An Effort toward Building more Secure and Efficient Physical Unclonable Functions

Dinesh Ganta

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Engineering

Leyla Nazhandali, Chair
Patrick R. Schaumont
Sandeep Shukla
Inyoung Kim
Mohammad Tehranipoor

December 8th, 2014
Blacksburg, Virginia

Keywords: Physical Unclonable Functions, Security, Fingerprinting, Integrated Circuits, Identifiers

An Effort toward Building more Secure and Efficient
Physical Unclonable Functions

Dinesh Ganta

ABSTRACT

Over the last decade, there has been a tremendous growth in the number of electronic devices and applications. One of the very important aspects to deal with such proliferation of ICs is their security. Establishing the Identity (ID) of a device is the cornerstone of any secure application. Typically, the IDs of devices are stored in non-volatile memories (NVM) or through burning fuses on ICs. However, through such traditional techniques, IDs are vulnerable to attacks. Further, maintaining such secrets in NVMs is expensive.

Physical Unclonable Functions (PUF) provide an alternative method for creating chip IDs. They exploit the uncontrollable variations that exist in IC manufacturing to generate identifiers. However, since PUFs exploit the small mismatch across identically designed circuits, the responses of PUFs are prone to error in the presence of unwanted variations in the operating temperature, supply voltage, and other noises. The overarching goal of this work is to develop silicon PUFs that are highly efficient and stable to such noises. In addition, to make PUFs more attractive for low cost and tiny embedded systems, our goal is to develop PUFs with minimal area and power consumption for a given ID length and security requirement.

Techniques to develop such PUFs span different abstraction levels ranging from technology-independent application-level techniques to technology-dependent device-level ones. In this dissertation, we present different technology-independent and technology-dependent techniques and evaluate which techniques are good candidates for improving different qualities of PUFs.

In technology-independent techniques, we propose two modifications to a conventional PUF architecture, which are detailed in this thesis. Both modifications result in a PUF that is more efficient in terms of area and power. Compared to the traditional architecture, for a

given silicon real estate, the proposed architecture provides over two orders of magnitude larger C/R space and it has higher resistance toward modeling attacks.

Under technology-dependent methods, we investigate multiple techniques that improve stability and efficiency of PUF designs. In one approach, we propose a novel PUF design with a similar architecture to that of a traditional design, where we replace large and power hungry digital components with more efficient analog components. In another technique, we exploit the differences between pMOS and nMOS transistors in their variation of threshold voltage (V_{th}) and in the temperature coefficients of V_{th} to significantly improve the stability of bistable PUFs. We also use circuit-level simulations to evaluate the stability of silicon PUFs to aging degradation.

We believe that our technology-independent techniques are good candidates for improving overall efficiency of PUFs in terms of both operation and implementation costs, suitable for PUFs with tight constraints on cost for design and test. However, with regards to improving the stability of PUFs, it is cost-effective to use our technology-dependent techniques as long as the extra effort for implementation and testing can be tolerated.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In practical cryptography, a Physical Unclonable Function, or PUF, is a function that is embodied in a physical structure, which has to be easy to evaluate but hard to characterize. PUFs have received a lot of attention in Challenge/Response (C/R) authentication schemes and secure key generation for public and private key cryptography. PUFs have demonstrated significant potential to provide secure alternative to the conventional techniques like storing secret keys and identities (ID) in non-volatile memories or fuses, which are prone to attacks [22, 23]

Among the several technologies proposed for building PUFs, silicon PUFs provide the most affordable solution for hardware secure design. Unlike optical PUFs and coating PUFs, silicon PUFs can be manufactured with the same process and at the same time as the rest of the Integrated Circuit (IC). Silicon PUFs are possible due to the fact that IC manufacturing process is imperfect and can never produce the exact same result as intended at the time of design. This imperfection results in random unintended variations in the manufactured ICs that can be captured using intelligent design. In the rest of this document, for brevity, we refer to a silicon PUF as PUF.

An ideal PUF can be viewed as a Challenge/Response (C/R) function, where for a given

Challenge, C, the Response, R, is random, is unique to the chip, and remains invariant to variations in the operating temperature and noises. However, in real circuits, some correlation exists between the responses from different PUFs. Further, some of the PUF responses become unstable because of varying operating conditions. Hence, the quality of a PUF architecture is typically measured with the help of metrics such as $Variability$ and $Stability$.

$Variability$ is the quality of being unique in a population of PUFs. In other words, given a population of ICs with PUFs, variability is a metric that tells us how uniquely an IC can be distinguished from the rest of the ICs. Typically, variability is observed with the use of histogram of the pairwise inter-chip hamming distances (HD) between the IDs of all the PUFs in the population. PUF's responses should have high *variability* as low variability means many PUFs have highly similar responses, which compromises the entire security application they are part of.

Stability is the quality of producing consistent output in spite of the unwanted variations in the supply voltage, temperature, or noises in the environment. It is typically measured as the intra-chip hamming distance between the responses of a PUF measured at different operating conditions. Although a stability of 100% is expected, in practical circuits, some of the PUF bits flip due to varying operating conditions. We will revisit $Variability$ and *Stability* metrics, and their methodology of evaluation in the later chapters along with the proposed techniques.

Stability requirements of a PUF depends on the application they are part of. For example, when PUFs are used as secret keys in cryptographic cores, they are required to have 100% stability, whereas in applications like Challenge/Response (C/R) authentication, certain number of bit flips can be tolerated. Hence, based on the application requirements, some form of Error Correction Codes (ECC) or helper data algorithms (HDA) are typically employed to achieve desired stability [8, 15, 28, 50]. However, the area and power overhead associated with ECC increases significantly with decrease in the inherent stability of PUF bits. Hence, it is desired to have a PUF that is inherently very stable such that a light weight ECC can

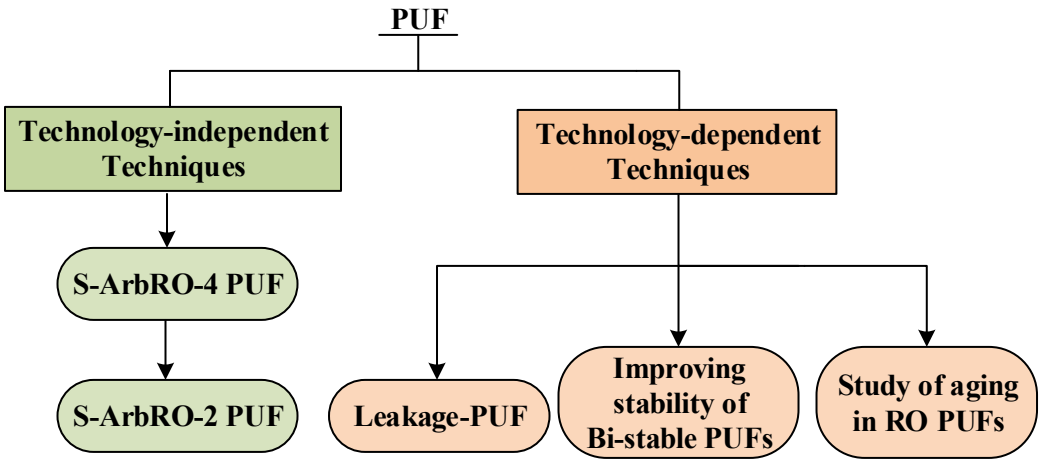

Figure 1.1: Classification of this dissertation

be sufficient in achieving required stability.

When PUFs are used in resource-constrained devices such as most embedded systems that cannot afford a full-fledged cryptographic-based authentication, it is necessary that the authentication mechanism is light-weight, where PUF response bits are generated for a series of challenges without hashing and/or extra processing. In such scenarios, the secure lifetime of a device directly depends on the total number of Challenge/Response (C/R) pairs of the PUF. Therefore, it is important to increase the C/R space of the PUF for a given amount of hardware.

In addition to the aforementioned, PUFs are required to be resistant to modeling attacks without which, the secure lifetime of a PUF can be limited. Modeling attacks are a class of security attacks, where the attacker develops mathematical models for PUFs based on a set of observed C/R pairs [16,36]. Typically, it is assumed that the attacker has the knowledge of the architecture of the PUF. It is particularly important for PUFs that do not use hashing or other forms of information hiding on their response bits to be resistant to modeling attacks.

PUF are components of electronic circuits. It is well known that techniques that drive low power consumption in standard ASICs leverage optimizations that spread across the

entire hierarchy of computation e.g. compiler optimizations, power-gating, clock-gating, body biasing, etc. Similarly, techniques to improve the qualities of PUFs mentioned above can span across different abstraction levels in the design of PUFs. In this dissertation, we broadly classify such techniques into technology-dependent and technology-independent efforts. Based on our work, we evaluate which techniques are better candidates for improving different qualities of PUFs.

Fig. 1.1 presents an overview of this dissertation with the proposed works classified as technology-independent and technology-dependent techniques. In technology-independent techniques, we propose a novel PUF architecture that brings together the positive traits of both Arbiter and Ring Oscillator PUFs. Our proposed PUF is easy to build, has high C/R space in a given area, and is also resistant to modeling attacks.

In technology-dependent techniques, we first propose a PUF that uses a traditional RO PUF architecture, but is highly stable as it uses efficient analog components that are more sensitive to underlying process variations. The proposed PUF uses leakage sensors as the variation capturing elements. Hence, we name this Leakage-PUF or L-PUF. Following that, we propose a stability enhancement technique for bi-stable PUFs. In this technique, the differences between pMOS and nMOS transistors in their variation and in the temperature coefficient of their threshold voltages are exploited. This results in a PUF that is much more stable to temperature variations. Finally, since aging has become one of the major reliability concerns in integrated circuits, we study the impact of aging on RO PUFs for different circuit-level choices and operating conditions.

Based on our work, we observe that technology-independent works are cost-effective in terms of both implementation and operation. They are easier to realize as compared to technology-dependent works as they involve standard design flows, which can be implemented in both ASICs and FPGAs. With the help of innovative architectures, large C/R space can be generated in a given area, which can result in lower operation costs. Further, techniques in technology-independent abstraction level are good candidates to make PUFs resistant to

modeling.

On the other hand, to improve stability of PUFs, we believe that our technology-dependent techniques are good candidates. This is because stability of PUF responses is tightly correlated with underlying circuit-level characteristics. Hence, technology-dependent techniques such as proposing circuits that exploit process variation better, circuit-level hacks to better tolerate unwanted variations in temperature and supply voltage, etc., can result in better stability in PUFs. However, the implementation cost of technology-dependent techniques is relatively higher as it requires custom ASIC designs.

The rest of this dissertation is organized as follows. In Chapter 2, sources of variation that are present in IC manufacturing process are presented along with a classification of prominent PUF architectures. Technology-independent techniques for improving PUFs are discussed in Chapter 3. In Chapter 4, different technology-dependent techniques proposed for the development of PUFs are presented. Conclusion of this dissertation is presented in Chapter 5.

Chapter 2

Background

In this chapter, we will first present the sources of variation in PUFs. Following that, a basic classification of PUFs is provided along with a brief description of well-known PUF architectures, which we will revisit in later chapters.

2.1 Sources of variation

PUFs are possible because of the uncontrollable variations that exist in the CMOS process. After fabrication, the quality of a PUF depends on its operating conditions. In order to design new PUF architectures, or to improve upon existing ones, we need to understand the sources of variability. So, in this section, we will present the major sources of variation in a PUF. We distinguish several different sources in Figure 2.1.

- **Manufacturing Process Variations (MPV):** These are random and permanent deviations from the designed, nominal value of a circuit structure, caused by random effects during manufacturing [18]. MPVs can be separated into two categories: The first category covers variations in process parameters, such as impurity concentration densities, oxide thicknesses, and diffusion depths. These result from non-uniform con-

ditions during the deposition and/or the diffusion of dopants. The second category covers variations in the dimensions of devices. These result from limited resolution of the photo-lithographic process, which in turn causes width and length variations in transistors.

- **Environmental Variations (EV):** These are temporary variations caused by changes in the environmental parameters, including temperature, operating voltage, and external noise coupling. As the PUF environment cannot always be controlled, the effect of these variations should be minimized. For example, at nominal operation, an increase in temperature or a decrease in power supply, will slow down a circuit. Moreover, the performance loss is not linear and it may affect different parts of a circuit differently. This will affect the stability of the PUF.

- **Aging:** Aging is one of the major reliability issues in integrated circuits. Aging is primarily due to the changes that occur in the gate dielectric due to elevated electric fields and switching, over the lifetime of a device. Two of the major sources of aging are Negative Bias Temperature Instability (NBTI) and Hot Carrier Injection (HCI). Although some of the aging degradation is recovered when the devices are turned off, there is a permanent degradation, which deteriorates circuit performance. Aging results in slower operation of circuits, irregular-timing characteristics, increase in power consumption and sometimes even in functional failures [3, 9, 13]. We will revisit aging in more detail in Chapter 4.3.

2.2 Classification of PUFs

PUFs exploit variation in identical circuit elements to generate chip specific IDs or responses. Based on the circuit characteristics that they exploit, most prominent PUF designs can be broadly classified into delay-based PUFs or bi-stable PUFs.

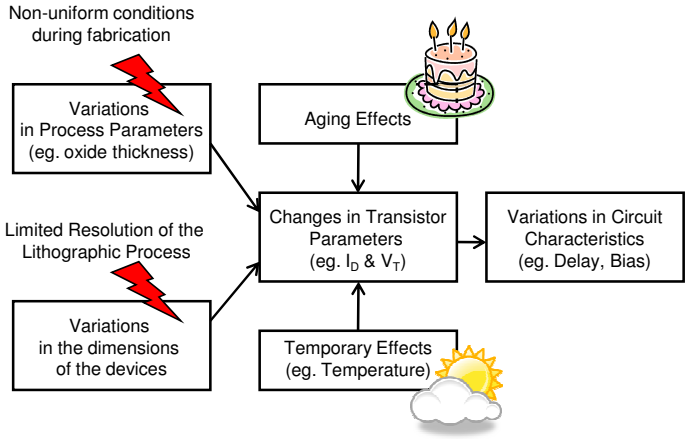

Figure 2.1: Causes of variability in circuit characteristics.

2.2.1 Delay-based PUFs

Delay-based PUFs are a class of PUFs that exploit the variation in the delay across identical circuit elements or identical circuit paths to generate PUF bits. Two major architectures that exploit path delay variation are Arbiter PUF [16] and RO PUF [42]. Based on these architectures, many other variations have been proposed that improve some quality of PUFs [11, 32, 35, 48]. In the following subsections, the architectures of Arbiter PUF and RO PUF are presented as they form the basis for different techniques proposed in this thesis.

Arbiter PUF (Arb PUF)

An Arb PUF consists of N identical switching elements connected in series where each element has two pairs of identical paths. Fig. 2.2a and 2.2b show the basic architecture of an Arb PUF and its MUX based switching element, respectively. Based on the MUX select line (challenge bit, c) to each element, signals A and B traverse a pair of the paths for that element. When a challenge (N-bit) is given to the PUF, signals A and B traverse through the switching elements in symmetrical paths before reaching an arbiter, which typically is a D flip-flop (DFF). If we label the delays of the four paths in each element as d_1, d_2, d_3 and d_4, in essence, each element calculates a variable, s_i, as $d_1 - d_2$ or $d_3 - d_4$ based on the

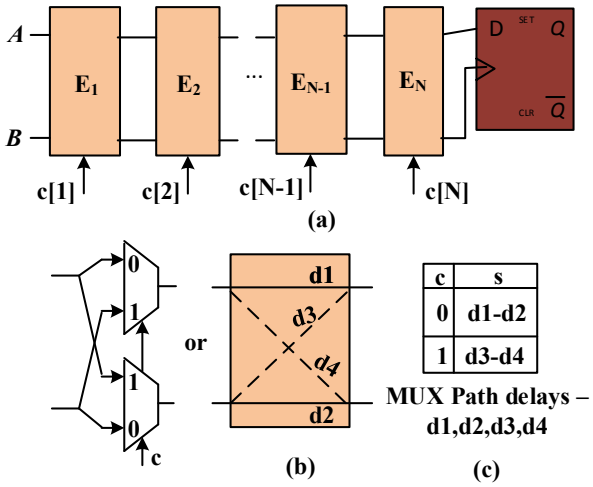

Figure 2.2: Arbiter PUF (a) Architecture. (b) MUX switch. (c) Effective contribution of a MUX switch to signal delay.

input challenge bit, c_i, as shown in Fig. 2.2c. The final binary response (1 bit) is evaluated based on when the signals reach DFF, or alternatively, the sign of Σs_i. An Arb PUF with N-elements (N-bit challenge) has a C/R space of 2^N. The advantages and disadvantages associated with Arbiter PUFs are discussed in more detail later in Section 3.2.1.

Ring Oscillator PUF (RO PUF)

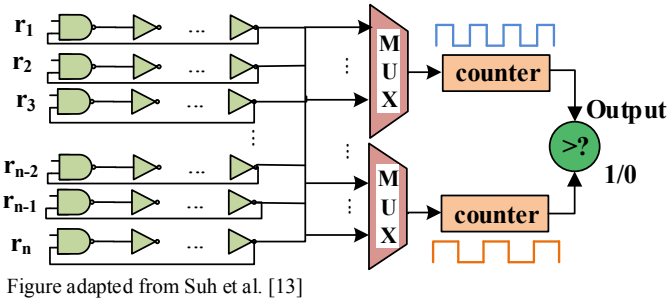

Figure 2.3: Ring Oscillator PUF

Fig. 2.3 illustrates the basic architecture of an RO PUF. In an RO PUF, the characteristic frequencies of identical ring oscillators (RO) are compared to generate random bits. For any

two chosen ROs, as they are identical at design time, they are supposed to oscillate at same frequency. However, due to random process variations, one of the oscillators is going to be faster than the other. The frequency comparison of these two ROs result in 1-bit of random data. An ID of required length is created by challenging the PUF with as many pairs of ROs.

2.2.2 Bi-stable PUFs

Bi-stable PUFs are a class of PUFs that exploit the random start-up state of identical cross-coupled circuits, at its core. The input for a bi-stable PUF is the address location of a bi-stable cell and output to such a PUF is 1-bit. Chip IDs of required length are generated by challenging the PUF with as many inputs. [17] and [19] have proposed SRAM PUF that uses random startup values of SRAM cells. [4] uses sense-amplifiers, [41] uses latches, [40] uses buskeepers, [25] uses latches in FPGAs, and [47] uses flip-flops. While some of the bi-stable PUFs proposed are targeted for FPGAs, most of the techniques apply to both ASICs and FPGAs. In the following subsection, the architecture of an SRAM PUF is presented.

SRAM PUF

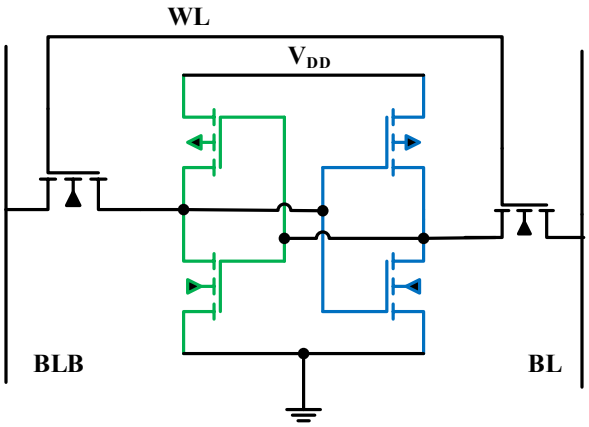

Figure 2.4: SRAM PUF

One of the well-received bi-stable PUF designs in the literature is an SRAM PUF. An SRAM PUF consists of N identical SRAM cells, where during power-up, each cell evaluates to logic 1 or logic 0 based on the mismatch between the cross- coupled inverters and surrounding noises [19]. Fig 2.4 illustrates a basic 6-T SRAM cell of an SRAM PUF. The inverters of the SRAM cell are typically minimum-sized such that the impact of process variation is high.

Chapter 3

Technology-independent PUF Techniques

In Chapter 1, we have introduced two important qualities of PUFs, *variability* and *stability*. Variability is the quality of being unique in a populations of PUFs, and stability is the quality of producing consistent output irrespective of variations in the operating environment. In addition to these two qualities, depending on PUF's usage and medium of operation, it requires other qualities. One such quality is *ease of manufacturing*. Silicon PUFs are relatively easy to fabricate and several such PUFs have been proposed and fabricated in the previous years. However, not all of them are equal in terms of ease of construction. One of the most easy-to-build silicon PUFs - as witnessed by several successful prototypes in ASIC and FPGA - is Ring Oscillator based PUF, where identical ring oscillators are used to extract device-unique responses.

Most resource-constrained embedded systems cannot afford to have a full-fledged cryptographic-based authentication. In such cases, it is necessary that the authentication mechanism is light-weight, where PUF response bits are generated without hashing or extra processing. The secure lifetime of such systems directly depends on the C/R space of the PUF. Hence, it is important for a PUF to have large C/R space in a given area. Further, the PUF has to

be resistant to modeling attacks.

