Leveraging Aging Effect to Improve SRAM-based True Random Number Generators

Saman Kiamehr Mohammad Saber Golanbari Mehdi B. Tahoori
Karlsruhe Institute of Technology, Karlsruhe, Germany
e-mails: {kiamehr, mohammad.golanbari, mehdi.tahoori}@kit.edu

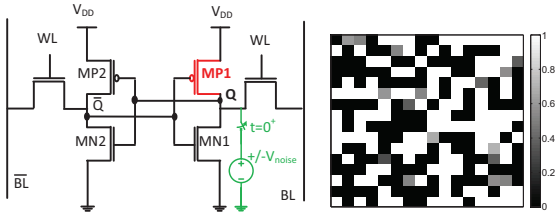
***Abstract*—The start-up value of SRAM cells can be used as the random number vector or a seed for the generation of a pseudo random number. However, the randomness of the generated number is pretty low since many of the cells are largely skewed due to process variation and their start-up value leans toward zero or one. In this paper, we propose an approach to increase the randomness of SRAM-based True Random Number Generators (TRNGs) by leveraging transistor aging impact. The idea is to iteratively power-up the SRAM cells and put them under accelerated aging to make the cells less skewed and hence obtaining a more random vector. The simulation results show that the min-entropy of SRAM-based TRNG increases by 10X using this approach.**

I. Introduction

True Random Number Generator (TRNG) is a key element of cryptographic systems providing secret keys and tokens. For ultra-low power secure applications, e.g. Internet of Things (IoT), designing low power and reliable TRNG is very crucial. There are different types of TRNGs which are based on non-deterministic physical randomness in circuits [1]. One common way of TRNG implementation is based on SRAM cells. In the hardware security context, SRAM-cells can be used as Physical Unclonable Function (PUF). A PUF is normally used for identification or authentication of an IC by generating a unique signature for the IC. In case of SRAM-based PUF, by powering up the SRAM array, a large portion of the bit cells tend to start-up with a particular value of either "zero" or "one" independent of noise. This is due to the fact that the transistor parameters of those cells are skewed from their nominal value due to process variation which makes the SRAM cell asymmetrical. Because of the random nature of process variation, the start-up pattern of a SRAM memory array results in a unique pattern for each chip which can be used as device fingerprint [2–5].

However, there are some cells in the SRAM-array which are less skewed and therefore, the start-up values of these cells exhibit more randomness dependent on the noise of each power-up. These random power-up bits can be used as a source of random number generation [6]. However, only a small portion of SRAM cells show the noisy behaviour and therefore the entropy of the SRAM power-up pattern has to be condensed into a full entropy random seed [1]. This is done by some compression approaches such as using a hash function. It is shown that in order to have 256 bits with full entropy, the size of the SRAM array should be at least 1600 bytes (50 times larger) [1].

In this paper, we propose an approach to improve the entropy of SRAM-based TRNG by leveraging the transistor aging impact. The idea behind our method is the fact that by powering up a skewed SRAM cell, and applying elevated stress (for accelerated aging) under this "preferred" state, its skewness reduces. Based on this fact, we propose an approach in which the SRAM-based TRNG is powered up iteratively and at each iteration the chip is aged at a high voltage and temperature (accelerated aging) during the burn-in phase in the in order to make SRAM cells less skewed and in turn increase randomness. Simulation results show that using our approach, the min-entropy of the SRAM-based TRNG can be improved by 10X. For the same min-entropy, this can be translated to a significant improvement (∼10X) in the area and power overhead of TRNGs which makes it very practical for IoT applications.



(a) (b)

Fig. 1: a) Structure of 6T SRAM cell b) Probability of start-up value being "one" (POP_1) for 256 bits SRAM-based TRNG

II. Motivation

The simplest structure of a SRAM cell is the 6T structure consisting of two back-to-back inverters and two access transistors (see Fig 1(a)). If there is no variation, the SRAM cell is *non-skewed* and the back-to-back inverters structure is symmetrical. By powering up the SRAM cell, due to its non-skewed symmetrical structure, it randomly gets a value of either "one" or "zero" dependent on noise, e.g. thermal noise. Here, we define probability of powering-up to "one" (POP_1) or "zero" (POP_0), showing the skewness of the cell:

