On the Theoretical Analysis of Memristor based True Random Number Generator

Mesbah Uddin, Md. Sakib Hasan, and Garrett S. Rose

Proceedings of the ACM Great Lakes Symposium on VLSI (GLSVLSI), May 2019.

Citation Information (BibTex):

```
@ARTICLE{MTRNG-GLSVLSI:Mesbah2019,
 author="Mesbah Uddin, Md. sakib Hasan, and Garrett S. Rose
 title="On the Theoretical Analysis of Memristor based True Random Number Generator",
 journal="Proceedings of the ACM Great Lakes Symposium on VLSI (GLSVLSI)",
 month="May",
 year="2019",
 publisher = "ACM",
}
```

On the Theoretical Analysis of Memristor based True Random Number Generator

Mesbah Uddin
The University of Tennessee
Knoxville, Tennessee
muddin6@vols.utk.edu

Md Sakib Hasan
The University of Tennessee
Knoxville, Tennessee, USA
mhasan4@vols.utk.edu

Dr. Garrett S. Rose
The University of Tennessee
Knoxville, Tennessee, USA
garose@utk.edu

ABSTRACT

Emerging nano-devices like memristors display stochastic switching behavior which poses a big uncertainty in their implementation as the next-generation CMOS alternative. However, this stochasticity gives opportunities to build circuits for hardware security. There are several examples in the literature where the stochastic switching time of memristors are used as the source of entropy to build true random number generators (TRNG). In many important applications, software-based pseudo-random numbers are not enough as they can be modeled to some degree and thus TRNGs become necessary. In this work, we have analyzed TRNG designs that utilize memristors' switching time and evaluated them in varying operating conditions and with process variation in mind. Specifically, we have shown mathematically how large process variation and strong temperature and voltage dependence of memristors can degrade the performance of these TRNGs. Therefore, we have proposed a new TRNG based on difference between high resistance states (HRS) of a pair of memristors. With theoretical analysis using simple probabilistic mathematics, we have evaluated our proposed method with existing ones and show that our proposed design works better even in unfavorable environmental conditions and in the presence of large process variation.

CCS CONCEPTS

• **Security and privacy** → **Embedded systems security**; **Hardware-based security protocols**.

KEYWORDS

True random number generator, TRNG, RNG, memristor, Hardware security, memristor based RNG

ACM Reference Format:
Mesbah Uddin, Md Sakib Hasan, and Dr. Garrett S. Rose. 2019. On the Theoretical Analysis of Memristor based True Random Number Generator. In *Proceedings of ACM Woodstock conference (GLSVLSI'19).* ACM, New York, NY, USA, Article 4, 7 pages. https://doi.org/10.475/123_4

1 INTRODUCTION

With the emergence of billions of smart interconnected devices, hardware security is becoming more important that ever. Most, if not all security implementations require secret keys. Key generation software usually follows some complex algorithms to generate their secret keys randomly. However, these software-generated keys are at best pseudo-random in nature and can be modeled or predicted with advanced machine learning tools. Therefore, there is a growing need for true random number generators (TRNGs) that can produce a stream of true random bits from hardware for cryptographic applications [5]. Using some stochastic and random physical phenomena within a circuit, these TRNGs produce a stream of random bits. True random numbers (TRN) are different from pseudo random numbers (PRN) where a seed is usually needed and the same stream of bits are produced if the initializing seed remains the same. A TRNG should ideally be insensitive to environmental variation, i.e. the source of entropy that produces the random bits should not be a strong function of temperature or supply voltage. Process variation should also have a small impact so that it doesn't disturb the randomness of the TRNG circuit. Naturally random circuit phenomena are used to build TRNGs. These phenomena include clock jitter, random telegraph noise (RTN), thermal noise, metastable state, any quantum phenomena, etc.

Recently, stochasticity in novel nano-devices like memristors [11] is being explored to build TRNGs [4, 15]. Memristors display probabilistic switching behavior, i.e. their switching time or memristive states can have different values at different periods of time [9]. Researchers have tried to use this as source of randomness to build different hardware security primitives [7, 12, 15]. In [15], probability distribution of memristors's OFF and ON states based on the switching time is used to design the MTRNG (memristor-based TRNG). In [6], researchers have utilized a differential readout scheme to harvest the RTN (random telegraph noise) in memristors along with LFSR (linear feedback shift register) to implement a TRNG. The differential nature of this circuit improves performance against temperature variation and supply voltage noise. Switching variability in memristor's set process is used to extract random numbers in [1]. In [4], authors have proposed TRNG design where the random switching time of a diffusive memristor is used as the source of entropy and is shown to produce true random numbers without any post-processing. This diffusive memristor is different than common metal-oxide memristors, does not require forming and reverts to its high resistance state spontaneously. However, this self-OFF switching ($\approx 1ms$) makes these memristors very slow.

