Metastability with Emerging Reconfigurable Transistors: Exploiting Ambipolarity for Throughput

Abhiroop Bhattacharjee[†][*], Shubham Rai[†], Ansh Rupani[†], Michael Raitza[†], Akash Kumar[†]
[†]Chair of Processor Design, Technische Universität Dresden, Germany
[*]Department of Electrical and Electronics Engineering, Birla Institute of Technology and Science Pilani, Pilani, India

***Abstract*— In this work, we leverage ambipolar transistors in the context of metastability for random number generation. We propose designs of a *Minority*-based SR latch and a dual-edge triggered *True Single Phase Clock D-Flip-Flop* (TSPC DFF) to sample two random bits in a single clock cycle. We demonstrate how metastable circuits based on ambipolar transistors allow doubling the throughput as compared to a similar standard CMOS-based design. The proposed design is compact in terms of the number of transistors per block ($60\,\%$ less transistors), power consumption (saving $94.5\,\%$ leakage power and $70.7\,\%$ dynamic power) and path delay ($77.3\,\%$ reduction) with respect to its CMOS counterpart.**

I. Introduction

Ambipolar transistors belong to an interesting class of emerging nanotechnologies which can be configured at runtime to behave either as a p-type or an n-type transistor. Transistors based on these technologies (such as silicon [9, 6] or germanium nanowire [29]) show electrical symmetry in both p- and n-type polarity. Owing to their transistor-level reconfiguration, devices made of such nanotechnologies are often termed as *Reconfigurable FETs* (RFETs). Reconfigurable transistors allow logic selection/reconfiguration without the need of extra multiplexers [21]. RFETs can encapsulate more logic and functionality into a smaller area and are able to achieve reduced power consumption and higher speed during their operation as compared to their CMOS counterparts [16, 21]. Due to their extended functionality [28], they have shown great potential for hardware security applications, particularly in the domain of logic locking and layout camouflaging [3].

In this work, we study metastability effects in RFETs-based circuit and demonstrate how ambipolarity at the transistor level can be exploited in circuit designs to enable security primitives such as random number generator. Random number generators extract noise from chaotic physical processes in the form of an unpredictable sequence of bits (e.g., thermal noise, flicker noise, clock-jitter, metastable states, power supply fluctuations) [10, 19, 2]. and are typically used in security applications to generate unique secret keys. Conventional *True Random Number Generators* (TRNGs) exploit metastability in various CMOS logic, such as ring oscillators (ROs) [15] or metastable latches [14], as their source for randomness. However, emerging technologies [1, 31, 3, 4] offer new and interesting alternatives due to their smaller area and lower power consumption.

Conventionally, each random bit generated using metastability-based RNGs is a result of two cross-coupled elements entering into a metastable state per clock cycle. While in typical CMOS circuits, metastable behavior of bi-stable circuits like cross-coupled inverters and latches is used to generate random bits, in RFETs-based circuits, two metastable states from the same bistable circuit can be realised using this transistor-level ambipolarity. Based on this core concept, we demonstrate two circuit components essential for a metastablity-based random number generators– a metastable *Minority-based SR latches* and a configurable *dual-edge triggered True-Single Phased Clock D-Flip Flop* (TSPC DFF).

978-1-6654-2614-5/21/$31.00 ©2021 IEEE

In our proposed design, the property of transistor-level reconfigurability allows to have both cross-coupled NAND and NOR operations (from *Minority* logic) in a single clock cycle, which are triggered into metastable states at the rising and falling clock edges respectively, thereby sampling two random bits (using dual-edge triggered TSPC DFF) per clock cycle. Our major contributions are as follows:

- Using Verilog-A model for RFETs, we demonstrate functioning of a *Minority*-based SR latch which allows reconfiguration between a NAND and NOR-based SR latch. This is essential to achieve double throughput and forms the core for our proposed RNG.
- Design of a reconfigurable dual edge-triggered D-flip flop using RFETs based on TSPC logic is proposed which allows random number sampling at both the edges of the clock.
- We show that the raw random bit sequences obtained from the proposed RNG have sufficient entropy to pass majority of the statistical tests. While all the tests are not expected to pass, as generated bit sequences do not follow a uniform distribution, post-processing becomes a prerequisite in this case. We show that on application of commonly used random extractors, all tests (except one) pass at three different frequencies.

