9-12-2014

Novel Transistor Resistance Variation-based Physical Unclonable Functions with On-Chip Voltage-to-Digital Converter Designed for Use in Cryptographic and Authentication Applications

Raj Chakraborty

Raj K. Chakraborty
Candidate

Electrical and Computer Engineering
Department

This dissertation is approved, and it is acceptable in quality and form for publication:

Approved by the Dissertation Committee:

Dr. Jim Plusquellic, Chairperson

Dr. Payman Zarkesh-Ha

Dr. Jedidiah Crandall

Dr. Charles Fleddermann

Novel Transistor Resistance Variation-based Physical Unclonable Functions with On-Chip Voltage-to-Digital Converter Designed for Use in Cryptographic and Authentication Applications

by

Raj K. Chakraborty

B.S., Microelectronics Engineering, Rochester Institute of Technology, 1997
M.S. (distinction), Electrical Engineering, San Jose State University, 2001

DISSERTATION

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctor of Philosophy
Engineering

The University of New Mexico
Albuquerque, New Mexico

July, 2014

Dedication

To my loving parents, Bejoy and Mita Chakraborty, for their relentless sacrifices and unconditional support throughout the years. All this would not have been possible without their constant encouragement and foresightedness. To my wife, Swagata, for her unwavering belief in me and patience through the years. To my lovely daughters, Sarmishta and Raima. You both give meaning to my life and you both are my heart and soul. To my late grand-father, for his guidance and teachings in my early years.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Jim Plusquellic, for his guidance, mentoring, and support throughout this entire endeavor. None of this would have been possible without his ingenuity and creativity. Thank you for having faith in me.

Second, I would like to sincerely thank members of my research group for all the idea brainstorming, useful discussions, and help with data collection.

Third, I would like to thank my committee members for taking the time out from their busy schedules to participate in my dissertation activities and for the useful discussions.

Last but not least, I would like to thank the professors of the Electrical and Computer Engineering department at the University of New Mexico, from whom I learnt a great deal through the classes I took.

Novel Transistor Resistance Variation-based Physical Unclonable Functions with On-Chip Voltage-to-Digital Converter Designed for Use in Cryptographic and Authentication Applications

by

Raj K. Chakraborty

B.S., Microelectronics Engineering, Rochester Institute of Technology, 1997
M.S. (distinction), Electrical Engineering, San Jose State University, 2001
Ph.D., Engineering, University of New Mexico, 2014

ABSTRACT

Security mechanisms such as encryption, authentication, and feature activation depend on the integrity of embedded secret keys. Currently, this keying material is stored as digital bitstrings in non-volatile memory on FPGAs and ASICs. However, secrets stored this way are not secure against a determined adversary, who can use specialized probing attacks to uncover the secret. Furthermore, storing these pre-determined bitstrings suffers from the disadvantage of not being able to generate the key only when needed. Physical Unclonable Functions (PUFs) have emerged as a superior alternative to this.

A PUF is an embedded Integrated Circuit (IC) structure that is designed to leverage random variations in physical parameters of on-chip components as the source of entropy for generating random and unique bitstrings. PUFs also incorporate an on-chip infrastructure for measuring and digitizing these variations in order to produce bitstrings.

Additionally, PUFs are designed to reproduce a bitstring on-demand and therefore eliminate the need for on-chip storage.

In this work, two novel PUFs are presented that leverage the random variations observed in the resistance of transistors. A thorough analysis of the randomness, uniqueness and stability characteristics of the bitstrings generated by these PUFs is presented. All results shown are based on an exhaustive testing of a set of 63 chips designed with numerous copies of the PUFs on each chip and fabricated in a 90 nm nine-metal layer technology. An on-chip voltage-to-digital conversion technique is also presented and tested on the set of 63 chips. Statistical results of the bitstrings generated by the on-chip digitization technique are compared with that of the voltage-derived bitstrings to evaluate the efficacy of the digitization technique. One of the most important quality metrics of the PUF and the on-chip voltage-to-digital converter, the stability, is evaluated through a lengthy temperature-voltage testing over the range of -40^{o}C to +85^{o}C and voltage variations of +/- 10% of the nominal supply voltage. The stability of both the bitstrings and the underlying physical parameters is evaluated for the PUFs using the data collected from the hardware experiments and supported with software simulations conducted on the devices.

Several novel techniques are proposed and successfully tested that address known issues related to instability of PUFs to changing temperature and voltage conditions, thus rendering our PUFs more resilient to these changing conditions faced in practical use.

Lastly, an analysis of the stability to changing temperature and voltage variations of a third PUF that leverages random variations in the resistance of the metal wires in the power and ground grids of a chip is also presented.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 What is a Physical Unclonable Function?

A Physical Unclonable Function (PUF) is an embedded IC structure designed to leverage naturally occurring variations in the physical parameters of on-chip components such as wires, transistors, etc. to produce a random bit string. These variations are unique to each chip and cannot be reproduced or duplicated in its exactness hence, depending on the parameter, can be leveraged to produce large numbers of random bits. Nothing in the manufacturing process of a chip is exact, and therefore, all fabricated physical components, e.g., wires and transistors, on the chip vary from their nominal characteristics. Although it is possible to measure these physical variations directly, it is extremely difficult or impossible to do so without sophisticated processes and equipment. The analog electrical and parametric variations that result, on the other hand, can be measured and processed more easily, and in many cases, this can be done using on-chip instrumentation. Many proposed PUF-based systems are defined in this manner, and are differentiated by the type of electrical variation they leverage. The magnitude and stability of variations in, e.g., transient current, delay, leakage, resistance, capacitance, etc. are dependent on the technology and the environment, and therefore, some PUF systems can better meet certain quality metrics than others.

Chapter 1. Introduction

PUFs are promising components for next generation of integrated circuit (IC) security and continue to gain momentum as an alternative to the current practice of embedding ‘secrets’ using fuses and non-volatile memory on ICs. PUFs can produce repeatedly random bitstrings on the fly using dedicated hardware primitives and processing engines, and therefore eliminate the need for a specialized non-volatile on-chip memory to store them. This feature not only improves their resilience to invasive attacks designed to steal the secret keying material, but it also reduces the cost of manufacturing the IC. The latter is true because, in many cases, PUFs are designed using components that can be fabricated using standard CMOS processing steps, and therefore, the cost of integrating non-standard components, such as non-volatile memories, is eliminated. PUFs generate random but reproducible bitstrings that can be used in security applications such as encryption, authentication, feature activation, metering, etc.

A PUF produces a bitstring by applying a set of “challenges” to specialized circuit primitives and measuring the corresponding “responses”. The challenges are typically ‘digital’ and therefore can be generated on-chip using a pseudo-random number generator such as a linear feedback shift register (LFSR). The challenges are used to configure one or more PUF circuit primitives prior to the application of a stimulus. The stimulus elicits an analog response from the PUF primitives, which is measured and digitized by other components of the PUF circuit. The digitized responses are then compared in a variety of combinations to produce a digital bitstring.

The PUF response is analog in nature, e.g., it can be a voltage drop or the propagation delay of a signal through the PUF primitive. The analog nature of the underlying random variable make the PUF sensitive to environmental variations such as temperature and power supply noise. Several important applications of a PUF require that they produce the same bitstring for a fixed challenge. Therefore, PUF architectures must be both random and resilient to noise sources such as Temperature and Voltage (TV).

Another important characteristic of the PUF as a next generation security mechanism is its potential for generating large numbers of repeatable random bits. This feature offers new opportunities for software processes to strengthen security mechanisms, for example, by allowing frequent re-keying in encrypted communication channels and by allowing a large, changing set of shared keys to be utilized among multiple communicating entities. PUFs are designed to be sensitive to variations in the printed and implanted features of wires and transistors on the IC. Precise control over the fabrication of IC components is becoming more difficult in advanced technology generations, resulting in a wider range of electrical variations among and within the replicated copies of the chip. Signal variations that occur within the IC are the source of entropy for the PUF.

1.2 Quality Metrics of PUF Generated Bitstrings

The ‘quality’ of the bitstring produced can be measured against many statistical metrics, but needs to meet three important criteria: 1) the bit string is unique for each chip, and

thereby able to distinguish each chip in the population, 2) the bit string is random and therefore difficult or impossible to model and predict by an adversary, and 3) the bit string is stable, i.e., it remains constant for a given chip over time, and under varying environmental conditions such as temperature and voltage. A PUF that is able to meet these requirements can be used in applications related to security including chip identification, authentication, as keys for encryption algorithms, for remote activation, and for protecting Intellectual Property (IP).

Several statistical parameters have emerged as important metrics for judging the quality of a PUF. Hamming Distance (HD) is defined as the number of bits that are different when two bit strings are compared. Interchip HD is used to determine the uniqueness of the bitstrings among the population of chips. An average inter-chip HD is defined by computing the HDs across all combinations of bit strings from the chip population. The best result occurs when exactly half of the bits from any two bit strings are different, i.e., when the average HD, expressed as a percentage, is 50%. The intra-chip HD can be used to evaluate stability of the bitstrings, i.e., the ability of each chip to reproduce the same bitstring time-after time, under varying TV conditions. An intra-chip HD is computed using all combinations of bit strings obtained from one chip in the population under repeated sampling at different TV corners. An average intra-chip HD is computed by averaging all of the individual intra-chip HDs. The ideal value in this case is 0%, i.e., each chip is able to reproduce the same bit string. Probability of failure, defined as the ratio of the number of bits that are different to the total number of bits produced

when comparing the bitstrings produced at different TV corners, is also used to evaluate the stability of a bitstring.

Similarly, the NIST statistical test suite can be used to evaluate the randomness of the bitstrings produced by each chip. These standardized battery of statistical tests developed at NIST are applied at a significance level of 0.01 (the default) [1]. In general, the NIST tests look for ‘patterns’ in the bit strings that are not likely to be found at all or above a given frequency in a ‘truly random’ bit string. For example, long or short strings of 0’s and 1’s, or specific patterns repeated in many places in the bit string work against randomness. The output of the NIST statistical evaluation engine is the number of chips that pass the null hypothesis for a given test. The null hypothesis is specified as the condition in which the bitstring-under-test is random. Therefore, a good result is obtained when the number of chips that pass the null hypothesis is large.

1.3 PUF Applications

As mentioned earlier, one of the main drawbacks of PUFs are their sensitivity to environmental conditions. The temperature sensitivity and measures of uniqueness of several popular PUF instantiations are captured in Table I [2]. As can be seen from Table I, the intra-chip HD indicates that some PUF instantiations are very sensitive to environmental conditions and as a result are very noisy. This noise can be random or deterministic. Random errors are caused by circuit noise such as shot noise that are not as predictable and are inherently present even in the absence of any environmental changes.

The deterministic errors are more predictable and are caused by environmental condition changes such as temperature variations, voltage variations, aging, etc. that induce local mismatches of internal components of the device.

Table I. Typical temperature sensitivity and uniqueness of several popular PUF instantiations [2][85]

PUF Type	**Temperature Range (ºC)**	**Intra-chip HD (%)**	**Inter-chip HD (%)**
Ring-oscillator PUF 1	20 to 120	0.48	46.14%
Ring-oscillator PUF 2	25 to 65	1.9	-
Arbiter	20 to 120	9	38%
SRAM PUF	-20 to 80	<12	49.97%
Latch based PUF	0 to 80	5.5	50.55%
Butterfly PUF	-20 to 80	<6	50%
D-flip flop PUF	-40 to 80	10	-
Glitch-based PUF	0 to 80	8	-

For some applications like key generation or authentication, noisy PUF output is unacceptable as the bitstring has to be reproducible with very little to no noise. Therefore, techniques to correct for this noise or error have to be applied for such applications. However, for applications such as identification, additional error correction techniques can be avoided as long as the IDs are unique enough to tolerate the error associated with

environmental variations. Therefore, even if relatively large errors occur, the chip can still be identified correctly.

1.3.1 Identification and Authentication

Identification is the simplest application of a PUF that can be implemented without additional error correction techniques. The process of identification is widely used in anti-counterfeiting technologies. The process of identification consists of two phases: enrollment and identification. During the enrollment phase, a PUF is challenged with different challenge-sets and the challenge-response pairs (CRP) are stored in a database. The identification phase allows identifying an entity that exhibits a challenge-response pair contained in that database. The result of the whole identification process is a chip ID which is assigned to the CRP in the database. The decision if the observed response matches the entity in the database is usually made by HD calculations. In a PUF without error correction techniques, the responses for the same set of challenges usually differ slightly due to noise, and this is captured by the intra-chip HD.

The acceptable noise level for a positive identification depends on the intersection of the intra-chip HD and the inter-chip HD distributions. An example of this is illustrated in Fig. 1 [2]. For a particular type of PUF, the inter- and intra-distance characteristics are often summarized by providing distributions showing the occurrence of both distances, observed over a number of different challenges and a number of different chips. In many cases, both distributions can be approximated by a gaussian distribution and are

summarized by providing their means and their standard deviations. The intra-chip HD distribution represents the average reproducibility of a measured response with respect to an earlier observation of the same response from a PUF while the interchip HD distribution represents the distinguishability and the uniqueness of this response amongst responses from other PUFs. A successful identification depends on the separation between the intra-distance and inter-distance distributions. If both distributions do not overlap, an errorless identification can be made. In the case of overlapping distributions, errors in identification are possible. Either the wrong chip is erroneously identified which is termed as False Acceptance Rate (FAR) or the correct chip is erroneously rejected which is termed as False Rejection Rate (FRR). For cases of overlapping distributions, a trade-off between FAR and FRR has to be made and the sum of these two errors has to be minimized.

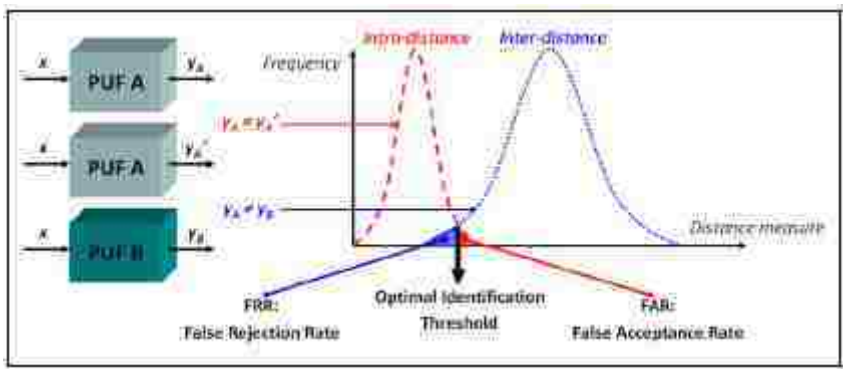


Fig. 1. Illustration of FRR and FAR [2]

PUFs can also be used to authenticate ICs with minimal hardware cost using a challenge-response protocol. In this process, a secure database stores a set of CRPs from

each PUF instance prior to the use of the IC. When the authenticity of the IC has to be queried, a set of CRPs are chosen randomly from this database and applied to the PUF circuit. The obtained response is compared with the responses stored in the database to authenticate the IC. It is important that challenges are never reused to prevent man-in-the middle attacks [3]. Hence it is extremely useful to have a PUF that can support large number of CRPs. This feature has been demonstrated by implementing PUF based RFID tags in 0.18um technology [8]. Results have shown that with a 128bit response, the FAR and FRR can be reduced to a few parts per billion. This can be improved further by using wider set of response bits. Hence PUFs are naturally suited for authentication and this has been explored in several lightweight protocols [9][10]. Mutual authentication and ownership transfer protocols to identify both RFID readers and tags by utilizing PUFs and LFSRs have also been proposed [4]. Existing hash functions require 8000 to 10,000 gates as compared to 784 gates used in this approach. It is also mentioned that an RFID tag can afford a maximum of 2000 gates for security features. The use of PUFs has also been proposed for IC activation and prevention of piracy in integrated circuits [10]. Roy et al. have proposed the concept of Ending Piracy in Integrated Circuits (EPIC) which involves embedding a combinational locking mechanism on the IC [11]. A random IC key pair is generated during initial power-up and this is utilized to create a common key between the user and the IP provider. The IP provider transmits this common key to the user to unlock the IC. In resource constrained platforms, the use of PUFs has been proposed to generate the unique signature necessary for this application.

Bolotnyy et al. [12] have proposed the use of PUFs to implement privacy preserving tag identification and secure message authentication code. When a reader interrogates a tag, the tag responds with its ID and updates its identifier to the PUFs response to the challenge ID. The backend database will need to store the challenge-response pairs. In this way, a PUF based MAC protocol can use the PUFs response to sign a message.

1.3.2 Key generation and Cryptography

Cryptographic primitives such as encryption and message authentication need the presence of a secret key. The use of PUFs for secret key generation was first proposed in [3]. A PUF can be used in secret key generation and the main requirement for such applications is a stable and reproducible PUF output. In order to produce exactly the same PUF response repeatedly over time, some error correction techniques have to be applied. The process of secret key generation consists of two phases. The first is the generation phase where the PUF is queried and the secret key is generated by an algorithm with the aid of some helper data stored off-chip in a database. The second phase is the regeneration phase where the PUF is queried again and the secret key is regenerated by the algorithm with the helper data from the database. Thus, the algorithm extracts the same secret key as in the generation phase. The helper data and algorithm are stored in an off-chip database and generally reveal nothing about the secret key.

Another application is the generation of secrets for cryptography. The advantage of a PUF is that these secrets do not have to be stored anywhere on the hardware, since they are generated dynamically at device reset. This is especially interesting for embedded devices. An example of a cryptographic application involves a mobile phone whose firmware must be decrypted on each startup. The cryptographic key must somehow be stored securely. Solutions using nonvolatile memory or volatile memory with a battery are vulnerable to physical attacks or side channel attacks. PUFs can reduce these vulnerabilities, since physically disassembling such a circuit will destroy its delay characteristics and therefore change its output.

As far as secure communication is concerned, there are several RFID (Radio Frequency Identification) authentication schemes proposed that intend to strongly reduce many of the vulnerabilities in today's RFID systems. These designs must be extremely efficient both in energy and complexity since a typical RFID card may only offer a few thousand logic gates. A proposed mutual-authentication scheme for RFID using PUFs appears in [4]. The work in [5] uses a PUF's output to encrypt the challenge-response pairs exchanged during RFID communication. In [6] SRAM PUFs are used to implement a PKI system to encrypt the transmission of a bitstream to an FPGA. FPGA bitstream encryption is also performed in [7] using an Anderson PUF.

Another example of the application of secret key generation by PUFs in cryptography is the protection of device firmware. Transmission of the firmware to the device utilizes a public/private key-pair. The server that maintains the firmware will encrypt the firmware and then sign it using its own private key. This encrypted data is

then sent to the mobile device along with the server's public key. In this case, the server is not concerned with someone being able to capture this transmission and decrypt the firmware, but is more concerned about proving to the devices that the packet is valid. To support this approach, a mathematical operation known as a "hash" is performed on the public key, which results in a non-reproducible, and often smaller value. This hash can be stored in non-volatile storage on the mobile device and reveals nothing about the secret key. When the device receives the firmware update package from the server, it performs the same hash on the received server public key. If the result of the hash matches what is in memory, the device knows that the key is valid and has not been altered. It can then use the key to verify the signature of the firmware package and decrypt the remaining data. Once the data has been decrypted, it can then be re-encrypted using the device-unique PUF generated secret key and stored on the device.

1.3.3 Controlled PUFs and Secure Environments

The notion of controlled PUFs (CPUFs) was introduced by Gassend et al. [19]. Controlled PUFs are entities in which the PUF can be accessed only by an algorithm tied to the physical device. The use of CPUFs to generate a secret to be shared between a remote user and a physical device is mentioned in this work. In addition, introduction of a user, renewal of CRPs, and anonymity preserving protocols have been discussed. Applications such as certified execution and software licensing using CPUFs have also been discussed. Certified execution involves producing a certificate verifying the

authenticity of the IC. In distributed computing scenarios, this allows a remote user to know that his program ran on a certified chip without being tampered. Similarly, use of PUFs for software licensing will allow only authentic software to be run on a processor.

PUF circuits are an ideal choice for security applications and they form a part of the security solutions offered in industry by Verayo and Intrinsic ID [13][14] amongst others. Verayo offers the PUF as an IP to be licensed for RFID, ASIC and FPGA applications. Intrinsic ID provides secure key storage to protect semiconductor products from cloning and reverse engineering.

The idea of the PUF-based secure environment based on hardware generated keys was introduced in [15]. In particular, the idea of this scenario is to generate a cryptographic key which depends on the underlying hardware, and thus implicitly identifies the device. Subsequently, the key is used to unlock encrypted software, which is installed on the device. The idea is to decrypt the bootloader, which is executed first during device start-up in the domain of embedded devices. After the bootloader has been decrypted using the key derived from the PUF response, it subsequently unlocks the kernel, which in turn decrypts user space applications. Since every layer relies on the preceding layer to be decrypted, it is possible to establish a chain-of-trust with the hardware constituting the anchor-of-trust. A full implementation of the scheme could provide an alternative to current device identification approaches. The Mobile Trusted Module (MTM) approach [16], for example, relies on storing several keys and certificates in dedicated chips or in software. The former option requires additional hardware, which induces extra costs from the manufacturer's point of view. Alternatively, software MTMs

cannot provide strong hardware-based anchors of trust. Traditional approaches are potentially prone to side-channel attacks to extract cryptographic material. In contrast, using intrinsic PUFs would bind a software instance to the hardware and not to a permanently stored cryptographic key.

1.3.4 Random Number Generator

With some modification, a PUF design can also be turned into a true and cryptographically secure random number generator. True random number generators have been created by exploiting D-Flip Flop metastability [17], Ring Oscillators [18], and SRAM PUFs [20]. In a similar way Deterministic Random Bit Generators (DRBG) can be created, such as in [20]. DRBGs employ a deterministic algorithm to create pseudo-random numbers, but seed it with the random signature generated by a PUF. As long as the seed remains secret, the numbers that are generated are not predictable. This system can create large numbers of random numbers very quickly.

1.4 Research Contributions

1.4.1 Proposed PUFs and On-Chip Voltage-to-Digital Converter

As part of this research, two novel PUF primitives that leverage resistance variations that occur in transistors are presented. Specifically, the resistance variations in transistors that make up the Transmission Gates (TGs) and Inverters in these two PUF primitives are leveraged. Therefore, these two PUFs are named the Transmission Gate PUF (TG-PUF) and the Inverter PUF (I-PUF) to refer to their respective primitives.

Hardware experiments are carried out on these PUFs built into a set of 63 chips manufactured with a 90 nm nine-metal layer process. Each of these chips had numerous copies of these PUF primitives designed in them, therefore allowing for a very statistically significant sample size. Furthermore, all 63 of these chips were put through rigorous and lengthy TV testing using a controlled temperature chamber allowing for data collection across industrial rated TV ranges. Nine TV corners, using all combinations of the temperatures -40ºC, 25ºC, and 85ºC and voltage variations of +/- 10% of the nominal supply voltage, were tested. This work is unique in the fact that a full-blown 9 TV corner testing was conducted on a sample size of this extent.

An embedded structure called a Voltage-to-Digital Converter (VDC) that was also designed into each of the chips for the purposes of digitizing the analog output signals from the PUF was also evaluated under these varying environmental conditions.

The analysis of stability to TV variations of a PUF called the Power Grid PUF (PG-PUF) [21][22] that is based on resistance variations which occur in the metal wires of the chip's power and ground grids is also presented. A significant benefit of using metal structures is that "noise-related" variations, such as those introduced by TV variations, result in linear changes to the measured voltages. This linear scaling characteristic allows the relative magnitude of two voltages to remain fairly consistent across changes in TV, which, in turn, improves the stability of the PUF to bit-flips, when compared, for example to PUFs which leverage transistor-based variations. The analysis presented is from experimental data collected on the same 63 chips fabricated with a 90 nm nine-metal layer process at 9 TV corners, i.e., over all combinations of 3 temperatures -40ºC, 25ºC, and 85ºC and voltage variations of +/- 10% of the nominal supply voltage.

All analyses related to the PUFs stability to TV variations were scrutinized down to the physical parameter level and not just the bit level. This work is unique and lacks precedence in shedding light on evaluating the extent to which physical parameters affect the stability characteristics of these novel PUFs. The hardware experimental data was also verified with simulation, and the hardware and simulations results were compared and contrasted.

The bitstrings generated from our PUFs are categorized into two types. First are those generated directly by comparing the digitized voltages from the PUF with each other and second are those that are generated by converting the digitized voltages into thermometer codes with the aid of the on-chip VDC and then comparing those codes with each other to generate the bitstring. The results presented include statistical analyses of

both the voltage-comparison generated bitstrings and the VDC-generated bitstrings. This allows for evaluation of the pros and cons of each method of generating the bitstring and more importantly, allows the evaluation of the penalties involved with the digitization process. Other work in the PUF research domain usually evaluates only the digitized bitstrings, so this research is unique in assessing both voltage-derived bitstrings and digitized bitstrings.

A study of the area and power consumption characteristics of the novel PUFs was also completed and the results compared against the characteristics of predominant competing PUF designs.

1.4.2 Bit Flips and Novel Techniques to Avoid Them

In general terms, bit flips are defined as the specific bits that change or flip from “1” to “0” or “0” to “1” when comparing two bitstrings generated at two different instances. Since environmental conditions can be different at any two instances that the bitstring is generated, the number of bit flips is an indication of the stability of the bitstring (and the PUF used to generate it) to changing environmental conditions.

Needless to say, the fewer the bit flips the better. However, there will always be a certain level of bit flips in any PUF and the key is to devise robust techniques to either correct for these bit flips or avoid them.

Two noise-resilient bit-flip avoidance schemes that are designed to increase the probability that the bitstring can be reproduced under varying environmental conditions

are demonstrated. These novel techniques are developed as an alternative to popular error correction [23] and helper data schemes [24] which tend to suffer from additional area overhead/cost and the increased chance of attacks and data compromise.

The first technique derives a threshold from a chip's digitized voltage drop distribution profile that is used to decide whether a given comparison generates a strong bit or a weak bit, where strong bits are defined as those that will not flip when the bitstring is regenerated and weak bits as those that are more susceptible to flipping. A second Triple Module Redundancy (TMR-based) scheme is proposed for fixed length bitstrings that further improves bit-flip resilience. Although these techniques discard a significant fraction of bits, they provide several significant advantages. The public (helper) data associated with these methods reveals nothing about the secret bitstrings that they encode. Second, for applications where the PUF responses are made public, the difficulty of model building is significantly increased (assuming the public data is obfuscated) because bitstrings are constructed using only a subset of all possible voltage pairings. These techniques are tested and demonstrated with data obtained from the 63 chips fabricated in a 90 nm technology and provide a significant improvement to inter-chip Hamming Distance and the results obtained from NIST statistical tests [1].

Lastly, a compression technique was presented that would help reduce the size of the public data associated with the thresholding and TMR techniques.

Chapter 2

Background

Research on PUFs and process variation has been gaining increasing interest since the concept of the PUF was formally introduced by Pappu, et. al. in [25] in 2001. In the simplest sense, a PUF is a device whose transfer function exploits physical phenomena in a way that cannot be replicated, even if the full design is known. The PUF designs that have been proposed over the years have been diverse. The introduction of the PUF as a mechanism to generate random bitstrings began in [25] and [19], although their use as chip identifiers began a couple years earlier [26]. Since their introduction, there have been many proposed architectures that are promising for PUF implementations, including those that leverage variations in transistor threshold voltages [26], in speckle patterns [25], in delay chains and ROs [19][25][27-31 + many others], in thin-film transistors [32], in SRAMs [6][33], in leakage current [34], in metal resistance [21][35], in optics and phase change [36], in sensors [37], in switching variations [38], in sub-threshold design [39], in ROMs [40], in buskeepers [41], in microprocessors [42], using lithography effects [43], and aging [44].

At the behavioral level, a PUF is often thought of as a hardware version of a cryptographic hash function. It is sometimes also referred to as a physical one-way hash function when implemented in a challenge-response framework. PUFs reduce the ability of attackers to circumvent security mechanisms, as these mechanisms are implemented in tamper-resistant hardware rather than at the software level. This property of tamper

evidence has already been demonstrated for optical PUFs [25] and coating PUFs [45]. Furthermore, the devices are conceptually unclonable in the sense that, although they may be physically copied, this provides no advantage to an attacker, because each copy will behave differently. PUF designs exist that consume very little power, meaning a high degree of security can be applied to embedded applications with extremely limited resources, such as RFID cards.

The physical phenomena that underlie a PUF should be computationally difficult to model. While sophisticated models for modeling transistor resistance in semiconductor devices exist, much of the process variation inherent in any manufacturing process can only be modeled as a statistical distribution. These variations exist within a die, between dies, between wafers, and between lots or production batches. These variations appear in the channel doping, channel width/length, and discrete transistor features, as well as the thickness of oxide layers.

This variation exists for every property of a silicon device, any of which can have an impact on the PUF's output. It is well-known that process variation is becoming harder and harder to control as feature size shrinks. Moore’s law has driven CMOS scaling technology over the years leading to increased complexities in designs. Deep sub-wavelength lithography used in lower technology nodes brings greater challenges in the manufacturing process. The amount of process variation seen follows an increasing trend with technology and is becoming increasingly significant. As a result, designs aiming for high performance will find it exceedingly difficult to meet the requirements in presence of these variations. However, an increase in process variation benefits the identification capability that can be

achieved by PUFs. PUF uniqueness is directly related to the amount of inter-chip variation seen as this is what dictates the entropy in the system and with technology scaling, we expect PUF uniqueness to increase. However, an increase in process variation will impact the reliability across different environmental conditions [46] thus undesirably increasing the intra-chip HD.

It has been shown that, at least between 90nm and 45nm processes, not only is variation increasing but it is also becoming less systematic and more random, or stochastic [47]. In [48] ring oscillators are used on a 90nm Field-Programmable Gate Array (FPGA) to estimate the impact of process variation on delay variation. In the study, the amount of variation is projected out to future process nodes. The delay through a lookup table (LUT) was measured to have a mean variation (3σ) of +/-3.5%. The authors projected that for 65nm this will increase to 4.5%, for 45nm 5.5% and 22nm 7.5%. The estimation for 45nm aligns well with the empirical study performed in [47] in 2008, suggesting the projection may be quite accurate.

2.1 PUF Classifications

PUFs can be classified into Strong and Weak PUFs based on the number of challenge response pairs supported which subsequently determines the applications in which they are used [49].

2.1.1 Strong PUFs

Strong PUFs support a large number of CRPs and a complete measurement of all CRPs within a feasible time frame is impossible. Further, it should be difficult for an attacker to predict the response of the PUF for a random selected challenge, even with the prior knowledge of a limited number of CRPs. This implies that the PUF should not be susceptible to modeling attacks. Hence it is tough to mimic the behavior of a strong PUF and this class of PUFs is ideally suited for IC identification and secret key generation. Examples of strong silicon PUF constructions include Arbiter PUFs, feed forward arbiter PUFs, XOR arbiter PUFs and lightweight secure PUFs.

2.1.2 Weak PUFs

Weak PUFs support a limited number of CRPs, sometimes just a single challenge. This prevents their use in IC authentication applications as they will be susceptible to replay attacks. Responses derived from weak PUFs are used to generate a secret key necessary for embedded cryptosystems. Weak PUFs offer a better mechanism to generate secret keys as opposed to storing them in non-volatile memory. The characteristics of a weak PUF will be harder to read out using invasive techniques compared to digital storage in memory. However the secret keys are still susceptible to side channel attacks just as in any physical cryptosystem. Typical examples of weak PUFs are SRAM PUFs and

butterfly PUFs. The concept of a Physically Obfuscated Key (POK) is similar to the idea of a weak PUF where the responses are not given out and are used to generate a secret key internally.

2.2 PUF Designs

A variety of PUF designs have appeared over the past decade. In fact, in [50] it is noted that a new PUF design has appeared roughly each year since 2000.

2.2.1 SRAM PUF

An SRAM PUF is a kind of memory-based and bistable PUF and is depicted in Fig. 2 using the 6 Transistor (6T) SRAM cell. Memory-based PUFs exploit the unpredictability of the startup value of volatile memory cells, which is caused by slight asymmetries in the cell's internal routing and transistor characteristics. This PUF strongly relies on randomness in transistor drive strength, i.e. the strongest inverter decides the startup preference of the cell. The startup value is mainly determined by the relative strength of the Threshold voltage (V_{th}).

SRAM PUFs are quite appealing due to the fact that they rely on commodity SRAM cells. In fact, after the PUF signature is extracted, it is possible to use the same cells as regular non-volatile memory. As an example, SRAM PUFs have even been

evaluated on a commodity microcontroller [51]. In that work, a set of criteria and metrics are proposed to determine whether a given SRAM can function as a PUF.

In their raw, uncorrected state, this type of PUF suffers from a relatively high error rate and thus generally requires complex ECC circuitry and algorithms. Instability occurs when the internal cell layout is too symmetrical; it becomes susceptible to environmental noise, temperature changes, and power supply transients. One proposed approach to combat unstable bits is to place more PUFs than needed, and add ADC circuitry to automatically select the most stable ones [52]. Unfortunately, this approach is not practical for FPGA-based studies since there is generally no flexible way to measure the analog aspect of an internal signal. Another technique applies helper data algorithms to normalize the output [6].

Lastly, SRAM cells are generally limited to 1 bit per cell which leads to a limited CRP count compared to competing designs.

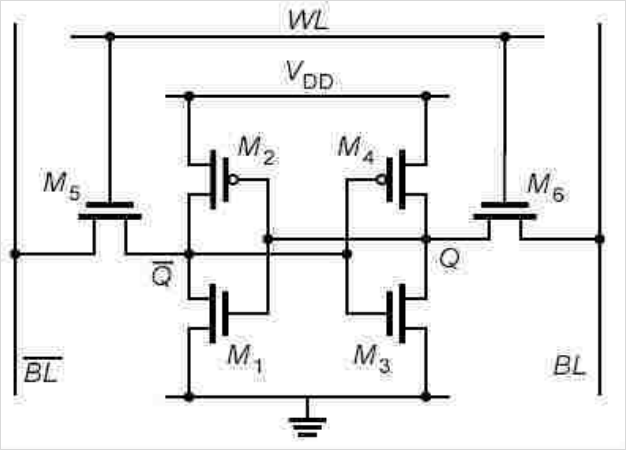


Fig. 2. 6T SRAM Cell PUF

2.2.2 Butterfly PUF

Conceptually, the Butterfly PUF is somewhat similar to the SRAM PUF, in that they both are memory and bistable cells whose startup value is hard to predict. However, it so happens that FPGA SRAM cells are all reset to a known state upon device reset. Therefore the Butterfly PUF was developed in [53] as a way to enable the study of memory-type PUFs on an FPGA. It exploits the cross-coupled D Flip-Flop design, shown in Fig. 3.

Initially the "excite" signal is raised high for a few clocks. Since the preset and clear pins on the D Flip-Flops are asserted, and due to the cross-coupling of the outputs, the circuit is held in an indeterminate, unstable state. When "excite" is released, the circuit output will resolve itself to a stable state of either '1' or '0' based on the delay mismatch between the interconnects. In the ideal case, in which the routes are totally symmetrical, the outcome is determined by the effect the process variation has on the delay.

The advantages of this design are that it uses only D- Flip Flops which are ubiquitous in FPGAs as well as in general design processes.

The disadvantage of this design is that it requires extra care to route due to the constraints of FPGA routing, and that the outputs of the latches can oscillate imposing precise timing requirements on the excite signal for reproducible keys. Also, attaining a metastable point prior to key generation is difficult due to the finite delays of latches and interconnects.

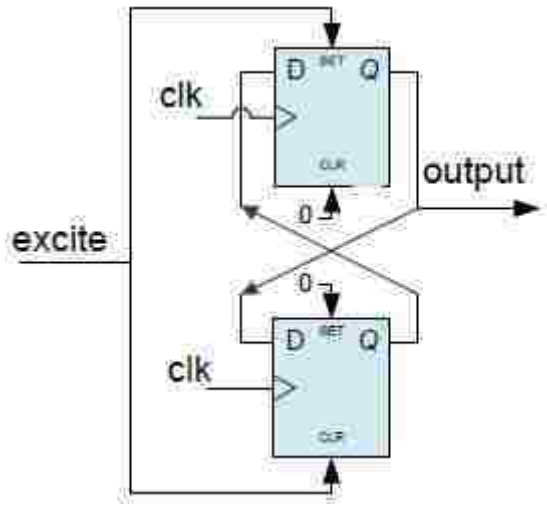


Fig. 3. Butterfly PUF

2.2.3 Ring Oscillator PUF

Generally PUFs based around design symmetry have been deemed less suited for implementation on FPGAs due to the limitations of routing [54]. This is one of the reasons for the popularity of RO-based designs on FPGAs, since absolute symmetry is not necessary to create an oscillator, and the error associated with making a single measurement is amortized across many oscillator cycles. The ring oscillator (RO) PUF, depicted in Fig. 4, is one of the earliest and mature classes of delay-based silicon PUFs, first introduced in [55][56]. A RO is simply a loop of inverters having an odd number of stages. The circuit will spontaneously begin to oscillate with a frequency that can be determined from the delay of each inverter stage.

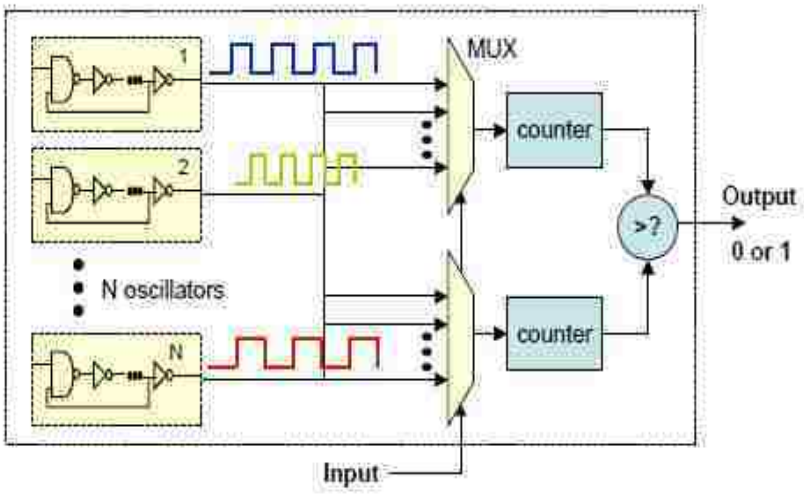


Fig. 4. RO PUF [3]

The RO PUF relies on the fact that any two rings will not oscillate at the exact same frequency, even if they are laid out exactly the same. This is due to process variation which impacts the delay of the signal propagating around the ring. The RO PUF shown in Fig. 4 affixes a counter to each RO and compares the counts after a period of time, in pair-wise fashion [3]. This "differential" measurement has been shown to give better results than a basic RO design.

In [57] is performed the largest-scale analysis of RO behavior that is known to date, using 90nm FPGAs as test platforms. The study confirmed that the RO PUFs generated signatures were unique among different chips, and quite consistent within a given chip.

It is clear from the workings of the RO PUF that it is very layout dependent and there is a high area cost and higher power consumption of this design. Since the

measurements are carried out over a relatively longer period of time, the RO PUF is fairly susceptible to environmental variations.

2.2.4 Arbiter PUF

The arbiter PUF, depicted in Fig. 5, is another well-studied delay-based PUF design, published in 2004 [58]. In the general sense, an arbiter PUF sets up a set of closely-matched race tracks with an arbiter at the end to determine which signal reached the end first. Typically the arbiter is a D Flip-Flop with one signal attached to the clock pin and another attached to the data pin.

Although shown as multiplexers, the adjustable delay portion of the circuit is implemented in different ways. In [59], LUTs are used to create extremely precise programmable delay lines.

A rigorous large-scale analysis of this kind of PUF is performed by [60]. In that work, it is demonstrated that is quite feasible to make a fully-functional arbiter PUF on an FPGA, despite the routing constraints. Interestingly, these results fall contrary to the results of [54] which used timing tools to conclude that FPGA routes could not be configured which are matched closely enough. This discrepancy demonstrates the challenge of measuring process variation and the importance of empirical study.

While arbiter PUFs have been shown to be quite good in terms of adhering to PUF properties, it has been shown that the basic form is vulnerable to model-building attacks as delay is additive in nature [61]. Using machine learning, after observing a

sufficient number of challenge-response pairs, it was possible to guess the outcome the PUF with a 0.6% error rate.

Subsequent designs add additional complexity in order for the challenge to control the delays in a non-linear way. An early attempt to introduce non-linearity is the feed forward arbiter PUF [61]. Since then, there have been several rounds of attack proposals followed by design modifications.

The arbiter PUF is also more layout dependent and susceptible to environmental variations compared to other PUF designs.

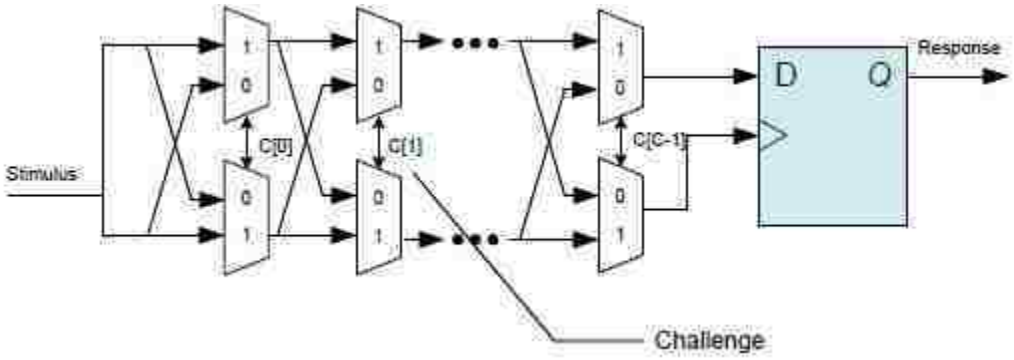


Fig. 5. Arbiter PUF

2.2.5 Power Grid PUF

The Power Grid PUF (PG-PUF) was introduced in 2012 and leverages the variations in resistance of the metal lines in the power and ground grids of a chip to implement a PUF

[21]. The voltage drops across the individual metal layers are digitized and then compared randomly with each other to generate the bitstring.

Since the resistance of metal wires varies linearly with temperature, the PG-PUF can easily be designed to be resistant to aging effects such as electromigration. The resistance can be measured using a simple DC process, which can improve the signal-to-noise ratio significantly over PUFs that leverage AC characteristics such as delay.

The PG-PUF is relatively easy to implement as the metal components are ubiquitous on a chip, with the power grid consuming a large fraction of the metal resources, e.g., 15-25% is typical in most commercial power grid designs.

The power grid is a stacked structure, offering a 3rd dimension in which to leverage entropy in a PG-PUF. Also, the interconnected structure of the wires in the power grid complicates the interaction among variations in resistance that occur, thereby increasing the complexity of model building attacks.

A more completion description of the PG-PUF operation is provided in later sections of this document.

2.2.6 Hardware-Embedded Delay PUF

The Hardware-Embedded Delay PUF (HELP), depicted in Fig. 6, is a delay-based PUF introduced in 2013 [62] and is designed to leverage the natural variations that occur in the

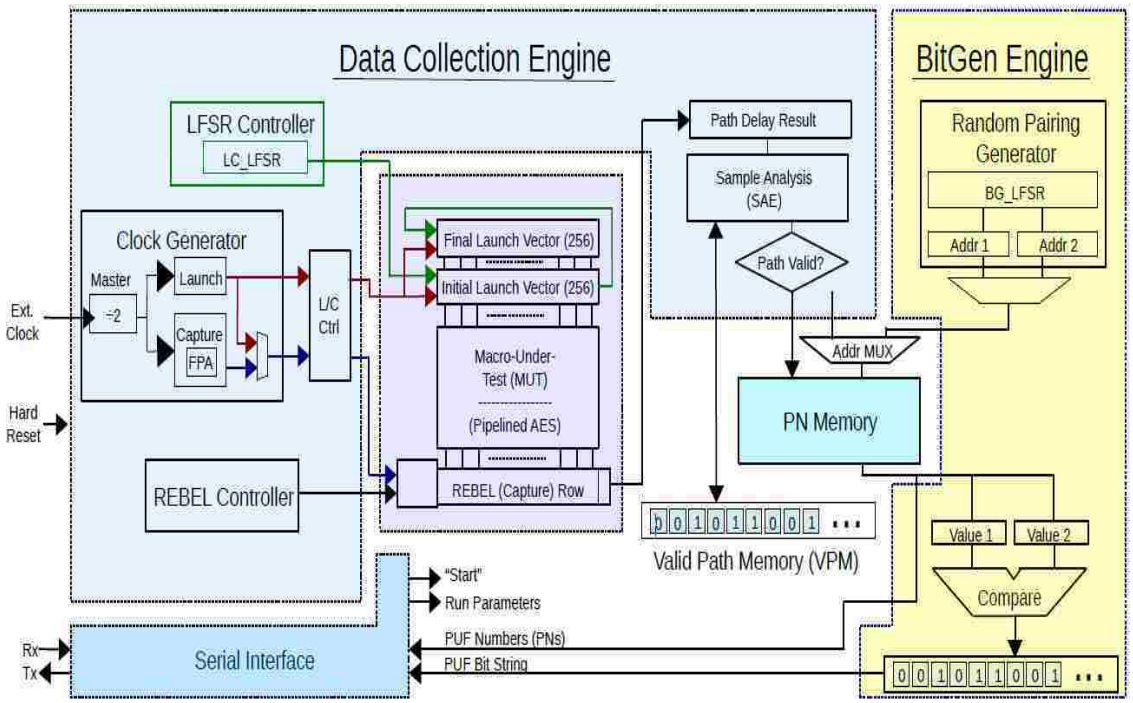


Fig. 6. Top-Level HELP system diagram [62]

path delays of a core macro on a chip to create a unique, stable, and random bitstring of virtually any length.

HELP has demonstrated the capability of comparing paths of widely differing lengths and eliminating the need for specially designed, layout-dependent delay elements that impose a high area cost while providing a relatively small amount of entropy. The design is supposed to be minimally invasive with low area and performance impact.

HELP also exhibits a large number of paths typically found in logic macros such as the Advanced Encryption Standard (AES). This large source of entropy allows HELP to generate large bitstrings, despite being extremely conservative in the paths selected for bit generation.

The challenge component for HELP consists of a randomly selected, two-vector test sequence applied to the inputs of the macro-under-test (MUT), which introduces a set of transitions that propagate through the core logic of the MUT and emerge on its outputs. The responses are the measured path delays on each of the outputs, and are expressed as 8-bit numbers that correspond to path delay. A single MUT output is isolated and measured individually. A bitstring is generated by comparing pairs of these path delays.

2.3. Attacks

While PUFs are reliable and secure because of their intrinsic unique properties obtained due to manufacturing process variations, there are vulnerabilities to some attacks.

A key characteristic of PUFs is the entropy of its responses. The entropy quantifies the number of independent and random IDs that can be generated by the same device architecture. This is proportional to the amount of variation in the physical parameter being leveraged to generate the ID.

Successful product counterfeiting involves the production of a clone. By definition, the cloned device needs to have a PUF with the exact same intrinsic properties as the original one. The probability of success in cloning a PUF depends on the PUF's entropy, thus it is very important to design a PUF with large entropy. The larger the entropy of the original PUF, the more instances are needed in order to successfully create an identical cloned PUF. If the entropy of the PUF is b bits, then it is theoretically possible to obtain 2^b number of unique identifiers from it. Assuming all these identifiers

are uniformly distributed, the probability of occurrence of each ID is equal. But because of the birthday paradox, the population of all possible identifiers is $2^{b/2}$. Due to the large number of instances needed for a successful cloning, such clone attacks makes sense only if the resulting profits are greater than the incurred expenses.

2.3.1 Modeling attacks

Modeling attacks are well described in [63] by Ruhrmair et al. They are based on machine learning algorithms when some CRPs are known and are outlined for Arbiter and Ring-Oscillator PUFs [3]. In these attacks, the adversary collects many CRPs and uses them to derive the runtime delays occurring in the subcomponents of the electrical circuit. Once they are known, simple simulation and prediction of the PUF becomes possible, breaking its security.

High modeling accuracies can be obtained through machine learning techniques like support vector machines and artificial neural networks [3]. Given a limited set of training CRPs, algorithms automatically learn the input-output behavior by trying to generalize the underlying interactions. The more linear a system, the easier to learn its behavior.

In the paper proposing arbiter PUFs as a security primitive, machine learning was already identified as a threat [64]. The authors reported a modeling accuracy of 97% for their 64-stage 0.18µm CMOS implementation.

2.3.2 Side-channel attacks

The result of reverse engineering the internal structure of a PUF by an attacker is the alteration of the CRP characteristics of the PUF. However, the attacker is able to measure external characteristics of the PUF circuit such as electromagnetic radiation, time various computations, power consumption, etc. Attacks formulated based on observation of external characteristics of the circuit are termed side-channel attacks. Previously published work outlines the susceptibility of the PUFs to side-channel attacks [65][66]. For example, the PUF output can be learnt and predicted by investigating the power leakage occurring in the error correction phase [65]. However, side channel attacks are harder to implement due to the resources required for external observation and correlation.

2.3.3 Invasive attacks

Invasive attacks involve the depackaging of the chip in order to get direct access to its inner components and enable the reading out of the states of register, latches, etc. There is a high probability that the removal of the chip layers causes the destroying of the unique chip fingerprint [67] therefore invasive techniques are seldom successful. Furthermore, invasive attacks generally require capital intensive failure analysis equipment. However, [68] proposes successful semi-invasive attacks based on EM signals.

Circuits containing secure data are vulnerable to invasive attacks if they use sequential logic or store secret unencrypted data in SRAM. It has been proposed that to prevent invasive attacks, the PUF response must be processed and stored in a serial manner, i.e. the whole circuit must be serialized [69]. Serialized PUFs cannot generate more than one response at a given point in time limiting the response exposure to external characterization. Thus, only a subset of the full hardware PUF response is ever vulnerable in this implementation. An arbiter PUF is a good example of a serialized PUF while the RO PUF is not serialized since the individual oscillators run simultaneously meaning that more than a single PUF response is present on the device at any given point in time. Making SRAM PUFs more resilient to invasive attacks is also possible by implementing an asynchronous reset for the SRAM [69].

Chapter 3

Design and Experiment Setup

3.1 Test Chip Architecture

Fig. 7 illustrates the block diagram of the 90 nm test chip architecture used for all the experimental data collection in this research. The chip pad-frame consists of 56 I/Os, and surrounds a chip area of approx. 1.5 mm x 1.5 mm. Four pads labeled PS_1, PS_2, NS_1 and NS_2 refer to voltage sense connections; the 'P' version for sensing voltages near V_{DD} and the 'N' version for voltages near GND. These four terminals wire onto the chip and connect to 85 copies of a Stimulus/Measure circuit (SMC). The SMCs are distributed across the entire chip (see small rectangles) as two arrays, a 7x7 outer array and a 6x6 inner array. Although not shown, a controlling scan chain connects serially to each of the SMCs.

The distance between the SMCs is 250μm and noteworthy from Fig. 7 is the fact that the length of the sense wires from the SMC to the voltage sense pads differ from SMC to SMC.

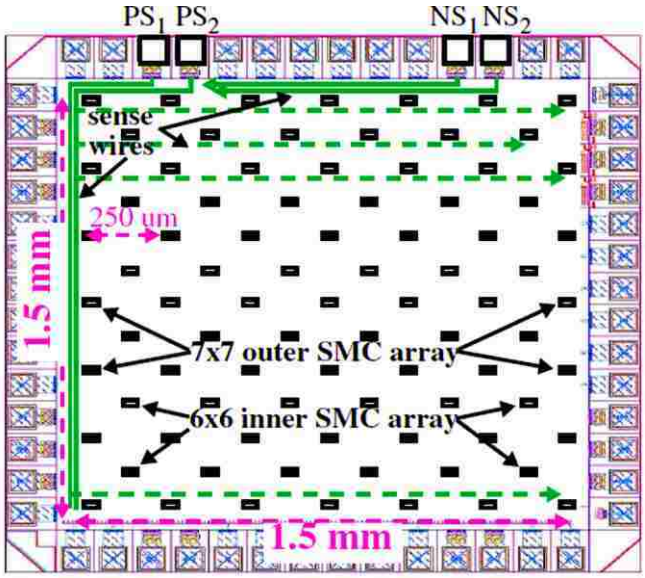


Fig. 7. Block diagram of 90nm test chip

3.2 Transmission Gate PUF (TG-PUF)

The TG-PUF was introduced in 2012 [21] and analyzed in detail in [70]. It leverages the resistance variations in transistors, specifically of those that make up the transmission gates of the TG-PUF primitive.

The schematic diagram of the SMC is shown in Fig. 8. A set of 20 ‘pseudo’ pass gates (hereafter referred to as transmissions gates or TGs) serve as both the PUF primitives and voltage sensing elements. Eight of the TGs, labeled 1 through 8, connect to the first 8 (of the 9) metal layers that define the V_{DD} grid, as shown on the left side of Fig. 8, while the other eight connect to the GND grid. Two additional TGs, labeled as 9

and 10, connect to the drains of the 1-8 TGs. Separate scan FFs control their connection to the chip-wide wires that route to the P/NS_x pins shown in Fig. 8.

For the TG-PUF, the SMC is configured so that the PS_1 and NS_1 sense wires are connected off-chip to GND and V_{DD}, respectively, to create the stimulus condition described as follows. PS_2 and NS_2 are routed to off-chip Agilent 34401A voltmeters (VMs). The pair of shorting transistors in Fig. 8 is always off during the TG-PUF experiments so as to allow the current path to consist of the TGs and not the shorting transistors.

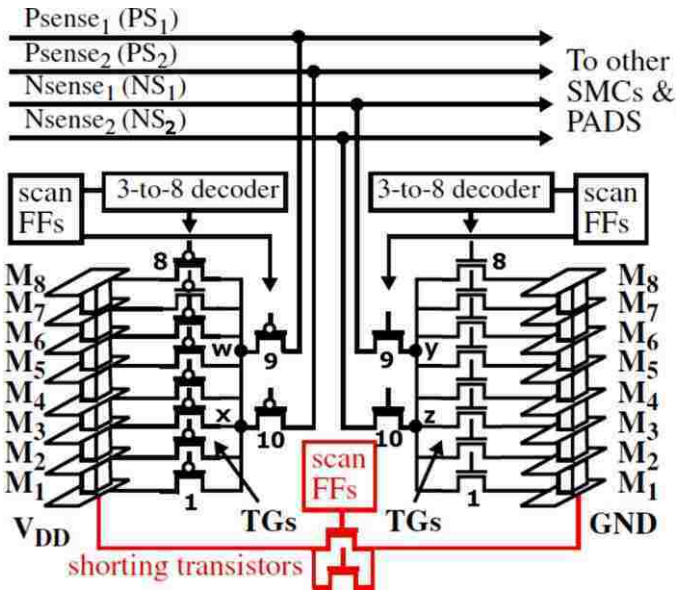
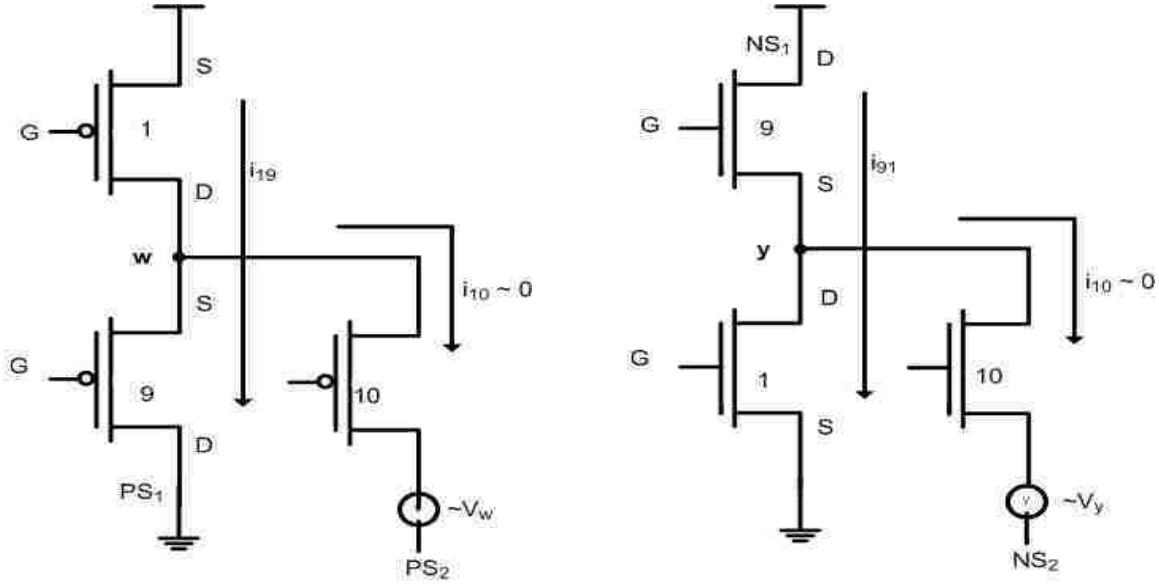


Fig. 8. Stimulus Measure Circuit (SMC) schematic

Voltage drop measurements are carried out by enabling three TGs, both of those labeled 9 and 10 and one from the group 1 through 8. For example, using the PFET TGs, enabling TG 1 and 9 creates a short between the V_{DD} grid on-chip and a GND node off-

chip. The voltage falls across the two TGs as well as the PS_1 wire. The voltage on the intermediate node w between TG1 and 9 can be sensed with TG10. Only a negligible amount of current flows through TG10 to the voltmeter, so the voltage on node w or y is nearly identical to that at the voltmeter. The on-resistances of the TGs (and the resistance of the PS_1 wire) determine how much of the V_{DD} voltage falls across each of TG1 and 9. Random variations in the on-resistances of TG1 through 8 (referred to subsequently as the stacked NFETs or PFETs) produce different voltage drops as each is enabled. We refer to the voltages at the intermediate node (w for PFETs or y for NFETs) as Transmission Gate Voltages (TGVs). Therefore, it is the TGV that represents a single unit of entropy in the TG-PUF and the basic primitives are shown in Fig. 9. It should be noted that the body of both NFETs are connected to GND while those of both PFETs are connected to V_{DD}.



(a) (b)

Fig. 9. TG-PUF primitives for (a) PFET (b) NFET

Referring to Fig. 9, it is clear that the TG-PUF primitive works as a voltage divider circuit, the intermediate voltages of which are a function of the R_{on} of the transistors. Considering the NFET primitive as an example, (1) illustrates the dependence of the TGV voltage V_y on the individual transistor R_{on}.

$$V_y = V_{DD}\left[\frac{R_1}{R_1 + R_9}\right] = V_{DD}\frac{1}{1+\frac{R_9}{R_1}} \tag{1}$$

where R_1 is the R_{on} of TG1 (Stack NFET TG) and R_9 is the R_{on} of NFET TG9.

The component of the TGV that falls across the sense wires represents a bias because, as mentioned previously, the length of the sense wires is different for each SMC in the array. This bias is illustrated in Fig. A1 in Appendix A using experimental data collected from one of our chips. The bias is eliminated by creating TGV differences (TGVDs) using the 8 TGVs measured within each SMC, separately for NFETs and PFETs. Refer to Fig. A2 in Appendix A for an illustration of the bias removal using experimental data collected from one of our chips. The TGVDs are obtained by subtracting pairs of TGV values. With 8 TGVs, a total of 8*7/2 = 28 TGVDs can be created in each stack. The total number of TGVDs obtained per chip is 2,380 for each of the PFETs and NFETs, obtained as 85 SMCs * 28 TGVDs/SMC. The NFET and PFET TGVDs, in turn, can be compared under all combinations to produce bitstrings of length 2,380*2,379/2 * 2 = 5,662,020 bits. It should be noted that the NFET and PFET TGVDs cannot be compared with each other primarily because of channel width differences

(PFETs are 2.5x wider than the NFETs) and as shown in Fig. 10, mobility variations with doping (NFET variations are larger than PFET variations). As a consequence, PFET voltage variations are only about half as large as the NFET variations (see Figs. A4 and A6 in Appendix A). In our experiments, the order in which the comparisons are made is randomized using srand(seed) and rand() from the C programming library. This operation is easily implemented on chip using an LFSR and a seed.

This "differences" comparison strategy to eliminate sense wire bias was devised after some preliminary experimentation with an "absolute" comparison strategy where no difference operation was done. The results of those experiments are presented in a later section.

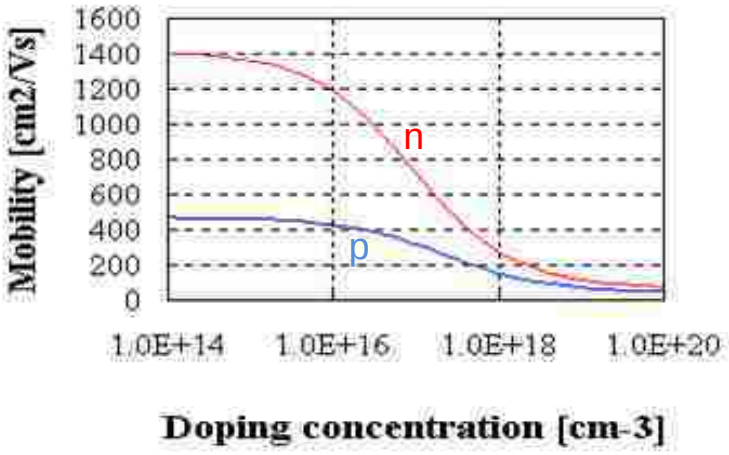


Fig. 10. Mobility as a function of doping concentration in Si [71]

3.3 Power Grid PUF (PG-PUF)

The PG-PUF was introduced in 2012 [21] and analyzed in detail in [22]. It leverages the resistance variations in the metal layers, specifically that of those that make up the power and ground grids of the chip. For the PG-PUF experiments, the SMC of Fig. 8 is configured so that the pair of shorting transistors is always on and sinks approx. 10 mA of current through the power grid. This results in a voltage drop/rise on the V_{DD} and GND grid, respectively of less than 10 mV. The set of 16 'pseudo' transmission gates (TGs) in the stack, labeled 1 through 8, serve as voltage sense devices for the PG-PUF. Eight of these TGs connect to the first 8 (of the 9) metal layers that define the V_{DD} stack-up of the power grid, as shown on the left side of Fig. 8, while the other 8 connect to the GND stack-up. Scan FFs and 3-to-8 decoders allow exactly one of the TGs to be enabled in each of the stack-ups.

For the PG-PUF experiments, TG9 (one for V_{DD} and one for GND) is enabled while TG10 is disabled. Separate scan FFs control the TG9 connection to the chip-wide wires that route to the PS_1 and NS_1 pins of Fig. 7, which are in turn routed to off-chip VMs. This configuration and control mechanism allows any V_{DD} and GND voltage to be measured using off-chip VMs.

A 'challenge' is applied by configuring the scan chain to 1) enable the shorting transistors within an SMC, and 2) enable two TGs in that same SMC, in particular, the TG labeled 9 in Fig. 8 and one from the group 1 through 8. Once enabled, the voltage

drop/rise, denoted as Power Grid Voltages (PGVs), is measured on the NS_1 and PS_1 pads using VMs.

In order to reduce bias effects and correlations that exist in the V_{DD} and GND stack-ups for the PG-PUF, inter-layer voltage drops/rises are computed by subtracting pair-wise, the voltages measured from consecutive metal layers, i.e., VM_1 - VM_2, VM_2 - VM_3, etc [21]. These voltage differences, called Power Grid Voltage Differences (PGVDs), also allow the PUF to leverage the independent resistance variations that occur in each of the metal layers of the power grid. The 8 TGs in the V_{DD} and GND stacks as shown in Fig. 8 indicate that 7 PGVDs can be computed per stack. However, the structure of the power grid on the chips reduces the voltage drops on the upper layers of the power grid. Therefore, the analysis is restricted to PGVDs generated using the lower 4 metal layers, which allows 3 PGVDs to be computed. Therefore, each chip generates 85 SMCs * 3 metal layer pairings = 255 PGVDs for each of the V_{DD} and GND stacks. Each of the PGVDs can be compared with other PGVDs in various combinations to produce a bitstring. Bitstrings are generated by comparing each PGVD with all others generated using the same metal layer pairing. Therefore, the total number of bits per chip is 85*84/2 per metal layer pairing * 3 metal layer pairings * 2 grids = 3,570* 6 = 21,420 bits.

Each of the 340 stacked NFET TGs (since we restrict our analysis to the lower 4 metal layers as described previously) is enabled, one at a time, and the corresponding PGV is measured using a VM connected to NS_1. The current through the shorting transistors path is also measured so as to allow Power Grid Equivalent Resistance

(PGER) calculations. This process is repeated for the stacked PFET TGs. The mean values of 11 samples are used to compute inter-layer voltage drops/rises (PGVDs).

In the experiments, the order in which the comparisons are made is randomized using srand(seed) and rand() from the C programming library. This operation is easily implemented on chip using an LFSR and a seed.

For the purposes of this thesis, only an analysis of the stability of the PG-PUF to changing TV conditions is presented. A more thorough analyses of other characteristics of this PUF and the generated bitstrings is covered in [21] and [22].

3.4 Inverter PUF (I-PUF)

The Inverter PUF was introduced in 2013 [72] and leverages the resistance variations in transistors, specifically of those that are configured in the SMC as inverter-like structures, although they do not operate as inverters in the traditional sense.

The schematic diagram and configuration of the SMC setup that enables the I-PUF is shown in Fig. 11.

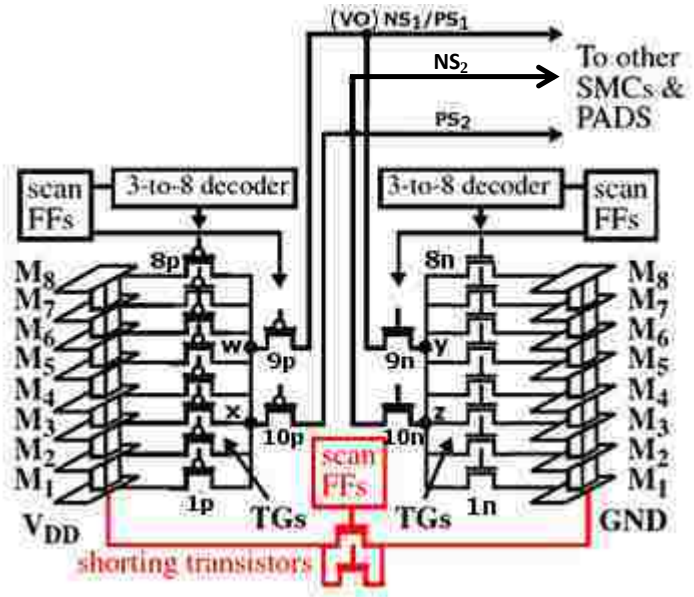


Fig. 11. I-PUF SMC setup schematic

A set of 20 MOSFETs (10 NFETs and 10 PFETs) serve as both the PUF primitives and voltage sensing elements for the I-PUF. The eight stacked PFETs (1p-8p) connect to the first 8 (of the 9) metal layers that define the V_{DD} grid, as shown on the left side of Fig. 11, while the eight stacked NFETs (1n-8n) connect to the GND grid. Two additional PFETs (and NFETs), labeled as 9p and 10p, connect to the drains of the stacked transistors. Separate scan FFs control their connection to the chip-wide wires that route to the P/NS_x pins shown in Fig. 11. The PS_1 and NS_1 sense wires are tied together and connect to an off-chip Agilent 34401A voltmeter that measures the voltage on the NS_1/PS_1 sense wire. The PS_2 and NS_2 sense wires are also tied together in a similar fashion and connect to a VM that measures the voltage on the NS_2/PS_2 wire. The pair of shorting transistors in Fig. 11 is always off during the Inverter PUF experiments so as to eliminate these shorting transistors from the current path.

Voltage drop measurements are carried out by enabling four transistors at a time. Referring to Fig. 11, these four transistors are NFET9n and PFET9p, and any one stacked PFET and any one stacked NFET. This setup creates a short between the V_{DD} and GND grids and the voltage falls across the four MOSFETs and the sense wire. The voltage on the intermediate node w between the stacked PFET and PFET9p can be sensed by enabling PFET10p. This would appear as a voltage measurement on the VM connected to the NS_2/PS_2 sense wire. The voltage on the node between PFET9p and NFET9n is measured by the VM connected to the NS_1/PS_1 sense wire, while the intermediate voltage y between NFET9n and the stacked NFET is measured by enabling NFET10n and registering the measurement on the VM connected to the NS_2/PS_2 sense wire. Therefore, we are able to create 8 paths/SMC X 85 SMCs/chip = 680 different paths per chip. Now, following the aforementioned setup, as we enable different stacked NFETs 1n-8n and stacked PFETs 1p-8p one at a time, we are able to get random and unique voltage measurements at each of the three intermediate nodes. The on-resistances of the MOSFETs (and the resistance of the sense wire) determine how much of the V_{DD} voltage falls across each of the MOSFETs. Random variations in the on-resistances of the stacked PFETs and NFETs produce different voltage drops as each is enabled. It should also be noted that the body of both NFETs are connected to GND while those of both PFETs are connected to V_{DD}. We refer to the voltage at the intermediate node between PFET9p and NFET9n as the inverter Output Voltage (VO). Therefore, it is the VO that represents a single unit of entropy in the I-PUF and the basic primitive is shown in Fig. 12.

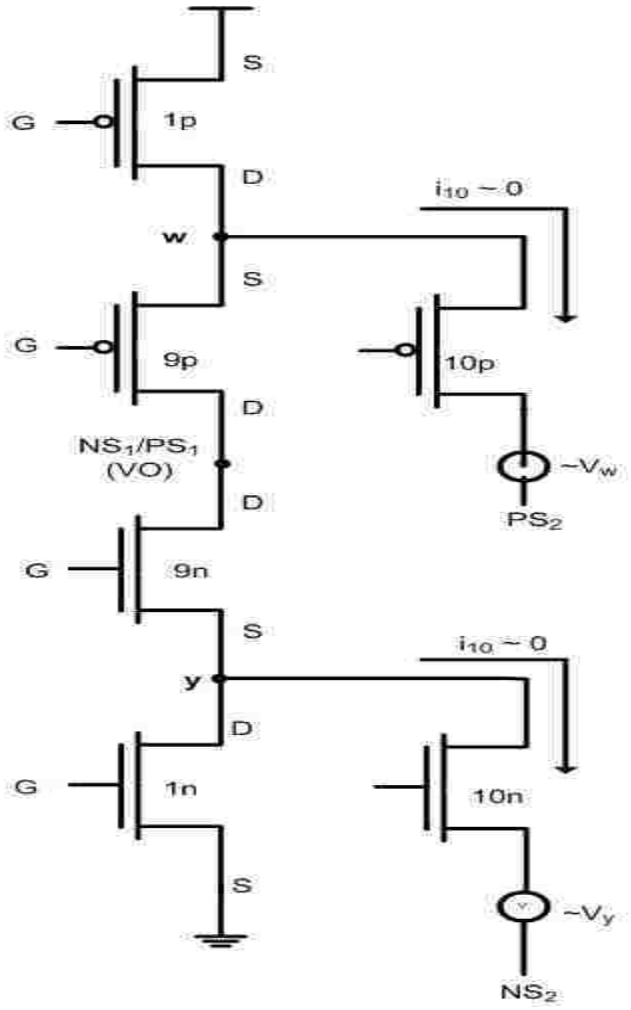


Fig. 12. I-PUF primitive

The I-PUF primitive also works as a voltage divider circuit, the intermediate voltages of which are a function of the R_{on} of the transistors. Equation (2) illustrates the dependence of the voltage VO on the individual transistor R_{on}.

$$VO = V_{DD}\left[\frac{R_{N1}}{R_{N1}+R_{P1}}\right] = V_{DD}\frac{1}{1+\frac{R_{P1}}{R_{N1}}} \tag{2}$$

where R_{N1} is the sum of the R_{on} of NFET1n (stacked NFET) and NFET9n and R_{P1} is the sum of the R_{on} of PFET1p (stacked PFET) and PFET9p.

The component of VO that falls across the sense wires represents a bias because the length of the sense wires is different for each SMC in the array. This bias is illustrated in Fig. A3 in Appendix A using experimental data collected from one of our chips. The bias is eliminated by creating output voltage differences (VODs) using the 8 VOs measured within each SMC. The VODs are obtained by subtracting pairs of VO values. With 8 VOs, a total of 8*7/2 = 28 VODs can be created for each SMC. The total number of VODs obtained per chip is 2,380, obtained as 85 SMCs * 28 VODs/SMC. These VODs, in turn, can be compared under all combinations to produce bitstrings of length 2,380*2,379/2 = 2,831,010 bits.

This “differences” comparison strategy to eliminate sense wire bias was devised after some preliminary experimentation with an “absolute” comparison strategy on the TG-PUF where no difference operation was done. The results of those experiments are presented in a later section.

The VO voltages exhibit much larger variation than the TGV voltages in the TG-PUF as illustrated in Fig. A8 of Appendix A. The reason for this is that the variation in the on-resistances of the two PFETs and the two NFETs are combined in the I-PUF primitive, whereas in the TG-PUF, the primitives contained either two NFETs or two PFETs but not both. This combined PFET path and the combined NFET path, which are responsible for determining the mid-point VO voltage, are much larger than the on-resistances of just the NFETs (used to determine the mid-point TGV voltage of the NFET

TG-PUF) or the PFETs (used to determine the mid-point TGV voltage of the PFET TG-PUF). Similar to how greater voltage variation is seen on the higher resistance lower metal layers of a chip versus the lower resistance upper metal layers, the variation in the higher on-resistance of the combined PFET and combined NFET paths of the I-PUF primitive result in a much larger voltage variation as compared to the TG-PUF.

3.5 On-chip Voltage-to-Digital Converter (VDC)

3.5.1 VDC Functionality

For the TG-PUF and I-PUF, in addition to analyzing the TV stability characteristics of the voltage drops and on-resistances, we also briefly analyze the TV stability of a digital representation of them that is produced by an on-chip VDC, similar to designs described in [73]. The architecture of the VDC is shown in Fig. 13. The VDC is designed to 'pulse shrink' a negative input pulse as it propagates down a current-starved inverter chain. As the pulse moves down the inverter chain, it activates a corresponding set of latches to record the passage of the pulse, where activation is defined as storing a '1'. A Thermometer Code (TC), i.e., a sequence of '1's followed by a sequence of '0's, represents the digitized voltage.

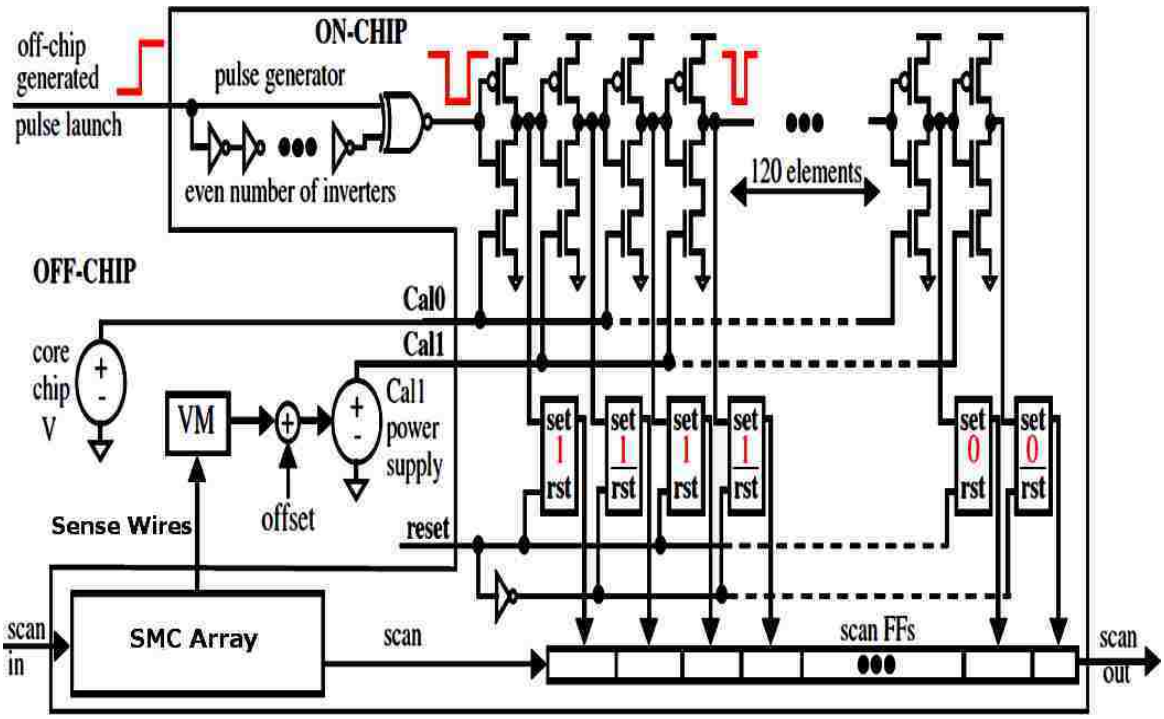


Fig. 13. On-chip Voltage-to-Digital-Converter architecture

The voltage-to-digital conversion is accomplished by introducing a fixed-width (constant) input pulse, which is generated by the pulse generator shown on the left side of Fig. 13. Two analog voltages, labeled Cal0 and Cal1 connect to a set of NFET transistors in the inverter chain, with Cal0 connecting to the NFETs in odd numbered inverters and Cal1 connecting to the NFETs in even numbered inverters. The propagation speed of the two edges associated with the pulse is controlled separately by these voltages. The Cal0 voltage controls the propagation speed of the back-edge while the Cal1 voltage controls the speed of the front-edge. The back-edge of the pulse catches up to the front-edge eventually and in order to ensure that the pulse shrinks as it propagates down the inverter chain, the Cal0 voltage needs to be larger than the Cal1 voltage. This behavior is

illustrated with the aid of Fig. 14 which depicts a delay element from the on-chip VDC and the associated rise and fall times of the pulse. Assuming the width of the input pulse in_1 is T and the width of the output pulse out_2 (after passing through 1 delay element) is T', then

$$T' = T + t_{d1f} + t_{d2r} - t_{d1r} - t_{d2f} = T + [(t_{d1f} - t_{d1r}) + (t_{d2r} - t_{d2f})] \quad (3)$$

where t_{d1f} and t_{d1r} are the fall time and rise time, respectively, of the out_1 pulse while t_{d2f} and t_{d2r} are the fall time and rise time, respectively, of the out_2 pulse. Therefore, if we define T' as $T - T_{LSB}$, then it can be deduced that the original pulse width T is reduced by one T_{LSB}, which is the resolution of the VDC defined by:

$$T_{LSB} = (t_{d1r} - t_{d1f}) - (t_{d2r} - t_{d2f}) \quad (4)$$

Also, noteworthy from Fig. 14 is that t_{d1f} is inversely proportional to the Cal0 voltage while t_{d2f} is inversely proportional to the Cal1 voltage. The pulse will eventually die out at some point along the inverter chain when the back edge of the pulse 'catches up' to the front edge. A digital representation of the voltages can then be obtained by counting the number of '1's in the latches.

