Emerging non-volatile memory based physical unclonable functions

Zhang, Le

2016

Zhang, L. (2016). Emerging non-volatile memory based physical unclonable functions.
Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/66438>

<https://doi.org/10.32657/10356/66438>

Emerging Non-Volatile Memory Based Physical Unclonable Functions

Zhang Le

School of Electrical and Electronic Engineering
Nanyang Technological University

A thesis submitted to Nanyang Technological University in partial
fulfillment for the degree of
Doctor of Philosophy

March 2016

*The worthwhile problems are the ones you can really solve or help solve, the ones you can really contribute something to.
... No problem is too small or too trivial if we can really do something about it.*

- Richard Feynman

Acknowledgements

I would like to express my deepest gratitude to my supervisors, without whom I cannot finish the Ph. D. program successfully. Firstly, I would like to thank Assistant Professor Kong Zhi Hui, who offered me the opportunity to pursue a Ph. D. degree in Nanyang Technological University (NTU) with sufficient financial aid. Her encouraging words were always the most powerful magic to relieve my stress and disperse my frustration. Then I would like to thank Associate Professor Chang Chip Hong, who has been mentoring me as a co-supervisor. Prof. Chang has influenced and remoulded me in many different ways. I appreciate every bit of knowledge and experience he has shared with me. His meticulous mind and passionate attitude to doing what he had committed to do greatly inspired me.

I also would like to thank those who helped me during the four years. Thank Professor Guido Torelli and Doctor Alessandro Cabrini for the invaluable discussion and earnest measurements on our collaborative project. Thank Professor Kaushik Roy and Doctor Xuanyao Fong for providing the opportunity to work with them and a group of talented Ph. D. students in Nanoelectronic Lab in Purdue University. Thank all my friends, especially the Chip Family members, the Nanovators and those working in VIRTUS. The great moments I have had with them are unforgettable.

In the last, special thanks go to my family members, who have been supporting me to finish the Ph. D. program all the way. Thank my parents. It was their love that motivated me coming through the tough journey. Also, thank my wife, Feng Xue, for accompanying me and tolerating my bad temperament.

Abstract

Cybersecurity has drawn tremendous attentions from academia and industry in the recent years, thanks to the proliferation of extensively connected devices and computers. While software-defined service has been demonstrated effective in many other applications, safeguarding sensitive information that virtually resides in the communication channels, computation nodes and data storage elements remains a critical issue that may require sagacious solution leveraging on the advantageous attributes of hardware. Hardware-based security solutions for accelerating cryptographic primitives like hash function and block cipher using hardware fabrics on FPGA and ASIC has been studied comprehensively in the literature. As an essential part, the storage element for preserving secret keys is always the target of adversarial attacks. Unfortunately, conventional methods for key preservation such as applying a “One-Time-Programmable” Non-Volatile Memory (OTP NVM) storage where the key is restricted for read only by authorized identities, has been experimentally demonstrated vulnerable to varieties of physical attacks. The technology of Physical Unclonable Function (PUF) emerged at the right moment, when traditional approaches failed to circumvent this issue. Compared to the OTP NVM, PUF is superior in the security strength against physical attacks owing to its inherently tamper-evident property and physical randomness that renders predicting unknown secret patterns extracted from it a difficult job.

The incentive of research in this dissertation is *to improve the security and reliability of existing PUF implementations supported by applying emerging Non-Volatile-Memory (eNVM) technologies*. Even though PUF is recognized as an efficient primitive for key preservation/generation and cryptographic protocols, it has been criticized for low reliability under certain environmental conditions and limited strength of security against intelligent attacks. The research in this dissertation will review these problems firstly, and propose several PUF designs using the eNVM media to achieve the best performance in security and reliability with minimum overhead.

The first PUF design is featured by physically reconfiguration property, that is, the key pool of the PUF can be dynamically refreshed as required. Compared to an existing instance called “logically reconfigurable PUF”, the proposed design does not need any auxiliary components for algorithmically refreshing challenges and responses of a PUF. An illustrative

design based on Phase Change Memory (PCM) by exploiting process variation and programming uncertainties of PCM cell devices have been used for demonstration. Statistics of measurement results on an experimental PCM chip proved the expected effectiveness of the proposed scheme.

The second design utilizes the non-volatility of eNVM technologies for enhancing reliability of memory-based PUFs. The core idea of this approach is the Automatic Write-Back (AWB) technique which latches the initially generated response bits by comparing the mismatch between two memory cells into a complementary state. The worst-case reliability of response regeneration using AWB can be remarkably improved in the extreme working conditions. While the proposed AWB can be generously applied for most of eNVM technologies to implement PUFs, an illustrative design based on the Spin-Torque Transfer Magnetic Random-Access Memory (STT-MRAM) was developed and simulated for demonstrating the efficacy. Simulation results on the experimentally calibrated device/circuit models proved that the reliability was increased by 10^6 times on average.

Finally, the design of an eNVM-based PUF for dual-mode applications of normal memory operation and secret key generation is proposed. Unlike the conventionally intrinsic PUFs based on SRAM or latch, the proposed eNVM based dual-mode memory-based PUF is theoretically proven to be vulnerable against a “fault-injection attack” that makes use of data remanence after re-write operations to reveal the precedingly generated response bits. Meanwhile, a generically applicable design method considering the impact of process variations on memory yield and PUF quality is proposed. Optimal point of sizing memory cell can be found so as to achieve a memory design with minimized failure probabilities in read and write operations under the reliability constraint in the PUF mode. As a case study, an STT-MRAM based PUF instance is designed using the proposed method and simulated afterwards. The results indicate the advantage of the method for designing dual-mode memory based PUFs in terms of security enhancement and area/power consumption saving.

Table of contents

List of figures	xiii
List of tables	xix
1 Introduction	1
1.1 Motivations	1
1.1.1 Physical Unclonable Function	3
1.1.2 Emerging non-volatile memory technology	6
1.2 Research Objectives	7
1.3 Contributions	9
1.3.1 eNVM based reconfigurable PUF	9
1.3.2 eNVM based PUF with enhanced reliability	10
1.3.3 eNVM based PUF for dual-mode application	10
1.4 Organization	11
2 Preliminaries and related work	13
2.1 Physical Unclonable Function	13
2.1.1 Definition	13
2.1.2 Categorization	14
2.2 Emerging Non-Volatile Memory Technology	18
2.3 Related Work	20
2.3.1 Alternative PUF types	20
2.3.2 PUF quality improvement techniques	23
2.3.3 eNVM based PUFs	25
2.4 Summary	27
3 Reconfigurable PUF	29
3.1 Fundamentals of rPUF	30
3.1.1 Definition and classification	30

3.1.2	Quality criteria	31
3.2	PCM-based rPUF Design	34
3.2.1	PCM fundamentals	34
3.2.2	PCM process variations and programming sensitivity	35
3.2.3	PCM-based rPUF design-I	37
3.2.4	PCM-based rPUF design-II	46
3.3	Summary	56
4	Quality Improvement of PUF Based on eNVM	57
4.1	Pros and Cons of MemPUF	59
4.1.1	Pros of MemPUFs	60
4.1.2	Cons of MemPUFs	61
4.2	Quality Enhancement Using eNVM Technology	62
4.2.1	Automatic Write-Back scheme	62
4.2.2	Multiple-bit capability per cell	63
4.3	Case Study: STT-MRAM Based PUF With Improved Quality	69
4.3.1	Design of STT-MRAM based PUF with Automatic Write-Back	71
4.3.2	Evaluations on quality criteria	73
4.3.3	Comparison	86
4.4	Summary	88
5	Dual-Mode Application of MemPUF	91
5.1	Reusing Memory for Dual-Mode Applications	92
5.1.1	Impact of process variations on memory operation and PUF qualities	92
5.1.2	Security concerns on MemPUF for dual-mode applications	95
5.2	Design Method for Dual-Mode BF-MemPUF	95
5.2.1	Resilience against data remanence decay attack	100
5.2.2	Design considerations	100
5.2.3	Design space exploration	104
5.3	Case Study: Dual-Mode BF-MemPUF Using STT-MRAM	108
5.3.1	Fundamentals of STT-MRAM	108
5.3.2	Simulation and discussion	109
5.3.3	Design optimization and results analysis	111
5.3.4	Dual-mode operation on other eNVM based PUFs	115
5.4	Summary	117

6	Conclusions And Future Work	119
6.1	Conclusions	119
6.2	Future Work	121
6.2.1	Dual-mode SRAM-based PUF system design and implementation .	122
6.2.2	Public PUF for FPGA IP licensing scheme	124
	Publications	127
	Bibliography	129

List of figures

1.1	An illustration of TPM architecture.	2
1.2	PUF application for authentication application.	4
2.1	Block diagrams of L-rPUF (top) and P-rPUF (bottom). The symbol “S” represents the configuration state of the rPUF.	17
2.2	Structural diagrams for eNVM technology devices: (a) PCM, (b) RRAM and (c) MTJ.	19
2.3	Identification protocol based on PPUF.	21
2.4	Adapting the ramp-up time of supply voltage to improve the reliability of SRAM PUF.	24
2.5	(Left) Aging effects to skew the threshold voltages of transistors in the SRAM-based PUF cell. (Right) Enrollment of SRAM-based PUF with accelerated aging procedure.	24
3.1	Basic characteristics of PCMs: (a) sketch of PCM cells in a row of a memory array and detail of the storage element; (b) programming pulses; (c) driving a PCM cell (voltage-driven programming approach); (d) single-bit per cell and multi-bit per cell storage.	34
3.2	Programming pulses for $Prog_s(A, t)$: (a) SCD pulses for full-SET programming; (b) uniform A and t in SP method for partial-RESET programming; (c) non-uniform A and t in SP method for partial-RESET programming. . .	38
3.3	Reconfiguration modes of rPUF: (a) hash mode; (b) TRNG mode; (c) ICR hash mode.	38
3.4	ICR schematic: (a) programming voltage generator (all PMOS transistors are nominally identical); (b) pulse generator (all delay chains, consisting of an odd number of inverters, are nominally identical); (c) PCM programming circuit. Voltage-mode programming is considered.	41
3.5	PCM read circuit. Noise model is also included.	42

3.6	Challenge selection methods: (left) horizontal; (centre) vertical; (right) random.	43
3.7	Fractional Hamming distances of measured 128-bit responses to the same challenge for 500 different configuration states.	46
3.8	Measured 1/0 occurrence proportions for 128 cells from 500 random state configurations of each cell (black="1", white="0").	47
3.9	Mean values of the fractional Hamming distances for the responses to the same challenges from 10 chips (the response of one of the chips to each challenge is used as a reference).	47
3.10	SCU scheme for multi-bit programming operation.	48
3.11	Simulation results for SCU partial-RESET programming: programmed cell resistances obtained by using a varying number of pulses with ΔV of 25 mV, 50 mV and 100 mV. Plots (a), (b) and (c) correspond to three different cases of pulse duration, i.e., 50 ns, 100 ns and 150 ns, respectively.	49
3.12	Measured cell state occurrence proportions in the case of (a) 1-bit, (b) 2-bit, and (c) 3-bit quantization for a 128-cell word.	52
3.13	Distribution of fractional Hamming distances of the responses generated by one PCM chip to the same challenge for different quantization levels (128-cell words): (a) 1-bit; (b) 2-bit; (c) 3-bit.	53
3.14	Mean values of the fractional Hamming distances for the responses generated by 10 PCM chips when using three quantization levels (a) before and (b) after being processed by the hash function.	54
4.1	Initial read-out response bit when two resistive memory cells in identical state are unstable in the presence of noise effects. Write-back operation sets one of the cells to an opposite state with respect to the other. The regenerated response bit from the cells in complementary states is more stable.	60
4.2	$RA - t_{MgO}$ (top) and $TMR - t_{MgO}$ (bottom) characteristics of MTJ. Bit alteration occurs when t_{MgO} and eOV meet certain condition. Labels A and B identify the two MTJs of an SB cell.	63
4.3	By resetting the state of each MemPUF cell (assuming the MemPUF array consists of N cells) with randomly generated $s_{(i)}$ ($1 \leq i \leq N$), bit produced from the cell may be altered.	64
4.4	Device model of MTJ with process parameters considered as normally distributed variables due to fabrication uncertainties (e.g., the contact area, $Area$, the thickness of oxide layer, t_{MgO} , and the effective oxygen vacancy, eOV).	65

4.5	The relationships (experimental data and simulation results) between MTJ resistance and applied voltage ($-0.6 \sim 0.6$ V) in P and AP states, respectively, with MTJ oxide layer deposited at (a) 180°C (when the oxide layer is free of OV and eOV is assumed to be 0) (b) room temperature (when the proportion of OV reaches its maximum value and eOV is assumed to be 1). The increased eOV results in a decrease of TMR owing to the lowered energy barrier, and also results in an increase of MTJ resistance because the non-coherent tunneling induced by OV is less conductive than the coherent ones.	66
4.6	RSD of TMR against RSD of eOV . The nominal value of eOV is assumed to be 0.03 to match the experimental results in [142]. A proper value of eOV RSD, i.e., 0.24%, can be found by sweeping it within the possible range. . .	67
4.7	(a) Fundamental components of the STT-MRAM circuitry: SB cell (2T2R), sense amplifier, bit-lines and word-lines; (b) memory organization; (c) system-level view of the secret key generator composed of the internal STT-MRAM and the fuzzy extractor. R, H, and K are the response, the helper data and the generated cryptographic key, respectively.	68
4.8	Reliability model of MemPUF. The reliability may be deteriorated by noise effects in STT-MRAM, i.e., $b_{t+\delta t} \neq b_t$	69
4.9	(a) (Top) Temporal distribution of the initial angle of an MTJ in P state from 1000 samples. Temperature is 300 K. (Bottom) The corresponding R_{MTJ} . (b) BERs at varying temperature points (from 200 K to 400 K) were calculated as the proportion of SB cells in the 1000 samples that produce unreliable bits in consecutive 10 read-out operations. The MTJs of each SB cell are initialized into P and AP states, respectively. Nominal values and variations (shown in brackets) of the relevant process parameters used in the simulation are shown in the text-box.	70
4.10	(a) Schematic and state-diagram of the AWB scheme. (b) and (c) show the timing diagram of reading a bit from an SB cell in the enrollment phase. (b) One of the MTJs in the SB cell is switched during the write-back while (c) another MTJ is kept unchanged according to the bit produced from the cell initially.	78
4.11	Failure models of (a) case #2 and (b) case #3. In case #2 and case #3, certain failure effects (framed by red dashed line) cause the bit regenerated at $t + \delta t$ to be different from that generated previously at t	79
4.12	BERs in the three cases with varying width of access transistor.	79

4.13	BERs for the three cases (initial states of both P and AP were considered for case #2 and case #3) at (a) varying supply voltage (fluctuating within a span of 10%) at 300 K and (b) different ambient temperatures (ranging from 200 K to 400 K) with supply voltage fixed at 1 V. The inner plots show the BERs for case #2 and case #3.	80
4.14	Distribution histograms of the calculated fractional Hamming distances for initial MTJ states of (a) P ($s = 0$) and (b) AP ($s = 1$).	81
4.15	Estimated compression ratio, i.e., upper bound of the entropy, of 50 64-bit strings generated with SB cells initialized in P and AP states, respectively. .	81
4.16	(a) The bit-maps of a 32×32 array for initial P (left) and AP (right) states, respectively. Black boxes represent “1”, whereas white boxes represent “0”. (b) Overall 117 (filled with black color) of the 1024 cells are capable of generating alterable bits.	82
4.17	(a) Complementary probability of “one-bit flip” in the generated bit strings using the same cells but different initial states, i.e., $1 - \Pr(\text{num}(b_{s=0} \oplus b_{s=1} = 1) \geq 1)$. The length of the strings was varied from 64 bits to 1024 bits. (b) Percentage of bit patterns ($(b_{s=0}, b_{s=1})$) generated from an SB cell with different initial states before and after using hash function for post-processing.	83
4.18	(a) 1-finger and (b) 2-finger configurations of STT-MRAM cell layout. W and λ refer to the width of the access transistor and the parameter of design rules, respectively.	86
5.1	Working mechanism of a conventional memory-based PUF system (illustrated with an SRAM-based PUF). C , R , H , and K are the challenge (normally the memory address), response, helper data and output key of the system. The figure at the bottom shows what happens inside a SRAM cell when it is switched from memory mode to PUF mode (write back operation after response bit generation is not shown).	96
5.2	Procedure of performing a fault-injection attack on SRAM based PUF utilizing data remanence of SRAM cell. t_i refers to the decay time, and Δt is the incremental value for t_i after each iteration. Initially, the SRAM is given a known state M_{i_0} . The attack tries to recover the intrinsic state of SRAM based PUF (in Phase-II) with the collected responses $R_1, R_2, \dots, R_f$ iteratively generated from the PUF with tuned decay time (in Phase-II). A more detailed description of the attack can be found in [94].	97

5.3	(a) Distribution of read-out current for an eNVM cell in HRS and LRS states, respectively. Setting reference currents to different values may result in different output bits from the cell; (b) working mechanism of BF-MemPUF.	98
5.4	Distributions of sensed current on a LRS eNVM cell (a) before and (b) after the variances of randomly distributed variables have been increased.	103
5.5	Adding redundant reference arrays average out the effect of skewness introduced by the reference cells which are shared by each row of data cells. . .	104
5.6	Algorithm to design an optimal BF-MemPUF.	105
5.7	(a) Circuitry of STT-MRAM array and cross-sectional diagram of STT-MRAM cell; (b) R-V characteristics of MTJ device, and resistivity switching dynamics between AP and P states of an MTJ.	107
5.8	Increasing the oxide layer thickness results in a prolonged lifetime of MTJ device with the same biasing condition. In this configuration, the thickness of 1.15 nm gives the MTJ device a life time of around $7 \sim 8$ years.	109
5.9	Memory array architecture (left-top), working mechanism (left-bottom) and read circuit diagram (right) of the BF-MemPUF design based on STT-MRAM technology.	111
5.10	(a) Resistance distribution of 1000 MTJ samples in AP and P states, respectively. Voltage biasing between the top and bottom nodes of the MTJ is 100 mV. (b) Resistance distribution of 50 MTJ samples, where the output state tendencies for the same MTJ sample in two different resistance states are correlated.	112
5.11	(a) Calculated Φ_{f-RD} and Φ_{f-WF} with varying W ; (b) Estimated BER in 2000 samples under different temperature points.	113
5.12	(a) Decreasing the scaling factor α will result in a deterioration of randomness owing to the correlation effects created by sharing reference cells. The value of α is annotated in blue. (b) Trend of σV_{th} with scaled effective channel length of a NMOS transistor [95].	114
5.13	Calculated (a) Φ_u and (b) Φ_e using different architecture arrangement strategies for the STT-MRAM array.	115
6.1	(a) Architecture of SRAM system for dual-mode application. (b) Mechanism of SRAM based PUF in two working modes.	122
6.2	The impact of variance of threshold voltage on the probability of failure effect in memory-mode operations and probability of generating unreliable bits in PUF-mode.	123

List of tables

3.1	Randomly distributed process parameters of 180 nm PCM technology with NMOS selector	36
3.2	Commonly used ECCs and their code parameters	40
3.3	Mean values of the fractional Hamming distances for the measured responses to the same challenge and corresponding reliability using different ECCs at different ambient temperatures	42
3.4	Mean values and coefficients of variation of fractional Hamming distances for the measured responses to different challenges before and (after) hash function - different wordlengths and CRP generation methods were analyzed	46
3.5	Mean values of Hamming distance of responses to the same challenge at different temperatures for different quantizations.	49
3.6	Mean values and coefficients of variation of the fractional Hamming distances for the measured responses to the same challenge before (after) hash function using the augmented scheme - different response word lengths, quantization levels, and CRP generation methods were analyzed	50
4.1	Properties of eNVM candidates for MemPUF with AWB scheme	71
4.2	Parameters used in the simulation	74
4.3	Secret bits generated from one SB cell with different initial states.	77
4.4	Comparisons of the proposed STT-MRAM (with and without AWB scheme) with conventional MemPUFs in terms of Hamming distance, BER, and equivalent entropy	84
4.5	Comparison of the proposed STT-MRAM PUFs with the conventional designs in terms of <i>energy/bit</i> and <i>area/bit</i>	88
5.1	SD structure and process variation sources of STT-MRAM, PCM, and RRAM	99
5.2	Parameters considered for designing BF-MemPUF	101
5.3	Parameters used in STT-MRAM based BF-MemPUF	110

5.4	Optimization Results	116
5.5	Comparison of eNVM based BF-MemPUF with conventional memory based PUFs	116

Chapter 1

Introduction

1.1 Motivations

The present information age witnesses a worldwide proliferation of mobile devices. What follows closely are inevitable security/privacy issues that call for immediate research attention. There is an increasing need to providing a sanctuary for storing secret information, for instance, the private key which is required for safeguarding sensitive data computation and storage in mobile computing platform. In fact, numerous efforts have been made to address this problem by both industry and academia. Conventional software-based strategies such as anti-virus or anti-spy software and applications with key management functions have been developed to detect intrusion or malicious malware that stealthily breach crucial information in the device. In addition, virtualization techniques help with the isolation of sensitive data computation from general data transmission within an environment where all execution requests are authorized and attested.

However, more and more academic reports have experimentally demonstrated the incompetence of software-based approaches to providing security and privacy for contemporary computing systems, which are almost incessantly connected to the omnipresent internet access [27, 30, 99, 100]. An experienced hacker can exploit the weakness of communication protocol or directly dump out the content of the main memory where a piece of secret key is contained via Direct Memory Access (DMA) method. To this end, effectual solutions from hardware perspective have been proposed as viable candidates to tackle the problems. The advantages of hardware-based methods as compared to software-based approaches to ensure the security of a mobile computing system include:

1. It is difficult, if it is not possible, to hack into a piece of hardware via physical means to eavesdrop sensitive data. This is because under normal circumstances, the hardware

Fig. 1.1 An illustration of TPM architecture.

will be intentionally and thoughtfully islanded if it is intended to prohibit users from physical access, and to obtain the useful information for an adversarial attack, one needs to dismantle the device and physically probe the internal delicate elementary components, which will either be tremendously time-consuming and expensive or ultimately lead to application failure owing to ruinous activity. Therefore, it is a safer option to use hardware module to safeguard confidential information such as secret key.

2. Integrating a hardware-oriented security/cryptographic primitive into the system is beneficial to the overall performance as well as the computational overhead. On one hand, one may expect the hardware-accelerated algorithms that execute authorized data computation to be faster compared to the software counterpart because the hardware modules are often dedicatedly designed and optimized to perform specific cryptographic/security tasks efficiently. On the other hand, the hard-wired secret key preservation module regenerates and updates (when necessary) the critical information more efficiently and securely than software-based approaches that preserve and generate secret keys virtually.

Hardware security has been studied extensively in the past decades. In 2009, the Trusted Computing Group proposed a generic model called the Trusted Platform Module (TPM) (Fig. 1.1) for secured computing in modern computing platforms [127]. Conceptually, it is a standard for cryptographic microprocessor chip that can securely store the artifacts which are applied to authenticate the computer platform where the TPM chip resides as well as the executions operated on the platform. The artifacts may include the password, certificates and/or encryption keys. GlobalPlatforms developed the Trusted Execution Environment (TEE), which is the result of a standardization effort of different hardware implementations (e.g., ARM TrustZone, AMD Platform Security Processor, etc.) originally developed by

individual vendors to achieve secured computing and data storage purposes [1, 4, 40]. It sets up an isolated region for the computation of sensitive data so that the data loaded into this region will be protected to ensure confidentiality and integrity. A closer look at both TPM and TEE suggests that both methodologies operate based on a common practice in which *a trustworthy hardware module that can be used to permanently and securely hold the secret key in a storage device to provide the Root-of-Trust (RoT) is required*. Normally, the RoT is instantiated using a piece of non-volatile memory such as Electrically Erasable Programmable Read-Only Memory (EEPROM) where the secret key is permanently burnt into. Architectural support in security standards such as TPM restricts access to reading the NVM unless it is authorized. In addition, the EEPROM is made unwritable so that any malicious attempt to distort or tamper the content is prohibited. While the adoption of hardware RoT seemed to adequately resolve the problem that have long been puzzling software security engineers, until recently, the ever unswerving belief in safeguarding secret key using such approach has been proven to be a wishful thinking without tangible efficacy. Experimental results have revealed the vulnerabilities of using EEPROM to store secret key, by simply utilizing more advanced probing techniques or dismantling the storage device to measure side-channel signals, which is in fact an easy task without incurring much cost or time investment [2, 3, 120]. These findings propelled a renewed search into other hardware alternatives to protect the secret information in computing platform.

1.1.1 Physical Unclonable Function

One of the most representative hardware solutions to ensure reliable management of sensitive data is to take advantage of the physical characteristics of hardware components to directly generate device-specific keys for security applications. The works proposed by Bauder and Simmons in 1983 and 1984, respectively, were widely recognized as the pioneering endeavor to exploit physical properties of disordered hardware for authentication purposes [7, 117]. In 1992, Naccache and Fremanteau applied hardware-oriented authentication scheme on memory card [91]. The notion of Physical Unclonable Function (PUF) appeared in the literature in 2002, when the MIT researchers Gassend *et al.* introduced the electronic device based design where process variation of circuit components were leveraged on for authentication application [38]. Research work in this area has since mushroomed following these forerunners to exploit hardware-intrinsic cryptographic schemes to ensure increased level of protection of secret keys. When the scale of semiconductor fabrication process went down into the nanometer regime, it became possible to leverage on the previously undesirable process variations in Integrated Circuit (IC) components to design a PUF [42, 47, 69, 72, 77, 123, 124]. The benefit of this approach is that the PUF primitive can

Fig. 1.2 PUF application for authentication application.

be efficiently integrated into computer systems, where secret keys safely reside within the system without being exposed to the outside world. Based on the database of Microsoft Academic Search, since 2003, approximately 300 articles with key word “Physical Unclonable Function” have been published. The number grows exponentially along with the ramping interest of people from different disciplines who have common concern on security issues.

PUF is believed to be a promising candidate to efficiently tackle the safety issues that long persecuted the EEPROM based key storage component in computers. Functionally, PUF is a physical device that probabilistically maps the input *challenge* (commonly denoted as C) to the output *response* (denoted as R) according to its innate stochastic properties resulting from the uncontrollable manufacturing variations. The challenge C is often an m -bit binary string that belongs the challenge set $\mathcal{C} = \{0, 1\}^m$ and the response is an n -bit binary string that belongs to the response set $\mathcal{R} = \{0, 1\}^n$. Using binary string to represent challenge and response makes the interface between PUF and the contemporary digital computer system more efficient. Since the Challenge-Response Pairs (CRPs) of a PUF are random, device-specific and unique, they can be extracted and post-processed for cryptographic applications such as device identification, IP protection and authentication. A typical authentication scheme using PUF between two parties, A and B , can be conducted as follows:

1. Initially, A physically holds the PUF and measures a set of k CRPs, $\{C_i, R_i\}_{1 \leq i \leq k}$ which are then stored into a list $\mathcal{L}$ as references.
2. A issues the PUF device to B who later uses it for credential communication between the two parties.
3. When B wants to use the PUF for an authenticated execution, he firstly sends a request to A and then receives a challenge C which is randomly selected from $\mathcal{L}$ by A .

4. B stimulates the PUF devices using C to generate the corresponding response R and then sends it to A .
5. A receives R and authenticates B if the CRP matches one in the list $\mathcal{L}$.

A graphical illustration of the process is shown in Fig. 1.2.

The advantages of using PUF as a cryptographic primitive as compared to conventional methodologies are summarized as follows:

- *PUF is not only tamper resilient but also tamper evident as any kind of physical attack that intends to break the device so as to retrieve sensitive information achieves nothing but damage the PUF chip.* This is the primary distinctive feature of a PUF in comparison to an EEPROM based key storage hardware. While it is easy, or cost-efficient, to physically attack an EEPROM where the secret key is stored, it may be difficult or impossible to invasively intrude a PUF chip without inducing any destructive effect on it. A PUF can be visualized as a black-box where the secret key preserved within it is trustworthy such that the key chain built upon it is also secure and reliable under the threat of physical attacks [109].
- *The challenge-response characteristics of a PUF are unique, unclonable and reliable.* Thanks to the intrinsic hardware disordered properties which originate from the ineluctable variations during fabrication process, the extractable physical signatures cannot be physically or mathematically cloned (this will be extensively discussed in the later chapters). In addition, it is qualified as a “good” PUF as response-string-derived secret keys can be reproduced reliably using the same challenges even under varying operating conditions. Similar to biometric-based authentication such as using iris, fingerprint and face, a good PUF based authentication system is featured by low false-acceptance rate and false-rejection rate.
- *Silicon PUFs can be fabricated using conventional CMOS technology and can be compatibly integrated into the modern computing system.* While the first PUF instances were prototyped using specific material with stochastic distributive properties when subjected to certain external stimulus for generating responses, modern PUFs are implemented using CMOS technology as chips such that they can be easily embedded into computer systems as a standalone key generator or integrated cryptographic primitive.
- *Emerging PUF designs can be applied to more advanced scenarios or protocols.* The most recent PUF designs not only inherit the tamper-evident attribute from its predecessors, but also endowed with novel appealing features. Examples include the

reconfigurable feature made possible by a dynamically refreshable key pool [67, 143], public PUF which is considered an analogy to asymmetric public cryptography [8, 85, 102, 111] and PUFs based on emerging material or technology that have enhanced qualities [104, 106].

1.1.2 Emerging non-volatile memory technology

There are several reported approaches in the implementation of silicon PUF. Most of the existing approaches are based on the CMOS transistors where either the delay or the mismatch characteristics of the devices are exploited to produce PUF responses. However, recent advancements in this research area have demonstrated some vulnerabilities in these designs. For example, modelling attack associated with machine learning techniques were developed to firstly model the physical characteristics of a strong PUF instance whose challenge-response pairs are correlated, and then predict the future response generated from the PUF with new challenges [109, 114]. Semi-invasive attacks have been proven to be possible on SRAM based PUF, i.e. by taking advantage of the data remanence effect in SRAM devices to reproduce response bits generated from the same chip [94].

Emerging Non-Volatile Memory (eNVM) technologies such as Phase Change Memory (PCM), Spin Transfer Torque Magnetic Random Access Memory (STT-MRAM) and Resistive RAM (RRAM) were recognized as promising candidatures for next-generation memory systems owing to several appealing properties.

- *Non-volatility* - As the name implies, eNVM is by nature non-volatile, which means that it will not lose its data bits preserved in the memory cells when the power is off. This is due to the data storage mechanism in eNVMs where a data bit is stored in the form of resistivity in an eNVM cell, i.e., bit 1 when the resistivity is high and bit 0 when the resistivity is low (or sometimes vice versa). The benefit of non-volatility is that there is no leakage power when the eNVM chip is at stand-by mode.
- *High density and scalability* - Most of the representative eNVM technologies are fabricated by forming different layers on a silicon wafer, which is compatible with the conventional CMOS fabrication technology. When compared to CMOS-based memory technologies with similar read/write speed and random accessibility (e.g., SRAM and DRAM), eNVM has much smaller area consumption for each memory cell. This in turn provides a higher density in eNVM memory array than other types of memory. In addition, most eNVM technologies such as PCRAM and STT-MRAM have experimentally been proven to have high scalability (below 5 nm [135, 137]).

- *Multi-bit capacity* - Some of the eNVM technologies have the capacity to preserve more than one bit in a single memory cell. By exploiting the continuity of resistivity of memory cell storage element, the read-out current sensed from a memory cell can be quantized using an analog-to-digital converter so as to produce multi-bit data output. In writing a cell, a programming pulse which is regulated by a fine-grained controlled signal is used to set/reset the cell state to an extreme or intermediate state. Multi-bit capacity increases the information capacity of an eNVM given a fixed array size.
- *Universal adaptability* - The read/write latencies of eNVMS are comparable to CMOS based counterparts. Coupled with its non-volatility feature, it is considered as a promising candidate for implementing a *universal memory* that flattens the memory hierarchy (formation of a universal memory sub-system) in the traditional computer architecture, thus favoring low latency of data transmission between different layers as well as low power consumption of preserving data [92, 135].

While high-density and scalability are appealing features of eNVM technology, similar to the CMOS based counterparts, eNVM storage elements (e.g., $\text{Ge}_2\text{Sb}_2\text{Te}_5$ for PCM) that are normally deposited as multiple layers on a minimally sized contact will suffer from process uncertainties. This causes the actual characteristics of eNVM devices to deviate from the expectations during design time. A direct consequence of the process variation effect is a yield loss of the eNVM chips, since normal read/write operations on certain memory cells with an outlier physical parameters may fail permanently. Nevertheless, these random and unpredictable properties can be leveraged for extracting device-specific fingerprints that can be used for alternative applications.

Although eNVMS have been extensively exploited in data storage and computation fields, they are relatively less explored for developing security primitives such as PUFs. Process variation of eNVM chips which are fabricated with miniaturized technology can be fully exploited to generate digitized device-specific response bits. Furthermore, some of the unique characteristics of eNVM materials such as non-volatility and multi-bit capacity make them suitable for developing PUF instances with enhanced security, multi-mode operation and higher power-area efficiency [67, 104, 106, 143–145].

1.2 Research Objectives

The research work in this thesis was philosophically inspired by prior research works published in [38, 47, 63], where the viability of using process variability of memory device for designing qualified PUFs were proposed. In general, it is to *design novel eNVM based*

PUFs with enhanced security and efficiency and augmented reliability while ensuring good compatibility with conventional computing platform. More specifically, the research work will be conducted in the following aspects in correspondence to the research bottlenecks that have not been satisfactorily resolved .