We also show that the proposed design is more efficient in terms of number of transistors (60% saving), delay (77.3% reduction) and power consumption (94.5% lower leakage power and 70.7% lower dynamic power) as compared to a similar architecture of an RNG using CMOS technology. Experimental evaluations over NIST benchmark suite [22] at three different frequencies—10 MHz, 100 MHz and 200 MHz, are carried out to empirically demonstrate that the generated bit sequence has high entropy.

II. Background

A. Reconfigurable nanowire-based FETs

Reconfigurable transistor technology is exhibited by transistors made with various materials such as silicon [9, 6], germanium [29] or MoTe$_2$ [17]. In this paper, we focus on nanowire-based RFETs since it is one of the most actively researched emerging technologies with Verilog-A models [8] and has been evaluated with a physical synthesis flow [20]. Silicon or germanium nanowire-based RFETs follow similar CMOS-like top-down fabrication process [24] and come in stacked nanowire geometry [34] and hence are commercially feasible. Moreover, all manufacturing related process which are applicable in case of CMOS can be applied to RFETs as well [16, 25]. Further details regarding the physics of such reconfigurable devices can be found in [16].

Reconfigurable nanowire-based transistors, unlike conventional CMOS based transistors, feature two kinds of gates- *program gate* that makes the device p-type or n-type by selectively suppressing the injection of one type of carrier, and *control gate* that receives voltage input to the FET and modulates the injection of the other type of carrier RFETs further allow multiple gate terminals on the same channel in a wired-AND configuration [23] thereby dramatically reducing transistor count in digital circuits as well as the parasitics and delays as compared to conventional CMOS devices [21, 28]. This enables logic gate (as shown in Fig. 1a and 1b) and circuit designs using fewer RFETs [21].

Circuits based on RFETs also follows the same complementary pull-up and pull-down networks for their functioning as in CMOS-based circuits. The only difference here is due to this simultaneous switch between the pull-up and pull-down network, multiple functionalities are realised.

An important aspect for nanowire-based reconfigurable nanotechnology which is favorable for the design of RNGs, is that, it is a dopant-free technology and hence, the numbers generated are totally an outcome of chaotic external noise.

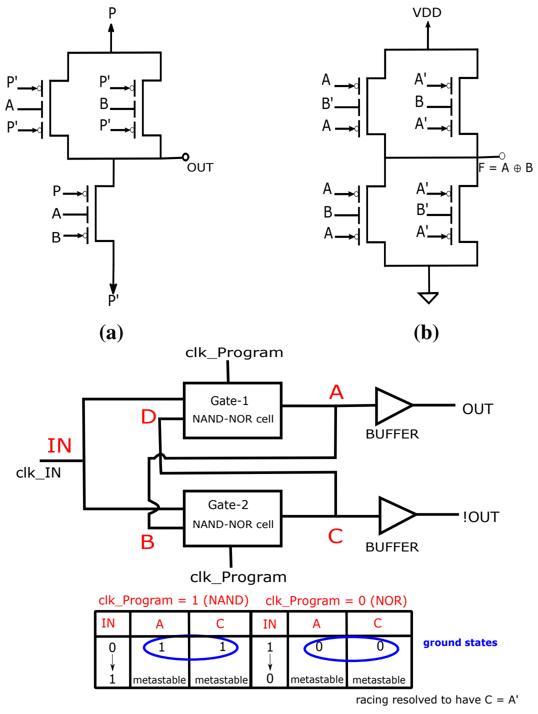
III. Design of the TRNG using RFETs

The potential of using transistor-level reconfiguration in RFETs to develop compact and power-efficient circuits with less parasitics motivates us to employ them for our proposed design. In this work, we use vertically-stacked all-around *Three-Independent-Gate SiNW RFETs*, called TIGFETs [36] to design digital circuits.

A. Minority gate based SR latch for proposed design

Conventional TRNGs employ the metastable state attained by cross-coupled elements as a source of randomness. In the present work, metastability-based RNG is designed using reconfigurable *Minority* (MIN) gates (Fig. 1a). The Minority gate shown in Fig. 1a can be reconfigured into NAND and NOR depending upon the value of P [21]. This transistor-level reconfigurability is employed to design a configurable SR latch as shown in Fig. 1c.

