# Variation-Resilient True Random Number Generators based on Multiple STT-MTJs

Yuanzhuo Qu, Bruce F. Cockburn, *Member, IEEE*, Zhe Huang, Hao Cai, *Member, IEEE*, Yue Zhang, *Member, IEEE*, Weisheng Zhao, *Senior Member, IEEE* and Jie Han, *Senior Member, IEEE*

***Abstract*—In the Internet of Things era, security concerns may require a cryptography system in every connected device. True random number generators (TRNGs) are preferred instead of pseudo-random number generators in the cryptography systems to achieve a higher level of security. For on-chip applications, we seek scalable and CMOS-compatible devices and designs for TRNGs. In this article, the stochastic behavior of the spin transfer torque magnetic tunnel junction (STT-MTJ) is utilized for the source of randomness. However, variations and correlations exist in MTJs due to fabrication limitations, so TRNG designs based on a single MTJ have to be post-processed or tracked in real time to ensure an acceptable level of randomness. Two novel designs are proposed in this article which can produce random sequences with high variation-resilience. The first design uses a parallel structure to minimize variation effects, and the second design leverages the symmetry of an MTJ-pair to take advantage of any correlations. Moreover, a universal circuit for quality improvement is proposed and it can be used with any random number generator. All of the designs are validated in a 28-nm CMOS process by Monte Carlo simulation with a compact model of the MTJ. The National Institute of Standards and Technology (NIST) statistical test suite is used to test the randomness quality of the generated sequences under the scenario of encryption keys in Transport Layer Security or Secure Sockets Layer (TLS/SSL) cryptographic protocol.**

***Index Terms*—Magnetic tunnel junctions, true random number generators, statistical tests, variations, correlations**

## I. Introduction

THE Internet of Things (IoT) puts a name to the accelerating trend to collect data from physically distributed networks of interconnected devices. More convenience is brought in the IoT era but security challenges become concerns. Inadequate levels of encryption may put transmitted data at risk and lead to privacy, property or even physical loss [1]. Therefore, stronger methods of on-chip encryption are required.

Random numbers are an essential part in an encryption algorithm. To produce random numbers, two categories of random number generators (RNGs) are used: pseudo-random number generators (PRNGs) and true random number generators (TRNGs) [2]. Tausworthe generators and a specific implementation, linear-feedback shift registers (LFSRs), are typical examples of PRNGs [3]. The sequences generated from a PRNG are fully deterministic but they have statistical properties that make them look random [4]. However, the predictability of PRNGs undermines the security level and thus TRNGs are sought for use in cryptography.

In contrast with PRNGs, TRNGs generate numbers with true randomness that originates from nondeterministic physical phenomena. For on-chip applications, only the schemes that are scalable and compatible with CMOS technology can be implemented. Also, energy-efficiency and high generation speed are important implementation criteria.

One major group of generators which does not involve devices other than CMOS, such as those using metastability [5] and oscillator jitter [6], are called all-digital TRNGs. They tend to have relatively poor randomness, so post-processing is usually needed, which increases the area and energy. Another group of generators leverages the stochastic behavior in some kinds of novel nanoscale devices, such as memristors [7-8] and magnetic tunnel junctions (MTJs). MTJs with spin transfer torque (STT) switching have the advantages of high density, high endurance, and compatibility with CMOS process. STT-MTJ based TRNGs are more power-efficient and have higher generation speed compared with memristor-based TRNG.

However, due to fabrication limitations, variations exist in MTJs [9], which will lead to a probability bias in the generated sequences. TRNG designs based on a single STT-MTJ device have to be post-processed or tracked in real time to ensure an acceptable level of randomness. One of the possible solutions in literature includes a feedback calibration circuit [10-12], in which the actual frequency of 1's in the output is calculated. Then the probability of the next bit to be generated can be adjusted according to the previous outputs in order to ensure an overall probability of 50%. Another method is to use multiple MTJs to generate multiple bits, and perform XOR operations among them. At least four MTJs and three XOR gates are needed to get one bit random number in [13], which wastes

Manuscript received July 5, 2017; revised February 12, 2018; revised April 23, 2018; accepted September 20, 2018. Date of publication MMM DD, YYYY; date of current version MMM DD, YYYY. This work was supported by the Natural Sciences and Engineering Research Council of Canada.

Y. Qu, B. F. Cockburn and J. Han are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada (e-mail: {yuanzhuo, cockburn, jhan8}@ualberta.ca).

Z. Huang, Y. Zhang and W. Zhao are with the School of Electronic and Information Engineering and Spintronics Interdisciplinary Center, Beihang University, Beijing 100191, China (e-mail: {huangzhe, yz, weisheng.zhao}@buaa.edu.cn).

H. Cai contributed to this work while he was visiting the Laboratoire Traitement et Communication de l'information (LTCI), Télécom-ParisTech, Université Paris-Saclay, 75013, France (E-mail: cai@telecom-paristech.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

generated bits and increases hardware cost.

The post-processing or real-time tracking circuits would be relatively large compared with the simple generation circuit. Also, using a calibration circuit undermines the randomness because the probability for each bit is either higher or lower than 50% according to previous outputs. Therefore we seek TRNG designs based on MTJs that can provide random sequences with high variation-resilience.

Two designs are proposed in this article: the first uses a parallel structure with multiple MTJs, which is based on our previous work appeared as [14] in DATE 2017. The new contributions of this article include the second design which uses an MTJ-pair leveraging the symmetry, as well as the following:

- Schematics and generating procedures of the proposed MTJ-based TRNG designs.
- Discussion of correlation issues of MTJ variations.
- Theoretical analysis of the quality improvement circuit (QIC) and its implementation.
- Randomness quality analysis of the generated random sequences and comprehensive comparisons on the performances of the proposed designs and other generators.

The proposed designs were verified in simulation using the perpendicular magnetic anisotropy (PMA) STT-MTJ compact model with ST Microelectronics' 28-nm fully depleted silicon-on-insulator (FD-SOI) CMOS technology. The randomness quality was validated using the National Institute of Standards and Technology (NIST) SP-800 statistical test suite.

The rest of the article is organized as follows. Section II reviews MTJ device and stochastic STT switching, as well as the problems that arise in single MTJ switching. The proposed schematics of the parallel design and the symmetry MTJ-pair design are demonstrated in Section III and Section IV, respectively. Section V discusses correlation issues and a quality improvement circuit is proposed in Section VI. Evaluations and comparisons of the designs follow in Section VII. Finally, conclusions are drawn in Section VIII.

## II. Single MTJ Switching

### A. MTJ device structure

An MTJ is a basic spintronic device that exploits the effects of tunnel magnetoresistance. Fig. 1 shows a typical structure of the MTJ, which has a sandwich structure with three layers: two relatively thick ferromagnetic layers (e.g., CoFeB) separated by one relatively thin tunneling barrier layer (e.g., MgO). One of the ferromagnetic layers is called the free layer for its switchable magnetization and the other one is called the pinned layer or fixed layer for its fixed magnetization. There are two stable states for an MTJ, parallel (P) or anti-parallel (AP), determined by the relative magnetization of the two ferromagnetic layers. The device has a lower electrical resistance $R_P$ in the P state and a higher resistance $R_{AP}$ in the AP state. The tunnel magnetoresistance ratio (TMR) = $(R_{AP} - R_P) / R_P$ characterizes the relative resistance difference between the two states, which is typically between 150% and 200% [15].

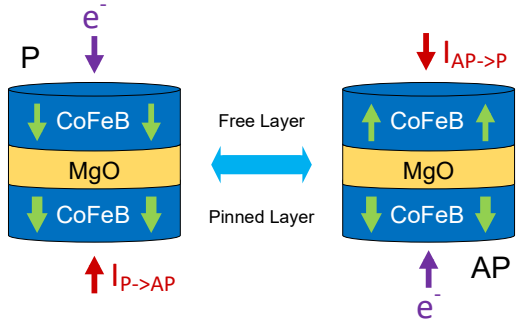



Fig. 1. The structure of an MTJ and the switching between two states.

The MTJ used in this work has perpendicular magnetic anisotropy (PMA). Thus it has a better thermal stability and a lower critical current compared with the in-plane magnetic anisotropy MTJ [16].

### B. MTJ probabilistic switching

To set the state of an MTJ, a current is injected into the MTJ from one direction to produce an effect called spin transfer torque (STT) switching. If the current is injected from the pinned layer side, the MTJ will be set to the AP state. If the current is injected from the free layer side, the MTJ will be set to the P state (Fig. 1). During the STT switching process, the current (electrons) is spin-polarized when going through the pinned layer, and the spin-polarized current will transfer sufficient spin-angular momentum to the magnetic moment in the free layer to switch its magnetization making it align with that of the current [17]. STT switching needs a lower current density compared with the switching method caused by the current-induced magnetic field, so the STT-MTJ is both more scalable and power-efficient [18-22].