The main purpose of this work is to analyze the underlying theory and mathematical basis for memristor based random number generators. We have analyzed a memristor-based TRNG design and formulated how temperature and supply voltage change or process variation could affect the TRNG performance. To improve robustness against process variation and environmental changes,

GLSVLSI'19, May 2019, Washington DC, USA

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

we have suggested a TRNG design that leverages the high resistance state (HRS) variability of a memristor. This design uses relative measurements, i.e. the HRS of one memristor is compared with another to generate a random bit. Since HRS usually has shown higher variability compared to other memristor characteristics, we have chosen HRS as the source of stochasticity for our design. The differential nature of this design makes it robust against process and environmental variation.

The contributions of this paper include (1) formulation of conventional memristor based TRNG design, (2) providing a mathematical insight to measure changes in performance with outside effects, (3) enumerating the data bias in TRNG output from temperature and supply voltage change, (4) proposing a new TRNG design, (5) evaluation of both existing and proposed design against environmental changes, and (6) evaluation of both design against process variation.

The paper is organized as follows: section 2 first presents a short introduction to memristor and its stochastic behavior for common memristor based TRNG designs. Section 3 derives the relationship of existing TRNG design with temperature shift and supply voltage and process variation. Section 4 presents and discusses our proposed design and again derives the relationship of this design with temperature, supply voltage and process variation. Section 5 presents the simulation results and explains the difference between these two designs. Finally, future prospects and concluding remarks are provided in section 6 and 7, respectively.

2 BACKGROUND

2.1 Introduction to Memristor

The memristor or ReRAM/RRAM (resistive random access memory) is one of the most promising post Moore-era devices because of their non-volatility, CMOS compatibility, low energy operation among other features [11]. Metal oxide memristors (metal-insulator-metal or MIM layer) are more common among various types of memristors. They are usually bi-polar, i.e. they have two stable resistive states namely high resistance state (LRS) or the OFF state and the low resistance state (LRS) or the ON state. Applying a minimum threshold voltage for a minimum amount of time switches a memristor from LRS to HRS or from HRS to LRS depending on the direction of applied voltage. The minimum threshold voltage and minimum time to switch from HRS to LRS are called positive threshold voltage and positive switching time, respectively, where from LRS to HRS, they would be called negative threshold and negative switching time, respectively. The transition from LRS to HRS is known as RESET where the transition from HRS to LRS is called SET. As memristors display RTN (random telegraph noise), their final HRS and LRS values along with threshold voltages and switching times are stochastic over time. Also, due to their nano-scale size and relatively newer fabrication process, memristors display a large die-to-die process variation [10, 14].

2.2 Stochasticity in Memristor

As mentioned before, in metal-oxide memristors, threshold behavior is observed. This means when the applied voltage is larger than a threshold voltage for at least the minimum switching time, the memristor switches its state from one memristive (or resistive) state to another. Applying a minimum voltage of opposite polarity for at least another specific period of time would then revert the memristor back to its previous state. Because of large manufacturing process variation, memristors display a wide range of characteristics. Also, because of their underlying physical nature of switching mechanism, they also display varied characteristics in time. Thus their switching process is stochastic i.e. the value of their memristive states, switching time and threshold voltages can display variation across time. Over a larger number of clock cycles, their stochastic behavior can be approximated as Gaussian or Normal distribution. A TRNG tries to extract the uncertainty in their switching over time to generate stream of random bits.