Fig. 1c shows a single SR latch unit consisting of two cross-coupled MIN gates and two buffers. Two clock signals ($clk\_Program$ and clk_IN) with the same time period T are fed into the unit, clk_IN being a time-delayed version of $clk\_Program$, delayed by t_d satisfying the condition $t_d < T/2$. In the first half-period of $clk\_Program$ ($clk\_Program$ = '1'), the MIN gates behave as NAND gates and as the rising edge of the clk_IN signal occurs; (when clk_IN = '0'), the outputs of both the gates are '1' (ground state). Post the transition in clk_IN signal, the outputs begin to race and temporarily enter into metastability. However, owing to the noise, the output 'OUT' stabilises in order to generate a random bit ('0' or '1'). Similarly, in the second half-period of $clk\_Program$ ($clk\_Program$ = '0'), the MIN gates behave as NOR gates and the falling edge of the clk_IN signal occurs. This time in the ground state the outputs of both the gates are '0' and metastability is attained at the '1'→ '0' transition of clk_IN signal, which eventually results in another random bit. Thus, in one complete clock cycle, two random bits are generated implying that the throughput of the SR latch unit is twice the input clock frequency.



clk_Program = 1 (NAND)			clk_Program = 0 (NOR)		
IN	A	C	IN	A	C
0 → 1	1 metastable	1 metastable	1 → 0	0 metastable	0 metastable

(c)

Fig. 1: (a) A configurable *Minority* (MIN) gate behaving as a NAND gate when P = '1' and NOR gate when P = '0'. (b) An XOR gate in double-gate configuration [36]. (c) An SR latch unit for proposed design based on minority based NAND-NOR cell.

We can imagine the cross-coupled MIN gates (of same driving capability) in the SR latch unit (Fig. 1c) to be two cross-coupled inverters (such as in an SRAM cell) powered-ON when the input clock makes a '0' → '1' transition for $clk\_Program$ = '1' or when it makes a '1' → '0' transition for $clk\_Program$ = '0'. 'B' and 'D' are respectively the inputs to *Gate-2* and *Gate-1* while, 'A' and 'C' are respectively the outputs of *Gate-1* and *Gate-2*. The corresponding butterfly-curve in the *Voltage-Transfer Characteristic* (VTC) for the SR latch unit is shown in Fig. 2. It can be clearly seen that point 'X', which is the point of metastability, lies on the identity line. This means that

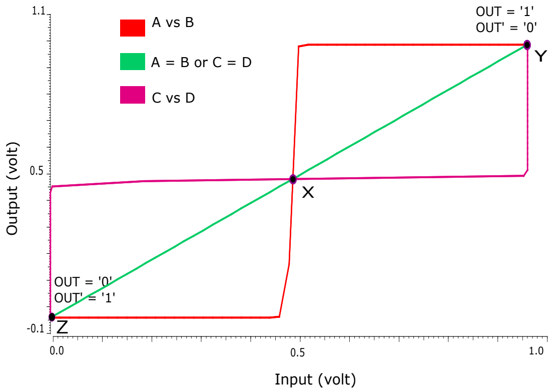


Fig. 2: Butterfly curve in the Voltage-Transfer Characteristic (VTC) for the SR latch unit (Fig. 1c) of the proposed design. B' and 'D' are respectively the inputs to *Gate-2* and *Gate-1*, while A' and 'C' are respectively the outputs of *Gate-1* and *Gate-2*

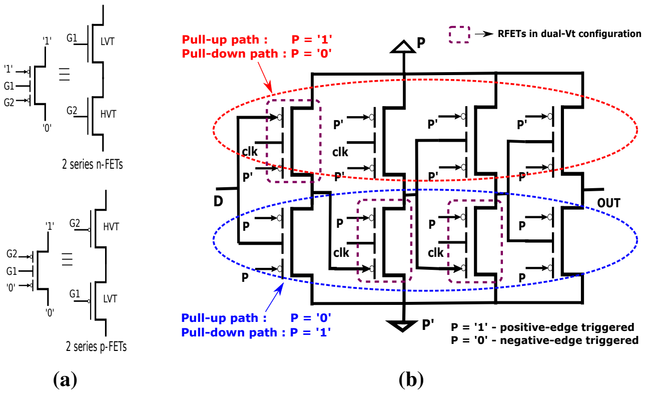


Fig. 3: (a) N-MOS and P-MOS transistor level equivalent models for TIGFETs in dual-threshold voltage configuration (b) A configurable dual edge-triggered D-flip flop based on TSPC logic style

both stable states demarcated by points 'Y' and 'Z' are equally preferred. Eventually, the latch attains either state 'Y' or 'Z' due to noise, thereby producing a random bit at the output (OUT).