In this chapter, we propose a novel technology-independent PUF architecture that is easy to build and has a very large Challenge/Response (C/R) space in a given area, while being resistant to modeling attacks. Our proposed design aims at improving the C/R space by bringing together the positive qualities of both Arbiter PUF and RO PUF. Hence, we name this design S-ArbRO PUF (or Select-ArbRO PUF). We study the variability of S-ArbRO PUF, and its stability to temperature variations. In addition, we perform a modeling attack on the PUF to get an insight into its resistance to modeling. For an RO count of 24, the total number of C/R pairs in S-ArbRO PUF is over 63K, compared to 276 for a traditional RO PUF with the same number of ROs. A logistic regression (LR)-based modeling attack requires a training set of over 50K C/R pairs to model S-ArbRO PUF with 96% correct prediction, while an equivalent sized Arbiter PUF breaks at only 250 C/R pairs.

The contributions of this technique are as follows:

- We propose a silicon PUF called S-ArbRO PUF, inspired from the architectures of Arbiter PUF and RO PUF. We present its C/R scheme, and PUF characteristics like variability and stability.
- We propose a modification to S-ArbRO PUF, which doubles the total secure C/R space for a given number of ring oscillators.
- We employ a logistic regression (LR) based modeling attack on S-ArbRO PUF. We show that S-ArbRO PUF improves the C/R space, while being resistant to modeling attacks.

The rest of this chapter is organized as follows: Section 3.1 presents the related work in this area. Section 3.2 introduces the S-ArbRO PUF architecture. In Section 3.3, the methodology of evaluation of PUF's *variability* and *stability* is presented. Results are presented in section 3.4. Finally, conclusion of the proposed architectural technique is presented in Section 3.5.

3.1 Related Work

A number of architectures are proposed in PUF literature, but a lot of them suffer from low C/R space. Arbiter PUFs [26] have exponential C/R space with respect to the number of switching elements but they are easy to model [16], and harder to design. RO PUFs [42] have polynomial number of C/R pairs in the number of ring oscillators, but the number of C/R pairs in a given area is low. Another class of PUFs called bi-stable PUFs have a very small C/R space and are best used as secret keys for crypto applications. Authors of [30], have proposed an enhancement for RO-PUF, in which the Euclidean distance of frequencies of a subset of identical ROs is quantized to maximize the C/R space. Authors of [24] have extracted more C/R pairs from an SRAM-based PUF by controlling the peak time of the word line while writing to an SRAM. Authors of [51] have proposed k-sum PUF which has similar construction as S-ArbRO PUF with respect to its response generation but the work is directed towards providing a novel Error Correction Code (ECC). This work differs from prior work in that it tries to improve the C/R space of the PUF, securely. We use the most time-tested PUF constructs, i.e. ring oscillators to build a PUF with a large C/R space for the same amount of hardware and we show that C/R space is more resistant to modeling as compared to an Arbiter PUF. Many of the techniques that were proposed in the literature to improve arbiter PUFs are still applicable to S-ArbRO PUF.

3.2 Architecture

In this section, we first briefly revisit the architecture of a traditional Arbiter PUF (Arb PUF) as it forms the backbone of our proposed architecture. Following that, we present the architecture of S-ArbRO PUF and highlight its advantages compared to other PUF designs.

3.2.1 Arbiter PUF (Arb PUF)

An Arb PUF consists of N identical switching elements connected in series where each element has two pairs of identical paths. Fig. 2.2a and 2.2b show the basic architecture of an Arb PUF and its MUX based switching element, respectively. An Arb PUF with N-elements (N-bit challenge) has a C/R space of 2^N.

Arb PUF as described in Section 2.2.1, has two major difficulties. First, in terms of their implementation and second, in terms of modeling. As for implementation, designing the PUF such that the two pairs of paths in each switch are identical requires relatively high design effort in ASICs and it is almost impossible in FPGAs. Further, some of the delays of the paths traversed by the signals A and B are so close that they result in unreliable responses due to metastability problems associated with DFFs. Although faster flip-flops with smaller timing constraints can be developed, it demands special design effort and they are not a part of general standard cell libraries. In terms of security with respect to modeling, Arb PUF is prone to modeling attacks.

Arb PUF uses delays of signal paths to generate unique responses. As the path delays can be modeled as the sum of the delays of segments that make up the path, there are only $4N$ unknowns in the PUF, where N is the number of switching elements. The security of the PUF is compromised if the attacker has the knowledge of the path segment delays. Assuming an attacker does not have means to open a PUF circuit and measure individual segment delays, [16] shows a modeling attack on an Arbiter PUF using additive-delay model, where the attacker needs to observe only a small set of C/R pairs to model path segment delays. After training the model, the attacker is able to predict PUF responses with high accuracy. The authors show that an Arbiter PUF with N switching elements can be modeled with only $2N+2$ parameters instead of $4N$.

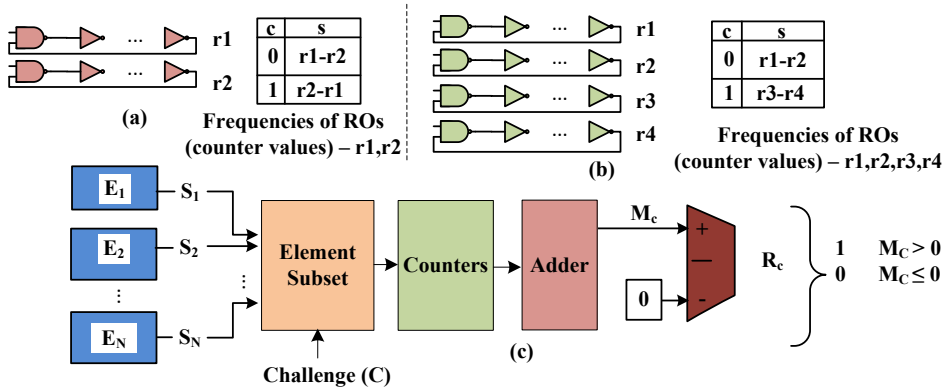

Figure 3.1: S-ArbRO PUF (a) 2-RO element. (b) 4-RO element. (c) Architecture.

3.2.2 Select-ArbRO PUF (S-ArbRO PUF)

Similar to Arb PUF, an S-ArbRO PUF consists of N identical elements. It is named Select-ArbRO PUF as its construction is similar to an Arb PUF but with ring oscillators, and *Select* indicates that only a selectable subset of K ($<=N$) elements of the total N elements is used to generate a PUF response.

The challenge bits in an S-ArbRO PUF are divided into two groups. In the first group of bits, $G1$, the particular combination of K-element subset is encoded. The second group, $G2$, has K bits, which form the challenge for the chosen subset of K elements. For example, for $N = 12$ and $K = 8$, $G1$ has 9 bits to be able to encode 495 (12 choose 8) different possible 8-element combinations and $G2$ has 8 bits, one for each element of the subset. So, a challenge for this PUF is 17 bits long and its response is 1 bit.

The architecture of S-ArbRO PUF has two main differences with Arb PUF, which help overcome the problems that are associated with Arb PUF.

First, S-ArbRO PUF solves both the problems associated with the construction of an Arb PUF by replacing the two pairs of paths in each element with two pairs of identical ring oscillators. Identical ROs can be built easily as macros in both ASICs and FPGAs. Further, if the frequencies of the ROs are very close (similar to a scenario of an Arb PUF, where path

delays are very close), the counters of the ROs can be counted up longer to distinguish them reliably.

Second, each PUF response in an S-ArbRO PUF is only generated by a subset K ($<=N$) of the N elements in the design. The motivation for this method is to improve the number of C/R pairs for the same amount of hardware, while not compromising on the security of the PUF. In section 3.4, we will show that this approach is more resistant to modeling as each response bit doesn't provide information about every element in the PUF.

The following subsections discuss the operation of the two versions of S-AbrRO PUFs, namely S-ArbRO-4 PUF and S-ArbRO-2 PUF.

3.2.3 S-ArbRO-4 PUF

The basic element of an S-ArbRO-4 PUF is built of 4 ROs. The architecture of S-ArbRO-4 PUF and its 4-RO element are illustrated in Fig. 3.1c and 3.1b, respectively. The PUF has N elements, from which, only a subset of K elements are used to generate a PUF response.

Based on the subset $G2$ of the input challenge, each element of the subset calculates either $r_1 - r_2$ or $r_3 - r_4$,where r_j is the value of the counter representing the frequency of jth ring oscillator. Finally, the result of the K elements will be added up to produce M_c, which is compared to 0, in order to generate a 1-bit response, R_c. Thus, each K-element subset has a C/R space of 2^K.

In S-ArbRO-4 PUF, as there are many possible K element subset combinations, it results in an increase in the total C/R space of the PUF. The total C/R space of S-ArbRO-4 PUF is the product of number of possible K-element subset combinations and the number of C/R pairs per subset as shown in Eq. 3.1.

$$Total\ C/R\ space\ of\ S-ArbRO-4\ PUF\ = \frac{N!}{(K! \times (N-K)!} \times 2^K \tag{3.1}$$

3.2.4 S-ArbRO-2 PUF

The problem with S-ArbRO-4 is that it requires $4N$ oscillators, where N is the number of elements. In this section, we present a modification where only two ring oscillators are used in each element, hence the name S-ArbRO-2. The architecture of S-ArbRO-2 PUF is similar to S-ArbRO-4 PUF as shown in Fig. 3.1c but it is made of 2-RO element shown in Fig. 3.1a. Based on the input challenge, either $r_1 - r_2$ or $r_2 - r_1$ is calculated for each element. The downside of this architecture is that half of the challenge space is unusable. This is because, in any chosen subset, the response to a challenge given to the elements ($G2$) is the complement of the response to the complement of that challenge ($G2$'). In other words, in any subset, if the response to a challenge is known, the response to the complement of that challenge is also known. Therefore, for a K-element subset, there are a total of 2^{K-1} C/R pairs. Despite this, in section 3.4, we will show that for a given number of ROs, using a 2-RO element has a larger *secure* C/R space than using a 4-RO element. The total C/R space of S-ArbRO-2 PUF is given by Eq. 3.2. Section 3.4 discusses the differences between S-ArbRO-4 and S-ArbRO-2 in more detail.

$$Total\ C/R\ space\ of\ S-ArbRO-2\ PUF\ = \frac{N!}{(K! \times (N-K)!} \times 2^{K-1} \tag{3.2}$$

3.3 Methodology

Ring Oscillators (RO) form the core of the presented techniques. In our experiments, we have used the RO frequency data reported by [29]. For variability analysis, the authors have implemented 512 ROs on 193 Xilinx 90nm Spartan (XC3S500E) FPGAs, and collected their frequencies at 25C and at nominal supply voltage of 1.2V. For stability analysis, 4 FPGAs were subjected to the temperatures of 65C and 45C at a supply voltage of 1.2V. The detailed implementation and experimental setup of the FPGAs are given by the authors in [29]. In our analysis, if an architecture requires X number of ROs, the first X ROs out of

the 512 ROs reported for each FPGA chip are used to represent each PUF, and the rest are discarded.

3.3.1 Variability

Variability of a population of chips is the quality with which a chip in the population can be distinguished from the rest of the chips in the population. To evaluate variability, Z random challenges are picked. As the challenges are random, the element subset chosen and the challenge bits for the chosen subset are random. These challenges are given to all the M chips in the population. Therefore, we get M number of responses that are each Z bits long. The Z-bit response of each chip is compared to the response of every other chip in the population. This comparison is performed by measuring the Hamming Distances (HD). Variability is represented with the histogram of the obtained HDs. Ideally, the histogram should have a peak around half the length of the response, which is $Z/2$. In our experiments, Z=256 and M=193.

3.3.2 Stability

In this work, stability of PUF's response to temperature variations has been studied. To evaluate stability, randomly selected challenges are given to 4 chips to produce a 256-bit PUF response from each chip. PUF responses are collected for the same set of challenges at temperatures of 25C, 45C, and 65C. For each chip in the population, the response at 25C (reference) is compared with the responses at 45C and 65C, independently.

3.3.3 Modeling

Modeling attacks are a class of attacks, which try to predict the future PUF responses based on an observed set of challenge-response pairs (training set) [36], [33], [16]. In this work,

logistic regression has been used to simulate the modeling attack. We have used logistic regression functions provided by Matlab to implement the attack. For each of the 193 chips in the population, we collect a set of randomly selected Challenge/Response (C/R) pairs from the entire C/R space. This fixed set is called a training set. The model fitting function feeds on the training set and iteratively converges to output a set of coefficients which represents the PUF model. This model is used to predict the responses of the PUF for all challenges that are not part of the training set. The responses predicted by the PUF are compared with the known PUF responses and percentage of the correct prediction is measured. We report an average percentage of the correct prediction among 193 chips for different training set sizes.

In our simulations, we assume the attack has the knowledge of the input challenge grouping. So, from the random challenge given to the PUF, the attacker identifies the element subset and models the corresponding subset. If there are M possible subsets in the PUF, the attack trains M logistic regression models, one for each subset. This is because, if we train the same logistic regression with challenges from different subsets, the coefficients of the model never converge to predict with good probability. Matlab functions, mnrfit and mnrval, have been used to obtain the PUF model coefficients and predict the model's responses, respectively.

3.4 Results

In this section, we first show that an S-ArbRO-2 PUF is more efficient than an S-ArbRO-4 PUF with respect to total C/R space. Following that, we show the rest of the results for S-ArbRO-2 PUF.

Fig. 3.2a shows the C/R space available for one subset of S-ArbRO-2 and S-ArbRO-4 for different number of ROs. For each point on the graph, the number of elements, K, is shown inside parentheses. For both S-ArbRO PUFs with the same number of ROs, the PUF with 2 ROs is more efficient than a PUF with 4 ROs as it has a larger C/R space. Fig. 3.2b shows

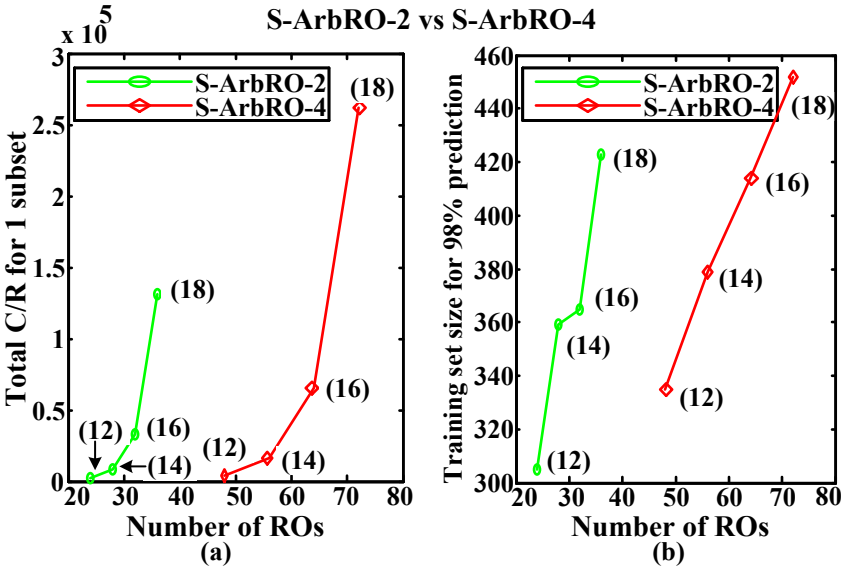

Figure 3.2: S-ArbRO-2 PUF vs. S-ArbRO-4 PUF (a) Maximum C/R pairs in one subset (b) Training C/R pairs required to model one subset

the average size of training set required to model one subset of the PUFs to achieve 98% correct prediction using the model attack methodology described in Section 3.3. Again, for the same number of ROs, S-ArbRO-2 is more efficient as it requires a much larger training set before it is broken by modeling attack. For the same K (e.g. 18), although S-ArbRO-2 has half the C/R space, for any given desired C/R space, S-ArbRO-2 requires less number of ROs, which means it is smaller and more power efficient. Hence, the results we present from here on are for S-ArbRO-2 PUF.

3.4.1 S-ArbRO-2 PUF Characteristics

Effect of K on C/R space

Fig. 3.3 shows the improvement in the C/R space for S-ArbRO-2 PUF compared to an Arb PUF with as many elements. At N=12, the total C/R space for S-ArbRO-2 PUF and Arb PUF are 63,360 and 4096, respectively. It can be observed that with increasing N, the

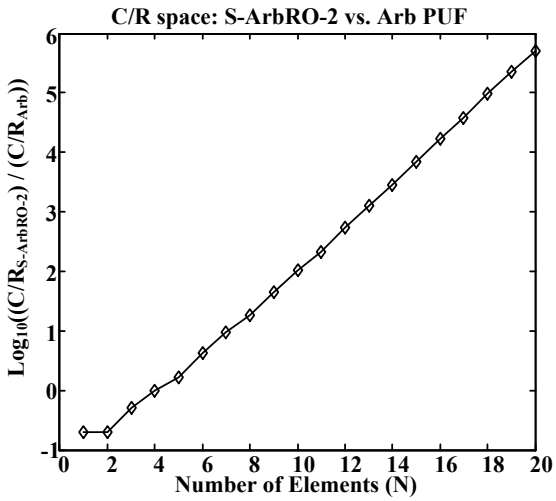

Figure 3.3: Total C/R space improvement of S-ArbRO-2 PUF over Arb PUF

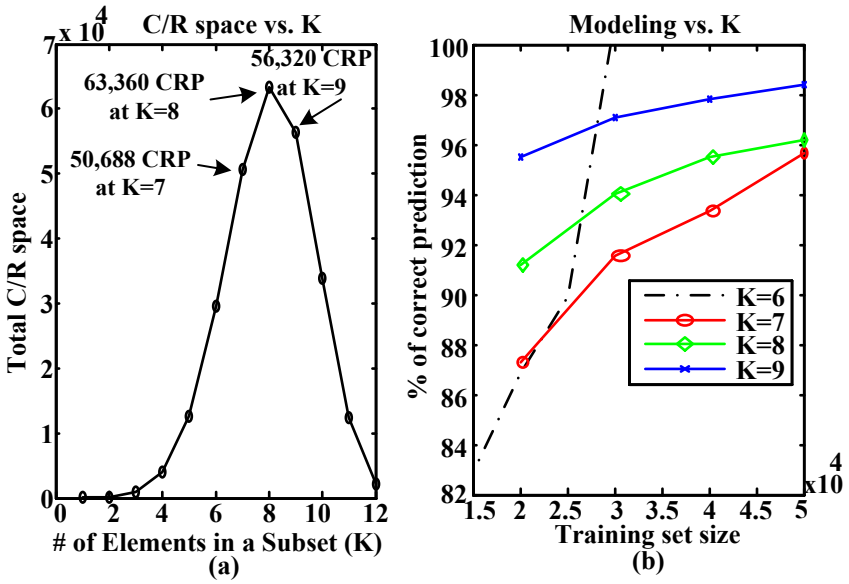

Figure 3.4: Effect of K on S-ArbRO-2 PUF (a) Total C/R pairs (b) Modeling

improvement in the total C/R space appears exponential. For S-ArbRO-2, at each N, K has been chosen to maximize the total C/R space, which is given by Eq. 3.2.

The value of K has a significant effect on the C/R space of an S-ArbRO-2 PUF. Fig. 3.4a shows the total size of C/R space for different values of K at N=12. It can be observed that the C/R space is maximum at K=8. In addition to the total C/R space, we study the effect

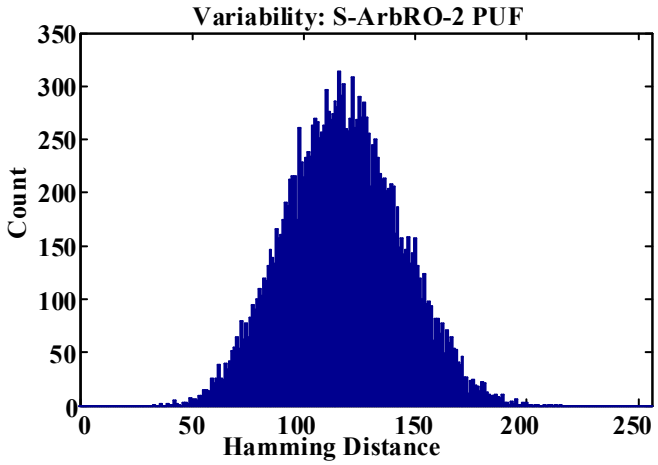

Figure 3.5: Variability of S-ArbRO-2 PUF

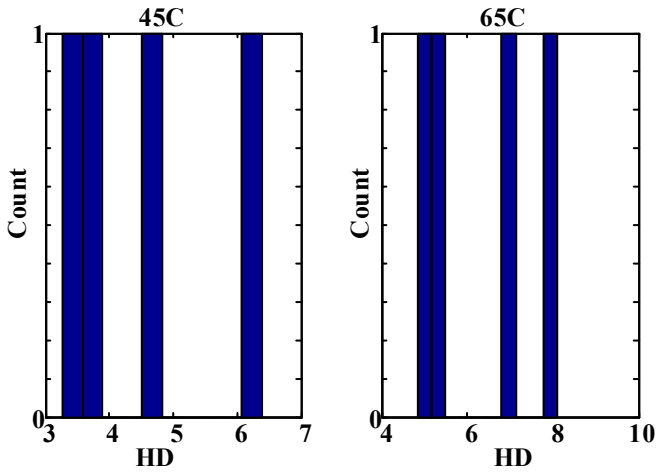

Figure 3.6: Stability of S-ArbRO-2 PUF

of K on resistance to modeling for this PUF in Fig. 3.4b. Although $K = 8$ has the maximum number of C/R pairs, $K = 7$ does slightly better in terms of modeling attack resistance. To explain this, we observe two extreme points one at K=9 and the other at K=6. At K=9, there are only 220 independent subset combinations and each C/R pair in the training set of the modeling attack is giving out information on 9 of the 12 elements. As a result, the training model reaches high prediction rates at low training set sizes. On the other hand, at K=6, although there are 924 subsets, the total C/R space itself is only about 29,000,

Table 3.1: Dependence of stability on K

Stability with K (HD)						
	45C			65C		
	K=5	K=7	K=9	K=5	K=7	K=9
Mean	4.83	4.55	5.20	6.85	6.35	7.05
Standard deviation	2.40	1.84	3.08	3.06	1.83	3.48

which is about half the size at K=8, resulting in small training set sizes. K=7 provides the optimum trade-off point between the number of subsets and the amount of information that each C/R pair gives out for the training model.

