The Applications of NVM Technology in Hardware Security

Chaofei Yang, Beiye Liu, Yandan Wang, Yiran Chen, Hai Li
Department of Electrical & Computer Engineering, University of Pittsburgh, Pittsburgh, PA, 15261, USA
{chy61, bel34, yaw46, yic52, hal66}@pitt.edu

Xian Zhang, Guangyu Sun
Center for Energy-Efficient Computing and Applications, Peking University, Beijing, 100871, China
{zhang.xian, gsun}@pku.edu.cn

ABSTRACT
The emerging *nonvolatile memory* (NVM) technologies have demonstrated great potentials in revolutionizing modern memory hierarchy because of their many promising properties: nanosecond read/write time, small cell area, non-volatility, and easy CMOS integration. It is also found that NVM devices can be leveraged to realize some hardware security solutions efficiently, such as *physical unclonable function* (PUF) and *random number generator* (RNG). In this paper, we summarize two of our works about using NVM devices to implement these hardware security features and compare them with conventional designs.

Keywords
Memristor; spin-transfer torque random access memory; hardware security.

1. INTRODUCTION
With significant effort in process and device development, the emerging *nonvolatile memory* (NVM) technologies gradually alleviate their high manufacturing cost and low performance, enabling many promising features like high-density cell structure, nanosecond read/write accesses, CMOS-compatible integration, *etc*. Accordingly, many research studies on NVM technologies have been conducted. In this paper, we will introduce two NVM applications developed by us, aiming at enhancing hardware security.

Physical unclonable functions (PUFs) have been extensively investigated for the purpose of secured and low-cost authentication. There exist various types of PUFs that take advantages of random physical disorders in CMOS process technologies, such as SRAM PUFs based on SRAM power-up states [1], RO PUF based on latency of oscillator [2], Arbiter PUF based on wire connection delay [3], *etc*. Recently, some NVM-based PUFs have also been proposed in order to improve efficiency in energy and area and to enhance the resistance to simulation attack [4] and invasive attack [5]. However, the existing NVM-based PUFs face two major design limitations – the environmental impacts and substantial modification to the peripheral circuitry. Thus, we proposed errPUF design [6], which maximizes the hardware reuse with existing read/write circuits in *spin-transfer torque RAM* (STT-RAM) to enhance the reliability under environmental variations.

Random number generators (RNGs) are widely used in various systems and applications where unpredictable data are required. RNG plays a crucial role in system protection of many applications. There are two types of typical RNG designs: *pseudo random number generator* (PRNG) and *true random number generator* (TRNG). PRNG generates a sequence of pseudo numbers by injecting an initial seed to a given computing algorithm. TRNG usually leverages unpredictable physical phenomenon to generate true random numbers. Memristors are emerging two-terminal nonlinear dynamic devices in which the stochastic processes are well demonstrated [7]. For instance, the distribution of static memristances at *high resistance state*/*low resistance state* (HRS/LRS) can be approximated by a lognormal probability density function. Previously, we presented a *memristor-based true random number generator* (MTRNG) design by leveraging the stochastic behaviors of memristor [9].

The remainder of the paper is organized as follows: We will start with a brief introduction about memristor and STT-RAM technologies. Then the use of NVM devices in hardware security, including PUF and RNG will be introduced. Finally, we will give our conclusion.

2. RELATED WORK

2.1 NVM-based PUF
NVM-based PUFs include Memristor PUF [4], FPUF [15], PCM PUF [17], DWM PUF [18] and STT-PUF [19], *etc*. They have several advantages over the CMOS-based ones. First, NVM-based PUFs are more energy and area efficient to be used in some resource-constraint scenarios such as sensor nodes [26]. Second, it is more complex to simulate a NVM-based PUF so that it becomes more difficult to launch a simulation attack [4]. Third, NVM-based PUFs are also less vulnerable to invasive attack [5] because the storage units, *e.g.*, GST for PCM, MTJ for STT-RAM, and metal-oxide for RRAM, are stacked atop of the control transistors [27]. Note that all these NVM-based PUFs are CMOS-compatible.

