Low-Power, Stable and Secure On-Chip Identifiers Design

Vignesh Vivekraja

A thesis submitted to the faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Engineering

Leyla Nazhandali, Chair

Dong S. Ha

Patrick Schaumont

August 27, 2010

Blacksburg, Virginia

Keywords: PUF, Process Variation, Low Power, Hardware Security

Low-Power, Stable and Secure On-Chip Identifiers Design

Vignesh Vivekraja

(ABSTRACT)

Trustworthy authentication of an object is of extreme importance for secure protocols. Traditional methods of storing the identity of an object using non-volatile memory is insecure. Novel chip-identifiers called Silicon Physical Unclonable Functions (PUFs) extract the random process characteristics of an Integrated Circuit to establish the identity. Though such types of IC identifiers are difficult to clone and provide a secure, yet an area and power efficient authentication mechanism, they suffer from instability due to variations in environmental conditions and noise. The decreased stability imposes a penalty on the area of the PUF circuit and the corresponding error correcting hardware, when trying to generate error-free bits using a PUF.

In this thesis, we propose techniques to improve the popular delay-based PUF architectures holistically, with a focus on its stability. In the first part, we investigate the effectiveness of circuit-level optimizations of the delay based PUF architectures. We show that PUFs which operate in the subthreshold region, where the transistor supply voltage is maintained below the threshold voltage of CMOS, are inherently more stable than PUFs operating at nominal voltage because of the increased difference in characteristics of transistors at this region. Also, we show that subthreshold PUF enjoys higher energy and area efficiency. In the second part of the thesis, we propose a feedback-based supply voltage control mechanism and a corresponding architecture to improve the stability of delay-based PUFs against variations in temperature.

Acknowledgments

Firstly, I would like to thank Dr. Leyla Nazhandali for her continued support and for providing me with interesting research topics.

Thanks to Dr. Dong Ha and Dr. Patrick Schaumont for their participation in my thesis committee.

Thanks to my friends at the PAC-Lab - Michael Henry, Dinesh Ganta, Steven Griffin, Kanu Priya and Sinan Huang for their assistance through the course of my masters.

Finally, I would like to thank my parents Dr. Vivekraja and Mrs. Santhi Vivekraja, and my family for their continued encouragement and support.

Contents

List of Figures

Chapter 1

Introduction

Over the past decades, the semiconductor industry has grown at a very rapid pace, governed by Moore's law. Consequently, silicon and computer technology has matured over these years and Integrated Circuits (ICs) are becoming increasingly efficient, cost-effective and powerful. As a result computers have become highly pervasive and are increasingly employed in almost every practical application. Some of these applications involve processing and storing of confidential user data, and authentication of a secure identifier to grant access to restricted areas/records. Examples include electronic passports, Radio-Frequency Identifiers (RFID) for secure access, anti-counterfeiting of expensive items and important documents, and authorized access to prescription medications in hospitals, to name a few. Any security vulnerabilities involved in the authentication of such devices, leading to access by rogue devices can prove to be very detrimental. Therefore, it is imperative that ICs are able to perform critical operations such as authentication of devices, secure communication, etc., in a highly secure, yet an inexpensive way.

Recently there has been a significant growth in the type and effectiveness of attacks on ICs to reveal the stored secret content and to create cloned rogue devices [1][2]. To give a few examples: FBI has announced that counterfeit Cisco products have been used unknowingly by the US government [3]. Also, unauthorized use of IPs especially meant for FPGAs is fairly common [4]. The most frequent form of attacks is non-invasive attacks. Such passive attacks involve man-in-the-middle attacks, wherein the data communication between the IC and the authentication device is intercepted to reveal the secret information in the communication channel. This information could be replayed later on by rogue devices to gain authentication. Also, side-channel attacks which are based on information gained from physical implementation of secure ICs rather than brute force have gained prominence [5]. Such attacks are based on analysis of the timing information, power consumption, electromagnetic leaks, etc, from the IC to extract extra sources of information, which could ultimately help in breaking the security of the system.

Various techniques and protocols are adopted by secure devices to thwart attacks, which reveal the secret information. Traditionally, the objects are identified by storing a unique key in a non-volatile memory and later by restoring the key for authentication. However, using such simple pieces of data as identifiers keeps the door open for stealing the secret identifiers easily. Modern security systems use cryptographic techniques to protect the data and provide secure authentication. Such crypto protocols are based on the premise that only authorized participants have access to a secret key, which is necessary to reveal the required information [6]. They use complex cryptographic hardware, which encrypts the secret data that can be revealed by decryption, using the shared secret key. Various public key and private key cryptographic techniques like AES, ECC, etc. provide a high level of security making it difficult to crack such systems. Breaking the security of such systems is costly and practically infeasible. However, the level of security provided has a direct relationship to the complexity of the cryptographic protocol and this translates into higher utilization of computing resources. This makes most of the secure cryptographic protocols infeasible for most of the embedded computing applications. For example, a light weight computing system like a passive RFID tag can hold only a maximum 2000 hardware gates for security purposes [7]. However, it has been reported that an efficient implementation of AES algorithm would consume 3400 gates. Also, an implementation of MD5 and SHA-256 hashing algorithms require 8000 to 10000 gates [8]. Furthermore, such cryptographic hardware are also power hungry making them unsuitable for ultra low power applications like smart cards.

In addition, since the keys used in cryptography are always stored in digital form, there is a finite probability that these keys could be extracted using another category of attacks on ICs called invasive attacks. These invasive attacks involve physically dismantling the package and probing key layers of the IC to reveal information [9][10]. Generally, tamper sensing circuits are embedded in the IC, which in turn, trigger certain areas of the IC to change its properties upon detection of any foul play. Such methods are very expensive and need careful physical design to realize tamper resistant hardware [11]. Therefore, it is a challenging problem to make an IC secure against all forms of attacks [12]. There is a need for a secure alternative to complement and to improve the traditional security mechanisms, especially for low cost and energy constrained applications.

Physical Unclonable Functions (PUFs) have recently been proposed and successfully implemented to provide security to computing devices in an inexpensive way [13]. Rather than storing keys in digital format, PUFs extract the randomness associated with any physical characteristics of an object to produce a unique and stable identifier for each object. There have been a variety of implementations including the optical PUFs [13] and coating PUFs [14]. However, the focus of this thesis is on one of the most popular and economic PUFs called silicon-based PUFs [15], which act as on-chip identifiers of an IC. These PUFs can be manufactured using the traditional CMOS fabrication process. Such PUFs employ innovative configurations of identically fabricated circuits to derive secrets from variations in wire delays, gate delays, leakage current, etc. rather than storing a unique identifier in physical memory. These variations are in turn dependent upon unpredictable factors, such as manufacturing variations, quantum mechanical fluctuations, thermal gradients, electromigration effects, parasitics, noise, etc. Since, silicon PUFs rely on the uncontrollable and unpredictable process variations during IC fabrication, the secret is extremely difficult to predict. Also, PUFs are tolerant towards invasive attacks and are less costly in terms of processing power and chip area.

It has been shown that PUFs could be employed as a standalone security system for Challenge/Response(C/R)[1] based authentication protocols or as a complement for traditional cryptography by producing device specific secret keys [16]. Therefore, silicon PUFs are ideal candidates for secure authentication of low cost devices like smart cards and for key generation for highly secure applications like user data protection, which typically involve key based crypto-hardware. For example, [17] discusses how PUFs can be used for secure device authentication, and [18][19] propose protocols using PUFs as key generators for cryptographic cores.

Though numerous silicon-based PUF architectures have been proposed and successfully implemented, most of them have low stability i.e. ability to replay the same response. This is because the output of the PUF is not only affected by the process variation but also by temporary operating conditions such as temperature and supply voltage. This means the same PUF may create different results at different times by having some of its output bits present a wrong value, which are called noisy bits. Based on the type of deployment of the PUF, this can have adverse effects on PUF security and cost. In C/R applications, lack of stability can result in denying authentication to a legitimate PUF device or worse, identifying a PUF device as another device. In the key generation applications, the effect of noisy bits is even more pronounced as a single noisy bit can result in 50% change in the encrypted message.

In order to alleviate the adverse effects of the lack of stability for PUFs, different methods have been proposed. For example, in C/R applications, if the hamming distance between the expected and the observed response of a PUF device does not exceed a certain threshold, the device is authenticated. Although this simple approach can reduce the number of false negatives, it can increase false positives and also make reverse engineering attacks simpler. Much more advanced and reliable methods, such as helper data functions and fuzzy

[1] C/R protocol is a lookup table based security protocol, wherein each of the devices to be authenticated provides a unique response. The response for a given challenge for each unique device is stored as a lookup table.

extractors [20][21][22] have been proposed to overcome the noisy bits problem in key generation applications. However, such techniques come with an extra cost of increased hardware complexity and consequently higher power consumption, both of which are critical for small embedded systems. The size of error correction hardware is proportional to the stability of the PUF. Therefore, there is an increased need of PUFs which are inherently more stable. Furthermore, these techniques are known to leak some of the secret data and reduce the security of the device and their extent of use has to remain limited. Finally, none of these techniques can result in a 100% stable key, since, theoretically, they can only reduce the probability of generating a wrong key but they cannot eliminate it. The probability of generating the correct key after error correction techniques depends on the original error probability of source bits. Therefore, it is very important to reduce the probability of a noisy bit generating the wrong value as much as possible.

A majority of the work in the PUF community have focused on architectural, protocol and algorithmic levels of abstractions to make the PUFs more stable and easily implementable in a reconfigurable platform. Also, improved error correcting schemes have been proposed. Some of this work has been mentioned in the future sections of this thesis. However, relatively less work has been done to make the circuit inherently more robust and stable, so that a lesser complex error correcting scheme could be applied to the PUF. The first part of this thesis is focused on this fundamental problem, where in we make the delay based PUF circuits more stable using circuit-level techniques. More precisely we study the effect of key circuit decisions on the popular ring oscillator and arbiter PUFs, namely 1) circuit topology and 2) the operating (supply) voltage of the circuit. Our goal is to uncover the best combination of these decisions to result in stable, yet area and power efficient realization of such PUFs.

It is well known that operating the circuit with a supply voltage less than threshold voltage, also known as subthreshold regions of operation, increases the effect of process variation drastically [23]. However, the power and performance of a circuit decreases exponentially as well [24]. In this work, we will show that operating the PUF circuit in subthreshold region has multiple advantages. As the most important advantage, we show a significant improvement in PUF stability when operating in this region. This is because the difference in characteristics of transistors gets amplified in the subthreshold region and it is difficult for environmental changes to get over this barrier to cause errors. Interestingly, this is contrary to normal CMOS circuits which are less stable in the subthreshold region. Other advantages of using subthreshold voltage operation are reduced power consumption and increased stability of the PUF device in the face of aging.

1.1 Contributions

In the first part of this work, we analyze the effect of circuit topology for increasing the difference in characteristics of transistors to yield more reliable PUFs. We will show thinner transistors are effective for increasing the stability of traditional super threshold PUFs. Later, we present a complete study of the two major silicon PUF architectures, i.e. Ring Oscillator PUFs (ROPUFs) and arbiter-based PUFs, and show how the amount of noisy bits are significantly reduced in these designs when operating in subthreshold voltage. Furthermore, in this thesis we present a theoretical analysis which establishes that PUFs have a higher life expectancy when operated at lower supply voltages. Finally, we will show that subthreshold PUFs are more energy efficient than their superthreshold counterparts, while the performance penalty is minimal. A part of this work has been published by the authors in [25].

In the second part of this thesis, we propose a novel feedback-based supply voltage control scheme and a corresponding architecture to enhance the stability of ROPUFs towards variations in temperature. We intend to apply this scheme over the circuit techniques proposed in this study, to further improve the stability of the popular ROPUFs. In this scheme we vary the supply voltage of the PUFs based on the operating temperature of the PUF IC. The optimal supply voltage to be applied to the PUF at each operating temperature is identified during the evaluation stage of the PUF and is stored in form of a micro-code. We will show that such a mechanism significantly improves the stability of the PUFs.

1.2 Thesis Organization

The rest of this thesis is organized as follows:

Chapter 2 presents the background. We introduce the concept of PUFs, and classify these novel identifiers based on the fabrication technique and the type of deployment. Then, the sources of variation in ICs are briefly explained.

Chapter 3 deals with circuit-level techniques to enhance the popular delay based PUFs including Ring Oscillator PUFs, and Arbiter PUF.

Chapter 4 proposes a feedback-based supply voltage control mechanism and a corresponding architecture to improve the stability of Ring Oscillator PUF against temperature variations.

Finally, Chapter 5 concludes this thesis.

Chapter 2

Background

In this chapter, we deal with the preliminaries of PUFs. Firstly, we define PUFs and describe the common deployment protocols followed by PUFs. Then, we classify PUFs according to the security and the fabrication process. Further, we describe the popular silicon PUF implementations. Finally, we discuss about the sources of variation in an IC, which is extracted by PUFs to produce a unique response.

2.1 Defining PUFs

Definition:

"A physical unclonable function (PUF) is a physical object that can take inputs and generate unpredictable outputs; it is unclonable in that the input/output behavior of a physical copy of one PUF will differ from that of the original one due to some uncontrollable randomness in the copying process." [26].

In other words, a PUF is a function which produces a secret output response based on the underlying properties of a physical device while adhering to various properties based on the deployment and the level of security intended from the device.

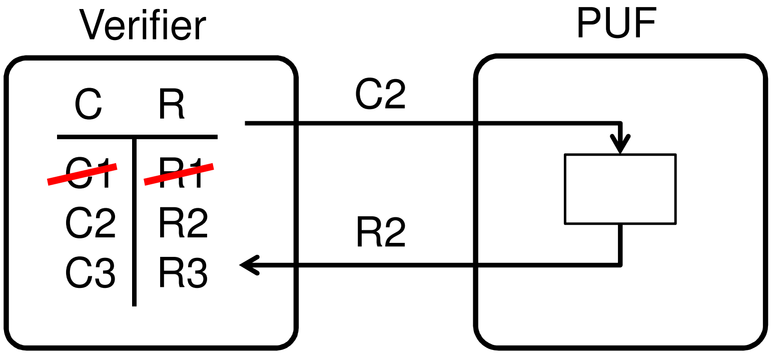


Figure 2.1: Challenge/Response based PUF Protocol

2.2 PUF Deployment

PUFs can be deployed for use in two distinct protocols that are described in this section.

2.2.1 Challenge/Response(C/R) based Authentication

One of the methods a PUF can be employed is using a C/R scheme. Figure 2.1 illustrates how a verifier tries to authenticate a PUF device. Upon fabrication, each PUF is subjected to an evaluation phase wherein the response for the set of all possible challenges is recorded by a trusted source and is provided to the verifier. The information recorded is represented as a table as shown in Figure 2.1. We refer to this set of responses for all possible challenges to a PUF device as its C/R space. Each PUF device would have its unique C/R table[1] due to the process variations in ICs. The C/R tables of all PUF devices are stored in the verifier system. To authenticate a PUF, the verifier sends a random challenge from the table to the PUF. It then compares the response to the set of pre-recorded PUF responses to match the corresponding PUF device. In order to prevent playback-attacks, each C/R pair in the table may be used only once. Therefore, the trustworthy lifetime of the PUF is determined by the number of C/R pairs in its table. In addition, to avoid aliasing between PUFs, each unique C/R pair can only be assigned to the table of one single PUF. Therefore, the total number

[1] Also referred to as C/R space.

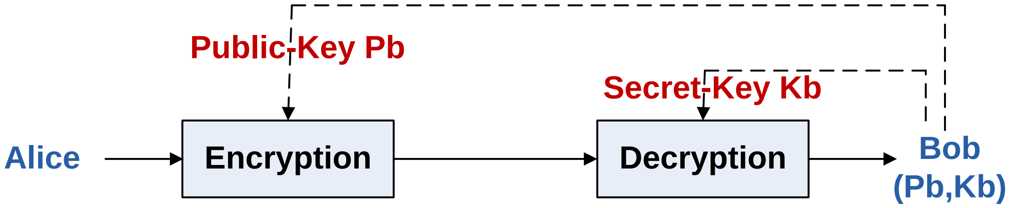


Figure 2.2: Private Key Cryptography Protocol

of unique C/R pairs determines the number of PUF circuits that can be fielded.

2.2.2 Key Generation for Cryptography

PUFs are also used for key generation in a variety of key based cryptographic protocols. In cryptographic protocols, the secret message to be transmitted is first encrypted, using a cryptographic function. The encrypted message can be decrypted to reveal the original plain text, only by devices which have the corresponding decryption key. Therefore, rogue devices which do not have the access to the correct decryption key would not be able to access the secret message. There are a variety of key based cryptographic protocols, for example, Figure 2.2 shows the asymmetric key cryptographic protocol. The message from Alice is encrypted using a public key, and the encrypted message could be decrypted only using a corresponding private key, which is possessed by Bob. Other rogue devices, which have no access to the private key would not be able to decrypt the message.