Variability and Stability

Fig. 3.5 shows the variability plot for S-ArbRO-2 PUF for a 256-bit response. We observe a mean Hamming Distance (HD) of 119 with a standard deviation of 25. The skew in the mean HD from the ideal HD of 128 is close to the skew observed for a traditional RO PUF. This is described in more detail in the following subsection. Fig. 3.6 shows the stability of S-ArbRO-2 PUF to temperature variations. The plots show the average HD for 100 randomly generated 256-bit responses at 45°C and 65°C. We can observe that the average HD is below 8 at both temperatures.

Effect of K on stability

Table 3.1 shows the stability of S-ArbRO-2 PUF for subset size, K, of 5, 7, and 9 for a 12-element PUF. It can be observed that the choice of K does not seem to affect stability at both 45°C and 65°C. Further, the stabilities are close to that of RO PUFs built using the same ROs.

Table 3.2: Comparison between S-ArbRO-2 PUF and RO PUF

PUF	Variability (HD)		Stability (HD)			
			45C		65C	
	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation
S-ArbRO-2 PUF	118.8	25.2	4.55	1.84	6.35	1.83
RO PUF	120.7	8.9	4.50	0.58	7.75	1.26

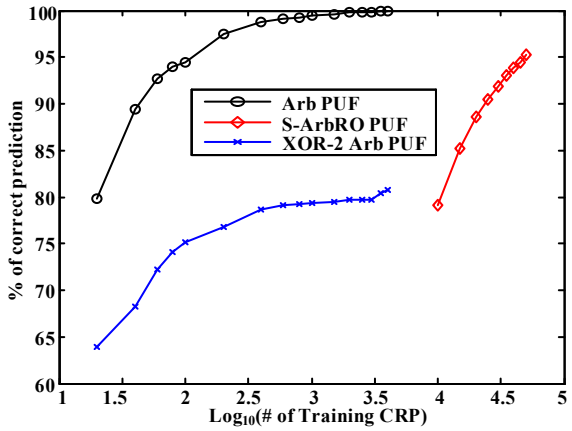

Figure 3.7: Modeling results for Arbiter PUF, S-ArbRO-2 PUF, and XOR-2 Arbiter PUF

3.4.2 Comparison to RO PUF

Table 3.2 compares the variability and stability of 12-element S-ArbRO-2 PUF with RO PUF. For a 256-bit response, the mean HD of the variability plot is almost equivalent for the two PUFs, whereas, S-ArbRO-2 ends up having a higher standard deviation. We observe the stability to temperature variations is very similar for both S-ArbRO-2 PUF and RO PUF.

3.4.3 Modeling

Fig. 3.7 shows the results of a logistic regression based modeling attack on Arbiter PUF (circles), S-ArbRO-2 PUF (diamonds), and XOR Arbiter PUF (stars). Besides regular Arbiter PUF, we compare our results to an XOR Arbiter PUF as it is more resistant to

Table 3.3: Comparison of stability between S-ArbRO-2 PUF and XOR-2 Arb PUF

PUF	Stability (HD)			
	45C		65C	
	Mean	Standard deviation	Mean	Standard deviation
S-ArbRO-2 PUF	4.55	1.84	6.35	1.83
XOR-2 Arb PUF	9.95	5.21	13.41	6.60

modeling attacks [42]. XOR Arbiter PUF is implemented by XORing the responses of two Arbiter PUFs and hence requires 2 times the number of elements to generate the same C/R space. The plot shows the percentage of correct prediction by the attack against the log of the number of training C/R pairs. The number of elements in Arb PUF and S-ArbRO PUF are 12, whereas XOR-2 Arb PUF has 24.

For S-ArbRO-2 PUF, the modeling attack is performed assuming the attacker has full knowledge of the architecture and challenge bit grouping. At K=7, S-ArbRO-2 PUF performs 495 independent logistic regressions in parallel. From Fig. 3.7, it can observed that it is only at a training set size of about 50K (98% of the total C/R space) that the correct prediction touches the maximum of 96%. For an Arbiter PUF, the prediction reaches 98% at only about 250 C/R pairs (6% of the total C/R space). For XOR-2 Arb PUF, which is more resistant to modeling attacks, the maximum prediction of 81% reaches when 97% of its C/R space is used. However, the number of secure C/R space is still much less compared to S-ArbRO-2 PUF as the total C/R space itself is only 4096 in spite of using double the number of elements. Further, XOR Arbiter PUF are known for their low stability. Table 3.3 shows the stability of both S-ArbRO-2 PUF and XOR-2 Arb PUF for randomly generated 256-bit IDs. At both 45C and 65C, the mean HD for XOR-2 Arb PUF is about 2 times that of S-ArbRO-2 PUF. Further, a standard deviation of 6.59 at 65C means that the number of unreliable bits in an XOR-2 Arb PUF can reach as high as 33 on a 256-bit ID.

Although, applying more sophisticated attacks can break the PUFs much sooner as the

number of unknowns in the PUF is still a constant between Arb PUF and S-ArbRO-2 PUF, we believe that this approach makes it more difficult for the attacker to model the PUF, while avoiding the stability issues that are associated with XOR based approaches. This is especially important for low cost embedded applications where techniques to prevent modeling attacks are too expensive to incorporate in the design.

3.5 Conclusion of technology-independent techniques

In this chapter, we proposed a technology-independent PUF architecture using ring oscillators that produces a very large Challenge/Response space, which is essential for low cost embedded systems. We proposed a modification to the basic element in the PUF to improve the C/R space per unit area. Further, by implementing a logistic regression-based modeling attack, we showed that S-ArbRO PUF is very resistant to modeling.

So far, in this work, we have looked into the technology-independent techniques for the development of PUFs. In the following chapter, we pay attention to the technology-dependent techniques of Ring Oscillator PUFs and SRAM PUFs, as they are the leading contenders for silicon PUFs.

Chapter 4

Technology-dependent PUF Techniques

In this chapter, we present three technology-dependent techniques to study and improve the qualities of Physical Unclonable Functions. In Section 4.1, we propose Leakage-PUF that uses efficient analog components to generate highly stable responses. Stability improvement techniques for bi-stable PUFs are presented in Section 4.2. Finally, in Section 4.3, we study the effect of aging on ring oscillator PUFs.

4.1 Leakage Physical Unclonable Function (L-PUF)

In this section, we propose a new silicon PUF using efficient analog components that can be fabricated on a standard CMOS process. Our proposed design is built using leakage sensors with each measuring the leakage current of a transistor. Multiple identical leakage sensors are fabricated on the same chip. Due to manufacturing process variations, each sensor produces slightly different leakage values that can be compared in order to create chip-unique responses. Therefore, we name our design Leakage PUF or L-PUF. The top-

level architecture of L-PUF is similar to that of an RO PUF, whereby ring oscillators are replaced with leakage sensors, and multiplexers and comparators are replaced with their analog counterparts. Despite the similarities, L-PUFs show a significant improvement in the performance metrics when compared to RO PUFs as leakage current is more closely affected by process variation. We study the stability of L-PUF with respect to temporary environmental variations like temperature and supply voltage. Our results show that nearly ideal stability can be achieved with minimal area overhead in our design. Comparing with a popular ring oscillator PUF architecture of the same entropy, our proposed PUF consumes about 80% less power, occupies about 85% less area, and has a high level of stability across a wide range of temperatures.

The contributions of this technique are as follows:

- We propose a new PUF design using leakage sensor circuitry and other analog components. We show that the new design can effectively identify the same number of chips with much less power and area compared to a traditional RO PUF.
- We study the stability of L-PUF with respect to environmental variations and show that it has very high stability.

The rest of this section is organized as follows: Section 4.1.1 presents the architecture of L-PUF with its analog circuit components. Section 4.1.2 details our methodology for studying L-PUF. Section 4.1.3 presents the results showing the stability of L-PUF to environmental variations and its inter-chip variability. It also provides power and area characteristics of L-PUF and compares it to RO PUF design. Conclusion of this technique is presented in Section 4.1.4.

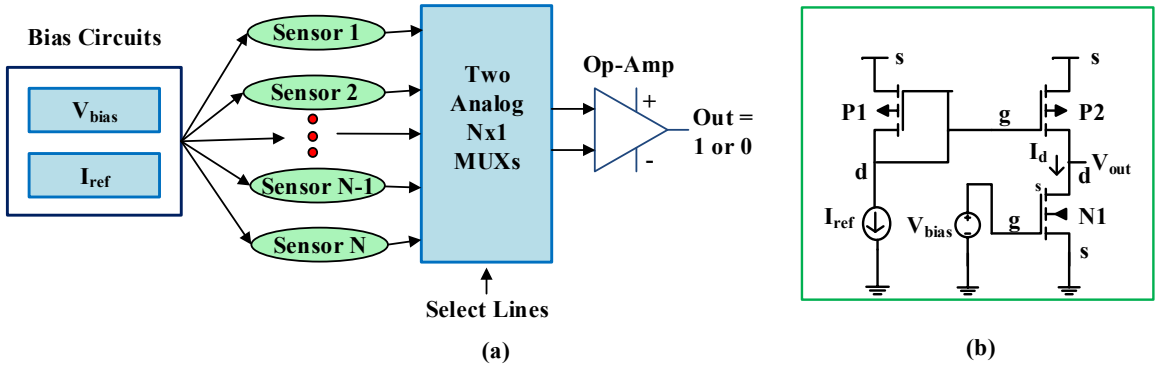

Figure 4.1: Leakage PUF (L-PUF) (a) Architecture. (b) Leakage sensor.

4.1.1 Overview of Leakage-PUF

Architecture

Fig. 4.1a shows the architecture of Leakage PUF. In L-PUF, leakage sensors are used to capture process variation. The N leakage sensors are designed to be identical, however, the output of each sensor, which is an analog voltage that is dependent on leakage current, is slightly different due to process variation. Two Nx1 analog multiplexers are used to select two out of N sensors. The output voltage of these sensors is then compared using an analog comparator, i.e. a differential amplifier of sufficient gain.

Based on the control logic of the select lines, the L-PUF can either be used as a Challenge/ Response (C/R) function or as a secure key generator for cryptographic applications. If the PUF is used as a C/R function, the challenge from the server is directly fed into the select lines to produce a 1-bit result. To achieve a k-bit digital response, k different comparisons are made between the sensing elements. In case of an N-sensor L-PUF, the total entropy held by PUF is $log_2(N!)$ as there are total $N!$ possible orderings of the sensors.

Leakage Based Analog Sensor

The leakage sensor is the heart of L-PUF, where the amount of leakage current passing through a transistor is captured as a voltage. We adopt the leakage sensor circuit proposed by Kim et al. [20] and modify it to achieve maximum level of variation among sensor outputs. Fig. 4.1b shows the circuit of the leakage sensor. The circuit consists of two PMOS (P1 and P2) and one NMOS (N1) transistors in addition to the extra circuitry for building I_{ref} and V_{bias}. The ports of the transistors are labeled with g (gate), d (drain) and s (source). The goal of the leakage sensor is to capture the leakage current passing through N1. The purpose of P1 and P2 is to force N1 to operate in a region where leakage current can be easily measured.

In order to explain how the circuit operates, we pay attention to the fact that there are two modes of operation for any given transistor: when the absolute difference between gate voltage and source voltage, i.e. V_{gs}, exceeds threshold voltage, V_{th}, the transistor turns on and a significant amount of current passes through it. This is known as superthreshold operation. When V_{gs} is below V_{th}, a limited amount of current known as leakage current passes through the transistor. This is called subthreshold operation.

In the leakage sensor circuit, I_{ref} is chosen such that P1 operates in superthreshold region. In our design, we choose I_{ref} to be 10uA as it provides acceptable variability at the sensor output, which is the ultimate goal of L-PUF. This amount of current results in V_{gs} of 570mV for P1, which is mirrored to P2 as their gate ports are connected. On the other hand, V_{bias}, which is equal to V_{gs} of N1, is selected such that N1 operates in subthreshold region. The choice of V_{bias} has a significant impact on variability and stability of L-PUF and for our design V_{bias} is at 125mV such that it offers good randomness in the sensor output voltages.

The output of the leakage sensor, V_{out} is read at the drain port of P2 and N1 that are tied together. Also, assuming a large impedance load at V_{out}, the current passing through P2 and N1, I_{ds}, are equal. However, since each of P2 and N1 operates under different V_{gs} and has different design parameters, each has a different characteristic curve that relates its I_{ds}

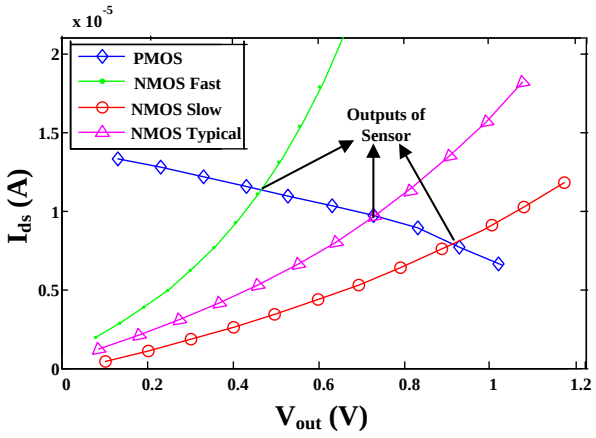

Figure 4.2: I-V curves for P2 and N1.

to V_{out}. Fig. 4.2 presents both curves on the same graph. The point that these two curves intersect, is where P2 and N1 share the same I_{ds} and V_{out} and therefore, represents the point that the circuit reaches its steady state and V_{out} becomes stable.

In order to explain how V_{out} is related to leakage current of N2, we plot I_{ds} to V_{out} for the N1 of three different sensors on the same graph. Basically, under the effect of process variation, the current curves for both PMOS and NMOS shift slightly. However, since P2 is in superthreshold region, the changes to its curve for the three sensors is negligible. Therefore, the V_{out} of different sensors practically reflects the leakage current of N1 as shown in Fig. 4.2. It can be seen that even a slight variation in NMOS characteristics is reflected by a significant change in the output voltage. This change in output voltage based on the process characteristics of the transistor is captured to produce a PUF response.

Analog MUX and Op-amp

The circuit of an analog 2x1 MUX consists of two transmission gates along with an inverter, which is much smaller and more power efficient than its digital counterpart. An analog Nx1 MUX, where N is a power of 2 can be easily built using a tree of 2x1 analog MUXs very similar to building a digital MUX.

The operational amplifier is designed to have enough gain to amplify the difference between the V_{out} of two selected sensors such that the output of the op-amp can be easily used to drive a flip-flop or a digital latch to store the response of the L-PUF. The op-amp consists of a bias circuit, differential amplifier, and a common source amplifier. It is designed to offer a gain of 67dB and consumes about 166uW of power.

Ideally, a positive offset is amplified by an op-amp to reach V_{DD} and a negative offset to reach 0. However, if the difference between the two sensors is below a certain threshold, minimum offset, the output of op-amp is unpredictable. In such a case, that particular pair of sensors should be excluded (rejected) from challenge space. The ratio of rejected bits to the whole space is called Bit Rejection Ratio (BRR). Minimum offset of our op-amp is 1mV and can be improved by larger and more power hungry designs.

4.1.2 Methodology

All the modules of the architecture proposed in Section 4.1.1 are implemented in SPICE with Monte Carlo analysis for 2000 chip instances. Simulations were performed in a 90nm technology with both the inter-chip and intra-chip variations. The simulations were carried out using the highest possible accuracy settings.

The area numbers for RO PUFs with 5 inverting stage Ring Oscillators are collected by synthesizing an RO PUF using 90nm standard cell library. The power numbers are collected using PrimeTime-PX. The area numbers for the analog components are calculated by adding the areas of the transistors in the design plus 50% for routing.

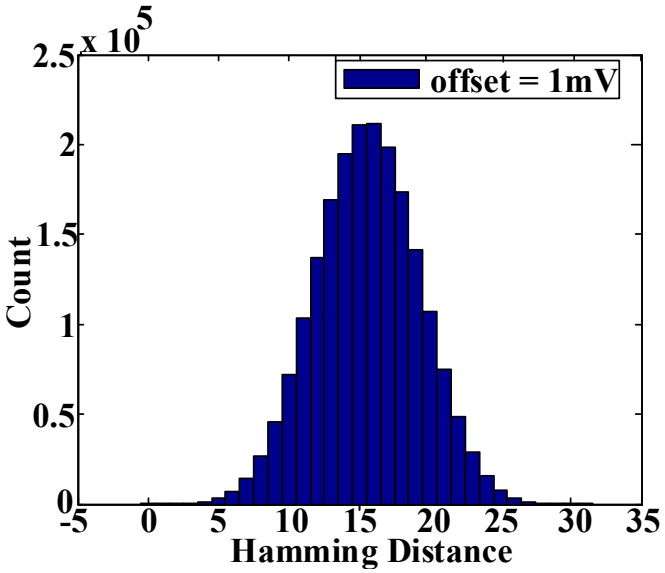

Figure 4.3: Inter-chip variation of L-PUF.

4.1.3 Results

Variability

Fig. 4.3 shows the inter-chip hamming distance plot for the pool of 2000 simulated chip instances. The ID length for the PUFs is 32-bit. Our results present a desirable distribution of inter-chip variation by showing a bell curve with its mean and standard deviation at 15.59 and 3.07, respectively. Any skew to the left or right would have meant a correlation among responses. However, based on this graph, the responses are not correlated and the PUF is able to effectively capture manufacturing process variations.

Power and Area

Table 4.1 shows the transistor count and the area occupied by the components of L-PUF. Table 4.2 provides the area and power consumption of L-PUF for varied number of leakage sensors on the same IC. For comparison, the same characteristics are reported for an RO PUF with similar number of ring oscillators each with 5 inverting stages, which is known to

Table 4.1: Transistor count and area for L-PUF components

Component	# of transistors	Area (um^2)
Sensor	3	1.32
V_{bias} & I_{ref}	18	101
32x1 MUX	206	47
Op-amp	23	653

Table 4.2: Area and power for RO PUF and L-PUF

# of ROs/ Sensors	Power (uW)		Area (um^2)		% Reduction	
	RO-PUF	L-PUF	RO-PUF	L-PUF	Power	Area
16	1739	679	5118	797	**61**	**84**
32	3372	694	8134	843	**79**	**89**
64	6639	725	8947	930	**89**	**89**
128	13172	788	15708	1110	**94**	**92**

provide acceptable variation. It is well-known that carefully designed analog components can achieve much higher efficiency compared to their digital counterparts. This fact is clearly reflected in our results. It can be seen that for a 128-element design, the L-PUF consumes about 95% less power and area.

It can also be observed that power consumption of L-PUF does not increase as rapidly as RO PUF, with increasing the number of sensing elements. This is because the major source of power consumption in L-PUF is the bias generation circuitry, which is common for all leakage sensors.

Stability

Stability is defined as the ability of a valid response bit to produce the same response even in the presence of environmental variations. A valid response bit is a bit that is generated by its corresponding sensors' output voltages which are different enough for detection to

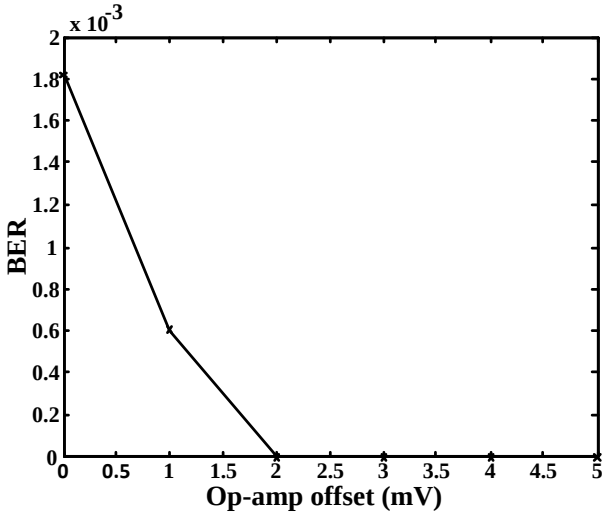

Figure 4.4: Bit Error Rate vs. Op-amp offset.