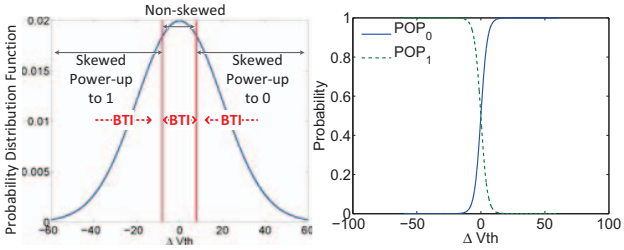
$$POP_1 = \frac{\#of(Startup\ value = \text{“}one\text{”})}{Total\ number\ of\ trials} = 1 - POP_0 \quad (1)$$

According to this definition, if the SRAM cell is not skewed, the POP_1 or POP_0 are close to 0.5. However, in the presence of process variation, some of the transistors might become stronger and the SRAM might become asymmetrical which we call here *skewed*. By powering up the skewed SRAM cell, the probability of settling at "one" could be higher or lower than "zero" according to the strength of the internal transistors impacted by process variation. Fig. 1(b), shows the POP_1 for a SRAM array with 256 bits. As can be seen in this figure, most of the cells are skewed such that their start-up values are mostly "one" or "zero", showing a low randomness.

Here, for the sake of simplicity, we explain the impact of process variation for a case in which only the PMOS transistor MP1 in Fig 1(a) is affected and its ΔVth shift from the nominal value is shown in Fig. 2(a). If due to process variation this transistor becomes stronger (weaker), this means that the probability that node "Q" becomes equal to "one" is higher (lower) (see Fig 2(b)). When MP1 is strong enough (large negative ΔVth), the start-up state of Q and $\overline{Q}$ are always "one" and "zero", respectively. In this case, if the SRAM cell stays ON under this power-up state, the transistor MP1 is under NBTI stress and therefore it becomes weaker, which means the SRAM cell becomes less skewed. However, if we continue aging the cell, MP1 might become that weak (even weaker than MP2) leading to a case in which the start-up state of Q and $\overline{Q}$ flips.

For the case in which MP1 is not skewed, the start-up value is random and if the cell is aged according to this value, the transistor MP1 might become more skewed. Although the above explanation is for the case in which the process variation and NBTI impacts only on the MP1 transistor are considered, the observation is valid for all transistors considering also PBTI effect. In general from the above observation we can conclude:

978-3-9815370-8-6/17/$31.00 ©2017 IEEE



(a) (b)

Fig. 2: a) ΔVth distribution of the MP1 transistor, the red arrows show the possible direction of BTI-induced ΔVth for each region b) Equivalent POP_1 and POP_0 of the SRAM cell

- Aging a skewed cell with its start-up state leads to a less skewed cell which is good for TRNG. However, over-aging the cell with the same state might make the cell skewed again in the other direction.
- Aging a non-skewed cell with its start-up state might lead to more skewness in the cell which is not good for generating randomness.

III. Proposed Approach

Here, we propose an approach to improve the randomness of TRNGs by leveraging the BTI effect. The overall flow of our approach is shown in Fig 3. In this approach, the SRAM array is powered-up iteratively and the cells are aged in an accelerated manner for each iteration according to their start-up value. At each iteration, the power-up value of a non-skewed cell is a random number of "zero" or "one" and if we age the cell iteratively, in almost half of the iterations the cell is aged according to the value "zero" and in the other half of iterations the cell is aged according to the value "one" leading to a symmetrical BTI aging and therefore the cell remains non-skewed. However, in case of a skewed cell, the iterative aging of the cell by its power-up value makes it less skewed. For example, if the cell is skewed to power-up to "one" (the left side of Fig 2(a) with large negative ΔVth) with a probability of $POP_1 = 0.9$, in 9 iterations out of 10, it ages in a way that the $|\Delta Vth|$ becomes smaller (shift to right and closer to zero in Fig 2(a)) which makes the cell less skewed. The accelerated aging can be performed during the burn-in phase of chip production by increasing the temperature and the supply voltage value [7] and more details is provided in Section IV-A3.