Traditionally, the private key is stored in the memory of the device. However, various invasive attacks could be used to reveal the key, thereby allowing rouge devices to gain access to the secure data. PUFs are used as key generators to produce a unique private key for each device. Due to the properties of PUFs stated earlier these keys are difficult to extract. Therefore, PUFs can be used as key generating devices for secure data sharing protocols. The public keys corresponding to the private keys from the PUF can be computed mathematically and be distributed to the required devices. Such applications typically use small PUF hardware with very stable output response. The main hardware cost in such applications is due to the encryption and decryption processes rather than the PUF producing the secret key.

Apart from the above two major protocols there are a variety of other secure protocols which use PUF. Some of them are listed in the following literature : [19] proposes Light weight PUF based RFID authentication, [27] details a hardware based public key cryptographic protocol and [16] describes a variety of PUF based authentication protocols.

2.3 Classification of PUFs

2.3.1 Classification Based on Security

PUFs are classified according to the application and the security features into the following major types [28] : Strong PUFs [13][29], Weak PUFs [4] and Controlled PUFs [30]. One common trait between all PUFs is that they must be easy to evaluate. Signifying, it must be easy to extract the necessary secret response from the PUF in a short time with high efficiency.

2.3.1.1 Strong PUF

A Strong PUF produces a secret response for a given challenge determined by the underlying properties of a physical device and adheres to the following properties apart from the generic characteristics of all PUFs:

1. Hard to Predict: An adversary who has access to polynomial number of physical measurements from the PUF device and has no further access to the device, can only extract insignificant amount of information about the response of the PUF to a randomly applied challenge.

2. Difficult to Clone: It must be almost impossible to fabricate a second strong PUF system, which exhibits the exact Challenge/Response behavior as the original PUF. This character-

istic must be held true even in the case of the manufacturer of the PUF.

3. Hard to Characterize: An adversary should not be able to extract the response for all possible challenges to the PUF. This signifies that the PUF should have a large amount of Challenge/Response space[2]. Also, reading consecutive responses from the PUF should take a finite time, so that the adversary cannot read out the entire PUF in a constrained environment[3].

Typically Strong PUFs are used for key generation, authentication and identification. These PUF based protocol are cheaper in terms of computation resources than traditional crypto-based protocols [11]. Some of the currently known Strong PUF implementations are: [13, 11, 31, 15, 29]. From security point of view, such PUFs typically do not have any mechanism to circumvent an adversary to apply a challenge and observe the response. Therefore, it is very important to maintain a large enough Challenge/Response space in the PUF.

2.3.1.2 Controlled PUFs

Controlled PUFs employ a wrapper logic built around a strong PUF to prevent challenges from being directly applied to the PUF and to thwart direct access to the responses of PUF. Typically such PUFs are employed to overcome modeling attacks [28]. However, the security provided by the controlled PUF is broken if the outputs of the underlying strong PUF are probed on its way to the control logic. These PUFs are generally used as a more secure alternative to Strong PUFs.

[2] C/R space - The total space of the C/R look up table.

[3] The terms finite time and polynomial in the above definition are relative to the size of the device and the level of security intended. In previous literature PUF have been referred to as Physical One Way Functions and Physical Random Functions [13].

2.3.1.3 Weak PUFs

Weak PUFs are a special case of Strong PUFs which have very few challenges. In the extreme case, they just have one fixed challenge. The secret response from the weak PUFs is used as a standard secret key for other key based cryptography functions (either public or private key cryptography). The outputs of such PUFs are never intended to be outputted to systems external to the security core. Weak PUFs are mainly used as a replacement to non-volatile memory based key storage for crypto-protocols; such PUFs are tolerant to invasive attacks unlike memory based systems. There are a variety of implementations of weak PUFs including Coating PUFs [14], Butterfly PUFs [32], SRAM PUFs [4] etc.

2.3.2 Classification Based on Fabrication

Various PUFs can be implemented to extract the physical properties of a variety of physical systems. Since most PUFs are used for security critical applications involving Integrated Circuits and computers, it is very important that PUFs could be easily integrated with ICs. Therefore, it would be advantageous if PUFs could be built along with such ICs using the existing ASIC fabrication flow. In this regard we classify PUFs into the following two major categories:

2.3.2.1 Silicon PUFs

These PUFs are fabricated using the existing ASIC fabrication process and therefore could be easily interfaced with other ICs or can be built on the same die with rest of the components. They are based on the uncontrollable process variations occurring during the fabrication of ICs. These variations make it impossible to manufacture two identical devices with identical characteristics. The device variations are captured using innovative configuration of identical circuits, leading to slight differences in the circuit characteristics like propagation delay, leakage current, voltage drop, etc. This difference in circuit characteristics is reflected as a unique response for a given set of challenges to the PUF device. Since these variations are uncontrollable and random in nature, each PUF device produces a device unique identifier/response. The sources of variation in silicon PUF and various implementations of such PUFs are discussed in the next few sections.

2.3.2.2 Non-Silicon PUFs

All other PUF implementations which are not classified as silicon PUFs are Non-Silicon PUFs. They derive secret keys from random variations occurring in physical systems other than Integrated Circuits. The first PUF was proposed by Pappu et al [13], and is based on variations occurring in optical systems. Such PUFs use the speckle pattern observed from an optical medium focused with a laser to derive secret keys. The other popular class of Non-Silicon PUFs include the coating PUFs from Phillips Labs [14]. In such PUFs random dielectric particles are coated on the top of the IC and this is reflected as changes in capacitance at different areas of the IC. Sensor arrays present in the different regions of the IC are used to detect the variation in capacitance and thereby produce a secret key. Though, such PUFs are fabricated on silicon systems, we classify it as Non-Silicon PUFs as they need special fabrication techniques which aren't part of generic CMOS fabrication technology. There are other Non-Silicon PUF implementations, some of which are discussed in [33][34]. In this thesis we focus only on silicon PUFs and we interchangeably refer silicon PUFs as PUFs.

2.4 Implementation of Silicon PUFs

2.4.1 Delay-Based PUFs

These PUFs utilize the variations in propagation delay of identical circuits to derive a secret response from the IC. The popular delay based PUF architectures include Ring Oscillator

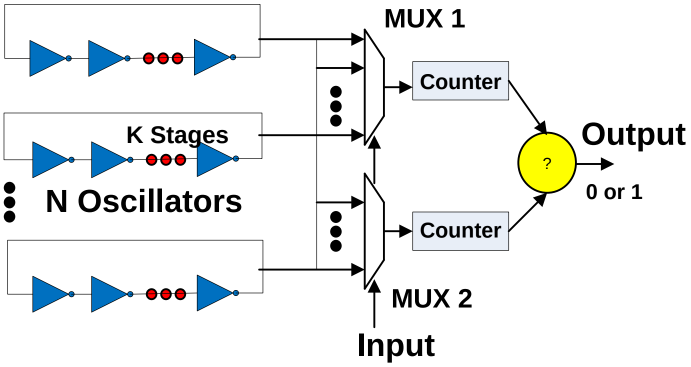


Figure 2.3: Architecture of Ring Oscillator PUF

PUFs , Switch-based PUFs and Tristate Buffer based PUFs.

2.4.1.1 Ring Oscillator PUFs

Ring Oscillator PUFs (ROPUFs) are based on variations in frequencies of identical ring oscillators to produce a secret PUF response. Figure 2.3 represents the architecture of a typical Ring Oscillator PUF [29]. It is comprised of N identical S-stage ring oscillators. Each of the oscillators oscillates at its characteristic frequency determined by the device characteristics of the underlying transistors. Theoretically, all the ring oscillator circuits would oscillate at the same frequency, but the inherent inter-chip, and intra-chip process variations, as well as the environmental conditions affect the oscillation frequency. This causes each oscillator to output a slightly different frequency. To derive digital values from these oscillators, a comparison is made between the frequencies of a pair of oscillators. The output bit is set to 1 or 0 based on which of the oscillators is faster. The selection of a pair of oscillators for comparison is controlled by the MUXs based on the input challenge to the circuit. To derive an M-bit output, M different comparison between the oscillators should be made. For a circuit with N ring oscillators, there are $N * (N-1)/2$ possible comparisons to make. Theoretically, the maximum possible M for maximum entropy is $log_2(N!)$[29]. This is because of the correlation between certain input challenges. For example, if a set of

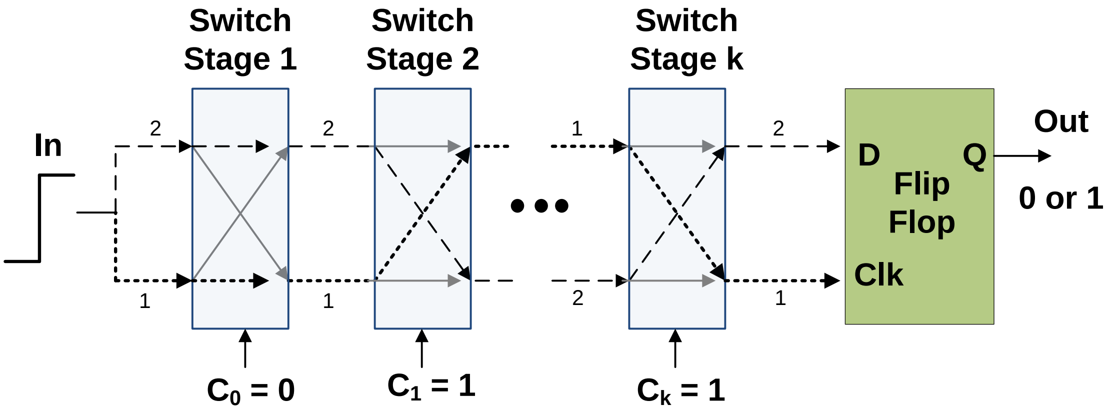


Figure 2.4: Architecture of Switch-Based PUF

challenges requires a comparison to be made between oscillators 1 and 2, 2 and 3, then the relationship between oscillators 1 and 3 can be deciphered.

Different ROPUFs have different output responses for the same challenge. This property is used to identify a given PUF based on the challenge response behavior or to produce a secret key.

2.4.1.2 Switch-Based PUFs

Switch-based PUFs [6] capture the variations in propagation delay of identical delay lines using an arbiter. Figure 2.4 shows the generic architecture for switch-based PUFs. Such PUFs consist of k stages of switching elements, with the outputs of each element connected to the inputs of the next stage. Each switching element has a two-bit input, two-bit output and a single bit challenge bit. By setting a 0 in the control bit, the inputs are mirrored to the outputs. However, the two input paths are switched, if a 1 is set in the control bit. The outputs of the last switching stage are connected to the D and Clock input of a flip flop (which is called arbiter). The structure of the switching element can lead to a variety of different designs for switch-based PUFs. However, the most popular implementation of switch-based PUF is the arbiter PUF [15] shown in Figure 2.5. Arbiter PUFs (A-PUFs) use

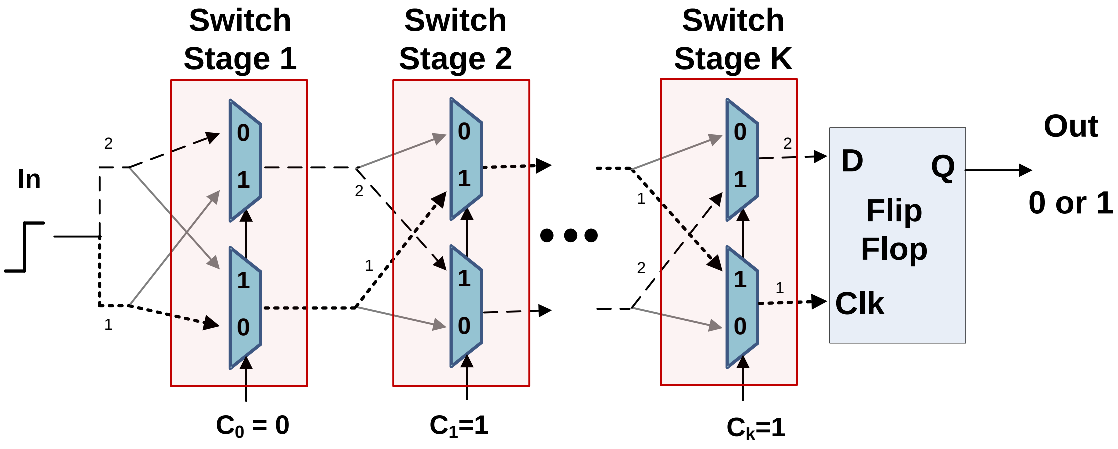


Figure 2.5: Architecture of Arbiter PUF

two multiplexers to implement each switching stage.

To evaluate such PUFs, the challenge key is fed on to each of the control bits of the switching stages. Since there are K stages, the PUF has a K-bit challenge key. By controlling the challenge bits of the switching elements it is possible to obtain different combination of delay lines. The common inputs to the PUF are fed with a rising waveform. Due to the process variations, one of the two delay lines has a shorter propagation delay than the other[4]. Thus, the rising waveform reaches either one of the D line or clock line of the flip flop, quicker than the other. The output of the flip flop is 1 if the signal to the D input arrives quicker, else the output is 0. For example, as shown in Figure 2.5, if an input challenge of 011 is fed to the PUF, the path indicated by the dashed line represents delay line 2 and the path indicated by the dotted line represents delay line 1. Both paths are perfectly symmetrical and when a rising input is given to this system, one of the delay lines propagates the input changes quicker than the other due to delay variations induced by process constraints. This is used to derive the digital 1-bit response. For a K-stage A-PUF, there are 2^K unique configurations or in other words, 2^K comparisons.

[4] The delay stages must be laid out in a symmetric fashion to make the outputs dependent only upon uncontrollable variations and not the skew due to asymmetry.

Compared to ROPUFs, such PUFs have a larger Challenge/Response space. However, some of the challenges lead to violations of setup or hold times of the arbiter, thereby causing metastability. This is because some of the paths have propagation delay close to each other, causing the transitions at the clock and D input of flip flop at an interval less than the setup or hold time. Under such conditions the outputs of the PUF is random, thereby leading to a loss in stability of the PUF. Traditionally, challenges causing metastability are pre-determined during the evaluation phase of the PUF and such challenges are forbidden to be used. However, such an evaluation is expensive and causes loss of certain challenges. Secondly, switch-based PUFs are not well suited for FPGA platforms, as it is difficult to build symmetric switching stages. Therefore, ROPUFs are traditionally preferred over switch-based PUFs.

2.4.1.3 Tristate Buffer PUFs

Tristate Buffer PUFs [35] borrow the idea of using an arbiter to capture the variations in identical delay lines for PUF response, from switch-based PUFs. However, they are more hardware efficient and use tristate buffers to select different delay paths. Tristate buffers have 2 inputs, and an output with three states: logic 1, logic 0 and high impedance. If the enable pin of the buffer is set, the input is reflected into the output pin, else the output will reach the high impedance state.

Figure 2.6 shows the architecture of the Tristate Buffer PUF. It has K delay stages, with the outputs of a certain stage cascaded with the inputs of the next stage. The final delay stage is connected to a flip flop, just like a switch-based PUF. Each delay stage has two delay units, each consisting of two tristate buffers. Each delay unit is constructed by connecting the input and output ports of the two tristate buffers. To ensure that only one of the buffers will be enabled, the two enable ports of the tristate buffers are connected to each other, with one of the buffers having an inverted enable input. The input to the enable bits of the tristate buffer forms the challenge for the system. By selecting different challenges, a

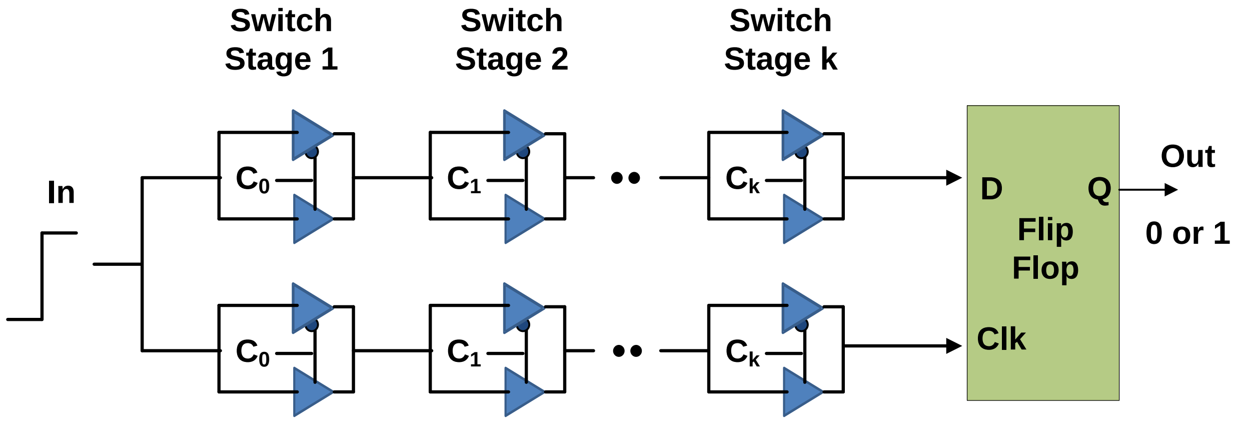


Figure 2.6: Architecture of Tristate Buffer Based PUF

path from In to D and In to clock can be selected with varying combinations of buffers. The bottom delay line from In to Clock and the top delay line from In to D, are completely independent unlike switch-based PUFs. Though all the buffers have identical designs, each of it has a slightly different propagation delay due to uncontrollable variations in the underlying transistors. Therefore, each challenge to the Tristate Buffer PUF, would lead to a slightly different propagation delay between the two delay paths from In to clock and D respectively. A rising transition is supplied to the input and this is reflected quicker in one of the inputs of the flip flop. This difference in transition of flip flop inputs is used to produce a single bit secret key for a given challenge, just as in the case of switch-based PUF.