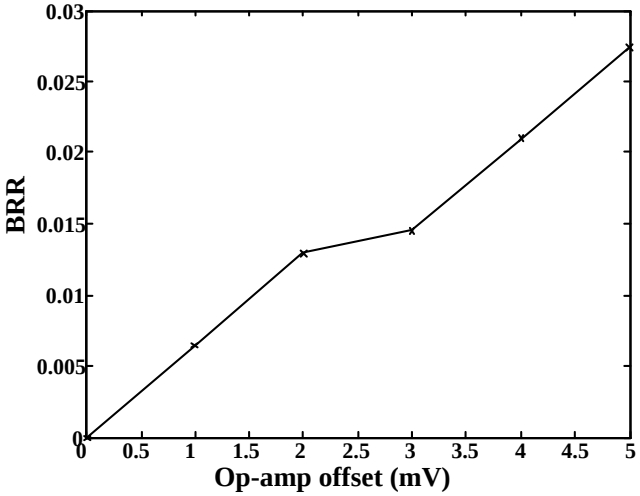

Figure 4.5: Bit Rejection Rate vs. Op-amp offset.

logic 1 or logic 0 by the op-amp during the initial reference measurement taken at the room temperature.

The stability of a bit can be observed by a metric Bit Error Rate (BER), which is defined as the probability of error of a valid response bit. Fig. 4.4 shows the BER with varying operating temperature of the PUF against the minimum offset of the op-amp. The operating

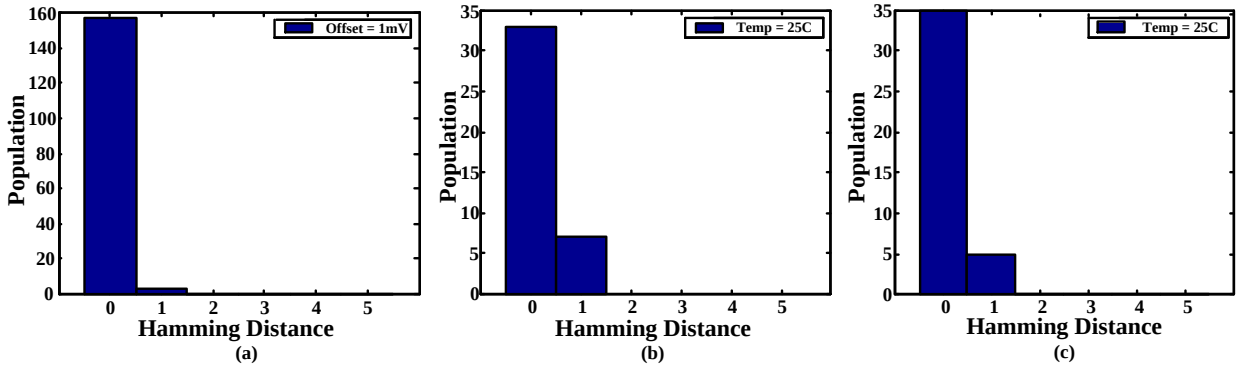

Figure 4.6: L-PUF stability (a) Temperature. (b) Supply. (c) Bias variations.

temperature has been swept from -15C to 95C in steps of 10C. It can be observed that for an offset of 2mV and above, the BER is 0. To make the analysis more balanced with respect to the total number of usable bits in the design, we have to observe the percentage of valid bits in the reference measurement, which is termed as Bit Rejection Ratio (BRR). Fig. 4.5 shows the BRR of L-PUF against the offset of the op-amp. At an offset of 1mV, BRR is as low as 1%. From Fig. 4.4 and 4.5, it can be observed that there is a trade-off between lowering BRR (which results in less area overhead) and lowering BER (which results in more stability). Since BRR is relatively low for our design, we suggest the L-PUF be used with an offset of 2mV.

Fig. 4.6a shows the stability of L-PUF to variation in temperature. It can be observed that for a 32-bit ID the maximum HD is 1, which means there is a maximum of one bit flip in the responses of the PUF at different temperatures compared to the reference response showing a remarkable stability in the generated bits.

Another important variation parameter that impacts the stability of the responses is the supply voltage. We studied the impact of supply voltage by varying it by a margin of 10% on either side of the nominal voltage. As the variation in bias for the subthreshold transistor (N1) has a significant impact on the L-PUF's behavior, we observed the stability of our design to V_{bias} by varying it by 10% on either side of 125mV. Fig. 4.6b and 4.6c show the

robustness of L-PUF to changes in supply voltage and V_{bias} respectively.

4.1.4 Summary of L-PUF discussion

In this technique, we proposed a new silicon PUF that uses leakage sensors to capture process variation. Our proposed design takes advantage of efficient analog components in order to build a PUF that is more stable, less power hungry, and smaller, while maintaining the same level of entropy as a digital counterpart such as an RO PUF.

In this chapter, so far, we have proposed leakage PUF, which is more appealing for applications like Challenge/Response authentication. For secure key generation applications, bi-stable PUFs like SRAM PUFs are much more suitable for two reasons: First, bi-stable PUFs directly generate binary data. Second, binary data of required length can be generated at the same time unlike RO PUFs, where counters are typically shared slowing the rate of response generation. Hence, in the following section, we investigate techniques to improve the stability of bi-stable PUFs.

4.2 Bi-stable PUF Stability

Among different silicon PUF designs, bi-stable PUFs are particularly attractive as they directly generate binary data. In order to achieve this, they exploit the random start-up state of identical cross-coupled circuits. During power-up, each bi-stable cell evaluates to logic 1 or logic 0 based on the mismatch between the cross-coupled inverters and surrounding noises [19]. Bi-stable PUFs are relatively more stable to supply voltage (V_{DD}) variations as compared to temperature variations [5, 39]. This is because V_{DD} typically reflects as common-mode noise on a bi-stable cell. Although temperature can be generalized to be common mode on a bi-stable cell, unlike V_{DD}, temperature directly affects the V_{th} of the transistors in the bi-stable cell. In the presence of process variation, the variation in V_{th}

due to temperature is not necessarily common mode across the transistors that form the bi-stable cell. This is further discussed in Section 4.2.2. In this work, we study the causes of instability to temperature variations in bi-stable ASIC PUFs and propose circuit-level techniques to improve their stability.

The contributions of this work are as follows:

- We show that temperature does not necessarily have a common-mode effect on the V_{th} variation of MOSFETs.
- We study the stability of bi-stable PUFs in the context of differences between pMOS and nMOS. We show that bi-stable PUFs are less reliable, exploiting pMOS variation as compared to nMOS variation. To the best of our knowledge, this is the first work that exploits the differences between pMOS and nMOS for PUFs.
- We propose circuit-level techniques to minimize the impact of pMOS variation. Our simulations confirm that SRAM PUF stability can be improved by about 70% in 90nm technology. We further show the applicability of the proposed technique in 130nm and 65nm technologies.
- We implement an ASIC in 90nm CMOS technology to validate the proposed technique in silicon. Our experiments show 35% improvement in stability to temperature variations.

The rest of Section 4.2 is organized as follows. Section 4.2.1 presents the related work. In Section 4.2.2, the dependence of threshold voltage on temperature is discussed. In the same section, the differences between nMOS and pMOS transistors are observed in the context of process variation and temperature. Section 4.2.3 presents the techniques proposed to improve the stability of bi-stable PUFs. Simulation results are presented in Section 4.2.4. ASIC implementation and experimental results are presented in Section 4.2.5. Section 4.2.6 presents the conclusion of this technique.

4.2.1 Related Work

There is a wealth of research on the topic of exploiting process variation (PV) to generate chip specific unique identifiers. [17] and [19] have proposed SRAM PUF that uses random startup values of SRAM cells. Many other PUF architectures exploit the random startup value of cross-coupled circuit elements [4, 25, 40, 41, 47]. [4] uses sense-amplifiers, [41] uses latches, [40] uses buskeepers, [25] uses latches in FPGAs, and [47] uses flip-flops. While some of the bi-stable PUFs proposed are targeted for FPGAs, most of the techniques apply to both ASICs and FPGAs.

In the PUF literature, not much work has been published in improving the stability of bi-stable PUFs. [6] has proposed using techniques like Directed Accelerated Aging (DAA) and multiple evaluation (ME) to improve bi-stable PUF stability. DAA involves characterizing and selectively aging the transistors of bi-stable PUF elements to increase the mismatch between the cross coupled elements. In ME, each bi-stable cell is evaluated for a large number of times to generate a soft response based on the probability with which a cell goes to logic 0 or logic 1. Both DAA and ME requires extensive post manufacturing characterization of each cell of all the PUFs in the population. Hence, they become very expensive to implement. Similarly, methods to improve stability in [4] also require expensive post-fabrication steps.

Contrary to the above, the techniques proposed in this work are at circuit-level. They improve the stability of bi-stable PUFs inherently with no related post-processing required. Although, techniques in [6], implemented in conjunction with our proposed technique may further improve the stability of bi-stable PUFs.

4.2.2 Temperature Stability of Bi-stable PUFs

In this section, first, we briefly talk about the dependence of threshold voltage (V_{th}) on temperature. Following that, we study the differences between nMOS and pMOS with respect to temperature, in the presence of process variation.

Dependence of V_{th} on temperature

Bi-stable PUFs exploit the mismatch in the threshold voltages (V_{th}) of the MOSFETs in the cross-coupled circuits [6, 10]. Post manufacturing, the two major sources of V_{th} variation in MOSFETs are temperature and body biasing. Assuming MOSFETs are not body biased, temperature is the major source of V_{th} variation. So, to solve the stability problems, it is important to study the effect of temperature on V_{th}.

Eq. 4.1 from [43] shows the V_{th} of an nMOS, where V_{FB} is the flat-band voltage, ψ_B is the bulk potential, ε_s is the permittivity of semiconductor, q is the unit electron charge, N_A is the acceptor impurity concentration, and C_{ox} is the oxide capacitance per unit area. ψ_B is given by the Eq. 4.2. Eq. 4.3 presents the relationship between n_i, which is the intrinsic carrier concentration, and temperature.

$$V_{th} = V_{FB} + 2\psi_B + \frac{\sqrt{2\varepsilon_s q N_A (2\psi_B)}}{C_{ox}} \tag{4.1}$$

$$\psi_B = \frac{kT}{q} ln(\frac{N_A}{n_i}) \tag{4.2}$$

$$n_i^2 \propto T^3 exp(\frac{-E_{g0}}{kT}) \tag{4.3}$$

Here, T is the temperature, k is the Boltzmann constant, and E_{g0} is the energy gap at T=0K.

From Eq. 4.1, 4.2 and 4.3, it can be observed that V_{th} is a complex function of Temperature (T). V_{th} is typically modeled as linearly dependent on temperature [49] as shown in the Eq. 4.4, where $V_{th}(T)$ is the threshold voltage at any temperature T, T_{Ref} is the reference temperature, and k_{vt} is the temperature coefficient. [14, 44] show that k_{vt} is not constant with respect to temperature.

$$V_{th}(T) = V_{th}(T_{Ref}) - k_{vt}(T - T_{ref}) \tag{4.4}$$

k_{vt} has been evaluated in [14, 44], and it is shown in Eq. 4.5, where ϕ_{ms} is the work-function

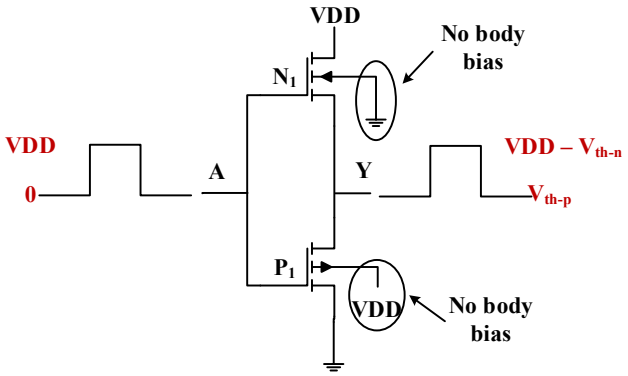

Figure 4.7: Non-Inverting Buffer (NIB) showing V_{th} drops

difference. It can be observed that k_{vt} is a function of temperature (T), carrier concentration (N_A), and oxide thickness (t_{ox}) (because of C_{ox}), where N_A and t_{ox} are process variation dependent. Hence, k_{vt} randomly varies between identical transistors while being a function of temperature. This means, an nMOS with a higher threshold voltage at one temperature can end up having a lower threshold voltage at another temperature compared to a second identically designed nMOS.

$$k_{vt} = \frac{\partial \phi_{ms}}{\partial T} + \frac{\partial \psi_B}{\partial T}\left(2 + \frac{1}{C_{ox}}\sqrt{\frac{\varepsilon_s q N_A}{\psi_B}}\right) \tag{4.5}$$

Although the discussion has so far been for an nMOS, the same argument applies to pMOS as well. These changes in the threshold voltages of MOSFETs with temperature result in a change in the mismatch of the cross-coupled circuits, which is the major cause of instability in bi-stable PUFs.

V_{th} variation: pMOS vs. nMOS

In the previous subsection, we analyzed the dependence of a given MOSFET's V_{th} on temperature, regardless of being nMOS or pMOS. In this subsection, through simulations, we focus on the differences between pMOS and nMOS in this context. Identical non-inverting buffers (NIB) are used to capture V_{th} variation. Fig.4.7 shows the circuit of a non-inverting

buffer. As a pMOS passes a weak 1 and an nMOS passes a weak 0, the output Y of an NIB charges only up to $V_{DD} - V_{th-n}$ and discharges only up to V_{th-p}, where V_{th-n} and V_{th-p} are the threshold voltages of nMOS and pMOS, respectively.

Based on industry 90nm MOSFET and variation models, Monte Carlo (MC) SPICE simulations have been performed to observe V_{th} variation. Both die-to-die and within-die variations have been turned on to the preset standard deviation (σ) levels provided by the foundry. 100 chip instances have been simulated with 50 NIBs in each chip. Temperature has been varied from -15C to 85C in steps of 20C. So, there are a total of $10K$ MOSFETs (100chips $\times$ 50NIBs $\times$ 2MOSFETs/NIB).

The goal of this experiment is to observe if relative threshold voltage values of identically designed MOSFETs change with temperature. Assume V_{th-n_x} and V_{th-n_y} are the threshold voltages of identical nMOSs, x and y. Both x and y are indices of nMOSs in one chip, which are between 1 and 50. If $V_{th-n_x}(T_1) > V_{th-n_y}(T_1)$ and $V_{th-n_x}(T_2) < V_{th-n_y}(T_2)$, where T_1 and T_2 are different temperatures, the relative threshold voltage of x and y has switched. From here on, for simplicity, we refer to this as V_{th} *Switch*. So, in each chip, the magnitudes of the V_{th} of transistors are compared to generate binary responses, Q_n and Q_p for pMOS and nMOS, respectively, as shown by Eq. 4.6 - 4.9.

$$V_{th-n_x}(T) > V_{th-n_y}(T) \Rightarrow Q_{n_{xy}}(T) = 1; \tag{4.6}$$

$$V_{th-n_x}(T) < V_{th-n_y}(T) \Rightarrow Q_{n_{xy}}(T) = 0; \tag{4.7}$$

$$V_{th-p_x}(T) > V_{th-p_y}(T) \Rightarrow Q_{p_{xy}}(T) = 1; \tag{4.8}$$

$$V_{th-p_x}(T) < V_{th-p_y}(T) \Rightarrow Q_{p_{xy}}(T) = 0; \tag{4.9}$$

At different temperatures, Q_n and Q_p have been measured for all NIBs of the 100 chips simulated. As there are 50 NIBs in each chip, a total of 122500 bits ($C_2^{50} \times 100$ chips) are evaluated at each temperature for both nMOS and pMOS.

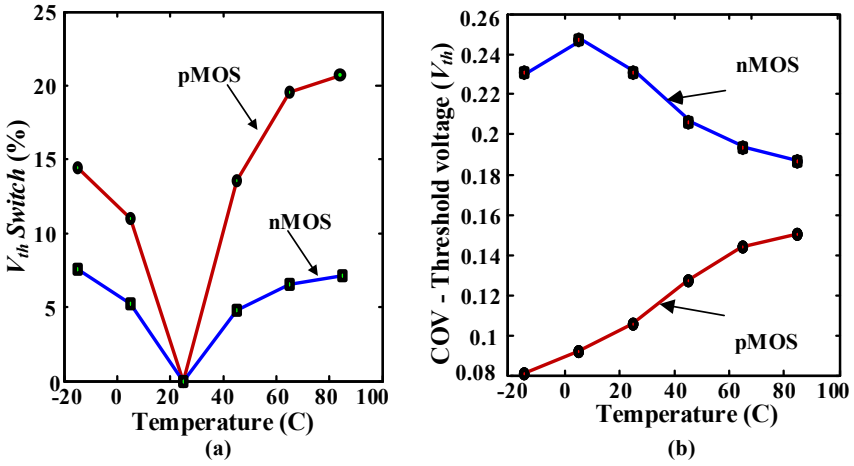

Figure 4.8: pMOS vs. nMOS (a) V_{th} *Switch* with temperature. (b) COV of V_{th} with temperature.

Fig. 4.8a shows the probability of V_{th} *Switch* for pMOS and nMOS. It can be observed that V_{th} *Switch* for nMOS is less than that of pMOS at all temperatures. This suggests that nMOS is more tolerant to random variations in the temperature coefficient of V_{th} than pMOS.

Further, [45, 46] have shown that nMOS transistors on a die have a higher variation in their V_{th} than pMOS transistors. The authors attribute that to a possible clustering of the dopant boron atoms in the channel of an nMOS. We observe this V_{th} variation phenomenon in simulations as well. Fig. 4.8b plots the coefficient of variation (COV) of 50 nMOS and 50 pMOS V_{th}s, independently. COV is given by Eq. 4.10, where i is the MOSFET index, n is the number of pMOS or nMOS (in our case, 50), $\overline{v_{th}}$ is the mean of respective MOSFET threshold voltages. The reported COV is the mean of the COVs of the 100 chips simulated. It can be observed that at all temperatures, the COV of V_{th} of pMOS is lower than that of nMOS. In other words, nMOS transistors see more process variation in V_{th} than pMOS transistors.

We apply these observations to a bi-stable PUF cell to verify if the stability of PUFs can be improved.

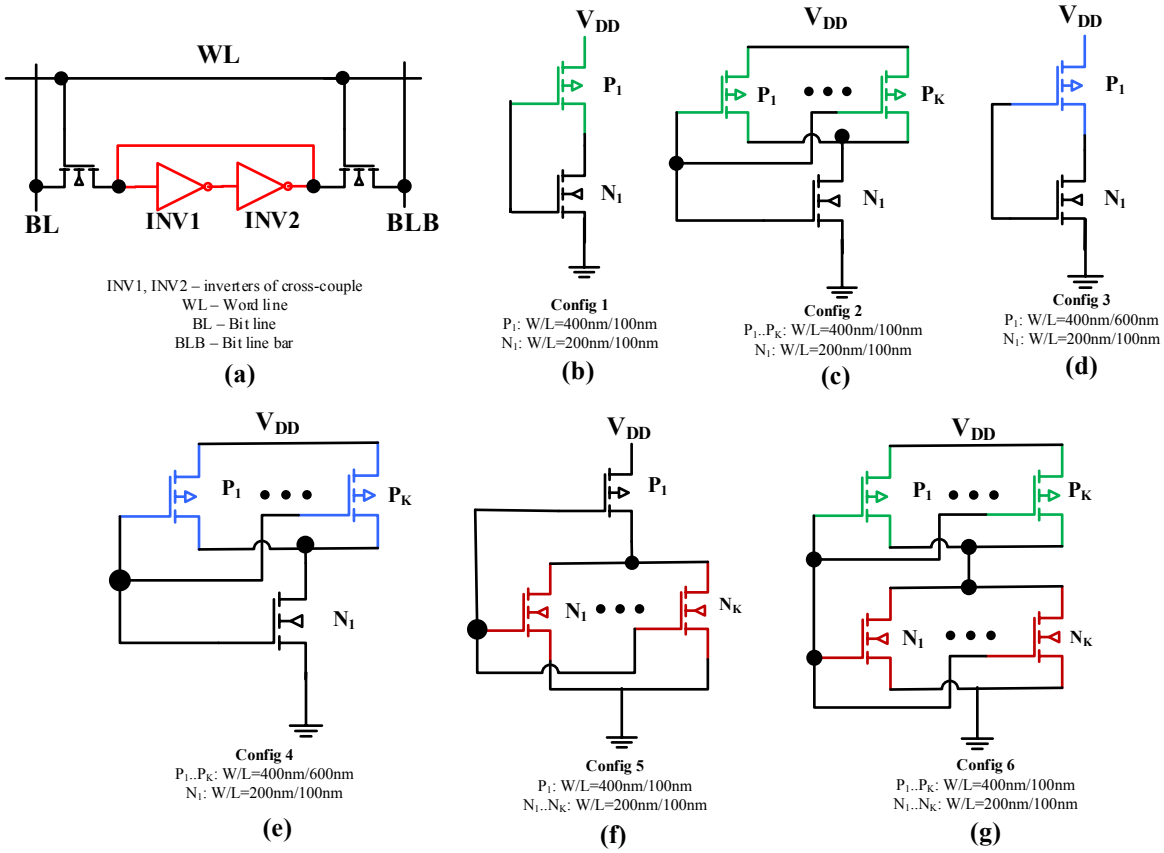

Figure 4.9: SRAM configurations (a) SRAM PUF cell showing cross-coupled inverters (INV). (b) Reference INV. (c) INV with parallel pMOS. (d) INV with longer pMOS. (e) INV with parallel and longer pMOSs. (f) INV with parallel nMOSs. (g) INV with parallel nMOSs and parallel pMOSs.

$$COV = \frac{\sqrt{\sum_{i=1}^{n} \frac{(v_{th}^{i} - \overline{v_{th}})^2}{n-1}}}{\overline{v_{th}}} \tag{4.10}$$

4.2.3 Circuit-level improvement for SRAM PUFs

In this section, we take an SRAM PUF as a representative candidate for bi-stable PUFs and implement circuit-level techniques to minimize the impact of pMOS variation.

