number of Stages

One of the most important design parameter for an RO structure utilized in an RO-PUF is the number of stages. Theoretically, any odd number of inverter stages connected serially will work as an RO and one bit PUF output can be generated using two of these ROs in conventional RO-PUFs. However, the number of stages has immense importance for the speed, jitter, spatial variation, and area of ROs. Therefore, the performance of RO-PUFs is closely related to this design parameter.

Frequency of an RO is directly determined by the delay of a single inverter and the number of inverters in the ring. In addition to these, the delay variation of inverters due to physical effects and random variation caused by noise affect the oscillation frequency of ROs [64]. The total noise in the RO manifests itself as the jitter on oscillations. In order to determine the delay characteristics of ROs in an RO-PUF, the following analysis is performed assuming a system of M ROs, RO_i, i=1,2..,M, that are composed of N inverters, $INV_{i,j}$, j=1,2..,N, each. Under these circumstances, the delay of the jth inverter in the ith RO at time k, $t_{d_{i,j,k}}$ can be calculated as

$$t_{d_{i,j,k}} = t_{inv} + \delta s_i + \delta s_{i,j} + \delta r_{i,j,k}, \tag{4.1}$$

where t_{inv} is the nominal delay of the inverter, δs_i is the inverter delay variation of the ith RO, $\delta s_{i,j}$ is gate delay variation of the jth inverter in the ith RO, and $\delta r_{i,j,k}$ is the random delay component. Here, δs_i, $\delta s_{i,j}$, and $\delta r_{i,j,k}$ can be assumed to be samples from Gaussian random variables (RV) with mean zero [65]. Therefore, $t_{d_{i,j,k}}$ is also a sample from a Gaussian RV with mean t_{inv}.

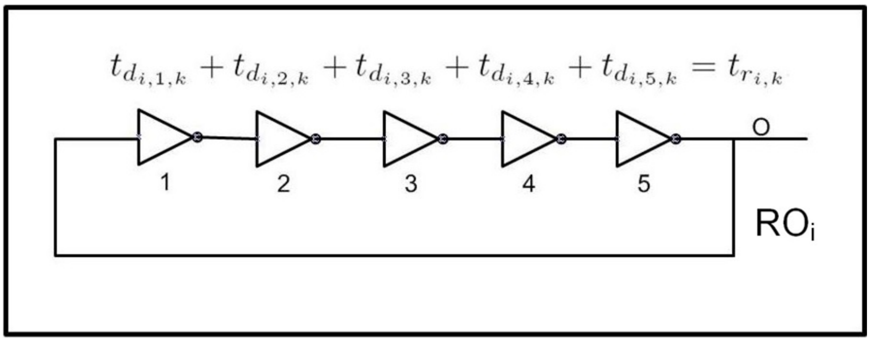

$t_{d_{i,1,k}} + t_{d_{i,2,k}} + t_{d_{i,3,k}} + t_{d_{i,4,k}} + t_{d_{i,5,k}} = t_{r_{i,k}}$
1
2
3
4
5
O
RO$_i$

Figure 4.1. 5-stage RO schematic with delay components mentioned.

After determining the delay of inverters, the delay of the ith RO, hence the period $t_{r_{i,k}}$, can be calculated by summing up the delays of inverters in the RO_i as

$$\begin{aligned} t_{r_{i,k}} &= \sum_{j=1}^{N}(t_{inv} + \delta s_i + \delta r_{i,j,k}) \\ &= N * t_{inv} + N * \delta s_i + \sum_{j=1}^{N}(\delta r_{i,j,k}). \end{aligned} \tag{4.2}$$

In Equation 4.2, the $\delta s_{i,j}$ component is discarded, since the inverters of an RO are located very near each other, hence the variation within the RO is small and this variation does not affect the total RO delay significantly. Similar to the inverter delay, $t_{r_{i,k}}$ is a sample from a Gaussian RV with mean $N * t_{inv} + N * \delta s_i$ and t_r is a sample from a Gaussian RV with mean $N * t_{inv}$. A five-stage RO with the delay components is presented in Figure 4.1.

Accumulated jitter is the main source of erroneous bits in RO-PUFs and is composed of correlated and uncorrelated noise, as explained in [61]. Correlated noise is strongly related to physical conditions such as layout and is directly proportional to the number of delay elements in the RO [66]. Random component in the inverter delay, $\delta r_{i,j,k}$, is the main source of the uncorrelated noise. As stated above, this random delay component can be safely assumed as Gaussian with mean zero and STD σ_r. When the whole RO is considered, N delay elements are connected serially and their delays add up. In this case, N random delay components, which are samples from Gaussian RVs are added. The resulting random delay component of the N-stage RO has again zero mean and N times the variance of a single inverter. Hence, the STD of RO delay, σ_{ro}, is proportional to $N^{1/2}$. When the effect of the correlated noise is considered, for large N, σ_{ro} may increase linearly with N as stated in [66]. However, employing a large number of stages is not very convenient for RO-PUFs due to speed, area, and power limitations.

Spatial variation is the main mechanism that enables PUF behavior in RO-PUF structures. It is basically the difference between identically laid out ROs at different locations of an integrated circuit and on different integrated circuits. In RO-PUFs, oscillation frequency differences of ROs are used to generate chip specific signatures. In order to determine the optimum number of stages in an RO, period variation of a set of different ROs is analyzed. In this case, analysis is done for M different ROs with N-stages. It is assumed that the measurements are repeated for a number of times and the results are averaged in order to vanish the random component. Thus, the delay of an inverter, $t_{d_{i,j}}$, can be stated as

$$t_{d_{i,j}} = t_{inv} + \delta s_i + \delta s_{i,j}. \tag{4.3}$$

The $\delta s_{i,j}$ component is discarded when summing the delays of inverters to determine the delay of an RO, since the variation within the RO is out of concern for spatial variation. Hence, the delay of the ith RO, t_{r_i}, can be defined as

$$t_{r_i} = N * t_{inv} + N * \delta s_i, \tag{4.4}$$

where δs_i is a sample from a Gaussian RV with mean zero and STD σ_i. Since multiplying a Gaussian RV with a constant,C, increases the STD with C and the variance with C^2, t_{r_i} is a sample from a Gaussian RV with mean $N * t_{inv}$ and STD $N * \sigma_i$.

Consequently, the STD of jitter increases by $N^{1/2}$, whereas spatial variation increases by N for an N-stage RO. When the frequency of oscillations is considered, jitter decreases by $N^{1/2}$ and spatial variation decreases by N, since the frequency and period are multiplicative inverses of each other.

Since the robustness is linearly proportional to spatial variation and inversely proportional to jitter, an RO-PUF consisting of ROs with small number of stages is the optimum structure. This is also the best case for speed and area of RO-PUF implementations, as validated through an FPGA implementation in Section 4.3.

4.2. Effect of the Measurement Time

As mentioned above, RO-PUF output generation depends on the oscillation counts of ROs within a certain time interval, t_m. Finding the optimum measurement time is a primary design objective, since it is closely related to speed, power consumption, and robustness of the system. In order to determine an optimum t_m, accumulated jitter and spatial variation are analyzed, since they are closely related to robustness. Accumulated jitter has two components, correlated and uncorrelated jitter, that depend on the measurement time [61]. Correlation coefficient of accumulated jitter in the system will be in the interval [0,1], depending on the magnitudes of the components. If the noise is totally uncorrelated, the correlation coefficient is equal to zero. For a fully correlated system, the correlation coefficient becomes one. Since the correlated and uncorrelated jitter depend on time, the correlation coefficient of the system changes dynamically. As explained in [61], the magnitude of the correlated component is proportional to t_m and dominates the uncorrelated jitter as t_m goes towards infinity. Correlated component becomes visible after a certain t_m, depending on the technology and layout. In spite of this, the uncorrelated component is dominant for small t_m and stabilizes within time since it is expected to be proportional to ${t_m}^{1/2}$. Therefore, when both components are considered, accumulated jitter should display less than linear increase for low t_m and a slope converging to one for large t_m.

To analyze the effect of t_m on spatial variation, delay differences of identically laid out ROs are considered. By using Equation 4.4, the delay difference of two N-stage

ROs for one period, $t_{\Delta r_{i,i+1}}$, is calculated as

$$t_{\Delta r_{i,i+1}} = N * \delta s_i - N * \delta s_{i+1}. \tag{4.5}$$

Since $t_{\Delta r_{i,i+1}}$ is stable for all periods, it is directly proportional to the number of periods after t_m. The number of periods after t_m can be calculated as t_m/t_{r_i}. As a result, the total delay difference between two ROs after t_m can be calculated as

$$t_{\Delta r_{i,i+1}}(t_m) = (t_m/t_{r_i}) * (N * \delta s_i - N * \delta s_{i+1}). \tag{4.6}$$

In order to find the optimum t_m, accumulated jitter and spatial variation should be measured for various values of t_m and the point, where the difference is the largest in favor to spatial variation should be chosen as the optimum t_m. At this optimum point, the effect of noise will be minimized; hence, the robustness of the system will be maximized. If there is more than one optimum point, the one with the lower t_m will be a better choice, due to speed and power considerations of the system. Selecting the optimum t_m has immense importance for the RO-PUF operation. Firstly, due to coupling problems, each output bit is generated one-by-one rather than activating all ROs and generating the entire output at the same time. This takes an important amount of time, $t_{PUF_K} = K * t_m$, that is proportional to the number of bits required and the time per bit generation. Even though one bit generation can be achieved within tens of microseconds, the total time required for a multi-bit key will be on the order of milliseconds, which slows down the system at each PUF output generation. Therefore, minimizing t_m without decreasing the robustness is crucial. Secondly, t_m is directly proportional to the energy used by the PUF circuit. The energy used by a single inverter for one oscillation can be defined as $C_L * {V_{DD}}^2$, where the C_L is the load capacitance of the inverter and V_{DD} is the supply voltage. The energy used by an RO during one period is N times the energy of a single inverter. Since the RO continues to oscillate during the interval t_m, the energy consumed is t_m/t_r times the energy of an RO for one period. For K bit PUF output generation $2 * K$ ROs are used and the total energy consumed is stated as

$$e_{PUF_{t_m}} = 2 * K * (t_m/t_r) * N * C_L * {V_{DD}}^2. \tag{4.7}$$

Therefore, optimizing t_m also optimizes the energy consumption and prevents unnecessary heating of the circuit. In addition to these, battery life of mobile systems will be extended.

4.3. Experimental Analysis and Validation of the Theory

ROs with 5, 7, 9, 11, 15, and 21 stages are built as hard macros on an FPGA to achieve identical layout, including the interconnects. 180 ROs are implemented on a Xilinx 3S5000 FPGA with frequency measurement circuitry based on counters and a serial port system. Since the oscillation frequencies 1 and 3-stage ROs are very high, reliable data cannot be data for them. In addition to this, due to area, power, and speed concerns of PUF implementations, measurements were not taken beyond 21-stage ROs. In this setup, each RO is activated one-by-one to minimize coupling and measured 50 times consecutively to calculate the jitter. Each measurement result is then sent to a PC via the RS-232 interface and analyzed in Matlab environment. Firstly, it is observed that the frequency of oscillations is inversely proportional to the number of stages, as shown in Figure 4.2. To determine the accumulated jitter, STD of 50 measurements from each RO is calculated and these values are averaged over 180 ROs, whose results are presented as the accumulated jitter in Figure 4.3. As can be seen from the figure, accumulated jitter of frequency decreases by $N^{1/2}$ as predicted via theoretical calculations. Spatial variation is measured by calculating the STD of 180 distinct RO frequencies. In order to minimize the random component, mean of 13 measurements from each RO is taken to represent the frequency of that RO. The results shown in Figure 4.4 again validate the theoretical calculations, since the spatial variation of frequency decreases inversely proportional to the number of stages.

Optimum measurement time is analyzed experimentally via implementing 180 5-stage ROs and collecting data for 16 t_m values ranging exponentially from 0.16

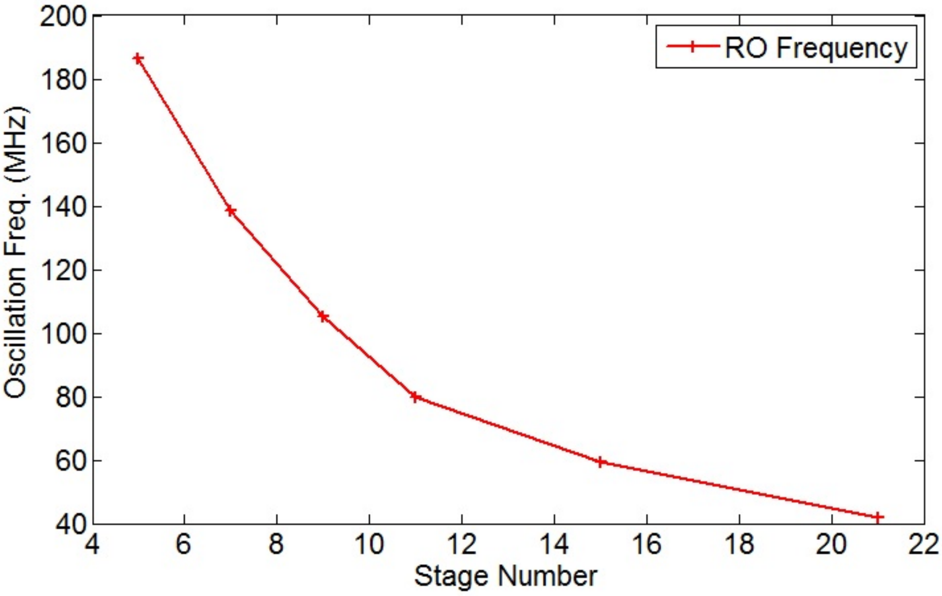


Figure 4.2. Frequency of ROs vs. Stage Number.

μs to 5.2 ms. Each RO is measured 50 times for each different t_m value. Based on the measurements, accumulated jitter and spatial variation are calculated. As shown in Figure 4.5, which is plotted logarithmically in both axes, spatial variation is directly proportional to t_m and accumulated jitter settles down after a certain time and starts to increase linearly as the correlated jitter dominates the system. In the time domain, until 10 μs, accumulated jitter is higher than the spatial variation, preventing PUF operation. Between 10 μs and 0.2 ms, accumulated jitter does not change significantly, whereas spatial variation continues to increase monotonically, which we call the critical region. After 0.2 ms, both accumulated jitter and spatial variation increase linearly; hence, their difference does not change significantly. Since the robustness of the system is closely related to the difference between accumulated jitter and spatial variation, t_m=0.2 ms seems to be the optimal point for this particular technology and RO structure.

The method of determining the optimum t_m is verified using the robustness results of previously built RO-PUF structures presented in Chapter 3 and [67]. As presented previously, two different RO-PUF structures are built and their robustness and uniqueness are analyzed experimentally based on a set of quality metrics. This analysis is repeated for four different t_m values, which are exactly the same four t_m

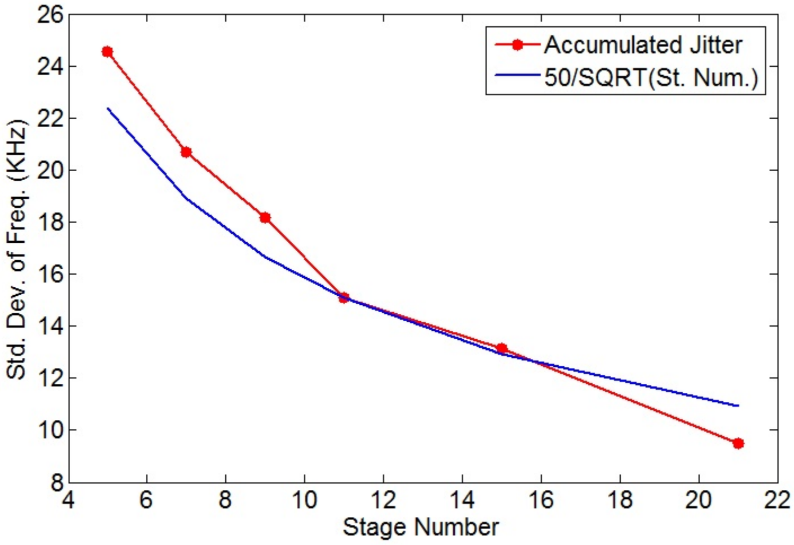


Figure 4.3. Accumulated Jitter vs. Stage Number.

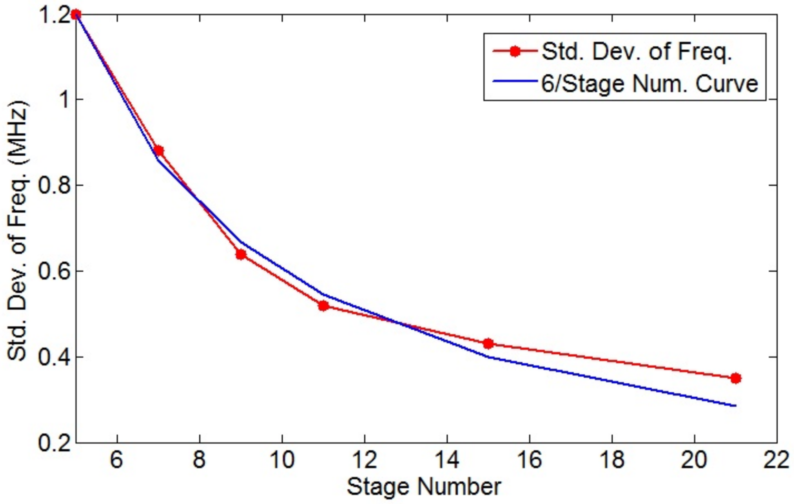


Figure 4.4. Spatial Variation vs. Stage Number.

values that are in the critical region. This helps to verify the theory for determining the optimum t_m value. As shown in Figure 4.6, error rate percentage decreases until 0.2 ms, where the error rate settles or starts to increase slightly for four different metrics. These four metrics are mean and maximum error rate under NOC, before and after the application of MV. This t_m value is also the optimum point for the best robustness performance as explained in the previous paragraph verifying the proposed technique.

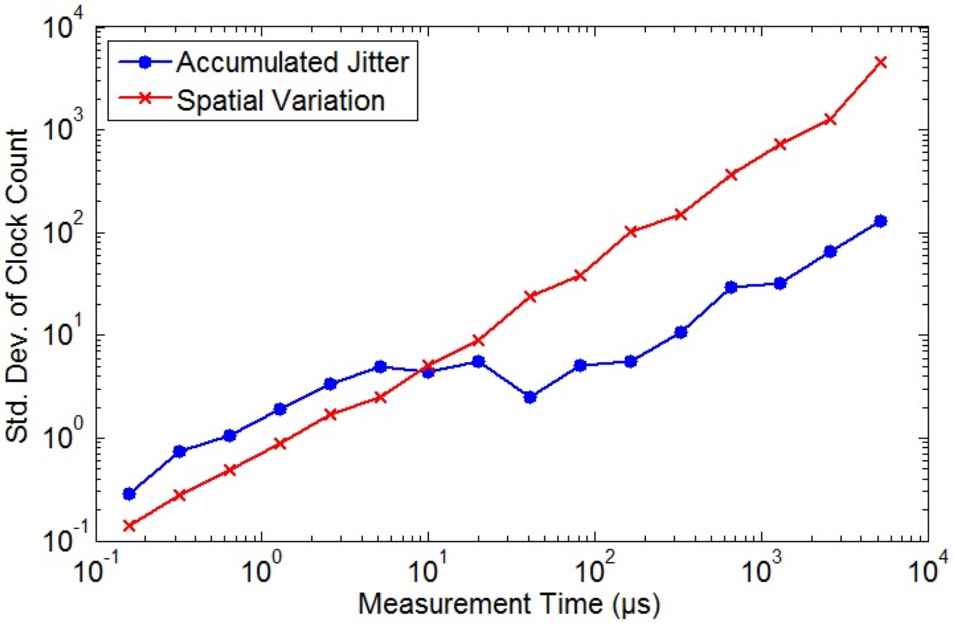


Figure 4.5. Accumulated Jitter and Spatial Variation.

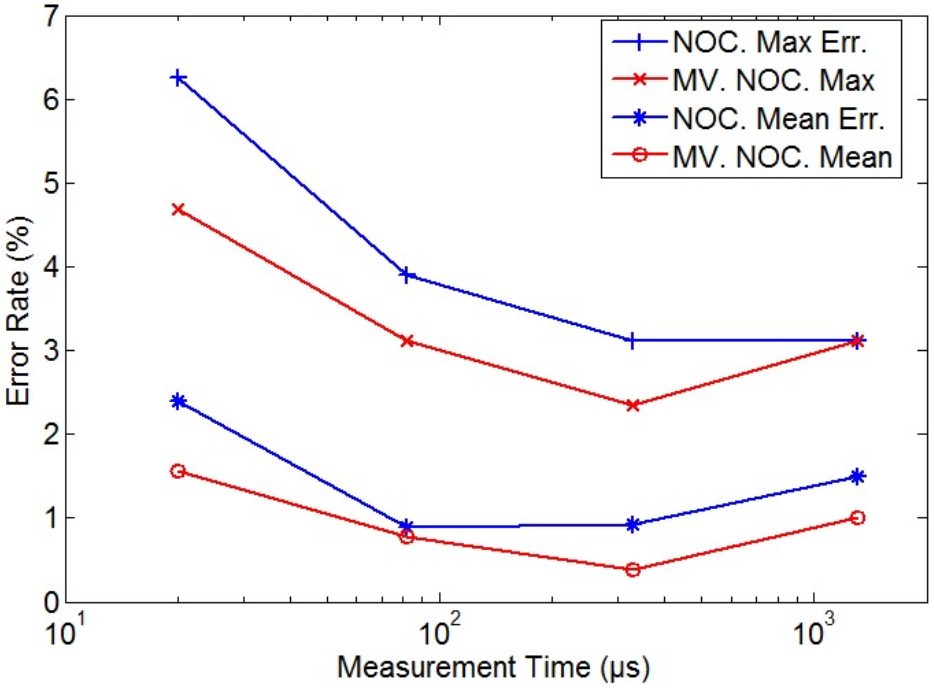


Figure 4.6. Error Rate vs. Meas. Time for RO-PUF structure.

Table 4.1. Notations and Meanings.

Notation	**Meaning of Notation**
t_m	Measurement time
N	Number of inverters in an RO
M	Number of ROs in the design or analysis
$t_{d_{i,j,k}}$	Delay of the jth inverter in the ith RO at time instance k
t_{inv}	Nominal delay of an inverter gate
δs_i	Variation of mean gate delays of inverters in the ith RO
$\delta s_{i,j}$	Gate delay variation of the jth inverter in the ith RO
$\delta r_{i,j,k}$	Random delay component of the jth inverter in the ith RO at time instance k
σ_r	Standard deviation of the random delay component
$t_{r_{i,k}}$	Delay of the ith RO at time instance k
σ_{ro}	Standard deviation of the RO delay
$t_{d_{i,j}}$	Mean delay of the jth inverter in the ith RO for a number of measurements
t_{r_i}	Mean delay of the ith RO for a number of measurements
$t_{\Delta r_{i,i+1}}$	Delay difference of the ith and *i+1*th RO for one period
$t_{\Delta r_{i,i+1}}(t_m)$	Accumulated delay difference of the ith and *i+1*th RO after t_m
K	PUF output length
t_{PUF_K}	K bit PUF output generation time
$e_{PUF_{t_m}}$	Energy consumed per PUF operation

5. ORDERING BASED RO-PUF CIRCUITS

5.1. Maximizing Robustness and Entropy in RO-PUF Circuits

As discussed previously, the vast majority of RO-PUFs generate one bit output by comparing the frequencies of two ROs [7, 8, 13]. This method does not exploit the maximum entropy present in the system; hence causes an increase in the required number of ROs for the generation of an output of certain length. To extract the maximum entropy from the system, frequency ordering of all ROs can be used, which can generate up to $\lfloor \log_2(N!) \rfloor$ bits by using N ROs [1]. Even though this theoretical upper-bound is not achievable due to the noise present in the system that causes unreliable bits, it is still much higher than the number of bits generated in conventional systems, which is upper-bounded by *N/2*.

The first step in output generation of ordering based RO-PUFs is grouping of the ROs. The frequencies of ROs in each group should be adequately separated from each other, which prevents changes in frequency ordering due to noise present in the system and temperature or supply voltage fluctuations. Then, frequency ordering of ROs within each group is used to generate the output bits. For a group of M ROs, *M!* different orderings that are equally likely may occur. By mapping each different ordering to a bitstream, $\lfloor \log_2(M!) \rfloor$ bits can be generated from each group [1]. Ideally, all ROs implemented on the circuit should be in the same group to extract the maximum entropy from the system. However, due to the noisy nature of integrated circuits and changing environmental conditions, this is almost impossible. Therefore, the main problem is to form the largest possible groups from the set of ROs implemented in the circuit.

In the literature, Longest Increasing Subsequence-Based Grouping Algorithm (LISA) is used to overcome the grouping problem [9]. In this method, frequencies of each implemented RO are measured at the lowest and highest temperatures that the circuit should work reliably, to determine the minimum and maximum possible frequencies of ROs. Then, the ROs are sorted according to the minimum frequencies measured. By using this sorted list, minimum and maximum frequencies of ROs, and a frequency threshold (f_{th}), groups are formed once at a time until all ROs are put into a group. For each group formation, $findReliableLIS$ function is called. This function creates a number of stacks and places the ROs to those stacks depending on the rules that are set to maintain reliability. After forming each group, its elements are removed from the sorted list to ensure each element is used only in one group and the function is repeated starting from the stack formation until all ROs are grouped. f_{th} used in the algorithm stands for the maximum frequency change of an RO due to noise present in the environment and measuring system. A sample RO-PUF system based on frequency ordering is depicted in Figure 5.1. Pseudo code of the LISA is presented in Figure 5.2.

In the ordering based approach, forming the groups is very closely related to the robustness problem. In LISA, noise in the system is compensated with the f_{th} value and the effect of environmental changes are compensated by measuring each RO in two extreme conditions and using both values in the algorithm. Even though this approach guarantees robustness, it is quite complex, since it requires two measurements for all ROs in all circuits which increases the computation cost of the algorithm. Also, the computation cost is high due to the redundant search of ROs performed in the algorithm. In order to solve the mentioned disadvantages of LISA, we propose exploiting DP to solve the grouping problem of ordering based RO-PUFs effectively and using a new parameter, so-called pre-determined frequency threshold (f_{thp}), which includes the frequency deviation that may result from temperature and supply voltage changes and noise present in the system. Therefore, the value of f_{thp} will be higher than f_{th}. With this approach, ROs are measured once under NOC and DP works with only one frequency value per RO.