Due to thermal fluctuations of magnetization during STT switching, the time to complete the switching follows a statistical distribution. In fact the switching is probabilistic given a fixed current and pulse duration. The relationship between the amplitude ($I$), duration ($t$) of the current pulse and the switching probability ($P$) can be expressed as follows:

$$P(I,t) = 1 - \exp\left(-\frac{t}{\tau}\right) \quad (1)$$

$$\tau(I) = \tau_0 \exp\left[\Delta\left(1 - \frac{I}{I_{c0}}\right)^2\right] \quad (2)$$

where $\tau$ is the mean switching time, $\tau_0$ is the attempt time, $I_{c0}$ is the critical switching current at 0 K and $\Delta$ is the thermal stability factor related to temperature [13].

Based on (1) and (2), when the current ($I$) and the pulse duration ($t$) are well controlled, a certain switching probability for an MTJ can be achieved. An MTJ will be in either state with equal probability after a carefully controlled current pulse aiming for 50% switching probability is applied. Then a random bit will be output by sensing the state of the MTJ. This intrinsic stochastic behavior is exploited to generate random numbers.

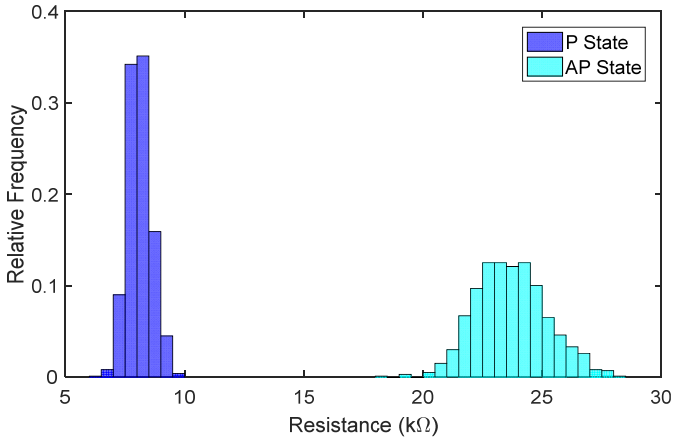



Fig. 2. The resistance distributions of $R_P$ and $R_{AP}$ in 28-nm PMA-STT-MTJ. 1000 Monte Carlo simulations were performed for each resistance state.

### C. Problems using a single MTJ

The two resistance values $R_P$ and $R_{AP}$ are affected by several factors such as the dimensions of the MTJ as well as other material properties. Due to the limitations in fabrication, especially the limited accuracy in the thickness of the three layers during thin film deposition, the resistances of the fabricated MTJs will vary from the nominal values [23]. To consider this effect at the design stage, three parameters are extracted to represent the MTJ variations: the thickness of the tunneling barrier layer $(t_{ox})$, the thickness of the free layer $(t_{sl})$ and the TMR value. These parameters are assumed to follow Gaussian distributions with standard deviations of 3% of the expected values [24]. The resistance is affected by the combined effects of these parameters.

The distributions of the two resistance values for the MTJ model used in the designs are shown in Fig. 2. The mean values of $R_P$ and $R_{AP}$ are 8.1 kΩ and 23.7 kΩ, respectively, and the standard deviation is 6.3% of the mean. In a TRNG design,

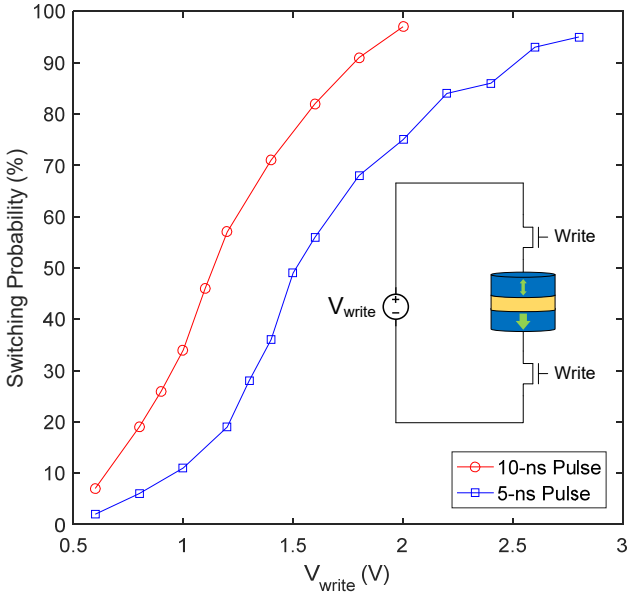



Fig. 3. The switching probability under different voltages with 5-ns and 10-ns pulse durations. The die temperature is 27 C. The initial state is the P state. Each result is an average from 100 Monte Carlo simulations.

TABLE I
MTJ PARAMETERS

| Parameter | Description | Value |
|---|---|---|
| $t_{ox}$ | Thickness of the MgO layer | 0.85 nm |
| $\sigma_{t_{ox}}$ | Standard deviation of $t_{ox}$ | 3% of 0.85 nm |
| $t_{sl}$ | Thickness of the free layer | 1.3 nm |
| $\sigma_{t_{sl}}$ | Standard deviation of $t_{sl}$ | 3% of 1.3 nm |
| *TMR* | Tunnel magnetoresistance ratio | 200% |
| $\sigma_{TMR}$ | Standard deviation of TMR | 3% of 200% |
| *Area* | MTJ dimensions | 28 nm × 28 nm × π/4 |

MTJ variations will affect the current in circuits and they can undermine the quality of the generated random numbers.

The switching probability of a single MTJ under different voltages and pulse durations was evaluated using Monte Carlo simulations. A PMA-STT-MTJ compact model [25] was used with 28-nm FD-SOI CMOS technology, and the hybrid MTJ/CMOS circuits were simulated in Cadence Virtuoso. The variations are integrated in the model by using the random functions and statistical block, which are provided by Verilog-A language under Cadence environment. For instance, $rdist_uniform generates a uniform distribution in a limited area and $rdist_normal generates a normal distribution with fixed mean value and standard deviation. The values of the parameters used in the MTJ model are listed in Table I. The die temperature in all simulations was set to 27 C. The choice of 200% for the TMR is justified from recent work [20-22]. The results of the stochastic switching are shown in Fig. 3, where different voltages and pulse durations are seen to affect the MTJ switching probabilities. The actual voltage and pulse width should be chosen according to the specific circuit parameters to achieve the desired switching probability.

Since parameter variations exist in all MTJs, the resistance of any particular MTJ will differ a little from the nominal value. Therefore, the current going through it will differ and so will the switching probability, which will lead to a probability bias in the generated sequences. The MTJ variation at the initial P state will lead to a standard deviation of 3.14% in the actual probability from the ideal 50%. Therefore, using only one MTJ is not sufficient to generate practical random sequences because the probability varies from 40.58% to 59.42% over $\pm 3\sigma$. Other methods are required to improve the randomness quality.

## III. THE PARALLEL DESIGN WITH MULTIPLE MTJS

The first proposed design uses parallel MTJs to compensate for the variation problem without the use of complicated feedback circuits. Since the standard deviation of the average of $N$ independent Gaussian-distributed random variables is

$$\sigma_{\frac{X_1+\cdots+X_N}{N}} = \frac{\sqrt{\sigma_1^2+\cdots+\sigma_N^2}}{N} \left(= \frac{\sigma_N}{\sqrt{N}}, if\ X_1 = \cdots = X_N\right), \quad (3)$$

the random sequences generated by multiple MTJs will have smaller standard deviations (divided by $\sqrt{N}$) in the probability. In other words, the parallel structure averages the biased probabilities of each single MTJ to get an overall probability

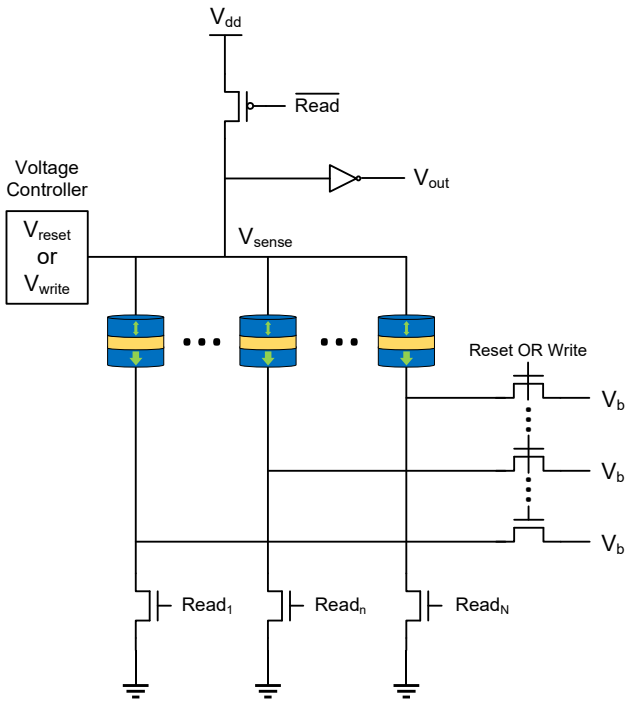



Fig. 4. Proposed TRNG with multiple parallel MTJs.

closer to 50%.