The proposed iterative approach can be mathematically modelled as a random process. The complete model of the random process is very complicated because its state space has at least six continuous dimensions corresponding to ΔV_{th} of all the six transistors in an SRAM cell. For simplification, we assume a skewness value for each memory cell instead of considering the ΔV_{th} of the transistors. When the cell is positively (negatively) skewed, it tends to fall into start-up value of "zero" ("one") in the next iteration. With no skewness, the choice of the next state is completely random. Also for the sake of simplicity we assume that the skew values are discrete. The continuous behaviour can be achieved by increasing the resolution of the skewness value. The discrete skewness values form a discrete state space ($x_1, \dots x_n \in \Omega$) and the process is defined as a discrete time process $\{X_t,\ t = 1, 2, \dots\}$ where $X_t \subseteq \Omega$. The process is memory less, i.e. the choice of the next state is only dependent on the current state of the cell and hence the process is a time-homogeneous Markov chain. The probabilities of the transition from any state to another are typically stored in a transition matrix P:

$$p_{x,y} = Prob[X_{t+1} = y | X_t = x] = Prob[X_1 = y | X_0 = x]. \quad (2)$$

The transition matrix models the states and the transitions as the nodes and edges of a directed graph, respectively. In the case of our problem, $p_{x,x} = 0$ because after any aging iteration the skewness changes. We simplify the problem by assuming the skewness of the cell as the threshold voltage of transistor MP1 ($x = \Delta V_{th}$) and in each iteration ΔV_{th} is changed by $\pm 1mV$. Therefore, the state space can be defined as $\Omega = \{-60, -59, ..., 59, 60\}$.

Fig 4 illustrates the Markov chain of the proposed iterative approach with the aforementioned assumptions. The nodes of the graph are the states and the edges are the probabilities to switch from one state to another. The elements of the transition matrix can be extracted from POP_1 and POP_0 plots in Fig 4. For example, the probability to fall to state -1 and 1 are equal when the previous state is zero ($\Delta V_{th} = 0$):

$$p_{0,-1} = p_{0,1} = 0.5$$

This is because at state 0 the SRAM cell is symmetric (no skew). However, when the state is 1 (or equally $\Delta V_{th} = 1mV$) the probability to fall to state 0 and 2 after aging are 0.8 and 0.2, due to the positive skewness of the SRAM cell.

$$p_{1,0} = 0.8 \qquad p_{1,2} = 0.2$$

Therefore, the transition matrix is built accordingly:

$$P = \begin{bmatrix} p_{-60,-60} & p_{-60,-59} & \cdots & p_{-60,59} & p_{-60,60} \\ \vdots & & \ddots & & \vdots \\ p_{60,-60} & p_{60,-59} & \cdots & p_{60,59} & p_{60,60} \end{bmatrix} \quad (3)$$

In fact, the superdiagonal and subdiagonal entries of the transition matrix are POP_1 and POP_0 probabilities (Fig. 2(b)), respectively.

Assuming that the probabilities of the states at iteration n is f_n which is a vector of length $|\Omega|$, the probabilities of the states at iteration $n+1$ would be $f_{n+1} = f_n P$, in other words $f_n = f_0 P^n$. According to the fundamental theorem of Markov chains [8], there is a unique probability vector π called *stationary distribution* which has the attribute:

$$\pi = \pi P. \quad (4)$$

After a number of iterations the Markov chain converges to the stationary distribution π , i.e. additional iterations have no impact on the distribution. It is evident from Equation (4) that the resulting distribution π is solely dependent on the transition matrix. Intuitively, we can say that the initial distribution is limited by the POP_1 and POP_0 distributions in each iteration, and after a number of

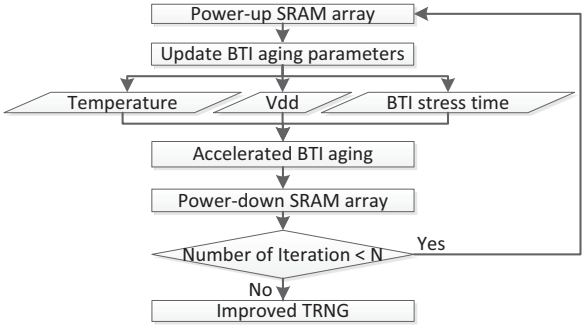


Fig. 3: The overall flow of proposed approach

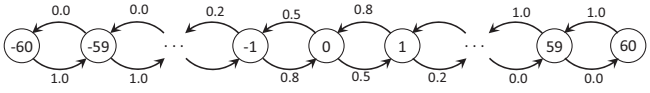
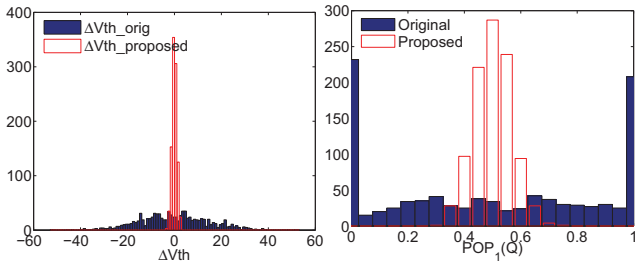


Fig. 4: Simple Markov chain structure of the proposed iterative approach with skewness $x \in \Omega = \{-60, ..., 60\}$.