The authors in [35] claim that Tristate Buffer PUFs are 18% more power efficient and 23% more area efficient than arbiter PUFs. Though such PUFs have a large C/R space, they suffer from metastability issues just like an arbiter PUF.

2.4.2 Memory-Based PUFs

Memory-based PUFs depend upon the unpredictable startup state of feedback based CMOS memory structures to produce a secret response. Most CMOS based memory structures including flip flops, SRAMs and latches, use a cross coupled structure with a positive feedback

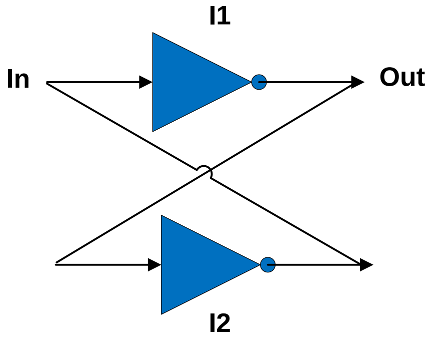


Figure 2.7: Cross Coupled Inverter Structure

to store the required logic. An example of such a structure is shown in Figure 2.7. It comprises of two inverters, the outputs of which are connected to the inputs of each other. Such structures have two stable outputs (logic 1 and logic 0) and an unstable output (logic X). Stable logics can be stored in the structure by driving appropriate values in the input. Upon power up with no input driven into the system, the output settles to either of the stable states, depending on the difference in characteristics of the inverters and the external noise. Ideally, if both the inverters are identical and there is no noise in the system, then the output of the system is in metastable condition forever. However, in practical systems, the inverters would have slightly different characteristics due to manufacturing limitations. This would result in the structure to output different logic states depending on the process characteristics of the devices involved. The output response of such systems can be used as PUF response. In order to provide multiple bits, an array of such devices can be used. There have been a variety of implementations of such memory-based PUF systems; in the following section we present two such examples: SRAM PUFs and butterfly PUFs.

The challenge and response of such memory PUFs is kept limited only to internal logic and never accessible to the output ports for higher security. Also, such PUFs are susceptible to environmental noise and therefore require a better error correcting hardware and redundant PUF structures to achieve reliable output key generation.

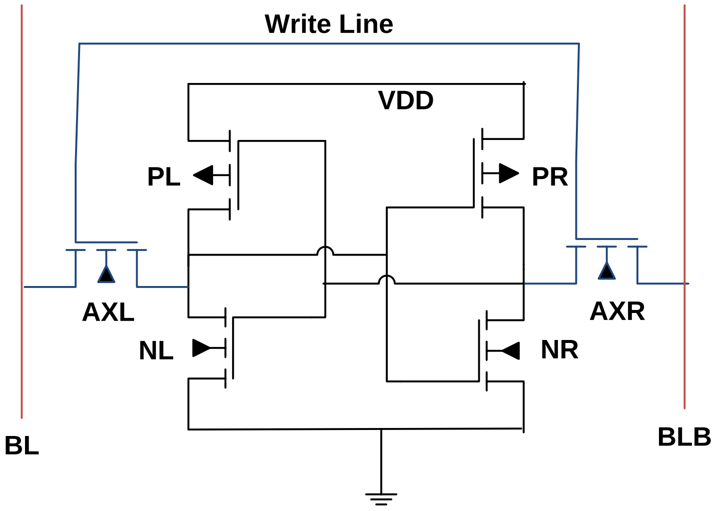


Figure 2.8: 6T SRAM Cell

2.4.2.1 SRAM PUFs

Figure 2.8 shows a traditional 6T SRAM memory cell [36]. It has two cross coupled inverters (load transistors PL,PR,NL and NR) and two access transistors (AXR,AXL). Traditionally, a write operation is achieved by loading the bit lines BL and BLB with appropriate values and turning the access transistors on. The read operation is achieved by forcing the two bit lines to logic 1 for a limited time and turning the access transistors on. Since the charge on the bit lines is stored dynamically, the bit lines are forced to the value stored in the cross coupled inverter structure. The sizes of transistors are carefully determined to achieve proper read and write operation of SRAMs. Secondly, the SNM (static noise margin), which is the amount of voltage needed to be applied to flip SRAM cells state, is set as high as possible.

However, for PUFs, SRAM cells [4] are used only to produce a random response based on process characteristics of the two load inverters. Therefore, the transistors are sized at minimal width to keep the cross coupled inverter structure more sensitive to device variations rather than noise. Upon power up, due to manufacturing variations, one of the inverters get a slightly higher voltage input than other, this is eventually amplified to logic 0 or 1 due to the feedback structure. This output is used as the response for the triggered PUF SRAM cell.

2.4.2.2 Butterfly PUF

Butterfly PUF [32] are memory-based PUF design with a focus towards reconfigurable platforms like FPGA, where the transistors are prefabricated and it is difficult to control device sizes, routing etc. They essentially use latches as a memory elements. Figure 2.9 shows the butterfly PUF structure. It has two latches the output of one connected to input of the other. Each latch has a PRE signal (turns the output to logic 1 on high) and a CLR signal (turns the output to logic 0 on high). The PRE signal of the latch1 and the CLR signal of latch2 are set to low. The clock of the latch is always set to high, to effectively make the latch function as a combinational circuit. The excite signal is connected to the CLR signal of latch1 and PRE signal of latch2. To evaluate the PUF, the excite is signal is pulled high, effectively making both the latches unstable (as the input and outputs of the latches are in opposite polarity). After a certain time, the evaluate signal is pulled low and the cross coupled latch structure settles down to a valid logic determined by the small manufacturing variations in the latches. This output is used as the outputs response for the butterfly PUF.

2.4.2.3 Other PUF Implementations

There are a variety of other silicon PUFs apart from the ones discussed previously. Many of which use analog circuit based approaches unlike the digital PUFs discussed above. [37] proposes an analog circuit based approach to produce secret keys from an IC. Such PUFs rely on slight variations in the on-current of identical PMOS and NMOS transistors to produce the secret response. [38] uses the variation in characteristics of identical Sense Amplifier circuits to derive a secret response. Also, there have been approaches to utilize variations in power rails, interconnects and even environmental variations to produce PUF responses. [39] proposes using the undesirable environmental variations as an additional layer of security to

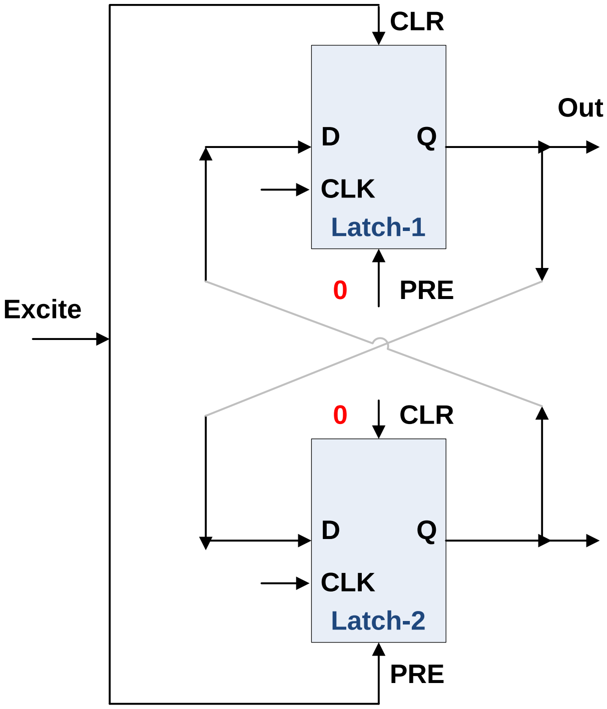


Figure 2.9: Butterfly PUF

the PUFs. [40] utilize the variation in resistance of identical power grids on an IC to produce the secret PUF response.

In this thesis we focus only on digital PUFs, more precisely the delay based PUFs.

2.5 Sources of Variability in Silicon ICs

To design better silicon PUF architectures, we need to understand the sources of variation in CMOS ICs. A variety of factors contribute towards the variation in the intended or designed properties/values of devices and circuits fabricated on an IC. These variations are reflected as a change in key parameters of transistors, such as threshold voltage, leakage current, delay etc, which effectively change the timing and sometimes even the logical behavior of the circuit. Figure 2.10 provides a brief overview of these sources of variation on an IC and

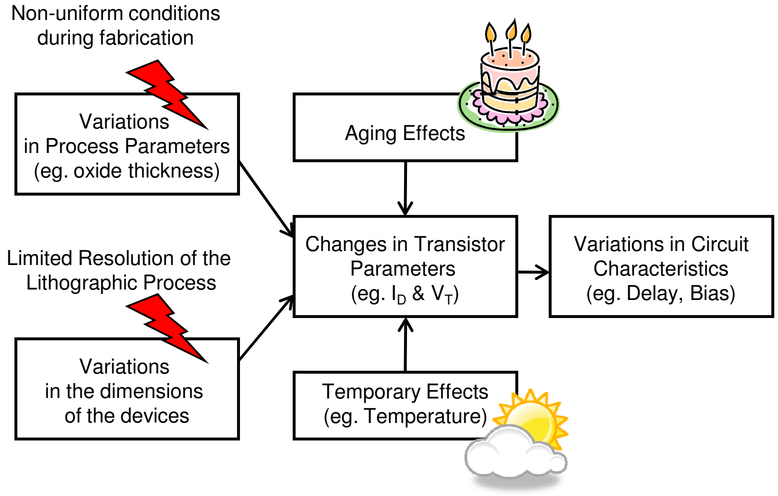

Non-uniform conditions during fabrication
Variations in Process Parameters (eg. oxide thickness)
Limited Resolution of the Lithographic Process
Variations in the dimensions of the devices
Aging Effects
Changes in Transistor Parameters (eg. I_D & V_T)
Temporary Effects (eg. Temperature)
Variations in Circuit Characteristics (eg. Delay, Bias)

Figure 2.10: Sources of Variability

they can be classified into the following factors: Permanent deviations due to manufacturing process, temporal variations due to environmental factors and variations due to aging.

2.5.1 Permanent deviations due to manufacturing process

Two identically designed circuits/interconnects have slightly different characteristics when fabricated using the same CMOS technology and equipment. Such manufacturing variations occurring between transistors in the same die/chip are called intra-die variation and that between different die is called inter-die variation. These manufacturing variations are largely random in nature occurring due to the limitations in the manufacturing. Such manufacturing variations can be divided into two major categories: The first being variations in process parameters such as oxide thicknesses, doping concentration, diffusion depths etc. They arise due to the non-uniform conditions in the silicon wafer and/or fluctuations in the diffusion of dopants. The second category covers the variations in dimensions of the devices. These include variations in the widths and lengths of the fabricated transistors. Such variations result from limited resolution of the photo-lithographic process.

2.5.2 Temporal variations due to environmental factors

These are time varying deviations of the operating conditions of the IC including temperature, external noise coupling, operating voltage fluctuations, etc. This causes the output characteristics of the circuit to vary based on the environmental factors. Since environmental variations are impossible to control, it is a challenging problem to maintain a stable output characteristic. Traditionally, various architecture and circuit-level approaches including feedback-based systems are used to make the IC tolerant towards such variation.

2.5.3 Perennial degradation due to aging

Aging is the degradation or deviation in characteristics of transistors with prolonged usage. Although very slow, the effects of aging are permanent. Aging results in slower operation of circuits, irregular-timing characteristics, increase in power consumption and sometimes even in functional failures. Traditionally, to combat aging, controlled scheduling of identical hardware blocks is done to allow all elements to age at the same rate, thereby increasing the life of an IC.

In traditional IC design, various techniques such as adaptive body biasing [41], multi threshold voltage transistors [42], etc. are used to minimize the impact of the above mentioned variations on the intended performance of the ICs. However, for PUFs it would be beneficial to extract the random manufacturing variations to derive a secret key. But, the PUF must be unaffected by temporal variations such as environment noise and aging to keep its response stable over time. In the next few chapters of the thesis, we will focus on various techniques to make the PUF response susceptible to the manufacturing variations but stable against all temporal and perennial deviations in device characteristics.

2.6 Summary

In this chapter, PUF was defined and the popular deployment protocols were explained. Then, the PUFs were classified into various categories based on the level of security and the fabrication process. Also, the popular silicon PUF architectures were briefed. Finally, the sources of variation in an IC were mentioned, along with their significance on the functioning of silicon-based PUF architectures.

Chapter 3

Circuit-Level Techniques for enhanced Delay-Based Silicon PUFs

3.1 Introduction

The characteristics of a PUF which forms the heart of a variety of security systems, drastically affects the performance, hardware requirements, power consumption and the lifetime of the system. Therefore, it is very important to design PUFs, whose characteristics meet the requirements of a secure system. Silicon PUFs have various conflicting design goals and it is difficult to satisfy them all at the same time. Figure 3.1 shows some such goals which have to be taken care during the design process. The desired characteristics from a PUF include - good stability against noise and environmental variation, as well as fast and low power operation. In this chapter, we will focus on circuit-level techniques to improve delay-based PUFs overall, to achieve a good balance between the various design goals.

Among the design requirements, the stability of a PUF is the most important criterion. This is because an unstable PUF would require better error correcting schemes to produce stable bits. Such error correcting hardware comes with an increased hardware and power cost,

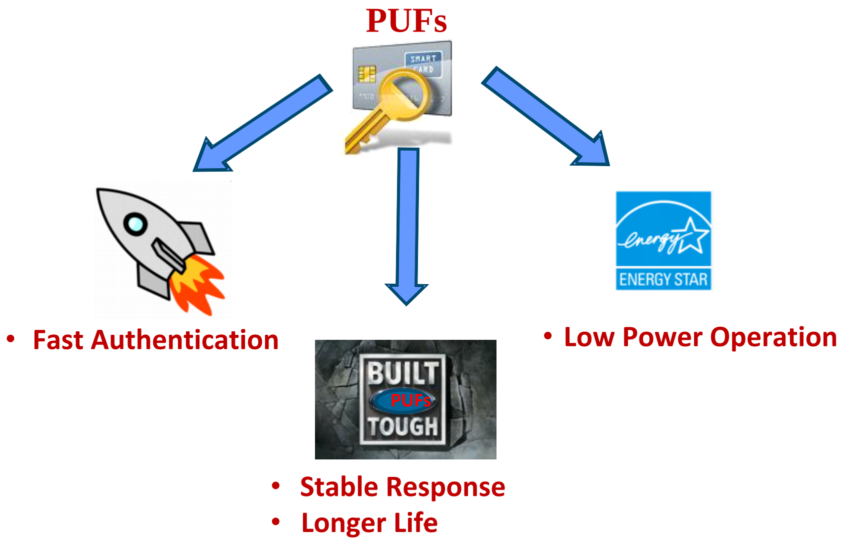


Figure 3.1: Design Goals of PUFs

which are very precious in low cost and energy constrained systems, for which PUFs are targeted. Also, more number of sources bits are required by the error correcting hardware, to produce stable output bits. This means that security systems based on unstable PUFs would require larger PUF structures to produce stable operation, thereby causing area and power penalty.

Increasing the stability of delay-based PUFs, especially against temperature and supply voltage variations is the main focus of this chapter. More precisely we focus on the two popular delay-based PUF architectures i.e. ROPUF and Arbiter PUF. We use circuit techniques to achieve higher stability. In the first part we analyze the effect of circuit topology including drive strength of gates and the number of stages of ROs to improve stability of ROPUFs. The motivation behind these decisions is the fact that PUF circuits which are more susceptible to process variation would have higher entropy and therefore would be more stable towards changes in the environment such as temperature changes. In the second part of this chapter, we show that the gains due to circuit topological alterations are limited and propose using subthreshold CMOS operation for delay-based PUF architectures. Our results show that using subthreshold CMOS operation for delay-based PUFs decreases the bit error probability from 0.069 to 0.012. This reduction can result in more than 50% in area savings, when a PUF is accompanied by error correcting hardware.