The schematic of the proposed parallel MTJ TRNG design is shown in Fig. 4. Three MTJs are shown in the figure, but the actual number of MTJs used can be adjusted according to the requirements.

For an array with $N$ MTJs, the control signals are *Reset*, *Write* and $Read_n$ ($n = 1, 2, \cdots, N$). To produce $N$ random bits, the circuit needs to go through $N + 2$ phases: 1) a reset phase, 2) a write phase and 3) $N$ read phases, with each phase taking 5 ns. In each phase, the corresponding control signal is driven high while the others are held low. In the first two phases, all MTJs work simultaneously. In the read phases, one MTJ is sensed at a time. Here the $N + 2$ phases are explained in detail:

*1) Group Reset*

In the reset phase, *Reset* is high and other control signals are low. The voltage controller drives $V_{reset}$, and current flows from the free layer (top) to the pinned layer (bottom) until all MTJs are switched to the P state. $V_{reset}$ is higher enough than $V_b$ to ensure an almost deterministic switching. At the end of the reset phase, all MTJs are in the P state waiting for the probabilistic switching in the write phase.

*2) Group Write*

In the write phase, *Write* is high and other control signals are low. The voltage controller drives $V_{write}$, which is lower than $V_b$ to induce a switching current going from the pinned layer to the free layer. The voltages are selected to target a 50% switching probability in 5 ns for each MTJ. Since the MTJs are connected in parallel, the voltages across each MTJ and the corresponding transistors are the same. All MTJs are written simultaneously, but each MTJ switches independently. The voltage controller ensures that $V_{write}$ is held steady despite MTJ switching. At the end of the write phase, an MTJ will change to the AP state if it switches; otherwise, it will remain in the P state.

*3) Read*

In the read phases, only one of the $N$ $Read_n$'s is high, from $Read_1$ to $Read_N$, while all other signals are low. The current flows from $V_{dd}$ to *GND* passing through only the selected MTJ. Depending on the resistance of that MTJ, the $V_{sense}$ will differ (the voltage controller is now off). The inverter (or some other kind of sense amplifier) will detect the difference and amplify it. Finally, the digital output at $V_{out}$ will indicate the resistance state of the selected MTJ. After $N$ cycles, the states of all the $N$ MTJs are sensed.

The proposed parallel structure will not only produce random numbers with higher randomness quality but also introduce other advantages compared with a single MTJ circuit. First, only one multiplexed sensing circuit is needed to read out all states of the $N$ MTJs at $V_{out}$, which saves hardware. Also, all MTJs are reset and written simultaneously, which requires less time compared with using a single MTJ to obtain the same number of random bits. Since $(N + 2) \times 5$ ns are needed to produce $N$ random bits, a generation speed of $\frac{N}{N+2} \times 200$ Mbit/s can be achieved. If $N$ is large enough, the read phase will dominate the operation and the speed will be about ~200 Mbit/s.

## IV. Symmetric MTJ-pair Design

### A. Basic idea and theory

In the parallel design, the accuracy of the switching probability is subject to the actual voltage and duration of the pulse applied to the MTJs, and PVT corners (process parameters, voltage and temperature). These global parameters will affect all MTJs in the circuit in the same way and to the same extent. In other words, each of the MTJs may produce random numbers with a probability biased to the same direction, either higher or lower than the expected 50%. In order to keep the probability precise, the pulses applied to the MTJs should be well controlled and the variations of the IC process should be insignificant.

However, note that, instead of producing random numbers by controlling pulses carefully, we can leverage the symmetries of multiple MTJs in the circuit and compare two independent

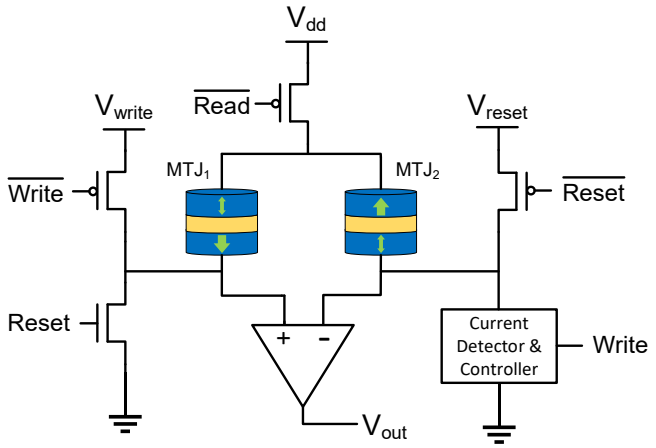



Fig. 5. Proposed TRNG with symmetric MTJ-pair.

random variables (such as switching times of two MTJs) which follow the same distribution to get a 50% probability. As long as the two random variables are equally affected by the variations, the distributions of them will be the same.

This method lies in the principal that there is equal probability that either variable is smaller than the other one, so the probability that the first variable is smaller than the second one is 50%. To prove it mathematically, suppose $X_1$ and $X_2$ are two independent random variables drawn from the same distribution $(X_1 = X_2 = X)$. The probability density function (PDF) of the random variables is $f(x)$, while the cumulative distribution function (CDF) is $F(x)$. The minimum and maximum possible values of $X$ are $\min(X)$ and $\max(X)$, respectively. The probability that $X_1$ is less than $X_2$ is

$$\begin{aligned} P(X_1 < X_2) &= \int_{a=\min(X)}^{\max(X)} \left[ f(a) \cdot \int_{b=a}^{\max(X)} f(b)\, db \right] da \\ &= \int_{a=\min(X)}^{\max(X)} \left[ f(a) \cdot F(b) |_{b=a}^{\max(X)} \right] da \\ &= \int_{a=\min(X)}^{\max(X)} \left[ f(a) \cdot \left(1 - F(a)\right) \right] da \\ &= \left[ F(a) - \frac{1}{2} F^2(a) \right] |_{a=\min(X)}^{\max(X)} \\ &= \frac{1}{2}. \end{aligned} \tag{4}$$

Therefore, it is proved that the result is fixed to 0.5 and is irrelevant to the actual distribution of the random variables $X_1$ and $X_2$, as long as they follow identical distributions.

An additional advantage is that the correlation problem of the MTJs is not a drawback anymore. Instead, a higher correlation will have improvements on the quality, which will be discussed in Section V.

### B. The proposed design

Fig. 5 shows the schematic of the proposed design. The core part of the design includes two MTJs of the same parameters connected in series to produce one bit random number. The principle idea is that both of the MTJs have equal probability of switching first, because the distributions of the switching time for each MTJ are independent and almost identical. As proven above, the probability that the switching time of the first MTJ is shorter than the second MTJ will be 50%.

The design works because of the following:

1. The two MTJs are connected in series, so the currents going through them are identical.

2. The parameters of the two MTJs are very similar to each other, so the two MTJs have the same properties such as the critical current and thermal stability factor.

3. The STT switching scheme ensures that the two MTJs switch individually and there's no correlation between them during the switching process.

However, it is impossible to know which MTJ switched first after the process if both of them switched. An alternative way is only allowing one of them to switch at a time. A current detector and controller is introduced to ensure only one of the two MTJs switches at a time in a vast majority of the cases.

### C. Generating procedures

To produce random numbers, the circuit needs to go through three phases: 1) a reset phase, 2) a write phase and 3) a read phase, with each phase taking 5 ns. One of the control signals *Reset, Write* and *Read* is driven high while the others are held low in each phase correspondingly.

#### 1) Reset

In the reset phase, *Reset* is high and other control signals are low. $MTJ_1$ and $MTJ_2$ in Fig. 5 are in series. The current flows from the free layer to the pinned layer for each MTJ until both MTJs are switched to the P state. $V_{reset}$ is high enough to ensure an almost deterministic switching to the P state. At the end of this phase, both MTJs are in the P state to be ready for the probabilistic switching in the write phase.

#### 2) Write

In the write phase, *Write* is high and other control signals are low. $MTJ_1$ and $MTJ_2$ are still in series, as well as the current detector and controller. $V_{write}$ induces a switching current going from the pinned layer to the free layer. Once any one of the two MTJs switches to the AP state, the current in the path decreases suddenly since the resistance of the AP state is higher than that of the P state and the voltage remains the same. The current detector and controller responds to this change and cut off the circuit path immediately. Once the circuit is cut off, there's no current going through the MTJs and the write phase comes to an end, so the MTJ that didn't switch will not switch anymore. In this case, one MTJ will be in the P state and the other one will be in the AP state.