(a) (b)

Fig. 5: a) ΔVth distribution of MP1 transistor b) Histogram of probability of "one" (POP_1) of node Q of the SRAM cell

iterations the final distribution would span where both POP_1 and POP_0 distributions are non-zero.

Therefore, π should have much smaller standard deviation compared to the original distribution of states leading to narrower distribution of the threshold voltage of the transistors after applying the proposed iterative approach. This is shown in Fig 5(a), where after N iterations ΔV_{th} has a narrower distribution compared to the original distribution which in turn leads to the cells with lower skewness, i.e. POP values closer to 0.5 (as shown in Fig 5(b)).

IV. Simulation Results

A. Simulation set-up and flow

1) TRNG setup: We assume that our SRAM-based TRNG consists of 256 SRAM cells (256 bits). We obtain the start-up pattern of the SRAM array for 100 trials and then we compare the original TRNG and the TRNG after applying our proposed approach considering the following metrics:

- POP_1 as defined in Equation (1): A value close to "zero" or "one" means that the cell is skewed and a value close to 0.5 means that the cell is non-skewed which is more suitable for TRNG.
- Min-entropy: is the worst-case (i.e., the greatest lower bound) measure of uncertainty for a random variable according to The National Institute of Standards and Technology (NIST) [1, 9]. The min-entropy of each $cell_i$ can be obtained by [1]:

$$H_{min}^{cell_i} = -log_2(POP_{max}) \quad (5)$$

$$POP_{max}^{cell_i} = max(POP_0, POP_1) \quad (6)$$

The total entropy of n independent bits can be obtained by:

$$(H_{min})_{total} = \sum_{i=1}^{n} H_{min}^{cell_i} \quad (7)$$

Higher values for $(H_{min})_{total}$, i.e., closer to 1, show a better TRNG with higher randomness.

2) SRAM simulation flow: For each SRAM cell, we consider a 6T structure as shown in Fig. 1(a). Process variation is considered as a shift in the threshold voltage (ΔVth) values of the internal transistors for all 256 cells according to the Pelgrom model [10]. For each cell, the start-up value is obtained using HSPICE simulation with 32nm SAED SPICE model. For this purpose, the word-line node (WL) is connected to ground and the nodes Q and $\overline{Q}$ are initialized with "zero" value and then a transient simulation is performed to obtain the value of Q and $\overline{Q}$ after a given time. To consider the noise, a voltage source is connected to node Q with a value of $+/-V_{noise}$ (see Fig. 1(a)). The sign of this voltage source is changed randomly for each trial. The value of V_{noise} is set in a way that the min-entropy of the original 256 bit SRAM-based TRNG is approximately equal to 5% which is consistent with the results provided for real SRAM-based TRNGs [1].

3) Accelerated BTI aging: As shown in Fig. 3, at each iteration of our proposed approach, the SRAM cell is aged under BTI stress in an accelerate manner according to its start-up value in order to decrease the skewness of the cell. The accelerated BTI aging can be performed by increasing the supply voltage and the temperature during BTI stress [11]. If the start-up value of Q is "zero" ("one"), the MP2 (MP1) and MN1 (MSN2) transistors are under NBTI and PBTI aging stress since these two transistors are ON. For this set of simulation, we model the BTI impact by a threshold voltage shift of transistors and we obtain the results for three different scenarios for accelerated aging as explained in the following and for each scenario the number of iterations is 100.

- Uniform BTI aging: In this scenario, we age the SRAM cells in a way that the ΔVth of the transistors under stress is equal at each iteration. The amount of aging (ΔVth) should be carefully selected for this approach because it causes a trade-off between the efficiency of the approach and the required number of iterations. More BTI aging at each iteration (larger ΔVth step) causes the strongly skewed cells to become less skewed faster, i.e., less number of iterations is needed, however it will lead to less efficiency of our approach at the end in terms of the randomness of TRNG. To explain this, please consider the Fig. 2(a). At each iteration, ΔVth shifts to the right or the left. The choice of a very large ΔVth step, e.g. 30mV, ages a skewed cell in a way that it becomes skewed in the other direction. A choice of smaller ΔVth steps, causes more cells to become non-skewed by enough number of iterations. For this set of simulation, we picked two values of 1mV and 0.5mV for the BTI aging-induced ΔVth at each iteration.
- Non-uniform BTI aging: In order to obtain a good trade-off between the efficiency of the approach (the obtained randomness at the end) and the required number of iterations, we also propose a non-uniform BTI aging scenario. In these scenario, the SRAM cells are aged more in the beginning (the earlier iterations) to improve the strong skewed cells faster, i.e. better efficiency is obtained with less number of iterations. Then, at each iteration, we reduce the amount of BTI aging in order to reach a better randomness by fine-tuning the final Vth of transistors at the end of all iterations. Here we chose a Non-uniform harmonic BTI aging in which a harmonic series for BTI-induced ΔVth at each iteration is used:

$$\Delta Vth_0 = 12mV \quad (8)$$

$$\Delta Vth_i = \Delta Vth_0/(i+1) \quad (9)$$

Vth_0 for non-uniform scenario is chosen according to the standard deviation of process variation-induced Vth shift ($\sigma_{\Delta Vth}$).

B. Min-entropy over iterations

Fig. 6(a) shows the min-entropy of 256-bits SRAM-based TRNG over iterations using different iterative aging scenarios (introduce in Section IV-A3). As shown in Fig. 6(a), the entropy under all these scenarios improves significantly over iterations. In case of uniform BTI aging, the choice of lower aging rate per iteration ($\Delta Vth = 0.5mV$) results in a slightly better min-entropy at the end of iterations, however the choice of larger aging rate ($\Delta Vth = 1.0mV$) converges faster to better min-entropy ranges.

Compared to the uniform aging scenarios, the choice of non-uniform aging rate per iteration results not only in better final min-entropy values, but also it leads to faster convergence. This is due to the fact that in case of non-uniform aging rate, in the first iterations the BTI aging and hence the induced ΔVth is large leading to faster

conversion improvement of strongly skewed cells. Over the iterations, the amount of BTI-induced ΔVth is reduced leading to a more fine tune improvement in the skewness of the SRAM cells.

According to the results of Fig. 6(a), the choice of non-uniform harmonic BTI aging provides the best trade-off between the obtained randomness and the number of required iterations. In general, the proposed approach can increase the min-entropy from 5% to more than 55% which means around 10X improvement in the min-entropy value. This is translated to a huge area and power saving because the same amount of min-entropy can be obtained with a much smaller TRNG size (10 times smaller).

C. POP results

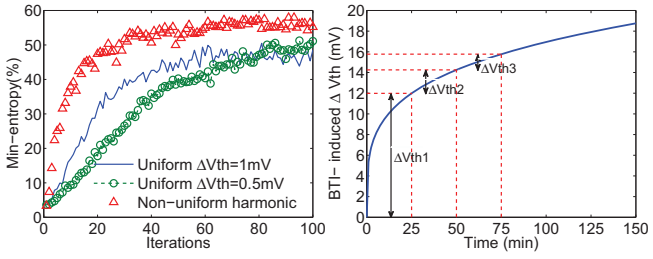
Fig. 7 depicts the probability of powering-up to "one" (POP_1) of 256 bits TRNG for different iterations using proposed approach for non-uniform harmonic BTI aging scenario (as the best choice according to last sub-section). According to the figure, before applying the proposed approach, the POP_1 value of most of the cells is either 0 or 1, meaning that the start-up value is mostly "zero" or "one" for trials of powering-up showing that the number generated by TRNG is not a random value. However, by applying our approach, POP_1 of most of the cells become close to 0.5 meaning that in half of the trials the start-up value is "zero" and in the other half it is "one" which is essential for a desirable TRNG.

D. Accelerated aging approach

As shown previously, the choice of non-uniform BTI aging provides the best trade-off between the efficiency of the approach and the number of required iterations. In this sub-section we explain the process of accelerated BTI aging for each iteration. According to the literature, the BTI aging-induced ΔVth is a function of different factors such as temperature and supply voltage. Here we assume that the delay of a digital circuit will degrade around 15% in three years in room temperature and nominal supply voltage ($Vdd = 1.0V$). This could be translated to $\Delta Vth = 20mV$ in 32nm technology node using HSPICE simulation.