Despite the advantages of NVM-based PUFs, there are two main limitations of current designs. The first limitation is that some designs do not evaluate the environmental impacts which may degrade PUFs' reliability. For example, in FPUF [15], the program cycles before inducing a disturb error for every cell are first evaluated. Then, the correlation coefficient is calculated to distinguish the genuine chip from the faked chips. Besides program cycles, variety of latency and program wear are also measured as a source of randomness. However, when environmental variations are considered, all the measured parameters above may greatly change [20] and FPUF's response may be unreliable. For Memristor-based PUFs, in which node voltage [4] is leveraged to

GLSVLSI'16, May 18–20, 2016, Boston, MA, USA
 ISBN 978-1-4503-4274-2/16/05…$15.00
DOI: http://dx.doi.org/10.1145/2902961.2903043

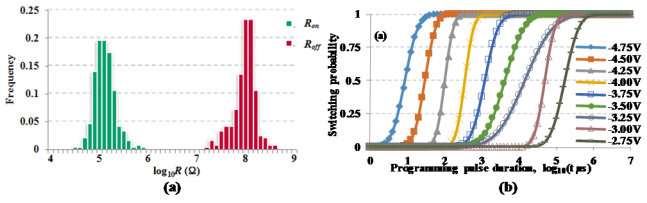

Figure 1: (a) Static stochastic behavior. (b) Probability distribution for ON switching [9].

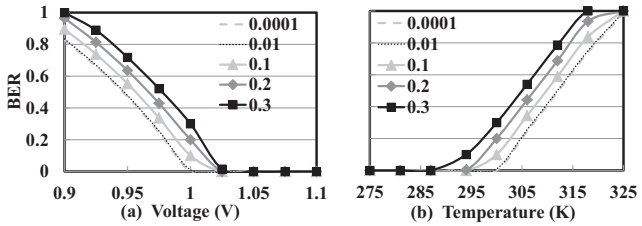

Figure 2: Variation of error rate of STTRAM cells due to change of environments: (a) different voltages (b) different temperatures [6].

generate random and unique fingerprints, the same limitation remains. The second limitation is that some designs require substantial modification to the peripheral circuitry to assist the extraction of device-level parameters, such as voltage sensors in every cell node [4], specified amplifier [17], differential circuit [19] and voltage to digital converter [21]. This will increase the design overhead and may affect normal read and write operations.

2.2 TRNG

TRNG usually leverages unpredictable physical phenomenon, such as thermal noise, random telegraph noise (RTN), atmospheric noise, electromagnetic and quantum [8]. Thermal noise is an intrinsic noise induced by thermal agitation of charge carriers (usually the electrons) inside an electrical conductor at equilibrium, which occurs regardless of applied voltage. RTN refers to a kind of electronic noise in semiconductors: when applying discrete voltage or current levels on semiconductors, sudden step-like RTN signals can be generated. Traditional thermal-noise-based TRNG usually is composed of a stochastic signal source, multi-level amplifiers, A/D converter and post-processing circuits [22]. Recently, a TRNG based on RTN in contact resistive random access memory (CRRAM) was proposed in which the high- and low-resistance states (HRS and LRS) of CRRAM are subject to RTN and therefore the resistance fluctuations can be converted to a stream of random bits [23]. Some TRNG designs leveraging the nanotechnologies have also been investigated. For example, Vivoli *et al.* presented a device-independent quantum TRNGs using a photon pair source based on spontaneous parametric down conversion (SPDC) which can gain both high entropy and high rate of random bit generation [24]. Spin dice is a spintronic-based TRNG that utilizes the stochastic nature of spin-torque switching in a magnetic tunnel junction (MTJ) to generate random numbers [25].

3. PRELIMINARY

3.1 Memristor Technology

3.1.1 Basics of Memristor

As the fourth fundamental component besides resistor, capacitor and inductor, memristor describes the dynamic relationship between charge (q) and flux (φ) [11]. Particularly, it can "remember" the total electric flux flowing through the device and its resistance is determined by the historical profile of the electrical excitations through the device.

3.1.2 Stochastic Behaviors

Stochastic behaviors have been widely observed in metal oxide-based memristor devices, including the variations in static states and dynamic switching processes. Figure 1 shows the static and dynamic stochastic behaviors observed in a TiO_2 device [16].

- ***Static stochastic behavior***: The final resistance value of a memristor during a programming operation is not deterministic but a stochastic variable related to the voltage amplitude and duration of programming pulse. The randomness of R_{on} and R_{off} is denoted as the static stochastic behavior of memristors. The distributions of R_{on} and R_{off} usually follow the lognormal probability density functions [12].
- ***Dynamic stochastic behavior*** is resulted by means of the complicated stochastic oxide electroforming process during ON/OFF switching in which the successful switching probability monotonically increases along with the increase of the amplitude and/or duration of programming pulse. More specific, the cumulative probability function of a successful switching between R_{on} and R_{off} follows a lognormal distribution [13].