B. *Dual edge-triggered TSPC-based D-flip flop*

The authors in [26] proposed a design of a single edge-triggered TSPC-based D-flip flop using RFETs that has been shown to have a reduced transistor count and area than its CMOS counterpart [35]. They employed a dual-threshold voltage configuration of the TIGFETs as shown in Fig. 3a for true single phase operation. In this design, input $G1$ has a *lower threshold voltage* (LVT) and input $G2$ has a *higher threshold voltage* (HVT) (corresponding to lower leakage current).

We exploit the runtime reconfigurability feature of RFETs to make the TSPC-based D-flip flop proposed in [26] dual-edge triggered. This can be done by using a program signal (P) instead of the power-rails as shown in Fig. 3b. If P = '1', the upper four transistors encircled in red provide the pull-up path while the lower four transistors encircled in blue provide the pull-down path. In this case, the flip flop samples data at the rising edge of the clock and hence, behaves as a positive edge-triggered flip flop. Conversely, if P = '0', the pull-up and pull-down paths get interchanged and the flip flop samples data at the falling clock-edge. This way it behaves as a negative edge-triggered flip flop. Dual-threshold voltage design style as shown in Fig. 3a has been adopted (for three transistors encircled in purple) to make the design compact and reduce leakage power consumption.

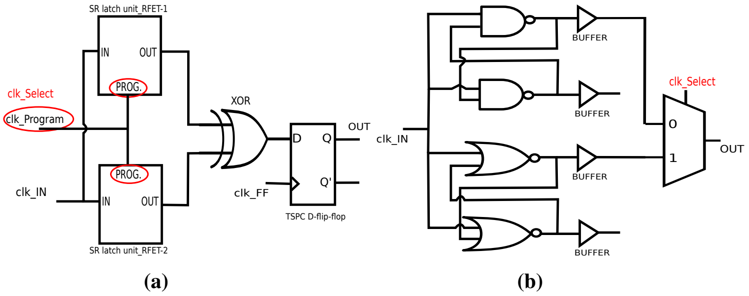


Fig. 4: (a) The simulation model for the proposed design consisting of two SR latch units, an XOR gate in DG configuration and the configurable dual edge-triggered D-flip flop **(The signals and nodes in the equivalent CMOS model have been marked in red)** (b) An SR latch unit for the CMOS equivalent of the proposed design

Thus, the same circuit of the flip flop can be reconfigured into both positive and negative edge-triggered functionalities based on the program signal during runtime. However, the same TSPC-based design of a D-flip flop in CMOS technology [35] cannot be reconfigured as both positive and negative edge-triggered and it also uses more number of transistors (11 transistors) as compared to our proposed design using RFETs (8 transistors).

C. *Complete circuit design*

Fig. 4a shows the complete circuit for the proposed design based on RFETs with all the components. Compact implementation of MIN gates (Fig. 1a), and the proposed TSPC-based D-flip flop has been carried out using dual-threshold-voltage design style that makes the design area-efficient with improved speed and reduced leakage power consumption [36]. For the simulation model of the proposed design, the output binary sequence can be assumed to be *i.i.d.* (independent and identically distributed). It is because before the occurrence of a metastability event, either at the rising or at the falling clock edge, the output node 'OUT' of the RNG attains a ground state in which it resets itself before generating another random bit. Hence, the model does not involve correlation between two consecutive bits generated at 'OUT' due to the metastability event.