1. *How can we enhance the security level of PUF primitives using alternative implementation material and/or design method?* - Given that the conventional PUF instances are vulnerable to either semi-invasive physical attacks or modelling attacks, PUF designs with enhanced security using alternative material are therefore desired. In this research direction, an eNVM based PUF with improved security based on physically refreshable challenge-response mapping characteristics to withstand cutting-edge physical and modeling attacks will be introduced. For the purpose of generalizing a theoretical approach to designing the reconfigurable PUF, formal conceptualization will be given to lay a foundation for understanding and evaluating the new PUF. Suitable material using eNVM technologies to implement the PUF following system-level security specifications will then be searched and identified. The proposed circuit-level design of the reconfigurable type of PUF will be physically implemented and experimentally tested.
2. *How can we improve the quality of a memory based PUF primitive without relying on costly architectural method or circuit-level modification?* - As PUF reliability is always one of the most vital concerns in PUF design and applications, the second direction of the research is to exploit the distinct characteristics of eNVM material to design PUF with augmented reliability. While most of the existing approaches circumvent the reliability issue with architectural methods such as error correction code or circuit-level techniques that involve heavy modifications on original circuitries, my research objective in this realm is to leverage on cost-efficient circuit-level method with marginal overhead in terms of area and power consumption to achieve a remarkable improvement of PUF quality compared to the state-of-the-arts. In addition, the research work will also generalize the design methodology so as to make it generically applicable in most contemporary technologies. An illustrative design will be presented to verify the hypothesis and comparatively evaluated to prove the effectiveness of the proposed method.
3. *How can we design a re-usable memory-based PUF with high qualities in both memory-mode and PUF-mode using generically applicable optimization approach?* - The design methodologies proposed in the literature mostly focus on improving the quality of PUF that can be intrinsically reused as another functional module in computer system.

Memory based PUF is one representative example in this category. This is particularly desirable for the purpose of minimizing the overhead of a dedicated chip area for a standalone PUF module or duplicating PUF chips (or arrays) for different application purposes, as reusing the same PUF is beneficial in terms of area cost. However, designing an IC with good quality of dual-mode requires optimal method that has not been well studied before. The incentive in the third research direction is therefore to find an optimization methodology in designing memory-based PUF that can be reused for dual-mode application while both applications meet the high quality specifications. Furthermore, eNVM based dual-mode PUF will also be proven to be secure against fault-injection attack that has been successfully mounted on conventional types of memory-based PUFs. Probabilistic method will be applied to evaluate a spin-transfer torque magnetic memory based PUF design using experimentally calibrated device model.

1.3 Contributions

Corresponding to the aforementioned objectives, the research work in this dissertation is conducted in three directions: 1) *eNVM based reconfigurable PUF*; 2) *eNVM based PUF with enhanced reliability*; 3) *eNVM based PUF for data storage and security dual-mode applications*.

1.3.1 eNVM based reconfigurable PUF

Reconfigurable PUF (rPUF) was introduced for the first time in [63]. However, various definitions of rPUF can be found in the literature but they are not totally agreeable. The research work in this part firstly generalizes the definition of rPUF - based on the reconfiguration methods, the existing rPUFs are dichotomized into two groups of Logically-rPUF (L-rPUF) and Physically-rPUF (P-rPUF). Besides, rPUF quality criteria that can be formally applied to evaluate an rPUF in terms of unpredictability, reliability and unclonability are formulated. Following that, based on the imperative research work conducted on rPUF definition and quality criteria, two rPUF schemes based on PCM are introduced. These rPUF schemes leverage the collective attributes of process parameters' variability and programming pulse uncertainties of PCM. Compared to the L-rPUFs proposed in [29, 54], the P-rPUFs proposed in this thesis are superior in terms of security against physical attacks. In addition to this, the response refreshment capability does not rely on auxiliary algorithmic blocks that may induce non-negligible hardware overhead.

The first proposed PCM rPUF scheme exploits the measurable differences of PCM resistances programmed by electrical pulses with randomly generated amplitudes and durations for CRP generation and reconfiguration. To maintain the anonymity of the configuration state, an Imprecisely Controlled Regulator (ICR) is proposed. Subsequently, the second PCM rPUF scheme is proposed. This is an augmented version, which relies on the number of varying programming pulses to converge a PCM cell to the target resistance value for responses extraction. Since this scheme reduces the accessible CRPs between two consecutive reconfigurations to only one, it can be safely utilized in broader cryptographic protocols. Both designs were experimentally tested by a PCM chip fabricated using 180 nm μ -trench technology.

1.3.2 eNVM based PUF with enhanced reliability

Memory-based PUF (MemPUF) has emerged as an inexpensive yet highly secure way of generating and storing the cryptographic key in memory for operations such as authentication of devices, protection and communication of confidential information. Although many MemPUFs have been proposed, most of them suffer from two key problems: 1) *the reliability of response-bit generation is poor*, and 2) *only one bit can be generated from each cell*. Techniques reported in the literature either relies on heavy architectural method such as error correction code that requires high-overhead implementation of digital modules, or circuit-level modifications, which are not generally suitable to all types of MemPUFs.

We proposed a novel MemPUF primitive based on eNVM technology. The uniqueness of the new design is that it takes advantage of the intrinsic device characteristics which are natively exploitable via normal memory operations, and avoids the use of techniques such as accelerated aging, ramping time adaption, and external effects that alter device properties for regular function. It has eliminated the need for both a pseudo random number generator and a protected non-volatile memory for the generation, storage and retrieval of secret keys in the algorithmic description of cryptographic primitives.

1.3.3 eNVM based PUF for dual-mode application

Using conventional memory circuitry (e.g., SRAM) for key generation purpose has several drawbacks. One of the critical issues is that it may help adversaries to exploit data remanence effect on SRAM cells to perform fault injection attack so as to obtain the secret bits preserved in the memory based PUF. Another problem is that, since process variations of memory devices act as a double-edged sword to produce reliable and random response bits for use as PUF and increase the yield loss when the devices are used as ordinary storage

elements, a novel design method is needed to find an optimal point to meet the requirements from both sides.

To overcome the aforementioned issues faced by conventional memory-based PUFs, a novel security primitive called *Buffer-Free Memory-based PUF* (BF-MemPUF) using eNVM technologies (i.e., PCM, STT-MRAM and RRAM, etc.) was proposed. Compared to other types of memory-based PUFs, the benefit of using eNVM is the area and energy consumption for both memory and PUF applications can be economized. Both the circuit- and architecture-level designs of a BF-MemPUF were presented. A general optimization method that maximizes the memory yield while maintaining an acceptable rate for PUF qualities were proposed and verified. An extensive analysis in comparison with the existing method proved the advantage of the proposed method.

1.4 Organization

The remaining content of this dissertation is organized as follows,

Chapter 2 presents a brief study on the related work and background. In this chapter, PUF technology will firstly be reviewed in terms of its basic functionality, categorization, and metrics for quality assessment. This is followed by an overview of the current research status of PUF technology, with an emphasis on the security vulnerabilities when subjected to advanced attack scenarios such as modelling attack and fault injection attack, as well as the reliability issues under varying conditions and limited entropy for PUF with limited challenge space. In the later part of Chapter 2, eNVM technology is briefly reviewed from the aspects of fundamental mechanisms, process variability and potential for designing PUF for enhanced qualities.

Chapter 3 is devoted to the research on PCM based reconfigurable PUF design. Firstly, a formal definition and dichotomy of rPUF are presented. The criteria such as reliability, unpredictability, unclonability and unbreakability for assessing rPUF quality are also introduced. Two rPUF designs that exploit the process variability and programming uncertainty of PCM devices are then presented. While the first design makes use of resistance variability of PCM cells after imprecisely regulated programming process to produce random and refreshable response bits, the second design, which is more resilient against prediction attack owing to the limited access to response bits between consecutive reconfigurations, uses programming dynamics of PCM devices for the same response generation and reconfiguration functionality. Both of the two designs were experimentally evaluated with a pre-fabricated PCM chip. The measurement results on the designed PCM based rPUF are comparatively studied and analyzed.

Chapter 4 introduces the design of a highly reliable MemPUF with multiple bit capacity based on eNVM technologies. The discussion in this chapter begins with a brief review of the reliability deterioration effects in a majority of MemPUF instances. Then an automatic write-back scheme which can be easily incorporated in the modern eNVM circuitries without heavy modifications is introduced. The efficacy of using the automatic write-back scheme for reliability enhancement is then analyzed on an illustrative medium, STT-MRAM, in a probabilistic approach based on an established device model of magnetic tunnel junction considering process variability and temperature dependency. A comprehensive analysis on the quality of the proposed design is then shown to demonstrate its advantage over the conventional designs.

Chapter 5 serves to report the design and analysis on dual-mode memory based PUF using eNVM technologies. In this chapter, the fault injection attack on the traditional SRAM based PUFs leveraging on data remanence effect will be reviewed. As a countermeasure, the idea of using eNVM for dual-mode PUF application with data store capability is proposed and analyzed. In the second part of Chapter 5, the optimization method to design a memory-based PUF with maximized yield under the constraints of PUF quality and area/power efficiency is presented. The method is validated by using an illustrative design on STT-MRAM. Subsequently, simulation results on the proposed design and the conventional approaches are compared and discussed.

Chapter 6 draws a conclusion of this dissertation by summarizing the results from simulative analysis and experimental testing. Future works worthy of further investigation are outlined.

Chapter 2

Preliminaries and related work

2.1 Physical Unclonable Function

The term Physical Unclonable Function was initially introduced by Pappu in 2002 as physical one-way functions in [98], and this idea was further developed in [39] by utilizing process variabilities resulting from fabrication of silicon devices to generate device-specific signatures. The following years witnessed a growing number of novel PUF designs such as arbiter PUF and SRAM PUF with advanced functionalities and optimized qualities [46, 67, 69, 79, 125, 143]. However, even though PUF has evolved tremendously since it came into being, the fundamental challenge-response-mapping mechanism and security attributes are mostly retained.

2.1.1 Definition

PUF can be formally defined as follows.

Definition 1 *Physical Unclonable Function is a physical system that probabilistically maps a stimulating challenge c selected from a finite space $\mathcal{C} = \{0, 1\}^k$ to the output response r from space $\mathcal{R} = \{0, 1\}^l$ according to its intrinsic stochastic properties $\mathcal{P}$ due to the uncertainties of manufacturing process,*

$$\text{PUF}_{\mathcal{P}} : \mathcal{C} \rightarrow \mathcal{R} \quad (2.1)$$

The operation of generating a response $r \in \mathcal{R}$ given a challenge $c \in \mathcal{C}$ is expressed as

$$r \leftarrow \text{PUF}_{\alpha_p}(c) \quad (2.2)$$

where $\alpha_p \in \mathcal{P}$ represents the disordered physical characteristics of the PUF instance PUF_{α_p} .

Normally a fuzzy extractor $\text{Ext}()$ is required in a PUF device to stabilize the noisy responses as well as to increase the entropy. The integration of the PUF device and the fuzzy extractor is called the PUF system, or in short PUF, which is defined as follows.

Definition 2 A PUF system consists of minimally a PUF device and a fuzzy extractor $\text{Ext}()$ such that,

$$\text{PUF}_{\mathcal{P},\mathcal{E}} : \mathcal{C} \rightarrow \mathcal{R} \times \mathcal{H} \quad (2.3)$$

where the $\mathcal{H}$ is the helper data .

The function of the extractor can be expressed as,

$$r \leftarrow \text{Ext}_{\alpha_e}(\text{PUF}_{\alpha_p}(c) + e, h) \quad (2.4)$$

where r is the output response, e is the error bit string induced by noise, $h \in \mathcal{H}$ is the helper data used by an Error Correction Code (ECC) to correct the noisy responses to the challenge c followed by a universal hash function to enhance the output response entropy. $\alpha_e \in \mathcal{E}$ is an attribute of ECC and hash function used in $\text{Ext}()$.

A PUF is considered qualified if it meets the following criteria:

- *Reliability* A PUF is reliable if its responses to the same challenge produced over time under varying operating conditions are identical.
- *Unpredictability* The response bits generated from a PUF should be unpredictable so that any adversary equipped with advanced computation capability and physical attack instruments is unable to predict responses to arbitrary challenges given a set of Challenge-Response Pairs (CRPs).
- *Unclonability* A PUF cannot be physically or mathematically cloned.
- *Physical Unbreakability* A PUF is tamper evident such that any physical attempts to probe, measure or maliciously using the device will achieve nothing but a chip malfunction or permanent damage.

2.1.2 Categorization

It is hard to find a generic categorization that meets a common set of specifications. Depending on different implementation styles, working mechanism and alternative attribute, PUFs can be classified into different families according to different criteria.

Delay-based PUF and memory-based PUF

Delay-based PUFs generate the CRPs by racing replicated delay chains of identically designed components to generate random response bits as a consequence of uncontrollable fabrication process variability. An example of delay-based PUFs is the arbiter PUF [69]. Other instantiations of delay-based PUFs include the Ring Oscillator PUF [124], the lightweight parallel PUFs structure [79] and the XOR PUF [125]. The advantages of the delay-based PUF are that they have larger CRP space which is exponential to the amount of hardware elements. This cripples any mathematical attack attempts infeasible in polynomial time. However, since CRPs of the PUF are correlated, machine learning techniques can be employed to efficiently break the PUF device. By training a properly selected machine algorithm with challenge-response pairs collected from a PUF, unknown responses generated from the PUF can be predicted [109].

Memory-based PUF exploits the metastable states of inverting structures (e.g. two cross-coupled inverters) to generate random bits determined by mismatched properties of the cross-coupling devices. Multiple variants of memory PUFs have been developed in the past few years, such as the latch PUF [123], the butterfly PUF [61], the SRAM PUF [47], D flip-flop PUF [130] and buskeeper PUF [118]. The mechanism of a particular memory based PUF may be unique but generally all memory-based PUFs are architected in a similar way as an array and random bits are produced individually from each cell of the array. The advantages of memory-based PUFs over delay-based PUFs are low complexity of design, low cost of area consumption and less mutual information between individual response bits. The disadvantage of memory-based PUFs is that it has limited CRPs hence rendering its challenge-response space to be entirely explored with trivial efforts [109]. This limitation causes memory-based PUFs to be inapplicable to certain advanced protocols that require uncompromised level of security.

Strong PUF and weak PUF

PUF can also be dichotomized into *weak PUF* and *strong PUF*.

Weak PUF has limited CRPs that can be generated from the entire PUF instance [109]. More specifically, a weak PUF can be characterized by the following aspects:

- The CRPs of a weak PUF with finite physical size can be exhausted within polynomial-time, because the CRPs are dependent merely on the physical disordered properties of individual PUF devices that have no (or negligible) correlations with each other.
- The major application of a weak PUF is to derive a secret key for security purposes. For the sake of safeguarding the key derivation process of a weak PUF, a fuzzy extrac-

tor module which is hard-wired with the internal weak PUF hardware component is normally required. The purpose of incorporating such module is to stabilize the noisy responses and increase the entropy of the generated key.

Representative weak PUFs include memory-based PUFs such as SRAM PUF and butterfly PUF as introduced in the previous sub-section. Coating PUF and resistive PUF as proposed in [67, 119, 143] can also be categorized as weak PUFs.

Strong PUF, on the other hand, is highly secure in terms of its resilience against an exhaustive search on the entire challenge space, owing to both the physical obfuscated structure and the complex internal signals. The prevailing features of strong PUFs are as follows.

- The number of possible CRPs of a strong PUF is huge, therefore evaluating all of them needs a period of time which is exponentially correlated to the size of the strong PUF device.
- Unlike a weak PUF, a strong PUF does not strictly require a post-processing or an encapsulating module that block the direct interface between the internally generated response bit string and the finalized secret key. This attribute makes strong PUFs suitable for use in more application scenarios and protocols as opposed to weak PUFs. Advanced protocols such as authentication, bit-commitment, zero knowledge proof and oblivious transfer have been proven possible on strong PUF based approaches.

Reconfigurable PUF

Another criterion for classification is based on the reconfigurability of a PUF. Reconfigurable PUF (rPUF) is a type of PUF whose challenge-response mapping characteristics can be dynamically and randomly altered according to the demand. The original conception of rPUF was proposed by Kursawe [63]. Le *et al.* [67] categorized rPUF into two groups, namely *Logically rPUF* (L-rPUF) and *Physically rPUF* (P-rPUF). As its name implies, the L-rPUF relies on logical interfaces (often called *control logic*) to obfuscate the challenge-response pairs of its internal PUF according to the configuration state. P-rPUF exploits the intrinsically alterable properties of the PUF device itself to realize the reconfiguration. This physical property alteration is triggered by stimulating the PUF with an external effect. The operating mechanisms of L-rPUF and P-rPUF are shown in Fig. 2.1.

An rPUF has the following advantages over the conventional PUFs particularly in security-critical systems and highly information-sensitive applications.

- Its challenge-response behavior can be dynamically updated.

Fig. 2.1 Block diagrams of L-rPUF (top) and P-rPUF (bottom). The symbol “S” represents the configuration state of the rPUF.

- The space of possible responses of PUF is significantly expanded.

In [63], rPUF was proposed for use in the protection of persistent storage of security module. Owing to the reconfigurability of rPUF derived keys, malicious updates of the keys such as invasive setting of the internal states of the storage, can be detected. In [67], the advantage of rPUF is discussed for its enhanced resilience against prediction attack. The authors of [108] introduced a key-exchange scheme based on the Erasable PUF, which can be regarded as a kind of P-rPUF, to resolve a security issue in a conventional protocol.

Apart from using rPUF to improve the security level of the PUF primitive itself or its developed protocol, rPUFs are also proposed to improve the reliability of certain PUF instances. The authors of [77, 140] independently proposed the use of configurable structure on FPGA platform to design ring-oscillator PUF that has better quality. The basic idea is to choose fundamental elements through a multiplexer to achieve a ring-oscillator PUF that has the best performance in terms of reliability and uniqueness.

Logically recyclable PUF introduced in [54] makes PUF reusable in certain applications such as recyclable RFID-based authentication token, air flight luggage tags, secure update of secret keys, prevention of malicious downgrading of software version, etc. The introduction of L-rPUF dates back to [38] where the “Controlled” PUF was proposed. It is basically a hash function that interfaces the input challenge and the output response. Due to the diffusion effect of the hash function, the actual challenge and response of the PUF are hidden. The controlled PUF can be used to thwart man-in-the-middle attacks. A generalized version of the controlled PUF is the “Logically Recyclable” PUF [29, 54]. It uses control logic to interface the challenge and response, but the interface is implemented as block cipher whose state is kept secret. By initializing the PUF with different configuration states, the chal-

lenge and response are diffused by the cipher module in a way different from the original challenge and response of the PUF before the configuration.

In [63], the authors introduced an implementation of P-rPUF that exploits the sensitivity of certain material under optical stimulation so as to achieve physical reconfiguration. Each time the material is exposed under a laser beam, its physical characteristics will be disordered. The response bits extracted from the PUF will correspondingly be altered after this process.

Another P-rPUF instance proposed in [63, 143] exploits the measurable resistance of phase change memory (PCM) cells to produce the response bits. By re-programming the cell with fuzzily controlled electrical pulses, resistance values of the PCM cells can be randomly changed, which in turn changes the response bits generated from the same cell. Detailed analysis and experimental demonstration of PCM-based rPUF are further presented in [67]. A novel design of PCM-based rPUF that offers enhanced security against prediction attacks was also developed. Instead of relying on the measurable resistance, the number of electrical pulses that converges a cell resistance to a pre-determined value is used as a response.

Rhumair *et al.* [108] defined a PUF primitive called “Erasable” PUF, which is essentially a P-rPUF. It is constructed based on resistive memory structured as a cross-bar array. Whenever a voltage across the memory cell is higher than the threshold, the I-V characteristics of the cell will be altered. Therefore, the measured electrical characteristic that is digitized as response bit is refreshed.

Device aging effects were also applied to change the device physics so as to alter the challenge-response mapping of a PUF. Meguerdichian and Potkonjak [85] proposed the aging PUF to match the challenge-response characteristics of two PUFs to establish an authentication protocol between two parties.

2.2 Emerging Non-Volatile Memory Technology

eNVM technologies such as PCM, Resistive RAM (RRAM) and Spin-Transfer Torque Magnetic RAM (STT-MRAM) emerged in the recent years as promising candidates to replace conventional memories such as SRAM and DRAM in computer systems for higher performance, lower power consumption and higher density.

In eNVMs, data bits are represented by the resistance states of the Storage Device (SD) in eNVM cells, i.e., High Resistance (HR) and Low Resistance (LR). Therefore, the resistance mismatch between two eNVM cells whose SDs are in the same state can be measured to generate a response bit. STT-MRAM cell stores data bits in the Magnetic Tunnel Junc-

Fig. 2.2 Structural diagrams for eNVM technology devices: (a) PCM, (b) RRAM and (c) MTJ.

tion (MTJ). The relative magnetization of the pinned and free magnetic layers determine the resistance state of the MTJ (i.e. LR when the two magnetizations are in parallel, and HR when they are anti-parallel) [135]. PCRAM uses the atomic structure of the SD to store a data bit (if GeSbTe-based SD is used, it is LR when the GeSbTe layer is in the crystalline state, and HR while in the amorphous state) [137]. The SD of RRAM considered in the work of this dissertation is constructed using HfO_x . The length of the conductive filament determines its resistivity (i.e. LR when the filament shorts the electrodes, and HR when the filament is broken) [136].

The configurations of SDs for the three eNVM technologies used in this dissertation are shown in Fig. 2.2. Each SD is connected to an access transistor (omitted in Fig. 2.2) to form a memory cell. Due to process variations, the eNVM cell resistances are randomly distributed. For example, the contact area of the SD may change as a result of lithographic variations. [15, 96]. Moreover, variations in the access transistors (e.g., threshold voltage and channel length) also contribute to resistance discrepancies of the memory cell. These variation sources can be extracted securely by using certain circuit level techniques for generating device-specific bit strings. Additionally, thanks to the unique features of eNVM storage devices, the PUF primitives designed based on them are appealing in many aspects, which will be discussed in details in the following chapters.

2.3 Related Work

2.3.1 Alternative PUF types

On top of the research and development on conventional PUFs such as delay-based PUF (e.g. variants of ring oscillator based PUF and arbiter-based PUF) and memory-based PUF (e.g. different implementations of cross-coupling structured memory PUFs), a group of emerging PUFs sprouted in the literature.

Development of reconfigurable PUF

Tremendous research efforts have been devoted to the development of rPUFs. Considered as a simplistic version of L-rPUF, controlled PUF was proposed earlier in the literature for enhancement of security against man-in-the-middle attacks [38]. Similar work proposed in [77, 140] used control blocks such as multiplexers or controlled hash functions to implement L-rPUFs. The work in [29, 54] firstly put up with the terminology of “logically” reconfigurable (refreshable) PUF, and used query algorithm as a generic approach to implementing such an L-rPUF.

Besides the research on logical type of reconfiguration, another stream focused mainly on P-rPUF, which can be found in [63, 67, 69, 85, 107, 108, 143, 146, 152]. The P-rPUFs achieve reconfigurability of challenge-response mapping property via physical approach. Emerging material characteristics, semiconductor device aging effects or alterable I-V characteristics were leveraged to achieve physical reconfigurability for a PUF instance.

Public PUFs

Simulation-Possible-but-Laborious (SIMPL) systems [21, 22, 111–113, 115] and Public PUFs (PPUFs) [8, 80, 81, 85, 102, 106, 133] refer to a group of PUFs (or PUF systems)¹ that can be used in a way analogous to public-key cryptography. Basically, a PPUF has the following features [113]:

- PPUF is a physical system that can generate a device-specific response R when stimulated by a challenge C due to the physical disordered properties of the system;
- The challenge-response mapping behavior of the physical system is assumed to be stable over time;

¹For convenience, both SIMPL system and PPUF collectively are called PPUF hereafter.

```

B chooses  $k$  challenges  $C_1, C_2, \dots, C_k$ ;
for  $i = 1 \rightarrow k$  do
    B sends  $C_i$  to A;
    A uses the PPUF to produce  $R_i$  according to  $C_i$ , and send  $R_i$  back to B;
    If B does not receive  $R_i$  from A within a time bound, he sets a verifier  $V_i$  as  $\perp$ ;
end for
B computes all  $R_i$  ( $1 \leq i \leq k$ ) and verifies if  $R_i = V_i \neq \perp$ ;
if This is the case then
    The identity of A is verified;
else
    The identity of A is denied.
end if

```

Fig. 2.3 Identification protocol based on PPUF.

- A numerical algorithm, denoted as Sim, exists to calculate R according to a given C . However, the speed of obtaining a response using Sim from the system is remarkably slower than physically deriving the responses from the system.

Public key cryptography (known as asymmetric cryptography in some context) is based on certain mathematical problems such as integer factorization that have no efficient solution unless a secret information is given to solve the problem [86]. Owing to the asymmetry between physical evaluation and computer simulation, the PPUF device always computes the responses to the given publicly known challenges faster than any other means via simulation. While the asymmetries of public-key cryptography stem from the use of different keys to perform the function and its corresponding inverse function, PPUF relies on the heavy mismatches between the efficiency of physically synthesized and software simulated response generation to establish security protocols that can achieve similar purpose. Instead of relying on the mathematical intractability of deriving the private key from the public key, the strength of PPUF protocol lies in the fact that the adversaries are unable to obtain the secret keys in time to break the cryptographic system.

This is an illustrative protocol where PPUF is involved. Assuming Alice (denoted as *A*) and Bob (denoted as *B*) are communicating based on a simple identification protocol. *A* physically holds the PPUF and has made the simulation method Sim of the PPUF public. *A* can then prove her identity to *B* via the protocol as shown in Fig. 2.3 [113]. Other protocols such as authentication, key-exchange, bit-commitment and zero-knowledge proof can also be established based on PPUFs.

A number of PPUF implementations have been proposed in the literature. In [21, 115], SRAM based and Cellular Non-linear Network (CNN)-based SIMPL systems are proposed. The former design essentially exploits the reliance of SRAM write operation on the scaling of power supply voltage. If the supply voltage is over-scaled, write malfunction will occur and the data bit will fail to be written into the cell. Simulating successive write-after-read operations performed iteratively on a large array is always much less efficient compared to the real-life physical execution of the same procedure on SRAM chips. The latter design exploits the two-dimensional analog array to realize a PPUF. Since the communication with the CNN is by nature via the transmission of analog signals, it is much faster compared to software simulation. A speed discrepancy between the physically evaluated responses using the CNN-based PUF and the simulated response is substantial to be utilized to construct a public-key cryptography.

In [8, 80], PPUFs are constructed based on a network of XOR gates. Although the delays of the XOR gates are publicly known, due to the interleaving inputs and outputs of the gates, numerically evaluating the output with the known XOR gate delays in simulations is a much more tedious job when compared to obtaining the results directly from the circuit itself. The idea of PPUF was further developed in [102] as differential PPUF. This new primitive lifted off the reliance of the XOR-gate based PPUF on accurate clock manipulation. Instead, it uses two input signals to race against each other to compute a final output. Matched PPUF is another variant following this train of thought [85]. It exploits aging effects of IC devices to achieve accurate match of two individual PPUF devices that cannot be matched with a third one. In this manner, two parties can establish a secure communication protocol by using their matched PPUFs, and any other malicious attempts to match the devices will be futile owing to their low efficiency.

Besides conventional CMOS-based implementation, PPUF can also be realized based on emerging technologies. In [106], the PPUF is designed by memristors configured in a cross-bar array. By making use of the non-linearity of memristive device and the bi-directionality of signal transmission between adjacent devices, a sneak path that is exponential to the size of the array can possibly be found. By randomly selecting M-omino shapes from the array to transmit signals to obtain the outputs at certain nodes, it takes much longer time to obtain the results via software simulation than physical evaluation. A similar idea was proposed in [133] as “Bidirectional Polyomino Partitioned” PPUF where either memristors or nanowires are used as the media to construct a PPUF that relies on the uniqueness of the partitions of bidirectional array to achieve asymmetries between simulation and physical evaluation.

2.3.2 PUF quality improvement techniques

Fuzzy extractor

A conventional approach to improve the reliability as well as randomness of responses generated from a PUF is to use a fuzzy extractor. As defined in the first section of this chapter, a fuzzy extractor essentially consists of an ECC codec and a universal hash function, whose parameters are given by the pre-determined helper data. The benefit of using a fuzzy extractor is that it leaks negligible information of the original response to adversary by making the helper data publicly known. However, one of the most critical issues of a fuzzy extractor is that it requires huge amount of hardware resources to meet the specifications of error correction capability. In addition, using ECC for reconciling noisy response will induce entropy-loss owing to the information rate trade-offs.

Research on applying ECC and universal hash function on PUF, as well as optimization and improvement on the base scheme can readily be found in the literature. A mathematical construction of a fuzzy extractor and a theoretical proof of its efficiency for biometric applications were originally provided [25]. Bosch *et al.* proposed an efficient fuzzy extractor algorithm that decreases the overhead of conventional ECC codec by using concatenating code [16]. Maes *et al.* proposed a fully functional work of PUF system where a fuzzy extractor consisting of a BCH codec as ECC and a Linear-Feedback-Shift-Register (LFSR) as universal hash function [75]. Besides the development of fuzzy extractors, researchers in [87] also pointed out the weakness of the commonly used fuzzy extractor module under the threat of side-channel attacks.

Circuit-level techniques

Memory-based PUFs (MemPUFs) such as SRAM-based PUF, butterfly PUF, sense-amplifier PUF, flip-flop PUF, buskeeper PUF, or the most recent non-volatile memory based PUFs (e.g., Phase Change Memory (PCM)-based PUF and Spin-Transfer Torque (STT) Magnetic RAM (MRAM) based-PUF) refer to a group of PUFs that exploit the mismatches of individual memory elements to generate response bits [47, 61, 67, 82, 118, 123, 130, 143]. Similar to other types of PUFs, MemPUF is also susceptible to the noise effects such that the response bit generated from each cell cannot be reproduced reliably over time. For example, if the mismatch of driving ability between the two cross-coupled inverters in an SRAM-based PUF is too small, the response bit generation process can be easily corrupted by noise which makes the output bit unreliable. The Bit-Error Rates (BERs) of most PUFs are as high as 10%, and can be further aggravated by the variations of environmental conditions such as the change of ambient temperature and fluctuation of supply voltage.

Fig. 2.4 Adapting the ramp-up time of supply voltage to improve the reliability of SRAM PUF.

One proposed approach to improving the reliability of memory-based PUFs is to adapt to the ramping time of the power-up process to generate random bits, as shown in Fig. 2.4. The experimental results in [14, 74, 116] show that ramping the power-up time helps to reduce the BER by $\sim 71\%$ at room temperature. The simulation results in [24] also demonstrate a reduction in the original BER of various types of memory-based PUFs effectively at different temperatures by employing the ramp-up time adaption technique.

Fig. 2.5 (Left) Aging effects to skew the threshold voltages of transistors in the SRAM-based PUF cell. (Right) Enrollment of SRAM-based PUF with accelerated aging procedure.

Another approach proposed in [14] used aging effects to improve the reliability of bi-stable PUFs (see Fig. 2.5). It deliberately skews the characteristics of the two cross-coupled inverter structure to its preferred data bit after the chip is manufactured. Since aging effects (negative bias temperature instability (NBTI), hot carrier injection (HCI), etc.) can be accelerated if the transistors in the inverters are subjected to proper stress conditions, the characteristics of the two inverters will be biased towards the preferred data bit to make the output state more distinguishable. An additional procedure is required to orchestrate this

reinforcement after the chip is fabricated and before it is used. A similar method to enhance the reliability of SRAM-based PUF is also proposed in [37]. The difference is that the latter method can also be used for optimizing the entropy of the PUF cells. The idea of using HCI effects to age the PUF cell is developed in [13]. The advantage of using HCI is the shorter time required to stress the devices to obtain an acceptable distinguishability. Furthermore, the aging procedure is autonomously controlled by an internal circuit so that no response bits generated from the PUF will be leaked to the outside world.

2.3.3 eNVM based PUFs

Memristive PUF

Rose *et al.* in [106] proposed a novel type of PUF which utilizes the write-time variability of memristive device to generate chip-specific keys which can be used for security applications. Considering the process variation effects in memristive devices, the time consumed by writing the storage element of a memristive device from high resistance state to low resistance state, or vice versa, will be randomly distributed. For example, if a minimum write time for SET process, denoted as t_{min} , is chosen to write a memristive cell from high resistance state to low resistance state, due to process variations, the actual time, t_{real} , taken by this writing operation may be higher or lower than t_{min} . If $t_{real} > t_{min}$, the output logic will be “1” or else the output logic will be “0”. Since the probability of the output logic being “1” or “0” is approximately 50%, the bit can be considered random. By concatenating the 1-bit implementation of the memristive PUF circuitry, an N -bit PUF can be constructed. Evaluation in [106] demonstrated the superiority of the proposed memristive PUF in terms of uniqueness, uniformity and bit-aliasing. When compared to the conventional CMOS-based PUF instances, the memristive PUF has better quality in terms of randomness. Similar to the approach proposed in [106], Patrick *et al.* in [58] also proposed a novel design of memristor-based PUF, which makes use of the weak write mechanism to produce device-specific random responses originating from process variations.