Simulation Setup

For each of the SRAM PUF configurations presented in the following subsection, we have performed Monte Carlo SPICE simulations using industry 90nm MOSFET models. Die-to-die and with-in-die variation levels are preset in the spice models provided by the industry. 1000 identical SRAM cells have been simulated in each chip and a total of 100 chips are instantiated. Supply voltage (V_{DD}) is 1.2V and temperature has been varied from -15C to 85C in steps of 20C. The ramp-up rate of V_{DD} has been chosen such that it does not impact the stability of the PUF. After the SRAM cells stabilize to one of the two stable states, the output voltages are measured. Post-processing is performed using Matlab.

SRAM PUF Configurations

Six variations of SRAM PUFs have been implemented in the simulations. Fig. 4.9a shows an SRAM cell with abstract identical inverters, INV1 and INV2. [10] shows that the startup values of an SRAM PUF are dependent on the MOSFETs that make up the cross-coupled inverters. The access transistors do not have any significant effect on the random startup value of an SRAM cell. Hence, in all the configurations below, changes are only made to the cross-coupled inverters, while the access transistors remain identical. Fig. 4.9b, 4.9c, 4.9d, 4.9e, 4.9f, and 4.9g show different inverter configurations implemented, which are discussed below.

Config 1 (Reference SRAM PUF) Fig. 4.9b shows the inverter of the reference SRAM cell. Minimum sized inverters from the 90nm standard cell libraries are used to implement the cross-coupled inverters.

Config 2 (SRAM with parallel pMOS) The inverter for Config 2 is shown in Fig. 4.9c. Here, to minimize the impact of pMOS variations, we use parallel pMOSs in each inverter such that it reduces the impact of process variation. The number of pMOSs in each inverter,

K, has been varied from 2 to 5.

Config 3 (SRAM with longer pMOS) As one of the major sources of process variation is the channel length (L), in this configuration, to minimize the impact of pMOS variation, the L of the pMOSs in the cross-coupled inverters is increased. This is shown in Fig. 4.9d.

Config 4 (SRAM with parallel and longer pMOS) As shown in Fig. 4.9e, this configuration brings together Configs 2 and 3, i.e. the pMOS of the inverters are longer, and parallel pMOSs are used to reduce the effect of V_{th-p} variation. Here, the number of parallel pMOSs, K, is set at 5.

Config 5 (SRAM with parallel nMOS) Analogous to Config 2, this configuration uses parallel transistors to minimize the effect of process variation. However, nMOSs are used in parallel instead of pMOSs, as shown in Fig. 4.9f. This configuration is not proposed to improve bi-stable PUF stability. Instead, it is presented to further emphasize the difference in bi-stable PUF's stability exploiting nMOS variation over pMOS variation. Here, the number of parallel nMOSs, K, is set at 3.

Config 6 (SRAM with parallel pMOS and nMOS) With this configuration, we show that it is not primarily because of the increase in the size of the pMOS transistor that the stability has improved, but because of the differences between nMOS and pMOS. As shown in Fig. 4.9g, this configuration uses both nMOS and pMOS transistors in parallel, which intuitively should improve the stability as the transistors are less impacted by process variation in the cross-coupled bi-stable cells. However, in Section 4.2.4, we will see that stability does not change significantly. Similar to Config 5, this configuration is not proposed to improve bi-stable PUF stability and only serves as comparison point. The number of parallel transistors, K, is set at 3.

Table 4.3: Variability of SRAM PUF configurations

Configuration	Variability on a 1000-bit ID			
	Mean	Std	Min	Max
Config 1 (Reference)	500.04	16.25	442	558
Config 2 at K=5	500.01	15.82	440	553
Config 3	500.10	16.11	443	559
Config 4	499.99	15.86	440	557

4.2.4 Simulation Results

In this section, PUF quality metrics, variability and stability, are measured for each of the configurations presented earlier in this section. Following that, we will also present the power and area numbers for each of the configurations. Finally, stability results for the proposed technique are shown in 130nm and 65nm technologies.

PUF Quality Metrics: Variability and Stability

Variability We recall variability is the quality of distinguishing an IC uniquely from the rest of the ICs in a population. As each simulated SRAM PUF has 1000 cells, we take all the 1000 bits as the unique identity (ID) for that PUF. PUF IDs are measured for all 100 chips in the population and the histogram of the pairwise inter-chip hamming distances gives us information about the correlation between the chip IDs. For brevity, in Table 4.3, the mean, standard deviation (std), minimum (min), and maximum (max) HDs of the variability histogram are presented for each configuration. It can be observed that in all configurations, in spite of minimizing the impact of pMOS variation in the SRAM cell, PUF variability stays very close to the reference SRAM PUF.

Stability For each chip in the population, the 1000-bit ID is measured at the temperature points mentioned in Section 4.2.3. With the IDs measured at 25C as reference, HDs are calculated at different temperatures compared to the reference. At each temperature, the

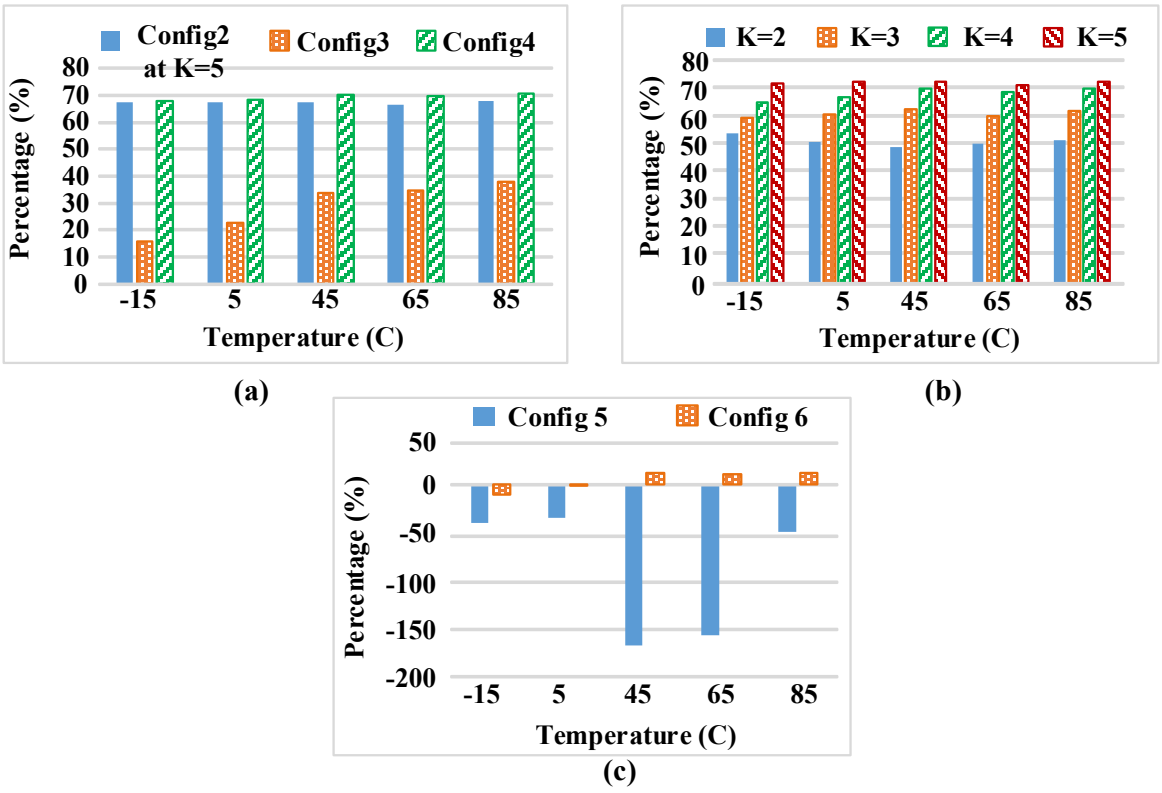

Figure 4.10: Stability of SRAM PUF configurations (a) Configs 2, 3, 4. (b) Config 2 at K=2,3,4,5. (c) Configs 5 and 6.

average HD is measured for all the chips in the population. Fig. 4.10a shows the percentage improvement in the number of error bits for Config 2 (at K=5), 3 and 4 compared to the reference Config 1. It can be observed that in Config 2, which uses parallel pMOSs to minimize the effect of V_{th-p} variation, the percentage of unreliable bits is reduced by about 70%. The improvement for Config 3 is around 30%. Using Config 4, where both parallel and longer pMOSs are used, stability has only slightly improved over Config 2.

Fig. 4.10b shows the percentage decrease in the number of error bits for Config 2 at different number of parallel pMOSs, K, compared to Config 1. It can be observed that with increase in K, there is steady improvement in PUF's stability, at all temperatures.

Fig. 4.10c shows stability of Config 5 and Config 6 both of which are not proposed to increase PUF stability, but to further emphasize the difference between pMOS and nMOS

with respect to bi-stable PUF stability. Config 5 uses parallel nMOSs, which averages the variation among nMOS transistors. As a result, the bi-stable cell is more reliant on pMOS variation to evaluate to logic 1 or logic 0, which is more prone to have unwanted V_{th} variations with temperature. Hence, we observe over 160% increase in the number of unreliable bits. This further illustrates that stability can be improved by exploiting nMOS variation over pMOS variation. Also, in Config 6, where both pMOS and nMOS transistors are used in parallel, it can be observed that there is a very small difference in PUF stability, suggesting that stability improvement is not an artifact of increased transistor sizing, but it is because of the differences between pMOS and nMOS.

Trade-off: Variability vs. Stability Typically, PUF systems observe a trade-off between variability and stability characteristics. In the case of SRAM PUFs, improved variability results from increase mismatch between cross-coupled inverters due to process variation. At the same time, this mismatch also results in increased variation in V_{th} due to temperature, resulting in less stability. In the configurations proposed in Section 4.2.3, by minimizing the impact of PV on pMOS, we have improved the stability. However, we do not observe an adverse effect on variability as the variation in nMOS transistors is adequate in maintaining the variability.

Power and Area

Table 4.4 shows the area and power consumption values for one SRAM cell. Reported are the power numbers averaged over the 100 chips simulated. For each chip, power is averaged from the time PUF is powered ON to the time when the SRAM cell stabilizes to logic 1 or logic 0. It can be observed that for both Config 3 and Config 4, which have longer transistors, there is a significant increase in the power consumption. This is because the SRAM cell takes longer to evaluate to logic 1 or logic 0 due to increased gate overlap capacitance, resulting in high short circuit power. For Config 2, there is a steady increase in the power consumption

Table 4.4: Power and area for SRAM configs

Configuration	Area (μm^2)	Power (nW)
Config 1 (Reference)	1.0	0.08
Config 2, K=2	1.3	0.13
Config 2, K=3	1.7	0.18
Config 2, K=4	2	0.23
Config 2, K=5	2.3	0.27
Config 3	4.3	0.20
Config 4	1.4	0.90

Table 4.5: Stability improvement at 130nm and 65nm

Technology	Stability Improvement (%)				
	-15C	5C	45C	65C	85C
130nm	47.82	42.76	45.03	43.39	41.00
65nm	87.12	92.57	77.41	50.35	57.96

with increasing K.

As SRAMs can have a variety of layout styles, we have reported the total active area of the transistors that form the SRAM cell, to rule out the SRAM layout style dependency. It can be observed that Config 3 with parallel and longer pMOSs has a much higher area consumption.

From Fig. 4.10 and Table 4.4, Config 2 has a better stability while not being power and area hungry. Config 4 proves only a small improvement over Config 2 and it is not recommended due to higher power and area consumptions.

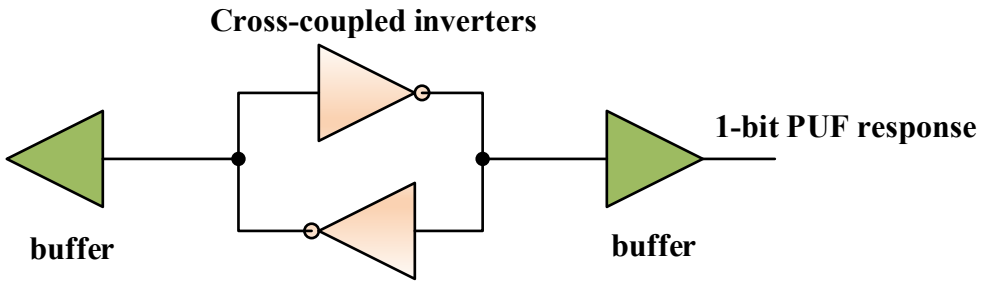

Figure 4.11: Abstract custom bi-stable cell.

Process Technology

Here, we observe the proposed stability improvement technique applied to different technology nodes of the same foundry. The simulation setup is similar to the 90nm setup described in section 4.2.3. SRAM PUF simulations have been performed in 130nm and 65nm technologies. For brevity, we have only shown the improvement for Config2 at K=5 compared to the reference SRAM design. Table 4.5 shows the improvement in stability across different temperatures for both technologies. It can be observed that on an average there is about 44% improvement at 130nm and about 73% improvement at 65nm.

4.2.5 ASIC Implementation

We implemented the proposed bi-stable PUF stability improvement technique in 90nm CMOS technology. The following subsections present the design and implementation, test setup, and experimental results of the ASIC.

Design and Implementation

In order to validate the proposed technique in silicon, we have designed PUFs with custom bi-stable cells that minimize the impact of pMOS variation in addition to reference bi-stable cells, which uses standard cross-coupled inverters. We have also designed PUFs with bi-

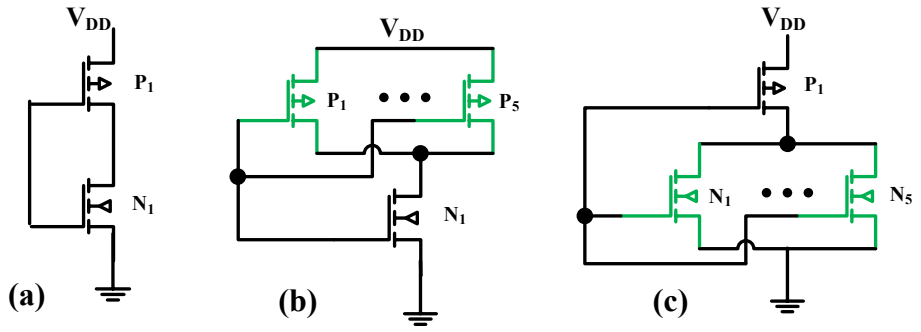

Figure 4.12: Inverter configurations (a) Reference inverter. (b) Inverter with parallel pMOSs. (c) Inverter with parallel nMOSs.

stable cells that minimize the impact of nMOS variation. The reason for this is to further show the difference between bi-stable PUFs that exploit pMOS variation versus the ones that exploit nMOS variation.

Custom cell design Fig. 4.11 shows the abstract bi-stable cell implemented in ASIC. Inverters in the cross-couple are varied to implement different custom bi-stable cells. It can be observed that buffers are used on both cross-couple nets to minimize systematic skew in the cell due to asymmetric wire loads on the cross-couple nets. Different inverters designed for the cross-couple are shown in Fig. 4.12. We have implemented three types of bi-stable cells using Cadence Virtuoso, namely, Reference, pMOS-dominant, and nMOS-dominant. A reference bi-stable uses a standard inverter (Fig. 4.12a), whereas, pMOS-dominant and nMOS-dominant cells use parallel pMOS (Fig. 4.12b) and parallel nMOS (Fig. 4.12c) inverters, respectively. A pMOS-dominant cell uses parallel pMOSs to minimize the impact of pMOS variation on the random outcome of the bi-stable cell. nMOS-dominant cells have a similar structure except that they use parallel nMOS instead of pMOS.

Fig. 4.13 a,b, and c show the layouts of the three custom bi-stable cells implemented. It can be observed that the cells are designed symmetrically. Each of the cells has a buffer on either end of the cell with the cross-couple inverters in between. Further, the routing with-in the cells is done symmetrically to minimize systematic skew. The configuration and area of

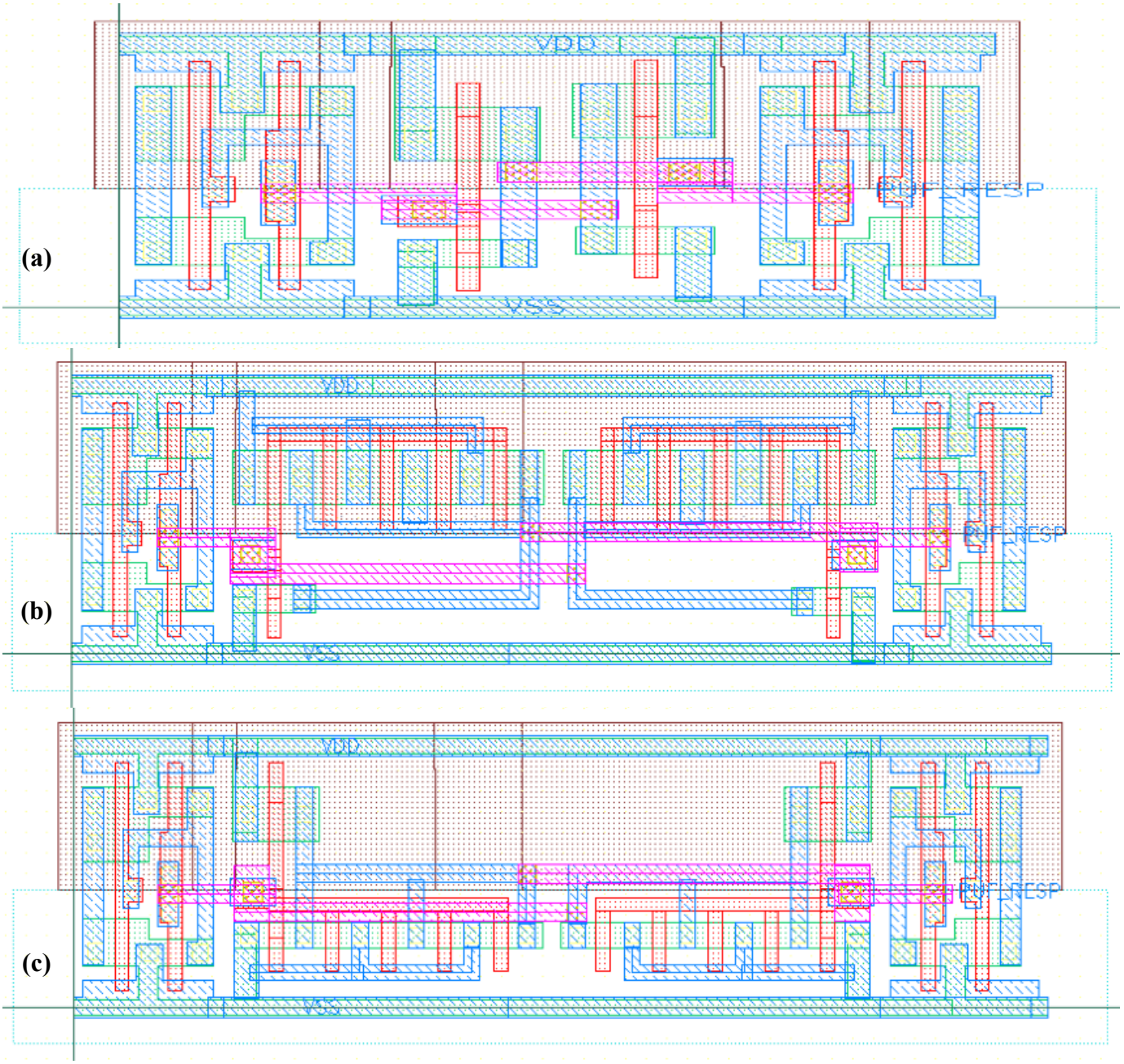

Figure 4.13: Bi-stable cell layouts (a) Reference. (b) pMOS-dominant. (c) nMOS-dominant.

the bi-stable cells implemented are shown in Table 4.6.