3 EFFECT OF ENVIRONMENTAL AND PROCESS VARIATION ON CONVENTIONAL DESIGN

As mentioned in earlier sections, most of the existing memristor based TRNG designs take advantage of the stochastic switching time (or the RTN). Memristors display cycle to cycle variation in its characteristics. The switching time (positive or negative) of a memristor can be approximated to have a normal distribution over many cycles. This is formulated as:

$$\mathbf{t_{swn}} = N(\mu_{tswn}, \sigma_{tswn}) \tag{1}$$

and,

$$\mathbf{P}(t \leq \mu_{tswn}) = \mathbf{P}(t > \mu_{tswn}) = 0.5 \tag{2}$$

To reiterate, equation 1 describes that the switching time of a memristor is a Gaussian (Normal) random variable with mean μ_{tswn} and standard deviation σ_{tswn}. Therefore, 2 expresses the probability that a random switching time, t ϵ t_{tswn} to be less than or equal to the mean is 0.5 or 50% according to the property of a Gaussian distribution. Thus if we determine the mean of that switching time (μ_{tswn}) and apply a voltage greater than its threshold voltage to a memristor repeatedly for many cycles, then half of the time, memristor would switch and the other half of the time, it would not. This is the basis for most memristor based RNG where memristors's stochastic switching time is the source of entropy.

This method requires a precise pulse of pulse width approximately equal to the mean switching time. Any clock jitter/skew or other source of noise could change this pulse width which may drastically change the overall switching probability. The switching time also depends heavily on the applied voltage magnitude across the memristor. Thus, any power supply noise would also strongly affect the switching time. Since the memristor's threshold voltages and memristive states are functions of temperature, temperature change would also affect the switching time and thus the overall switching probability. Thus any change in any of these factors would bring a change in the probability of switching and thus equation 2 would not hold for the changed distribution. Then the produced bit stream from that memristor could be heavily biased towards either a '1' or a '0'. In the next few subsections, we are going to formulate the change in switching time distribution for these factors.

3.1 Supply voltage variation

To simplify our analysis and to get an approximate idea how voltage variation can affect the random number generator performance, we have considered a linear model for a memristor [8, 13]. The RESET of a memristor (i.e. from LRS to HRS) is governed by this equation [13]:

$$M(t_{i+1}) = M(t_i) + \frac{R_{diff}\Delta t V(t_{i+1})}{t_{swn}V_{tn}}, \tag{3}$$

where, t_{swn} and V_{tn} are negative switching time and negative threshold voltage of a memristor, respectively. R_{diff} is the difference between HRS and LRS. $M(t)$ presents the memristance and $V(t)$ is the applied voltage across the memristor at a time instant, t. Equation 3 can be written as:

$$\Delta M = \frac{R_{diff}}{t_{swn}V_{tn}} * \Delta t, \tag{4}$$

By integrating this equation 4, we can establish a relationship between switching time and applied voltage. Let's assume that the applied voltage remains the same for one switching cycle, from equation 4, we can estimate the actual switching time for any applied voltage:

$$\int_{LRS}^{HRS} dM = \frac{R_{diff}\left|V_{applied}\right|}{t_{swn}V_{tn}} * \int_{0}^{t_{sw,actual}} dt \tag{5}$$

$$\implies t_{sw,actual} = t_{swn} * \frac{V_{tn}}{\left|V_{applied}\right|} \tag{6}$$

The resulting relationship between applied voltage and actual time to switch is found to be proportional which is expected from a linear switching model of a memristor. If the change in applied voltage is limited (max ±10% here), then a linear equation should provide reasonable accuracy. If $V_{applied}$ equals to the threshold V_{tn}, then the actual switching time is equal to the minimum switching time. However, from equation 6, a ±10% change in $V_{applied}$ from V_{tn} would result in a (≈ -9%,+11%) change in actual switching time.

3.2 Temperature variation

Temperature directly affects memristive states and threshold voltages and thus in turn affect the actual switching time. From [2, 3], we can see threshold voltage magnitude decreases with increasing temperature within the range $\approx[0, 100]^0$C and we can approximate the change with a linear equation as:

$$V_{tn}(\theta) = V_{tn0}[1 + \alpha_{Vtn}(\theta - \theta_0)], \tag{7}$$

where, V_{tn0} is the threshold voltage at room temperature, θ_0=25^{0}C. From [3], the temperature coefficient for negative threshold voltage is calculated as ≈-0.007/^{0}C. Therefore, for a $\Delta\theta$ temperature change, the new V$_{tn}$ would be:

$$V_{tn}(\theta) = [1 - 0.007 * \Delta\theta] * V_{tn0}, \tag{8}$$

The modified switching time would then be:

$$t_{sw,actual} = t_{swn} * \frac{[1 - 0.007 * \Delta\theta] * V_{tn0}}{V_{applied}} \tag{9}$$

3.3 Process variation

Because of large process variation of an emerging devices like memristor, all of its major parameters display a relatively larger variance. For existing RNG designs which are based on switching time, we are only interested in the variation of switching time. For our purpose, we have considered a 5% standard deviation (σ_{tswn}) on negative switching time, t$_{swn}$.