SHIC PUF

SHIC PUF was proposed by Ulrich *et al.* in [110]. The term of “SHIC” stands for “Super High Information Content”, and the SHIC PUF is mainly featured by its extremely high volume of physical information and low read-out rate of this information. As a result, exhausting all the potential random bits producible from the SHIC PUF will take unreasonably long time. In addition, since there is no complexity enhancement owing to interactions between the internal device nodes, the modelling attack, which takes advantage of machine

learning techniques to model and predict the output responses from a PUF with the known CRPs, is not applicable to SHIC PUF. This requires a SHIC PUF to have the following specifications as delineated in [110]:

1. The read-out speed of response bits from the SHIC PUF is limited to a small value by the design of the SHIC PUF circuit. In addition, the slow read-out speed is dependent on the intrinsic physical properties of the SHIC PUF device and/or circuit, instead of any auxiliary component that is controlled by external signals.
2. The total time for characterizing the whole SHIC PUF will exceed the lifetime or the maximum access times of the PUF in its application scenario.
3. The physical properties of individual devices in SHIC PUF should be random and unpredictable due to the inevitable process variation in fabricating the chip.

The authors in [110] specified the above criteria for a SHIC PUF and suggested an illustrative implementation using cross-point structure, which has the following features:

- The SHIC PUF is not writable. The physical information of the SHIC PUF is physically and permanently preserved inside each individual device, and can only be retrieved in a read operation.
- By bridging the bit cell and word-line with high- k material, the junction and interwire capacitance can be increased so that the read-out time can be prolonged.
- The SHIC PUF is laid out as a monolithic block which makes it impossible to exhaust information from the entire PUF in a parallel manner.

The SHIC PUF can be categorized into “strong PUF”, whose possible challenge-response pairs cannot be obtained within a period time that is proportional to the physical size of the PUF chip. As mentioned earlier, the advantage of a SHIC PUF over other strong PUF instances is that it is resilient against modelling attacks based on machine learning techniques.

Memristor based Public-PUF

Another interesting implementation of memristor based PUF was proposed by Rajendran *et al.* in [104]. The design, called “nano-PPUF” by the authors, is based on nano-scale memristor, and the mechanism of challenge-response mapping characteristics is in correspondence with “Public PUF” that was proposed years ago by Beckmann in [8]. The nano-PPUF can be constructed based on the memristor device using TiO_2 material. Owing to process variation sources such as oxide layer thickness, lateral dimension, doping concentration of

oxygen vacancies in the TiO_2 layer, the memristance of the device may vary randomly. The nano-PPUF can be constructed into a cross-bar structure with $N \times N$ individual memristor cells. The security of the nano-PPUF lies in the fact that *simulating the PPUF will take exponentially longer time than physically evaluating it*. The complexity of simulation is a result of the complexity of the memristor device model and the complexity of simulating the in-out behaviour of the cross-bar structure. To further create the asymmetry between simulation and physical evaluation on a given nano-PPUF in the secure communication protocol between two parties where a nano-PPUF is applied, polyominoes are used to select a particular sub-segment in the $N \times N$ cross bar structure so that finding an exactly the same shape of polyominoes wherein the memristor devices are used for response generation is extremely hard if not impossible. Experimental results conducted on a 4×4 cross-bar nano-PPUF demonstrated the efficacy of this approach in creating abundant asymmetry for public-key-like cryptography using PUF primitive.

2.4 Summary

In this chapter, the fundamentals of PUF and eNVM technologies were introduced. A comprehensive review of PUF research in the recent years was also presented. By extensively studying the existing research in PUF technology, it was observed that discovering novel types of PUF, improving PUF qualities and developing optimal design methodologies for good PUF primitives have great impact on revolutionizing cryptographic protocols and addressing emerging security issues with device authentication. Research in these directions will result in designs of alternative PUF types that can be used for versatile applications. In addition, the quality of existing/emerging PUF instances can be boosted by applying emerging technologies so as to achieve higher security and reliability performance in industrial/commercial application. Inspired by the recently published work on PUF, the work conducted in this dissertation, which will be introduced in the following chapters endeavor to resolve the issues and challenges in this realm.

Chapter 3

Reconfigurable PUF

In this chapter, a design of an emerging type of PUF which is called the “reconfigurable PUF” (rPUF), based on eNVM technology, is presented. One distinguishing property of an rPUF is that *its challenge-response mapping characteristics can be dynamically reconfigured logically or physically as demanded*. The benefit of having response bits refreshable is that the security of the PUF itself and/or the security protocol where the PUF primitive is applied can be enhanced. For instance, a potential adversary may try to access a PUF to retrieve a set of response bit strings using various means of attack to achieve specific purpose. However, the success rate of achieving his objective may be lower if the PUF can be reconfigured in a random fashion. This is simply because the information about challenge-response characteristics that the adversary obtained from the PUF before its reconfiguration is not enough for him to perform any effective attack on the PUF after reconfigurations.

For the first time in the literature, a formal definition and dichotomization on rPUFs will be given. This formal definition follows that of the ordinary PUF proposed in the work of [5]. Then a set of quality evaluation criteria for an rPUF will be discussed in details. With these evaluation criteria, one can assess a given rPUF design in terms of unpredictability (both static and dynamic), randomness, reliability and unclonability, which are considered the key properties of an rPUF in any application scenario. Two rPUF designs based on Phase Change Memory (PCM) technology are then introduced. Both designs take advantage of process variability and programming uncertainty in PCM device to generate unpredictable, random and dynamically refreshable response bits. Evaluation on the proposed PCM-based rPUF designs using an experimental PCM chip fabricated based on a 180-nm μ -trench technology will be presented at the end of this chapter to demonstrate the efficiency of the proposed scheme.

The following original contributions will be presented in this chapter:

- A formal definition of reconfigurable PUF is given, which can be used as a framework to qualify and evaluate any given rPUF instance.
- Two PCM-based rPUF primitives are proposed. The first design utilizes the impacts of process variations and programming uncertainties on PCM cell resistances to produce randomness and refreshable response bits. The second design makes use of programming dynamics of PCM cells to produce response bits.
- Both novel designs are evaluated using experimental PCM chips. On contrary to the existing approaches, the proposed PCM-based rPUFs are featured by physical reconfigurability, which makes it more secure in certain application scenarios.

3.1 Fundamentals of rPUF

3.1.1 Definition and classification

An rPUF device is a physical device that possesses a set of configurable states $\mathcal{S} = \{0, 1\}^n$ and can probabilistically map an input challenge c from space $\mathcal{C} = \{0, 1\}^l$ to a response r in space $\mathcal{R} = \{0, 1\}^m$. The response r to a challenge c is dependent on its configuration state $s \in \mathcal{S}$:

$$\text{rPUF} : \mathcal{S} \times \mathcal{C} \rightarrow \mathcal{R} \quad (3.1)$$

Normally, a *fuzzy extractor* is required in an rPUF device to stabilize the noisy responses as well as to increase their entropy [16, 25]. The integrated system consisting of the rPUF device and the fuzzy extractor is called rPUF system or in short, rPUF.

Depending on the method used for reconfiguring the rPUF state, two types of rPUF, namely the *L-rPUF* and the *P-rPUF*, are distinguished. An L-rPUF realizes the reconfiguration via a control logic to provide the reconfigurable interface for a challenge-response pair (c_{int}, r_{int}) of an internal PUF. Depending on the configuration state s , the control logic obfuscates the mapping of internal CRPs, (c_{int}, r_{int}) , to external CRPs, (c, r) . The reconfigurable logic can be implemented by a multiplexer [65, 77, 140] or a cryptographic primitive [29, 38, 54]. Logical reconfigurability efficiently reinforces the reliability of the PUF [77, 140], strengthens its resilience against certain types of attacks (e.g., man-in-the-middle attack [38]), or enables the PUF device to be used repeatedly without compromising security [29, 54].

On the other hand, a P-rPUF achieves reconfigurability by altering its intrinsic stochastic physical properties in such a way that the effects manifested externally can be digitized and represented by s . Examples of P-rPUF can be found in the literature [63, 69, 85, 107, 108,

143]. When compared to L-rPUF, P-rPUF is more area efficient because it does not need additional hardware-induced entropy obtained, for instance, by means of replicated PUF devices such as the selectable oscillators used in the Ring Oscillator PUF [77, 140] or cryptographic primitives such as the block ciphers used in the Logically Recyclable PUF [54]. Since measuring the physical effects utilized to reconfigure P-rPUFs can be more difficult than obtaining the digital signals used to reconfigure L-rPUFs, P-rPUF is also believed to provide higher level of security.

3.1.2 Quality criteria

rPUFs are qualified by *reliability*, *unpredictability*, *unclonability*, and *unbreakability*, which are similar to but not exactly the same as the corresponding parameters used for PUF qualifications.

Reliability of rPUFs

Similar to the definition of reliability for a conventional type of PUF, reliability measures the reproducibility of response bits generated from an rPUF. More specifically, the *reliability* of an rPUF, β_{rel} , is defined as the probability that the response r to a challenge c is identical to the response r' to the same challenge c under the same configuration state s evaluated at a later time, i.e., $\beta_{\text{rel}} := \Pr(r = r')$, where $r' = r + e$, e being an error bit string induced by noise. For conventional PUFs, reliability is defined as the probability of regenerating the same response to the same challenge under different working conditions. The difference between the definitions of reliability for a conventional PUF and that of an rPUF is that

- the responses generated from an rPUF to the same challenge should be reliable under the same configuration states;
- altering configuration state of an rPUF should not deteriorate the reliability of an rPUF.

In some context, Bit-Error-Rate is a metric to evaluate the reliability of a PUF or an rPUF. It is defined as the proportion of erroneous responses bits in the overall volume of response bits generated from a PUF (an rPUF) by using the same challenge (under different configuration states).

Unpredictability of rPUFs

Unpredictability is another critical criterion for conventional PUFs. Intuitively, it is a property of PUF that makes the responses generated from a PUF to some unknown challenges

unpredictable even for cases where some previous challenge-response pairs are known. Under some circumstances, an adversary may get access to the internal challenge and/or response interface, or even the physical layer of the PUF chip to directly read or obtain some responses to some challenges and response properties of the PUF, and in turn predict the responses generated from the same PUF device using some other challenges. This attack is a generic prediction attack where the actual form of attack can be of any kind which makes the successive prediction following a challenge-response collection successful. A specific example of this attack is the *modeling attack* where the output responses of a strong PUF are collected and trained with a PUF model by using machine learning techniques. The model will then be used to predict responses to other possible challenges by using the learning algorithm.

For rPUFs, the definition of unpredictability is similar but with subtle differences. Consider an adversary $\mathcal{A}$ who is able to access an rPUF system so as to collect p CRPs under each of the q different configuration states $\{s_1, s_2, \dots, s_q\}$ but is unable to set the rPUF configuration state. The subset of CRPs collected by $\mathcal{A}$ is denoted by $\mathcal{A}_L$. Based on $\mathcal{A}_L$, $\mathcal{A}$ can mount a prediction attack to derive a numerical rPUF system model to generate responses $\hat{r} \in \mathcal{R}$ to arbitrary challenges $\hat{c} \in \mathcal{C}$ under some configuration states $\hat{s} \in \mathcal{S}$. The set of CRPs produced by this attack is denoted by $\mathcal{A}_P$. The rPUF *predictability*, β_{pre} , is then defined as $\Pr[(\hat{c}, r) \in \mathcal{A}_P]$. Specifically, this probability is called *static predictability* if the predicted state $\hat{s}$ is one of the states from which $\mathcal{A}_L$ is collected in the learning phase, and *dynamic predictability* otherwise. Correspondingly, unpredictability is defined as the complement of predictability, i.e., $1 - \beta_{\text{pre}}$.

It should be noted that the unpredictability is a probabilistic measure of the generated responses to different challenges (under different configuration states) of a (an) PUF (rPUF). It is mostly useful in formally evaluating the security level (e.g., semantic security) of a PUF or rPUF system under a given adversarial scenario. Sometimes, “randomness”, measured by “entropy”, can be found in the literature and is equivalently used to represent a statistical measure of unpredictability. It is mostly used as a figure-of-merit in statistically design a PUF or rPUF physical device to meet the criterion of unpredictability. Another statistical metric, called “uniqueness”, is also conventionally used to evaluate the quality of a PUF or an rPUF by calculating the average Hamming distance between response bit strings generated from a PUF or rPUF.

Unclonability of rPUFs

rPUF is identical to conventional PUFs in terms of physical unclonability. This is because the challenge-response mapping characteristics and dynamic reconfiguration functionality

are both consequences of physical disordered properties due to uncontrollable process variations. This implies that even the foundry that manufactures the PUF chips has no idea about its physical properties hence it is impossible to deliberately fabricate a clone of it. In addition, unclonability sometimes refers to the resilience of a PUF instance to shield itself against mathematical approaches to create a clone of it. This is, to some extent, similar to the definition of unpredictability since either mathematical clone or prediction relies on the previous learnt responses to know the responses generated from a PUF. The only difference is that cloning a PUF is a stronger assertion, since it often implies that the entire response space of a PUF is predictable.

Based on the above preliminaries, unclonability of an rPUF can be formally delineated as follows. A *clone* $\widetilde{\text{rPUF}}$ of an rPUF system rPUF is a physical system, or a mathematical or functional model, that can generate the same CRPs as the rPUF under all possible configuration states of the rPUF, i.e., $\widetilde{\text{rPUF}} = \text{rPUF}$, if $(c, r) = (c, \tilde{r}) \forall s \in \mathcal{S}$, $\tilde{r}$ being the response of $\widetilde{\text{rPUF}}$. $\widetilde{\text{rPUF}}$ is called a *mathematical clone* if it is a numerical model of rPUF, and a *physical clone* if it is another physical rPUF system. The *clonability* of an rPUF system (denoted by β_{clo}) is defined as the probability of making a clone $\widetilde{\text{rPUF}}$ of rPUF, and unclonability is given by $1 - \beta_{\text{clo}}$.

Physical Unbreakability of rPUFs

Tamper evidence is another important property of PUFs. In a nutshell, the tamper evidence property makes any attempt to physically break a PUF instance in order to retrieve the internally information conspicuous. This is because whenever an invasive attack is conducted on a PUF instance, the delicate chip will be inevitably destroyed by the intrusive activities. This will then cause to the PUF to malfunction such that the endeavor to obtain response bits by this process is in vain.

The *physical breakability* (denoted by β_{bre}), also known as the tamper-evidence property of an rPUF is defined as the probability that an rPUF device can be physically probed or tested to obtain the correct response r without destructing its normal CRP mapping function under any of the configuration states. Most of the PUFs fabricated in sub-micron or nano-scale technology are believed to have very high physical unbreakability ($1 - \beta_{\text{bre}}$) because it is extremely difficult, if not impossible, to extract the secret bits in a PUF device via physical means without damaging the chip [87].

Conversely, to change from the RESET to the SET state, either a moderate magnitude and long duration pulse is required to trigger the crystallization process of the GST material [137], or a high magnitude and long fall time pulse is applied to melt the GST material, which then crystallizes during the subsequent slow quenching (the latter approach, which is preferred to drive the cell into its full-SET state, is considered in Fig. 3.1(b)). Since the change in the programmed GST resistivity is continuous, it is possible to program PCM cells into “Partial RESET” (or alternatively, into “Partial SET”) states with pulses that have intermediate values of magnitude, duration, and/or fall time, thus achieving Multi-Level Cell (MLC) storage capability, which allows more than one bit to be stored in a single cell [11, 53, 93]. The logic map and the corresponding cell states (single bit and two bits per cell) are shown in Fig. 3.1(d) [17].

3.2.2 PCM process variations and programming sensitivity

Process variations

Due to fabrication process variations, the physical characteristics and, hence, the electrical properties of PCM cells deviate stochastically from die to die and even within the same die. In other words, the resistances of identically programmed PCM cells will spread randomly [17, 131, 149]. The dominating process parameters considered responsible for this variability include the cell geometry (e.g., GST layer thickness, heater thickness, bottom electrode contact diameter) and the electrical and thermal conductivities of the GST and the heater material. Geometrical variations are ascribed to tolerances of the fabrication process. These variations cause spreads in the GST temperature during the phase change process. Amorphization and crystallization rates under varying temperature will result in a random distribution of programmed cell resistances. Electrical and thermal conductivity variations are caused by local fluctuations in the material composition [131], and result in variations in electrical and thermal resistances of the PCM storage elements.

The reasons for these parameter variations and their effects on programmed resistances are analyzed as follows.

- Variations of geometric parameters such as t_{GST} , t_{heater} and BCD are attributed to the imperfect lithography during the fabrication process. They cause the resistances of the GST layer and the heater to vary randomly. They can also cause the temperature of the programmed cell during phase change process to spread because of their influences on the thermal resistivities of the GST layer and the heater. As a consequence, amorphization and crystallization rates under the varying temperature will result in a random distribution of the programmed resistances.

Table 3.1 Randomly distributed process parameters of 180 nm PCM technology with NMOS selector

PCM cell	Parameter	Description	Nominal Value
Storage element	t_{GST}	GST layer thickness	100 nm
	t_{heater}	Heater thickness	50 nm
	BCD	Bottom contact diameter	180 nm
	σ_{heater}	Electrical conductivity of heater (TiN)	$10^4 (\Omega m)^{-1}$
	σ_{amor}	Electrical conductivity of amorphous GST	$0.1 (\Omega m)^{-1}$
	σ_{cryst}	Electrical conductivity of crystalline GST	$10^4 (\Omega m)^{-1}$
	κ_{heater}	Thermal conductivity of heater (TiN)	20 W/mC
	κ_{amor}	Thermal conductivity of amorphous GST	0.2 W/mC
	κ_{cryst}	Thermal conductivity of crystalline GST	0.5 W/mC
MOSFET selector	E_a	Activation energy	2.3 eV
	L	Channel length	180 nm
	V_{th}	Threshold voltage	0.4 V

- The electrical conductivity (σ_{heater} , σ_{amor} and σ_{cryst}) and thermal conductivity (κ_{heater} , κ_{amor} and κ_{cryst}) variations of the heater and the GST layer are caused by the local compositional fluctuations [131], which result in the varying resistances of the corresponding components.
- The activation energy E_a is considered as a random variable responsible for the stochastic fluctuation in the number of carriers contributing to GST conductivity [31]. This variable can affect the nucleation and growth of crystalline phase material of GST. Therefore, the variation of E_a affects merely the resistance distribution of the SET programming operation.
- Variations of MOSFET selector parameters L and V_{th} lead to non-ideal programming pulse amplitudes and shapes that vary both the SET and RESET programmed resistances.

It is important to note that the programmed resistances of the PCM cells are also known to be sensitive to the applied programming pulses [17, 131]. If the programmed resistance variations caused by pulse fluctuations are comparable to those due to process variations, the combined effects of PCM process variations and programming sensitivity can be used advantageously to obfuscate the cell resistances of the PCM for rPUF.

Programming sensitivity

The programmed resistance of PCM cells is also known to be sensitive to the parameters of the applied programming pulses [17, 131]. Indeed, variations in the amplitude and in the time duration of the program pulse lead to changes in the temperature profile within the

GST layer and in the time allowed for the phase change process to take place, and variations in the pulse fall time result in different quenching of the melted alloy, thereby affecting the extent of GST amorphization/crystallization. If the programmed resistance variations caused by pulse parameter fluctuations are comparable to those due to process variations, the combined effects of PCM process variations and programming sensitivity can be used advantageously to vary the resistance of the PCM cells for the construction of an rPUF.

3.2.3 PCM-based rPUF design-I

CRP Reconfiguration and Generation Scheme

Given a PCM array with N cells, the challenge c to the PCM rPUF is a series of memory addresses $\{c_1||c_2||\dots||c_n\}$ and the responses r are the measured digitized resistances of the PCM cells addressed by the challenge. The PCM array can be reconfigured by a program pulse $Prog_s(A, t)$ controlled by a binary string s representing the configuration state of the rPUF. A and t denote the amplitude and the duration of the pulse, respectively. $Prog_s(A, t)$ can be either a partial-SET or a partial-RESET operation. Partial RESET is used in the rPUF design because it takes less time to program PCM cells. The algorithm for CRP reconfiguration and generation is shown in Algorithm 1.

Algorithm 1: CRP reconfiguration and generation

```

Input:  $s, c = \{c_1||c_2\dots||c_n\}$ 
Output:  $r$ 
/* Initialization and configuration */
1 Initialize the PCM array to full-SET state;
2 for  $i = 1 \rightarrow N$  do
3   if  $i == 1$  then
4      $s_i \leftarrow s$ ;
5   else
6      $s_i \leftarrow \text{Update}(s_{i-1})$ ;
7   end
8   Program  $cell_i$  by  $Prog_{s_i}(A, t)$ ;
9 end
/* CRP evaluation */
10 for  $j = 1 \rightarrow n$  do
11   Select a PCM cell  $cell_j$  using  $c_j$ ;
12   Compare the sensed current of  $cell_i$ ,  $I_{cell_i}$  with  $k$  predefined references  $[I_{ref_1}, \dots, I_{ref_k}]$ ;
13    $r_i \leftarrow$  encoded comparison results;
14 end
15  $r = \{r_1||r_2\dots||r_n\}$ ;

```

Before the PCM array is configured for CRP generation, it is initialized into a full-SET state, so that variations in subsequent partial-RESET programming results will depend only

Fig. 3.2 Programming pulses for $Prog_s(A, t)$: (a) SCD pulses for full-SET programming; (b) uniform A and t in SP method for partial-RESET programming; (c) non-uniform A and t in SP method for partial-RESET programming.

Fig. 3.3 Reconfiguration modes of rPUF: (a) hash mode; (b) TRNG mode; (c) ICR hash mode.

on variations in fabrication process and program pulse parameters. Stair Case Down (SCD) pulses, which in practice emulate pulses with a very long fall time [10], can be used to meet the accuracy requirements of this initialization process, as shown in Fig. 3.2(a). After the whole array is initialized, a Single Pulse (SP) is applied to reconfigure each of the PCM cells to a partial-RESET state. If the same $Prog_s(A, t)$ is used for partial-RESET programming of all PCM cells, the differences in programming results will depend merely on fabrication process variations, as shown in Fig. 3.2(b). If different pulses $Prog_{s1}(A, t)$ and $Prog_{s2}(A, t)$ are used to program the cells, changes in programming results will vary according to both fabrication process spreads and pulse parameter variations, as depicted in Fig. 3.2(c).

Imprecisely Controlled Regulators

To guarantee high static and dynamic unpredictability of an rPUF, its configuration state should be (1) kept private and (2) non-resettable [54]. This implies the requirement of a highly secured scheme to generate $Prog_s(A, t)$ for the reconfiguration operation. Conventional hash-mode [Fig. 3.3(a)] and True Random Number Generator (TRNG)-mode [Fig. 3.3(b)] methods utilize a pseudo- and a true-random number generator, respectively, to produce random binary strings (i.e., random configuration states s), which are then converted to analog values A and t by a Digital-to-Analog Converter (DAC), to reconfigure the rPUF [54, 63, 65]. A common problem of these two approaches is that the configuration state s is at risk if the primitives of the random number generation module (i.e., the hash function in the hash-mode and the TRNG in the TRNG-mode) are attacked physically.

To circumvent this problem, an Imprecisely Controlled Regulator (ICR) [Fig. 3.3(c)] is proposed to reconfigure the PUF [143]. An ICR implementation is shown in Fig. 3.4. It consists of a programming voltage generator [Fig. 3.4(a)] and a pulse generator [Fig. 3.4(b)]: the output, V_{prog} , of the voltage generator feeds the selected cell with the voltage required for a programming operation with the help of the enable pulse, EN , provided by the pulse generator [Fig. 3.4(c)]. The uniqueness of the ICR for any configuration state of the rPUF is guaranteed by the non-deterministic analog values it generates due to the mismatches between replicated circuit elements. If resistance variations caused by programming pulse variations are appreciable, a PCM cell may generate different output bit patterns even when it is programmed with pulses having the same nominal parameter values.

As shown in Fig. 3.4(a), the programming voltage V_{prog} is generated by applying a predetermined reference voltage, V_{ref} , across a resistor, R_1 , to obtain a current, $I_1 = V_{ref}/R_1$. It is then mirrored with a mirror factor nominally equal to m through a resistor, R_2 , matched to R_1 , to obtain a programming voltage $V_{prog} = I_2 R_2 = m V_{ref} R_2 / R_1$. The output section of the current mirror consists of m pairs of identically designed PMOS transistors, each able to generate a fraction I_i ($1 \leq i \leq m$) of the total mirrored current I_2 (nominally, $I_i = I_1$). Each fractional current I_i is obtained by selecting one PMOS transistor in the corresponding pair by means of an m -bit binary string $s_c = [s_{c,1}, s_{c,2}, \dots, s_{c,m}]$. Due to the mismatches between the two selectable PMOS devices in any pair, the current contributions generated for each bit are nominally the same but slightly deviate from one another (i.e., $(I_i)_A \approx (I_i)_B$). The pulse generator (Fig. 3.4(b)) generates the program enable pulse, EN . To this end, a pulse P_i is generated by selecting one of 2^n identically designed inverter chains by means of an n -bit binary string $s_p = [s_{p,1}, s_{p,2}, \dots, s_{p,n}]$; consequently, the time width of P_i , t_i ($1 \leq i \leq n$), which is equal to the time delay provided by the selected inverter chain, spreads stochastically because of the random mismatches between the delay chains. The width of

pulse P_i , t_i , is compared to a predetermined fixed pulse width t_0 (set sufficiently smaller than the minimum t_i) which is produced by a reference delay path, thus generating a pulse, P_Δ , with a time duration $\Delta t = t_i - t_0$. To increase the time duration of the enable pulse to the value required for programming operations, the time duration Δt due to the mismatches between the replicated delay chains is stretched, thus generating the program enable pulse, EN , with a duration of $N \times \Delta t$, where N is the stretching coefficient. The stochastic spread of the time duration of the programming pulse will be on the same order as the pulse time duration, which ensures a sufficiently high spread of the programmed resistance. The configuration state s is the concatenation of two binary strings, i.e., $s_c || s_p$.

Table 3.2 Commonly used ECCs and their code parameters

Code	$[n, k, d]$	$t = \lfloor d - 1 \rfloor / 2$	Code rate
Repetition	$[33, 1, 33]$	16	1/33
RM	$[128, 8, 64]$	31	8/128
Golay	$[23, 12, 7]$	3	12/23
BCH-1	$[127, 8, 63]$	31	8/127
BCH-2	$[127, 15, 55]$	27	15/127

The benefit of adopting an ICR in the proposed PCM rPUF is that the physical effects which reconfigure the rPUF device is concealed even if s is known. This is because the significances of all bits in s are uniform, so that one cannot get any information from s to predict the PCM cell resistances and, hence, the output responses, unless the replicated components of the ICR (PMOS transistors in the programming voltage generator and inverter chains in the pulse generator) are experimentally characterized.

Read Resolution

In the case of conventional bi-level PCMs, the read current flowing through each of the cells selected by the challenge is compared to a reference current, which is set as a function of the sensed current from a cell having the expected SET resistance. The polarity of the comparison results determines the output response bits of the rPUF. For MLC PCMs, the current flowing through each selected cell is compared to $2^k - 1$ references to quantize the cell current into 2^k discrete levels. The values of the reference currents are chosen as a function of the sensed currents obtained with the expected programmed resistance range, taking both process parameter spreads and programming pulse variations into account. The comparison result is a thermometric encoded digital version of the cell current, which is then converted to a binary code to obtain the output response bits of the rPUF. The circuit used for multi-bit read operation and response generation is shown in Fig. 3.5. Using multi-bit readout can effectively increase the entropy contained in each memory cell at the cost of

Fig. 3.4 ICR schematic: (a) programming voltage generator (all PMOS transistors are nominally identical); (b) pulse generator (all delay chains, consisting of an odd number of inverters, are nominally identical); (c) PCM programming circuit. Voltage-mode programming is considered.

increased read resolution. Increasing read resolution degrades bit stability due to the ensuing reduced noise margin, which in turn deteriorates the reliability of the generated responses.

Experimental evaluation

The proposed PCM-based rPUF scheme was evaluated by using 1M-bit arrays of ten 180-nm experimental PCM chips. *Reliability*, *unpredictability*, and *unclonability* were tested, whereas *unbreakability* was not tested¹. Single-bit readout² in Direct Memory Access mode was carried out.

¹Testing *unbreakability* may damage the experimental chips. We assume that PCM based rPUF is *unbreakable*, i.e., $\beta_{bre} = 0$, like other PUFs/rPUFs. This assumption is valid since the chips under test are fabricated by using deep sub-micron technology, and any attempt of non-destructive testing is in practice infeasible.

²One bit was read from each cell since the current references of the used chips were designed for the complete cell current range with cell resistance spanning from full-SET to full-RESET values.

Temperature	$\overline{\text{HD}}$	β_{rel}				
		Repetition	RM	Golay	BCH-1	BCH-2
25°C	6.0%	$1 - 7.4 \times 10^{-13}$	$1 - 3.5 \times 10^{-12}$	$1 - 4.6 \times 10^{-2}$	$1 - 2.8 \times 10^{-12}$	$1 - 1.7 \times 10^{-9}$
45°C	6.0%	$1 - 8.2 \times 10^{-13}$	$1 - 4.1 \times 10^{-12}$	$1 - 4.6 \times 10^{-2}$	$1 - 3.3 \times 10^{-12}$	$1 - 2.0 \times 10^{-9}$
65°C	6.6%	$1 - 3.2 \times 10^{-12}$	$1 - 3.8 \times 10^{-11}$	$1 - 6.0 \times 10^{-2}$	$1 - 3.0 \times 10^{-11}$	$1 - 1.3 \times 10^{-8}$
85°C	6.9%	$1 - 7.6 \times 10^{-12}$	$1 - 1.5 \times 10^{-10}$	$1 - 7.1 \times 10^{-2}$	$1 - 1.2 \times 10^{-10}$	$1 - 3.9 \times 10^{-8}$

Reliability

According to the definition, rPUF reliability was tested by evaluating the responses to a given challenge under a certain configuration state at different time instants and then calculating their Hamming distances. Since all the PCM cell resistances under different configuration states are within the partial-RESET range in the scheme, the noise impact is uniform [23]. Therefore, the reliability deterioration due to reconfiguration can be neglected. A total of 2000 128-cell words were selected to generate response bits for six times. ECCs shown in Table 3.2 were used to stabilize the generated responses (in Table 3.2, n , k , d , and t are the

codeword bit length, the information bit length, the minimum distance between codewords, and the number of correctable errors, respectively). To account for the effects of ambient temperature, reliability tests were repeated for the different ambient temperatures, namely, 25°C, 45°C, 65°C, and 85°C. The obtained results in all cases are summarized in Table 3.3. There is a slight increase in the fractional Hamming distance with increasing ambient temperature due to larger fluctuations of mean trap energy in the chalcogenide material [12]. The reliability β_{rel} can be computed by assuming that the error bits in a response are binomially distributed, that is,

$$\beta_{\text{rel}} = F(t, n, p_e) \quad (3.2)$$

where $F(t, n, p_e) = \sum_{i=0}^t \binom{n}{i} p_e^i (1 - p_e)^{n-i}$ is the binomial cumulative distribution function, t and n are the parameters of the ECCs shown in Table 3.2, and p_e is the probability of an error bit in the response, which can be estimated by the calculated fractional Hamming distance of the responses evaluated at different time instants.

As seen in Table 3.3, most of the computed values of β_{rel} are close to the ideal value of 1, except in the case when Golay code is used, due to its low correction capability, as shown in Table 3.2. ECC code selection comes from a trade-off between code rate and error correction capability. Although Golay code possesses the highest code rate among the listed ECCs in Table 3.2, its error correction capability is inadequate for use in the rPUF design, as it results in a value of β_{rel} smaller than the acceptable standard of $1 - 10^{-6}$.

Unpredictability

Fig. 3.6 Challenge selection methods: (left) horizontal; (centre) vertical; (right) random.

Taking spatial correlations and PCM bit-line parasitics into consideration, the predictability of rPUF responses generated by the following three different methods of challenge selection was estimated:

- *Horizontal* - In this method, a set of memory rows in the array is selected and the responses are generated from m cells of each of these rows as shown in Fig. 3.6(left). For each row except the uppermost and the lowermost ones, the fractional Hamming distance is calculated between its response r_i and the responses, r_{i-1} and r_{i+1} , produced by its two adjacent rows, that is,

$$\text{HD}(i) = [\text{HD}(r_i, r_{i-1}) + \text{HD}(r_i, r_{i+1})]/2m.$$
- *Vertical* - In this method, the responses of m cells are generated from a set of memory columns as shown in Fig. 3.6(centre). In a way similar to the above method, for each column, the fractional Hamming distance is calculated between its response and the responses obtained from its two adjacent columns.
- *Random* - In this method, any response is generated by randomly accessing m cells from the whole PCM array as shown in Fig. 3.6(right). The fractional Hamming distance is then computed between any two responses r_i and r_j , that is,

$$\text{HD}([r_i[1], r_i[2], \dots, r_i[m]], [r_j[1], r_j[2], \dots, r_j[m]])/m.$$

2000 samples of responses with length $m = 64, 128$, and 256 were generated by using the above three methods of challenge selection. The mean values μ and the coefficients of variation σ/μ of the fractional Hamming distances obtained are shown in Table 3.4. The obtained values of μ and σ/μ are fairly consistent among the three challenge-selection methods, which implies that the response bits obtained from a PCM array can be assumed to be spatially independent. The mean values of the fractional Hamming distances are $\sim 30\%$, which are lower than the desired value of 50% . One possible reason for this skew of the output bit pattern is the log-normal distribution of PCM resistances, as pointed out in the preliminary study [143]. Such skew of the raw device responses is not uncommon, and is usually eliminated by post processing the stabilized response bits with a hash function incorporated in the fuzzy extractor [16, 25, 75]. We have implemented a Linear Feed-back Shift Register (LFSR) based hash function [59] to post process the raw responses, and the mean values of fractional Hamming distances of the post-processed responses (shown in brackets in Table 3.4) were enhanced to $\approx 50\%$.