3.2 STT-RAM

3.2.1 Basics

The popular STT-RAM cell design contains of one *magnetic tunneling junction* (MTJ) for data storage and one NMOS selective transistor. The logic bit represented by the MTJ is determined by the relative *magnetic direction* (MD) of its free and reference layers. When the MD is at the anti-parallel or parallel state, the MTJ demonstrates high or low resistances, representing logic '1' or '0', respectively. The state of the MTJ can be switched through a polarized current. The larger the write current is, the faster the MTJ switches. The read operation of STT-RAM is similar to those of conventional memories. Errors may happen during both read and write operations but the latter one contributes the most.

3.2.2 Sources of Write Errors

A write error occurs if the write current pulse stops before the MTJ successfully completes the switching. There are two sources of such errors:

Process variations of both the transistor and MTJ can affect the amplitude of write current. For example, the variations of transistor channel size can result in variance of write current driving ability. The variations in MTJ resistance can also influence the bias condition of the transistor and affect the driving current, causing an incomplete MTJ switching.

Thermal fluctuation happens during the MTJ switching. It is an intrinsic character demonstrating a random impact on the MTJ switching time. Thus, the error caused by the thermal fluctuation can only be detected occasionally.

Since both process variations and thermal fluctuation are random effects, the error rates of cells in an STT-RAM array follow a random distribution.

3.2.3 Impact of Environmental Variations

When taking the error rates of STT-RAM cells as a vector, it can be used as a fingerprint [15]. However, environmental variations have a significant impact on the reliability of this fingerprint, *e.g.*, the change of voltage and temperature can vary the amplitude of write currents, which in turns affects the error rates for cells.

Figure 2 shows the impact of different environmental variations on the bit error rates. The legends in the figure represent the error rates when the supply voltage is 1V and the temperature is 300K. It can be observed that the difference of error rates is more stable

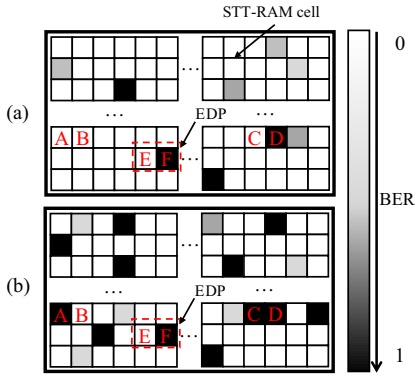

Figure 3: Error rates of a STT-RAM array in (a) Error-Least-State and (b) Error-Most-State. A-B and C-D are not EDPs. E-F is a valid EDP [6].

than the absolute values of error rates. Thus, we introduce a novel concept called *error-rate differential pair* (EDP). It reflects a stable relationship of bit error rates between two cells even with environmental variations.

4. HARDWARE SECURITY SOLUTIONS

4.1 STT-RAM-based PUF

4.1.1 Methodology

In order to simplify the description of EDP, we first introduce several definitions as follows:

Normal Working Environment refers to the working environment under which the PUF authentication works, *e.g.*, the supply voltage and temperature vary in the ranges of 0.9V −1.1V and 275K−325K, respectively.

Error-Most-State means the working environment under which the STT-RAM has the highest error rate, *e.g.*, the supply voltage is 0.9V and the temperature is 325K in Figure 2.

Error-Least-State means the working environment under which the STT-RAM has the lowest error rate, *e.g.*, the supply voltage is 1.1V and the temperature is 275K in Figure 2.

Read-Write-Read (RWR) Test is carried out in three steps: 1) read out the bit value in a cell, 2) write back the compliment bit to the cell, and 3) read the bit out again for comparison.

Having these terminologies, an EDP is defined as a pair of cells, A and B, that satisfy the following condition in both Error-Least-State and Error-Most-State. For N-round RWR tests, we have:

$$|Err_A - Err_B| \geqslant N_{th} \quad (1)$$

Err_A and Err_B represent the total number of errors occur in N rounds of tests for cell A and cell B, respectively.

EDP is the foundation of proposed err-PUF design. An example of valid EDP is shown in Figure 3. One STT-RAM cell is abstracted as one block. The color depth of a block represents the bit error rate (BER). Error rates of the same STT-RAM array under Error-Least-State and Error-Most-State are shown in Figure 3 (a) and (b), respectively. In this case, three pairs of cells are highlighted in the figure: A-B, C-D, and E-F. Only the E-F pair is a valid EDP. Pair A-B and C-D are not EDPs, because the difference is below the threshold under Error-Least-State and Error-Most-State, respectively.