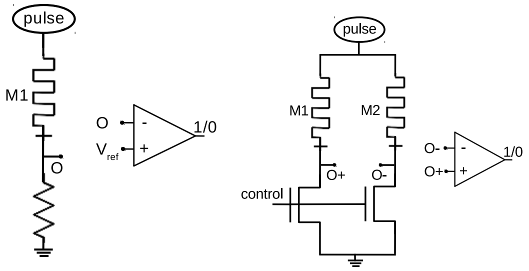

Figure 1: Conceptual circuit diagram of (left) conventional switching time based single memristor TRNG, and (right) the proposed twin memristor TRNG

$$\mathbf{t_{swn}} = N(\mu_{tswn}, \sigma_{tswn}), die - to - die \tag{10}$$

where σ_{tswn} = 0.05*μ_{tswn}. Thus for a 1σ variation of this switching time, the shifted mean of this distribution would be:

$$t_{sw,actual} = \begin{cases} 1.05 * \mu_{tswn}, & 1\sigma \ above \ mean \\ 0.95 * \mu_{tswn}, & 1\sigma \ below \ mean \end{cases}. \tag{11}$$

4 PROPOSED TRNG DESIGN

We have seen that process variation, temperature change and supply voltage variation can affect and thus shift the switching time distribution of a memristor. Therefore, any TRNG design based on the initial distribution of cycle to cycle switching time around the mean of this distribution would create biased proportion of ones and zeros and thus degrade the RNG performance. In this work, we are instead proposing a RNG design based on relative distribution of HRS. In this design, the main source of entropy is the cycle to cycle HRS variation of a pair of memristors. With equal voltage bias of a fixed duration, the memristive states of two memristors are compared to generate random bit streams. Figure 1 displays the high level circuit diagram of the existing switching time based single memristor TRNG and the proposed relative HRS based twin memristor TRNG. Temperature, voltage change and process variation have smaller effect on this proposed TRNG performance as we'll analyze now.

Suppose we have two memristors which are physically close in a circuit layout and their mean HRS are 1σ = 1% (mismatch) apart from each other. Each memristor's HRS has a cycle to cycle Normal distribution of mean μ_{HRS} and standard deviation σ_{HRS} = 10% of μ_{HRS}.

$$\begin{aligned} \mathbf{HRS_1} &= N(\mu_{HRS1}, \sigma_{HRS1}) \\ \mathbf{HRS_2} &= N(\mu_{HRS2}, \sigma_{HRS2}) \end{aligned} \tag{12}$$

Since our proposed design compares between these HRS states, this can be expressed as another Normal random variable as:

$$\begin{aligned} \mathbf{Z} &= \mathbf{HRS_1} - \mathbf{HRS_2} \\ &= N((\mu_{HRS1} \sim \mu_{HRS2}), \sqrt{\sigma^2_{HRS1} + \sigma^2_{HRS2}}) \end{aligned} \tag{13}$$

If $\mu_{HRS2} > \mu_{HRS1}$ and thus μ_{HRS2} = 1.01*μ_{HRS1} (1% larger) and σ_{HRS1} = 0.1*μ_{HRS1}, σ_{HRS2} = 0.1*μ_{HRS2} (10% standard deviation), then equation 13 becomes:

$$\begin{aligned}\mathbf{Z} &= N(.01 * \mu_{HRS1}, \sqrt{(0.1 * \mu_{HRS1})^2 + (0.1 * 1.01\mu_{HRS1})^2)}) \\ &\approx N(.01 * \mu_{HRS1}, 0.1 * \sqrt{2} * \mu_{HRS1})\end{aligned} \tag{14}$$

Since $\mathbf{Z}$ doesn't have a mean of zero, the ratio of ones and zeros wouldn't be the same. However, it should be close to zero as can be deduced using a Z-table. The z-score of this normal distribution to find the probability of a randomly chosen value to be less than 0 is:

$$\begin{aligned}z = \frac{x-\mu}{\sigma} &= \frac{0 - 0.01 * \mu_{HRS1}}{0.1 * \sqrt{2} * \mu_{HRS1}} \\ &= -0.0707\end{aligned} \tag{15}$$

where, z represents a Normal random variable of mean 0 and standard deviation 1, transformed from $\mathbf{Z}$ to calculate the probability using a z-table.