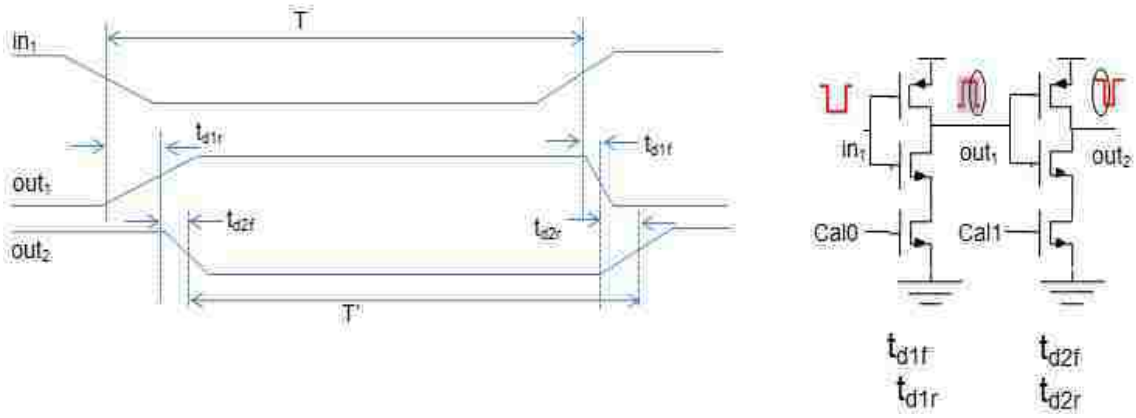


Fig. 14. Delay element of VDC

It was determined through preliminary experimentation that in order to enable the desirable type of pulse shrinking behavior, the Cal1 voltage needs to be set to a value between 500 mV and 800 mV. It was determined that if the Cal1 voltage is less than 500mV, the pulse dies too quickly to maintain a good sensitivity and register an accurate measurement, while if the Cal1 voltage is greater than 800mV, the pulse doesn't die and causes overflow. This is because of the relative pull-down strengths of the current-starved inverters and how that pertains to the rise and fall times of (3).

3.5.2 VDC Data Collection Process

For the TG-PUF, each of the 680 stacked NFET TGs is enabled, one at a time, and the corresponding TGV is measured using a VM connected to NS_2. The current through the path is also measured so as to allow on-resistance calculations of the individual TGs. The

Cal1 power supply is programmed with this TGV plus an offset and calibration factor (the details of which are described in a later section) and 11 TC samples are collected from the VDC. This process is repeated for the 680 stacked PFET TGs. The mean value of the 11 samples is used to compute a 'difference' value, synonymous to the TGVDs described previously. We use the term TCD to refer to these Thermometer Code Differences in the remainder of this thesis. The Cal0 power supply is programmed with the core chip power supply voltage value (V_{DD}) in order to produce a negative shrinking pulse.

For the I-PUF, each of the 680 four-transistor paths are enabled one at a time and the intermediate output voltages (VO and those at w and y) are measured using a VM connected to the corresponding sense wire. The current through the path is also measured so as to allow on-resistance (R_{on}) calculations of the individual MOSFETs. 11 voltage and current samples are collected to ensure statistically valid data. The Cal1 power supply is programmed with these VO values plus a calibration factor (the details of which are described in a later section) and 11 TC samples are collected from the VDC. The mean value of the 11 samples is used to compute a TCD value, synonymous to the VODs described previously. The Cal0 power supply is programmed with the core chip power supply voltage value (V_{DD}) in order to produce a negative shrinking pulse.

3.5.3 Need for Voltage Offsets and Calibration Factors

As stated previously, for the TG-PUF and the I-PUF, the Cal1 power supply is programmed with the measured PUF primitive voltage plus an offset and/or a calibration factor. This is required for two different reasons as explained below.

For the TG-PUF, the voltage-divider (series) arrangement of the identically-sized TGs shown in Fig. 9 should provide voltages at the midpoint of the supply voltage, e.g., approx. 600 mV for a 1.2V V_{DD}. This is not the case, however, for two reasons; 1) a portion of the voltage falls across the NS_1 (for NFETs) and PS_1 (for PFETs) sense wires, and 2) the series-connected transistors in the shorting path operate in different regions of operation, e.g., for both NFETs and PFETs, TG9 and TG1 in Fig. 9 operate in saturation mode and linear modes, respectively. As a consequence, the range of the TGVs observed in our experiments at node w in Fig. 9 for PFETs is between 950 mV to 1050 mV, and at node y for NFETs is 150 mV to 250 mV. As stated earlier, the desirable range for the Cal1 voltage is between 500mV to 800mV, therefore in order to move Cal1 into that range, an offset voltage is added (subtracted) to the TGV voltages measured by the VM as shown in Fig. 10 for NFETs (PFETs). This offset voltage is computed as part of a calibration process briefly described below and expanded upon in the next section.

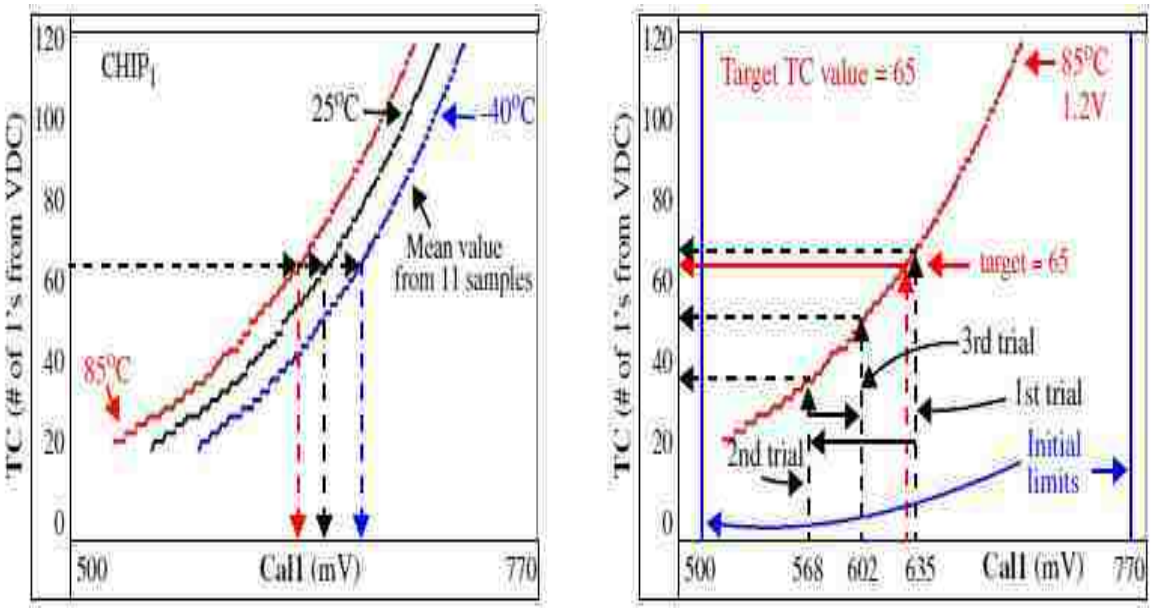
The calibration process is needed because the required offset voltage changes as a function of changing TV conditions. From our experiments, the results of which are presented in a later section, we found that the VDC curves shift with changing TV conditions. To be precise, when the TC versus Cal1 voltage characteristics of the VDC

are plotted for all 9 TV corners, it is seen that the curves shift along the x-axis. Although the VDC remains stable across the TV corners, this shift along the x-axis causes overflow in the VDC; a situation where the pulse propagates through all 120 delay chain elements and therefore, no meaningful data is discerned. A calibration process is carried out that tunes the 'offset' at each TV corner, and effectively eliminates the adverse effects of the curve shift. Basically, the calibration process tests a distributed set of 9 TGs, e.g., of the 680 NFET TGs, and uses binary search to find an offset voltage that produces a 'target' TC, separately for each of the 9 tests. This is done for each of the 9 TV corners. We set the target TCs for NFET and PFET TGVs to 65 and 85, respectively. These targets worked well to prevent overflow in all of the 1,360 TG measurements, across all TVs and chips used in our experiments. The median offset from the 9 calibration tests (for each of the 9 TV corners) is then added to all the TGV voltages measured during the subsequent data collection process. This calibration procedure only approximates the best offset, but does not need to be precise because the goal is only to prevent overflow in the VDC. A more detailed explanation of the process is given next.

It should be noted that unlike the TG-PUF, the I-PUF does not require a voltage offset to be added to the VO voltages to bring it into the optimal 500 mV – 800 mV range for Cal1. This is because the inverter-like design of the I-PUF primitive ensures that the VO voltage is close to half of the power supply voltage or around 600 mV. However, the calibration process is still needed for the I-PUF due to the shifts seen in the VDC curves with changing TV conditions.

3.5.4 VDC Calibration Process

The calibration process briefly described in the previous section is further illustrated using the Cal1 vs. TC curves for the TG-PUF shown in Fig. 15. As indicated earlier, calibration is carried out before enrollment and regeneration only once during PUF characterization, and its objective is to find an appropriate Cal1 voltage offset that prevents overflow in the VDC for any of the TGVs that will be measured during bit generation at any TV.



(a) (b)

Fig. 15. (a) VDC calibration curves at 85, 25, and -40ºC and 1.2V illustrating the offset calculation process (b) Illustration of the binary search process used during calibration at 85ºC, 1.2V

We determined that testing a subset of 9 TGs during calibration is sufficient to obtain a good predictor for offset voltage that prevents overflow. The goal of calibration is to select an offset voltage such that the TG-under-test produces the same TC value independent of the TV corner. This objective is illustrated in Fig. 15(a) with the horizontal dashed line at TC = 65. The 3 curves shown represent the mean values produced by the VDC on Chip1 as the Cal1 voltage is swept across a range of values at 3 different temperatures. The different positions of the dashed vertical lines from each curve make it clear that the offset voltage needs to change in order to maintain a value of 65 in the VDC. Note that the TGV itself measured from the TG-under-test will also change as a function of temperature. This situation is handled by using the TGVs directly in the calibration process (as opposed to using a special voltage source).

Calibration is carried out by enabling each of a select, distributed group of TGs, one at a time, and performing a binary search. The search process varies the Cal1 voltage offset until the TG-under-test produces a specific TC value. The process is illustrated in Fig. 15(b) using the 85ºC Cal1-TC curve from Fig. 15(a). The initial limits are set to 500 mV and 770 mV. The 1st trial selects the midpoint between these limits, i.e., 635 mV. Note this midpoint voltage is the sum of the TGV and the offset voltage that is being tuned in the search. The 1st trial produces a TC of approx. 68, which is larger than the target. Therefore, the next trial uses 635 mV as the upper limit and the new midpoint voltage becomes 568. The 2nd trial produces a TC of 35, so 568 is used as the lower limit for the new midpoint. The process continues until an offset is found that produces a TC of 65. The binary search process is repeated using 9 TGs as a means of obtaining a value

that best approximates the average behavior. The median value from the 9 calibration tests is used as the final offset, which is added to all subsequent TGVs measured at this TV corner.

Chapter 4

Unstable Bits – Cause and Effect

In our experiments, we found that unstable bits, defined as bits that are susceptible to 'flipping' because their TGVDs and VODs (for the TG-PUF and IPUF) or PGVDs (for the PG-PUF) are very similar, actually reduce several quality metrics associated with the overall bitstring, including inter-chip HD and NIST statistical test scores [22][70]. Moreover, including unstable bits in the bitstring requires the inclusion of error correction [19] and Helper Data schemes [24] that weaken security and increase overhead. An alternative scheme called thresholding that identifies and discards unstable bits, was proposed in [22][70] to address the unstable bit issue. However, this thresholding scheme eliminates a large percentage of the bits indicating that a large percentage of the bits produced by the PUFs are unstable and thus, unusable.

Bit flips occur when the relative ordering of a pair of TGVDs, VODs, or PGVDs defined during enrollment reverse order during regeneration at different TV conditions. This manifests itself as a reversal in order of a pair of TCDs for a specific TV, as illustrated in Fig. 16, since TCD is the VDC-digitized representation of the TGVD, VOD, or PGVD voltages. Therefore, from Fig. 16 it is clear that the reversal of the slope from positive to negative or vice versa is what constitutes a bit flip. This reversal is much more likely to occur for pairs of TGVDs, VODs, or PGVDs that are similar in magnitude. Since we observe a significant number of bit flips with changing TV, it implies that the

individual TGV, VO, or PGV values do not change equally with TV changes, causing non-linear shifts in TGVDs, VODs, and PGVDs with TV changes. Upon investigation of this hypothesis using data collected from our chips, we were able to confirm this hypothesis. Figs. A22, A23, and A24 in Appendix A illustrate this based on representative data from one of our chips. Fig. A22 illustrates how the 8 TGV voltages (for the 8 stacked TGs) measured on one of the SMCs of our NFET TG-PUF change with changing TV. As can be seen by the circled points, the TGV voltages do not change equally with changing TV for every stacked TG. These unequal changes in the individual TGVs cause non-linear and disproportionate shifts in the TGVD with TV. Fig. A23 illustrates this same idea for the I-PUF and the magnified view of the circled region of Fig. A23 displayed in Fig. A24 shows the cause of the bit flips. In Fig. A24, the first and second VODs are each calculated by taking the difference in VOs between paths 3 and 4 (of 8) and paths 4 and 5 (of 8) respectively. The enrollment condition establishes the reference for the relative difference in the VOD values and as can be seen, the two VOD values at 85C, 1.2V exhibit a reversal in relative difference as compared to the enrollment condition. Thus, the 85C, 1.2V VO voltage pairings are responsible for the bit flip. It is evident that the underlying unequal shifts in VOs with TV are responsible for the non-linear and disproportionate shifts in the VODs when a VO pairing is taken.

In order to understand the reason for these bit flips in the TG-PUF and I-PUF, it is essential to understand the physical behavior of the transistors that causes unequal shifts in TGVs and VOs at different TV's. Similarly, in order to understand the cause for these

bit flips in the PG-PUF, it is essential to understand the physical behavior of the power grid that causes unequal shifts in PGVs at different TVs.

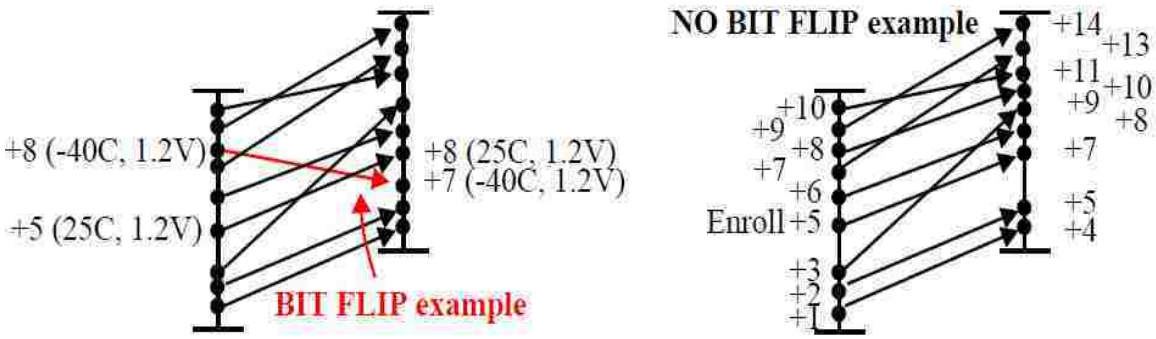


Fig. 16. Illustrative example of a bit flip

4.1 Unstable Bits in the TG-PUF

It is well-known that the On-Resistance (R_{on}) of transistors change non-linearly with TV and the R_{on} shifts with Temperature are a function of both the V_{GS} and V_{DS} of the transistor (operating region of the transistor) [74].

Using the NFET TGs as an example and referring to Fig. 9 (b), it is clear that the TG-PUF primitive works as a voltage divider circuit, the intermediate voltages of which are a function of the R_{on} of the transistors. (5) illustrates the dependence of the TGV voltage V_y on the individual transistor R_{on} at a certain TV.

$$V_y = V_{DD}\left[\frac{R_1}{R_1 + R_9}\right] = V_{DD}\frac{1}{1+\frac{R_9}{R_1}} \qquad (5)$$

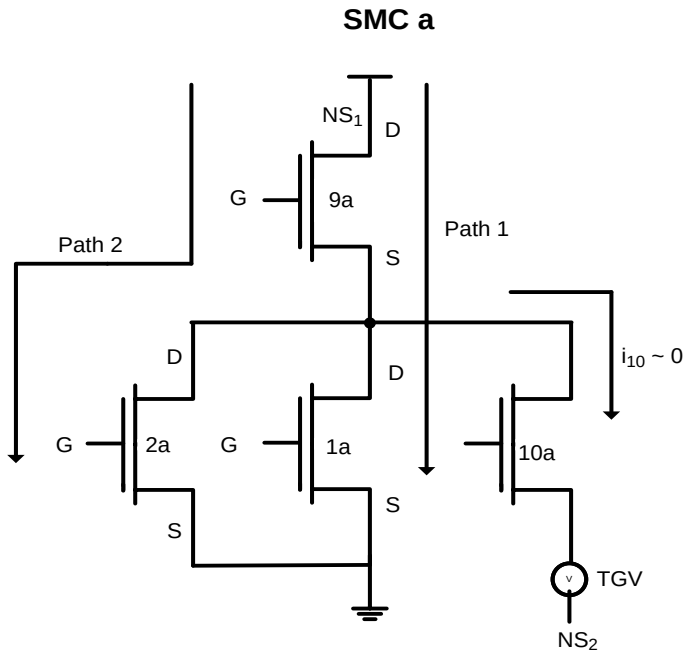
where R_1 is the R_{on} of TG1 (Stack NFET TG) and R_9 is the R_{on} of TG9.

Let us define X_1 as the % change in R_{on} of TG1 at a certain TV from the R_{on} of TG1 at the enrollment condition (25C, 1.2V) and X_9 as the % change in R_{on} of TG9 at a certain TV from the R_{on} of TG9 at the enrollment condition. Note that X_1 or X_9 would be positive for a % increase and negative for a % decrease. Therefore, V_y at a TV other than enrollment is:

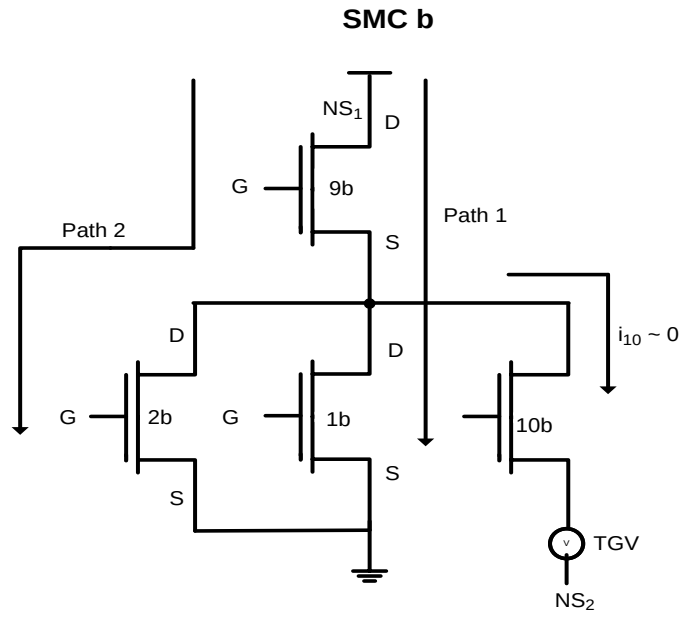
$$V_{DD} \frac{1}{1 + \frac{R_9}{R_1} \cdot \frac{(1 + X_9)}{(1 + X_1)}} \qquad (6)$$

Thus, it is evident that the TGV voltage at a certain TV is inversely proportional to the ratio of the R_{on} of the two transistors at enrollment conditions of 25C, 1.2V (this is the temperature insensitive term) and the ratio of the (1 + %) change in R_{on} in those transistors with respect to their values at enrollment.

Figs. 17 (a) - 17 (b) depict an example of the TG-PUF primitives involved in the calculation of two TGVD values. For example, the first TGVD ($TGVD_a$) is calculated by taking the difference of the TGVs of path 1 and path 2 in SMC a while the second TGVD ($TGVD_b$) is calculated by taking the difference of the TGVs of path 1 and path 2 in SMC b.



(a)



(b)

Fig. 17. NFET TG-PUF primitive for (a) SMC a (b) SMC b

Now, referring to (6), the TV dependence of the difference in the TGV voltages i.e., $TGVD_a$ for SMC a is:

$$TGVD_a = V_{DD}\left[\frac{1}{1+\frac{R_{9a}}{R_{1a}}\frac{(1+X_{9a})}{(1+X_{1a})}} - \frac{1}{1+\frac{R_{9a}'}{R_{2a}}\frac{(1+X_{9a}')}{(1+X_{2a})}}\right] \quad (7)$$

and for SMC b is:

$$TGVD_b = V_{DD}\left[\frac{1}{1+\frac{R_{9b}}{R_{1b}}\frac{(1+X_{9b})}{(1+X_{1b})}} - \frac{1}{1+\frac{R_{9b}'}{R_{2b}}\frac{(1+X_{9b}')}{(1+X_{2b})}}\right] \quad (8)$$

In (7) and (8), the subscripts 1 and 2 refer to the TGV pairing (stacked NFETs TG1a and TG2a or TG1b and TG2b) involved in the specific TGVD calculation, while the subscripts a and b refer to SMC a and SMC b. It should be noted that at a cursory glance at Figure 17 (a), the R_{on} of TG9a (or R_{9a}) and the % change in R_{9a} with TV (or X_{9a}) appear to be unchanged for the TGV pairing being compared to generate the TGVD value. However, this is inaccurate because although TG9a is common for the TGV pairing, the TGV magnitudes are not the same for the pairing and it is the TGV magnitude that determines the V_{GS}, the V_{DS}, the operating region, and the R_{on} value of TG9a, and therefore the amount of shift of R_{on} of TG9a with changing TV. These

different on-resistances and % changes in on-resistance with TV from enrollment are designated by $R_{9a,}$ R_{9a}', X_{9a}, and X_{9a}' in (7).

Upon inspection of (7) and (8), it is evident that the shifts in TGVD with changing TV are non-linear and disproportional. Furthermore, it is evident that these non-linear shifts are a strong function of the ratio of the individual transistor R_{on} values, which change with TGV. Therefore, when comparing two digitized TGVD values to generate a bitstring, it is these non-linear shifts in TGVD with TV that is the primary cause of bit flips. As expected, this same behavior is observed with the TCDs. Therefore, to illustrate this phenomenon in a clear manner, the disproportionate shifts in TCDs with TV for a particular TCD comparison (which would result in a bit flip) from one of our chips is shown in Fig. 81. Fig. 81 is representative of the problem associated with the disproportional TCD and TGVD shifts seen that are responsible for the bit flips in the bitstrings generated from our chips.

4.2 Unstable Bits in the PG-PUF

As stated earlier, the PG-PUF is based on resistance variations that occur in the metal wires of the chip's power grid [21][22]. The TV stability of the PG-PUF has a direct bearing on the number of unstable bits produced by the PUF. A significant benefit of using metal structures is that "noise-related" variations, such as those introduced by TV variations, result in linear changes to the measured voltages. This linear scaling characteristic allows the relative magnitude of two voltages to remain fairly consistent

across changes in TV, which in turn improves the stability of the PUF to bit-flips, when compared, for example to PUFs which leverage transistor-based variations, e.g., TG-PUFs.

The temperature dependence of the electrical resistance of a conductor, such as a copper wire, can be linearly approximated by the following equation:

$$R(T) = R_o + \alpha R_o(T - T_o) \tag{9}$$

where α is called the temperature coefficient of resistance which is an empirical parameter measured at a reference temperature, T_o is a fixed reference temperature (usually room temperature), and R_o is the resistance at temperature T_o.

It should be noted that the shorting transistors from Fig. 8 are very large (57x minimum size) and therefore exhibit smaller variations with TV in comparison to minimum-sized transistors. While these variations are small, we still eliminate them by dividing the PGV voltages by the shorting current and use the term PGERs, for Power Grid Equivalent Resistances, to refer to them. In order to get as 'pure' a form as possible of the PGERs, we also subtract the leakage voltage and leakage current from the values measured with the shorting transistors enabled. Similar to the creation of PGVDs from PGVs, we create PGER differences (PGERDs) by subtracting pairings of PGERs.

In order to determine the magnitude of the TV variations (or 'TV noise'), we calibrate the PGVD and PGERD data. Calibration removes the DC offsets introduced by

TV noise in the data but preserves the variation. Calibration is carried out by computing the mean PGERD and PGVD over the entire set of SMCs for a given metal layer pairing and TV corner. Correction factors are then computed by subtracting the mean value at each of the TV corners from a reference TV corner. In our case, the reference is the data collected at 25C, 1.2V (enrollment conditions). The correction factors are then added to the corresponding data from the TV corners.

4.3 Unstable Bits in the I-PUF

An illustration of the I-PUF primitive is shown in Fig. 18. For representative purposes, only 2 of the 8 primitive paths have been shown in Fig. 18 for a given SMC. Just as was in the case of the TG-PUF, the R_{on} of the I-PUF transistors also change non-linearly with TV and the R_{on} shifts with temperature are a function of both the V_{GS} and V_{DS} of the transistors (operating region of the transistors) [74].

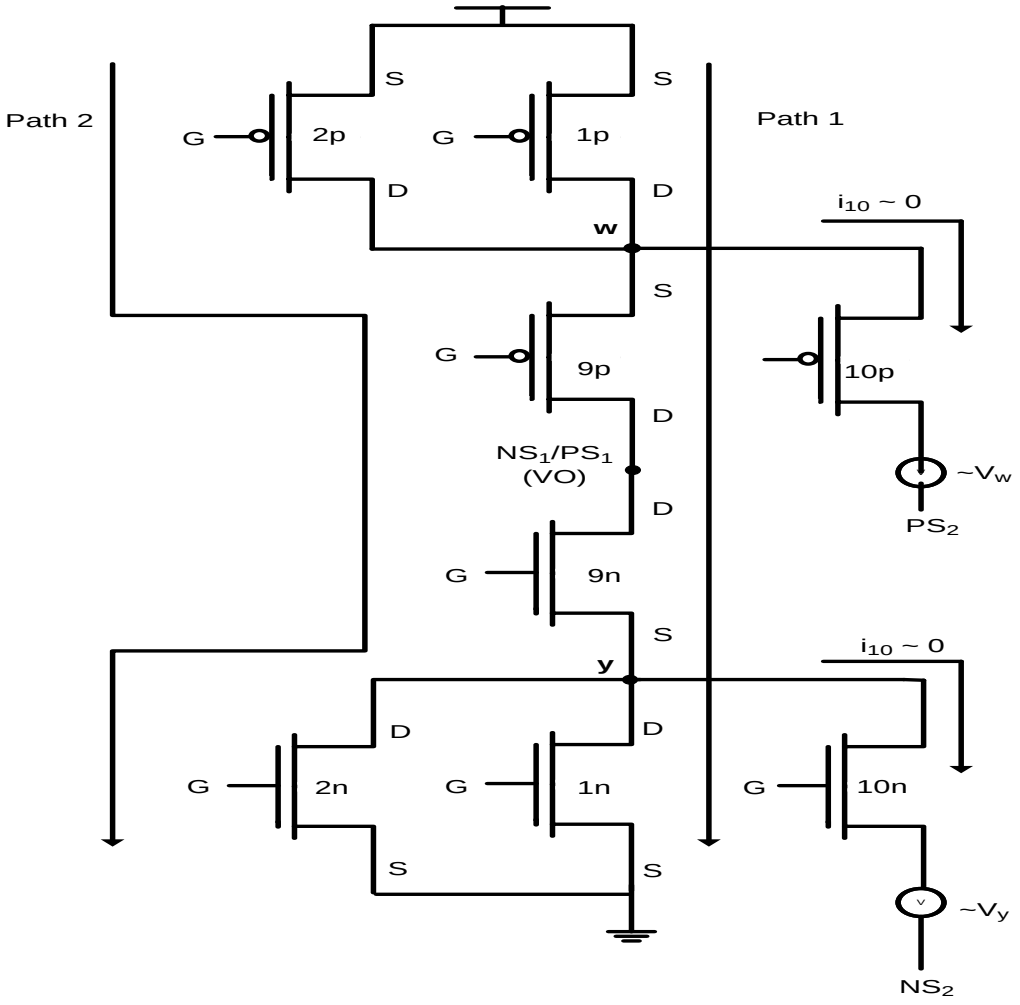


Fig. 18. I-PUF primitive

The I-PUF primitive also works as a voltage divider circuit, the intermediate voltages of which are a function of the R_{on} of the transistors. For example for path 1 in Fig. 18, (10) illustrates the dependence of the voltage VO_1 on the individual transistor R_{on} at a certain TV.

$$VO_1 = V_{DD}\left[\frac{R_{N1}}{R_{N1}+R_{P1}}\right] = V_{DD}\frac{1}{1+\frac{R_{P1}}{R_{N1}}} \qquad (10)$$

where R_{N1} is the sum of the R_{on} of NFET1n (stacked NFET) and NFET9n and R_{P1} is the sum of the R_{on} of PFET1p (stacked PFET) and PFET9p, all at enrollment conditions (25C, 1.2V).

Similar to the TG-PUF, let us define X_{N1} as the % change in R_{N1} at a certain TV from the R_{N1} at the enrollment condition and X_{P1} as the % change in R_{P1} at a certain TV from the R_{P1} at the enrollment condition. Note that X_{N1} or X_{P1} would be positive for a % increase and negative for a % decrease. Therefore for path 1, VO_1 at a TV other than enrollment is:

$$V_{DD}\frac{1}{1+\frac{R_{P1}}{R_{N1}}\cdot\frac{(1+X_{P1})}{(1+X_{N1})}} \qquad (11)$$

The calculation of a VOD value can also be understood by referring to Fig. 18. For example, a VOD value could be calculated by taking the difference of the VOs of path 1 and 2. Referring to (11) and altering subscripts to reflect path 2, this VOD can be written as:

$$\mathrm{VOD} = \mathrm{VO}_1 - \mathrm{VO}_2 = V_{DD}\left[\frac{1}{1+\frac{(R_{1p}+R_{9p})}{(R_{1n}+R_{9n})}\frac{(1+X_{P1})}{(1+X_{N1})}} - \frac{1}{1+\frac{(R_{2p}+R_{9p}')}{(R_{2n}+R_{9n}')}\frac{(1+X_{P2})}{(1+X_{N2})}}\right] \quad (12)$$

where $R_{1p,}$ R_{9p}, $R_{1n,}$ $R_{9n,}$ $R_{2p,}$ R_{9p}', $R_{2n,}$ and R_{9n}' are the individual resistances of the transistors indicated by the subscript. The distinction drawn between R_{9p} and R_{9p}'(or R_{9n} and R_{9n}') is because of the fact that although these are the resistances of the same common transistor, their magnitudes are different when computing two different VOs as the resistances of transistors 9n and 9p change as a function of VO. X_{N2} and X_{P2} are the % changes in R_{N2} (defined as R_{2n} + R_{9n}') and R_{P2} (defined as R_{2p} + R_{9p}') respectively, from their values at the enrollment condition.

Upon inspection of (12) it should be clear that shifts in VOD with TV will be non-linear and disproportional. It is further evident that the % shifts in the individual transistor resistances with TV (from enrollment) contribute to the non-linearity as a weighted function of the individual transistor resistances, e.g. if $R_{1p} > R_{9p,}$ then the magnitude of the % change in R_{1p} with TV will dominate in the overall % change of the combined transistor resistances term (X_{P1}). Therefore, when comparing digitized versions of two VODs to generate a bit string, these non-linear and disproportionate shifts of the individual VODs with changing TV are what leads to bit flips.

Chapter 5

Bit Flip Avoidance Schemes

As discussed earlier, TCDs are computed by subtracting TCs within the same SMC as a means of eliminating the voltage bias introduced by the sense wires. Computing differences also has the benefit of significantly increasing the number of bits that can be produced from each chip. For example, for the TG-PUF, 2380 TCDs are produced from the 680 NFET TCs.

Using difference values, however, has two main drawbacks. First, subtracting two TCs reduces the signal-to-noise ratio because the noise from two separate measurements is combined in the difference. More importantly, TCDs ‘re-use’ the base entropy of the array, therefore, re-use makes model building attacks possible in cases where the bitstring is made public.

As stated previously, the bitstrings generated from our PUFs are categorized into two types. First are those generated directly by comparing the digitized voltages (TGVDs) from the PUF with each other and second are those that are generated by converting the digitized voltages into TCDs with the aid of the on-chip VDC and then comparing those TCDs with each other to generate the bitstring. The results presented include those for both the voltage-comparison derived bitstrings and the VDC-derived bitstrings.

5.1 Thresholding Technique

The thresholding scheme shares characteristics with the shielding function proposed in [75] but is simpler because it is based entirely on strong bits, referred to as ‘robust’ bits in the reference. This fact changes the nature of the public data and eliminates information leakage that, although unlikely, is possible with shielding functions.

A thresholding technique as a means of dealing with model-building attacks and preventing information leakage in the public helper data is proposed. Our thresholding technique discards TCD (or TGVD) comparisons that are susceptible to producing bit flips in the bitstring.

Bit flips occur when the relative ordering of a pair of TCDs (or TGVDs) defined during enrollment reverse order during regeneration. This is much more likely to occur for pairs of TCDs (or TGVDs) that are similar in magnitude. It is shown in the experimental results that it is possible to define a threshold that filters all TCD (or TGVD) pairings that introduce bit flips during regeneration at one or more of the TV corners. The threshold is derived using the distribution characteristics of TCDs (or TGVDs) obtained during enrollment, which is carried out in the experiments at 25ºC and 1.20V.

5.1.1 Thresholding technique applied to the TG-PUF

Using the data derived from the VDC, Fig. 19 shows the TCD enrollment (at 25ºC, 1.2V) distributions for NFETs and PFETs from one of our chips (Chip1). It is clear from the spread of the distributions that the NFET TCDs have more variation than the PFET TCDs, the reasons for which are outlined in Section 3.2.

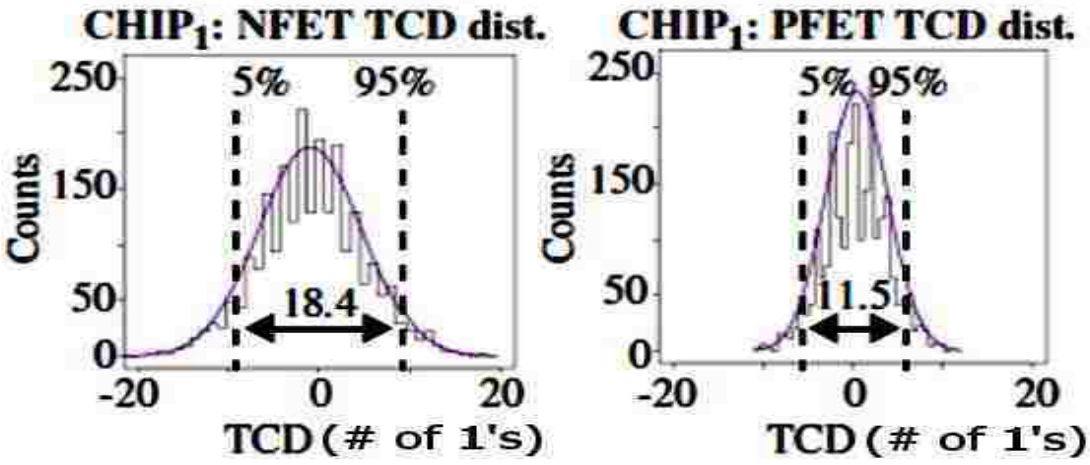


Fig. 19. TG-PUF enrollment NFET (left) and PFET (right) TCD distributions with 2,380 components from Chip1, with inter-percentile ranges delineated.

Without using the VDC and just using the data derived from TGV voltage comparisons, Figs. 20 and 21 shows the TGVD enrollment (at 25ºC, 1.2V) distributions for NFETs and PFETs from Chip1 respectively. It is clear from the spread of the distributions that the NFET TGVDs have more variation than the PFET TGVDs, the reasons for which are

explained in section 3.2. The greater underlying variation in the NFET voltages as compared to the PFET voltages are depicted in Figs. A4 and A6 in Appendix A.

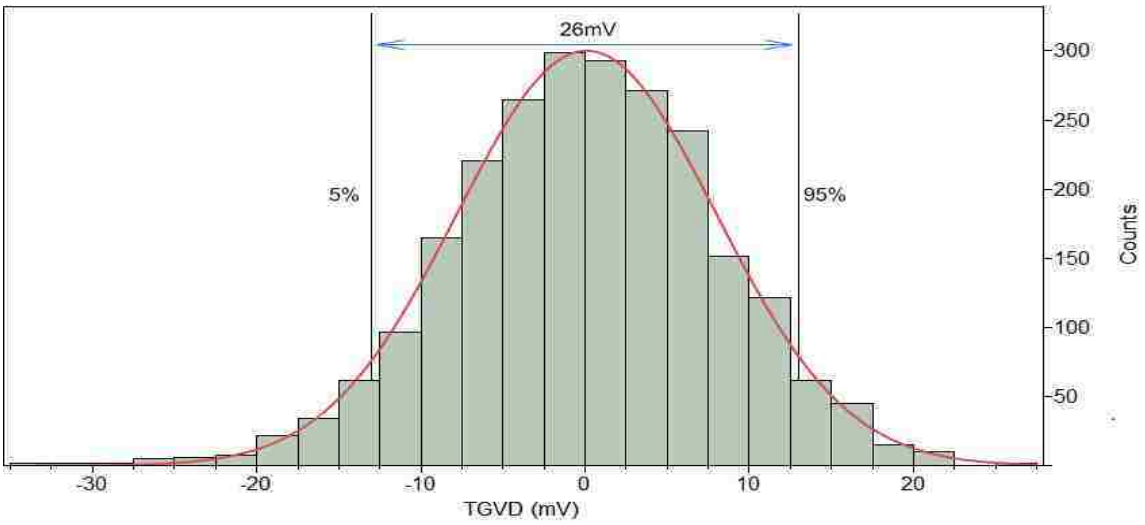


Fig. 20. Enrollment NFET TGVD distributions with 2,380 components from one chip (Chip 1), with inter-percentile ranges delineated.

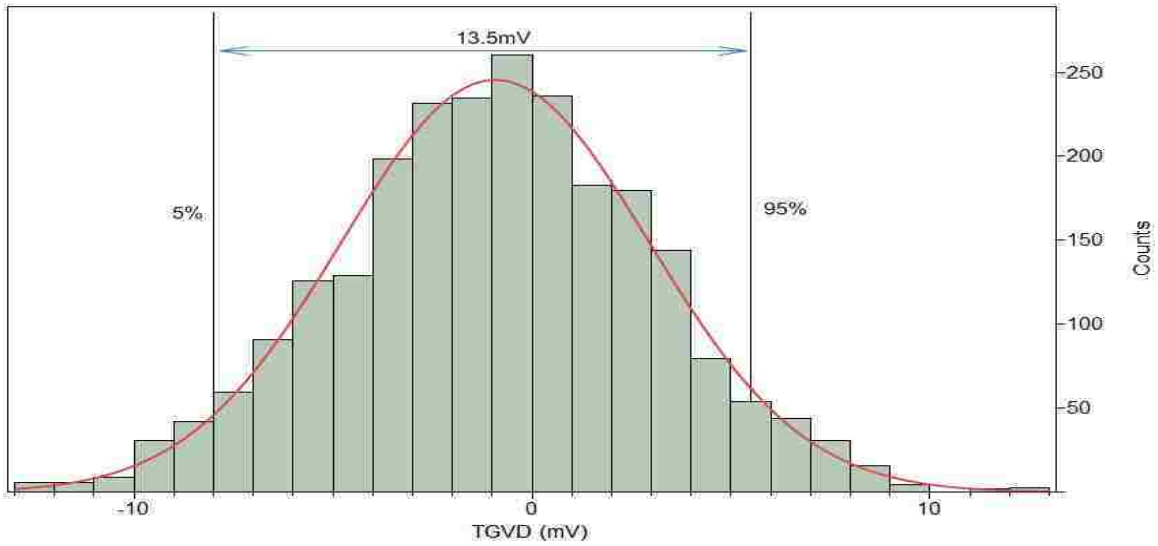


Fig. 21. Enrollment PFET TGVD distributions with 2,380 components from one chip (Chip 1), with inter-percentile ranges delineated.

The objective is to derive a threshold from these distributions that serves three primary goals: 1) avoids bit flips under different TV conditions in the subsequent bit generation phase, 2) preserves as many strong bits as possible for each chip and 3) makes the number of strong bits as consistent as possible across chips, i.e., scales with the range of variation that occurs on each chip. We define strong bits as those generated by TCD or TGVD comparisons where the differences in the TCDs or TGVDs exceed the threshold and those that do not flip during regeneration of the bitstring at any TV.

In our experiments, we found the limits defined by the two vertical lines labeled 5% and 95% in Figs. 19, 20, and 21 achieve these goals. These limits capture the spread of the distribution while ignoring the outliers on the tails of the distributions, which, when included, introduce large variations in the number of strong bits preserved across the chip population, i.e., they degrade criteria 3 above. We then multiply the 2 inter-percentile ranges defined as the distances between these limits by 2 scaling factors, one for NFETs and one for PFETs, to define the 2 TCD or TGVD thresholds for the chip. The scaling factors are set to 0.42 (NFET) and 0.39 (PFET) for the TGVD voltage analysis and 0.53 (NFET) and 0.78 (PFET) for the TCD analysis. These scaling factors were derived by analyzing the bitstrings across all 9 TV corners and tuning the values until no bit flips occurred.

Figs. 22(a) and (b) provide an illustration of the thresholding process applied using the VDC-derived TCD data from one of the chips (Chip1) for the TG-PUF NFETs. The graphs plot bit number along the x-axis against the differences of the TCDs being compared. Only the first 390 strong bits are shown. The horizontal lines at 9.7 and -9.7

delineate the threshold boundaries for the NFET TCDs, which are derived from Fig. 19 using a scaling factor of 0.53. Fig 22(a) shows those TCD differences which produce strong bits during enrollment. From Fig. 22(a), the bitstring is generated by defining the TCD differences greater than the positive threshold value (+Tr) as 1's and the differences smaller than the negative threshold value (-Tr) as 0's. In addition to generating the secret bitstring, a thresholding bitstring is also constructed during enrollment which indicates which comparisons produce strong bits and which produce weak bits. The thresholding bitstring is recorded in public data storage, and using techniques such as run-length encoding (explained in a later section), is proportional in size to the secret bitstring. This type of public data reveals nothing about the secret bitstring, and represents the helper data for our PUF. It should be noted that the thresholding process is implemented only during enrollment, and is disabled during regeneration.

Fig. 22(b) superimposes the TCD difference data points generated under the remaining 8 TV corner experiments, which represent the regeneration scenarios in our experiments. The thresholding bitstring is consulted to ensure regeneration uses the same comparisons as enrollment. The data points associated with the regenerations appear above and below the enrollment data points. Only those that move toward 0 line are problematic however. Although none occur in these plots, points that move over the 0 line from above or below indicate the relative ordering has changed in the TCD pairing. A bit flip will occur during regeneration if this condition is met. Fig. 23 illustrates the threshold method for the Chip1 TG-PUF PFETs using the VDC-derived TCD data.

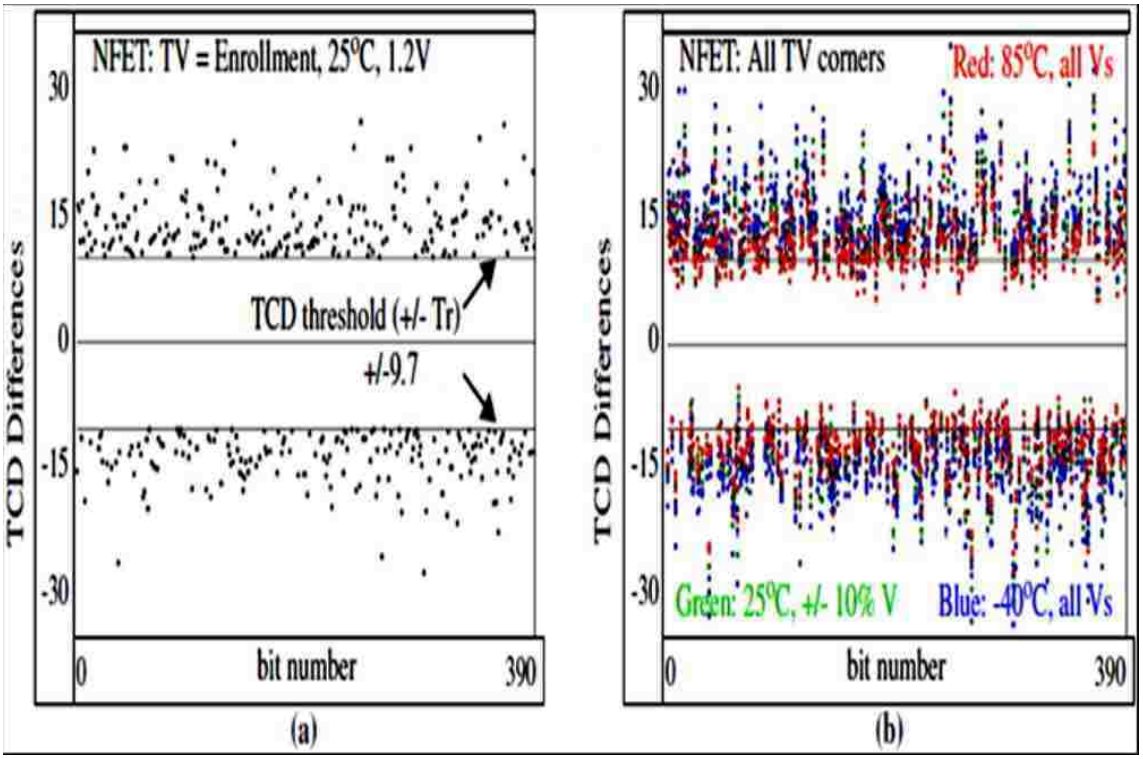


Fig. 22. Threshold method showing the first 390 strong bit comparisons for Chip1 during (a) enrollment and (b) regeneration across 8 TV corners for the NFETs in the TG-PUF

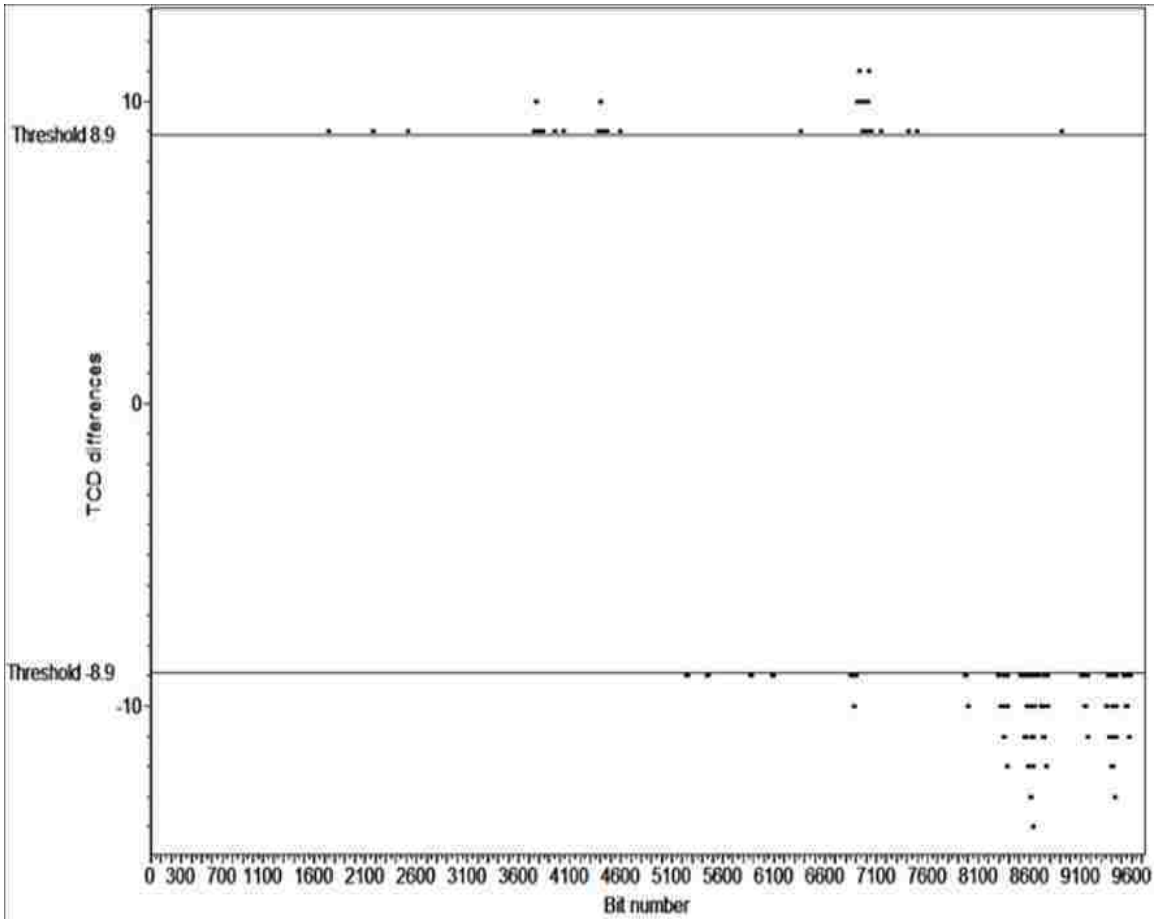


Fig. 23. Threshold method showing the first 9730 (out of 2,831,010) TCD comparisons during enrollment for the Chip1 PFET

Fig. 24 provides an illustration of the thresholding process applied using the voltage-derived TGVD data from one of the chips (Chip1) for the TG-PUF NFETs. The graphs plot bit number along the x-axis against the differences of the TGVDs being compared at enrollment conditions (25C, 1.2V). Only the first 500 bits are shown. The horizontal lines at 10.9mV and -10.9mV delineate the threshold boundaries for the NFET TGVDs, which are derived from Fig. 20 using a scaling factor of 0.42. Fig 24 shows

those TGVD differences which produce strong bits during enrollment. From Fig. 24, the bitstring is generated by defining the TGVD differences greater than the positive threshold value as 1's and the differences smaller than the negative threshold value as 0's. Fig. 25 illustrates the threshold method using the voltage-derived TGVD data for the Chip1 TG-PUF PFETs.

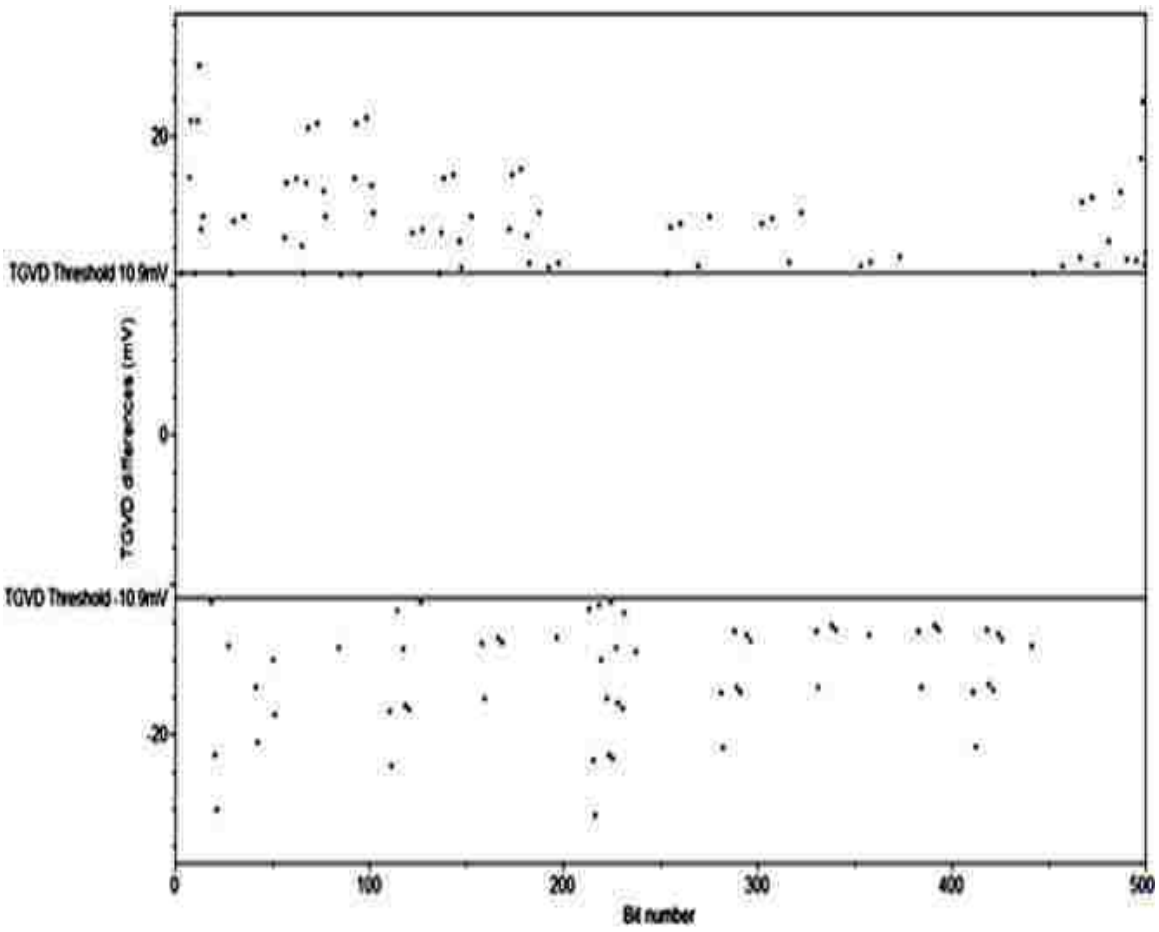


Fig. 24. Threshold method showing the first 500 (out of 2,831,010) TGVD voltage comparisons during enrollment for the Chip1 NFETs

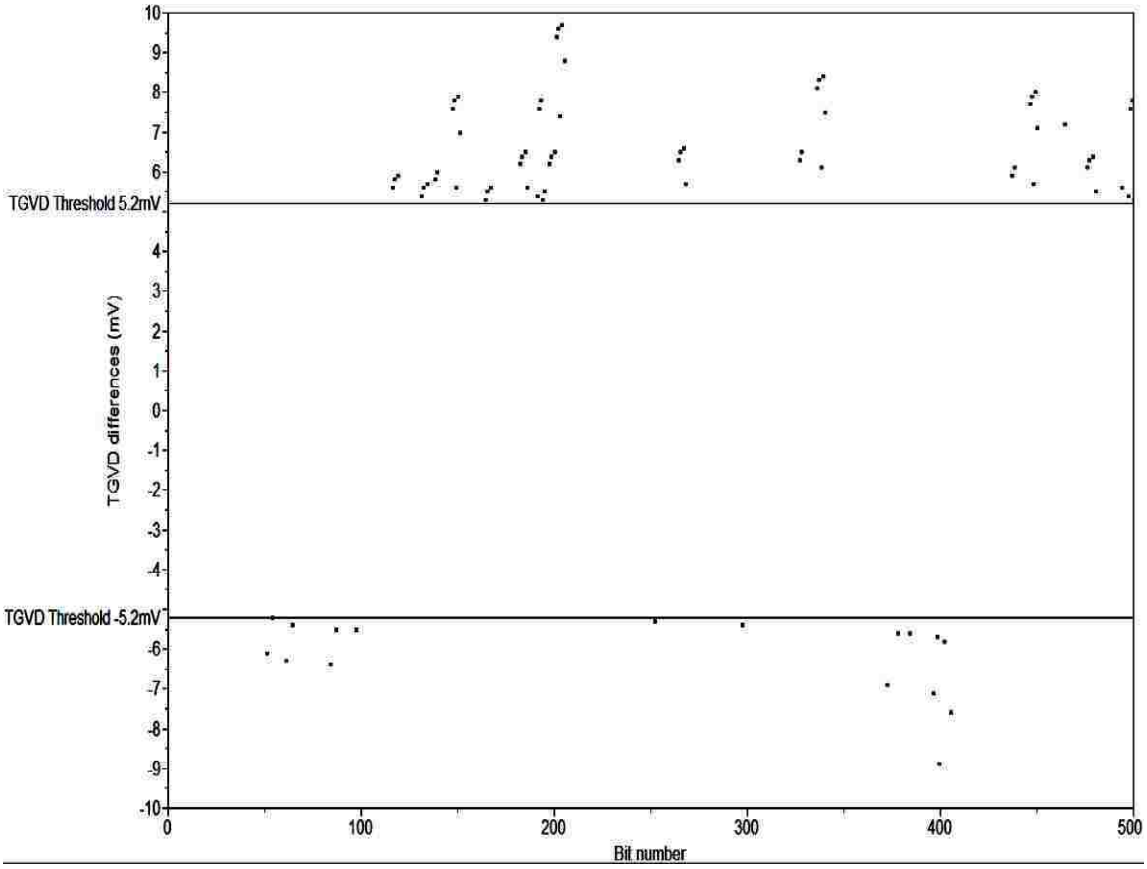


Fig. 25. Threshold method showing the first 500 (out of 2,831,010) TGVD voltage comparisons during enrollment for the Chip1 PFETs

The TCD differences plotted in Figs. 22 and 23 span a larger range than the TCDs used to compute the inter-percentile range from Fig. 19 because the TCDs themselves are both positive and negative and therefore, their differences will have a larger range. It should be clear that the wider the TCD distribution, the larger the magnitude of the differences between TCDs. Despite their larger range, only about 21% of the 2,831,010 possible comparisons, i.e., approx. 595,000 bits, survive the thresholding for NFETs. A similar analysis using the TGVD voltages shows approx. 33% surviving the thresholding, which

suggests that the digitization process adds substantially to the noise. This is even more dramatic in the PFET analysis, where approx. 7% of the TCDs survive and approx. 36% of the TGVDs survive. The smaller variation in the PFET TCDs reduces the signal-to-noise for the VDC even further. However, the 832,343 TCD-based bits for this chip that survive are reproducible across the TV corners and exhibit excellent statistical characteristics. Table II below summarizes these results for the Chip1 TG-PUF while Table III summarizes the results for all 63 chips tested. These results from Chip1 are representative of the results seen from all the chips in the population.

Table II: Threshold and length of VDC-derived and voltage-derived bitstrings for Chip 1 TG-PUF

	VDC-derived bitstrings		Voltage-derived bitstrings	
	NFET	PFET	NFET	PFET
Inter-percentile range	18.4	11.5	26mV	13.5mV
Threshold scaling factor	0.53	0.78	0.42	0.39
Threshold	+/- 9.7	+/- 8.9	+/- 10.9mV	+/- 5.2mV
% of strong bits	21%	7%	33%	36%
Total strong bitstring length	832,343 bits (14.7%)		1,953,397 bits (34.5%)	
Truncated bitstring length	725,230 bits (12.8%)		1,901,845 bits (33.6%)	

Table III. Threshold and ranges of VDC-derived and voltage-derived bitstrings for the TG-PUF from all 63 chips

	VDC-derived bitstrings		Voltage-derived bitstrings	
	NFET	PFET	NFET	PFET
Range of Inter-percentile range	15.7 (Chip9) – 23.9 (Chip56)	8.2 (Chip41) – 11.7 (Chip2)	25.12mV (Chip9) – 36.9mV (Chip21)	10.7mV (Chip34) – 15.56mV (Chip13)
Threshold scaling factor	0.53	0.78	0.42	0.39
Range of Threshold	+/- 8.32 (Chip9) to +/- 12.6 (Chip56)	+/- 6.4 (Chip41) to +/- 9.1 (Chip2)	+/- 10.55mV (Chip9) to +/- 15.5mV (Chip21)	+/- 4.17mV (Chip34) to +/- 6.06mV (Chip13)
Range of % of strong bits	19.3% - 22.9%	5.6% - 8.05%	31.3% - 34.1%	34.9% - 37.9%

From Table III, it can be seen that there is a considerable spread in the distribution of the TGVDs and TCDs of the 63 chips, signified by the range of the inter-percentile ranges. However, it can be seen that the threshold technique does a good job in keeping the preserved bits, represented by the % of strong bits, fairly consistent across the 63 chips. Even though the TGVD and TCD distributions of the chips have a large range, the % of strong bits preserved stays fairly consistent and does not suffer for the chips with tighter distributions. Therefore, this highlights the importance of scaling the threshold of each chip as a function of its specific distribution as that is the only way to ensure a consistent level of strong usable bits across all chips.

The next few pages capture the threshold technique results of all the 63 chips tested for both the VDC-derived bitstrings and the voltage-derived bitstrings. Fig. 26 depicts the behavior of the thresholds and inter-percentile ranges of each of the chips for the NFETs and PFETs (denoted by the different color) for the VDC-derived bitstrings. Each point on this graph is the data from one chip and as expected, the slope of the curve equals the scaling factor. Fig. 27 depicts this for the voltage-derived bitstrings. As mentioned earlier, the NFETs have more variation than the PFETs and this exhibits itself in a wider distribution and thus, a larger inter-percentile range in the NFETs as compared to the PFETs. Noteworthy is the fact that the scaling factors (slopes) are fairly similar for the NFETs and PFETs for the voltage-derived bitstrings but are higher for the VDC-derived bitstrings and markedly higher for the PFETs. This is because of the added noise due to the digitization process involved in the VDC-derived bitstrings and the fact that the TV noise is markedly higher for the VDC-derived bistrings from the PFET. Also, as shown in Fig. 28 for the voltage-derived bitstrings as an example, the PFET threshold is directly proportional to the NFET threshold, i.e. chips with a larger (or smaller) NFET variation also have a larger (or smaller) PFET variation.

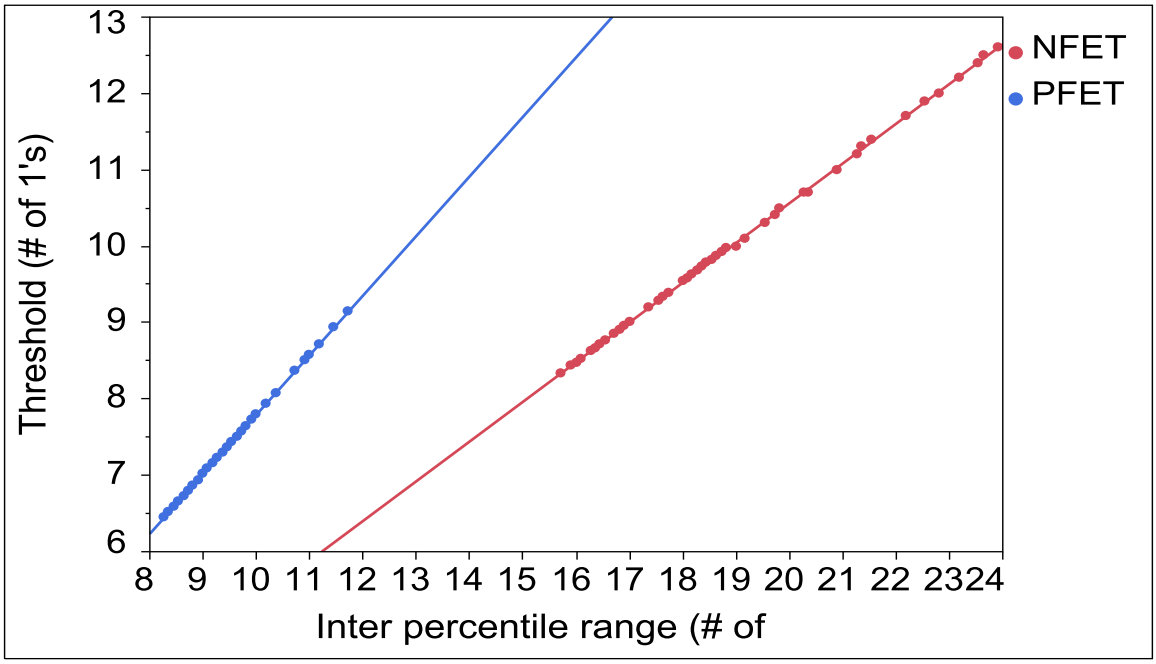


Fig. 26. Threshold versus inter-percentile ranges of VDC-derived bitstrings for the TG-PUF from all 63 chips

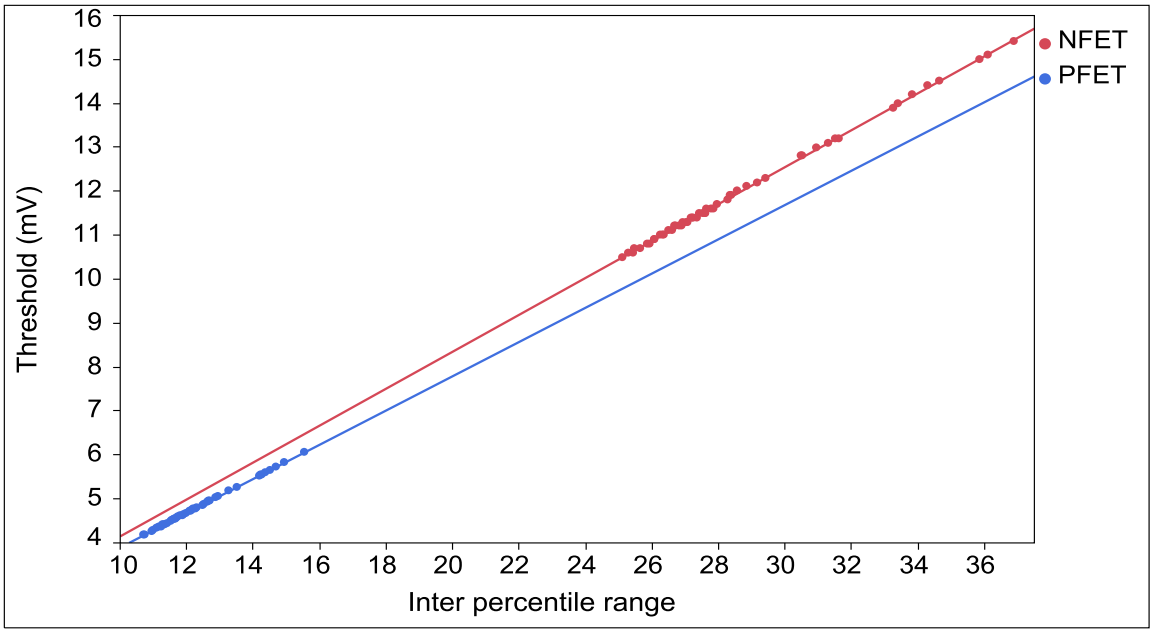


Fig. 27. Threshold versus inter-percentile ranges of voltage-derived bitstrings for the TG-PUF from all 63 chips

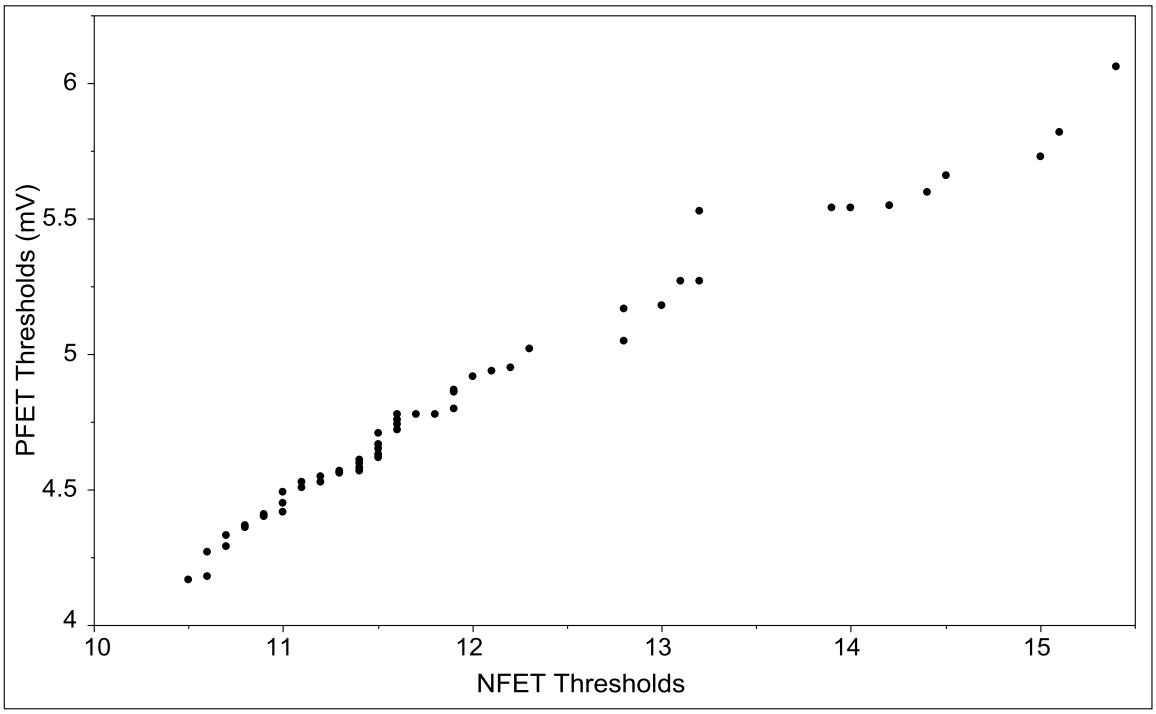


Fig. 28. PFET versus NFET Thresholds for the TG-PUFs from 63 chips

Fig. 29 plots the VDC-derived thresholds versus the voltage-derived thresholds for all 63 chips. Each point in the graph is a single chip and as can be seen from the graph, the VDC-derived thresholds are proportional to the voltage-derived thresholds although the PFETs exhibit a slightly greater slope which is due to the greater resolution of the VDC at the higher PFET TGV voltages.

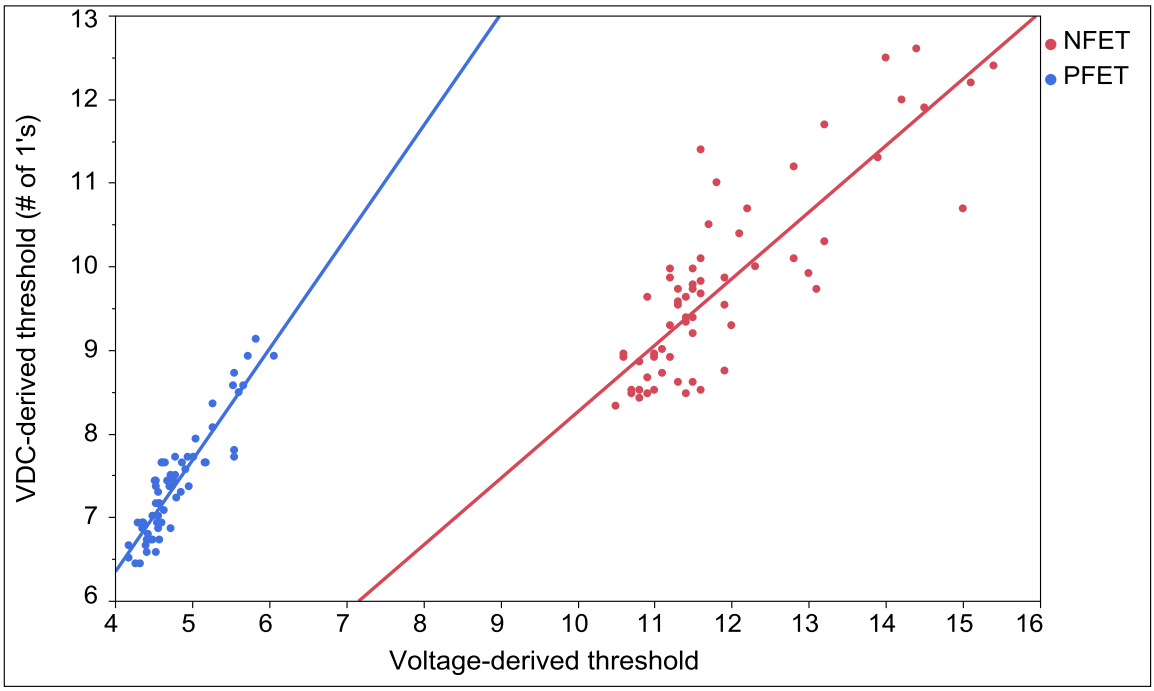


Fig. 29. VDC-derived thresholds versus the voltage-derived thresholds for the TG-PUFs from 63 chips

As explained earlier, the scaling factor is a constant for all chips and its function is to scale the inter-percentile range of each chip to produce the threshold for each chip. This way, each chip has its own unique threshold value that is dependent on its own unique distribution and this strategy renders the number of strong bits consistent from chip to chip irrespective of their distributions. Figs. 30 and 31 illustrate this advantage of the thresholding technique for the VDC-derived and voltage-derived bitstrings respectively. The lack of dependency of the % of strong bits preserved on the threshold value is one of the hallmarks of this technique. Had there been a dependency, that would indicate that chips with too wide or two narrow of a TCD or TGVD distribution would

generate bitstrings with a disproportionately large or small number of strong bits. This would render the number of strong bits preserved inconsistent from chip to chip. As can be seen from Figs. 30 and 31, although there is considerable chip to chip variation in the thresholds, the % of strong bits preserved is within a fairly tight distribution from chip to chip.

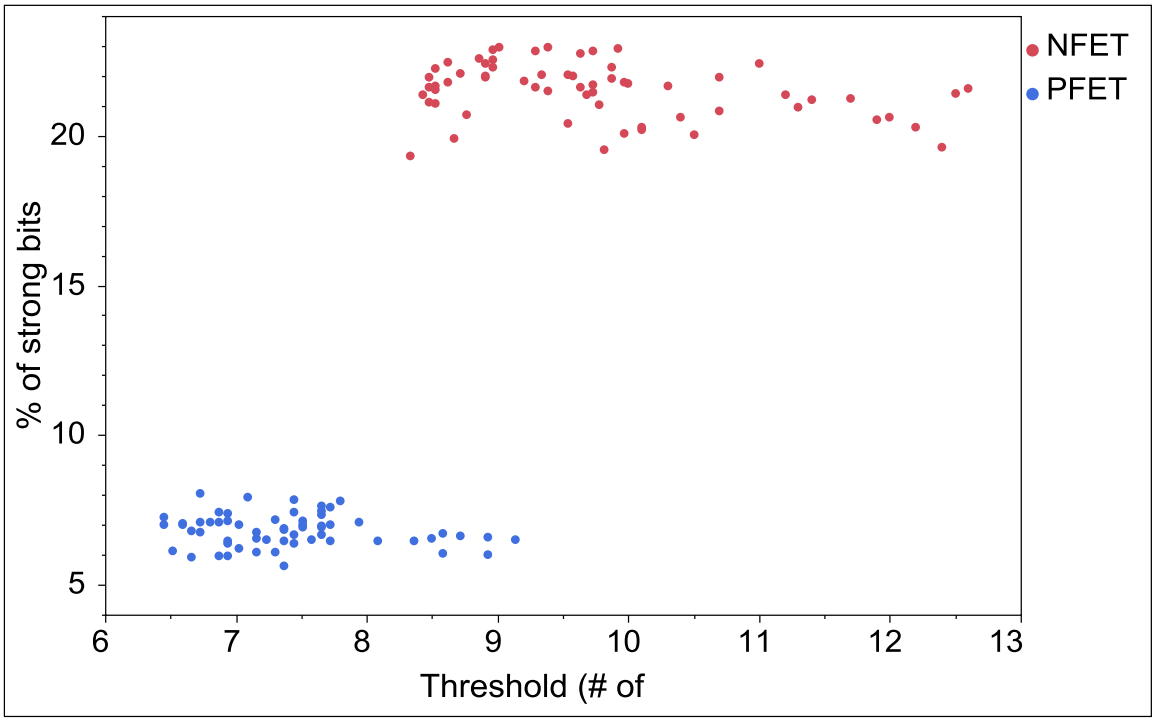


Fig. 30. % of strong bits versus threshold of VDC-derived bitstrings for the TG-PUF from all 63 chips

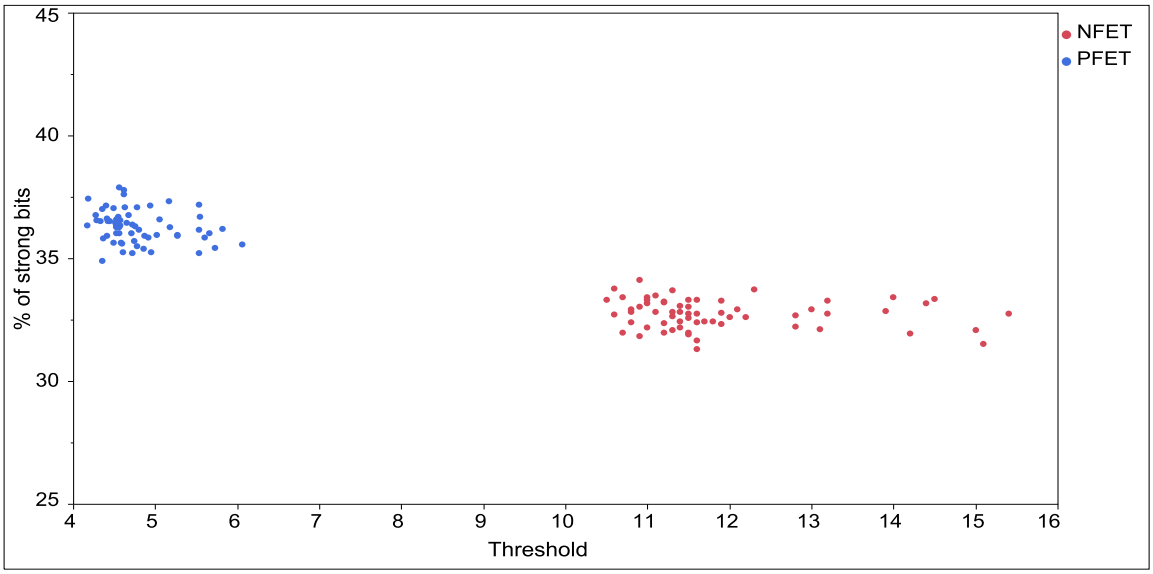


Fig. 31. % of strong bits versus threshold of voltage-derived bitstrings for the TG-PUF from all 63 chips

Figs. 32 and 33 depict the margin and the thresholds for the NFET and PFET VDC-derived bitstrings respectively, of each of the 63 chips tested. The margin is defined as the largest enrollment TCD (for VDC-derived bitstring) or TGVD (for voltage-derived bitstring) difference that exhibits a bit flip at some other TV. A small margin value is desirable as that would mean a smaller scaling factor to insure that difference values above the threshold do not have bit flips and a smaller scaling factor helps with preserving more strong bits. The margin should always be a value between +/- threshold as that will insure that the largest enrollment TCD or TGVD difference that exhibits a bit flip at some other TV is discarded and denoted as a weak bit in the public data. If the margin is a value outside the +/- threshold limits, that indicates that a bit being preserved and denoted as a strong bit exhibits a bit flip at some other TV, and this then defeats the

purpose of thresholding. It should be clear from the figures below that the scaling factor for the PUF (which is a constant for all chips) is decided by the chip that has the margin value closest to the threshold. A smaller scaling factor would cause the thresholding technique to fail on this chip first as the margin would fall outside of the threshold. This does have the tradeoff of causing wasted bits in chips where the margin value is farther away from the threshold as stable bits that do not flip are discarded as they fall within the threshold. But it can be seen from Figs. 32 and 33 that for every chip, the margin falls within the +/- threshold range indicating that the thresholding ensures that all enrollment TCD differences falling outside of the +/- threshold values, defined as strong bits, do not exhibit any bit flips at any TV. Figs. 34 and 35 depict the margins and thresholds of all the chips for the voltage-derived bitstrings.