In our err-PUF design, EDPs with a large difference of error rates are preferred. Since the detection of EDP is based on statistical testing results, it is possible that two cells with close error rates

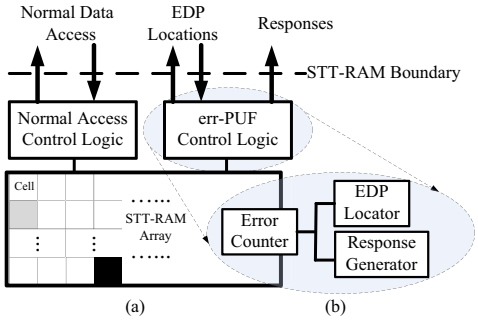

Figure 4: Illustration of err-PUF architecture [6].

are detected as a pair of EDP. We can find that identifying EDPs in a STT-RAM array relies on the setup of N and N_{th}. If we increase N and N_{th}, the probability is decreased for identifying an EDP with low error rate difference. However, increasing N includes more timing overhead in the process of detecting EDPs in a STT-RAM array.

4.1.2 err-PUF Design

Based on EDP, we propose a robust PUF design. The architecture of a STT-RAM with err-PUF is illustrated in Figure 4. As the PUF itself is embedded inside the STT-RAM array naturally, only the memory access control logic needs modification. As shown in the Figure 4, we add a component called "err-PUF control logic". The logic shares the same read/write interfaces for normal data access except for the ECC component. The whole process consists of several components for different phases of the err-PUF work flow, which is described as follows.

Pre-process phase includes two steps:

1) Identify all EDPs by scanning all cells in pair.
2) Store the location information of these EDPs to a database for later PUF verification.

The purpose of this phase is to identify all EDPs in a STT-RAM array after it is fabricated. Then, the location information of these EDPs in the array is stored in a database for later PUF verification. In order to achieve a secured PUF design, there should be enough number of EDPs in the array.

Enrollment phase includes four steps:

1) Randomly select N_{sec} EDP locations from the database as an input (i.e. a challenge).
2) When err-PUF receives the input, it will perform R-round RWR tests to the correlated EDPs. For each pair of two cells under test, if the first cell has more errors, a bit '0' is generated. Otherwise, a bit '1' is generated.
3) Add up all N_{sec} bits generated in the last step together and then compare it with $N_{sec}/2$. The comparison result is the final output of PUF circuits.
4) Store the output to a secure database as a reference.

The purpose of enrollment phase is to find challenge-response-pairs (CRPs) for PUF verification. Step-2 is to detect which cell in the EDP has a higher bit error rate. N_{sec} is a parameter that determines the security strength of our design.

Evaluation phase includes five steps:

1) - 3) is the same as Enrollment phase.
4) Compare the output bit with reference output in the database. If two values are different, it means that the response is incorrect.

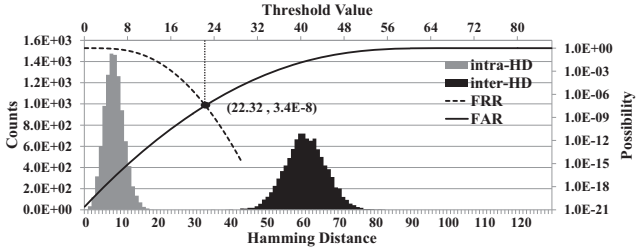

Figure 5: Inter-HD/intra-HD distribution and correlated FAR/FRR curves at 300K, 1.0V [6].

5) Repeat step-1 to step-4 for certain times (*e.g.* 128) and record the total number of incorrect responses, *i.e.*, Hamming Distance of two multi-bit outputs. If it is below a threshold value, the authentication succeeds. Otherwise, the authentication fails.

4.1.3 Experiment

Here we define two evaluation metrics: intra-HD and the inter-HD. **Intra-HD** represents the Hamming Distance between two responses of the same err-PUF design. **Inter-HD** represents the Hamming Distance of two responses from two different err-PUFs.

Reliability without environmental variation: The err-PUF is evaluated in a fixed environment state (V = 1.0V and T = 300K) for both enrollment and evaluation phases. In the Monte Carlo simulation, 10000 sets of challenges are used. The experimental results of intra-HD and inter-HD in our experiments are illustrated in Figure 5. The mean value of intra-HD is 7.76 with a variance of 7.29. For inter-HD, the mean value is 60.56 and the variance is 32.31. Based on these distributions, we can generate results of False Acceptance Rate(FAR) and False Rejection Rate(FRR) with different thresholds, which is also shown in Figure 5. From the results, we have the minimum *max (FAR, FRR)* = 3.4×10^{-8} when the threshold is set to 22.32. If we set the threshold at 23, the FAR and FRR are still less than 1×10^{-7}. Thus, it is acceptable for authentication for a large population of STT-RAM devices.