However, it takes a small amount of time for the current detector and controller to cut off the circuit after the current changes, which cannot be completely ignored. If the second MTJ happens to switch just after the first one switching, both MTJs will be in the AP state.

Another case is that neither of the MTJs switches. Since the actual switching time follows a Gaussian distribution, but the pulse only lasts a finite period of time, there is the chance that neither MTJ switches before the pulse ends. If neither MTJ switches, both of them will remain in the initial P state.

In conclusion, there are actually three cases that might happen in the write phase:

- Case 1: only one MTJ switches and the two MTJs end up in different states.
- Case 2: Both MTJs switch.
- Case 3: Neither MTJ switches.

Case 1 is common while case 2 and case 3 are rare.

#### 3) Read

In the read phase, *Read* is high and other control signals are low. The current branches to the two MTJs and the path that has the MTJ with a higher resistance will have a lower current flowing through, and vice versa. A current comparator is used to determine the relative magnitude of

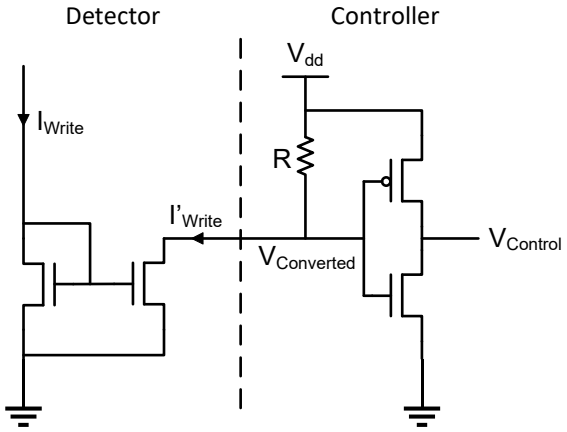



Fig. 6. Proposed schematics of the current detector and controller.

the currents. Finally, the digital output at $V_{out}$ will indicate the relative resistance of the MTJs. If $V_{out}$ is low, then there is a lower current in the left path, which means that $MTJ_1$ has the higher resistance. If $V_{out}$ is high, it means that $MTJ_2$ has the higher resistance.

For the cases that might happen in the write phase, the output is given slightly differently. When case 1 happens, the MTJs are in different states. The MTJ in the AP state must have a higher resistance than the one in the P state. Therefore, the output reveals which MTJ switched: if $MTJ_1$ switched, $V_{out}$ is low. If $MTJ_2$ switched, $V_{out}$ is high. When case 2 or case 3 happens, the MTJs are in the same state. However, the resistances of them are slightly different due to inevitable fabrication variations. The output will still reflect the relative resistance of the two MTJs: if the resistance of the $MTJ_1$ is higher, $V_{out}$ is low; otherwise, $V_{out}$ is high.

### D. *Discussions and evaluations about the feasibility*

Since the proposed design is based on the equal probability that either MTJ will switch first, we have to ensure that the probability of the rare cases 2 and 3 happening is small enough to ensure correct function.

#### 1) *Delay of the current detector and controller*

The delay of the current detector and controller should be short enough to prevent the second MTJ from switching as much as possible. The delay of the current detector and controller is defined as the time interval between when the first MTJ switches and when the circuit is cut off. The less the delay is, the less the probability that case 2 will happen. In our proposed design shown in Fig. 6, the detector is based on a current mirror which can duplicate the current in the path using only two transistors. The current mirror can also duplicate the current by a certain proportion to save energy. The controller is based on a current-voltage converter and an amplifier, which converts the duplicated current to a digital voltage signal. The amplifier then regulates the voltage and provides an output. The I-V converter can be simply implemented by a resistor, and the amplifier can be as simple as an inverter. Therefore, the change of current in the path is converted into the change of a digital control signal, and the signal is sent to cut off the circuit.

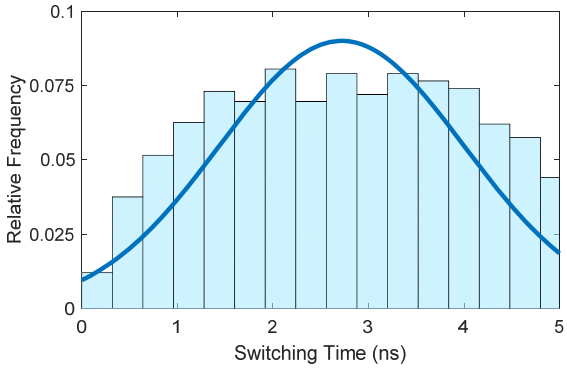



Fig. 7. The distribution of the actual switching time.

The simulation results show that the delay for the current detector and controller circuit described above is approximately 19.9 ps. Therefore, if the second MTJ happened to switch in less than 20 ps after the first one switched before the circuit is cut off, then both MTJs will end up in the AP state.

The probability of case 2 happening can be calculated theoretically as follows: the actual switching time can be taken to be a Gaussian distribution with a mean of $\mu = 2.72$ ns and a standard deviation of $\sigma = 1.28$ ns shown in Fig. 7. The distribution of the switching interval, which is the difference of the two independent Gaussian distributions, is also Gaussian. Since the two distributions are identical, the difference of the two distributions has a mean of $\mu' = \mu - \mu = 0$ and a standard deviation of $\sigma' = \sqrt{\sigma^2 + \sigma^2} = 1.81$ ns. Therefore, the probability that the switching interval lies between $\pm$ 20 ps is 0.88%.

#### 2) *Pulse width*

Another issue is that neither MTJ might switch since the pulse only lasts a finite period of time but the actual switching time is Gaussian distributed. Although increasing the duration of the pulse can reduce the probability that case 3 happens, the generation speed and the power consumption are also concerns. A moderate pulse length of 5-ns will keep this undesirable case happening little while maintaining a fast operation. The probability that one MTJ will not switch in 5 ns is 3.77% (Fig. 7). Since the switching times for the two MTJs are independent, the probability that neither of them switches is approximately $(3.77\%)^2 = 0.142\%$ in theory.

The simulation results verified the calculation by showing an approximately 0.9% probability of case 2 happening, and a less than 0.2% probability of case 3 happening. The total probability that the two rare cases 2 and 3 happen is approximately 1%.

## V. Discussions on Correlation Issues

Due to fabrication limitations, the parameters of the two MTJs are slightly different, and this will affect the probability that each of the three cases happens. The critical switching current is proportional to the size of the free layer [25],

$$I_{c0} = k \cdot t_{sl} \cdot Area\ , \qquad (5)$$

so the MTJ with a smaller size has a smaller critical current. The series connection of the two MTJs ensures that the currents ($I$) flowing through them are the same, and according to (2), the smaller MTJ will have a shorter mean switching time and thus is more likely to switch first.

For example, if $MTJ_1$ in Fig. 5 is slightly smaller than $MTJ_2$, then $MTJ_1$ is more likely to switch first, and the probability that $V_{out}$ is low is slightly higher than the probability that $V_{out}$ is high. The difference of the two MTJs leads to a probability bias that will undermine the quality of the random sequences.

However, the correlation in the MTJs actually helps to relieve this problem. Due to the correlations in the fabrication process, some parameters, such as the dimensions, of MTJs fabricated close to each other will be similar, and this leads to correlations in the mean switching time of the two MTJs [26-27]. As analyzed in Section IV.A, when two independent variables have identical distributions, the probability that one variable is smaller than the other is 50%. The more similar the two distributions are, the closer the probability will be towards 50%. Therefore, the more correlations the two MTJs have, the more similar the distributions of the switching time will be.

To analyze the correlation, a simplified mathematical model is built assuming that the mean switching time $\tau$ in (2) for the two MTJs follows a multivariate Gaussian distribution impacted by the fabrication effects. The distribution is determined by the mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

$$\boldsymbol{X} \sim \boldsymbol{\mathcal{N}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{6}$$

$$\boldsymbol{X} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}, \boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \tag{7}$$

We aim for a pair of MTJs with the same parameters, so the expected value of mean switching time $\mu$ is the same for both MTJs, as well as the standard deviation $\sigma$.

$$\begin{gathered} \mu_1 = \mu_2 = \mu, \boldsymbol{\mu} = \begin{pmatrix} \mu \\ \mu \end{pmatrix}, \\ \sigma_1 = \sigma_2 = \sigma, \sigma_1^2 = \sigma_1\sigma_2 = \sigma_2^2, \boldsymbol{\Sigma} = \sigma^2 \cdot \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \end{gathered} \tag{8}$$

Note that the "expected value of mean switching time" here is the mean value of the mean switching time of multiple devices, which is determined by the process parameters and design objectives before fabrication. The standard deviation of the mean switching time is relatively small (see Fig. 8). While the "mean switching time" is the mean value of the actual switching time of a certain device in multiple switching processes, which is determined by the material parameters and the size dimensions after fabrication. The standard deviation of the switching time is relatively large (see Fig. 7). In conclusion, the first distribution is for a set of devices, while the second distribution is for one device and is slightly different for each device.