The key point in this new approach is determining the value of f_{thp} parameter. If the value determined is less than the required amount, bigger groups will be formed, but their RO frequencies will not be far enough from each other. In this case, ordering may change under certain conditions preventing the PUF outputs from being 100%

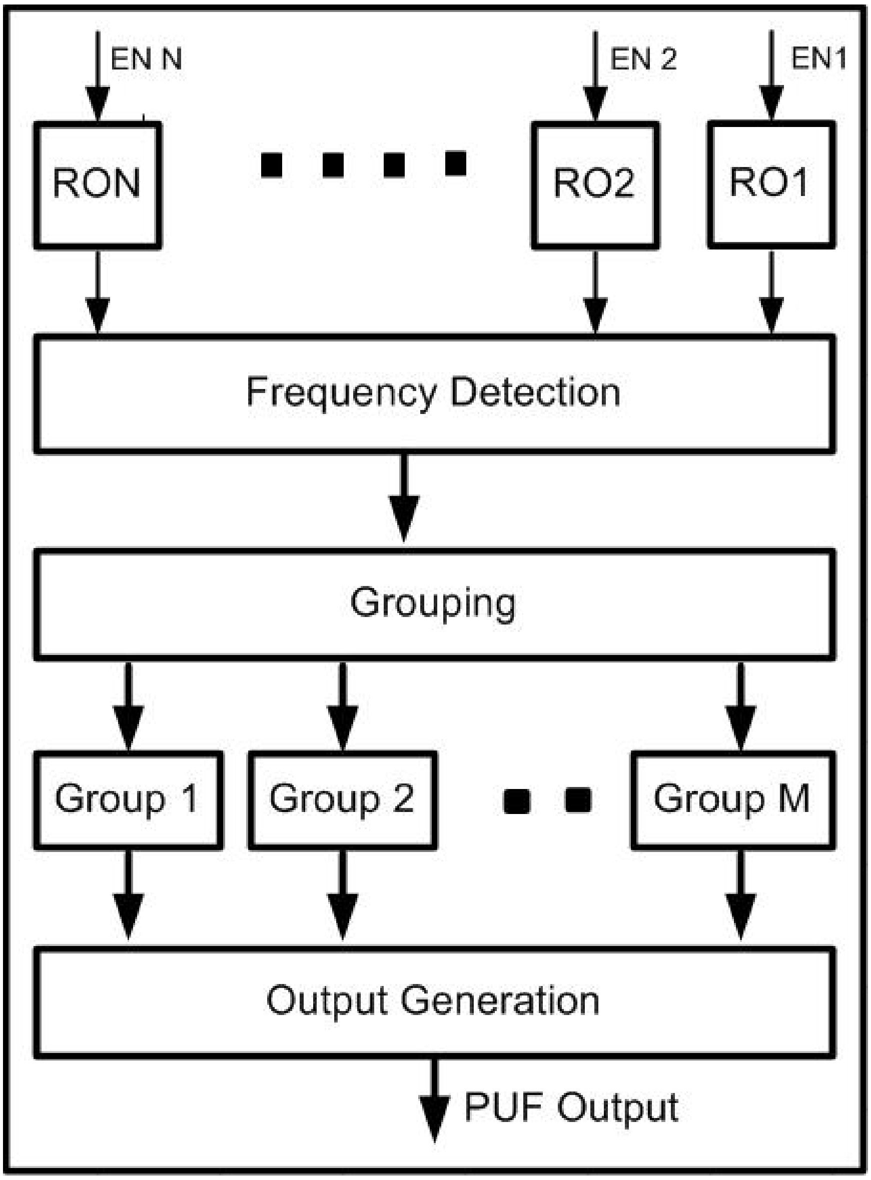


Figure 5.1. PUF output generation by ordering based systems.

reliable. On the other hand, if the value of f_{thp} parameter is higher than the required amount, smaller but more reliable groups will be formed and the extracted entropy, hence the number of output bits generated, will be lower. In order to determine the optimum f_{thp} value, a very small subset of circuits is formed and the frequencies of the ROs in this subset is measured at two extreme temperatures that the circuit is expected to work reliably. Normally, all ROs will be slower at higher temperatures, however their frequency change will be different from each other. This difference is the reason for unreliable bits and should be used as the minimum f_{thp} value in the PUF output generation algorithms. A formal structure of determining the f_{thp} value

```
Data:
1. RO linked list phy[n]. Each element of phy[n] contains the lowest frequency phy [i].$f_{min}$ and the highest frequency phy[i].$f_{max}$.
2. The linked list size $n$. For simplicity, $n \geq 2$ and $n$ is an even number.
3. The frequency threshold $f_{th}$ for reliability.
Result: 1. The partition of the ROs.
Sort phy[n] by $f_{min}$ in the increasing order and name the result sorted[n].
$i \leftarrow 0$
while there are ungrouped ROs do
    $S_i \leftarrow$ findReliableLIS(sorted[n])
    Mark ROs in $S_i$ grouped, remove from sorted[n] and update $n$, i++
end
findReliableLIS()
Create a stack ST, and push the first RO sorted[l] to it, $h \leftarrow 1$
for $j \leftarrow 1$ to $n$ do
    Top $\leftarrow$ the top RO on stack $ST_h$;
    if ((sorted[j]. $f_{min}$ - top. $f_{min} \geq f_{th}$) and (sorted[j] . $f_{max}$ - top. $f_{max} \geq f_{th}$) )
    then
        h++, new a stack $ST_h$, push sorted[j] to it.
        Sorted [j].PRE $\leftarrow$ top
    end
    else
        Find the stack $ST_p$ with the largest index $p$ that has its top elements $f_{max}$ ¡ sorted [j].$f_{max} - f_{th}$ and $f_{min}$ ¡ sorted [j].$f_{min} - f_{th}$.
        if $p! = null$ then
            Push sorted[j] to $ST_p$
            Sorted [j].PRE $\leftarrow$ top element of $ST_p$
        end
    end
end
Return sequence sorted [$j_1$], sorted [$j_2$] ... sorted[$j_h$]; where sorted[$j_h$] is the top element of $ST_h$ and sorted [$j_{q-1}$] sorted[$j_q$].PRE, $q$ from 2 to $h$
```

Figure 5.2. LISA in pseudo code [9].

```
**Data**:
1. A list of minimum RO frequencies $fmin[n]$.
2. A list of maximum RO frequencies, $fmax[n]$.
**Result**: $f_{thp}$
for $i \leftarrow 1$ to $n$ do
    $diff(i) = fmax(i) - fmin(i)$;
end
$f_{thp}$=$max(diff) - min(diff)$;
```

Figure 5.3. Determining f_{thp} in pseudo code.

is given in Figure 5.3 and explained in the next paragraph.

180 ROs are implemented on an FPGA board and their oscillation frequencies are measured at 20^{o}C and 100^{o}C. As can be seen from Figure 5.4, all ROs become slower when the IC temperature increases, whereas the frequency deviation changes within the RO set as shown in Figure 5.5. For instance RO_{63} slows down by 8.35 MHz, whereas RO_{115} slows down by 9.335 MHz when the temperature increases from 20^{o}C to 100^{o}C. Since these ROs are the ones with the highest and lowest frequency deviations, f_{thp} should be determined by considering the frequency variation difference of these ROs. In this case, adding a certain safety margin to the measured maximum frequency deviation to compensate the noise present in the system will result in an f_{thp} value of approximately 1 MHz to guarantee the robustness.

With the proposed method, a realistic value of the f_{thp} parameter is determined. However, a small amount of overhead should be added to this value for compensating the noise in the system and guaranteeing robustness in all manufactured circuits, since a subset of all circuits is used during the determination of the f_{thp} parameter.

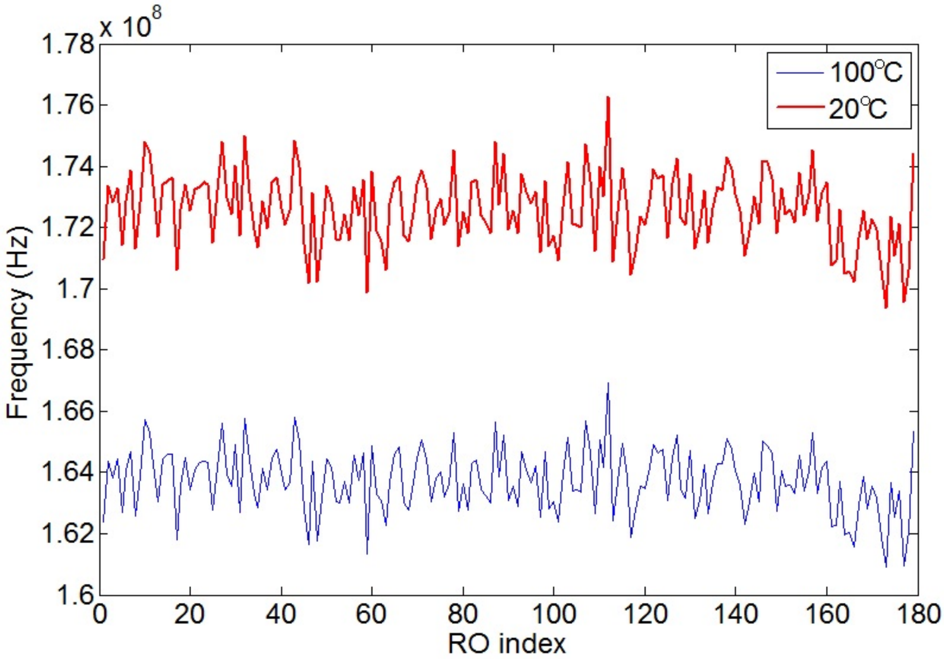


Figure 5.4. Frequency of ROs measured at 20^{o}C and 100^{o}C.

5.2. Adapting Dynamic Programming to RO-PUFs

Even though LISA extracts the maximum available entropy from the system with guaranteed robustness, it is very costly in terms of computational power, mainly due to redundant search for ROs to form the optimum groups. By using LISA, it may be hard to achieve low computation times for output generation on devices with limited capability. In addition to this, it requires two measurements for each circuit at two extreme temperatures, which complicates and increases the cost of initialization and output generation phases.

To overcome the drawbacks of LISA without decreasing its capability of extracting entropy and achieving high robustness levels, we have adapted the DP approach to the grouping of ROs in PUF output generation. With this approach, the computational complexity of output generation decreases considerably and the requirement to measure each circuit at two extreme temperatures during the initialization phase is avoided. Measuring a small subset of circuits at extreme conditions is enough to determine f_{thp} value, which will guarantee reliability with just one measurement taken under NOC for the rest of the circuits.

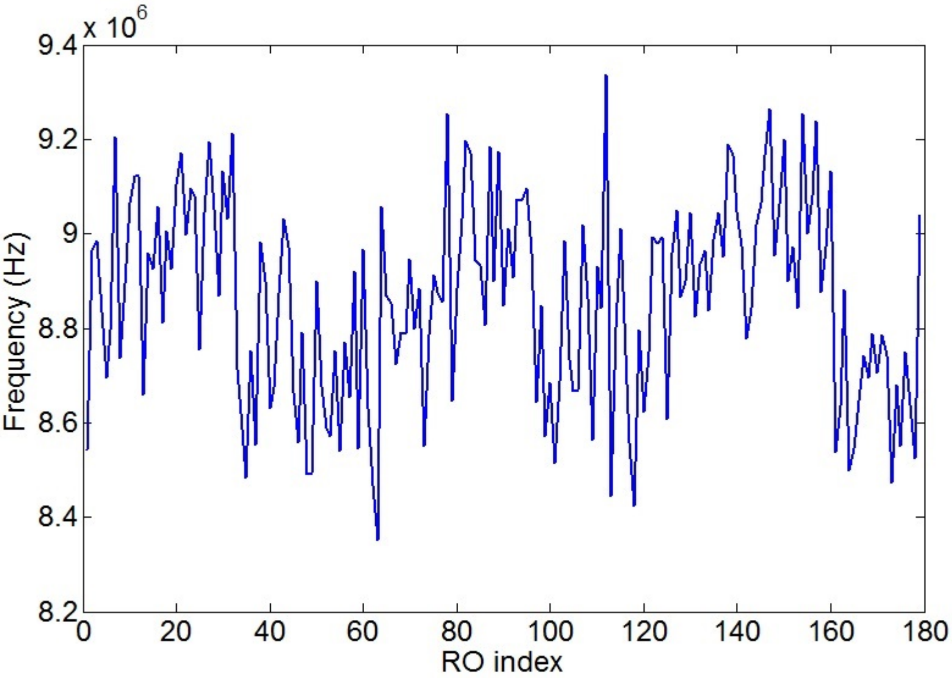


Figure 5.5. Frequency deviation of ROs.

The inputs to the DP algorithm for RO grouping are similar to the inputs of LISA. Each RO frequency is measured during the initialization phase and given as input to the algorithm. In addition to this, f_{thp} parameter is also used by the algorithm for the generation of reliable outputs. By using this f_{thp} parameter, it is possible to avoid measuring and working on two different frequencies for each RO, reducing the complexity of the initialization phase and PUF output generation.

DP algorithm has three steps. In the first step, ROs are sorted according to their frequencies in increasing order and $Fsorted[n]$ list is created. In the second step, for each RO, the nearest RO whose frequency is at least f_{thp} higher is found and a linked list is created, $list[n]$. In the third step, groups are formed using the linked list, that are satisfying the requirements of maximum entropy and 100% robustness even in unstable environmental conditions. In this step, the algorithm starts from the first position in the list, $list[1]$, and groups RO_1 with the RO that $list[1]$ shows, RO_j. Then, DP continues by jumping to the position of the last grouped RO, j, in $list$ and group the one that $list[j]$ shows. This continues until the last position is reached in $list$. After the first run, first group is formed and this step is repeated until all ROs are grouped. During the grouping process, if the RO that $list$ shows is grouped, the nearest ungrouped RO through the end of $list$ is added to the group. DP approach is explained in the next paragraph and its pseudo code is presented in Figure 5.7.

An RO set of 12 ROs are created as the first step and their frequencies are placed into an array, $FreqRO[n]$. In the second step, a sorted list of RO frequencies are created, $Fsorted[n]$. Then, a linked list, $list[n]$, is created using an f_{thp} value of 1.5 MHz, which shows the first available RO to be placed in the same group. As the last step, groups are formed one-by-one, and ROs placed in a group is removed from $list[n]$. Last step is repeated until all ROs are grouped. As can be seen from Figure 5.6, when the algorithm is applied, 3 distinct groups that will work reliably and extract maximum entropy from the available resources are formed. The first group with 6 ROs can generate $6! = 720$ possible orderings and $\lfloor \log_2(6!) \rfloor = 9$ bits. The second group with 4 ROs can generate 24 possible orderings resulting in 4 bits. The third group with 2 ROs can generate a single bit. As a result, 14 bits of output can be generated using 12 ROs in such a system.

The reduced complexity of the proposed DP approach is a result of avoiding the redundant RO frequency search done by the LISA algorithm. For each addition to a group, LISA searches over all RO frequencies that have at least f_{th} higher frequency than the last group member to form the largest possible group. Once a possible RO is identified, this RO is added to the group and the group size is incremented by one. In the DP approach, benefiting from the fact that we are operating on a sorted RO frequency list, the first qualifying candidate is chosen. This simplifies the task, since it allows choosing the nearest item to the last group member in the sorted RO frequency list. This search is illustrated in Figure 5.8, where the last group member is assumed to be i. The LISA algorithm searches over the region of the remaining sorted RO frequency list, where there is at least f_{th} frequency difference (the region that starts with i'' and extends to the end of the list), whereas the DP approach simply chooses i''. Choosing the first available candidate seems like a suboptimal solution; however, as we will prove next, this approach is indeed optimal in the sense that it always adds the same group member as LISA, thanks to the sorted nature of the input RO frequency list. We attempt to prove this via proof by contradiction.

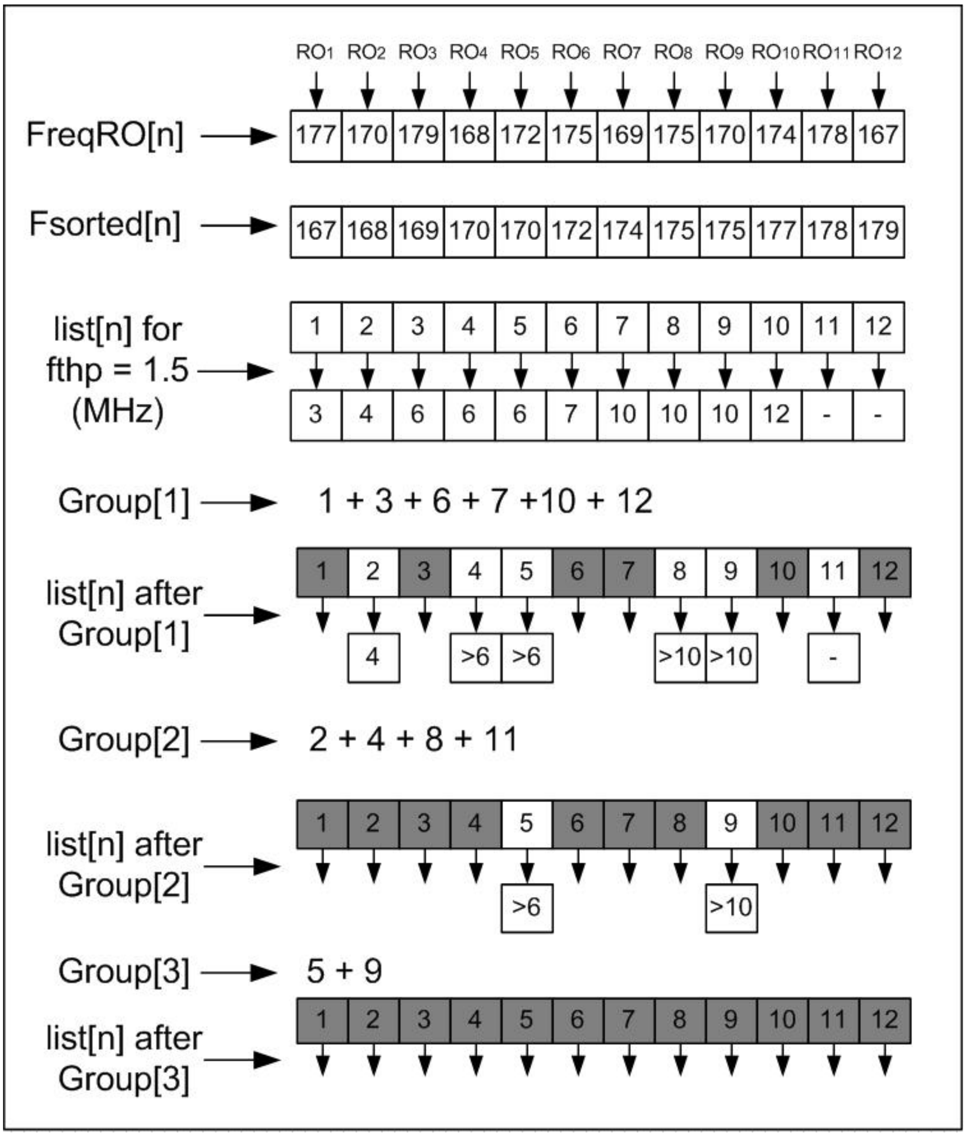


Figure 5.6. DP sample execution for 12 elements.

5.2.1. Proof of Lower Complexity of DP Method over LISA by Contradiction

Let $\{S_i\}$ denote the largest group that starts at position i (and ends at position n) and g_i denote the size of this group, $1 \leq i \leq n$. Also note that f_i denotes the frequency for the corresponding RO. The LISA algorithm searches over all possible future positions to obtain i' that belongs to the largest group, i.e., $i' = \arg\max(g_{i'})$ for all $i' > i$, such that $f_{i'} - f_i > f_{th}$. In this case, we can form the largest group for

```
Data:
1. A linked list of ROs with their frequencies measured under NOC, $FreqRO[n]$.
2. $f_{thp}$ for robustness
Result: Groups of ROs.
Sort $FreqRO[n]$ by frequency in increasing order: $Fsorted[n]$
for $i \leftarrow 1$ to $n - 1$ do
    find the nearest element $Fsorted[j]$ that is

    ($Fsorted[i]$ < $Fsorted[j]$-$f_{thp}$) and link $i$ to $j$ in $list[n]$
end
$i = 1$
while ungrouped RO exists do
    if ROi is ungrouped then
        Add $ROj$ to the group of $ROi$
        Jump to $ROj(i = j)$
    end
    if ROi is grouped then
        Increment $i$ until $ROi$ is ungrouped
    end
    if i=n and still ungrouped RO exists then
        $i = 1$
    end
end
```

Figure 5.7. Dynamic Programming in pseudo code.

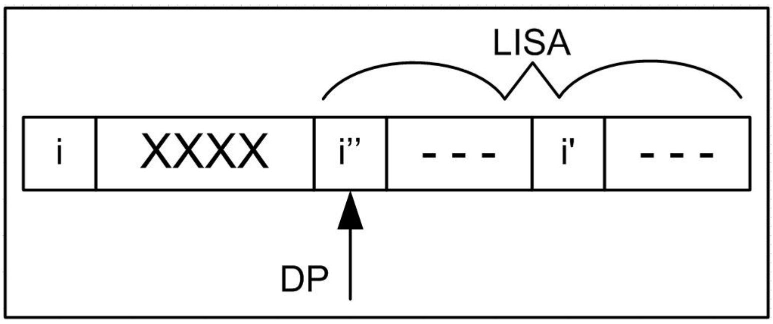


Figure 5.8. Search for the largest group.

position i, by simply adding it to this group as

$$\begin{aligned} g_i &= 1 + g_{i'}, & (5.1) \\ S_i &= S_{i'} \cup \{i\}. & (5.2) \end{aligned}$$

On the other hand, the DP approach simply looks for the smallest i'' such that the f_{thp} condition is satisfied using $i'' = \arg\min(i'')$ for all $i'' > i$, such that $f_{i''} - f_i > f_{thp}$. The corresponding group can be formed by

$$\begin{aligned} g_i &= 1 + g_{i''}, & (5.3) \\ S_i &= S_{i''} \cup \{i\}. & (5.4) \end{aligned}$$

Our claim is that, the newly formed group using the DP approach is at least as large as that formed by the LISA algorithm, i.e., $g_{i''} \geq g_{i'}$. Now, let's assume this is incorrect, i.e., assume that $g_{i''} < g_{i'}$. In this case, we can simply take $S_{i'}$ and replace i' with i'' using $S_{i''} = S_{i'} \setminus \{i'\} \cup \{i''\}$. This is indeed a valid set, since $f_{i''} \leq f_{i'}$ and any group that has the position i' as its lowest value would be still valid if this change is completed. This leads to the fact that we have a group that starts at position i'' and has the same number of elements as that of $S_{i'}$, i.e., $g_{i''} = g_{i'}$. This is a contradiction, since we had started with assuming $g_{i''} < g_{i'}$. Hence, the original claim of $g_{i''} \geq g_{i'}$ is valid and the DP approach can indeed form groups that are at least as large as the ones formed by the LISA algorithm.

Both LISA and DP share the step of sorting the ROs according to their frequencies prior to the grouping step. The sorting operation has a complexity of $O(N(\log(N)))$ [68] and may take significant time with increasing number of inputs. In the PUF case, this operation will be either done with a microprocessor that is available in the system or it will be done with additional circuitry, which will increase the area cost. This may not be convenient in devices with limited capability and decreases the usability of PUF on such devices. To overcome the drawback of sorting, a new method that does not utilize sorting for the PUF output generation mechanism is required. The Patience Sorting (PS) algorithm, which efficiently computes the lengths of the longest increasing subsequences in an unsorted data array, is a promising candidate for grouping the ROs [69]. However, this approach proved to be inefficient for PUF output generation, since both the IDs and frequencies of ROs in each group are sorted in increasing order by the PS algorithm and different orderings are not achievable. This renders the PS algorithm useless for entropy generation. Therefore, it can be concluded that removing the sorting step in ordering based PUF output generation mechanisms is not an option.

5.3. Experimental Analysis and Validation of the Theory

In the system we have set up, 160 ROs are placed on a Xilinx 3S5000 chip. Each RO is enabled one-by-one after power up and their oscillations within a certain amount of time is counted with a counter. The results are sent to PC via Matlab.

The frequencies of each RO are measured at six different temperatures, 0^oC, 20^oC, 40^oC, 60^oC, 80^oC, 100^oC to be able to calculate related f_{thp} values under different environmental conditions. It is assumed that the initialization of the PUF circuit is done at 20^oC and all f_{thp} values are calculated with reference to the frequencies measured at this temperature. Calculated f_{thp} values are given in Table 5.1. When the results are analyzed, it is seen that f_{thp} value increases as the temperature difference between two measurements increases. This is an expected result, since the frequencies of some ROs decrease slightly more than other ones with increasing temperature. As the temperature difference increases, frequency deviation increases as well.

Table 5.1. Maximum Frequency Deviation of ROs due to Temperature Change.

Initialization Temperature (^{o}C)	**Min./Max. Operation Temp. (^{o}C)**	**Max. Frequency Deviation (kHz)**
20	0	296
20	40	242
20	60	362
20	80	661
20	100	985

From this point on, we have analyzed the effectiveness of DP for different f_{thp} values in a wide range, since different designs may require different operating regimes; hence, different f_{thp} values. In the real case, it is the designers responsibility to determine the correct temperature and/or supply voltage for the reference and extreme measurement cases for an effective f_{thp} determination.

As discussed previously, in conventional systems, N ROs can generate $N/2$ bits without any dependency. In ordering based systems, the theoretical upper-bound is $\lfloor \log_2(N!) \rfloor$ bits using again N ROs. By using 160 ROs, an 80 bit output can be generated with conventional systems, whereas the upper-bound is 1086 bits in ordering based systems, which is more than 13 times higher. However, this is not achievable due to the noise present in the system and changing environmental conditions. When f_{thp} parameter is added to the system to compensate these effects, number of output bits generated decreases. For instance, by using an f_{thp} value of 1000 kHz, DP generates 127 bits of output, which is significantly higher than the 80 bits of conventional systems. Moreover, these conventional systems do not guarantee 100% robustness. Number of bits generated by DP with respect to conventional systems by using different f_{thp} values are shown in Figure 5.9.

Since more entropy is extracted from the system with ordering based output generation mechanisms, fewer ROs are enough for the generation of same length outputs. In Table 5.2, required number of ROs for 80 bit output generation with DP algorithm is presented for different f_{thp} values. As seen from the table, by using an f_{thp} value of

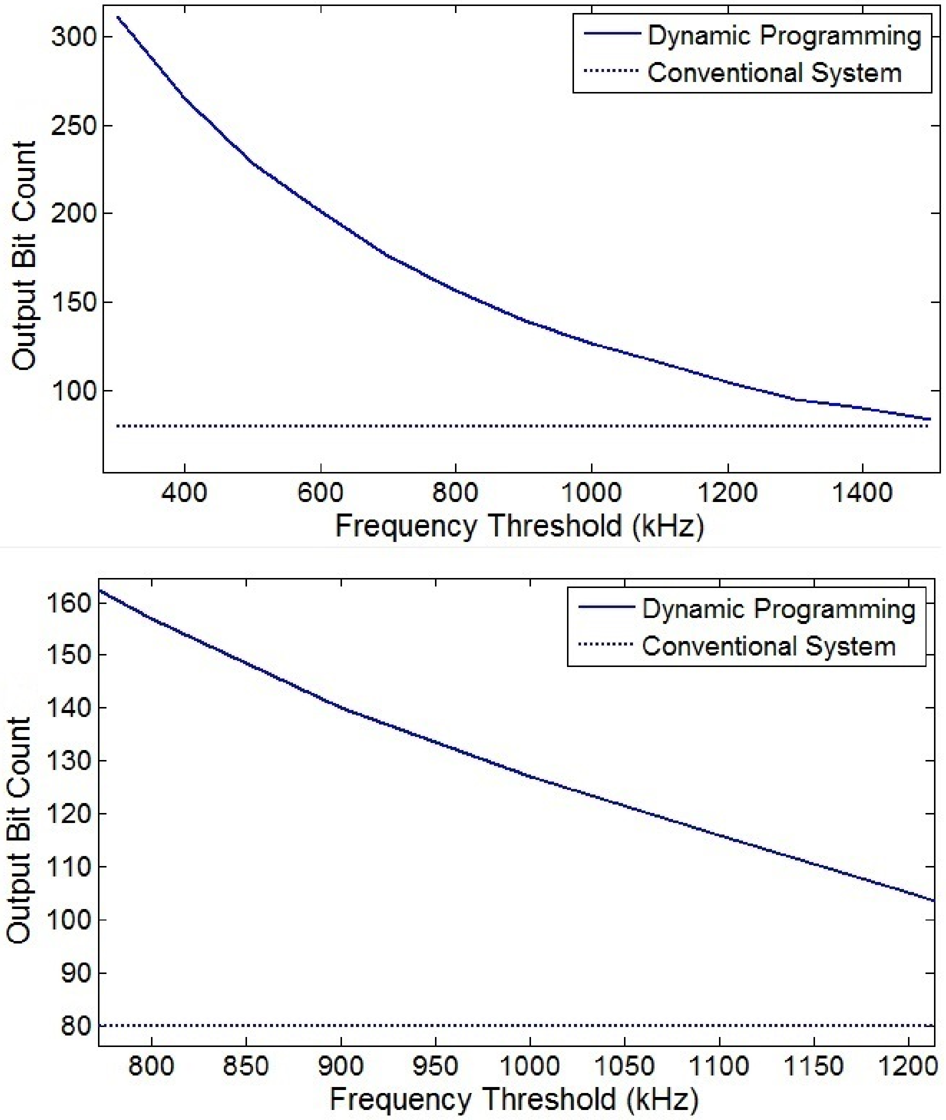


Figure 5.9. PUF output bit generation comparison.

600 kHz, more than 50% area reduction is achieved since fewer than half the number of ROs required for conventional systems is enough for an 80 bit output by using DP.

Another aim in this study was to decrease the computational cost of output generation in ordering based systems. To analyze the computational cost of the mentioned methods, both algorithms are implemented on Matlab and computation times are measured under the same conditions. As can be seen from Figure 5.10, DP algorithm has a significant advantage over LISA in this sense. For instance, for 160 RO implementation, execution time of LISA is more than 5.5 times longer than execution time of DP. Even for fewer ROs, such as 40 or 60, DP is 3 times faster than LISA,

Table 5.2. Area Reduction with DP.

f_{thp} (kHz)	Number of ROs for 80 bit output with DP	RO number decrease (%)
600	75	53
1000	105	34
1200	122	23

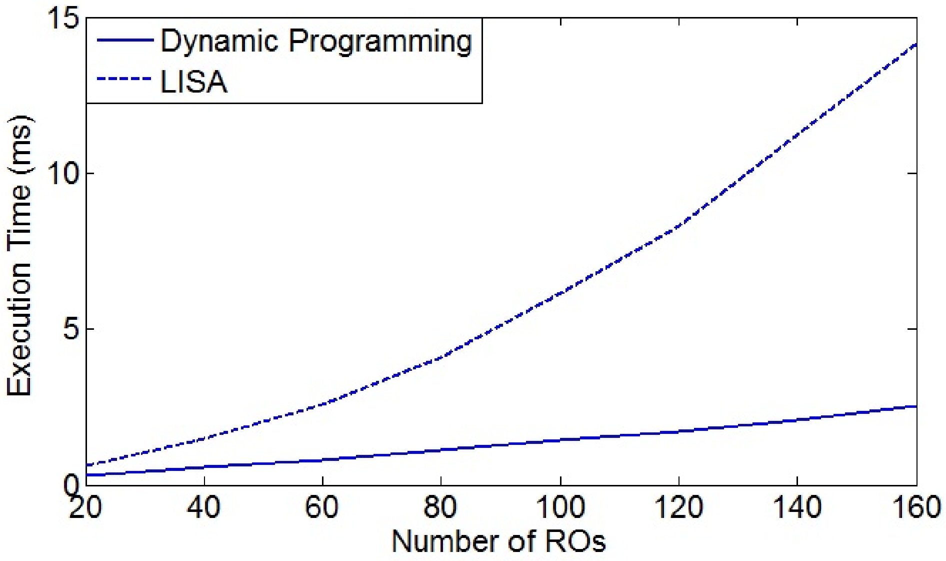


Figure 5.10. Execution time of algorithms in Matlab.

which will be very helpful for on chip implementation.