Reliability with environmental variation: We first set the working state with V = 1.0V and T = 300K for the enrollment phase to get the reference. Then we explore the worst case of evaluation phase when both variations are considered to demonstrate the reliability of errPUF. By exhausted experiments, we find that the worst case (highest FAR and FRR) happens at the state-A (V = 1.1V, T = 275K) for inter-HD and state-B (V = 0.9V, T = 325K) for intra-HD, which is shown in Figure 6. We can see that there is an obvious bias of inter-HD distribution. But even in the worst case, we have FAR=6.2×10^{-8} and FRR=1.3×10^{-7}, when the threshold is selected as 23. In conclusion, we can set the threshold as 23 to ensure FAR and FRR below 1×10^{-7} even with environmental variation.

Randomness of err-PUF: Note that there is nearly no overlap between different STT-RAM arrays' EDP positions, thus the false PUF will always output 0 with the same input of location information. Therefore, the inter-HD can present the randomness of our PUF's outputs which is shown in Figure 5. We can find that there is a slight bias between our result and the ideal one whose mean value should be 64. Despite the slight bias, the guessing probability of err-PUF's 128-bit output is still very low (about 2^{-118}).

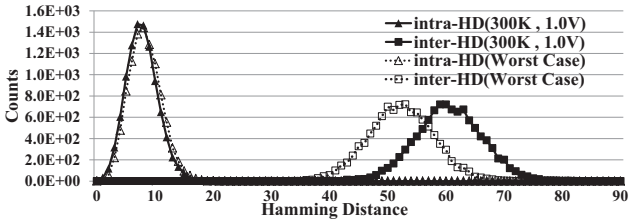

Figure 6: Inter-HD and intra-HD distribution at 300K, 1.0V and the worst case of inter-HD (@1.1V, 275K) and intra-HD (@0.9V, 325K) [6].

Comparison Results with other NVM-based PUFs: In order to compare err-PUF with other typical NVM-based PUFs, we synthesize a prototype of our err-PUF control logic with the 45nm technology. Results of other PUFs are listed in Table 1 from references. Our hardware cost is trivial mainly because we share most hardware with existing structure including the Read/Write control logic and STT-RAM cells. The only cost comes from little extra multiplexors and adders. Also, the test rounds of evaluation phase in our design are substantially fewer than those of SRAM PUF or FPUF. In the evaluation phase, only the cells within EDPs are being read or written, which reduces the dynamic power of our design. Note that we assume that the read/write width of the STT-RAM array is 512-bit. And the energy consumption and latency are calculated when 128-bit response is generated.

4.2 Memristor-based TRNG

4.2.1 Methodology

As aforementioned, the resistance of a memristor in ON or OFF state is not deterministic but random, even for a single identical device. Figure 1(a) presents the real measurement data of a TiO_2 memristor [16]. The distributions of static state resistances R_{on} and R_{off} both can be approximated to the lognormal probability density function such as [13]:

$$f_x(x;\mu,\sigma)=\frac{1}{x\sigma\sqrt{2\pi}}\cdot\exp\left(-\frac{(\ln x/\mu)^2}{2\sigma^2}\right),\quad x>0. \qquad (2)$$

where, μ is the normal mean and σ is the standard deviation of the normal distribution of the initial barrier width of the memristor device. Giving $E[R_{on}]$ and $E[R_{off}]$ as the means of R_{on} and R_{off}, respectively, and their standard deviations are $D[R_{on}]$ and $D[R_{off}]$, respectively. The device demonstrated in Figure 1(a) has $E[R_{on}] \approx 10^5\Omega$ and $E[R_{off}] \approx 10^8\Omega$. Both $D[R_{on}]$ and $D[R_{off}]$, are more than 2 orders smaller than the difference between the means ($E[R_{off}] - E[R_{on}]$). Such a highly isolated binary characteristic in memristors guarantees an ideal physical mechanism for MTRNG design.

The dynamic stochastic behavior refers to the successful switching probability between ON and OFF state. Under an exter-

Table 1. Comparison result between different NVM-based PUFs.

	STT-PUF [19]	PCM PUF [17]	Memristor PUF [4]	FPUF [15]	err-PUF [9]
Technology Node	45nm	45nm	45nm	45nm	45nm
Extra Circuit Area (μm^2)	5.5×10^3	1.7×10^3	9.1×10^3	3.3×10^2	2.9×10^2
Evaluation Phase Latency (μs)	7.18	12.8	10.1	1.5×10^7	2.32
Energy Consumption (pJ)	4.1×10^2	1.1×10^3	9.0×10^4	4.8×10^4	3.1×10^2