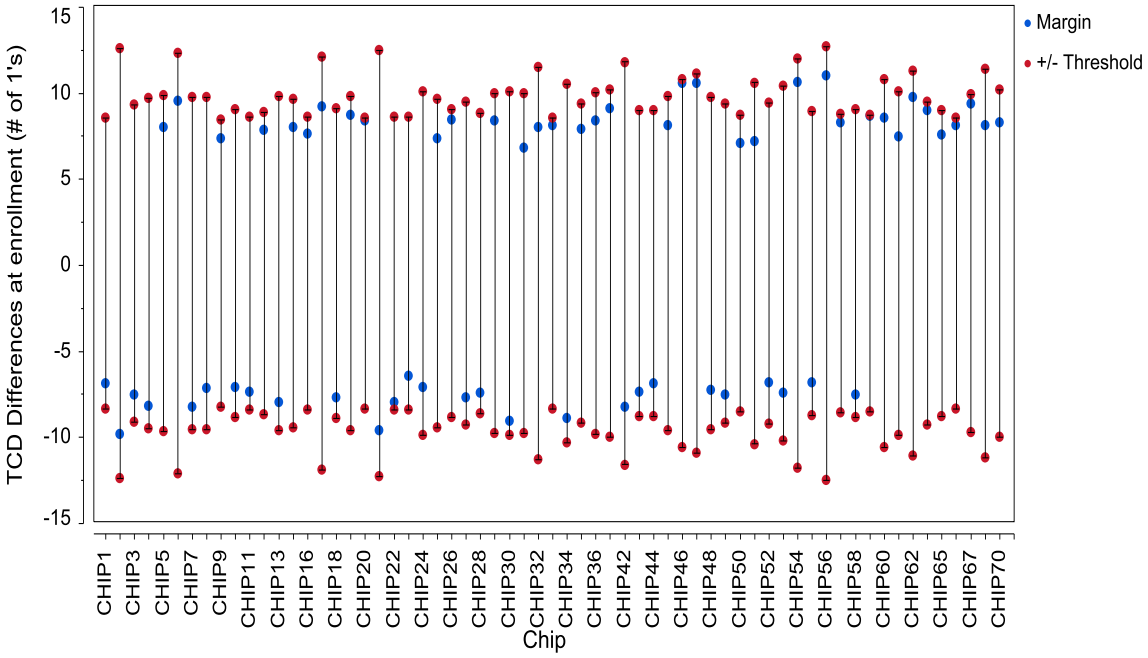


Fig. 32. Threshold and margin of NFET TG-PUF for VDC-derived bitstrings from all 63 chips

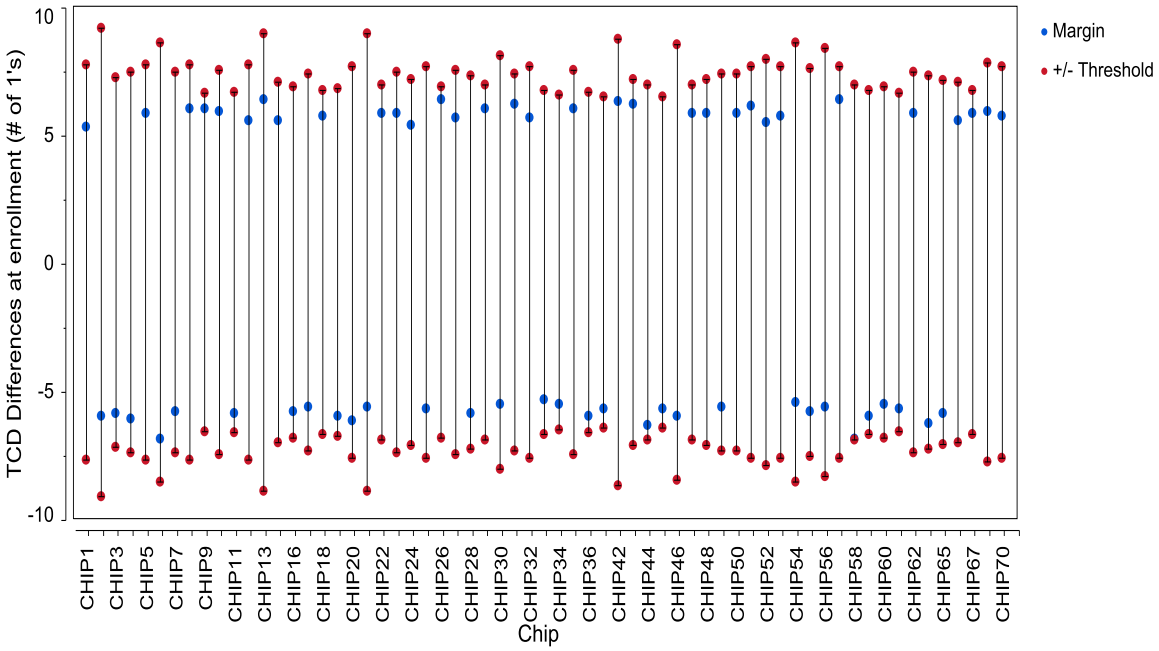


Fig. 33. Threshold and margin of PFET TG-PUF for VDC-derived bitstrings from all 63 chips

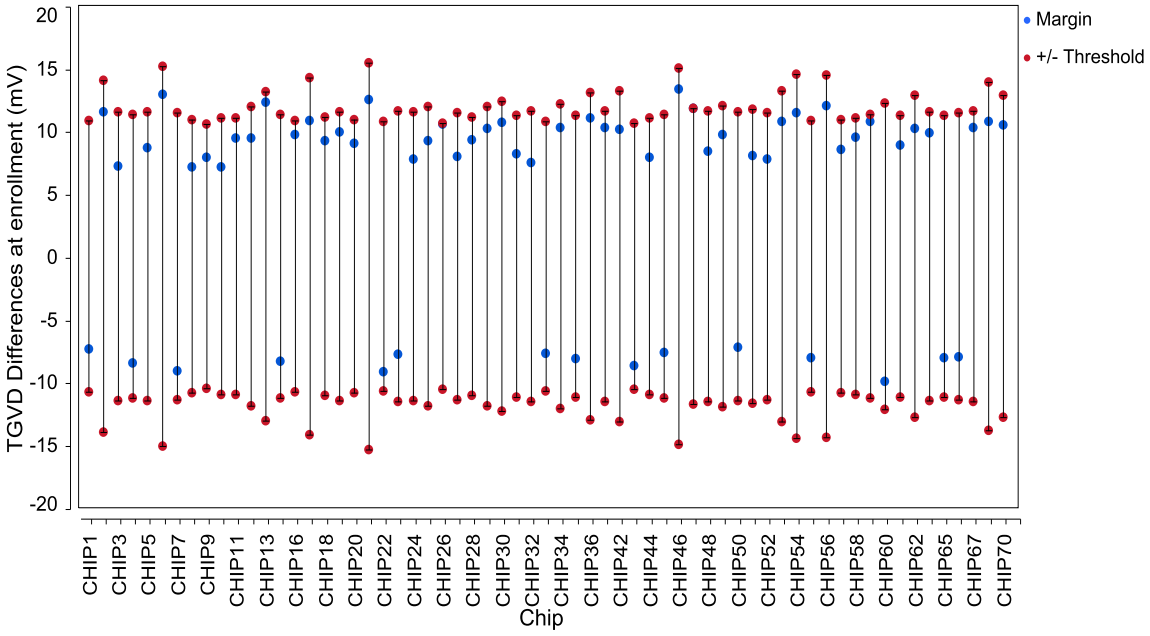


Fig. 34. Threshold and margin of NFET TG-PUF for voltage-derived bitstrings from all 63 chips

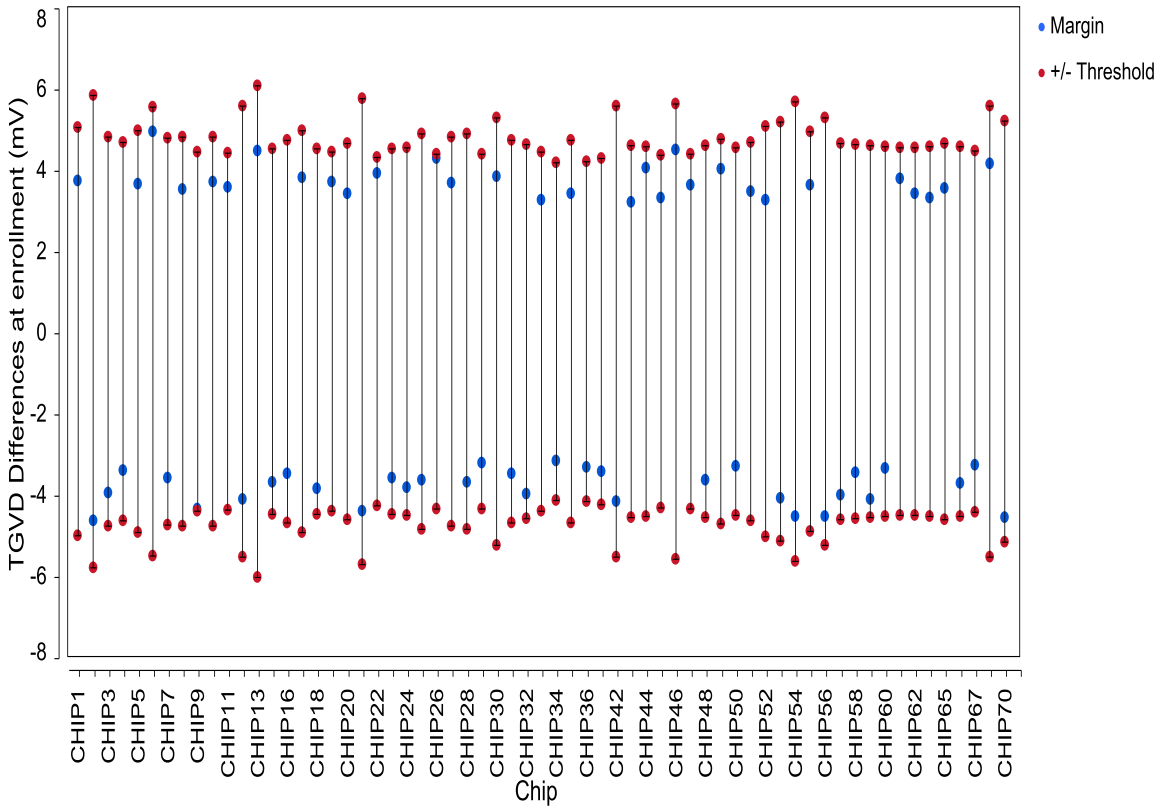


Fig. 35. Threshold and margin of PFET TG-PUF for voltage-derived bitstrings from all 63 chips

Next, it was prudent to investigate the relationship of the margin with the threshold and Figs. 36 and 37 depict that relationship for the VDC-derived and voltage-derived bitstrings respectively. Each datapoint is a chip and it can be seen that a larger margin does translate into a larger threshold to insure no bit flips occur on the enrollment differences above and below the +/- threshold values.

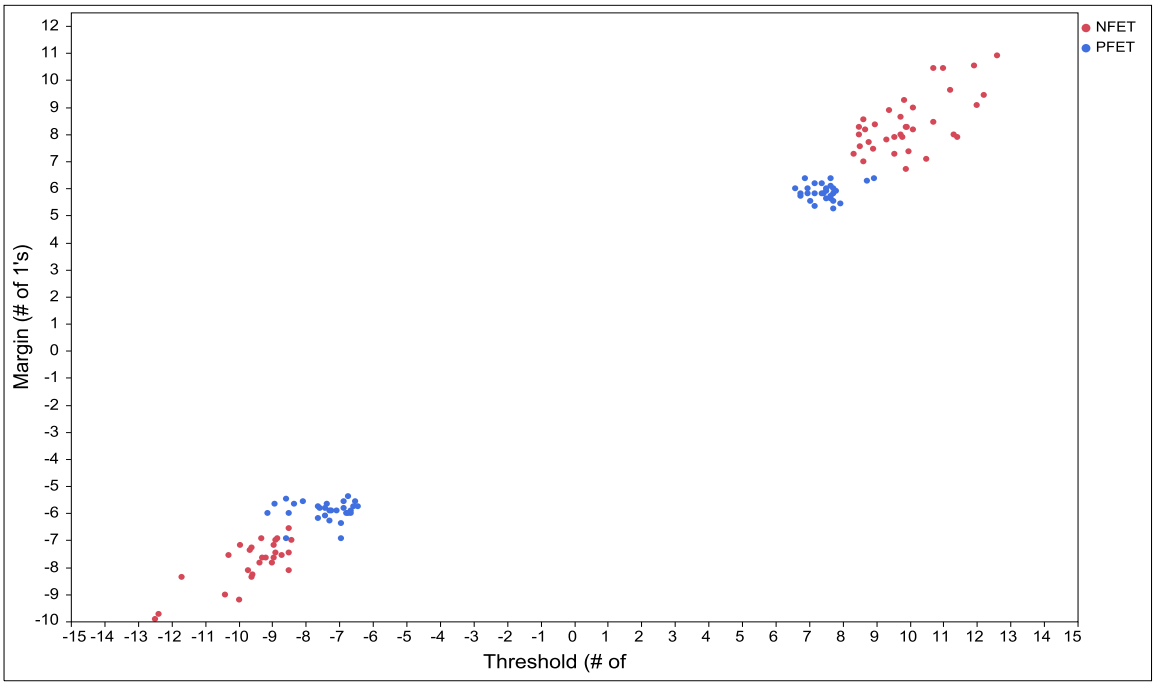


Fig. 36. Margin versus Threshold of TG-PUF for VDC-derived bitstrings from all 63 chips

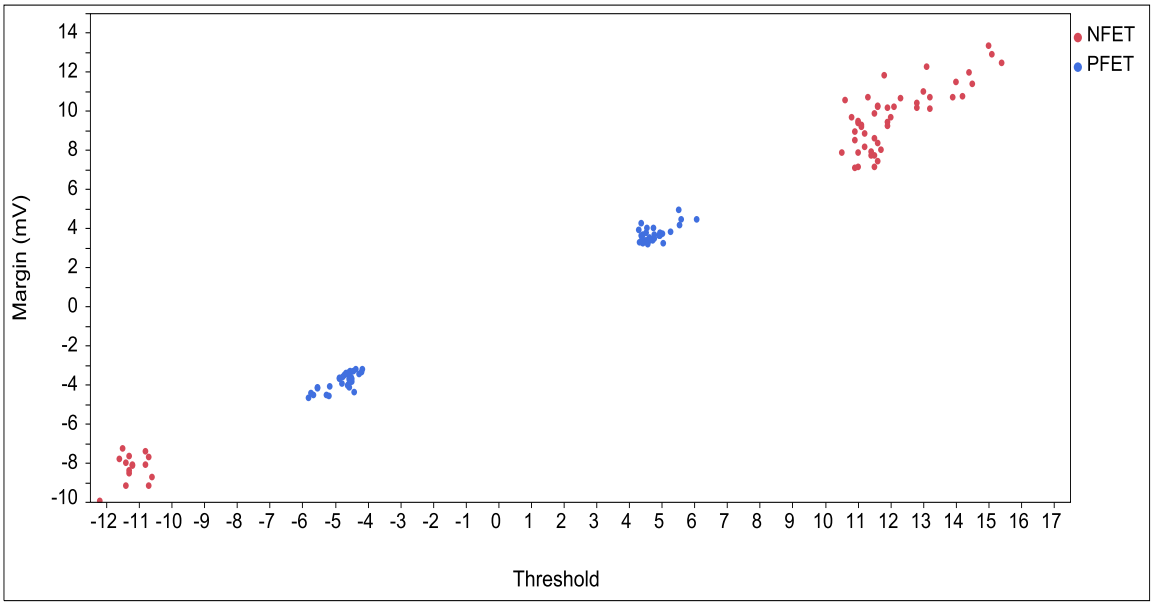


Fig. 37. Margin versus Threshold of TG-PUF for voltage-derived bitstrings from all 63 chips

The cutoff values for the TG-PUF on all the chips are investigated next. The cutoff value of a chip is defined as the smallest TCD (for VDC-derived bitstring) or TGVD (for voltage-derived bitstring) difference at any TV for a comparison identified as a strong bit during enrollment. A value close to 0.0 here is undesirable because it represents how close the strong bit is to flipping at some TV other than enrollment. Figs. 38 and 39 depict the cutoff values for all 63 chips for the VDC-derived and voltage-derived bitstrings respectively. Noteworthy is the fact that the voltage-derived analysis of the PFET cutoffs shows that they fall into a much tighter range and are closer to 0.0 than the NFETs. This is because, for the voltage-derived bitstrings, the threshold of the NFETs are almost double that of the PFETs so the probability of a strong bit getting close to 0 at any TV is lower for the NFETs as compared to the PFETs. The VDC-derived analysis of the cutoff shows that the NFET and PFET cutoffs are in a much comparable range, although the PFETs still do show slightly more points closer to the 0.0 value. Again, this is because of the slightly lower threshold of the PFET as compared to the NFET for the VDC-derived bitstrings. It should be noted that each point in these graphs is one chip and as can be seen from Figs. 38 and 39, the chip 26 NFETs exhibit a cutoff of 0.0 which means that this specific strong bit with the smallest difference in TCDs is right at the edge of flipping.

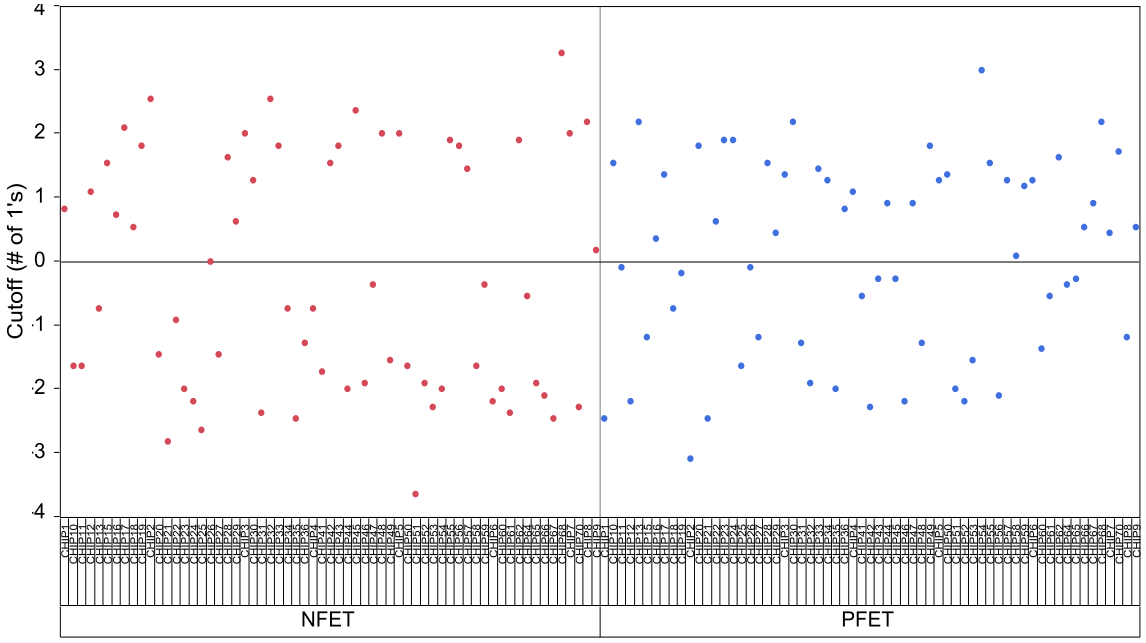


Fig. 38. Cutoff values for VDC-derived bitstrings from TG-PUFs of all 63 chips

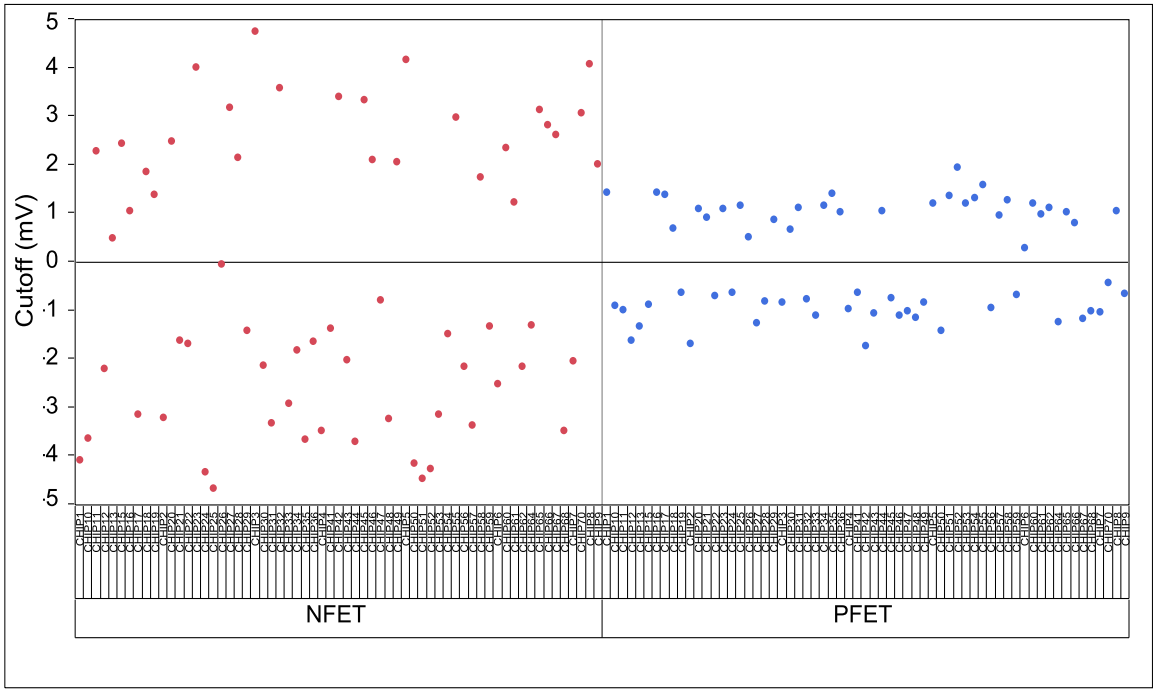


Fig. 39. Cutoff values for voltage-derived bitstrings from TG-PUFs of all 63 chips

Next, the relationship between the cutoff values and the thresholds for each of the chips was investigated via the aid of Figs. 40 and 41. As expected, a weak relationship exists between the two parameters which means unlike as seen for the margins, the cutoff of a chip is weakly proportional to the threshold of the chip.

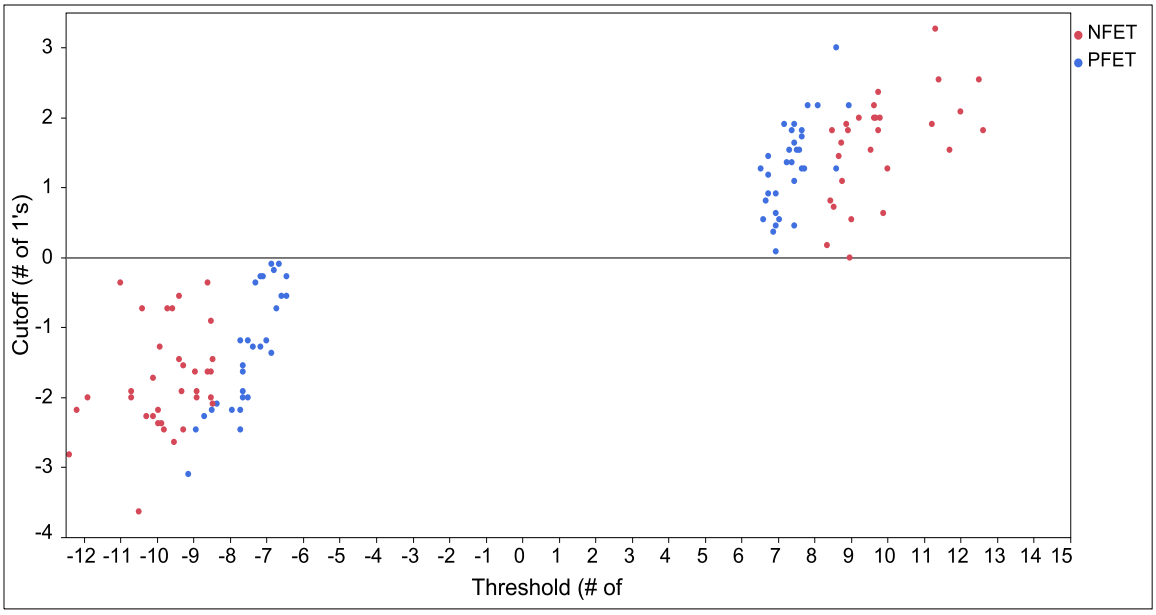


Fig. 40. Cutoff values versus Threshold for VDC-derived bitstrings from TG-PUFs of all 63 chips

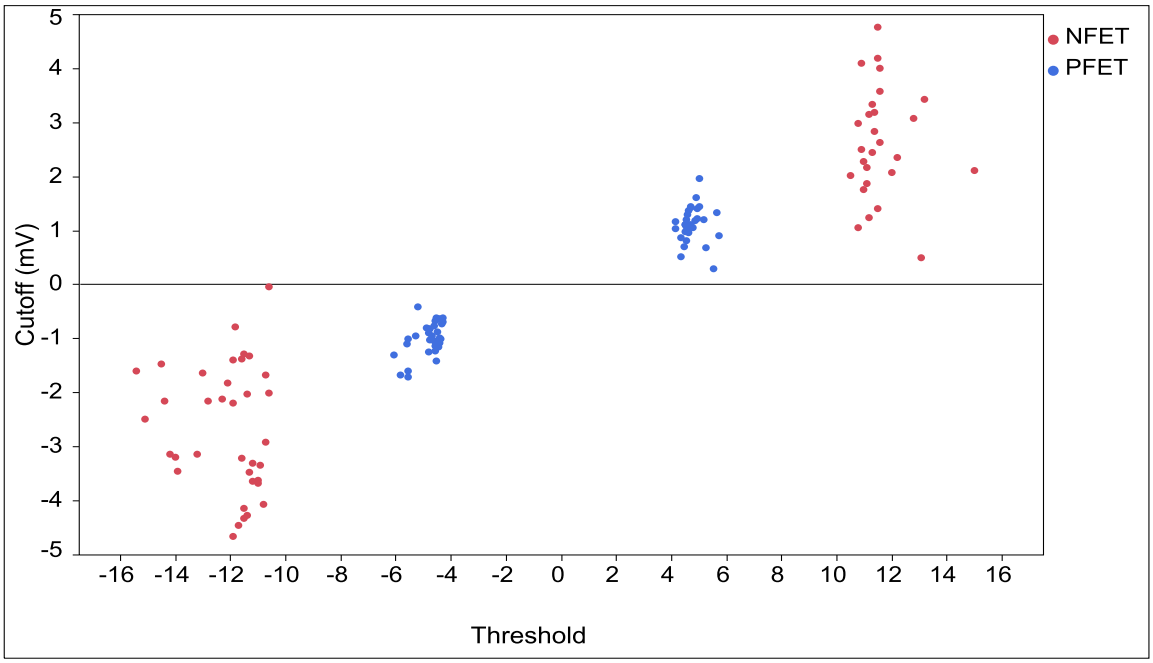


Fig. 41. Cutoff values versus Threshold for voltage-derived bitstrings from TG-PUFs of all 63 chips

5.1.2 Thresholding technique applied to the I-PUF

Similar to the approach taken to applying the thresholding technique to the TG-PUF in section 5.1.1, Fig. 42 illustrates the distribution of the voltage-derived VODs from Chip2 at enrollment conditions (25C, 1,2V). Just as in the TG-PUF, the inter-percentile ranges were defined at the 5% and 95% limits which resulted in a value of 76mV.

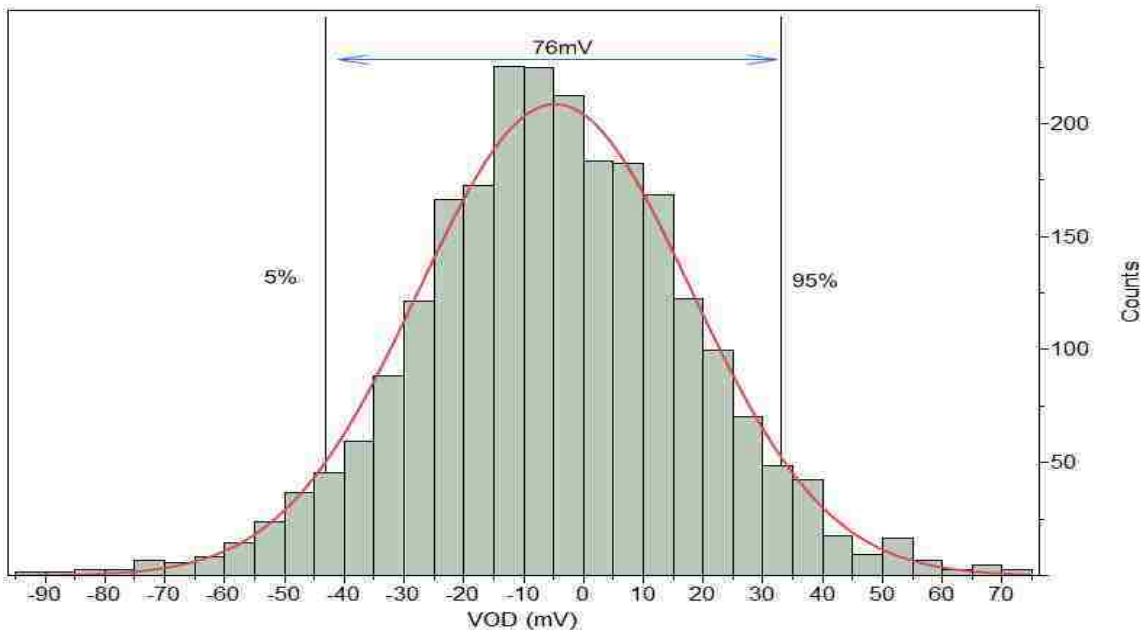


Fig. 42. Enrollment I-PUF VOD distributions with 2,380 components from one chip (Chip 2), with inter-percentile ranges delineated

As can be seen from Fig. 42, the distribution of the VODs is much wider than what was seen in the TG-PUF. This reason for this can be understood by examining Fig. A8 in Appendix A which shows that the distribution of the underlying VO voltages are much wider indicating that the variation in voltages is much larger in the I-PUF.

Next, the scaling factors were derived by analyzing the voltage-derived bitstrings across all 9 TV corners and tuning the values until no bit flips occurred. The scaling factor obtained for the I-PUF was 0.52. We then multiply the inter-percentile range by the scaling factor to define the threshold of +/- 39.5mV for the chip.

Fig. 43 provides an illustration of the thresholding process applied using the voltage-derived VOD data from one of the chips (Chip2) for the I-PUF. The graph plots bit number along the x-axis against the differences of the VODs being compared. Only

the first 5000 bits are shown. The horizontal lines at 39.5mV and -39.5mV delineate the threshold boundaries, which are derived from Fig. 42 using a scaling factor of 0.52. Fig 43 shows those VOD differences which produce strong bits during enrollment. From Fig. 43, the voltage-derived bitstring is generated by defining the VOD differences greater than the positive threshold value as 1's and the differences smaller than the negative threshold value as 0's.

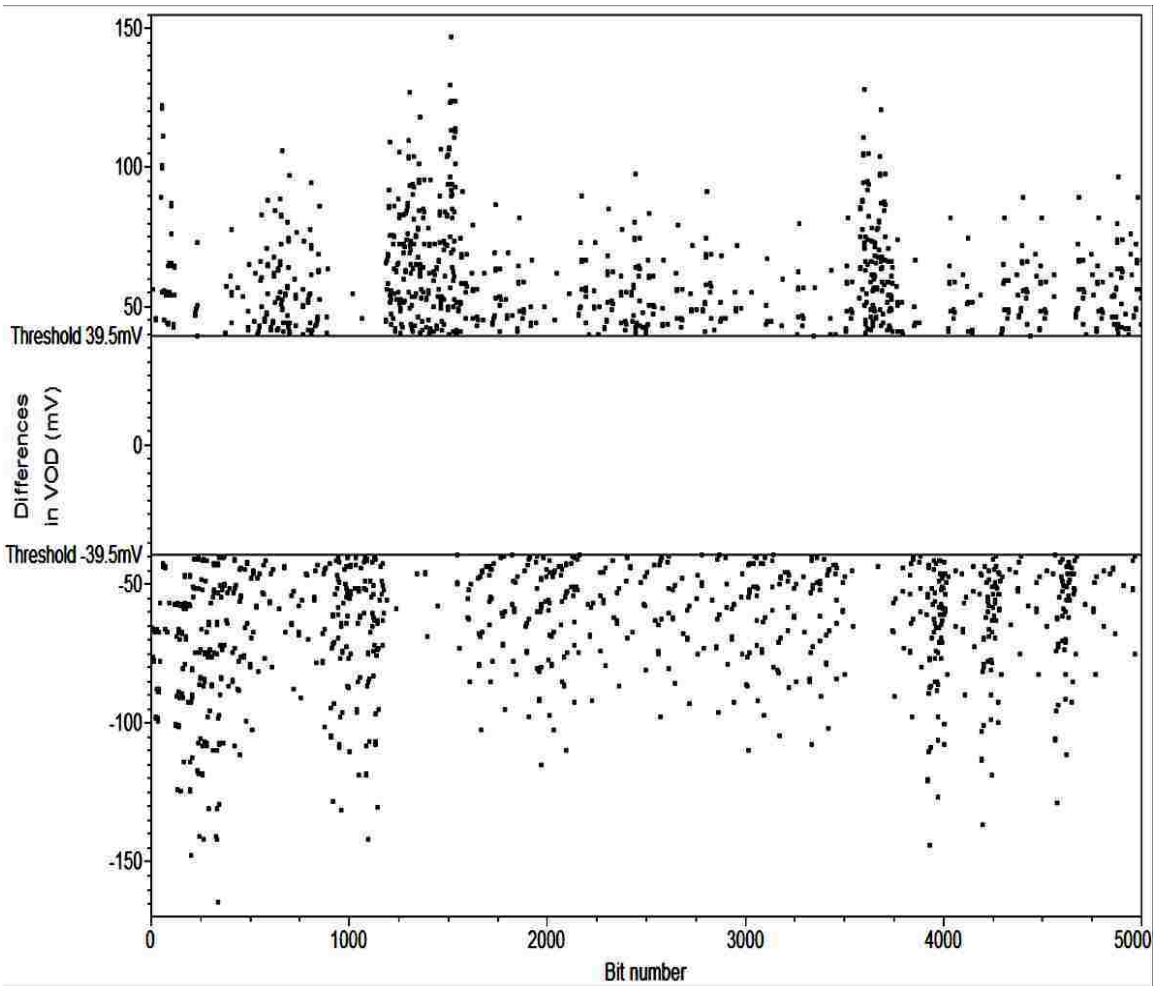


Fig. 43. Threshold method showing the first 5000 (out of 2,831,010) TGVD voltage comparisons during enrollment for the Chip2 I-PUF

Table IV captures the key threshold parameters of the voltage-derived bitstrings from the Chip2 I-PUF while Table V captures that for the voltage-derived bitstrings from all the 62 chips tested.

Table IV: Threshold and length of voltage-derived bitstrings for Chip 2 I-PUF

Voltage-derived bitstrings for Chip2	
Inter-percentile range	76mV
Threshold scaling factor	0.52
Threshold	+/- 39.5mV
% of strong bits	22.18%

Table V: Threshold and range of voltage-derived bitstrings for the I-PUF from all 62 chips

Voltage-derived bitstrings for all 62 chips	
Range of Inter-percentile range	58.78mV (Chip31) – 86.29mV (Chip36)
Threshold scaling factor	0.52
Range of Threshold	+/- 30.56mV (Chip31) to +/- 44.87mV (Chip36)
Range of % of strong bits	20.57% - 23.19%
Truncated bitstring length	582554 (20.5%)

From Table V, it can be seen that there is a considerable spread in the distribution of the VODs of the 62 chips, signified by the range of the inter-percentile ranges. However, it can be seen that the threshold technique does a good job in keeping the preserved bits, represented by the % of strong bits, fairly consistent across the 62 chips. Therefore, this highlights the importance of scaling the threshold of each chip as a

function of its distribution as that is the only way to ensure a consistent level of strong usable bits across all chips.

Upon analysis of the VDC-generated bitstrings for the I-PUF, it was found that the thresholding technique could not be applied to these bitstrings to completely eliminate all bit flips. A constant scaling factor that would eliminate all bit flips on all the chips was unattainable and a scaling factor up to a value of 1.0 was tested and still yielded bit flips on several of the chips. The number of strong bits was also drastically reduced at this scaling factor to the point of being impractical for use. Upon investigation of the raw I-PUF TC data from the VDC, it was discovered that the reason for these results was that numerous TC counts were maxed out at the 120 limit of the VDC and thus, recorded as 120. What that meant is that several TC counts that would be recorded as a finite number greater than 120 were recorded simply as 120 due to the VDC maximum limit being 120. This effectively resulted in erroneously registering a bit flip at a certain TV when in reality, this would likely not be a bit flip had the correct TC count (greater than 120) been recorded. The VDC's maximum limit of 120 bits was not able to handle the large voltage variation of the I-PUF. As stated earlier, considering an approximate resolution of 1bit/mV for the VDC means that the VDC can faithfully digitize about 120mV of voltage variation. However, depending on the TV corner, the I-PUF voltage variation ranges anywhere from 140mV to 256mV (see Fig. A8 in Appendix A) which was well above the capacity of the VDC. On the other hand, the NFET and PFET TG-PUF exhibited a voltage variation in the range of 80mV and 40mV, respectively (see Figs. A4 and A6 in Appendix A), which was well within the 120 bit capacity of the VDC.

Due to the above reasons, the VDC-generated bitstrings were deemed unusable for the I-PUF. The solution to this would be to implement a VDC with greater capacity so that TC counts above 120 can be registered.

The above issue is better visualized by plotting the threshold and margin values of each chip as depicted in Fig. 44 for the VDC-derived bitstring. As explained earlier, the margin should always fall within the +/- threshold limits in order to ensure that all enrollment TCD differences falling outside of the +/- threshold values, defined as strong bits, do not exhibit any bit flips at any TV. As can be seen from Fig. 44, the margins for every chip fall outside the +/- threshold limits indicating that the largest enrollment TCD difference with a bit flip at some TV is being preserved and denoted as a strong bit since it is outside the threshold limits. This is contrary to the goal of thresholding. The solution to this would be to increase the scaling factor to increase the threshold limits, but as described earlier, the tradeoff to this would be the preservation of a fewer number of bits. The data presented in Fig. 44 is using a scaling factor of 1.0 and as can be seen from Fig. 45, the % of bits preserved with a scaling factor of 1.0 is below a dismal 4.5%. Therefore, as described earlier, the large voltage variation in the I-PUF and the inability of our VDC range in handling this limits us to using only the voltage-derived bistrings for the I-PUF.

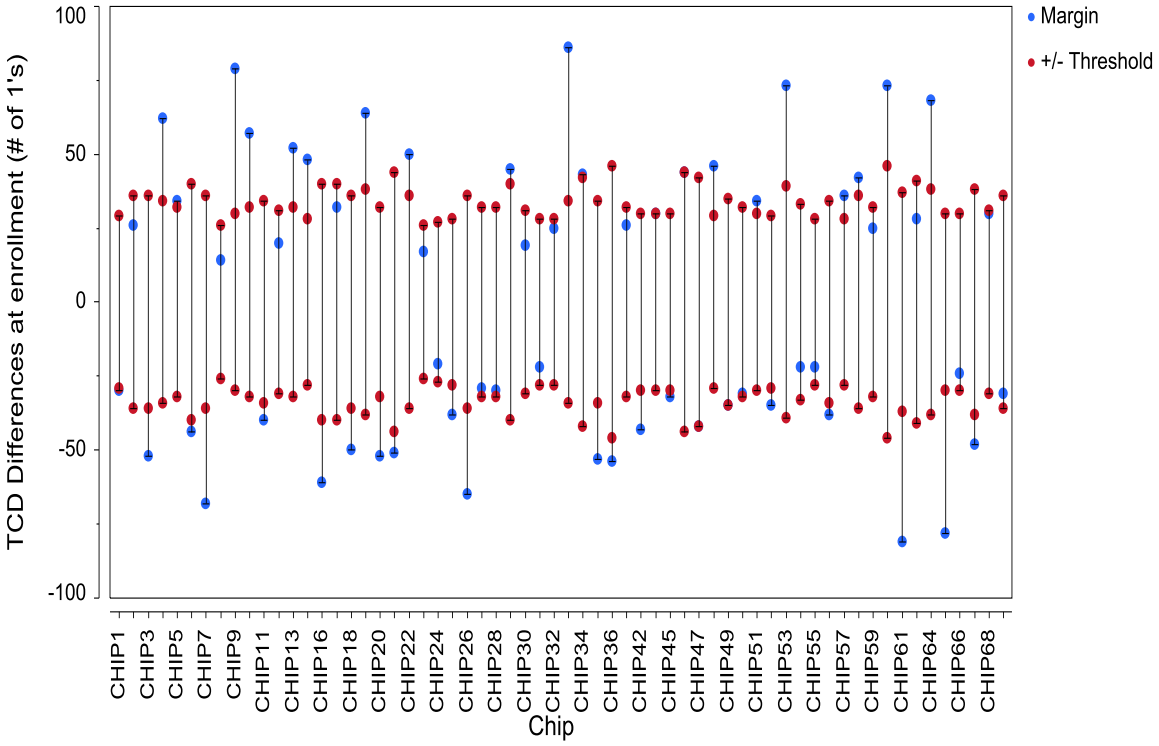


Fig. 44. Threshold and margin of I-PUF for VDC-derived bitstrings from all 62 chips

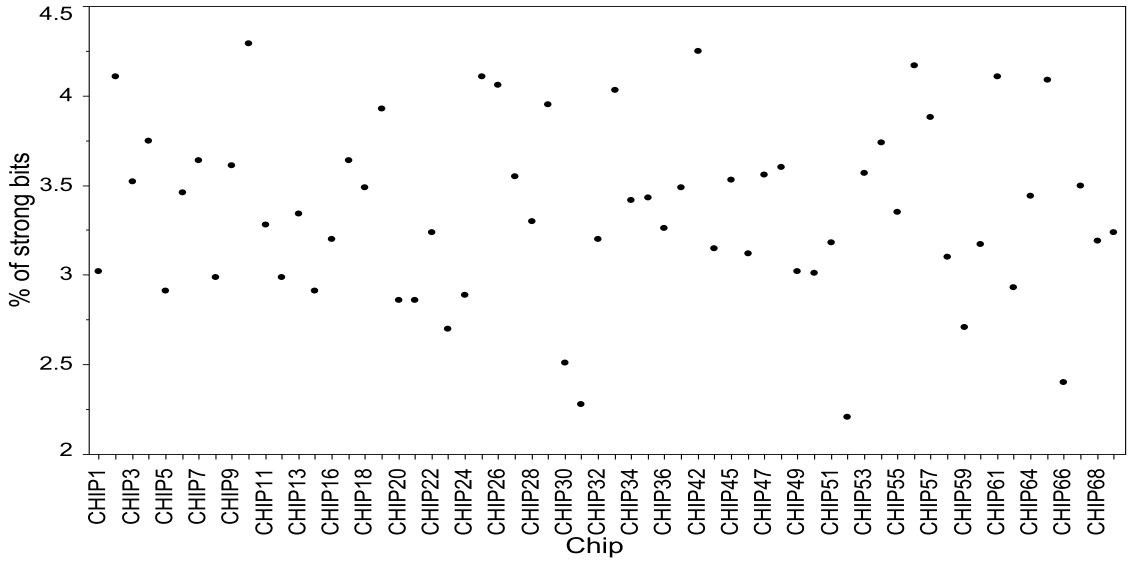


Fig. 45. % of strong bits of VDC-derived bitstrings by chip for the I-PUF from all 62 chips

The next few pages capture the threshold technique results of all the 62 chips tested for the voltage-derived bitstrings. Fig. 46 depicts the behavior of the thresholds and inter-percentile ranges of each of the chips. Each point on this graph is the data from one chip and as expected, the slope of the curve equals the scaling factor.

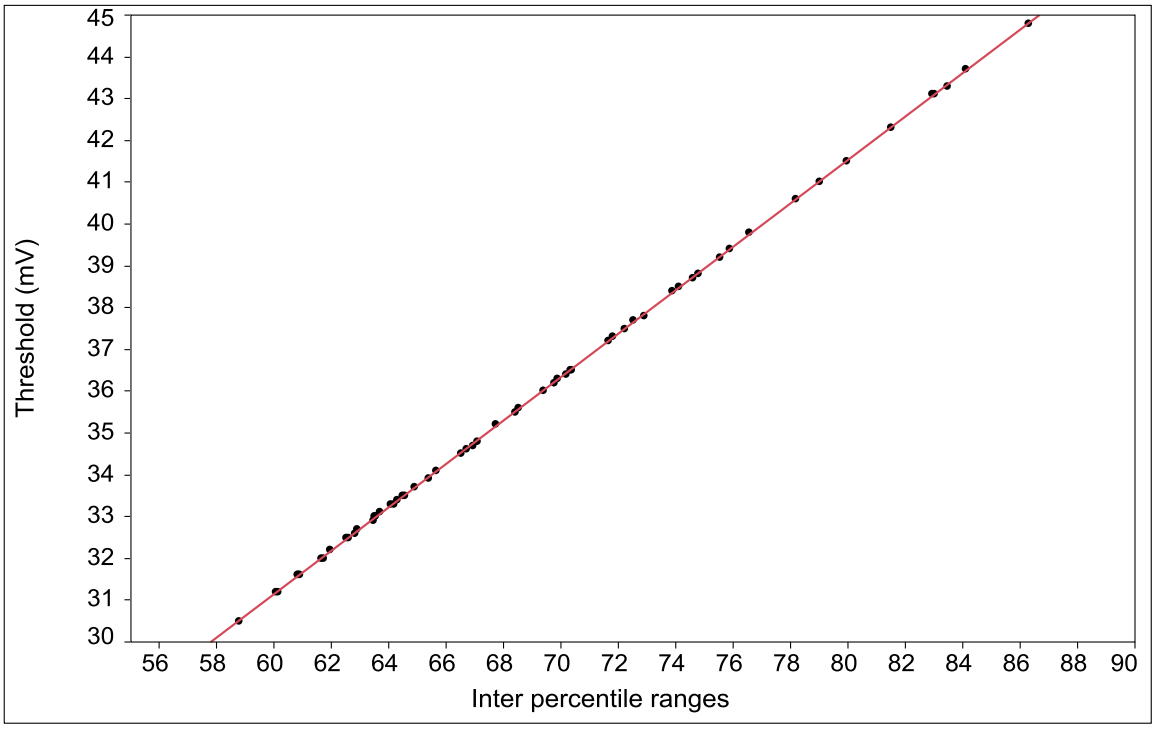


Fig. 46. Threshold versus inter-percentile ranges of voltage-derived bitstrings for the I-PUF from all 62 chips

As explained earlier, the scaling factor is a constant for all chips and its function is to scale the inter-percentile range of each chip to produce the threshold for each chip. This way, each chip has its own unique threshold value that is dependent on its own unique distribution and this strategy renders the number of strong bits consistent from

chip to chip irrespective of their distributions. Fig. 47 illustrates this advantage of the thresholding technique for the voltage-derived bitstrings. The observed lack of dependency of the % of strong bits preserved on the threshold value is one of the hallmarks of this technique.

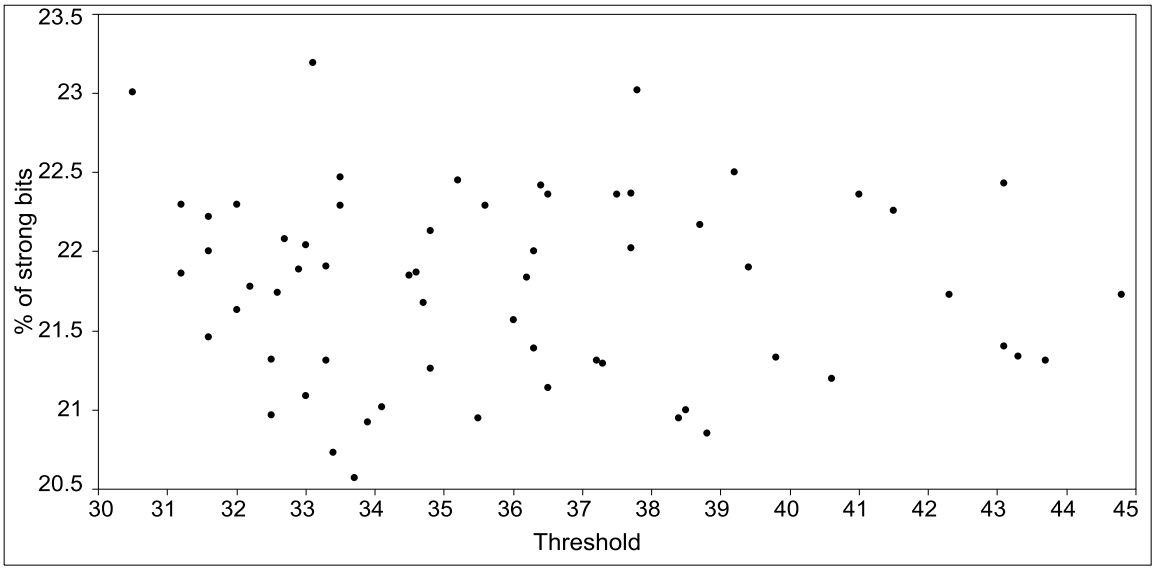


Fig. 47. % of strong bits versus threshold of voltage-derived bitstrings for the I-PUF from all 62 chips

As stated earlier, the margin should fall within the limits of +/- threshold and as can be seen from Fig. 48, this is the case for the voltage-derived bitstrings for all the chips.

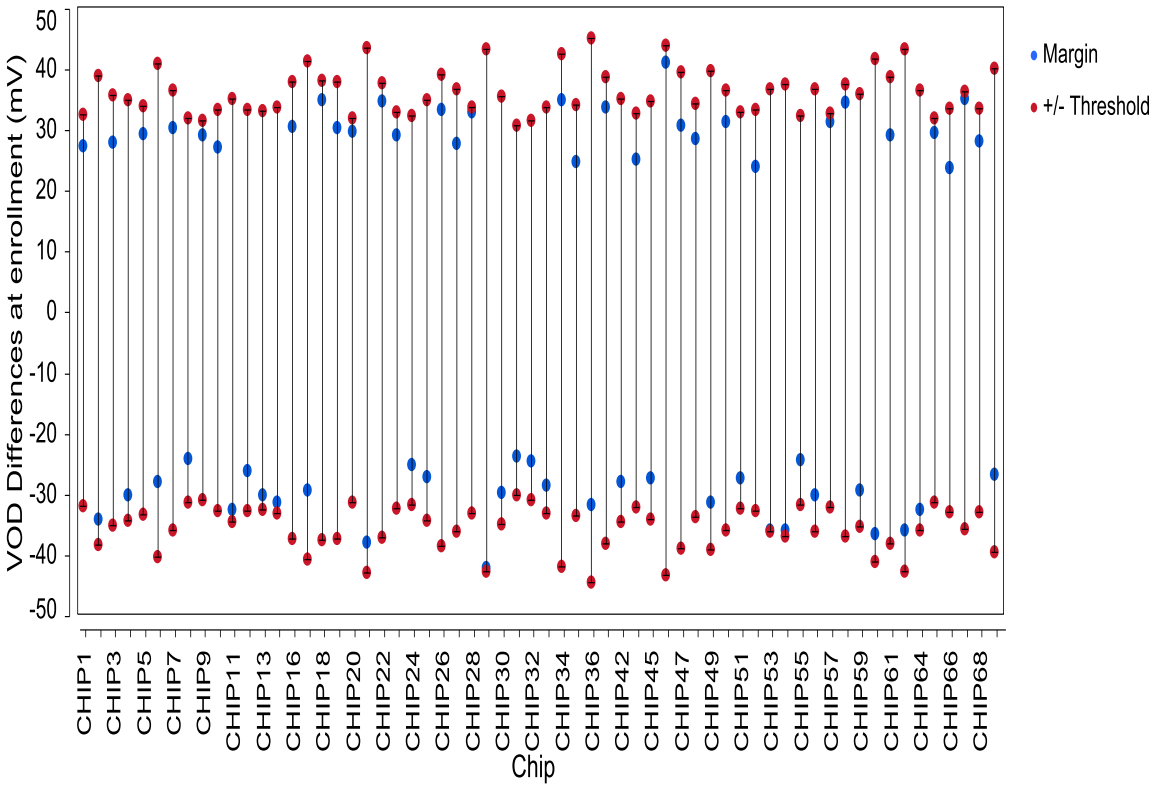


Fig. 48. Threshold and margin of I-PUF for voltage-derived bitstrings from all 62 chips

The relationship between the margin and threshold is depicted in Fig. 49 for all the chips and as seen in the case of the TG-PUF as well, there is a distinct relationship between the two parameters. As the margin gets more positive, so does the threshold.

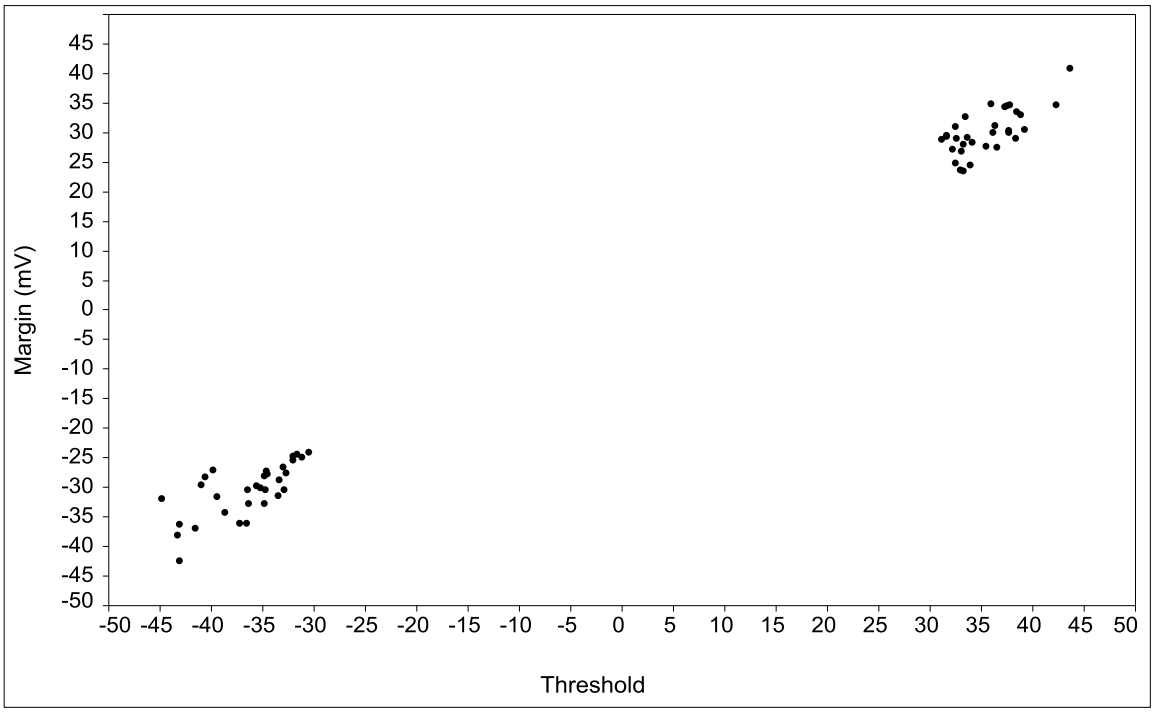


Fig. 49. Margin versus Threshold of I-PUF for voltage-derived bitstrings from all 62 chips

Lastly, the cutoff of all the chips is plotted in Fig. 50 and it is evident that none of the chips exhibit a cutoff of 0.0 indicating that the strong bits of the chips are not close to flipping. From Fig. 51, as expected, there is only a weak relationship between the cutoff and the threshold for the chip population tested.

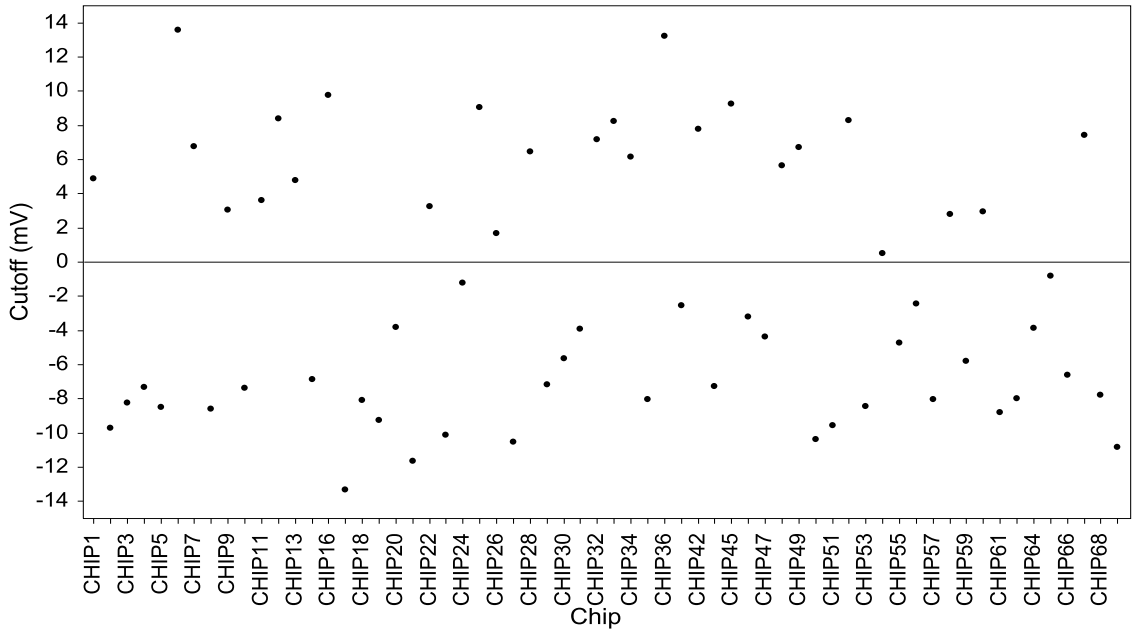


Fig. 50. Cutoff values for voltage-derived bitstrings from I-PUFs of all 62 chips

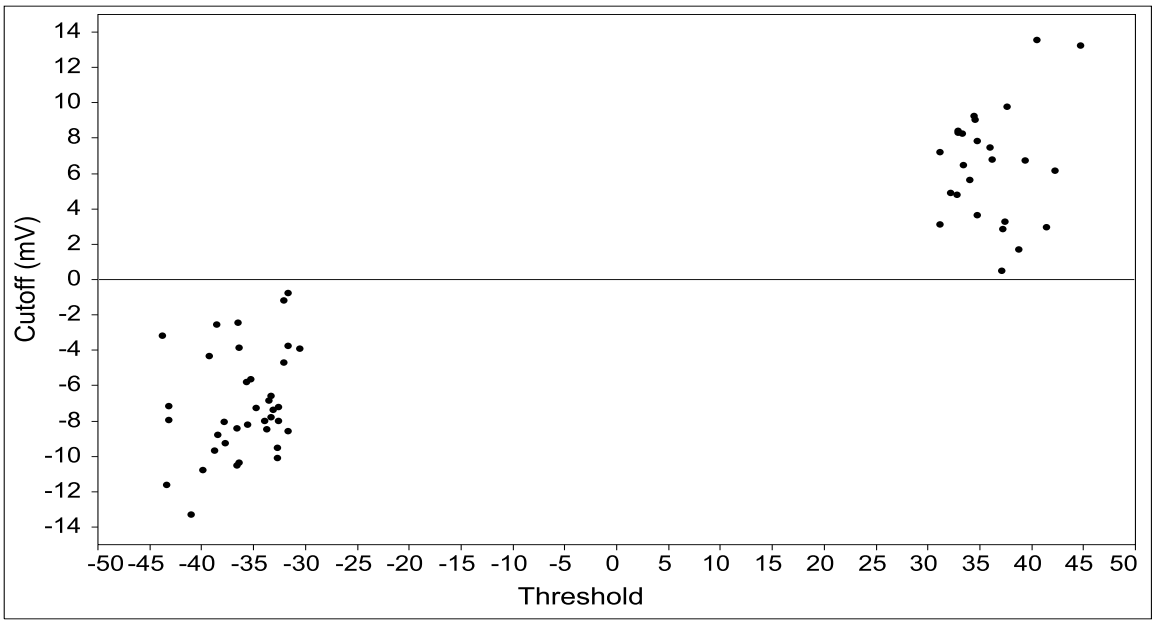


Fig. 51. Cutoff values versus Threshold for voltage-derived bitstrings from I-PUFs of all 62 chips

5.2 Fixed Length Bitstrings and TMR

In actual applications, only a fixed number of bits are needed. With encryption, the values vary between 128 to 1024 bits, depending on the encryption algorithm. The large number of bits available from the PUF can be beneficial, however, by allowing a distinct set of fixed-length secret keys to be generated over time during successive enrollments.

A second possible usage scenario leverages this large pool of strong bits to further increase the resiliency to bit flip failures, i.e., beyond that provided by thresholding. A bitstring replication method is proposed that mimics a popular scheme used in fault tolerance called triple-module-redundancy or TMR. In this technique, a fixed length, e.g., 1,024-bit, bitstring is generated as described above using the thresholding technique. TMR is then applied to generate two more copies of the bitstring. The two copies are generated by parsing the strong bit sequence until a match is found to each bit in the first bitstring.

During regeneration, a majority voting scheme is applied to each of the columns in the three identically regenerated bitstrings as a means of avoiding single bit flip failures. In other words, the final bitstring is constructed by using the majority of the 3 column bits as the final bit for each bit position, i.e., a ‘1’ is assigned in the final bitstring when 2 or more of the 3 bits in the column are ‘1’, and a ‘0’ otherwise.

Fig. 52 illustrates the proposed thresholding and TMR-based scheme using data from a hypothetical chip. The x-axis plots a sequence of comparisons that would be used to generate a bitstring, while the y-axis plots the differences between the pairings of

TCDs. Each difference reflects the relative ordering of the two TCDs, e.g., positive difference values indicate that the first TCD is larger than the second. For strong bits, the TCD difference data points must lie above or below the thresholds, labeled '+Tr' and '-Tr' in the figure. This condition, when met, is recorded using a '1' in the thresholding bitstring shown below the data points. Weak bits, on the other hand fall within the thresholds and are indicated with a '0'. The bold (and blue) '0's indicate strong bits that are skipped under the TMR scheme.

The TMR-based method constructs 3 identical bitstrings during enrollment as shown along the bottom of Fig. 52. The left-most bitstring labeled 'Secret BS' is generated from the first 4 strong bits encountered as the sequence of data points is parsed from left to right. The second bitstring labeled 'Redundant BS_1' is produced from the next sequence of data points but has the additional constraint that each of its bits must match those in the first bitstring. During its construction, it may happen in the continued left-to-right parsing of the data points that a strong bit is encountered that does not match the corresponding position in the 'Secret BS'. In the example, this occurs at the position indicated by the left-most bold '0' in the thresholding bitstring. Here, we encountered a strong bit with a value of '0'. But the 'Secret BS' requires the first bit to be a '1', so this strong bit is skipped. This process continues until redundant bitstrings BS_1 and BS_2 bitstrings are constructed.

A PUF that is able to generate strong bit sequences that are locally random (a quality measured by the NIST tests [1]) ensures that a match occurs for each bit during the generation of the two copies every 2 bits on average. Under these conditions, it

follows that a TMR-based bitstring, and its public data, consumes on average 5 times more strong bits than a non-TMR-based bitstring. The benefit, on the other hand, is a significant decrease in the 'probability of failure', i.e., the likelihood of a bit flip occurring during regeneration. Moreover, this scheme offers flexibility by allowing a trade-off between tolerance to bit flips and public data size. Therefore, the number of strong bits required to generate a secret bitstring of length 4 is approx 5x or 20. From the example, this is evaluated by counting the number of '1's and bolded '0's in the thresholding bitstring, which is given as 19. The benefit of creating these redundant bitstrings is the improved tolerance that they provide to bit flips. For example, during regeneration, the three bitstrings are again produced, but this time using the thresholding bitstring to determine which TCDs to compare.

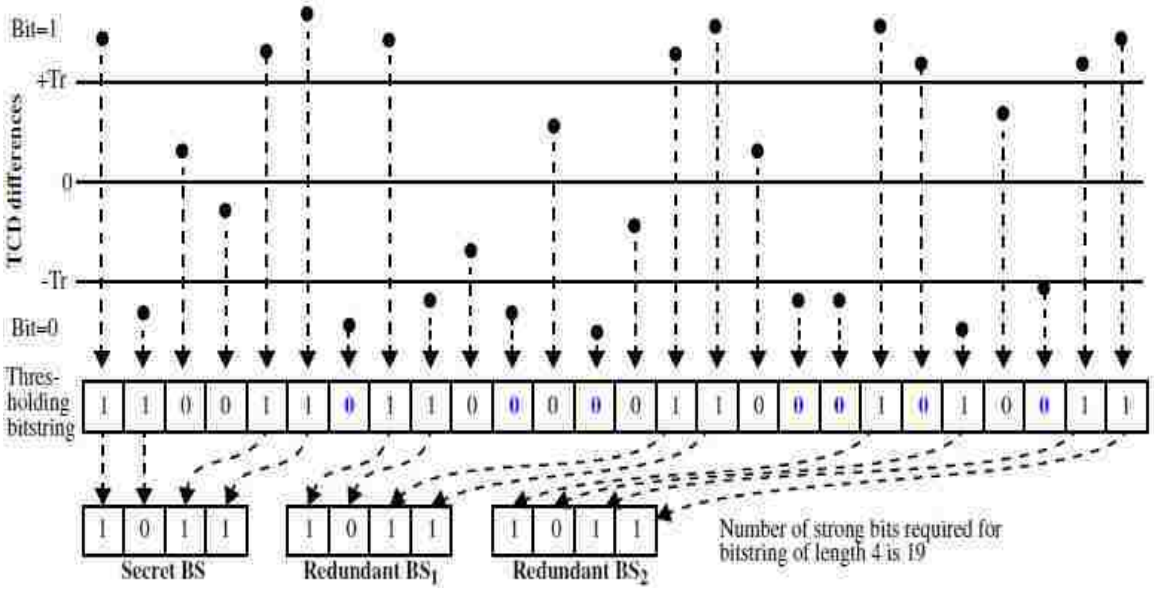


Fig. 52. Secret bitstring generation example using the proposed thresholding and TMR-based method

In scenarios where the threshold is set too low, it is possible that a strong data point used in enrollment is displaced across both the threshold and the '0' line because of different TV conditions in regeneration, causing a bit flip. However, with TMR, a bit flip can be avoided if no more than 1 bit flip occurs in a single column of the matrix of bits created from the 3 bitstrings. For example, the first 3 rows of the matrix of bits in Fig. 53 are constructed during regeneration in a similar way to those shown in Fig. 52 for enrollment. The bottom row represents the final secret bitstring and is constructed by using a majority vote scheme (in the spirit of TMR). The bit flip shown in the third column has no effect on the final bitstring because the other two bits in that column are '1', and under the rule of majority voting, the final secret bit is therefore defined as '1'.

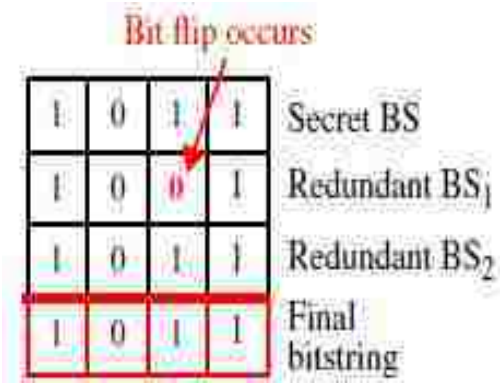


Fig. 53. Bit flip avoidance illustration using example from Fig. 52

5.3 Probability of failure

As discussed previously, the TMR scheme improves resiliency to bit flips over the thresholding scheme alone. The curves shown in Fig. 54 illustrate the improvement for the TG-PUF VDC-derived bitstrings. The scaling factor used for NFETs (the PFET

scaling factor is also changed proportionally) is plotted along the x-axis against the probability of failure on the y-axis. The probability of failure is computed at each scaling factor value by dividing the number of bit flips that occur in all 63 chips by the total number of strong bits produced. The curve on the left is the result obtained using the TMR + thresholding technique, while the curve on the right uses only thresholding. Both curves are exponential in shape. From the positions of the curves, it is clear that the TMR scheme requires a lower scaling factor, 0.34 vs. 0.53, before any bit flips occur. Using 0.53 as the scaling factor, the probability of failure is 1.1e-6 with thresholding but improves significantly to 1.5e-12 after adding TMR. These values were obtained by fitting the discrete-valued curves produced from repeatedly running the analysis at different scaling factors with exponential functions.

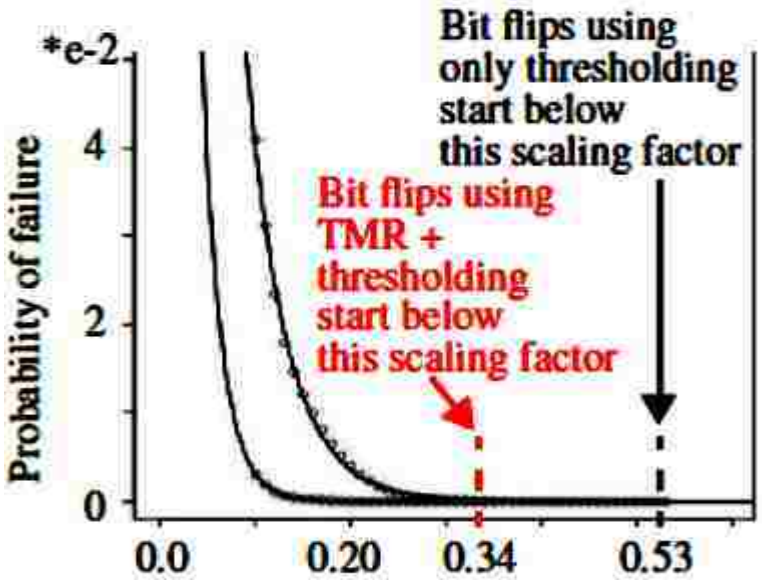


Fig. 54. TG-PUF NFET TCD scaling factor (x-axis) vs. probability of failure (y-axis)

Fig. 55(a) shows the data for the TMR + thresholding curve in Fig. 54 with the fitted exponential curve. The exponential is clearly a good fit to the data points. Fig. 55(b) shows a blow-up of the region around the NFET scaling factor of 0.53 from which the estimate of 1.5e-12 was derived.

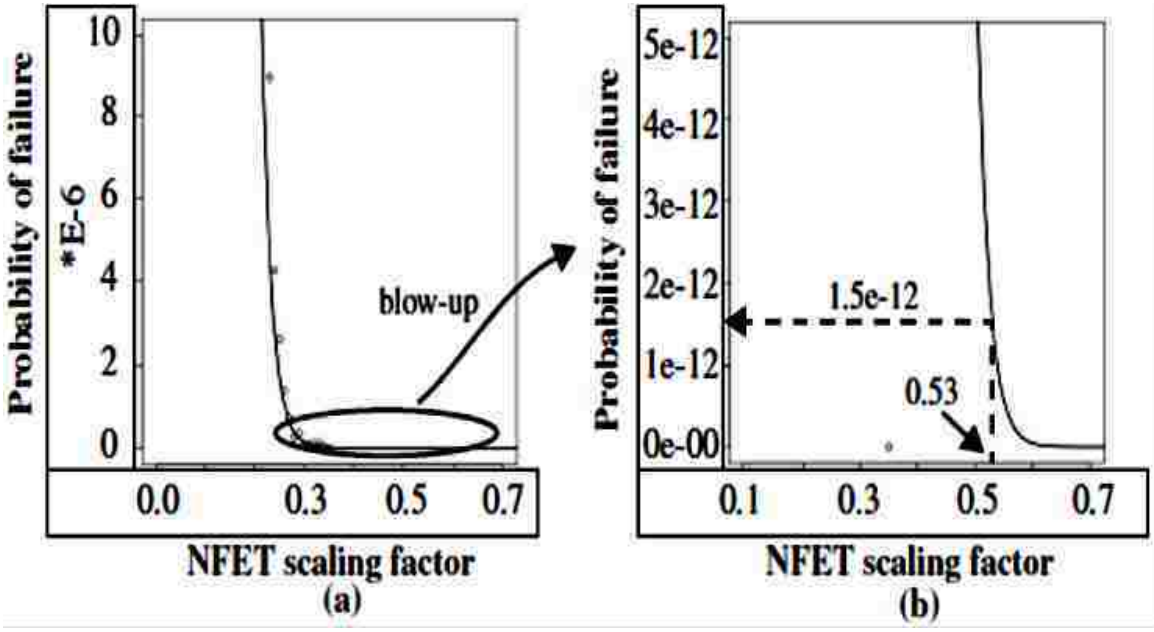


Fig. 55. (a) TMR probability of error curve and (b) blow-up of the designated region. The discrete curve is fitted with a superimposed exponential function.

Performing a similar analysis to assess the benefits of a TMR+threshold technique versus just a threshold technique for the voltage-derived bitstrings of the I-PUF results in the data graphed in Fig. 56. As seen before, the y-axis of the graph represents the probability of failure while the x-axis represents the scaling factor. The probability of failure is computed at each scaling factor value by dividing the number of bit flips that

occur in all 62 chips by the total number of strong bits produced. The curve on the left is the result obtained using the TMR + thresholding technique, while the curve on the right uses only thresholding. Both curves are exponential in shape. From the positions of the curves, it is clear that the TMR scheme requires a lower scaling factor, 0.29 vs. 0.51, before any bit flips occur. Using 0.51 as the scaling factor, the probability of failure is 5.03e-8 with thresholding. Using a scaling factor of 0.29, the probability of failure using thresholding jumps to 12.84e-5 but with the TMR+Thresholding technique drops to 5.9e-8, which is at the level observed by just the threshold technique although at a larger scaling factor of 0.51. As mentioned earlier, a smaller scaling factor is desired as it allows for preservation of more strong bits.