Also as an additional advantage of using subthreshold PUFs, we show that the other important design goals i.e.power and energy consumption of PUF are reduced by more than two order of magnitude. Finally, we present a theoretical analysis which shows that subthreshold PUFs age slower than traditional PUFs operating at nominal voltage. All these advantages of subthreshold PUF come at the cost of reduction operating speed. However, since PUFs are targeted for use in low cost and low performance systems, speed loss is not a major concern. Thereby we show that subthreshold PUF helps in achieving a favourable combination of the various design goals discussed above.

The rest of the chapter is organized as follows :

Section 3.2 presents the common experimental methodology followed in the rest of this thesis. In section 3.3, we discuss the effects of circuit topology on the performance of PUFs. Further, we analyze the effect of subthreshold voltage operation on delay-based PUF in section 3.4. Finally, section 3.5 presents the summary of this chapter.

3.2 Methodology

Before explaining the effect of various circuit techniques on the performance of PUFs, we present the common experimental methodology followed in the rest of this thesis. Methodologies used in specific cases, which are not mentioned in this section would be explained along with the relevant text.

All of the experiments were carried out using SPICE simulations. Monte Carlo analysis was carried out to simulate the effect of process variations. The simulations were performed on the 90-nm technology node. Industry standard transistor and process variation models from a commercial foundry (UMC) were used for all simulations.

The architectures of the PUF circuits are similar to the ones specified in Section 2.4. For experiments relating to ROPUFs: each ROPUF consists of 32 ring oscillators, each containing varying stages and drive strengths of NAND gates. One input of NAND gate is set to 1 to make it function effectively as an inverter. Such a circuit was simulated in SPICE over 20 different Monte Carlo runs. Both the intra-die and inter-die process variation flags were set during the simulation. This setup is analogous to the simulation of 20 different PUF ICs.

For, the arbiter-based PUFs, we simulated 10 different 2x1 multiplexers in parallel over 20 different Monte Carlo runs. This is analogous to simulating 20 different ICs each containing 10 different MUXs. We observed the gate propagation delay of each of these MUXs and later used Matlab to process this data and obtain a digital output. Such a setup was used in order to expedite the simulation process, while maintaining a high accuracy.

The simulations were carried out using the highest possible accuracy settings. Since it is a common practice in the design industry to rely on statistical transistor level simulations before actual implementation and since we used statistical models provided by a commercial foundry, we strongly believe that the results of the simulation are very close to that of actual implementation. In the next section, we explain the concept of digital signatures which would be used in quantifying the experimental results.

3.2.1 Extracting the Digital Signature

3.2.1.1 ROPUF - Digital Signature

As it is a common practice in the PUF design community, we derived a digital signature for each ROPUF IC instance by comparing adjacent oscillators. In other words, the frequency of the first oscillator was compared with that of the second, the frequency of the second was compared with that of the third, and so on. Therefore, except for the first and last oscillator, each oscillator was compared to two other oscillators. In our experiment, the comparison was done in an ideal fashion and no actual circuitry is implemented. Consequently, we obtained a 31-bit signature for each ROPUF.

3.2.1.2 Arbiter PUF - Digital Signature

After obtaining the propagation delay of the 10 MUX's, we assumed they were configured as a 5-stage arbiter-based PUF as shown in Figure 2.5. The propagation delay of the two delay lines under various input challenges were computed using a high level programming tool (Matlab). The arbiter PUF with 5 delay stages would consist of 2^5 different distinct challenges. For our experiments, we consider the digital signature of the arbiter PUF to be the output obtained under all of the distinct challenges. Consequently, we obtained a 2^5, i.e. 32-bit digital signature for each arbiter PUF.

3.2.2 Quantifying PUFs quality

Two major metrics are used to assess quality of PUFs, namely variability and stability. These two quality factors help in establishing the effectiveness of various techniques on the characteristics of PUFs.

Variability is a measure of how easily a PUF can be differentiated from others. For instance, it can be measured by the hamming distance between the outputs of two PUFs challenged with the same input. Since there are many possible input challenges, we just observe the hamming distance between the digital signatures of various PUF ICs. The probability density distribution of the hamming distances between different PUF IC effectively characterizes variability. PUFs with PDF curves which are taller centered on half the value of the number of bits in the digital signature are more easily identifiable i.e. unique than PUFs which have a flatter distribution curve.

Stability is the ability of the PUF to reproduce the same output response for the same input challenge at different times in the presence of environmental variations. It can be characterized by measuring the bits that remain unchanged while changing the environment variables, but keeping the input challenge the same. Again, since there are many possible input challenges, the digital signature is used for the analysis. Ideally, the PUF must be able to reproduce the response to a particular challenge in all 100% of occurrences, implying it must be able to reproduce the same response even at worst case scenario. We quantify stability by observing the PDF curves representing hamming distance of the digital signature, of the same PUF IC subject to different environmental conditions i.e changes in temperature, supply voltage, crosstalk, etc. A PUF producing a PDF curve centered on 0 hamming distance is more stable than a PUF having a flat PDF curve.

Both the stability and variability of PUFs were captured for the different delay-based PUF architectures operating at supply voltages in the superthreshold and nominal voltage region i.e. 0.2 V and 1 V . Also each single IC instance was subjected to temperatures ranging from −30 °C to 80 °C , at intervals of 10 °C.

3.3 Effect of Circuit Topology

In this section, we analyze the effect of certain circuit topologies on the performance of PUFs. More precisely we analyze the effect of the drive strength and the number of stages of ring oscillators in the quality of ROPUFs. The motivation behind these decisions is the fact that PUF circuit which is more susceptible to process variation will be more affected by random process variation and less by environmental changes. Thereby, they become more stable.

3.3.1 Effect of Number of Stages of RO

The number of delay stages of the ring oscillator affects the stability of ROPUFs. By having more number of them in a ring oscillator, the positive deviations (from average delay) tend to cancel out the negative deviations of another gate in the same chain. Therefore, we expect more ring oscillator pairs to have oscillating frequency relatively closer to each other and

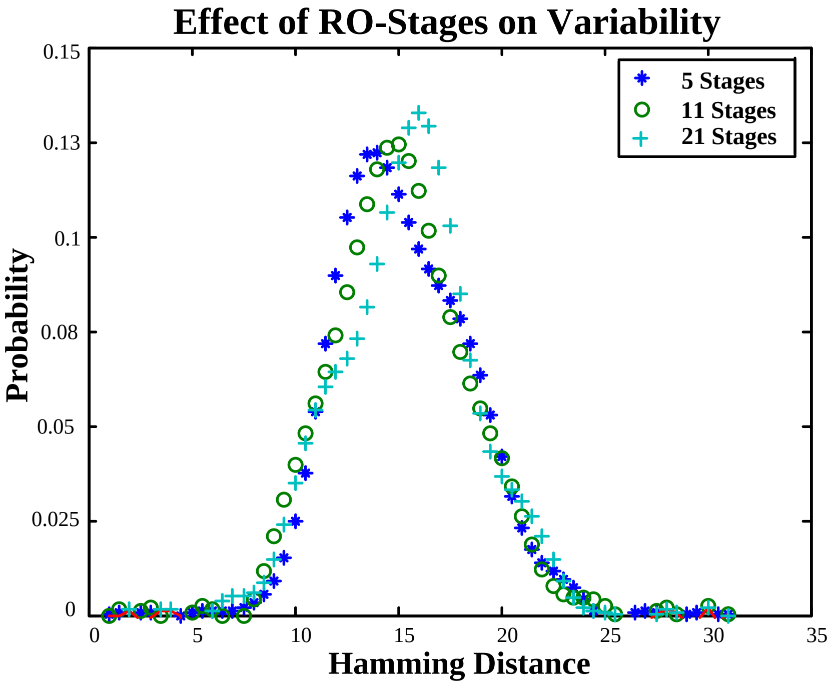


Figure 3.2: Variability in ROPUFs with varying number of stages of ROs

thereby possibly less stable. In this study, ring oscillators containing 5, 11 and 21 stages of NAND gates have been compared.

Figure 3.2 represents the variability in terms of hamming distance between the digital signatures of 20 different IC instances, each containing 32 ring oscillators. It is observed that irrespective of the number of stages of the NAND gates in the ring oscillator, the variability remains similar. This is because even a small variation in the process characteristics is enough to differentiate the two ring oscillators. However, the stability is affected if the process characteristics of two different ring oscillators are very close to each other. Figure 3.3 shows the effect of the stage length of the ring oscillator on the stability. In the following experiments only the effect of temperature variation has been considered for stability calculations. It is observed that ring oscillators with 5 stages are most stable followed by 11 stages and 21 stages. The probability of instances with 0 error bits decreases from 78% to 35% when the number of stages is increased from 5 to 21. This is in line with our initial premise that shorter stages leads to better stability.

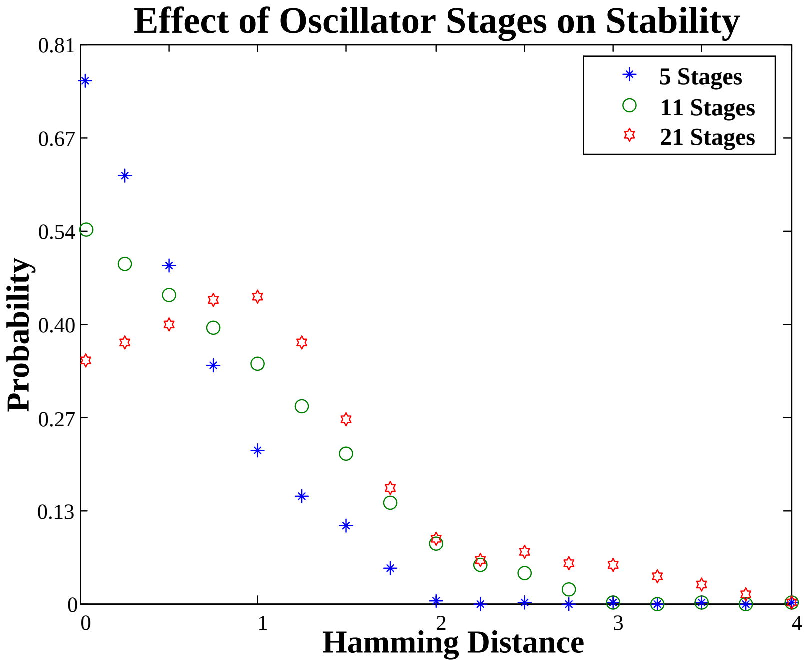


Figure 3.3: Stability of ROPUFs with varying number of stages of ROs

3.3.2 Effect of Transistor Width

Wider transistors are inherently more variation tolerant than thinner ones. This is because, wider transistors drive more current and small variations have a lesser effect on the circuit characteristics like delay, etc, than the thinner ones. Also, lithographic limitations during fabrication make the thinner transistors relatively more variant than the wider ones. For delay-based PUFs, the obvious choice would be the thinner ones as more variation implies that the frequencies of oscillators are distributed farther from each other implying a higher tolerance to noise, thereby an increase in stability. The result is shown in Figure 3.5. The widths of the thinner transistors are 0.2 times the normal ones and the wider transistors are 5 times wider than the normal ones. It is observed from Figure 3.5 that transistors with small widths are more stable than the ones with medium and large widths.It is also observed from Figure 3.4 that the variability is independent of the width of the transistors which is similar to the effect of ring oscillator stage length on the variability.

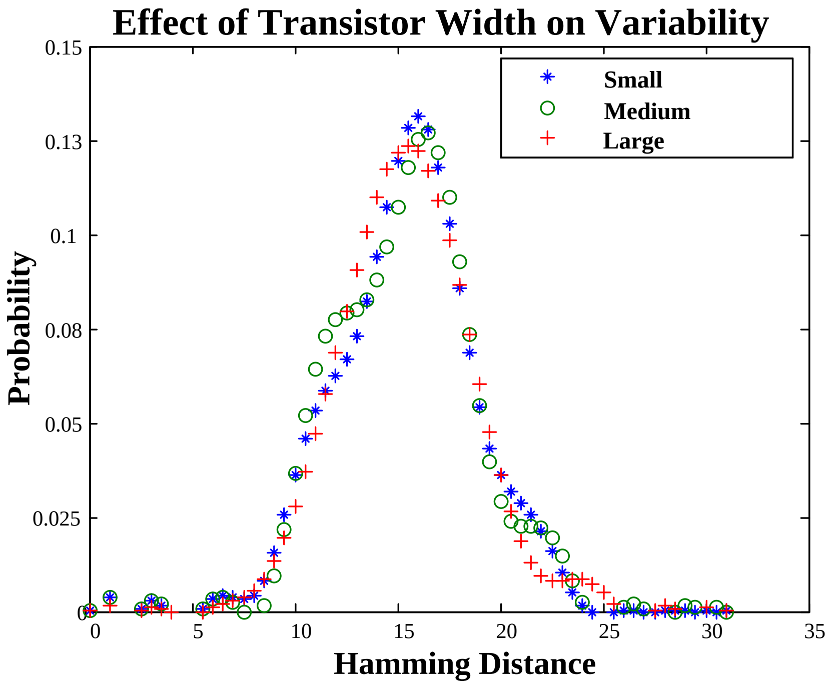


Figure 3.4: Variability of ROPUFs with varying drive strendths of transistors

Using the above techniques, the stability of ROPUFs has been improved from the worst case hamming distance of 4 to 2. However, the other benefits offered by circuit sizing on the overall characteristics (considing the various PUF design goals) of PUFs are limited. Also, the resolution of the counters in ROPUF hampers the number of stages which could be employed by ROPUF. Consequently, ROPUFs employing ring oscillators with lesser number of stages than the threshold for the technology are difficult to design and this imposes a restriction on the least number of stages of ring oscillators to be employed by ROPUF. Thereby, posing a limitation on realizing highly stable PUFs. In the following sections we will show that circuit techniques like voltage scaling, applied over circuit topological optimizations can help in achieving a more stable and power efficient delay-based PUF.

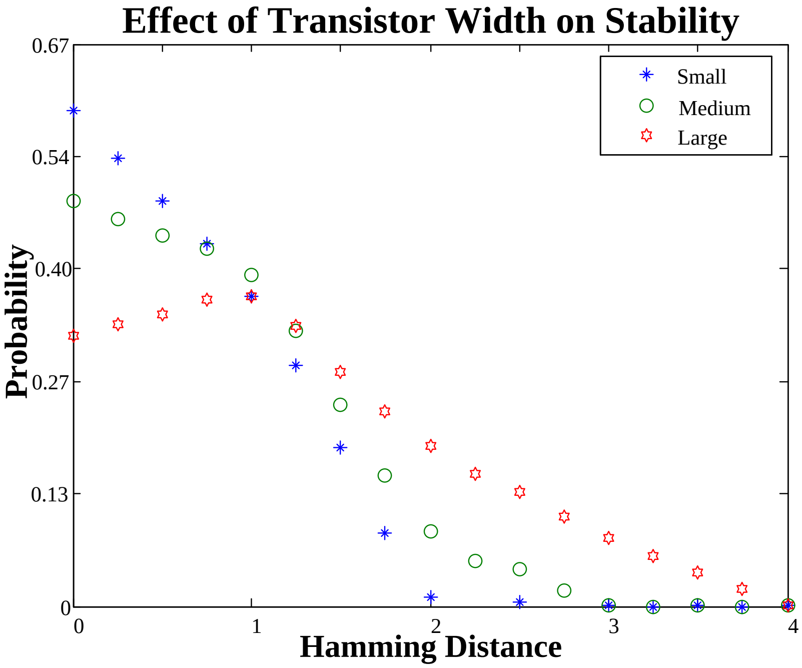


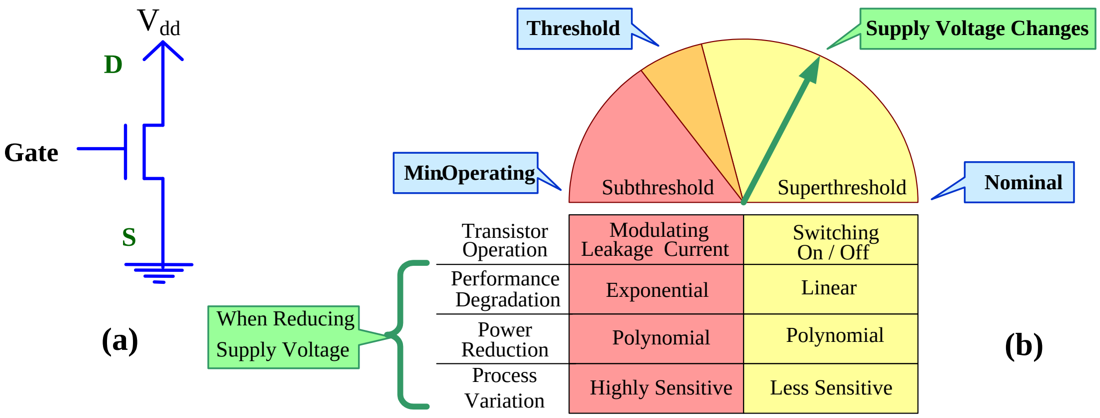
Figure 3.5: Stability of ROPUFs with varying drive strengths of transistors

3.4 Subthreshold Voltage Delay-Based PUFs

3.4.1 Subthreshold CMOS Operation

It is well-known that the power consumption and the frequency of a CMOS circuit are critically controlled by the supply voltage. The purpose of this section is to explain the impact of subthreshold operation in functioning of digital CMOS circuits. Figure 3.6(a) shows a CMOS transistor identifying its source and gate. When the gate-source voltage is above a certain threshold, the transistor effectively functions like a switch responding to the changes that come from gate voltage. Lowering the voltage source or voltage scaling has been a prevalent method for improving the energy efficiency of CMOS circuits [43].