The performance of the proposed PUF structure is also measured in terms of uniqueness and robustness for various f_{thp} values. Due to the shortage of different FPGA ICs, PUF structure is mapped to 25 distinct parts on the FPGA and 128 bit outputs are generated from each implementation using four different f_{thp} values ranging from 1 MHz to 1.6 MHz. Uniqueness is measured with U_QM1, U_QM2, and U_QM3, which are proposed in Chapter 3 and the results showed that the proposed design behaves as a PUF. As can be seen from the results presented in Table 5.3, implemented PUF structure exhibit a very good uniqueness performance in terms of all quality metrics. We have also verified the robustness of design by generating outputs at six different temperatures, 0^{o}C, 20^{o}C, 40^{o}C, 60^{o}C, 80^{o}C, and 100^{o}C using four

Table 5.3. Uniqueness Results.

f_{thp} (MHz)	**U_QM1**	**U_QM2**	**U_QM3**
1	50	95,55	68,61
1,2	49,38	92,74	54,75
1,4	49,19	93,62	68,61
1,6	48,6	94,67	68,61

different f_{thp} values ranging from 1 MHz to 1.6 MHz. Each measurement taken under different temperature values generates the exact true PUF output that is generated during the initialization phase for all f_{thp} values, maintaining 100% robustness. As can be seen from the analysis results presented, ordering based RO-PUFs generate 100% reliable and highly unique outputs, and have better area efficiency than conventional RO-PUFs, which encourages their use in various applications.

Even though the number of required ROs for the generation of certain length outputs is significantly reduced with ordering based RO-PUFs compared to the conventional structures, comparison of the output generation mechanisms in terms of area and speed will be beneficial for a fair comparison. For this purpose, frequency detection, ordering, and output generation circuits are generated for different number of ROs and group lengths.

Frequency detection is the step that is the same for both conventional and ordering based RO-PUFs. In this step, oscillation counts of all ROs are detected within a certain measurement time, t_m. With the proposed design, a multiplexer and a counter are implemented. Each RO is selected one-by-one with the multiplexer and their frequency is detected with the counter. Six sample structures are implemented using combinatorial circuits for systems composed of 96, 128, 160, 192, 224, and 256 ROs. Area utilization results for Xilinx Virtex5 and Spartan3 FPGAs are presented in Table 5.4. Maximum achievable frequency for the proposed frequency detection circuit is 430 MHz for Virtex5 and 230 MHz for Spartan3 devices, which is higher than the oscillation frequency of 5-stage RO structures.

Table 5.4. Area Utilization of Frequency Detection Circuit.

RO Number	**Device Type**	**Slice Count**
96	Virtex5	31
96	Spartan3	40
128	Virtex5	44
128	Spartan3	48
160	Virtex5	44
160	Spartan3	57
192	Virtex5	57
192	Spartan3	65
224	Virtex5	62
224	Spartan3	73
256	Virtex5	68
256	Spartan3	81

Determining the ordering of ROs in a group and generating the output depending on this ordering is a mandatory step in ordering based RO-PUFs and critical for the performance and cost of the system. This step can be performed using a microprocessor already present in the system, or implementing a dedicated hardware. Assuming a microprocessor is not present in the system, dedicated hardware blocks are designed and implemented for ordering and output generation. Ordering of the oscillation counts is performed sequentially. RO IDs and their counts are stored in an array of registers in increasing order of the oscillation counts. Ordering of four ROs are illustrated in Figure 5.12. Execution time of ordering circuit is upper-bounded by $m^2/2$ for a group of m oscillators. However, since the ordering can overlap with the frequency detection of ROs, only the ordering time of the last group will decrease the speed of the operation. Output generation of ordering based RO-PUFs is performed by mapping each ordering to a different bitstream using a sequential circuit. Execution time of the ordering circuit is upper-bounded by m for a group of m oscillators. Similar to the ordering case, only the output generation time of the last group will decrease the speed of the operation. Pseudo code of the output generation is presented in Figure

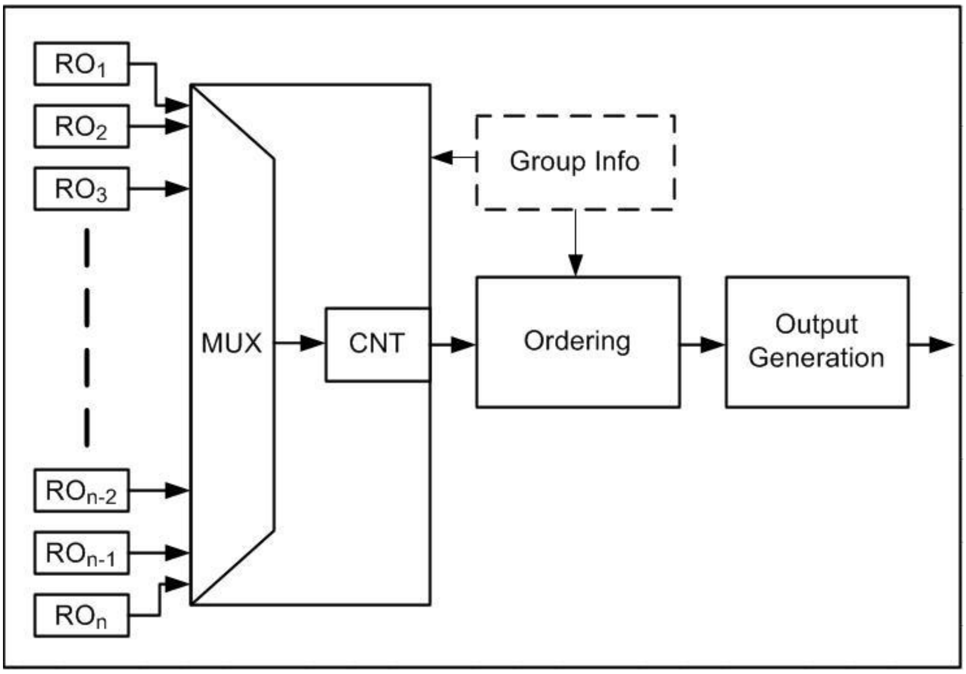
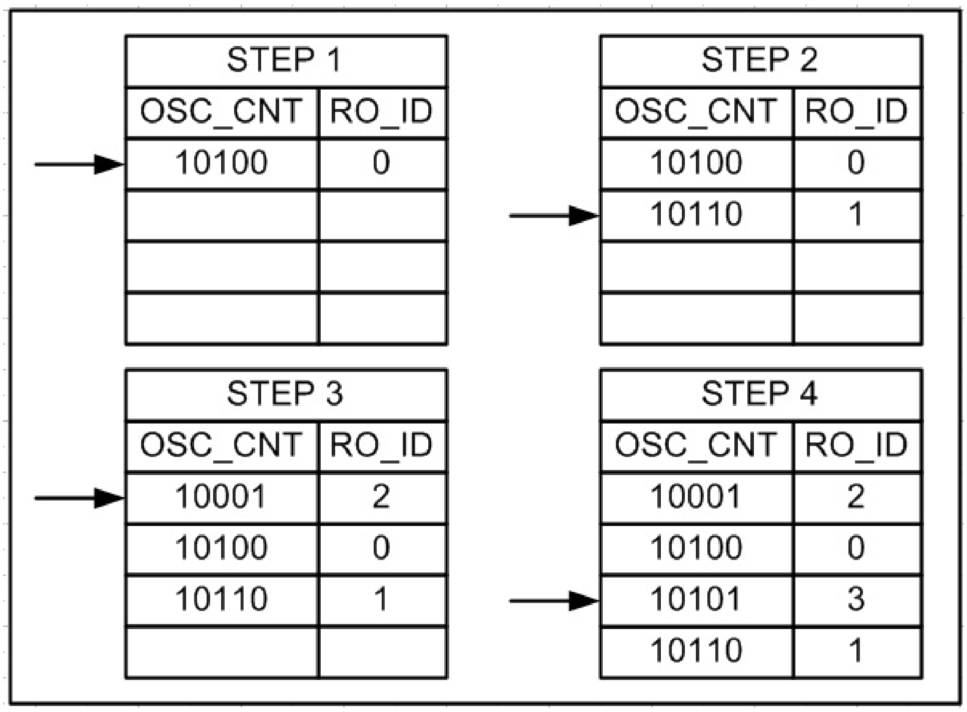


Figure 5.11. Output generation sample execution.

5.14 and output generation of a group of four ROs is illustrated in Figure 5.13. Output generation mechanism of the proposed ordering based RO-PUF is illustrated in Figure 5.11. According to this structure, it is assumed that grouping is done either by a PC during the initialization step and resulting groups are stored in a memory on-chip or off-chip, or done by a microprocessor present on the IC.

Since measuring the ROs one-by-one is a good design practice to prevent the inter-locking of ROs, implementing one ordering detection and output generation circuit according to the largest group present in the system is the most convenient way. In this method, an upper-bound for the group lengths will be set and the grouping step will form the groups according to this upper-bound. The proposed ordering and output generation circuits are implemented for different group lengths in the range of 3 to 10 and their area utilization results are presented for Xilinx Virtex5 and Spartan3 devices in Tables 5.5 and 5.6. Total number of slices for the generation of 128 bit outputs using conventional RO-PUFs and ordering based RO-PUFs with different maximum group lengths are presented in Tables 5.5 and 5.6 as well. According to the presented results, ordering based RO-PUFs with the maximum group lengths of 4 and 3 seems to be the optimum case for Virtex5 and Spartan3 devices, respectively, for the



STEP 1	
OSC_CNT	RO_ID
10100	0

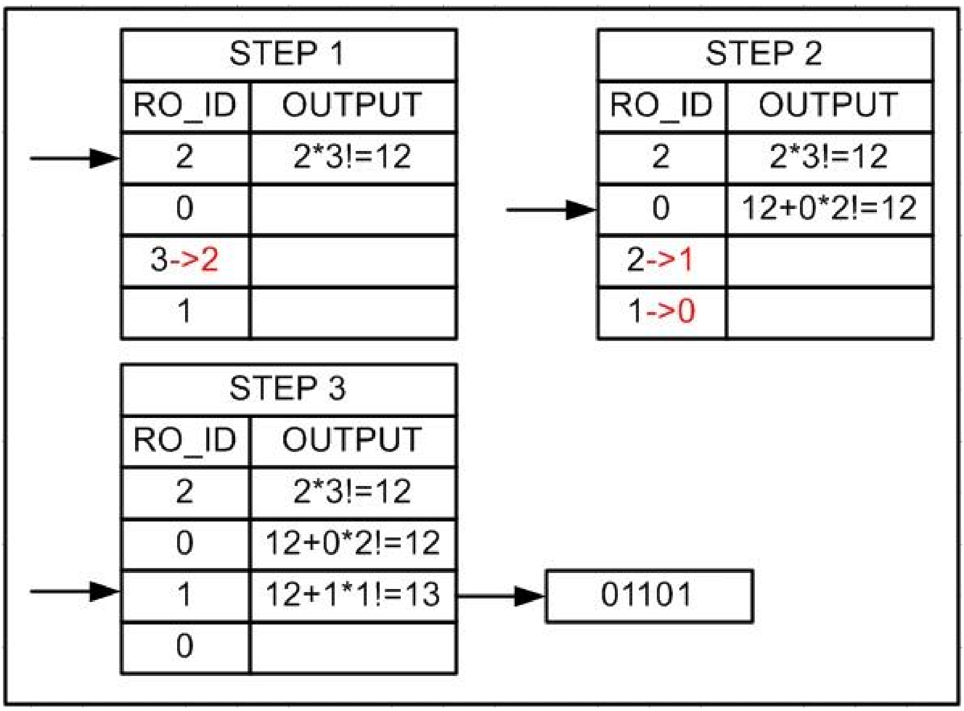
STEP 2	
OSC_CNT	RO_ID
10100	0
10110	1

STEP 3	
OSC_CNT	RO_ID
10001	2
10100	0
10110	1

STEP 4	
OSC_CNT	RO_ID
10001	2
10100	0
10101	3
10110	1

Figure 5.12. Ordering circuit sample execution.

area performance of the system. Increasing the group lengths more than the indicated values does not contribute to the overall performance due to the increasing cost of ordering and output generation circuits. It should be also noted that even though the area performance of the conventional circuit is compatible with the ordering based structures, conventional RO-PUF does not guarantee 100% robustness and require the implementation of ECCs for the systems that need 100% reliable outputs.



STEP 1	
RO_ID	OUTPUT
2	2*3!=12
0	
3->2	
1	

STEP 2	
RO_ID	OUTPUT
2	2*3!=12
0	12+0*2!=12
2->1	
1->0	

STEP 3	
RO_ID	OUTPUT
2	2*3!=12
0	12+0*2!=12
1	12+1*1!=13
0	

Figure 5.13. Output generation sample execution.

Data:

List of RO IDs in a group sorted according to their frequencies, $RO[m]$.

Result: Output bitstream.

```
for $i \leftarrow 1$ to $m - 1$ do
    Output = Output + RO[i]*(m-i)!
    for $j \leftarrow i$ to $m - 1$ do
        if $RO[i] < RO[j]$ then
            Decrement RO[i]
        end
    end
end
```

Figure 5.14. Output generation in pseudo code.

Table 5.5. Area Utilization of RO-PUFs for Virtex5 Devices.

PUF Type	**RO Num**	**RO Area (Slice)**	**Freq. Det. (Slice)**	**Ord. (Slice)**	**Out. Gen. (Slice)**	**Total (Slice)**
Conventional	256	512	68	0	5	585
Ord. B. (3)	195	390	57	10	7	464
Ord. B. (4)	185	370	57	26	10	463
Ord. B. (5)	175	350	57	49	15	471
Ord. B. (6)	170	340	57	54	23	474
Ord. B. (7)	165	330	57	71	45	503
Ord. B. (8)	160	320	44	114	56	534
Ord. B. (9)	155	310	44	117	98	569
Ord. B. (10)	150	300	44	181	123	648

Table 5.6. Area Utilization of RO-PUFs for Spartan3 Devices.

PUF Type	**RO Num**	**RO Area (Slice)**	**Freq. Det. (Slice)**	**Ord. (Slice)**	**Out. Gen. (Slice)**	**Total (Slice)**
Conventional	256	512	81	0	9	602
Ord. B. (3)	195	390	65	28	10	493
Ord. B. (4)	185	370	65	57	16	508
Ord. B. (5)	175	350	65	97	36	548
Ord. B. (6)	170	340	65	128	57	590
Ord. B. (7)	165	330	65	163	82	640
Ord. B. (8)	160	320	48	213	98	679
Ord. B. (9)	155	310	48	260	175	793
Ord. B. (10)	150	300	48	336	210	894

6. ERROR PROBABILITY IN ORDERING BASED RO-PUFs AND ANALYSIS OF ERROR CORRECTION CODES

Determination of the f_{thp} value is the most critical step in ordering based systems. An ideal f_{thp} value should guarantee the robustness of PUF outputs by grouping the ROs, whose frequencies are far enough from each other while maintaining the largest group sizes to extract the maximum entropy present in the system. Determining the f_{thp} value smaller than the ideal value increases the group sizes and lowers the robustness of the system. If the distance between RO frequencies in the same group are smaller than the amount of noise and fluctuations in the environmental conditions, such as process variations, temperature, and supply voltage, robustness of the system may be compromised. This situation arises due to so-called problematic RO pairs, which have frequency deviation more than the selected f_{thp} value and have the potential of generating errors in the output. In this chapter, error probability of ordering based RO-PUFs is subject to analysis based on f_{thp} value determination. In addition to this, ECC, which have the capability of removing noise present in PUF outputs, are presented and their implementation results are analyzed.

6.1. Systematic Analysis of Bit Error Probability

Even though the f_{thp} value determination presented in Chapter 5 gives an idea about the error vulnerability of the system, it does not fully express the bit error probability that will result in the output. Since erroneous outputs are acceptable up to a certain degree in some applications that utilize PUF circuits, a bit error probability analysis is required. The analysis of bit errors due to problematic RO pairs being placed in the same group is complicated, since it is closely related to group lengths, symbol error probability, and symbol error to bit error conversion. Here, symbol error is the change in ordering within a group due to noise present in the system or environmental variations. In the following subsections, these issues are addressed.

6.1.1. Group Length Analysis

In ordering based RO-PUFs, ROs are grouped according to the selected f_{thp} value and each group generates a certain number of output bits according to the conversion method selected. If the ordering in a group changes due to one of the problematic RO pairs, a symbol error occurs and some of the output bits become erroneous. Since the length of the group that generates the symbol error affects the number of erroneous bits, an analysis on group lengths is required for a complete bit error probability estimation. One way of determining the group lengths is to measure a large number of samples and calculate the average number of groups per size. However, since it is impractical to build and measure many samples, a large synthetic data set can be created using the mean and STD of RO frequencies measured from a sample system. Then, DP algorithm is applied to the data set and group lengths are analyzed according to the selected f_{thp} values. This is explained in the next paragraph.

25 sample RO-PUFs of 160 ROs each are implemented, and mean and STD of RO frequencies are calculated. Then, 10,000 sets of 160 RO frequencies are generated by Matlab based on the calculated mean and STD. The reason for using 160 ROs for each RO-PUF is that they can generate approximately 128 bits of 100% robust outputs by using ordering based methods, which is adequate for many applications, such as AES encryption. Then, DP algorithm is used to determine the lengths of groups formed with respect to the selected f_{thp} value. A range of 0.5 MHz to 1.1 MHz is used for the f_{thp} value, since 1 MHz seems to be the optimum value for this case. A smaller value of f_{thp} allows ROs with frequencies closer to each other to be placed in the same group; hence, larger groups are more likely to be formed. On the other hand, larger f_{thp} values limit the sizes of groups, allowing only those ROs with frequencies far enough from each other to be in the same group. As can be seen from Figure 6.1, among 10,000 sets, the largest group formed with an f_{thp} value of 1.1 MHz involves 11 ROs (This group size appears in 11 sets among 10,000), whereas the largest group formed with an f_{thp} value of 0.5 MHz involves up to 19 ROs (This group size appears in 4 sets among 10,000). An interesting result observed from this analysis is the number of groups with a single RO, which do not contribute to the output generation by default.

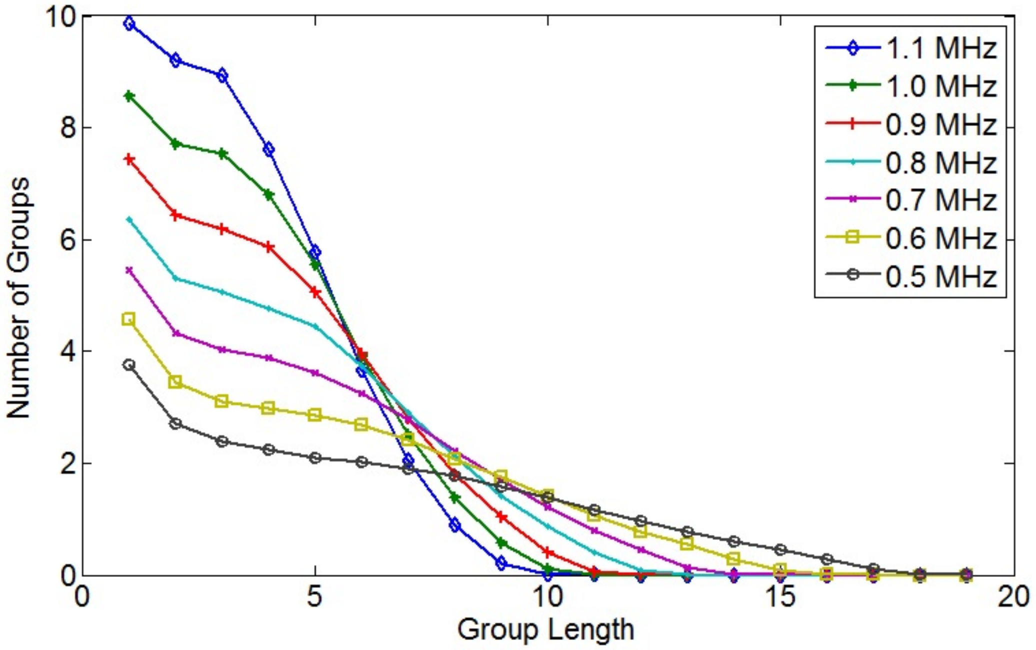


Figure 6.1. Number of groups per group length for different f_{thp} values.

With an f_{thp} value of 1.1 MHz, approximately 10 ROs could not be grouped with any other ROs on the average. This number becomes smaller than 4, when the f_{thp} value is 0.5 MHz.

6.1.2. Symbol Error Probability and Validation

As discussed previously, if the selected f_{thp} value is smaller than the ideal f_{thp} value, problematic RO pairs arise, which have the possibility to create symbol errors. The required conditions for a problematic RO pair to create a symbol error are stated as follows:

(i) Both ROs of a problematic RO pair should be in the same group and next to each other in frequency ordering.[1]

(ii) ROs of a problematic RO pair should be in the correct order.[2]

[1] Assuming that the selected f_{thp} value is greater than half of the ideal f_{thp} value.

[2] If the frequency of RO_1 is smaller than the frequency of RO_2, but RO_1 becomes faster than RO_2, a symbol error occurs. Then, this situation is called the correct order for symbol error creation. On the other hand, if RO_1 slows down more than RO_2, the frequency order does not change even if the frequency deviation is smaller than the f_{thp} value, and a symbol error is not created.

(iii) Environmental changes and noise should be large enough to create the symbol error.

ROs in the same group affect the output bit generation. If the problematic ROs are distributed to different groups or placed into the same group, but are not next to each other in the frequency ordering, ordering in any group does not change and a symbol error is not created. The symbol error probability is closely related to the number of groups formed and their lengths. Fewer groups with higher number of ROs tend to have a higher probability of resulting symbol error from problematic RO pairs. In the extreme case, if the RO-PUF has a single group with all the ROs included, a symbol error will definitely occur from a problematic RO pair, when the ROs of the problematic pair are next to each other and in the correct order in frequency, and when certain environmental conditions occur. The number of different placement combinations for two ROs in a list of M ROs is

$$C_M = (M * (M - 1)). \tag{6.1}$$

Then, the probability of the two elements of the RO pair to be placed next to each other in a group of length s, when the total number of ROs in the PUF is M, can be stated as

$$p_s = 2 * (s - 1)/(M * (M - 1)). \tag{6.2}$$

Based on this formula, the probability of an RO pair to be placed in a group of size two, in a system of 160 ROs, can be calculated as 7.86×10^{-5}. This probability increases up to 1.1×10^{-3} if the group size is 15, as shown in Figure 6.2. Since the total number of groups and their distribution in terms of lengths depends on the f_{thp} value, this value determines the probability of placing the problematic RO pair in the same group. Let k_s be the average number of groups with size s and m be the size of the largest group. Then, the probability of placing the RO pair in a group of size s

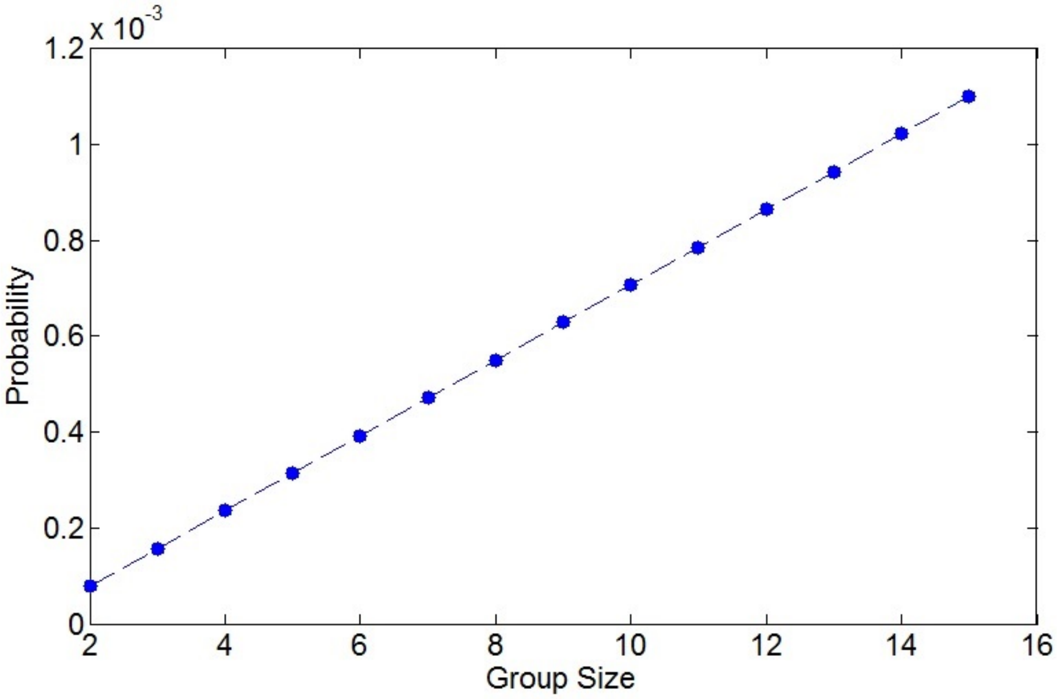


Figure 6.2. Probability of two ROs to be placed in the same group.

can be calculated as

$$k_s * p_s, \tag{6.3}$$

and the probability of placing the RO pair in a group can be calculated as

$$p_{av} = \sum_{s=2}^{m} k_s * p_s. \tag{6.4}$$

Even though the problematic RO pair is placed in a group, this does not mean that a symbol error will occur when the required environmental conditions are realized. The RO pair should be in the correct order for a symbol error creation. As explained in Chapter 5, frequency deviation of ROs differ with changing temperature or supply voltage. Considering an RO pair with frequency difference smaller than the ideal f_{thp} value, if the slower RO slows down more than the faster RO when the environmental fluctuations occur, a symbol error does not occur, since the ordering does not change. This situation is illustrated in Figure 6.3. Since two different orderings may occur

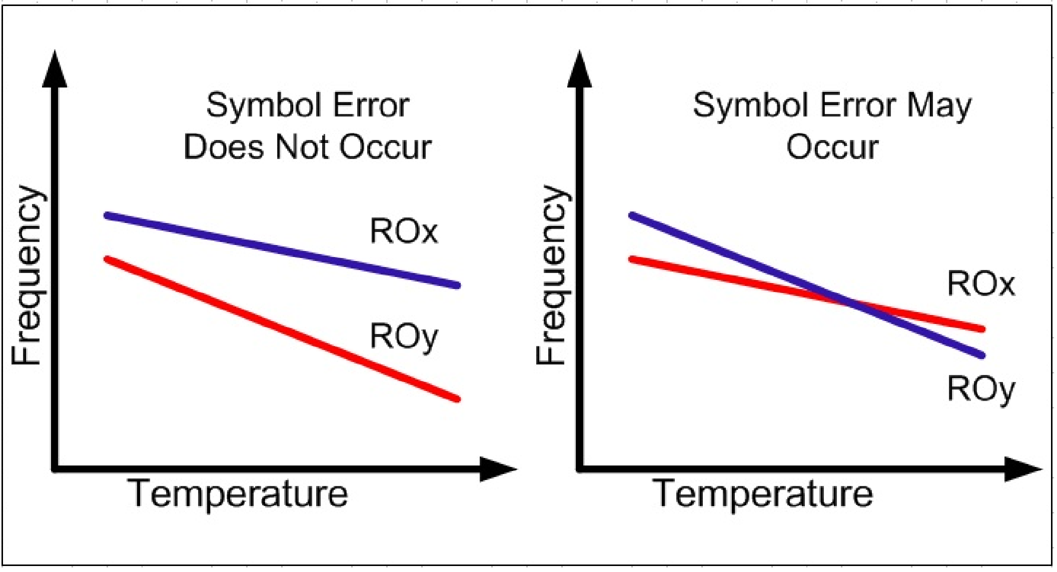


Figure 6.3. Effect of ordering on symbol error [10].

within the RO pair with equal probability, probability of the correct order is 0.5.

When the problematic RO pair is placed in a group with the correct order, a symbol error is likely to occur due to extreme environmental conditions. Since we cannot determine the probability of these conditions and these depend on the working conditions of the device, the probability of environmental fluctuations to create a symbol error is assumed to be one (worst-case scenario). Resulting probability of symbol error is the product of probabilities of conditions stated above and presented as

$$p = 0.5 * p_{av} = \sum_{s=2}^{m} k_s * (s-1)/(M*(M-1)). \tag{6.5}$$

Assuming that a problematic RO pair is somehow present in a system of 160 ROs, the symbol error rate is calculated based on the data presented in the previous section and equations (8.3) and (8.2) for f_{thp} values in the range of 0.5 MHz to 1.1 MHz. As can be seen from Figure 6.4, theoretically calculated symbol error rate varies linearly between 5.26×10^{-3} and 4.40×10^{-3}.

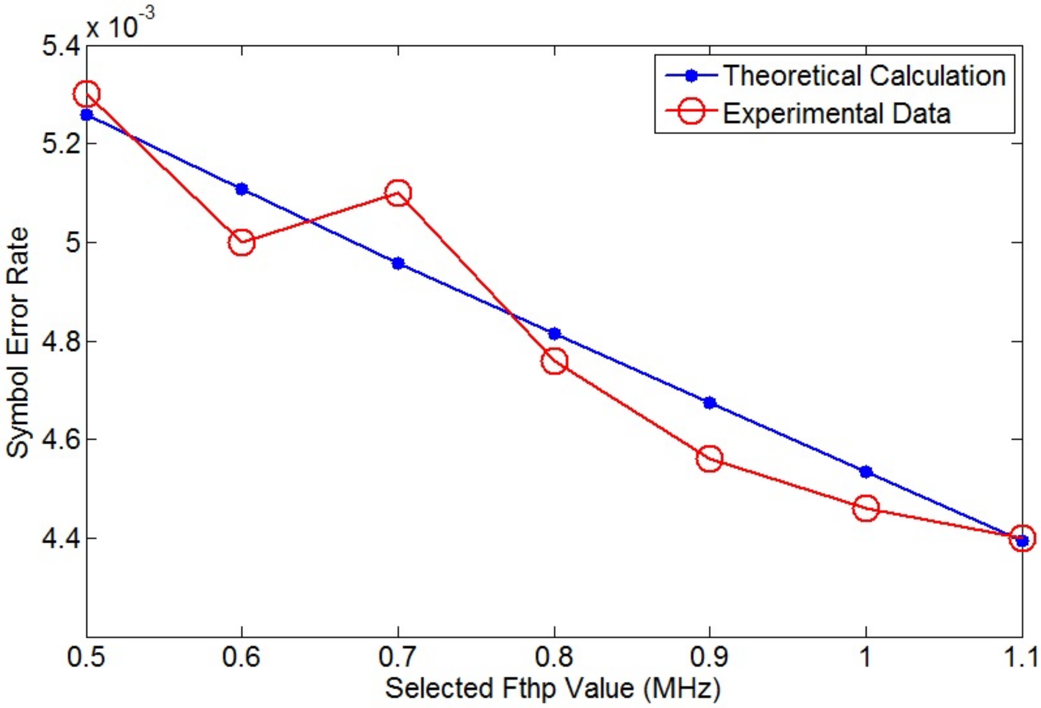


Figure 6.4. Symbol error rate.