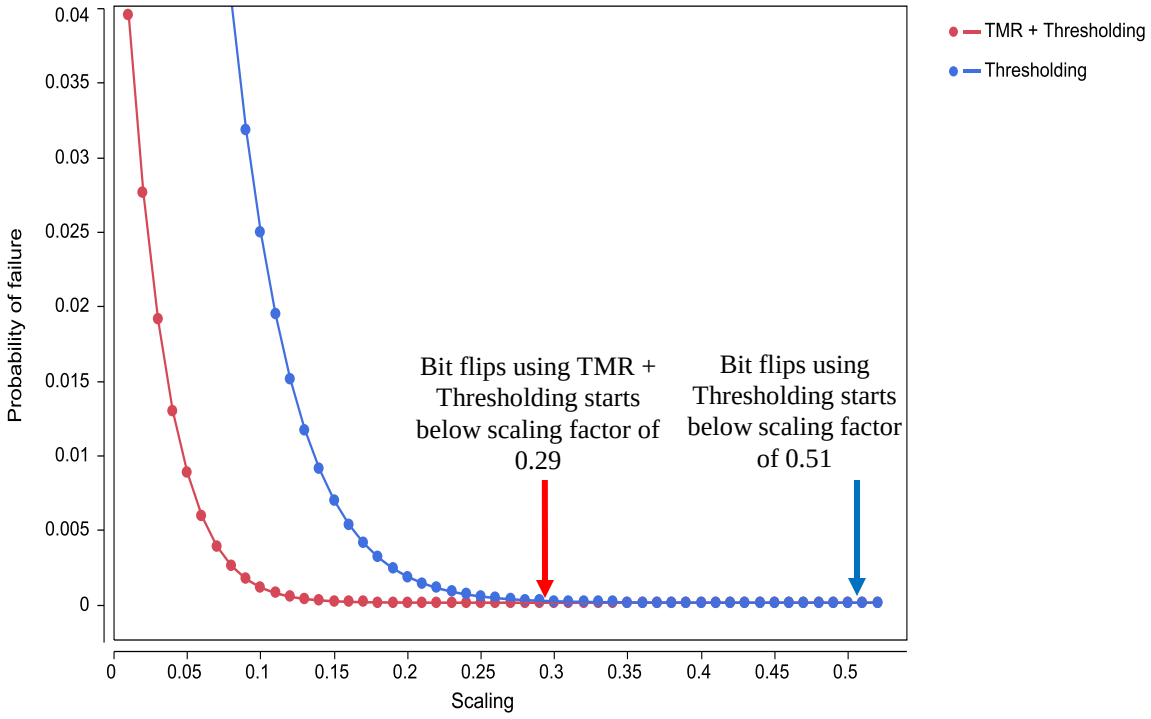


Fig. 56. Probability of error curves for TMR+Threshold and only Threshold techniques applied to the voltage-derived bitstrings from the I-PUF

5.4 Run-Length Encoding of Public Data

The size of the public (helper) data under the thresholding and TMR-based schemes can be reduced using compression techniques such as run-length encoding. The benefit of run-length encoding is its simplicity. Fig. 57 shows an example of a thresholding bitstring with 26 bits. The long strings of '0's can be run-length encoded by simply counting them and replacing the '0' sequence with a field which represents the number of '0's in each sequence. In the example, the run-length encoded bitstring uses 19 bits instead of 26. The longer the sequences of '0's, the more efficient the scheme becomes. The best choice for the field width depends on the nature of the public data, i.e., the average length of the '0' strings.

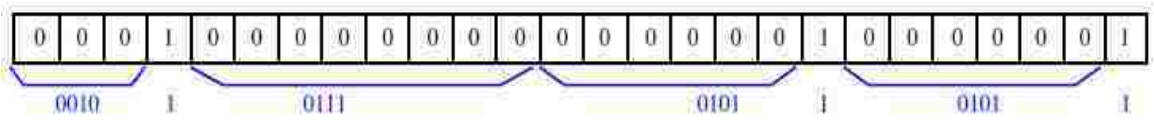


Fig. 57. Examples of run-length encoding as a compression technique to reduce public data size. Original public data string has 26 bits. Run-length encoded using a field width of 4 yields 19 bits

The public data for the TCD analysis of the TG-PUF indicates that approx. 14% of the bits survive the thresholding, and even fewer, approx. 8.4%, are marked with '1's in the public data when TMR is added. The public data is therefore expected to contain strings of 0's with average lengths of approx. 11 under thresholding + TMR. Therefore, a field

width between 3 and 4 (which allows counting up to 8 and 16, resp.) should be optimal. It was found that a field width of 5 is best and yields a 42% reduction on average to the size of the original public data string. The plan is also to explore other compression techniques in future work.

It should be noted that in addition to compression, obfuscation is required for the thresholding bitstring when the PUF usage scenario involves authentication. This is true because the 'secret' bitstring is not kept on chip as it is for encryption but rather is also made public. With both bitstrings available, an adversary can reverse engineer the relative ordering of the TCDs. In order to prevent this, we propose to obfuscate a portion of the thresholding bitstring as follows. During enrollment, the first *n* strong bits, e.g., 128, are used as a key to encrypt the thresholding bitstring, excluding those public data bits that correspond to the encryption key itself. These bits do not need to be encrypted because the key is never made public.

Chapter 6

Statistical Characterization of Bitstrings

6.1 TG-PUF Bitstrings

In this section, we evaluate the several important statistical properties of the voltage-derived (TGVD) and VDC-derived (TCD) bitstrings including randomness, uniqueness and probability of bit flips, e.g., failures to regenerate the bitstring under different environmental conditions. The TGVD analysis is carried out on bitstrings generated from digitized voltages obtained from an off-chip voltmeter (no VDC involvement), while the TCD analysis is carried out on bitstrings generated by the PUF after digitizing the voltages using an on-chip VDC that is subjected to the same TV corners as the PUF itself. As discussed earlier, the process of digitizing the voltages using the VDC adds noise and reduces the number of corresponding strong bits. The penalty of the digitization process is evaluated by carrying out the same analysis using the TGVDs directly, and serves to illustrate the best that can be achieved in the absence of digitization noise.

Fig. 58(a) gives the inter-chip hamming distance (HD) distribution using the TGVDs while Fig. 58(b) shows the distribution using TCDs for the bitstrings after thresholding was applied for all chips. The graphs plot HD along the x-axis against the number of instances on the y-axis. With 63 chips, the total number of instances is 63*62/2 = 1,953. The distributions are ‘fitted’ with Gaussian curves to illustrate the level of conformity they exhibit to this distribution.

Since HDs must be computed across bitstrings of equal length, it was necessary to truncate the bitstrings used in Fig. 58 to the length obtained for the chip with the fewest number of strong bits. This defines the length of all bit strings for the purposes of the HD analysis. Truncation reduced the lengths to 1,901,845 for the TGVD analysis and 725,230 for the TCD analysis, which are approx. 33.6% and 12.8%, resp., of the maximum possible length, i.e., 5,662,020 bits. The chip with the longest bitstring, in comparison, uses 35.6% of the maximum for the TGVD analysis and 15.0% for the TCD analysis. The term truncated bitstrings is used to refer to the shorter, equal-length bitstrings.

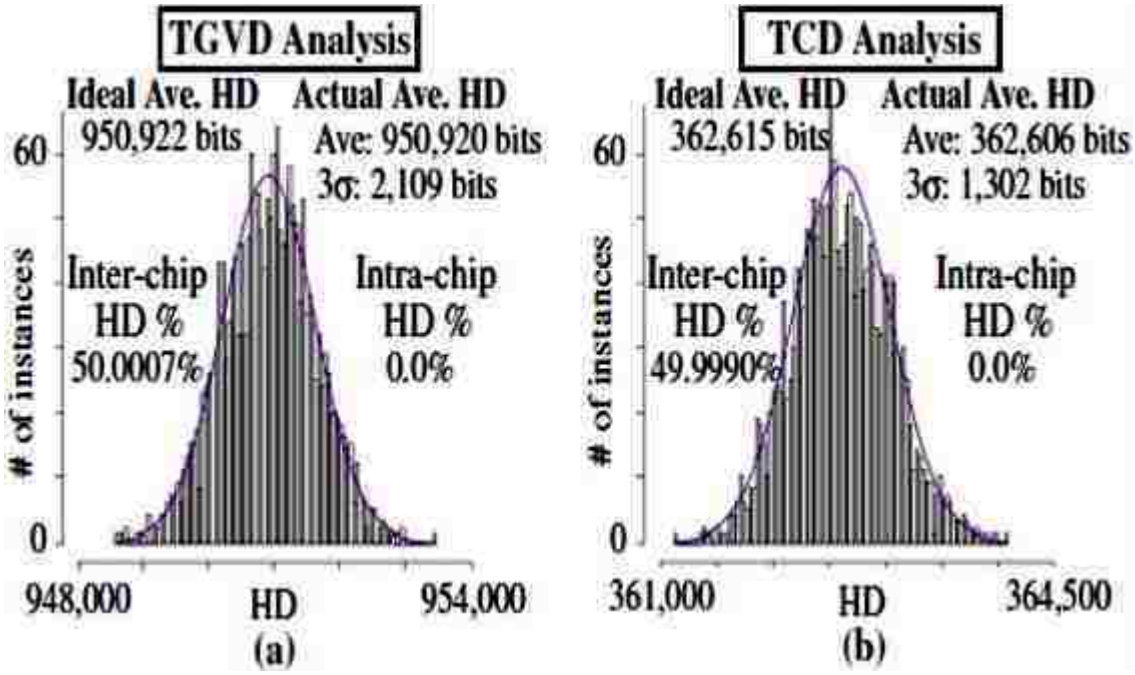


Fig. 58. Inter-chip Hamming Distance using a) TGVDs and b) TCDs.

The actual average inter-chip HDs listed in Fig. 58 are nearly equal to the ideal value of 50%. In contrast, the average inter-chip HDs for the bitstrings of length 5,662,020, i.e., those with the weak bits included (not shown), is 48.4% and 48.5% for TGVD and TCD, resp., so removing the weak bits improves the inter-chip HDs. The 3σ values shown in the figure are derived from the Gaussian curves and represent the spread of the distributions (where smaller is better). These values are small relative to the length of the truncated bitstrings, e.g., they are only 0.11% and 0.18% of the lengths for the TGVD and TCD analysis, resp.

The thresholding technique ensured that the average intra-chip HD across all chips is 0.0% as shown in Fig. 58 for both analyses. However, the underlying noise levels can be measured by disabling thresholding, yielding average intra-chip HDs of 5.11% and 8.68% for the TGVD and TCD analyses, resp. The increase in the TCD intra-chip HD over that given for TGVD reflects the noise added by the VDC digitization process. Fig. 59 depicts the intra-chip HDs for each of our chips tested for the unstable VDC-derived bitstrings (defined as bitstrings generated by disabling thresholding) while Fig. 60 depicts this for the voltage-derived bitstrings.

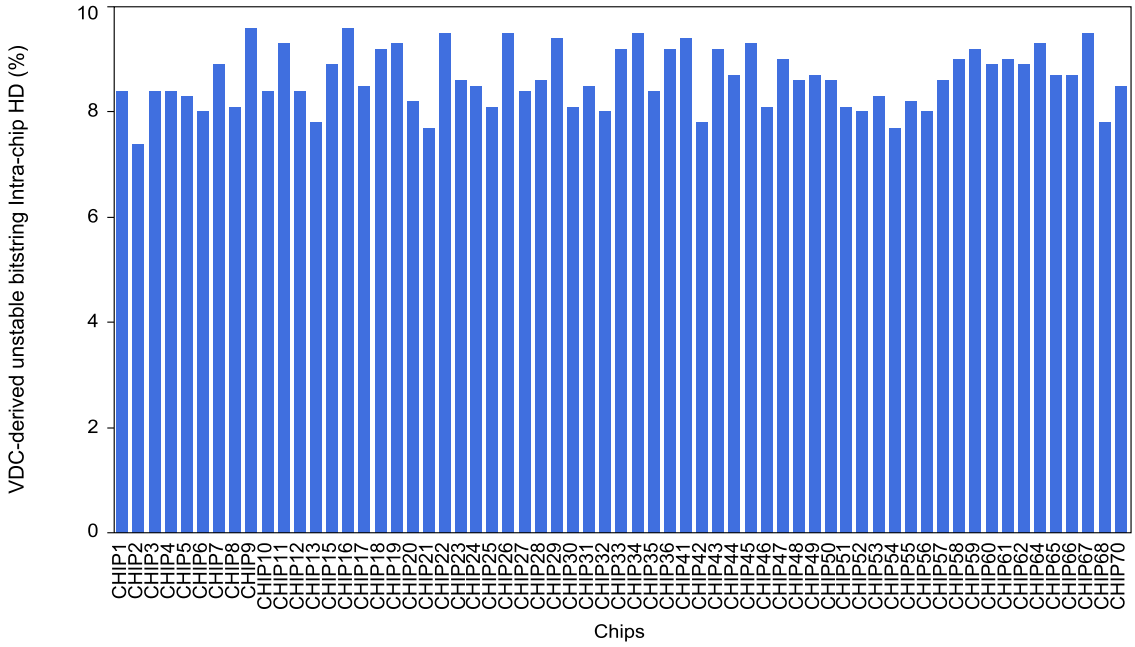


Fig. 59. Intra-chip HD of the unstable VDC-derived bitstrings for the 63 chips tested

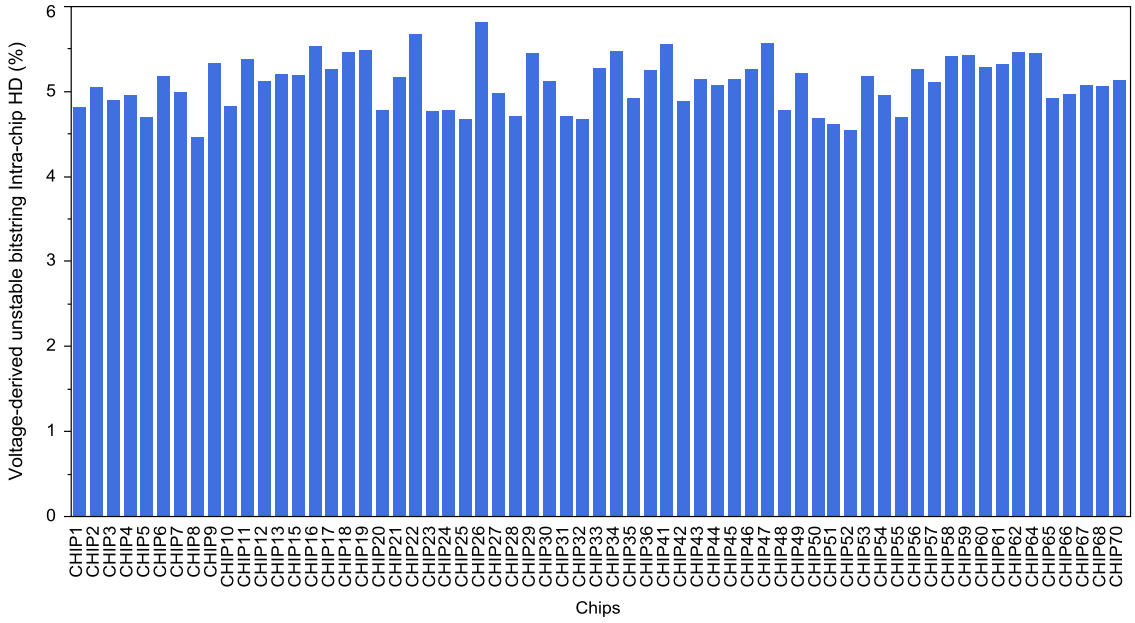


Fig. 60. Intra-chip HD of the unstable voltage-derived bitstrings for the 63 chips tested

The NIST tests [1] look for ‘patterns’ in the bit strings that are not likely to be found at all or above a given frequency in a ‘truly random’ bit string. For example, long or short strings of 0’s and 1’s, or specific patterns repeated in many places in the bit string work against randomness. The output of the NIST statistical evaluation engine is the number of chips that pass the null hypothesis for a given test. The null hypothesis is specified as the condition in which the bitstring-under-test is random. Therefore, a good result is obtained when the number of chips that pass the null hypothesis is large. We applied the NIST statistical tests to the truncated bitstrings of the 63 chips at a significance level of 0.01 (the default). The TGVD and TCD bitstrings pass all tests, with no fewer than 60 passing chips per test (the number required by NIST for the test to be considered ‘passed’). Moreover, all tests passed the P value-of-the-P values metric. Fig. 61 depicts the NIST test results for the 13 tests applied to the voltage-derived and VDC-derived stable bitstrings from the TG-PUF for all 63 chips. As can be seen, at least 61 or more chips pass all tests with the required number for a pass being 60 chips.

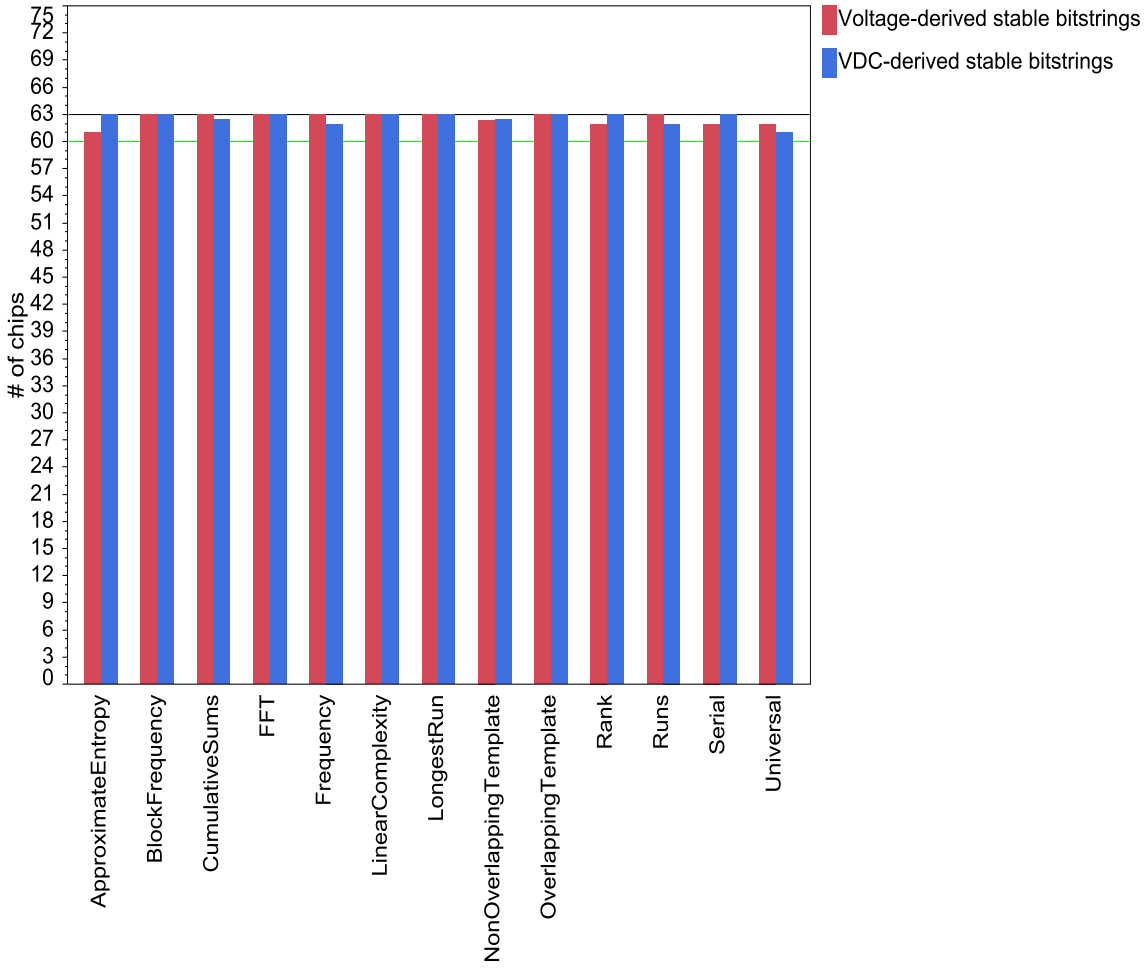


Fig. 61. NIST test results for the voltage-derived and VDC-derived stable bitstrings

Fixed-length bitstrings were also created using the TMR-based scheme. In the experiments, we were able to create, on average, 381 1024-bit TMR-based bitstrings per chip using TGVD data, and 156 on average using TCD data. Although not shown, the statistical test results are similar to those discussed above for the longer bitstrings.

6.2 I-PUF Bitstrings

Similar to the TG-PUF analysis in Section 6.1, we evaluate the several important statistical properties of the voltage-derived (VOD) bitstrings including randomness, uniqueness and probability of bit flips, e.g., failures to regenerate the bitstring under different environmental conditions. The VOD analysis is carried out on bitstrings generated from digitized voltages obtained from an off-chip voltmeter (no VDC involvement). No VDC-derived bitstrings were evaluated due to the reasons elaborated in Section 5.1.2. Fig. 62 depicts the inter-chip hamming distance (HD) distribution of the voltage-derived stable bitstrings after thresholding was applied while Fig. 63 depicts the distribution of the voltage-derived unstable bitstrings before any thresholding was applied. The graphs plot HD along the x-axis against the number of instances on the y-axis. With 62 chips tested, the total number of instances is 62*61/2 = 1,891. The distributions are ‘fitted’ with Gaussian curves to illustrate the level of conformity they exhibit to this distribution.

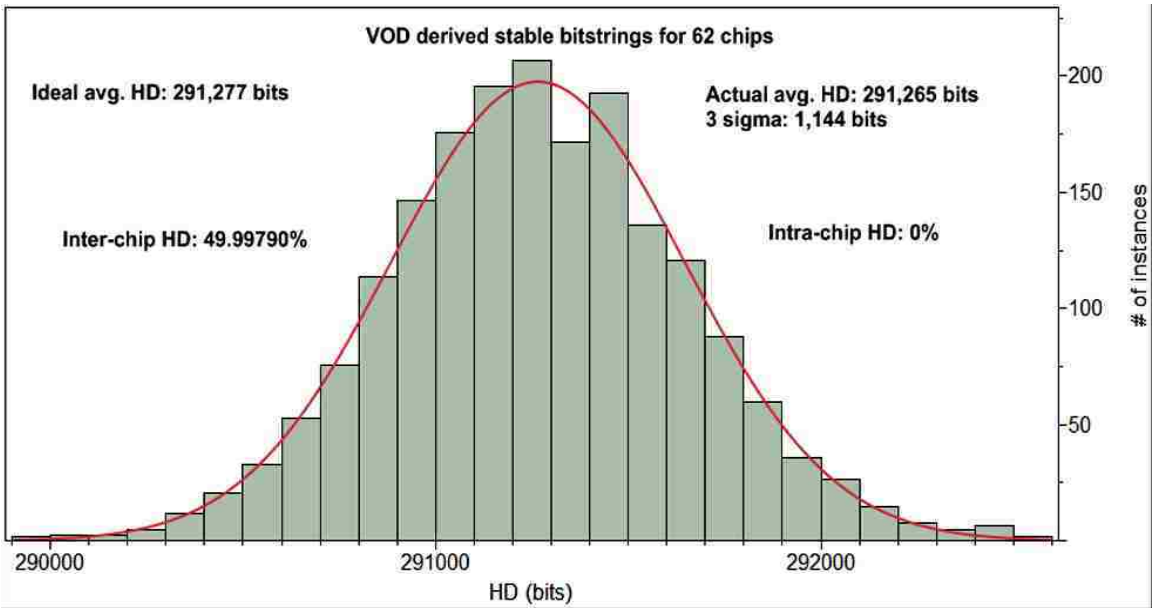


Fig. 62. HD analysis of VOD derived stable bitstrings for the I-PUF based on data collected from 62 chips

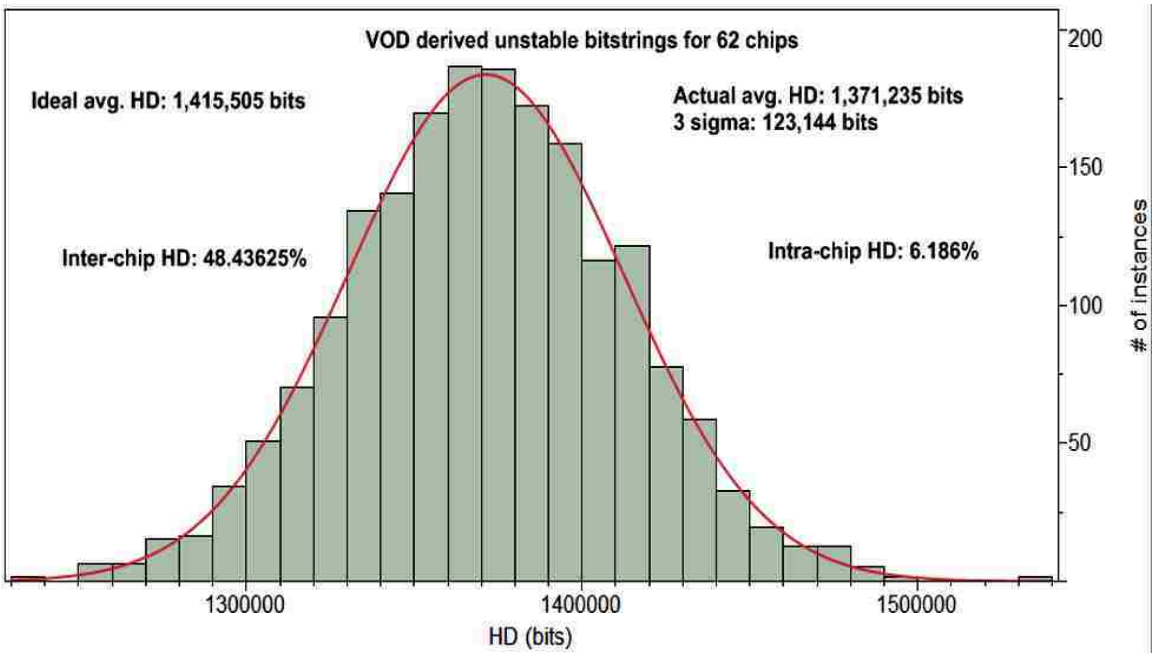


Fig. 63. HD analysis of VOD derived unstable bitstrings for the I-PUF based on data collected from 62 chips

Since HDs must be computed across bitstrings of equal length, it was necessary to truncate the bitstrings used in Fig. 62 to the length obtained for the chip with the fewest number of strong bits. This defines the length of all bit strings for the purposes of the HD analysis. Truncation reduced the lengths to 582,554, which is approx. 20.5% of the maximum possible length, i.e., 2,831,010 bits. The chip with the longest bitstring, in comparison, uses 23.19% of the maximum possible length. The term truncated bitstrings is used to refer to the shorter, equal-length bitstrings.

From Fig. 63, it can be seen that before any thresholding is applied to eliminate weak bits, the average inter-chip HD calculated across all 62 chips was 48.43625% based on analysis of unstable bitstrings of length 2,831,010. Also, the average intra-chip HD across the 62 chips and 9 TV corners was 6.186%. Fig. 64 depicts the intra-chip HD of each of the chips tested based on analyses of the unstable voltage-derived bitstrings. Recall that the ideal value for inter-chip HD is 50% and for intra-chip HD is 0%. From Fig. 62, it can be seen that after application of the thresholding technique and eliminating the weak bits, the average inter-chip HD calculated across all 62 chips increased to 49.9979% based on analysis of truncated stable bitstrings of length 582,554, while the average intra-chip HD dropped to the ideal value of 0%.

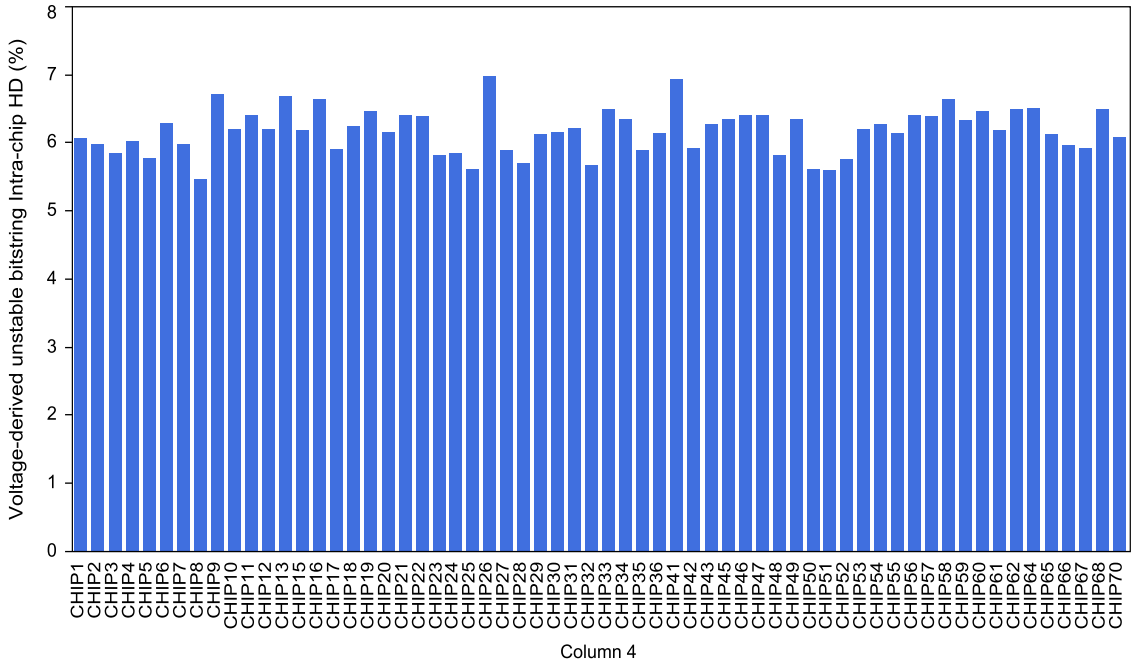


Fig. 64. Intra-chip HD of each of the 62 chips based on analyses of the unstable voltage-derived bitstrings

Similar to the TG-PUF bitstring analyses, NIST tests were performed on the stable voltage-derived bitstrings generated from the I-PUF. Fig. 65 depicts the results of the 13 NIST tests there were applied to the bitstrings generated from all 62 chips. As can be seen, atleast 60 of the 62 chips tested passed all the NIST tests and the number of passing chips required in order to statistically consider a test as a pass was 59.

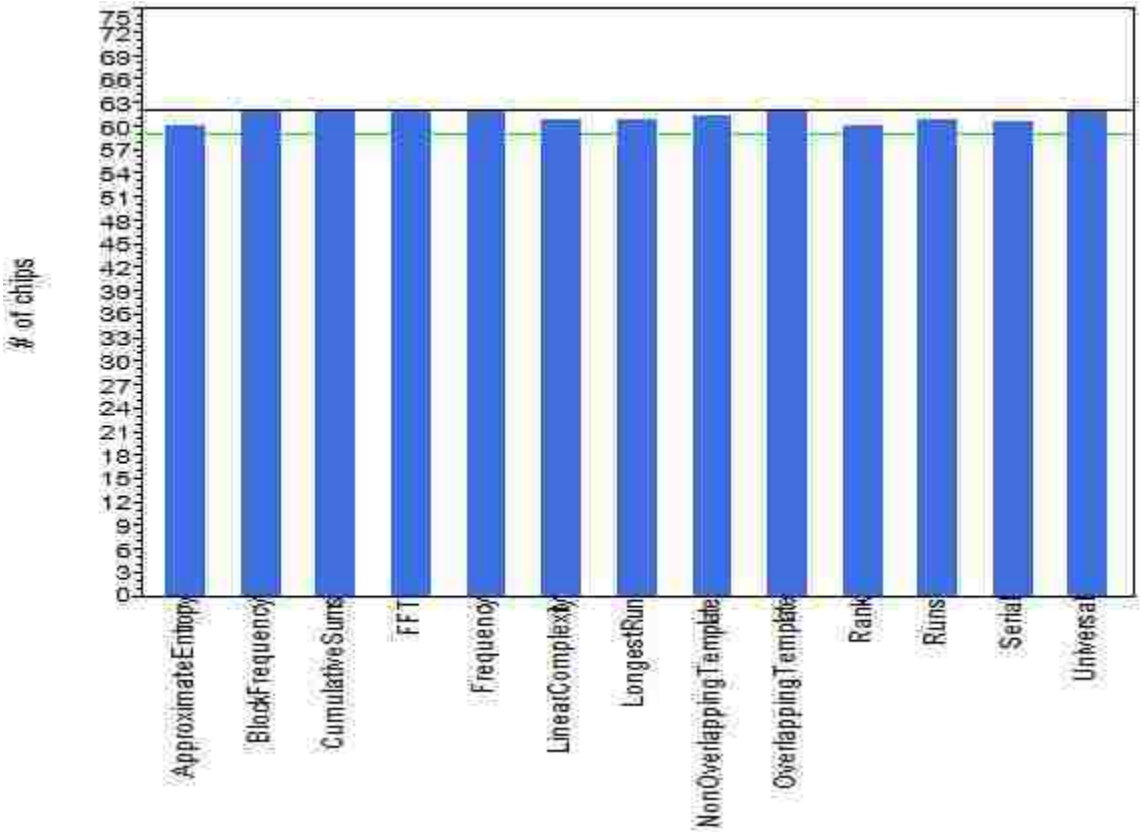


Fig. 65. NIST test results for the voltage-derived stable bitstrings from the I-PUF

6.3 Bitstring Construction Strategies – "ABS" vs. "DIFF"

As stated previously, preliminary experimentation with different bitstring construction strategies revealed the effects of sense wire bias in certain cases that prompted the selection of a strategy that eliminated the effects of wire bias. Two bitstring construction strategies, hereon referred to as the "DIFF" and "ABS" methods, were studied using the TG-PUF as a test vehicle.

In the DIFF method, voltage differences between consecutive TGVs in the stack are taken and those differences are then compared with each other in all combinations to generate a bitstring. Therefore, the bitstring length per chip for the TG-PUF is calculated as (7 voltage differences X 85 SMCs) X 2 MOS types and then all combinations of that result. This method generates a bitstring of length 707,000. As stated previously, this method of bitstring construction eliminates the bias effects of unequal sense wires to the SMCs.

In the ABS method, the absolute TGV voltages (and not their differences) are compared with each other in all combinations to generate a bitstring. Therefore, the bitstring length per chip for the TG-PUF is calculated as (8 voltages X 85 SMCs) X 2 MOS types and then all combinations of that result. This method generates a bitstring of length 924,000. The bias effects of unequal sense wires to the SMCs should be visible with this method.

In these preliminary experiments, no testing was done at different temperature and voltage conditions, so the Intra-chip HD, which is a measure of the reproducibility, was calculated based on repeated sampling (measurement noise and not TV noise). The TG-PUFs on 60 copies of the 90nm chips were tested and the results of those experiments follow.

The distribution of the HDs for the bitstrings generated using the DIFF and ABS methods are shown in Figs. 66(a) and (b), resp. HD is plotted along the x-axis against the number of instances on the y-axis. The total number of instances is given by all combinations of the chips' bit strings, i.e., 60*59/2 = 1,770. After removing unstable or

weak bits using the thresholding technique described previously and a 1mV voltage threshold, the "stable" bit string lengths reduce, on average, by 15.3% and 4.3% for the DIFF and ABS analyses, respectively. It is the HD distribution of these "stable" bitstrings that are depicted in Figs. 66(a) and (b). The size of the bitstrings used in the HD analysis is smaller still at 588,230 and 874,240 for the DIFF and ABS methods respectively. This adjustment is necessary because the HD analysis must be carried out on bit strings of equal length. To accomplish this, the chip with the shortest stable bit string is used to define the length of all bit strings. Although the number of bits discarded as unstable using the threshold method is relatively large, the benefits of constructing a stable bit string in this fashion are significant. The average inter-chip HD was seen to improve for both the DIFF and ABS methods as a function of applying the thresholding technique to discard the unstable bits. The superimposed Gaussian curve on the DIFF distribution of Fig. 66(a) illustrates the distribution is close to ideal, and reflects the power of differential analysis. The ABS distribution, on the other hand, is skewed somewhat to the left, which indicates that a component of the bias discussed earlier is still present.

The bias issue associated with the sense wire routing appears to be significantly reduced using the threshold technique, as depicted by shape of the distribution and results given in Fig. 66(b). However, the shape of the DIFF distribution is a better fit to a Gaussian than the ABS distribution and the Std. Dev. is desirably smaller, e.g., 388 vs. 617. The inter-chip HD of the stable bitstrings is clearly superior for the DIFF method (50.002%) than the ABS method (48.983%). The intra-chip HD of the unstable bit string from Figs. 66(a) and (b) indicates that the ABS method has a slightly better repeatedly,

but as stated earlier this was based on repeated sampling and not extensive TV testing. Again, application of the thresholding technique to bitstrings generated by both methods reduced the intra-chip HD of the stable bitstrings to the ideal value of 0%.

The length of the bit strings allowed 11 of the 15 NIST statistical tests to be performed. All tests except the Overlapping Template, Random Excursions, Random Excursions Variant, and Linear Complexity tests were performed. The ABS bit strings do poorly, producing 0 passing chips on several tests including Runs, Longest Runs, Approx. Entropy and Serial for all seeds. The poor performance is caused by the sense wire bias that still remains in the data and demonstrates that the threshold technique is limited in how much bias it can remove for the ABS method. In contrast, the DIFF bit strings pass all tests except 2 Non Overlapping Template tests out of 148 sub-tests, both of which fail by only 1 chip. For our sample size of 60 chips, the NIST tests consider the tests as a pass as long as there are at least 57 passing chips. This result clearly demonstrates the power of differential analysis to extract randomness and eliminate the adverse effects of bias.

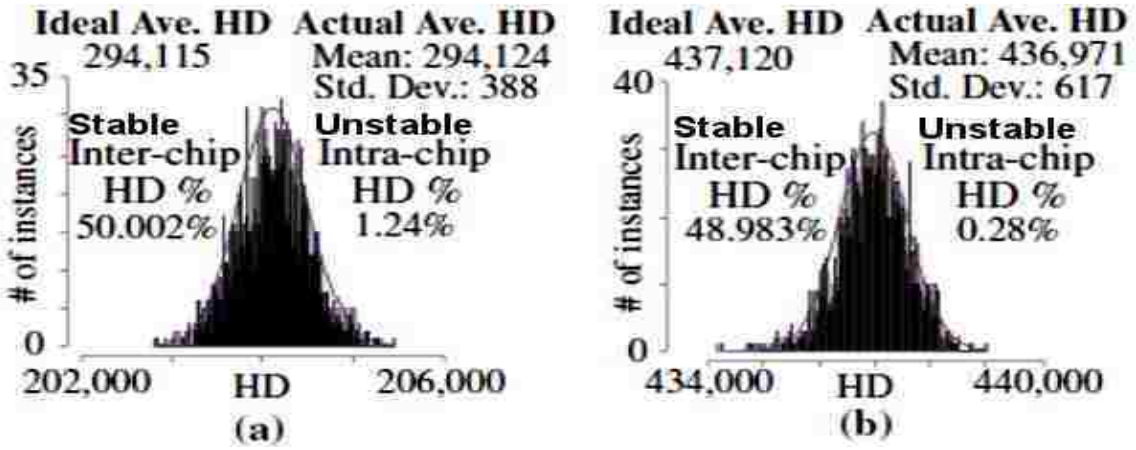


Fig. 66. Distribution of HDs using bitstrings generated from (a) DIFF (b) ABS

Chapter 7

PUF Stability to Temperature and Voltage Variations

7.1 TG-PUF Analyses

Referring to (7) and (8), it is clear that two parameters contribute to the non-linear behavior of the TGVD shifts with changing TV conditions. First is the ratio of the individual transistor R_{on} values at enrollment conditions and their dependency on the TGV voltage. Second is the % change of the individual transistor R_{on} with TV from its R_{on} value at enrollment conditions. Figs. 67-69 depict these parameters for our collected dataset from one of the chips (Chip 1). It should be noted that in the rest of this chapter, when stating V_{GS} and I_{DS} for PFETs, it is implied that it is actually being referred to $|V_{GS}|$ and $|I_{DS}|$ for PFETs.

From Figs. 67(a) and (b), we see that the R_{on} ratio changes significantly with TGV indicating that the two R_{on} ratios in the denominators of (7) or (8) could be significantly different. From Figs. 68-69, it is clear that the % changes in R_{on} with TV also exhibit a dependency on the TGV values. This is more apparent when referring to Figs. A25-A28 in Appendix A which shows the dependency is more pronounced for the stacked NFETs and PFETs, due to the fact that they operate in the linear region, indicating that the four % shifts in R_{on} in the denominators of (7) or (8) could be significantly different from each other. Also, it is evident that the % changes in R_{on} are greater for NFET9 and PFET9 than those of the stacked NFETs and PFETs for any given TV, with the NFETs

generally higher than the PFETs. Another noteworthy observation from Figs. 68-69 is that there is significantly greater noise in the % changes in R_{on} at 1.08V regardless of the operating temperature, because the R_{on} is the highest at 1.08V and therefore is expected to have a larger variation in deviations from enrollment. The TGV pairings at 1.08V are also responsible for a large portion of the bit flips due to the greater variation. Explanations for these observations are provided later in this section.

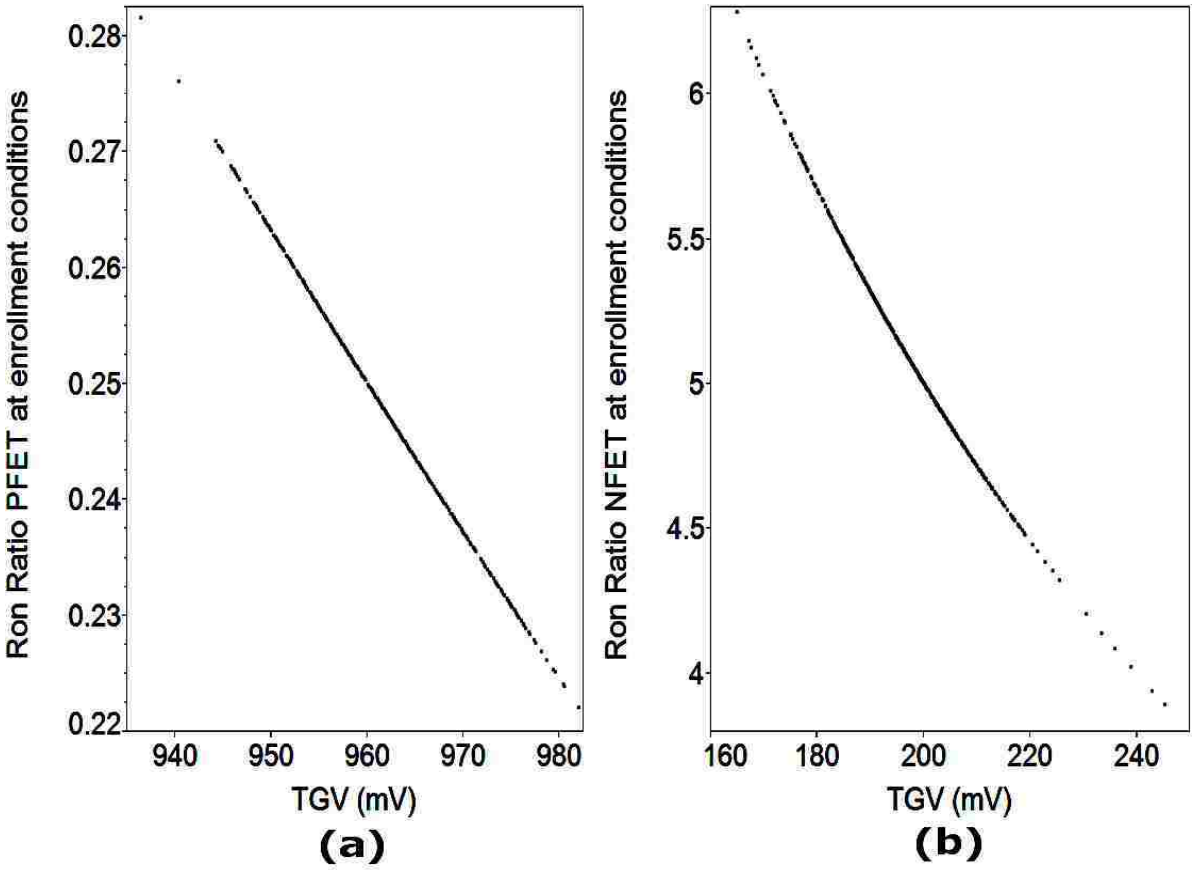


Fig. 67. Chip1 R_{on} ratio versus TGV for (a) PFETs (b) NFETs

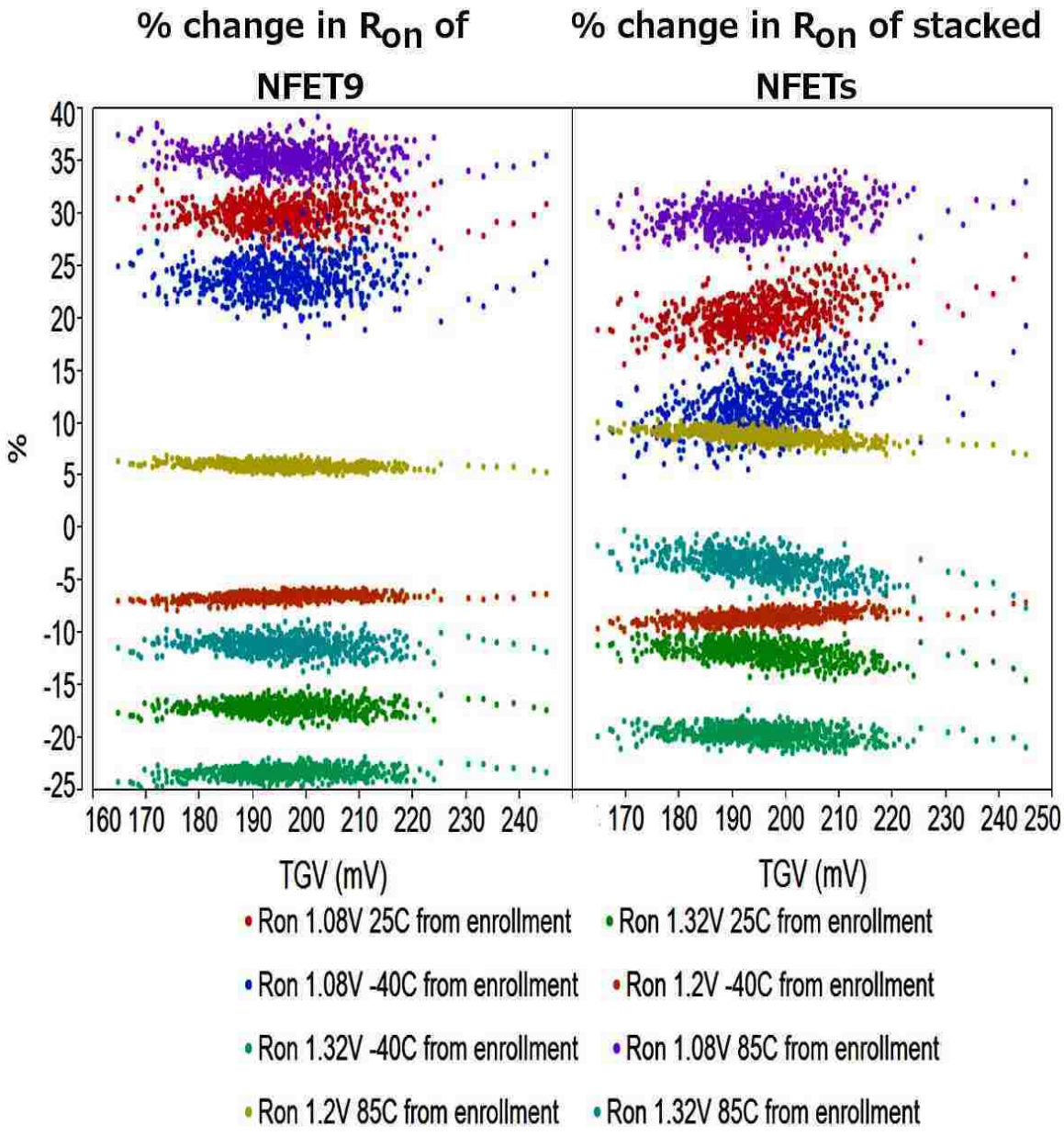


Fig. 68. % changes in NFET R_{on} versus TGV for Chip1

From Figs. A25 and A26 in Appendix A, it should be noted that the stacked NFETs, which operate in the linear region, exhibit a larger change from enrollment conditions as the V_{DS} changes for small V_{DS} values, and a smaller change from enrollment conditions as V_{DS} changes for large V_{DS} values. From Figs. A27-A28, we see that this is also true for the NFET9 transistors, which operate in the saturation region, however a distinct increase in change from enrollment with increasing V_{DS} is not observed due to the fact that at these much higher V_{DS} levels, the R_{on} changes with V_{DS} are much smaller and not noticeable. These behaviors are expected and more comprehensible when considering the generic I_{DS} vs. V_{GS} curves for changing V_{DS} of a NFET shown in Fig. A29 in Appendix A.

All these aforementioned factors lead to disproportionate and sometimes unpredictable shifts in TGVD with TV. Depending on the TGV pairing being compared, if the R_{on} ratio in the denominator of the first term in (7) happens to be significantly different than that in the denominator of the second term, the R_{on} % change terms in (7) play a weaker role in determining the shift in TGVD with TV. However, if the R_{on} ratios are similar, then the differences between the % change in R_{on} for NFET9 (or PFET9) and that of the stacked NFETs (or PFETs) for a given TV will play a dominating role in dictating the shift in TGVD with TV. These are the key reasons for the bit flips seen when comparing two different closely-spaced TGVD values at all 9 TV corners.

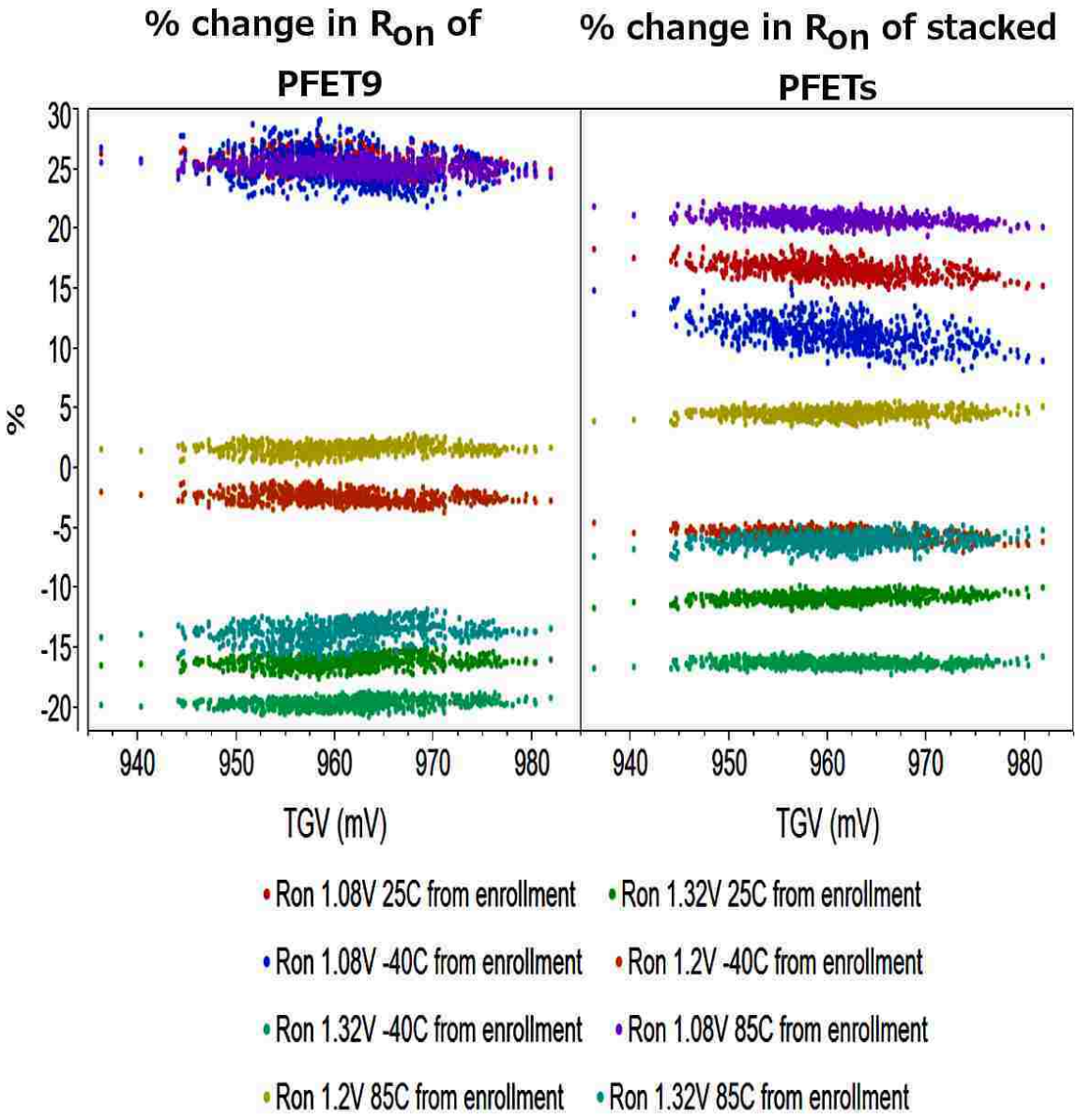


Fig. 69. % changes in PFET R_{on} versus TGV for Chip1

Using the NFET primitive of the TG-PUF as an example and referring to Fig. 9(b), NFET9 always operates in the saturation region whereas the stacked NFETs (NFET1) operate in the linear region. Thus, $R_9 > R_1$. The I_{DS} of MOSFETs exhibit a

bimodal dependence on temperature [74]. At small V_{GS} values, the I_{DS} generally increases with increasing temperature whereas it decreases with increasing temperature at large V_{GS} values. At small V_{GS} values, the threshold voltage's (V_T) dependence on temperature dominates while at large V_{GS} values, the mobility's (μ_n) dependence on temperature dominates. A typical example of this behavior taken from [74] is illustrated in Fig. 70. Assuming a constant V_{DS} (as in Fig. 70), this relationship of I_{DS} with temperature can be an indication of the relationship of R_{on} with temperature, i.e. increase in R_{on} with increasing temperature at larger V_{GS} and decrease in R_{on} with increasing temperature at lower V_{GS}.

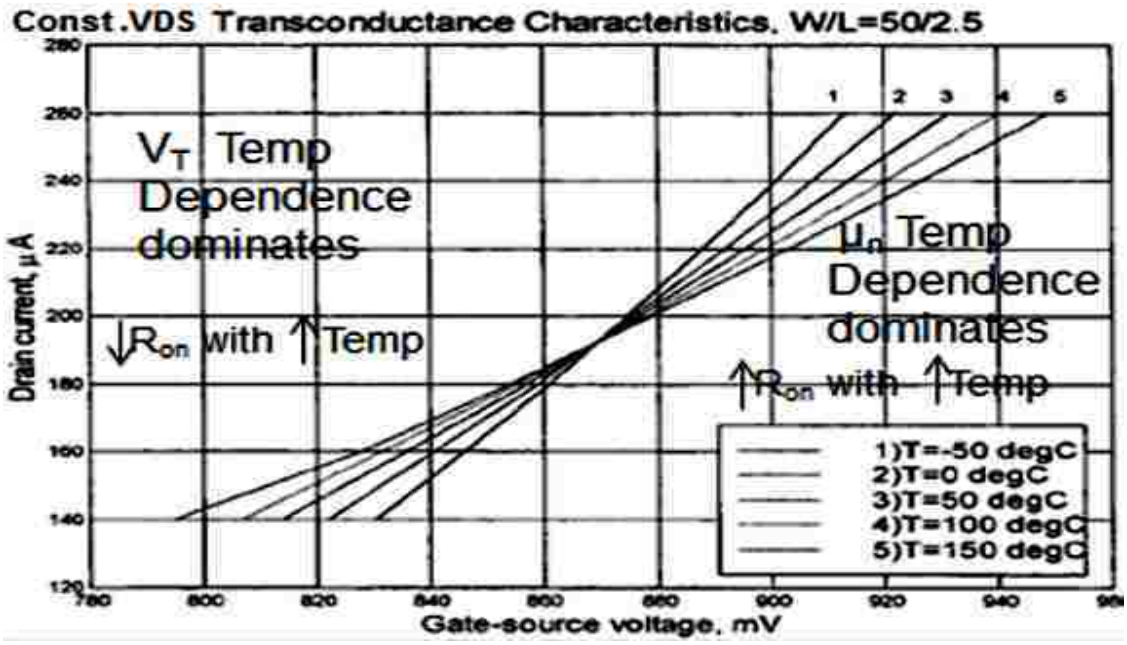


Fig. 70. Illustration of bimodal dependency of R_{on} on temperature

Both V_T and μ_n decrease with increasing temperature but they have opposite effects on the R_{on}. V_T generally decreases at a rate of -2.4mV/ºC. As the temperature increases, a

decreasing V_T leads to a decrease in R_{on} while a decreasing μ_n leads to an increase in R_{on} leading to the bimodal behavior observed in Fig. 70. All our transistors generally operate in the larger V_{GS} region so we see an increase in R_{on} with increasing temperatures for a fixed V_{DD} voltage. This exhibits itself in the form of the slope of the I_{DS} versus V_{GS} curve of the MOSFET being inversely proportional to the operating temperature, as shown in Fig. 70. On the other hand, a decrease in R_{on} with increasing V_{DD} voltages at a fixed temperature, is seen for the transistors in our primitive. Due to these changes in R_{on} with TV, the TGV voltage also exhibits an increase with increasing temperature and voltage. With data from one of our chips, these behaviors of the NFET and PFET primitives are illustrated in Figs. 71 through 77 and Figs. A11 through A13 in Appendix A.

Defining the TV noise of transistor R_{on} as the standard deviation of the transistor R_{on} for the 9 TV corners tested, the summary statistics of all the chips tested revealed that the Coefficient of Variation (CV), defined as the standard deviation divided by the mean, for the TV noise in R_{on} for the stacked NFETs was 15% while that for the NFET9 was 20%. The change in the R_{on} of the stacked transistors with changing temperature is larger than that of NFET9 and PFET9, however, the change in R_{on} of NFET9 and PFET9 with changing V_{DD} is much larger than the stacked transistors causing the higher overall TV noise in R_{on} for NFET9 and PFET9. This larger change in R_{on} of NFET9 and PFET9 with changing V_{DD} is consistent with the fact that voltages on 3 (of 4) terminals (G, D, S) of NFET9 and PFET9 change as V_{DD} changes, whereas the voltage on only 2 (of 4) terminals (G, D) of the stacked NFETs and PFETs change as V_{DD} changes. The larger change in R_{on} of the stacked transistors with temperature is consistent with the fact that

those transistors operate at a higher V_{GS} and a much lower V_{DS} than NFET9 and PFET9 and thus, operate in the linear region where the changes in R_{on} with temperature are larger than in the saturation region. See Figs. 72-73 and Figs. 75-76 for details. It should be noted that both PFET and NFET transistors exhibit an increasing change in R_{on} with changing temperature as the V_{GS} increases for a fixed V_{DS} (or as the V_{DS} decreases for a fixed V_{GS} [74]. In our case, both the larger V_{GS} and a much lower V_{DS} of the stacked transistors (that operate in the linear region) as compared to PFET9 and NFET9 (that operate in saturation region) cause a larger change in R_{on} with temperature for the stacked transistors.

It should also be noted from Fig. 75 that for the saturated transistor PFET9, the changes in R_{on} with changing temperature get smaller with decreasing V_{DD}. The reason for this is clear when referring to the PFET9 I_{DS} vs. V_{GS} curves of Fig. A12. A lower V_{DD} leads to a lower operating V_{GS} for all the transistors and the saturated transistors operate at a lower V_{GS} than the linear transistors, and therefore closer to the V_{GS} inflection point below which the temperature dependency of I_{DS} reverses. As can be seen from Fig. A12, the change in I_{DS} with changing temperature gets smaller with decreasing V_{GS}. Decreasing V_{DD} is what results in the decreasing V_{GS} in Fig. A12, and as V_{DD} decreases, the V_{DS} of PFET9 (not shown) is also not constant and is changing. Therefore, these changing behaviors of V_{DS} and I_{DS} with temperature at different V_{DD} are what cause the R_{on} dependency on temperature to change with V_{DD}. This decrease in R_{on} changes with changing temperature at lower V_{DD} is more pronounced for the PFET9 as compared to the NFET9 because PFET9 operates at a lower V_{GS} and V_{DS} than NFET9 (see Fig. 71 and Fig.

A12) and the inflection point for PFET9 is at a higher V_{GS} (937mV) than NFET9 (870mV); therefore PFET9 operates closer to the V_{GS} inflection point than the NFET9. The R_{on} changes with temperature get smaller the closer the transistor gets to operating at the inflection point. Noteworthy is also the fact that the inflection point of where R_{on}'s dependency on temperature reverses will not match the inflection point of where I_{DS}'s dependency on temperature reverses. This is because unlike in Fig. 70, the V_{DS} of the transistor also changes with changing V_{DD} and is a central component in determining the R_{on}. That is why when analyzing Fig. A12, it appears that the inflection point of R_{on} should also be somewhere between a V_{DD} of 1.08V and 1.2V based on the I_{DS} inflection point. However, as is obvious from Fig. 75, it is somewhere slightly lower than a V_{DD} of 1.08V.

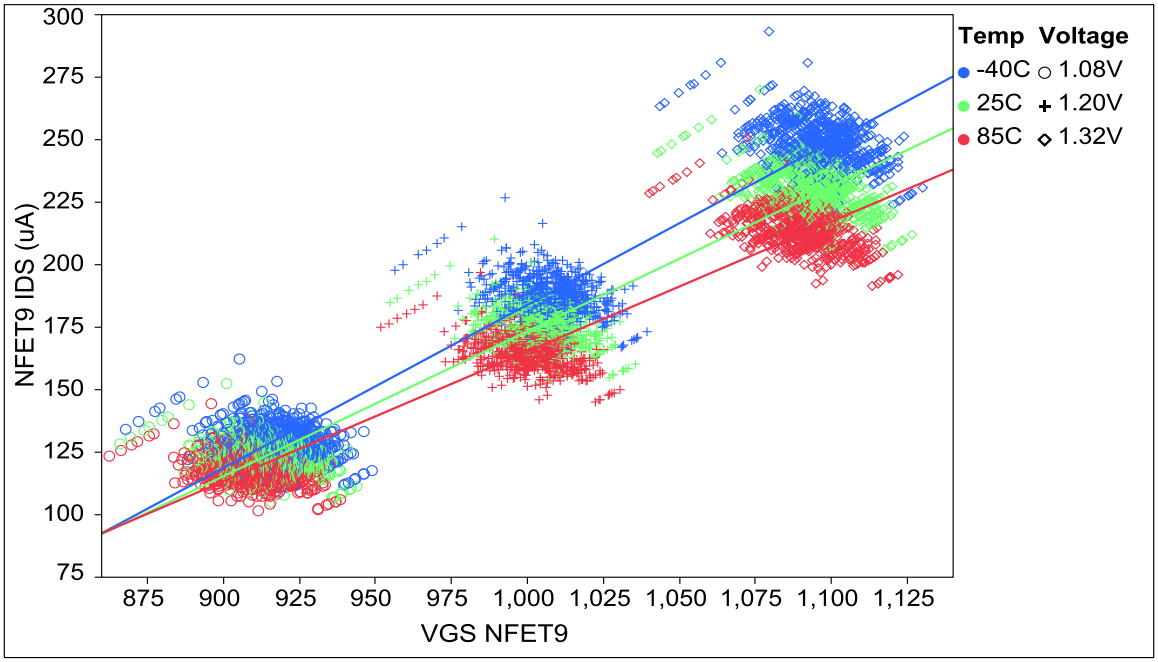


Fig. 71. I_{DS} vs. V_{GS} for NFET9 of Chip1

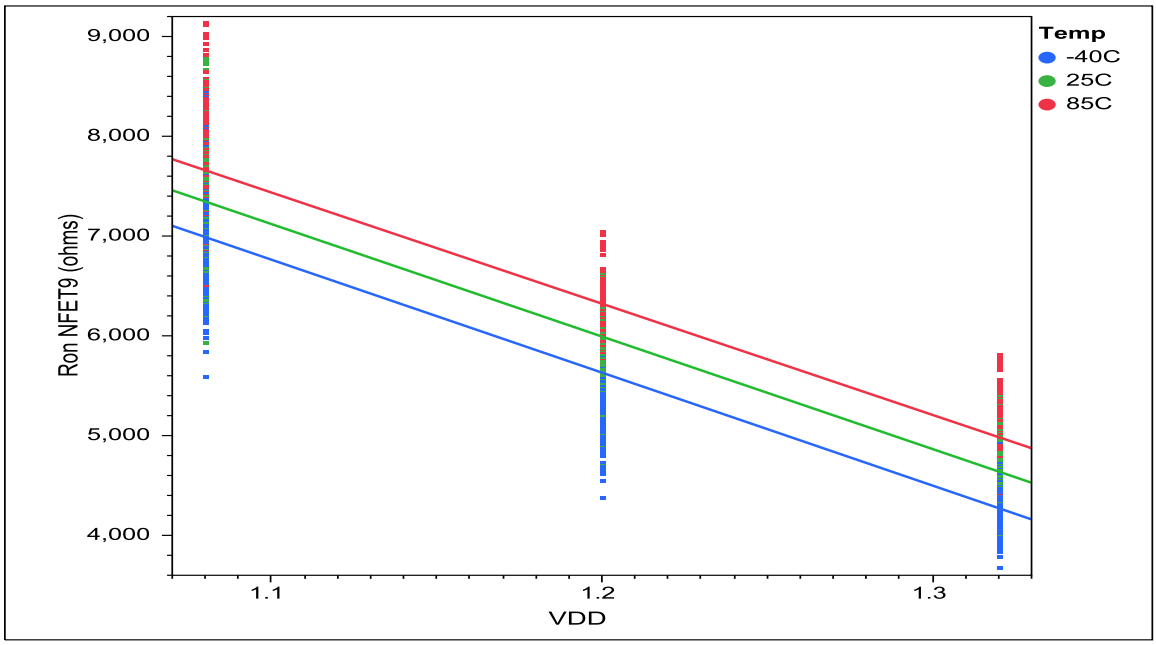


Fig. 72. NFET9 R_{on} changes with TV for Chip1

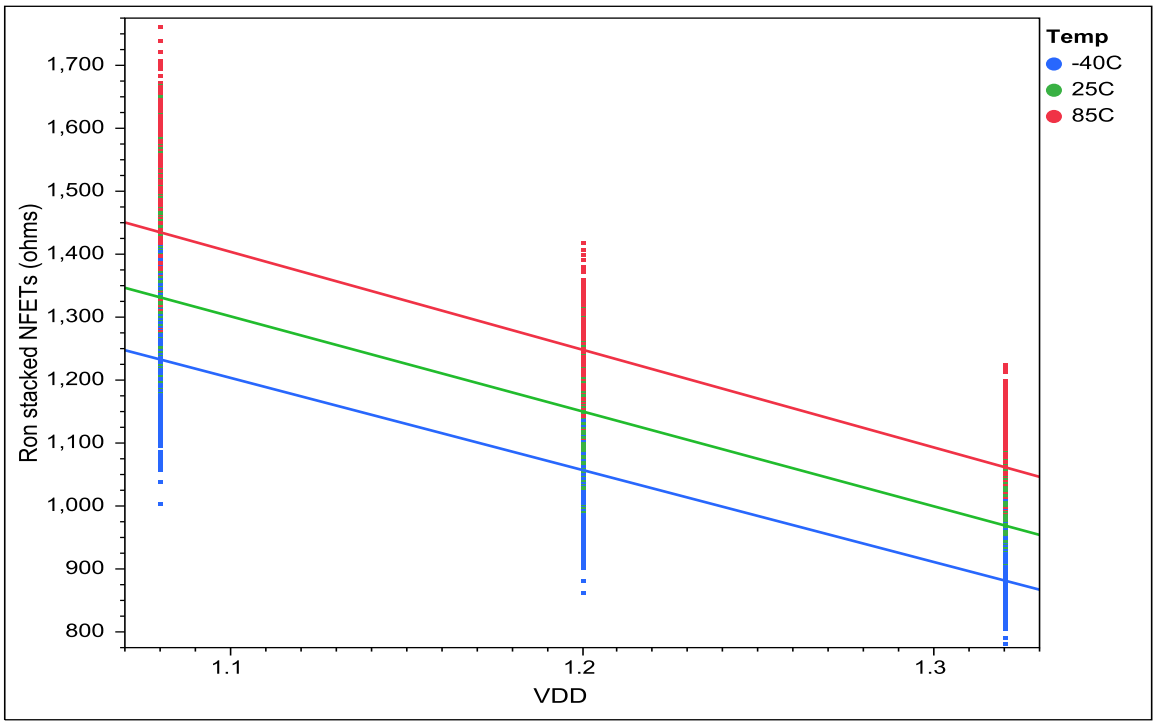


Fig. 73. Stacked NFETs R_{on} changes with TV for Chip1

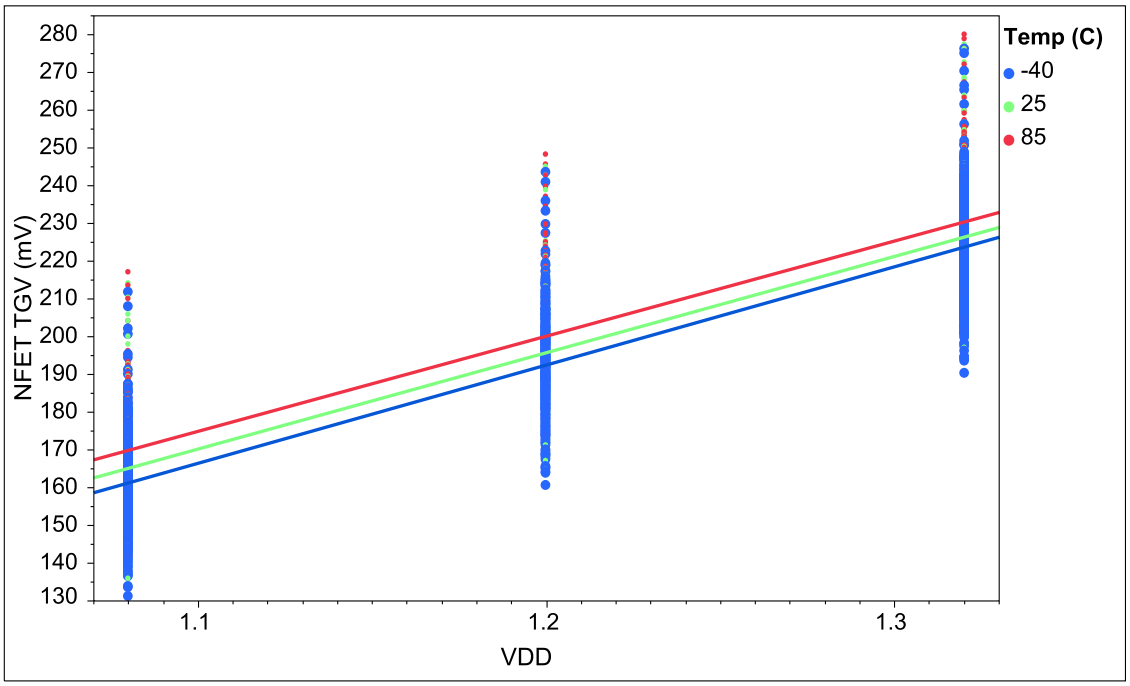


Fig. 74. NFET TGV changes with TV for Chip1

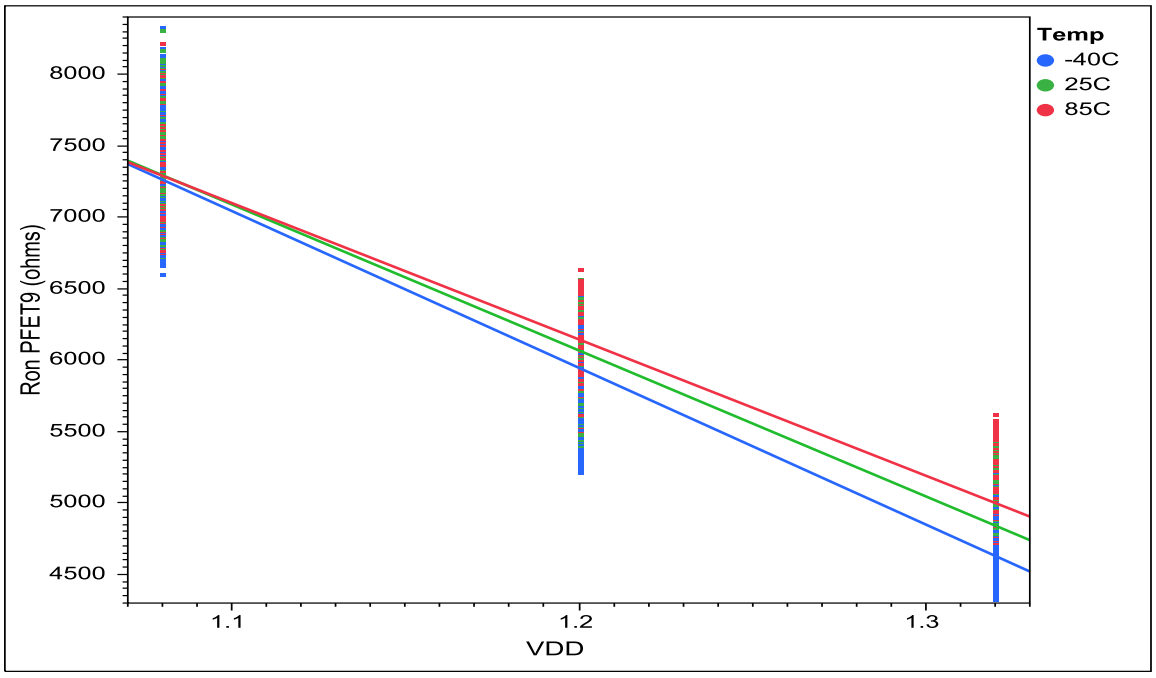


Fig. 75. PFET9 R_{on} changes with TV for Chip1

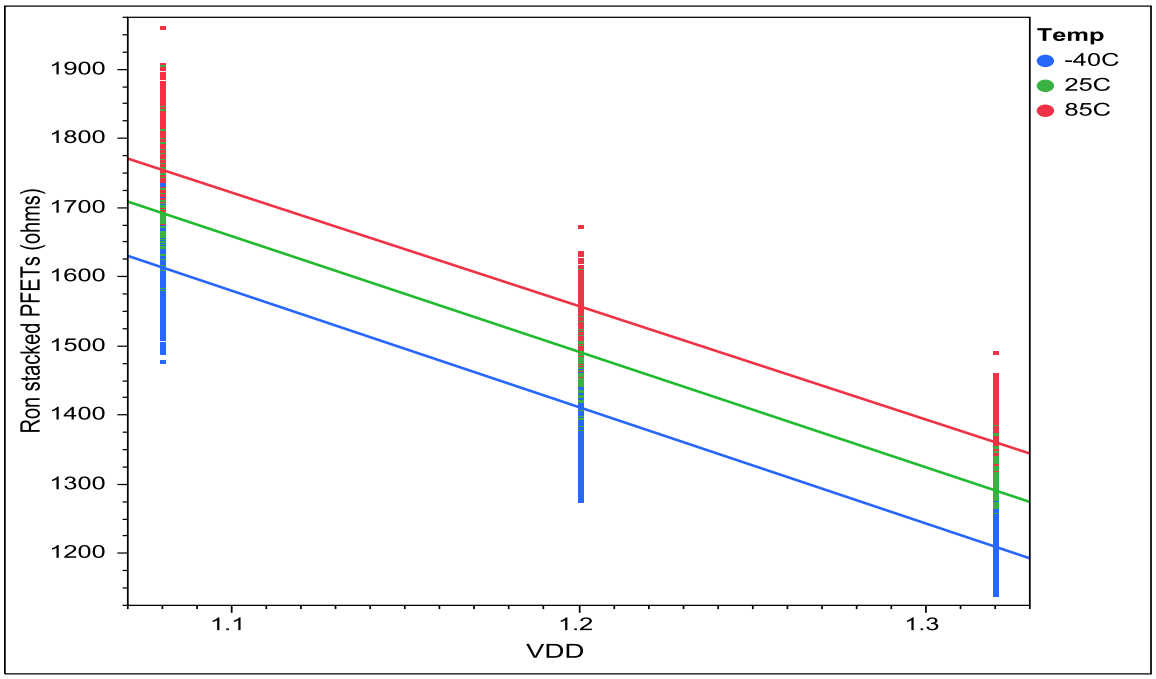


Fig. 76. Stacked PFETs R_{on} changes with TV for Chip1

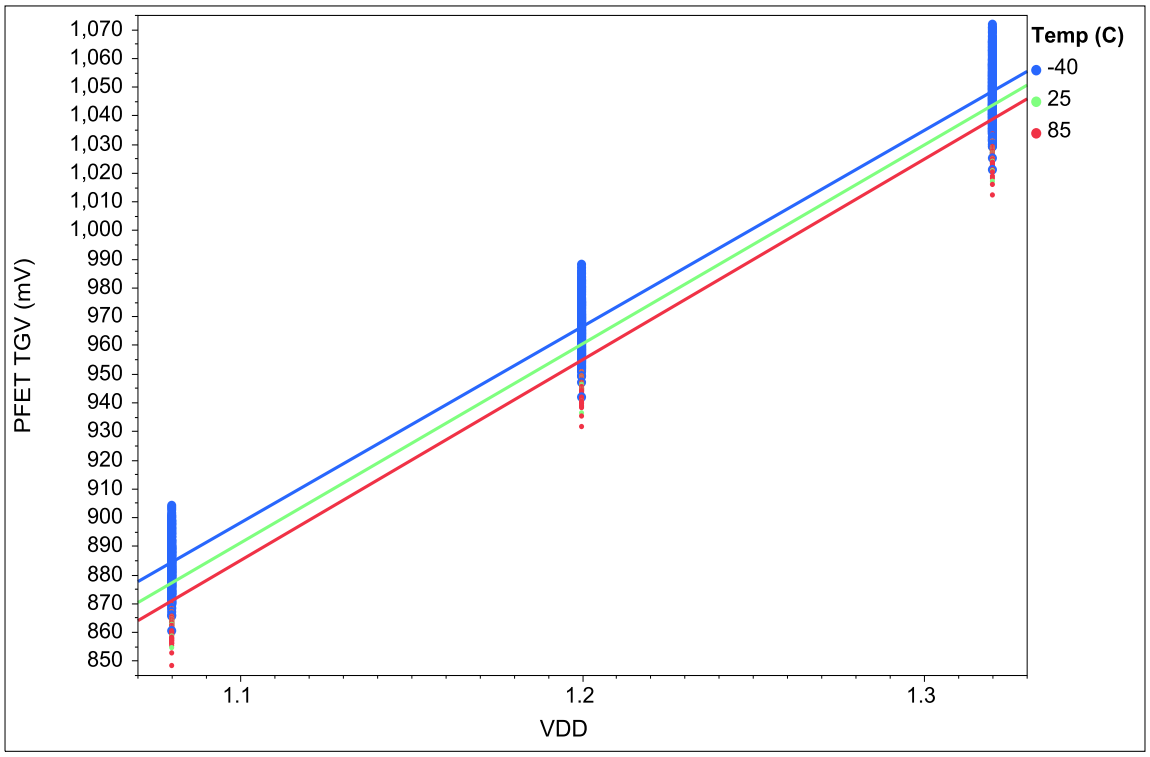


Fig. 77. PFET TGV changes with TV for Chip1

The CV for the TV noise in R_{on} for the stacked PFETs was 12.5% while that for the PFET9 was 17.8%, indicating that the NFETs are more sensitive to TV variations. These results are summarized in Table VI below. The higher sensitivity of NFETs to TV variations is mostly due to a higher sensitivity to temperature variations and can be explained with Fig. A10 in Appendix A. Fig. A10 illustrates the much larger change in electron mobility as compared to the hole mobility as temperature changes, and the mobility has a direct effect on the R_{on} of a transistor.