From z-table, the probability of getting a '0' value for z score of -0.0707 is:

$$\mathbf{P}(x \in \mathbf{Z} \leq 0) = 0.4718 \tag{16}$$

It should be noted that if the mean HRS between two memristors are less than 1σ, then the the probability number in equation 16 would be closer to the ideal of 0.5. Our proposed design has inherently some bias. But this provides us with a better robustness against environmental changes compared to existing designs as we'll analyze now.

4.1 Supply voltage variation

Changing the voltage applied across the memristors would change the switching speed. However, there is no direct relationship between this small voltage variation(±10%) with final HRS values. In our design, we chose a large enough switching time so that memristors always switch, because we are only interested in variation in the final HRS values, not the switching probability itself. Therefore, due to rail voltage variation, the time to generate a valid bit from the proposed TRNG changes slightly, but the value of that bit itself is assumed to be unaffected.

4.2 Temperature variation

Temperature would change the HRS of both memristors in our design. For a relatively small temperature window $[0-100]^0$C, this change is approximated by a linear equation, similar to threshold voltage change. Suppose, the temperature is increased by an amount of $\Delta\theta$. Then from equation 7, we can find the two changed HRS as:

$$\begin{aligned}HRS1'(\theta) &= HRS1[1 + \alpha_{HRS} * \Delta\theta] \\ HRS2'(\theta) &= HRS2[1 + \alpha_{HRS} * \Delta\theta]\end{aligned} \tag{17}$$

where, HRS_0 represent HRS at room temperature and α_{HRS} is the linear temperature coefficient of HRS and approximated as -0.008/^{0}C. The die-to-die standard deviation of HRS considered here is σ_{HRS} = 0.1*μ_{HRS} (10% of HRS).

Now, let's use the term $C_{\Delta\theta} = 1 + \alpha_{HRS} * \Delta\theta$. Thus we can rewrite equations 12 for these two new HRS values at a different temperature. Here, we assume that the standard deviation (and variance) stays the same.

$$\begin{aligned}\mathbf{HRS'}_1 &= N(\mu_{HRS1'}, \sigma_{HRS1'}) \\ &= N(\mu_{HRS1} * C_{\Delta\theta}, \sigma_{HRS1}) \\ \mathbf{HRS'}_2 &= N(\mu_{HRS2'}, \sigma_{HRS2'}) \\ &= N(\mu_{HRS2} * C_{\Delta\theta}, \sigma_{HRS2})\end{aligned} \tag{18}$$

and

$$\begin{aligned}\mathbf{Z'} &= \mathbf{HRS}'_1 - \mathbf{HRS}'_2 \\ &= N(C_{\Delta\theta} * (\mu_{HRS1} \sim \mu_{HRS2}), \sqrt{\sigma^2_{HRS1} + \sigma^2_{HRS2}})\end{aligned} \tag{19}$$

Thus equation 18 and 19 actually represents scaled versions of the normal distribution of equation 12 and 13, respectively. Now we can find the new distribution of our proposed design by using equation 18 with equations 13 and 14. The z-score of this scaled normal random variable is:

$$\begin{aligned}z = \frac{x-\mu}{\sigma} &= \frac{0 - C_{\Delta\theta} * 0.01 * \mu_{HRS1}}{0.1 * \sqrt{2} * \mu_{HRS1}} \\ &= -C_{\Delta\theta} * 0.0707\end{aligned} \tag{20}$$

For 50^{0}C temperature increase, $C_{\Delta\theta}$ = (1-.008*50) = 0.6 and thus the z-score would be z = 0.0424. The probability of getting 0:

$$\mathbf{P}(x \in \mathbf{Z} \leq 0) = 0.4831 \tag{21}$$

This value is very close to the value in equation 16. Thus changing the temperature does not bring in a significant change on the ratio of 1's and 0's and the RNG performance does not degrade with changing temperature.