The static predictability was estimated by calculating the entropy of the raw response bits. The entropy of a random source $\mathcal{X}$ is defined as

$$H(\mathcal{X}) = -\sum_i \log(p(x_i)) \cdot p(x_i) \quad (3.3)$$

where x_i is a set of discrete events of $\mathcal{X}$ and $\sum_i p(x_i) = 1$. The $\log()$ function refers to binary logarithm.

The effort to compute the exact entropy of a large random source, such as the responses generated from a PUF, is non-trivial. Therefore, an upper bound of the actual entropy was estimated instead by means of a statistical algorithm called Context Tree Weighting (CTW) [5, 134]. The compression ratio, which is the number of bits required to encode one bit of the raw responses generated from the 1M-bit PCM array, was calculated to be 0.69 by the CTW algorithm. This implies that each cell has an entropy of no more than 0.69 bit. Thus, the predictability β_{pre} of a cell is at least $2^{-0.69}$. Since the output bits of the PCM cells are uncorrelated, the predictability of responses generated from n cells is $2^{-0.69 \times n}$. If an ECC is incorporated in the rPUF system, the predictability of an encoded response will be $2^{-0.69 \times k}$ where k is the information length of the ECC.

Dynamic predictability was evaluated by considering the configuration state probability of each cell. The fractional Hamming distances were calculated from 128-bit responses generated from a unique set of memory cells programmed with 500 different $\text{Prog}_s(A, t)$ controlled by s . The obtained distribution of the fractional Hamming distance of raw responses is shown in the histogram of Fig. 3.7. The mean value μ of the fractional Hamming distance is 29.61%, and is improved to 50.01% after the hash function is used.

It is desirable that the output bit patterns of every single PCM cell for all possible configuration states are uniformly distributed. Proportions of the occurrences of bit “1” and bit “0” were calculated in order to approximate the probability of cell states. The obtained results are shown in Fig. 3.8. Each bar in Fig. 3.8 represents the 1/0 occurrence proportion for a cell over 500 trials of state reconfiguration, and is colored black to indicate the fraction of bit “1” occurrences. The average entropy of a cell of the 128-bit word for all configuration states was estimated directly by Eq. (3.3), i.e., $-\log(p(1)) \cdot p(1) - \log(p(0)) \cdot p(0)$, which turns out to be equal to 0.59. In other words, the dynamic predictability β_{pre} of the cell is $2^{-0.59}$.

The static and dynamic predictabilities for each raw response bit are deviated from the ideal value, i.e., 2^{-1} , due to the log-normal characteristics of PCM resistance distribution [143]. After being processed by the hash function, the above values were improved to $\approx 2^{-1}$.

Unclonability

The unclonability of the proposed rPUF was estimated by calculating the Hamming distances of the raw responses to the same challenge generated by different rPUF devices. 64-bit responses to 100 different challenges generated from each of the 10 chips were collected. The mean values of fractional Hamming distance between the responses of one of the 10 chips used as a reference and the responses of each of the other 9 chips are shown in Fig.

Table 3.4 Mean values and coefficients of variation of fractional Hamming distances for the measured responses to different challenges before and (after) hash function - different wordlengths and CRP generation methods were analyzed

Method	64-bit		128-bit		256-bit	
	μ	σ/μ	μ	σ/μ	μ	σ/μ
Horizontal	29.5% (50.3%)	17.1% (1.9%)	29.8% (51.2%)	17.9% (1.6%)	29.6% (49.9%)	11.2% (1.3%)
Vertical	35.4% (50.4%)	13.9% (1.9%)	32.9% (50.1%)	11.6% (1.6%)	30.7% (50.7%)	10.1% (1.1%)
Random	28.2% (48.9%)	20.2% (1.2%)	28.0% (49.9%)	13.7% (1.7%)	28.2% (49.2%)	10.0% (1.9%)

Fig. 3.7 Fractional Hamming distances of measured 128-bit responses to the same challenge for 500 different configuration states.

3.9. The obtained values are mostly around 40%. The probability of successfully finding a match of two 64-bit responses from any two chips is $F(0, 64, HD)$ where HD is the Hamming distance between the responses, which indicates the disparity of the two chips under comparison. In this case, the clonability of a 64-bit response is equal to $(1 - 0.4)^{64} = 6.33 \times 10^{-15}$. For a 1M-bit array containing 16384 mutually independent 64-bit words, the probability of matching all the responses from these cells, or equivalently the clonability (β_{clo}), is infinitesimally small. The non-ideal values of unclonability, which deviate from 50%, can be further improved by the incorporated hash function, which disperses the processed output bit strings among different chips [16, 75]. As shown in Fig. 3.9, the Hamming distance after the hash function has been improved to almost 50%.

3.2.4 PCM-based rPUF design-II

Due to the nature of the CRP reconfiguration and evaluation mechanism depicted in Fig. 1(b), the CRPs of the PCM rPUF in the previous section (and of any generic P-rPUF device)

Fig. 3.8 Measured 1/0 occurrence proportions for 128 cells from 500 random state configurations of each cell (black=“1”, white=“0”).

Fig. 3.9 Mean values of the fractional Hamming distances for the responses to the same challenges from 10 chips (the response of one of the chips to each challenge is used as a reference).

can still be exhaustively accessed as in the case of a normal PUF during the time interval between two reconfigurations. An alternative rPUF scheme is introduced in this section to circumvent this problem.

Time- and Frequency-based Reconfiguration Schemes

There are two possible ways to refresh the CRPs of a P-rPUF: 1) the rPUF is reconfigured periodically at a time interval equal to t_R ; and 2) the rPUF is reconfigured after n_R CRPs have been evaluated. A clock or an internal counter can be used to monitor the time t_R or the number of CRP evaluation, respectively, after the last reconfiguration has been carried out over the rPUF. Upon expiry of the timer or the counter, the state of rPUF will be reconfigured to refresh its CRPs. For the time-based reconfiguration scheme, if the time taken to complete a CRP evaluation is t_E , the number of CRPs accessible by the adversary $\mathcal{A}$ in the learning

phase of the prediction attack will be limited within

$$p \times q = \left\lfloor \frac{t_R}{t_E} \right\rfloor \times q \quad (3.4)$$

where q and p are defined as the number of configuration states and the number of CRPs collected for each configuration state, respectively. Time t_E is very short for the majority of PUFs and typically lies in the range of nano-seconds. The time interval between two consecutive state reconfiguration operations has to be very brief in order to reduce the number of CRPs that can be collected by $\mathcal{A}$. For the frequency-based reconfiguration scheme, $\mathcal{A}$ can obtain $n_R \times q$ CRPs during the learning phase. Prediction attacks can be thwarted by making n_R reasonably small so as to limit the maximum number of CRPs obtainable by $\mathcal{A}$ for each configuration state.

CRP Reconfiguration and Generation Scheme

Fig. 3.10 SCU scheme for multi-bit programming operation.

Fig. 3.11 Simulation results for SCU partial-RESET programming: programmed cell resistances obtained by using a varying number of pulses with ΔV of 25 mV, 50 mV and 100 mV. Plots (a), (b) and (c) correspond to three different cases of pulse duration, i.e., 50 ns, 100 ns and 150 ns, respectively.

Algorithm 2: CRP reconfiguration and generation

Input: $s, c, res(target), V_{start}, t, \Delta V$
Output: r
 /* Reconfiguration is merged into challenge-response evaluation */

```

1 for  $i = 1 \rightarrow n$  do
2   Select a PCM cell  $cell_i$  using  $c_i$ , initialize it to full-SET state;
3   Set  $V_p = V_{start}$ ,  $t_p = t$ , and  $num_i = 0$  initially;
4   while  $res(cell_i) \leq res(target)$  do
5     Partially RESET  $cell_i$  using  $Prog(V_p, t_p)$ ;
6     Update  $s$ ;
7      $V_p = V_p + (\Delta V)_s$ ;
8     Determine  $t_p$  for the next iteration according to  $s$ ;
9      $num_i \leftarrow num_i + 1$ ;
10  end
11   $r_i \leftarrow Codec(num_i)$ 
12 end
13  $r = \{r_1 || r_2 \dots || r_n\}$ ;
```

Table 3.5 Mean values of Hamming distance of responses to the same challenge at different temperatures for different quantizations.

Temperature	Quantization resolution		
	1-bit	2-bit	3-bit
25°C	9.0%	7.4%	11.5%
45°C	9.8%	10.0%	16.0%
65°C	10.9%	8.8%	16.8%
85°C	9.8%	9.7%	18.2%

Table 3.6 Mean values and coefficients of variation of the fractional Hamming distances for the measured responses to the same challenge before (after) hash function using the augmented scheme - different response word lengths, quantization levels, and CRP generation methods were analyzed

Quantization resolution	Challenge selection method	64-bit		128-bit		256-bit	
		μ	σ/μ	μ	σ/μ	μ	σ/μ
1-bit	Horizontal	48.8% (49.2%)	8.6% (1.3%)	48.5% (49.8%)	6.0% (1.3%)	49.8% (49.7%)	4.2% (1.1%)
	Vertical	48.6% (49.9%)	9.1% (1.2%)	48.6% (49.9%)	6.9% (1.2%)	48.8% (50.0%)	4.5% (1.0%)
	Random	48.4% (50.2%)	12.8% (1.3%)	48.7% (49.8%)	9.1% (1.1%)	48.7% (50.0%)	6.4% (1.1%)
2-bit	Horizontal	49.4% (49.8%)	8.6% (1.2%)	49.5% (50.1%)	6.0% (1.3%)	49.4% (50.1%)	4.3% (1.0%)
	Vertical	48.4% (49.9%)	12.6% (1.3%)	48.8% (50.2%)	8.9% (1.3%)	48.8% (50.9%)	6.8% (1.3%)
	Random	50.1% (49.9%)	12.5% (1.1%)	50.1% (49.7%)	8.9% (1.3%)	49.9% (49.9%)	6.2% (1.2%)
3-bit	Horizontal	40.5% (50.1%)	15.8% (1.3%)	40.5% (50.2%)	15.1% (1.1%)	40.5% (49.8%)	14.9% (1.1%)
	Vertical	42.7% (49.9%)	14.0% (1.2%)	42.7% (50.1%)	9.7% (1.4%)	42.7% (50.1%)	6.8% (1.2%)
	Random	49.5% (50.2%)	12.6% (1.3%)	49.6% (50.0%)	8.9% (1.1%)	49.5% (50.2%)	6.4% (1.3%)

To increase both static and dynamic unpredictability, an alternative rPUF design is proposed, which allows the PUF to be reconfigured periodically with a very short time window, which is set equal to t_E . Consider the Stair-Case Up (SCU) partial-RESET programming scheme, where the PCM cells are programmed to a partial-RESET state with electrical pulses whose amplitude is incrementally increased by a constant step ΔV [17]. The PCM resistance increases in steps with increasing pulse amplitude, as shown in Fig. 3.10 [17]. Since programming results are sensitive to the parameters of programming pulses, if the voltage step and/or pulse duration are changed, the programmed PCM resistances will also change. The above programming strategy is simulated by using a PCM compact model [122]. Fig. 3.11(a), 3.11(b), and 3.11(c) illustrate the relationship between the simulated PCM cell resistance value and the number of programming pulses for three different time durations, namely 50 ns, 100 ns and 150 ns. The pulses have an initial voltage amplitude of 3 V and increase in steps of ΔV . The three curves on each chart correspond to ΔV of 25 mV, 50 mV, and 100 mV, respectively.

Based on the SCU programming method, a Program-and-Verify (P&V) approach [128] was implemented. Upon application of each programming pulse, the cell resistance is verified against the target value to determine if an additional pulse is needed. This process is iterated until the cell resistance reaches the target range [17]. Due to fabrication process variations, the number of programming pulses required to make the cell resistance converge to the target value varies randomly. In addition, the programmed PCM resistances will also vary if the step size and the time duration of the pulses are randomly altered by the voltage regulator according to the controlling configuration state s . These phenomena were leveraged to devise the improved CRP reconfiguration and generation scheme (Algorithm 2).

In the new rPUF scheme, the initial magnitude of the programming pulse, V_{start} , is chosen depending on the target resistance $res(target)$, which is an intermediate value between the full-SET and the full-RESET resistance. ΔV is controlled by the configuration state s . Due to the variations in PCM fabrication process parameters, voltage step size and pulse duration, the number, num_i , of pulses required to program the i -th cell to the target resistance $res(target)$ is distributed stochastically. An encoder `Codec()` is used to quantize num_i . Quantization of n bits per cell can be achieved by dividing the range of num_i into 2^n sub-ranges. When compared to the approach proposed in the previous section, the improved rPUF scheme limits the number of accessible CRPs between two reconfiguration operations to only one, because the time interval t_R between two consecutive reconfiguration processes is merged with t_E .

Experimentally evaluation

Fig. 3.12 Measured cell state occurrence proportions in the case of (a) 1-bit, (b) 2-bit, and (c) 3-bit quantization for a 128-cell word.

The augmented PCM-based rPUF scheme was tested for *reliability*, *unpredictability*, and *unclonability* by using the same PCM chips as in PCM-based rPUF design I. CRP re-configuration and generation were accomplished by using Algorithm 2 instead. The same SCD pulses in the PCM-based rPUF design I were used for initialization. The target resistance for the P&V procedure was set to $20\text{ k}\Omega$. V_{start} was set to 3 V. A method similar to that used in the previous section was applied to select ΔV from the set $\mathcal{V} = \{20\text{ mV}, 25\text{ mV}, \dots, 50\text{ mV}, 55\text{ mV}\}$ and the duration of each pulse in the SCU consecutive pulses from the set $\mathcal{T} = \{50\text{ ns}, 100\text{ ns}, 150\text{ ns}, 200\text{ ns}\}$.

Fig. 3.13 Distribution of fractional Hamming distances of the responses generated by one PCM chip to the same challenge for different quantization levels (128-cell words): (a) 1-bit; (b) 2-bit; (c) 3-bit.

Reliability

Reliability was tested at the ambient temperatures, i.e., 25°C , 45°C , 65°C , and 85°C , over 500 128-cell words. The mean values of fractional Hamming distance of 500 responses to the same challenge under different configuration states were calculated. Each of the responses was evaluated six times. Three quantization levels, i.e., 1-bit, 2-bit, and 3-bit of Codec() were considered. The obtained results are shown in Table 3.5.

The mean values of the fractional Hamming distance are higher than those of the baseline rPUF scheme (Table 3.3) because programming pulses in the SCU P&V procedure can also be influenced by environmental condition variations. Increasing the number of output bits to 3 leads to remarkably rising error probability because output patterns are more prone to fluctuations if a fine-grained quantization is applied. Under extreme conditions, not all error bits can be successfully corrected by the ECCs in Table 3.2 so as to keep the rPUF reliability above an acceptable bound of $1 - 10^{-6}$. Concatenated codes with higher error

Fig. 3.14 Mean values of the fractional Hamming distances for the responses generated by 10 PCM chips when using three quantization levels (a) before and (b) after being processed by the hash function.

correction capability can be used to resolve this problem [16]. By concatenating an inner code $C_1[n_1, k_1, d_1]$ with an outer code $C_2[n_2, k_2, d_2]$, the reliability β_{rel} can be improved from

$$\beta_{\text{rel}} = F(t_1, n_1, p_e) \quad (3.5)$$

when only C_1 is used for error correction to

$$\beta_{\text{rel}} = F(t_2, n_2, 1 - F(t_1, n_1, p_e)) \quad (3.6)$$

where $F(t, n, p_e)$ is the binomial cumulative distribution function as introduced in the previous section (Eq. (3.2)). For example, if RM-[128, 8, 64] code and repetition-[3, 1, 3] code are used as an inner code C_1 and an outer code C_2 , respectively, to form a concatenated code, the reliability β_{rel} of the augmented rPUF with 3-bit quantized output at 85°C will be $1 - 3.83 \times 10^{-8}$, which is much higher than the value $\beta_{\text{rel}} = F(31, 128, 0.1817) = 0.97$ obtained by using only the RM-[128, 8, 64] code.

Unpredictability

The three challenge selection methods described in the previous section were used to generate the responses from PCM cells. Responses of different word lengths were generated from 64, 128 and 256 selected cells. The fractional Hamming distances of the generated responses were calculated. The mean values and the coefficients of variation for different combinations of response word lengths, quantization and CRP generation methods before hash function and after hash function (in brackets) are shown in Table 3.6. The mean values of the fractional Hamming distances for the raw responses are close to the ideal value of 50% except for the case of 3-bit quantization level: indeed, when the resolution increases, the log-normal characteristic of the PCM resistance distribution tends to affect the Ham-

ming distance and, hence, the unpredictability of rPUF more heavily. The upper bound of the response entropy of a cell was estimated by using the CTW algorithm. The obtained values were 0.79, 0.83, and 0.86 for 1-bit, 2-bit, and 3-bit quantization level, respectively. The static predictabilities of the output patterns of a single cell for the three quantization levels are therefore $2^{-0.79}$, $2^{-2 \times 0.83}$ and $2^{-3 \times 0.86}$, respectively. After the adoption of an LFSR-based hash function, the fractional Hamming distance converged to $\approx 50\%$ and the estimated predictability was improved to $\approx 2^{-1}$ for each bit.

The dynamic predictability was estimated by computing the cell state occurrence proportions. 128 cells were randomly selected to generate responses for three different configuration states. The obtained state occurrence proportions for each cell are depicted in Fig. 3.12. The fractional Hamming distances of the responses to the same challenge for different configuration states before and after hash function were also plotted (Fig. 3.13). By using Eq. (3.3), the dynamic predictabilities of the raw responses for the three quantization levels were obtained as $2^{-0.68}$, $2^{-0.49 \times 2}$ and $2^{-0.48 \times 3}$, respectively, for each PCM cell. The dynamic predictability drops with increasing quantization level, but deviates from the ideal values because the proportions of certain patterns for some cells become zero. For example, in Fig. 3.12(c), the proportion of pattern “111” is zero for some cells in the 128 samples, which leads to the lower than expected entropy value. The LFSR-based hash function successfully brought the dynamic predictability of the responses to $\approx 2^{-x}$ for each cell where x denotes the quantization level.

Unclonability

The unclonability of the augmented rPUF was tested on 10 PCM chips. 100 words of 64 cells were selected from each of the 10 chips to generate the responses by using the three quantization levels (1-bit, 2-bit and 3-bit). Hamming distances between the responses from one reference chip and the other 9 chips were calculated. The obtained mean values of the fractional Hamming distances of the 100 64-cell raw response samples from these chips are shown in Fig. 3.14(a). The mean values of the fractional Hamming distances are almost 50% for all the considered quantization levels. The physical clonability β_{clo} of a response generated from 64 randomly selected cells quantized by an n -bit Codec() is estimated to be $(1 - \text{HD})^{64 \times n}$. For the chips tested, the obtained values of β_{clo} were 5.42×10^{-20} , 2.94×10^{-39} and 1.59×10^{-58} for 1-bit, 2-bit, and 3-bit quantization level, respectively. These values indicate the practical impossibility to fabricate two PCM-based rPUFs that have exactly the same CRP behaviour. Fig. 3.14(b) illustrates the fractional Hamming distances after the raw response bits have been processed by the hash function.

3.3 Summary

In this section, rPUF is formally conceptualized and dichotomized into two categories - logically rPUF (L-rPUF) and physically rPUF (P-rPUF). Then two PCM-based rPUF designs were introduced. In both designs, process variability and programming uncertainty of PCM cell devices were exploited to produce response bits which are dynamically refreshable as required. The first design extracts the randomness of PCM cell device resistance for response generation. Owing to the cumulative effect of fabrication process variations and uncertainty during programming operation, the randomness may be distributed arbitrarily from cell to cell and from program cycle to cycle. The advantage of this design over the conventional PUF is physically reconfigurable so that the physical security of the reconfiguration part of the rPUF is not compromised assuming the tamper-proof property of the PUF primitive. The second design augments the security attribute of the first one. It utilizes the uncertainty during each programming procedure to generate response bits. In this manner, the response bits are produced at the same time when the configuration state is updated so that any attempt to measure-and-then-predict at separated time frame will become difficult. Both designs were verified by physical measurements on an experimental PCM chip which was fabricated using 180 nm μ -trench technology.

The theoretical analysis and evaluative approach in this chapter demonstrated the promising features of the proposed rPUF design based on PCM, which can be applied in different applications requiring high security standard that is not achievable using traditional instances. In the next chapter, possibility of exploiting instinct properties of eNVM technologies for designing quality-enhanced PUFs will be discussed.

Chapter 4

Quality Improvement of PUF Based on eNVM

Memory-based PUF (MemPUF) has emerged as an inexpensive yet highly secure way of generating and storing the cryptographic key in memory for operations such as authentication of devices, protection and communication of confidential information [47, 73, 118, 123, 143]. It harvests the intrinsic manufacturing process variations of memory or memory-like array to produce a set of chip-unique output bits (responses) from a set of addresses (challenges) that are prohibitively hard to simulate, emulate or predict. The exact mechanism of the challenge-response generation cannot be copied to another physical device due to the imperfections and uncertainties in the fabrication technology, hence the term PUF. Nonetheless, as the number of possible challenge-response pairs (CRPs) is linearly related to the size of the memory, MemPUFs are usually classified as “weak PUF” or “obfuscating PUF”. This is in contrast to “strong PUF” that possesses exponential number of possible CRPs with respect to the number of physical devices [115]. The advantages of MemPUFs over other types of PUFs are their simple structure and the efficiency of implementation in memory sub-systems.

Although many MemPUFs have been proposed, most of them suffer from two key problems: 1) *the reliability of response bit generation is poor*, and 2) *only one bit can be generated from each cell*. The most common and convenient way to solve the aforementioned issues is to use architectural techniques such as Error Correction Code (ECC) [16, 25] and interface module [29, 38]. ECC stabilizes the noisy response bits generated from the MemPUF, while the interface module such as control-logic obfuscates the stabilized bits to equivalently increase the response bit capacity for each MemPUF cell. However, if each cell can originally produce only one response bit and reliability of the raw response bits is poor, the overheads of these architectural techniques may be very costly. This is because control logic

components for entropy increase and more redundant cells to compensate the low reliability of generated response bits will be added. Circuit-level techniques can help to relax the burden of architecture-level methods if they are properly applied. Bhargava *et al* [13, 14] used the accelerated aging effects to skew the bi-stable PUF cells so as to produce reliable bits. However, aging acceleration requires dedicated circuit components or additional testing procedure, and also degrades the performance of the MemPUF cells if the MemPUF is used as normal memory [62]. Cortez *et al* [24] proposed to adapt the ramping time during power-up process of SRAM-PUF to temperature so as to improve the reliability of bit generation, but this degrades the speed of the bit generation process and thus makes the MemPUF inapplicable in certain conditions.

Another approach to improving MemPUF design is to expand the response bit capacity by physical means, which may be achievable in certain MemPUFs. For example, the “re-configurable PUF” [63, 67, 143], which may be implemented using phase change material, is capable of producing refreshable response bits when the PUF devices are reconfigured by external effects. Zheng *et al* in [152] makes use of SRAM write-failure effects to produce random bits, and tunes the supply voltage to increase the entropy contained in each cell. These MemPUFs may be compromised if adversaries know the digital signals that regulate the external effects (e.g. electrical pulse [63, 143] and supply voltage [152]) for bit expansion, i.e., the adversaries may exploit the obtained information to predict the results of response bit expansion.

To this end, a MemPUF with multi-response-bit per cell capacity and inexpensive effective reliability enhancement technique is required. In this work, a MemPUF exploiting the physical variability of the emerging Spin-Transfer Torque Magnetic RAM (STT-MRAM) is proposed. STT-MRAM has emerged as the most promising candidate for universal memory technology, due to its non-volatility, compatibility with CMOS process, near unlimited endurance and high integration density [20, 135]. Recently, some PUF designs based on other emerging non-volatile memory technologies have been proposed [58, 104, 106], but none have explicitly suggested any efficient design techniques to enhance the reliability and response bit capacity for each memory cell. In the proposed PUF, multiple response bits can be extracted by using resistance mismatches between two STT-MRAM cells that store the same state in the Magnetic Tunnel Junctions (MTJs). In addition, circuit-level design techniques that take advantage of the unique features of the MTJ is also proposed to enhance the reliability of the PUF. Compared to the conventional methods such as those proposed in [13, 14, 24, 63, 67, 143], the design takes advantage of the intrinsic device characteristics which are natively exploitable via normal memory operations, and avoids the use of techniques such as accelerated aging, ramping time adaption, and external effects that alter

device properties for regular function. It has eliminated the need for both a pseudo random number generator and a protected non-volatile memory for the generation, storage and retrieval of secret keys in the algorithmic description of cryptographic primitives.

The main content of this chapter can be summarized as follows.

- A generic approach to improving reliability and bit capacity of a MemPUF using eNVM technology is introduced. Automatic write-back scheme which enhances the reproducibility of response bits without disturbing security level is introduced. Physical reconfigurability of eNVM-based MemPUF (e.g., bit alteration effect) is exploited to achieve multi-bit per cell.
- As an illustrative case study, the design of STT-MRAM-based MemPUF is introduced and evaluated in comparison with existing PUF instances implemented using CMOS technology.

4.1 Pros and Cons of MemPUF

MemPUF is regarded as one of the most promising PUF representatives for commercial use thanks to its several appealing features. From the early-stage designs such as SRAM-based PUF, butterfly PUF, latch-based PUF and flip-flop PUF, to the most recent ones like eNVM-based PUFs, MemPUF instances share some common characteristics that distinguish them from other PUFs.

Conceptually, a MemPUF is an array of elements where each individual element of the array, often called the “memory cell”, can be addressed to produce a response bit. The mechanism of response generation of a MemPUF depends on different physical effects of the memory cell used to extract the analog signals affected by the process variability. This mechanism can be formally expressed as

$$\begin{cases} r_i^j = 1, & \text{if } p_i + n_i^j \leq t \\ r_i^j = 0, & \text{if } p_i + n_i^j > t \end{cases} \quad (4.1)$$

where r_i^j is the response bit generated from the i -th cell for the j -th time, p_i is the random physical property which is extracted to generate the response bit from the cell, n_i^j is the noise impact of the cell in the j -th response generation process, and t is the threshold for determining if the output should be 0 or 1. It should be noted that the “noise” may refer to a broader range of effects. For example, aging effect may lead to evolutionary changes in physical properties of semiconductor devices. Resistance drifting and wear-out are those

phenomena found in eNVM devices [17, 51, 66]. All of these impacts may temporarily alter the behaviour of a specific MemPUF and make the generated response bits noisy.

Fig. 4.1 Initial read-out response bit when two resistive memory cells in identical state are unstable in the presence of noise effects. Write-back operation sets one of the cells to an opposite state with respect to the other. The regenerated response bit from the cells in complementary states is more stable.

4.1.1 Pros of MemPUFs

MemPUF is one of the first silicon-based PUF primitives invented by the researchers. SRAM based PUF, being one of the most representative MemPUF implementations, has already been commercialized. MemPUF has several appealing features that give rise to its superiority over other types of PUFs.

- A MemPUF can be easily implemented on various platforms using conventional design approaches. For example, an SRAM PUF can be obtained directly from a computer system since all of the modern computer architecture requires SRAM blocks to serve as a cache memory. Other types of MemPUFs can be readily implemented on FPGA or ASIC without much effort since the basic structure and circuitry is simple and straightforward.
- As the response bits of a MemPUF are generated from individual cells of a memory or memory-like array, the randomness and unpredictability of the generated response bits are better compared to other PUF implementations where internal interleaving of random physical effects may introduce correlations. Therefore, without considerations of spatial correlations or other unexpected and uncontrollable correlation effects that may arise from the manufacturing process, SRAM is expected to be immune against

modelling attack. This is because the information obtainable from an SRAM by learning the responses generated from a subset of cells will not be useful in predicting the response bits from other cells.

- Another advantage of MemPUF is that it can in principle be reusable for two modes of applications, i.e., data storage application and PUF application, although this fact is not currently exploitable in existing MemPUFs due to certain practical issues, which will be discussed in the next chapter. Normally the response generation process is based on the intrinsic random effects of elementary memory cells of the MemPUF. To retrieve the random information from the cell for the purpose of response generation, the auxiliary circuitry of the memory array can be directly used with a minor modification. The benefit of having this reusability is that one can save a specific PUF chip or segment of dedicated IC for PUF application, which is appealing to embedded hardware system with limited hardware resources. While process variations contribute to quality enhancement in PUF application, they jeopardize the stability in regular memory access operations. The conflict between normal operations and PUF application will be resolved by the novel optimization strategy to be presented in the next chapter.

4.1.2 Cons of MemPUFs

Notwithstanding the advantages of MemPUFs, some inherent drawbacks causes them to be less attractive in certain application scenarios. The drawbacks are summarized as follows:

- MemPUF is essentially a weak PUF because its response bits are generated from individual cells. The total number of possible challenge-response pairs is proportional to the size of the memory array. Therefore, the exhaustive search of its entire challenge-response space can be completed in polynomial time. The applications of weak PUF has been restricted to mainly secret key generation. Even then, access to the MemPUF bit pool needs to be protected in case the MemPUF is physically held by an adversary a. A MemPUF can never be used for advanced security application, such as authentication, without encapsulating the internal response bits by a post-processing module that is often implemented as a controlled hash function.
- The reliability of MemPUF is an issue, especially in varying operating conditions. The response bits of MemPUF can be easily deteriorated by noise. The situation gets worse when the operating conditions such as ambient temperature and supply voltage are unregulated. Normally, architectural techniques such as error correction

codes modules and dedicated circuit-level modifications are applied to enhance its reliability. While these approaches have been demonstrated to be effective, they incur non-negligible overheads to the MemPUF system and negate the original advantages of MemPUF.

- Another limitation of MemPUF is that the maximum achievable entropy of each individual cell of a MemPUF is only one. For instance, in an SRAM-based PUF, the response bit is generated from an SRAM cell where the cross-coupling inverters are biased to a metastable state or charged in a power-up process. Therefore, the output bit is essentially binary since it is the result of a “comparison” between the driving strengths of the two inverters in a cell.

4.2 Quality Enhancement Using eNVM Technology

Some quality improvement techniques for PUFs have been reviewed in Chapter 2. In particular, these approaches are applicable in MemPUF instances to improve qualities such as reliability and randomness. The most convenient way to solve the aforementioned issues is to use architectural techniques such as Error Correction Code (ECC) [16, 25] and interface module [29, 38]. ECC stabilizes the noisy response bits generated from the MemPUF, while the interface module such as control-logic obfuscates the stabilized bits to equivalently increase the response bit capacity for each MemPUF cell. In the following sections, a novel method to improve the reliability and bit entropy capacity of MemPUF based on eNVM technology will be introduced.

4.2.1 Automatic Write-Back scheme

Write-back is a common technique used in SRAM design for circumventing stability issues in half-selected cells. The idea of write-back is that the bit read from a cell will be automatically written back to the same cell as the read operation will deteriorate the data integrity. Inspired by this technique, without relying on the aging effects as well as the testing procedure, an automatic write-back scheme can also be used directly in an eNVM based MemPUF to improve the reliability of bit generation [67]. This scheme is depicted in Fig. 4.1. To illustrate this idea, in the following part of this section, an STT-MRAM based PUF was designed, whose initially generated bit can be written back to the cell upon completion of the read operation. The response bit can be reliably regenerated from the cell. Since the two eNVM cells selected by the challenge to generate a response bit are in complementary

Fig. 4.2 $RA - t_{MgO}$ (top) and $TMR - t_{MgO}$ (bottom) characteristics of MTJ. Bit alteration occurs when t_{MgO} and eOV meet certain condition. Labels A and B identify the two MTJs of an SB cell.

state after a write-back operation, the BER of response bit regeneration from the cell will be very low.

4.2.2 Multiple-bit capability per cell

Most of the existing MemPUF is limited to produce only one bit. This limitation comes from the bit generation principle of the MemPUF instance. In the recent research work, attempts to producing multiple bits from a MemPUF cell were taken to increase the entropy content of each MemPUF cell.

MECCA and Retrofitted SRAM PUF

MECCA, in short of (ME)mory (C)ell-based (C)hip (A)uthentication PUF, was proposed by Aswin *et al.* based on the failure mechanism of SRAM memory to produce reliable response bits for authentication and key generation purpose [60]. The distinct feature of MECCA PUF is that, instead of relying on the random and unpredictable power-up behavior of SRAM cells to produce chip-specific response bits, the response bits of a MECCA PUF is generated from the failure effects of SRAM operations ascribed to the process variations in devices' geometries. For example, due to random fluctuations of physical parameters of the

Fig. 4.3 By resetting the state of each MemPUF cell (assuming the MemPUF array consists of N cells) with randomly generated $s_{(i)}$ ($1 \leq i \leq N$), bit produced from the cell may be altered.

transistors in the fabrication process, write/read operation may occasionally fail on certain cells. The steps required to produce random response bits from a MECCA PUF are given as follows:

1. R SRAM cells are indexed by a challenge.
2. For each of the selected SRAM cells, it is initially written into a pre-determined state. Then the bitlines are precharged to the corresponding values, respectively, to write the cells.
3. By reducing the word line duration, the write operations on the cells may or may not fail depending on the physical properties of the cell devices.
4. The bit values in the cells are finally read-out to produce the random bits.

By using a “Word Line Duty Cycle Controller” which modulates the duration of the word-line pulse, the bit generated from a cell may be altered according to the word-line pulse duty cycle. In this manner, equivalently, more than one-bit entropy may be extracted from a cell, because the response bit is dependent on the collective effects of both the physical parameters of the devices and the controllable word-line duty cycles.