Validation of the symbol error probability analysis is done via creating random sets of RO frequencies in Matlab environment. For this purpose, 30,000 sets of 160 RO frequencies with Gaussian distribution are created. The mean and STD of the distribution are obtained from real data collected from FPGA implementation. Then, DP algorithm is applied to each set of 160 ROs using an f_{thp} value of 0.5 MHz. Next, a problematic RO pair is defined by choosing two ROs randomly from RO_1 to RO_{160}. Then, the RO sets that include the two elements of the selected RO pair next to each other and in the correct order are counted to determine the symbol error rate. This process is repeated for different f_{thp} values in the range of 0.5 MHz to 1.1 MHz. The results are presented in Figure 6.4. As can be seen from the figure, experimental data agrees with the theoretical calculations validating the analysis presented above.

6.1.3. Symbol Error to Bit Error Conversion and Bit Error Probability

When the ordering in a group changes, a symbol error occurs and the output becomes erroneous. The number of bits affected by a symbol error depends on the output generation method. In order to analyze the bit error probability, two simple and

Table 6.1. Output Generation Mapping.

Frequency Ordering	**Output Bits by Direct Mapping**	**Output Bits by Gray Encoding**
RO1>RO2>RO3	000	000
RO1>RO3>RO2	001	001
RO2>RO1>RO3	010	011
RO2>RO3>RO1	011	010
RO3>RO1>RO2	100	110
RO3>RO2>RO1	101	111

efficient output generation methods are selected, direct mapping and Gray encoding. With these methods, each ordering in a group of length s is mapped to a bitstream with $\lceil \log_2(s!) \rceil$ bits. An example mapping for a group size three is presented in Table 6.1.

By using the mapping functions, a symbol error results in a certain number of changes in the output bitstream. Since the number of erroneous bits depends on the group size and pre-error and post-error orderings, a complete analysis is required to determine the bit error probability. The number of symbol error cases, E_c, for a group of size s, can be calculated using the number of possible pre-error orderings, $s!$, and possible post-error orderings, $s-1$, given as

$$E_c = s! * (s-1). \tag{6.6}$$

For group sizes of up to 25 ROs, the number of bit errors is calculated for E_c and the average number of bit errors for a group size of s, $bepg_s$, is presented in Figure 6.5.[3] Next, bit error probabilities per generated bit are calculated for both mapping methods and presented in Figure 6.6. As can be seen from the figure, the bit error probability is lower in larger groups. Finally, by using the symbol error rate and $bepg_s$,

[3] Bit errors are calculated for all E_c up to group sizes of 10. For group sizes larger than 10, Monte Carlo method is applied and bit errors are calculated via 500,000 random trials.

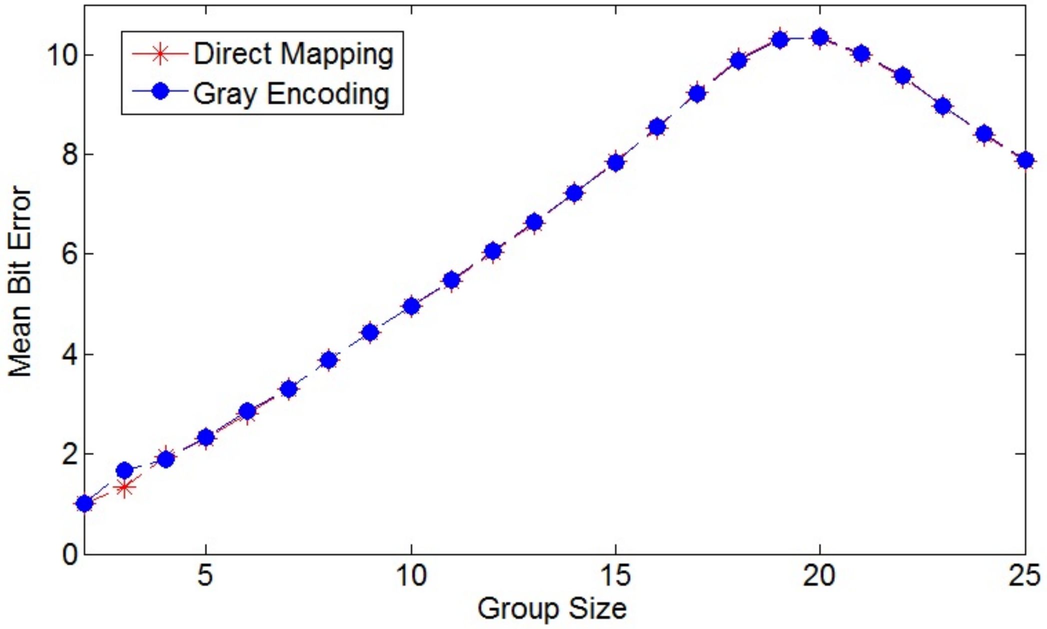


Figure 6.5. Number of bit errors per group.

the bit error probability per problematic RO pair, *bep* can be calculated as

$$bep = p * bepg = \sum_{s=2}^{m} (k_s * bepg_s) * (s-1)/(M * (M-1)). \tag{6.7}$$

Bit error probability for f_{thp} values in the range of 0.5 MHz to 1.1 MHz is presented in Figure 6.7. Interestingly, both mappings result in similar bit error probabilities. For both mappings, bit error probabilities are as low as 10^{-2} for an f_{thp} value of 1.1 MHz, whereas for an f_{thp} value of 0.5 MHz, they are slightly higher than 2.4×10^{-2}.

6.1.4. Worst Case Bit Errors per Problematic RO Pair

As discussed above, the number of erroneous bits due to a symbol error differs due to group size, and pre-error and post-error ordering of RO frequencies. Even though the average bit error rate gives a clear idea about the erroneous output, the worst case data will also be helpful for some applications, such as key generation, where the system performance heavily relies on the maximum number of errors in an

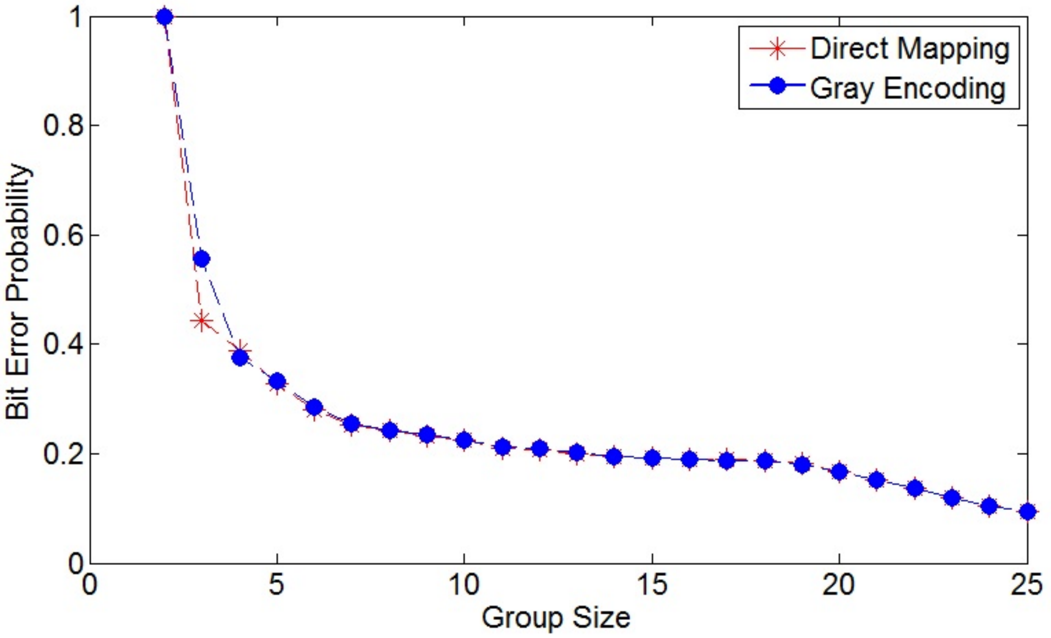


Figure 6.6. Bit error probability per generated bit.

output. In this sense, based on group sizes, all pre-error and post-error orderings are analyzed to determine the maximum number of output bits that can flip due to a symbol error. For both direct mapping and Gray encoding, the maximum number of bit errors at the output is shown in Figure 6.8 as a function of the group size. As can be seen from the figure, using larger groups has a disadvantage in terms of the worst case bit error despite their area efficiency. In addition to this, Gray encoding has a significant advantage over direct mapping for almost all group sizes, since gray encoding aims minimum number of bit change between similar orderings.

Even though the number of erroneous bits at the worst case is a very critical information, their occurrence probability is also important. For this purpose, the occurrences of worst symbol errors among all possible errors are counted and their probabilities are presented in Figure 6.9 for group sizes of up to 20.[4] As can be seen from the figure, the worst case symbol error probabilities, $wcsep_s$, decrease swiftly with increasing group sizes and approach 10^{-6} for a group size of 20.

[4] Bit errors are calculated for all E_c up to group sizes of 10. For group sizes larger than 10, Monte Carlo method is applied and bit errors are calculated via 500,000 random trials.

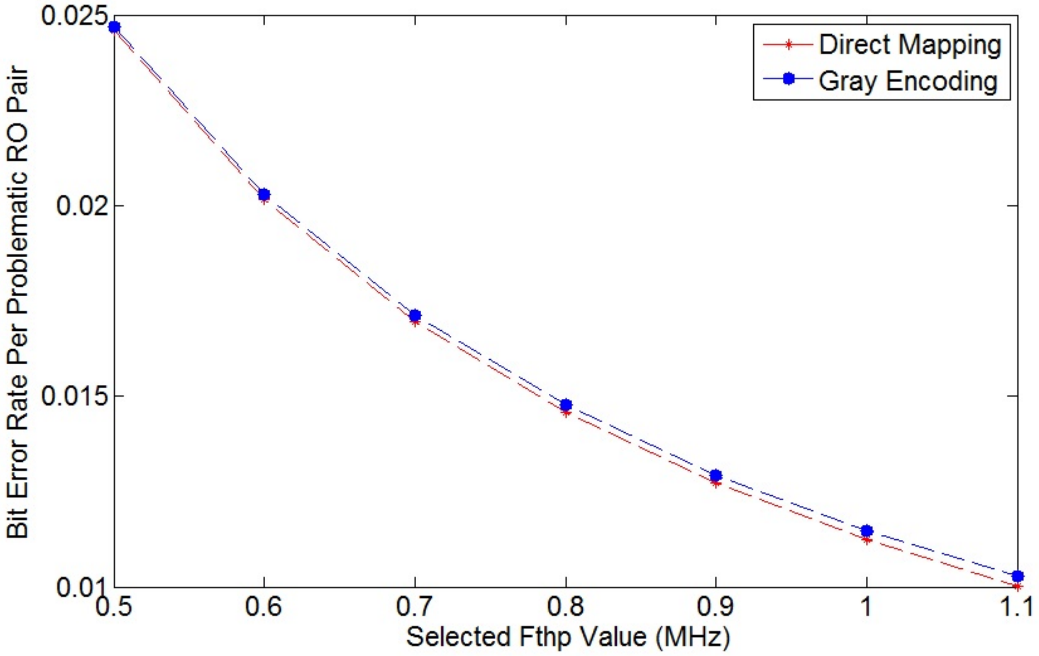


Figure 6.7. Bit error probability per problematic RO pair.

6.2. Area Usage vs. Robustness in Ordering Based RO-PUFs

Unlike cryptographic key generation applications, erroneous outputs are acceptable up to a certain degree in some other applications, such as authentication [1, 21]. The degree of the acceptable error rate depends on the false rejection rate (FRR) and false acceptance rate (FAR) requirements. Due to the errors at the output, a circuit may be authenticated as another circuit, which contributes to the FAR. Similarly, a circuit may unnecessarily fail to be authenticated, which contributes to the FRR [1]. [70] specifies the relation between the number of erroneous bits in the output and the false acceptance and rejection rates for authentication. For instance, if 10 bits out of a total output length of 128 bits are allowed to be erroneous, the FAR can be calculated as 2.1×10^{-21} and the FRR can be calculated as less than 5×10^{-11}.

The immunity of these applications to erroneous outputs up to a degree enables increasing the area efficiency of the underlying PUF circuit. As described in Chapter 5, choosing an f_{thp} value smaller than the optimum value results in forming larger groups by the DP algorithm; hence, increasing the entropy extraction from the system. This enables an implementation consisting of a smaller number of ROs to achieve a target

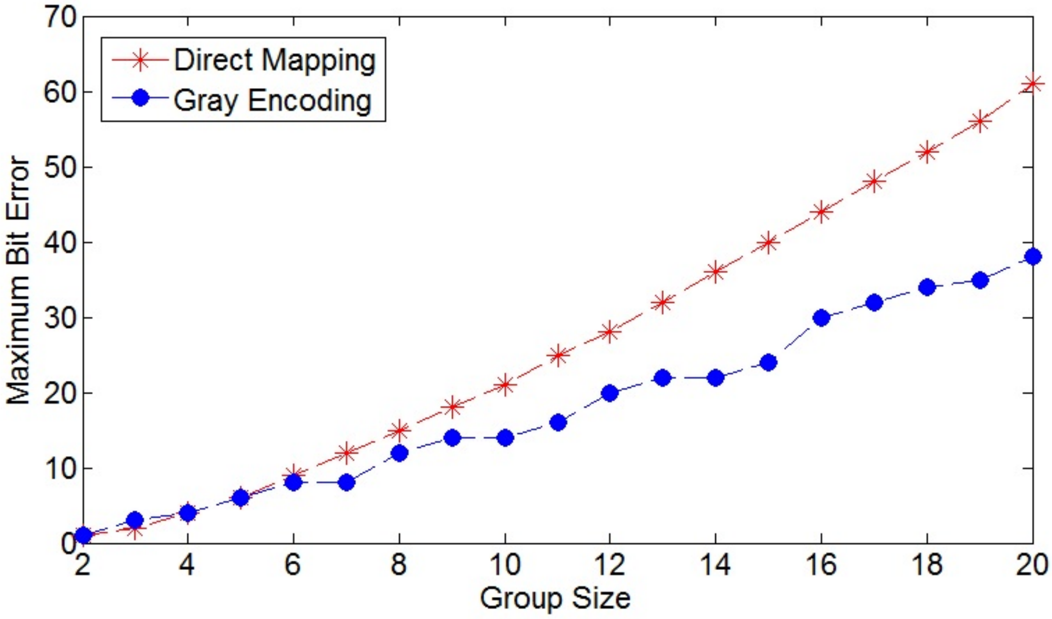


Figure 6.8. Maximum error vs. group size.

number of output bits.

In order to quantify the effects of non-ideal f_{thp} values on the number of erroneous bits at the output, an analysis is performed to determine the number of RO pairs that may change their ordering under extreme conditions. For this purpose, the RO frequency data, collected from the sample implementation is used and an f_{thp} value of 1 MHz is assumed as ideal. Then, the RO pairs with frequency deviation more than the so-called non-ideal f_{thp} values in the range of 0.85 MHz to 1 MHz are counted to determine the maximum number of problematic RO pairs. This results in an upper bound on the error vulnerability with respect to the non-ideal f_{thp} values. It can be seen from Table 6.2 that, as the non-ideal f_{thp} values get smaller and smaller, the number of problematic RO pairs increases significantly, as expected. On the other hand, due to the increase in group sizes, the entropy extraction is increased and so is the area efficiency. In order to analyze the improvement in the area consumption of the system, the number of required ROs for a fixed 128 bits of output is analyzed on a sample system using the DP algorithm with non-ideal f_{thp} values. As shown in Table 6.2, the required number of ROs decreases 9% with a reduction of 150 kHz in the f_{thp} value.

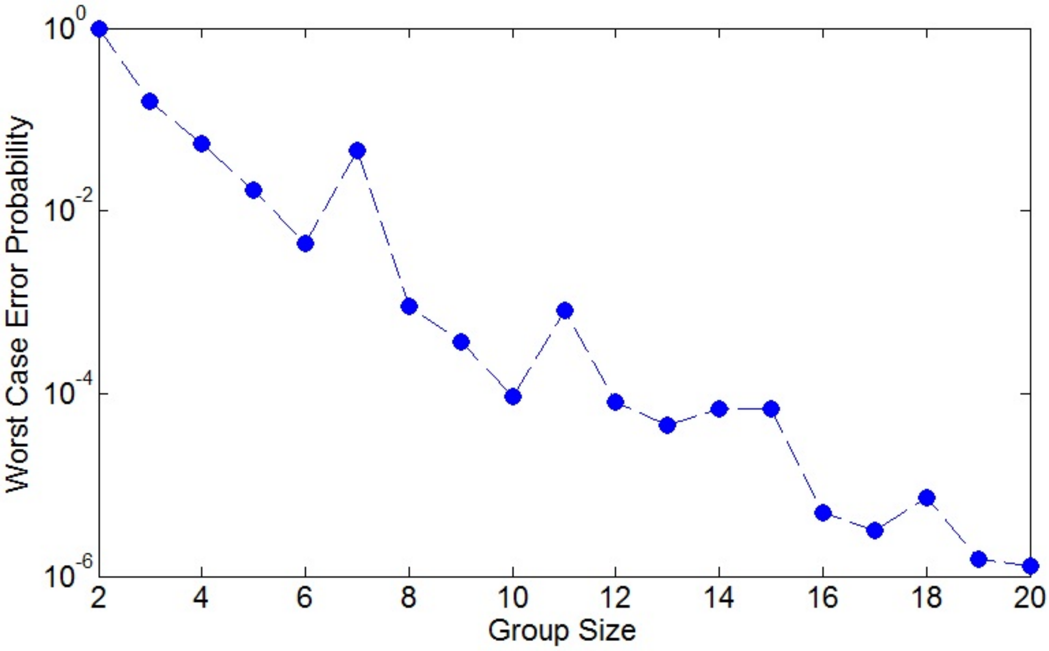


Figure 6.9. Worst case symbol error probability vs. group size.

Table 6.2. Number of Problematic RO Pairs.

Selected f_{thp} (MHz)	**1**	**0.95**	**0.9**	**0.85**
Num. of Problematic Pairs	0	2	5	15
Num. of Required ROs	114	110	108	105

Since each problematic RO pair contributes to the generation of symbol errors, the bit error probability of the system will increase directly proportional to the number of problematic RO pairs. Based on the bit error probability per problematic RO pair analysis presented in Figure 6.7, and the analysis on the number of problematic RO pairs given in Table 6.2, the bit error probability of the sample system with respect to the selected f_{thp} value can also be calculated. As can be seen from Figure 6.10, the bit error probability increases significantly as the selected f_{thp} value decreases and a 150 kHz reduction causes over 20% of the output bits to be erroneous in the average.

The worst case bit error amount depends on the number of symbol errors that occur at the same time, n. Based on the number of problematic RO pairs k, and the symbol error probability per problematic RO pair p, the probability of simultaneous

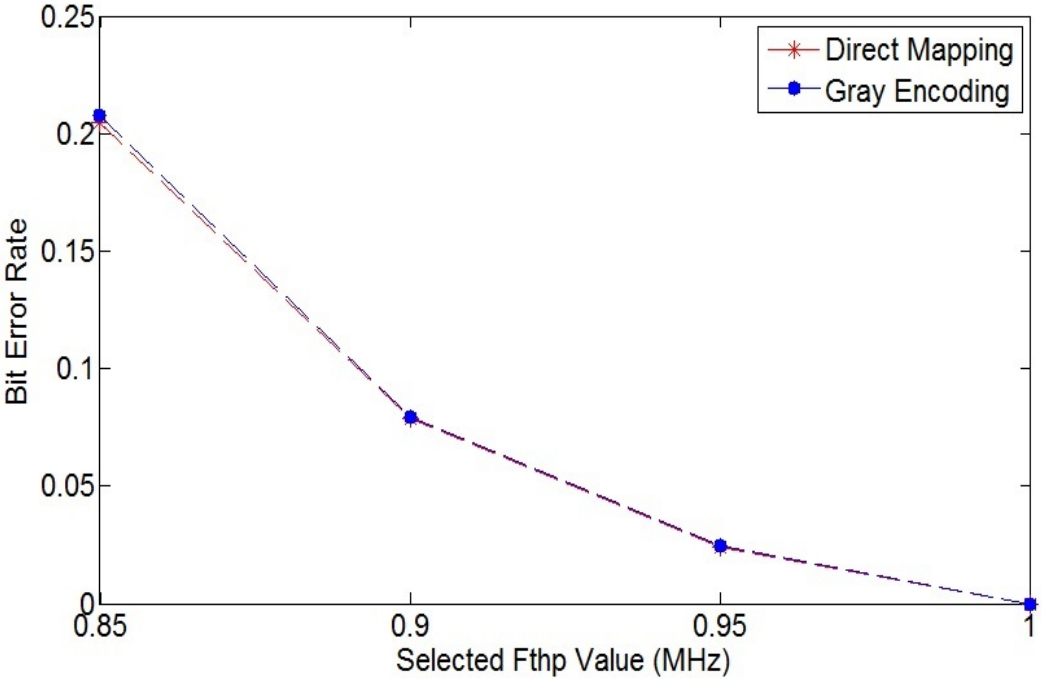


Figure 6.10. Bit error probability vs. f_{thp} chosen.

n errors, $p_{n,k}$, can be calculated as

$$p_{n,k} = p^n * (1-p)^{k-n} * \binom{k}{n}, \tag{6.8}$$

meaning n out of k problematic pairs will create errors and $k - n$ problematic pairs will not create errors.

Based on this relation, the symbol error rates for different numbers of simultaneous errors are calculated and presented in Table 6.3. As can be seen from these results, the probability of multiple errors occurring simultaneously decreases exponentially as n increases. Using this data, the maximum error amount per group size, and the expected largest group size, a designer can choose the f_{thp} value maintaining the system requirements. The probability of the worst case error situation, $wcep_n$, can be calculated as

$$wcep_n = p_{n,k} * wcsep_s^n, \tag{6.9}$$

Table 6.3. Simultaneous Symbol error probabilities for non-ideal f_{thp} values.

Selected f_{thp} **(MHz)**	**1 error**	**2 error**	**3 error**	**4 error**	**5 error**
0.95	0.0092	2.1178e-005			
0.90	0.0229	2.15e-004	1.01e-006	2.37e-009	2.22e-012
0.85	0.0666	0.0022	4.58e-005	6.56e-007	6.88e-009

where $wcsep_s$ is the worst case symbol error probability of a group size s, $p_{n,k}$ is the probability of simultaneous n errors when k problematic RO pairs are present in the system, and n is the number of simultaneous errors, assuming that all errors will take place within the largest groups. For instance, if the probability of three or more errors at a time can be disregarded and the largest group size is limited by 10, the worst case bit error amount for an f_{thp} value of 0.9 MHz will be 28 with Gray encoding and the probability of error caused by this will be 1.83×10^{-12}.

6.3. Implementation and Analysis of Error Correction Codes

As discussed previously, f_{thp} parameter used in DP algorithm determines the robustness level and area consumption of the ordering based RO-PUF systems. Based on this property, lowering the robustness of the PUF outputs by decreasing the f_{thp} value and increasing the area efficiency is an option for the systems that are tolerant to errors up to a certain level. For the systems that are not immune to errors, such as key generation applications, ECC can be utilized if the cost of ECC are less than the expected area gain achieved by lowering the f_{thp} value. For this purpose, ECC are implemented to determine if a better area utilization can be achieved in ordering based RO-PUFs, without decreasing the robustness of the system.

Error detection and correction techniques, which have been developed in the areas of information theory and coding theory, enable reliable delivery of digital data over unreliable communication channels. In many systems, transmitted data become erroneous up to a degree at the receiver side due to noise introduced by the transmission channel [71]. In addition to this, many applications are very vulnerable to

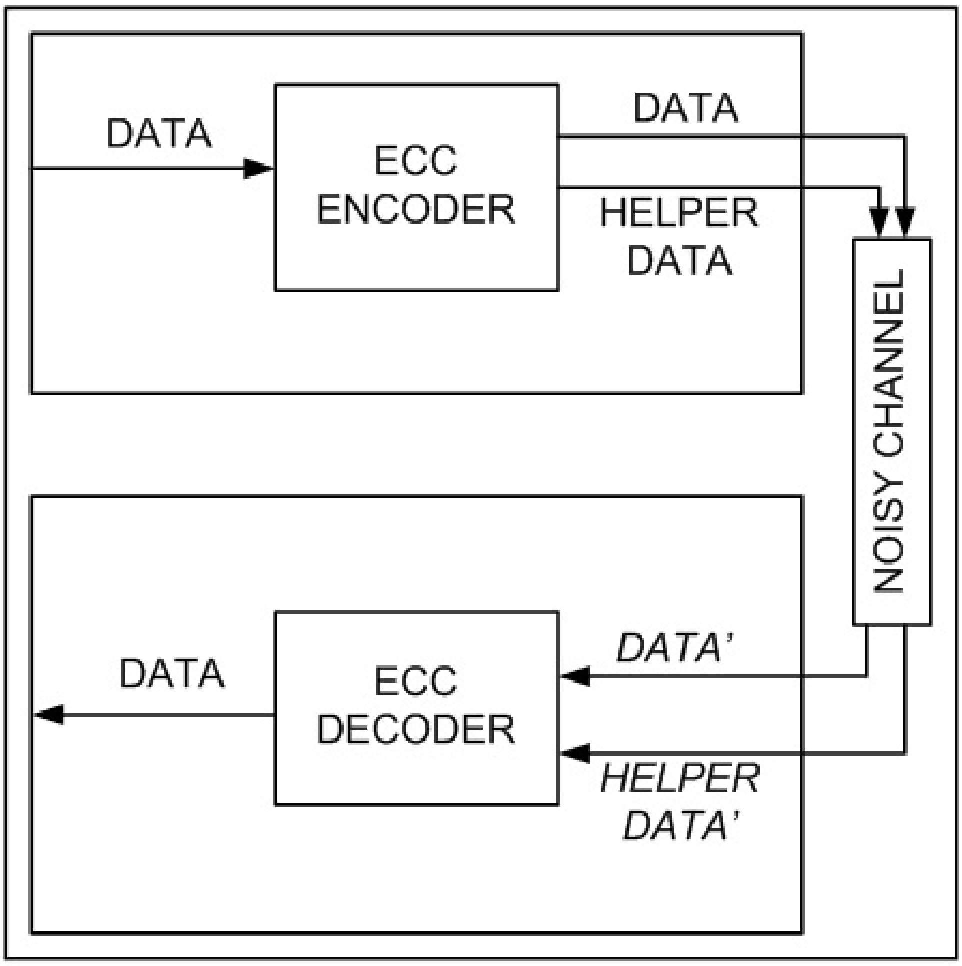


Figure 6.11. A system utilizing ECC.

erroneous data and removing the noise is mandatory. Therefore, ECC are very useful in many applications with their ability of removing the noise present in the data [72]. Some of the areas that ECC are utilized are NAND flash memories, wireless communications, fiber communications, optical communications, finite field multiplier circuits, watermarking algorithms, video watermarking extraction, and solid-state device controllers [73–84]. The main working principle of ECC is adding redundancy to the data that is required to be recovered when it is corrupted by noise [85]. This is done by the encoder part of the ECC and helper data is generated. Then, data and helper data are transmitted over a noisy channel and the noise introduced is removed by the ECC decoder using the helper data. This is illustrated in Figure 6.11. In this figure, DATA is the original bitstream and $DATA'$ is the noisy bitstream. The amount of noise that can be removed is limited and depends on the complexity of the ECC algorithm and length of the helper data.

One of the commonly used ECC is linear block code, and Hamming codes are the simplest block codes within the linear block code structures [86]. Hamming codes are composed of very trivial error correction schemes; hence, result in very low area overhead in the system. However, they can only correct one random error in the bitstream. This property limits their widespread usage [71]. BCH codes are the generalization of Hamming codes that are designed for multiple-error correction. BCH codes are implemented on Galois fields or finite fields [87]. They form a large class of powerful random error correcting cyclic codes. Cyclic codes utilize circular shifts and each code word generates another word that belongs to the code after the circular shift operation. Algebraic properties of cyclic codes allow building efficient error correction mechanisms [88].

The usage of ECC in PUF implementations are illustrated in Figure 6.12. As can be seen from the figure, PUF output is applied to the ECC encoder and helper data is generated and recorded to a database during the initialization phase. Then, during the usage phase, ECC decoder removes the noise present in the PUF output by using the information stored in the helper data. Since it is almost impossible to guarantee single error in outputs, BCH codes are convenient for data recovery in PUF circuits. In this study, BCH codes are implemented and analyzed in terms of area and timing performance.

The capabilities of multi-bit correcting ECC are shown with a three item notation, (a, b, c). In this format, a represents the total number of data and helper data bits, b represents the total number of data bits, and c represents the maximum number of erroneous bits that the ECC can recover successfully in noisy data. As the number of maximum number of erroneous bits that can be recovered increases, the complexity; hence, the area, time, and power consumption of both ECC encoder and decoder increase as well.