Additionally, it has been mathematically proven in [30, 5] that by XOR-ing outputs of more than one RNG (in this case, the SR latch units), the randomness (entropy) of the resultant output sequence can be increased and the RNG becomes more robust against PVT variations. Hence for our simulations, we have XOR-ed (XOR gate Fig. 1b) the outputs of two SR latch units and fed the result into a dual edge-triggered TSPC-based D-flip flop (Fig. 3b). To the best of our knowledge, none of the earlier works have explored an RNG design using device-level reconfigurability offered by reconfigurable emerging nanotechnologies.

TABLE I: A comparison between proposed design SR latch unit and its CMOS equivalent

SR latch unit	St. power(nW)	Dy. power(nW)	Delay(ps)	#transistors
proposed design	16.85	79.65	206	26
CMOS eq.	308.25	271.5	909	65

IV. Experiments

A. Experimental setup

The simulation of the proposed design has been carried out in Cadence Virtuoso. The Verilog-A model for the RFET in three-independent gate configuration (TIGFET) from [8] was used during the circuit-level simulations. This model has been adapted to incorporate flicker and white noise parameters.

B. Comparison with an equivalent CMOS-based RNG

In the corresponding CMOS-based implementation (designed for double-throughput) of the proposed design, operating at supply voltage of 1.0 V, we have used PTM 16nm low power CMOS model for the simulation of the MOSFETs [32]. The SR latch unit in this case is shown in Fig. 4b consisting of two cross-coupled NAND gates, two cross-coupled NOR gates, four buffers and one 2×1 MUX.

TABLE I presents a comparison between the SR latch units of the simulated proposed design and its CMOS counterpart (both for double-throughput) on the basis of transistor count, power consumption and delay operating at a clock frequency of 100 MHz. It can be seen that there is a 60% saving in the number of transistors by employing an RFET-based design. Furthermore, we observe a 94.5% reduction in leakage power, 70.7% reduction in dynamic power and 77.3% reduction in path delay in case of the SR latch unit based on RFETs with respect to its CMOS equivalent. This reduction in delay and hence power can be ascertained due to the fact that RFETs have lower parasitics as compared to series connection of transistors [28]. Note, from Fig. 1c, the path delay for the SR latch unit of proposed design includes only clk_IN to 'OUT' delay (inclusive of buffer delay) while, for the equivalent CMOS SR latch unit (Fig. 4b), it includes clk_IN to output delay of the NOR-based SR latch (including buffer delay) and delay of the MUX.

C. Simulation for proposed design

For the circuit shown in Fig. 4a, the transient waveforms for the input and output signals are shown in Fig. 5. All the analyses have been done for supply voltage of 1.0 V. Here, all the clock signals *viz.*, $clk\_Program$, clk_IN and clk_FF operate at a frequency of 100 MHz. Also, clk_IN and clk_FF are time-delayed versions of $clk\_Program$, delayed by $1ns$ and $3ns$ respectively. We run a transient analysis using the embedded transient noise feature in *Virtuoso Spectre Circuit Simulator* and obtain random bits at the 'OUT' node after every 5 ns. The ground states (**G.S.**) and metastable states (**M.S.**) attained by the RNG in a clock cycle have been marked in Fig. 5. We find that the throughput is equal to 200 Mbps which is twice the input clock frequency of 100 MHz. The above procedure is repeated for clock frequency of 200 MHz and 10 MHz as well.

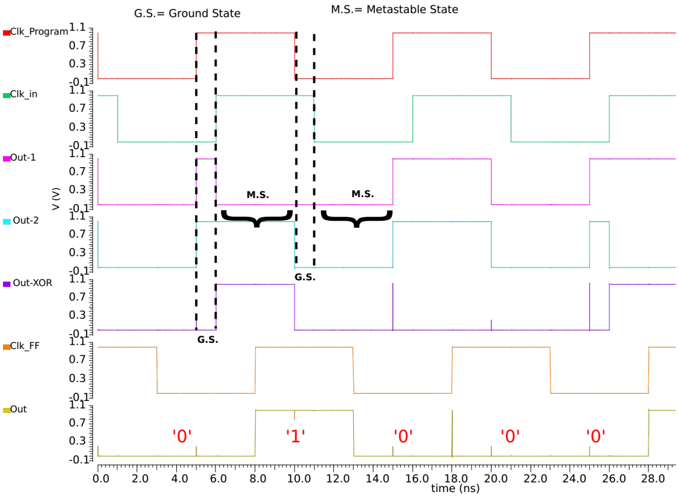


Fig. 5: Transient waveforms on operating the proposed design at 100 MHz clock frequency with the ground states and metastable states marked for a clock cycle