4.3 Process variation

In last subsection, we observe that temperature change creates a scaled version of the original HRS distributions. Process variation would cause the HRS values to be different from the die-to-die mean HRS value, causing a shift in the distribution. However, since we are taking the difference among two memristors, the resulting distribution would just display a scaled mean and scaled standard deviation of the original distribution.

$$\begin{aligned}\mathbf{Z} &= \mathbf{HRS}''_1 - \mathbf{HRS}''_2 \\ &= N((\mu_{HRS1''} \sim \mu_{HRS2''}), \sqrt{\sigma^2_{HRS1''} + \sigma^2_{HRS2''}}) \\ &\approx N(.01 * \mu_{HRS1''}, 0.1 * \sqrt{2} * \mu_{HRS1''})\end{aligned} \tag{22}$$

Suppose, $\mu_{HRS1''}$ is 1σ larger than the mean μ_{HRS1}. Thus $\mu_{HRS1''}$ = 1.1*μ_{HRS1} and the z-score of this would again be:

$$\begin{aligned}z = \frac{x-\mu}{\sigma} &= \frac{0 - 0.01 * \mu_{HRS1''}}{0.1 * \sqrt{2} * \mu_{HRS1}} \\ &= \frac{0 - 0.01 * 1.1 * \mu_{HRS1}}{0.1 * \sqrt{2} * \mu_{HRS1}} \\ &= -0.0778\end{aligned} \tag{23}$$

The probability of getting 0 for 1σ variation of HRS is thus:

$$\mathbf{P}(x \in \mathbf{Z} \leq 0) = 0.469 \tag{24}$$

Thus the ratio of 1's and 0's change by a very small amount with process variation. This is an excellent result since it states that we do not need to change any design parameters even if memristors

Table 1: Percentage of 0's to the sum of 0's and 1's for different operating conditions

Design with	**Ideal/No**	**Temperature variation**		**Voltage variation**		**Process variation**	
varying conditions	**change**	**$+25^0$C**	**-25^0C**	**+10%**	**-10%**	**+1 σ**	**-1 σ**
Single-memristor	50	4.01	95.99	18.16	86.67	69.15	30.85
Proposed (twin memristor)	47.18	47.76	46.62	47.18	47.18	46.90	47.46

show a large variance across different die as the RNG performance should remain very close to optimum operating condition.

4.4 Output Correction

Since our proposed design inherently has a small percentage of bias i.e. amount of 1's and 0's produced are not equal even in ideal condition, some form of post processing could be useful to improve the data quality. In this design, we have proposed to use simple Von-Neumann correction which can be implemented very easily with minimum hardware overhead. Von-Neumann correction discards when two consecutive RNG output bits are both '1' or '0'. But if the bit sequence is '10' then it becomes a '1', and if the sequence is '01' then it becomes a '0'. It helps to remove simple bias and reduce correlation between consecutive bits. XORing of two TRNGs is another technique. If output from two TRNGs are not correlated, then XORing them could help to improve randomness in the output.

5 RESULTS AND ANALYSES

First, we have analyzed both the conventional single memristor based TRNG design and proposed twin memristor based TRNG design in terms of the proportion of 1's and 0's with respect to varying operating conditions. We have used MATLAB to generate simulation for our experiments. We calculated the percentage of 0's compared to the sum of 1's and 0's in normal operating condition, at temperature $\pm 50^0 C$ change compared to room temperature, applied voltage with ±10% variation and with 1σ process variation. This 1σ process variation for single memristor design corresponds to the negative switching time variation and is assumed to be 5%. For the proposed design, the variation corresponds to of HRS and is assumed to be 10% as HRS displays higher variation than switching time or other memristor characteristics. Table 1 displays the result. As expected, process variation and temperature variation strongly affect switching time and thus the single memristor based design shows a large bias towards 1's or 0's except for normal condition. The proposed design is based on relative measurement of HRS and because of mismatch, inherently has a small bias i.e. the proportion of 0 is not 50% ideally. However, due to its relative nature, our proposed design only show a small change in performance with changing operating conditions. Thus our design can be considered robust against environmental changes or against large process variation. Table 1 lists the performance of the two design for different operating conditions. As expected, performance of the proposed design doesn't show a large deviation from ideal condition where a single memristor TRNG performance could be affected severely.

To better understand the effect of process variation, we have run Monte Carlo analysis with 10,000 different chips and generated 5000 random bits from each chip. Figure 2 shows the percent of 1's for different chips for both existing and proposed TRNG design. A TRNG should ideally has 50% of 1's i.e. the mean should be 0.5. As expected for our theoretical analysis, existing switching time based TRNG design shows a wide distribution with process variation with a standard deviation of 0.1805. But our proposed twin memristor design only has a narrow distribution over die-to-die process variation with a standard deviation of only 0.0322. This result shows our proposed design is robust against process variation and eliminates the need to redesign for every circuit instance.