In order to determine the area overhead of ECC on PUF systems, BCH encoders and decoders for different output lengths and error correction capabilities are implemented on Xilinx Spartan 3S5000 type of FPGAs and their area usages are analyzed.

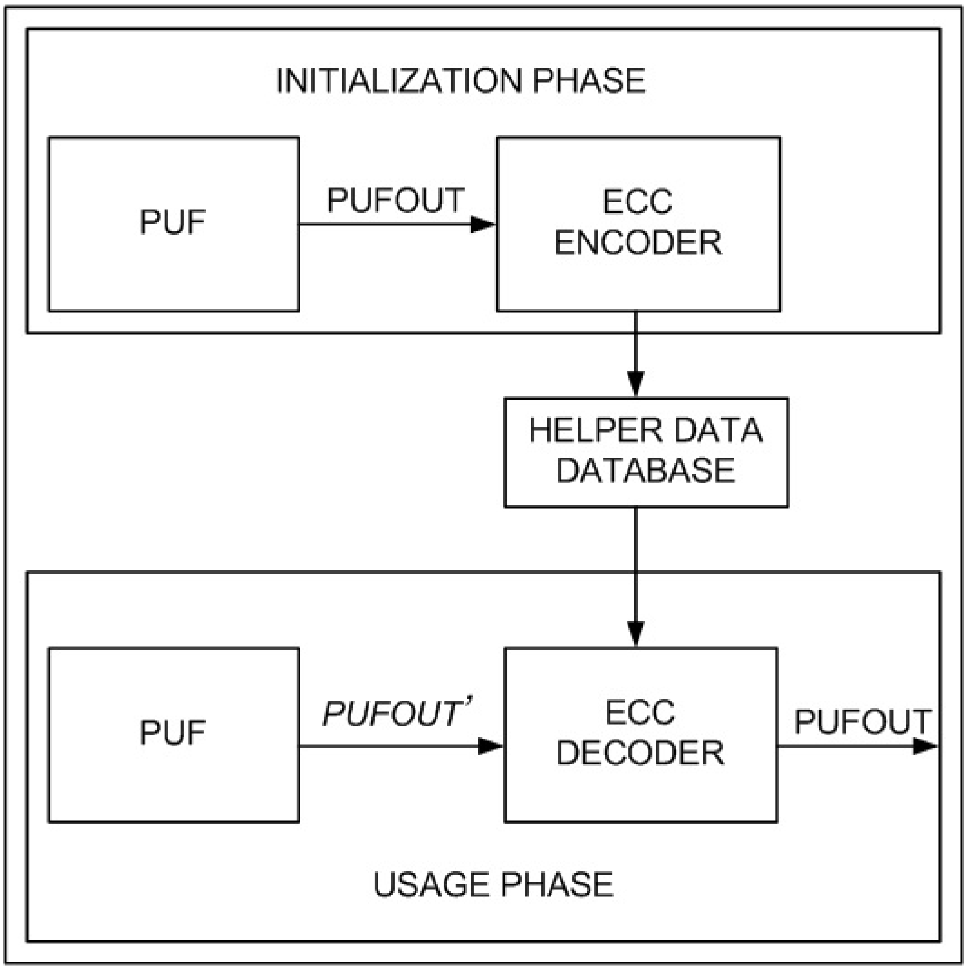


Figure 6.12. Use of ECC in PUF based systems.

In Figure 6.13, area usage of the BCH encoders with output lengths 15 to 511 are presented. In this analysis, a is chosen as $2^n - 1$, where n is an integer in the range of 4 to 9 and b is chosen almost half the value of a in all cases for a fair comparison. As can be seen from the figure, area usage increases with a. For instance, $(127, 64, 10)$ BCH encoder consumes 32 slices, whereas $(255, 131, 18)$ BCH encoder consumes 60 slices on the specified FPGA.

The same analysis is repeated for BCH decoders and area consumptions of the BCH decoders with output lengths 15 to 511 are presented in Figure 6.14. As can be seen from the figure, area usage increases linearly with the output length similar to the encoder case. However, area consumptions of the BCH decoders are significantly higher than the BCH encoders with the same parameters, especially for longer output lengths. For instance, $(511, 259, 30)$ BCH decoder consumes an area of more than

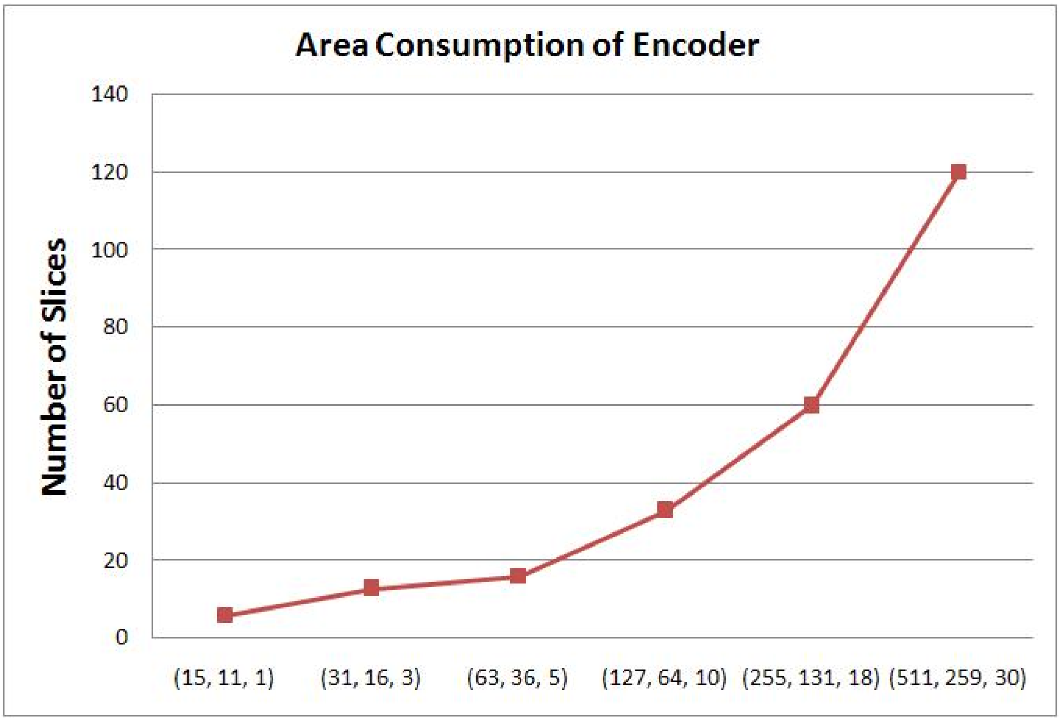


Figure 6.13. Area consumptions of BCH encoders with different data lengths and error correction capabilities.

1,400 slices, which is 12 times higher than the $(511, 259, 30)$ BCH encoder circuit.

Next, a parameter is kept constant at 255 bits and the area consumptions of the BCH encoders with six different c values, 3, 6, 9, 12, 15, and 18 are analyzed and presented in Figure 6.15. As can be seen from the figure, area usage of the encoder increases with increasing error correction capability. For instance, $(255, 131, 18)$ BCH encoder has three times more area consumption than the $(255, 231, 3)$ BCH encoder.

The same analysis is repeated for the BCH decoder. Based on this analysis, the number of occupied slices for BCH decoders with different error correction capabilities are presented in Figure 6.16. As can be seen from the figure, area usage increases linearly with increasing number of recoverable erroneous bits. For instance, $(255, 131, 18)$ BCH decoder has four times more area consumption than the $(255, 231, 3)$ BCH decoder.

Timing behavior of ECC is another important performance parameter. Execu-

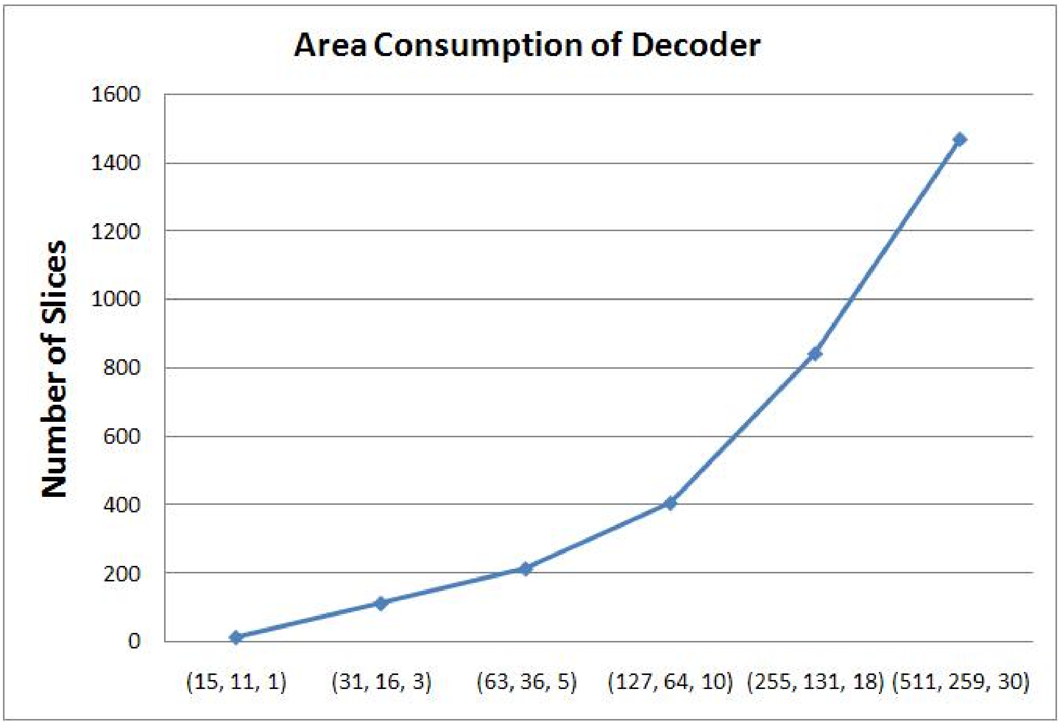


Figure 6.14. Area consumptions of BCH decoders with different data lengths and error correction capabilities.

tion times of the BCH encoder and decoder implementations are analyzed for different output lengths and error correction capabilities. As can be seen from Figure 6.17, execution time of encoding operation is a clock cycles. For instance, $(255, 131, 18)$ and $(511, 259, 30)$ BCH encoders operate in 255 and 511 clock cycles, respectively. When the analysis is repeated for the decoding operation, the number of clock cycles required is more than twice of the a clock cycles. As can be seen from Figure 6.18, $(255, 131, 18)$ and $(511, 259, 30)$ BCH decoders operate in 567 and 1101 clock cycles, respectively.

Even though using ECC ensures the robustness of PUF outputs and it is a reliable solution, implementation results of BCH codes have shown that the cost of adding ECC to the system is quite high in terms of area and brings complexities, such as the need for storing and using the helper data generated. Therefore, using the ideal f_{thp} value rather than using a lower f_{thp} value and adding ECC to the system is a more efficient way in terms of area consumption of the system in ordering based RO-PUFs. However, if an ECC implementation is already present in the system for other purposes, it will be convenient to utilize them during the PUF operation. This will enable lowering

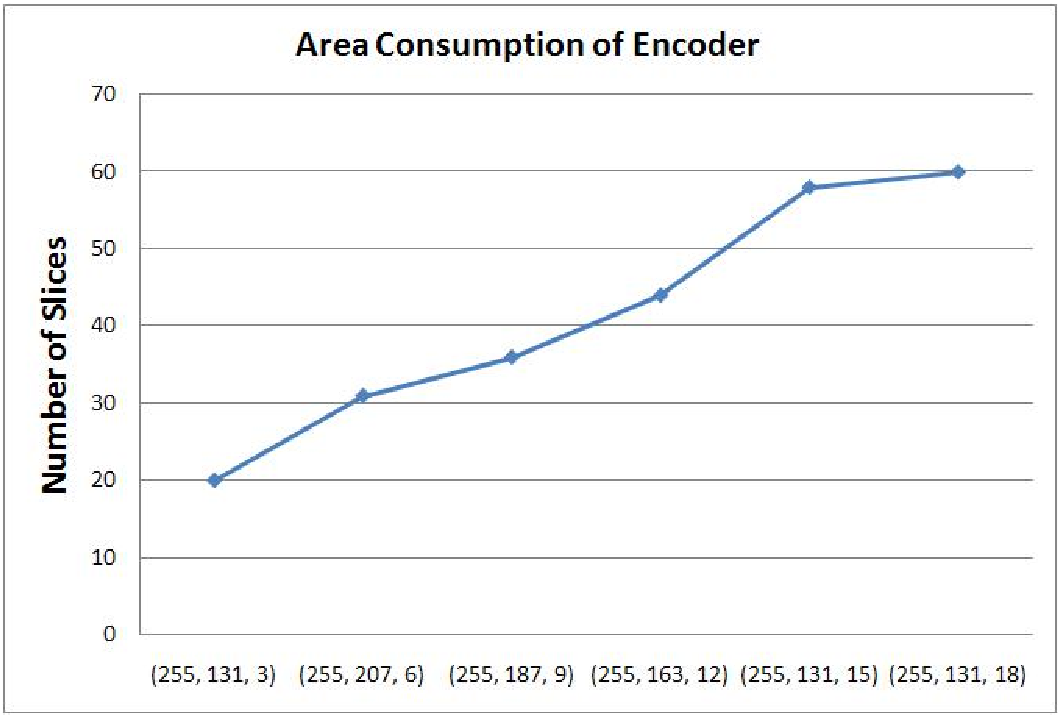


Figure 6.15. Area consumptions of BCH encoders with different error correction capabilities.

the f_{thp} value and increasing the area efficiency of the system as discussed previously. In addition to these, using ECC with conventional RO-PUFs, where noisy outputs are generated by comparing two ROs for each bit, is a good option for generating robust outputs. When the timing behavior of ECC implementations are considered, both the operating times of BCH encoder and decoder is reasonable, since the output generation time of RO-PUFs are quite long compared to the ECC operation.

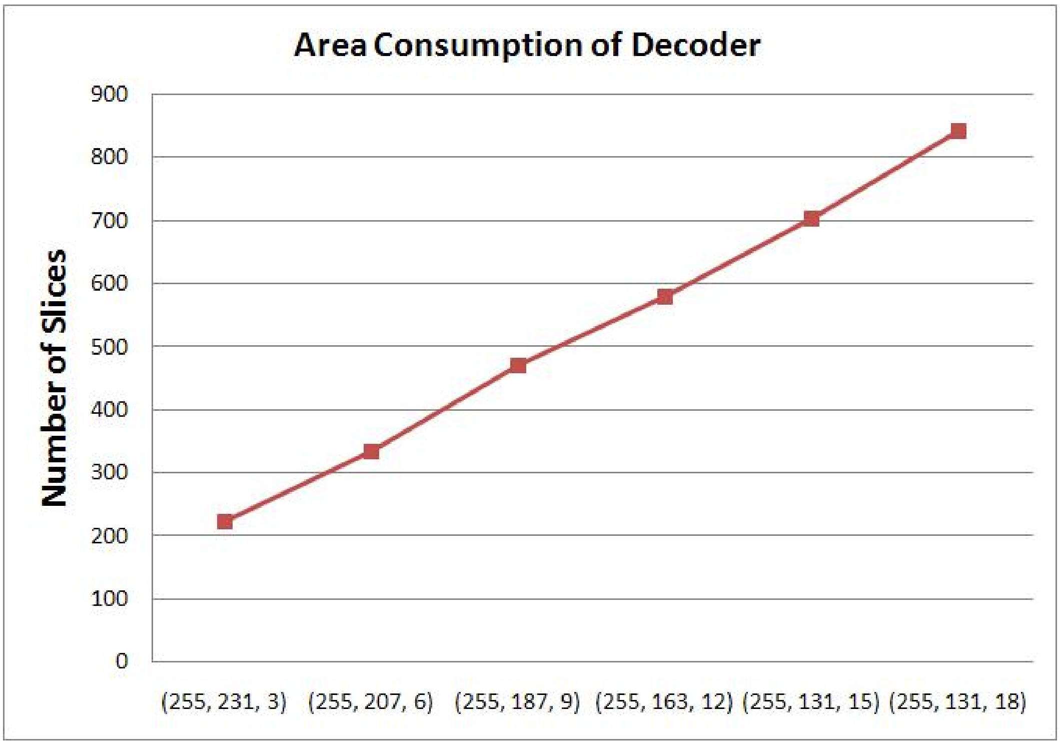


Figure 6.16. Area consumptions of BCH decoders with different error correction capabilities.

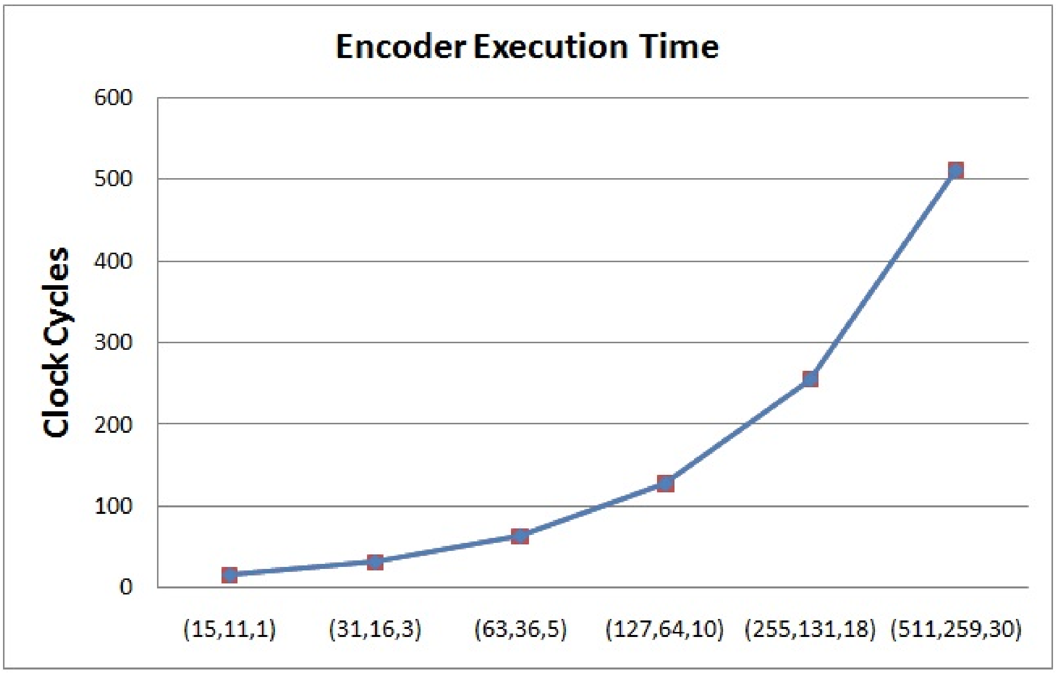


Figure 6.17. Execution times of BCH encoders with different data lengths and error correction capabilities.

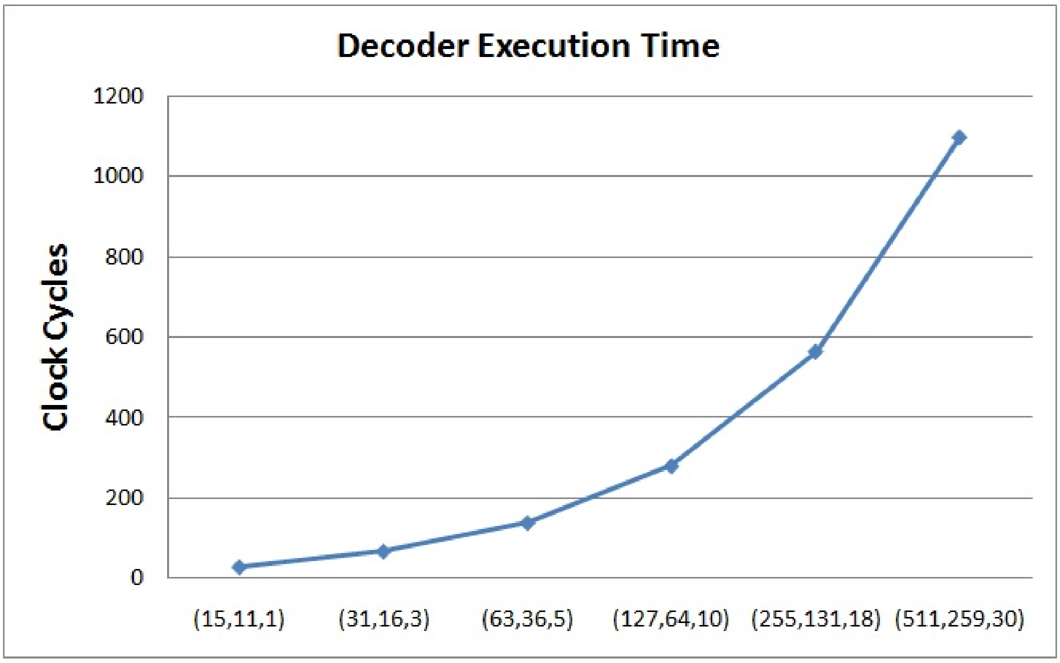


Figure 6.18. Execution times of BCH decoders with different data lengths and error correction capabilities.

7. CRP GENERATION

7.1. CRP Concept in PUFs

PUFs are used to generate signatures on individual ICs by utilizing the random components in the manufacturing process. Generating a static digital output without using an input is the first method developed to identify circuits. The second method is to generate many CRPs on each IC, which is more convenient for security applications, such as authentication. In this method, the challenge is a stimulus to the system and the response is the output that depends on the challenge and the transient behavior of the IC. The number of CRPs is strongly related to the number of inputs to the system [30].

PUF types are divided into two groups as weak PUFs and strong PUFs based on the number of unique CRPs provided [19]. Strong PUFs provide a large number of CRPs based on the high amount of entropy present in the system and thus they can be used in authentication. However, weak PUFs do not support the CRP concept or allow only a small number of challenges to be applied. Arbiter PUFs support an exponential number of CRPs based on the number of stages. In such a system, reading all CRPs is impossible. However, arbiter PUFs have weaknesses allowing modeling attacks and they are not suitable for FPGA implementation, which limit their usage [28]. SRAM PUFs are not suitable for CRP applications, since the number of SRAM cells is limited on any device and full read-out is possible and very fast [24]. RO-PUFs, which are the most convenient PUFs for FPGA implementation and work reliably under changing environmental conditions, suffer from a small number of CRPs [10,45]. A conventional RO-PUF, which compares RO frequencies one-by-one, can be characterized by $n(\log n)$ bits of information and can supply a maximum number of n^2 CRPs. This makes full read-out possible [89]. As discussed in Chapter 5, ordering based RO-PUFs enable 100% robust, noise-free outputs with higher entropy extraction than the conventional RO-PUFs. In these systems, a single output is generated, which can be used as a secret key without adding any error correction mechanism. In spite of these advantages, ordering based RO-PUFs presented in the literature do not support authentication systems, since the CRP concept is not yet defined for them [9]. In this chapter, two CRP enhancement methods, the Pre-determined Frequency Threshold Selection and the RO Selection methods are proposed and their analysis results are presented in terms of CRP quality, CRP count, and area, time, and power efficiency.

7.1.1. CRP Properties

The properties of PUF circuits in terms of CRP behavior can be stated as follows [19, 30]:

(i) A response R_i to a challenge C_i should not give much information about response R_j to challenge C_j, $i \neq j$.

(ii) It should be almost impossible to predict the response R_i to a challenge C_i without using the corresponding PUF circuit.

(iii) The CRP behavior of the PUF should change drastically when an invasive attack is performed on device (tamper evidence property).

(iv) CRPs should be easily evaluated by the PUF circuit.

7.1.2. Importance of Large Number of CRPs

The number of CRPs that a PUF type provides is an important parameter for five reasons that are stated as follows [4, 30]:

(i) Since each CRP can be used only once during authentication, higher number of CRPs allow higher number of authentication processes with the same circuit.

(ii) Large number of CRPs allow generation of longer and stronger PUF outputs with limited resources.

(iii) Large number of CRPs allow identification of bigger populations.

(iv) Emulation attack, which aims at storing all possible CRPs in a memory is not applicable due to insufficient storage when the PUF circuit supports an exponential number of CRPs.

(v) If CRPs are used more than once during the authentication process due to their scarcity, an attacker can make a copy of the database by a man-in-the-middle attack and unauthorized accesses to the system may become possible.

As presented above, the number of CRPs supported by the PUF circuit is very important. Adding CRP support to PUF types like ordering-based RO-PUFs, where a single output is generated, allows the primitive to be used in a wider range of application areas. In addition to this, increasing the number of CRPs makes attacks such as emulation and man-in-the-middle impractical.

Enhancing the CRP set in ordering based RO-PUFs can be achieved via utilizing the inputs as challenges. As discussed in Chapter 5, the DP algorithm has two sets of inputs. The first input is the list of RO frequencies measured on IC and the second input is the chosen f_{thp} value. Two methods utilizing these inputs as challenges are presented in the following two sections.

7.2. Enhanced CRP Set with the f_{thp} Selection Method

f_{thp} is the main parameter in determining the RO groups via DP. Since the outputs are created by frequency comparison within these groups, different sets of groups will result in different outputs. In this context, the f_{thp} parameter itself can be used as the challenge to the system and the PUF output will behave as the response.

The main problem in this so-called f_{thp} selection method is to determine the range of f_{thp} values and the minimum difference between any two f_{thp} values that will be used as challenges. The minimum f_{thp} value, f_{thpmin}, depends on the noise present

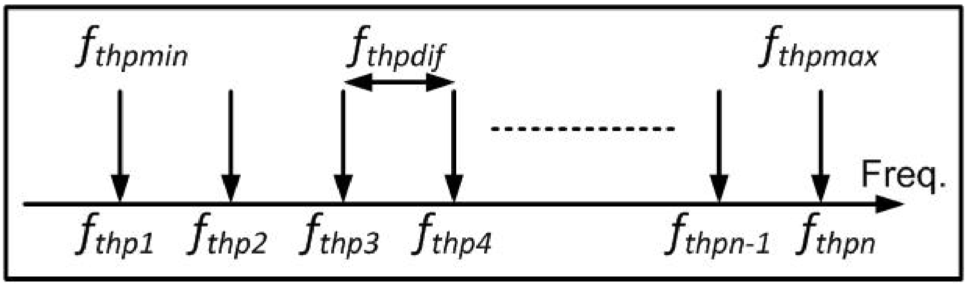


Figure 7.1. The relation between the f_{thpmin}, f_{thpmax}, and f_{thpdif} values.

in the system and the frequency fluctuations of ROs due to environmental variations, as described in [90]. The maximum f_{thp} value, f_{thpmax}, depends on the number of CRPs required by the application and the area, speed, and power consumption requirements of the system. As the f_{thpmax} value increases, the range for selecting f_{thp} values to be used as challenges gets bigger; hence, the number of CRPs provided by the PUF circuit increases. However, increasing the f_{thpmax} value decreases the efficiencies of area, speed, and power of the system. With higher f_{thp} values, ROs will form smaller groups and the entropy extraction of the system will be lower. This will increase the minimum number of ROs that will be implemented in the system in order to generate the required output length. Increasing the number of ROs also increases the evaluation time of the PUF output and the power consumption of the system. The relation between the f_{thpmin}, f_{thpmax}, and f_{thpdif} values are illustrated in Figure 7.1.

f_{thpdif} value is the minimum frequency difference allowed between any two f_{thp} values. f_{thpdif} value directly determines the number of CRPs provided by the system and the independence of the outputs. When the f_{thpdif} value is small, adjacent f_{thp} values will be near each other and more CRPs will be available within the defined range. However, the formed RO groups may be similar for certain challenges, increasing the correlation of the outputs, which is a disadvantage. As the f_{thpdif} increases, the number of CRPs will diminish, but the independence of the outputs will be maintained. Since the frequency distribution of the implemented ROs in the system depends on the technology used, design and layout of the ROs, and environmental properties, the optimum value for f_{thpdif} should be determined by measuring a subset of the ICs manufactured or FPGAs programmed. The number of CRPs generated with the f_{thp} selection method, CRP_{num}, can be calculated as

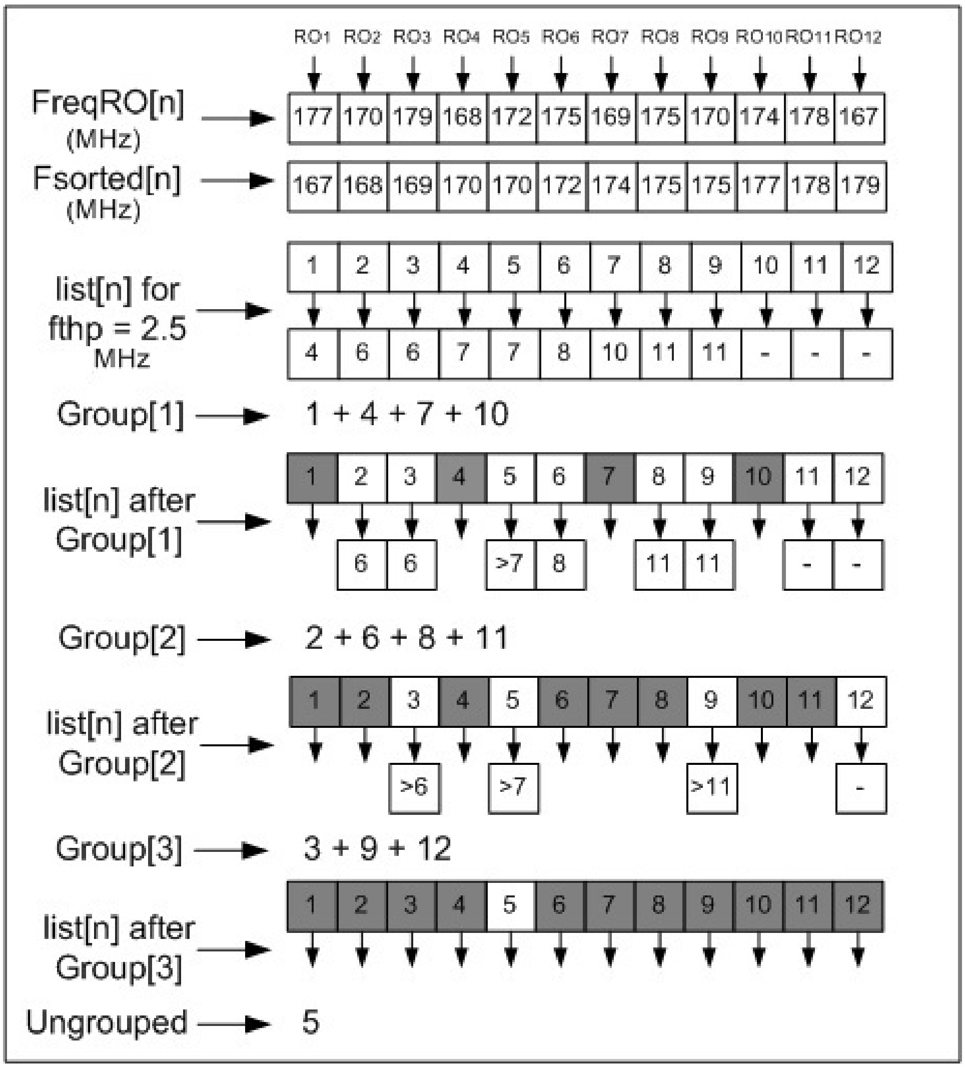


Figure 7.2. DP sample execution for 12 elements with an f_{thp} value of 2.5 MHz.

$$CRP_{num} = \frac{f_{thpmax} - f_{thpmin}}{f_{thpdif}} + 1. \tag{7.1}$$

To verify the effectiveness of the f_{thp} selection method, example given in Chapter 5 is repeated by choosing a different f_{thp} value. As shown in Figure 7.2, the same RO set is grouped with an f_{thp} value of 2.5 MHz instead of 1.5 MHz using DP. As expected, the formed group contents changes drastically when the f_{thp} value is modified. This verifies the effectiveness of the f_{thp} selection method for enhancing the CRP set in ordering based RO-PUFs.