Table 4.6: Configuration and area of bi-stable cells

Cell type	# of pMOS/ inverter	# of nMOS/ inverter	Width (µm)	Height (µm)	Area ($µm^2$)
Reference	1	1	3.92	1.96	7.6832
pMOS-dominant	5	1	7.28	1.96	14.2688
nMOS-dominant	1	5	7.28	1.96	14.2688

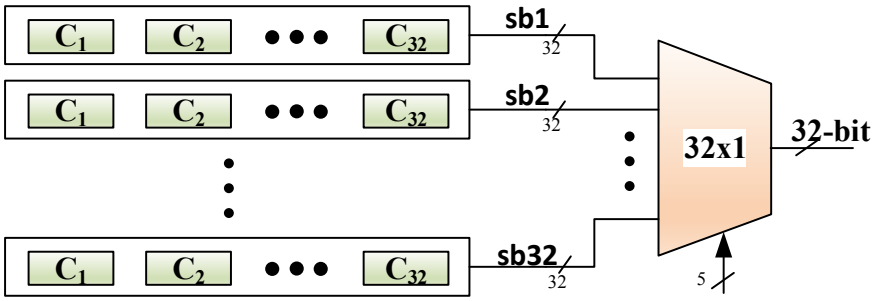

Figure 4.14: PUF block

PUF block Using the custom bi-stable cells presented in the previous paragraph, PUF blocks are designed, where each has 1024 bi-stable cells with multiplexers. The abstract design of a PUF block is shown in Fig. 4.14. It can be observed that the 1024 bi-stable cells are logically divided into 32 sub-blocks (sb1 to sb32), each with 32 cells, and a single sub-block's output is read with the help of multiplexers.

PUF module Fig. 4.15 shows the PUF module of the chip. It can be observed that there are two blocks with reference bi-stable cells, two blocks with pMOS-dominant cells, and two blocks with nMOS-dominant cells. The controller selects a puf block and a sub-block in it, and the chosen 32-bits are read out serially. Fig. 4.15 also shows power gates, one for each PUF block. Power to the PUF blocks that are not activated is shut off with the help of power gates. The layout of the full chip is shown in Fig. 4.16, where the PUF module is highlighted in white. The number of PUF related pins for the chip is 18.

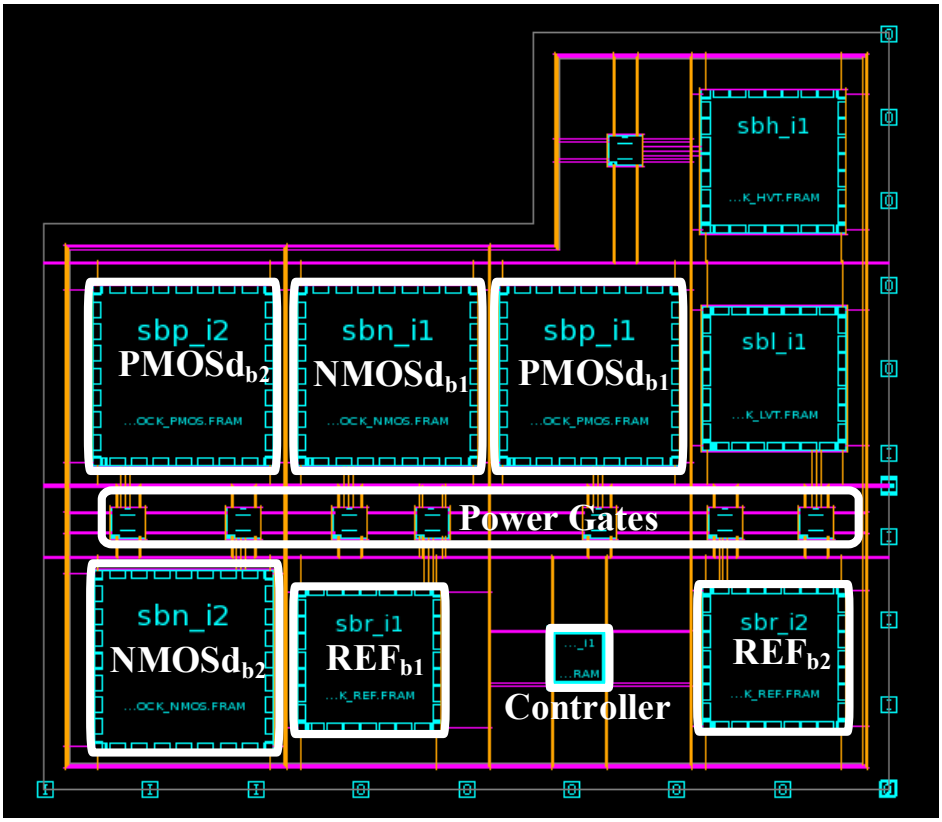

Figure 4.15: PUF module

Test Environment

In the previous subsection, we have presented the design of the PUF ASIC. Here, we outline the environment used for testing the chips.

The chips have been packaged using 72-pin ceramic pin grid array (CPGA) packages. Custom printed circuit board has been designed to house the packaged chips. We have automated the testing process using Labview, which provides input to the PUF and collects its responses with the help of a digital stimulus/response generator.

An overview of the testing environment is presented in Fig. 4.17. With the help of Labview environment (PC-based), we communicate with ASIC on the PCB through the digital stimulus/response generator. Further, we control the power supply to PUF blocks using

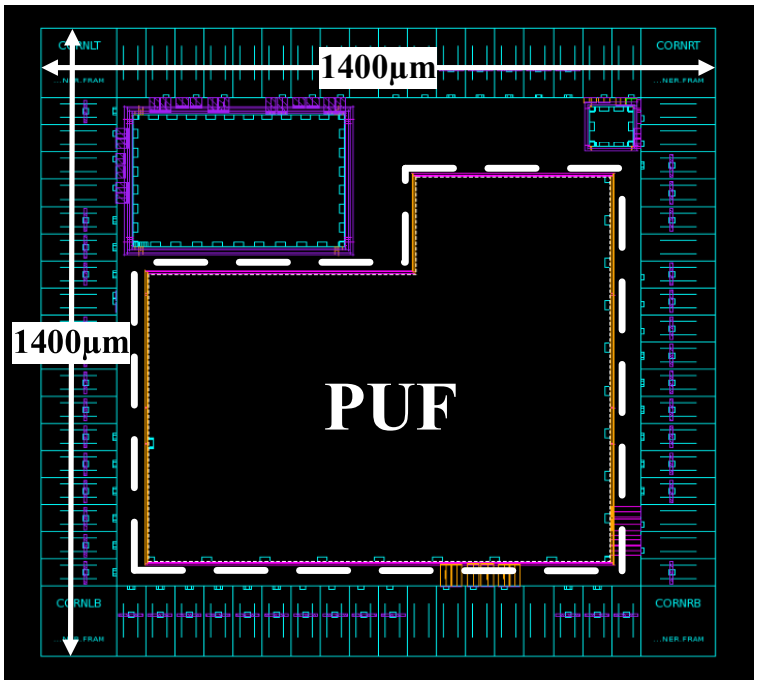

Figure 4.16: Full chip layout

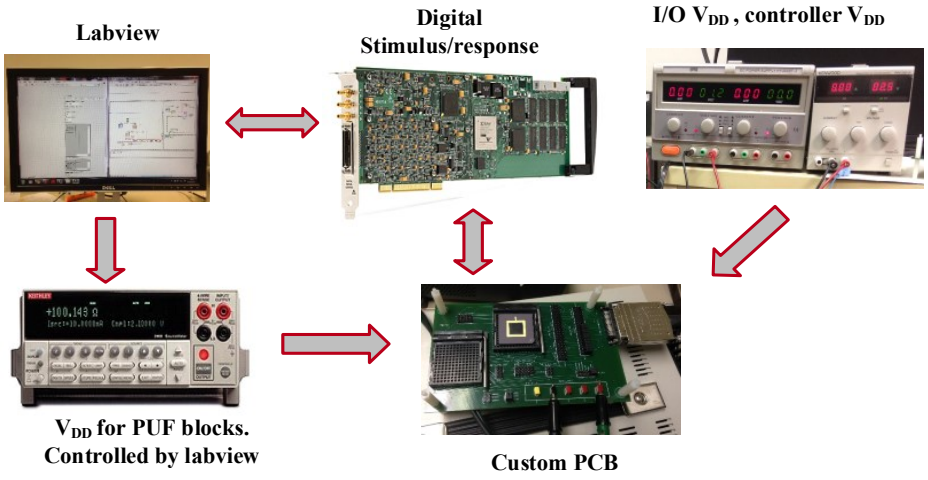

Figure 4.17: PUF ASIC test environment

Labview. This helps in controlling the power-up of the PUF blocks. In addition to these, we can also observe the power supplies for the controller and pads (I/O Vdd). The nominal V_{DD} for PUF blocks and controller is 1.2 V and I/O V_{DD} is 2.5 V.

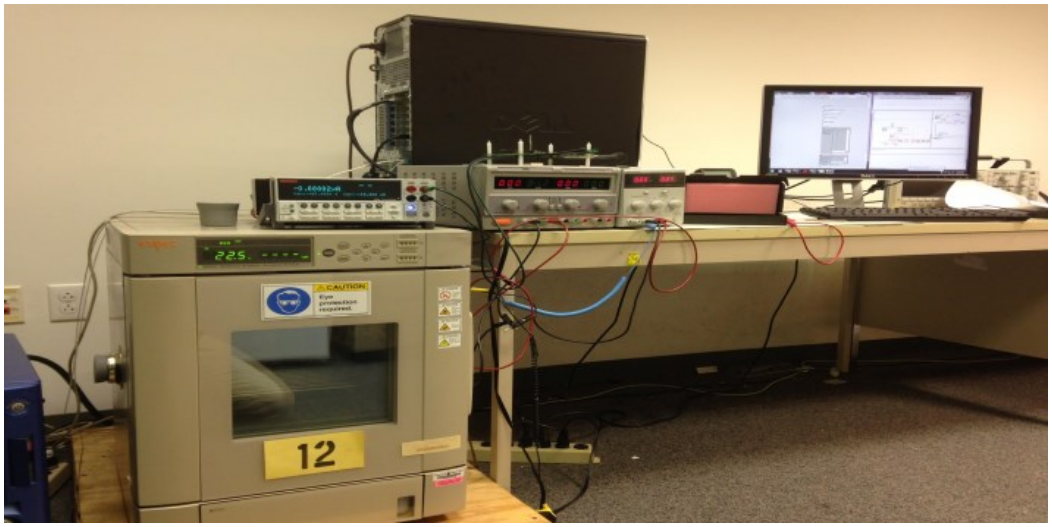

Figure 4.18: PUF ASIC test setup

Table 4.7: Stability at 25C

	REF (%)	PMOSd (%)	NMOSd (%)
Mean	91.0	94.8	90.5
Std	1.8	2.6	1.8

Fig. 4.18 shows the entire test setup. The figure shows a temperature chamber, which is used to control the operating temperature of the PUF. We have collected data from 21 chips at room temperature, and for 9 chips at 5C and 45C.

Experimental Results

Here, we present stability and variability results for the three types of bi-stable PUF variants implemented in ASIC.

Stability at 25C As mentioned earlier, stability is the quality of a PUF to produce consistent output. A bi-stable cell evaluates to logic 1 or 0 during its power up. Due to

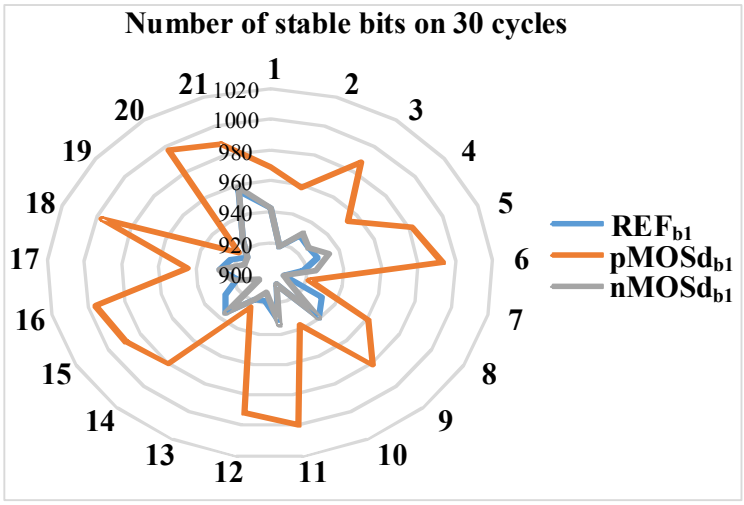

Figure 4.19: Number of stable bits at 25C

noises in the system such as Johnson-Nyquist noise, crosstalk, etc., the probability that a cell goes to a particular logic is not always 1. Traditionally, bi-stable cells are evaluated many times (multiple power ups) at nominal conditions to observe the stability of the cell.

In our experiments, for each of the 21 chips at room temperature, we have evaluated PUF responses for 30 cycles. In each cycle, PUF blocks are powered up, responses are collected, and powered back off. From Fig. 4.15, we can observe that each chip has 2 PUF blocks of each type (reference, pMOS-dominant, and nMOS-dominant). Effectively, there are 42 PUF blocks of each type with 1024-bits in each block.

In each block, we observe the number of bits that have consistently gone to either logic 1 or logic 0 in all 30 cycles. Table 4.7 shows the mean and standard deviation of the percentage of stable bits in the three PUF types. It can be observed that on average, a pMOS-dominant bi-stable PUF has about 94.8% stable bits as compared to 91% in a reference design. An nMOS-dominant PUF has shown slightly lower stability compared to a reference design. Fig. 4.19 shows the number of stable bits for three PUF blocks on the 21 chips for which data was collected. The contours in the plot represent the number of stable bits and numbers on the perimeter are the chip numbers. The maximum bit count can be 1024 (number of bits

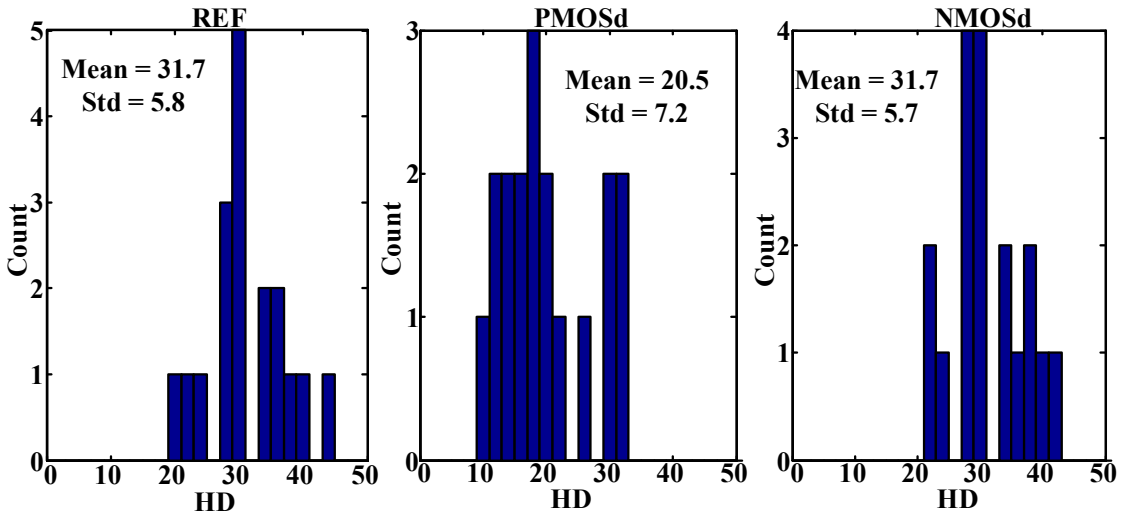

Figure 4.20: Stability of PUFs at 5C

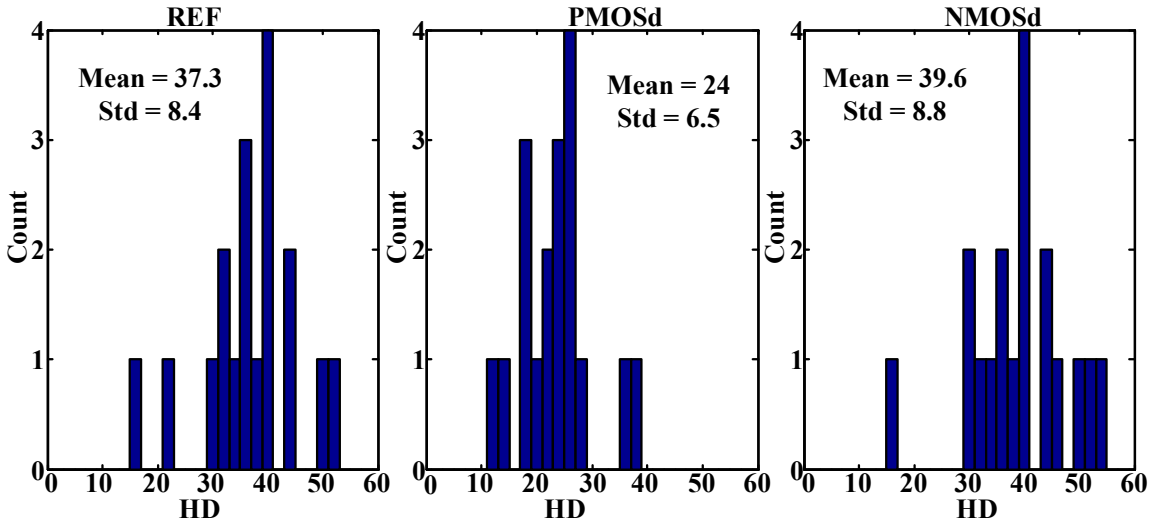

Figure 4.21: Stability of PUFs at 45C

per block). It can be observed that in all the chips, pMOS-dominant block ($pMOSd_{b1}$) has got higher number of stable bits compared to both reference (REF_{b1}) and nMOS-dominant block ($nMOSd_{b1}$).

Stability at 5C and 45C To observe stability of the PUFs to temperature variations, we have collected PUFs' responses at 45C and 5C and compared with the responses at 25C.

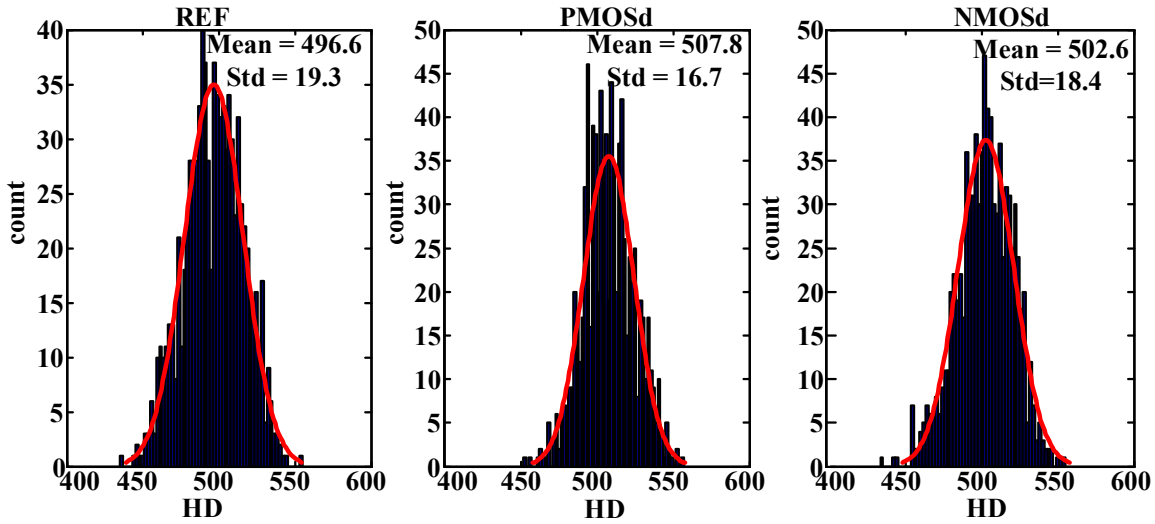

Figure 4.22: Variability of PUFs

Fig. 4.20 shows the histograms of hamming distances (HD) between PUF responses at 5C with responses at the room temperature for the three PUF types. It can be observed that the mean HD for a pMOS-dominant design is 20.5, whereas, it is 31.7 for both reference and nMOS-dominant designs. Similarly, Fig. 4.21 shows the stability of the PUFs at 45C. It can be observed that the mean HD goes down from 37.3 for a reference design to 24 for a pMOS-dominant design. Mean percentage improvements in stability are 35.4% and 35.6% at 5C and 45C, respectively.

Variability Fig. 4.22 shows the variability plots for the three PUF types. The number of PUF instances in each histogram of the plot is 42. So, each histogram has 861 (42 choose 2) HDs. As each PUF has 1024 bits, the ideal expected mean HD is 512. We observe a small skew from the ideal in all the designs. The mean HDs of the histograms are 496.6, 507.8, and 502.6 for the reference, pMOS-dominant, and nMOS-dominant designs, respectively. We believe that this could be because of the small population. Alternatively, this can also result from bias in the number of 1s and 0s in PUF responses.

Table 4.8: Area of PUF blocks

PUF Block	Width (µm)	Height (µm)	Area (μm^2)
REF	145.3	147.4	21417.2
PMOSd	186.2	188.1	35024.2

Area Table 4.8 shows the area for the reference and pMOS-dominated PUF blocks. For one block, which has 1024 bi-stable cells, we observe 63% increase in the area with the proposed technique compared to the reference.