The lower limit for voltage-scaling has typically been restricted to half the nominal voltage - the voltage that hardware is designed to typically operate at - and has been imposed upon by a few sensitive circuits with analog-like operation such as sense amplifiers. However, it has



	Subthreshold	Superthreshold
Transistor Operation	Modulating Leakage Current	Switching On / Off
Performance Degradation	Exponential	Linear
Power Reduction	Polynomial	Polynomial
Process Variation	Highly Sensitive	Less Sensitive

Figure 3.6: Overview of Subthreshold/Superthreshold Operation

been known for some time that standard CMOS gates operate seamlessly from full-voltage source to well below the threshold voltage - the voltage that turns the transistor on - at times reaching as low as 100mV [44].

In the past decade, a number of prototype designs have demonstrated that with careful design and replacement of these analog-like devices with standard switching counterparts, it is possible to extend the traditional voltage-scaling limit to below the threshold voltage, i.e., subthreshold voltage region [45][46]. Figure 3.6(b) provides an overview of subthreshold and superthreshold operation differences. First, the transistors are not switching as normal in the subthreshold region; instead they simply use the changes in the current that passes through them to charge or discharge their load and eventually perform computation. This, in turn, results in exponential degradation of performance in the subthreshold region as opposed to linear degradation when voltage is reduced in superthreshold.

Moreover, because the system operates at much lower voltages, it becomes more susceptible to some manufacturing and operational problems such as process variation and soft errors. These issues as well as accurate modeling of subthreshold leakage are currently under investigation by several research groups in the VLSI and digital electronics area who have shown promising results [46][47][24]. The focus of this thesis is not on building variation tolerant subthreshold CMOS circuits, but instead on using the susceptibility of subthreshold CMOS circuits to build more unique and stable PUFs.

To quantify the effect of process variation on circuit operation we use a key metric: ***Coefficient of Variation (COV)***. In statistics, COV is used in place of standard deviation as a measure of dispersion in different populations, when the populations have different average values. The COV of a group of 'n' ring oscillators is the ratio of the standard deviation of its characteristic frequency *'f'* to the average characteristic frequency of all ring oscillators in that population, as shown in formula-(3.1). Consequently, a higher value of COV indicates that the frequency of the ring oscillators in different chips are more spread apart. In other words, the design is more susceptible and less tolerant toward process variation. The motivation behind studying COV is that we believe, a larger dispersion in frequencies can result in higher stability of delay-based PUFs. We investigate this claim in future sections.

$$CoefficientofVariation = \frac{\sigma(f_1,f_2,..f_n)(n)}{\Sigma(f_1,f_2,..f_n)} \tag{3.1}$$

Figure 3.7, shows the scaling of COV with supply voltage. It is observed that COV increases at a slow but steady pace as we reduce the voltage from nominal voltage to the threshold voltage, around 500 mV. However, around this point the circuit starts to show significant increase in susceptibility towards process variation. The reason behind this is that below the threshold voltage, transistors are not switching as usual and rely heavily on leakage current for charging and discharging the load capacitance. Leakage current is more affected by process variation, which results in overall higher variability in subthreshold region. This effect is considered a drawback of subthreshold operation in general designs. However, we believe this effect can be employed to our advantage when designing a PUF circuit.

Building subthreshold ASICs comes with certain hardware overhead. Level converters would need to be attached to each output interface in order to boost the subtheshold voltage levels to interact with other ICs operating at nominal voltage. However, this is a small penalty compared to the advantages subthreshold logic provides to energy critical embedded systems.

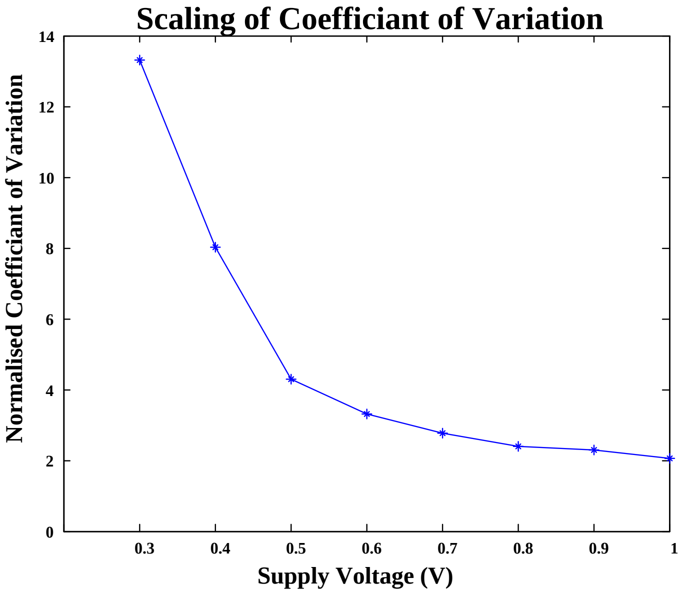


Figure 3.7: Scaling of Coefficient of Variation with Supply

Also, the standard 6T SRAM cell do not operate correctly in the subthreshold region. This is because; the decreased static noise margin of SRAM cells during the subthreshold operation causes the read operation to corrupt the contents of the cell. Also, subthreshold SRAM cells face issues during writing due to the ratioed contention between the access and the cell transistors. In our work we do not consider the effectiveness of subthreshold operation on SRAM based PUF.

3.4.2 Effect of Subthreshold CMOS on Delay-Based PUFs

In this section we will characterize and quantify the effectiveness of subthreshold voltage operation on delay-based silicon PUFs.

Figure 3.8 presents the variability associated with ROPUFs operated at the superthreshold and subthreshold region of operation. It is observed that the histogram of the hamming distances between the ICs and the corresponding PDF curves, representing the superthreshold and subthreshold regions of operation, fall on top of each other. It is clear that irrespective

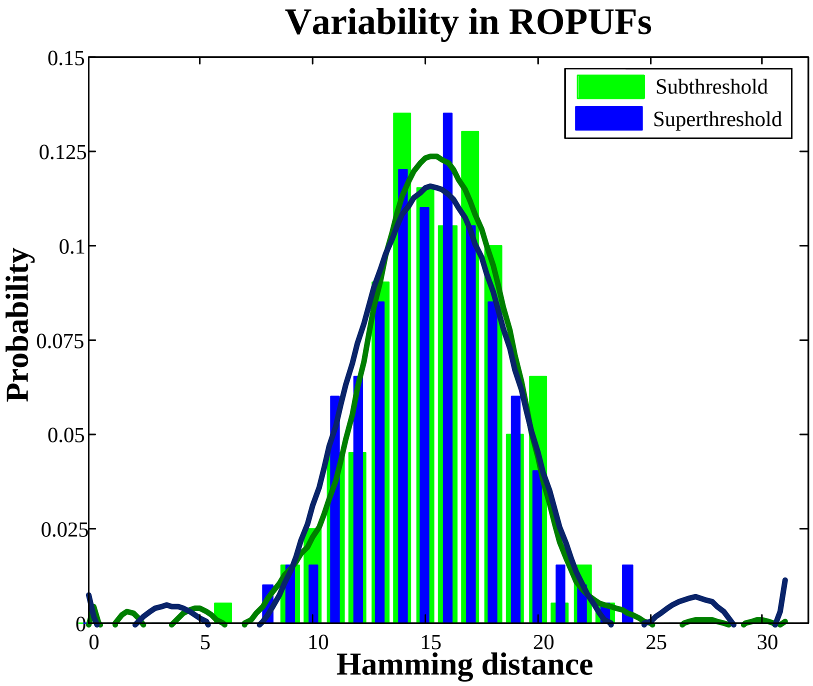


Figure 3.8: Variability in ROPUF

of the operating voltage, the hamming distance between the digital signatures of ICs follow a Gaussian distribution. This is because even the slightest of variations, irrespective of the magnitude, is enough to produce a unique digital signature for the IC. Also, since the process characteristics of the devices on the IC follow a Gaussian distribution, the PDF curves above maintain the same shape as well. However, the magnitude of deviation would determine the stability of PUFs.

Figure 3.9 presents the stability of a ROPUF with varying supply voltage. The stability is determined by comparing the digital signature of the same IC under varying environmental variables. In our case we varied temperature and supply voltage. A single IC instance was subjected to temperatures ranging from $-30\,°\mathrm{C}$ to $80\,°\mathrm{C}$ degree centigrade, at intervals of $10\,°\mathrm{C}$. Also, the ICs were subject to variations in supply voltage of up to 10%. While in the majority of the cases (66%), a hamming distance of zero i.e. no error bits are observed at the subthreshold region, only 17% of the cases of the superthreshold ROPUF are completely error free. Furthermore, a maximum of 1-bit error is obtained in the subthreshold ROPUF,

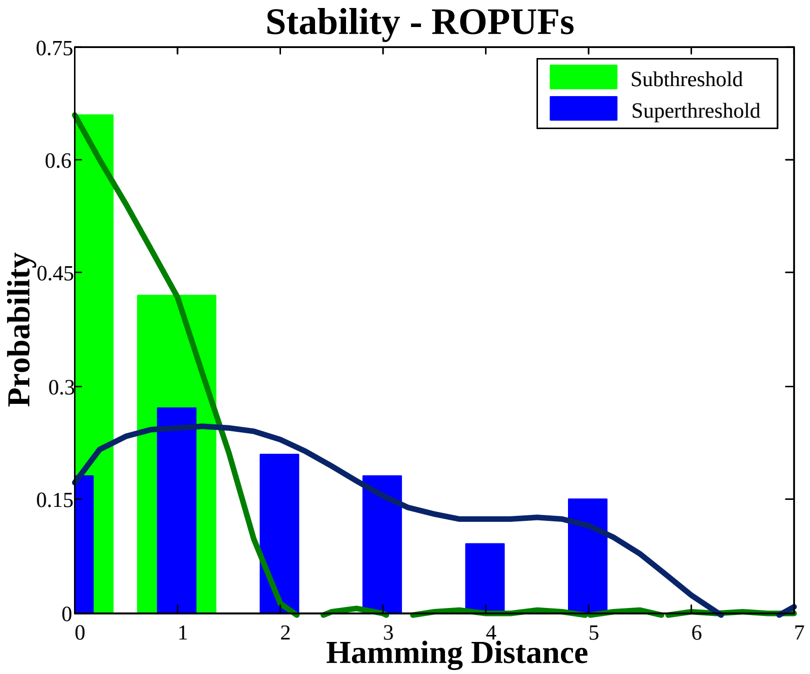


Figure 3.9: Stability in ROPUF

whereas a 5-bit error is obtained in the superthreshold ROPUF. Ideally, the number of bits with errors would increase with a larger PUF signature space. The subthreshold PUFs are more stable because the relative difference between the delays of the two circuit paths is much larger than the superthreshold region. Therefore, it is more difficult for external noise sources and changes in operating conditions, to break over this barrier to cause an error in the PUFs output response.

To get an idea of the hardware savings obtained due to the higher stability at the subthreshold region, we will consider the error correcting algorithms from [22], to produce stable output from noisy responses. From our previous results, we calculated that the probability for a given bit to be wrong (bit error rate (BER)) is 0.069 for superthreshold ROPUFs and it is 0.012 for subthreshold ROPUFs. From [22] we determined that the repetition code algorithm is the ideal error correcting code for the above bit error rates. It was observed that the number of source bits necessary to correct a PUF output with a BER of 0.069 is twice that of a PUF output with a BER of 0.012. More precisely, if we need 171 error free bits, then the subthreshold ROPUF would require 855 sources bits, whereas the superthreshold ROPUF would need 2227 sources bits. This translates to about 150 RO structures for subthreshold operation, but about 320 RO structures for the superthreshold operation. Therefore, in this case the hardware requirement of the subtheshold ROPUF is just about 46.4% that of a superthreshold ROPUF. This results in savings of about 53.6% in area. It must be noted that, the above example neglects the overhead due to the error correcting code algorithm itself, and only considers the number of source bits required. This is just a representative case and with better error coding algorithms the difference in hardware requirements of superthreshold and subthreshold PUFs may vary. However, this analysis helps in establishing that subthreshold ROPUFs would require lesser hardware than superthreshold ROPUFs.

Figure 3.10 helps in quantifying the stability of ROPUF without a specific environmental condition as a trigger. This provides a good measure of comparing superthreshold versus subthershold PUFs in terms of stability for unaccounted environmental changes. The figure shows the PDF of the following value for a certain PUF :

$$NormalizedDifferenceinFrequency = \frac{|fi - fj|}{Average(f)} \tag{3.2}$$

In other words, the graph shows how the differences between frequencies of any pair of ROs on a PUF (normalized w.r.t average frequency) are distributed. It can be seen that for superthreshold PUF the population is centered at 2% and has a very tall peak. This means most of the RO pairs have a 2% difference in frequency and almost all of them have a difference of less than 5%.

On the other hand, in subthreshold PUFs, RO pairs have relative differences of frequencies centered around 11%. Also, more than 94% of the RO pairs have a relative difference in frequecy of more than 2%, which is median for the superthreshold ROPUFs. This shows, that the relative difference in frequecies of oscillators is much larger in the subthreshold region. Therefore it would me more difficult for the unaccounted noise and environmental factors to flip the relationship of the oscillators in the subthreshold PUFs, to produce a wrong output, in comparison to superthreshold PUFs.

To show the effectiveness of subthreshold operation in other delay based PUF architectures we performed the above experiments on an arbiter-based PUF. Figure 3.11 presents the variability of subthreshold and superthreshold Arbiter PUFs. It is observed that at both these regions the variability is similar, as expected from the ROPUF experiments i.e. both the subthreshold and superthreshold regions can effectively distinguish a population of ICs from each other.

Figure 3.12 presents the stabilty of Arbiter PUFs. The superthreshold region has a flatter PDF curve than the subthreshold region indicating that subthreshold Arbiter PUFs are more stable than their superthreshold counterparts. The histograms show that, in the majority of the cases (75%), a hamming distance of one or zero i.e. a maximum of 1 error, bits are observed at the subthreshold region, only 31% of the cases of the superthreshold Arbiter have a maximum of 1 error. Furthermore, a maximum of 2-bit error is obtained in the subthreshold Arbiter PUF, whereas a 10-bit error is obtained in the superthreshold Arbiter PUF. This shows, that subthreshold Arbiter PUFs are more stable than their superthreshold counterparts.

Figure 3.13 analyses the performance trends in terms of time requirements, for a ROPUF simulated using the 90nm technology model. The measurement time to produce a single output bit, shown in the graph, is the amount of time required to produce a difference of 10 in the two counters connected to two different ring oscillators in a ROPUF.