Analysis of the f_{thp} selection method is done by creating three random sets of ROs in Matlab environment. For this purpose, 160 RO frequencies with Gaussian distribution are generated for three different RO structures with 5, 11, and 21 stage ROs. The mean frequency and the standard deviation of the three distributions are derived based on real data measured from FPGA implementations. Then, DP algorithm is applied to each set of 160 ROs by using various f_{thp} values based on the previously determined f_{thpmin}, f_{thpmax}, and f_{thpdif} parameters. In this analysis, the f_{thpmin} value is set to 1 MHz for the 5-stage RO structure, which is close to the ideal value according to the measurements presented in Chapter 5, 400 kHz and 200 kHz are chosen for the 11 and 21-stage structures, respectively. f_{thpmax} parameter is selected as 2 MHz, 1.4 MHz, and 1.2 MHz for the 5, 11, and 21-stage ROs, respectively, which provides a range of 1 MHz for possible f_{thp} values without increasing the area consumption drastically. In order to determine the optimum value for the f_{thpdif} parameter, the analysis is repeated for five different f_{thpdif} values, 10 kHz, 25 kHz, 50 kHz, 100 kHz, and 200 kHz. Starting from the f_{thpmin} value, PUF outputs are generated with DP algorithm for each possible f_{thp} value that are f_{thpdif} apart from each other, until the f_{thpmax} value is reached. For instance, 101 outputs are generated for an f_{thpdif} value of 10 kHz within the range of 1 MHz to 2 MHz for the 5-stage RO structure.

The uniqueness of the outputs are analyzed with the metrics defined in Chapter 3, U_QM1, U_QM2, and U_QM3, to determine the independence of CRPs, which is the most important quality factor for the proposed CRP enhancement methods.

Uniqueness analysis results of the f_{thp} selection method are shown in Figure 7.3. As can be seen from the figure, U_QM1 is close to the ideal value of 0.5 for the three structures, for all f_{thpdif} values investigated. In spite of this, U_QM2 is slightly lower than the ideal value for the 5-stage and 11-stage RO structures and it is significantly lower than the ideal value for the 21-stage RO structure. U_QM3 is very low for all the structures when an f_{thpdif} value of 10 kHz is used. For f_{thpdif} values of 25 kHz, 50 kHz, 100 kHz, and 200 kHz, the results are close to each other. Uniqueness results confirm the validity of the method. Since the results are similar for 25 kHz, 50 kHz, 100 kHz, and 200 kHz, using the smaller f_{thpdif} value is better for creating more CRPs

Table 7.1. f_{thp} Selection Method Area Consumption vs. CRP count.

Largest f_{thp}	Required RO Number	Area Increase (%)	CRP Count
1	88	0	1
1.2	100	13	9
1.4	110	25	17
1.6	119	35	25
1.8	132	50	33
2	145	64	41

with minimum area, time, and power consumption. In addition to this, all analyzed RO structures with different number of stages exhibit acceptable levels of uniqueness. Therefore, using the one with 5-stages is the best choice for area, time, and power efficiency of the system.

Next, the area efficiency of the method is analyzed using the 5-stage RO structure. For this purpose, area overhead vs. CRP count is evaluated by using an f_{thpdif} value of 25 kHz and f_{thpmin} value of 1 MHz, which seem to be the optimum values according to the analysis described above for the specific PUF design, FPGA type, and technology node. Area requirement of the system is measured for an output length of 128 bits using different f_{thpmax} values. As can be seen from Table 7.1, the area overhead is limited to 35% and 64% for supplying 25 and 41 CRPs, respectively.

7.3. Enhanced CRP Set with RO Selection Method

Another possible approach to enhance the CRP set is to change the RO frequencies that are used by the DP algorithm to generate the PUF outputs. This can be done by implementing more ROs than the minimum required number and selecting a subset of these ROs according to the applied challenge to apply the DP based grouping method. Let RO_{min} be the minimum number of ROs to generate output with a certain bit length and f_{thp} value, and let RO_{imp} be the number of implemented ROs

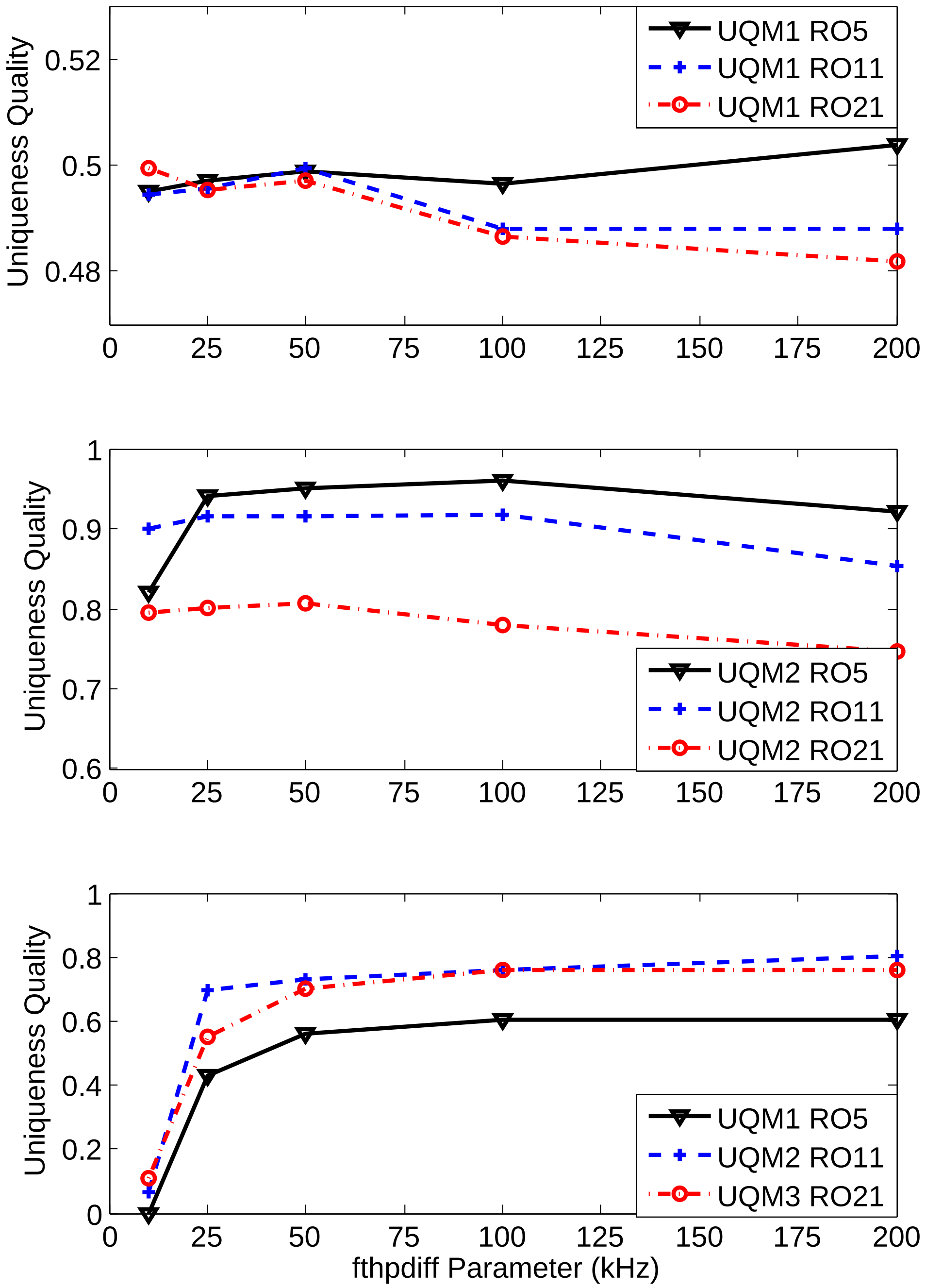


Figure 7.3. Uniqueness Quality vs. f_{thpdif} Parameter.

```
Data:
1. $RO_{min}$ number of random RO selection information applied as challenge.
2. $f_{thp}$ for robustness
Choose $RO_{min}$ number of ROs with their frequencies measured under nominal operating conditions and form a linked list, $FreqRO[n]$.
Result: Groups of ROs.
```

Figure 7.4. Revised part of the DP algorithm in pseudo code.

in the system. In this case, the first part of the DP algorithm until the sorting step is updated as presented in Figure 7.4. With this update, DP algorithm selects the RO_{min} number of ROs out of RO_{imp} number of ROs that are already implemented in the system depending on the applied challenge. When the selection is completed, the algorithm remains unchanged starting from the sorting step.

In this case, the number of different RO sets composed of RO_{min} out of RO_{imp} ROs can be calculated as

$$C(RO_{imp}, RO_{min}) = \frac{RO_{imp}!}{RO_{min}!(RO_{imp} - RO_{min})!}, \tag{7.2}$$

which increases factorially with RO_{imp} when the RO_{min} is constant. In this method, RO_{min} number of selected RO identities act as a challenge and the PUF output acts as the response. Since the possible number of RO subsets increases factorially, the number of possible CRPs increases similarly, which is a desired property that is not presented for RO-PUFs previously.

The drawback of this method is the possible similarity of the groups after the DP algorithm is applied. This may occur if most of the ROs within the two subsets are the same. The proposed solution for this problem is to generate more RO subsets than the system's CRP requirement and use the sets that are quite different from each other as challenges. In addition to this, if the number of possible sets is much bigger than the CRP requirement, random selection of the challenges will have a small probability of

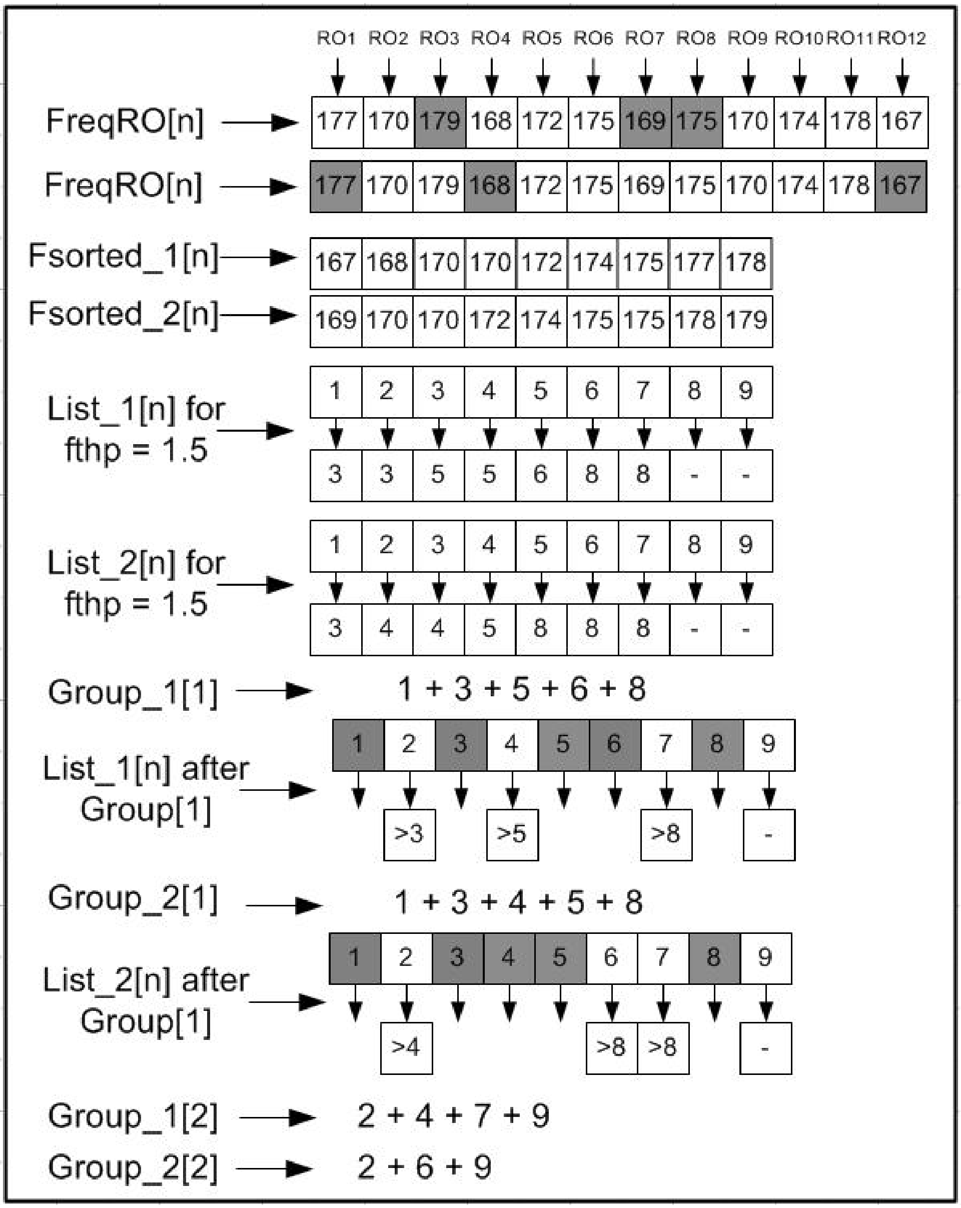


Figure 7.5. DP algorithm sample execution for RO selection method.

creating similar outputs. This situation is analyzed with real implementation results at the end of this section.

To verify the effectiveness of the RO selection method, example given in Chapter 5 is repeated for two RO subsets composed of 9 ROs each, that are selected from the whole RO set of 12 ROs. An f_{thp} value of 1.5 MHz is used for the analysis again. As can be seen in Figure 7.5, two different RO selections resulted in different groupings when the DP algorithm is applied. This verifies the effectiveness of the RO selection method for enhancing the CRP set in ordering based RO-PUFs.

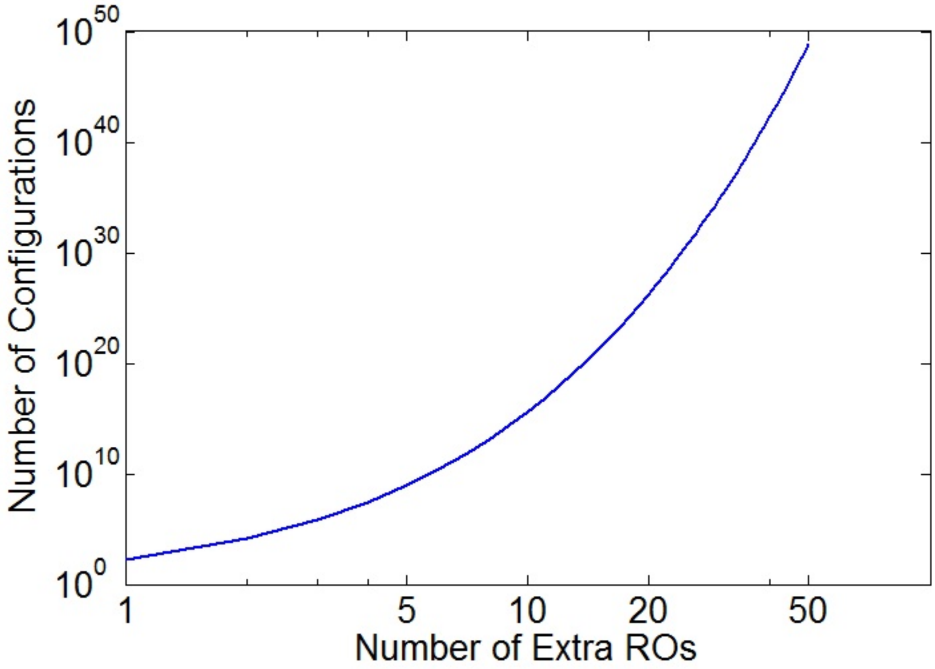


Figure 7.6. Number of CRPs vs. additional ROs.

CRP generation capability of the RO selection method is illustrated in Figure 7.6 by calculating the number of possible configurations for each additional RO within an RO-PUF of 160 ROs. The number of possible CRPs increases factorially with the number of additional ROs. Even adding 5 ROs results in more than 10^{10} different combinations and this number exceeds 10^{50} when 50 ROs are added to the system.

Analysis of the RO selection method is performed by creating 10 different RO sets composed of 165 to 210 ROs for 3 different RO structures composed of 5, 11, and 21 stages in Matlab environment, similar to the manner described in the previous section. Then, 10,000 subsets from each of these sets are created by selecting 160 ROs randomly, and DP algorithm is applied to the subsets to generate 128 bit long outputs. As a result, 10,000 PUF outputs for each RO set are generated. Then, U_QM1, U_QM2, and U_QM3, are used to analyze the uniqueness of the outputs. As can be seen in Figure 7.7, uniqueness of the outputs increases with increasing number of additional ROs for all three structures. U_QM1 reaches 0.95 and U_QM2 reaches 0.45 by adding only 20 ROs to the system, which are quite close to the targeted values of 1.00 and 0.50, respectively. U_QM3 shows that adding fewer than 20 ROs may result in generating identical responses to different challenges within a set of 10,000

CRPs.

When the results of the two proposed CRP enhancement methods are compared, it is observed that U_QM3 is significantly lower in the RO selection method. This indicates that some output pairs generated with the RO selection method are closer to each other than the ones generated with the f_{thp} selection method. The probability of output pairs that have more than a certain level of HD may be beneficial from a system designer's perspective. For this purpose, the probability of output pairs with an HD of fewer than 10 to 50 bits is calculated for each RO set of the 5-stage structure and presented in Table 7.2. As the number of additional ROs increases, the probability of output pairs with low HD decreases as expected. For instance, when a total of 210 ROs are implemented, none of the output pairs within the 10,000 outputs have an HD of less than or equal to 20 bits. Moreover, almost 98% of output pairs have an HD of more than 50 bits.

Another analysis performed on the proposed method is determining the minimum HD within the output pairs. This is meaningful if the target system has a minimum HD requirement within the responses. Hence, the system designer can choose the optimum number of ROs that should be implemented in order to simultaneously maintain the required quality of the outputs and the minimum area consumption. Results of the indicated analysis and the area overhead of the RO selection method are presented in Table 7.3.

7.4. Comparison of the f_{thp} Selection and RO Selection Methods

Both the f_{thp} selection and RO selection methods provide CRP support to ordering based RO-PUFs. However, they have different behaviors in terms of CRP count and quality, area, time, and power efficiency. Therefore, a comprehensive comparison of the proposed methods is presented in this section. For this purpose, an ordering based RO-PUF without CRP support and two ordering based RO-PUFs with CRP support that are based on f_{thp} selection and RO selection methods are compared and the results are presented in Table 7.4. In this comparison, f_{thp} selection method with

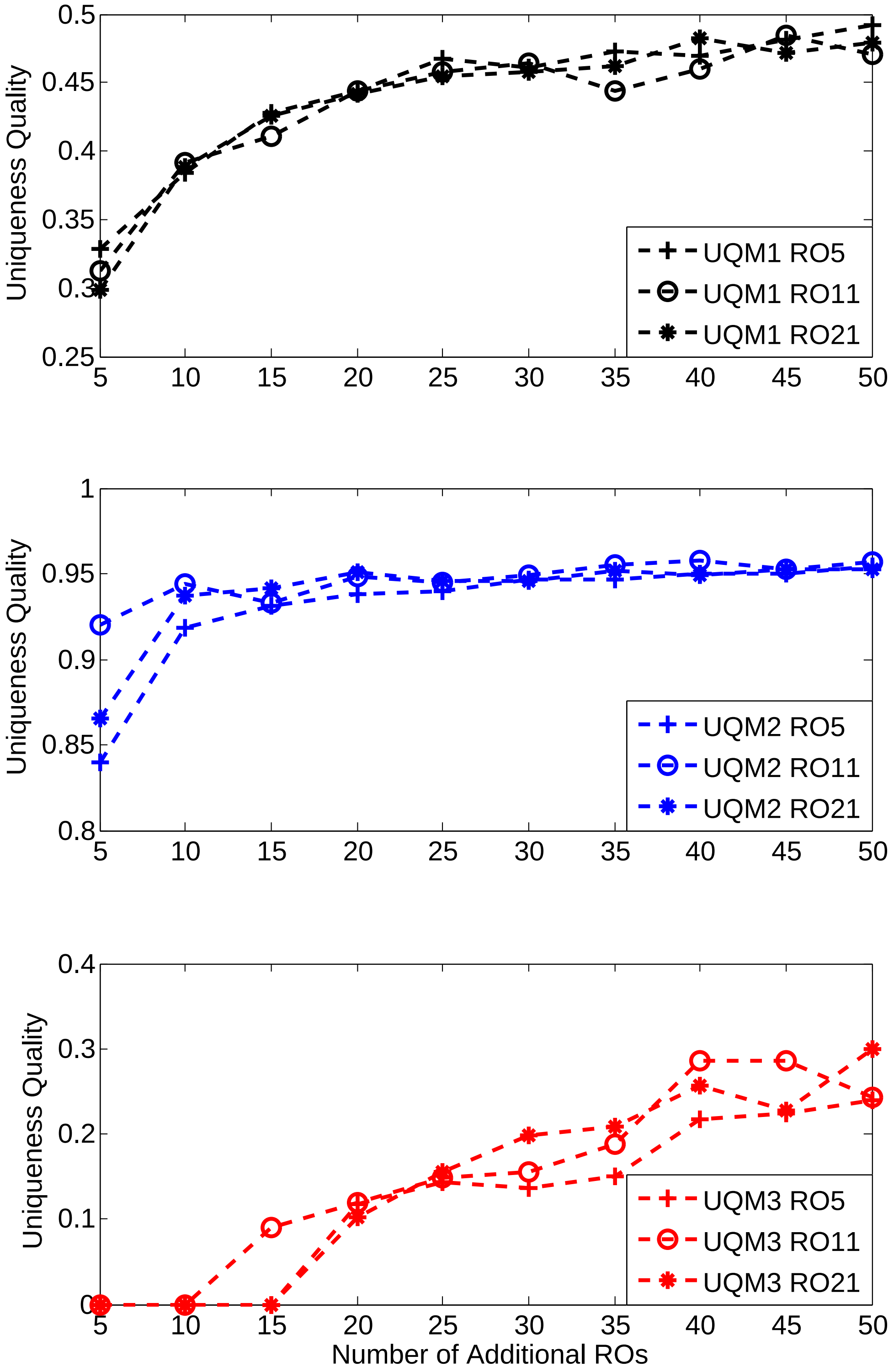


Figure 7.7. Uniqueness quality vs. additional ROs.

Table 7.2. Probability of Output Couples with HD Less than the Minimum HD Defined.

Number of ROs	Min HD ≤ 10	Min HD ≤ 20	Min HD ≤ 30	Min HD ≤ 40	Min HD ≤ 50
165	4.00E-02	1.40E-01	2.70E-01	4.11E-01	6.06E-01
170	3.80E-03	2.50E-02	8.40E-02	2.20E-01	4.80E-01
175	2.90E-04	3.70E-03	2.20E-02	9.80E-02	3.10E-01
180	8.30E-06	4.80E-04	6.80E-03	4.70E-02	2.10E-01
185	6.40E-07	4.70E-05	1.00E-03	1.20E-02	1.00E-01
190	8.00E-08	2.30E-05	8.10E-04	1.30E-02	1.27E-01
195	6.00E-08	6.60E-06	2.95E-04	6.60E-03	8.00E-02
200	0	5.00E-07	7.60E-05	4.00E-03	7.80E-02
205	0	4.40E-07	5.10E-05	2.30E-03	4.90E-02
210	0	0	7.44E-06	5.80E-04	2.30E-02

41 CRP generation capability and RO selection method with 10^{50} CRP generation capability are compared. One of the main performance parameters of PUF structures is area consumption. As can be seen from the table, ordering based RO-PUF without CRP support requires 88 ROs, whereas CRP supporting structures with f_{thp} selection and RO selection require 145 and 210 ROs, respectively. Another important performance parameter is output generation time, which is directly proportional to the number of ROs to be measured during the output generation and measurement time for each RO. 81 μs is used for the measurement time of each RO, since it is the optimum measurement time according to the analysis presented in Chapter 4. As can be seen from the table, ordering based RO-PUF without CRP support requires 7.2 ms to measure 88 ROs, whereas CRP supporting structures with f_{thp} selection and RO selection require 11.9 ms and 13.1 ms to measure 145 and 160 ROs, respectively. Even though the structure with RO selection method has 210 ROs implemented, only 160 of them will be measured, since the DP algorithm will only use the frequencies of the selected ROs. Power consumption of the RO structures are also directly proportional to the number of ROs to be measured; hence, output generation time. Uniqueness anal-

Table 7.3. Minimum HD Among 128 bit Outputs within 10,000 CRPs Based on the Number of ROs.

Number of ROs	Min HD among 128 bit output	Area Overhead
165	0	0.031
170	0	0.063
175	0	0.094
180	2	0.125
185	6	0.156
190	10	0.188
195	11	0.219
200	15	0.250
205	18	0.281
210	22	0.313

ysis of the CRPs are also presented for the proposed methods for the three metrics proposed in this work.

Even though the number of CRPs provided with the f_{thp} selection method is limited, uniqueness quality of the outputs is very good and the area overhead of the system is reasonable. This method is very convenient especially for applications that use PUF for secret key generation that require highly unique outputs and reconfiguration of the keys for a limited number of times within the lifetime of the IC.

The main advantage of the RO selection method is its capability of generating highly unique and very large number of CRPs with an acceptable area overhead. The method is also very flexible and can be customized for the desired number of CRPs and uniqueness of the outputs, using the analysis presented above. RO selection method is very convenient especially for applications that use PUF for authentication purposes.

As discussed in this section, f_{thp} selection and RO selection methods have their own advantages and disadvantages. In order to achieve a better PUF design that

Table 7.4. Comparison of Ordering Based RO-PUFs in terms of Response Uniqueness.

	Ordering Based **RO-PUF** **CRP non-supported**	**Ordering Based** **RO-PUF** f_{thp} **Selection**	**Ordering Based** **RO-PUF** **RO Selection**
CRP Count	1	41	10^{50}
Area (RO Count)	88	145	210
Output Gen. Time (ms)	7.2	11.9	13.1
U_QM1	N/A	0.4972	0.4925
U_QM2	N/A	0.953	0.954
U_QM3	N/A	0.46	0.2387

is suitable for a bigger variety of applications, it is possible to combine both of these methods. In this case, both f_{thp} selection and RO selection information will be applied as challenges to the DP algorithm. With this approach, higher uniqueness levels and CRP count can be achieved by utilizing the features of the f_{thp} selection and RO selection methods. However, using both of the methods at the same structure does not increase the total number of CRPs available, since the CRP set of the RO selection method is a superset of the CRP set of the f_{thp} selection method. This means that the groups formed using different f_{thp} values can also be formed via RO selection method as well. Therefore, the advantage of combining the methods will be the ability of achieving higher uniqueness levels by applying different f_{thp} values when required.

8. EFFECTS OF AGING AND COMPENSATION MECHANISMS

8.1. Aging Mechanisms and PUFs

Electrical characteristics of ICs change gradually with continuous use and may result in faulty behavior, causing reliability issues. The changes that appear due to aging are irreversible and affect the IC through the end of its lifetime. The downscaling of the physical dimensions of ICs with respectively high supply voltages makes circuits more prune to aging effects and reliability becomes a serious concern. Negative-Bias Temperature Instability (NBTI), Hot Carrier Injection (HCI), Temperature-Dependent Dielectric Breakdown (TDDB), Positive-Bias Temperature Instability (PBTI), electromigration, and soft errors are the main mechanisms that lead to the aging phenomenon. Among these, NBTI and HCI are considered as the dominant ones [91].