Table VI: TV Noise of the various transistors in the TG-PUF primitive based on all chips tested

	TV Noise (CV %)
R_{on} of Stacked NFETs	15%
R_{on} of NFET9	20%
R_{on} of Stacked PFETs	12.5%
R_{on} of PFET9	17.8%

Table VII lists the average calculated TV noise for the voltages and their digitized form. It can be seen when comparing the TGVD TV noise that the NFETs have a larger TV noise than the PFETs, and this is consistent with the larger TV noise seen in the NFET R_{on} values. It is the TGVD TV noise that dictates the sensitivity of bit flips to TV variations in the unstable voltage-derived bitstring, as it is the differences of the TGVDs

that are used to generate the bitstring. Evident is also the fact that the difference operation for both the TGV and TC result in a lowering of the magnitude of the TV noise. This is due to the fact that when calculating the TV noise of the TGVs, we are calculating a standard deviation of the TGV voltages (which are large numbers as evident from Figs. A4 and A6 in Appendix A) across different TVs and when calculating the TV noise for the TGVD, we are calculating a standard deviation of the differences in the TGV voltages (which are smaller numbers as evident from Figs. 20 and 21) across different TVs. The TGVD TV noise should be proportional to the TGV TV noise because if the standard deviation (across TV) of the TGV voltages is higher, this increased variation in the TGV voltages will induce an increase in variation in their differences across TV; this will be captured in the TV noise or standard deviation of the TGVDs. Also, as can be seen from Table VII, the magnitude of the TGV TV noise is larger for larger TGV voltages, i.e. the PFET PUFs that operate with a TGV in the 950mV range exhibit a TGV TV noise of 72.4mV and NFET PUFs that operate with a TGV in the 200mV range exhibit a TGV TV noise of 26.8mV. That is the reason their values are normalized using the CV (μ/σ) %. However, it should be clear that the TV noise in TGVD is not dependent on the magnitude of TGVD as evident from Figs. A20 and A21 in Appendix A. Therefore, it is not valid to conclude that the NFETs are expected to exhibit a larger TGVD TV noise magnitude just because their TGVD range is larger; the two are not related. It is however a valid conclusion that the magnitude of the TGVD TV noise will be higher for higher TGV voltages as higher TGV voltages will have larger standard deviations and therefore larger standard deviations of their TGVDs, i.e. a larger TGVD TV noise magnitude. That

is why the magnitude of the TGVD TV noise is normalized with respect to their corresponding TGV TV noise, indicated by the percentages. With this in mind, when comparing the NFET and PFET TGVD TV noise, we would expect a lower value for the NFETs due to their lower TGV voltages and lower TGV TV noise. But, we see the opposite which is explained by the significantly higher TV noise in the NFETs.

Also, the lowering of the TV noise magnitude with the difference operation is not as dramatic for the TCDs as compared to the TGVDs indicating that the TCD TV noise is much larger in comparison to the TGVD TV noise. The TV noise in the TCD determines the sensitivity of bit flips to TV variations in the unstable VDC-derived bitstring. This explains the increased TV noise with digitization and the associated higher intra-chip HD of the VDC-derived unstable bitstring as compared to that of the unstable voltage-derived bitstring. A reduced difference in TCD TV noise between NFETs and PFETs is also seen due to the digitization process. The 2.04X TGVD TV noise for the NFETs compared to the PFETs reduces to 1.29X TCD TV noise for the NFETs compared to the PFETs.

Table VII. TGV, TC, TGVD, and TCD TV Noise for Chip1 NFET and PFET

	TGV TV noise (CV %)	**TC TV noise (CV %)**	**TGVD TV noise**	**TCD TV noise**
Chip1 NFET	26.8 mV (13.7%)	1.36 bits (2.1%)	0.9 mV (3.3%*) * normalized to TGV TV noise	1.14 bits
Chip1 PFET	72.4 mV (7.5%)	1.4 bits (1.6%)	0.44 mV (0.6%*) * normalized to TGV TV noise	0.88 bits

Fig. 78 illustrates the general behavior of the VDC as a function of changing TV. Here, the TC is plotted as the Cal1 voltage is swept at different TVs. The mean and 3σ curves are superimposed. The average 3σ, computed using the individual 3σ in each curve, is less than 1 for all curves. The small non-linearity in the curves does not degrade the statistical properties of the bitstrings, as shown below. This figure illustrates the need for the calibration process described in Section 3.5.4 to compensate for the TV shifts in the offset voltages. A fixed TGV voltage could cause overflow problems past the 120 bit capacity in the VDC due to shifts in the curves along the x-axis with changing TV. An offset voltage and a calibration factor are applied to the TGV voltages before they are programmed into the Cal1 power supply for the VDC. From Fig. 78, it can be seen that

the sensitivity of the VDC is approx. 1 TC bit per millivolt change in Cal1. The TGVs for a typical chip vary over the range of 40 to 80 mV so about half of the 120 bit range of the VDC is used in our experiments.

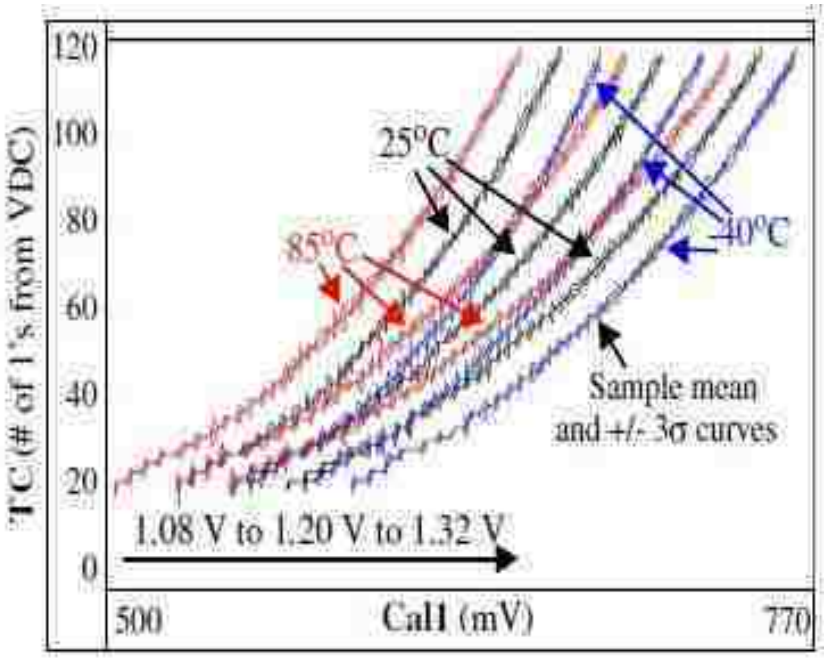


Fig. 78. VDC TC versus Cal1 for Chip1 across 9 TV corners

Fig. 79 depicts the behavior of the VDC output (TC) as a function of changing TV for the NFET TGV voltages after voltage offsets are added to the TGV voltages using the calibration process described in Section 3.5.4. Fig. 80 depicts this behavior for the PFETs. It can be seen that the shifts in the curves are greater for changing power supply voltages than changing temperatures for both NFETs and PFETs. It can also be seen that the slopes of the curves (or resolution) decreases slightly with increasing power supply voltages for any given temperature and the shifts of the curves with changing temperature

get slightly smaller with increasing supply voltage. Two conclusions can be made from Figs. 79 and 80. First, the resolution of the VDC is about 1 bit/mV which allows capability to measure variation of up to 120mV as the VDC has a measurement capacity of 120 bits. Second, the PFETs have a much tighter range of TCs as compared to the NFETs due to the smaller voltage variation or voltage range for the PFETs.

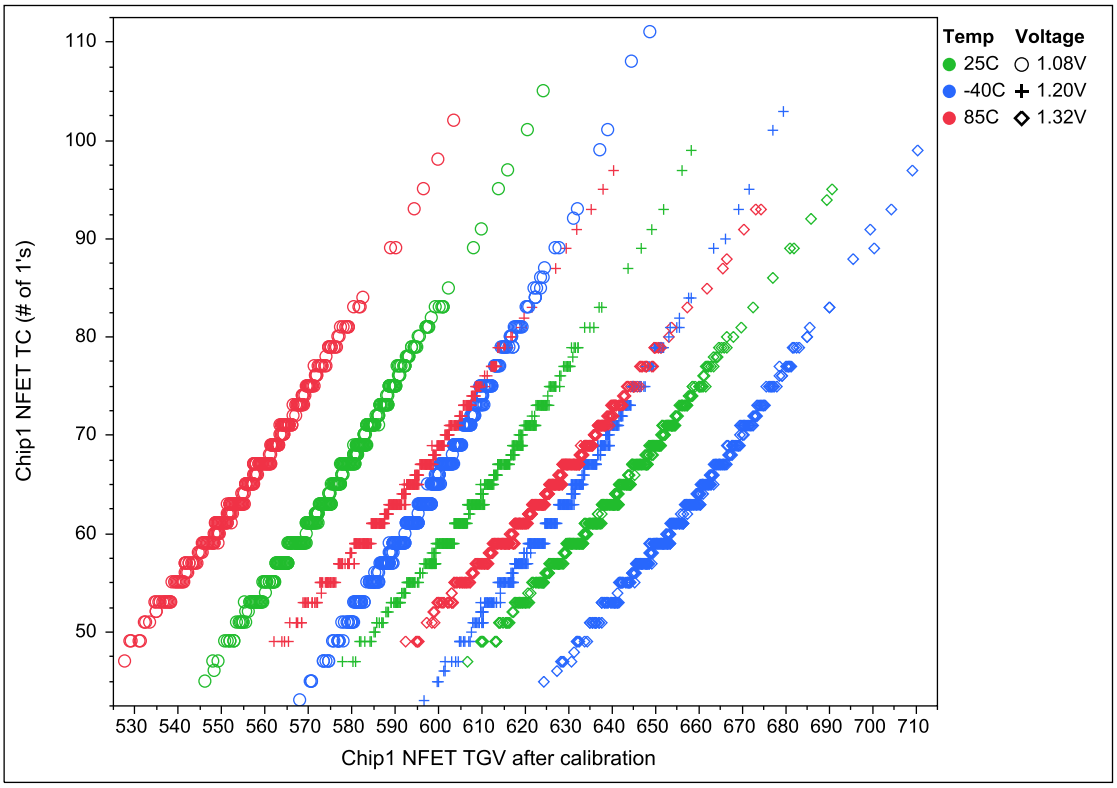


Fig. 79. TC versus calibrated TGV for the Chip1 NFETs

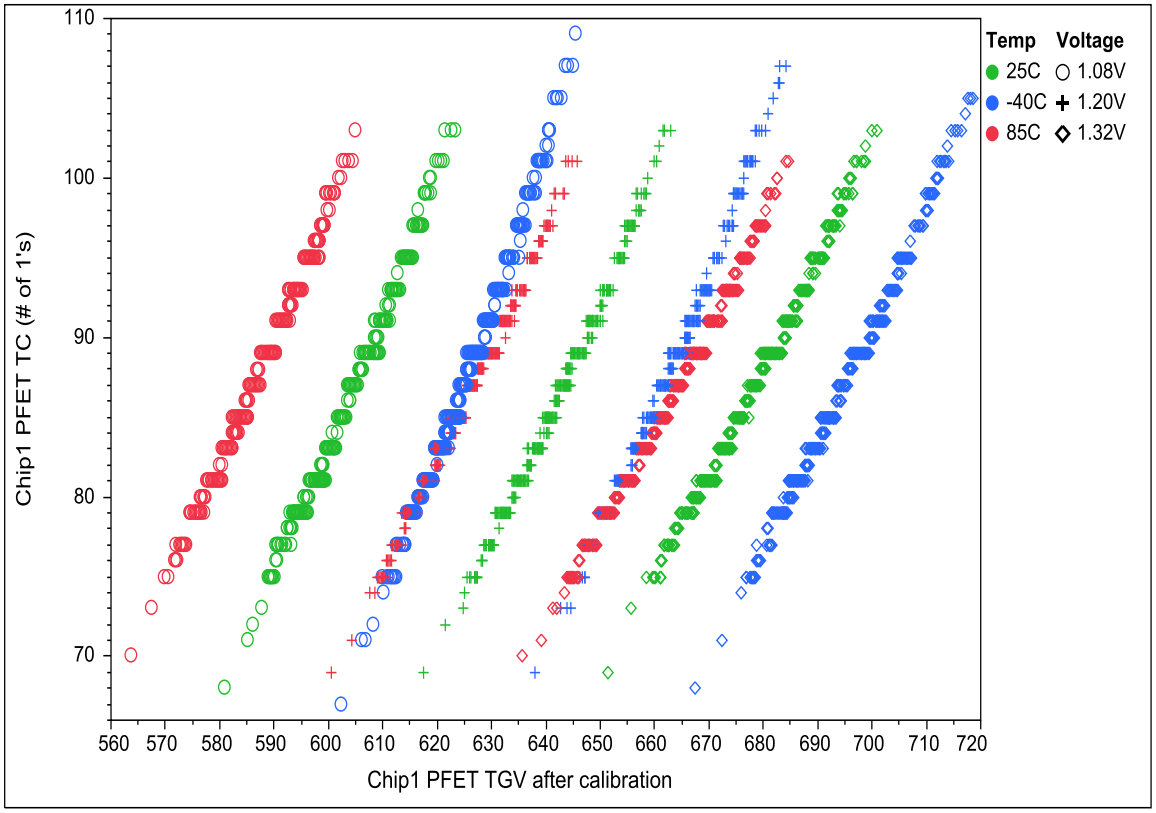


Fig. 80. TC versus calibrated TGV for the Chip1 PFETs

Fig. 81 illustrates a bit flip when comparing a TCD from SMC6 with a TCD from SMC7. It can be seen that this bit flip is caused by the reversal of the relative ordering of the 25C,

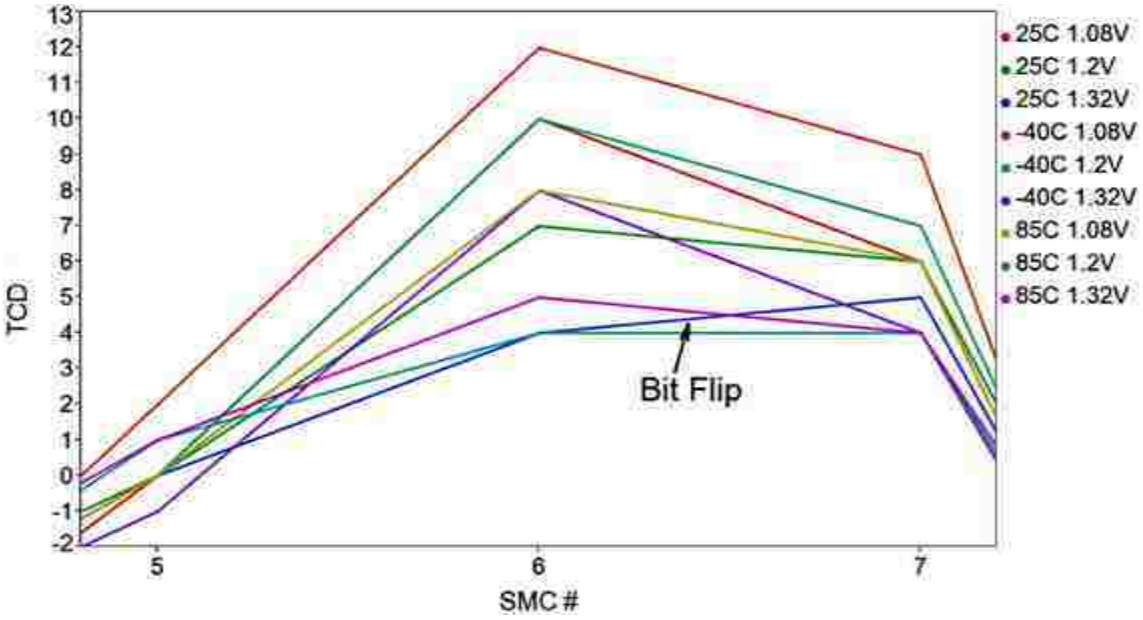


Fig. 81. Illustration of a bit flip in Chip1 NFETs

1.32V TCD comparison which manifests itself as a positive slope, while the slopes for all the other TV comparisons are negative. On the other hand, when comparing the TCD from SMC5 with that from SMC6, there are no bit flips as all the slopes are positive.

7.2 PG-PUF Analyses

Referring to (9), it is evident that plotting PGERD versus temperature yields a fit the equation of which is represented by (13).

$$PGERD(T) = PGERD_o - \alpha(PGERD_o)(T_o) + \alpha(PGERD_o)(T) \qquad (13)$$

In (13), $PGERD_o - \alpha(PGERD_o)(T_o)$ is the y-intercept and $\alpha(PGERD_o)$ is the slope. Fig. 82 displays this relationship for our collected dataset from one of the chips. Given that we now know the slopes and y-intercepts of the curves in Fig. 82, we calculate the α to be 0.0025/C and the $PGERD_o$ as 0.176Ω at 25C based on our experimental data.

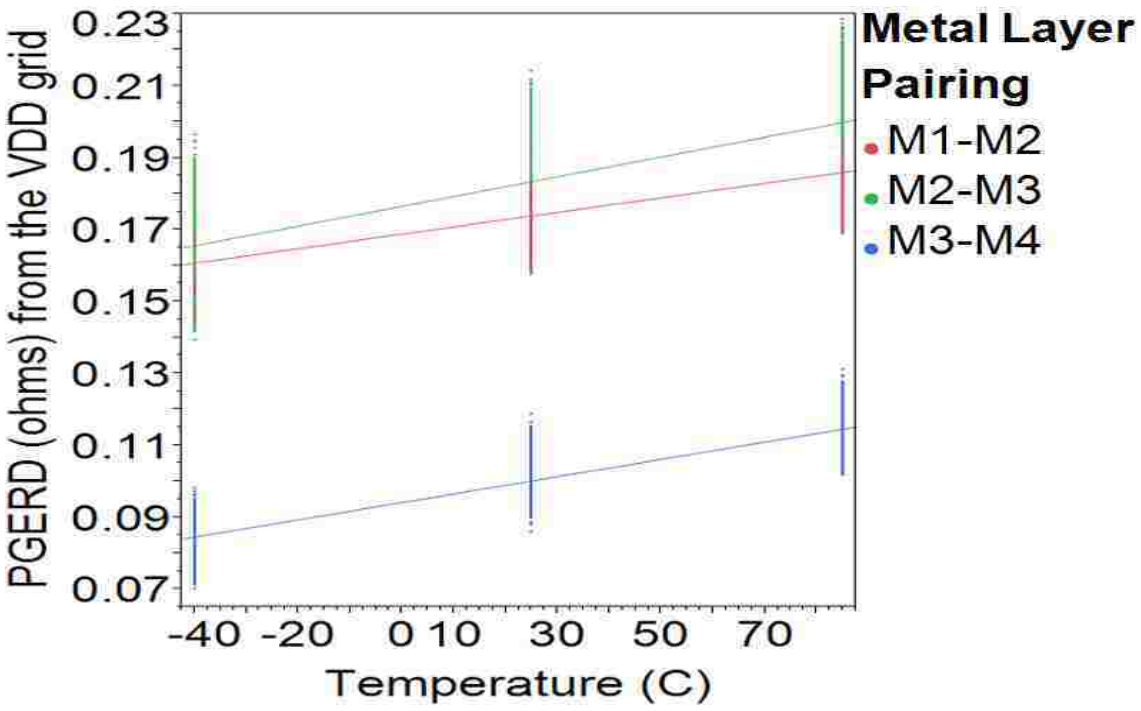


Fig. 82. PGERD versus Temperature for Chip15 V_{DD} grid

Figs. 83 and 84 illustrate the PGERD behavior as a function of TV for both V_{DD} and GND grids. It can be seen that the PGERDs increase with increasing temperature but have no dependency on changing power supply voltages.

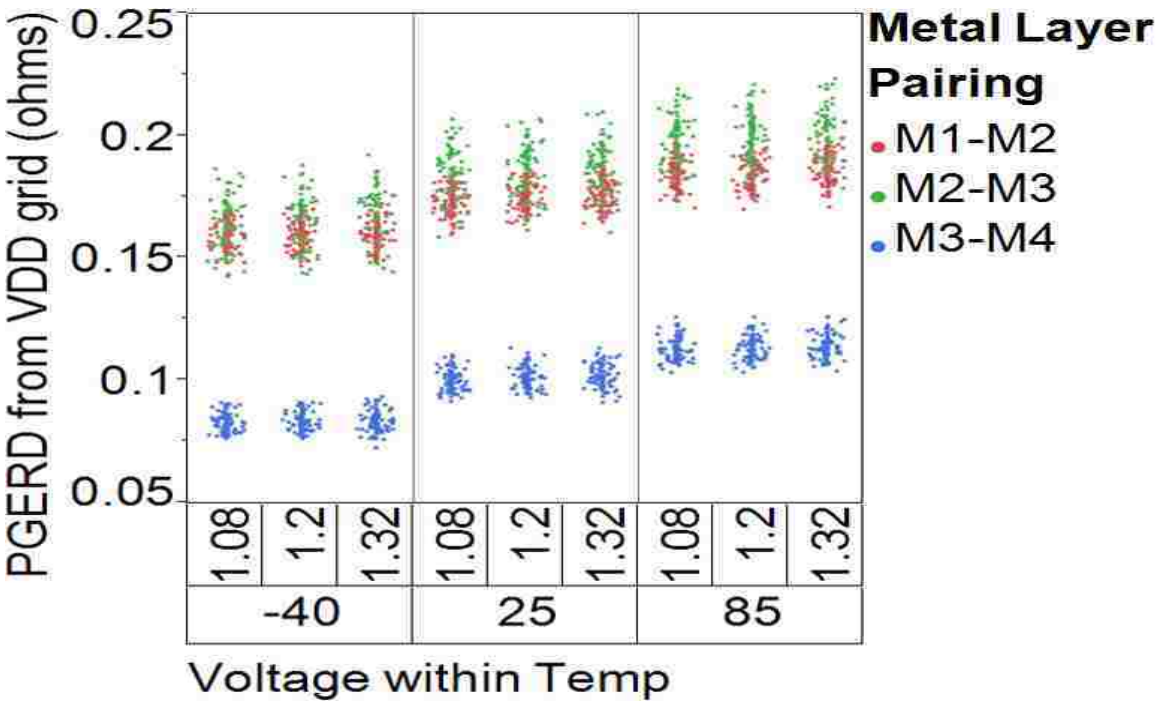


Fig. 83. PGERD versus TV for Chip15 in V_{DD} grid

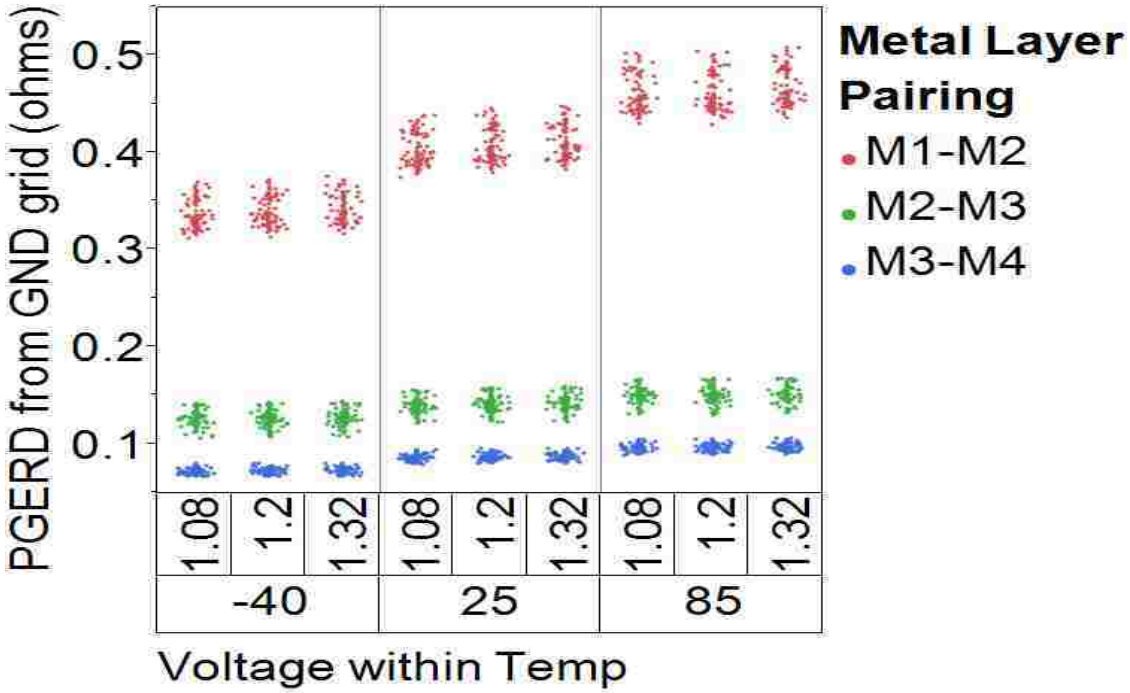


Fig. 84. PGERD versus TV for Chip15 in GND grid

It is further observed that while the M1-M2 metal layer pairing exhibits larger PGERDs for the GND grid compared to the V_{DD} grid, the PGERDs for M2-3 and M3-M4 metal layer pairings are smaller for the GND grid. The PGERDs for the GND grid are expected to be lower due to a better a current sink to GND, however the larger M1-M2 PGERD in the GND grid is due to the fact that the tap point for our M1-M2 voltage measurement for the GND grid was placed in a location that caused the inclusion of some finite routing wire resistance. This had an additive effect on the M1-M2 PGERD of the GND grid causing it to be higher than the V_{DD} grid. This also leads to a greater TV noise in the PGERDs for the M1-M2 pairing of the GND grid, as shown in Table VIII. Fig. 85 shows that unlike the PGERDs, the PGVD voltages increase with increasing temperature and voltage.

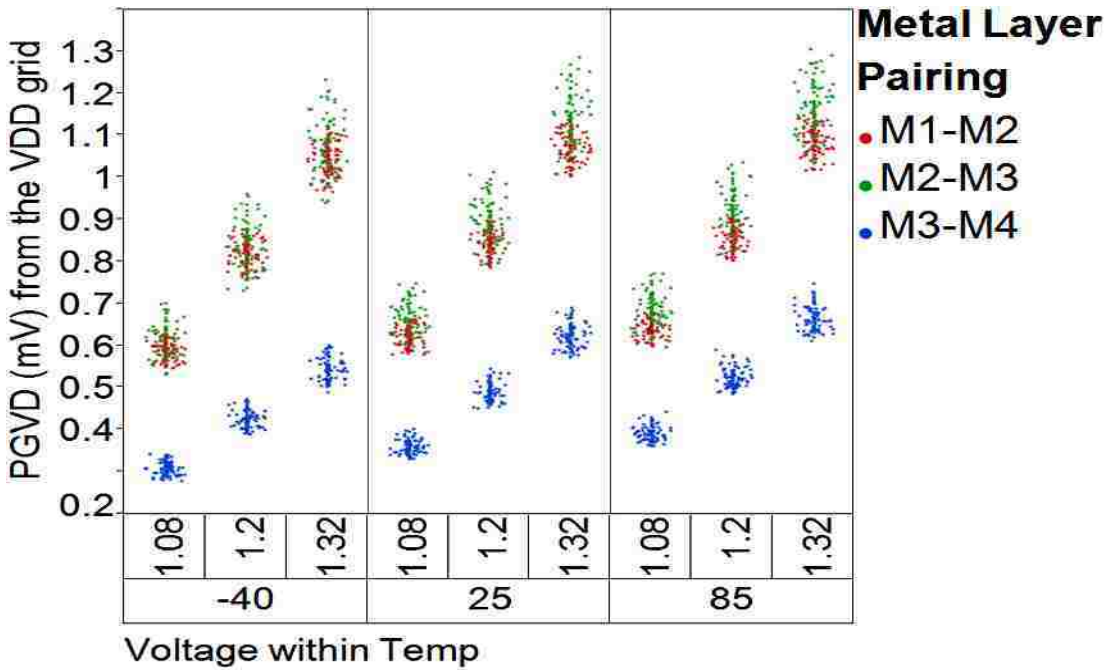


Fig. 85. PGVD versus TV for Chip15 V_{DD} grid

Fig. 86 displays the % shifts in PGERDs at different TV conditions from that of its value at enrollment conditions (25C, 1.2V). This is shown for all three metal pairings and both the V_{DD} and GND grids.

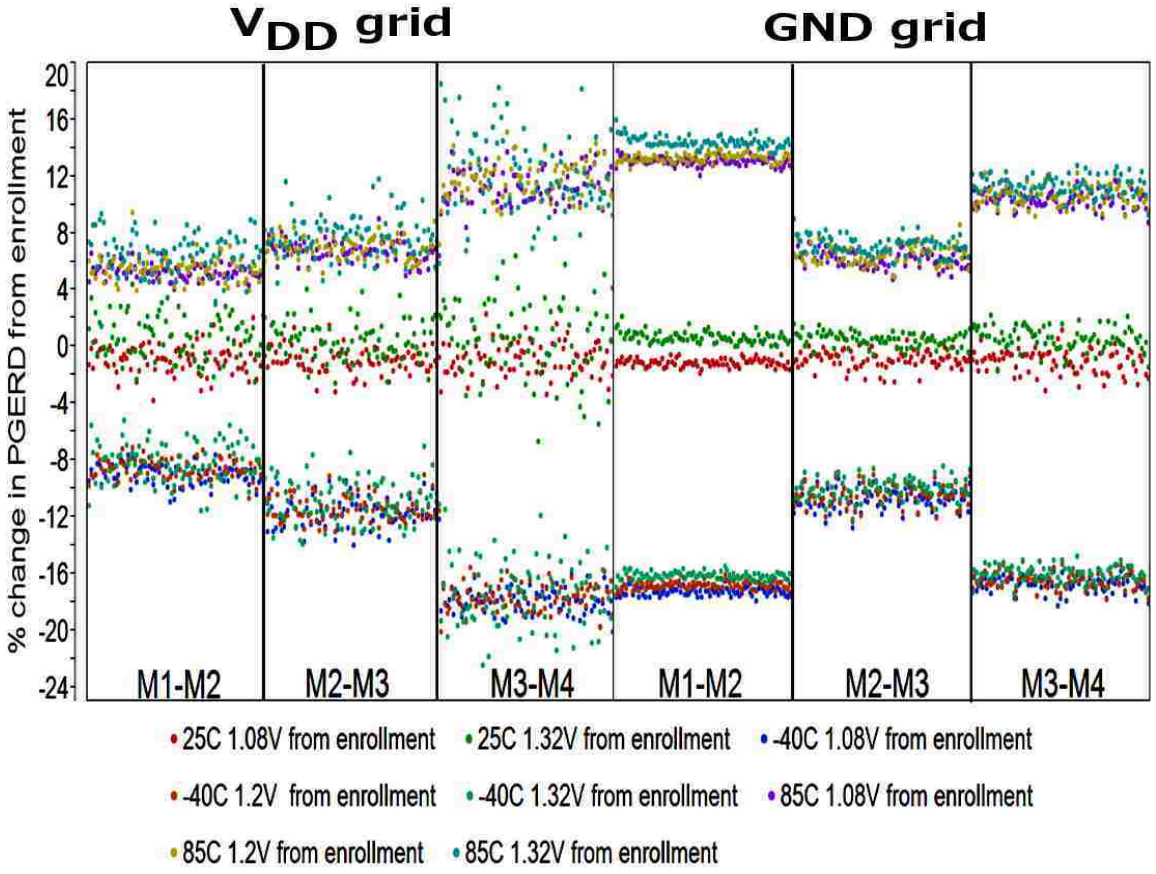


Fig. 86. % changes in PGERD for V_{DD} and GND grids of Chip15

It is noteworthy from Fig. 86 that the % changes in PGERDs for the V_{DD} grid exhibit a greater level of noise than the GND grid, and the noise level progressively increases up the metal layer stack with the M3-M4 metal layer pairing being the noisiest

for both grids. Therefore, it is these PGERDs that would be responsible for the majority of the bit flips in our bitstring.

Table VIII quantifies the TV and measurement noise for the calibrated PGERDs (calibration procedure described in Section 4.2 and [22]) and PGVDs for one of the chips. The measurement noise, defined as the standard deviation of the repeated 11 samples, accounts for 65% of the TV noise for the V_{DD} grid and only 20% for that of the GND grid. Also noteworthy is that the TV noise is 1.6X greater for the PGERDs derived from the V_{DD} grid than those derived from the GND grid, while the measurement noise is 5X greater. This is explained by the fact that the capacitance of the GND grid is much larger than the V_{DD} grid and also better distributed in the substrate.

Table VIII. TV Noise in calibrated PGERDs and PGVDs for V_{DD} and GND grids of Chip15

	TV Noise (mΩ)	**Avg TV Noise (CV)**	**Avg Measurement Noise (CV)**
Calibrated PGERDs for V_{DD} grid	M1-M2: 1.53 M2-M3: 1.53 M3:M4: 1.53	1.53 mΩ (1.07%)	1.0 mΩ (0.76%)
Calibrated PGERDs for GND grid	M1-M2: 1.90 M2-M3: 0.55 M3-M4: 0.50	0.98 mΩ (0.47%)	0.2 mΩ (0.14%)

Furthermore, the greater measurement noise for the V_{DD} grid is largely driven by the measurements at 1.32V as shown in Fig. 87, which displays the measurement noise in the PGERDs derived from the V_{DD} grid as a function of TV for the various metal layer pairings. The reason for the greater measurement noise at 1.32V is the larger voltages measured at this TV corner and the fact that the instrument's range needed to be changed to accommodate this larger range.

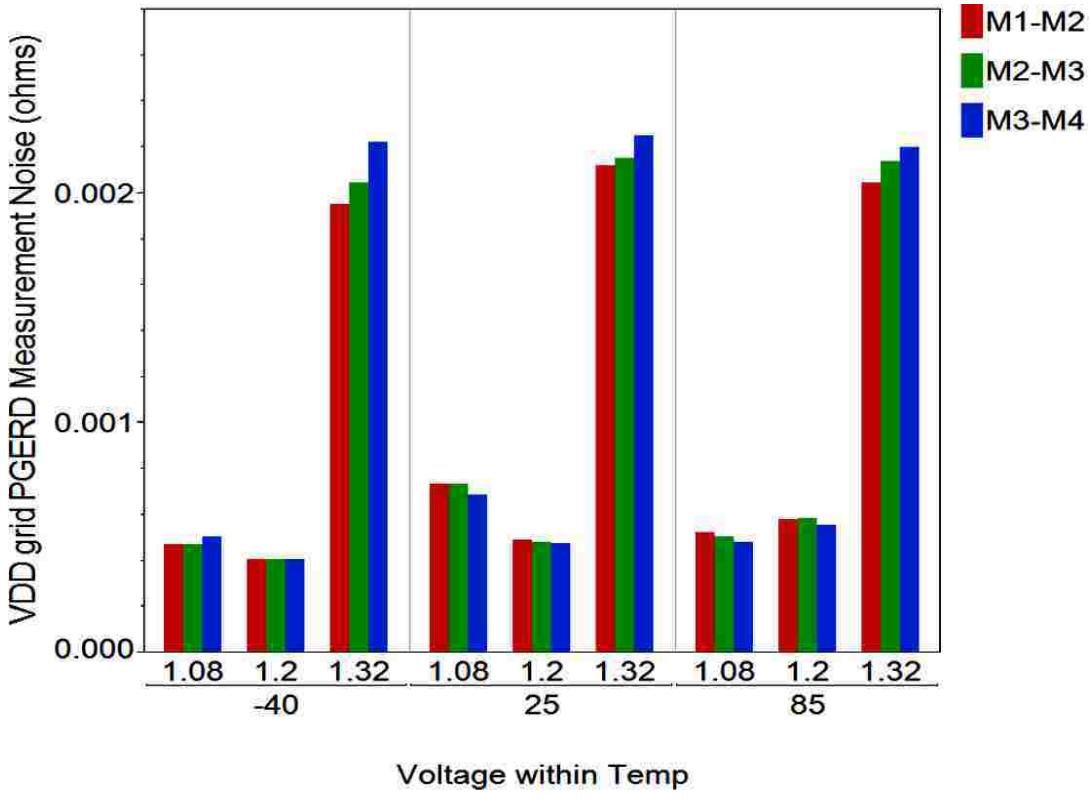


Fig. 87. PGERD measurement noise for V_{DD} grid of Chip15

7.3 I-PUF Analyses

Referring to (12), it is clear that two parameters contribute to the non-linear behavior of the VOD shifts with changing TV conditions. First is the ratio of the sum of the R_{on} of the PFET transistors to the sum of the NFET transistors at enrollment conditions and the dependency of this ratio on the VO voltage. Second is the % change of the combined PFET or NFET transistor R_{on} with TV from the R_{on} value at enrollment conditions. Figs. 88-90, and A18-A19 (in Appendix A) depict these parameters for our collected dataset from one of the chips (Chip 2). It can be seen from Fig. 88 that the R_{on} ratio changes significantly as a function of VO thus indicating that the two R_{on} ratios in the denominators of (12) could be significantly different. Fig. 89 illustrates the % changes in the combined R_{on} of the 2 PFETs (R_{on} of stacked PFET + R_{on} of PFET9p) from that at enrollment while Fig. 90 illustrates this for the combined R_{on} of the 2 NFETs (R_{on} of stacked NFET + NFET9n). From Figs. 89-90, it is clear that the % changes in R_{on} with TV also exhibit a dependency on the VO values or paths being compared. This dependency is more pronounced for the NFETs indicating that the % shifts in R_{on} in the denominator of (12) could be significantly different for the two terms. In addition to this VO dependency, the NFETs also exhibit a cross-over between the 85C, 1.32V and the -40C, 1.2V curves in Fig. 90 indicating that the VO shifts for these two TV corners could be even more unpredictable depending on the paths being compared. Also, it is evident that the % changes in R_{on} are greater for the NFETs than the PFETs for any given TV, with the % changes in the NFET R_{on} being the greatest at 1.08V. Lastly, another

noteworthy observation from Figs. 89-90 is that there is significantly greater variation in the % changes in R_{on} at 1.08V regardless of the operating temperature. This increased variation at 1.08V is a major cause of the larger number of bit flips seen at this voltage. Explanations and causes for these observations are provided later in this section.

All these aforementioned factors lead to disproportionate and unpredictable shifts in VO with TV. Depending on the VOs being compared, if the R_{on} ratio in the

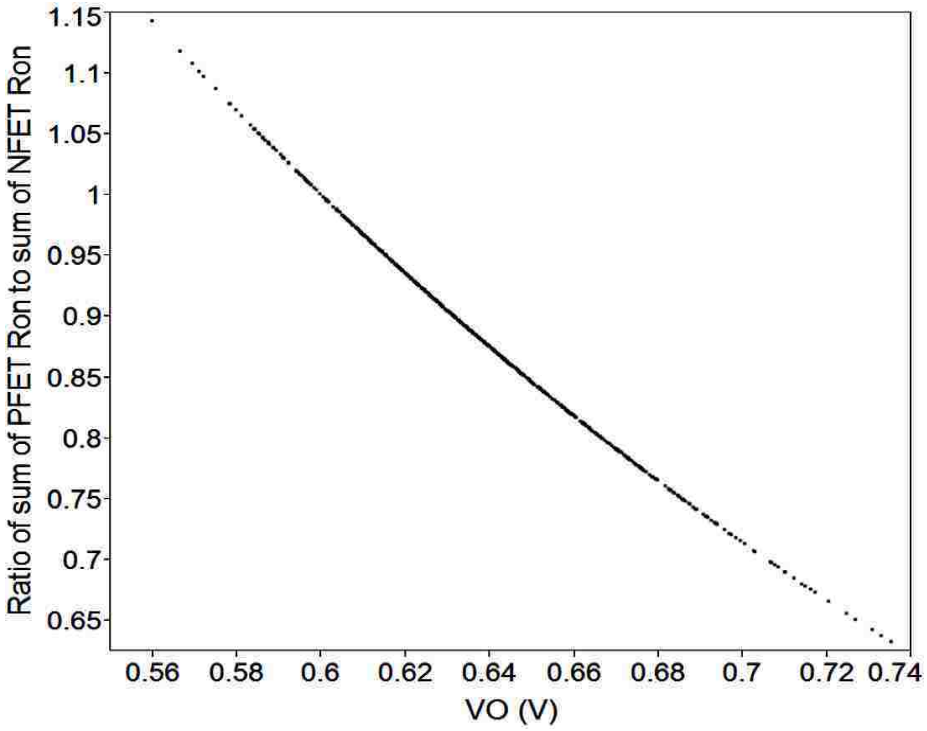


Fig. 88. Ratio of the sum of the R_{on} of the PFET transistors to sum of NFET transistors vs. VO at 25C, 1.2V for Chip2

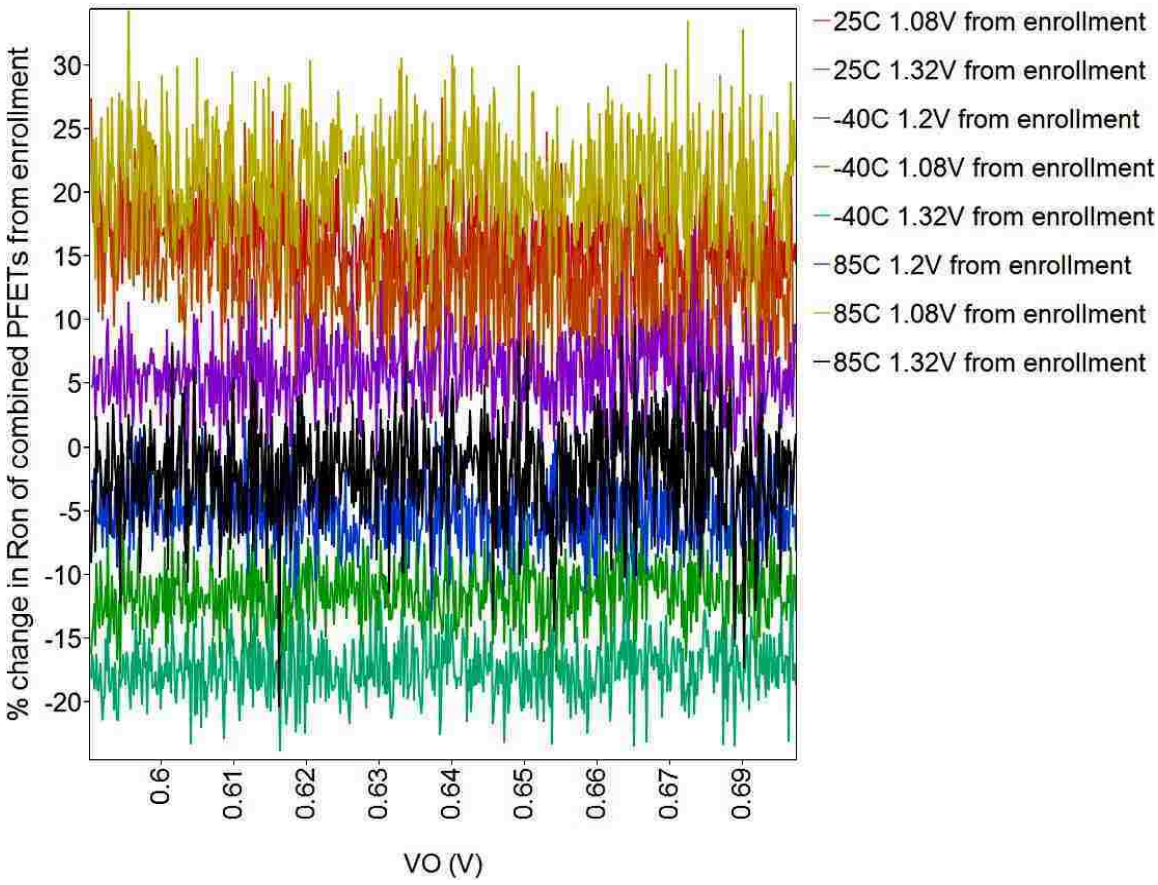


Fig. 89. % change in combined R_{on} of the 2 PFETs versus VO for Chip2

denominator of the first term in (12) happens to be significantly different than that in the denominator of the second term, the R_{on} % change terms in (12) play a weaker role in determining the shift in VO with TV. However, if the R_{on} ratios are similar, then the differences between the % change in combined R_{on} for the 2 PFETs and that of the 2 NFETs for a given VO will play a dominating role in dictating the shift in VO with TV. These are the key reasons for the bit flips seen when comparing two different closely-spaced VO values at all 9 TV corners.

It should be noted that, for any given path of the I-PUF primitive, the stacked NFET and PFET transistors operate in the linear region while the PFET9p and NFET9n transistors operate in the saturation region. All our transistors operate in the larger V_{GS} region so we generally see an increase in R_{on} with increasing temperatures for a fixed V_{DD} voltage. On the other hand, a decrease in R_{on} with increasing V_{DD} voltages at a fixed temperature is seen for the transistors in our primitive.

Due to these changes in R_{on} with TV, the VO voltage exhibits some interesting behaviors as a function of changing TV. The I-PUF displays a self-compensating

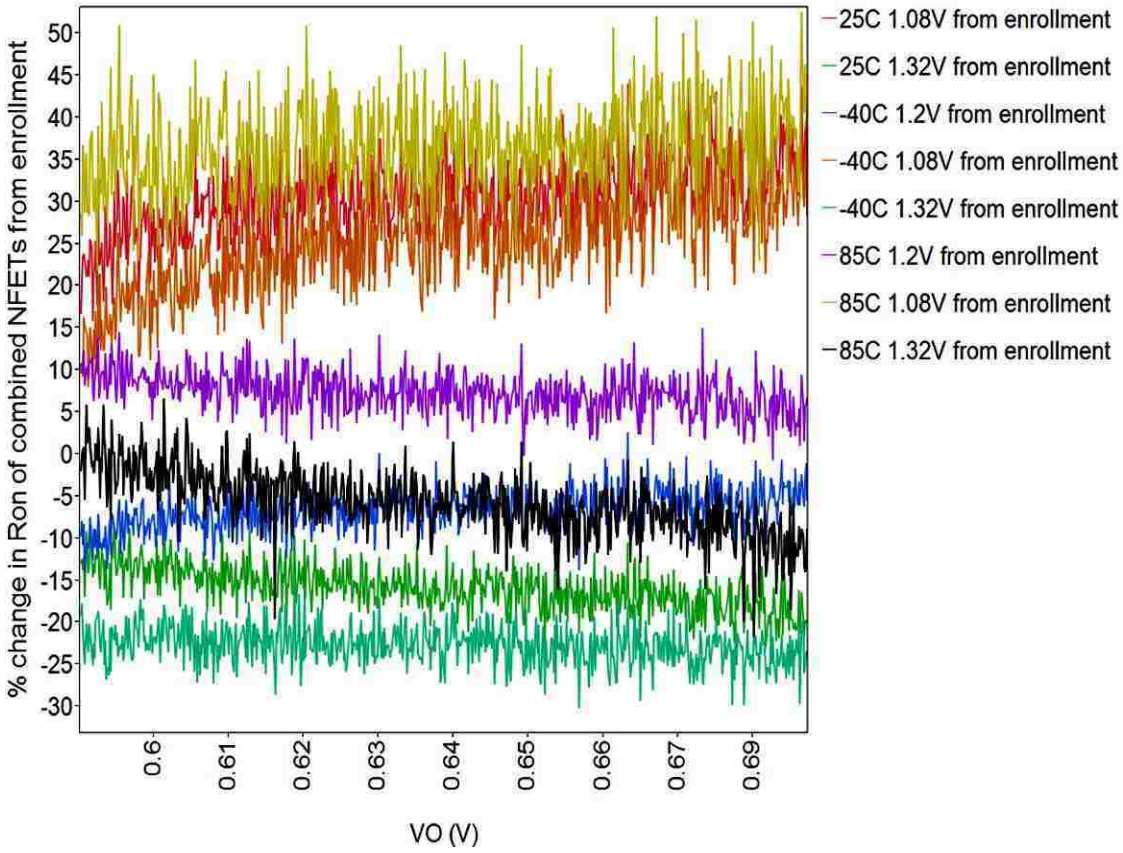


Fig. 90. % change in combined R_{on} of the 2 NFETs versus VO for Chip2

characteristic as a function of changing temperature. That is, as the temperature changes for a fixed V_{DD}, the R_{on} of each of the four transistors change in an unequal fashion, however, the average ratio of the combined R_{on} of the 2 PFETs to that of the 2 NFETs stays fairly constant rendering little to no change in the VO of the I-PUF with changing temperature, as displayed in Figs. 91 and 92. On the other hand, as seen in Fig. 91, VO does increase with increasing V_{DD} but less than the expected V_{DD} scaling factor of ~1.1X. This stems from the fact that we see increasing ratios of combined resistance of the 2 PFETs to that of the 2 NFETs with increasing V_{DD}, as depicted in Fig. 92. To be precise, the combined R_{on} of the 2 PFETs decrease at a much smaller rate than that of the 2 NFETs with increasing V_{DD}, thus causing the increasing ratios seen in Fig. 92. Referring to (10), it is clear that these increasing ratios mitigate the effect of an increasing V_{DD}, thus resulting in a lower than expected increase in VO.

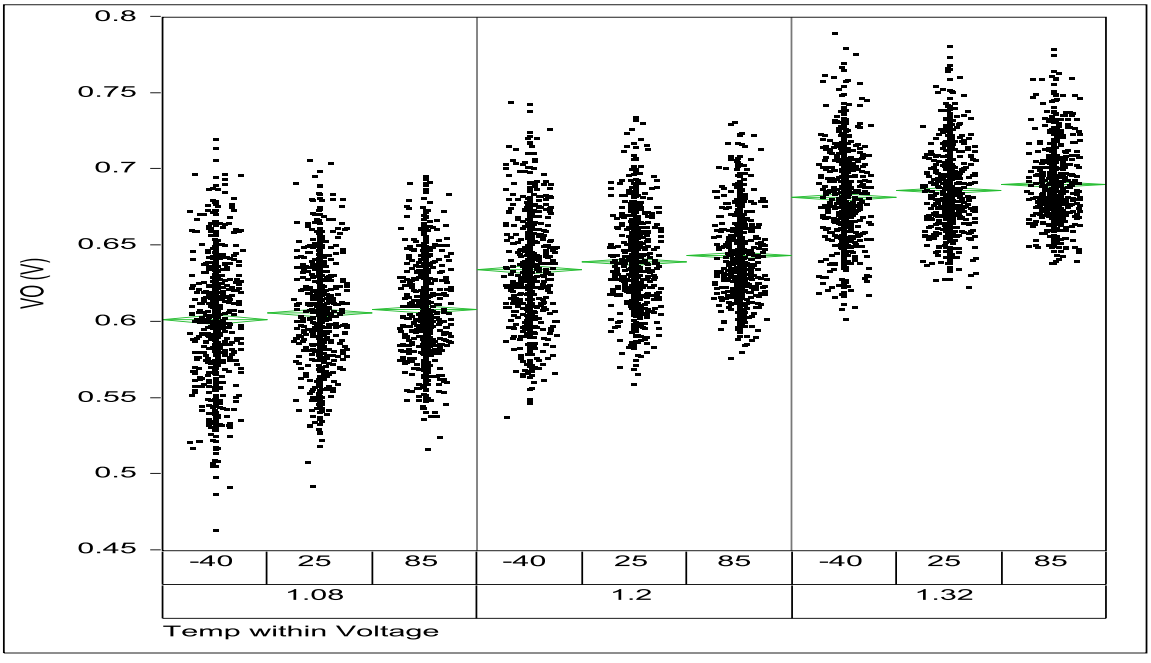


Fig. 91. VO versus TV for Chip2

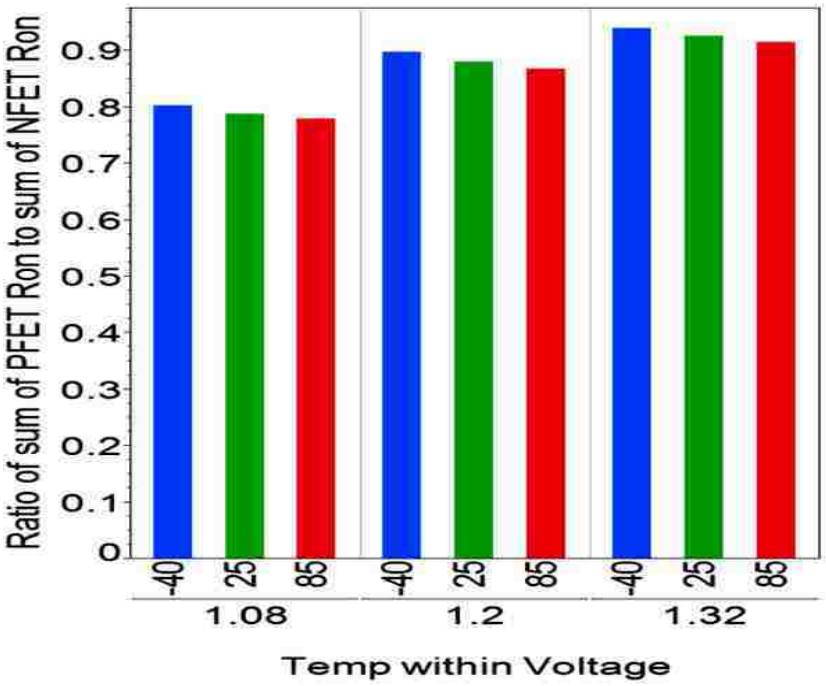


Fig. 92. Ratio of combined path resistance of PFETs to NFETs versus TV for Chip2

Investigating the larger rate of decrease in the combined R_{on} of the 2 NFETs with increasing V_{DD} reveals that the NFET9n R_{on} plays a large role in this. The NFET9n R_{on} decreases at a rate of approximately 1.7X that of the other transistors as a function of increasing $V_{DD,}$ thus rendering the combined R_{on} of the 2 NFETs to decrease at a rate of 1.7X the combined R_{on} of the 2 PFETs. This is illustrated in Fig. 96 and is also reflected in Table IX (a) which outlines the average TV noise of R_{on} in each of the four transistors in our primitive. The reason that the NFET9n R_{on} shifts so much more with changing V_{DD} than the other transistors is due to the fact that unlike any of the other three transistors, NFET9n experiences voltage changes on 3 of its terminals (G, S, D) with changing $V_{DD.}$ This has a greater effect in modulating the R_{on} of NFET9n with changing V_{DD} compared

to the other three transistors. It is also noteworthy that although the NFET9n R_{on} changes at a much larger rate as a function of V_{DD} compared to the other 3 transistors, its temperature dependency is in line with that of PFET9p. From Figs. 93 through 96, it is observed that the change in the R_{on} of the stacked transistors with changing temperature is larger than that of NFET9n and PFET9p. As explained in section 7.1, this is because the stacked transistors are operating in the linear region and at a larger V_{GS} and smaller V_{DS}, where the R_{on} changes with temperature are larger than at the saturation region (smaller V_{GS} and larger V_{DS}). However, the difference in TV noise of NFET9n R_{on} from that of the other transistors is mostly driven by its larger dependency on V_{DD}. It should also be noted from Figs. 93 through 96 that for the saturated transistors NFET9n and PFET9p, the changes in R_{on} with temperature get smaller with decreasing V_{DD}. The reason for this is similar to what was seen and explained in Section 7.1 for the TG-PUF. A lower V_{DD} leads to a lower operating V_{GS} for all the transistors and the saturated transistors operate at a lower V_{GS} than the linear transistors, and therefore closer to the V_{GS} inflection point below which the temperature dependency of I_{DS} reverses (refer to the I_{DS} vs. V_{GS} curves of PFET9p and NFET9n in Figs. A15 and A17 of Appendix A). The R_{on} and I_{DS} changes with temperature get smaller the closer the transistor gets to operating at their respective inflection points. As can be seen from Fig. A15 and A17, the change in I_{DS} with changing temperature gets smaller with decreasing V_{GS}. Decreasing V_{DD} is what results in the decreasing V_{GS} in Figs. A15 and A17, and as V_{DD} decreases, the V_{DS} of PFET9p and NFET9n is also not constant and is changing. Therefore, these changing behaviors of V_{DS} and I_{DS} with temperature at different V_{DD} are what cause the R_{on} dependency on

temperature to change with V_{DD}. As mentioned in Section 7.1, noteworthy is the fact that the inflection point of where R_{on}'s dependency on temperature reverses will not match the inflection point of where I_{DS}'s dependency on temperature reverses. This is because the V_{DS} of the transistor also changes with changing V_{DD} and is a central component in determining the R_{on}. That is why when analyzing Fig. A17, it appears that the inflection point of the NFET9n R_{on} should be somewhere slightly below a V_{DD} of 1.08V, however as is obvious from Fig. 96, it is somewhere well below a V_{DD} of 1.08V after factoring in changing V_{DS}.

It is also evident from Fig. 91 and Figs. 93 - 96 that the R_{on} and VO variation gets progressively worse with decreasing V_{DD}, confirming the observation from Figs. 89 - 90. Additionally, the measurement noise in R_{on} also gets worse with decreasing V_{DD} and temperature as observed from Fig. 97 but is negligible compared to the TV noise as shown in Table IX (a). The R_{on} of the PFET9p and NFET9n transistors exhibit a higher measurement noise value as shown in Fig. 97 but a quick look at Table IX (a) shows that this is due to higher R_{on} values for these transistors and therefore the CV % of the measurement noise in R_{on} is very similar for all four transistors. It is clear from Table IX (a) that the combined R_{on} of the NFETs is more sensitive to TV variations than that of the PFETs. This is because mobility variations with temperature cause NFETs to exhibit a lot higher noise in R_{on} due to temperature variations than PFETs (see Fig. A10 in Appendix A). Additionally, the NFET9n R_{on} is more sensitive to V_{DD} variations than any of the other transistors and thus, plays a bigger role in determining the TV sensitivity of the I-PUF.

From Table IX (a), it is indicative that the TV noise in the R_{on} of the stacked PFET and PFET9p are very similar. This is explained by the fact that the noise in R_{on} due to V_{DD} variations is almost identical due to the fact that both the stacked PFET and PFET9p have variation on 3 (S, D, B) of their terminals due to change in V_{DD}. However, as expected, the noise due to temperature variations is slightly larger for the linear region operated stacked PFET than the saturation region operated PFET9p and this is what causes the slightly larger overall TV noise of the stacked PFET. Similarly, the NFET9n has a much larger noise in R_{on} due to V_{DD} variations (G, D, S affected by V_{DD} variation) compared to the stacked NFET (G, D affected by V_{DD} variation) however, the stacked NFETs have a larger noise in R_{on} due to temperature variations due to the fact that they operate in the linear region. The noise due to V_{DD} is comparatively much larger in the NFET9n transistor thus causing the higher overall TV noise in R_{on} of the NFET9n transistor as compared to the stacked NFET transistor.

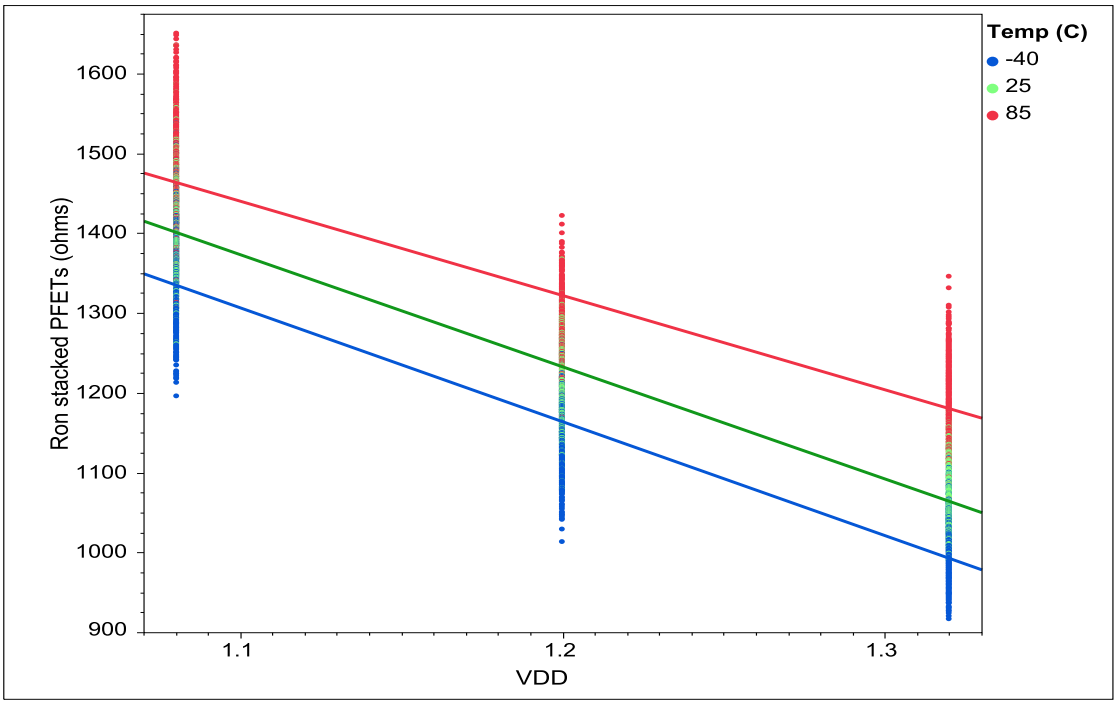


Fig. 93. R_{on} versus TV for Chip2 stacked PFETs

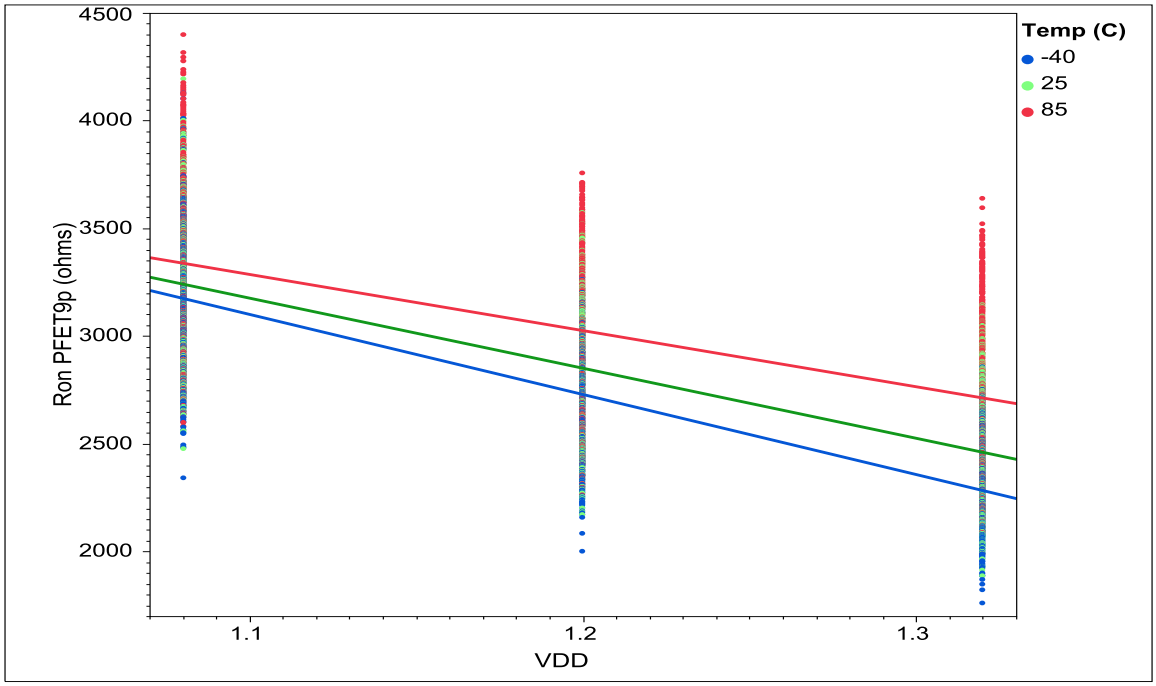


Fig. 94. R_{on} versus TV for Chip2 PFET9p

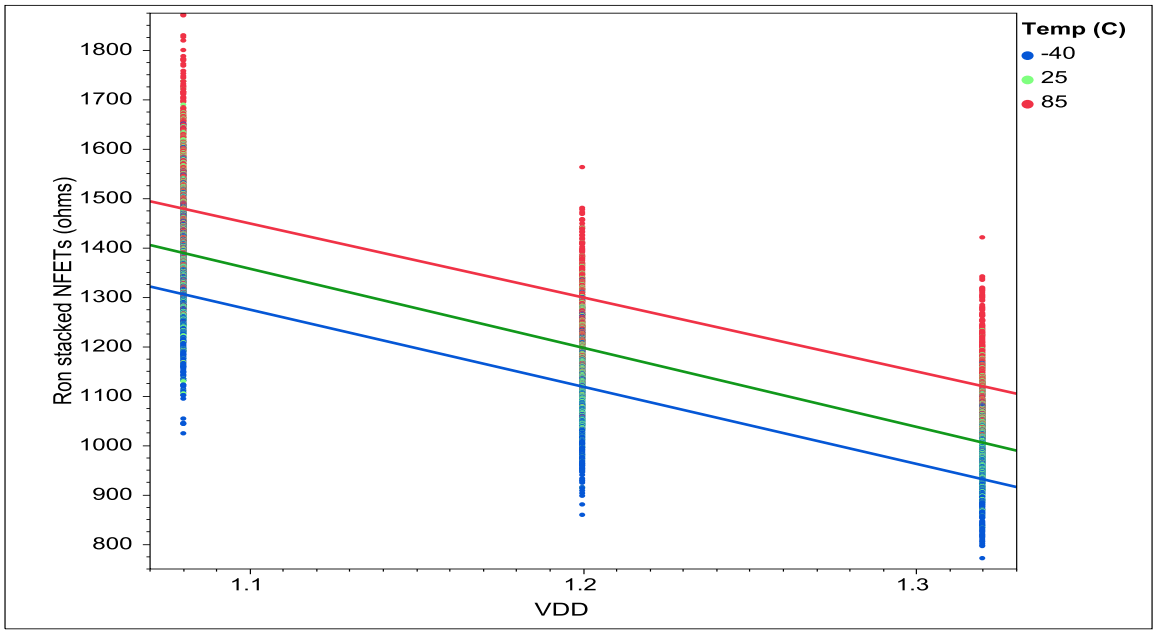


Fig. 95. R_{on} versus TV for Chip2 stacked NFETs

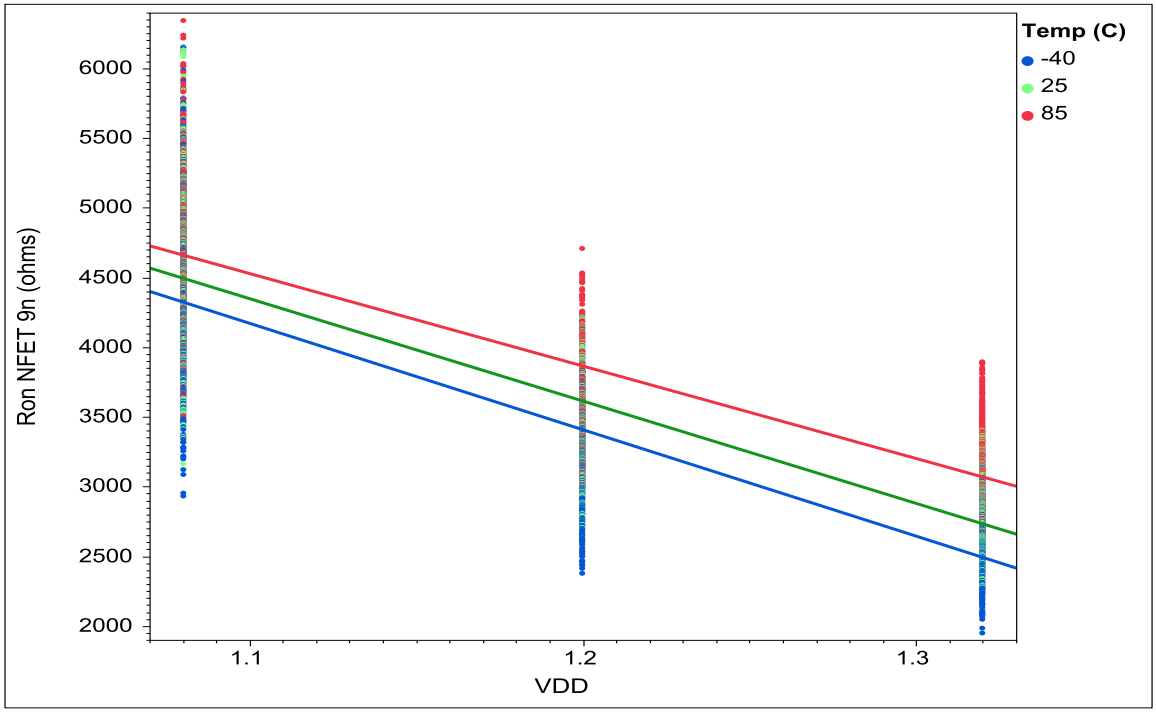


Fig. 96. R_{on} versus TV for Chip2 NFET9n

Table IX. Average TV and measurement noise in (a) R_{on} of Chip2 (b) voltages of Chip2

	CV of TV Noise (%)	**TV noise (Ω)**	**Measurement noise (CV)**
Stacked PFET R_{on}	13.6	169	0.97 Ω (0.08%)
PFET9p R_{on}	13.02	373	3.44 Ω (0.12%)
NFET9n R_{on}	21.83	796	3.46 Ω (0.09%)
Stacked NFET R_{on}	15.17	183	0.59 Ω (0.05%)
Combined R_{on} of 2 PFETs	12.8	528	-
Combined R_{on} of 2 NFETs	20.00	976	-

(a)

	TV noise (CV)	**Measurement noise (CV)**
VO	36mV (15.6%)	0.47mV (0.07%)
VOD	6mV (16.6%*) * normalized to VO TV noise	-

(b)

From Table IX (b), it can be seen that the measurement noise is negligible compared to the TV noise. Just as in the TG-PUF, the difference operation of the voltages leads to a lower magnitude of TV noise. A detailed explanation of what causes this and its implications are provided in Section 7.1 and are directly applicable to the I-PUF characteristics observed in Table IX (b). Making the assumption that the I-PUF operates with TV noise levels similar to the NFET and PFET TG-PUF and considering the VO voltages are around 620mV (about half-way between the NFET TG-PUF and PFET TG-PUF TGV voltages), we would expect the TV noise in VOD to be within the range of TGVD TV noise of the PFET TG-PUF (0.44mV) and the NFET TG-PUF (0.9mV). However, the TV noise in VOD is 6mV indicating the much larger TV noise levels of the I-PUF as compared to the TG-PUF. The TV noise in VOD is what dictates the sensitivity of bit flips to TV variations in the unstable bitstring, as it is the differences of the VODs that are used to generate the bitstring. This explains the higher intra-chip HD of the unstable voltage-generated bitstrings of the I-PUF (6.18%) as compared to that of the TG-

PUF (5.11%). This result was expected due to the incorporation of 2 PFETs and 2 NFETs in the I-PUF primitive versus just 2 PFETs or 2 NFETs in each of the TG-PUF primitives.

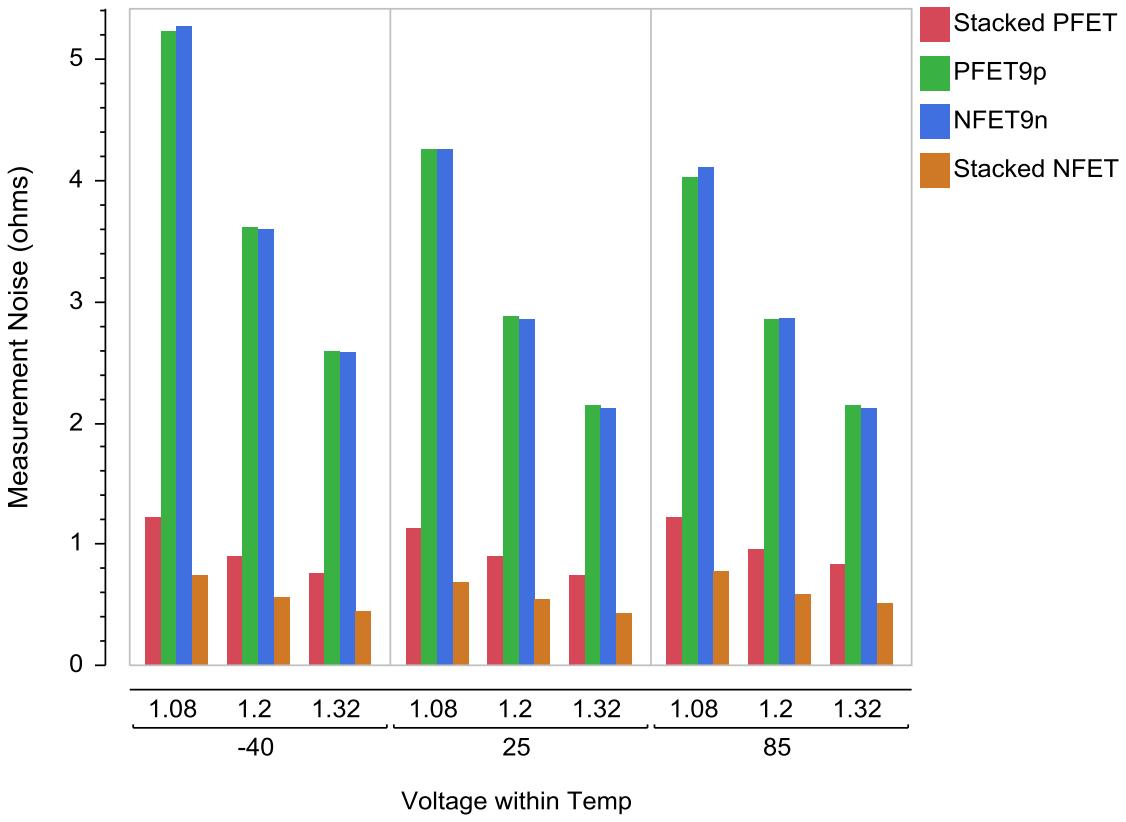


Fig. 97. R_{on} measurement noise versus TV for Chip2

An interesting observation emerges from the analyses of the TV noise in VO for the I-PUF. From Fig. 98, it can be seen that the TV noise in VO increases with decreasing VO. This occurs because a decrease in VO translates to a smaller combined R_{on} of PFETs to NFETs ratio (see Figs. 91 and 92). This implies that the TV noise in the combined R_{on} of NFETs, which is larger than that in the combined R_{on} of PFETs, plays a larger part in

determining the overall TV noise in VO for decreasing VO. This leads to the observed increase in the overall VO TV noise with decreasing VO. This indicates that we could make our I-PUF more resilient to TV variations by increasing the combined R_{on} of PFETs to NFETs ratio.

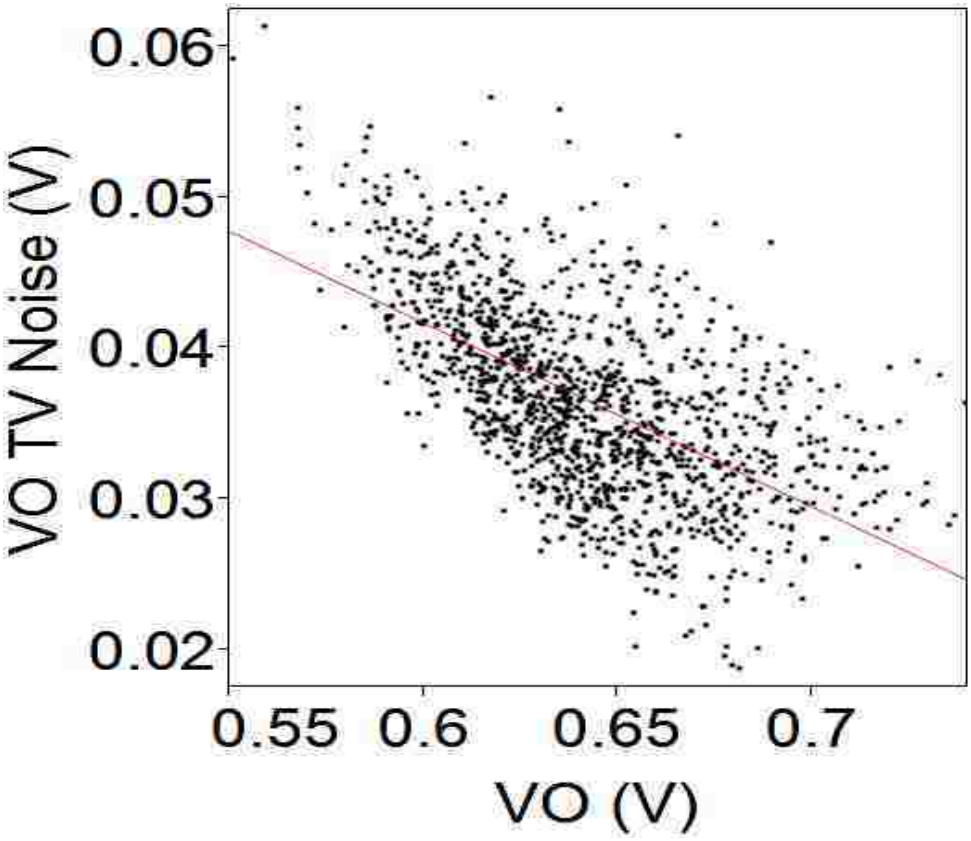


Fig. 98. VO TV noise versus VO for Chip2

Chapter 8

Simulation Results

All the PUF primitives were simulated using Cadence Virtuoso® Design Environment version IC6.1.5.500 and Virtuoso® Analog Design Environment Spectre IC6.1.5.500. This chapter presents the simulation results and compares and contrasts these results with those obtained from hardware experimental data. All model files, cell libraries, etc. used in the design of the chips were used in the simulation. It should be noted that in the rest of this chapter, when stating V_{GS} and I_{DS} for PFETs, it is implied that it is actually being referred to $|V_{GS}|$ and $|I_{DS}|$ for PFETs.