Generic subthreshold CMOS circuits operate at much slower speeds than CMOS circuits operating at nominal voltage. The time period of a ring oscillator, operating at 0.2 V is around 200 times higher when it is operated at 1V. This is because of the decrease in drive current in the subthreshold region. This is considered to be a major disadvantage of subthreshold logic circuits, especially for high performance applications. However for PUFs the performance degradation with voltage scaling in terms of time is not much as compared

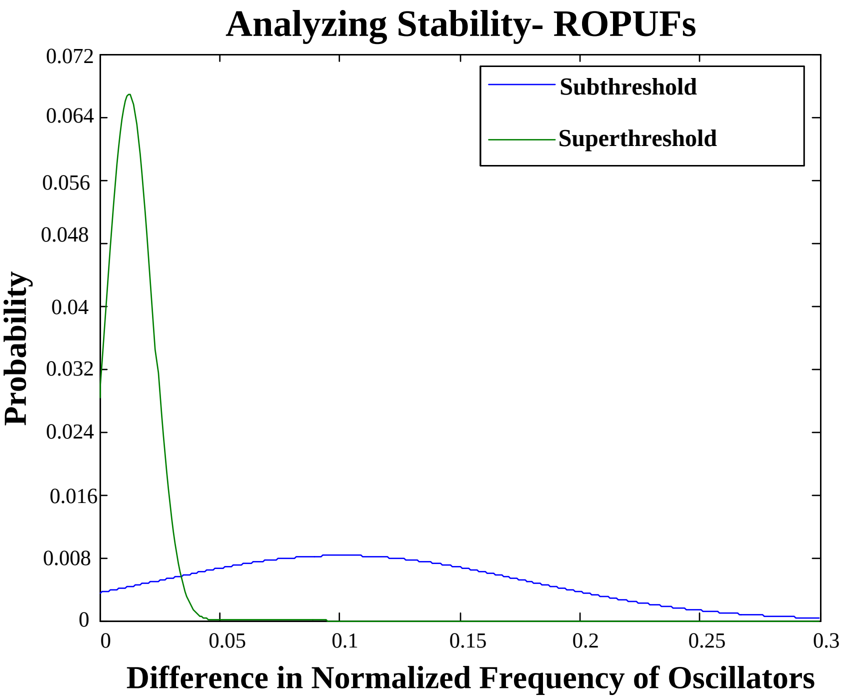


Figure 3.10: Normalized Difference in Frequency Distribution of ROs

to generic subthreshold logic circuits. Figure 3.13 shows the scaling of the measurement time with supply voltage. The graph shows that when a ROPUF is operated at 0.2V, it is just 10 times slower than when it is operated at nominal voltage. This is around 20 times increase in performance compared to generic subthreshold digital circuits. Such a phenomenon is observed because the performance of PUFs is tightly coupled with the amount of variation in the circuit. The large amount of variation in subthreshold region is able to produce a difference of 10 in the counter values in a fewer number of cycles as compared to superthreshold region. Also, the graph indicates the performance degrades by just 1.69 times when operated at 0.4V in comparison to 1V. This is not a huge penalty considering the other benefits of operating the PUF in the subthreshold region. Therefore, performance of subthreshold ROPUFs in terms of speed of operation is comparable to that of superthreshold ROPUFs. As for arbiter-based PUFs, the performance degradation is the same as any other digital circuit and therefore, somewhere between 50-100 times slower. However, since arbiter-based PUFs need to employ a hash function in order to avoid the well

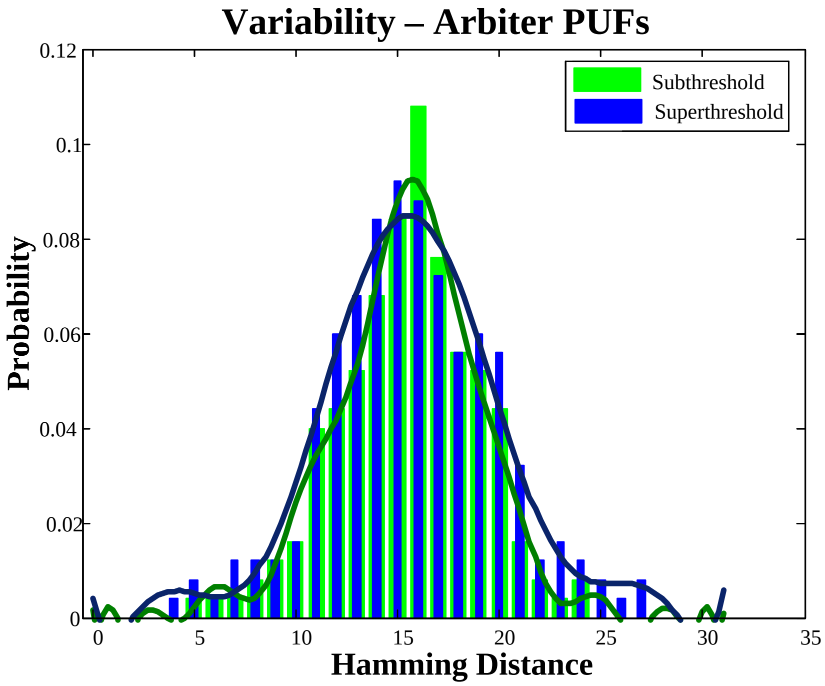


Figure 3.11: Variability in Arbiter PUF

documented reverse engineering attacks, the delay of the PUF devices is dominated by the hash module not the measurement module. Therefore, this reduction in performance does not affect the overall performance of the device.

Figure 3.14, shows the energy consumption of a ROPUF simulated using the 90nm technology model. To characterize the energy consumption: First, the average power consumed by the oscillator is obtained using Spice simulations, then, the energy consumed is calculated by multiplying the power dissipation with the measurement time and number of oscillators in the circuit. The figure 3.14 shows the scaling of energy consumed for obtaining a difference of 10, in the value of the 2 counters. This curve helps in understanding the trends of energy consumption of a delay-based PUF circuit. It is observed that the energy consumed at 0.2V is more than 2 orders of magnitude lesser than the energy consumed at nominal voltage. This is because of the decrease in dynamic energy consumed by CMOS circuits when operated at lower supply voltages as explained in the previous sections.

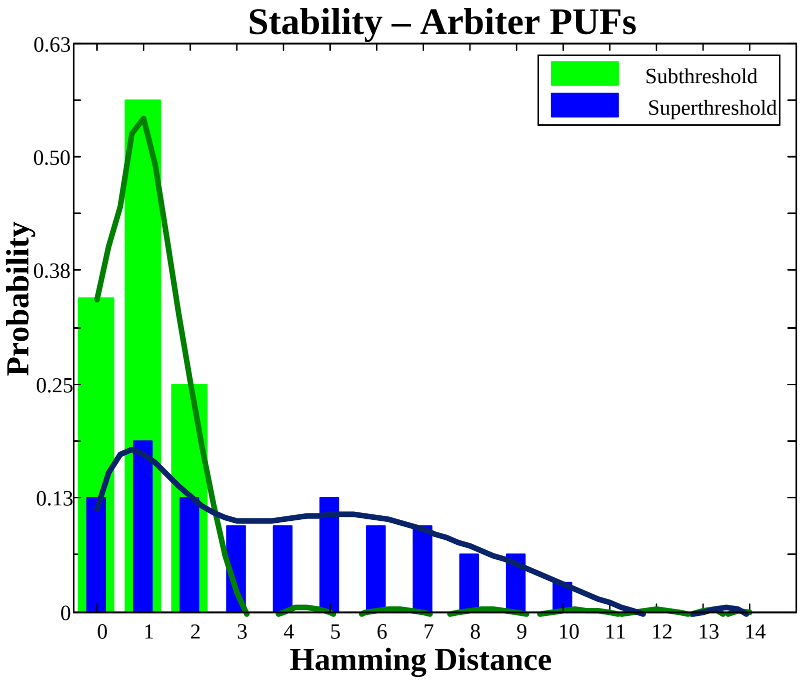


Figure 3.12: Stability in Arbiter PUF

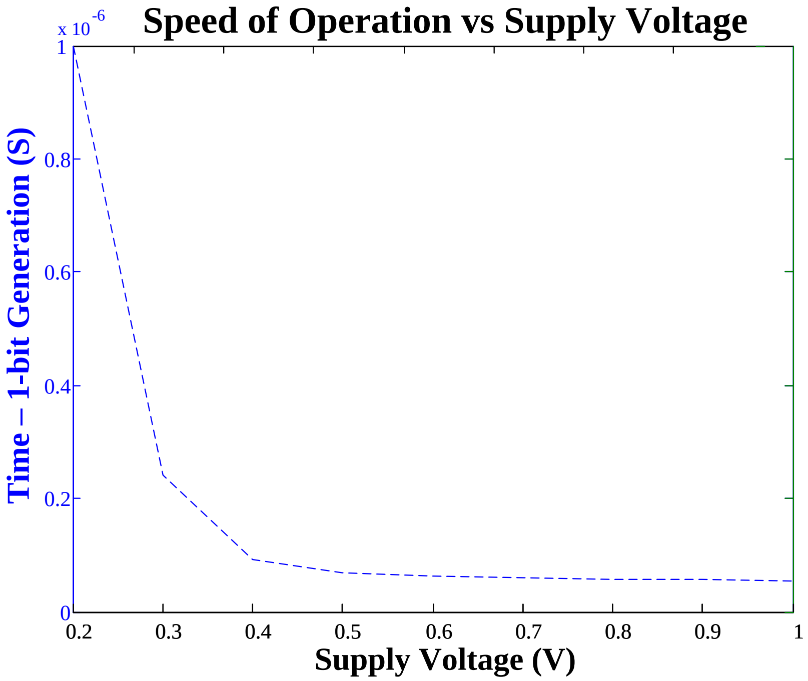


Figure 3.13: Performance of ROPUF

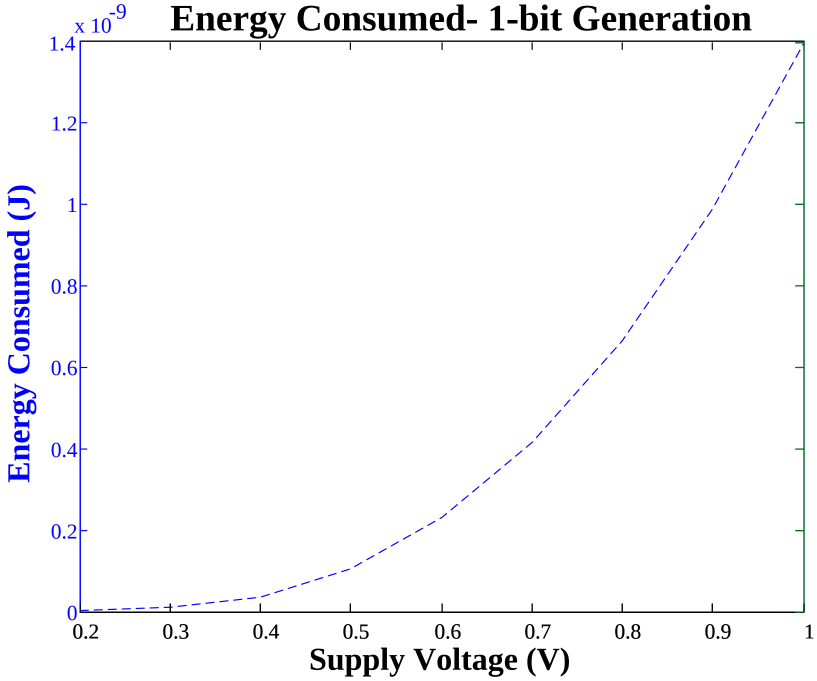


Figure 3.14: Energy Consumed by ROPUF for 1-bit Generation

This section helped in establishing the effectiveness of subthreshold operation in holistically improving delay-based PUF architectures. We showed that the stability, area and the energy consumption of delay-based PUFs is improved drastically with a small penalty in the speed of operation. In the next section, we will present a theoretical analysis to show that subthreshold CMOS age much slower than the traditional superthreshold CMOS circuit.

3.4.3 Aging and Overcoming it using Subthreshold Operation

With prolonged usage, MOS devices undergo changes in characteristics and this affects the circuits performance. In traditional digital circuits, aging would cause changes in critical paths of the circuit and this would eventually affect the circuit's performance and functionality. This can be overcome by using redundant circuits for checking operation. However, for PUFs changes in relative characteristics of MOSFETs can cause the PUF to output a different identifier with prolonged operation. This can cause the PUF to be wrongly authenticated as another device or never be authenticated even though it is the correct device. Redundancy based checks cannot be used for PUFs as each PUF circuit is unique and cannot be replicated. Therefore, it is very important to decrease the effect of aging in PUFs. In the following section we make a theoretical claim on why subthreshold operation decreases the ill effects of aging.

Hot carrier induction(HCI) degradation is chiefly responsible for degradation of individual transistors. When carriers travel through the channel from source to drain, the collision with the surrounding silicon atoms result in formation of a new electron and hole pair. These, new carriers over time, cause a damage to the physical structure of the transistor. Typically the threshold voltage of NMOS increases and that of PMOS decreases due to such a process [48]. However, when the operating voltage of transistor is decreased the electric field influencing the carriers and electrons decrease.Therefore, the collision results in less damage. [49] shows the degradation of transistors fabricated in 0.8 um technology increases with increasing supply voltage. Therefore, operating the PUF in subthreshold region would decrease the effect of HCI degradation.

The other most important phenomenon responsible for circuit degradation, especially in deep submicron technology is Negative Bias Temperature Instability(NBTI) [50] . The cause of Bias Temperature Instability (BTI) is the building up of positive charges at the interface of Si/SiO2 regions and near the oxide layer under elevated temperature and constant gate voltages. This causes degradation in functioning of MOS by indirectly affecting its threshold voltage. BTI occurring on PMOS transistors is NBTI and is the more dominant source of variation [51]. Equation 2 [52] models the variation in threshold voltage due to NBTI:

$$\delta V_{th} = \frac{q * T_{ox}}{\epsilon_{ox}} \sqrt{k^2 C_{ox}(V_{gs} - V_{th}) e^{\frac{2E_{ox}}{E_0}}} ct^{1/6} \tag{3.3}$$

For the sake of brevity, we refer the reader to [52] for the details of the parameters in this equation. However, the only parameter in this equation that is affected by supply voltage is V_{gs}, which is reduced when supply voltage is reduced. Therefore, ΔV_{th} will be smaller in subthreshold region. Also [53] shows that the expected life time of a digital circuit with a gate oxide thickness of 1.5 nm is increased by three orders of magnitude when operated at 1V as opposed to 2V. From, these results we can conclude that the degradation in threshold voltage of transistors in PUF circuits, would decrease when operated at a lower supply voltage.

3.5 Conclusion

This chapter establishes the effectiveness of circuit-level techniques to enhance delay-based PUFs. The initial study showed that circuit topological optimizations provide limited improvement in the stability of PUFs. Later, we discussed how subthreshold operation holistically improves delay-based PUF architectures. The experimental results and the theoretical analysis show that subthreshold PUFs are inherently more stable than superthreshold PUFs. This is because the difference in characteristics of transistors gets amplified in the subthreshold region and it is difficult for noise signals to get over this barrier to cause errors. Also, the variability of the delay-based PUF architectures is not affected by the operating voltage. Since, the subthrehold PUFs are more stable, the hardware required would be lesser as a fewer helper data bits would be sufficient to produce stable outputs. In our experiments the savings in hardware while using subthreshold PUFs was about 53.6% when a repetition code error correcting algorithm was used. Subthreshold PUFs are also more energy efficient than their superthreshold counterparts. The savings in energy of PUFs when operated at the subthreshold regions is more than two orders of magnitude in comparison to superthreshold region. Inspite of all these advantages the subthreshold PUF is just 10 times slower than a superthreshold PUF, which is minimal considering the other advantages. In essence subthreshold voltage operation is a very effective technique to improve delay-based PUF architectures.

Chapter 4

Feedback-Based Supply Voltage Control for Improving Stability of PUFs

4.1 Introduction

In this chapter, we propose a feedback-based supply voltage control mechanism to improve the stability of ROPUFs against variations in environmental temperature - which is one of the chief sources of errors in PUF ICs. Though, the subthreshold CMOS operation proposed in the previous chapter provides an overall improvement in the PUF characteristics, the feedback-based supply voltage control system can be used incrementally over some of the circuit-level optimizations, to further improve stability of delay-based PUF systems. Especially, such a system would be beneficial for delay-based CMOS PUFs operating at nominal voltage.

Nominal voltage PUFs may be preferred over subthreshold PUFs due to certain design constraints. For example, in mixed signal chips, where the analog and digital components are fabricated on the same die, the fast transitions of the analog components causes substantial noise in the power network and substrate node. Such noise signals ranging near 0.2V or more can make the operation of subthreshold CMOS infeasible. But, CMOS operating at nominal voltage could operate under such conditions, however, with a large deviation from its intended characteristics.

The above example just shows one such example where superthreshold CMOS PUFs would be preferred. Other scenarios includes avoidance of multi-voltage ICs. In this chapter we do not make a comparison of the scenarios or reason to why nominal voltage CMOS may be used, but focus on a feedback-based control system and a corresponding architecture to improve the stability of CMOS PUFs. The proposed mechanism could be applied on subthreshold PUFs as well, to further improve their stability against temperature variation.

The rest of this chapter is organized as follows : Section 4.2 presents the background regarding the effect of temperature variation on the operation of CMOS and PUF circuits as well as some of the related work. Section 4.3.1 details the proposed feedback-based supply voltage control mechanism and an architecture to achieve the above. Section 4.3.2 discusses the results to prove the effectiveness of the proposed mechanism. Finally, Section 4.4 concludes this chapter.