Fig. 4.4 Device model of MTJ with process parameters considered as normally distributed variables due to fabrication uncertainties (e.g., the contact area, $Area$, the thickness of oxide layer, t_{MgO} , and the effective oxygen vacancy, eOV).

In the retrofitted SRAM PUF proposed by the same research group, failure effects in SRAM write operations are used to produce random bits, and the bit entropy per cell is further increased by scaling the supply voltage to the SRAM chip. Simulation results show that 6-bit entropy per cell can be achieved by the retrofitted SRAM PUF.

PCM based rPUF

As introduced in Chapter 2, PCM based rPUF is another illustration to demonstrate the possibility of designing MemPUF that has multiple-bit-entropy for each memory cell. The principle of PCM based rPUF is somewhat similar to that in MECCA and retrofitted SRAM, that is, the response bits generated from memory cells are the collective results of process variations and PCM programming uncertainties. By tuning the programming pulse attributes such as pulse duration and pulse amplitude by using an imprecisely controlled regulator, the programming results on a cell may be varied randomly so that the generated response bits will be random and unpredictable.

STT-MRAM bit alteration effects

The approaches introduced in the previous two sections have one thing in common, that is, the multi-bit capacity per cell relied on an externally controlled effect such as controllable word-line duty cycle or inaccurately regulated programming pulse. The study on STT-MRAM storage device, MTJ, revealed that by utilizing the intrinsically random properties of MTJ TMR, more than 1 bits can be extracted from a single STT-MRAM cell without any auxiliary signals.

Fig. 4.5 The relationships (experimental data and simulation results) between MTJ resistance and applied voltage ($-0.6 \sim 0.6$ V) in P and AP states, respectively, with MTJ oxide layer deposited at (a) 180°C (when the oxide layer is free of OV and eOV is assumed to be 0) (b) room temperature (when the proportion of OV reaches its maximum value and eOV is assumed to be 1). The increased eOV results in a decrease of TMR owing to the lowered energy barrier, and also results in an increase of MTJ resistance because the non-coherent tunneling induced by OV is less conductive than the coherent ones.

Consider two MTJs in an SB cell, represented by MTJ_A and MTJ_B , and with parameters, $t_{MgO,A}$ and eOV_A , and $t_{MgO,B}$ and eOV_B , respectively¹. Assuming $t_{MgO,A} = t_{MgO,B}$ but $eOV_A > eOV_B$, then $RA_A > RA_B$ and $TMR_A < TMR_B$, where RA represents the resistance-area product of an MTJ. Since $R_{AP} = (TMR + 1)R_P$, the relationship of $RA_{AP,A}$ and $RA_{AP,B}$ is determined by $\Delta RA_P = RA_{P,A} - RA_{P,B}$ and $\Delta TMR = TMR_A - TMR_B$. If the signs of ΔRA_P and ΔTMR are opposite and the impact of ΔTMR is more significant than ΔRA_P , $RA_{AP,A}$ will be smaller than $RA_{AP,B}$ (as shown in the case when $t_{MgO,A} = t_1$ in Fig. 4.2). At this time, $RA_A > RA_B$ but $RA_{AP,A} < RA_{AP,B}$, and the bit produced from the SB cell will be altered if the MTJ is initialized into different states during enrollment. This phenomenon is called *bit alteration*. If ΔTMR is less significant than ΔRA_P ($t_{MgO,A} = t_2$ in Fig. 4.2) or the signs of

¹Other process parameters (e.g., the MTJ contact area and parameters of the access transistor) do not affect both MTJ TMR and resistance, and are therefore not taken into account in this analysis.

Fig. 4.6 RSD of TMR against RSD of eOV . The nominal value of eOV is assumed to be 0.03 to match the experimental results in [142]. A proper value of eOV RSD, i.e., 0.24%, can be found by sweeping it within the possible range.

ΔRA and ΔTMR are the same ($t_{MgO,A} = t_3$ in Fig. 4.2), the relationships of RA_{AP} and RA_P between MTJ_A and MTJ_B are consistent and there is no bit alteration.

Bit alteration may be exploited to increase the amount of extractable information from each SB cell, i.e., ≥ 1 bit can be generated from one SB cell. Ideally 2 bits can be achieved since there are two initial states for an SB cell, i.e., AP and P. For convenience, the initial state is represented by s which belongs to the set $\{0, 1\}$. Due to the imperfection of randomness originated from each cell, normally less than 2 response bits can be generated from a cell.

It is possible to *re-enroll* the STT-MRAM based MemPUF so that its response bit map can be refreshed (see Fig. 4.3): in a new enrollment phase, before the response bits are generated, the configuration states s of the SB cells in an N -bit array are iteratively set. The configuration state $s_{(i)}$ ($1 \leq i \leq N$) of the i -th cell is generated using the function $\text{Update}(s_{(i-1)}, \tau)$, where $s_{(i-1)}$ is the configuration state of the $(i-1)$ -th cell, and τ is a binary bit from a one-time pad produced using an on-chip random source. The function $\text{Update}(\cdot, \cdot)$ may be implemented by an exclusive OR operator, i.e., $s_{(i)} = s_{(i-1)} \oplus \tau$. After the configuration states for all the N cells are set, the response generation starts. Owing to the bit alteration effects, after resetting the configuration state of the array, the response bits generated from the PUF are refreshed. This property is called “physical reconfigurability” [67]. It can be exploited for enlarging the key space of the PUF, protecting PUF-based persistent storage, recycling the PUF and enhancing the security of PUF against possible leakage [54, 63, 67, 147].

Process variations significantly affect the characteristics of STT-MRAM. The geometry of the STT-MRAM structure, for example, MTJ oxide thickness, t_{MgO} , and cross-section area, $Area$, NMOS channel length, L , and width, W , may vary due to imperfections in

Fig. 4.7 (a) Fundamental components of the STT-MRAM circuitry: SB cell (2T2R), sense amplifier, bit-lines and word-lines; (b) memory organization; (c) system-level view of the secret key generator composed of the internal STT-MRAM and the fuzzy extractor. R, H, and K are the response, the helper data and the generated cryptographic key, respectively.

the fabrication process [33, 68]. In addition, point defect sites such as Oxygen Vacancies (OVs) may also be formed during deposition if the process condition is not well-controlled [83, 88, 142]. Fig. 4.4 shows the MTJ geometry and its variation effects considered in this work. Variations in $Area$ lead to variations in R_{MTJ} . On the other hand, variations in the MTJ tunneling oxide layer (i.e., the oxide thickness t_{MgO} and OV) contribute to variations in both R_{MTJ} and TMR . Variations in the access transistor (i.e., threshold voltage V_{th} and channel length L) cause variations in the overall STT-MRAM cell resistance.

To measure OVs in the MTJ oxide layer, a metric called *effective OV* (eOV) is introduced. It is assumed that $eOV = 0$ when the MTJ oxide layer is free of OVs and $eOV = 1$ when the OV reaches its maximum value. eOV was integrated into the MTJ device model [33] and calibrated against the experimental data in [88, 142] to capture variations in the MTJ due to OV variations. Fig. 4.5(a) ($eOV = 0$) and 4.5(b) ($eOV = 1$) compare the experimentally measured and fitted R_{AP} and R_P for different applied voltages across the MTJ. eOV is assumed to follow a Gaussian distribution, and its Relative Standard Deviation (RSD) can be determined by calibrating the simulation results with data in [88, 142]. Fig. 4.6 plots the RSD of eOV against the TMR of the model. The RSD of TMR in the model matches that in [142] when the RSD of eOV is 0.24%.

Fig. 4.8 Reliability model of MemPUF. The reliability may be deteriorated by noise effects in STT-MRAM, i.e., $b_{t+\delta t} \neq b_t$.

Physically refreshing the response bits is different from *logically* remapping the input-output of a PUF. For example, control-logic interfaces, which can be implemented by a hash function or block cipher, have been used to diffuse the raw bits produced from an internal PUF [54]. Compared to the physical means, the logical reconfigurability consumes larger chip area because the control-logic interfaces are hardware-expensive. The response bit refreshability of the STT-MRAM MemPUF is originated from the intrinsic properties of the MTJ. No additional hardware entropy sources are needed to realize it. Furthermore, the response bit refreshability does not rely on any quantization of electrical signals (e.g., supply voltage or programming pulses, etc.) to increase the extractable entropy from each cell, as used in the contemporary designs [143, 152]. Leakage of these signals may help an adversary to guess the output response bit from the MemPUF cell during/after the re-enrollment. For example, in the PCM-based MemPUF, learning the configuration state that regulates the programming pulses for response refreshment allows adversaries to predict the output bit according to the characteristics of the pulses. However, in the STT-MRAM MemPUF, the adversary has no idea about the response bit generated from the i -th SB cell even if $s_{(i)}$ is known.

4.3 Case Study: STT-MRAM Based PUF With Improved Quality

In this section, a MemPUF design based on the emerging STT-MRAM technology will be introduced. The MemPUF is featured by enhanced reliability and multi-bit capacity per cell. Compared to the conventional methods, the proposed STT-MRAM MemPUF does not rely on any architectural method or heavy circuit-level modifications.

An STT-MRAM cell consists of an MTJ and an access transistor. The MTJ is the basic storage element in STT-MRAM. An MTJ consists of two ferromagnetic layers sandwiching a thin tunneling oxide barrier. One ferromagnetic layer is magnetically pinned (called the pinned layer PL), while the other is not (called the free layer, FL). The MTJ state is stored

Fig. 4.9 (a) (Top) Temporal distribution of the initial angle of an MTJ in P state from 1000 samples. Temperature is 300 K. (Bottom) The corresponding R_{MTJ} . (b) BERs at varying temperature points (from 200 K to 400 K) were calculated as the proportion of SB cells in the 1000 samples that produce unreliable bits in consecutive 10 read-out operations. The MTJs of each SB cell are initialized into P and AP states, respectively. Nominal values and variations (shown in brackets) of the relevant process parameters used in the simulation are shown in the text-box.

as the magnetization of the FL with respect to that of the PL, and may be sensed as the MTJ resistance, R_{MTJ} . When the magnetization of the FL is the same as that of the PL, the MTJ is in the *parallel* state (or P) and $R_{MTJ} = R_P$ (low resistance). When the magnetization of the FL and the PL are opposite (known as *anti-parallel*, or AP state), $R_{MTJ} = R_{AP}$ (high resistance). Hence, different logic bits can be obtained by sensing R_{MTJ} (“0”: $R_{MTJ} = R_P$; “1”: $R_{MTJ} = R_{AP}$). The distinguishability between R_P and R_{AP} is defined as the Tunnel Magnetoresistance Ratio (TMR), and is given by

$$TMR = \frac{R_{AP} - R_P}{R_P} \quad (4.2)$$

Table 4.1 Properties of eNVM candidates for MemPUF with AWB scheme

	STT-MRAM	MRAM	PCM	FLASH
Endurance	$> 10^{15}$	$> 10^{15}$	10^{10}	10^5
Neighbor-disturbance	No	Magneto interaction [50]	Thermal disturbance [101]	Cell interference [56]
Tamper evidence	High	High	High	Low
Side-channel	No	EM emanation [50]	Heat profile [101]	Fault injection [121]

Data of endurance were from [135].

The MTJ state may be programmed by passing a current pulse through it to change the FL magnetization. The current flows from FL to PL to program the MTJ from AP to P. The current direction is reversed to program the MTJ from P to AP. To successfully program the MTJ within a given delay, the current amplitude needs to be larger than a critical current (denoted as I_C). The write operation is accomplished by setting the voltages of the Bit-Line (BL) and the Source-Line (SL).

The conventional STT-MRAM may be modified to design MemPUF. A pair of STT-MRAM cells (2T2R) forms a Secret Bit (SB) cell, which is replicated to construct an array (Fig. 4.7). Consider when both MTJs in an SB cell selected by the memory address (by setting RSE and WL) are identically set to AP (or to P) and a read operation is performed on the cell (i.e. $RDEN = 1$). Due to mismatches between the STT-MRAM cell resistances in the SB cell, the outputs of the sense amplifier, which have been pre-discharged to $VDD/2$, will be driven to different potentials. Note that the differential nature of the sensing scheme cancels out the systematic variations of the STT-MRAM cells in the SB cell. The voltage at node BIT is then latched as a full-swing voltage signal, which represents the response bit extracted from the SB cell. To ensure that the output bits are dominated by the mismatches of the MTJs in the SB cell, the other transistors ($M1$, $M2$, $M5$, and $M6$, and those used in the sense amplifier) are up-sized to mitigate their impact on the output bit generated.

4.3.1 Design of STT-MRAM based PUF with Automatic Write-Back

Maintaining the reliability of response bit generation under varying conditions is a major challenge for MemPUF. The reliability of producing a response bit is defined as *the probability that bit b_t , generated from a selected SB cell at time point t , is reproduced as $b_{t+\delta t}$ at time $t + \delta t$ ($\delta t > 0$)* (see Fig. 4.8). If the mismatches between memory cells in MemPUFs are too small, the reliability of bits generated from the cells may be degraded by time-varying effects such as noise from the memory devices themselves, cross-talk from external components and/or evolutionary change of memory storage element's resistance value. For example, the thermal fluctuation in the fixed layer magnetization (measured as

angle θ relative to the pinned layer magnetization) [89] may cause the actual R_{MTJ} to vary over time due to the thermal field [see Fig. 4.9(a)] [33, 89]. Hence, the reliability of the proposed STT-MRAM based MemPUF may be degraded if the thermal fluctuation effect affects the bit generation process more than the other intrinsic MTJ variations. Assuming that θ is distributed as in [89], Fig. 4.9(b) plots the complementary reliabilities (represented by BERs) at different temperatures as estimated by the simulation. It can be seen that the BERs increase with increasing ambient temperature and reaches 20% in the worst case.

Automatic Write-Back (AWB) scheme is proposed to exploit the non-volatility of the MTJ to improve the reliability of raw response bits generated from the proposed MemPUF. The basic idea of AWB is that *when a response bit is initially produced from an SB cell, the generated bit is automatically written back into the SB cell upon completion of the read operation to make the bit reproducible thereafter* (see Fig. 4.10). Incorporated with the AWB scheme, the STT-MRAM PUF operates in two phases:

- *Enrollment* phase: The MTJs in a selected SB cell are initialized ($INIT = 1$) into the same state (either AP or P, depending on the value of Data_init). The mismatch of MTJ resistances in the SB cell is then sensed to produce a random bit. When the read operation finishes, the generated bit (Data_wb) is triggered by the completion signal of read operation, hence automatically written back to the SB cell by setting the two MTJs into complementary states.
- *Regeneration* phase: Response bits are directly produced from the SB cells whose MTJs have been set to the complementary states ($INIT = 0$ in this phase).

Fig. 4.10(a) and 4.10(b) show the timing diagrams of a write-back-after-read process in an SB cell with both MTJs initialized to “0” or P. Once the bit is read (= 1 in this case), the MTJ in the left branch is written back with “1” (i.e., switched to AP).

The security of AWB is guaranteed by its “spontaneous” execution upon the completion of read operation in the enrollment phase. Therefore, any attempted malicious writing is prohibited during the write-back process. Note that the proposed AWB scheme may also be used in MemPUFs based on other emerging non-volatile memory (eNVM) technologies. However, most eNVMs have very limited endurance that may limit the total cycles of write-back operations. Table 4.1 lists the possible eNVMs and the relevant properties that are crucial to the MemPUF qualities. The write endurance of MTJs is much higher than other eNVM (e.g., Phase Change Memory and FLASH). Hence, the AWB scheme does not wear out STT-MRAM based MemPUF as fast as MemPUFs based on the other nonvolatile memory technologies. Furthermore, STT-MRAM has several advantages over other types of eNVMs: 1) the density of STT-MRAM is high, which makes it infeasible for tampering

activities such as probing the internal nodes or wires; 2) since the current required for successful MTJ switching is not as large as that used in MRAMs [135], the electro-magnetic emanation may be too weak to be exploited by side-channel attacks. The other eNVMs listed in Table 4.1 may leak measurable information (e.g., electro-magnetic emanation, heat generation, and optical radiation) which can be exploited to speculate the bit that is written back.

Compared to the reliability improvement techniques proposed in the literature [13, 14, 24], the method is superior in several aspects. Firstly, unlike the approach in [24], generation of the response bits in the scheme does not necessarily require prolonging the time delay which may degrade performance. Secondly, the method does not rely on aging effects on the memory devices for reliability enhancement. Aging effects such as negative-bias temperature instability (NBTI) and hot-carrier injection (HCI) may change the properties of the CMOS devices permanently. Thus, the deliberately aged memory-based PUF is unsuitable for use as regular memory as the stability of read/write operations will be vastly deteriorated [62]. On the other hand, the STT-MRAM can still be reused as a regular memory even with the AWB scheme incorporated. Though reading/writing MTJ devices excessively may also result in aging effect such as time-dependent dielectric breakdown, previous studies concluded that an MJT device can normally endure $\sim 10^{15}$ cycles before breaking down [97]. This period of time is sufficient for the STT-MRAM to be used for both reliable PUF and regular memory. When switching from PUF to memory mode of operation, the MTJ states of a MemPUF cell will be erased (i.e., written by 0) before any read/write operations are performed. In this way, the secret bit generated in the PUF mode cannot be reproduced or predicted.

4.3.2 Evaluations on quality criteria

In this section, the proposed STT-MRAM based MemPUF is evaluated in terms of *reliability*, *uniqueness* and *randomness*. The definitions of these metrics were previously introduced in Chapter 3, Section 3.1.2. Due to the inaccessibility to STT-MRAM fabrication, the analysis is based on an MTJ device model and the 45 nm bulk CMOS Predictive Technology Model (PTM) [19]. The MTJ model was developed in Verilog-AMS language and the parameters were calibrated against experimental results in the Object Oriented MicroMagnetic Framework (OOMMF) [26] to precisely mimic both the static and dynamic characteristics of MTJ device [32, 34]. Dependencies of MTJ resistivity on eOV density in MgO layer, temperature and biasing voltage were also taken into account and carefully tuned to align with the reported data [142]. The area and energy consumption of the design were compared with conventional MemPUFs. The parameters used in the simulation are listed in Table 4.2.

Table 4.2 Parameters used in the simulation

Parameters	Nominal values	RSD
Contact area	45 nm×116 nm	5%
Oxide layer thickness	1.15 nm	2%
Effective oxygen vacancy	0.03	2.4%
MOSFET channel length	45 nm	10%
MOSFET threshold voltage	0.466 V	10%
Supply voltage	1 V	-
Write cycle	40 ns	-
Read cycle	20 ns	-

The simulations were performed in HSPICE and the results were processed and analyzed in MATLAB.

Reliability

Based on the definition introduced in Section 4.3.1, three cases need to be considered for evaluating the MemPUF reliability:

1. b_t and $b_{t+\delta t}$ are both generated in the enrollment phase. STT-MRAM MemPUF operating in this case can be regarded as a MemPUF without AWB scheme.
2. b_t is generated in the enrollment phase while $b_{t+\delta t}$ is generated in the subsequent regeneration phase.
3. b_t and $b_{t+\delta t}$ are both generated in the regeneration phase.

In case #1, as introduced earlier in Section 4.3.1, the reliability can be significantly degraded by noise effects under varying environmental conditions [see Fig. 4.9(b)]. In case #2 and case #3, the reliability is higher than that in case #1 because response bits are reproduced from two MTJs (in an SB cell) which are in the complementary states after write-back. However, failure effects, i.e., read/write failures, may affect the reliability of bit regeneration. Write failure occurs when the current injected into the MTJ is not able to switch the MTJ state [35], thus resulting in unsuccessful write-back and the two MTJs in the SB cell will remain in the same state. As a consequence, the reliability becomes as low as that in case #1. Read-disturb failures affect the SB cell reliability in a similar way. During a read operation in the regeneration phase, read-current that is unexpectedly higher than the critical current may switch the state of MTJ. Hence, the two MTJs in the SB cell are back to the identical state. Again, the reliability becomes as low as that in case #1. Sense-decision failure occurs when R_P is larger than R_{AP} of the other MTJ in the SB cell, and therefore,

the response bits reproduced from this kind of cells will be different from expected. The failures for case #2 and case #3 are modeled in Fig. 4.11(a) and Fig. 4.11(b), respectively.

Failure effects in STT-MRAM can be attributed to process variations as well as thermal fluctuation effects [105, 150]. Based on the failure models in Fig. 4.11(a) and 4.11(b), BERs due to these effects can be estimated from the probabilities of relevant failure effects that cause $b_{t+\delta t} \neq b_t$. The Response Surface Modeling technique was used to numerically compute the probabilities given the distribution of the relevant variables (e.g. process parameters, initial angles, etc.) [33, 105]. In the STT-MRAM PUF, the read/write delays are fixed to meet the performance requirements (see Table 4.2). The access transistors can be sized properly to achieve the minimum BER [33]. Fig. 4.12 shows the BERs for case #2 and case #3 with varying access transistor width in an SB cell. Increasing transistor width reduces write failures because the driving capability of the access transistor is enhanced. Also, the sense-decision failure is reduced because the equivalent resistance of the memory cell becomes dominated by the MTJ resistances. On the other hand, read disturbance failure is increased and may become significant when the transistor size is larger than certain value [35]. An optimum width of 1060 nm is found to have the lowest error rate considering both failure effects.

The reliability of an optimally designed STT-MRAM MemPUF may also be deteriorated by supply voltage fluctuations and ambient temperature variations. Fig. 4.13(a) plots the BERs for all three cases when the supply voltage varies by $\pm 10\%$. The BERs for case #2 are high when the supply voltage is low because write-failure is the dominant failure. When the supply voltage ramps up, the BERs drop and eventually converge to the BER for case #3 because read failures become dominating at high supply voltage. Fig. 4.13(b) shows BERs under different temperatures (from 200 K to 400 K) with supply voltage fixed at 1 V. BERs for all three cases increase with temperature because increasing temperature deteriorates thermal fluctuation effects, which in turn leads to increased read-disturb failures. Furthermore, the increase in temperature results in a lower TMR [151] and intensifies sense-decision failures. Although the reliability of the STT-MRAM MemPUF with AWB scheme may be degraded by supply voltage fluctuations and temperature variations, the worst-case BER for all three cases under varying conditions is within the range of $10^{-6} \sim 10^{-5}$, which is far better than that of conventional MemPUFs or the original STT-MRAM MemPUF without AWB. As will be discussed later, the high reliability of raw response bits significantly relaxes the burden of ECC by reducing the redundancies required for error correction.

Uniqueness and randomness

Uniqueness and randomness are crucial to the security of MemPUF and are commonly used for PUF quality evaluation as introduced in Chapter 2 and Chapter 3 [5, 67]. The uniqueness measures *how the generated response bits using different cells vary from each other*. This is estimated by computing the fractional Hamming distance (denoted by H) of m l -bit strings $B = b_1 || b_2 || \dots || b_l$ generated from randomly selected SB cells [78], i.e.,

$$H = \frac{2l}{m \times (m-1)} \sum_{i=1}^{m-1} \sum_{j=i+1}^m \text{HD}(B_i, B_j) \quad (4.3)$$

where the function HD computes the Hamming distance between two binary strings. Fig. 4.14 shows the results calculated from the 64-bit strings generated from the SB cells initialized into P and AP states, respectively. The mean values of H in both cases are close to 50%, which indicates a high uniqueness of the produced bits.

Entropy was used as a metric for evaluating the randomness of MemPUF, which is defined as follows,

$$E = - \sum_{X \in \mathcal{X}} \log(\text{Pr}(X)) \text{Pr}(X) \quad (4.4)$$

where X is a random event from the set $\mathcal{X}$. In this case, X refers to an l -bit string generated using l randomly selected SB cells. Since estimating the entropy of such a large output space using Eq. (5.10) is non-trivial, the Context-Tree Weighting (CTW) algorithm [5, 134] was adopted to compute the compression ratio of the generated bit strings, which is used as an alternative indicator of entropy. Fig. 4.15 shows the estimated entropies when the initial states are P and AP, with mean values equal to 0.99 and 0.98, respectively. These results were achievable due to the symmetric 2T2R structure of SB cells which cancels out the systematic variations of process parameters in the response bit generation process. Therefore, the variations due entirely to the mismatches between devices within an SB cell, offer excellent randomness for the generated bits.

Bit alteration also impacts the uniqueness and randomness of the generated bit strings when different initial states are used. Fig. 4.16(a) shows the bit-maps of a 32×32 SB cell array when the MTJs in each cell are initialized to P and AP states, respectively. Fig. 4.16(b) shows the bits obtained by XOR-ing the two bit-maps in Fig. 4.16(a), and 117 of the 1024 cells are alterable. The proportion of the alterable cells is fairly low due to *self-correlation* effects. Although the variations of t_{MgO} and eOV may result in bit alteration phenomenon, other relevant parameters (e.g., $Area$, V_{th} , etc.) create strong correlations between the cell resistances in the two spin states. For example, if the resistance of MTJ_A is larger than that of MTJ_B in P state mainly due to the difference in their junction areas, the resistance of

Table 4.3 Secret bits generated from one SB cell with different initial states.

Bit pattern ($b_{s=0}, b_{s=1}$)	(0, 0)	(0, 1)	(1, 0)	(1, 1)
Percentage	44.3%	4.9%	5.3%	45.5%

MTJ_A in AP state will be larger than that of MTJ_B in AP state because *Area* impacts MTJ resistances in P and AP states in the same way.

To measure the randomness of an SB cell considering bit alteration, *equivalent entropy* ($\hat{E}$) based on Eq. (4.5) is defined, but assuming that the random event X here refers to the possible bit patterns that can be generated from one SB cell with different initial states, i.e.,

$$\begin{aligned}
\hat{E} = & \\
& - [\log(\mathbf{Pr}(b_{s=0} = 0, b_{s=1} = 0)) \times \mathbf{Pr}(b_{s=0} = 0, b_{s=1} = 0) \\
& + \log(\mathbf{Pr}(b_{s=0} = 0, b_{s=1} = 1)) \times \mathbf{Pr}(b_{s=0} = 0, b_{s=1} = 1) \\
& + \log(\mathbf{Pr}(b_{s=0} = 1, b_{s=1} = 0)) \times \mathbf{Pr}(b_{s=0} = 1, b_{s=1} = 0) \\
& + \log(\mathbf{Pr}(b_{s=0} = 1, b_{s=1} = 1)) \times \mathbf{Pr}(b_{s=0} = 1, b_{s=1} = 1)]
\end{aligned} \tag{4.5}$$

Fig. 4.10 (a) Schematic and state-diagram of the AWB scheme. (b) and (c) show the timing diagram of reading a bit from an SB cell in the enrollment phase. (b) One of the MTJs in the SB cell is switched during the write-back while (c) another MTJ is kept unchanged according to the bit produced from the cell initially.

Fig. 4.11 Failure models of (a) case #2 and (b) case #3. In case #2 and case #3, certain failure effects (framed by red dashed line) cause the bit regenerated at $t + \delta t$ to be different from that generated previously at t .

Fig. 4.12 BERs in the three cases with varying width of access transistor.

Fig. 4.13 BERs for the three cases (initial states of both P and AP were considered for case #2 and case #3) at (a) varying supply voltage (fluctuating within a span of 10%) at 300 K and (b) different ambient temperatures (ranging from 200 K to 400 K) with supply voltage fixed at 1 V. The inner plots show the BERs for case #2 and case #3.

Fig. 4.14 Distribution histograms of the calculated fractional Hamming distances for initial MTJ states of (a) P ($s = 0$) and (b) AP ($s = 1$).

Fig. 4.15 Estimated compression ratio, i.e., upper bound of the entropy, of 50 64-bit strings generated with SB cells initialized in P and AP states, respectively.

Fig. 4.16 (a) The bit-maps of a 32×32 array for initial P (left) and AP (right) states, respectively. Black boxes represent “1”, whereas white boxes represent “0”. (b) Overall 117 (filled with black color) of the 1024 cells are capable of generating alterable bits.

Fig. 4.17 (a) Complementary probability of “one-bit flip” in the generated bit strings using the same cells but different initial states, i.e., $1 - \Pr(\text{num}(b_{s=0} \oplus b_{s=1} = 1) \geq 1)$. The length of the strings was varied from 64 bits to 1024 bits. (b) Percentage of bit patterns $((b_{s=0}, b_{s=1}))$ generated from an SB cell with different initial states before and after using hash function for post-processing.

Table 4.4 Comparisons of the proposed STT-MRAM (with and without AWB scheme) with conventional MemPUFs in terms of Hamming distance, BER, and equivalent entropy

Instances	Fractional H	BER (233 K)	BER (298 K)	BER (358 K)	Equivalent Entropy
		BER (0.9 V)	BER (1.0 V)	BER (1.1 V)	
SRAM	49.7%	$\frac{8\%}{7\%}$	$\frac{6\%}{6\%}$	$\frac{8\%}{7\%}$	0.99
FF	41.8%	$\frac{17\%}{7\%}$	$\frac{4\%}{4\%}$	$\frac{21\%}{7\%}$	1
Latch	36.9%	$\frac{28\%}{-}$	$\frac{3\%}{-}$	$\frac{18\%}{-}$	-
Buskeeper	49.1%	$\frac{11\%}{5\%}$	$\frac{4\%}{4\%}$	$\frac{20\%}{5\%}$	0.99
STT-MRAM, original (AP/P)	49.9%/50.2%	$\frac{5\%}{8\%}$	$\frac{8\%}{8\%}$	$\frac{21\%}{8\%}$	$\approx 1.48 \xrightarrow{\text{hash}} \approx 2$
STT-MRAM, with AWB (AP/P)	49.9%/50.2%	$\frac{4.1 \times 10^{-6}}{6.6 \times 10^{-6}}$	$\frac{4.2 \times 10^{-6}}{4.2 \times 10^{-6}}$	$\frac{4.6 \times 10^{-6}}{4.0 \times 10^{-6}}$	$\approx 1.48 \xrightarrow{\text{hash}} \approx 2$

¹Data of the fractional Hamming distances and the BERs of SRAM, FF, Latch and Buskeeper were from [75, 129]. The data of entropies were from [129].

²Worst-case BERs under varying conditions are considered in the comparison. It should be noted that the PUF instances in comparison [75, 129] were fabricated using 65-nm technology while the proposed STT-MRAM was evaluated using 45 nm technology. Nevertheless, the comparison is valid as the worst-case reliability of these MemPUF instances will remain the same even when scaled CMOS technology is used.

Table 4.3 lists the percentage of the corresponding patterns estimated by simulating 10^6 SB cells. According to Eq. (4.5), $\hat{E}$ is calculated by

$$\begin{aligned}\hat{E} &= -[\log_2(0.443) \times 0.443 + \log_2(0.049) \times 0.049 \\ &\quad + \log_2(0.053) \times 0.053 + \log_2(0.455) \times 0.455] \\ &\approx 1.48\end{aligned}\tag{4.6}$$

which means that there are 1.48 bits contained in each SB cell on average. To meet the requirement of cryptographic application, $\hat{E}$ per cell should be further enhanced (to the ideal value, 2). This enhancement may be difficult to achieve via device engineering because bit alteration depends completely on the physical characteristics of the MTJ (e.g., t_{MgO} and eOV). The most efficient way is to apply post-processing techniques such as hash function. If the applied hash function satisfies the *strict avalanche criterion*² [132] and at least one SB cell among the selected cells is alterable, the bit string post-processed by the hash function will be completely diffused. The probability that at least one bit is complemented in an l -bit string given different initial states of the l SB cells can be calculated as

$$\begin{aligned}\Pr(\text{num}(b_{s=0} \oplus b_{s=1} = 1) \geq 1) \\ = \sum_{i=1}^l \binom{l}{i} \Pr(b_{s=0} \oplus b_{s=1} = 1)^i (1 - \Pr(b_{s=0} \oplus b_{s=1} = 1))^{l-i}\end{aligned}\tag{4.7}$$

The length of the response-bit string was varied to obtain the results shown in Fig. 4.17(a). When the string-length is more than 512 bits, the probability³ that at least one in 512 cells is alterable is very close to 1. Further increase in string-length will converge the probability to 1. Using the post-processing hash function, the bit patterns will have more balanced percentages [see Fig. 4.17(b)]. Hence, $\hat{E}$ contained in each SB cell is improved to

$$\begin{aligned}\hat{E} &= -[\log_2(0.252) \times 0.252 + \log_2(0.249) \times 0.249 \\ &\quad + \log_2(0.248) \times 0.248 + \log_2(0.251) \times 0.251] \\ &\approx 1.99997\end{aligned}\tag{4.8}$$

which is ≈ 2 .

²More specifically, strict avalanche criterion requires that *one bit flip in the input string to the hash function will cause each bit of the output string to flip with an average probability of 50%*.

³For convenience of displaying the calculated values, complementary probability, i.e., $1 - \Pr(\text{num}(b_{s=0} \oplus b_{s=1} = 1) \geq 1)$ is plotted instead.

Fig. 4.18 (a) 1-finger and (b) 2-finger configurations of STT-MRAM cell layout. W and λ refer to the width of the access transistor and the parameter of design rules, respectively.

4.3.3 Comparison

The proposed STT-MRAM based MemPUF was compared to conventional MemPUFs in terms of *Hamming distance*, *BER* (at different supply voltages and ambient temperatures), and *equivalent entropy*. The results are shown in Table 4.4. The proposed STT-MRAM based MemPUF without AWB exhibits randomness, uniqueness and reliability that are comparable to conventional PUFs. The reliability of the STT-MRAM based MemPUF after adopting the AWB scheme is significantly improved ($\sim 10^4\times$). Additionally, 1.48 bits can be produced from each cell of the proposed MemPUF owing to the bit alteration phenomenon, and this value can be improved to ≈ 2 using a hash function to post-process the generated bits.