4.2.6 Summary of Bi-stable PUF stability discussion

In this technique, we showed that the stability of bi-stable PUFs can be improved by exploiting the inherent differences between pMOS and nMOS in the context of process variation and temperature variations. We proposed circuit-level techniques to improve the stability of bi-stable PUFs with no related post-processing required. We have implemented the proposed technique in 90nm CMOS technology and showed the improvement in stability to temperature variations.

In this chapter, so far, we have looked into stability of PUFs to temporary environmental conditions. Recently, a lot of research has gone into studying the effects of aging on integrated circuits. Reliability is particularly aggravated as the supply voltage is not scaling as aggressively as the device dimensions resulting in high electric fields. Aging mechanisms like Bias Temperature Instability (BTI) and Hot Carrier Injection (HCI) have become one of the major reliability concerns. Hence, in the following section, we look into the stability of PUFs to aging degradation.

4.3 Study of IC Aging

One of the major sources of unreliability in the technology nodes 90nm and below is device aging. Aging is primarily due to phenomena like Bias Temperature Instability (BTI) and Hot Carrier Injection (HCI). In this section, we study the effect of device aging on the stability of Ring Oscillator PUFs for different circuit-level choices and operating conditions. We observe that most of the PUF's aging instability happens early in its lifetime. Due to the typical differential nature of PUF structures, stability does not change significantly with age. Further, a high correlation has been observed between instability that is caused due to aging and instability that is caused due to temperature. In various RO-PUF setups and operating conditions, we observe that around 4% of the PUF bits are prone to instability due to aging.

The contributions of this work are as follows:

- We study the impact of circuit aging on the stability of Ring Oscillator PUFs. Different circuit-level choices in building an RO-PUF are studied in the context of aging.
- Unlike earlier works, we consider the frequency of authentication as a parameter in observing the stability to aging degradation. This presents a more realistic approach to understanding PUF aging instability. With this approach, we observe about 35% reduction in the estimated bit error rate compared to the situation where PUF is assumed to be constantly under stress.
- For the first time, we study the correlation between RO-PUF instabilities that arise due to temperature and aging. Our results show that 70% of the PUF bits that become unstable due to aging are also unstable under temperature variations.

The rest of this section is organized as follows. In Section 4.3.1, major IC aging phenomena are discussed. Related work is presented in Section 4.3.2. Section 4.3.3 presents the simulation setup, the methodology of evaluation of PUF stability, and different RO-PUF setups

studied in this work. In Section 4.3.4, parameters that are varied to observe the effect of aging are presented along with their results. In the same section, we talk about the correlation between aging instability and temperature instability. Section 4.3.5 presents the conclusion.

4.3.1 Background on IC aging

Changes that occur in the gate oxide are the primary source of aging in integrated circuits. In the following subsections, we briefly discuss the major aging phenomena such as Negative Bias Temperature Instability (NBTI), Hot Carrier Injection (HCI), and Time Dependent Dielectric Breakdown (TDDB).

Negative Bias Temperature Instability (NBTI)

NBTI in pMOSs is a major reliability issue in integrated circuits. When an electric field is applied across the gate oxide of a pMOS, traps are generated at the $Si-SiO_2$ interface [37,49]. Another effect is the trapping of holes in the dielectric bulk. These effects increase the magnitude of V_{th} of pMOS, lowering its drain current and hence slowing down the circuits. When the number of traps are high enough, the resulting slowdown of the gates may result in failures in ICs [49]. NBTI can shift the V_{th} of the pMOS transistors by as much as 50mV over ten years, which translates to over 20% increase in the circuit delay [7, 38]. However, when the stress is removed or when the device is turned off, part of the V_{th} shift that happens due to hole trapping can be recovered.

Since the introduction of high-k dielectrics, nMOS transistors are affected by degradation in the form of Positive Bias Temperature Instability (PBTI) [12]. Nevertheless, the degradation due to PBTI is less prominent compared to NBTI.

Hot Carrier Injection (HCI)

During the switching of a transistor, some of the high energy charge carriers get injected into the gate oxide. HCI is a major reliability issue for nMOS as electrons have higher mobility than holes [49]. HCI results in a decreased nMOS drain current, while the drain current of pMOS increases. The amount of HCI-related aging degradation of a circuit is directly dependent on the activity of the circuit. Circuits operating at higher V_{DD}, elevated temperatures, and with high switching activity are very susceptible to HCI-related aging.

Time Dependent Dielectric Breakdown (TDDB)

When the density of traps created in the dielectric is high, it can cause the dielectric to breakdown, resulting in a conductive path across the gate oxide. This is a problem in the lower technologies as supply voltages and the associated electric fields have not been scaling as aggressively as the device dimensions. Because of this, only a smaller density of traps are sufficient for the oxide to breakdown. However, with the introduction of high-k gates, the effect of TDDB has become less severe.

4.3.2 Related Work on PUF aging

Although aging phenomena have been studied in a number of works, not much work has been performed in the context of PUFs. [21] proposed software-based techniques to detect and prevent aging-related instability in PUFs. They use Feige-Fiat-Shamir identification scheme to detect when aging starts to affect the stability, and update the golden response table. In a different approach using the Merkle hash trees, unstable bits are prevented by controlling the valid life time of challenge/response pairs. In [27], impact of aging has been reported for SRAM PUF after a few reads and writes to SRAM cells. Extended usage or accelerated aging has not been performed on the PUF. Maiti et al. [31] have performed accelerated aging on RO-PUFs implemented in FPGAs. Although this work gives a good insight into

aging phenomena on FPGA PUFs, it is limited in its flexibility to mimic the activity of a PUF. In other words, it does not consider frequency at which PUF circuits are turned on. Hence, it provides a pessimistic approach to observing the effect of aging. This work differs from prior work on aging in its closeness to mimic a real PUF application. We consider the frequency at which a PUF is used for authentication as it significantly impacts the reliable life time of a PUF. We study the impact of aging on RO-PUF stability for different design options and operating conditions. Further, we provide an insight into the correlation between temperature and aging in the context of PUF stability.

4.3.3 Methodology

In this section, we first outline the simulation setup for RO-PUF operation and aging. Following that, we will present the methodology of evaluating the stability of the PUF. In the last subsection, PUF operating setups that are used to observe the impact of different parameters on aging are discussed.

Simulation Setup

Different Ring Oscillator PUF (RO-PUF) setups have been simulated using HSPICE. HSPICE MOS Reliability Analysis (MOSRA) has been used to incorporate Negative Bias Temperature Instability (NBTI), Positive Bias Temperature Instability (PBTI), and Hot Carrier Injection (HCI) aging effects. 90nm Predictive Technology Models [1] have been used to perform the simulations. The nominal supply voltage (V_{DD}) is 1.2V. Monte Carlo (MC) statistical methodology has been used to implement process variation. Mukhopadhyay et al. [34] show that the variations in channel length (L), oxide thickness (t_{ox}), and a few other less prominent variations can be summed up as variation in the threshold voltage (V_{th}). Hence, we use variation on the threshold voltage (V_{th}) to implement process variation (PV). Both global and local variations on V_{th} are implemented as Gaussian distributions using

the Monte Carlo DEV/LOT approach offered in HSPICE [2]. The frequency of ROs are measured from the simulations and post processing to generate PUF IDs is performed using Matlab.

Aging simulation occurs in two phases - pre-stress and post-stress [2]. During the pre-stress phase, the degradation in the V_{th} of all the transistors in the chip are measured based on their operating voltage, temperature, activity, etc. In the post-stress phase, the degradation calculated during pre-stress is taken into account to evaluate the effect of aging. Operating environment is independently controlled for pre-stress and post-stress phases to simulate different PUF operating conditions.

PUF Stability Evaluation

In each RO-PUF chip, 50 identical ring oscillators (RO) are simulated. 50 such chips are simulated to get a good distribution of process variations. ROs with minimum sized inverters are used because PUFs are typically designed with minimum sized devices to maximize the effect of process variation. As shown in Fig. 2.3, in an RO-PUF, a single bit is generated by comparing the frequencies of any two ROs in the PUF. As 50 ROs are simulated in each chip, we obtain a 1225-bit (C_2^{50}) ID for each chip. Although all of the 1225 ID bits are not independent, for the sake of observing instability due to aging, we use the entire C/R space available in a chip, i.e.1225 bits.

Stability of a PUF is the quality of producing the same output irrespective of PUF's age or operating environment. Hamming Distance (HD) is typically employed to observe PUF's stability. Stability of a PUF is evaluated as its Bit Error Rate (BER). In other words, it is the probability of bit flips in the ID of the PUF compared to the reference ID of the PUF. The stability at a particular PUF setup, and operating conditions (OC) is given by the average BER of the 50 PUFs simulated, which is given by Eq. 4.11

Table 4.9: Process variation levels

PV level	Inter-chip (3σ V_{th})	Intra-chip (3σ V_{th})
pv_low	3%	1.5%
pv_nom(nominal)	6%	3%
pv_high	8%	6%

Table 4.10: RO-PUF reference ID setups

Reference ID setup	Number of RO stages	Supply (V_{DD}) (V)	Temperature (C)	PV level
1	11	1.2	25	pv_low
2	11	1.2	25	pv_nom
3	11	1.2	25	pv_high
4	11	0.8	25	pv_nom
5	11	0.4	25	pv_nom
6	21	1.2	25	pv_nom
7	5	1.2	25	pv_nom

$$BER_{OC}(\%) = \frac{\sum_{i=1}^{N}(HD_{REF,OC})_i \times 100}{N \times L_{ID}} \tag{4.11}$$

Where, N is the number of chips in the population (in our case, 50), $HD_{REF,OC}$ is the hamming distance between the ID of a chip at the reference operating condition measured during the PUF registration (REF) and the ID at any specific PUF operating conditions (OC), i is the chip index, and L_{ID} is the length of the ID (in our case, 1225).

PUF Operating Setups

In this work, we observe the effect of aging for different operating conditions and circuit-level options. The reference IDs of chips are established under certain conditions present at the time of ID registration. Chip IDs are again evaluated for a specific setup, and operating conditions (e.g. PUF aged for 1 year). Stability is evaluated by comparing these IDs with

Table 4.11: RO-PUF operating setups

RO PUF Operating Setup*	Number Of stages in RO	PUF Activity	Supply (V_{DD})		Temperature		PV level	Reference ID setup
			Pre-stress (V)	Post-stress (V)	Pre-stress (C)	Post-stress (C)		
stp_pv1	11	60 sec	1.2	1.2	25	25	pv_low	1
stp_pv2	11	60 sec	1.2	1.2	25	25	pv_nom	2
stp_pv3	11	60 sec	1.2	1.2	25	25	pv_high	3
stp_on	11	ON always	1.2	1.2	25	25	pv_nom	2
stp_min	11	60 sec	1.2	1.2	25	25	pv_nom	2
stp_hr	11	1 hour	1.2	1.2	25	25	pv_nom	2
stp_day	11	1 day	1.2	1.2	25	25	pv_nom	2
stp_t25	11	60 sec	1.2	1.2	25	25	pv_nom	2
stp_t85	11	60 sec	1.2	1.2	85	85	pv_nom	2
stp_tn40	11	60 sec	1.2	1.2	-40	-40	pv_nom	2
stp_v1.32	11	60 sec	1.32	1.32	25	25	pv_nom	2
stp_v1.08	11	60 sec	1.08	1.08	25	25	pv_nom	2
stp_v1.2	11	60 sec	1.2	1.2	25	25	pv_nom	2
stp_v0.8	11	60 sec	0.8	0.8	25	25	pv_nom	4
stp_v0.4	11	60 sec	0.4	0.4	25	25	pv_nom	5
stp_11	11	60 sec	1.2	1.2	25	25	pv_nom	2
stp_21	21	60 sec	1.2	1.2	25	25	pv_nom	6
stp_5	5	60 sec	1.2	1.2	25	25	pv_nom	7

* In this column and rest of the section, *setup* has been abbreviated as *stp*

the reference IDs, as given by Eq. 4.11. We have simulated 18 different RO-PUF setups to observe the effect of aging at different levels of process variation, activity of the PUF, temperature, supply voltage, and configuration of ROs.

Table 4.9 shows the levels of process variation used in this work along with their 3σ inter-chip and intra-chip threshold voltage (V_{th}) standard deviation levels. We believe, *pv_nom* is the level that should closely represent the variation levels from a foundry as we closely matched the variation levels in *pv_nom* to the ones in a leading 90nm technology data. Table 4.10 shows different reference ID registration setups. It can be observed that reference IDs have been taken for different process variation levels, supply voltage, and the number of stages in ROs. Table 4.11 summarizes 18 different RO-PUF setups simulated in this work for obtaining PUF IDs. For each setup, the table shows the number of RO inverting stages, the activity of the PUF showing the time interval between successive PUF ID evaluations,

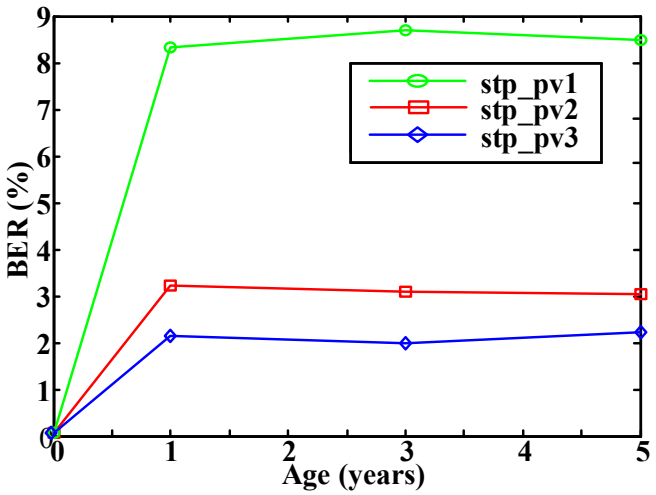

Figure 4.23: BER with respect to age at different process variation levels

supply and temperature conditions at both pre-stress and post-stress stages, and the level of process variation used. The table also shows the reference ID (REF) to which the ID at this setup is compared, to obtain the BER for that setup. These setups are explained in more detail in section 4.3.4. For each of the setups shown in Table 4.11, IDs of the PUFs have been computed at the ages of 1 year, 3 years, and 5 years.

4.3.4 Results

In this section, we present different parameters that are considered to observe the impact of aging on stability. In each subsection, the RO-PUF setups used to observe the impact of a parameter are described, along with the results.

Process Variation

PUFs exploit the mismatch between identical circuit elements that occurs due to process variation (PV). Higher the PV, higher is the mismatch between identical circuit elements. PUFs with higher mismatch are more tolerant to temporary variations in the environmental conditions, which results in more stable PUFs. We observe the same in the impact of aging

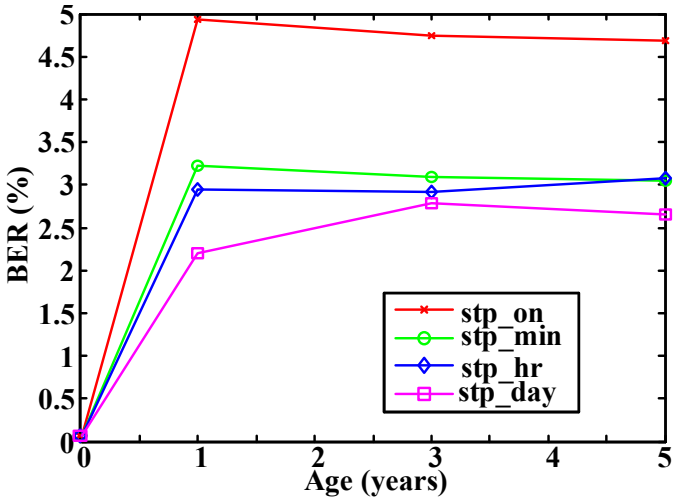

Figure 4.24: BER with respect to age at different PUF activity levels

as well. From Table 4.11, the setups used for this experiment are *stp_pv*1, *stp_pv*2, and *stp_pv*3, which have a low, nominal and high levels of PV, respectively. Between *stp_pv*1, *stp_pv*2, and *stp_pv*3, the operating conditions are identical and they only differ by the level of variation assumed in the process. Fig. 4.23 shows the BER of the PUF with respect to age for these setups. It can be observed that for *stp_pv*1, which has a low level of process variation (*pv_low*), BER reaches around 9%. For *stp_pv*2, which represents the nominal PV level, BER is around 3%.

From Fig. 4.23, for any given level of PV, it can be observed that BER is about the same for ages 1, 3, and 5 years. Further, BER slightly drops with increased age in some cases, for instance in *stp_pv*3 going from 1 year to 3 years of age. The reason for this is explained under BER discussion, which is at the end of this subsection.

For the rest of the work, we use process variation level *pv_nom*, which can be the typical variation seen in a process technology.

PUF Activity

Since the activity of a PUF directly impacts the associated aging degradation, we vary the frequency at which the PUF is authenticated. The PUFs are turned on once every minute, hour, and day as shown by the setups *stp_min*, *stp_hr*, and *stp_day* in Table 4.11. In each of these setups, the PUF turns on, evaluates the ID, and then turns off until the next cycle. These setups are compared with the worst case aging setup, *stp_on*, where the PUF is always on.

In RO-PUFs, if a pair of RO frequencies are very close to each other, they can be allowed to count longer to observe a finite difference in the counter values. The time required to evaluate an ID is set such that the pair of ROs in the PUF, whose frequencies are closest to each other would evaluate to have a finite difference in their counter values. In our experiments, this evaluation time is set at 0.5ms. So, this is the time the ROs of a PUF are on to compute the ID, and then go back to sleep. While in sleep, there is no aging stress on the PUF.

Fig. 4.24 plots the BER against the age for RO-PUFs for the activity setups discussed. Although a fully on PUF (*stp_on*) has a higher BER than other setups, BERs for much lesser activity such as once a day (*stp_day*) still has BER of over 2%. It can also be noticed that the fully on assumption (similar to accelerated aging tests) could be very pessimistic in estimating the instability caused due to aging. For the rest of this work, we have chosen a PUF activity of once every minute.

Temperature

Here, we observe the effect of operating temperature on PUF's stability, in the presence of aging. PUFs are operated at 85C, 25C, and -40C, and the IDs are measured at the ages of 1year, 3years, and 5 years. From Table 4.11, the setups used for this are *stp_t*85, *stp_t*25, and *stp_tn*40 for PUFs stressed at 85C, 25C, and -40C, respectively. The post-stress PUF ID measurements are also performed at the respective temperatures of 85C, 25C, and -40C.

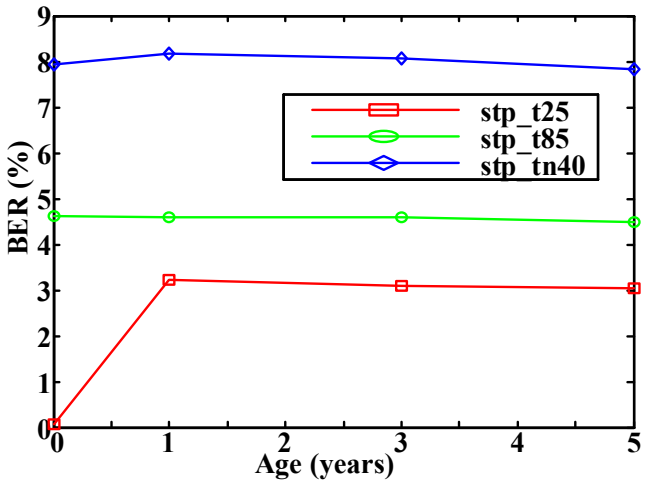

Figure 4.25: BER with respect to age at different operating temperatures

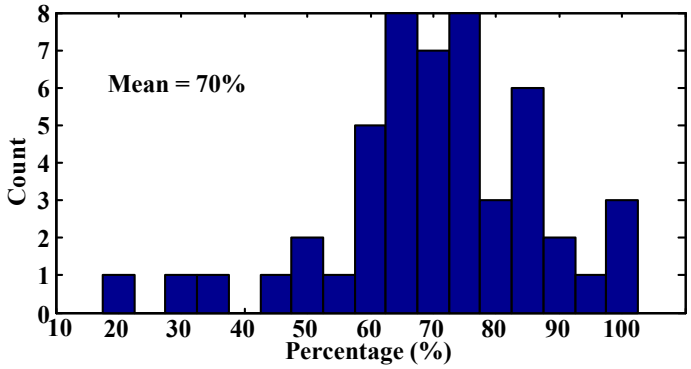

Figure 4.26: Histogram of the correlation of instabilities between aging and temperature variations

Fig. 4.25 shows the BER plot for these setups. It can be observed that for new chips (age of 0 years), the BERs of the PUFs operated at 85C and -40C are high at around 5% and 8%, respectively. These errors are due to temperature instability only. At other ages, the BER does not seem to increase over the years.

In order to understand the above phenomenon better, we observe the correlation between the instability that is caused due to temperature and the instability that is caused due to aging. In the literature, a lot of work has gone into making PUFs more reliable to temperature. If there is a correlation between the RO pairs that become unstable due to aging and the RO pairs that become unstable due to temperature, it might be possible the same stability

Table 4.12: Correlation between temperature and aging

Correlation type	Age = 1year		Age = 3 years		Age = 5years	
	Mean	Std	Mean	Std	Mean	Std
new chips (85C and -40C) aged chips (25C)	70.26%	16.55%	72.37%	15.13%	67.69%	13.73%
new chips (85C and -40C) aged chips (85c and -40C)	63.27%	13.03%	62.73%	11.04%	64.73%	10.15%

improvement techniques can be applied to deal with both instabilities.