D. Statistical evaluation of the generated bit sequence

In order to carry out a statistical evaluation, we use the *National Institute of Standards and Technology* (NIST) benchmark suite [22]. The test suite is used to evaluate the randomness of the binary sequences generated at the output of an RNG. This benchmark suite is commonly used to evaluate both hardware and software-based RNGs and indicates whether the bitstream is likely to come from an uniform i.i.d. [22, 12]. The test suite consists of several benchmarks that are run on all the binary sequences to evaluate different characteristics of the generated bit sequence. For each benchmark in the suite, two statistical metrics are used namely, *success rate* (S-rate) and *P'-value*. We calculate *P-value* corresponding to each sequence per benchmark, which denotes the distance between the test results and the expected results. It needs to be greater than a particular threshold, to conform that the specific benchmark has passed successfully. The success rate for a benchmark is the proportion of the binary sequences passing the benchmark, while the P'-value quantifies the uniformity in the distribution of all the P-values for a benchmark in the suite. The P'-value is a number between 0 and 1. An RNG is said to pass a benchmark if the success rate and the P'-value are greater than a threshold [22].

Owing to the complexity of the simulations due to large number of parameters, high precision and a bulk of simulated and stored data points, two types of analysis are carried out– Firstly, 110 sequences of 1000 bits each are formed from the overall 110,000 bits for each frequency of operation and are subjected to various statistical evaluation. This is required to evaluate the randomness in smaller chunk of the bit patterns. Secondly, statistical analysis is performed by consolidating all the 110 sequences, thereby forming a 110,000-bits long sequence each for the clock frequencies of 10 MHz, 100 MHz and 200 MHz. This is necessary to carry out evaluation for the complete sequence. By performing a transient analysis in the *Spectre* simulator, we generate 110,000 bits as output from the proposed design for the clock frequencies of 10 MHz, 100 MHz and 200 MHz respectively. The statistical tests are performed

TABLE II: Results of the NIST benchmark suite for the proposed design using 110 sequences of 1000 bits each. The threshold for P'-value is 0.0001 and for success rate is 105/110 = 0.954 [12, 19]. **(Failed benchmark results have been highlighted in red)**

	10 MHz		100 MHz		200 MHz	
Benchmark name	**S-rate**	**P'-value**	**S-rate**	**P'-value**	**S-rate**	**P'-value**
Monobit Frequency	0.991	0.2238	0.964	0.0052	1.0	0.7757
Block frequency	0.982	0.0004	0.936	3.19E-12	0.918	1.08E-10
Runs	1.0	0.7399	1.0	0.6276	0.973	0.3807
Longest run	1.0	0.5159	1.0	0.5899	1.0	0.1431
DFT	0.991	7.99E-18	0.973	5.04E-14	0.991	2.25E-20
Overlap template matching	0.964	0.0004	0.945	0.0020	0.964	2.02E-07
Non-overlap template matching	0.991	0.0064	1.0	0.3218	1.0	0.6655
Cumulative sum-1	0.991	0.1359	0.973	0.0002	1.0	0.5526
Cumulative sum-2	1.0	0.2820	0.945	6.66E-05	1.0	0.5159
Serial - 1	0.991	0.7216	0.954	0.0011	0.973	0.0027
Serial - 2	0.982	0.3654	0.991	0.9230	0.973	0.1506
Approximate entropy	0.991	0.8421	0.964	0.0597	0.982	0.0083
Binary matrix rank	1.0	0.2949	0.991	0.0885	0.982	0.2346

TABLE III: P-values for the NIST benchmarks with the binary sequences of 110,000 bits from the proposed design taken altogether (without post-processing). The threshold for P-value is 0.01 for a benchmark to pass [22]. **(Failed benchmark results have been highlighted in red)**