Finally, there are only three independent parameters, namely the expected value of mean switching time $\mu$, the standard deviation of mean switching time $\sigma$, and the correlation coefficient $\rho$:

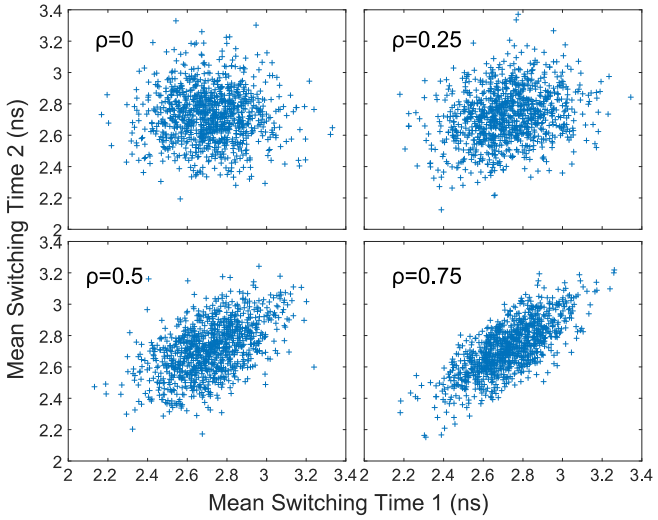



Fig. 8. The mean switching time for the two MTJs with different correlation coefficients ($\rho$).

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim \boldsymbol{\mathcal{N}}\left(\begin{pmatrix} \mu \\ \mu \end{pmatrix}, \sigma^2 \cdot \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}\right) \tag{9}$$

Under the simulation conditions, the expected value of mean switching time is $\mu$ = 2.72 ns, and the MTJs have a variation of 6.28% with respect to the expected value, which makes the standard deviation $\sigma = 6.28\% \times 2.72$ ns.

The population correlation coefficient $\rho$ reflects the correlations between the two MTJs. Since the correlation is non-negative, $0 \le \rho \le 1$. When $\rho = 0$, there's no correlation. And when $\rho = 1$, the two MTJs are identical. The correlation coefficient mainly depends on the limited accuracy during the fabrication process. For analytical purpose, we simulated different levels of correlation with $0 \le \rho \le 1$.

After all three parameters are set, the mean switching time for the two MTJs can be generated according to the multivariate Gaussian distribution. The random number generation will then be conducted based on the known distributions of the switching time of the two MTJs. The subfigures in Fig. 8 illustrate the correlation in two switching times for $\rho$ = 0, 0.25, 0.5 and 0.75, respectively, with 1000 samples for each case.

We have to point out that the correlations exist only during the fabrication process. After fabrication, the distributions of the switching time of both MTJs are determined. During each switching, there are no correlations between the two MTJs since they switch individually, nor are there correlations in the time domain since the switching is based on quantum effects.

## VI. The Quality Improvement Circuit (QIC)

If the distributions of the switching time for the two MTJs in the proposed circuit are less identical than expected, there will be a probability bias in the generated sequences. Actually, a drawback of nondeterministic physical phenomena based TRNGs is that the probability is more sensitive to various factors than PRNGs which are based on deterministic generation algorithms. An idea is to combine the advantages of both kinds of generators to get an unbiased true random number generator [28]. The true random source provides the

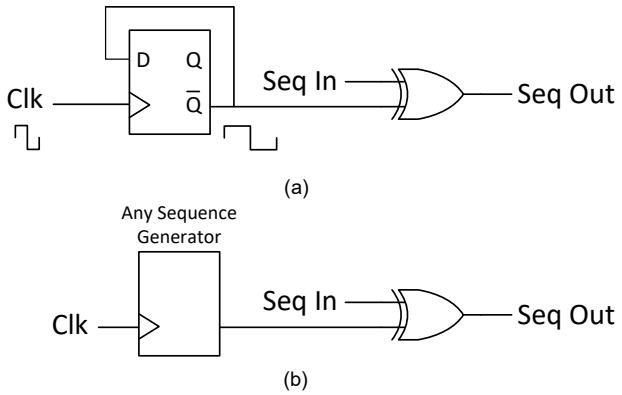



Fig. 9. (a) A simple implementation for flipping every other bit, and (b) Proposed quality improvement circuit.

nondeterministic property while the deterministic source ensures an unbiased frequency of 1's. Therefore, the probability bias issue can be mitigated by regulating the frequency of 1's occurred in the sequences closer to 50%, while keeping the true randomness in the combined generator.

### A. Theory

The simplest way to do the combination is to use XOR (or XNOR) gates [29], because other 2-input logic gates will not maintain the equal probability of 1's and 0's in the output given that the inputs are of equal probability. Using probabilistic logic, the theory of using XOR gates to improve the quality of random sequences in terms of frequency can be given [3].

If the inputs are independent, Boolean function $C = A \,\mathrm{XOR}\, B = \bar{A}B + A\bar{B}$ corresponds to $c = (1-a)\cdot b + a\cdot(1-b)$ where $a = P(A=1)$, $b = P(B=1)$ and $c = P(C=1)$. Suppose A is the sequence from the true random source with a probability bias $\delta$, so $a = 0.5+\delta$. Then suppose B is the sequence from a deterministic source used for improvement. We then have

$$\begin{aligned} c &= \big(1-(0.5+\delta)\big)\cdot b + (0.5+\delta)\cdot(1-b) \\ &= 0.5 + (1-2b)\delta. \end{aligned} \tag{10}$$

Since $0 < b < 1$, then $-1 < 1-2b < 1$, and finally $0.5-\delta < c < 0.5+\delta$. Therefore, the quality of the random sequences is surely improved from $a = 0.5+\delta$ to $0.5-\delta < c < 0.5+\delta$, even though the sequence from the deterministic source can be unbiased to some extent.

If the sequence from the deterministic source has a probability of exactly 0.5, then the result will be the best since $c = 0.5$ when $b = 0.5$.

### B. Simplest implementation

The randomness of the combined generator comes from the true random source, so the randomness is not required for the deterministic source. In fact, any sequence generator with nearly equal portions of 1's and 0's may help. Some simple sequence generators including binary counters and small LFSRs can contribute significantly to the improvement of the randomness quality.

The simplest sequence generator which meets the criteria is a 1-bit counter, and it can be simply implemented by a flip-flop (Fig. 9(a)). The output of a 1-bit counter is a sequence of alternating 1's and 0's. For an XOR gate, a 1 at one input will let the other input become its logical complement at the output, while a 0 at one input will let the other input remain its value at the output. So in short, the function of the XOR gate with a 1-bit counter is flipping every other bit in the original sequence. Intuitively, it can make the frequency of 1's in a biased sequence turn closer to 50% and break sub-sequences of consecutive 1's or 0's, which improves the randomness quality of both frequency-related and non-frequency-related properties.

### C. General QICs

With the analysis, we propose a quality improvement circuit (QIC) for the TRNG shown in Fig. 9(b). The QIC is composed of a sequence generator and an XOR gate. For the sequence generator, it could be any one producing approximately 50% of 1's and 50% of 0's in the sequence with a simple structure. For the XOR gate, one of the inputs is the original sequence from a TRNG (Seq In), and the other input is from the sequence generator. The output of the XOR gate is the sequence with improved randomness quality (Seq Out). The generator can be chosen with the consideration of the quality requirements and hardware cost. Examples include 1-bit counter, 2-bit counter and 4-bit LFSR, which will be tested and compared in the next section.

The QIC can be applied to any types of random number generators and it is a universal way to improve the quality of random sequences without complicated circuits.

## VII. Evaluations

In cryptography applications, such as Internet security, the typical key length is 256 bits for a Transport Layer Security or Secure Sockets Layer (TLS/SSL) cryptographic protocol [30]. Therefore, 256-bit sequences were generated using the proposed TRNGs.

The quality of the random sequences needs to be evaluated in aspects other than frequency to demonstrate the effectiveness of our approaches. Therefore, we applied the widely used statistical test suite National Institute of Standards and Technology (NIST) Special Publication 800-22 rev.1a [31].

The tests are based on statistical hypothesis testing. A significance level ($\alpha$) is chosen, and a decision is derived that accepts either the null hypothesis (the sequence tested is random) or the alternative hypothesis (the sequence tested is not random). Seven types with a total of 9 tests in the suite were selected to evaluate the sequences because other tests in the suite require millions of bits in a sequence. The tests are divided into two categories according to their relation with frequency:

- Frequency-related tests
  - Frequency (Monobits) Test
  - Frequency Test within a Block
  - Cumulative Sums (2 tests)

Frequency-related tests examine whether a sequence has a reasonable portion of 1's as a whole or in any sub-sequences.