8.1 TG-PUF

The NFET and PFET primitives of the TG-PUF were simulated separately to better observe the effects of varying temperature and voltage conditions on each primitive. Figs. 99 and 100 illustrate the schematic of the NFET and PFET primitives, respectively, used in the simulation. These were also the same primitives used in the chip design. It should be noted from Fig. 100 that the PFET primitive is actually constructed from 4 PFETs of the same size. The top 2 PFETs, 1p1 and 1p2, are connected in parallel and their total parallel resistance is denoted later in this section as 1p, while the bottom 2 PFETs, 9p1 and 9p2, are connected in parallel and their total parallel resistance is denoted later in this section as 9p. It should be obvious from the schematics below that NFET1n

is what is referred to as the stacked NFETs previously in this document while PFET1p is the stacked PFETs. Similarly, what is referred to NFET9n and PFET9p here is referred to NFET9 and PFET9 in section 7.1, respectively.

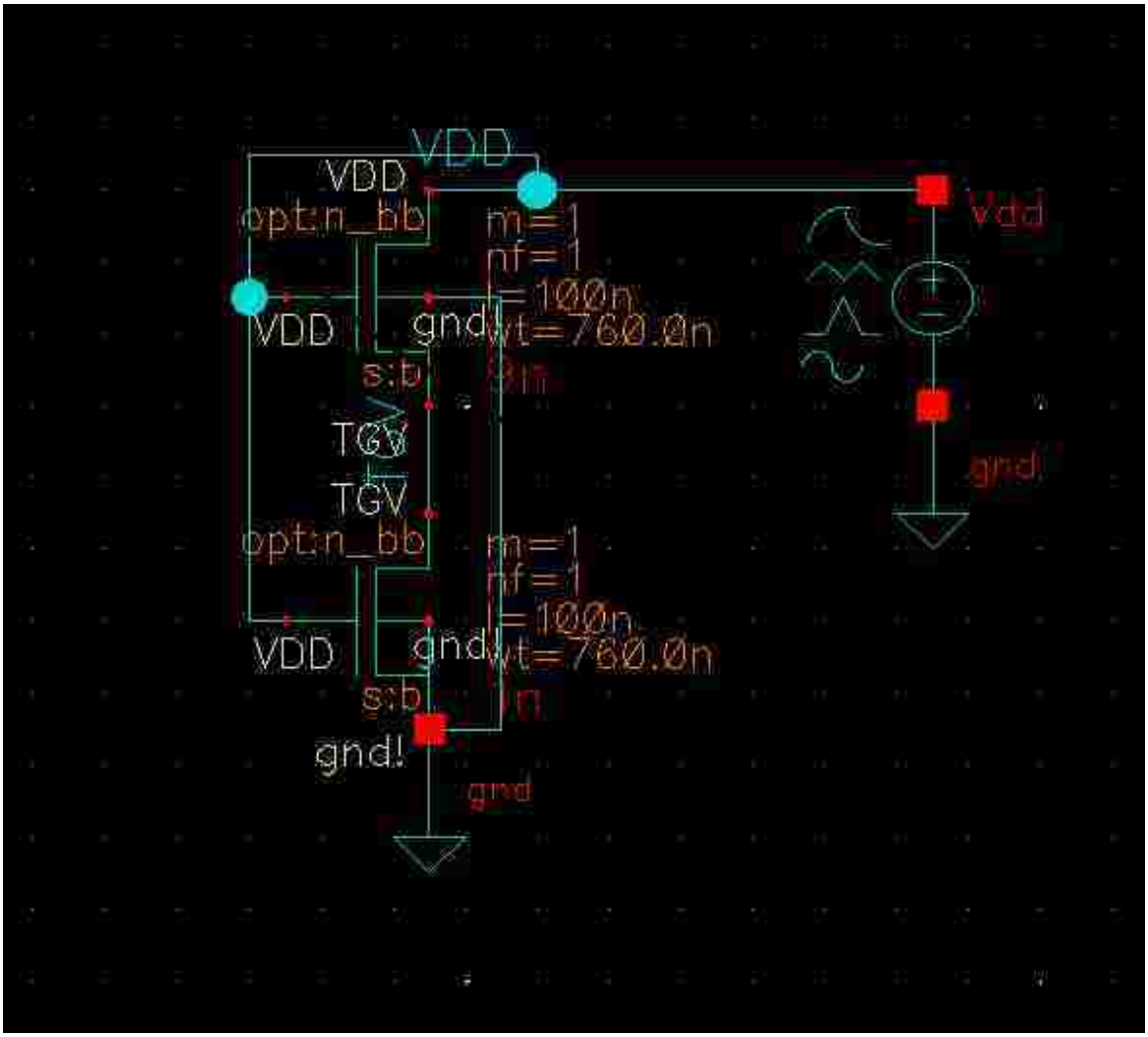


Fig. 99. Schematic of NFET TG-PUF primitive

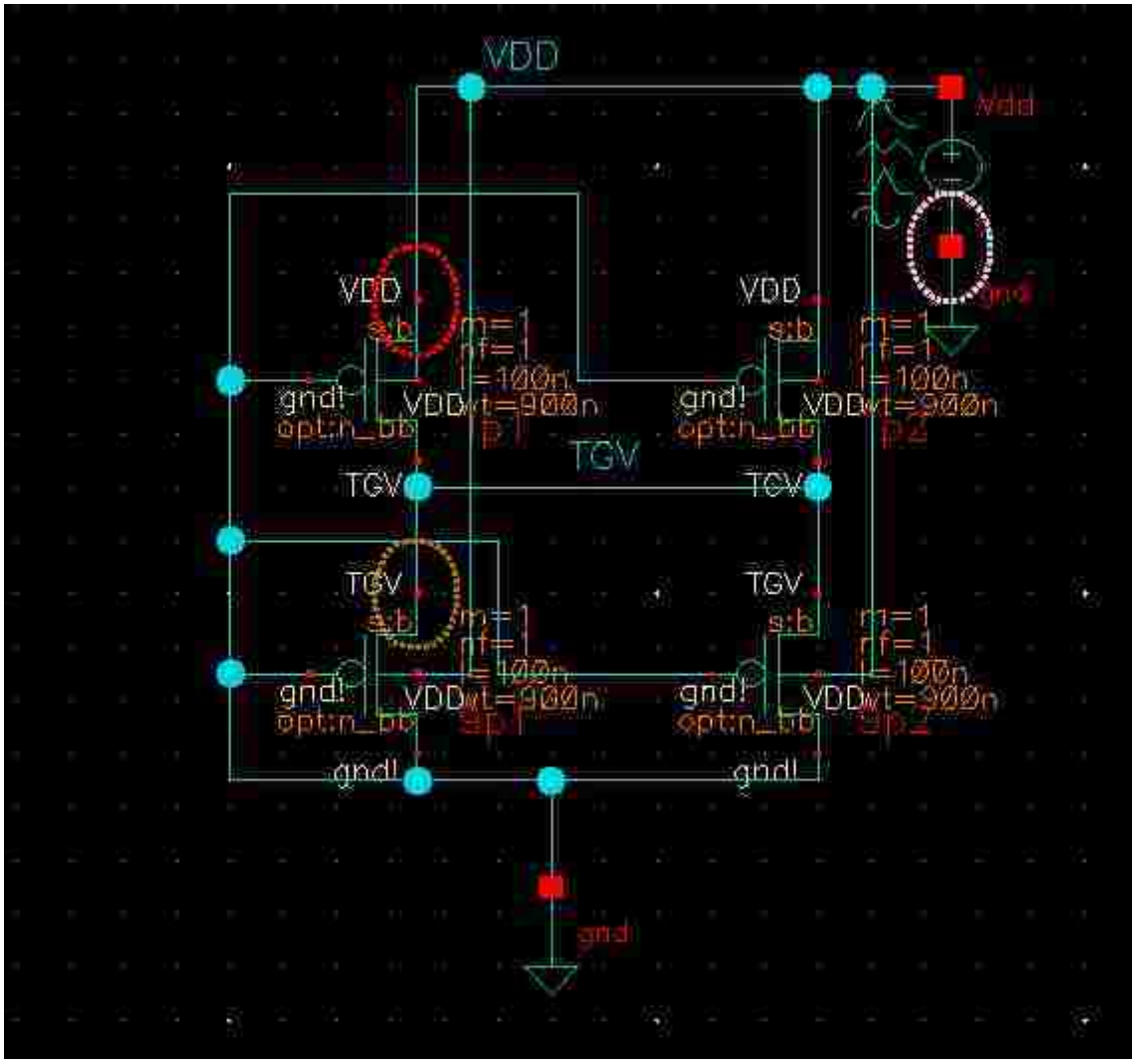


Fig. 100. Schematic of PFET TG-PUF primitive

Next, the V_{DD} voltage is swept from 1.08V to 1.32V to yield the V_{DD} corner information at 1.08V, 1.2V, and 1.32V (1.2V +/- 10%). This V_{DD} sweep is conducted in conjunction with a parametric analysis sweep of the temperature at -40C, 25C, and 85C. This yields the 9 corner TV simulation data. It should be noted that in the rest of this

chapter, the -40C data is color coded as blue, the 25C data as green, and the 85C data as red.

Fig. 101 depicts the TGV data at all 9 TV corners for the NFET TG-PUF primitive. The x-axis represents the V_{DD} voltage sweep while the 3 different temperatures are color coded. It can be seen that the TGV voltage increases with increasing temperature and increasing V_{DD}.

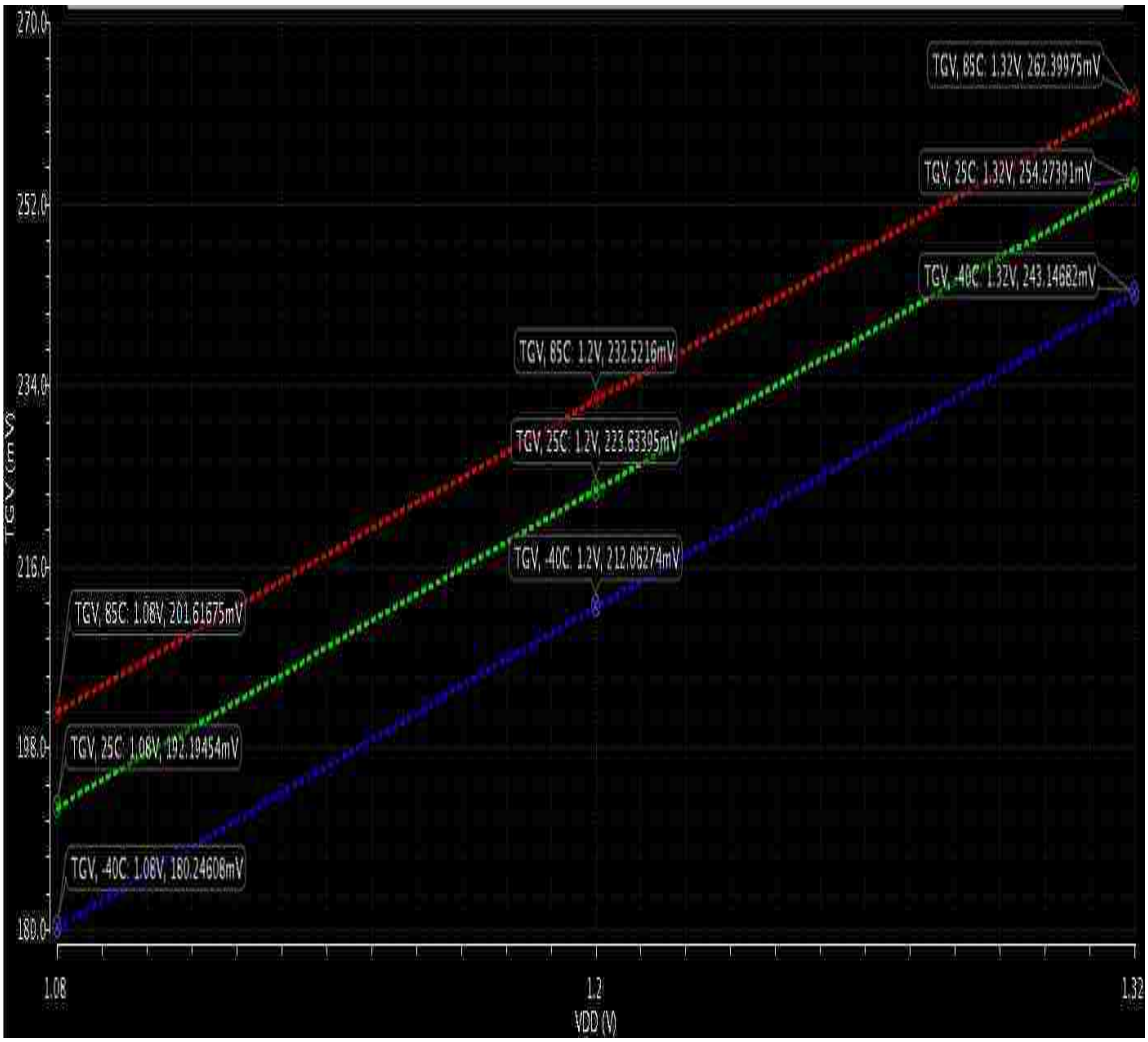


Fig. 101. Simulation results for TGV vs. TV for the NFET TG-PUF

From hardware experiments, it was also seen that TGV increased with increasing V_{DD} and temperature and a quick review of Section 7.1 confirms that. It should be noted that the data plotted in Section 7.1 is for all 85 SMCs for a single chip. The spread in the TGV data for a given TV corner exemplifies the spread across the SMCs. As can be seen, the simulation results match the experimental results for all the 9 TV corners and fall within the spread of the hardware results.

Fig. 102 depicts the simulation results for the PFET TG-PUF. Again, good agreement between simulation and experimental results was seen when comparing to Section 7.1.

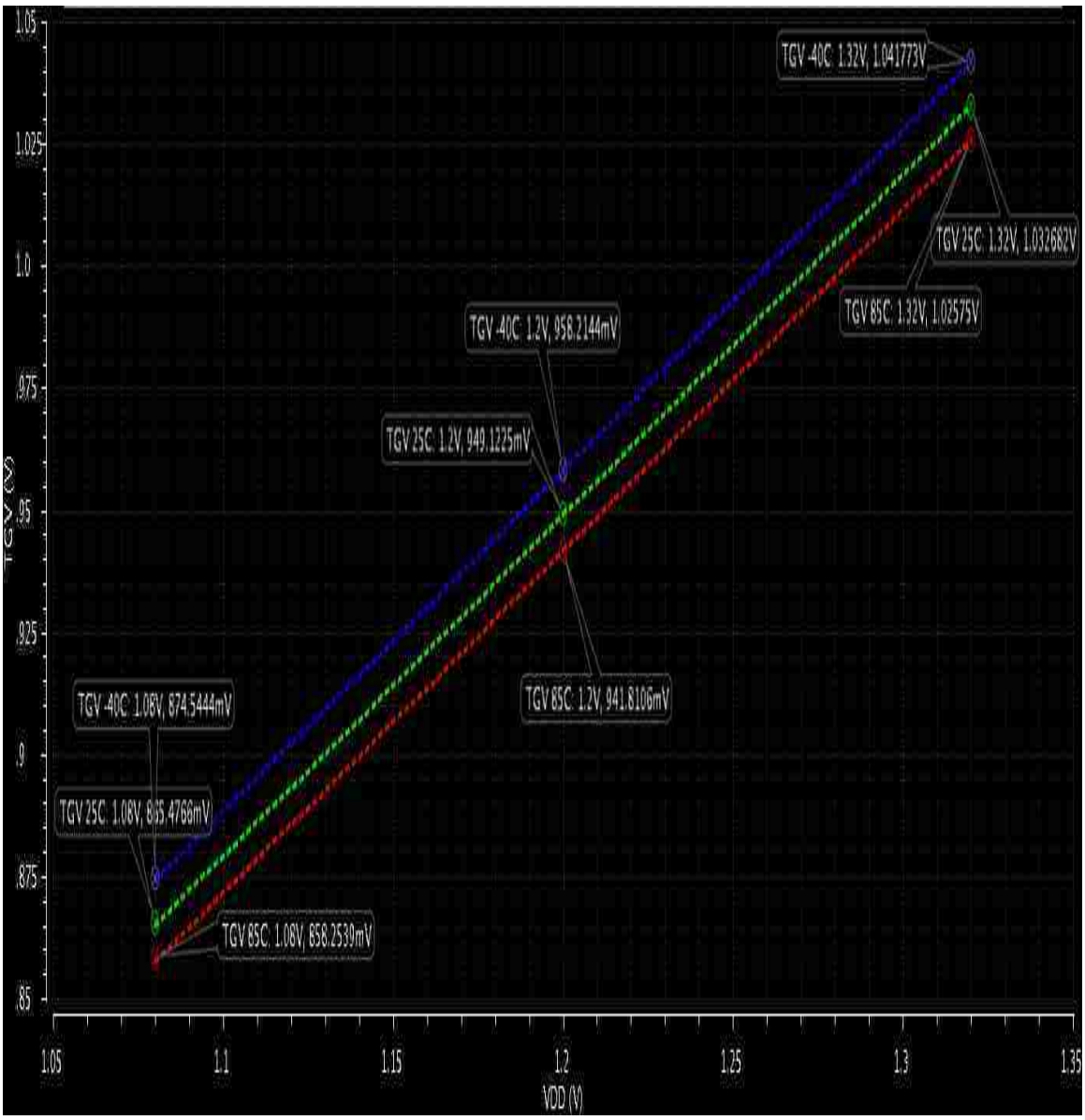


Fig. 102. Simulation results for TGV vs. TV for the PFET TG-PUF

Next, the relationship of I_{DS} versus TV is simulated for the NFET and PFET TG-PUF primitives. This is depicted in Figs. 103 and 104. It should be noted from Fig. 104 that two sets of curves have been shown for the PFETs. The bottom set of curves are the I_{DS} vs TV curves for each of the four individual PFETs in the PFET TG-PUF schematic

whereas the top set of curves are the total I_{DS} through the primitive (double of the bottom set of curves). I_{DS} increases with decreasing temperature and increasing V_{DD}, and this matches the experimental results from Section 7.1.

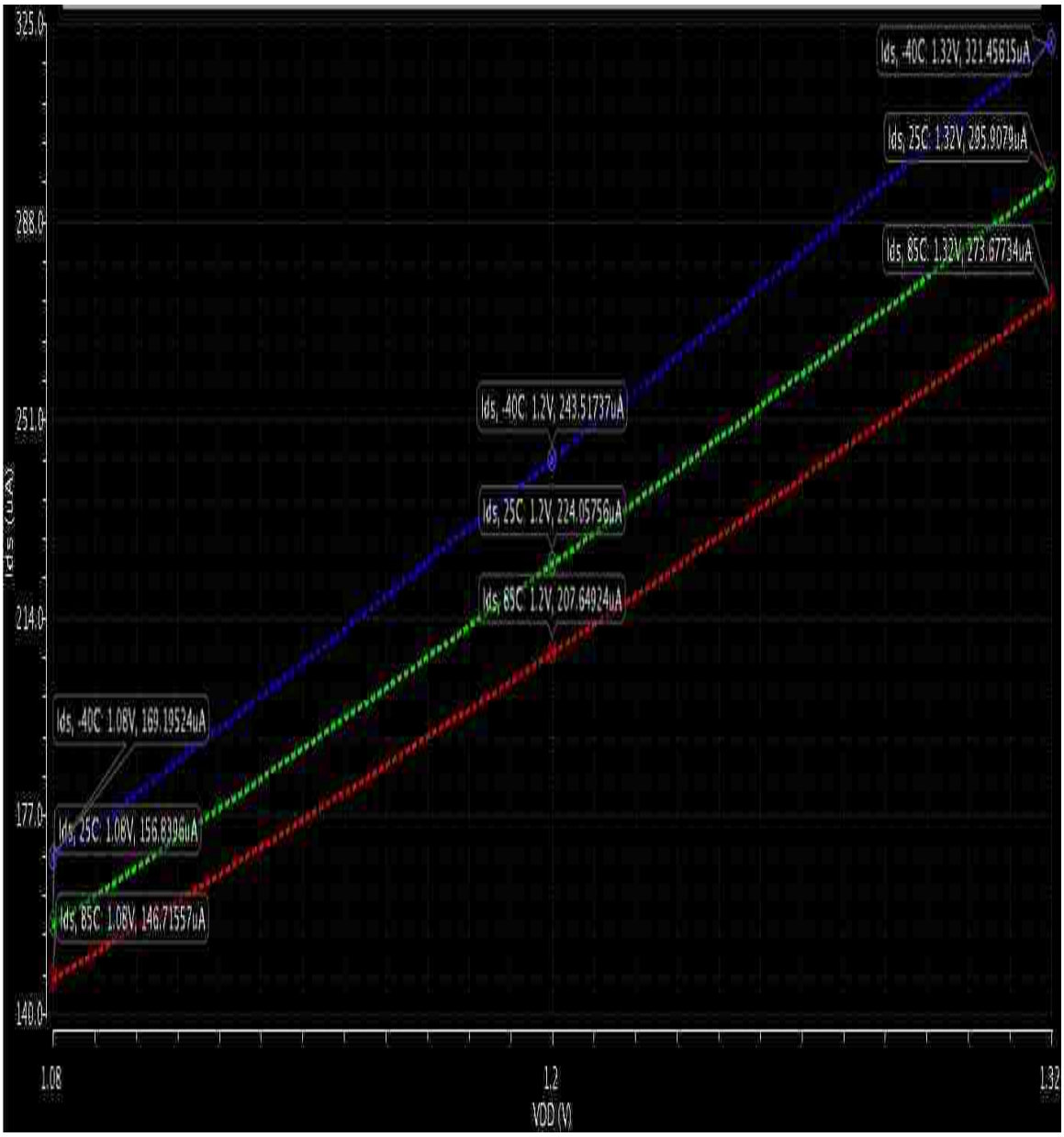


Fig. 103. Simulation results for I_{DS} vs. TV for the NFET TG-PUF

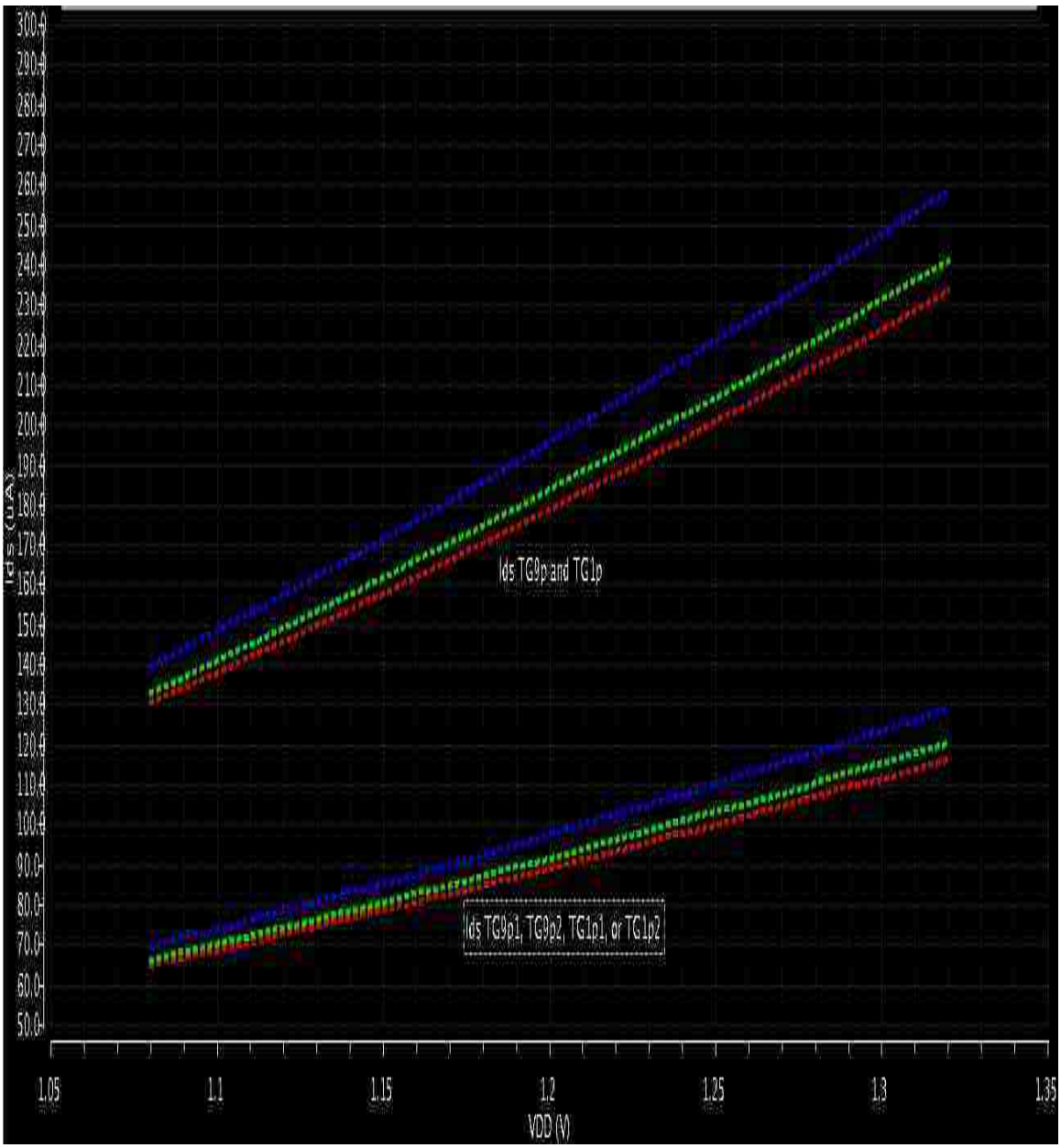


Fig. 104. Simulation results for I_{DS} vs. TV for the PFET TG-PUF

A sweep of the V_{GS} of NFET1n in the NFET TG-PUF primitive across all 3 temperatures yields the I_{DS} versus V_{GS} relationship depicted in Fig. 105. It is clear from this graph that as shown in Fig. 70 and described in [74], the I_{DS} (and therefore the R_{on}) of the transistors

exhibit a bimodal dependency on temperature. The inflection point where there is no dependency on temperature is at around a V_{GS} of 0.87V. Below this value, the I_{DS} increases with increasing temperature and above this value, I_{DS} decreases with increasing temperature. As stated in Chapter 7, our TG-PUF operates in the region where R_{on} increases with increasing temperature.

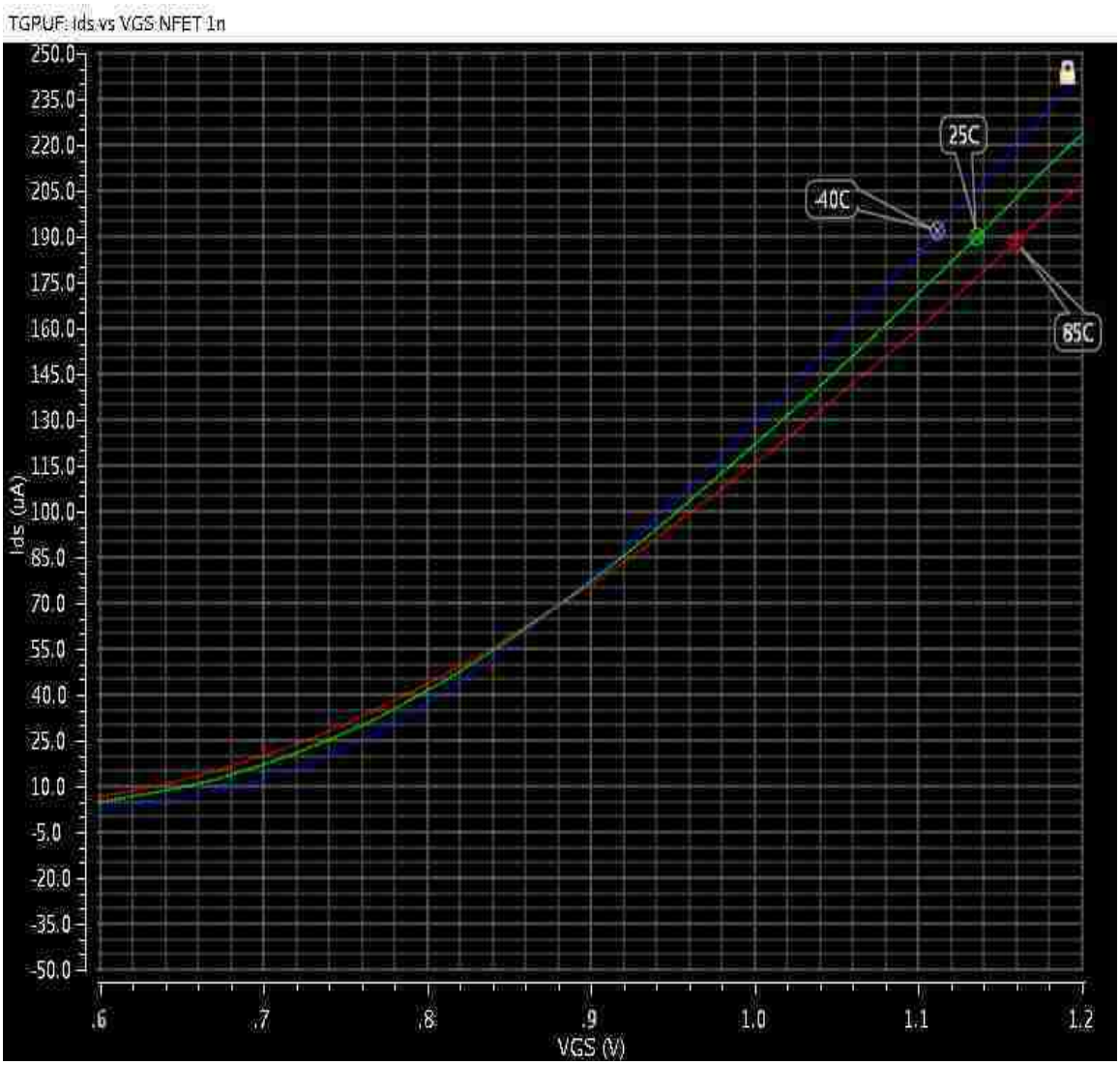


Fig. 105. I_{DS} vs. V_{GS} for NFETs in TG-PUF

Fig. 106 depicts the I_{DS} vs. V_{GS} relationship of the PFETs in the TG-PUF. Similar behavior as the NFETs is seen from this figure except the facts that the changes in I_{DS} (and therefore R_{on}) are much smaller with changing temperature for the PFETs than they are for the NFETs and that the changes in I_{DS} (and therefore R_{on}) from -40C to 25C are much larger than those from 25C to 85C. Again, we operate in the region where R_{on} increases with increasing temperature and the inflection point for I_{DS}'s reversal in temperature dependency appears to be at 0.92V.

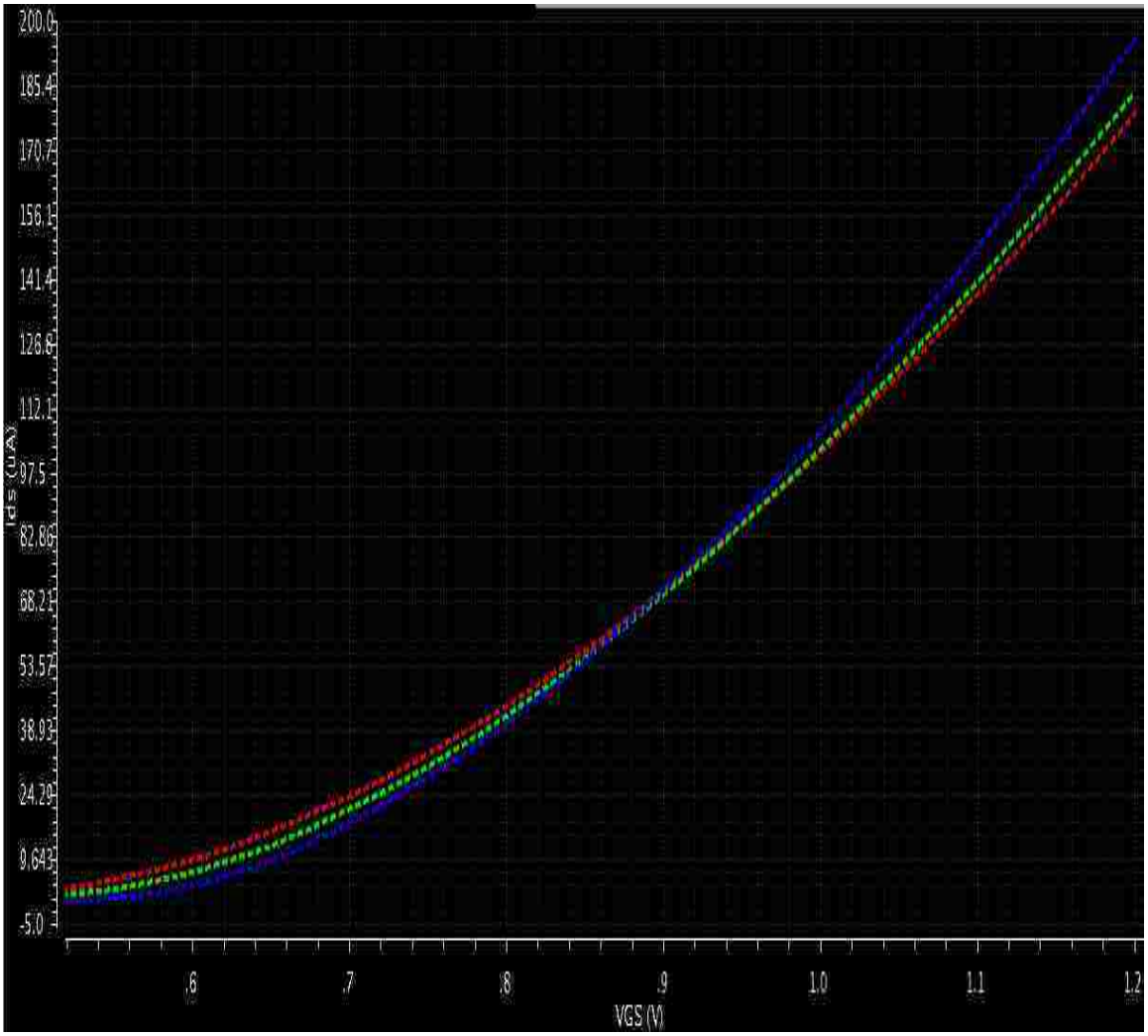


Fig. 106. I_{DS} vs. V_{GS} for PFETs in TG-PUF

Next, we plot the simulation results of the DC operating points for I_{DS} vs. V_{GS} for each of the transistors in the NFET primitive and the PFET primitive. Figs. 107 and 108 demonstrate this for the NFETs 1n and 9n for the NFET TG-PUF primitive, whereas Figs. 109 and 110 demonstrate this for the PFETs 1p and 9p for the PFET TG-PUF primitive, respectively. It can be seen from Figs. 107 and 108 that 1n exhibits larger shifts in I_{DS} with changing temperature compared to 9n and this is due to the larger V_{GS} DC operating points of 1n. 1n also operates farther away from the inflection point of 0.87V while 9n operates a lot closer to it.

From Figs. 109 and 110, it is evident that the PFETs exhibit similar behavior as the NFETs. 1p exhibits larger shifts in I_{DS} with changing temperature compared to 9p, and 1p also operates farther away from the inflection point of 0.92V compared to 9p. Also noteworthy from Fig. 110 is that the I_{DS} of PFET9p appears to reverse its dependency on temperature around the 0.88V region, meaning that the PFET9p operating conditions cause it to cross the inflection point where the I_{DS} dependency on temperature reverses. This is very similar to the PFET9p reversal that was seen in the experimental results of Section 7.1 except that the inflection point is predicted a little lower than what was observed (0.937V).

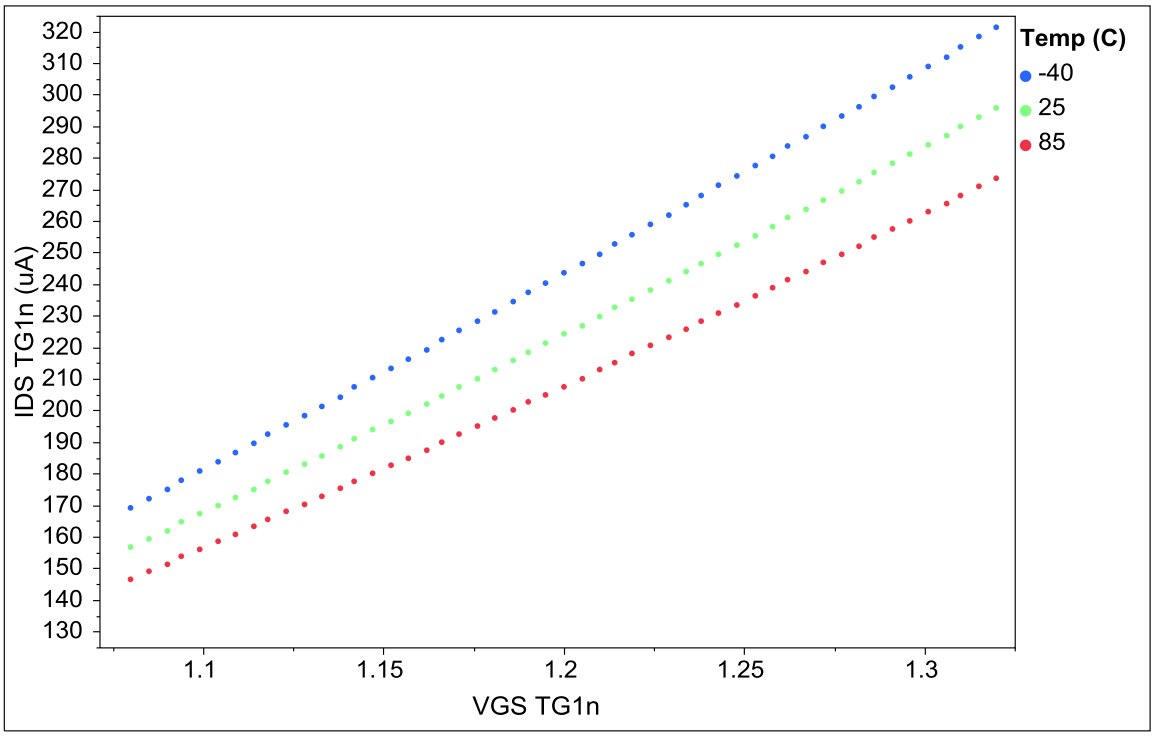


Fig. 107. Simulation results of I_{DS} vs. V_{GS} by Temperature for NFET1n

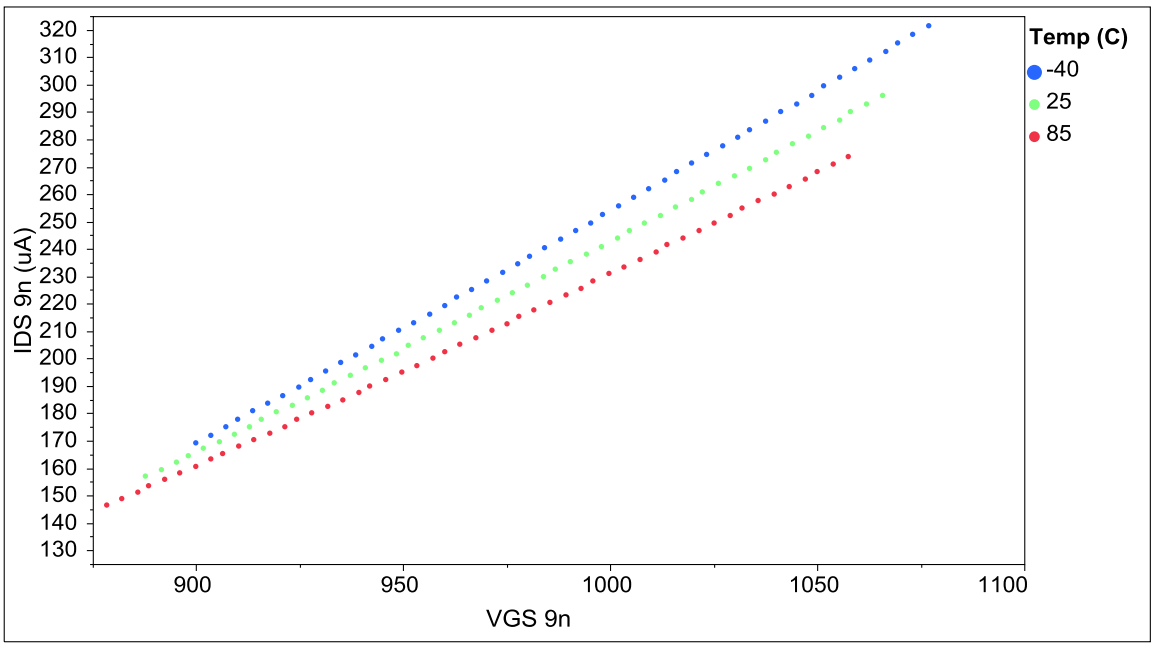


Fig. 108. Simulation results of I_{DS} vs. V_{GS} by Temperature for NFET9n

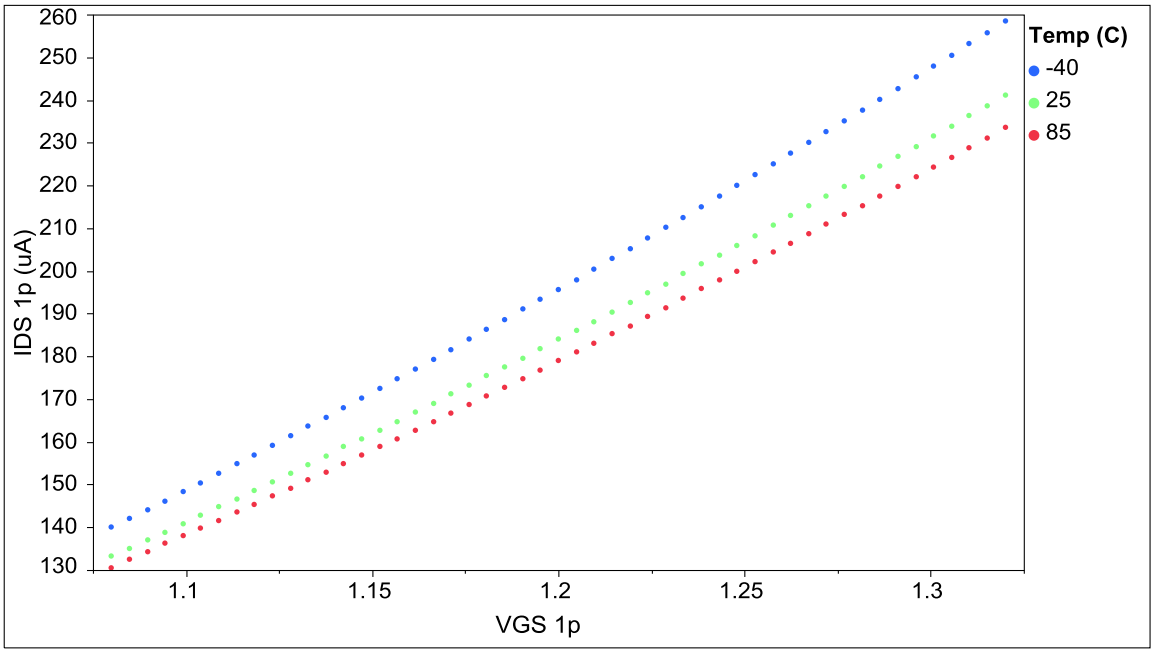


Fig. 109. Simulation results of I_{DS} vs. V_{GS} by Temperature for PFET1p

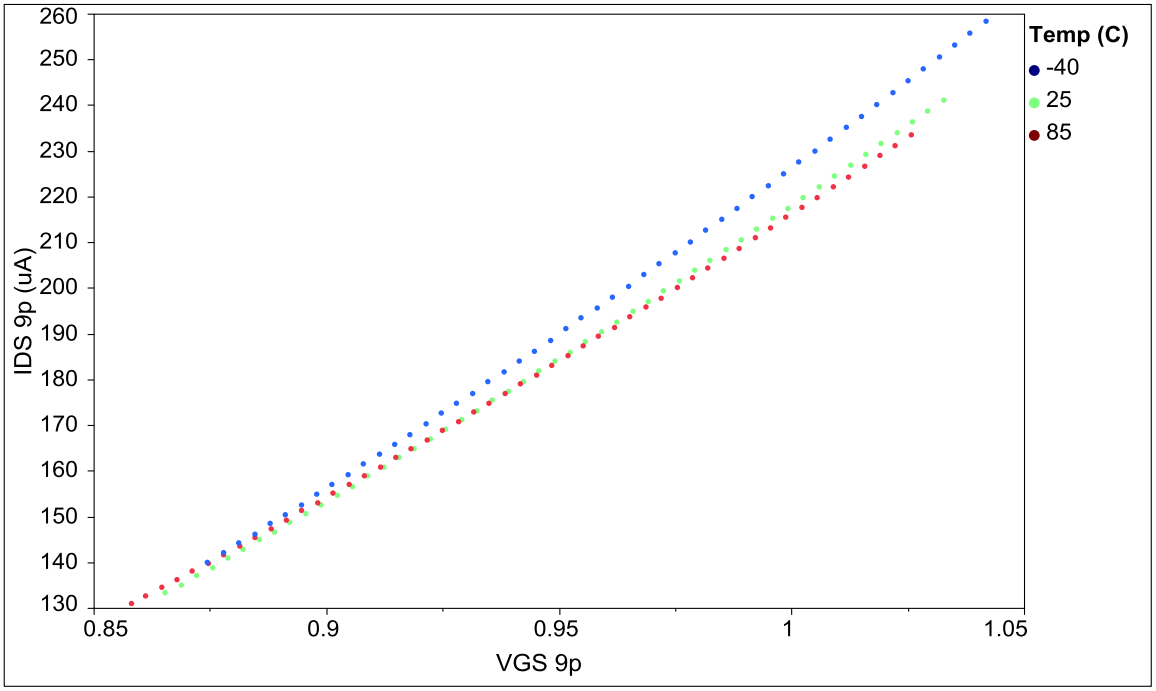


Fig. 110. Simulation results of I_{DS} vs. V_{GS} by Temperature for PFET9p

As an example, Fig. 111 below recaps the experimental results obtained from testing the NFET primitive on all SMCs of one of the chips. It can be seen that the simulation results of the I_{DS} vs V_{GS} DC operating points for NFET9n match pretty well with the experimental results in Fig. 111 when considering the spread of the experimental data.

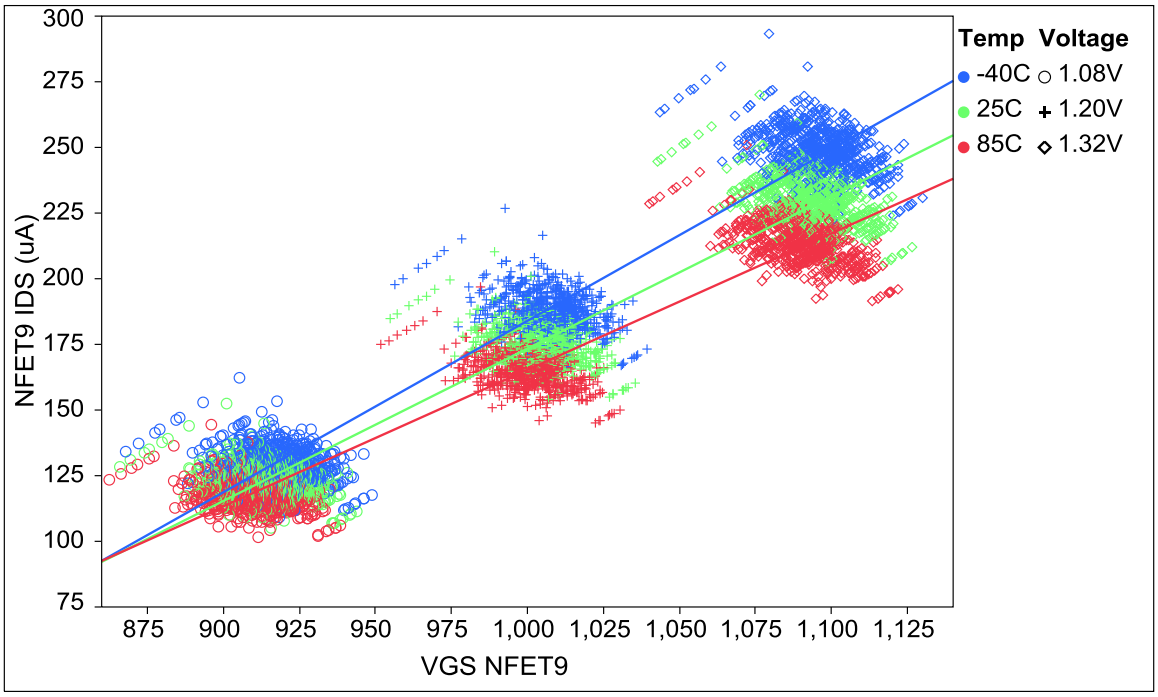


Fig. 111. Experimental results of I_{DS} vs. V_{GS} by Temperature for NFET9n

Next, the R_{on} of the transistors in the NFET and PFET primitive are simulated at the 9 TV corners based on the DC operating points of the TG-PUF. Figs. 112 and 113 depict the behavior of the R_{on} of NFETs 1n and 9n as a function of changing TV.

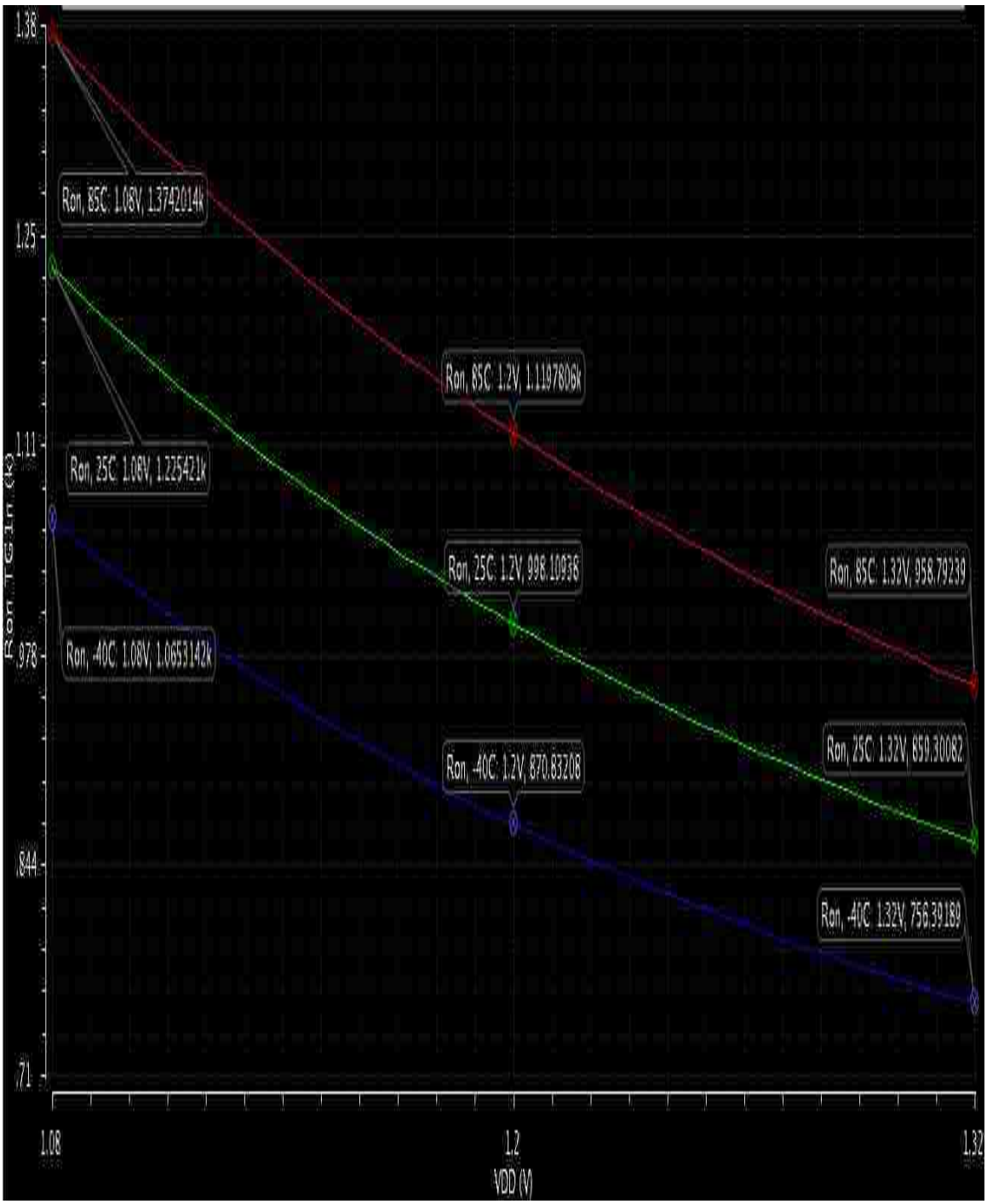


Fig. 112. Simulation results of R_{on} of NFET1n at 9 TV corners

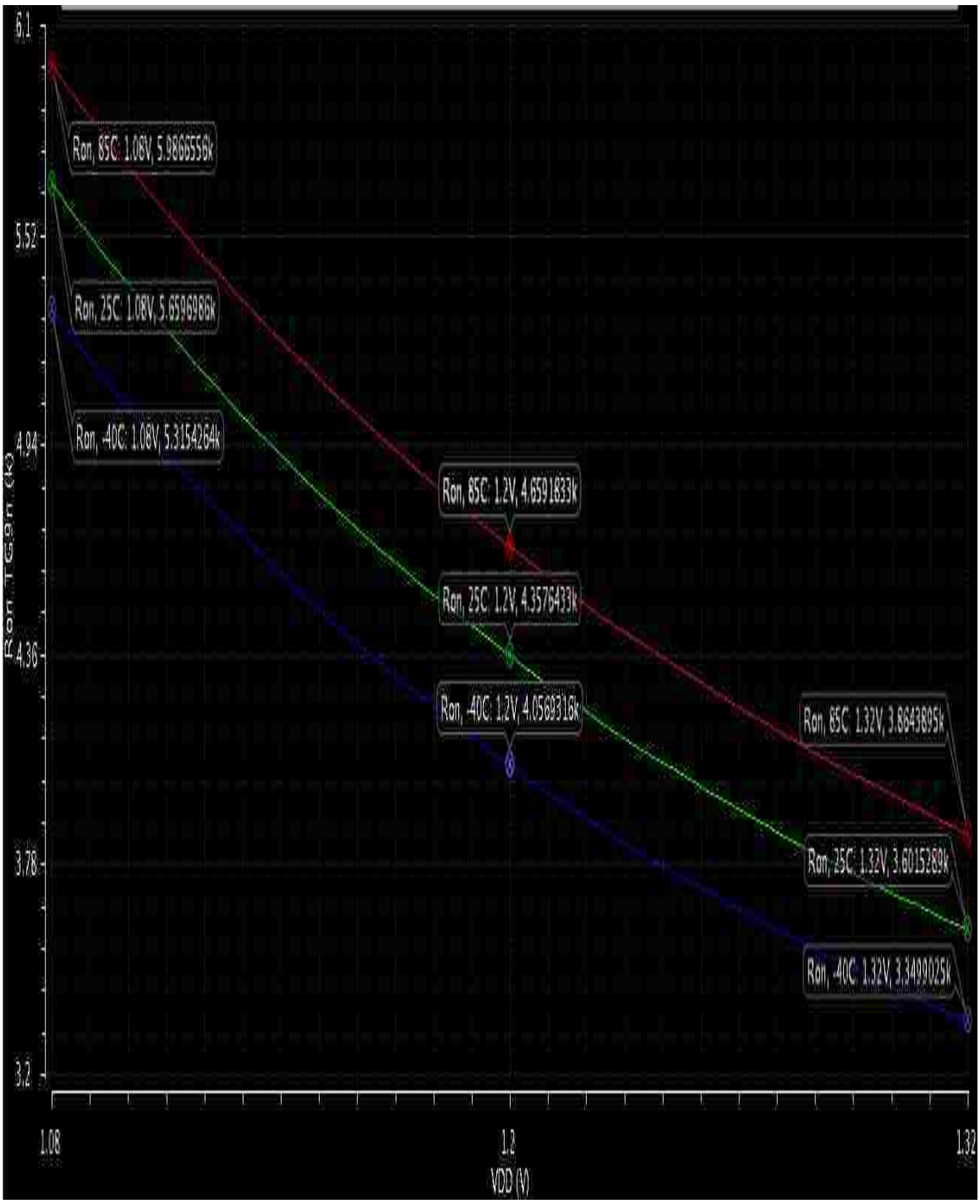


Fig. 113. Simulation results of R_{on} of NFET9n at 9 TV corners

As expected from Figs. 112 and 113, the R_{on} of NFETs 1n and 9n decrease with decreasing temperature and increasing V_{DD}. Comparing these simulation results with the experimental results of Section 7.1, we see that the R_{on} of NFET1n matches within 2% of the value obtained by experiment, while the experimental results for the R_{on} of NFET9n is about 15% higher than that obtained from simulation. It should be kept in mind that the experimental result comparison is based on data from just one chip and due to the larger Ron of NFET9n, the discrepancy appears fairly large. When compared to the data from all chips, the simulated results are well within the distribution of the R_{on} obtained from the hardware experiments.

Figs. 114 and 115 illustrate the behavior of the R_{on} of the PFETs in the PFET TG-PUF primitive as a function of changing TV conditions.

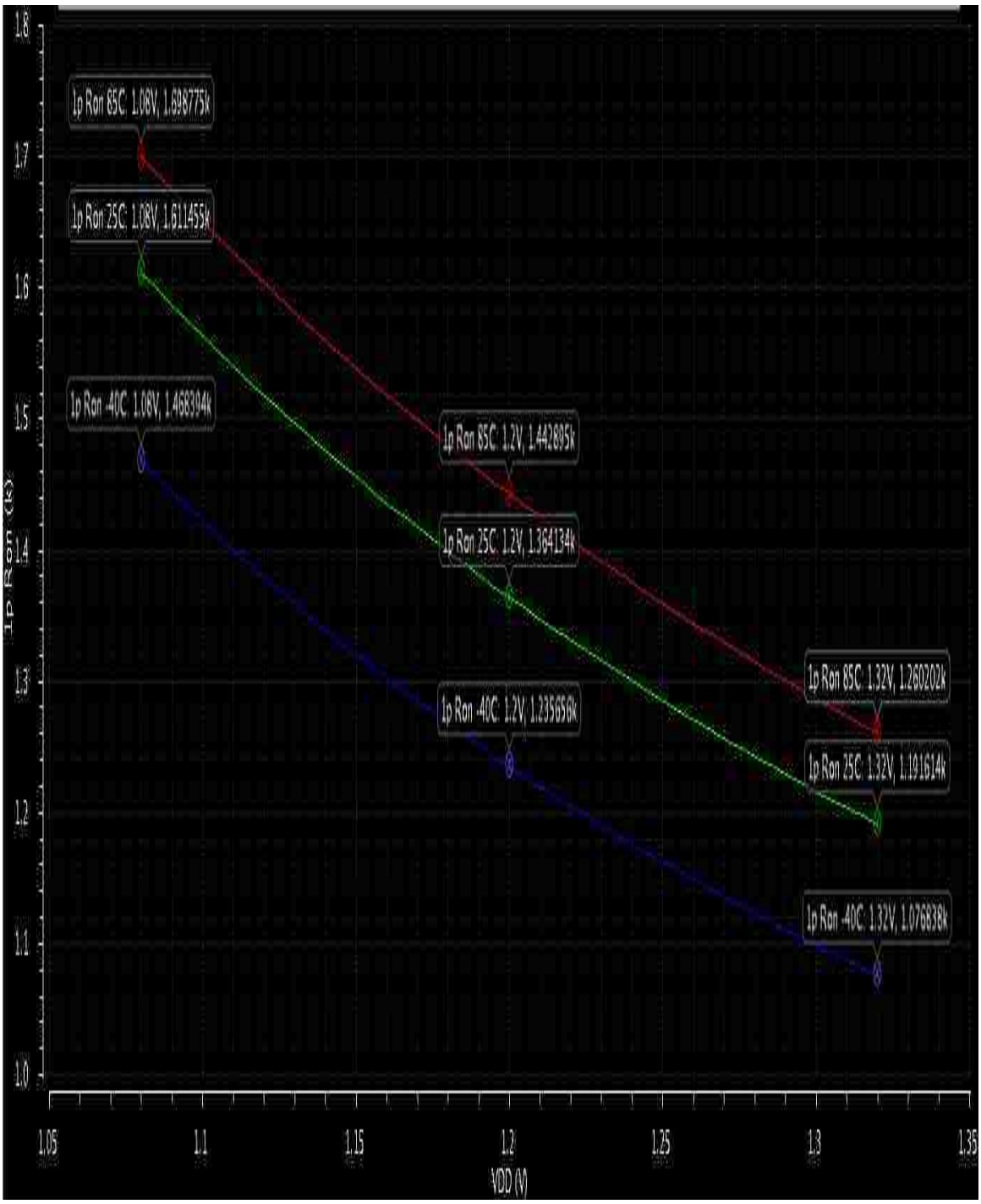


Fig. 114. Simulation results of R_{on} of PFET1p at 9 TV corners

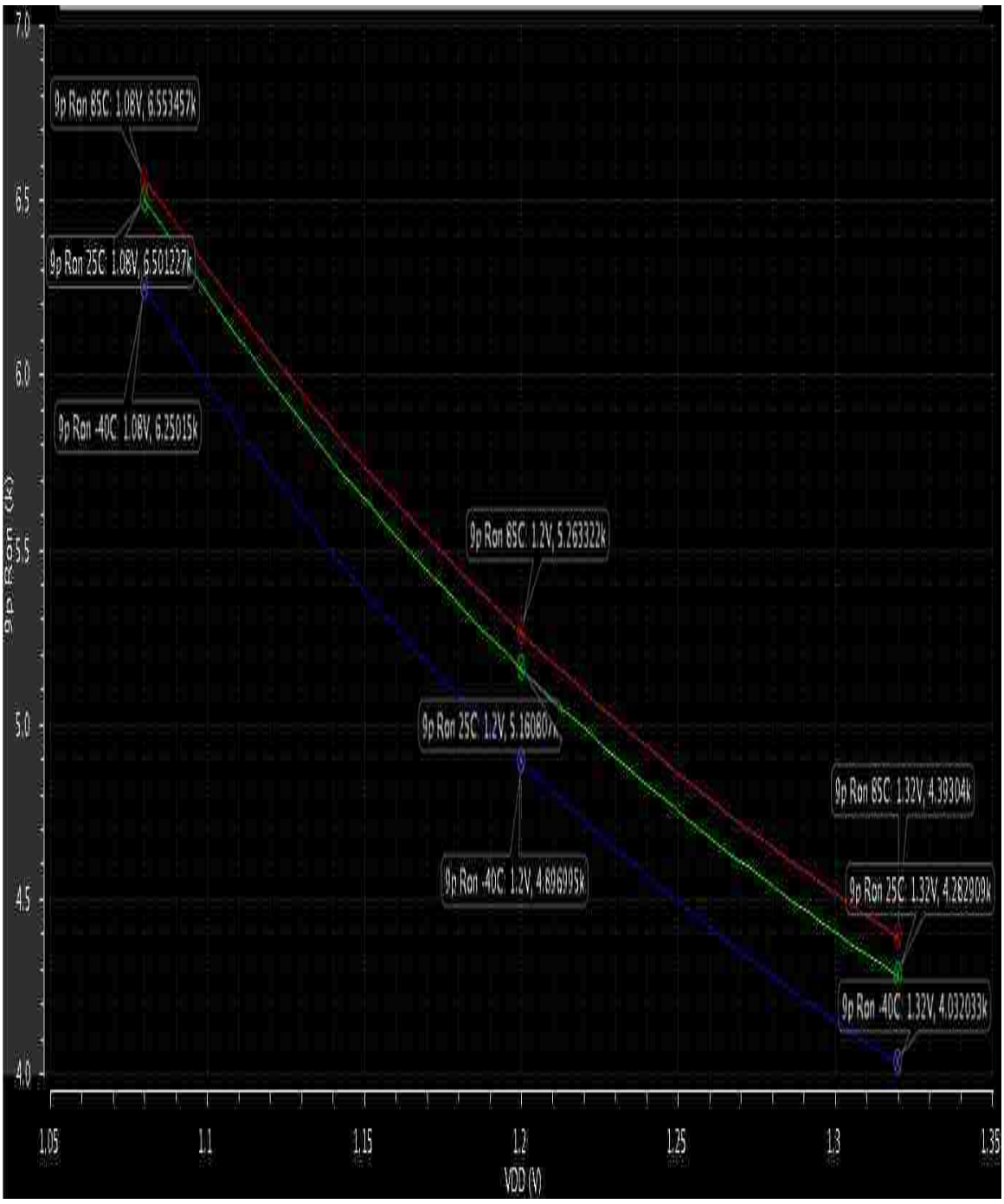


Fig. 115. Simulation results of R_{on} of PFET9p at 9 TV corners

Comparing the PFET R_{on} simulation results of Figs. 114 and 115 to the experimental results of Section 7.1, it is evident that the values are within 1% for the PFET1p and within about 10% for the PFET9p which operates fairly close to the inflection point.

Next, the regions of operation of each of the transistors in the NFET and PFET primitives were investigated. Figs. 116 and 117 depict the regions of operation at all 9 TV corners for the NFETs and PFETs respectively. The y-axis designates the regions whereas the x-axis represents the 3 temperatures of -40C, 25C, and 85C. These temperature sweeps are conducted at all 3 V_{DD} settings of 1.08V, 1.2V, and 1.32V to give us the information depicted below.

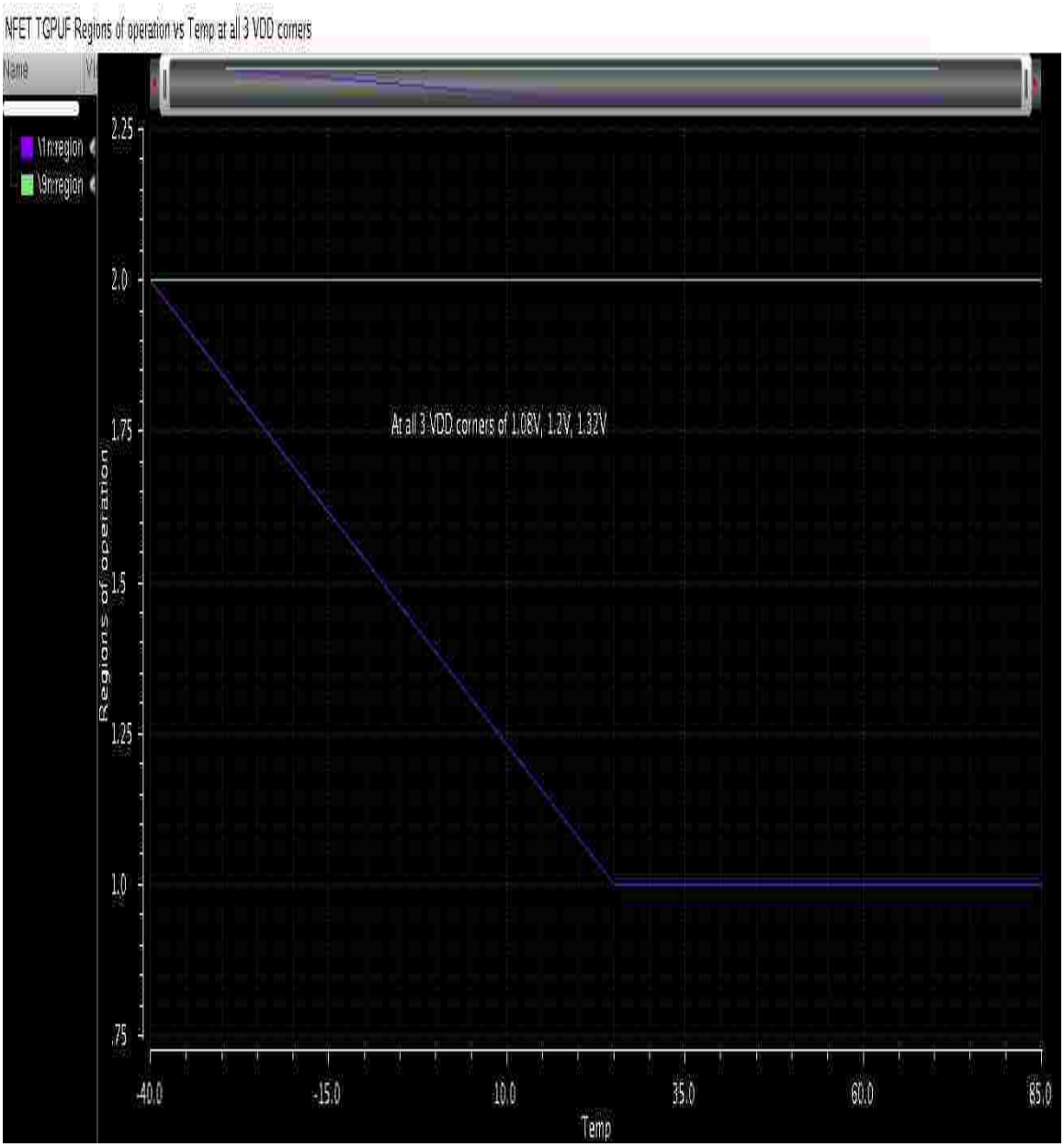


Fig. 116. NFETs 1n and 9n regions of operation at the 9 TV corners

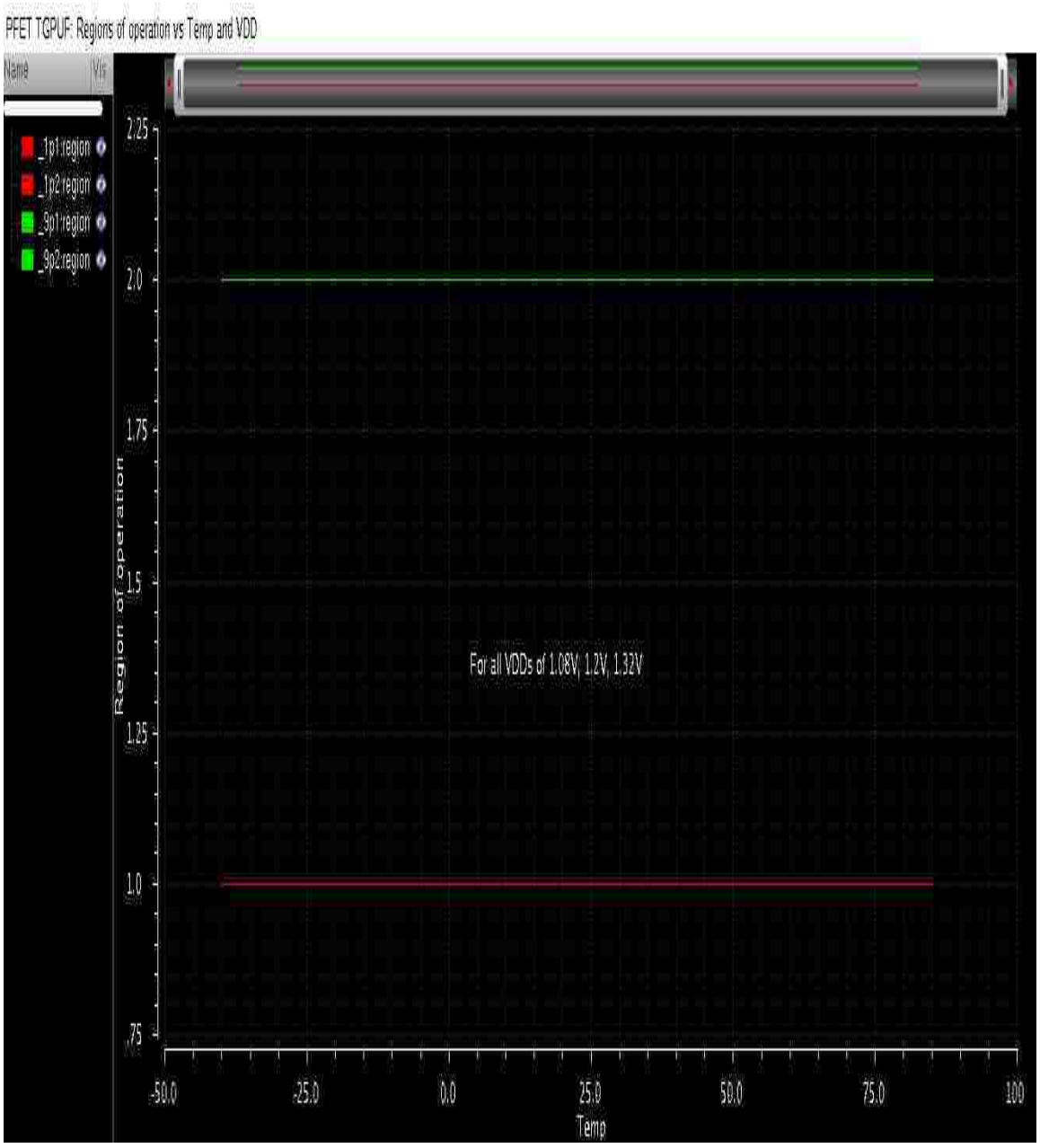


Fig. 117. PFETs 1p and 9p regions of operation at the 9 TV corners

A region of operation of 1 corresponds to the linear region whereas 2 corresponds to the saturation region. Evident from the region of operation graphs is that NFET9n stays in saturation at all TV corners, however NFET 1n stays in the linear region at 25C and

85C but flips to the saturation region at -40C. On the other hand, PFET1p stays in the linear region at all TV corners and PFET9p stays in the saturation region for all TV corners. These results are in complete agreement with what was obtained from experimental results because the R_{on} of NFET9n and PFET9p are much higher than NFET1n and PFET1p, respectively, and higher R_{on} is associated with transistors in saturation. Furthermore, as corroborated from experimental results that agree with the simulated results, the saturated transistors operate at a lower V_{GS} and thus closer to the inflection point where temperature changes have smaller impact to the R_{on} compared to that of the transistors operating in the linear region.

The total power dissipation of the NFET and PFET TG-PUF primitive was simulated at the 9 TV corners. These results are depicted in Figs. 118 and 119 for the NFET and PFET primitives respectively. It is evident that the power dissipation increases with decreasing temperature and increase V_{DD}. Another noteworthy fact is that the NFET primitive dissipates about 21% more power than the PFET primitive mainly due to the higher currents in the NFET TG-PUF primitive. The simulated power dissipation results agree fairly well with what was observed in the experimental results.