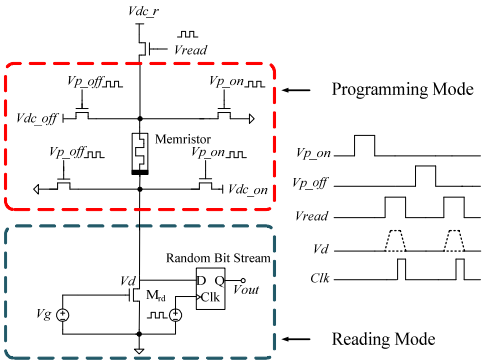

Figure 7: 1-branch MTRNG design [9].

nal programming pulse, the switching probability is determined by the voltage amplitude and the pulse width (duration) t. The cumulative distribution can be approximated by a lognormal distribution [13]:

$$F(t;\tau,\sigma_t)=\int_0^t \frac{1}{\sqrt{2\pi}\sigma_t T} e^{-\left(\frac{\ln T}{\sqrt{2}\tau\sigma_t}\right)^2} dT = \frac{1}{2} erfc\left(-\frac{\ln t}{\sqrt{2}\tau\sigma_t}\right). \quad (3)$$

Where, τ is the mean of the switching time, which has an exponential dependency on the applied voltage amplitude, while its deviation σ_t only has a weak dependence on the voltage [13]. Figure 1(b) shows the cumulative switching probability distributions of ON switching (OFF is similar). Increasing the programming duration of a constant-amplitude pulse can increase the switching probability, and a larger voltage amplitude decreases the required programming duration to reach a given switch probability.

4.2.2 MTRNG Design

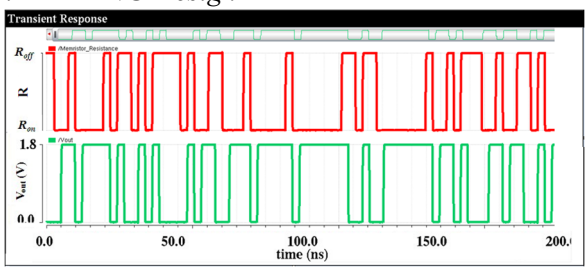

Figure 9. 1-branch MTRNG (R_{on}=$10^5\Omega$ and R_{off}=$10^8\Omega$) [9].

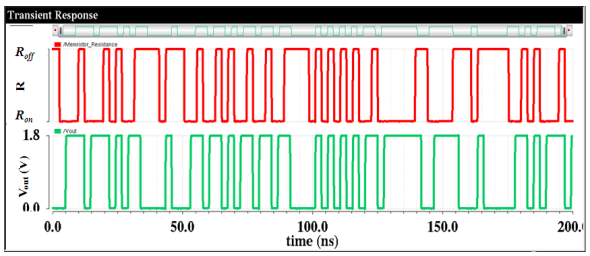

Figure 10. 1-branch MTRNG (R_{on}=$10^6\Omega$ and R_{off}=$10^7\Omega$) [9].

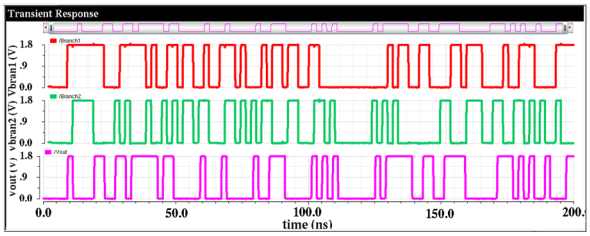

Figure 11. Simulation of 2-branch MTRNG [9].

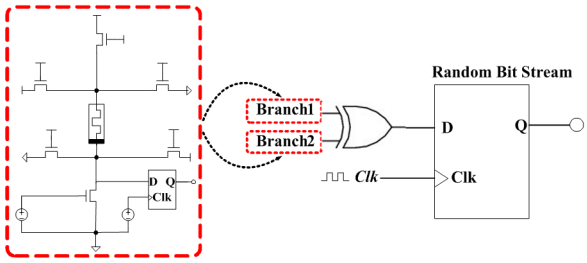

Figure 8: 2-branch MTRNG design [9].

The proposed MTRNG design switches between the programming mode and the reading mode to generate the random bit stream. In the programming mode, a programming pulse is applied on the memristor to trigger a dynamic switching between ON and OFF states. In the reading mode, the programmed resistance is converted to a binary bit. In the design, the selection of the programming pulse amplitude determines the maximal allowable sampling rate of the bit stream. We can control the ratio of the probability of 0's and 1's by modulating the programming duration. Ideally, a uniform distribution of 0/1 bit-stream can be obtained by aligning the pulse width to the switching probability of 0.5 under a given pulse voltage (refer Figure 1(b)).