4.2 Background

4.2.1 Instability due to Temperature Variation

Process variations in Integrated Circuits cause identical designs to exhibit slightly different operational characteristics. In traditional digital design, the worst case process corner affecting the circuit the most is considered to design stable circuits. However, in PUFs we are rather interested in the relative difference in process characteristics of a circuit to produce a secret and stable response. Time varying parameters such as environmental variations

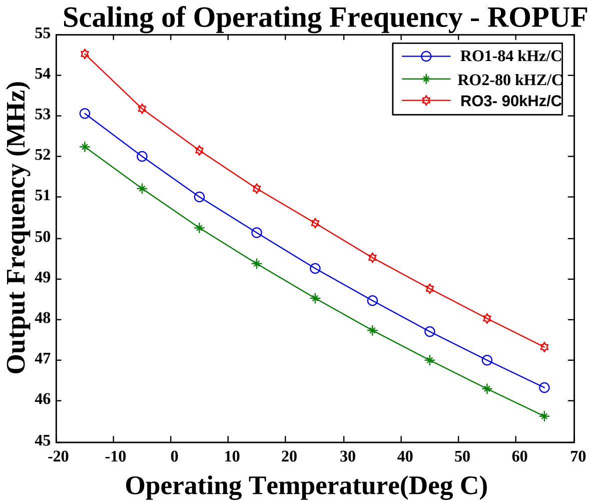


Figure 4.1: Scaling of Frequency with Temperature

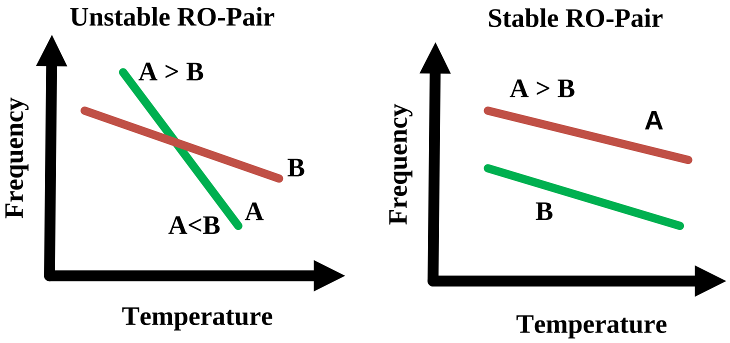


Figure 4.2: Fictional scaling of Frequency with Temperature

and other circuit-level noise signals cause deviations in circuit characteristics of PUF chips. For PUFs if the characteristics of all circuits scale the same way with these time varying environments then the outputs of the PUFs would be stable with respect to them. However, in reality, the various circuit characteristics of similar designs scale slightly differently based on the process characteristics. Therefore, it is possible that the relative characteristics of a circuit could completely toggle under different environmental conditions leading to an unstable response.

Temperature variation is one of the chief time varying conditions which affect the stability of PUFs. Figure 4.1 shows the scaling of frequency of identical ring oscillators with varying temperature. It is observed that the frequency of operation of all oscillators decreases with increasing temperature. However, the rate of decrease in frequency with temperature is different for different oscillators. If the relative difference between the frequencies of a pair of oscillators is high enough, then the relationship between the frequencies of the oscillators would hold across a broad range of temperatures. However, if the relative difference in frequencies is small, then the relationship between the frequencies of oscillators could toggle, i.e. the faster oscillator becomes slower and vice versa. This phenomenon is represented in Figure 4.2. Therefore, techniques that can inhibit the crossing over of the frequency of oscillators with changes in temperature can enhance the stability of PUFs.

4.2.2 Related Work

There have been a few attempts to address this problem. [54] proposes cooperative clustering of ring oscillators to produce stable output at varying temperature. [29] addresses this problem by using multiple ring oscillators to produce a stable bit. However, both of these methods require increased number of ring oscillators and therefore require a higher silicon footprint. In the next section of this chapter, we propose a feedback-based supply voltage control technique to enhance the stability of PUFs with a minimal hardware penalty.

4.3 Feedback-based supply voltage control for improved stability

The operating temperature of a PUF IC drastically affects the stability. The stability could be enhanced by preventing the crossing over of the frequency of oscillators by controlling certain parameters of the transistor. This parameter could be temperature itself, but the designer has no control over the environmental temperature. One possible approach, which could be easily controlled by the designer, is controlling the operating voltage of the transistor to overcome the flipping of bits due to changes in temperature.

Figure 4.3 shows the fictional scaling of frequency of ring oscillators with varying supply voltage. The fictional graph is shown because the actual graphs are too close to each other to be shown effectively. It is observed that the frequency of oscillators decreases with lowering of supply voltage. However, the rate of decrease in frequency of operation varies based on process characteristic. Therefore, if the frequencies of oscillators are too close to each other at the nominal voltage, then at a lower supply voltage the relationship between the frequencies of oscillators could flip.

In our method we improve the stability of ROPUFs against variations in operating temperature by controlling the supply voltage based on the operating temperature of the PUF IC. The key idea is that if the relationship between a certain pair of oscillators flip, then the supply voltage of the ring oscillators is scaled to overcome this flip in relationship, thereby making the ROPUF stable against temperature variations. However, scaling of supply voltage could flip the frequency relationship between pairs of oscillators which are stable against temperature changes. *To achieve 100% stability each pair of ring oscillator could be supplied with a different supply voltage based on temperature. But, such a method reveals information about the relationship between the critical paths of pairs of ring oscillators and is not a secure approach.* Alternatively, the supply voltage of all ring oscillators could be scaled to a same operating voltage such that the least number of bit flips occur at that particular temperature. Therefore, there is an optimal supply voltage to be maintained at each temperature range for each IC in order to produce the most stable response. In this paper we propose a feedback-based architecture to maintain the supply voltage at it optimal voltage depending upon the operating temperature.

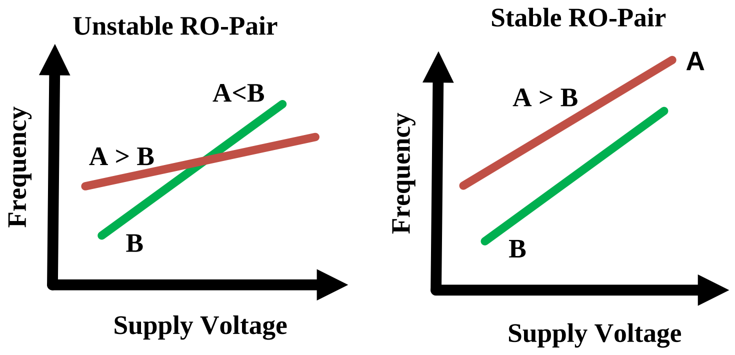
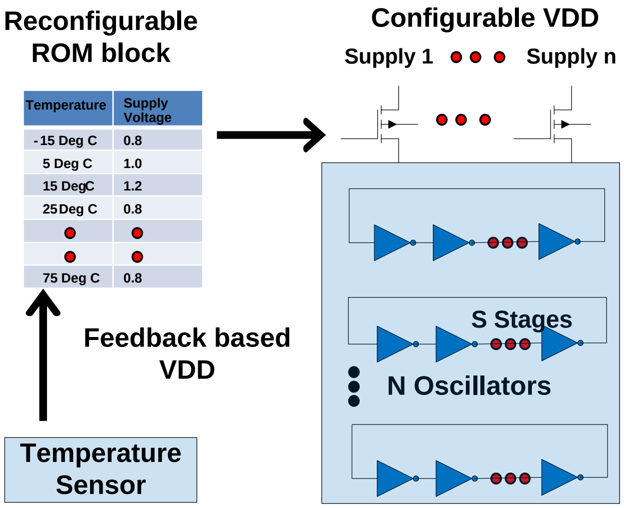


Figure 4.3: Scaling of Frequency with Supply Voltage



Temperature	Supply Voltage
-15 Deg C	0.8
5 Deg C	1.0
15 DegC	1.2
25 Deg C	0.8
●	●
●	●
75 Deg C	0.8

Figure 4.4: Feedback-based voltage control

4.3.1 Proposed Architecture

Figure 4.4 presents the architecture of our proposed system. Each ROPUF can be operated under multiple predefined supply voltages controlled by a programmable ROM. The PUF IC also contains a temperature sensor which keeps track of the operating temperature. The outputs of the temperature sensor are provided to the ROM, and in turn the ROM selects the supply voltage to be applied to the ROPUFs. The ROMs are programmed individually for each IC during the testing and configuration phase of the PUF after the fabrication. This feedback-based control of the supply voltage of the PUF based on monitoring the operating temperature improves the stability of ROPUFs against temperature variations. Also, as indicated in future section this architecture requires less hardware than the existing state of the art temperature variation tolerant PUFs briefed earlier in this paper.

The granularity, the precision and the placement of the temperature sensor and the controllable operating voltages have a significant impact on the stability of the PUFs. However, a more granular structure would require more hardware in terms of the number of bits in ROM, complexity in temperature sensor etc. This study is just a proof of concept to show the effectiveness of the feedback-based approach and we do not experiment with the effect of precision, granularity and geometric placement of the sensors. The temperature sensor in our case is a track and hold based thermal sensor proposed in [55] . The following steps show how the feedback controlled ROPUF IC is constructed, characterized and operated to yield stable output bits.

1. **Design Phase:** In this phase the PUF architecture specified above is designed. It is desirable to design ROPUFs, using the circuit techniques specified in the previous chapter. Thereby, making the PUFs further stable using this novel approach. The PUF is then sent for the manufacturing process.

2. **Testing and Configuration Phase:** In this phase the manufactured PUF ICs are tested across varying operating temperatures. Initially, the temperature sensors in each IC are calibrated to capture the various operating temperatures. Then, the output signature of each PUF IC is characterized at various operating voltages and across various operating temperatures. The operating voltage which yields the least difference in the hamming distance of the output signatures, in comparison to the base case signature at nominal voltage and 25 °C at different operating temperatures are characterized. These characterized supply voltages across temperatures are used to program the ROM controlling the supply voltage of the ring oscillators. Once the ROMs are programmed they cannot be tampered or reconfigured to avoid leakage of

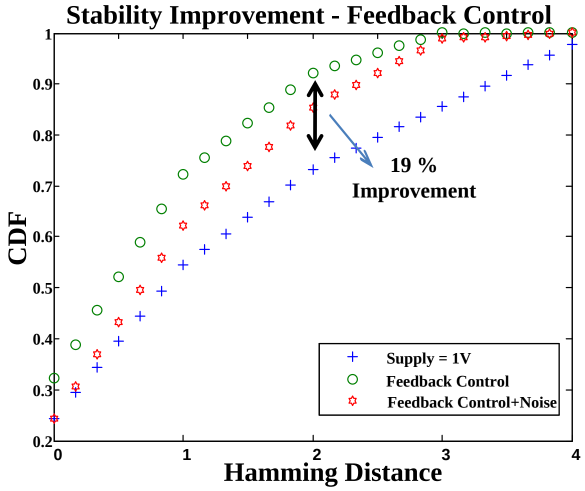


Figure 4.5: Stability - Feedback-based voltage control

PUF characteristics to an adversary.

3. **Operation Phase:** To an external user who authenticates an IC based on the PUF response, the feedback-based PUF operates in the same way as a traditional ROPUF. The only difference is that internally the feedback-based ROPUF would automatically vary its supply voltage to produce the most stable response at a given temperature.

4.3.2 Results

The PUF shown in figure 4.4 was implemented using the simulation based environment mentioned in the methodology section. The ring oscillators were implemented with 15 stages of NAND gates. Each ROPUF IC consists of 32 ROs. Three different voltages of 1.1V, 1V and 0.9V were provided as the switchable supply voltages. The temperature was varied between 15 °C to 65 °C at intervals of 10 °C. For each IC instance the supply voltage producing the best stability at each temperature bin was characterized. All experiments were carried using the experimental setup detailed in Section 3.1.

Figure 4.5 shows the results for the stability experiment. The bottom curve shows the cumulative density function of the number of error bits in the digital signatures at various temperatures, under the base case of a ROPUF IC operating at 1V supply voltage. For example, the probability of two or less number of error bits in the digital signature is 0.73. The CDF curves for ROPUFs operating at 0.9V and 1.1V lies below the curve for 1V and are not reported in the graph for brevity. The top curve shows the CDF of the number error bits for the PUF IC consisting of the novel feedback-based architecture. It is observed that the probability of two or less number of error bits in the digital signature shoots up to 0.92. This is a 19% increase in this probability. It is also observed that the novel feedback-based control method outperforms the traditional single supply based ROPUFs across all points in the curve.

The effect of noise in supply voltage of the PUFs is considered as well. The middle curve in Figure 4.5 represents the worst case CDF for the error probability when noise is induced to the three supply voltages of the feedback-based PUF architecture, by up to 10% of its nominal value. In spite of this high supply voltage noise, the feedback-based approach outperforms the traditional ROPUF operated at a supply voltage of 1V void of noise. For example, the probability for two or less number of error bits is 0.87 for the feedback-based approach induced with noise. This is still a 14% improvement over the traditional ROPUF at 1V.

The decreased probability in obtaining an error in the output code by employing the feedback scheme translates to a smaller hardware for generating a fixed amount of error free bits from the PUF. To get an idea of the hardware savings, we will consider the repetition code error correcting algorithm [22]. Based on our calculations from the previous data we obtained that the probability for a given bit to be wrong is 0.031 for the feedback-based approach and 0.053 for the traditional ROPUF. Using this data, from [22] we obtain that if 171 error free bits are needed, then the feedback-based ROPUF would require 928 sources bits whereas traditional ROPUF would need 1600 sources bits when a Reed Muller error correcting code is used. This translates to about 40% decrease in PUF hardware when the feedback-based scheme is used.

Also, this novel approach is more efficient than the state of the art temperature variation tolerant ROPUF proposed in [29] and [54]. [29] proposes a redundant based architecture of using multiple (8 in the paper) ring oscillators to produce one single bit. This method achieves a best case hardware utilization of 25 %. [54] improves over this method by a factor of 80% using a cooperative ring oscillator approach. Our method almost uses 100% hardware utilization as no bits are wasted in redundancy.

However, the control architecture presents an area overhead. The temperature sensor consists of 30 transistors, and has similar area as a 15-stage RO used in the ROPUF. Since, we use 10 different temperature levels, and 3 voltage levels in this work, a 30-bit ROM would be required. The area occupied by the 30-bit ROM would be equivalent to 2 15-stage RO structures. The only other overhead in our architecture is the need for three different voltage levels and the corresponding power gating transistors. We assume the different supply voltages are provided from off-chip, therefore the area penalty is minimum. Therefore, the overhead due to the control architecture is equivalent to 3 RO structure for a ROPUF consisting of 32 ROs. This is roughly 10% of the total area. However, as the size of the ROPUF grows, the overhead due to the control architecture is going to remain constant, and therefore it would consume a very small area in comparison to the entire ROPUF itself. Therefore, the hardware penalty would be minimal.

Since the stability towards temperature variation is improved by the feedback-based supply voltage control scheme and the hardware penalty is minimal, it offers a promising method for improving ROPUFs.

4.4 Conclusion

In this chapter, we presented a feedback-based supply voltage control scheme to enhance stability of ring oscillator PUFs. Results indicate that the probability of having error in PUF output due to temperature variation is decreased by 19% by using this approach. Also, the PUF hardware requirements could be reduced by 40% due to the increased stability offered by the novel approach for achieving a fixed number of error free output bits.

Chapter 5

Conclusion and Future Work

Silicon PUFs are novel chip identifiers, which produce a unique response per IC, based on the slight variations in process characteristics of identically designed devices. In the initial sections of this thesis, PUFs were classified based on the fabrication techniques and the type of deployment. Further, the sources of variation in CMOS PUFs were explained.

Later, we showed that it is important to optimize PUFs considering the various conflicting design goals, as the characteristics of the PUFs drastically affects the performance of security system, which employ such novel identifiers. In this regard, we classified the various conflicting design goals as the following: fast authentication, low power operation and stable response. Among the design goals the stability is the most important criterion. This is because the PUFs are not only affected by process characteristics but are also affected by varying environmental conditions like temperature, supply voltage noise, etc. Thus, the output of PUF consists of certain noisy bits. Such noisy responses lead to a penalty in area and power. This is because, better error correcting schemes and the extra sources bits are required to produce error free bits. As our first contribution, we proposed using circuit techniques to improve the popular delay-based PUF architectures i.e. ROPUF and Arbiter PUFs, with a focus on stability. Initially, we explored optimizations of circuit topology including transistor sizing and the number of stages of the ring oscillator. We showed that the benefits offered by optimizations in circuit topology are limited.

Further, we proposed using subthreshold voltage operation of delay-based PUFs, to further improve their characteristics. Traditionally, subthreshold circuits are used only for extremely low power applications and they suffer from the exaggerated effect of process variation in this region. We showed that at the subthreshold region, the PUF circuit will be affected more by random process variation and less by environmental changes. Thereby, PUFs become more stable when operated in the subthreshold region, unlike traditional digital CMOS circuits which are less stable. Also, operating at the subthreshold region would require smaller PUF hardware due to its increased stability and it enjoys higher energy efficiency as well. However, all these benefits are achieved at the cost of ten times reduction in speed of operation. Thereby, we showed that subthreshold operation of delay-based PUF helps in achieving a favorable combination of the various design goals for applications with low power and area requirement, and relaxed performance constraints.

As our second contribution, we proposed a feedback-based supply voltage control technique and a corresponding architecture, to improve the stability of ROPUFs against variations in temperature. In this scheme we varied the supply voltage of the PUFs based on the operating temperature of the PUF IC. The optimal supply voltage to be applied to the PUF at each operating temperature is identified during the evaluation stage of the PUF and is stored in form of a micro-code. Results indicate that this technique helps in decreasing the bit error probability from 5% to 3%.

Overall, the focus of this thesis was to improve the novel on chip identifiers - Physical Unclonable Functions (PUFs) holistically.