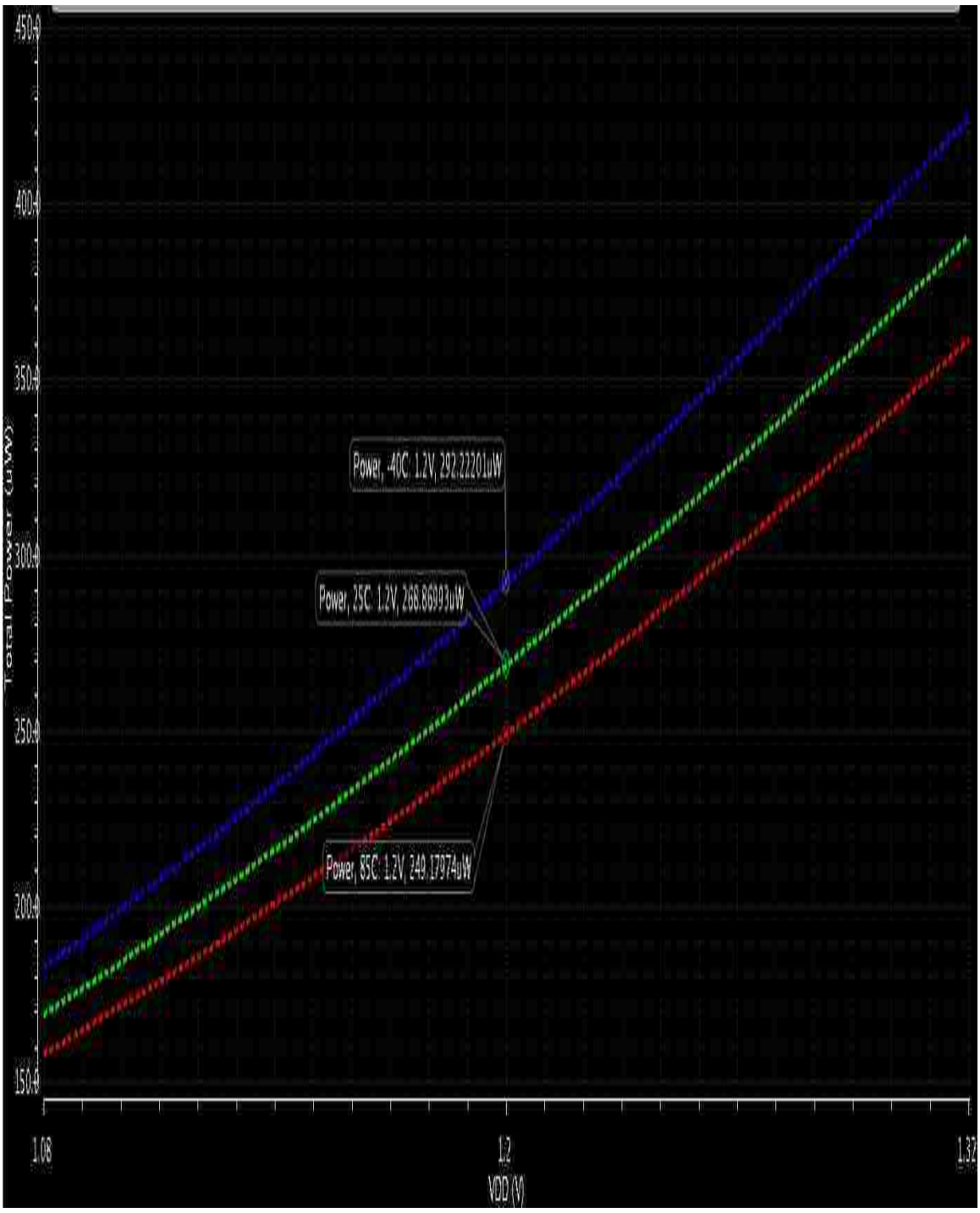


Fig. 118. Total power dissipation of NFET TG-PUF primitive at the 9 TV corners

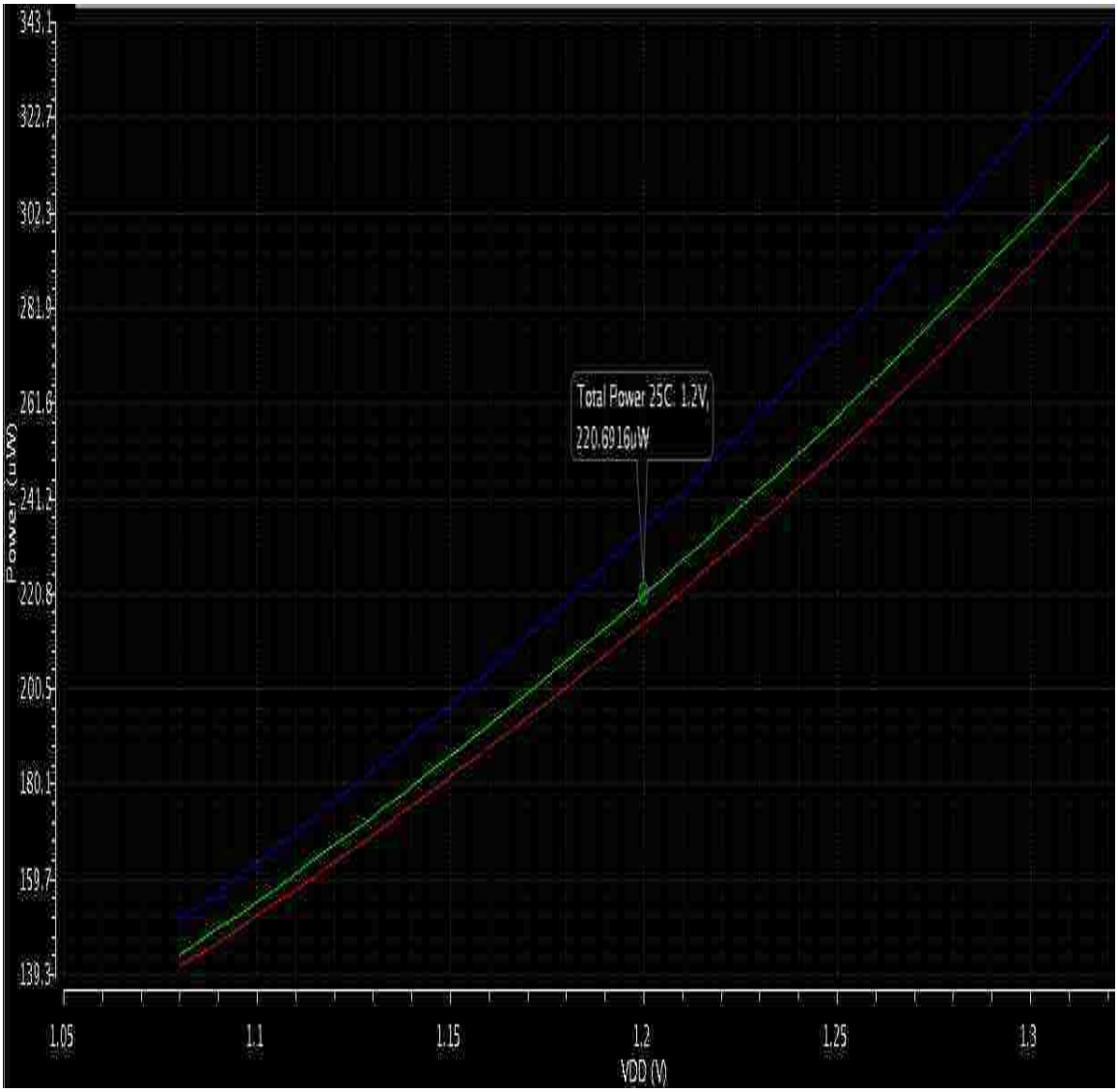


Fig. 119. Total power dissipation of PFET TG-PUF primitive at the 9 TV corners

8.2 I-PUF

Fig. 120 illustrates the schematic of the I-PUF primitive used in the simulation. This was also the same primitive used in the chip design. It should be noted from Fig. 120 that the

PFET primitive is actually constructed from 4 PFETs of the same size. The top 2 PFETs, 1p1 and 1p2, are connected in parallel and their total parallel resistance is denoted later in this section as 1p, while the bottom 2 PFETs, 9p1 and 9p2, are connected in parallel and their total parallel resistance is denoted later in this section as 9p. It should be obvious from the schematic below that NFET1n is what is referred to as the stacked NFETs previously in this document while PFET1p is the stacked PFETs.

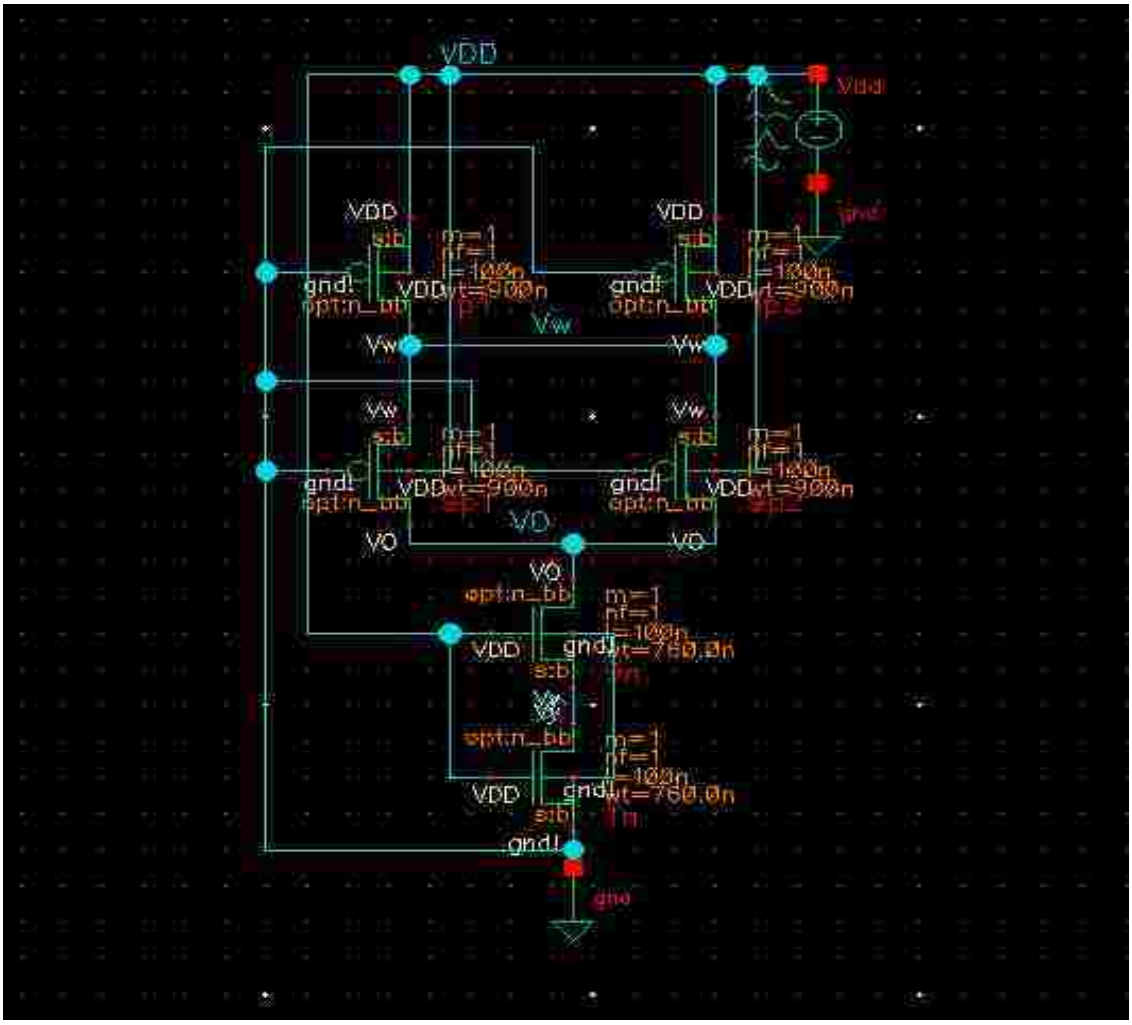


Fig. 120. Schematic of I-PUF primitive

Next, the V_{DD} voltage is swept from 1.08V to 1.32V to yield the V_{DD} corner information at 1.08V, 1.2V, and 1.32V (1.2V +/- 10%). This V_{DD} sweep is conducted in conjunction with a parametric analysis sweep of the temperature at -40C, 25C, and 85C. This yields the 9 corner TV simulation data.

Fig. 121 depicts the VO data at all 9 TV corners. The x-axis represents the V_{DD} voltage sweep while the 3 different temperatures are color coded. It can be seen that the VO voltage increases with increasing temperature and increasing $V_{DD.}$ Noteworthy is the fact that the rate of increase in VO gets larger with increasing V_{DD}, which was also evident in the experimental results and is due to the slower rate of decrease in the combined NFET R_{on} at higher V_{DD}. The increase in VO with temperature was not seen at these levels in the experimental results and this discrepancy is likely attributed to temperature modeling gaps in the transistor standard cells. The VO values seem to be predicted lower by simulation than what the experimental results yielded. This discrepancy is due to the fact that the simulated results are being compared to the experimental results of just one chip. When compared to the data from all chips, the simulated results are well within the distribution of the VOs obtained from the hardware experiments.

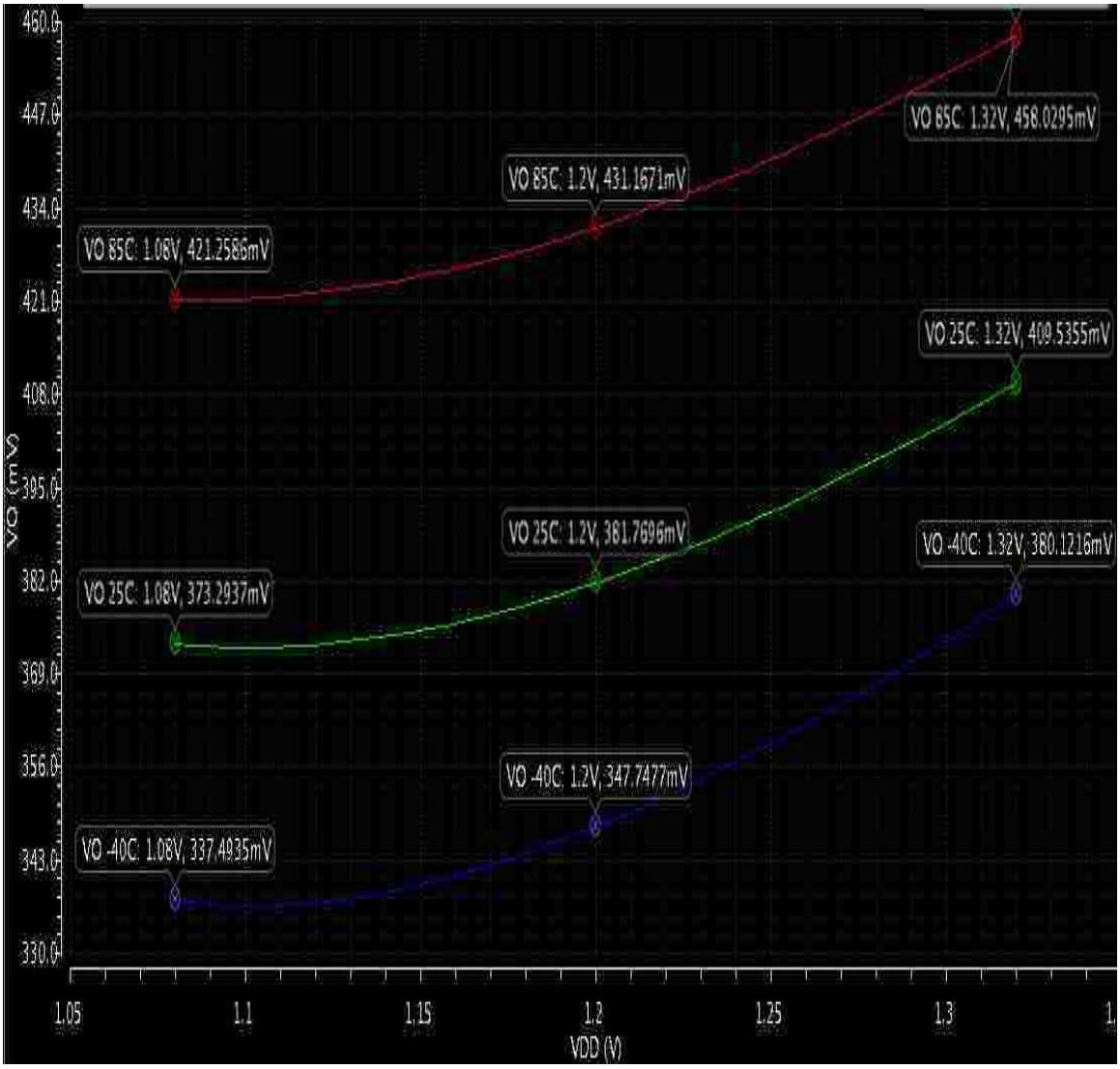


Fig. 121. Simulation results of VO at the 9 TV corners for the I-PUF primitive

Next, the intermediate NFET and PFET voltages of V_y and V_w, respectively, are obtained as a function of the 9 TV corners. This is simulated by conducting a DC sweep of V_{DD} between 1.08V and 1.32V at the 3 temperature points. The results are depicted in Figs. 122 and 123. Noteworthy from these graphs is the fact that the intermediate NFET voltage V_y exhibits a much larger shift with changing temperature than the intermediate

PFET voltage V_w, and this is attributed to the fact that the R_{on} of the NFETs has a larger change with temperature as compared to the R_{on} of PFETs. Also, while both voltages increase with increasing V_{DD}, V_y increases with increasing temperature whereas V_w increases with decreasing temperature. This is consistent with the expected behavior as the V_w voltage is calculated as a drop from V_{DD} whereas V_y is calculated as an increase from GND.

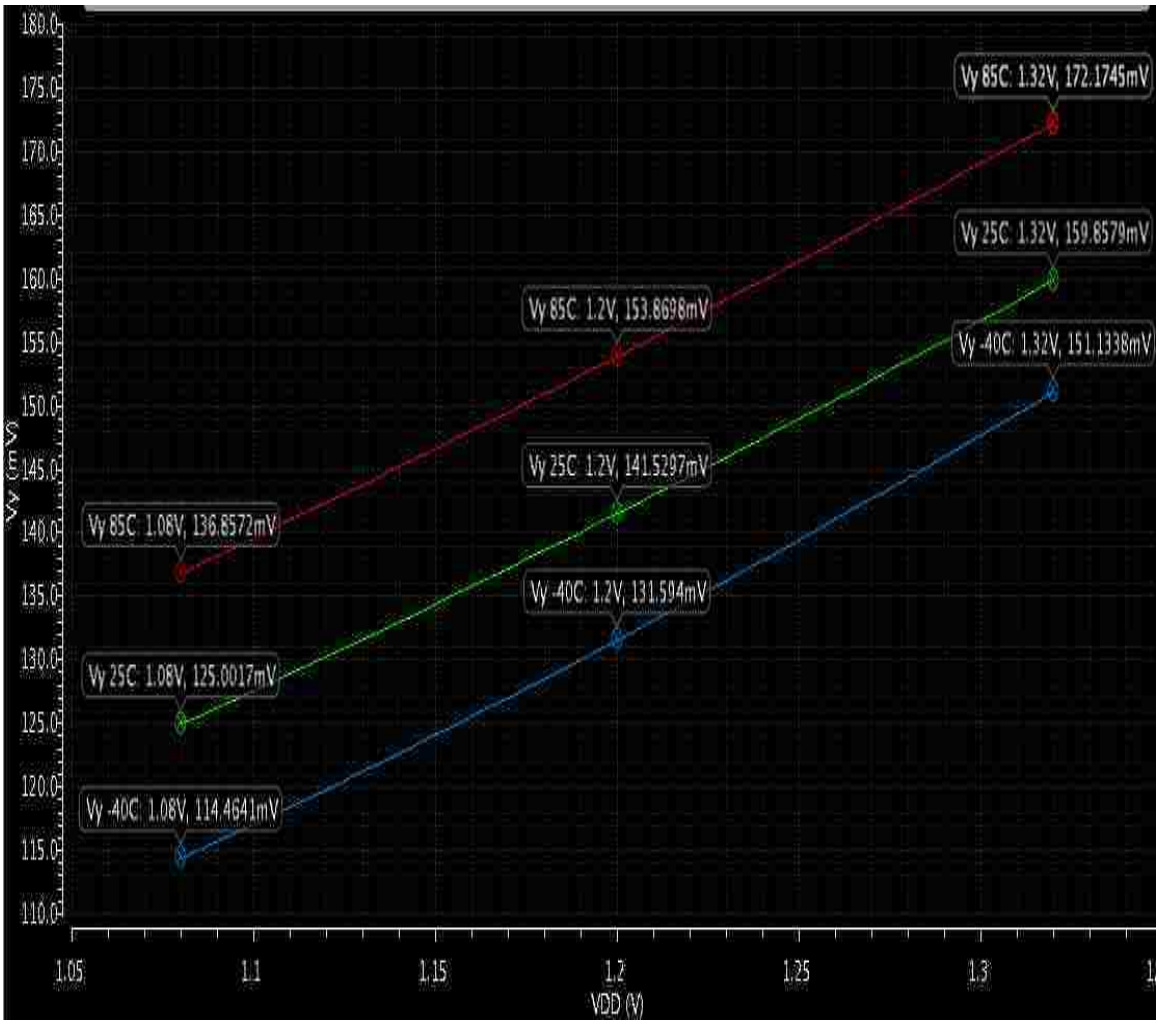


Fig. 122. Simulation results of V_y at the 9 TV corners for the I-PUF primitive

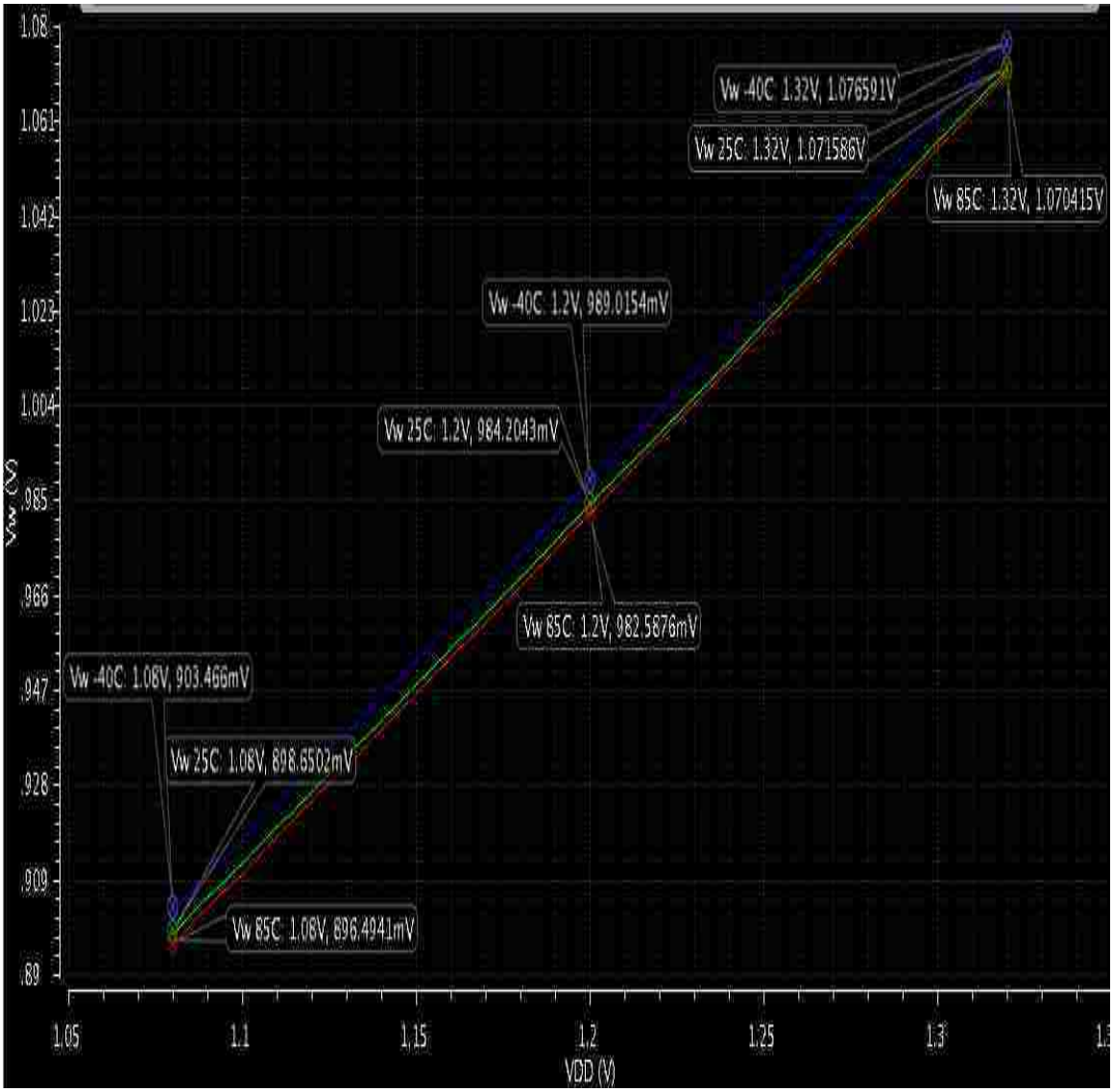


Fig. 123. Simulation results of V_w at the 9 TV corners for the I-PUF primitive

Figs. 124 and 125 recap the V_y and V_w voltages obtained from the experimental results. As can be seen when comparing the simulated results to those of the hardware experiments, the V_y and V_w voltages exhibit the same characteristics at the 9 TV corners. The V_y and V_w values, however, seem to be predicted a little lower than what the

experimental results yielded. This discrepancy is due to the fact that the simulated results are being compared to the experimental results of just one chip. When compared to the data from all chips, the simulated results are well within the distribution of the VOs obtained from the hardware experiments.

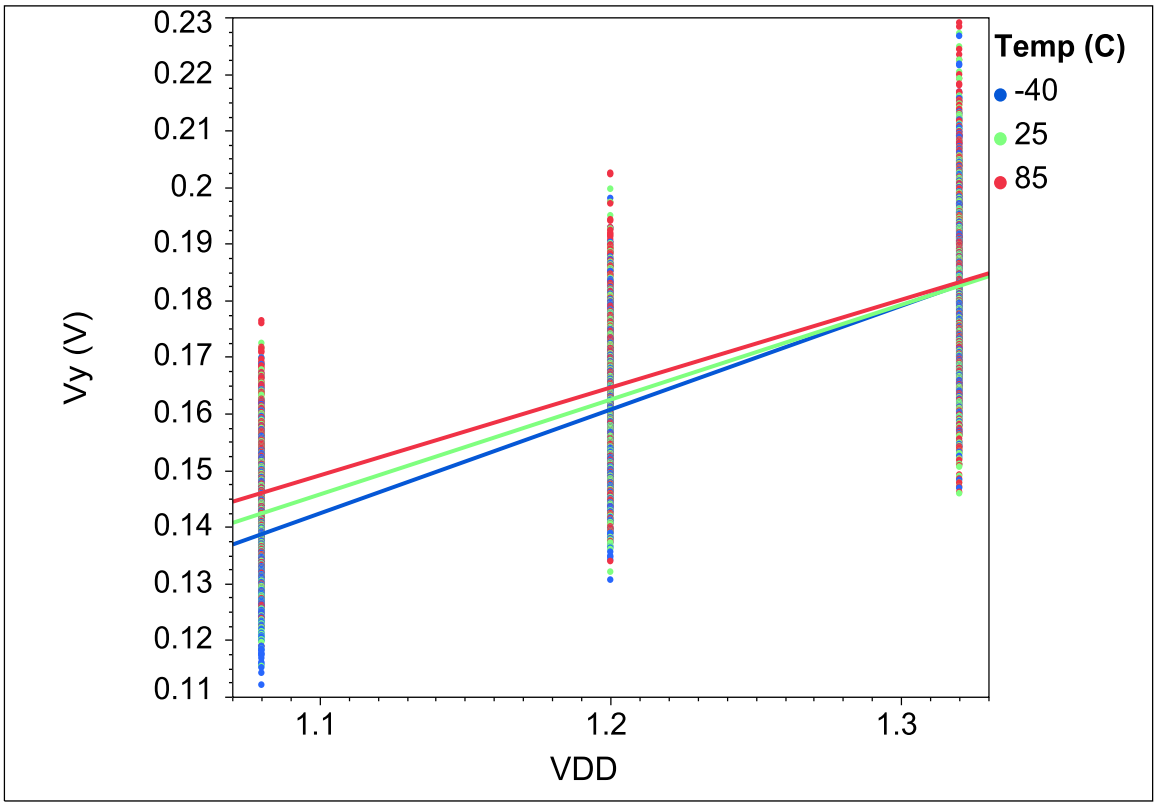


Fig. 124. Hardware experimental results of V_y at the 9 TV corners for the I-PUF primitive

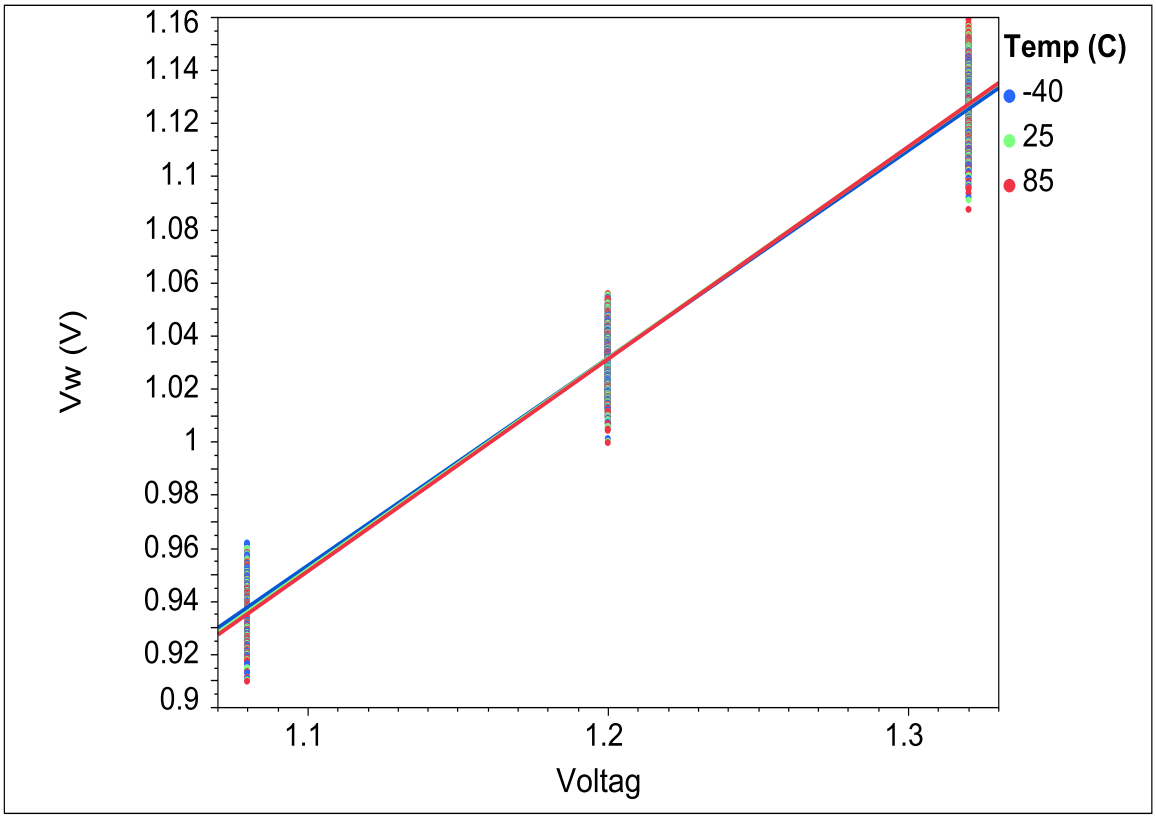


Fig. 125. Hardware experimental results of V_w at the 9 TV corners for the I-PUF primitive

Next, the R_{on} of the transistors in the I-PUF primitive is simulated at the 9 TV corners based on the DC operating points of the I-PUF. Figs. 126 through 129 depict the behavior of the R_{on} of NFET1n, NFET9n, PFET1p, and PFET9p as a function of changing TV. Comparing these simulation results with the experimental results of Section 7.3, we see an overall good match with the following exceptions. As stated earlier, the VO was predicted a little lower by simulation than what the experimental results yielded. This therefore had a greater impact on the simulated R_{on} results of NFET9n and PFET9p as their values are directly dependent on VO, unlike the NFET1n and PFET1p R_{on} values. Therefore, when comparing the simulated R_{on} values of NFET1n

and PFET1p with the experimental results, we see an excellent match between simulation and experimental results. However, when comparing the results of the R_{on} of NFET9n and PFET9p, we see that the experimental results are about 30% higher than the simulated results and this is mainly due to the fact that the VO values obtained by experiments are higher than those obtained by simulation. Once again, it should be noted that this is based on comparison with experiment results from just one chip. The R_{on} results obtained by simulation are well within the distribution of the R_{on} obtained by experiments for all chips. Noteworthy is the simulated behavior of the R_{on} of PFET9p. As can be seen from Fig. 129, the R_{on} of PFET9p increases with decreasing temperature at 1.08V, with the R_{on} at -40C and 25C almost identical. However, the -40C R_{on} curve crosses over with increasing V_{DD} and at 1.32V, is almost identical to the R_{on} at 85C. The reason for this behavior is attributed to the fact that PFET9p, which operates in the saturation region, also operates in a region which is fairly close to the cross-over inflection point where the temperature dependence of the R_{on} of the transistor flips, i.e. the R_{on} increases with increasing temperature above this inflection point but decreases with temperature below it. Since 9p operates near this point, we see the cross-over behavior of the R_{on} curves as a function of TV.

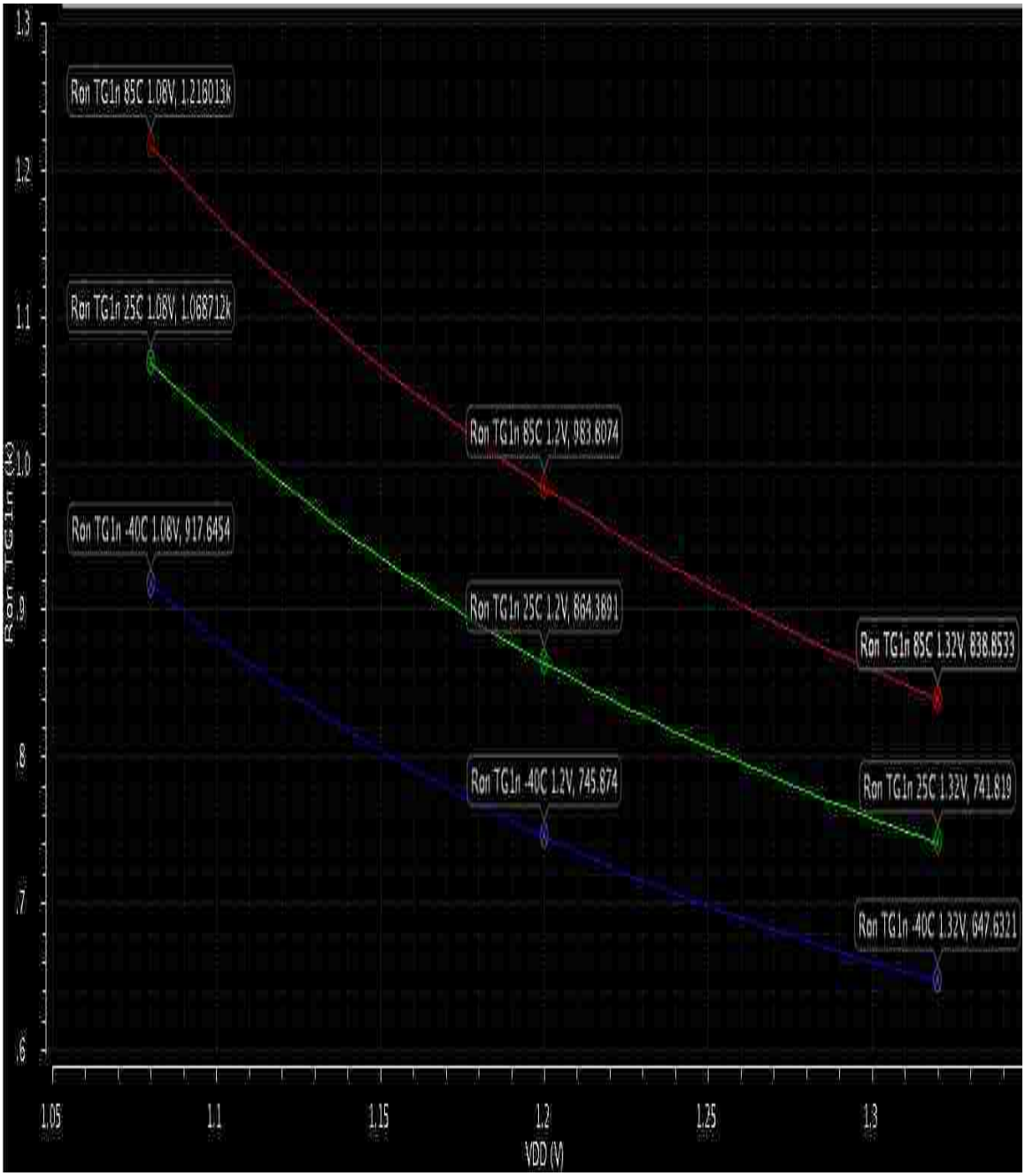


Fig. 126. Simulation results of R_{on} of NFET1n at the 9 TV corners for the I-PUF

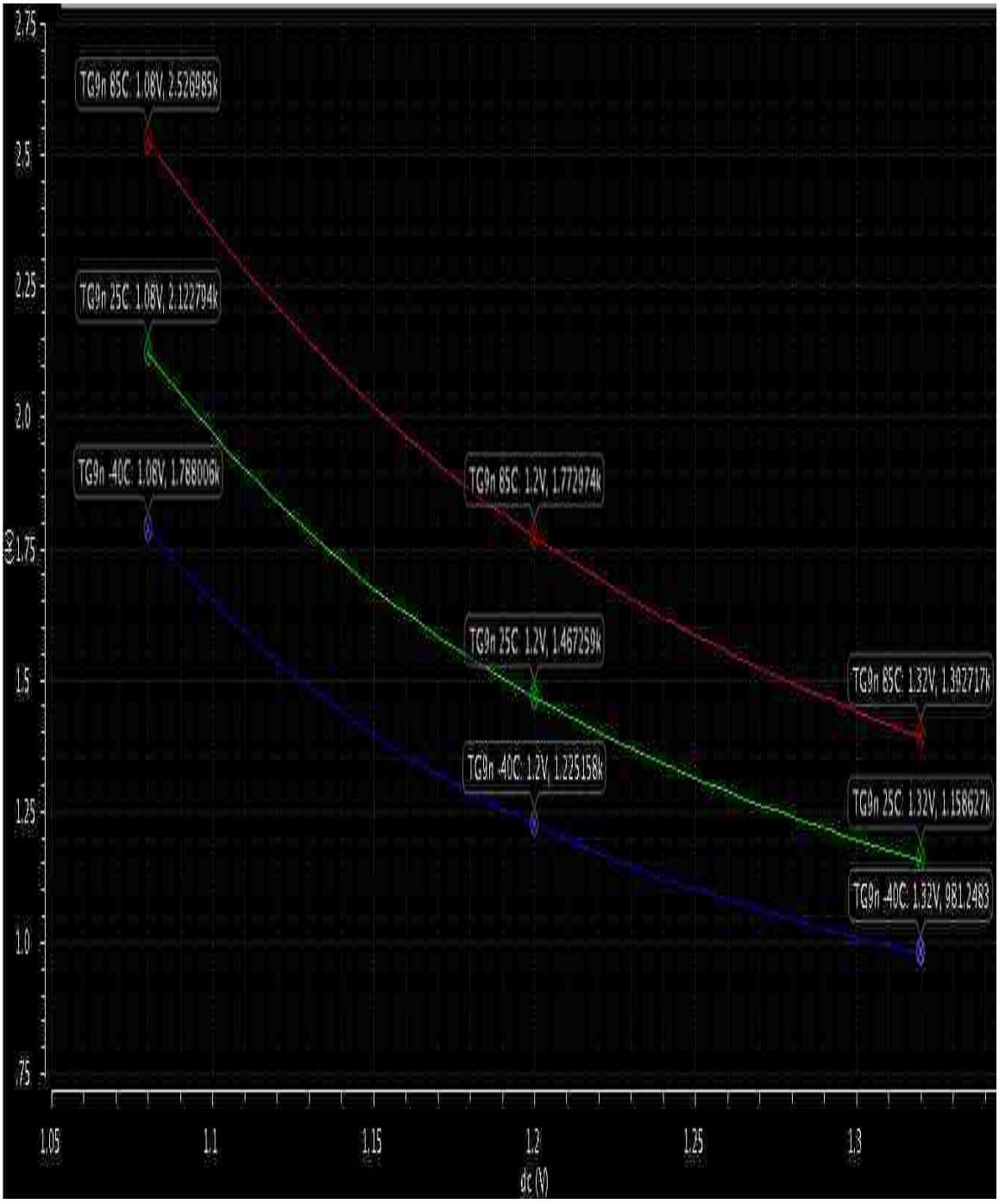


Fig. 127. Simulation results of R_{on} of NFET9n at the 9 TV corners for the I-PUF

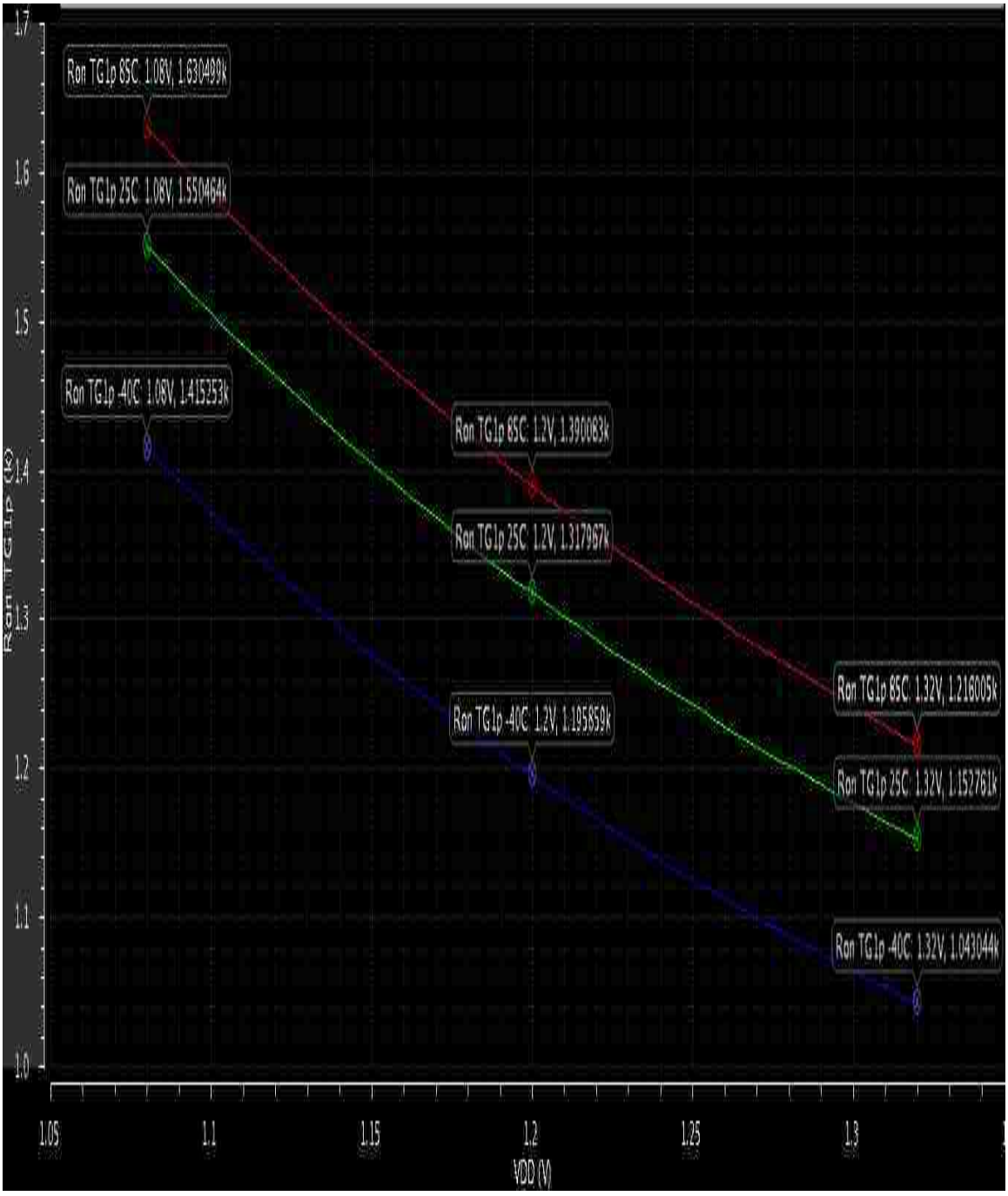


Fig. 128. Simulation results of R_{on} of PFET1p at the 9 TV corners for the I-PUF

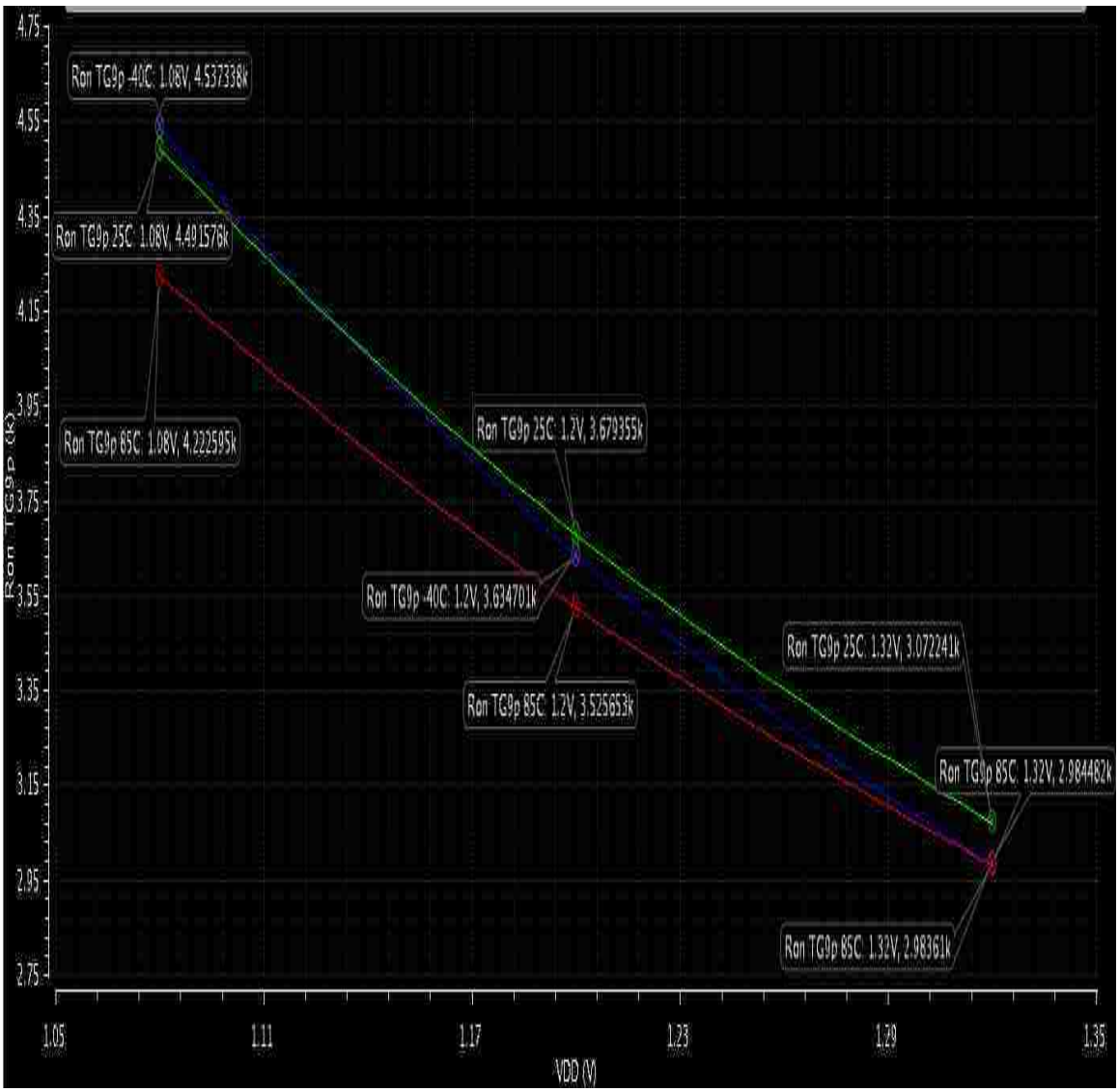


Fig. 129. Simulation results of R_{on} of PFET9p at the 9 TV corners for the I-PUF

The regions of operation of the individual transistors in the I-PUF primitive are illustrated in Fig. 130 at all TV corners. It should be noted that Fig. 130 represents the regions of operation at all the V_{DD} voltages while the x-axis represents the 3 temperature settings, so it should be clear that the regions do not change as a function of TV. The simulated regions of operation match those seen in the experimental results. The NFET9n

and PFET9p operate in saturation region for all TVs while the NFET1n and PFET1p operate in the linear regions. This is also consistent with the higher R_{on} values of these saturated transistors.

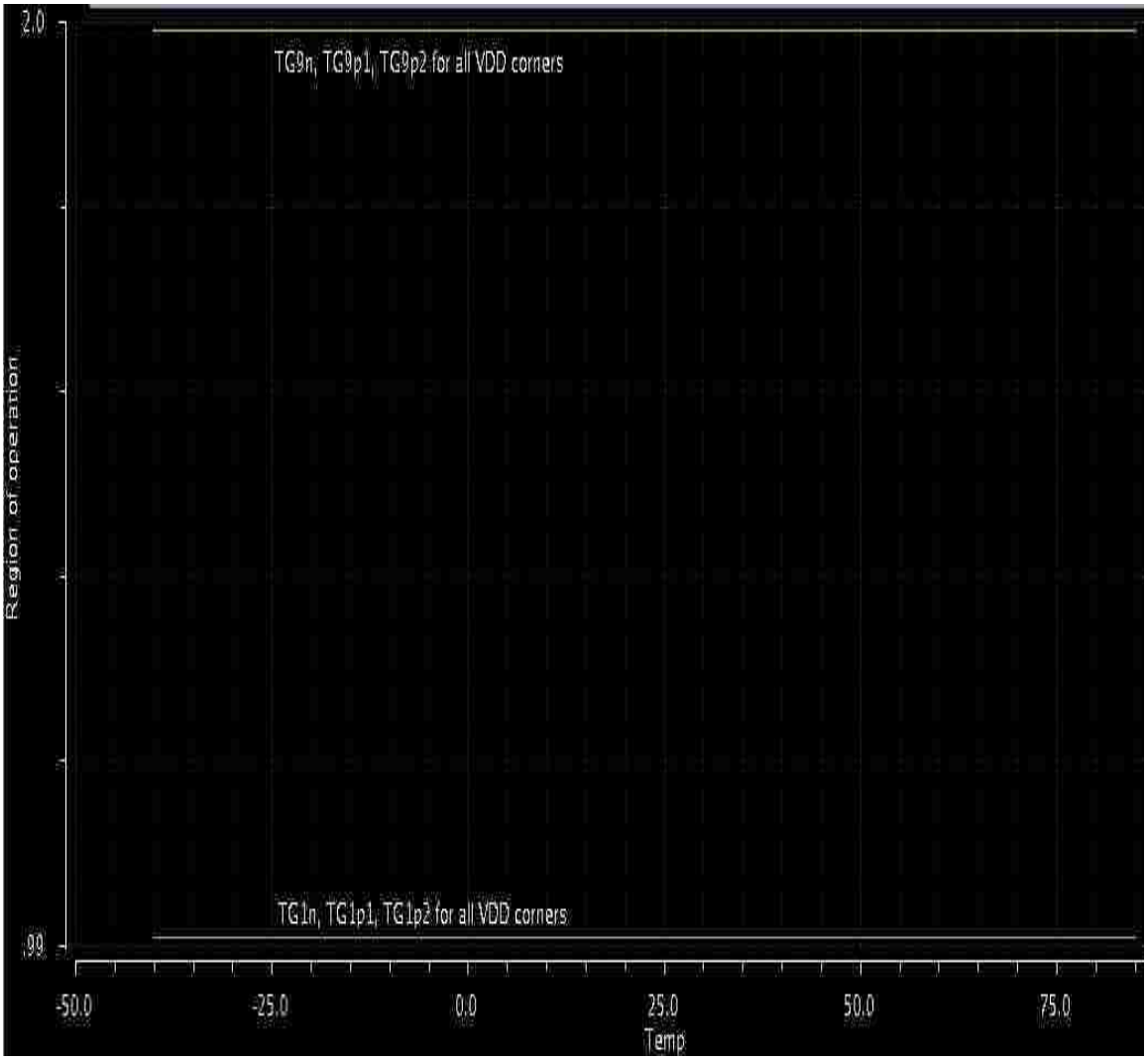


Fig. 130. PFET1p, PFET9p, NFET1n, and NFET9n regions of operation at the 9 TV corners

Lastly, the total power dissipation of the I-PUF primitive as a function of the different TV corners is depicted in Fig. 131. As can be seen, the power dissipation of the I-PUF primitive is lower than that of either the NFET or PFET TG-PUF primitive. This is mainly due to the stacked effect of the multiple transistors which has the effect of increased source to body bias of the stacked transistors and subsequent increase in threshold voltage, causing a drop in sub-threshold leakage current.

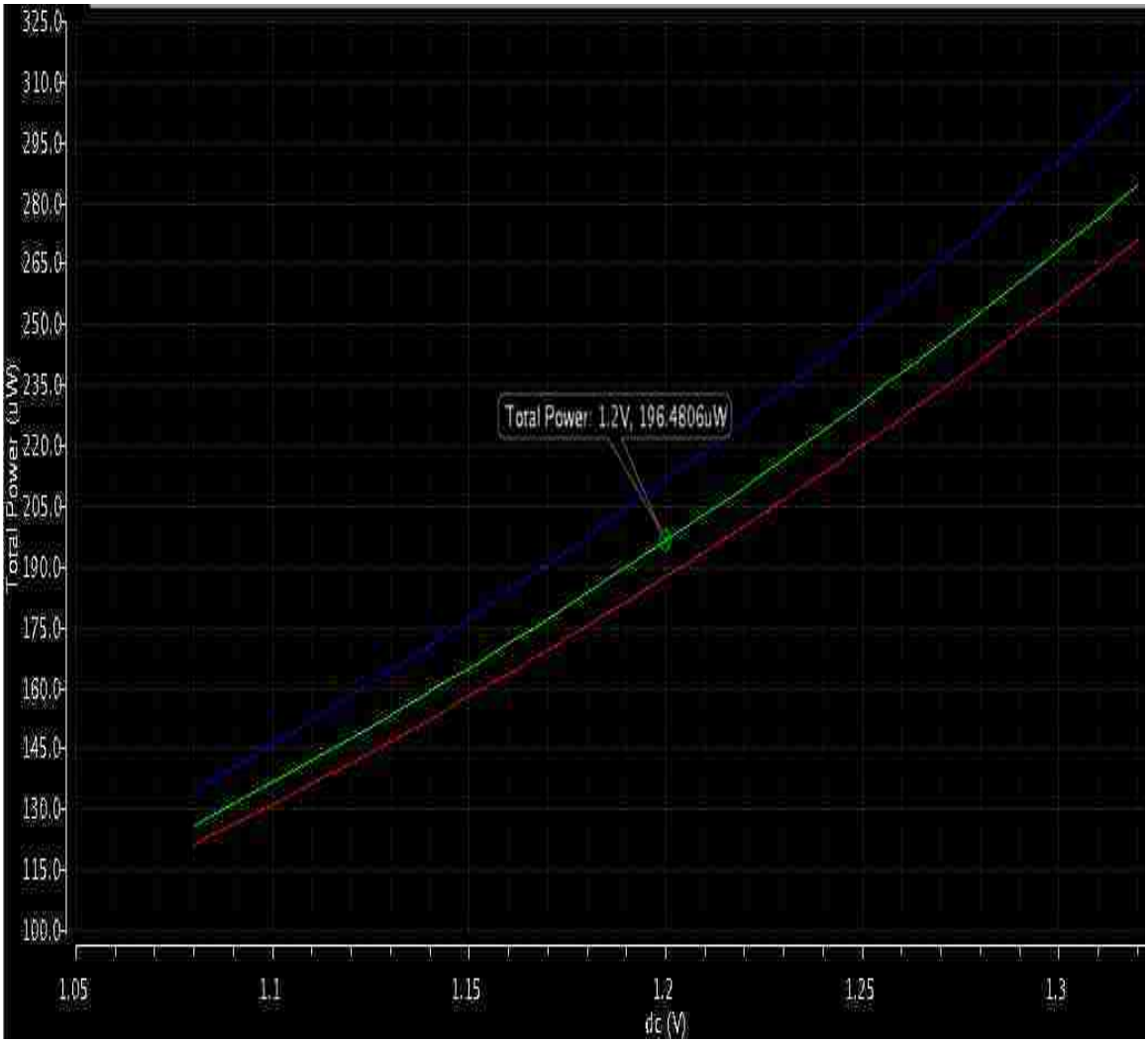


Fig. 131. Total power dissipation of I-PUF primitive at the 9 TV corners

8.3 Comparative Area and Power Characteristics

Area and power consumption of a PUF design is of paramount importance in assessing the viability of implementation of the design. This section presents some of the area and power characteristics of our PUF designs and based on available literature, a comparison of our designs to some of the competing PUF designs have been made as it pertains to area and power.

For our TG-PUF and I-PUF designs, each SMC occupies an area of approx. 500 μm^2, so the total area occupied by the array of 85 SMCs is approx. 42,500 μm^2. If the SMCs are placed adjacent to each other (instead of being distributed as in our design), the array would occupy a 206 µm x 206 µm region. The VDC occupies an area of 136 µm x 60 µm. The area of the digital components, i.e., the LFSR and bit generation engine, is estimated at 300 µm x 300 µm. On-chip memory requirements for the array of 680 NFET and PFET TGs is approx. 2,380 bytes for our design. Therefore, a total area of approx. 140,660 μm^2 per chip would be allocated to the PUF based on our current design and assuming existing on-chip memory could be shared for PUF use. This results in an area per unit base entropy of 500 μm^2 / 16 voltages = 31 μm^2 / unit base entropy (V) for the TG-PUF and 62 μm^2 / unit base entropy (V) for the I-PUF .

Based on available literature, the RO PUF at 130nm node exhibits 43 μm^2 / unit base entropy (frequency) [30]. The area characteristics of the other types of PUFs are listed below in Table X [30][76][78][84].

The total power consumption is calculated as the sum of the static power, dynamic power, and short circuit power. The dynamic and static power consumption, which consists of the charging/discharging of load capacitance and the sub-threshold leakage current, is negligible compared to the short circuit power consumption in our PUF primitives as we have a direct path from V_{DD} to GND through 2 NFETs (or PFETs) in series for the TG-PUF and 2 NFETs and 2 PFETs in series for the I-PUF. Therefore, the short circuit power consumption is dominated by the DC power of approx. 210 µW at 25C, 1.2V per NFET primitive for the TG-PUF. Assuming an approx. 10 nsec on-time per NFET primitive (100Mbps) results in an energy consumption of 210 µW * 1e-8s = 2.1 pJ per NFET primitive of the TG-PUF. The short circuit power consumption for the PFET TG-PUF and the I-PUF are 196 µW and 164 µW respectively. This calculates to an energy consumption of 1.96 pJ and 1.64 pJ for the PFET TG-PUF and the I-PUF respectively.

For comparison, the Arbiter PUF and the Delay-Line PUF exhibit an energy consumption of 0.239 pJ and 0.066 pJ, respectively while the RO PUF exhibits an energy consumption of 244.2 pJ [30]. The IC Identification circuit PUF (ICID) [26][78] exhibits a power consumption of 120 µW and at a throughput of 5 Mbps, calculates to an energy consumption of 24 pJ per bit. This is for 130nm technology to generate an ID length of 256 bits. Table X summarizes the results for these different technologies.

The sub-threshold leakage power component of the static power consumption of our PUFs are minimal due to the higher V_{th} and an inherent stacked transistor design of our PUF primitive, the design of which is known to be very resilient to leakage current

due to the increased source to body bias of the stacked transistors and subsequent increase in threshold voltage [79][80]. Additionally, Adaptive Reverse Body Biasing (RBB) techniques can be applied to further reduce the leakage component [81] of scaled devices where the leakage component may be unacceptable. However, an estimate of the sub-threshold leakage power and dynamic power of our NFET TG-PUF primitive is 6.5pW and 4nW at 25C, 1.2V, respectively.

Table X. Area, Power, and Energy characteristics of various PUF designs [30][76][78][84][85].

PUF type	**Area per unit base entropy**	**Power/Energy dissipation**	**Technology**
TG-PUF	31 μm^2	196µW/1.96pJ	90nm
I-PUF	62 μm^2	164µW/1.64pJ	90nm
RO PUF	43 μm^2, 62 μm^2 1000 μm^2	244.2pJ 10^4 pJ	65nm, 130nm 65nm
Arbiter PUF	1089 μm^2 3000 μm^2	0.239 pJ 0.5 pJ	130nm 65nm
Delay-line PUF	Unavailable	0.066 pJ	65nm
SRAM PUF	0.81 μm^2 2.5 μm^2	100µW 0.09 pJ	65nm 65nm
DFF PUF	11.9 μm^2	Unavailable	130nm
Buskeeper PUF	4.63 μm^2	Unavailable	65nm
ICID	4.63 μm^2	120µW/24pJ	130nm

Chapter 9

Conclusions and Future Work

Both of the novel PUFs described in this work have demonstrated the production of high quality bit strings that perform exceptionally well under stringent statistical metrics including stability, randomness, and uniqueness on all 63 chips tested. The 63 chips with 85 SMCs designed on each chip allowed for a robust sample size that allowed for statistically sound conclusions to be drawn.

The generated bitstrings were tested with industry-standard NIST tests to validate that they were of cryptographic quality. The stability of these bitstrings was evaluated using exhaustive controlled environmental testing at industry-standard 9 TV corner ratings on all 63 chips. This work was unique in the fact that a full-blown 9 TV corner testing was conducted on a sample size of this extent.

Significant was the fact that the TG-PUF bitstrings produced by voltage comparisons and those by VDC digitized voltage comparisons were evaluated and compared using identical quality standards. This allowed for evaluation of the pros and cons of each method of generating the bitstring and more importantly, allowed the evaluation of the penalties involved with the digitization process. Other related work in the PUF research space usually evaluates only the digitized bitstrings, so this research was unique in assessing both voltage-derived bitstrings and digitized bitstrings. It should be noted that this was not possible for the I-PUF as the voltage range to be digitized was much higher than it was for the TG-PUF (for reasons explained earlier) and the limited

capacity of our VDC did not allow for stable I-PUF bitstrings to be generated. Therefore for the I-PUF, only bitstrings generated by voltage comparisons were successfully analyzed.

For the TG-PUF, the % of bits from the original voltage-derived bitstring that were preserved as strong bits was 34.5% and dropped to 14.7% for the VDC-derived bitstrings. This was an indication of the added noise induced by the VDC digitization process. Furthermore, by disabling the threshold technique to get an idea of the underlying noise levels and comparing the intra-chip HD of the voltage-derived bitstrings with that of the VDC-derived bitstring, we saw an increase in its value from 5.11% to 8.68%. This was also an indication of the increased noise levels associated with the digitization process in the TG-PUF. For the I-PUF, the % of bits from the original voltage-derived bitstring that were preserved as strong bits was 22.18%, indicating that the TV noise was higher in the I-PUF as compared to the TG-PUF. Another indication of the higher TV noise levels in the I-PUF was revealed when comparing the intra-chip HD of the TG-PUF voltage-derived bitstrings to that of the I-PUF voltage-derived bitstrings when thresholding was disabled. An increase of the intra-chip HD from 5.11% (for the TG-PUF) to 6.18% (for the I-PUF) was noted indicating the increased TV noise levels.

The increased TV noise levels in the I-PUF was explainable by a comparison of the 0.9mV TV noise observed in the NFET TGVD distribution with the 6mV TV noise observed in the VOD distribution. Analysis of the underlying TV noise associated with the R_{on} of the combined PFET path and the combined NFET path comprising the I-PUF primitive revealed that the R_{on} variation in each of these paths due to TV changes was

much larger than the TG-PUF because of the much higher resistance values of each of these paths in the I-PUF. This also resulted in the TV noise (as reflected in the CV% of the noise) of the VO voltage in the I-PUF to be higher than the TV noise of the TGV voltages of the NFET and PFET TG-PUFs.

The PFET TG-PUF exhibited the least TV noise as indicated by a 0.44mV TGVD TV noise. The higher TGVD TV noise of the NFET TG-PUF compared to the PFET TG-PUF was further understood by analyzing the TV noise levels associated with the R_{on} of the NFET transistors as compared to the PFET transistors. The TV noise in the R_{on} of the NFETs was significantly higher than that of the PFETs and this was driven mainly by larger electron mobility shifts with varying temperature, as shown in Appendix A.

Novel techniques such as thresholding and TMR were demonstrated to improve the inter-chip HD close to an ideal value of 50% and the intra-chip HD to an ideal value of 0%. These achieved metrics are equal or better than those of the predominant competing PUF designs based on literature referenced in this work [38, 69, 75, 76] and Table I. These novel techniques were developed as an alternative to popular error correction and helper data schemes which tend to suffer from additional area overhead/cost and the increased chance of attacks and data compromise. The probability of failure data was also presented and showed how the TMR technique can decrease the probability of failure by several orders of magnitude for a given scaling factor, when used in conjunction with the thresholding technique. Compression techniques such as run length encoding were presented in an effort to reduce the size of the public helper data.

Chapter 9. Conclusions and Future Work

All analyses related to the PUFs stability to TV variations were scrutinized down to the physical parameter level to understand how those were impacting the bitstring stability and causing the unpredictable bit flips. This work lacks precedence in shedding light on evaluating the extent to which physical parameters affect the stability characteristics of these novel PUFs. Analysis of the unstable bits revealed that they were primarily being caused by non-linear and disproportional shifts in the voltages with changing temperature and V_{DD} conditions. Understanding the root cause of this was imperative and led to the analyses of the R_{on} of the transistors with changing temperature and V_{DD} conditions, and this was found to be inducing the non-linear shifts in the voltages. A mathematical model consisting of transistor R_{on} ratios at enrollment conditions and % change of R_{on} from enrollment conditions was derived to better understand the disproportional and unpredictable voltage shifts that were causing the bit flips. Furthermore, it was seen that this non-linear behavior in R_{on} was also a function of the V_{GS} and $V_{DS,}$ and thereby the operating regions of the transistors making up the PUF primitive. These characteristics were simulated and the results were compared and contrasted with the experimental results to reveal a satisfactory match.

The PG-PUF voltage variation as a function of changing temperature and V_{DD} were also analyzed and shown to be linear, but no simulation or bitstring analysis of this PUF was done as that was beyond the scope of this work. A study of two different voltage comparison strategies to generate the bitstring was conducted and revealed vital information related to undesirable voltage bias that proved crucial in selecting a differential comparison approach to generating the final bitstring.

Several strengths of the TG-PUF and I-PUF architectures were highlighted in this work. The salient characteristics were that the architectures were entirely silicon-based and used the existing power grid array of a chip to generate a unique ID that is also digitized on-chip. Also, the architecture scales with process nodes as the primitive consists of minimum-size transistors that are expected to exhibit larger variation with technology scaling. Furthermore, our PUF has no complex ECC circuitry on-chip and having this circuitry on-chip is generally a disadvantage of other PUF architectures as it takes up valuable area and could render the architecture more prone to attacks.

Future work would entail designing a VDC with greater capacity or even looking at different VDC architectures that may exhibit a lot less digitization noise than our current design. As shown on the left side of Fig. 13, integrating the instrumentation used to measure the voltages and to add an offset and control the Cal1 voltage, will be one of the goals for the next version of the chip.

The offset calibration process is challenging and would require the possible implementation of a state machine and temperature feedback circuit. The Cal1 offset voltages can be derived using a resistor-ladder network [82], and added to the TG voltage using a voltage subtractor/adder circuit [83]. The offset only needs to be accurate to approx. 5 mV, which significantly reduces the area overhead of the ladder network. With the availability of these on-chip components, a state machine can be designed to carry out the calibration process.

Also, a closer look at implementing effective compression techniques to reduce the public helper data size would be beneficial as the thresholding and TMR techniques render the public helper data size large.

A review of Table X reveals that the area and power characteristics of the TG-PUF and I-PUF are fairly comparable to other PUF designs based on available literature. However, as expected, there appears to be a large range in the area and power characteristics of the different PUF designs based on the process technology node used. It would be beneficial to implement our PUF designs on a more current process node and evaluate the area and power characteristics or better yet, design some of the predominant competing PUFs on the same chip and evaluate their performance side-by-side. This is also another goal of the next chip design.

Lastly, aging studies of PUFs seems to be picking up in the research community as this has a very practical implication on the PUF usage scenario and reliability. Although our chips have been through numerous temperature and voltage cycles as part of several studies related to the designs on the chips, there have been no controlled aging studies done on our PUF designs. This would be something to explore in the future.