Figure 7 depicts the proposed MTRNG circuit with the following key control and internal signals:

- V_{dc_r}, V_{dc_on} and V_{dc_off} are the DC voltage sources used in reading mode, the ON switching and the OFF switching programming, respectively.
- V_{read} is the control signal to enable the reading mode to detect the state of the memristor.
- V_{p_on} and V_{p_off} are used for programing the memristor to ON and OFF states, respectively.
- V_d is the bias voltage representing the state of memristor. It determines the generated output bit of the MTRNG.
- V_g is used to modulate V_d for bit generation.
- ***Clk*** is a clock signal to control the data capture at D flip-flop.

The sequence of control signals is also illustrated in Figure 7. V_{p_on} and V_{p_off} are turned on alternatively to enable the ON and OFF switching. Under the ideal condition with the sufficient programming voltage and pulse duration, the memristor can always be programmed, that is, the device switches between ON and OFF states. By properly controlling the programming voltage amplitude together with the pulse duration corresponding to the required bit distribution, the switching of the memristor becomes more random. In our design, following every programming period is a read operation enabled by V_{read}. The ON and OFF states of the memristor will be transferred to 1 or 0, respectively, under appropriate V_g setup. Here, a D flip-flop is used to recover distorted binary signal resulted by stochastic memristance values.

The simple MTRNG in Figure 7 can be used to generate a stream of random bits. However, the scheme cannot obtain the maximal entropy since the memristor will keep at the ON or OFF state if the previous switching fails. Assume the previous state of the memristor is OFF: if an ON switching triggered by V_{p_on} fails so that the memristor remains as OFF, the following OFF switching initialized by V_{p_off} does not affect the state of the memristor. In such a case, this OFF switching is not a stochastic process.

To improve the entropy of the random bit stream, we further enhance the design. As illustrated in Figure 8, it integrates two basic (1-branch) MTRNGs through an XOR gate. Because the

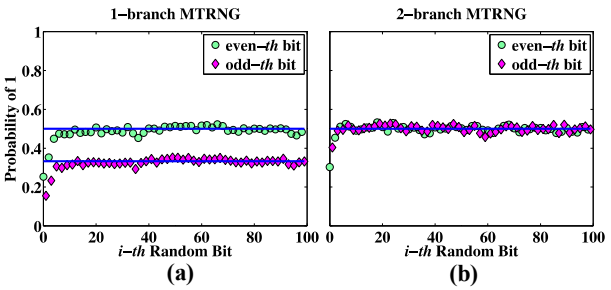

Figure 12: The probability distribution of random bit in the stream generated by 1-branch (a) and 2-branch (b) MTRNG [9].

stochastic switching of one memristor is independent to the other, the entropy of the random bit stream through the XOR function can be maximized under appropriate dynamic switching probability. We name this scheme as 2-branch MTRNG design.

4.2.3 Experiment

Figure 9 and 10 show the simulation results of the basic 1-branch MTRNG at the typical (R_{on}=$10^5\Omega$ and R_{off} =$10^8\Omega$) and the worst-case (R_{on} =$10^6\Omega$ and R_{off} =$10^7\Omega$) conditions, respectively. The simulations show that stochastic binary states of memristor can be successfully converted to random bit stream. Even in the extreme situation when R_{off} is very close to R_{on}, the basic 1-branch MTRNG design still functions properly.

Figure 11 shows the simulation result of the enhanced 2-branch MTRNG, the output random bit stream of which is dependent on the signals of two bit sequences generated by the two 1-branch MTRNGs. The simulations show that stochastic binary states of memristor can be successfully converted to random bit stream.

To analyze the probability distribution of the 1-branch and 2-branch MTRNG designs, the memristor ON switching and OFF switching probabilities are set to P_{on} = 1/4 and P_{off} = 1/3, respectively. To ease the explanation, we show the probability distributions of the first 100 bits generated by 1-branch and 2-branch MTRNG in Figure 12. Here, each point represents the probability of logic 1 at the bit. Simulation shows that both MTRNG schemes rapidly converge towards their stationary distributions in only a few steps.

For the 1-branch MTRNG design, the probability distribution of the odd-*th* bits is non-uniform. The situation can be solved by passing two bit streams of the 1-branch design through an XOR gate. Consistent to the theoretical analysis in Section 3.2.2, a uniformly distributed random bit stream can be generated via the 2-branch MTRNG design.

5. CONCLUSION

In this paper, we summarize two of our hardware security solutions. The proposed STT-RAM-based PUF employed the cell error rates. With the help of EDP, we can overcome the major challenge of PUF – environmental variations. Memristor-based TRNG leverages the stochastic behavior of memristors and converts the programmed resistances to random binary bit stream. 2-branch scheme promises the maximum entropy of random number generation. Both designs exhibit characteristics of simple structure, compact area, low power and show potentiality in hardware security.