- Non-frequency tests
  - Runs
  - Longest Run of Ones in a Block

- Approximate Entropy
- Serial (2 tests)

Non-frequency tests evaluate a sequence in aspects other than frequency such as the presence of oscillations and special patterns.

Only aiming at passing the frequency-related tests may lead to undesired results. For example, a sequence with alternating 1's and 0's (10101010…) will definitely pass all frequency-related tests, since it has perfect proportions of 1's and 0's in every part of the sequence. However, this sequence is absolutely not random. With the non-frequency tests, it is easy to exclude this sequence from the choices of good random sequences. First, there are too many runs (sub-sequences of consecutive 1's or 0's) in this sequence, or we could say the oscillation is too fast, which will let the sequence fail the tests of Runs and Longest Run. Second, the patterns of "10" and "01" occur far more frequently than the patterns of "00" and "11", which will let the sequence fail the Serial tests.

A confidence interval is used to determine whether a certain test is passed or not. To have a convincing conclusion, 1000 sequences were generated in each test: when the significance level is $\alpha = 0.01$ and the number of sequences tested is $m = 1000$, the confidence interval is $(1-\alpha) \pm 3 \times \frac{\alpha(1-\alpha)}{m} = 0.99 \pm 0.0094392$. Therefore, the pass rate needs to be greater than or equal to 0.981 to satisfy acceptable randomness. In other words, at least 981 in 1000 sequences should pass the test. Note that to validate the randomness quality of a generator, all of the 9 tests must pass with no less than 0.981 pass rates, and any average pass rates are for illustration and comparison purposes only.

### A. *Results of the parallel design*

For each $N$ value, the proposed generation procedure was repeated $\frac{256}{N}$ times, and each MTJ was used $\frac{256}{N}$ times to generate $\frac{256}{N}$ random bits, where $N$ is the number of MTJs in the array. After one sequence of 256 bits is generated, a new set of $N$ MTJs is used to generate the next sequence. Altogether 1000 sequences were generated for each $N$ value.

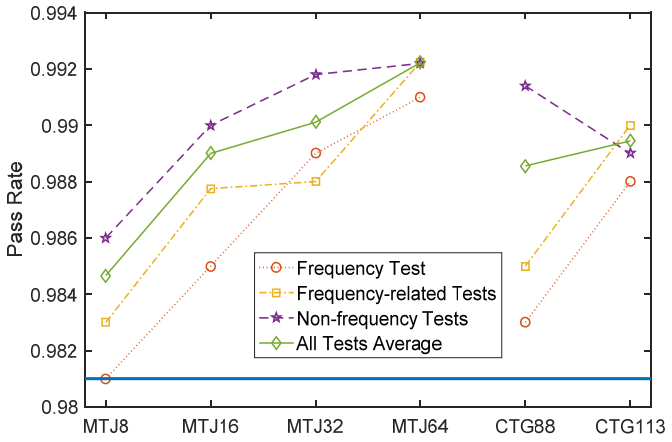



Fig. 10. Statistical quality pass rates of four MTJ-based TRNGs and two combined Tausworthe generators. MTJN denotes N parallel MTJs used in the proposed design, and CTG88 and CTG113 are combined Tausworthe generators with a period of nearly $2^{88}$ and $2^{113}$, respectively.

The four curves at the left side of Fig. 10 show the pass rate trends for different categories of tests, and illustrate the quality improvement of the generators with increasing number of MTJs used. The bold horizontal line is the threshold of 0.981 for passing the tests (same for Figs. 11 to 12). When using at least 16 MTJs, the pass rates for all tests are no less than 0.981, which means that the corresponding generators can pass all 9 randomness tests. Therefore, it was shown by the statistical test suite that using at least 16 MTJs in the proposed TRNG can generate high-quality 256-bit random sequences.

In addition, the combined Tausworthe generators (CTGs) and LFSRs were tested for comparison purposes [32-33]. The results show that the simple Tausworthe generator with a period of $2^{28}$-1 and the LFSR with a period of $2^{52}$-1 behave quite poorly, with an overall pass rate of 0.89 and 0.96, respectively. However, with the more complex combined Tausworthe generators, the statistical quality is improved. The comparison results are shown in Fig. 10: using 16 MTJs can produce random sequences with a similar quality of randomness as using any of the CTGs, while using 32 and 64 MTJs will lead to better results. However, the test suite can only evaluate the statistical properties of the random sequences. The advantages of a TRNG over a PRNG are not shown from the numerical results: the MTJ-based generators generate true random numbers and are better for cryptographic applications.

As a trade-off between quality, speed and area, using 16 MTJs is sufficient to satisfy basic quality concerns while providing a fast generation speed. 32 or more MTJs can be implemented in applications that require a higher security level where a better quality or a faster speed is needed. However, more hardware resources are required as the number of MTJs increases.

### B. *Results of the symmetric MTJ-pair design*

For each correlation coefficient, the proposed generation procedure was repeated 256 times to obtain a 256-bit sequence. After one sequence is generated, a new pair of MTJs is used to generate the next sequence. Altogether 1000 sequences were generated for each $\rho$ value.

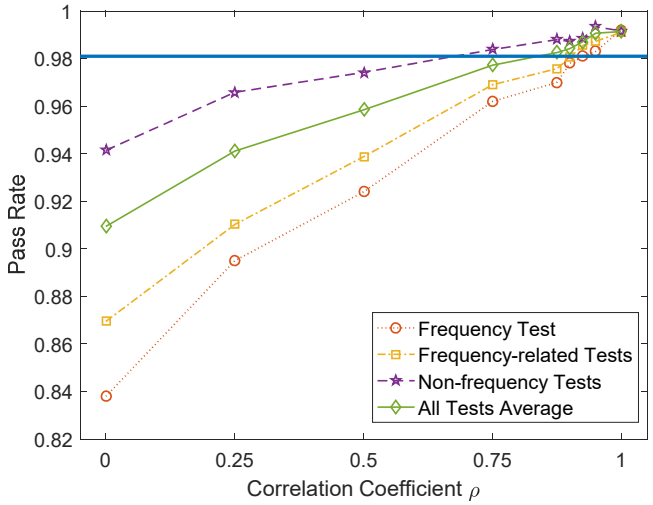



Fig. 11. Statistical quality pass rates of the proposed MTJ-pair design with different correlation coefficients.

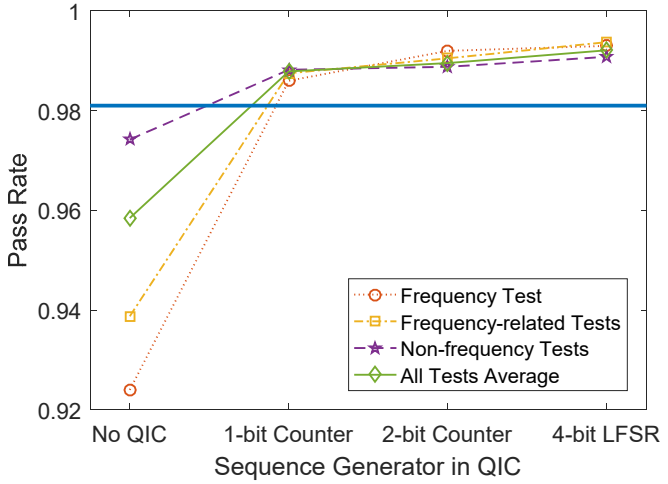



Fig. 12. Statistical quality pass rates of the proposed MTJ-pair design with different QICs ($\rho = 0.5$).

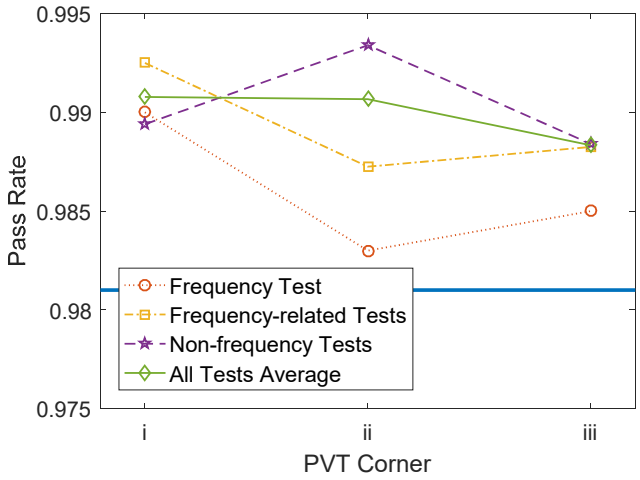



Fig. 13. Statistical quality pass rates of the proposed MTJ-pair design with different PVT corners in Table II.

Fig. 11 shows the pass rate trends for different categories of tests. The four curves illustrate the quality improvement of the generators with an increasing large correlation coefficient, showing that this design is especially suitable for MTJs with highly correlated physical properties. Actually, all 9 tests are passed when $\rho \geq 0.925$. However, if the correlation of the MTJs is less significant and some tests fail, the quality improvement circuit can be added.