Appendix A

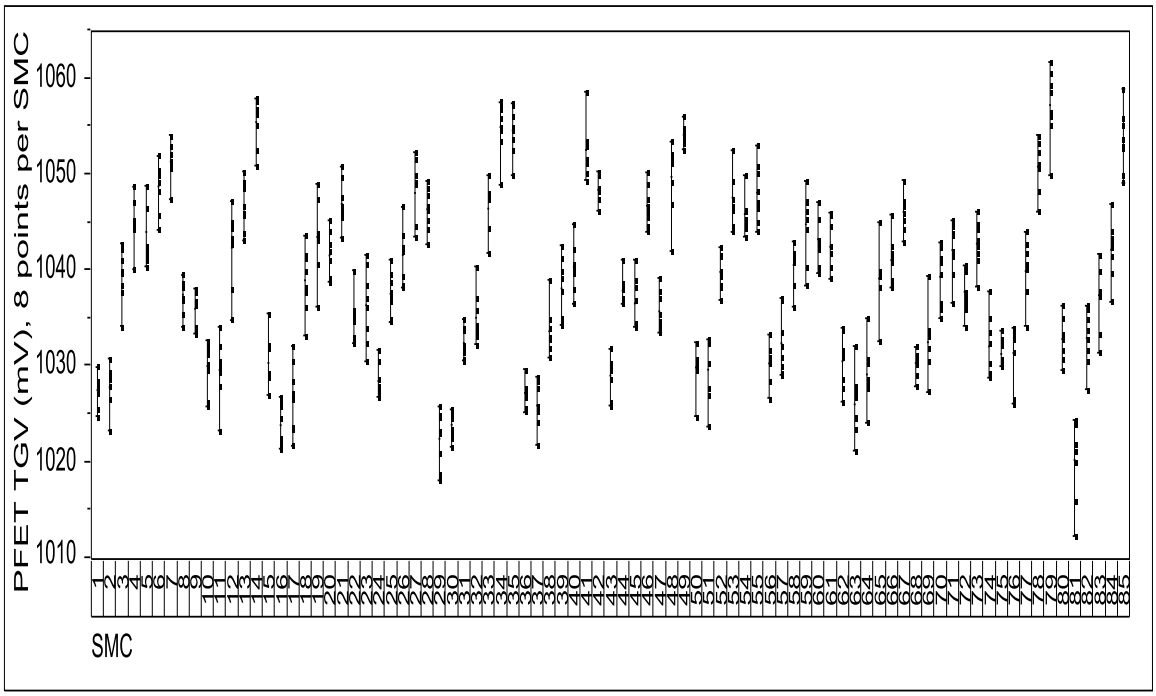


Fig. A1. Illustration of sense wire bias showing up in the voltages of the TG-PUF

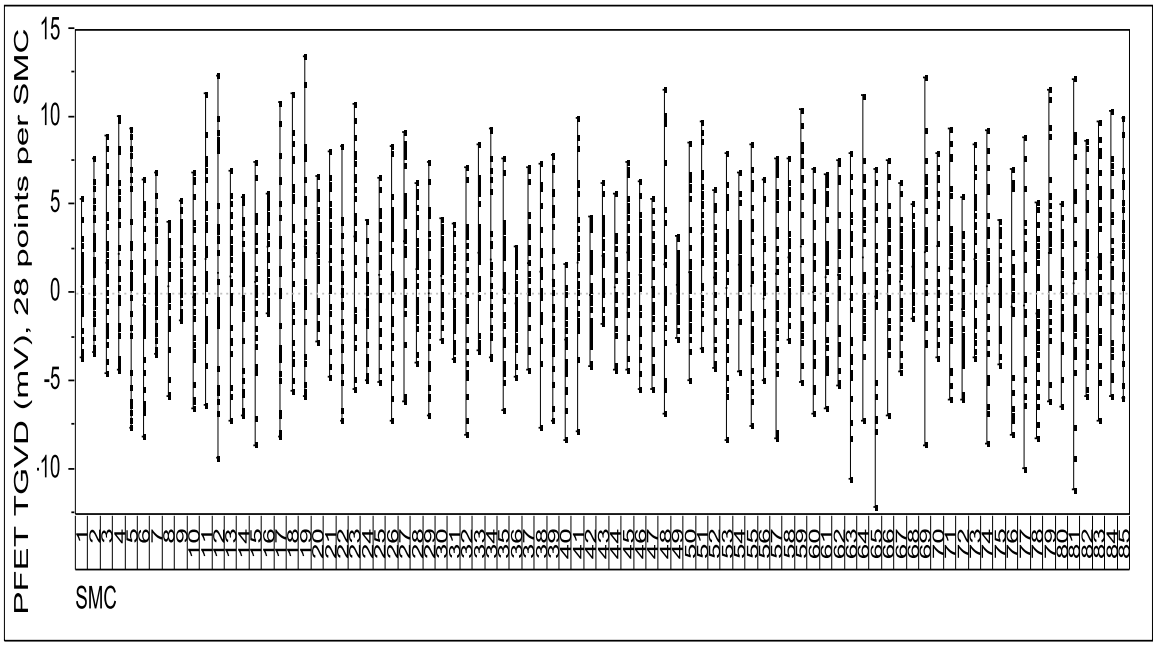


Fig. A2. Illustration of removal of sense wire bias by taking differences of TGV voltages of the TG-PUF

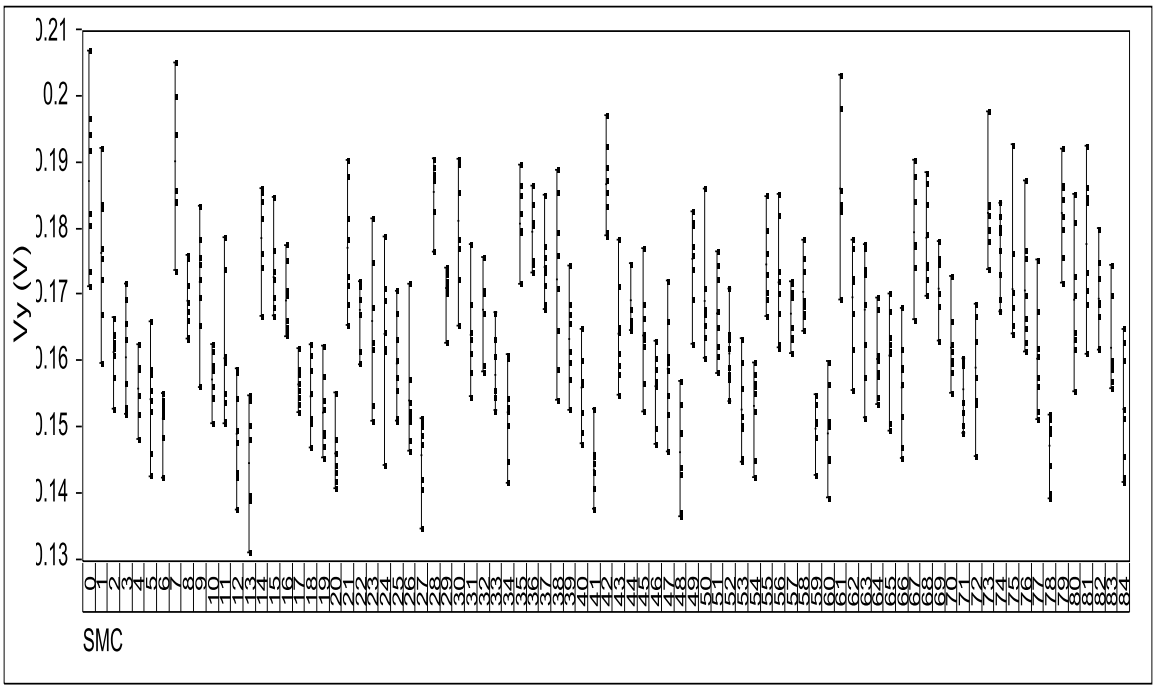


Fig. A3. Illustration of sense wire bias showing up in the voltages of the I-PUF

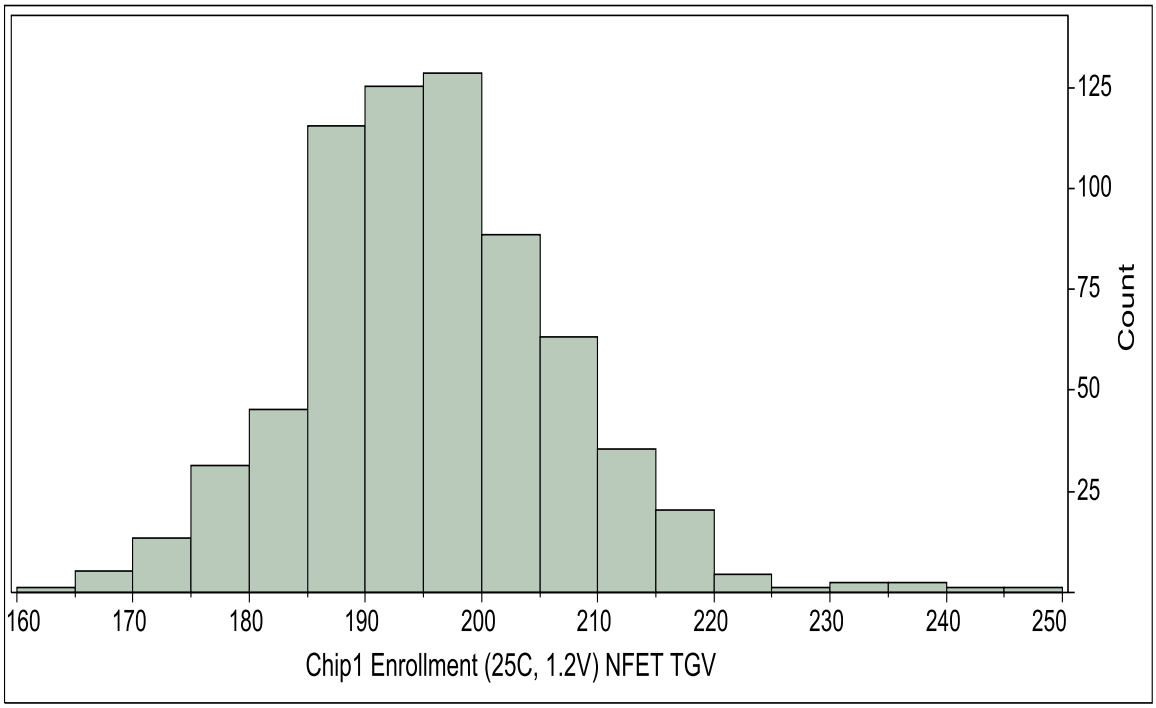


Fig. A4. Chip1 NFET TGV distribution at enrollment

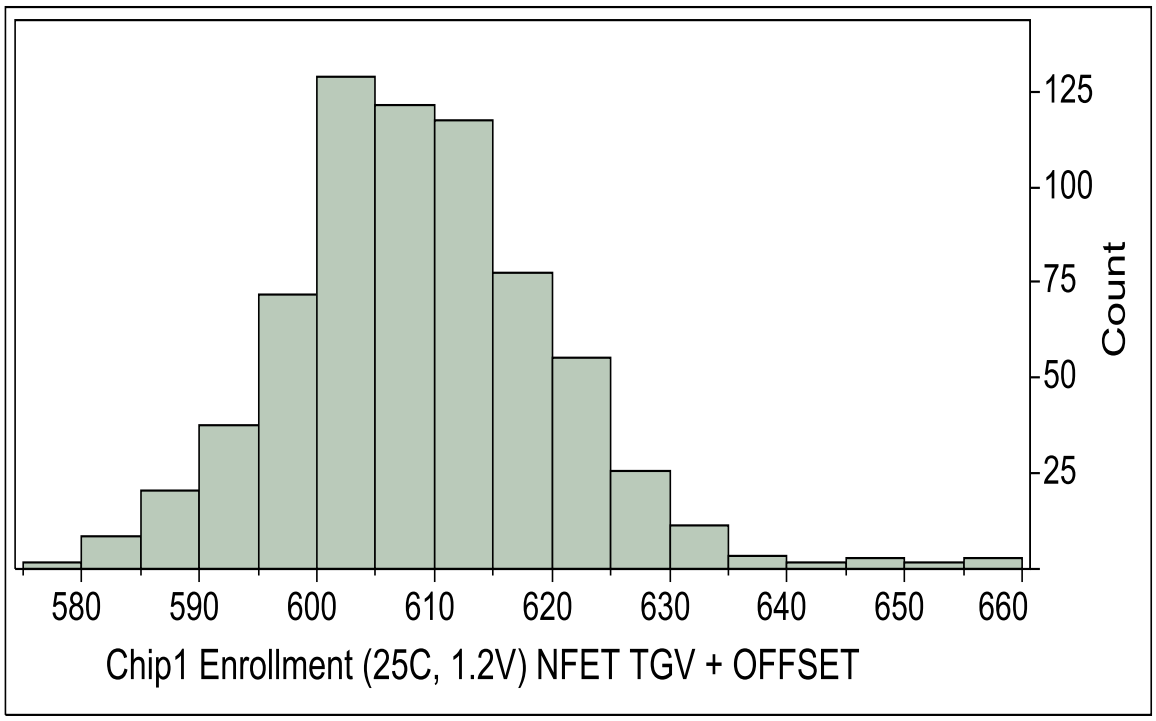


Fig. A5. Chip1 NFET TGV + OFFSET distribution at enrollment

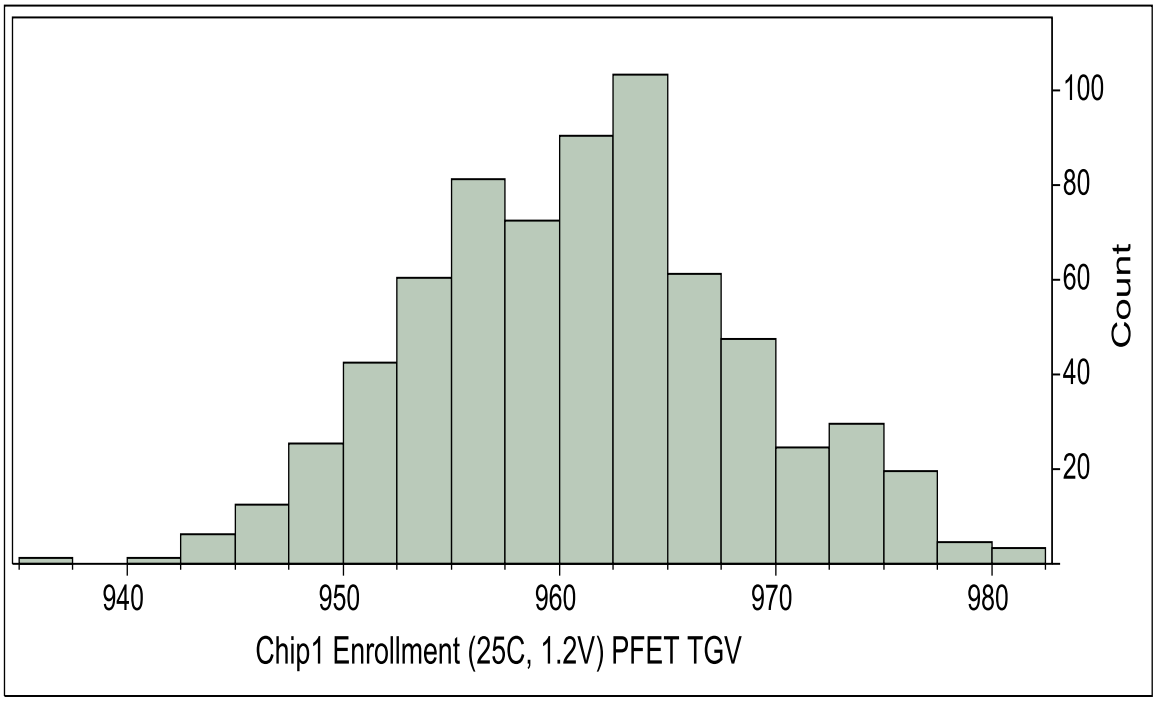


Fig. A6. Chip1 PFET TGV distribution at enrollment

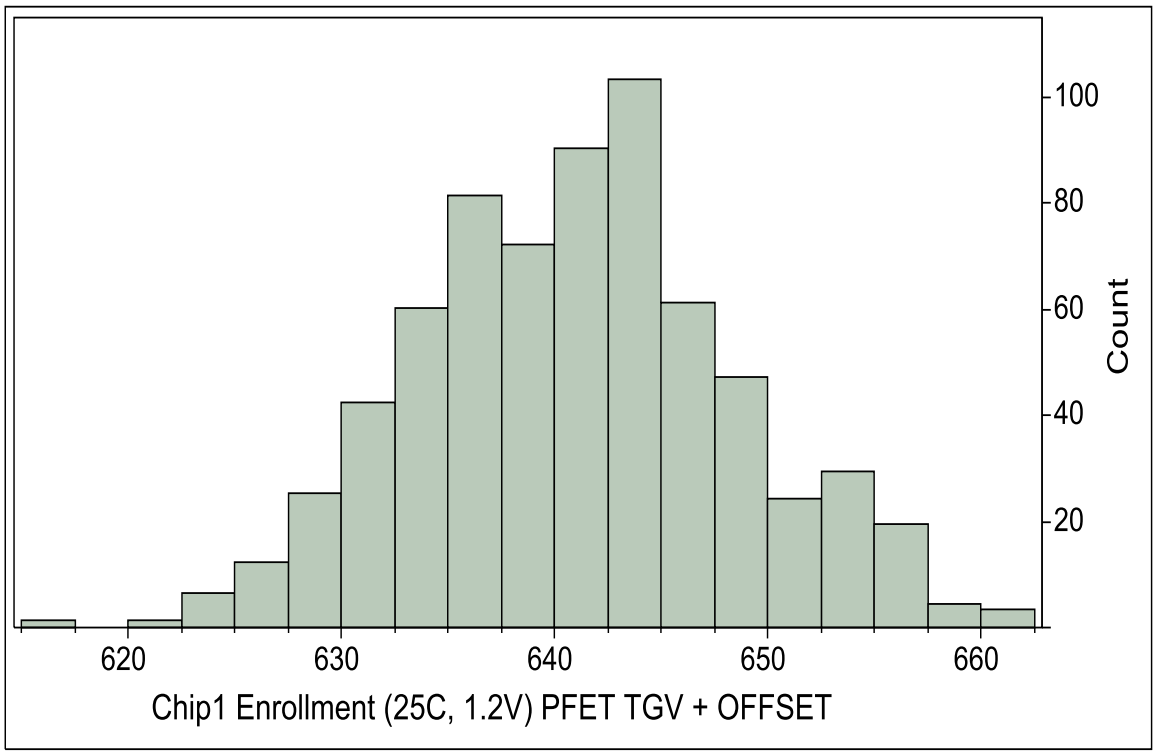


Fig. A7. Chip1 PFET TGV + OFFSET distribution at enrollment

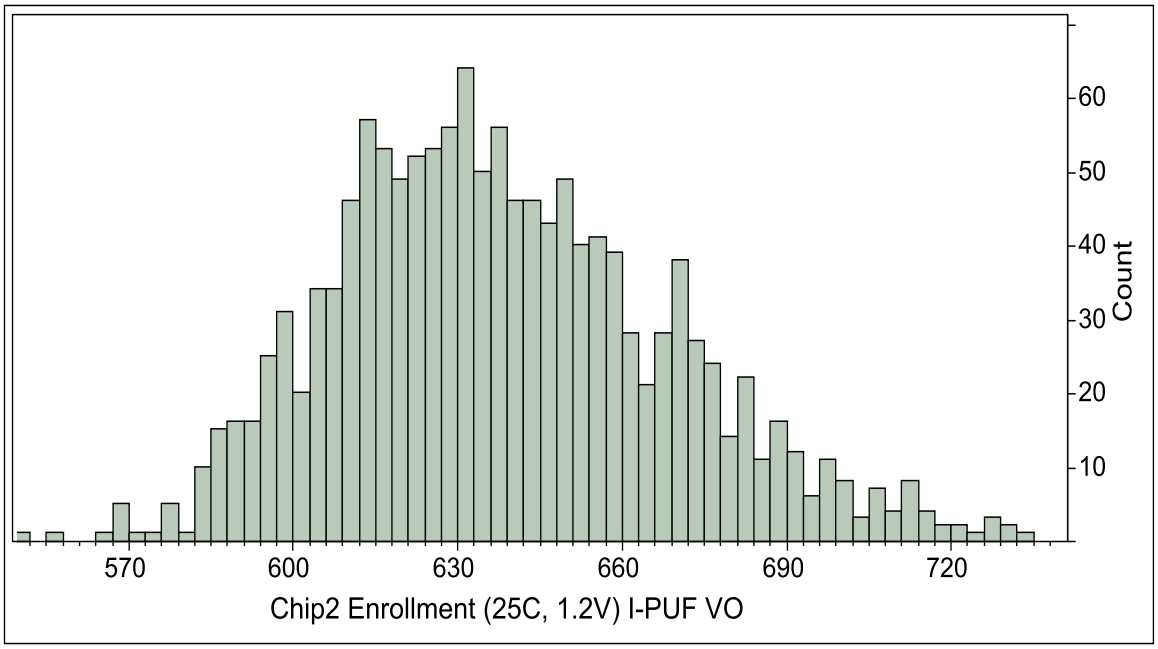


Fig. A8. Chip2 I-PUF VO distribution at enrollment

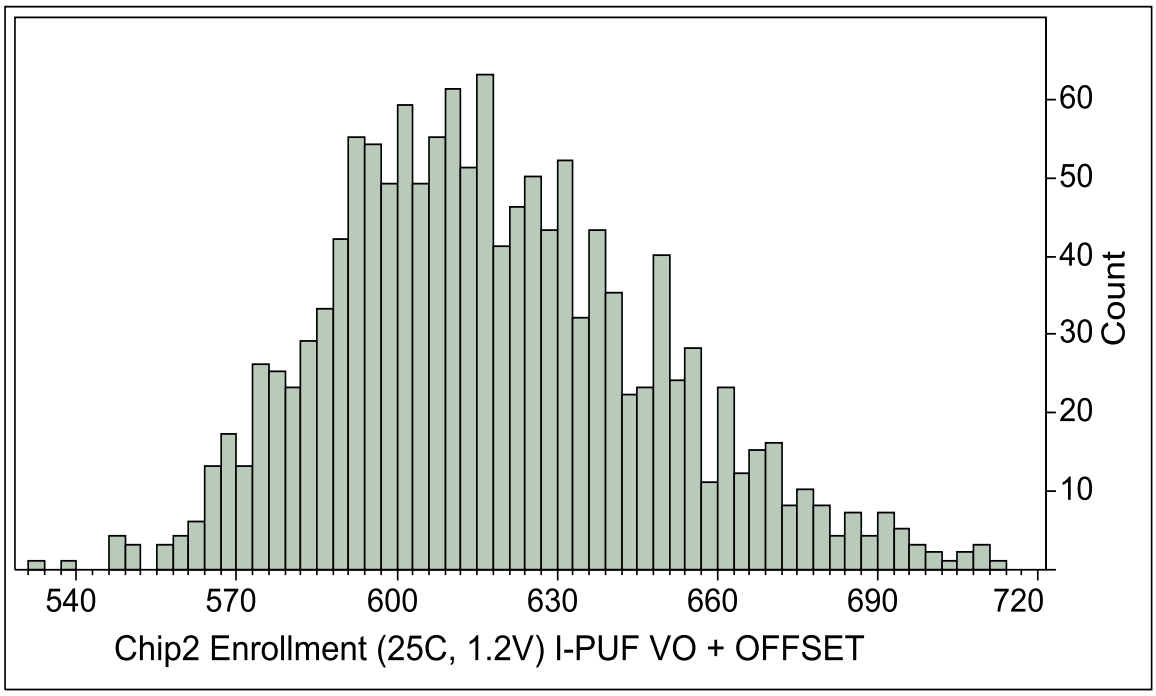


Fig. A9. Chip2 I-PUF VO + OFFSET distribution at enrollment

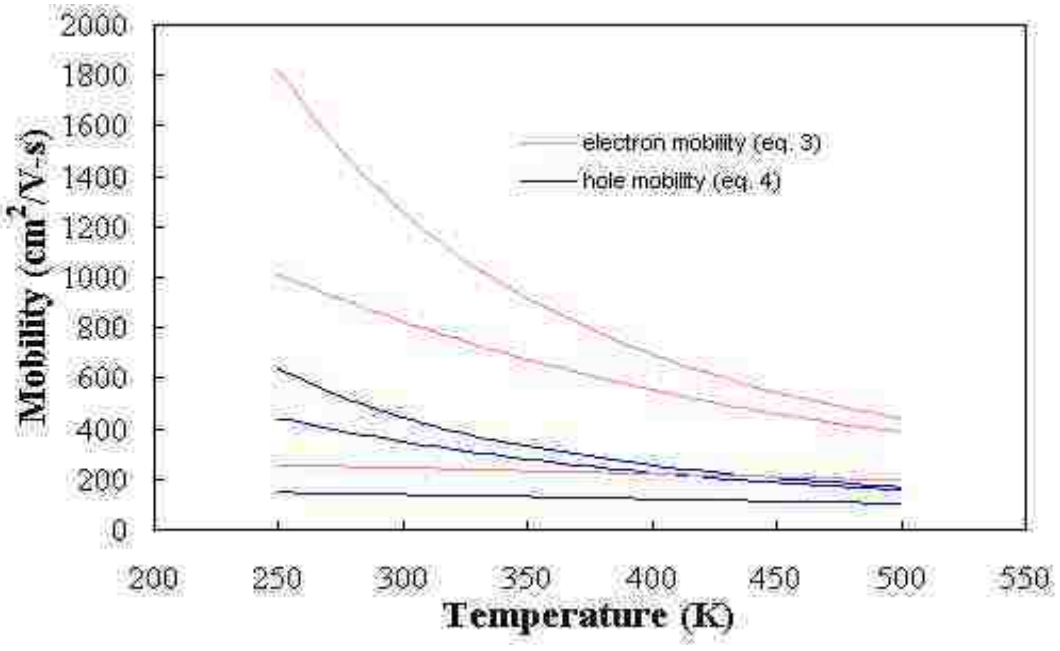


Fig. A10: Example mobility vs. temperature curves for electron and holes [71] for varying doping densities (10^{16} (top curve), 10^{17} and 10^{18} (bottom curve) cm^{-3})

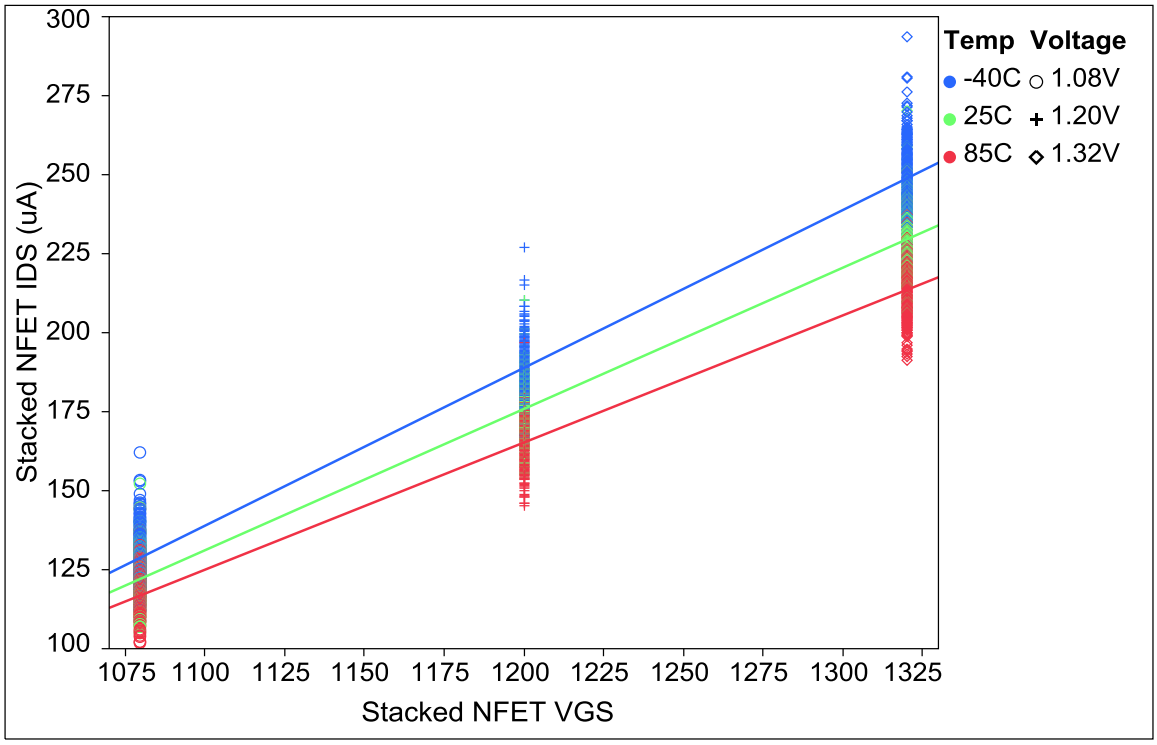


Fig. A11. TG-PUF Stacked NFET I_{DS} vs. V_{GS} at different temperatures

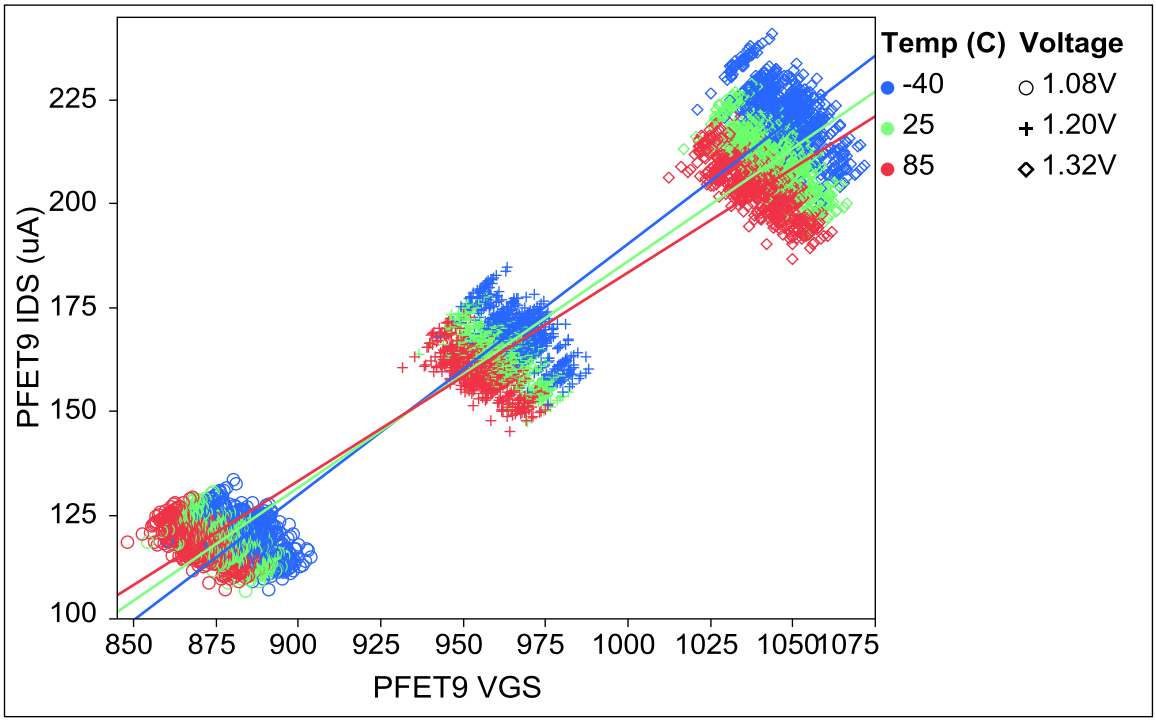


Fig. A12. TG-PUF PFET9 I_{DS} vs. V_{GS} at different temperatures

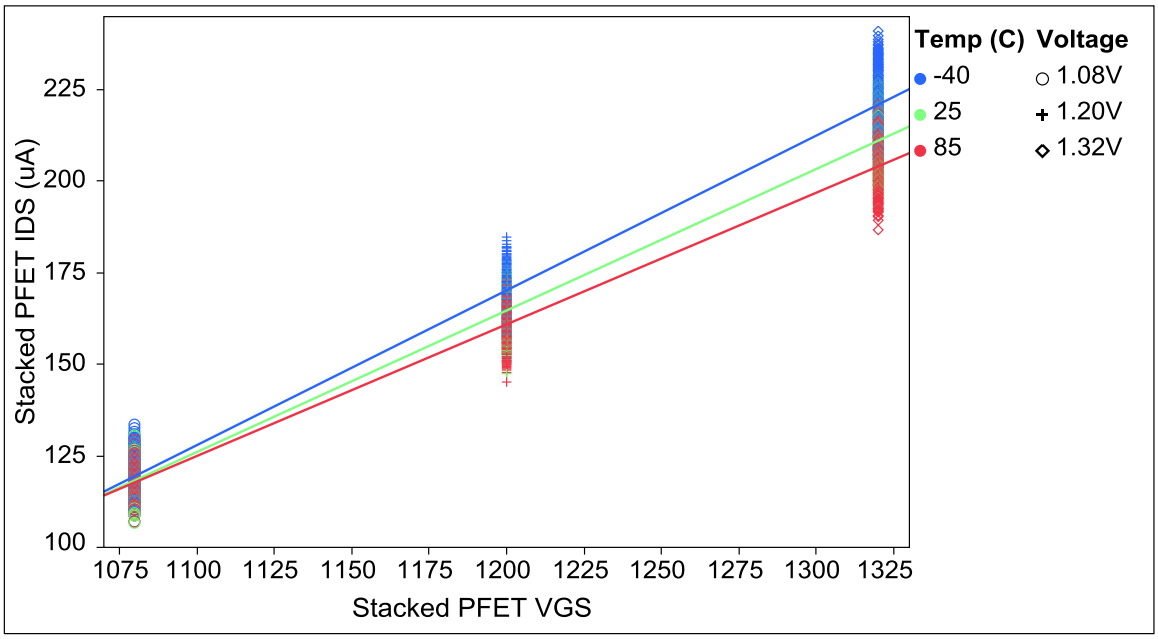


Fig. A13. TG-PUF Stacked PFET I_{DS} vs. V_{GS} at different temperatures

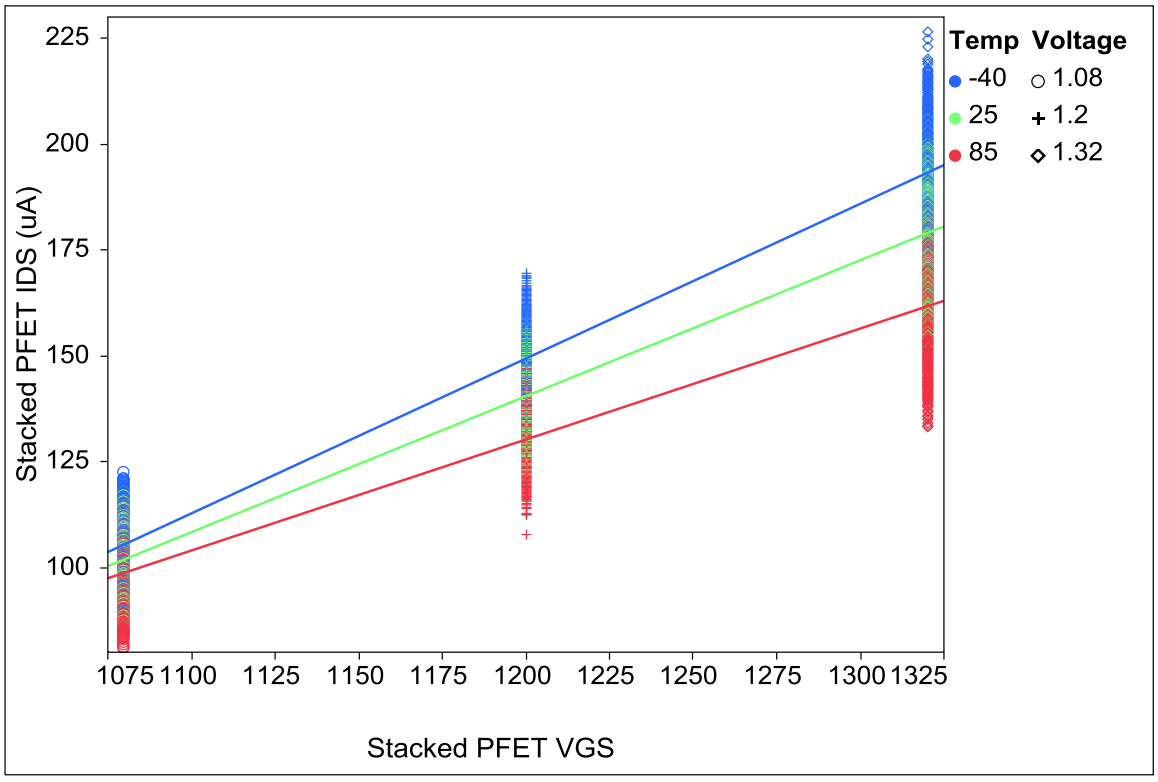


Fig. A14. I-PUF Stacked PFET I_{DS} vs. V_{GS} at different temperatures

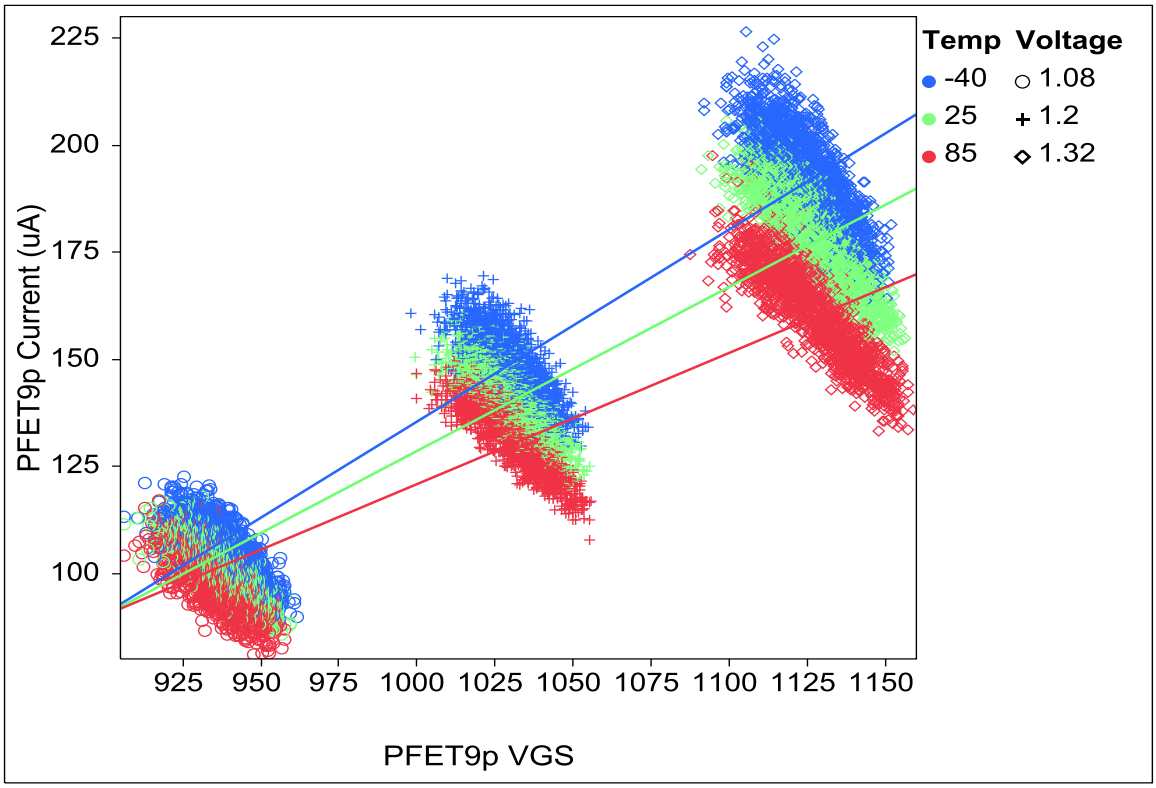


Fig. A15. I-PUF PFET9p I_{DS} vs. V_{GS} at different temperatures

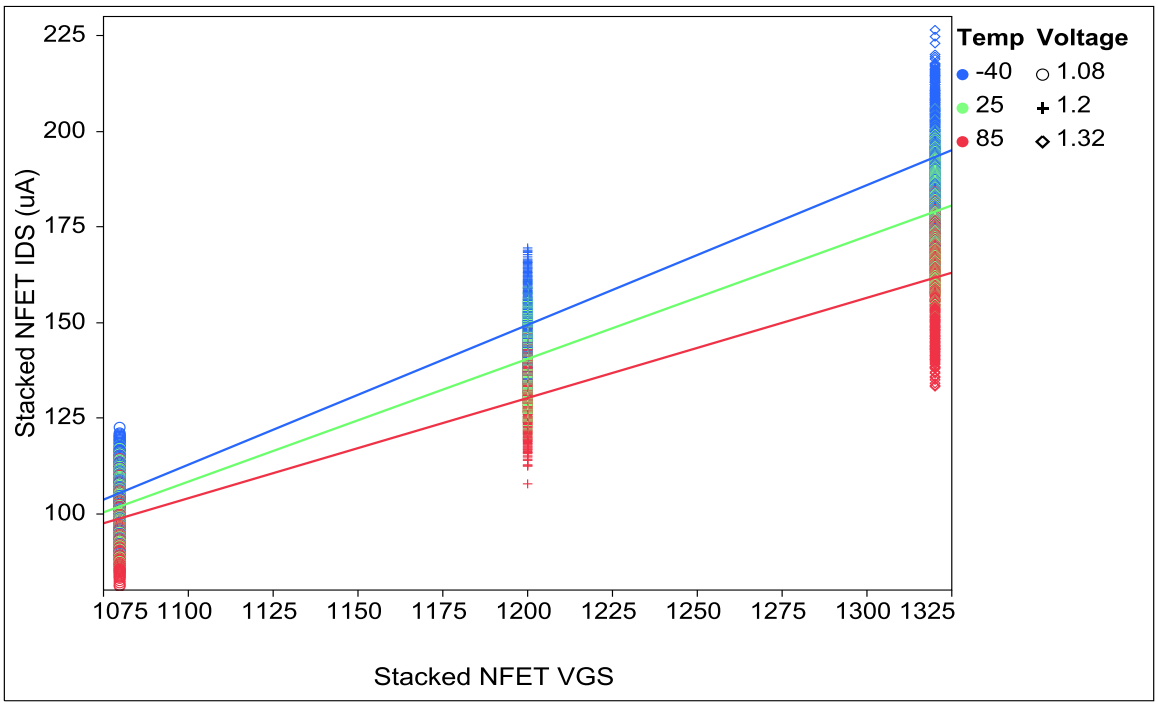


Fig. A16. I-PUF Stacked NFET I_{DS} vs. V_{GS} at different temperatures

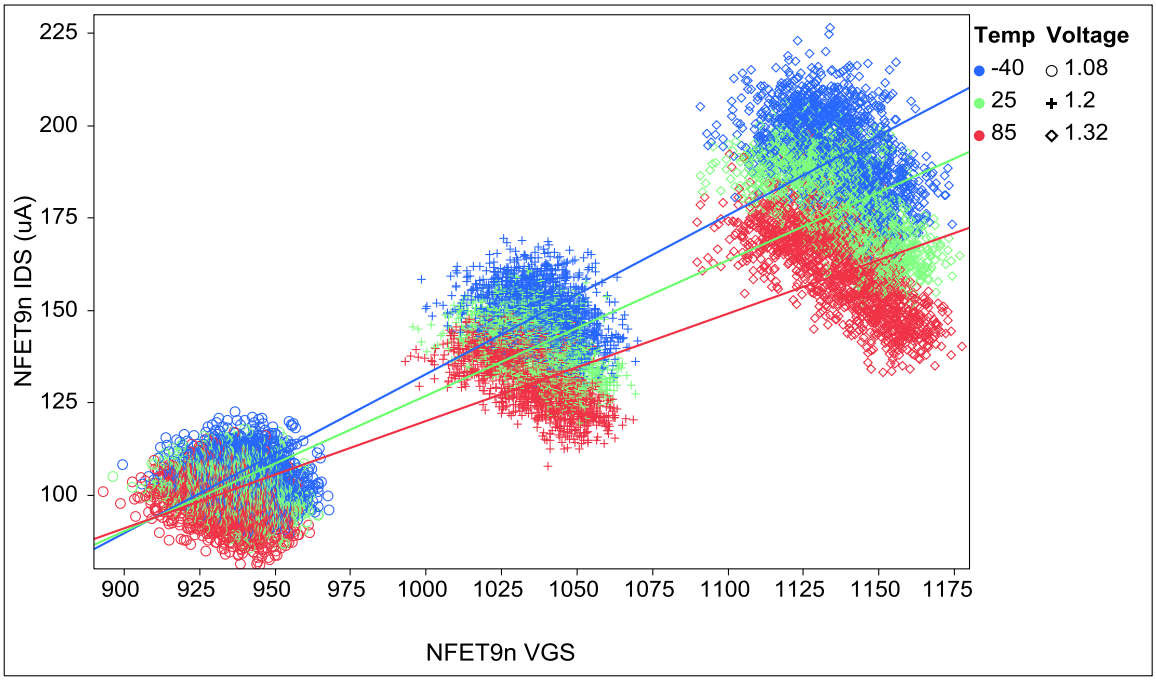


Fig. A17. I-PUF NFET9n I_{DS} vs. V_{GS} at different temperatures

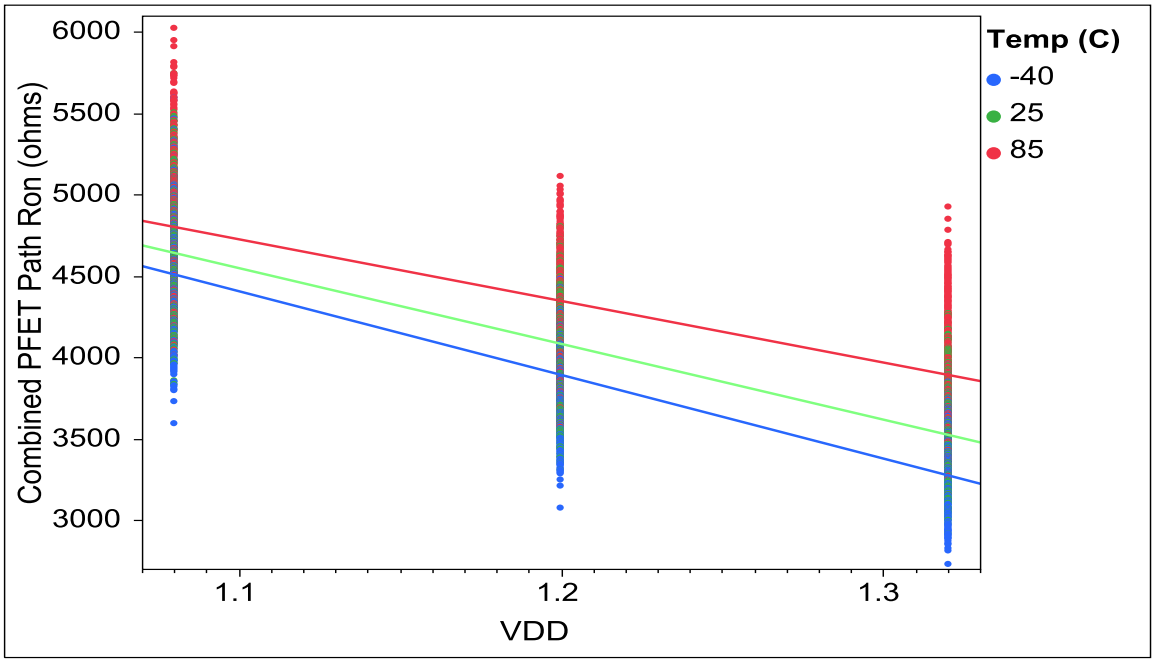


Fig. A18. I-PUF combined PFET path R_{on} at the 9 TV corners

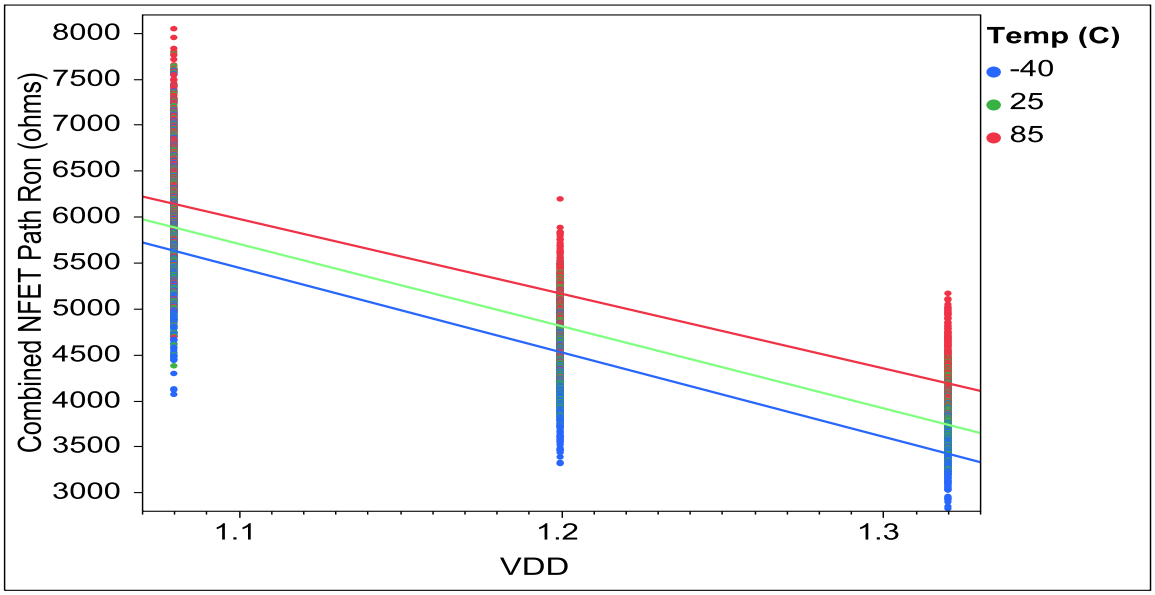


Fig. A19. I-PUF combined NFET path R_{on} at the 9 TV corners

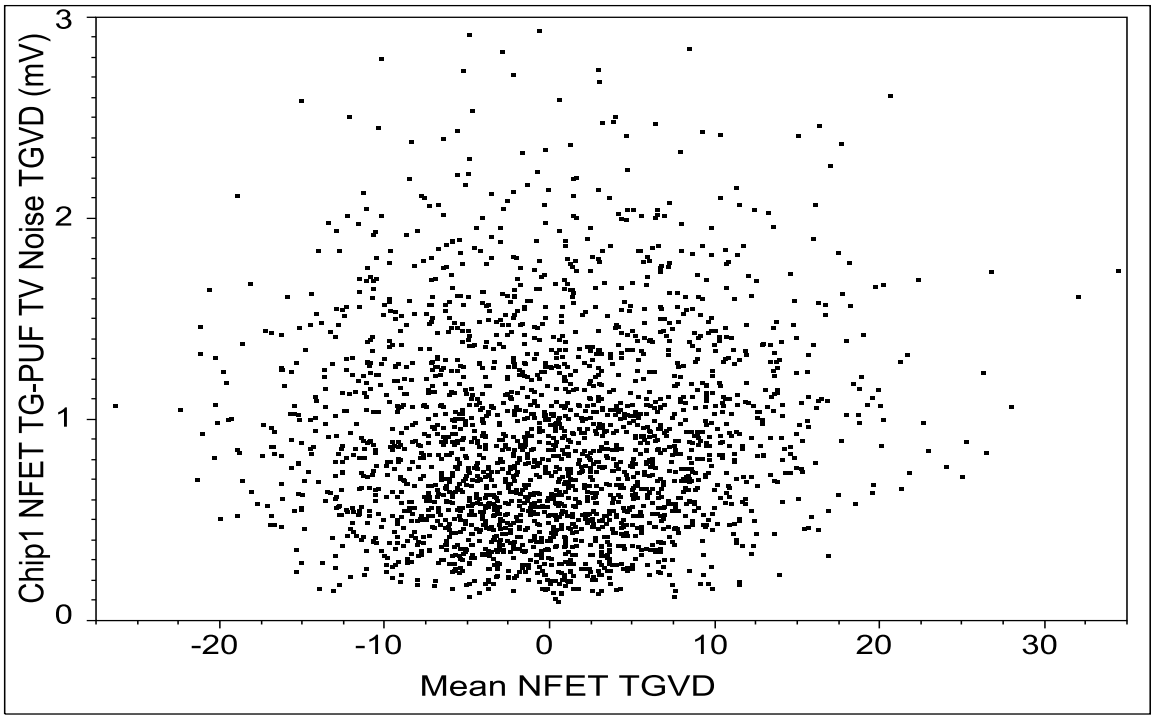


Fig. A20. Chip1 NFET TG-PUF TGVD TV noise vs. TGVD

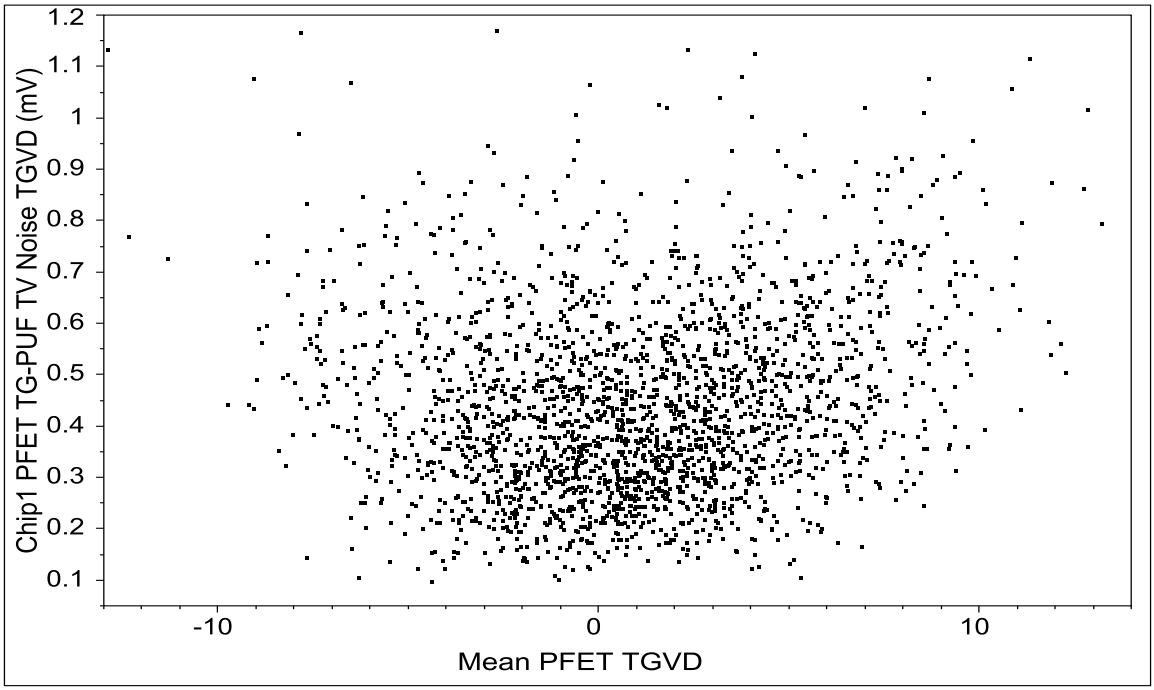


Fig. A21. Chip1 PFET TG-PUF TGVD TV noise vs. TGVD

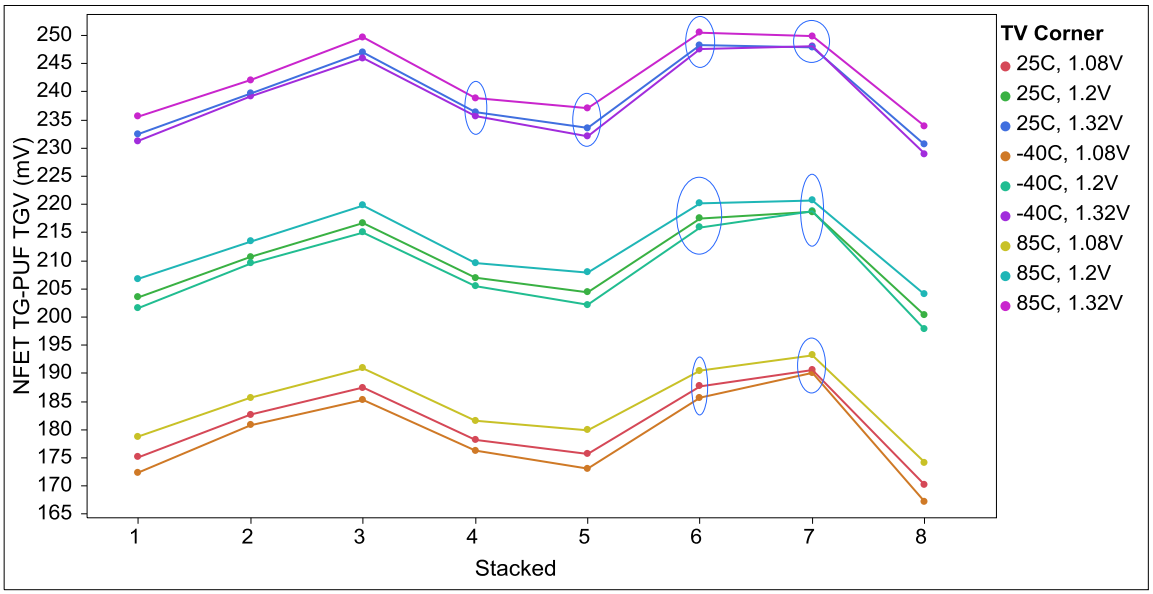


Fig. A22. Unequal shifts in TGV with changing TV for Chip1 SMC53 NFET TG-PUF

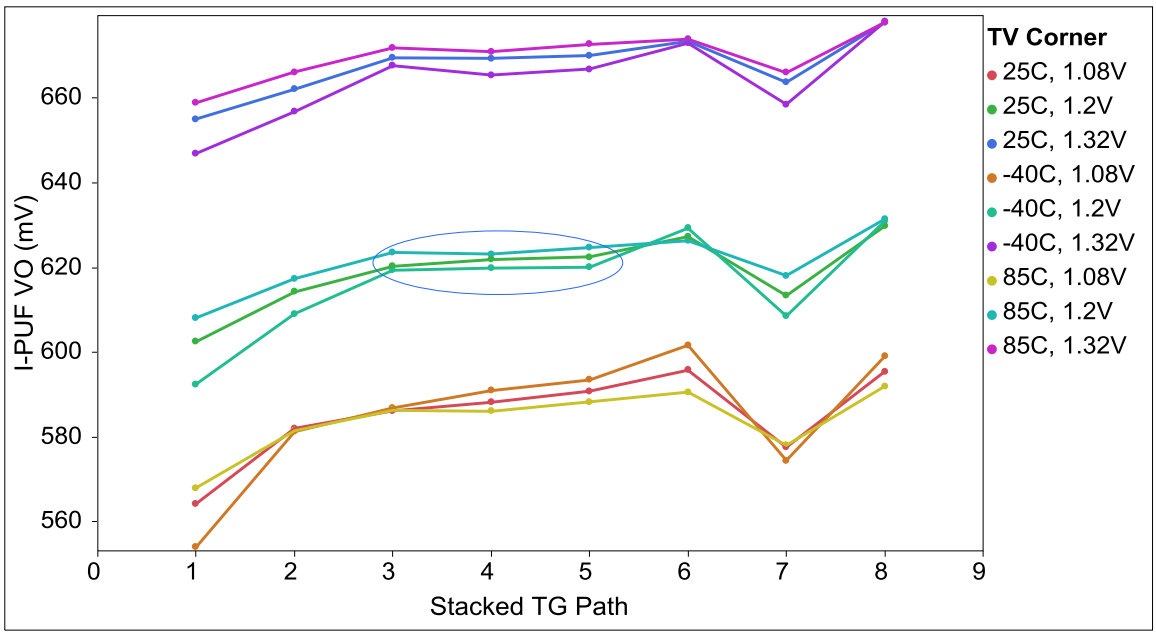


Fig. A23. Unequal shifts in VO with changing TV for Chip2 SMC0 I-PUF

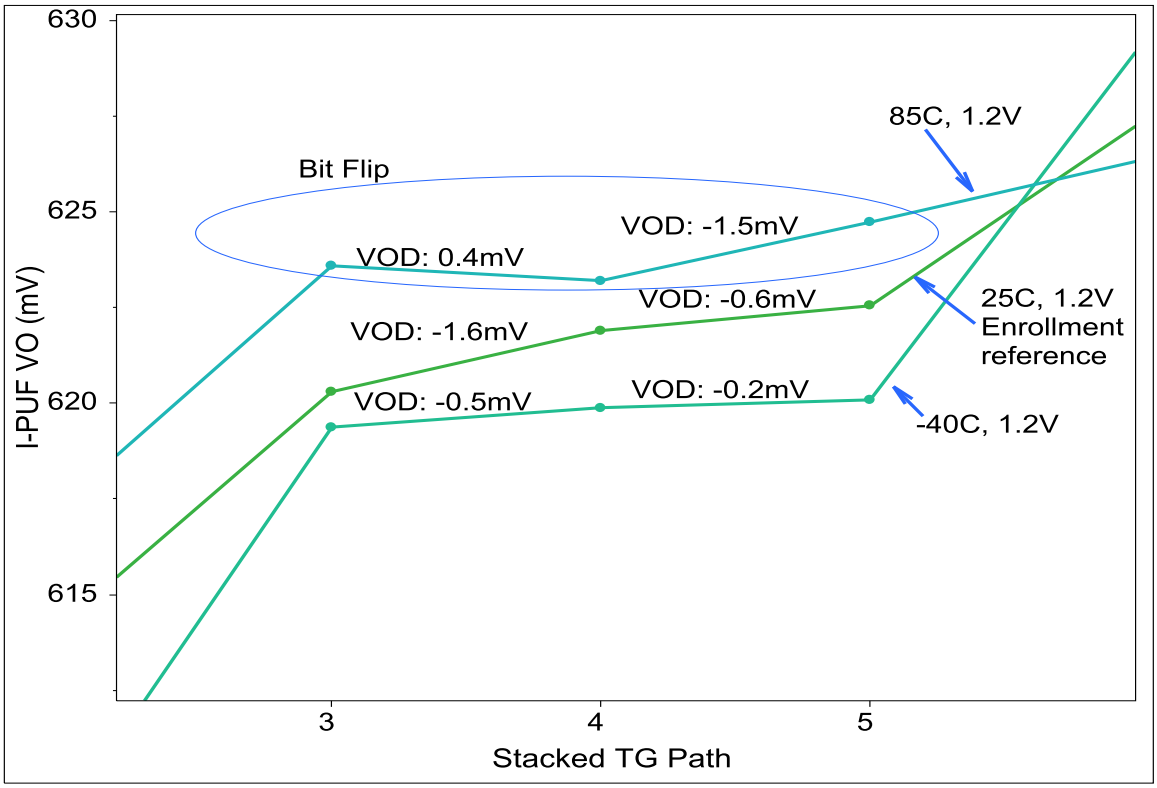


Fig. A24. Magnified view of circled region in Fig. A22 showing unequal shifts in VO with changing TV causing a bit flip

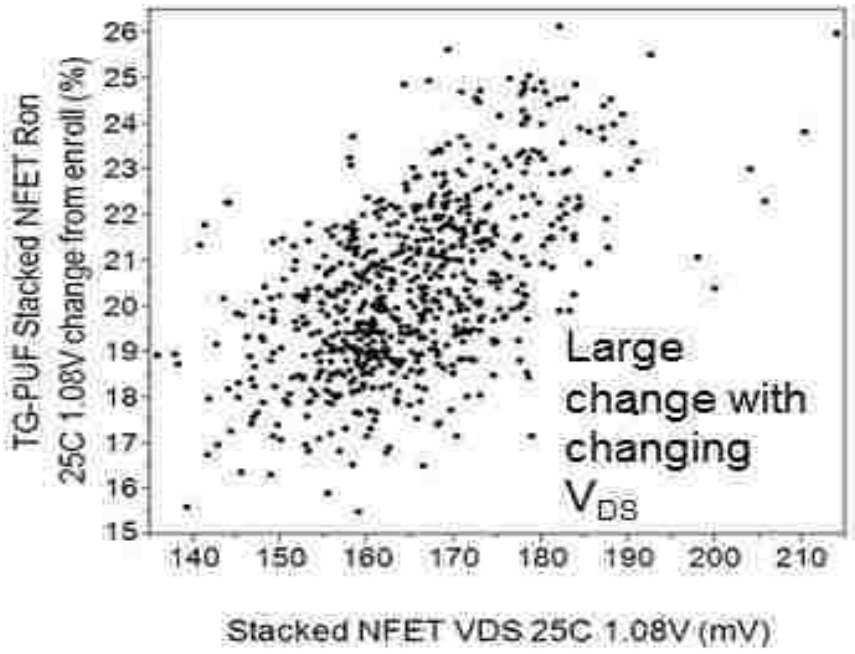


Fig. A25. Large R_{on} changes from enrollment with V_{DS} for stacked NFETs of TG-PUF at 25C, 1.08V

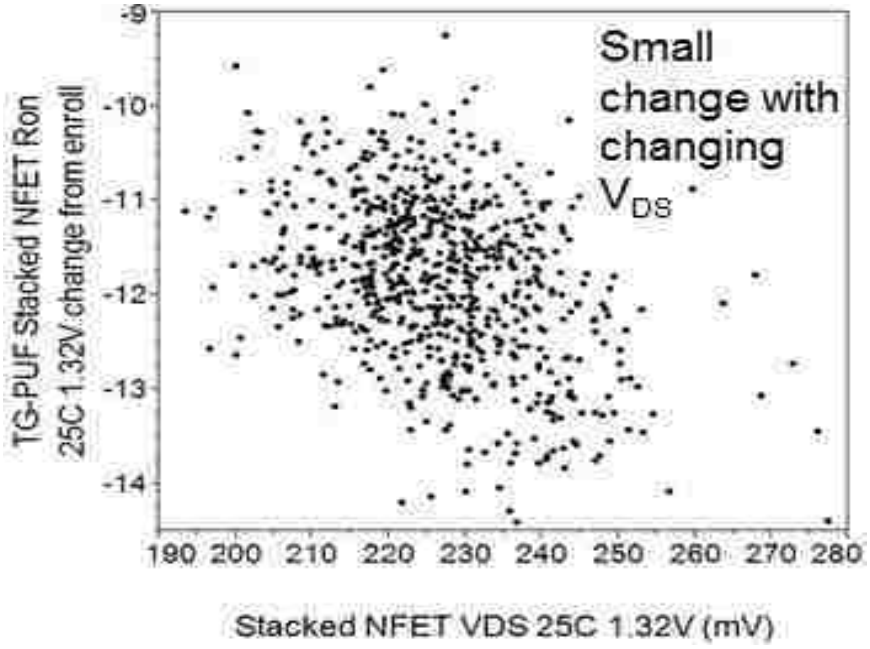


Fig. A26. Small R_{on} changes from enrollment with V_{DS} for stacked NFETs of TG-PUF at 25C, 1.32V

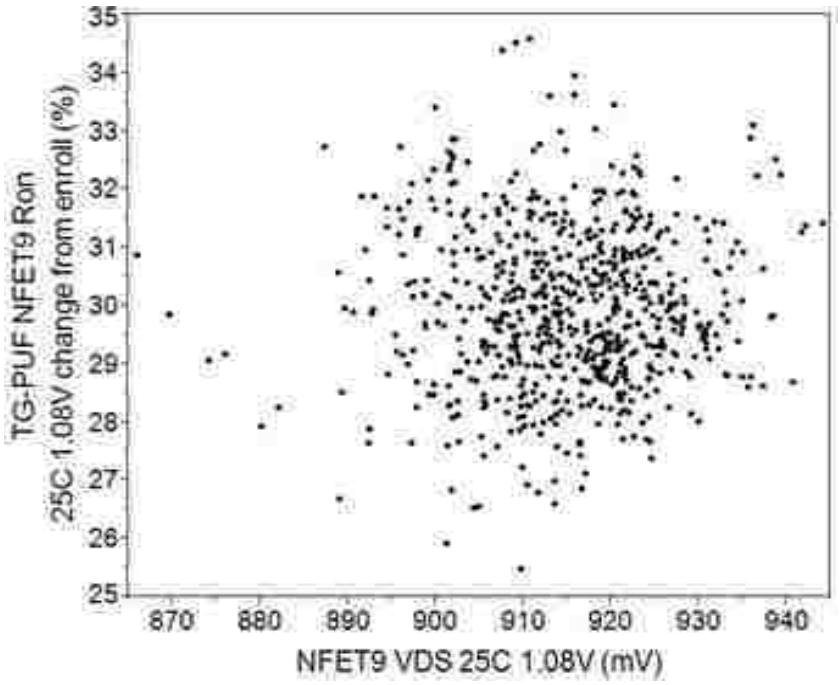


Fig. A27. R_{on} changes from enrollment with V_{DS} for NFET9 of TG-PUF at 25C, 1.08V

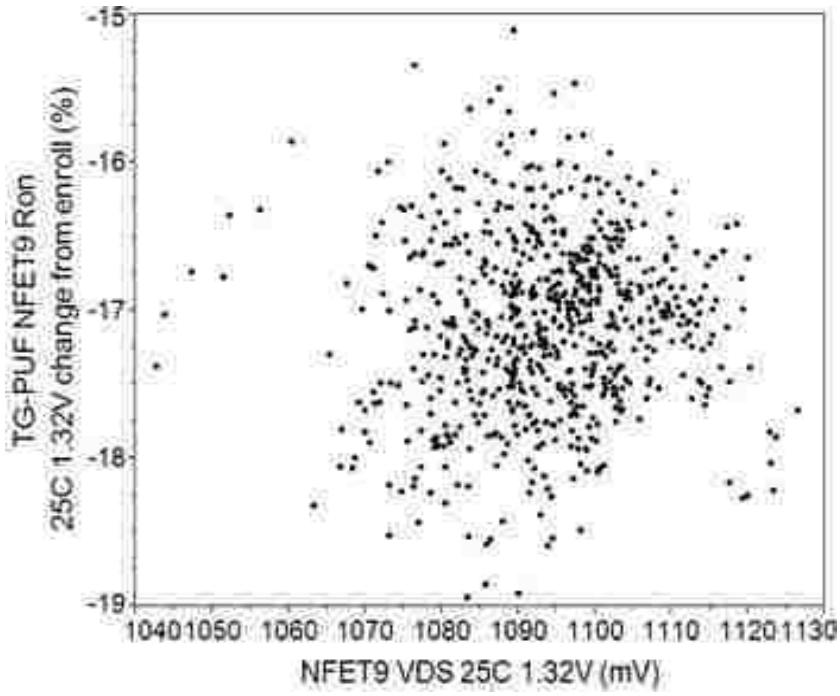


Fig. A28. R_{on} changes from enrollment with V_{DS} for NFET9 of TG-PUF at 25C, 1.32V

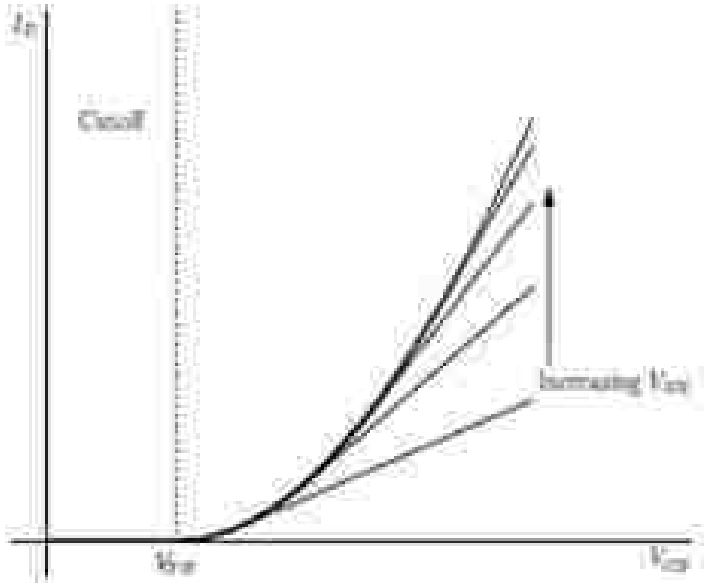


Fig. A29. Example I_{DS} vs. V_{GS} curves for NFET at different V_{DS}

References

[1] NIST: Computer Security Division, Statistical Tests, http://csrc.nist.gov/groups/ST/toolkit/rng/stats_tests.html

[2] R. Maes and I. Verbauwhede, "Physically unclonable functions: a study on the state of the art and future research directions," in Towards Hardware-Intrinsic Security, Springer-Verlag, 2010, pp. 3–37.

[3] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation", in ACM/IEEE Design Automation Conference, pages 9-14, 2007.

[4] L. Kulseng et al., "Lightweight mutual authentication and ownership transfer for RFID systems," in INFOCOM, Mar. 2010, pp. 1–5.

[5] W. Choi et al., "PUF-based Encryption Processor for the RFID Systems," IEEE International Conference on CIT, pp. 2323-2328, 2010.

[6] J. Guajardo et al., "Physical unclonable functions and public-key crypto for FPGA IP protection," in FPL, Aug. 2007, pp. 189–195.

[7] S. Goren et al., "FPGA bitstream protection with PUFs, obfuscation, and multi-boot," in ReCoSoC, Jun. 2011, pp. 1–2.

[8] S. Devadas, E. Suh, S. Paral, R. Sowell, T. Ziola and V. Khandelwal, "Design and Implementation of PUF-Based Unclonable RFID ICs for Anti-Counterfeiting and Security Applications", in IEEE International Conference on RFID, 2008.

[9] P.F. Cortese, F. Gemmiti, B. Palazzi, M. Pizzonia and M. Rimondini, "Efficient and Practical Authentication of PUF-Based RFID Tags in Supply Chains", in IEEE International Conference on RFID-Technology and Applications, June 2010.

[10] J. Huang and J. Lach, "IC Activation and User Authentication for Security-Sensitive Systems," in IEEE International Workshop on Hardware-Oriented Security and Trust, HOST 2008.

[11] J.A. Roy, F. Koushanfar and I.L. Markov, "EPIC: Ending Piracy of Integrated Circuits", in Design, Automation and Test in Europe, 2008.

[12] L. Bolotnyy and G. Robins, "Physically Unclonable Function -Based Security and Privacy in RFID Systems", in PerCom 2007.

[13] http://www.verayo.com.

[14] http://www.intrinsic-id.com.

[15] J. Riordan and B. Schneier, “Environmental Key Generation Towards Clueless Agents,” in Mobile Agents and Security. London, UK, UK: Springer-Verlag, 1998. [Online]. Available: http://dl.acm.org/citation.cfm?id=648051.746194

[16] TCG, “Mobile Trusted Module Specification, Version 1.0, Revision 7.02,” Trusted Computing Group, Tech. Rep., 2010.

[17] M. Majzoobi, F. Koushanfar, and S. Devadas, “FPGA-Based True Random Number Generation Using Circuit Metastability with Adaptive Feedback Control”, Proc. CHES 2011.

[18] K. Wold and C. H. Tan, “Analysis and enhancement of random number generator in FPGA based on oscillator rings”, Reconfigurable Computing and FPGAs, International Conference on, 0:385–390, 2008.

[19] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, “Controlled physical random functions”, In Proceedings of 18th Annual Computer Security Applications Conference, December 2002.

[20] V. van der Leest, E. Sluis, G.-J. Schrijen, P. Tuyls, and H. Handschuh, “Efficient implementation of true random number generator based on SRAM PUFs”, in Cryptography and Security: From Theory to Applications, ser. Lecture Notes in Computer Science, D. Naccache, Ed. Springer Berlin Heidelberg, 2012, vol. 6805, pp. 300–318.

[21] J. Ju, R. Chakraborty, R. Rad, J. Plusquellic, “Bit String Analysis of Physical Unclonable Functions based on Resistance Variations in Metals and Transistors”, HOST, 2012, pp. 13-20.

[22] Ju J., Chakraborty R., Lamech C., Plusquellic J, “Stability Analysis of a Physical Unclonable Function based on Metal Resistance Variations”, In: 6th International Workshop on Hardware-Oriented Security and Trust (HOST’13) Austin, TX, USA (2013)

[23] M. Yu and S. Devadas, “Secure and robust error correction for physical unclonable functions”, IEEE Des. Test. Comput., Jan. 2010, pp. 48–64.

[24] Delvaux, et.al, “Key-recovery Attacks on Various RO PUF Constructions via Helper Data Manipulation”, Technical Report, IACR, June 2012

[25] R. Pappu, R. Recht, and J. Taylor, “Physical one-way functions,” Science, pp. 2026 2030, Sep. 2002.

[26] K. Lofstrom, et al., "IC Identification Circuits using Device Mismatch," SSCC, 2000, pp. 372-373.

[27] M. Majzoobi, et al., "Lightweight Secure PUFs", ICCAD, 2008.

[28] G. Qu and C. Yin, "Temperature-Aware Cooperative Ring Oscillator PUF", Workshop on HOST, 2009, pp. 36-42.

[29] A. Maiti and P.Schaumont, "Improving the Quality of a Physical Unclonable Function using Configurable Ring Oscillators", FPLA, 2009. pp. 703-707.

[30] Y. Meng-Day, et al., "Performance Metrics and Empirical Results of a PUF Cryptographic Key Generation ASIC," HOST, 2012, pp. 108-115.

[31] S. S. Mansouri and E. Dubrova, "Ring Oscillator Physical Unclonable Function with Multi Level Supply Voltages", ICCD, 2012, pp. 520-521.

[32] S. Maeda, et al., "An Artificial Fingerprint Device (AFD): a Study of Identification Number Applications Utilizing Characteristics Variation of Polycrystalline Silicon TFTs," Trans. on Electron Devices, number 50, issue 6, June, 2003, pp.1451- 1458.

[33] M. Bhargava, et al., "Reliability Enhancement of Bi-Stable PUFs in 65nm Bulk CMOS", HOST, 2012, 79-83.

[34] Y. Alkabani, et al., "Trusted Integrated Circuits: A Nondestructive Hidden Characteristics Extraction Approach," Information Hiding, 2008.

[35] R. Helinski, et al., "Physical Unclonable Function Defined Using Power Distribution System Equivalent Resistance Variations", DAC, 2009, pp. 676-681.

[36] K. Kursawe, et al., "Reconfigurable Physical Unclonable Functions - Enabling Technology for Tamper-Resistant Storage", HOST, 2009, pp.22-29.

[37] K. Rosenfeld, et al., "Sensor Physical Unclonable Functions", HOST, 2010, pp. 112-117.

[38] W. Xiaoxiao and M. Tehranipoor, "Novel Physical Unclonable Function with Process and Environmental Variations", DATE, 2010, pp. 1065-1070.

[39] L. Lin, et al., "Low-Power Sub-Threshold Design of Secure Physical Unclonable Functions", LPED, 2010, pp. 43-48.

[40] U. Ruhrmair, et al., “Applications of High-Capacity Crossbar Memories in Cryptography”, Trans. on Nanotechnology, Volume: 10, Issue: 3, 2011, pp. 489-498.

[41] P. Simons, et al., “Buskeeper PUFs, a Promising Alternative to D Flip-Flop PUFs”, HOST, 2012, pp. 7-12.

[42] A. Maiti and P. Schaumont, “A Novel Microprocessor-Intrinsic Physical Unclonable Function,” FPLA, 2012, pp. 380-387.

[43] A. Sreedhar and S. Kundu, “Physically Unclonable Functions for Embeded Security based on Lithographic Variation”, DATE, 2011, pp. 1-6.

[44] S. Meguerdichian and M. Potkonjak, “Device Aging-Based Physically Unclonable Functions”, DAC, 2011, pp. 288-289

[45] P. Tuyls et al., “Read-proof hardware from protective coatings”, CHES, 2006, pp. 369–383.

[46] L. Lin, S. Srivathsa, D. K. Krishnappa, P. Shabadi and W. Burleson, “Design and Validation of Arbiter-Based PUFs for Sub-45nm Low-Power Security Applications”, submitted to IEEE Transactions on Information Forensics and Security. August 2012

[47] L.-T. Pang and B. Nikolic, “Measurement and analysis of variability in 45nm strained-Si CMOS technology," in Custom Integrated Circuits Conference, 2008. CICC 2008. IEEE, Sept. 2008, pp. 129-132.

[48] P. Sedcole and P. Cheung, “Within-die delay variability in 90nm FPGAs and beyond," in Proceedings of IEEE International Conference on Field Programmable Technology, 2006.

[49] Ahmad-Reza Sadeghi and D. Naccache, “Towards Hardware-Intrinsic Security”, Springer, 1st Edition, 2010

[50] A. Maiti, V. Gunreddy, and P. Schaumon, “A systematic method to evaluate and compare the performance of physical unclonable functions," in IACR ePrint, 2011.

[51] C. Bohm, M. Hofer, and W. Pribyl, “A microcontroller SRAM PUF”, in Network and System Security (NSS), 2011 5th International Conference on, Sept. 2011, pp. 269-273

[52] M. Hofer and C. Bohm, “An alternative to error correction for SRAM-like PUF," in CHES - Workshop on Cryptographic Hardware and Embedded Systems, p. 335350.

[53] 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, June 2008, pp. 67-70.

[54] S. Morozov, A. Maiti, and P. Schaumon, “An analysis of delay based puf implementations on FPGA," in Reconfigurable Computing: Architectures, Tools and Applications, vol. 5992, 2010, pp. 382-387.

[55] B. Gassend, “Physical Random Functions," Master's thesis, MIT, MA, USA, 2003.

[56] 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, ser. CCS '02. New York, NY, USA: ACM, 2002, pp. 148-160

[57] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, “A large scale characterization of RO-PUF," in Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on, June 2010, pp. 94-99

[58] D. Lim et al., “Extracting secret keys from integrated circuits,” VLSI Systems, vol. 13, no. 10, pp. 1200–1205, Oct. 2005.

[59] M. Majzoobi, F. Koushanfar, and S. Devadas, “FPGA PUF using programmable delay lines," in Information Forensics and Security (WIFS), 2010 IEEE International Workshop on, Dec. 2010, pp. 1-6

[60] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh, “Quantitative and statistical performance evaluation of arbiter physical unclonable functions on FPGAs," in Recongurable Computing and FPGAs (ReConFig), 2010 International Conference on, Dec. 2010, pp. 298-303.

[61] Daihyun, et.al, “Extracting secret keys from integrated circuits,” Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 13, no. 10, pp. 1200-1205, Oct. 2005.

[62] J. Aarestad, P. Ortiz, D. Acharyya, J. Plusquellic, “HELP: A Hardware-Embedded Delay PUF”, IEEE Design & Test, Volume: PP, Issue: 99, March/April, 2013, pp. 1-8

[63] Ruhrmair, et.al, “Modeling attacks on physical unclonable functions. In CCS 2010: Proceedings of the 17th ACM conference on Computer and communications security, pages 237-249. ACM, New York, NY, USA, 2010.

[64] J.W. Lee, D. Lim, B. Gassend, G.E. Suh, M. van Dijk, and S. Devadas, “A technique to build a secret key in integrated circuits for identification and authentication

applications," in IEEE Symposium on VLSI Circuits, VLSIC 2004, pp. 176-179, Jun. 2004.

[65] Guajardo, Sandeep S. Kumar, Geert Jan Schrijen, and Pim Tuyls, "FPGA intrinsic PUFs and their use for IP protection", In Pascal Paillier and Ingrid Verbauwhede, editors, Cryptographic Hardware and Embedded Systems (CHES), volume 4727 of Lecture Notes in Computer Science, pages 63-80. Springer Berlin / Heidelberg, Berlin, Heidelberg, September 2007.

[66] Dieter Schuster, Side-Channel Analysis of Physical Unclonable Functions (PUFs). Master's thesis, Technische Universitat Munchen, 2010.

[67] P. Tuyls, B. Skoric, "Strong Authentication with PUFs", In: Security, Privacy and Trust in Modern Data Management, M. Petkovic, W. Jonker (Eds.), Springer, 2007.

[68] Dominik Merli, Dieter Schuster, Frederic Stumpf, and Georg Sigl, "Semi-invasive EM attack on FPGA RO PUFs and countermeasures", In Proceedings of the Workshop on Embedded Systems Security, WESS'11, pages 2:1-2:9. ACM, New York, NY, USA, 2011.

[69] D. Nedospasov, C. Helfmeier, J.-P. Seifert, and C. Boit, "Invasive PUF analysis," in Fault Diagnosis and Tolerance in Cryptography (FDTC), 2013 Workshop on, 2013.

[70] Chakraborty, R., Lamech, C., Acharyya, D., Plusquellic, J, "A Transmission Gate Physical Unclonable Function and on-chip Voltage-to-digital Conversion Technique", In: DAC, p. 59 (2013), http://doi.acm.org/10.1145/2463209.2488806

[71] Principles of Semiconductor Devices and Heterojunctions, B. Van Zeghbroeck, 2008, Prentice Hall, pp 180-182, Sec 2.7.2

[72] R. Chakraborty, J. Plusquellic, "Analyses of a Transmission Gate, Power Grid, and Inverter Physical Unclonable Function's Stability to Changing Temperature and Voltage Conditions', IEEE Transactions in VLSI, Nov 2013 (Submitted)

[73] L. Guansheng, Y.M. Tousi, A. Hassibi and E. Afshari, "Delay-Line-Based Analog-to Digital Converters", Trans. On CAS II, Volume: 56, Issue: 6, 2009, pp. 464-468.

[74] I. M. Filanovsky, "Mutual Compensation of Mobility and V_T Temperature Effects with Applications in CMOS Circuits", IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications, VOL.48, NO. 7, July 2001

[75] B. Skoric, P. Tuyls, W. Ophey, "Robust Key Extraction from Physical Uncloneable Functions", Chapter in Applied Cryptography and Network Security, 2005.

[76] Maes, R., Rožić, V., Verbauwhede, I., Koeberl, P., van der Sluis, E., and van der Leest, V, "Experimental Evaluation of Physically Unclonable Functions in 65 nm CMOS", In European Solid-State Circuits Conference – ESSCIRC 2012.

[78] K. Lofstrom, "ICID–A robust, low cost integrated circuit identification method, ver. 0.9," KLIC, Mar. 2007 [Online]. Available: http://www.kl-ic.com/white9.pdf

[79] Janaka Rani et.al, "Analysis of Pseudo-NMOS logic with reduced static power in deep sub-micron regime", Proceedings of International Conference on Advances in Electronics and Communications Engineering ECE2012, July 2012, pp 1 -4

[80] M. Johnson, et. al., "Leakage Control with Efficient Use of Transistor Stacks in Single Threshold CMOS," IEEE TVLSI, Feb 202.0

[81] Yang, J. and Y. Kim, "Self adaptive body biasing scheme for leakage power reduction in nanoscale CMOS circuit", Proceedings of the great lakes symposium on VLSI. Salt Lake City, Utah, USA, ACM: 111-116, 2012

[82] Dan O'Sullivan & Tom Igoe, "Physical Computing: Sensing and Controlling the Physical World with Computers," Thomson Course Technology Publishers, 2004, pp 388-391.

[83] R. Fried and C. C. Enz, "Simple and Accurate Voltage Adder/Subtractor," Electronics Letters, vol. 33, 1997, pp. 944-945.

[84] M. Kassem, M. Mansour, A. Chehab, and A. Kayssi, "A sub-threshold SRAM based PUF," in International Conference on Energy Aware Compuling (ICEAC), Dec. 2010, pp. 1 -4.

[85] "Comparison of bi-stable and delay-based physical unclonable functions from measurements in 65nm bulk CMOS," Center for Silicon Systems Implementation (CSSI) Technical Report, CSSI 12-1, http://www.ece.cmu.edu/ cssi/, Tech. Rep., April 2012.