The instances in terms of energy and area consumption were also compared. Considering the entropy-loss due to the use of ECC⁴, a metric called “cell/bit” is used to measure *the number of cells needed for generating a bit*. Using an ECC which is specified by $[n, k, d]$ [71], cell/bit is equal to n/k . Also, due to the bit alteration effect, the number of raw bits (before being stabilized by ECC and obfuscated by hash function) that can be produced from each cell is equal to the equivalent entropy, $\hat{E}$, of the cell.

BCH-[255, k , t] was used as the ECC in the comparison, and the parameter k was chosen to keep the worst-case BERs of the PUFs below 10^{-6} after stabilization by ECC. Then

⁴Using an ECC will induce an entropy-loss of the originally generated bits, i.e., a number of bits which are equal to the ECC redundancy will be truncated.

area/bit was calculated by

$$area/bit = (area/cell) \times \frac{n}{\hat{E}k} \quad (4.9)$$

and energy/bit was calculated by

$$energy/bit = (energy/cell) \times \frac{n}{\hat{E}k} \quad (4.10)$$

where “area/cell” is estimated based on the λ rule (λ is equal to half of the smallest feature size) for 2-finger SB cell [see Fig. 4.18(a)] layout which consumes less area than 1-finger layout (Fig. 4.18(b)) [44]. For STT-MRAM based MemPUF incorporated with the AWB scheme, “energy/cell” in the enrollment phase is calculated by

$$energy_E = \frac{1}{2} \sum_{s=\{0,1\}} [energy_{(s)}(INIT) + energy_{(s)}(READ) + energy_{(s)}(AWB)] \quad (4.11)$$

and the energy consumed in the regeneration phase is

$$energy_R = energy(READ) \quad (4.12)$$

For STT-MRAM based MemPUF without AWB scheme, only read operation has to be considered.

The calculated results are listed in Table 4.5. Note that the energy consumed by enrollment is relatively high due to the high write current required by an MTJ for state switching, while regenerating bits, i.e., sensing R_{MTJ} , consumes much lower energy ($\approx 2 \times$ lower compared to the latch-based MemPUF, which is meant to be the most energy-efficient). Note also that the frequency of enrollment is related to the security level of the system. Intuitively, the more frequent the enrollment is performed, the more difficult it is for adversaries to predict the response bits. In practical designs, when the energy budget and hardware resources are limited, optimization between energy consumption and system-security should be taken into consideration.

SB cell implemented using STT-MRAM has a smaller footprint compared to the conventional implementations using CMOS technology and hence, the proposed STT-MRAM based MemPUF is the most area-efficient ($\approx 11 \times$ smaller than its SRAM-based counterpart). Furthermore, AWB scheme incorporation reduces the area/bit tremendously by almost 200% (see Table 4.5) due to the high reliability of the raw response bits, which relaxes the ECC requirements to achieve an acceptably high reliability.

Table 4.5 Comparison of the proposed STT-MRAM PUFs with the conventional designs in terms of *energy/bit* and *area/bit*

Instances	energy/bit (pJ)	area/bit (μm^2)
SRAM	9.78×10^{-3}	4.81
FF	4.33×10^{-2}	35.47
Latch	1.48×10^{-2}	20.16
Buskeeper	5.10×10^{-3}	14.43
Proposed, original	0.63×10^{-3}	1.28
Proposed, with AWB	14.31(E), 0.69×10^{-3} (R)	0.43

Energy/cell of the referenced PUF designs were estimated in HSPICE with 45 nm bulk PTM. Area/cell were estimated with the cell layout based on the λ -based design rules.

MemPUF designs based on other types of non-volatile memory technologies [58, 104, 106], which may not specifically tackle secure key storage problem, use different methods to extract the entropies. For example, in [104], a Public PUF (PPUF) is proposed for secure time-bounded authentication. The principle is similar to the classical public-key cryptography except that it relies on the complexity discrepancy between using a hardware simulator and physical device to evaluate the same amount of response bits. It is difficult to carry out a fair comparison with these designs since their reliabilities are not reported. The proposed work nonetheless offers comparable ($\sim 1\%$ differences) response randomness (measured in uniformity) and uniqueness of with the aforementioned PUFs. As long as entropy, or equivalent entropy as defined and used is concerned, the proposed STT-MRAM PUF has an added advantage since the other instances produce only one bit from an individual cell.

4.4 Summary

In this chapter, the research work on improving the quality of MemPUF was reviewed. A novel design based on STT-MRAM with high reliability and multi-bit capacity per cell was introduced. The novelty of this work is that the reliability of the proposed STT-MRAM is remarkably enhanced by 10^6 times without architectural support or heavy circuit-level modification. This implies that the overhead of using ECC to reconcile noisy bits can be substantially saved. The automatic write-back scheme was integrated directly in the write operation of STT-MRAM, and the only ancillary circuit component is the write-back control logic which is essentially lightweight and easy-to-implement. The multi-bit capacity per cell is realized based on the bit alteration phenomenon in MTJ devices, that is, by initializing

an MTJ in different resistance states, the response bits produced from the MTJ will be inherently different owing to the randomness of TMR of the MTJ device. The simulation results demonstrated the efficiency of the proposed approach.

In the next chapter, another topic on MemPUF using eNVM technology is introduced, which has successfully overcome a critical attack on SRAM-based PUF that was proven successful in a recent report. In order to reuse the same eNVM MemPUF chip for dual-mode applications, a generic a design methodology to resolve the conflict between yield loss and PUF quality will be proposed. it can be applied on all types of eNVM-based MemPUF, and demonstrated specifically by an STT-MRAM based design.

Chapter 5

Dual-Mode Application of MemPUF

Intrinsically, most MemPUF instances rely on the inherent process variability of memory cell or memory-like structure element to produce device-specific random bit, and the extraction process is independent of normal operations of the memory or memory-like logic. Reusing the same chip or hardware module for two distinct functionalities saves hardware resources, which is especially appealing to the lightweight application in today's embedded systems. However, process variations may deteriorate memory yield although it aid to generate uniqueness and uncertainty of response when the memory is used as a PUF. Due to the conflicting effect of process variations on memory yield and PUF qualities, , novel design method is needed to make both modes of operation practical. The solution requires strategic optimization taking into consideration the target memory structure, device characteristics, circuit implementation constraints and long-term mutual parametric influence of the two supposedly independent working modes.

In this chapter, an optimal design methodology for dual-mode application based on MemPUF is proposed. The method is generically applicable to all MemPUF designs. By considering the different impact of process variations on the qualities of both normal data storage application and PUF application, the sizing of device geometries is optimized to achieve the best quality criteria for both working modes. The feasibility of the proposed method was demonstrated using two case studies on SRAM based MemPUF and STT-MRAM based MemPUF. Specifically, it will be proven by STT-MRAM based design that the buffer-free attribute offers resilience against certain type of fault-injection attack which has been demonstrated successful on conventional SRAM-based MemPUFs. The following contributions will be described in this chapter.

- A Buffer-Free Memory-based PUF (BF-MemPUF) is originally proposed to utilize eNVM for the generation of secret bits without disturbing the data already present in

the memory. The data stored in the memory do not need to be transferred to a buffer before the response bit generation, thus mitigating the overhead of additional data buffer memory and the threat of data remanence decay based fault-injection attacks.

- The implementation steps of a BF-MemPUF is shown by using eNVM technologies:
 - Circuit- and architecture- level design of a BF-MemPUF for dual-mode applications are introduced.
 - A general design optimization method is proposed to maximize both the memory yield and PUF qualities of the BF-MemPUF under the constraints determined by the high-level requirements.
 - The effectiveness of the method is demonstrated by a BF-MemPUF design based on Spin-Transfer Torque MRAM (STT-MRAM).

5.1 Reusing Memory for Dual-Mode Applications

Reusability is one of the most appealing features of MemPUF. In a nutshell, reusability refers to the capability of using the same piece of memory chip to serve dual purposes, namely data storage and PUF application. Since most of the existing MemPUF instances leverage on the intrinsic process variabilities of memory or memory-like structure devices to produce device-specific keys, they can also be used for normal data storage application. For example, in the PUF working mode of an SRAM-based PUF, random bits can be generated by powering up the entire array and reading the initial state of the memory cell selected by memory address. In the normal memory working mode, read and write operation can be executed on the same cells again once the previous state of the cell for generating random bits is erased. To design a MemPUF featuring two working modes, one needs to carefully consider the impacts of process variations on normal memory operation and PUF operation, as well as the intrinsic conflicts between the two modes of applications.

5.1.1 Impact of process variations on memory operation and PUF qualities

For most MemPUF designs, the response bits rely on process variations of memory cell devices. In SRAM MemPUFs and those belonging to the same category, such as latch-based PUF, flip-flop PUF and buskeeper PUF, the randomness specifically refers to the mismatch of threshold voltage fluctuations and transistor geometric variations. Threshold voltage variations may be due to random dopant fluctuation effects in CMOS transistor active region and

systematic variations in manufacturing process. For other types of MemPUFs e.g. those which are based on NVM technologies, the process variation sources may be some other physical parameter fluctuations of the storage element. While process variations are favored by PUF-mode applications, they are not welcomed by normal memory operations. Owing to the random distribution of physical parameters of memory devices, normal memory operations such as write and read may occasionally fail as a consequence of inadequate charging time to flip the state of a memory cell or accidental flipping of the previously preserved bit on a memory cell.

For instance, an SRAM based MemPUF is essentially a SRAM which can work in alternative modes of memory and PUF. In a good SRAM-based MemPUF, 1) the functionality of one application mode does not affect the other; 2) the qualities of both application modes should meet the system-level specifications; 3) no hardware design modifications are needed for the dual-mode applications. In the PUF mode, a set of memory addresses (i.e., the *challenge* denoted as C) is used to randomly interrogate a subset of SRAM cells to obtain a binary string (i.e., the *response* denoted as R). The response is usually post-processed by a *fuzzy extractor* to produce the final cryptographic key. The response randomness of the SRAM-based MemPUF is ascribed to race condition of the power-up process. During power-up, as the supply voltage ramps up, the two cross-coupled inverters will race to reach their stable and discriminative output values. The final state of the cell is determined by the mismatch of the two inverters.

SRAM-based PUF is featured by its superior randomness and uniqueness when compared to other PUF instances. This is because the response bits are generated from individual cells with mutually independent stochastic parametric variations. However, the mismatch between the inverters of a SRAM cell is not sufficiently large, which may be easily overwhelmed by noise effects. Therefore, reliability is always the utmost concern in SRAM-based PUF design. The reliability is defined as *the probability that the same bit can be generated from the cell at different time instances*. Apparently, the reliability of the SRAM-based PUF can be affected by circuit noises. Besides, the environmental conditions in which the devices operate will also impact the reliability.

As a result of the correlation of the transistor size to the random dopant density of MOSFET transistors, the intra-die variation of threshold voltage can be related to transistor geometry by the following formula [96]:

$$\sigma V_{th} = \frac{\Lambda \cdot \sigma V_{th, \max}}{\sqrt{(W \cdot L)}} \quad (5.1)$$

where Λ is a technology-dependent parameter and $\sigma V_{th,max}$ is the maximum value of σV_{th} . Considering this effect, sizing up the transistors will make the mismatch of V_{th} between the two inverters smaller, and consequently lead to lowered reliability of generated response bits.

In the memory mode, the primary consideration is the failure probability. In this work, two main mechanisms of failures, i.e., read failure (RF) and write failure (WF) are considered.

- RF occurs when the voltage read at node BIT (or BITB) rises above the trip point of the inverter to cause the data preserved in the SRAM cell to flip accidentally. When the bitline (BL) is precharged, the access transistor (M6) and the pull-down transistor (M4) act as a voltage divider, and the voltage at node BITB may trigger a flip of the data bit in the SRAM cell.
- WF occurs when the inverters of the SRAM cell fail to respond to the data bit within an acceptable time when a bit is written into an SRAM cell. Here, it is assumed that bit 0 will be written into the cell which preserves data 1, that is, the voltage at node BIT will be discharged to a low value. If this value is not smaller than the trip point of the inverter INV-2, the write operation will fail and the data bit stored in the SRAM remains unchanged.

The impact of process variations (e.g., V_{th}) will cause certain SRAM cell to fail in operation occasionally. The overall failure probability [$\Pr(RF,WF)$] of an SRAM cell can be estimated by

$$\Pr(RF,WF) = \Pr(RF) + \Pr(WF) - \Pr(RF \cdot WF) \quad (5.2)$$

It is possible to reduce this failure probability by manipulating the SRAM cell transistor sizes because of the relationship between $\Pr(RF,WF)$ and the sizing of each transistor¹ in the SRAM cell.

To design an SRAM-based PUF with dual application modes, the size of the SRAM cell needs to be optimized so that the quality criteria and specifications can be met for both applications. For memory mode, the failure probability should be kept as low as possible. For PUF mode, the randomness and reliability of the generated bits should be made as high as possible. The research focuses on the reliability since it is a more critical metric for SRAM-based PUF compared to other types of PUF as mentioned earlier. Besides the requirement of failure probability and PUF reliability, area and leakage constraints of the targeted process technology will also be considered.

¹The readers may refer to [90] to find out more about how geometric parameters of transistors affect the failure probabilities.

According to the above analysis, the objective is to acquire an optimal design point for the following problem.

$$\min_{\mathbf{X}} \Pr(\text{RF}, \text{WF}) = f(\mathbf{X})$$

where $\mathbf{X}$ is the SRAM cell parameters (e.g., W). The objective function is subjected to the following constraints:

$$\begin{cases} \text{reliability}(\text{ECC}[n, k, t]) \geq \text{reliability}_{\min} \\ \text{area_cell} \leq \text{area_cell}_{\max} \\ \text{leakage_cell} \leq \text{leakage_cell}_{\max} \end{cases} \quad (5.3)$$

where $\text{ECC}[n, k, t]$ is the Error Correction Code specified by code parameters n , k and t .

5.1.2 Security concerns on MemPUF for dual-mode applications

Another critical concern on reusing a memory for dual-mode application is that the two working modes should not bring about any negative impact on each other. For example, in an SRAM-based MemPUF, each time the SRAM is used for normal memory operations, the response bit preserved in the memory cells should be erased in entirety. However, this has been proven to be intricate in practice. Data remanence becomes a dreadful threat to SRAM-based MemPUF since the adversaries may make use of the residual information to recover the response bit previously generated from the cell. This “fault-injection” attack will be discussed in detail in the next section. It will also be theoretically and experimentally evaluated using the proposed design based on STT-MRAM technology to elucidate the security problem.

5.2 Design Method for Dual-Mode BF-MemPUF

The key generation process in conventional memory-based PUFs inevitably overwrites data which are already stored in the memory. Therefore, buffer storage is needed to temporarily hold the data that are presently stored in the memory until the completion of key generation process, as illustrated in Fig. 5.1. On one hand, this process introduces undesired area/performance/energy overhead. On the other hand, the data buffer operations may create potential security vulnerability. For example, data remanence decay of SRAM-based MemPUF may be exploited to reveal the response bits [94]. Since powering off the volatile SRAM will not cause immediate data loss, overwriting the SRAM-based PUF with known

Fig. 5.1 Working mechanism of a conventional memory-based PUF system (illustrated with an SRAM-based PUF). C , R , H , and K are the challenge (normally the memory address), response, helper data and output key of the system. The figure at the bottom shows what happens inside a SRAM cell when it is switched from memory mode to PUF mode (write back operation after response bit generation is not shown).

pattern after it has been used for key generation can result in a successful fault-injection attack to recover the response bits that have been generated. There are several countermeasures suggested in [94] to deal with the problem, but these methods either introduce non-negligible hardware implementation overhead or require complicated cryptographic primitives.

Since most existing memory-based PUFs are constructed based on volatile memory, one critical issue is that *data already stored in the memory needs to be moved to a buffer storage before the secret bits are generated and then written back when the key generation process has completed* (see Fig 5.1). This additional data buffering and writing back operations not only incur unnecessary overheads in chip area, performance and power consumption, but also provide opportunities for adversaries to perform fault-injection attack to reveal the response bits. The attack, as introduced in [94], makes use of the data remanence decay of data bit in an SRAM cell, to recover the information of response bits generated from the cell. The process of this attack is illustrated in Fig. 5.2.

Even if the security issues may be resolved by applying certain countermeasures (e.g., prolonging time interval between a PUF operation and successive memory operations, post-

Fig. 5.2 Procedure of performing a fault-injection attack on SRAM based PUF utilizing data remanence of SRAM cell. t_i refers to the decay time, and Δt is the incremental value for t_i after each iteration. Initially, the SRAM is given a known state M_{t_0} . The attack tries to recover the intrinsic state of SRAM based PUF (in **Phase-II**) with the collected responses $R_1, R_2, \dots, R_f$ iteratively generated from the PUF with tuned decay time (in **Phase-II**). A more detailed description of the attack can be found in [94].

processing responses with cryptographic algorithms [94], or memory obfuscating techniques [45]) such that the resilience of memory-based PUF against certain invasive/semi-invasive attacks can be enhanced, it is imperative to ensure that the memory cells that are designed consistently can still meet the different performance criteria in both modes of operation upon fabrication. Process variations, which are desirable for PUF application, may result in read/write failures in certain memory cells. In the design phase of a memory-based PUF, new optimization strategy is needed to adequately satisfy the quality requirements for both modes of operation.

To circumvent the above-mentioned problem, a novel memory-based PUF primitive, called BF-MemPUF, that produces random keys without disturbing the data originally stored in the memory, is proposed. BF-MemPUF may be implemented more efficiently using emerging non-volatile memories (eNVMs)². eNVM technology (e.g., phase change memory, spin-transfer torque memory, resistive memory, etc.) is equipped with desirable characteristics of universal memory such as non-volatility, high integration density, and good

²Resistive eNVM technologies is targeted in this dissertation. Otherwise, the proposed BF-MemPUF and design method can also be applied in NVMs using conventional technologies such as FLASH.

Fig. 5.3 (a) Distribution of read-out current for an eNVM cell in HRS and LRS states, respectively. Setting reference currents to different values may result in different output bits from the cell; (b) working mechanism of BF-MemPUF.

scalability [135–137]. In contrast to conventional memories, which preserve bit information as charge, which is volatile when the power is turned off, eNVMs store data by exploiting the resistivity state of the Storage Device (SD). That is, it represents logic “1” if the SD is in High Resistance State (HRS), and represents logic “0” if it is in Low Resistance State (LRS). To read the data bit from a cell, a voltage is applied across the data cell and the current flowing through it (I_{dat}) is compared with a reference $I_{ref} = (I_H + I_L)/2$, where I_H and I_L are the current of HRS and LRS states, respectively. The polarity of the comparison is output as a digital bit. Normally I_H and I_L are also generated from a referencing eNVM SD that is designed identically with those in the memory array.

Due to process variations, certain parameters of eNVM SDs are stochastically distributed, which lead to the random distribution of SD resistances and, in turn, the read-out current I_{dat} . Table 5.1 lists the SD structures of three representatives of eNVMs as well as their physical process variations contributors that are commonly considered in the literature [33, 43, 67, 149]. Consider a data cell with an SD storing a data bit, which can be 0 or 1. By setting the reference current (denoted as I_{ref}) to be the same as that flowing through the accessed data cell (denoted as I_{dat}), the outcome of the comparison between I_{ref} and I_{dat} will be random due to process variations [see Fig. 5.3(a)]. This stochastic feature allows us to extract a random bit from an eNVM cell without disturbing the data originally stored in it. The basic mechanism of eNVM based BF-MemPUF is shown in Fig. 5.3(b).

Table 5.1 SD structure and process variation sources of STT-MRAM, PCM, and RRAM

eNVM	STT-MRAM [49]	PCM [52]	RRAM [41]
SD Structure
Variations	Contact area, thickness of the MgO layer Threshold voltage and effective channel length of NMOS transistor which is used as the access device	Contact area, thickness of the heater and the GeSbTe layer	Contact area, thickness of the HfO ₂ layer

It should be noted that since the extraction of random bits relies on the physical random properties of the eNVM cells, its randomness is uncorrelated with the state of eNVM SD. Even if the same cell has been written multiple times with different resistance states, it should still produce consistent response bit in the PUF mode except in some rare conditions when the bit may be accidentally flipped by the noise effects. This is because the variation sources such as contact area and thickness of certain layer (as delineated in Table 5.1) are constant and have strong correlation to the distribution of SD resistances. Based on the configurations listed in Table 5.1 for each eNVM technology, a large number of eNVM cells were simulated by programming them into HRS and LRS before the response bits were collected in PUF mode. The variations in the minimum feature size of device fabrication technology, the threshold voltage of access transistor and the oxide layer thickness are assumed to be 5%, 10% and 2%, respectively. The results indicated that the probability of an eNVM cell producing the same response bit in two alternative states is 100%. Nevertheless, small random noise may be introduced during the state switching process for certain eNVM material, and gain sufficient energy to alter the response bit generated from the same cell in different states under some rare conditions (e.g., at a temperature much higher than the typical range of operation [43, 48, 97, 140]). Similar to any other type of temporary noise effects, the error induced by random fluctuations in programming process, device aging effect and endurance loss in a long-term use can be eliminated by algorithmic error correction method.

5.2.1 Resilience against data remanence decay attack

As compared to conventional memory based PUF, the eNVM based BF-MemPUF preserves data bits in non-volatile manner, and it does not rely on the power-up process to extract the intrinsic mismatch between two memory components. Instead, random bit generation is realized by setting the reference cell to a state identical to that of the accessed data cell. An adversary may not be able to perform a successful attack as shown in Fig. 5.2 on the eNVM based PUF because

1. switching on and off the power supply does not affect the state as well as the intrinsic randomness of an eNVM cell if the cell is properly biased. If the biasing voltage is excessively large, *read disturbance* may occur and accidentally switch the state of the cell;
2. regulating the time of power-on or power-off process has no effect on the data retention of eNVM cell because the data preserved in it is non-volatile;
3. reading a random bit from an eVM cell is not only independent of the already preserved data bit, but also the successive read/write operations on the eNVM cell as long as the response bit from the same cell can be reliably reproduced regardless of its resistance state.

Given these specifications, the adversary cannot succeed in the attack modeled in Fig. 5.2 by analyzing the start-up state of an eNVM based PUF, thus will not be able to recover the response bits as desired. Moreover, it is also non-trivial to take advantage of any other form of side-channel information to attack the PUF. For example, differential power analysis may not be applicable because eNVM consumes no leakage power owing to its non-volatility [28, 55, 57]. Physical attacks such as direct probing or invasive techniques will not be successful since the eNVMs are often fabricated using nano-scale technology such that physically accessing the internal devices will permanently damage them [135, 137].

5.2.2 Design considerations

Using memory for PUF applications require specific circuit-level design strategies to meet the specifications of PUF system. For example, devices may be sized small to deliberately aggravate the impacts of process variations for reliable and random bit generation. However, this will in turn increase failure probabilities in read/write operations, which deteriorate the ordinary memory applications. In this section, a general method to optimally design an eNVM-based PUF that satisfies the criteria of both operations will be introduced.

Table 5.2 Parameters considered for designing BF-MemPUF

Parameter	Description
E (electrical para.)	supply, word-line voltage, read/write cycle
P (physical para.)	parameters of transistors and SD devices
T (environmental para.)	ambient temperature
D (design para.)	device geometries, array arrangement

Prior to a detailed discussion on the design of BF-MemPUF, some metrics that are commonly used to evaluate the quality of memory-based PUFs are introduced. To make it convenient for quantitative evaluation, it is assumed that these metrics are quantified by the responses of the memory circuitry under certain input parameters. These parameters can be classified into the following four groups: electrical parameters (**E**), physical parameters (**P**), environmental parameters (**T**), and design parameters (**D**) (see Table 5.2).

Memory metrics

Memory yield may be significantly degraded by process variations. The probabilities of failure effects are often considered to evaluate memory yield [68, 90]. Three types of failure effects in eNVMs are considered:

1. Write-Failure (Φ_{f-WF}): Magnitude of the electrical pulse³ to write the cell is lower than the threshold, i.e., $I_{dat} < I_{crit}$;
2. Read-Failure (Φ_{f-RF}): The sensed current from the cell does not correspond to its resistivity state, i.e., $I_{dat} > I_{ref}$ when SD is in HR state, or $I_{dat} < I_{ref}$ when SD is in LR state;
3. Read-Disturbance (Φ_{f-RD}): The read current is larger than the threshold to switch the state of the SD due to process variations, i.e., $I_{dat} > I_{crit}$

The failure probabilities may be computed directly using the joint Probability Density Function (PDF) of the relevant parameters, which are assumed to be randomly distributed variables, i.e.,

$$\Phi = \int_{-\infty}^{+\infty} F(\mathbf{x})P(\mathbf{x})d\mathbf{x} \quad (5.4)$$

³Though some of the eNVM designs are featured by voltage-driven programming method, it is assumed that switching of the state of eNVM SD is triggered by current pulses without loss of generality. The voltage pulse can also be treated as the responses of memory circuitry if voltage-driven programming is applied.

where $F(\mathbf{x})$ is the criterion function which is formulated as

$$F(\mathbf{x}) = \begin{cases} 0, & \text{fail;} \\ 1, & \text{pass.} \end{cases} \quad (5.5)$$

and $P(\mathbf{x})$ is the probability density function

$$P(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^k |\boldsymbol{\sigma}|}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right) \quad (5.6)$$

$\mathbf{x} \sim \mathcal{N}_k(\boldsymbol{\mu}, \boldsymbol{\sigma})$ is the parameter vector that consists of k parameters in $\mathbf{P}$ and/or $\mathbf{D}$ which are considered as normally distributed variables. Without loss of generality, the parameters in $\mathbf{x}$ are assumed to be mutually independent.

Hence, the overall failure probability (Φ_f) can be computed as [68]

$$\Phi_f = \Phi_{f-WF} \cup \Phi_{f-RF} \cup \Phi_{f-RD} \quad (5.7)$$

Also, since there are two distinct resistance states in the SD, the worst-case failure probabilities, Φ_{f-RD} and Φ_{f-RF} are of more concern. They are computed as

$$\begin{aligned} \Phi_{f-RD} &= \max(\Phi_{f-RD}^H, \Phi_{f-RD}^L) \\ \Phi_{f-RF} &= \max(\Phi_{f-RF}^H, \Phi_{f-RF}^L) \end{aligned} \quad (5.8)$$

Since these failure probabilities are correlated with design parameters $\mathbf{D}$, by specifying the metrics of memory applications, the parameters in $\mathbf{P}$ can be selected appropriately to minimize the failure probabilities.

PUF metrics

The quality of memory-based PUF is often quantified in terms of *reliability* and *randomness*⁴ [5]. Reliability measures the repeatability of response bit generation in the presence of noise effects under varying operating conditions. In eNVM PUFs, the reliability may be measured as the amount of noise that can be tolerated when sensing the difference between I_{ref} and I_{dat} . That is, if $|I_{ref} - I_{dat}| < \varepsilon$, where ε is the noise current, the random bit generated from the cell may be *unstable*, otherwise, it is *stable*. Therefore, the complementary reliability, represented as Φ_u hereafter in this chapter, i.e., the probability that an unstable

⁴For simplicity, the less significant systematic variations and spatial correlations of intra-die variations are not considered. Therefore, the *uniqueness* of within-die response bits and die-to-die response bits, which are always better than the inter-die case, are not reported in this work.

Fig. 5.4 Distributions of sensed current on a LRS eNVM cell (a) before and (b) after the variances of randomly distributed variables have been increased.

bit is generated, can be estimated as

$$\Phi_u = \Pr(|I_{ref} - I_{dat}| < \epsilon) \quad (5.9)$$

Similarly, Φ_u can be directly calculated by using Eq. (5.4) with a suitable criterion function. Just as in the cases of Φ_{f-RD} and Φ_{f-RF} [see Eq. (5.8)], the maximum values of Φ_u in HRS and LRS are considered for the worst-case scenario.

The randomness of a PUF is measured by the entropy contained in each cell, which indicates how much information is carried in a response bit generated from a memory cell. Since the response output generated from one memory cell is at most 1 bit, the entropy of a cell, termed as Φ_e in this chapter, can therefore be computed as

$$\begin{aligned} \Phi_e &= - \sum_{r \in \mathcal{R}} \log[\Pr(r)] \Pr(r) \\ &= -\log[\Pr(r=0)] \times \Pr(r=0) \\ &\quad -\log[\Pr(r=1)] \times \Pr(r=1) \\ &= -\log[\Pr(I_{dat} > I_{ref})] \times \Pr(I_{dat} > I_{ref}) \\ &\quad -\log[\Pr(I_{dat} < I_{ref})] \times \Pr(I_{dat} < I_{ref}) \end{aligned} \quad (5.10)$$

To obtain Φ_e , $\Pr(I_{dat} > I_{ref})$ is first computed using Eq. (5.4). Φ_e is then calculated for each state using Eq. (5.10). Assuming that the SD has equal probability of being in HRS or LRS, the final Φ_e is computed as

$$\Phi_e = \frac{\Phi_e^H + \Phi_e^L}{2} \quad (5.11)$$

Fig. 5.5 Adding redundant reference arrays average out the effect of skewness introduced by the reference cells which are shared by each row of data cells.

5.2.3 Design space exploration

Device geometry scaling

Scaling down the size of SDs makes them more susceptible to process variations. As a result, Φ_f will be degraded if the variations of the parameters become larger. This is because the probabilities of read/write failures that may occur when process parameters $\mathbf{P}$ are located far from nominal values will become larger. However, $1 - \Phi_u$ will be increased when the deviation of certain parameters become larger. As a consequence, the tolerable noise, when reading response bit from a cell, $|I_{ref} - I_{dat}|$, is increased. The generated random bits will then become more reliable. Fig. 5.4 illustrates the changes of Φ_{f-RD} and Φ_u before and after the device parameters have been scaled down.

Sensing scheme

The conventional sensing scheme used for the read operation of SRAM-based PUF and other conventional memory-based PUFs is self-referenced and differential by nature. However, the sensing scheme employed in most eNVMs is inherently single-ended. There are two main referencing approaches to read the state of an eNVM cell. The first approach generates reference current with a carefully biased current source. The current source can be designed

- 1: Given $\mathbf{P}$, $\mathbf{E}$ and $\mathbf{T} = \mathbf{T}[0]$, where $\mathbf{T}[0]$ is the parameter in nominal condition, specify a fuzzy extractor: $\text{ECC}(n, k, t)$ and $\text{Hash}(\cdot)$, and set $\text{FIND} = 0$;
- 2: **for** $\mathbf{T}[j] = \mathbf{T}[1] \rightarrow \mathbf{T}[n]$, where $\mathbf{T}[1] = \mathbf{T}_{\min}$ and $\mathbf{T}[n] = \mathbf{T}_{\max}$ **do**
- 3: **for** $\log_2(\alpha)$ from $1 \rightarrow N$ **do**
- 4: Determine $F_{\max}$ [using Eq. (5.13)] by $F \leq \frac{\text{Area}_{\max}}{M(N + 2\alpha)}$;
- 5: Find $\mathbf{D}_o$ satisfying objective function $\min_{\mathbf{D}}(\Phi_f)$ subject to

$$\begin{cases} F \leq F_{\max} \\ 1 - \sum_{i=0}^t \binom{n}{i} \Phi_u^i (1 - \Phi_u)^{n-i} \leq \text{BER}_{\max} \\ \Phi_e \geq E_{\min} \end{cases} \quad (5.12)$$

- 6: **end for**
- 7: **for** $\mathbf{T}[j] = \mathbf{T}[1] \rightarrow \mathbf{T}[n]$ **do**
- 8: Compute $\Phi_u[j]$ with $\mathbf{D}_o$ and $\mathbf{T}[j]$;
- 9: **if** $\text{BER}(\Phi_u[j]) \geq \text{BER}_{\max}$ **then**
- 10: Stop the iteration;
- 11: **else**
- 12: $\mathbf{D}_{\text{opt}} = \mathbf{D}_o$;
- 13: Set $\text{FIND} = 1$;
- 14: **end if**
- 15: **end for**
- 16: **if** $\text{FIND} = 1$ **then**
- 17: Stop and output $\mathbf{D}_{\text{opt}}$ as the optimal design parameters;
- 18: **else**
- 19: Continue the optimization process;
- 20: **end if**
- 21: **end for**

Fig. 5.6 Algorithm to design an optimal BF-MemPUF.

using bandgap reference to eliminate its dependency on temperature variations. In different modes of read operation, the current source is configured by a control signal to generate a proper reference current. Another approach is to use a reference cell which is identically sized as the data cells in the array. In different modes of operation, the state of the reference cell is switched accordingly to read a data cell.

Typically, the current flowing through the selected eNVM cell is compared to that from a reference that is shared by all the data cells in the same row during sensing. Consequently, since the reference cell is also affected by process variations, the random bits generated from this row of data cells will be correlated due to the skew of the reference cell (see top of Fig. 5.5). As a result of the aforementioned biasing effects due to shared sensing circuit by a row of memory cells, the reliability as well as randomness of the generated random

bits may be affected [67]. Splitting a row of cells and inserting redundant reference arrays can reduce the correlation because the effects of the skew introduced by the reference cells will be averaged out. Intuitively, when the number of data cells in a row sharing the same reference cell is reduced to one, the sensing scheme becomes differential as in an SRAM cell. However, the area of the entire array will also be increased if redundant reference cells are added. Assuming the *splitting factor* (denoted as α) is equal to $\alpha = 2^i$ ($1 \leq i \leq \log_2(N)$), the area of BF-MemPUF array can be computed as

$$\text{Area} = M(N \cdot F + \alpha \cdot 2F) \quad (5.13)$$

where F is the footprint of an eNVM cell, and $2F$ is the footprint of a reference cell (assuming a reference cell comprises two cells storing complementary data).