5.0.1 Future Work

We have designed a subthreshold ROPUF ASIC and have sent it out for tapeout, to prove the effectiveness of subthreshold voltage operation on ROPUFs in real silicon. A 130nm

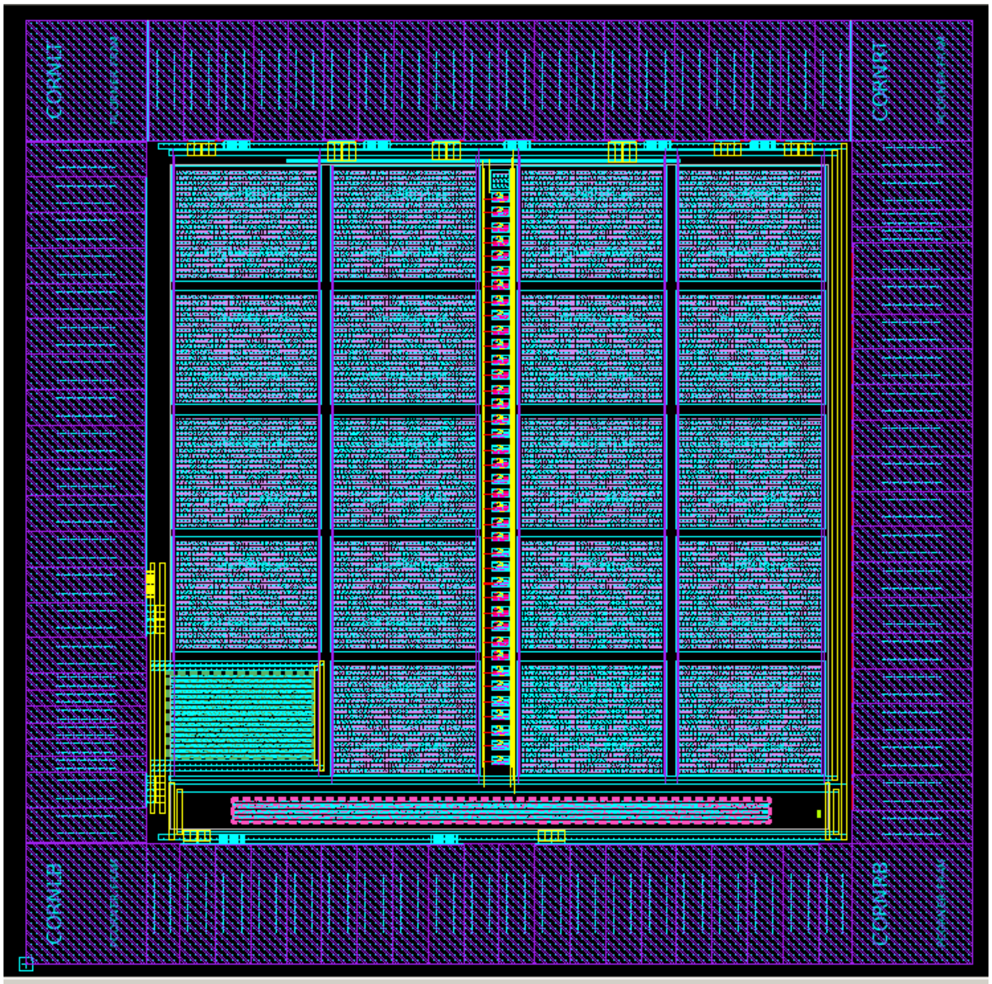


Figure 5.1: Layout - Subthreshold ROPUF ASIC

process technology was used for fabrication. The IC is capable of operating at both the superthreshold and the subthreshold voltage regions. Figure 5.1 shows the final gdsii layout of the IC sent for fabrication. It has 19 identical PUF blocks, each consisting of 2 ROPUF structures. Each ROPUF structure is made of 32 ring oscillators and the corresponding multiplexers and counter modules.

The 2 different ROPUF structures differ in the type of oscillators used. One of the ROPUF structure is made of a traditional 21 stage NAND gate chain and the other consists of 3 stage current starved inverter based ring oscillators. The counter and the multiplexer blocks are implemented to be operated in the subthreshold region. Also, asynchronous ripple counters are used to minimize the effect of the metastability issues causing erroneous outputs in flip flops.

A centralized controller, which is present in the bottom of the IC, is used to serially interface with external world and evaluate the corresponding PUF blocks. The controller operates in the superthreshold region, to achieve maximum performance. Hence, level converters are inserted between the interfaces from PUF blocks to the controller. The level converters are present in the central vertical rail in the IC as shown in the layout.

The results from this IC would enable to completely characterize the effect of distance based variation on PUF outputs. Also, the results would help in establishing the advantages of subthreshold operation of ROPUF, on silicon. The IC also consists of current starved oscillator based ROPUFs, which are novel PUF structures. Overall, the IC would help in further improving and tuning the results from simulation based analysis of PUF structures and prove the effectiveness of circuit level techniques to enhance delay based PUFs.

Bibliography

[1] S. P. Skorobogatov and R. J. Anderson, "Optical fault induction attacks," vol. 2523 of *Lecture Notes in Computer Science*, pp. 31–48, Springer Berlin / Heidelberg, 2003.

[2] L. Goubin and J. Patarin, "Des and differential power analysis the duplication method," in *Cryptographic Hardware and Embedded Systems* (e. Ko and C. Paar, eds.), vol. 1717 of *Lecture Notes in Computer Science*, pp. 728–728, Springer Berlin / Heidelberg, 1999.

[3] "Cisco statement on counterfeit goods," *Cisco Corp.,San Jose*, 2001.

[4] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," vol. 4727 of *Lecture Notes in Computer Science*, pp. 63–80, Springer Berlin / Heidelberg, 2007.

[5] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Test Conference, 2004. Proceedings. ITC 2004. International*, pp. 339–344, Oct. 2004.

[6] D. Lim, "Extracting secrets keys from integrated circuits," *Master of Science Thesis,Massachusetts Institute of Technology*, 2004.

[7] A. Juels and S. A. Weis, "Authenticating pervasive devices with human protocols," pp. 293–308, Springer-Verlag, 2005.

[8] M. Feldhofer and C. Rechberger, "A case against currently used hash functions in rfid protocols," in *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*

(R. Meersman, Z. Tari, and P. Herrero, eds.), vol. 4277 of *Lecture Notes in Computer Science*, pp. 372–381, Springer Berlin / Heidelberg, 2006.

[9] I. McLoughlin, "Secure embedded systems: The threat of reverse engineering," in *Parallel and Distributed Systems, 2008. ICPADS '08. 14th IEEE International Conference on*, pp. 729 –736, 8-10 2008.

[10] G. E. Suh, D. Clarke, B. Gassend, M. van Dijk, and S. Devadas, "Aegis: architecture for tamper-evident and tamper-resistant processing," in *ICS '03: Proceedings of the 17th annual international conference on Supercomputing*, (New York, NY, USA), pp. 160–171, ACM, 2003.

[11] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*, (New York, NY, USA), pp. 148–160, ACM, 2002.

[12] R. J. Anderson, "Security engineering: A guide to building dependable distributed systems," *John Wiley and Sons*, 2001.

[13] R. Pappu, "Physical one-way functions," *PhD thesis,Massachusetts Institute of Technology*, 2001.

[14] B. Skoric, G.-J. Schrijen, P. Tuyls, T. Ignatenko, and F. Willems, "Secure key storage with PUFs," pp. 269–292, Springer London, 2008.

[15] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola, and V. Khandelwal, "Design and implementation of PUF-based "unclonable" RFID ICs for anti-counterfeiting and security applications," in *RFID, 2008 IEEE International Conference on*, pp. 58–64, Apr. 2008.

[16] H. Busch, S. Katzenbeisser, and P. Baecher, "Puf-based authentication protocols revisited," in *Information Security Applications* (H. Youm and M. Yung, eds.), vol. 5932 of *Lecture Notes in Computer Science*, pp. 296–308, Springer Berlin / Heidelberg, 2009.

[17] E. Simpson and P. Schaumont, "Offline hardware/software authentication for reconfigurable platforms," in *Cryptographic Hardware and Embedded Systems - CHES 2006* (L. Goubin and M. Matsui, eds.), vol. 4249 of *Lecture Notes in Computer Science*, pp. 311–323, Springer Berlin / Heidelberg, 2006.

[18] L. Kulseng, Z. Yu, Y. Wei, and Y. Guan, "Lightweight secure search protocols for low-cost rfid systems," in *ICDCS '09: Proceedings of the 2009 29th IEEE International Conference on Distributed Computing Systems*, (Washington, DC, USA), pp. 40–48, IEEE Computer Society, 2009.

[19] G. Hammouri and B. Sunar, "PUF-HB: A tamper-resilient HB based authentication protocol," vol. 5037 of *Lecture Notes in Computer Science*, pp. 346–365, Springer Berlin / Heidelberg, 2008.

[20] R. Maes, P. Tuyls, and I. Verbauwhede, "Low-overhead implementation of a soft decision helper data algorithm for sram pufs," in *CHES '09: Proceedings of the 11th International Workshop on Cryptographic Hardware and Embedded Systems*, (Berlin, Heidelberg), pp. 332–347, Springer-Verlag, 2009.

[21] M.-D. M. Yu and S. Devadas, "Secure and robust error correction for physical unclonable functions," *IEEE Design and Test of Computers*, vol. 27, pp. 48–65, 2010.

[22] C. Bosch, "Efficient fuzzy extractors for reconfigurable hardware," *Masters Thesis - Ruhr-University Bochum,Germany*, 2008.

[23] B. Zhai, S. Hanson, D. Blaauw, and D. Sylvester, "Analysis and mitigation of variability in subthreshold design," in *ISLPED '05: Proceedings of the 2005 international symposium on Low power electronics and design*, (New York, NY, USA), pp. 20–25, ACM, 2005.

[24] M. B. Henry and L. Nazhandali, "Hybrid super/subthreshold design of a low power scalable-throughput FFT architecture," vol. 5409 of *Lecture Notes in Computer Science*, pp. 278–292, Springer Berlin / Heidelberg, 2009.

[25] V. Vivekraja and L. Nazhandali, "Circuit-level techniques for reliable physically uncloneable functions," *Hardware-Oriented Security and Trust, IEEE International Workshop on*, vol. 0, pp. 30–35, 2009.

[26] J. Wu and M. O'Neill, "On foundation and construction of physical unclonable functions." Cryptology ePrint Archive, Report 2010/171, 2010.

[27] N. Beckmann and M. Potkonjak, "Hardware-based public-key cryptography with public physically unclonable functions," in *Information Hiding* (S. Katzenbeisser and A.-R. Sadeghi, eds.), vol. 5806 of *Lecture Notes in Computer Science*, pp. 206–220, Springer Berlin / Heidelberg, 2009.

[28] U. Rhrmair, F. Sehnke, J. Slter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on physical unclonable functions." Cryptology ePrint Archive, Report 2010/251, 2010.

[29] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *DAC '07: Proceedings of the 44th annual conference on Design automation*, (New York, NY, USA), pp. 9–14, ACM, 2007.

[30] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Controlled physical random functions," in *In Proceedings of the 18th Annual Computer Security Conference*, 2002.

[31] B. Gassend, D. Lim, D. Clarke, M. van Dijk, and S. Devadas, "Identification and authentication of integrated circuits: Research articles," *Concurr. Comput. : Pract. Exper.*, vol. 16, no. 11, pp. 1077–1098, 2004.

[32] S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen, and P. Tuyls, "Extended abstract: The butterfly puf protecting ip on every fpga," in *Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on*, pp. 67 –70, 9-9 2008.

[33] P. Tuyls, B. skoric, S. Stallinga, A. Akkermans, and W. Ophey, "Information-theoretic security analysis of physical uncloneable functions," in *Financial Cryptography and Data*

Security (A. S. Patrick and M. Yung, eds.), vol. 3570 of *Lecture Notes in Computer Science*, pp. 141–155, Springer Berlin / Heidelberg, 2005.

[34] B. skoric, P. Tuyls, and W. Ophey, "Robust key extraction from physical uncloneable functions," in *Applied Cryptography and Network Security* (J. Ioannidis, A. Keromytis, and M. Yung, eds.), vol. 3531 of *Lecture Notes in Computer Science*, pp. 407–422, Springer Berlin / Heidelberg, 2005.

[35] E. Ozturk, G. Hammouri, and B. Sunar, "Physical unclonable function with tristate buffers," in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, pp. 3194 –3197, 18-21 2008.

[36] L. Chang, D. Fried, J. Hergenrother, J. Sleight, R. Dennard, R. Montoye, L. Sekaric, S. McNab, A. Topol, C. Adams, K. Guarini, and W. Haensch, "Stable sram cell design for the 32 nm node and beyond," in *VLSI Technology, 2005. Digest of Technical Papers. 2005 Symposium on*, pp. 128 – 129, 14-16 2005.

[37] D. Puntin, S. Stanzione, and G. Iannaccone, "Cmos unclonable system for secure authentication based on device variability," in *Solid-State Circuits Conference, 2008. ESSCIRC 2008. 34th European*, pp. 130 –133, 15-19 2008.

[38] "Attack resistant sense amplifier based pufs (sa-puf) with deterministic and controllable reliability of puf responses," in *Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on*, pp. 106 –111, 13-14 2010.

[39] X. Wang and M. Tehranipoor, "Novel physical unclonable function with process and environmental variations," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, pp. 1065 –1070, 8-12 2010.

[40] R. Helinski, D. Acharyya, and J. Plusquellic, "A physical unclonable function defined using power distribution system equivalent resistance variations," in *Design Automation Conference, 2009. DAC '09. 46th ACM/IEEE*, pp. 676 –681, 26-31 2009.

[41] T. Mudge, K. Flautner, D. Blaauw, and S. M. Martin, "Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads," *Computer-Aided Design, International Conference on*, vol. 0, pp. 721–725, 2002.

[42] L. Wei, Z. Chen, K. Roy, M. Johnson, Y. Ye, and V. De, "Design and optimization of dual-threshold circuits for low-voltage low-power applications," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 7, pp. 16 –24, mar 1999.

[43] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low power cmos digital design," *IEEE Journal of Solid State Circuits*, vol. 27, pp. 473–484, 1995.

[44] J. Meindl and J. Davis, "The fundamental limit on binary switching energy for terascale integration (tsi)," *IEEE Journal of Solid State Circuits vol 35.*, 2002.

[45] H.-i. Kim and K. Roy, "Ultra-low power dlms adaptive filter for hearing aid applications," in *ISLPED '01: Proceedings of the 2001 international symposium on Low power electronics and design*, (New York, NY, USA), pp. 352–357, ACM, 2001.

[46] N. Jayakumar and S. P. Khatri, "A variation tolerant subthreshold design approach," in *DAC '05: Proceedings of the 42nd annual Design Automation Conference*, (New York, NY, USA), pp. 716–719, ACM, 2005.

[47] W. Alice and C. Anantha, "A 180-mv subthreshold fft processor using a minimum energy design methodology," *IEEE J. Solid-State Circuits*, vol. 40, pp. 310–319, 2005.

[48] T. Douseki, M. Harada, and T. Tsuchiya, "Ultra-low-voltage mtcmos/simox technology hardened to temperature variation," *Solid-State Electronics*, vol. 41, pp. 519–525, 1997.

[49] Y. Leblebici, "Design considerations for cmos digital circuits with improved hot-carrier reliability," *Solid-State Electronics,IEEE Journal of*, vol. 31, pp. 1014–1024, 1996.

[50] A. H. Baba and S. Mitra, "Testing for transistor aging," *VLSI Test Symposium, IEEE*, vol. 0, pp. 215–220, 2009.

[51] V.Huard, M.Denais, and C.Parthasarathy, "Nbti degradatin : From physical mechanisms to modelling," *Microelectronics and reliability*, vol. 46, pp. 1–23, 2006.

[52] X. Chen, Y. Wang, Y. Cao, Y. Ma, and H. Yang, "Variation-aware supply voltage assignment for minimizing circuit degradation and leakage," in *ISLPED '09: Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design*, (New York, NY, USA), pp. 39–44, ACM, 2009.

[53] A. Calimera, E. Macii, and M. Poncino, "Nbti-aware sleep transistor design for reliable power-gating," in *GLSVLSI '09: Proceedings of the 19th ACM Great Lakes symposium on VLSI*, (New York, NY, USA), pp. 333–338, ACM, 2009.

[54] C.-E. Yin and G. Qu, "Temperature-aware cooperative ring oscillator puf," *Hardware-Oriented Security and Trust, IEEE International Workshop on*, vol. 0, pp. 36–42, 2009.

[55] B. Datta and W. Burleson, "Low-power, process-variation tolerant on-chip thermal monitoring using track and hold based thermal sensors," in *GLSVLSI '09: Proceedings of the 19th ACM Great Lakes symposium on VLSI*, (New York, NY, USA), pp. 145–148, ACM, 2009.