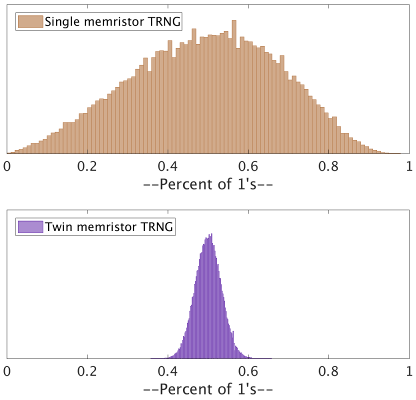

Figure 2: Histogram of bit bias with process variation for (top) existing single memristor design and (bottom) proposed twin memristor design

We have also run Monte Carlo to show how the histogram shifts from ideal for a $\pm 10^0 C$ change in temperature and for a ±10% change supply voltage variation. This result is shown in Figure 3. Conventional single memristor based design shows a large deviation from ideal (i.e. with a mean of 0.5) distribution for both increasing and decreasing temperature and supply voltage. But our proposed design shows almost zero or very negligible change.

6 FUTURE WORK

In this work, we have provided with a general idea about what to consider when designing a RNG using stochasticity of memristor's switching behavior. However, as different memristor materials change how a memristor switching behaves in varying conditions, the performance of a TRNG would also be different. If we can fully characterize several different types of memristors, then we can analyze and compare their performances with more confidence. Finally, our design is based on theoretical analysis only and thus other source of randomness of a circuit might affect the performance. Therefore, our immediate step would be to build our design using actual memristors and then evaluate it. The design of a good comparator to compare the memristive states is another important

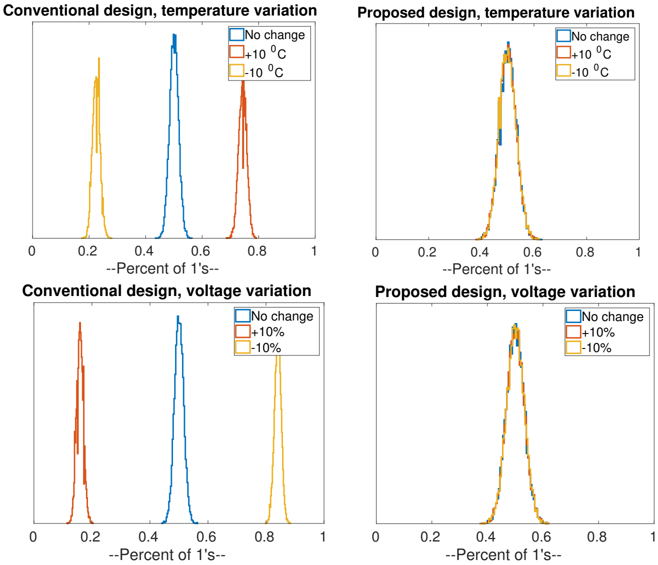

Figure 3: Histogram plot showing the trend for shift in distribution with temperature and voltage variation for both conventional and proposed design. Conventional design displays a large variation while proposed design almost unchanged.

task. Finally, since our results are based on simplified mathematical model based simulation, any statistical true random number tests are not shown in this work as softwares can only produce pseudo-random numbers but it can be done from real hardware designs.

7 CONCLUSION

We have used statistical distribution of random numbers to analyze the theoretical strength of existing TRNG designs. To improve upon these designs, we have proposed our twin memristor based TRNG design which theoretically performs better than existing designs in varying operation conditions and against large process variation of memristors. It also doesn't require sophisticated pulse width control to generate random bits and thus complexity of input signal generation circuit is reduced. Our design is also generic and unlike existing designs, it doesn't need to be experimented on heavily to determine input parameters. However, as we know memristor is a novel and immature device, thus many different memristors consisting of many different materials with different underlying physics are out there. Therefore, the temperature and other environmental factor dependence might be different for individual memristor. Thus the actual performance of a particular TRNG design would be very dependent on its underlying materials and their characteristics.

ACKNOWLEDGMENT

The authors would like to thank Md. Badruddoja Majumder of our research group for interesting discussions related to this topic.

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-16-1-0301. Any opinions, finding, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the United States Air Force.

REFERENCES

[1] Simone Balatti, Stefano Ambrogio, Zhongqiang Wang, and Daniele Ielmini. 2015. True random number generation by variability of resistive switching in oxide-based devices. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 5, 2 (2015), 214–221.