In some eNVM technologies, the resistance of SD in a cell may drift after many cycles of programming operations [48, 97, 139, 141]. Since a reference cell is accessed and programmed more frequently relative to the data cells, resistance drift may bias the value of reference cell and alter the response bit. This problem can be avoided if the reference cell is replaced by a current source for sensing. The technique of reducing the supply voltage of frequently accessed cells can also be applied to reduce the stress on reference cells. For certain technologies, reversing current flow in read operations can recover the aged cells [48]. Other wear leveling technique [103] can also be adapted for the scheme. Instead of a single reference cell, multiple references may be used. In read operations, one of the reference cells is randomly selected for sensing to average out the stress on all of the reference cells. The aging effects may not be fully avoided but the process can be slowed down to sustain the response quality in PUF mode for the life time of its security application.

Optimal design method

Due to the conflicting design requirements for memory yield, PUF qualities and array area cost, the design space (i.e. sizing of the devices and arrangement of the array) needs to be explored to find an optimal design point that satisfies all design specifications. For example, Φ_f should be minimized under the constraint of the array area ($\text{Area}_{\max}$). Additionally, in the PUF-mode, Φ_u of the generated response bits in the worst-case should guarantee that the BER of the response bits after post-processing using the fuzzy extractor with ECC

Fig. 5.7 (a) Circuitry of STT-MRAM array and cross-sectional diagram of STT-MRAM cell; (b) R-V characteristics of MTJ device, and resistivity switching dynamics between AP and P states of an MTJ.

incorporated is low enough⁵. The BER can be calculated using

$$\text{BER} = 1 - \sum_{i=0}^t \binom{n}{t} \Phi_u^i (1 - \Phi_u)^{n-i} \leq \text{BER}_{\max} \quad (5.14)$$

where n and t are the parameters of the ECC- $[n, k, t]$, where n , k and t are the code wordlength, information wordlength and number of correctable error bits, respectively. Also, the entropy extractable from a cell needs to be above a threshold $E_{\min}$ determined by the system requirement.

In general, the problem of BF-MemPUF design is to find the design parameters, $\mathbf{D}$, such that Φ_f , Φ_u and Φ_e are optimized to meet the constraints on array area and ECC tolerance. Based on the metrics introduced previously, the proposed optimization methodology is shown in Fig. 5.6. The method searches for the most optimal design point under varying temperature conditions. Since the method does not specify the parameters for $\mathbf{P}$, $\mathbf{E}$ and $\mathbf{D}$, it can be generally used for all eNVMS designed as a BF-MemPUF.

⁵The industrial standard for the reliability of a 128-bit response is $\leq 10^{-6}$. Therefore, the BER should be kept $\leq 10^{-9}$.

5.3 Case Study: Dual-Mode BF-MemPUF Using STT-MRAM

In this section, a BF-MemPUF design based on STT-MRAM is used to illustrate the optimization methodology.

5.3.1 Fundamentals of STT-MRAM

STT-MRAM stores data bits based on the spin-transfer torque and giant magnetoresistance (GMR) phenomena in certain material. The SD device in STT-MRAM is the Magneto-Tunneling Junction (MTJ), which consists of two ferromagnetic layers sandwiching a thin tunneling oxide barrier. One ferromagnetic layer is magnetically pinned (which is called the pinned layer, PL), while the other is free to be manipulated (which is called the free layer, FL). STT-MRAM array and cell structure are shown in Fig. 5.7(a). The MTJ state is stored as the relative magnetization of the FL to that of the PL, which can be sensed as the MTJ resistance, R_{MTJ} . When the magnetization of the FL is the same as that of the PL, the MTJ is in the *parallel* state (or P) and $R_{MTJ} = R_P$ (LRS). Instead, when the magnetization of the FL and the PL are opposite (known as *anti-parallel*, or AP state), then $R_{MTJ} = R_{AP}$ (HRS). Hence, different logic bits can be obtained by sensing R_{MTJ} (“0”: $R_{MTJ} = R_P$; “1”: $R_{MTJ} = R_{AP}$). The distinguishability between R_P and R_{AP} is defined as the Tunnel Magnetoresistance Ratio (*TMR*), and is given by

$$TMR = \frac{R_{AP} - R_P}{R_P} \quad (5.15)$$

The MTJ may be programmed by passing a current through it. If this current is larger than a critical value, I_{crit} , within certain period, the direction of magnetization of the FL will be switched [see Fig. 5.7(b)].

Due to the uncertainties in the manufacturing process, some parameters of MTJ may vary randomly. The variations in the MTJ are those in the oxide thickness (t_{ox}) and the bottom contact area (*area*). In addition, parameters of the access transistors (e.g., the threshold voltage, V_{th} , and the effective channel length, L_{eff}) are also assumed to be normally distributed. The variations of these parameters affect read/write failure probabilities of STT-MRAM, and the qualities of STT-MRAM based PUF.

The reliability of STT-MRAM may be deteriorated by the time-dependent dielectric breakdown in oxide layer [97, 139]. It was pointed out in [97] that STT-MRAM has an endurance of 10 years before breaking down if the oxide layer thickness is properly designed. In the case, the thickness of the oxide layer is 1.15 nm. From Fig. 5.8, it has a lifetime of $7 \sim 8$ years.

Fig. 5.8 Increasing the oxide layer thickness results in a prolonged lifetime of MTJ device with the same biasing condition. In this configuration, the thickness of 1.15 nm gives the MTJ device a life time of around 7 ~ 8 years.

5.3.2 Simulation and discussion

Based on the principle depicted in Fig. 5.3(b), the STT-MRAM based BF-MemPUF is designed. The architecture and read circuit diagram [76] is shown in Fig. 5.9, where the write-driver and column/row decoders are omitted for clarity. The state of the two reference cells is programmed by setting the voltages on RBL , $RSLL$ and $RSLR$ to provide the I_{ref} required for the memory- and PUF-mode accordingly. After the voltage of the reference cell has been settled, a data bit or random response bit is generated by comparing the current difference $I_{ref} - I_{dat}$ with a sense amplifier.

The design was evaluated in the HSPICE environment [126] and the MTJ was modeled using a publicly available MTJ model [36]. The resistance-area (RA) characteristics and switching dynamics of the MTJ is modeled by consistently self-solving the Non-Equilibrium Green Function (NEGF) and Landau-Lifshitz-Gilbert equation, including the Slonczewski spin-torque term (LLGS) [33]. The RA characteristics was calibrated to the experimental results in [142]. We used 45 nm bulk Predictive Technology Model (PTM) [19] for NMOS transistors in the circuitry. Table 5.3 lists the parameters assumed for the STT-MRAM based BF-MemPUF, as well as the nominal values and relative standard deviations of these parameters. Fig. 5.10(a) depicts the distributions of R_{MTJ} when a fixed biasing voltage of 100 mV is used. Fig. 5.10(b) shows the scattered plot of the randomly distributed resistances of 50 MTJ samples in AP and P states. It can be seen that the same propensity of output state is exhibited in both resistance states of the same MTJ sample.

In memory-mode, STT-MRAM read/write failure probabilities strongly depend on the sizing of the memory cells [33]. If the size of the memory cell is smaller, the current driving capability of the transistor will become weaker, thus resulting in high failure probability of

Table 5.3 Parameters used in STT-MRAM based BF-MemPUF

	Parameter	Nominal value	RSD
D	MOSFET channel width	W	-
	Reduction factor of the array size	α	-
P	Contact area	$45 \times 116 \text{ nm}^2$	5%
	Oxide layer thickness	1.15 nm	2%
	PMA energy barrier	$51k_B T$	-
	Saturation magnetization	850 emu/cm ³	-
	Damping factor	0.028	-
	Gyromagnetic factor	17.6 GHz/Oe	-
	MOSFET channel length	45 nm	10%
	MOSFET threshold voltage	0.466 V	10%
E	Supply voltage	1 V	-
	Wordline voltage	1 V	-
	Write cycle	40 ns	-
	Read cycle	20 ns	-
T	Temperature	200 ~ 400 K	-

write operations. However, if the size of the cell transistor is too large, occasional read-disturbance may occur which accidentally switch the data bit stored in the cell. To analyze the relationship between Φ_f and the size of the memory cell, the design parameters are varied within typical ranges. The values of t_{ox} and $area$ are held fixed while W is varied to analyze Φ_f . The read voltage is set to 0.1 V such that RD failure is dominant in the read operation [33]. When W is large, the equivalent TMR⁶ becomes larger as well, and this leads to a decrease in Φ_{f-RD} at some point. As mentioned before, increasing W also boosts the current driving capability of NMOS access transistors so that Φ_{f-WF} goes down. Fig. 5.11(a) plots Φ_{f-RD} and Φ_{f-WF} when W is increased from 1200 nm to 2000 nm. As can be observed from the figure, there is an optimal point where both Φ_{f-RD} and Φ_{f-WF} are minimized.

On the other hand, reducing the NMOS size increases its variations [6]. For instance, as an empirical rule, the deviation of NMOS transistor threshold voltage (σV_{th}) becomes larger when the transistor area becomes smaller, i.e.,

$$\sigma V_{th} \propto \frac{A}{\sqrt{W \cdot L}} \quad (5.16)$$

⁶Considering the impact of the resistance of the access transistor, equivalent TMR = $[(R_{AP} + R_T^{(AP)}) - (R_P + R_T^{(P)})] / (R_P + R_T^{(P)})$, where $R_T^{(AP)}$ and $R_T^{(P)}$ are the resistances of the access transistor in the read operation when the state of the MTJ is AP and P, respectively.

Fig. 5.9 Memory array architecture (left-top), working mechanism (left-bottom) and read circuit diagram (right) of the BF-MemPUF design based on STT-MRAM technology.

where A is the technology-dependent parameter. W and L are the width and length of the NMOS transistor, respectively. Fig. 5.12(a) depicts the relationship between σV_{th} and the size of the NMOS transistor [6, 95]. According to Eq. (5.9), Φ_u is evaluated by measuring the distinguishability between I_{dat} and I_{ref} given a certain level of noise. Noise effects in MTJ read operations are dominated by *thermal fluctuation*, which is modelled as a randomly distributed deviation of FL angle from the easy-axis [92]. The probability function of θ is characterized as

$$\mathbf{Pr}(\theta) = \frac{2H_k M_s}{k_B T} \exp \left[-\frac{H_k M_s}{k_B T} \sin^2(\theta) \right] \quad (5.17)$$

Therefore, Φ_u can be calculated using the spread of noise current ε in Eq (5.9) obtained from the distribution of θ . Fig. 5.12(b) depicts the calculated BERs (i.e., Φ_u) with varying W at $T = 200$ K, 300 K and 400 K.

As mentioned in the previous section, changing α at the architecture level affects the qualities of PUF. Fig. 5.12(a) shows the relationship between α and the proportion of bit 1's in the generated 1024-bit response string. It is observed that when α is decreased, the generated response bits skews more to 1 s. Fig. 5.13(a) and Fig. 5.13(b) plot the calculated Φ_u and Φ_e with varying α while all other parameters are held constant. The results indicate that both Φ_u and Φ_e can be effectively improved by increasing α .

5.3.3 Design optimization and results analysis

The STT-MRAM based BF-MemPUF is optimized using the proposed methodology in Fig. 5.6. Assuming the BF-MemPUF capacity is 128 kb (i.e., $M = 1024$ and $N = 128$), and using the data in [70], the array area is constrained to $2 \times 10^{-3} \text{cm}^2$. For the 2-finger

Fig. 5.10 (a) Resistance distribution of 1000 MTJ samples in AP and P states, respectively. Voltage biasing between the top and bottom nodes of the MTJ is 100 mV. (b) Resistance distribution of 50 MTJ samples, where the output state tendencies for the same MTJ sample in two different resistance states are correlated.

STT-MRAM cell layout in the design [44], the footprint of the cell is given by

$$F = (W/2 + 3\lambda) \times 16\lambda \quad (5.18)$$

where λ is the feature size of the fabrication technology, i.e., 45 nm.

Since Φ_{f-RD} can be neglected when the read voltage is small (≤ 0.1 V) [33], the objective function at step 5 in Fig. 5.6 is reduced to

$$\begin{aligned} \Phi_f &= \Phi_{f-WF} \cup \Phi_{f-RD} \\ &= \Pr(WF) + \Pr(RD) - \Pr(WF, RD) \end{aligned} \quad (5.19)$$

To accurately calculate the probability of failure effects which are rare events in memory operations, the Response Surface Modeling technique instead of Monte Carlo simulation method was used to evaluate Eq. (5.4) [33]. The calculated probabilities as well as the sys-

Fig. 5.11 (a) Calculated Φ_{f-RD} and Φ_{f-WF} with varying W ; (b) Estimated BER in 2000 samples under different temperature points.

tem constraints are fed into the optimization algorithm (Fig. 5.6) programmed in MATLAB [84] to find the solution efficiently.

Considering the overall area/energy consumption of the PUF system, the overhead of ECC code needs to be taken into consideration [16, 75]. For convenience of comparison, the BCH codec that is parameterized by different specifications is used to stabilize responses generated from the BF-MemPUF. It should be noted that the selection of ECC is a tradeoff between redundancy and error correction capability. Higher redundancy is desired for better error correction capability [71]. For example, a BCH-[127, 15, 27] code requires 127 memory cells to produce 15 bit output that may have as high as 27 bits of random errors. Instead of using a single ECC, concatenating two ECCs is beneficial to improve both the code efficiency and error correction capability [16, 71].

Table 5.4 lists the optimization results of using BCH-[127, 15, 27], BCH-[127, 29, 21] alone, concatenated code of BCH-[127, 29, 21] and repetition code-[3, 1, 3], respectively (For convenience, these three codecs are denoted as ECC-I, ECC-II, and ECC-III). As can be seen from the results, the BF-MemPUF designs using three different kinds of error correction codecs all achieve the expected qualities. For ECC-I, since the error correction

Fig. 5.12 (a) Decreasing the scaling factor α will result in a deterioration of randomness owing to the correlation effects created by sharing reference cells. The value of α is annotated in blue. (b) Trend of σV_{th} with scaled effective channel length of a NMOS transistor [95].

capability is larger than that of ECC-II, it results in lower Φ_u . Also, the optimized cell size is larger than that of PUF using ECC-II, hence the overall overhead and Φ_f are larger. Using ECC-III relaxes the burden of redundant cells to guarantee a sufficiently high BER, and that is why it does not have to sacrifice Φ_f for Φ_u .

Table 5.5 compares the proposed BF-MemPUF based on STT-MRAM technology optimized using ECC-I with some other representative implementations of memory-based PUF in terms of PUF qualities, area and energy consumption for generating a bit from a cell. The area/bit and energy/bit can be calculated by

$$\begin{aligned} \text{area/bit} &= \text{area/cell} \times \frac{n}{k} \\ \text{energy/bit} &= \text{energy/cell} \times \frac{n}{k} \end{aligned} \quad (5.20)$$

where area/cell and energy/cell are the area and energy consumption of a memory cell, respectively, and n and k are the parameters of the ECC used for response bit stabilization.

Fig. 5.13 Calculated (a) Φ_u and (b) Φ_e using different architecture arrangement strategies for the STT-MRAM array.

The temperature is 300 K and the supply voltages for the referred implementations are assumed to be the nominal values. It can be seen that the PUF quality metrics of the proposed eNVM based PUF are comparable to those of the conventional memory-based PUFs (it is even better than SRAM based PUF). However, the area and energy consumed for generating a bit are much smaller than the conventional instances because of the smaller footprint of a cell and lower power consumption in sensing a cell.

Since there is no data available in the cited papers about the values of Φ_f for the referenced memory-based PUF design, Φ_f is not compared. However, the superiority of eNVM for data storage has been widely acknowledged because of its nearly-zero leakage power consumption, comparatively high performance, and higher data retention time over the counterparts implemented using merely the conventional CMOS technology [135, 137].

5.3.4 Dual-mode operation on other eNVM based PUFs

While the major objective of running dual-mode application on eNVM based PUF is to prevent data remanence attacks, migrating the idea to other eNVM based PUFs is feasible.

Table 5.4 Optimization Results

ECC	ECC-I	ECC-II	ECC-III
α	32	64	8
W	1176 nm	815 nm	1773 nm
Φ_f	5.2×10^{-5}	2.2×10^{-4}	4.5×10^{-6}
$\Phi_u(200 \text{ K})$	6.7×10^{-14}	1.7×10^{-13}	5.1×10^{-13}
$\Phi_u(300 \text{ K})$	3.5×10^{-13}	5.1×10^{-12}	3.7×10^{-11}
$\Phi_u(400 \text{ K})$	1.8×10^{-12}	6.2×10^{-10}	2.5×10^{-10}
Φ_e	0.95	0.97	0.90

Table 5.5 Comparison of eNVM based BF-MemPUF with conventional memory based PUFs

PUF instances	BER	Φ_u	Φ_e	area/bit (μm^2)	energy/bit (pJ)
SRAM	6%	1.8×10^{-9}	0.99	7.5	0.015
Flip-Flop	4%	1.5×10^{-13}	1	55.3	0.068
Latch	3%	1.1×10^{-16}	-	31.5	0.023
Buskeeper	4%	1.5×10^{-13}	0.99	22.5	0.008
Proposed	4%	3.5×10^{-13}	0.95	2.0	0.001

¹Data of fractional Hamming distances and BERs of SRAM, FF, Latch and Buskeeper were from [75, 129]. Data of entropies were from [129].

¹Energy/cell of the referenced PUF designs were estimated in HSPICE with 45 nm bulk PTM. Area/cell were estimated with the cell layout based on the λ -based design rules.

For example, in the PCM based rPUF schemes proposed in Chapter 3, dual-mode operation can be made possible by modifying the quantization method in the sensing circuitry. Similar to the referencing circuits used in the BF-MemPUF in this Chapter, a sensing scheme can be properly designed to adaptively measure the subtle resistance variation of an eNVM cell and the resistivity state of the same cell. The tricky part here is how to deal with the reliability degradation issue when the resolution of reading circuit is increased for multiple-bit per cell. It is obvious that when the resolution is one bit, the scheme becomes no different from the BF-MemPUF. To reliably differentiate intermediate resistance range for producing multiple bits requires careful design on reference circuits as well as enhanced post-processing modules for error correction.

Compared to the PCM based rPUF, the highly reliable eNVM based PUF proposed in Chapter 4 cannot take in the dual-mode feature without extensive modifications. Consider the STT-MRAM based PUF introduced in Chapter 4, Section 4.3.1, whenever a bit is automatically written back to a secret cell in the enrollment phase, any following write oper-

ation is prohibited until the next enrollment. There is no simple yet effective circuit-level approach to reading/writing data bits in the memory cells while maintaining the response information. In case an eNVM sub-system should be re-used for normal data storage and response bit generation, alternating the two modes can be applied. For short-term cycles of communication scenarios where key-exchange, authentication and key revocation are executed frequently, such a highly reliable and reusable eNVM based PUF can be a good choice.

5.4 Summary

In this chapter, a novel memory-based PUF primitive, namely the BF-MemPUF, is introduced to overcome the design issues in conventional memory-based PUFs. The advantage of this new primitive over the conventional memory-based PUFs such as SRAM-based PUF is that it successfully eliminates the threat of fault-injection attack that makes use of data remanence effect to reveal PUF response bits from an SRAM-based PUF. An optimization methodology has also been proposed to maximize the memory yield and maintain the security metric of the eNVM based BF-MemPUFs. This method effectively solves the conflicting quality requirements of an MPUF affected differently by the process variations for dual applications in memory-mode and PUF-mode. The design methodology was illustrated by using STT-MRAM BF-MemPUF as an example and the efficacy of this novel memory-based PUF has been comparatively evaluated. On contrary to conventional PUFs, the proposed BF-MemPUF does not require buffer storage and hence, consumes less area. Furthermore, the potential risk of leakage or side-channel attacks on the remanent secret bits are inhibited, making it suitable for a wide variety of security applications.

Chapter 6

Conclusions And Future Work

6.1 Conclusions

Research in the realm of hardware security emerges as the time requires. With the advent and rise of the internet, which offers low-cost and ubiquitous connectivity, massive quantities of electronic devices are connected to the network on an almost permanent basis. This in turn requires veritable safeguarding techniques for reliable data transmission and secure data storage. A number of successful attacks have been demonstrated in the literature in the recent years, which prompted researchers worldwide to urgently venture into new paradigms to escalate security for data protection. Hardware security emerged as a promising solution to ensure privacy and data protection. The emerging PUF primitive, which utilizes the process variability of nano-scale electronic devices for generating device-specific signatures, has received global research attention. Published literature demonstrated the efficiency of PUF in myriad applications including device identification, authentication and key storage/generation. Research in this area begins to step into improving the quality (i.e. security and reliability), exploring alternative usage for advanced protocols and efficiency enhancement in modern computing systems.

The initial idea of developing eNVM technologies for PUF was invoked by a literature review on the process variability issues of memory devices and circuits. Inspired by the work in [47, 63, 98], research and development of physically reconfigurable PUF based on PCM was conducted as the first step to explore the possibilities of using alternative semiconductor materials for PUF with the intention to achieve enhanced characteristics. The subsequent work exploited unique features of NVM technologies to improve memory-based PUF qualities such as reliability and efficiency. The proposed design methods that are generically applicable on most of the contemporary eNVM technologies were verified on the representative designs based on experimentally calibrated eNVM device models and

industrial-standard CMOS device library. Simulation results demonstrated the superiority of the proposed work in terms of PUF quality (security and/or reliability) enhancement and hardware overhead (area and/or power consumption) reduction. On top of the contributions on practical PUF design, theoretical cryptographic analysis was also provided to formally prove the efficacy of deploying the proposed PUF primitives in certain security application scenarios.

In a nutshell, this dissertation presents extensive works that leverage on the emerging eNVM technologies for designing memory-based PUFs with advanced properties such as reconfigurability of challenge-response mapping characteristics, improved reliability under varying operating conditions, and dual-mode functionality with optimized quality. The main contributions are summarized as follows:

- Pragmatic designs of PCM-based physically rPUF were proposed and experimentally evaluated. Using PCM could endow a memory-based PUF primitive with multi-bit capacity per cell. This essentially enlarges the key pool of a memory-based PUF given a fixed array size. The auxiliary circuitry for reconfiguration is based on a fuzzy approach, where the electrical pulse for programming a PCM cell is stochastically controlled by the random physical properties of the pulse generator. The collective random effects originating from both the pulse generator and the PCM cells result in the randomness of the generated response bits, and altering the attributes of pulse generator will reconfigure the characteristics of the PUF. Aside from using PCM cell resistance variability for generating random and refreshable response bits, an alternative design that made use of the stochastic nature of PCM programming dynamics for the same functionality of challenge-response mapping and reconfiguration has also been proposed. Since dynamically producing response bits along with configuring the rPUF device happens simultaneously, the security level against prediction attack where an adversary may collect response bits for each configuration state of an rPUF at certain time point can be prevented. The experimental results demonstrated the efficacy of the design. Reliability, randomness, uniqueness and unclonability of both the proposed rPUF designs under different working conditions and configuration states were within expectation.
- The second contribution in this dissertation is reliability improvement of memory-based PUF by using eNVM technologies without heavy reliance on architectural methods or circuit-level modifications. The basic idea is to take advantage of the non-volatility of eNVM where random bits initially generated from an eNVM cell are “latched” into the same cell as reliable digital bits. The primary advantage of doing this is that one does not have to rely on hardware consuming architectural methods

such as ECC, or circuit-level techniques such as aging-effects, to enhance the reliability of bit regeneration from memory-based PUF cells. In addition, since writing back the initial random bit into the memory cell is automatically triggered internally, it will not cause any security vulnerability with the additional write-back operation. The idea was simulated with an illustrative design using STT-MRAM. Considering variations in operating conditions, the worst-case error rate of regenerating a random bit from an STT-MRAM cell is within the range of 10^{-6} . Furthermore, the equivalent energy and area consumption of generating a reliable bit in an illustrative design of STT-MRAM PUF were reduced by ~ 10 times compared to its representative SRAM counterparts.

- In the last part of the thesis, a dual-mode application, namely, data storage and random bit generation, which is based on eNVM was proposed and studied. Dual-mode model is possible due to the fact that the resistance value of an eNVM cell is continuous, and it can be either measured accurately with a fine-grained reference to generate a random bit or a coarse-grained reference to produce a digital data bit. The benefit of using eNVM for this purpose is that both working modes physically share one chip but functionally operate independently (not simultaneously). Another advantage of using eNVM is that it could save a data buffer for temporarily preserving data bits when switching between the two working modes, whereas the incapability of conventional memory (e.g. SRAM) based PUF in achieving this opens doors for attacks using fault-injection technique to retrieve response bit information from memory cells. This idea was verified by developing an STT-MRAM based PUF, where a design algorithm to achieve the optimal point in memory failure probability and PUF quality was applied. When compared to the conventional memory-based PUFs that can be re-used as both memory (or memory-like structure) and PUF, the optimized STT-MRAM based design has advantages in terms of enhanced security against fault-injection attack and improved efficiency in area and energy consumption (approximately 65% and 1000% reduction, respectively).

6.2 Future Work

Based on the author's extensive research on emerging eNVM-based PUF, several important yet untackled scientific problems which are worthy of further investigations have also been identified.

Fig. 6.1 (a) Architecture of SRAM system for dual-mode application. (b) Mechanism of SRAM based PUF in two working modes.

6.2.1 Dual-mode SRAM-based PUF system design and implementation

Reusing memory for dual-mode applications of data storage and secret bit generation was studied extensively in Chapter 5 of this dissertation. Most demonstration and evaluation in this work were based on simulation results due to the limitation in fabrication opportunities for eNVM technologies. While simulated results are meaningful and adequate to prove the efficiency of the proposed technique, the work is in fact more conclusive if a complete PUF system based on a contemporary CMOS technology (e.g., 45 nm CMOS technology) were to be fabricated and tested. SRAM, which is considered one of the most representative memory-based PUFs, can be easily taped out with CMOS technology. Similar to the eNVM-based PUF design, a dual-mode SRAM PUF also faces the same conflict considering the

Fig. 6.2 The impact of variance of threshold voltage on the probability of failure effect in memory-mode operations and probability of generating unreliable bits in PUF-mode.

different quality specifications in the two distinct working modes and therefore requires a proper design method to search for an optimal point for both modes. Fig. 6.1(a) illustrates the mechanism of an SRAM-based PUF working in different modes. In the PUF mode, an SRAM cell is randomly selected by a *challenge* to produce a *response* bit corresponding to discrepancies between the two inverters in the cell [47]. In the memory mode, regular write and read operations are executed on the same SRAM array where the cell that is to be written or read is accessed by memory address. The working mechanisms in two alternative modes on an SRAM individual cell are shown in Fig. 6.1(b).

A good SRAM-based PUF is characterized by 1) the functionality of one application mode does not affect the other; 2) the qualities of both application modes should meet the system-level specifications; 3) no major hardware design modifications is needed for dual-mode applications. The impact of process variations on the two working modes of an SRAM makes designing a good SRAM-based PUF an up-hill challenge. As shown in Fig. 6.2, when the variance of physical parameter(s)¹, $\sigma_{V_{th}}$, increases while its mean value, $\mu_{V_{th}}$, is kept the same, the probability of bit failure will be lowered while the probability of generating reliable bits is improved. This design paradox necessitates an optimal strategy that caters to both.

To design a SRAM-based PUF with dual application modes, the SRAM cell needs to be appropriately sized so that the quality expectations can be met for both applications. For memory mode, the failure probability should be kept as low as possible. For PUF mode, the randomness and reliability of the generated bits should be made as high as possible. This dissertation focuses on the reliability since it is a more critical metric for SRAM-based PUF

¹For simplicity, only the variation of threshold voltage is considered here.

compared to other types of PUF as mentioned earlier. Besides the requirement of failure probability and PUF reliability, area constraints and leakage issues of the targeted process technology will also need to be considered.

Future works for SRAM-based PUF can be conducted in the following aspects:

- An optimal sizing strategy considering both the quality of PUF operation and failure probability of memory operation need to be proposed. Moreover, circuit-level design techniques such as word-line boosting, supply voltage scaling and adaptive body biasing may be used to further enhance the qualities of the two working modes.
- To implement a whole PUF system, a fuzzy extractor can be incorporated. The fuzzy extractor can be designed based on a Reed-Muller codec and an LFSR based universal hash function, which can improve the reproducibility of PUF response bits and information entropy of the reconciled response bits, respectively.

6.2.2 Public PUF for FPGA IP licensing scheme

Another promising research direction is to apply a PUF primitive (particularly the PPUF primitive as introduced in Chapter 2) in IP protection scheme. FPGA technology is often used by system vendors for small/medium-scale designs as well as pilot projects. Using hardware-based parallel processing to accelerate operations and reduce power consumption with good post-design flexibility, FPGA-based solutions offer the benefits of both ASIC and multi-processor designs. Even then, designing a complete system on FPGA from scratch to meet the much narrower time-to-market window is a great challenge. To shorten the design turnaround time, Intellectual Property (IP)-based design methodology has prevailed. Unfortunately, the existing upfront IP core licensing model is impractical for small to medium volume of FPGA-based applications owing to the inefficiency of licensing process. Furthermore, the incorporation of the IP cores into the FPGA chips are highly vulnerable to malicious activities such as data piracy, secrecy misappropriation and reverse engineering.

The preliminary work proposed a new pragmatic security platform that will support the licensing of IP cores on a per-device basis without the need for a contractual agreement with an external trusted third party, large bandwidth and complicated flow of communications [148]. Besides ensuring that the incentives of all stakeholders involved in the FPGA IP core trading are not compromised, the proposed scheme guarantees total secrecy and integrity of licensed IP cores based on available primitives of contemporary FPGA devices (etc., the Device DNA of Xilinx FPGA board). Unlike existing cryptographic schemes for secure configuration of FPGA bitstream, which assume that the system developer has access to all content of the system and protect the entire system as a monolithic IP, our proposed

security framework safeguards the confidentiality of proprietary designs of each core vendor while allowing the system developer to independently authenticate multiple IP cores from different vendors. A new public-key cryptography equivalence of PUF based on the Digital Bimodal Fuction (DBF) is to be developed as a reliable, random and unique FPGA chip signature to simplify the security protocol for the IP core transaction [138]. This will prohibit the implementation of protected IP cores on unscreened excess devices and counterfeit chips sold in the gray market, and protects the IP core content from being cloned, reversed engineered and tampered. Using the self-reconfigurable feature of modern FPGAs, the hardware components used to enroll and install the licensed IP cores on authorized device are marginal and will be released upon secure configuration of the protected IP cores.

Future works in this direction may include:

- An improved version of DBF can be proposed and designed. To enhance its reliability, design optimization methodology proposed in Chapter 5 can be deployed. Along with the future work suggested in the previous sub-section, SRAM circuitry that can be re-used as both FPGA fabric and PUF primitive will be designed and evaluated. Customized SRAM chip fabricated using a cutting-edge technology can then be tested and studied. Following that, based on the measurement results, the possibility and feasibility of applying improved SRAM-based PUF for constructing DBF primitive can be evaluated.
- A new protocol for FPGA based IP licensing scheme using PPUF primitive need to be proposed. Unlike the existing approaches which is based on the symmetric cryptographic algorithm, the new method, which relies on the asymmetry in evaluation time between the simulation model and physical execution of the PPUFs, guarantees security as well as efficiency of the licensing scheme.

Publications

- **Book Chapters**

- [1] Z. Siarhei, **L. Zhang**, V. Klybik, A. Ivaniuk and C. H. Chang, “Design and Implementation of High-Quality Physical Unclonable Functions for Hardware-oriented Cryptography” in *Secure System Design and Trustable Computing*. Springer International Publishing.

- **Journal Papers**

- [1] **L. Zhang**, Z. H. Kong, C. H. Chang, A. Cabrini and G. Torelli, “Exploiting Process Variations and Programming Sensitivity of Phase Change Memory for Reconfigurable Physical Unclonable Functions”, *IEEE Trans. Information Forensics and Security*, vol. 9, no. 6, pp 921-932, April, 2014.
- [2] **L. Zhang**, X. Fong, C. H. Chang, Z. H. Kong and K. Roy, “Highly Reliable Spin-Transfer Torque Magnetic RAM based Physical Unclonable Function With Multi-Response-Bits Per Cell”, *IEEE Trans. Information Forensics and Security*, vol. 10, no. 8, pp 1630-1642, June, 2015.
- [3] **L. Zhang**, X. Fong, C. H. Chang, Z. H. Kong and K. Roy “Optimizing Emerging Non-Volatile Memories for Dual-Mode Applications: Data Storage and Key Generator”, *IEEE Trans Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 7, pp 1176-1187, June, 2015.
- [4] C. Yuan, **L. Zhang**, C. H. Chang and S. S. Chen, “A Low-power Hybrid Ring Oscillator PUF with Improved Thermal Stability for Lightweight Applications”, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 7, pp 1143-1147, June, 2015.
- [5] C. Yuan, **L. Zhang**, Z. Siarhei, C. H. Chang and S. S. Chen, “CMOS Image Sensor based Physical Unclonable Function for Coherent Sensor-level Authentication”, *IEEE Trans. Circuits and Systems: I - Regular Paper*, vol. 63, no. 11, pp 2629-2640, November, 2015.