Acknowledgement

The presented works were supported by NSF CNS-1253424, ECCS-1202225, and XPS-1337198.

6. REFERENCE

[1] D. E. Holcomb, *et al.*, "Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Numbers," IEEE Transactions on Computers, vol. 58, no. 9, pp. 1198-1210, 2009.

[2] G. E. Suh, *et al.*, "Physical Unclonable Functions for Device Authentication and Secret Key Generation," Design Automation Conference, 2007.

[3] J. W. Lee, *et al.*, "A technique to build a secret key in integrated circuits for identification and authentication applications," VLSI Circuits, 2004.

[4] J. Rajendran, *et al.*, "Nano-ppuf: A memristor-based security primitive," IEEE Computer Society Annual Symposium on VLSI, 2012

[5] D. Nedospasov, *et al.*, "Invasive PUF analysis," Fault Diagnosis and Tolerance in Cryptography, 2013

[6] X. Zhang, *et al.*, "A Novel PUF based on Cell Error Rate Distribution of STT-RAM," in proceeding of Asia and South Pacific Design Automation Conference, 2016

[7] S. Gaba, *et al.*, "Stochastic memristive devices for computing and neuromorphic applications," Nanoscale, vol. 5, pp. 5872-5878, 2013

[8] R. S. DeBellis, *et al.*, "Pseudo random number generator," US Patents US6061703, 2000.

[9] Y. Wang, *et al.*, "A Novel True Random Number Generator Design Leveraging Emerging Memristor Technology," Great Lakes Symposium on VLSI, 2015

[10] F. Pedregosa, "Scikit-learn: Machine Learning in Python," The Journal of Machine Learning Research, 2011.

[11] L. O. Chua, "Memristor-the missing circuit element," IEEE Transactions on Circuit Theory, vol. 18, pp. 507-519, 1971.

[12] M. Hu, *et al.*, "The stochastic modeling of TiO2 memristor and its usage in neuromorphic system de-sign," Asia and South Pacific Design Automation Conference, 2014.

[13] G. Medeiros-Ribeiro, *et al.*, "Lognormal switching times for titanium dioxide bipolar memristors: origin and resolution," Nanotechnology, vol. 22, p. 095702, 2011.

[14] T. Chang, *et al.*, "Short-Term Memory to Long-Term Memory Transition in a Nanoscale Memristor," ACS Nano vol. 5, no. 9, pp. 7669-7676, 2011.

[15] P. Prabhu, *et al.*, "Extracting device fingerprints from flash memory by exploiting physical variations," Trust and Trustworthy Computing, 2011.

[16] W. Yi, *et al.*, "Feedback write scheme for memristive switching devices," Applied Physics A, vol. 102, pp. 973-982, 2011.

[17] L. Zhang, *et al.*, "PCKGen: A Phase Change Memory based cryptographic key generator," International Symposium on Circuits and Systems, 2013.

[18] A. Iyengar, *et al.*, "DWM-PUF: A low-overhead, memory-based security primitive," International Symposium on Hardware-Oriented Security and Trust, 2014.

[19] L. Zhang, *et al.*, "Highly reliable memory-based Physical Unclonable Function using Spin-Transfer Torque MRAM," IEEE International Symposium on Circuits and Systems, 2014.

[20] N. Tega, *et al.*, "Anomalously Large Threshold Voltage Fluctuation by Complex Random Telegraph Signal in Floating Gate Flash Memory," International Electron Devices Meeting, 2006.

[21] W. Che, *et al.*, "A non-volatile memory based physically unclonable function without helper data," International Conference on Computer-Aided Design, 2014.

[22] S. Fujita, *et al.*, "Si nanodevices for random number generating circuits for cryptographic security," International Solid-State Circuits Conference, 2004.

[23] C. Y. Huang, *et al.*, "A Contact-Resistive Random-Access-Memory-Based True Random Number Generator," IEEE Electron Device Letters, vol. 33, no. 8, pp. 1108-1110, Aug. 2012.

[24] V. C. Vivoli, *et al.*, "Device-independent quantum random number generator with a photon pair source," arXiv preprint, 2014.

[25] A. Fukushima, *et al.*, "Spin dice: A scalable truly random number generator based on spintronics," Applied Physics Express, vol. 7, pp. 083001, 2014.

[26] R. Jeyavijayan, *et al.,* "Nano Meets Security: Exploring Nanoelectronic Devices for Security Applications." Proceedings of the IEEE, 2015.

[27] Valamehr, Jonathan, *et al.,* "A qualitative security analysis of a new class of 3-D integrated crypto co-processors." Cryptography and Security: From Theory to Applications, Springer, pp. 364-382, 2012.