For each of the 50 chips simulated, we observe all the ring oscillator pairs that become unstable due to temperature variations. In our simulations, RO PUFs have been subjected to the temperatures of 85C and -40C. After that, the PUFs are aged for a period of 1 year at nominal operating conditions with an activity of one evaluation per minute. For all the aged chips, ring oscillator pairs that become unstable are again accounted for.

Fig. 4.26 shows the histogram of percentage of RO pairs that are unstable due to aging that are also unstable with temperature variations, for the 50 chips simulated. For example, in a chip, if the total number of RO pairs that become unstable due to aging is M, and of which, N are unstable to temperature variations, the reported percentages are $N \times 100/M$. We observe a mean percentage of 70% among 50 chips with a standard deviation of 16%. In other words, in a population of chips, 70% of the bits that become unreliable with aging are also unstable to temperature variations. The histogram suggests that there are only a few chips, where instability due to aging is not closely related to instability to temperature.

In addition to the above observation, we observe correlation between instability due to temperature in new chips and instability in aged chips with temperature variations. The chips are aged at nominal conditions, but the measurements are made at temperatures of 85°C and -40°C. Table 4.12 shows the mean and standard deviations (std) of the correlation histograms of both cases for PUFs aged by 1, 3 and 5 years. It can be observed that the mean percentage across all ages for both correlation scenarios is over 62%. These results show that

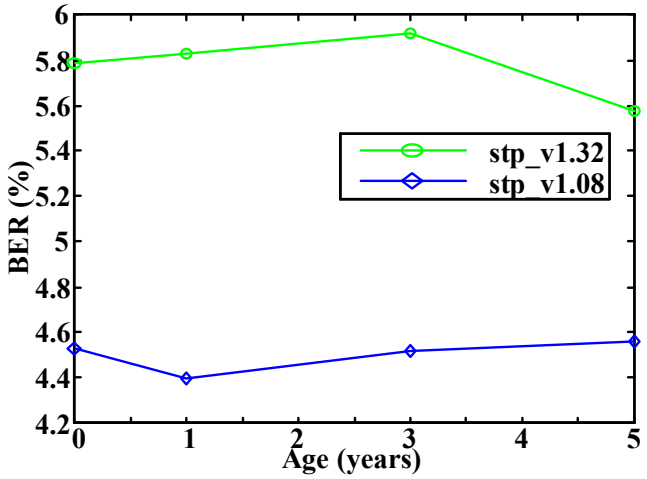

Figure 4.27: BER with respect to age in the presence of V_{DD} noise

for PUFs equipped with dealing with temperature instability, the net aging instability can be much less severe compared to what was predicted initially.

Supply Voltage

In this subsection, we observe the impact of supply voltage noise and supply voltage scaling on the stability of PUFs, in the presence of aging.

Effect of supply voltage noise Here, to observe the effect of supply noise, we vary the nominal supply voltage by 10%, and observe PUF's stability. From Table 4.11, the setups used to observe this are *stp_v*1.32 and *stp_v*1.08. Fig. 4.27 shows the BER with respect to age in the presence of supply noise. The baseline BER for new chips are around 4.6% and 5.8% for *stp_v*1.32 and *stp_v*1.08, respectively. It can be observed that the effect of supply noise in the presence of aging is limited as BER is less than 0.2% from the baseline.

Effect of the supply voltage scaling Here, we observe the impact of supply voltage scaling on the PUF as supply voltage (V_{DD}) is one of the major sources of aging stress in ICs. RO-PUFs have been operated at V_{DD}s of 1.2V, 0.8V, and 0.4V as shown by the setups

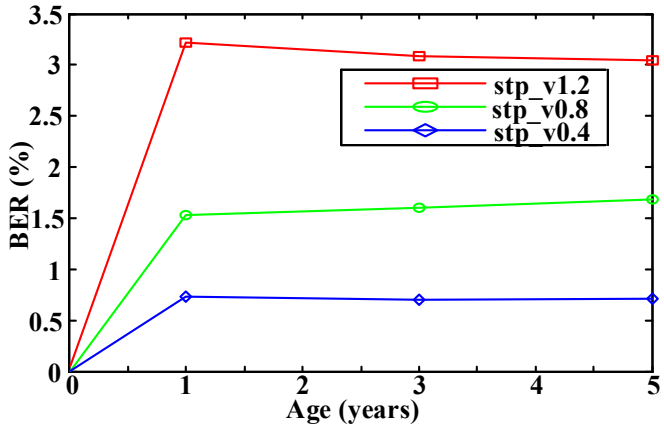

Figure 4.28: BER with respect to age for RO-PUFs at different operating V_{DD}

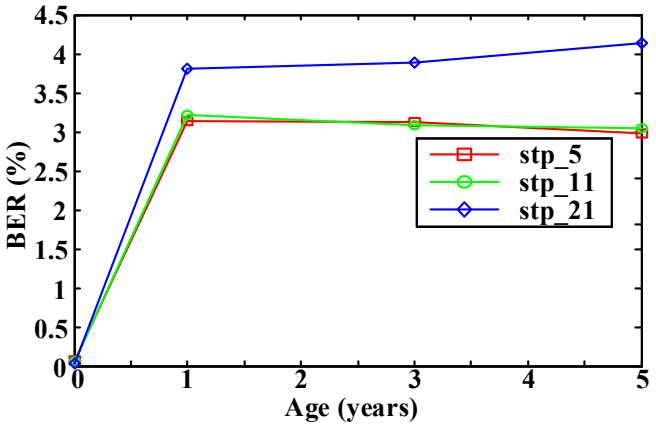

Figure 4.29: BER with respect to age for different RO stage lengths

*stp_v*1.2, *stp_v*0.8, and *stp_v*0.4, respectively, in Table 4.11. Fig. 4.28 shows the impact of V_{DD} on the stability of RO-PUFs. As it can be predicted, PUF operated at a higher V_{DD} has higher BER. At a near-threshold supply voltage of 0.4V, the BER is as low as 0.7%.

RO Configuration

Ring Oscillators (ROs) with higher number of inverting stages tend to have less variation among their frequencies. This is because with higher inverting stages, the variations among the inverter delays cancel out each other. As the RO frequencies in the PUF are very close

to each other, such PUFs are typically more susceptible to environmental variations. Here, we observe the effect of number of RO inverting stages on stability in the presence of aging. Fig. 4.29 shows BERs for RO PUFs implemented with 5, 11 and 21 stage ROs. The 21-stage PUF has slightly higher BER than 5-stage or 11-stage PUF.

BER Discussion

In each of the Figs. 4.23, 4.24, 4.25, 4.27, 4.28, and 4.29, it can be observed that the BER of the PUFs are nearly identical for ages of 1, 3, and 5 years. In regular CMOS circuits, with aging, circuits become less stable as they slow down. However, the advantage with PUFs is that, as two identical circuit elements (ROs in an RO-PUF) are compared to generate a PUF ID bit, they are likely to produce the correct ID bit since the ROs are aged under similar environmental conditions. However, if the frequencies of ROs under comparison are close to one another, they might undergo slightly different aging degradation resulting in unstable PUF ID bits. Hence, in spite of the frequencies of ROs going down due to aging, PUF BERs are nearly identical between ages 1, 3 and 5 years. Further, we observe the BER increase much before 1 year. For illustration purposes, discrete points of 1, 3, and 5 years are chosen.

4.3.5 Summary of PUF aging discussion

In this section, we studied the impact of aging on RO PUFs for different circuit-level choices and operating conditions. The stability of the PUF has been studied, varying the parameters like process variation, activity of the PUF, temperature, supply voltage, and the number of RO stages. We observed that due to inherent nature of comparing identical circuit elements in PUFs, and the fact that all ROs undergo aging under similar conditions, RO PUFs are fairly tolerant to aging. Also, ICs with high process variation are more tolerant to aging related PUF instability. Further, there is a significant correlation between PUF instabilities that are caused due to aging and temperature, which can be leveraged by PUF designers in

designing stable PUFs.

4.4 Conclusion of technology-dependent techniques

In this chapter, we proposed three technology-dependent circuit-level techniques and analysis for PUFs. In the first technique, we proposed Leakage-PUF (L-PUF), which employs the architecture of an RO PUF, but uses efficient analog components. L-PUF offers very high stability, has small area, and is more suitable for resource constrained applications. In the second technique, we proposed a stability improvement method for bi-stable PUFs, which are apt for key generation applications. Unlike the techniques proposed in the literature, which use expensive post-fabrication steps, our approach is circuit-level, which has significantly lower cost. In the third part, we looked into device aging, which is a major long-term reliability concern in ICs. We presented the effect of device aging on ring oscillator PUFs for different circuit-level choices and operating conditions.

Chapter 5

Conclusion

In this dissertation, we proposed different technology-independent and technology dependent techniques to improve the security, stability, and efficiency of silicon physical unclonable functions (PUFs).

As part of the technology-independent techniques, we proposed a novel PUF design called Select-ArbRO PUF (S-ArbRO PUF), which brings together the good qualities of Arbiter PUFs and RO PUFs. We showed that S-ArbRO PUF is easy to build and it has a very large Challenge/Response (C/R) space in a given area. Two variants, S-ArbRO-4 and S-ArbRO-2, of the proposed design were presented. Both PUFs have good variability and stability characteristics. Further, we showed that the increased C/R space is also resistant to modeling by implementing a logistic regression-based modeling attack.

Under technology-dependent methods, we investigated multiple techniques to improve stability and efficiency of PUF designs. In one approach, we proposed an efficient silicon PUF called Leakage-PUF (L-PUF) that can be fabricated on a standard CMOS process. L-PUF has a similar architecture to that of an RO PUF, where the power hungry digital components are replaced with more efficient analog components. Our results showed that L-PUF is more stable, less power hungry, and smaller, while maintaining the same level of entropy as an

RO PUF.

In another technique, we focused on improving the stability of bi-stable PUFs to temperature variations. We showed that the stability of bi-stable PUFs can be improved by exploiting the inherent differences between pMOS and nMOS transistors. Unlike the works in the literature, which predominantly rely on expensive post-manufacturing characterization, we proposed circuit-level techniques that are easy to implement and inexpensive. In addition to the simulation results, we have implemented an ASIC in 90nm CMOS technology. Our experiments showed about 35% improvement in the stability to temperature variations.

In the third technology-dependent work, we studied the impact of aging on Ring Oscillator PUFs. We observed that due to inherent nature of comparing identical circuit elements in PUFs, and the fact that all ROs undergo aging under similar conditions, RO-PUFs are fairly tolerant to aging. We used circuit-level simulations to evaluate the stability of silicon PUFs to aging degradation. Further, we showed that there is a significant correlation between PUF instabilities that are caused due to aging and temperature, which can be leveraged by PUF designers in developing stable PUFs.

Based on our work, we believe that technology-independent techniques are better candidates for improving overall efficiency of the designs in terms of both operation and implementation costs. However, with regards to improving the stability of PUFs, it is cost-effective to use technology-dependent techniques.

Bibliography

[1] Predictive Technology Model (PTM). `ptm.asu.edu/`.

[2] Synopsys HSPICE Documentation 2012.

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

[4] M. Bhargava, C. Cakir, and K. Mai. 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*, pages 106–111, 2010.

[5] M. Bhargava, C. Cakir, and K. Mai. Comparison of bi-stable and delay-based physical unclonable functions from measurements in 65nm bulk cmos. In *Custom Integrated Circuits Conference (CICC), 2012 IEEE*, pages 1–4, 2012.

[6] M. Bhargava, C. Cakir, and K. Mai. Reliability enhancement of bi-stable pufs in 65nm bulk cmos. In *Hardware-Oriented Security and Trust (HOST), 2012 IEEE International Symposium on*, pages 25–30, 2012.

[7] S. Borkar. Electronics beyond nano-scale cmos. In *Design Automation Conference, 2006 43rd ACM/IEEE*, pages 807–808, 2006.

[8] C. Bösch, J. Guajardo, A.-R. Sadeghi, J. Shokrollahi, and P. Tuyls. Efficient helper data key extractor on fpgas. In *Proceeding sof the 10th international workshop on Cryptographic Hardware and Embedded Systems*, CHES '08, pages 181–197, Berlin, Heidelberg, 2008. Springer-Verlag.

[9] 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*, pages 39–44, New York, NY, USA, 2009. ACM.

[10] M. Cortez, A. Dargar, S. Hamdioui, and G.-J. Schrijen. Modeling sram start-up behavior for physical unclonable functions. In *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2012 IEEE International Symposium on*, pages 1–6, 2012.

[11] C. Costea, F. Bernard, V. Fischer, and R. Fouquet. Analysis and enhancement of ring oscillators based physical unclonable functions in fpgas. In *Reconfigurable Computing and FPGAs (ReConFig), 2010 International Conference on*, pages 262–267, 2010.

[12] R. Degraeve, M. Aoulaiche, B. Kaczer, P. Roussel, T. Kauerauf, S. Sahhaf, and G. Groeseneken. Review of reliability issues in high-k/metal gate stacks. In *Physical and Failure Analysis of Integrated Circuits, 2008. IPFA 2008. 15th International Symposium on the*, pages 1–6, 2008.

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

[14] I. Filanovsky and A. Allam. Mutual compensation of mobility and threshold voltage temperature effects with applications in cmos circuits. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 48(7):876–884, 2001.

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

[16] B. Gassend, D. Lim, D. Clarke, M. van Dijk, and S. Devadas. Identification and authentication of integrated circuits. *Concurrency and Computation: Practice and Experience*, 16(11):1077–1098, 2004.

[17] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls. Fpga intrinsic pufs and their use for ip protection. In *Proceedings of the 9th international workshop on Cryptographic Hardware and Embedded Systems*, CHES '07, pages 63–80, Berlin, Heidelberg, 2007. Springer-Verlag.

[18] P. Gupta and A. B. Kahng. Manufacturing-aware physical design. In *ICCAD '03: Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design*, page 681, Washington, DC, USA, 2003. IEEE Computer Society.

[19] D. E. Holcomb, W. P. Burleson, and K. Fu. Initial sram state as a fingerprint and source of true random numbers for rfid tags. In *In Proceedings of the Conference on RFID Security*, 2007.

[20] C. Kim, K. Roy, S. Hsu, R. Krishnamurthy, and S. Borkar. An on-die cmos leakage current sensor for measuring process variation in sub-90nm generations. pages 221 – 222, may 2005.

[21] M. S. Kirkpatrick and E. Bertino. Software techniques to combat drift in puf-based authentication systems. In *Workshop on Secure Component and System Identification (SECSI 2010)*, page 9, Cologne,DE, 2010.

[22] P. C. Kocher. Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems. In *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '96, pages 104–113, London, UK, UK, 1996. Springer-Verlag.

[23] P. C. Kocher, J. Jaffe, and B. Jun. Differential power analysis. In *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '99, pages 388–397, London, UK, UK, 1999. Springer-Verlag.

[24] A. R. Krishna, S. Narasimhan, X. Wang, and X. Wang. Mecca: a robust low-overhead puf using embedded memory array. In *Proceedings of the 13th international conference on Cryptographic hardware and embedded systems*, CHES'11, pages 407–420, Berlin, Heidelberg, 2011. Springer-Verlag.

[25] 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*, pages 67–70, 2008.

[26] J. Lee, D. Lim, B. Gassend, G. Suh, M. van Dijk, and S. Devadas. A technique to build a secret key in integrated circuits for identification and authentication applications. In *VLSI Circuits, 2004. Digest of Technical Papers. 2004 Symposium on*, pages 176 – 179, june 2004.

[27] D. Lim, J. Lee, B. Gassend, G. Suh, M. van Dijk, and S. Devadas. Extracting secret keys from integrated circuits. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 13(10):1200–1205, 2005.

[28] R. Maes, P. Tuyls, and I. Verbauwhede. A soft decision helper data algorithm for sram pufs. In *Information Theory, 2009. ISIT 2009. IEEE International Symposium on*, pages 2101–2105, 2009.

[29] A. Maiti.

[30] A. Maiti, I. Kim, and P. Schaumont. A robust physical unclonable function with enhanced challenge-response set. *Information Forensics and Security, IEEE Transactions on*, PP(99):1, 2011.

[31] A. Maiti, L. McDougall, and P. Schaumont. The impact of aging on an fpga-based physical unclonable function. In *Field Programmable Logic and Applications (FPL), 2011 International Conference on*, pages 151–156, 2011.

[32] A. Maiti and P. Schaumont. Improving the quality of a physical unclonable function using configurable ring oscillators. In *Field Programmable Logic and Applications, 2009. FPL 2009. International Conference on*, pages 703–707, 2009.

[33] M. Majzoobi, F. Koushanfar, and M. Potkonjak. Testing techniques for hardware security. In *Test Conference, 2008. ITC 2008. IEEE International*, pages 1 –10, oct. 2008.

[34] S. Mukhopadhyay, H. Mahmoodi, and K. Roy. Modeling of failure probability and statistical design of sram array for yield enhancement in nanoscaled cmos. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 24(12):1859–1880, 2005.

[35] G. Qu and C.-E. Yin. Temperature-aware cooperative ring oscillator puf. In *Hardware-Oriented Security and Trust, 2009. HOST '09. IEEE International Workshop on*, pages 36–42, 2009.

[36] U. R ührmair, F. Sehnke, J. S ölter, G. Dror, S. Devadas, and J. u. Schmidhuber. Modeling attacks on physical unclonable functions. In *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, pages 237–249, New York, NY, USA, 2010. ACM.

[37] V. Reddy, A. Krishnan, A. Marshall, J. Rodriguez, S. Natarajan, T. Rost, and S. Krishnan. Impact of negative bias temperature instability on digital circuit reliability. In *Reliability Physics Symposium Proceedings, 2002. 40th Annual*, pages 248–254, 2002.

[38] D. Schroder and J. Babcock. Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing. *Journal of Applied Physics*, 94(1):1–18, 2003.

[39] G. Selimis, M. Konijnenburg, M. Ashouei, J. Huisken, H. De Groot, V. van der Leest, G.-J. Schrijen, M. van Hulst, and P. Tuyls. Evaluation of 90nm 6t-sram as physical unclonable function for secure key generation in wireless sensor nodes. In *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on*, pages 567–570, 2011.

[40] P. Simons, E. van der Sluis, and V. van der Leest. Buskeeper pufs, a promising alternative to d flip-flop pufs. In *Hardware-Oriented Security and Trust (HOST), 2012 IEEE International Symposium on*, pages 7–12, 2012.

[41] Y. Su, J. Holleman, and B. Otis. A 1.6pj/bit 96variations. In *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pages 406–611, 2007.

[42] G. Suh and S. Devadas. Physical unclonable functions for device authentication and secret key generation. In *Design Automation Conference, 2007. DAC '07. 44th ACM/IEEE*, pages 9–14, 2007.

[43] S. Sze and K. K. Ng. *MOSFETs*, pages 293–373. John Wiley & Sons, Inc., 2006.

[44] Y. Tsividis. *Operation and Modeling of the Mos Transistor (The Oxford Series in Electrical and Computer Engineering)*. Oxford University Press, 2004.

[45] T. Tsunomura, A. Nishida, and T. Hiramoto. Analysis of NMOS and PMOS Difference in Variation With Large-Scale DMA-TEG. *IEEE Transactions on Electron Devices*, 56:2073–2080, 2009.

[46] T. Tsunomura, A. Nishida, F. Yano, A. Putra, K. Takeuchi, S. Inaba, S. Kamohara, K. Terada, T. Hiramoto, and T. Mogami. Analyses of 5 #x03c3; vth fluctuation in 65nm-mosfets using takeuchi plot. In *VLSI Technology, 2008 Symposium on*, pages 156–157, 2008.

[47] V. van der Leest, G.-J. Schrijen, H. Handschuh, and P. Tuyls. Hardware intrinsic security from d flip-flops. In *Proceedings of the fifth ACM workshop on Scalable trusted computing*, STC '10, pages 53–62, New York, NY, USA, 2010. ACM.

[48] V. Vivekraja and L. Nazhandali. Circuit-level techniques for reliable physically uncloneable functions. In *Proceedings of the 2009 IEEE International Workshop on Hardware-Oriented Security and Trust*, HST '09, pages 30–35, Washington, DC, USA, 2009. IEEE Computer Society.

[49] N. Weste and D. Harris. *CMOS VLSI Design: A Circuits and Systems Perspective*. Addison-Wesley Publishing Company, USA, 4th edition, 2010.

[50] M.-D. Yu and S. Devadas. Secure and robust error correction for physical unclonable functions. *Design Test of Computers, IEEE*, 27(1):48–65, 2010.

[51] M.-D. M. Yu, D. M'Raihi, R. Sowell, and S. Devadas. Lightweight and secure puf key storage using limits of machine learning. In *Proceedings of the 13th international conference on Cryptographic hardware and embedded systems*, CHES'11, pages 358–373, Berlin, Heidelberg, 2011. Springer-Verlag.