- **Conference Papers**

- [1] **L. Zhang**, Z. H. Kong and C. H. Chang, “PCKGen: A Phase Change Memory based Cryptographic Key Generator” in *IEEE ISCAS-2013*, May, Beijing, China, pp 1444-1447, May, 2013.
- [2] **L. Zhang**, X. Fong, C. H. Chang, Z. H. Kong and K. Roy, “Feasibility Study of Emerging Non-Volatile Memory Based Physical Unclonable Functions” in *IEEE IMW 2014*, May, Taipei, Taiwan, pp 1-4, May, 2014.
- [3] **L. Zhang**, X. Fong, C. H. Chang, Z. H. Kong and K. Roy, “Highly Reliable Physical Unclonable Function Using Spin-Transfer Torque MRAM” in *IEEE ISCAS-2014*, June, Melbourne, Australia, pp 2169-2172, May, 2014.
- [4] **L. Zhang**, Z. H. Kong, C. H. Chang, A. Cabrini and G. Torelli, “Leakage-Resilience Enhancement of Memory-based Physical Unclonable Function Using Phase Change Memory” in *IEEE ICCST-2014*, Rome, Italy, pp 1-6, October, 2014.
- [5] C. Yuan, Z. Siarhei, **L. Zhang**, C. H. Chang and S. S. Chen, “CMOS Image Sensor Based Physical Unclonable Function for Smart Phone Security Applications” in *IEEE ISIC-2014*, Singapore, pp 392-395, November, 2014.
- [6] **L. Zhang**, C. H. Chang, Z. H. Kong and C. Q. Liu, “Statistical Analysis and Design of 6T SRAM Cell For Physical Unclonable Function With Dual Application Modes”, *IEEE ISCAS-2015*, Lisbon, Portugal, pp 1410-1413, May, 2015.

Bibliography

- [1] AMD. Platform security processor, June 2015. URL <http://www.amd.com/en-us/innovations/software-technologies/security>.
- [2] Ross Anderson and Markus Kuhn. Tamper resistance-a cautionary note. In *Proceedings of the second Usenix workshop on electronic commerce*, volume 2, pages 1–11, 1996.
- [3] Ross J. Anderson. *Security Engineering: A Guide to Building Dependable Distributed Systems*. Wiley, 2nd edition, March 2008.
- [4] ARM. Trust zone, June 2015. URL www.arm.com/trustzone.
- [5] Frederik Armknecht, Roel Maes, Ahmad-Reza Sadeghi, Francois-Xavier Standaert, and Christian Wachsmann. A formal foundation for security features of physical functions. In *Proc. IEEE Symp. Security and Privacy*, Oakland, CA, USA, May 2011.
- [6] Asen Asenov, Savas Kaya, and John H Davies. Intrinsic threshold voltage fluctuations in decanano MOSFETs due to local oxide thickness variations. *IEEE Trans. Electron Devices*, 49(1):112–119, 2002.
- [7] DW Bauder. An anti-counterfeiting concept for currency systems. *Sandia National Labs, Albuquerque, NM, Tech. Rep. PTK-11990*, 1983.
- [8] Nathan Beckmann and Miodrag Potkonjak. Hardware-based public-key cryptography with public physically unclonable functions. In *Information Hiding*, pages 206–220, 2009.
- [9] F. Bedeschi et al. 4-Mb MOSFET-selected μ trench phase-change memory experimental chip. *IEEE J. Solid-State Circuits*, 40(7):1557–1565, Jul. 2005.
- [10] Ferdinando Bedeschi et al. Staircase-down set programming approach for phase-change memories. *Microelectronics J.*, 38(10):1064–1069, 2007.
- [11] Ferdinando Bedeschi et al. A bipolar-selected phase change memory featuring multi-level cell storage. *IEEE J. Solid-State Circuits*, 44(1):217–227, Jan. 2009.
- [12] G Betti Beneventi et al. Analytical model for low-frequency noise in amorphous chalcogenide-based phase-change memory devices. *J. Applied Physics*, 106(5):054506.1–054506.8, Sept. 2009.

- [13] Mudit Bhargava and Ken Mai. A high reliability PUF using hot carrier injection based response reinforcement. In *Workshop on Cryptographic Hardware and Embedded Syst.*, pages 90–106. Santa Barbara, California, USA, Aug. 2013.
- [14] Mudit Bhargava, Cagla Cakir, and Ken Mai. Reliability enhancement of bi-stable PUFs in 65nm bulk CMOS. In *IEEE Symp. on Hardware-Oriented Security and Trust*, pages 25–30, San Francisco, CA, USA, Jun. 2012.
- [15] Swarup Bhunia and Saibal Mukhopadhyay. *Low-power variation-tolerant design in nanometer silicon*. Springer, 2011.
- [16] C. Bösch et al. Efficient helper data key extractor on FPGAs. In *Proc. Workshop on Cryptographic Hardware and Embedded Syst.*, pages 181–197, Washington, DC, USA, Aug. 2008.
- [17] S. Braga et al. Voltage-driven partial-reset multilevel programming in phase-change memories. *IEEE Trans. Electron Devices*, 57(10):2556–2563, Oct. 2010.
- [18] Geoffrey W Burr et al. Phase change memory technology. *J. Vacuum Science & Technology B*, 28(2):223–262, March-Apr. 2010.
- [19] Y. Cao et al. New paradigm of predictive MOSFET and interconnect modeling for early circuit simulation. In *Proc. IEEE Custom Integrated Circuits Conf.*, pages 201–204, Orlando, FL, USA, May 2000.
- [20] E Chen et al. Advances and future prospects of spin-transfer torque random access memory. *IEEE Trans. Magn.*, 46(6):1873–1878, 2010.
- [21] Q Chen, G Csaba, X Ju, SB Natarajan, P Lugli, M Stutzmann, U Schlichtmann, and U Ruhrmair. Analog circuits for physical cryptography. In *IEEE International Symposium on Integrated Circuits*, pages 121–124, Singapore, December 2009.
- [22] Qingqing Chen, György Csaba, Paolo Lugli, Ulf Schlichtmann, Martin Stutzmann, and Ulrich Ruhrmair. Circuit-based approaches to SIMPL systems. *Journal of Circuits, Systems, and Computers*, 20(01):107–123, 2011.
- [23] G.F. Close et al. Device, circuit and system-level analysis of noise in multi-bit phase-change memory. In *Proc. IEEE Int. Electron Devices Meeting*, pages 29–5, San Francisco, CA, USA, Dec. 2010.
- [24] Mafalda Cortez et al. Adapting voltage ramp-up time for temperature noise reduction on memory-based PUFs. In *IEEE Symp. Hardware-Oriented Security and Trust*, pages 35–40, Austin, TX, USA, Jun. 2013.
- [25] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Proc. Eurocrypt*, pages 523–540, Interlaken, Switzerland, May 2004.
- [26] M. J. Donahue and D. G. Porter. OOMMF user’s guide, version 1.0, Sept. 1999.
- [27] Jeffrey S Dwoskin and Ruby B Lee. Hardware-rooted trust for secure key management and transient trust. In *ACM Conf. Comp. Comm. Security*, pages 389–400, Alexandria, VA, USA, Oct. 2007.

- [28] Zoya Dyka, Christian Walczyk, Damian Walczyk, Christian Wenger, and Peter Langendoerfer. Side channel attacks and the non volatile memory of the future. In *ACM Int. Conf. on Compilers, Architectures and Synthesis for Embedded Syst.*, pages 13–16, Tampere, Finland, October 2012.
- [29] I. Eichhorn, P. Koeberl, and V. van der Leest. Logically reconfigurable PUFs: memory-based secure key storage. In *Proc. ACM Workshop on Scalable Trusted Computing*, pages 59–64, Chicago, IL, USA, Oct. 2011.
- [30] Karim Eldefrawy, Gene Tsudik, Aurélien Francillon, and Daniele Perito. SMART: secure and minimal architecture for (establishing dynamic) root of trust. In *Annual Network & Distributed Sys. Security Symp.*, pages 1–15, San Diego, CA, USA, Feb. 2012.
- [31] P Fantini et al. Characterization and modelling of low-frequency noise in pcm devices. In *Proc. IEEE Int. Electron Devices Meeting*, pages 1–4, San Francisco, CA, USA, Dec. 2008.
- [32] Xuanyao Fong, S.K. Gupta, N.N. Mojumder, S.H. Choday, C. Augustine, and K. Roy. KNACK: A hybrid spin-charge mixed-mode simulator for evaluating different genres of spin-transfer torque MRAM bit-cells. In *Int. Conf. Simulation of Semiconductor Processes and Devices*, pages 51–54, Sept 2011.
- [33] Xuanyao Fong, Sri Harsha Choday, and Kaushik Roy. Bit-cell level optimization for non-volatile memories using magnetic tunnel junctions and spin-transfer torque switching. *IEEE Trans. Nanotechnol.*, 11(1):172–181, 2012.
- [34] Xuanyao Fong, Sri Harsha Choday, Panagopoulos Georgios, Charles Augustine, and Kaushik Roy. Spice models for magnetic tunnel junctions based on monodomain approximation, Aug 2016. URL <https://nanohub.org/resources/19048>.
- [35] Xuanyao Fong et al. Failure mitigation techniques for 1T-1MTJ spin-transfer torque MRAM bit-cells. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, PP(99):1–1, 2013.
- [36] Xuanyao Fong et al. SPICE models for magnetic tunnel junctions based on monodomain approximation, August 2013. URL <https://nanohub.org/resources/19048>.
- [37] Archiranshu Garg and Tony T. Kim. Design of SRAM PUF with improved uniformity and reliability utilizing device aging effect. In *IEEE International Symposium on Circuits and Systems*, Melbourne, VIC, Australia, June 2014. To appear.
- [38] B. Gassend et al. Controlled physical random functions. In *Proc. Computer Security Applications Conf.*, pages 149–160, Las Vegas, NV, USA, Dec. 2002.
- [39] B. Gassend et al. Silicon physical random functions. In *Proc. ACM Conf. on Computer and Communications Security*, Washiton, DC, US, Nov. 2002.
- [40] GlobalPlatform. Trusted execution environment, 2015. URL www.globalplatform.org.

- [41] B Govoreanu et al. $10 \times 10\text{nm}^2$ Hf/HfO_x crossbar resistive RAM with excellent performance, reliability and low-energy operation. In *IEEE Int. Electron Device Meeting*, pages 31–6, Washington, DC, USA, December 2011.
- [42] J. Guajardo et al. FPGA intrinsic PUFs and their use for IP protection. In *Proc. Workshop on Cryptographic Hardware and Embedded Syst.*, pages 63–80, Vienna, Austria, Sept. 2007.
- [43] Ximeng Guan, Shimeng Yu, and H-SP Wong. On the switching parameter variation of metal - oxide RRAM - part I: Physical modeling and simulation methodology. *IEEE Trans. Electron Devices*, 59(4):1172–1182, 2012.
- [44] Sumeet Kumar Gupta et al. Layout-aware optimization of STT MRAMs. In *IEEE Conf. Design, Automation and Test in Europe*, pages 1455–1458, Dresden, Germany, Mar. 2012.
- [45] Clemens Helfmeier, Christian Boit, Dmitry Nedospasov, and Jean-Pierre Seifert. Cloning physically unclonable functions. In *IEEE Int. Symp. on Hardware-Oriented Security and Trust*, pages 1–6, Austin, TX, USA, Jun. 2013.
- [46] R. Helinski, D. Acharyya, and J. Plusquellic. A physical unclonable function defined using power distribution system equivalent resistance variations. In *Proc. IEEE/ACM Design Automation Conf.*, pages 676–681, San Francisco, CA, USA, Jul. 2009.
- [47] D.E. Holcomb, W.P. Burleson, and K. Fu. Power-up SRAM state as an identifying fingerprint and source of true random numbers. *IEEE Trans. Comput.*, 58(9):1198–1210, Sept. 2009.
- [48] D. Ielmini and Y. Zhang. Analytical model for subthreshold conduction and threshold switching in chalcogenide-based memory devices. *J. Appl. Phys.*, 102(5):54517–54517, 2007.
- [49] S Ikeda et al. A perpendicular-anisotropy CoFeB-MgO magnetic tunnel junction. *Nature materials*, 9(9):721–724, 2010.
- [50] J. Janesky et al. Magnetostatic interactions between sub-micrometer patterned magnetic elements. *IEEE Trans. Magn.*, 37(4):2052–2054, 2001.
- [51] Yongsoo Joo, Dimin Niu, Xiangyu Dong, Guangyu Sun, Naehyuck Chang, and Yuan Xie. Energy- and endurance-aware design of phase change memory caches. In *Design, Automation Test in Europe Conf.*, pages 136–141, March 2010.
- [52] Dae-Hwan Kang et al. One-dimensional heat conduction model for an electrical phase change random access memory device with an 8F^2 memory cell ($F=0.15\mu\text{m}$). *J. Applied Physics*, 94(5):3536–3542, Jun. 2003.
- [53] DH Kang et al. Two-bit cell operation in diode-switch phase change memory cells with 90nm technology. In *IEEE Proc. Symp. VLSI Technology*, pages 98–99, Honolulu, HI, USA, Jun. 2008.

- [54] S. Katzenbeisser, Ü. Kocabaş, V. van der Leest, A.R. Sadeghi, G.J. Schrijen, and C. Wachsmann. Recyclable PUFs: Logically reconfigurable PUFs. *J. Cryptographic Engineering*, 1(3):177–186, Nov. 2011.
- [55] Ganesh Khedkar and Dhireesha Kudithipudi. RRAM motifs for mitigating differential power analysis attacks (DPA). In *IEEE Computer Society Annual Symp. on VLSI*, pages 88–93, Amherst, MA, USA, August 2012.
- [56] Kinam Kim. Future memory technology: challenges and opportunities. In *IEEE Symp. VLSI Technology, Systems and Applications*, pages 5–9, Chutung, Hsinchu, Taiwan, Apr. 2008.
- [57] Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *Advances in Cryptology*, pages 388–397, Prague, Czech Republic, May 1999.
- [58] Patrick Koeberl et al. Memristor PUFs: a new generation of memory-based physically unclonable functions. In *IEEE Conf. Design, Automation and Test in Europe*, pages 428–431, Grenoble, France, Mar. 2013.
- [59] Hugo Krawczyk. LFSR-based hashing and authentication. In *Proc. Advances in Cryptology*, pages 129–139, Santa Barbara, CA, USA, Aug. 1994.
- [60] Aswin Raghav Krishna et al. Mecca: a robust low-overhead puf using embedded memory array. In *CHES*, pages 407–420. Sept. 2011.
- [61] Sandeep S Kumar, Jorge Guajardo, Roel Maes, G-J Schrijen, and Pim Tuyls. The butterfly PUF protecting IP on every FPGA. In *Proc. IEEE Int. Workshop on Hardware Oriented Security and Trust*, pages 67–70, Austin, TX, USA, Jun. 2008.
- [62] Sanjay V Kumar, Chris H Kim, and Sachin S Sapatnekar. Impact of NBTI on SRAM read stability and design for reliability. In *IEEE Int. Symp. Quality Electronic Design*, pages 6–pp, Santa Clara, CA, USA, Mar. 2006.
- [63] K. Kursawe et al. Reconfigurable physical unclonable functions-enabling technology for tamper-resistant storage. In *Proc. IEEE Int. Workshop on Hardware-Oriented Security and Trust*, pages 22–29, San Francisco, CA, USA, Jul. 2009.
- [64] Stephan Lai and Tyler Lowrey. OUM-a 180 nm nonvolatile memory cell element technology for stand alone and embedded applications. In *IEEE Proc. Int. Electron Devices Meeting Tech. Dig.*, pages 36.5.1–36.5.4, Washington, DC, USA, Dec. 2001.
- [65] Y. Lao and K.K. Parhi. Reconfigurable architectures for silicon physical unclonable functions. In *Proc. IEEE Int. Conf. on Electro/Information Technol.*, pages 1–7, Mankato, MN, USA, May 2011.
- [66] Simone Lavizzari et al. Reliability impact of chalcogenide-structure relaxation in phase-change memory (PCM) cells part II: Physics-based modeling. *IEEE Trans. Electron Devices*, 56(5):1078–1085, May 2009.
- [67] Zhang Le, Kong Zhi Hui, Chang Chip-Hong, A. Cabrini, and G. Torelli. Exploiting process variations and programming sensitivity of phase change memory for reconfigurable physical unclonable functions. *IEEE Trans. Inf. Forens. Security*, 9(6): 921–932, June 2014.

- [68] Jing Li et al. Design paradigm for robust spin-torque transfer magnetic RAM (STT MRAM) from circuit/architecture perspective. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 18(12):1710–1723, 2010.
- [69] Daihyun Lim et al. Extracting secret keys from integrated circuits. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 13(10):1200–1205, Oct. 2005.
- [70] C.J. Lin et al. 45nm low power cmos logic compatible embedded stt mram utilizing a reverse-connection 1t/1mtj cell. In *IEDM*, pages 1–4, Dec. 2009.
- [71] S. Lin and D.J. Costello. *Error control coding*, volume 123. Prentice-hall Englewood Cliffs, NJ, 2004.
- [72] K. Lofstrom, W.R. Daasch, and D. Taylor. IC identification circuit using device mismatch. In *Proc. IEEE Int. Solid-State Circuits Conf.*, pages 372–373, San Francisco, CA, USA, Feb. 2000.
- [73] Roel Maes, Pim Tuyls, and Ingrid Verbauwhede. Intrinsic PUFs from flip-flops on reconfigurable devices. In *3rd Benelux workshop on information and system security*, volume 17, Eindhoven, The Netherlands, Nov. 2008.
- [74] Roel Maes, Vladimir Rozic, Ingrid Verbauwhede, Patrick Koeberl, Erik Van der Sluis, and Vincent van der Leest. Experimental evaluation of physically unclonable functions in 65 nm CMOS. In *Proc. IEEE European Solid-State Circuits Conf.*, pages 486–489, 2012.
- [75] Roel Maes et al. PUFKY: A fully functional PUF-based cryptographic key generator. In *CHES*, pages 302–319. Sept 2012.
- [76] T.M. Maffitt et al. Design considerations for MRAM. *IBM J. Research and Development*, 50(1):25–39, 2006.
- [77] A. Maiti and P. Schaumont. Improving the quality of a physical unclonable function using configurable ring oscillators. In *Proc. IEEE Int. Conf. Field Programmable Logic and Applications*, pages 703–707, Czech Republic, Aug. 2009.
- [78] Abhranil Maiti et al. A large scale characterization of RO-PUF. In *IEEE Intl. Symp. on Hardware-Oriented Security and Trust*, pages 94–99, Anaheim, CA, USA, Jun. 2010.
- [79] M. Majzoobi, F. Koushanfar, and M. Potkonjak. Lightweight secure PUFs. In *Proc. IEEE/ACM Int. Conf. on Computer-Aided Design*, pages 670–673, San Jose, CA, USA, Nov. 2008.
- [80] M. Majzoobi et al. Techniques for design and implementation of secure reconfigurable PUFs. *ACM Trans. Reconfigurable Technol. and Syst.*, 2(1):5, 2009.
- [81] Mehrdad Majzoobi and Farinaz Koushanfar. Time-bounded authentication for FPGAs. *IEEE Trans. Inf. Forens. Security*, 6(3):1123–1135, 2011.
- [82] Mehrdad Majzoobi, Golsa Ghiaasi, Farinaz Koushanfar, and Sani R Nassif. Ultra-low power current-based puf. In *IEEE Symp. Circuits and Syst.*, pages 2071–2074, Brazil, May 2011.

- [83] PG Mather, JC Read, and RA Buhrman. Disorder, defects, and band gaps in ultrathin (001) MgO tunnel barrier layers. *Physical Review B*, 73(20):205412, 2006.
- [84] Mathworks. Matlab. URL <http://www.mathworks.com/>.
- [85] S. Meguerdichian and M. Potkonjak. Device aging-based physically unclonable functions. In *Proc. IEEE Design Automation Conf.*, pages 670–673, San Diego, CA, USA, Jun. 2011.
- [86] Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone. *Handbook of applied cryptography*. CRC press, 1996.
- [87] D. Merli et al. Side-channel analysis of PUFs and fuzzy extractors. In *Proc. Int. Conf. on Trust and Trustworthy Computing*, pages 33–47, Pittsburgh, PA, USA, Jun. 2011.
- [88] GX Miao et al. Disturbance of tunneling coherence by oxygen vacancy in epitaxial Fe/MgO/Fe magnetic tunnel junctions. *Physical review letters*, 100(24):246803, 2008.
- [89] N.N. Mojumder and K. Roy. Proposal for switching current reduction using reference layer with tilted magnetic anisotropy in magnetic tunnel junctions for spin-transfer torque (STT) MRAM. *IEEE Trans. Electron Devices*, 59(11):3054–3060, 2012.
- [90] Saibal Mukhopadhyay, Hamid Mahmoodi, and Kaushik Roy. Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, 24(12):1859–1880, 2005.
- [91] David Naccache and Patrice Fremanteau. Unforgeable identification device, identification device reader and method of identification, July 1995. US Patent 5,434,917.
- [92] Anurag Nigam, CW Smullen, Vidyabhushan Mohan, Eugene Chen, Sudhanva Gurumurthi, and Mircea R Stan. Delivering on the promise of universal memory for spin-transfer torque ram (STT-RAM). In *Proc. IEEE Intl. Symp. Low Power Electronics and Design*, pages 121–126, 2011.
- [93] T Nirschl et al. Write strategies for 2 and 4-bit multi-level phase-change memory. In *Proc. IEEE Int. Electron Devices Meeting*, pages 461–464, Washington, DC, USA, Dec. 2007.
- [94] Yossef Oren, Ahmad-Reza Sadeghi, and Christian Wachsmann. On the effectiveness of the remanence decay side-channel to clone memory-based PUFs. In *Workshop on Cryptographic Hardware and Embedded Systems*, pages 107–125. Santa Barbara, CA, USA, August 2013.
- [95] Michael Orshansky, Sani Nassif, and Duane Boning. *Design for manufacturability and statistical design: a constructive approach*. Springer, 2007.
- [96] Michael Orshansky et al. *Design for manufacturability and statistical design: a constructive approach*. Springer, 2008.

- [97] G. Panagopoulos, C. Augustine, and K. Roy. Modeling of dielectric breakdown-induced time-dependent stt-mram performance degradation. In *IEEE Device Research Conf.*, pages 125–126, Santa Barbara, CA, USA, June 2011.
- [98] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld. Physical one-way functions. *Science*, 297(5589):2026–2030, 2002.
- [99] Siani Pearson. Trusted computing platforms, the next security solution. *HP Labs*, 2002.
- [100] Ronald Perez, Leendert Van Doorn, and Reiner Sailer. Virtualization and hardware-based security. *IEEE Security & Privacy*, (5):24–31, 2008.
- [101] A. Pirovano et al. Reliability study of phase-change nonvolatile memories. *IEEE Trans. Dev. Mat. Rel.*, 4(3):422–427, Sept 2004. ISSN 1530-4388. doi: 10.1109/TDMR.2004.836724.
- [102] M. Potkonjak, S. Meguerdichian, A. Nahapetian, and Sheng Wei. Differential public physically unclonable functions: Architecture and applications. In *Design Automation Conference (DAC), 2011 48th ACM/EDAC/IEEE*, pages 242–247, June 2011.
- [103] Moinuddin K Qureshi et al. Enhancing lifetime and security of PCM-based main memory with start-gap wear leveling. In *IEEE/ACM International Symp. Microarchitecture*, pages 14–23, New York, NY, USA, Dec. 2009.
- [104] Jeyavijayan Rajendran et al. Nano-PPUF: A memristor-based security primitive. In *IEEE Comput. Society Annual Symp. VLSI*, pages 84–87, Amherst, USA, Aug. 2012.
- [105] Arijit Raychowdhury et al. Design space and scalability exploration of 1T-1STT MTJ memory arrays in the presence of variability and disturbances. In *IEEE Intl. Electron Devices Meeting*, pages 1–4, Baltimore, MD, USA, Dec. 2009.
- [106] Garrett S Rose et al. A write-time based memristive PUF for hardware security applications. In *IEEE/ACM Int. Conf. on Computer-Aided Design*, pages 830–833, San Jose, CA, USA, Nov. 2013.
- [107] U. Ruhrmair and M. van Dijk. PUFs in security protocols: Attack models and security evaluations. In *IEEE Symp. on Security and Privacy*, pages 286–300, San Francisco, CA, USA, May 2013.
- [108] U. Rührmair, C. Jaeger, and M. Algasinger. An attack on PUF-based session key exchange and a hardware-based countermeasure: Erasable PUFs. In *Proc. Financial Cryptography and Data Security*, pages 190–204, Divi Flamingo Beach Resort, Bonaire, Feb. 2012.
- [109] U. Rührmair et al. Modeling attacks on physical unclonable functions. In *Proc. ACM Conf. on Comput. and Communications Security*, pages 237–249, Chicago, IL, USA, Oct. 2010.
- [110] U Ruhrmair et al. Applications of high-capacity crossbar memories in cryptography. *IEEE Trans. Nanotechnol.*, 10(3):489–498, 2011.

- [111] Ulrich Rührmair. SIMPL systems: On a public key variant of physical unclonable functions. *IACR Cryptology ePrint Archive*, 2009:255, 2009.
- [112] Ulrich Rührmair. SIMPL systems, or: can we design cryptographic hardware without secret key information? In *Conference on Current Trends in Theory and Practice of Computer Science*, pages 26–45. Nový Smokovec, Slovakia, January 2011.
- [113] Ulrich Rührmair and Marten van Dijk. Practical security analysis of PUF-based two-player protocols. In *Proc. Workshop on Cryptographic Hardware and Embedded Syst.*, pages 251–267. Leuven, Belgium, Sept. 2012.
- [114] Ulrich Rührmair and Marten van Dijk. PUFs in security protocols: Attack models and security evaluations. In *IEEE SP*, pages 286–300, May 2013.
- [115] Ulrich Rührmair, Heike Busch, and Stefan Katzenbeisser. Strong PUFs: models, constructions, and security proofs. In *Towards hardware-intrinsic security*, pages 79–96. Springer, 2010.
- [116] Georgios Selimis et al. Evaluation of 90nm 6T-SRAM as physical unclonable function for secure key generation in wireless sensor nodes. In *IEEE ISCAS*, pages 567–570, 2011.
- [117] Gustavus J Simmons. A system for verifying user identity and authorization at the point-of sale or access. *Cryptologia*, 8(1):1–21, 1984.
- [118] Peter Simons et al. Buskeeper PUFs, a promising alternative to D flip-flop PUFs. In *Proc. IEEE Int. Symp. on Hardware Oriented Security and Trust*, pages 7–12, San Francisco, CA, USA, Jun. 2012.
- [119] Boris Skoric, Stefan Maubach, Tom AM Kevenaar, and Pim Tuyls. Information-theoretic analysis of coating PUFs. *IACR Cryptology ePrint Archive*, 2006:101, 2006.
- [120] Sergei Skorobogatov. *Semi-Invasive Attacks - A New Approach to Hardware Security Analysis*. PhD thesis, University of Cambridge, September 2004.
- [121] Sergei Skorobogatov. Flash memory "bumping" attacks. In *Workshop on Cryptographic Hardware and Embedded Systems*, pages 158–172. Santa Barbara, California, USA, Aug. 2010.
- [122] K. Sonoda et al. A compact model of phase-change memory based on rate equations of crystallization and amorphization. *IEEE Trans. Electron Devices*, 55(7):1672–1681, Jul. 2008.
- [123] Ying Su, J. Holleman, and B.P. Otis. A digital 1.6 pJ/bit chip identification circuit using process variations. *IEEE J. Solid-State Circuits*, 43(1):69–77, Jan. 2008.
- [124] G.E. Suh and S. Devadas. Physical unclonable functions for device authentication and secret key generation. In *Proc. ACM/IEEE Design Automation Conf.*, pages 9–14, San Diego, CA, USA, Jun. 2007.
- [125] D. Suzuki and K. Shimizu. The glitch PUF: a new delay-PUF architecture exploiting glitch shapes. In *Proc. Workshop on Cryptographic Hardware and Embedded Syst.*, pages 366–382, San Barbara, CA, USA, Aug. 2010.

- [126] Synopsys. HSPICE. URL <http://www.synopsys.com/>.
- [127] TCG. Trusted platform module (tpm) summary, Apr. 2008. <http://www.trustedcomputinggroup.org/resources/>.
- [128] Guido Torelli and P Lupi. An improved method for programming a word-erasable EEPROM. *Alta Frequenza*, 52(6):487–494, Nov. 1983.
- [129] Robbert van den Berg. *Entropy analysis of Physical Unclonable Functions*. PhD thesis, Eindhoven University of Technology, Aug. 2012.
- [130] Vincent van der Leest et al. Hardware intrinsic security from D flip-flops. In *Proc. ACM Workshop on Scalable Trusted Computing*, pages 53–62, Chicago, IL, USA, Oct. 2010.
- [131] D. Ventrice et al. Statistical modeling of bit distributions in phase change memories. In *Proc. IEEE European Solid-State Device Research Conf.*, pages 157–160, Athens, Greece, Sept. 2009.
- [132] AF Webster and Stafford E Tavares. On the design of S-boxes. In *Advances in Cryptology*, pages 523–534, Santa Barbara, CA, USA, Aug. 1986.
- [133] James B Wendt and Miodrag Potkonjak. The bidirectional polyomino partitioned PPUF as a hardware security primitive. In *IEEE Global Conference on Signal and Information Processing*, pages 257–260, Austin, TX, USA, December 2013.
- [134] Frans M. J. Willems, Yuri M. Shtarkov, and Tjalling J. Tjalkens. The context-tree weighting method: Basic properties. *IEEE Trans. Inf. Theory*, 41(3):653–664, May 1995.
- [135] Stuart A Wolf et al. The promise of nanomagnetism and spintronics for future logic and universal memory. *Proc. IEEE*, 98(12):2155–2168, 2010.
- [136] H-SP Wong et al. Metal-oxide RRAM. *Proc. IEEE*, 100(6):1951–1970, 2012.
- [137] H.P. Wong, S. Raoux, S. Kim, J. Liang, J.P. Reifenberg, B. Rajendran, M. Asheghi, and K.E. Goodson. Phase change memory. *Proc. IEEE*, 98(12):2201–2227, Dec. 2010.
- [138] Teng Xu, James B Wendt, and Miodrag Potkonjak. Digital bimodal function: an ultra-low energy security primitive. In *IEEE Int. Symp. Low Power Electronics and Design*, pages 292–296, Beijing, China, Sept. 2013.
- [139] Chikako Yoshida et al. A study of dielectric breakdown mechanism in CoFeB/MgO/CoFeB magnetic tunnel junction. In *IEEE Reliability Physics Symp.*, pages 139–142, Montreal, QC, Canada, Apr. 2009.
- [140] H. Yu, P.H.W. Leong, and Q. Xu. An FPGA chip identification generator using configurable ring oscillators. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, 20(12):2198–2207, Dec. 2012.

- [141] Shimeng Yu, Ximeng Guan, and H-SP Wong. Understanding metal oxide RRAM current overshoot and reliability using kinetic monte carlo simulation. In *IEEE Int. Electron Devices Meeting*, pages 26–1, San Francisco, CA, USA, Dec. 2012.
- [142] Shinji Yuasa, Taro Nagahama, Akio Fukushima, Yoshishige Suzuki, and Koji Ando. Giant room-temperature magnetoresistance in single-crystal Fe/MgO/Fe magnetic tunnel junctions. *Nature materials*, 3(12):868–871, 2004.
- [143] L. Zhang, Z.H. Kong, and C-H. Chang. PCKGen: a phase change memory based cryptographic key generator. In *Proc. IEEE Int. Symp. Circuits and Syst.*, Beijing, China, May 2013.
- [144] L. Zhang, X. Fong, C. Chang, Z.H. Kong, and K. Roy. Optimizing emerging non-volatile memories for dual-mode applications: Data storage and key generator. *IEEE Trans. Computer-Aided Design of Integrated Circuits and Syst.*, 34(7):1176–1187, July 2015.
- [145] L. Zhang, X. Fong, C. Chang, Z.H. Kong, and K. Roy. Highly reliable spin-transfer torque magnetic RAM-based physical unclonable function with multi-response-bits per cell. *IEEE Trans. Information Forensics and Security*, 10(8):1630–1642, Aug 2015.
- [146] Le Zhang, Chip-Hong Chang, A. Cabrini, G. Torelli, and Zhi Hui Kong. Leakage-resilient memory-based physical unclonable function using phase change material. In *IEEE Int. Carnahan Conf. Security Technology*, pages 1–6, Rome, Italy, Oct. 2014.
- [147] Le Zhang et al. Leakage-resilient memory-based physical unclonable function using phase change material. In *IEEE Int. Carnahan Conf. on Security Technology*, Rome, Italy, October 2014. To appear.
- [148] Li Zhang and Chip-Hong Chang. A pragmatic per-device licensing scheme for hardware IP cores on SRAM-based FPGAs. *IEEE Trans. Information Forensics and Security*, 9(11):1893–1905, 2014.
- [149] Wangyuan Zhang and Tao Li. Characterizing and mitigating the impact of process variations on phase change based memory systems. In *Proc. IEEE/ACM Int. Symp. Microarchitecture*, pages 2–13, New York, NY, USA, Dec. 2009.
- [150] Yaojun Zhang, Xiaobin Wang, and Yiran Chen. STT-RAM cell design optimization for persistent and non-persistent error rate reduction: a statistical design view. In *Proc. IEEE Intl. Conf. Computer-Aided Design*, pages 471–477, San Jose, CA, USA, Nov. 2010.
- [151] Hui Zhao et al. Spin-transfer torque switching above ambient temperature. *IEEE Magnetism Letters*, 3:300–304, 2012.
- [152] Yu Zheng, Maryam S Hashemian, and Swarup Bhunia. RESP: a robust physical unclonable function retrofitted into embedded SRAM array. In *IEEE Design Automation Conf.*, page 60, Austin, TX, USA, Jun. 2013.