NBTI is the result of negative bias applied at the gate of the PMOS device. The negative bias generates interface traps by causing holes in the channel to dissociate the Si-H bonds. These interface traps increase the threshold voltage of the transistor and decrease the performance of the device by making the switching difficult. High temperature and high supply voltage enhance the effect of the NBTI [24, 92].

HCI is a phenomenon that occurs due to high energy carriers, which collide with the gate oxide layer and get trapped. As a result of this effect, the oxide layer is damaged and the threshold voltage is shifted, which causes performance degradation over a period of time. HCI affects both PMOS and NMOS devices. High supply voltage and switching rates increase the effect of the HCI [93].

Electromigration is the movement of metal atoms due to high currents flowing through the interconnects. Transport of metal atoms lead to void and hillock formations, which change the resistance of wires and may result in open or short circuits. Main causes of electromigration are the direct application of the electric field on the charged metal atoms and the frictional force between the metal atoms and the flowing electrons. High temperature enhances the effect of the electromigration.

Soft errors usually occur due to the high energy particles that enter the substrate near the drain of a transistor. These particles may cause many electron-hole pair formations. The formed electrons are collected by the drain node and decrease the voltage at that node. If the amount of electrons collected exceeds a certain amount, operation of the transistor fails [92].

TDDB is the gradual breakdown of the gate oxide. Dielectric breakdown happens due to the voltage applied across the gate oxide and conduction takes place utilizing the trapped charges. High temperature and high supply voltage accelerates the process of the TDDB [94].

Even though aging is an important concern for the reliability of all ICs, it becomes especially critical for PUF circuits, since their working principle is based on small mismatches present in the manufacturing process. As a result of this, PUF behavior may change drastically due to small aging related drifts in the electrical characteristics of the circuit before the failure of other parts. Limited amount of work is present on the aging of PUF circuits in the literature. Software based aging detection and CRP modification techniques are presented in [95]. Implementation results of the proposed protocol-level techniques are not presented in this work. In [29], an aging test of one month is applied to arbiter PUFs. However, since the test is performed under NOC, significant changes in the behavior of the PUF circuit could not be detected. Another aging test performed on SRAM PUFs under NOC is presented in [19]. In this work, it is stated that the change of initial values of SRAM cells remain under 4.5%. In [96], aging is used to develop a new kind of PUF structure, rather than detecting possible problems in previously presented PUF types. Six different PUF structures are analyzed with AAT in [97]. The results of the AAT applied to four memory based PUFs, one SRAM PUF, and one RO-PUF structure are presented. Analysis results indicate that aging decreases the robustness of PUF circuits significantly. [94, 98] focus on conventional RO-PUF structures and analyze the effects of aging by applying an AAT.

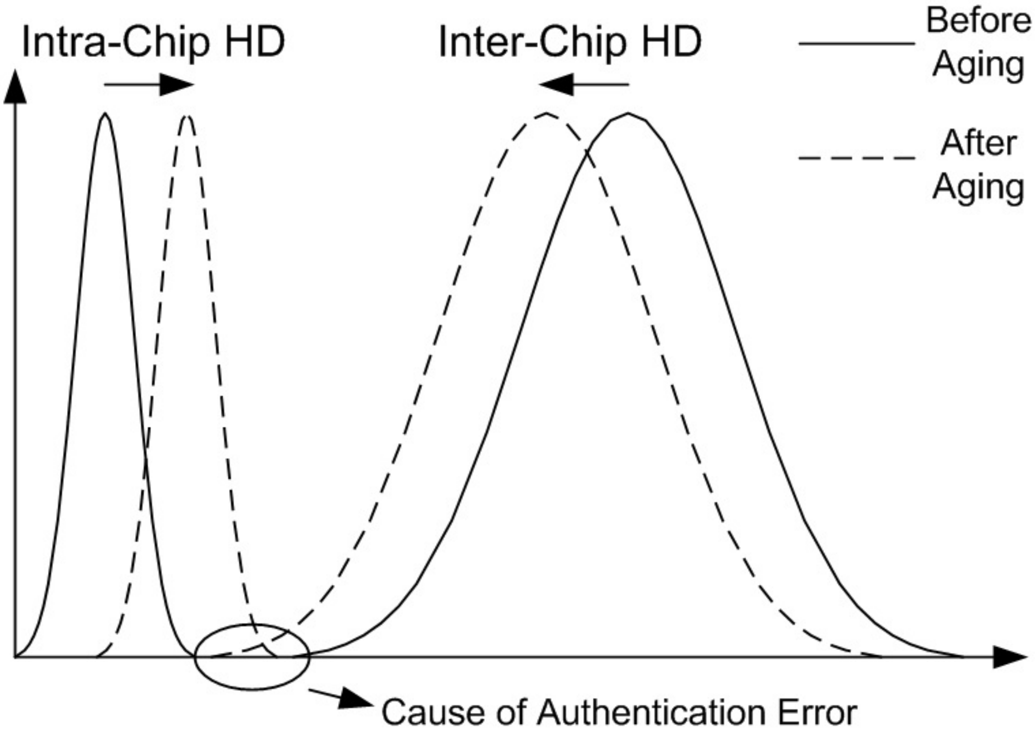


Figure 8.1. Effect of aging on conventional PUFs.

These works also propose a compensation technique against aging in conventional RO-PUFs. Finally, an aging-resistant RO-PUF structure is presented in [99], which aims to slow down the aging process using transistor level design techniques. However, due to the custom design requirement of ROs, proposed method is not applicable in FPGA environment.

Different from the works cited above, this study focuses on the effects of aging on ordering-based RO-PUFs via real implementation results. A mechanism to compensate aging effects in ordering-based RO-PUFs is also proposed. The cost of the compensation mechanism in terms of area efficiency is presented as well.

8.2. Ordering Based RO-PUFs and Aging

Aging decreases the uniqueness and robustness performances of the PUF structures [94]. This is illustrated in Figure 8.1. As can be seen from the figure, erroneous outputs increase and uniqueness of the outputs decreases. This behavior increases the authentication error probability.

Until the introduction of ordering based RO-PUFs in [9, 90], 100% robust PUF outputs were generated by adding ECC to the system [100]. Even though adding ECC is a reliable solution, it increases the area cost of the system drastically as discussed in Chapter 6. Ordering based RO-PUFs eliminate the need for ECC by introducing the f_{th} [9] or the f_{thp} parameters. The f_{thp} parameter defines the minimum frequency distance between ROs that are grouped together. It is the summation of noise in the system, $f_{th_{noise}}$, and environmental variations, $f_{th_{env}}$.

As long as the noise in the system and environmental variations affect the RO frequency differences within a group by frequency smaller than the f_{thp} value, reliable outputs are generated. However, effects of aging on the system have not been analyzed before. It is expected that combination of aging with the noise in the system and environmental variations may cause the ordering in the groups of ROs to change erroneously and the PUF to start generating noisy outputs. For ordering based RO-PUFs, expected aging effects on intra-PUF HD and inter-PUF HD are illustrated in Figure 8.2. As can seen from the figure, intra-PUF HD is a Dirac delta function located at *0* before aging, exhibiting ideal PUF behavior. When the circuit is aged, changes in ordering within the groups will lead to noisy outputs and intra-PUF HD will move away from *0*.

8.3. Accelerated Aging Test and Analysis of Results

Integrated circuits are expected to have a long life span that can be measured in years for consumer electronics and decades for military applications. Since it is impossible to test an IC for years in order to analyze the effects of aging, AAT is frequently employed to emulate the aging phenomenon. As explained in Section 8.1, contributors of aging, such as NBTI and HCI, are more effective on the IC under high temperature and/or high supply voltage. Thus, it is possible to accelerate the aging process by increasing the temperature and/or supply voltage of the IC under test.

In order to analyze the effects of aging on ordering based RO-PUFs, an unused Xilinx FPGA board is utilized. Since it was not possible to change the operating

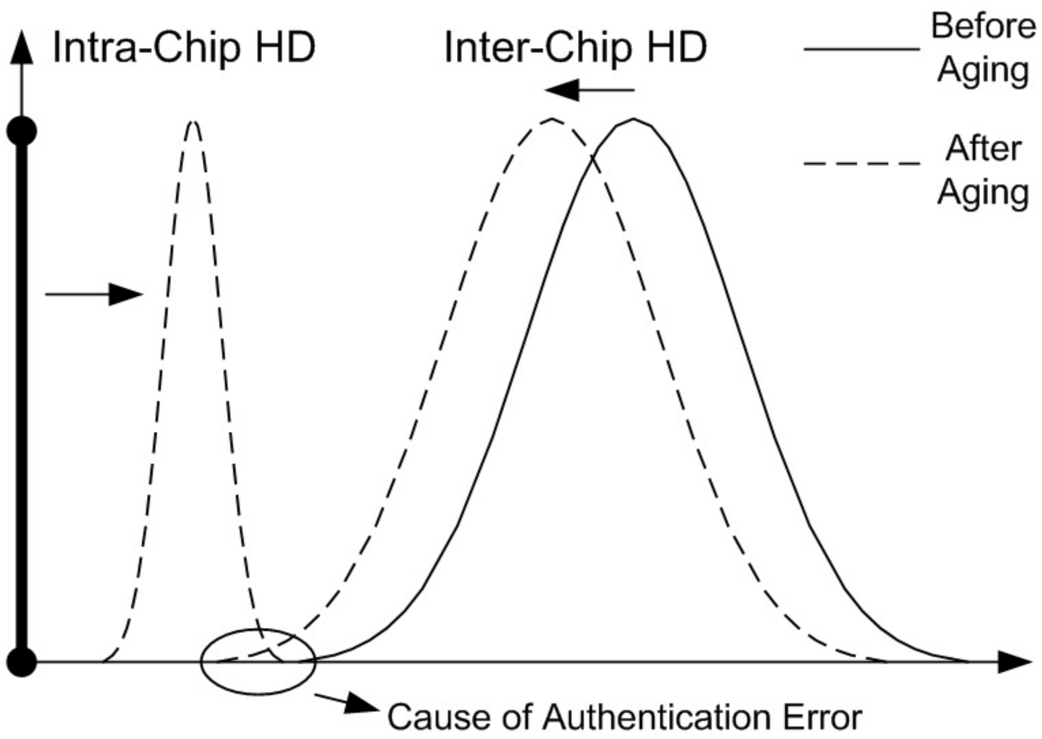


Figure 8.2. Effect of aging on ordering based RO-PUFs.

voltage of the FPGA due to the voltage regulators present in the system, only high temperature is used to accelerate the aging process. Acceleration factor of aging test under high temperature is called T_{factor} and can be calculated as

$$T_{factor} = e^{(E_a/k)\left(\frac{1}{T_{noc}} - \frac{1}{T_{aging}}\right)}, \tag{8.1}$$

where T_{noc} is the normal operating temperature of the circuit in Kelvin, T_{aging} is the temperature of the AAT in Kelvin, k is the Boltzman constant, and $E_a = 0.5\ eV$ is the activation energy [94]. In our case, the operating temperature is assumed as 25^oC and the test is applied at 100^oC. Under these circumstances, T_{factor} is calculated as 50.08. This means that one hour of AAT will correspond to 50.08 hours of operation under NOC for the IC.

As mentioned above, a Xilinx FPGA board is used for the AAT. For this pur-

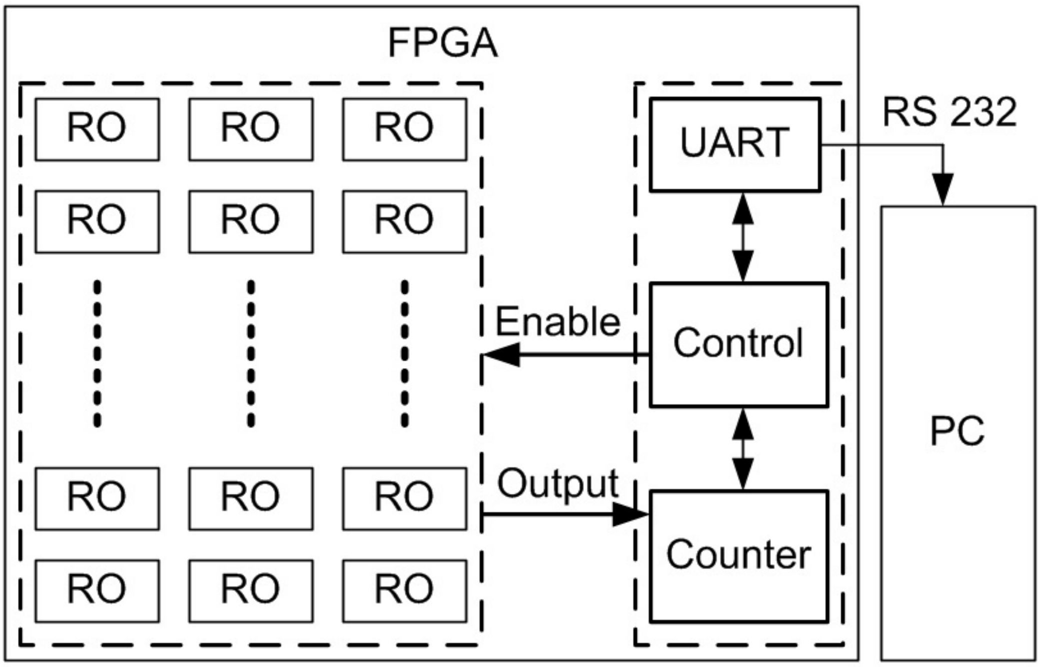


Figure 8.3. Accelerated aging test setup.

pose, 200 ROs composed of five inverting stages with enable inputs are implemented using the macro function. One counter is implemented to detect the number of oscillations within a certain amount of time. Universal Asynchronous Receiver/Transmitter (UART) is used to transfer the data from the FPGA to the PC via RS-232 serial bus. In this system, the controller unit enables all ROs at the same time during the AAT phase and collects and sends the oscillation counts to the PC by enabling ROs one-by-one during the measurement phase. The AAT setup is illustrated in Figure 8.3. The duration of the AAT is set to 500 hours, which is equivalent to approximately three years of operation under NOC. Equivalent IC working duration of the applied AAT for each 100 hours is given in Table 8.1.

Oscillation counts of all implemented ROs are recorded before the test and after each 25 hours of AAT, one measurement at 25^oC and other at 100^oC. Each measurement is repeated 50 times to determine the noise in the system and analyze the effects of aging on the noise. The average frequencies of ROs measured at 25^oC and 100^oC are presented in Figure 8.4. As can be seen from the figure, 500 hours of AAT decreases the average RO frequency by more than 1.3%. Deceleration rate of the ROs also decreases as the test duration increases. This is an expected behavior, since the

Table 8.1. Equivalent IC Working Duration of Accelerated Aging Test.

T_{Factor}	Test Duration (Hours)	Working Duration under NOC (Days)
50.08	100	208
50.08	200	417
50.08	300	626
50.08	400	834
50.08	500	1044

effect of NBTI decreases with aging as well.

Since the PUF output depends on the ordering of ROs within a group, relative frequency reduction of independent ROs are critical, rather than the average frequency change. If all of the ROs within a group slow down at the same rate, ordering does not change and the robustness of the system is not affected. However, if the slowing rates of ROs in a group differ from each other and the frequency ordering changes, erroneous outputs are generated. In order to determine the relative frequency change of ROs, deceleration of individual ROs are measured after each 100 hours of AAT. The distribution of ROs depending on frequency reduction rate is presented in Figure 8.5. From the analysis, it is seen that different ROs slow down at different rates. In addition to this, the distributions cover a wider range as the AAT progresses. For a more detailed analysis, the standard deviations of the distributions are calculated and presented in Figure 8.6. As can be seen from the figure, the increase in the standard deviations of the distributions slows down over time. This behavior indicates that the effects of aging on the relative RO frequency reduction diminishes after 300 hours of AAT and continuing the test after this point does not contribute much to the analysis. It can be concluded that, even though the measurements that are done correspond to 1044 days of aging, the robustness analysis applied based on this data will be valid for longer operation durations as well.

The noise in the environment is an important parameter for the robustness of PUF circuits and directly affects the determination of the f_{thp} parameter. Within the

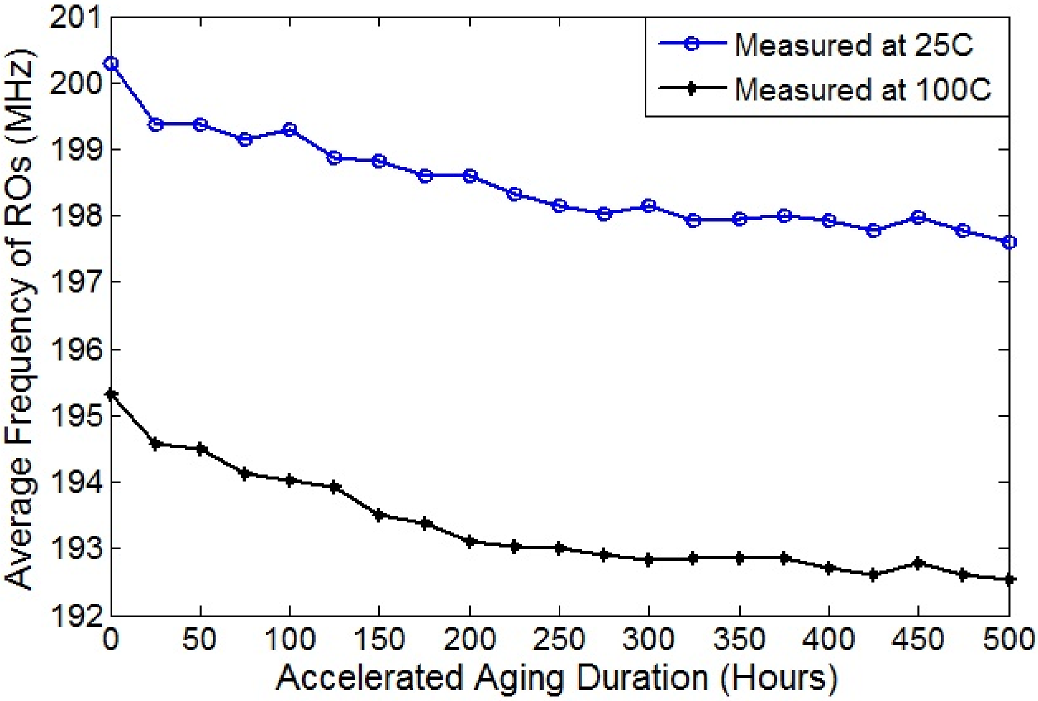


Figure 8.4. Mean frequency of ROs vs. accelerated aging test duration.

scope of this work, the effects of aging on the noise in the system are analyzed. For this purpose, each measurement collected from each RO is repeated 50 times and the standard deviations of these 50 measurements are calculated during different phases of the AAT at both 25^oC and 100^oC. Mean standard deviations of consecutively measured RO frequencies are presented in Figure 8.7. As can be seen from the figure, the noise level is not correlated with the duration of the AAT at neither 25^oC nor 100^oC and remains below a certain level at all times. As a result, the noise component in the f_{thp} parameter, $f_{th_{noise}}$, does not need to be updated due to aging in ordering based RO-PUF circuits.

The effect of RO locations on the aging process is also analyzed via calculating the frequency reduction rates of ROs that are located at 9 distinct sites on the FPGA from top-left to right bottom. As can be seen from Table 8.2, a clear correlation between the frequency reduction vs. RO locations does not exist. It can be concluded that power distribution and heat dissipation of the FPGA seem balanced; hence, each location on the IC indicates a similar behavior in terms of aging.

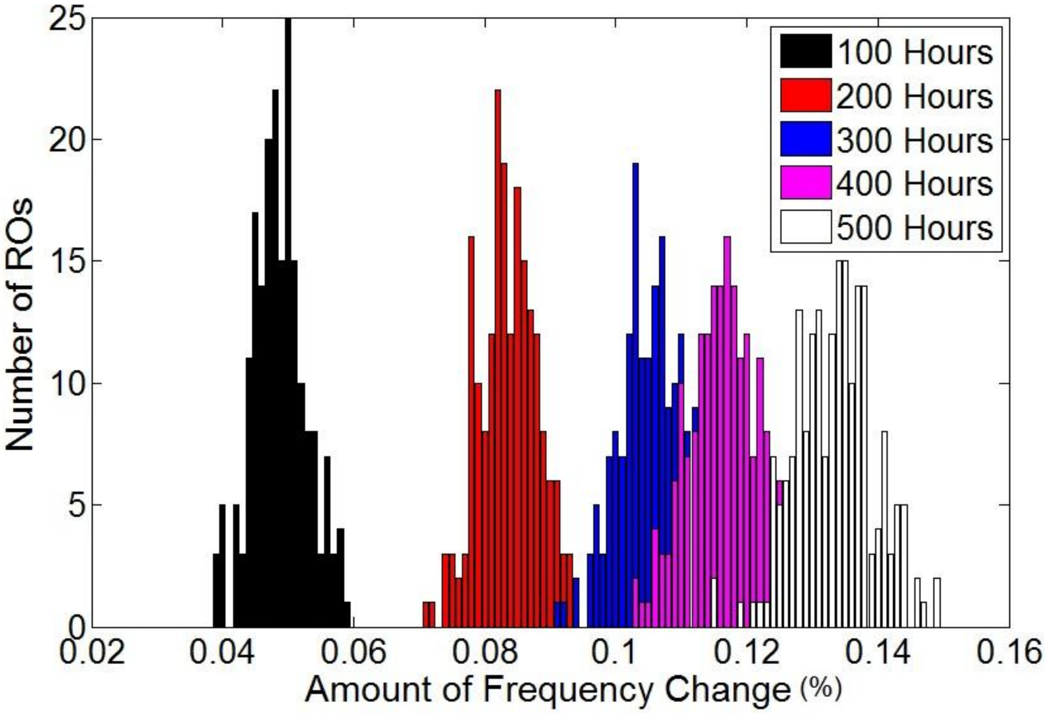


Figure 8.5. Distribution of ROs depending on frequency reduction due to aging.

Table 8.2. RO Frequency Reduction due to Aging vs. RO Location.

	Left	**Middle**	**Right**
Top	0.0134	0.0138	0.0125
Middle	0.0132	0.0140	0.0134
Bottom	0.0130	0.0134	0.0131

8.4. Effects of Aging on Ordering Based RO-PUFs and Compensation Mechanisms

According to the analysis presented based on the AAT applied, different ROs are affected differently from the irreversible changes that occur due to aging. Since the robustness of ordering based RO-PUFs depends on the f_{thp} parameter, different frequency reduction rates may cause frequency fluctuation values greater than the f_{thp} value and result in ordering changes within some groups. Each RO pair, which has frequency deviation more than the selected f_{thp} value is called a problematic RO pair. The probability of erroneous output generation due to problematic RO pairs is discussed in Chapter 6. An unintended change of ordering in a group is called a

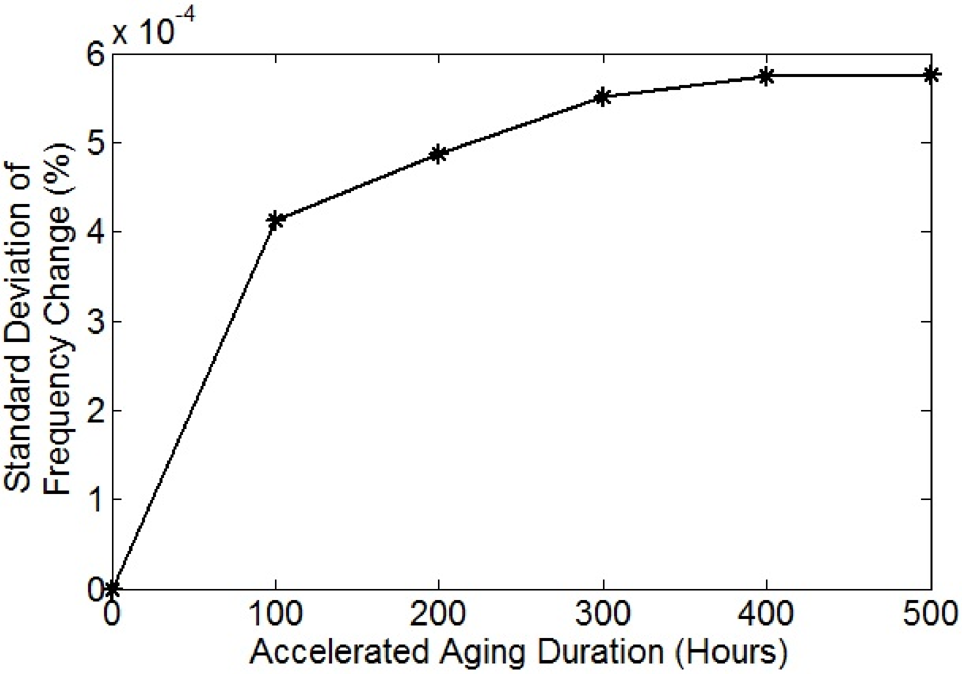


Figure 8.6. STD of frequency reduction due to aging.

symbol error and depends on three conditions. Initially, problematic RO pair should be placed in the same group and next to each other in frequency ordering. Secondly, ordering of the RO pair should be prone to error generation, i.e., if the slower RO slows down more than the faster RO due to environmental conditions, noise, or aging effects, ordering does not change and an error is not generated. The third condition is to have the worst case environmental conditions and noise in the system to trigger the symbol error by pushing the frequency change within a pair by more than the f_{thp} value. The probability of symbol error rate is then given as

$$p = \sum_{s=2}^{m} k_s * (s-1)/(M*(M-1)), \tag{8.2}$$

where M is the total number of ROs implemented, s is the group length, k_s is the average number of groups with size s, and m is the size of the largest group. Based on this formula, the symbol error rate for a system of 160 ROs that has a single problematic RO pair is given as 4.52×10^{-3} for an f_{thp} value of 1 MHz. The number

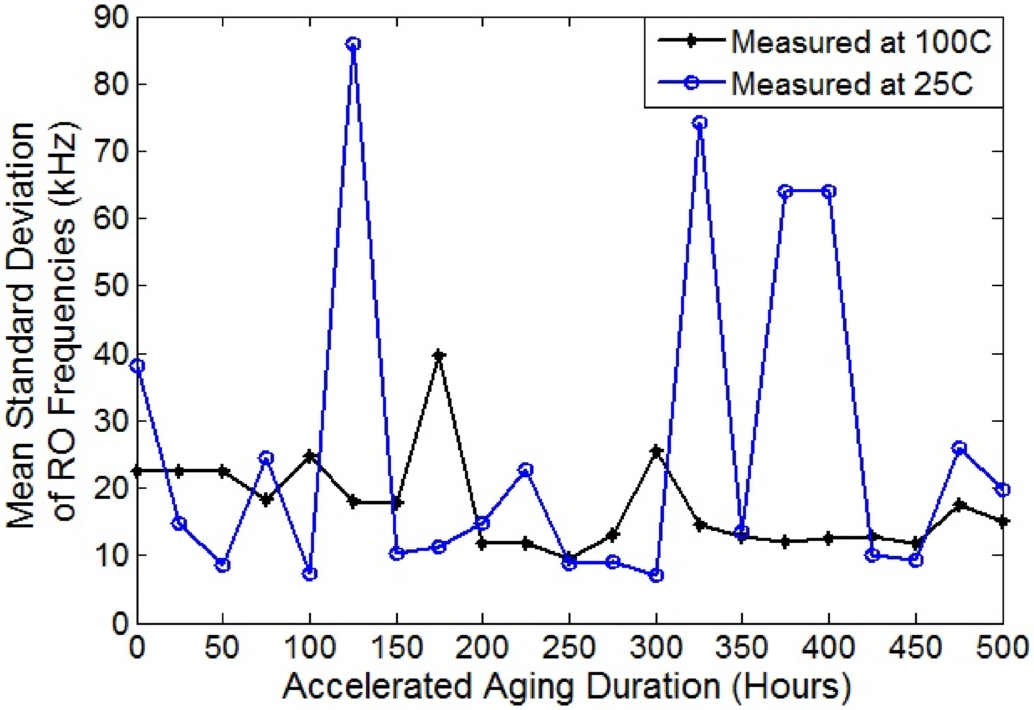


Figure 8.7. Noise on RO frequency measurements vs. aging.

Table 8.3. Number of Problematic RO Pairs.

f_{thp} **error (kHz)**	**50**	**100**	**150**
Num. of Problematic Pairs	2	5	15

of problematic RO pairs depends on the effect of aging on the f_{thp} value, which is called $f_{th_{aging}}$. $f_{th_{aging}}$ is basically the difference between the frequency changes of the most and least slowing ROs after the AAT. If the $f_{th_{aging}}$ value is not taken into account during the calculation of the f_{thp} value, higher values of $f_{th_{aging}}$ will result in a higher number of problematic RO pairs; hence, increasing the probability of the symbol errors. The effect of a wrong f_{thp} value on the number of problematic RO pairs is presented in Table 8.3. Based on this data, if an $f_{th_{aging}}$ value of 100 kHz is introduced during the lifetime of the system, five problematic RO pairs may be generated. In this case, the symbol error probability of the overall system can be calculated as $5 \times 4.52 \times 10^{-3} = 2.26 \times 10^{-2}$. This error probability can be acceptable especially in identification and authentication systems.

Even though the symbol error probability introduced due to aging in ordering based RO-PUFs is low, 100% robustness is mandatory for systems that utilize PUF circuits as cryptographic key generators. For this purpose, a compensation method is proposed based on updating the f_{thp} value according to the expected effects of aging. In this method, the f_{thp} value is updated by adding $f_{th_{aging}}$ value as

$$f_{thp} = f_{th_{noise}} + f_{th_{env}} + f_{th_{aging}}. \tag{8.3}$$

Since DP algorithm will use a larger f_{thp} value due to aging compensation, frequency difference between the ROs in a group will be larger. This will prevent the system from unintended ordering changes that may be result a of aging and maintain the robustness of outputs.