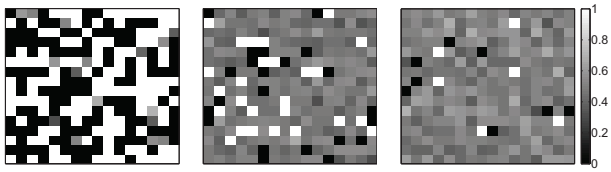
As discussed in Section IV-A3, the amount of ΔVth required for the first iteration is around $\Delta Vth = 12mV$ which can be translated to 350 days of BTI aging in room temperature and nominal supply voltage using equations provided in [12]. However, the BTI aging can be accelerated using higher supply voltages and temperatures [11]. If we increase the temperature to 125 $^\circ C$ and supply voltage to $1.8V$, it takes only 25 minutes for the first iteration.

For the next iterations, even if we stress the cells under the same supply voltage, temperature and stress time, the amount of ΔVth is decreased since the BTI induced degradation has a logarithmic dependency with the time (see Fig. 6(b)). This means that the amount of BTI aging-induced ΔVth decreases over iterations even under the same condition and this is exactly what we need in the non-uniform scenario. For the stress conditions of $Temp = 125^\circ C$, $Vdd = 1.8V$ and $Stress\ time = 25min$, the ΔVth of first, second and third iterations are equal to $12mV$, $2.27mV$, and $1.52mV$, respectively satisfying the condition of our non-uniform scenarios. According to the results depicted in Fig. 6(a) and Fig. 7, using a non-uniform scenario, 50 iterations are enough to obtain a desirable TRNG. This means that the entire process of our approach takes around $25min \cdot 50 = 20hours$ during the burn-in phase. Of course, the numbers provided here are just examples to show the feasibility of our proposed approach, however, the BTI aging conditions (temperature, supply voltage and stress time) at each iteration could be adapted according to the technology.



(a) (b)

Fig. 6: a) Min-entropy over iterations using proposed approach b) BTI-induced ΔVth over time at different iterations



(a) iteration1 (b) iteration25 (c) iteration50

Fig. 7: POP_1 over iterations for a) iteration=1 b) iteration=25 c) iteration=50.

It should be noted that, by performing accelerated BTI for the entire memory block during the burn-in phase not only the efficiency of the TRNG part is improving, but also the yield and Signal Noise Margin (SNM) of the other parts of the memory array, not used for TRNG, can be improved [13].

V. Conclusion

In this paper, we propose an approach in which Bias Temperature Instability (BTI) is used to improve the randomness of SRAM-based TRNGs. The idea behind the proposed approach is to iteratively power-up the SRAM cells and age them in an accelerated manner to make the cells less skewed. Simulation results show that the min-entropy of SRAM-based TRNGs could be improved by 10X using our proposed approach.

References

[1] V. van der Leest *et al.*, "Efficient implementation of true random number generator based on sram pufs," in *Cryptography and Security: From Theory to Applications*. Springer, 2012, pp. 300–318.

[2] S. Chellappa and L. T. Clark, "Sram-based unique chip identifier techniques," *VLSI*, vol. 24, no. 4, pp. 1213–1222, 2016.

[3] C. Herder *et al.*, "Physical unclonable functions and applications: A tutorial," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1126–1141, 2014.

[4] P. Koeberl *et al.*, "Evaluation of a puf device authentication scheme on a discrete 0.13 um sram," in *INTRUST*, 2011.

[5] A. Garg and T. T. Kim, "Design of sram puf with improved uniformity and reliability utilizing device aging effect," in *ISCAS*. IEEE, 2014, pp. 1941–1944.

[6] D. E. Holcomb *et al.*, "Power-up sram state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, 2009.

[7] R.-P. Vollertsen, "Burn-in," in *Integrated Reliability Workshop Final Report,*. IEEE, 1999, pp. 167–173.

[8] J. R. Norris, *Markov chains*. Cambridge university press, 1998, no. 2.

[9] E. Barker *et al.*, "Nist special publication 800-90a: Recommendation for random number generation using deterministic random bit generators," 2012.

[10] M. J. Pelgrom *et al.*, "Matching properties of mos transistors," *IEEE Journal of solid-state circuits*, vol. 24, no. 5, pp. 1433–1439, 1989.

[11] A. Amouri *et al.*, "Aging effects in fpgas: an experimental analysis," in *FPL*. IEEE, 2014, pp. 1–4.

[12] S. Bhardwaj *et al.*, "Predictive modeling of the nbti effect for reliable design," in *CICC*. IEEE, 2006, pp. 189–192.

[13] J. Wang *et al.*, "Improving sram vmin and yield by using variation-aware bti stress," in *CICC*. IEEE, 2010, pp. 1–4.