For example, when $\rho = 0.5$, the sequences fail the frequency test with a pass rate of 0.924. Moreover, none of the frequency-related tests are passed and only two of the non-frequency tests are passed. However, with the implementation of the QIC, the randomness quality improves significantly as shown in Fig. 12. Even combined with the simplest 1-bit counter, the output sequences can pass all 9 tests. The use of 2-bit counter or 4-bit LFSR will improve the quality even more, although the additional quality improvement is relatively small.

To test the variation-resilience of the design, experiments were conducted with different combinations of the process parameters, voltage and temperature. First, the mean switching time with different PVT corners were obtained in simulation,

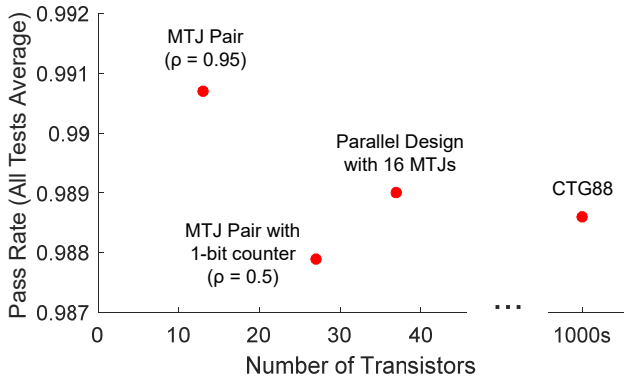



Fig. 14. The overall comparison of the two proposed designs and a benchmark PRNG in terms of quality and hardware cost.

TABLE II
PVT CORNER TEST

| PVT corners | Mean switching time $\mu$ ($ns$) |
|---|---|
| i. FF, high voltage (1.1x, 825 mV), 0 °C | 2.66 (lowest) |
| ii. TT, nominal voltage (750 mV), 27 °C | 2.72 |
| iii. SS, low voltage (0.9x, 675 mV), 70 °C | 2.88 (highest) |

TABLE III
PERFORMANCE COMPARISONS

| | The parallel design (with 16 MTJs) | The MTJ-pair design | Design in [10] |
|---|---|---|---|
| Technology | 28 nm | 28 nm | 90 nm |
| Frequency | 177.8 MHz | 66.7 MHz | 66.7 MHz |
| Area Estimation | 7.64 $\mu m^2$ | 3.84 $\mu m^2$ | Large |
| Energy | 0.64 pJ/bit | 0.81 pJ/bit | Unknown |
| Statistical Tests | Passed | Passed with QIC | Unknown |

assuming a write pulse width of 5 ns, and the simulation results are shown in Table II. Next, with the correlation coefficient $\rho$ set to 0.95, the random sequences are generated under different PVT corners. Fig. 13 shows that with all combinations of the process parameters, voltage and temperature, the generated sequences can pass all tests with similar pass rates. Therefore, it is confirmed that the PVT corners will not significantly affect the randomness quality, and the proposed design has an intrinsic resistance to all major variations in the circuit.

If the operating voltage varies from 0.9 to 1.1 times the nominal voltage, a TRNG based on a single MTJ switching will have a probability bias of more than $\pm$ 10% (see Fig. 3), which will severely undermine the randomness quality. Compared with other TRNG designs based on a single MTJ switching, the main advantage of the MTJ-pair design is its resistance to variations. Since all variations will affect both MTJs in the circuit to almost the same extent, the difference between the properties of the two MTJs will still be small. The random number generation depends on the similarity of the statistical distribution of the two MTJs instead of the actual value of a certain parameter, so the quality of the generated sequences will remain unimpaired (as long as the variation is moderate keeping the mean switching time within the expected range).

### C. Comparisons

To have an overall idea of randomness quality and hardware properties, this part provides comprehensive comparisons of

the proposed TRNG designs with other random number generators from the literature.

The trade-offs in terms of quality and the number of transistors are displayed in Fig. 14. Note that the MTJs are fabricated above the metal layers without occupying additional chip area in integrated circuit design, so the number of transistors is a good representation of the area.

As shown in Fig. 14, the symmetric MTJ-pair design saves more hardware than the parallel design when producing the same quality level. Moreover, when compared with other random number generators, both of the proposed designs are very compact without compromising randomness quality. Actually, the comparable PRNGs, such as CTGs, require much more hardware resources, since they contain hundreds of shift registers and other cells; other MTJ-based TRNGs with post-processing or real-time tracking circuits also have much more hardware overhead.

The hardware simulation results for both proposed designs are summarized in Table III and are compared with those in [10], which include a probability-locked loop. Our designs are energy-efficient (less than 1 pJ/bit) with a high generation speed (tens of MHz). Note that the energy cost of the required voltage controller has been omitted from the comparison in Table III. The voltage controller for an array of STT-MTJs could be designed by adapting a standard on-chip reference voltage generator circuit for semiconductor memories [34].

Some other main characteristics of the two proposed designs include:

1) *The parallel design*
   This design has a high frequency and it can be adjusted according to the quality requirements by choosing the proper number of parallel MTJs. The parallel design has a speed advantage because of the parallel resets and writes.
2) *The symmetric MTJ-pair design*
   This design is especially suitable for circuits with significant variations and MTJs with high correlations. In other words, this design is highly robust. Also, it saves more hardware compared with the first design.

In conclusion, each of the designs has its own advantages. However, the symmetric MTJ-pair design can maintain a good behavior under various PVT corners and the operations are simpler with fewer control signals.

## VIII. Conclusions

Two designs of true random number generators based on multiple magnetic tunnel junctions are proposed in this article, and both of the designs are variation-resilient. The parallel structure averages the biased probabilities of each single MTJ, and the MTJ-pair leverages the symmetry of the two MTJs fabricated close to each other. Each of the designs has some specific advantages: the parallel design has a higher generation speed while the MTJ-pair design is more robust. The designs are validated in a 28-nm CMOS process by Monte Carlo simulation with a compact model of the MTJ. It is verified by a statistical test suite that both designs can generate high-quality random sequences for cryptography applications. The designs save much hardware compared with pseudo-random number generators and other MTJ-based random number generators. Hardware simulations show that the designs are energy-efficient (less than 1 pJ/bit) with high generation speeds (177.8 or 66.7 MHz).

## References


[1] A. Botta, W. de Donato, V. Persico and A. Pescapé, "Integration of Cloud computing and Internet of Things: A survey", *Future Generation Computer Systems*, vol. 56, pp. 684-700, 2016.

[2] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL: CRC Press, 1997.

[3] J. Han, H. Chen, J. Liang, P. Zhu, Z. Yang and F. Lombardi, "A Stochastic Computational Approach for Accurate and Efficient Reliability Evaluation", *IEEE Trans. Computers*, vol. 63, no. 6, pp. 1336-1350, 2014.

[4] A. Alaghi and J. Hayes, "Survey of Stochastic Computing", *ACM Trans. Embedded computing systems*, vol. 12, no. 2, pp. 1-19, 2013.

[5] S. Mathew, S. Srinivasan, M. Anders, H. Kaul, S. Hsu, F. Sheikh, A. Agarwal, S. Satpathy and R. Krishnamurthy, "2.4 Gbps, 7 mW All-Digital PVT-Variation Tolerant True Random Number Generator for 45 nm CMOS High-Performance Microprocessors", *IEEE J. Solid-State Circuits*, vol. 47, no. 11, pp. 2807-2821, 2012.

[6] K. Yang, D. Fick, M. Henry, Y. Lee, D. Blaauw and D. Sylvester, "A 23Mb/s 23pJ/b fully synthesized true-random-number generator in 28nm and 65nm CMOS", *2014 IEEE International Solid-State Circuits Conference (ISSCC)*, 2014.

[7] P. Knag, W. Lu and Z. Zhang, "A Native Stochastic Computing Architecture Enabled by Memristors", *IEEE Trans. Nanotechnology*, vol. 13, no. 2, pp. 283-293, 2014.

[8] Y. Wang, W. Wen, H. Li and M. Hu, "A Novel True Random Number Generator Design Leveraging Emerging Memristor Technology", *The 25th Great Lakes Symposium on VLSI (GLSVLSI)*, 2015.

[9] N. Rizzo, F. B. Mancoff, R. Whig, K. Smith, K. Nagel, T. Andre, P. G. Mather, S. Aggarwal, J. M. Slaughter, D. Mitchell, and S. Tehrani, "Toggle and spin torque: MRAM at Everspin Technologies," *Proc. Non-Volatile Memories Workshop*, 2010

[10] S. Oosawa, T. Konishi, N. Onizawa and T. Hanyu, "Design of an STT-MTJ based true random number generator using digitally controlled probability-locked loop", *2015 IEEE 13th International New Circuits and Systems Conference (NEWCAS)*, 2015.