Maximum frequency reduction difference ratio based on the duration of AAT is presented in Table 8.4. The frequency difference between the most and least slowing ROs is calculated as 0.2% after 100 hours and 0.34% after 500 hours of AAT. The corresponding $f_{th_{aging}}$ value is also calculated and presented based on the average oscillation frequency, 200 MHz. According to the presented data, if the system designer wants to secure the system up to three years of fulltime working duration, an $f_{th_{aging}}$ value of 680 kHz should be added to the f_{thp} value. Since increasing the f_{thp} value requires increasing the number of ROs implemented to maintain the generation of similar length outputs, area efficiency of the system is degraded. Using the frequencies collected from the FPGA under NOC prior to AAT, the number of ROs required for an ordering based RO-PUF that generates 128 bit length outputs is calculated for an $f_{th_{noise}} + f_{th_{env}}$ value of 1 MHz and different $f_{th_{aging}}$ values ranging from 0 to 680 kHz. As can be seen from Table 8.4, the minimum required number of ROs varies from 88 to 126, depending on the expected working duration. Depending on these values, an area overhead of up to 43% arises for guaranteed reliability. Even though an area overhead is introduced due to aging, an easily applicable aging compensation mechanism is an advantage of the ordering based RO-PUF structures, which will sustain their widespread use in security systems.

Table 8.4. Effect of Aging on f_{thp} Parameter and Corresponding Area Overhead.

Acc. Aging Duration (Hours)	0	100	200	300	400	500
Max. Freq Diff. Ratio (%)	0	0.002	0.0027	0.003	0.0028	0.0034
Corresponding $f_{thp_{aging}}$ effect	0 kHz	400 kHz	540 kHz	600 kHz	560 kHz	680 kHz
Required Number of RO's	88	110	119	119	119	126
Area Overhead (%)	0	25	35	35	35	43

9. CONCLUSION

PUFs have received an important amount of attention in the last decade and several security applications utilizing these primitives have emerged. However, due to the relatively short time since the concept was first introduced, many aspects of PUF circuits have not been fully investigated yet. In this thesis study, important aspects of PUFs have been studied.

In Chapter 2, information on PUFs is given based on a detailed literature survey. Basic principles of PUFs are defined and application areas that utilize PUFs are summarized. Next, uniqueness, robustness, unclonability, and unpredictability, which are the main characteristics of PUFs, are explained in detail. Final section of this chapter presents different PUF types proposed in the literature with their advantages and disadvantages.

Quality metrics for the evaluation of PUF circuits are developed and presented in Chapter 3. Using the quality metrics proposed, a fair performance comparison of PUF implementations became possible. In addition to this, confidence interval and confidence level concepts are used in PUF evaluation phase to maintain the reliability of the results. Then, two conventional RO-PUF circuits are implemented on FPGA and analyzed according to the proposed quality metrics.

Chapter 4 focuses on the theoretical foundations of ROs. Based on the mathematical formulations developed, optimization techniques for the number of stages and measurement time of ROs are proposed. Then, RO-PUFs composed of ROs with different number of stages are built on FPGA and outputs of each RO-PUF are collected using a set of different measurement times. Next, proposed optimization techniques are validated by analyzing the collected outputs.

Ordering based RO-PUFs, which aim maximizing the robustness and entropy extraction, are discussed in Chapter 5. DP is adapted to ordering based RO-PUFs for lowering the complexity of the grouping algorithm proposed in the literature. Finally, area advantage of ordering based RO-PUFs over conventional RO-PUFs and timing advantage of DP over LISA are presented based on experimental results.

Error probability of ordering based RO-PUFs are analyzed in Chapter 6. In the first section, group length analysis is performed, symbol error and bit error probabilities are presented, and worst case bit errors based on group sizes are analyzed. Then, area usage vs. robustness in ordering based RO-PUFs is analyzed. Finally, ECC are introduced and implementation results of the BCH codes for various output lengths and error correction capabilities are presented.

CRP enhancement methods for ordering based RO-PUFs are presented in Chapter 7. Initially, CRP properties and importance of large number of CRPs are discussed. Then, f_{thp} selection and RO selection methods, which provide CRP support to ordering based RO-PUFs, are introduced. These methods are compared in terms of CRP quality and number, area consumption, and timing efficiency.

Chapter 8 focuses on the effects of aging on the performance and behavior changes of ordering based RO-PUFs. Aging mechanisms are presented and the effects of aging on RO-PUFs are investigated through an AAT applied in FPGA environment. Finally, a compensation mechanism to maintain the reliability of ordering based RO-PUFs for many years of operation is proposed.

With the progress achieved in this thesis, designers will be able to calculate the optimum number of stages of ROs and measurement time theoretically depending on the system requirements and will achieve building high performance RO-PUF structures. Applications requiring 100% robustness, such as key generators, will be able utilize area, time, and power effective PUFs without the need of implementing ECCs. RO-PUFs will be able to utilized by authentication applications with the added high number of CRP support. Degrading the effect of aging on ordering based RO-PUFs will not threaten the functionality of the system. However, the proposed structures and methods are verified using limited type of FPGAs and are not implemented on

ASICs yet. In addition to these, the effects of shrinking of the device sizes with developing technology on RO-PUFs are not investigated as well. Thus, the thesis can be considered as a basis for future studies on RO-PUFs. Future works are planned as

(i) Investigating the effects of different place&route strategies of ROs within slices of the FPGA on RO-PUFs,

(ii) Investigating the effects of different RO topologies on RO-PUFs using ASICs,

(iii) Developing and analyzing aging resistant RO topologies using ASICs for long lifetime RO-PUFs,

(iv) Investigating the effects of shrinking of the device sizes with developing technology on RO-PUFs.

APPENDIX A: LIST OF PUBLICATIONS

In the course of Ph.D. studies, following journal and conference papers have been published.

Chapter 3

- Komurcu G. and G. Dundar, ”Determining the Quality Metrics for PUFs and Performance Evaluation of Two RO-PUFs”, *IEEE 10th International New Circuits and Systems Conference, (NEWCAS)*, pp. 73-76, 2012.

- Komurcu G. and A. E. Pusane and G. Dundar, ”FPGA Üzerinde Ring Osilatörü Tabanlı PUF Gerçeklemesi”, *Elektrik - Elektronik, Bilgisayar ve Biyomedikal Mühendisliği Sempozyumu, (ELECO)*, 2012.

- Komurcu G. and A. E. Pusane and G. Dundar, ”A Ring Oscillator Based PUF Implementation on FPGA”, *IU-Journal of Electrical and Electronics Engineering, (IU-JEEE)*, vol.13, no. 2, pp. 1647-1652, 2012.

Chapter 4

- Komurcu G. and A. E. Pusane and G. Dundar, ”Analysis of Ring Oscillator Structures to Develop a Design Methodology for RO-PUF Circuits”, *IEEE 21st International Conference on Very Large Scale Integration (VLSI-SoC)*, pp. 332-335, 2013.

Chapter 5

- Komurcu G. and A. E. Pusane and G. Dundar, ”Dynamic Programming Based Grouping Method for RO-PUFs”, *9th Conference on Ph. D. Research in Microelectronics and Electronics (PRIME)*, pp. 329-332, 2013.

Chapter 6

- Komurcu G. and A. E. Pusane and G. Dundar, ”An Efficient Grouping Method and Error Probability Analysis for RO-PUFs”, Submitted to *Journal of Cryptology*, 2014.

Chapter 7

- Komurcu G. and A. E. Pusane and G. Dundar, ”Robust RO-PUFs with Enhanced Challenge-Response Set”, *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology Conference (ECTI-CON)*, 2014.

- Komurcu G. and A. E. Pusane and G. Dundar, ”Enhanced Challenge-Response Set and Secure Usage Scenarios for Ordering Based RO-PUFs”, *IET-Circuits, Devices, and Systems, (IET-CDS)*, accepted for publication, 2014.

Chapter 8

- Komurcu G. and A. E. Pusane and G. Dundar, ”Effects of Aging and Compensation Mechanisms in Ordering Based RO-PUFs”, Submitted to *IET-Information Security, (IET-IFS)*, 2014.

REFERENCES

1. Suh, G. E. and S. Devadas, “Physical Unclonable Functions for Device Authentication and Secret Key Generation”, *Design Automation Conference (DAC)*, pp. 9–14, 2007.

2. Guajardo, J., S. Kumar, G.-J. Schrijen and P. Tuyls, “Brand and IP Protection with Physical Unclonable Functions”, *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 3186–3189, 2008.

3. Tuyls, P., G. J. Shrijen, B. Skoric, J. V. Geloven, N. Verhaegh and R. Walters, “Read Proof Hardware from Protective Coatings”, *18th Annual Computer Security Applications Conference (CHES)*, Vol. 4249, pp. 369–383, 2006.

4. Majzoobi, M. and F. Koushanfar, “Techniques for Design and Implementation of Secure Reconfigurable PUFs”, *ACM Transactions on Reconfigurable Technology and Systems*, Vol. 2, No. 1, 2009.

5. Guajardo, J., S. Kumar, R. Maes, G. Schrijen and P. Tuyls, “Extended Abstract: The Butterfly PUF Protecting IP on every FPGA”, *Hardware-Oriented Security and Trust (HOST)*, pp. 67–70, 2008.

6. Anderson, J., “A PUF Design for Secure FPGA-Based Embedded Systems”, *15th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 1–6, 2010.

7. Maiti, A. and P. Schaumont, “Improving the Quality of a Physical Unclonable Function Using Configurable Ring Oscillators”, *International Conference on Field Programmable Logic and Applications (FPL)*, pp. 703–707, 2009.

8. Yu, H., P. Leong, H. Hinkelmann, L. Moller, M. Glesner and P. Zipf, “Towards a Unique FPGA-Based Identification Circuit Using Process Variations”, *Inter-*

national Conference on Field Programmable Logic and Applications (FPL), pp. 397–402, 2009.

9. Yin, C. and G. Qu, “LISA: Maximizing RO-PUF’s Secret Extraction”, *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pp. 100–105, 2010.

10. Yin, C. and G. Qu, “Temperature Aware Cooperative Ring Oscillator PUF”, *IEEE International Workshop on Hardware Oriented Security and Trust (HOST)*, pp. 36–42, 2009.

11. Pappu, R. S., *Physical One-Way Functions*, Ph.D. dissertation, Massachusetts Institute of Technology, Massachusetts, 2001.

12. Pappu, R. S., B. Recht, J. Taylor and N. Gershenfeld, *Physical One-Way Functions*, Science, Vol. 297, No. 6, pp. 2026-2030, 2002.

13. Gassend, B., D. Clarke, M. van Dijk and S. Devadas, “Silicon Pysical Random Functions”, *ACM Conference on Computer and Communications Security (CCS)*, pp. 148–160, 2002.

14. Morozov, S., A. Maiti and P. Schaumont, “An Analysis of Delay Based PUF Implementations on FPGA”, *Reconfigurable Computing: Architectures, Tools and Applications*, Vol. 5992, pp. 382–387, Springer Berlin Heidelberg, 2010.

15. Verbauwhede, I. and R. Maes, “Physically Unclonable Functions: Manufacturing Variability as an Unclonable Device Identifier.”, *ACM Great Lakes Symposium on VLSI*, pp. 455–460, ACM, 2011.

16. Beckmann, N. and M. Potkonjak, “Hardware-Based Public-Key Cryptography with Public Physically Unclonable Functions”, *Information Hiding*, Vol. 5806 of *Lecture Notes in Computer Science*, pp. 206–220, Springer Berlin Heidelberg, 2009.

17. Maes, R. and P. Tuyls, “Process Variations for Security: PUFs”, *Secure Integrated Circuits and Systems*, pp. 125–141, Springer US, 2010.

18. D.E.Lazich and M.Wuensche, “Protection of Sensitive Security Parameters in Integrated Circuits”, *LNCS*, Vol. 5393, No. 393, pp. 157–178, 2008.

19. Guajardo, J., S. Kumar, G. Schrijen and P. Tuyls, “FPGA Intrinsic PUFs and Their Use for IP Protection”, *18th Annual Computer Security Applications Conference (CHES)*, Vol. 4727, pp. 63–80, 2007.

20. Guajardo, J., S. Kumar, G.-J. Schrijen and P. Tuyls, “Physical Unclonable Functions and Public-Key Crypto for FPGA IP Protection”, *International Conference on Field Programmable Logic and Applications (FPL)*, pp. 189–195, 2007.

21. Devadas, S., E. Suh, S. Paral, R. R. Sowell, T. Ziola and V. Khandelwal, “Design and Implementation of PUF-Based ”Unclonable” RFID ICs for Anti-Counterfeiting and Security Applications”, *IEEE International Conference on RFID*, pp. 58–64, 2008.

22. Asim, M., J. Guajardo, S. Kumar and P. Tuyls, “Physical Unclonable Functions and Their Applications to Vehicle System Security”, *IEEE 69th Vehicular Technology Conference (VTC)*, pp. 1–5, 2009.

23. Hammouri, G., K. Akdemir and B. Sunar, “Novel PUF-Based Error Detection Methods in Finite State Machines”, *Information Security and Cryptology (ICISC)*, Vol. 5461 of *Lecture Notes in Computer Science*, pp. 235–252, Springer Berlin Heidelberg, 2009.

24. Wang, X. and M. Tehranipoor, “Novel Physical Unclonable Function with Process and Environmental Variations”, *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pp. 1065–1070, 2010.

25. Dodis, Y., L. Reyzin and A. Smith, “Fuzzy Extractors: How to Generate Strong

Keys from Biometrics and Other Noisy Data", *Advances in Cryptology - EUROCRYPT 2004*, Vol. 3027 of *Lecture Notes in Computer Science*, pp. 523–540, Springer Berlin Heidelberg, 2004.

26. Kursawe, K., A. Sadeghi, D. Schellekens, B. Skoric and P. Tuyls, "Reconfigurable Physical Unclonable Functions - Enabling Technology for Tamper-Resistant Storage", *IEEE International Workshop on Hardware Oriented Security and Trust (HOST)*, pp. 22–29, 2009.

27. Gassend, B., D. Clarke, M. V. Dijk and S. Devadas, "Delay-Based Circuit Authentication and Applications", *ACM Symposium on Applied Computing*, pp. 294–301, 2003.

28. Lee, J. W., D. Lim, B. Gassend, G. E. Suh, M. Dijk and S. Devadas, "A Technique to Build a Secret Key in Integrated Circuits for Identification and Authentication Applications", *Symposium On VLSl Circuits Digest of Technical Papers*, 2004.

29. Lim, D., J. Lee, B. Gasend, G.E.Suh, M. V. Dijk and S. Devadas, "Extracting Secret Keys from Integrated Circuits", *IEEE Transactions on VLSI Systems*, Vol. 13, No. 10, pp. 1200–1205, 2005.

30. Gassend, B., D. Clarke, M. V. Dijk, S. Devadas and D. Lim, "Identification and Authentication of Integrated Circuits", *Concurrency and Computation: Practice and Experience*, Vol. 16, No. 11, pp. 1077–1098, 2004.

31. Ruhrmair, U. and J. Solter, "PUF Modeling Attacks: An Introduction and Overview", *Design Test and Automation in Europe (DATE)*, 2014.

32. Xu, X. and W. Burleson, "Hybrid Side-Channel/Machine-Learning Attacks on PUFs: A New Threat?", *Design Test and Automation in Europe (DATE)*, 2014.

33. Ozturk, E., G. Hammouri and B. Sunar, "Physical Unclonable Function with Tristate Buffers", *IEEE International Symposium on Circuits and Systems (IS-*

CAS), pp. 3194–3197, 2008.

34. Majzoobi, M., F. Koushanfar and M. Potkonjak, "Lightweight Secure PUFs", *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 670–673, 2008.

35. Suzuki, D. and K. Shimizu, "The Glitch PUF: A New Delay-PUF Architecture Exploiting Glitch Shapes", *Cryptographic Hardware and Embedded Systems (CHES)*, pp. 366–382, 2010.

36. Patel, H., J. Crouch, Y. Kim and T. Kim, "Creating a Unique Digital Fingerprint Using Existing Combinational Logic", *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2693–2696, 2009.

37. Crouch, J., H. Patel, Y. Kim and R. Bennington, "Creating Unique Identifiers on Field Programmable Gate Arrays Using Natural Processing Variations", *International Conference on Field Programmable Logic and Applications, (FPL)*, pp. 579–582, 2008.

38. Patel, H., Y. Kim, J. McDonald and L. Starman, "Increasing Stability and Distinguishability of the Digital Fingerprint in FPGAs through Input Word Analysis", *International Conference on Field Programmable Logic and Applications, (FPL)*, pp. 391–396, 2009.

39. Bellaouar, A. and M. Elmasry, *Low-Power Digital VLSI Design. Circuits and Systems, 1st edn*, Kluwer Academic Publishers, 1995.

40. Helfmeier, C., C. Boit, D. Nedospasov, S. Tajik and J. P. Seifert, "Physical Vulnerabilities of Physically Unclonable Functions", *Design Test and Automation in Europe (DATE)*, 2014.

41. Su, Y., J. Holleman and B. Otis, "A 1.6pJ/bit 96% Stable Chip-ID Generating Circuit using Process Variations", *IEEE International Solid-State Circuits*

Conferenc (ISSCC), pp. 406–611, 2007.

42. Maes, R., P. Tuyls and I. Verbauwhede, “Intrinsic PUFs from Flip Flops on Reconfigurable Devices”, *Benelux Workshop Information and System Security (WISSec)*, 2008.

43. Busch, H., M. Sotáková, S. Katzenbeisser and R. Sion, “The PUF Promise”, *Trust and Trustworthy Computing*, Vol. 6101 of *Lecture Notes in Computer Science*, pp. 290–297, Springer Berlin Heidelberg, 2010.

44. Xin, X., J. Kaps and K. Gaj, “A Configurable Ring-Oscillator-Based PUF for Xilinx FPGAs”, *14th Euromicro Conference on Digital System Design (DSD)*, pp. 651–657, 2011.

45. Maiti, A. and P. Schaumont, “Improved Ring Oscillator PUF: An FPGA-friendly Secure Primitive”, *Journal of Cryptology*, Vol. 24, No. 2, pp. 375–397, 2011.

46. Maiti, A., P. Schaumont, J. Casarona and L. McHale, “A Large Scale Characterization of RO-PUF”, *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2010.

47. Eiroa, S. and I. Baturone, “An Analysis of Ring Oscillator PUF Behavior on FPGAs”, *2011 International Conference on Field-Programmable Technology (FPT)*, pp. 1–4, 2011.

48. Gassend, B., D. Clarke, M. Dijk and S. Devadas, “Controlled Physical Random Functions”, *18th Annual Computer Security Applications Conference (ACSAC)*, 2002.

49. Goren, S., H. Ugurdag, A. Yildiz and O. Ozkurt, “FPGA Design Security with Time Division Multiplexed PUFs”, *International Conference on High Performance Computing and Simulation (HPCS)*, pp. 608–614, 2010.

50. Bhargava, M., C. Cakir and K. Mai, “Attack Resistant Sense Amplifier Based PUFs (SA-PUF) with Deterministic and Controllable Reliability of PUF Responses”, *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 106–111, 2010.

51. Bhargava, M. and K. Mai, “An Efficient Reliable PUF-Based Cryptographic Key Generator in 65nm CMOS”, *Design Test and Automation in Europe (DATE)*, 2014.

52. Majzoobi, M., G. Ghiaasi, F. Koushanfar and S. Nassif, “Ultra-Low Power Current-Based PUF”, *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2071–2074, 2011.

53. Chen, Q., G. Csaba, P. Lugli, U. Schlichtmann and U. Ruhrmair, “The Bistable Ring PUF: A New Architecture for Strong Physical Unclonable Functions”, *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST).*, pp. 134–141, 2011.

54. Lin, L., D. Holcomb, D. Krishnappa, P. Shabadi and W. Burleson, “Low-Power Sub-Threshold Design of Secure Physical Unclonable Functions”, *ACM/IEEE International Symposium on Low-Power Electronics and Design (ISLPED)*, pp. 43–48, 2010.

55. Helinski, R., D. Acharyya and J. Plusquellic, “A Physical Unclonable Function Defined Using Power Distribution System Equivalent Resistance Variations”, *46th ACM/IEEE Design Automation Conference (DAC)*, pp. 676–681, 2009.

56. Csaba, G., X. Ju, Q. Chen, W. Porod, J. Schimidhuber, U. Schlichtmann, P. Lugli and U. Rührmair, “On-Chip Electric Waves: An Analog Circuit Approach to Physical Uncloneable Functions”, *IACR Cryptology ePrint Archive*, Vol. 246, 2009.

57. Abramowitz, M. and I. A. Stegun, *Handbook of Mathematical Functions with*

Formulas, Graphs, and Mathematical Tables, p. 11, New York: Dover, 1972.

58. Fischer, V., F. Bernard, N. Bochard and M. Varchola, "Enhancing Security of Ring Oscillator-Based TRNG Implemented in FPGA", *International Conference on Field Programmable Logic and Applications (FPL)*, pp. 245–250, 2008.

59. Fischer, V., F. Bernard, N. Bochard, A. Aubert and J. Danger, "True Random Number Generators in Configurable Logic Devices", *Project ANR - ICTeR*, pp. 23–28, 2009.

60. S.Yoo, D. Karakoyunlu, B. Birand and B. Sunar, "Improving the Robustness of Ring Oscillator TRNGs", *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, Vol. 3, 2010.

61. Liu, C., "Jitter in Oscillators with 1/f Noise Sources and Application to True RNG for Cryptography", Ph.D. dissertation, Worchester Polytechnic Institute, 2001.

62. Cherkaoui, A., V. Fischer, A. Aubert and L. Fesquet, "Comparison of Self-Timed Ring and Inverter Ring Oscillators as Entropy Sources in FPGAs", *Design, Automation Test in Europe Conference Exhibition (DATE), 2012*, pp. 1325–1330, 2012.

63. Johguchi, K., A. Kaya, H. Mattausch and T. Koide, "Measurement-Based Ring Oscillator Variation Analysis", *Design and Test of Computers, IEEE*, Vol. 27, 2010.

64. Eiroa, S. and I. Baturone, "Circuit Authentication Based on Ring-Oscillator PUFs", *18th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pp. 691–694, 2011.

65. Wold, K., *Security Properties of a Class of True Random Number Generators in Programmable Logic*, Ph.D. Thesis, Gjovik University College, 2011.

66. Valtchanov, B., V. Fischer, A. Aubert and F. Bernard, "Characterization of Randomness Sources in Ring Oscillator-Based True Random Number Generators in FPGAs", *13th International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, 2010.

67. Komurcu, G. and G. Dundar, "Determining the Quality Metrics for PUFs and Performance Evaluation of Two RO-PUFs", *IEEE 10th International New Circuits and Systems Conference, (NEWCAS)*, pp. 73–76, 2012.

68. Cormen, T. H., C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms*, MIT Press, 2 edn., 2001.

69. Aldous, D. and P. Diaconis, "Longest Increasing Subsequences: From Patience Sorting to the Baik-Deift-Johansson Theorem", *Bull. (new series) of the Amer. Math. Society.*, Vol. 36, No. 4, pp. 413–432, 2008.

70. Bohm, C. and M. Hofer, *Physical Unclonable Functions in Theory and Practice*, Springer, 2013.

71. Panda, A., S. Sarik and A. Awasthi, "FPGA Implementation of Encoder for (15, k) Binary BCH Code Using VHDL and Performance Comparison for Multiple Error Correction Control", *International Conference on Communication Systems and Network Technologies (CSNT)*, pp. 780–784, 2012.

72. Ruhrmair, U. and D. E. Holcomb, "PUFs at a Glance", *Design Test and Automation in Europe (DATE)*, 2014.

73. Gong, X., Z. Dai, W. Li and L. Feng, "Design and Implementation of a SoC for Privacy Storage Equipment", *International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE)*, pp. 435–438, 2011.

74. Choi, H., W. Liu and W. Sung, "VLSI Implementation of BCH Error Correction for Multilevel Cell NAND Flash Memory", *IEEE Transactions on Very Large*

Scale Integration (VLSI) Systems, Vol. 18, No. 5, pp. 843–847, May 2010.

75. Lin, C. S. and L. R. Dung, “A NAND Flash Memory Controller for SD/MMC Flash Memory Card”, *IEEE Transactions on Magnetics*, Vol. 43, No. 2, pp. 933–935, Feb 2007.

76. Xueqiang, W., P. Liyang, W. Dong, H. Chaohong and Z. Runde, “A High-Speed Two-Cell BCH Decoder for Error Correcting in MLC nor Flash Memories”, *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 56, No. 11, pp. 865–869, Nov 2009.

77. Qi, X., X. Ma, D. Li and Y. Zhao, “Implementation of Accelerated BCH Decoders on GPU”, *International Conference on Wireless Communications Signal Processing (WCSP)*, pp. 1–6, 2013.

78. Labiod, H., “A BCH Error Recovery Scheme for Adaptive Error Control in Wireless Networks”, *IEEE 49th Vehicular Technology Conference*, Vol. 3, pp. 2019–2023, 1999.

79. Liu, X. and Q. Hu, “10Gb/s Orthogonally Concatenated BCH Encoder for Fiber Communications”, *2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA)*, pp. 1018–1021, 2013.

80. Yoon, S., H. Lee, K. Lee, C.-S. Choi, J. Shin, J. Kim and J.-S. Ko, “Two-parallel Concatenated BCH Super-FEC Architecture for 100-GB/S Optical Communications”, *IEEE Workshop on Signal Processing Systems (SiPS)*, pp. 036–039, 2009.

81. Poolakkaparambil, M., J. Mathew, A. Jabir, D. Pradhan and S. Mohanty, “BCH Code Based Multiple Bit Error Correction in Finite Field Multiplier Circuits”, *12th International Symposium on Quality Electronic Design (ISQED)*, pp. 1–6, 2011.

82. Qi, B., L. Tang and J. Li, “A Novel Robust Watermarking Algorithm Based on Error Correction Coding”, *3rd International Conference on Innovative Computing Information and Control (ICICIC)*, pp. 22–22, 2008.

83. Queluz, M., T. Brandao and A. Rodrigues, “Signal Combining Techniques for Video Watermarking Extraction”, *IEEE Workshop on Multimedia Signal Processing*, pp. 347–350, 2002.

84. Lee, Y., H. Yoo, I. Yoo and I.-C. Park, “6.4Gb/s Multi-threaded BCH Encoder and Decoder for Multi-channel SSD Controllers”, *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, pp. 426–428, 2012.

85. Hiller, M. and G. Sigl, “Increasing the Efficiency of Syndrome Coding for PUFs with Helper Data Compression”, *Design Test and Automation in Europe (DATE)*, 2014.

86. MacWilliams, F. and N. Sloane, *The Theory of Error-Correcting Codes*, Northholland Publishing Company, 2nd edn., 1978.

87. Lin, S. and D. J. Costello, *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, 1983.

88. Peterson, W. W., “Encoding and Error-Correction Procedures for the Bose-Chaudhuri Codes.”, *IRE Transactions on Information Theory*, Vol. 6, No. 4, pp. 459–470, 1960.

89. Ruhrmair, U., J. Solter and F. Sehnke, “On the Foundations of Physical Unclonable Functions”, *Cryptology ePrint Archive*, Vol. 277, 2009.

90. Komurcu, G., A. E. Pusane and G. Dundar, “Dynamic Programming Based Grouping Method for RO-PUFs”, *9th Conference on Ph. D. Research in Microelectronics and Electronics (PRIME)*, pp. 329–332, 2013.

91. Kufluoglu, Z. and M. A. Alam, "A Computational Model of NBTI and Hot Carrier Injection Time-Exponents for MOSFET Reliability", *Journal of Computational Electronics*, Vol. 3, No. 3-4, pp. 165–169, 2004.

92. Kayam, N. R., "Experimental Analysis on Aging of Integrated Circuits", M.S. Thesis, University of Connecticut, Connecticut, 2011.

93. Simevski, A., R. Kraemer and M. Krstic, "Low-Complexity Integrated Circuit Aging Monitor", *IEEE 14th International Symposium on Design and Diagnostics of Electronic Circuits Systems (DDECS)*, pp. 121–125, 2011.

94. Maiti, A., L. McDougall and P. Schaumont, "The Impact of Aging on an FPGA-Based Physical Unclonable Function", *International Conference on Field Programmable Logic and Applications (FPL)*, pp. 151–156, 2011.

95. Kirkpatrick, M. and E. Bertino, "Software Techniques to Combat Drift in PUF-based Authentication Systems", *Secure Component and System Identification (SECSI)*, 2010.

96. Meguerdichian, S. and M. Potkonjak, "Device Aging-Based Physically Unclonable Functions", *48th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 288–289, 2011.

97. Maes, R., V. Rozic, I. Verbauwhede, P. Koeberl, E. van der Sluis and V. van der Leest, "Experimental evaluation of Physically Unclonable Functions in 65 nm CMOS", *Proceedings of the ESSCIRC*, pp. 486–489, 2012.

98. Maiti, A. and P. Schaumont, "The Impact of Aging on a Physical Unclonable Function", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2013.

99. Rahman, T., F. Domenic, J. Fahrny and M. Tehranipoor, "ARO-PUF: An Aging-Resistant Ring Oscillator PUF Design", *Design Test and Automation in Europe*

(DATE), 2014.

100. Yu, M.-D. and S. Devadas, "Secure and Robust Error Correction for Physical Unclonable Functions", *IEEE Design Test of Computers*, Vol. 27, No. 1, pp. 48–65, 2010.