[2] T. Cabout et al. 2013. Temperature impact (up to 200C) on performance and reliability of HfO_2-based RRAMs. In *5th IEEE International Memory Workshop*. 116–119. https://doi.org/10.1109/IMW.2013.6582112

[3] Z. Fang, H. Y. Yu, W. J. Liu, Z. R. Wang, X. A. Tran, B. Gao, and J. F. Kang. 2010. Temperature Instability of Resistive Switching on HfO_2 -Based RRAM Devices. *IEEE Electron Device Lett.* 31 (May 2010), 476–478. Issue 5. https://doi.org/10.1109/LED.2010.2041893

[4] Hao Jiang, Daniel Belkin, Sergey E SavelâĂŹev, Siyan Lin, Zhongrui Wang, Yunning Li, Saumil Joshi, Rivu Midya, Can Li, Mingyi Rao, et al. 2017. A novel true random number generator based on a stochastic diffusive memristor. *Nature communications* 8, 1 (2017), 882.

[5] Benjamin Jun and Paul Kocher. 1999. *THE INTEL Âő RANDOM NUMBER GENERATOR.* Technical Report. CRYPTOGRAPHY RESEARCH, INC. WHITE PAPER PREPARED FOR INTEL CORPORATION. 8 pages.

[6] J. Kim, T. Ahmed, H. Nili, N. Duy Truong, J. Yang, D. S. Jeong, S. Sriram, D. C. Ranasinghe, and O. Kavehei. 2017. Nano-Intrinsic True Random Number Generation. *ArXiv e-prints* (Jan. 2017). arXiv:1701.06020

[7] M. B. Majumder, M. Uddin, G. S. Rose, and J. Rajendran. 2016. Sneak path enabled authentication for memristive crossbar memories. In *2016 IEEE Asian Hardware-Oriented Security and Trust (AsianHOST)*. 1–6. https://doi.org/10.1109/AsianHOST.2016.7835568

[8] Nathan R. McDonald. 2012. *Al/Cu_x O/Cu Memristive Devices: Fabrication, Characterization, and Modeling*. Master's thesis. College of Nanoscale Science and Engineering, University at Albany, Albany, NY.

[9] Rawan Naous, Maruan Al-Shedivat, and Khaled Nabil Salama. 2016. Stochasticity modeling in memristors. *IEEE Transactions on Nanotechnology* 15, 1 (2016), 15–28.

[10] P. Pouyan, E. Amat, and A. Rubio. 2014. Reliability challenges in design of memristive memories. In *5th European Workshop on CMOS Variability (VARI)*. 1–6. https://doi.org/10.1109/VARI.2014.6957074

[11] Dmitri B. Strukov, Gregory S. Snider, Duncan R. Stewart, and R. Stanley Williams. 2008. The missing memristor found. *Nature* 453 (May 2008), 80–83.

[12] M. Uddin, M. B. Majumder, and G. S. Rose. 2017. Robustness Analysis of a Memristive Crossbar PUF Against Modeling Attacks. *IEEE Transactions on Nanotechnology* 16, 3 (May 2017), 396–405. https://doi.org/10.1109/TNANO.2017.2677882

[13] Mesbah Uddin, Md. Badruddoja Majumder, Garrett S. Rose, Karsten Beckmann, Harika Manem, Zahiruddin Alamgir, and Nathaniel C. Cady. 2016. Techniques for Improved Reliability in Memristive Crossbar PUF Circuits. In *IEEE Comp. Society Annual Symp. on VLSI (ISVLSI)*. 212–217. https://doi.org/10.1109/ISVLSI.2016.33

[14] Christian Walczyk, Damian Walczyk, Thomas Schroeder, Thomas Bertaud, MaÅĆgorzata Sowinska, Mindaugas Lukosius, Mirko Fraschke, Dirk Wolansky, Bernd Tillack, Enrique Miranda, and Christian Wenger. 2011. Impact of Temperature on the Resistive Switching Behavior of Embedded HfO_2 -Based RRAM Devices. *IEEE Trans. on Electron Devices* 58 (September 2011), 3124–3131. Issue 9. https://doi.org/10.1109/TED.2011.2160265

[15] Chaofei Yang, Beiye Liu, Yandan Wang, Yiran Chen, Hai Li, Xian Zhang, and Guangyu Sun. 2016. The applications of NVM technology in hardware security. In *Proceedings of the 26th edition on Great Lakes Symposium on VLSI*. ACM, 311–316.