[11] Y. Wang, H. Cai, L. A. B. Naviner, J. Klein, J. Yang and W. Zhao, "A novel circuit design of true random number generator using magnetic tunnel junction," *2016 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH)*, 2016.

[12] W. H. Choi, Y. Lv, J. Kim, A. Deshpande, G. Kang, J.-P. Wang and C. Kim, "A Magnetic Tunnel Junction based True Random Number Generator with conditional perturb and real-time output probability tracking", *2014 IEEE International Electron Devices Meeting*, 2014.

[13] A. Fukushima, T. Seki, K. Yakushiji, H. Kubota, H. Imamura, S. Yuasa and K. Ando, "Spin dice: A scalable truly random number generator based on spintronics", *Appl. Phys. Express*, vol. 7, no. 8, p. 083001, 2014.

[14] Y. Qu, J. Han, B. Cockburn, Y. Zhang, W. Zhao and W. Pedrycz, "A True Random Number Generator based on Parallel STT-MTJs," *2017 Design, Automation & Test in Europe (DATE)*, 2017.

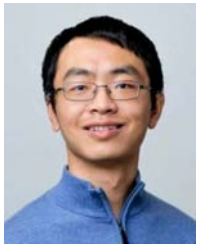
[15] W. Kang, Z. Li, J. Klein, Y. Chen, Y. Zhang, D. Ravelosona, C. Chappert and W. Zhao, "Variation-Tolerant and Disturbance-Free Sensing Circuit for Deep Nanometer STT-MRAM", *IEEE Trans. Nanotechnology*, vol. 13, no. 6, pp. 1088-1092, 2014.

[16] D. Zhang, L. Zeng, Y. Qu, Y. Zhang, M. Wang, W. Zhao, T. Tang and Y. Wang, "Energy-efficient neuromorphic computation based on compound spin synapse with stochastic learning", *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015.

[17] R. Sbiaa, H. Meng and S. Piramanayagam, "Materials with perpendicular magnetic anisotropy for magnetic random access memory", *physica status solidi (RRL) - Rapid Research Letters*, vol. 5, no. 12, pp. 413-419, 2011.

[18] S. Ikeda, K. Miura, H. Yamamoto, K. Mizunuma, H. Gan, M. Endo, S. Kanai, J. Hayakawa, F. Matsukura and H. Ohno, "A

perpendicular-anisotropy CoFeB–MgO magnetic tunnel junction", *Nature Materials*, vol. 9, no. 9, pp. 721-724, 2010.
[19] D. C. Worledge et al, "Spin torque switching of perpendicular Ta|CoFeB|MgO-based magnetic tunnel junctions", *Applied Physics Letters*, Vol. 98, pp. 022501. 2, 2011.
[20] G. Hu, et al, "Key Parameters Affecting STT-MRAM Switching Efficiency and Improved Device Performance of 400° C-Compatible p-MTJs", *IEEE IEDM*, pp. 38.3.1-38.3.4, 2017.
[21] J. Swerts, et al, "Solving the BEOL compatibility challenge of toppinned magnetic tunnel junction stacks", *IEEE IEDM*, pp. 38.6.1-38.6.4, 2017.
[22] M. Wang, et al, "Current-induced magnetization switching in atom-thick tungsten engineered perpendicular magnetic tunnel junctions with large tunnel magnetoresistance", *Nature Communications*, vol. 9, article no. 671, Feb. 14, 2018.
[23] M. Martin, B. Dlubak, R. Weatherup, H. Yang, C. Deranlot, K. Bouzehouane, F. Petroff, A. Anane, S. Hofmann, J. Robertson, A. Fert and P. Seneor, "Sub-nanometer Atomic Layer Deposition for Spintronics in Magnetic Tunnel Junctions Based on Graphene Spin-Filtering Membranes", *ACS Nano*, vol. 8, no. 8, pp. 7890-7895, 2014.
[24] O. Manos, A. Böhnke, P. Bougiatioti, R. Klett, K. Rott, A. Niesen, J. M. Schmalhorst, and G. Reiss, "Tunneling magnetoresistance of perpendicular CoFeB-based junctions with exchange bias," *J. Appl. Phys.*, vol. 122, no. 10, 2017.
[25] Y. Zhang, B. Yan, W. Kang, Y. Cheng, J. Klein, Y. Zhang, Y. Chen and W. Zhao, "Compact Model of Subvolume MTJ and Its Design Application at Nanoscale Technology Nodes", *IEEE Trans. Electron Devices*, vol. 62, no. 6, pp. 2048-2055, 2015.
[26] T. H. P. Chang, "Proximity effect in electron-beam lithography," *J. Vac. Sci. Technol.*, vol. 12, no. 6, pp. 1271–1275, 1975.
[27] D. Güttler, R. Grötzschel, and W. Möller, "Lateral variation of target poisoning during reactive magnetron sputtering," *Appl. Phys. Lett.*, vol. 90, no. 26, 2007.
[28] D. Eastlake, S. Crocker and J. Schiller, "Randomness Recommendations for Security", 1994. [Online]. Available: http://tools.ietf.org/html/rfc1750
[29] R. B. Davies, "Exclusive OR (XOR) and hardware random number generators," Tech. Rep., 2002. [Online]. Available: http://www.robertnz.net/pdf/xor2.pdf
[30] T. Dierks, "The transport layer security (TLS) protocol version 1.2.", 2008. [Online]. Available: http://tools.ietf.org/html/rfc5246
[31] National Institute of Standards and Technology, Special Publication 800-22 rev.1a, 2010. [Online]. Available: http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html
[32] A. Alimohammad, S. Fard, B. Cockburn and C. Schlegel, "On the efficiency and accuracy of hybrid pseudo-random number generators for FPGA-based simulations", *2008 IEEE International Symposium on Parallel and Distributed Processing*, 2008.
[33] P. L'Ecuyer, "Maximally equidistributed combined Tausworthe generators", *Mathematics of Computation*, vol. 65, no. 213, pp. 203-214, 1996.
[34] K. Itoh, *VLSI Memory Chip Design,* Berlin: Springer-Verlag, 2001.



**Yuanzhuo Qu** received the B.Eng. degree in Electronic and Information Engineering from Beihang University, Beijing, China in 2015. In 2017 he completed the M.Sc. degree from the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. Mr. Qu is presently working as a systems analyst in Edmonton, AB, Canada.

**Bruce F. Cockburn** (S'86-M'90) received the B.Sc. degree in engineering physics in 1981 from Queen's University, Kingston, ON, Canada. In 1985 and 1990 he received the M.Math. and Ph.D. degrees, respectively, in computer science from the University of Waterloo, Waterloo, ON. He is presently a Professor in the Department of Electrical and Computer Engineering at the University of Alberta, Edmonton, AB, Canada. His research interests include FPGA-based hardware accelerators, parallel computing, stochastic and approximate computing, and bioinformatics.

**Zhe Huang** received the B.Eng. degree in Electronic and Information Engineering from Beihang University, Beijing, China, in 2018. He is pursuing an M.S. degree of Microelectronics at Beihang University. His research interests are focused on the design of TRNG circuits and stochastic computing.

**Hao Cai** (M'15) received the PhD degree in electrical engineering from Télécom Paristech, Université Paris-Saclay, France, in 2013. From 2012 to 2014, he was involved in the European EUREKA program CATRENE-RELY for high reliability nanoscale integrated circuits and systems. He is currently an associate professor at the National ASIC System Engineering Center, Southeast University, Nanjing, China. His research interests include circuit techniques for emerging technologies, ultra-low-power VLSI, and reliability-aware design.

**Yue Zhang** (S'11–M'14) received the B.S. degree in optoelectronics from Huazhong University of Science and Technology, Wuhan, China, in 2009; he then received the M.S. and Ph.D. degrees in microelectronics from the University of Paris-Sud, France, in 2011 and 2014, respectively. He is currently an Associate Professor with Beihang University, China.

His current research focuses on emerging non-volatile memory technologies and hybrid low-power circuit designs.

**Weisheng Zhao** (M'06–SM'14) received the Ph.D. degree in physics from the University of Paris-Sud, France in 2007. In 2009, he joined the CNRS as a Tenured Research Scientist. Since 2014, he has been Distinguished Professor with Beihang University, Beijing, China. His interests include the hybrid integration of nanodevices with CMOS circuit and new non-volatile memory (40-nm technology node and below) like MRAM circuit and architecture design.

**Jie Han** (SM'16) received the B.Sc. degree in electronic engineering from Tsinghua University, Beijing, China, in 1999 and the Ph.D. degree from the Delft University of Technology, Delft, The Netherlands, in 2004. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His research interests include approximate computing, stochastic computation, reliability and fault tolerance, nanoelectronic circuits and systems, and novel computational models for nanoscale and biological applications.