Benchmark name	**10 MHz**	**100 MHz**	**200 MHz**
Monobit Frequency	0.59	0.21	0.51
Block frequency	0.01	7.32E-05	0.57
Runs	6.15E-08	1.33E-10	6.78E-07
Longest run	0.11	0.68	0.51
DFT	0.72	0.27	0.23
Overlap template matching	0.02	0.46	0.02
Non-overlap template matching	0.42	0.03	0.32
Cumulative sum - 1	0.39	0.25	0.35
Cumulative sum - 2	0.64	0.12	0.67
Serial - 1	6.44E-07	1.23E-17	1.76E-17
Serial - 2	0.08	0.01	0.02
Approximate entropy	9.19E-07	5.69E-17	5.10E-17
Binary matrix rank	0.68	0.35	0.06
Shannon Entropy	**0.9999980685**	**0.9999896832**	**0.9999972186**

on these generated random bits. Only those benchmark are performed from the suite which can be run on the generated number of bits (110,000)[1].

V. Results and Discussion

TABLE II shows the the NIST benchmark results (success rates and P'-values) for the 110 raw bit sequences generated from the proposed design operating at clock frequencies of 10 MHz, 100 MHz and 200 MHz. TABLE III shows the the NIST benchmark results (P-values) and Shannon entropies for the 110,000 bits-long binary sequence each for the clock frequencies of 10 MHz, 100 MHz and 200 MHz.

We observe that when the RNG operates at a lower frequency of 10 MHz, the success-rate for raw output binary sequence passes all the NIST benchmarks as shown in TABLE II. At higher values of operating frequency or throughput, only a few benchmarks (in this case, *Block frequency*, *DFT*, *Overlap template matching* and *Cumulative sum - 2* benchmarks) fail from the perspective of success rate and/or P'-value. However, the observed success rates for the failed benchmarks are still more than 90% for all the binary sequences tested. Moreover, we observe that the *Monobit Frequency* benchmark passes for both higher and lower frequencies of operation. It implies that the number of '0's and '1's produced by the RNG are approximately equal as would be expected for a truly random sequence [22]. It is important to note here that the *Monobit Frequency* benchmark is compulsory to pass as other subsequent benchmarks in the NIST suite depend on it [22].

Hardware generated bit sequence generally has a skewed (biased) distribution of '0's and '1's [7, 2]. Such statistical weaknesses may also arise from PVT variations that hamper the source of entropy among other factors. Hence, certain benchmarks in the NIST test suite are expected to fail without postprocessing. Failing the NIST test suite does not imply that the RNG is not random; it only means that the RNG distribution does not appear to be uniformly distributed. This is taken care by commonly used post-procesing techniques [13]. Postprocessing is generally used regardless to compensate for process variation and interference effect.

A. Post-processing

In practise, randomness extractors can be employed to compensate the output properties of an RNG to be uniform and *i.i.d.* for cryptographic use [13]. However, the closer the output bits are towards being uniform and *i.i.d.*, the simpler the random extractors can be used. Since in the previous experiment, only few benchmarks failed we use a simple postprocessing technique. In this algorithm, the raw bit stream is grouped into non-overlapping pairs of consecutive bits. For each pair, in case both the bits are equal then we discard the pair, otherwise, the first bit in the pair is taken to be the output. Thus, this algorithm essentially uses two input bits to produce either zero or one output bit. We employ this algorithm to post-process the raw binary sequences consisting of the entire set of 110,000 bits from the proposed design operating at frequencies of 10 MHz, 100 MHz and 200 MHz. Subsequently, all the NIST benchmarks are run on the processed output sequences and the corresponding p-values are recorded in TABLE IV. On comparing the data shown in TABLE III and TABLE IV, we observe that all scenarios (except one) in the NIST benchmark suite passes after postprocessing, for all the three frequencies of operation. It is to be noted that this post-processing algorithm is not a qualified or a standard cryptographic function or a randomness extractor. It is solely used to show that the generated bit sequence has sufficient randomness to pass the NIST test suite.

Typically every metastability-based RNG has an integration of two units – a physical source of entropy (in this case, the SR latch units generating the raw binary sequences) and a post-processing unit that transforms the raw binary sequences into a sequence which is computationally tedious to differentiate from a purely random sequence [7, 27]. However, post-processing leads to reduction in the throughput irrespective of the underlying technology. Post processing is also used to compensate for PVT variations and inference effects [2, 13].

B. Robustness against environmental factors

Since RFETs follow the same CMOS manufacturing process and are made of similar materials such as silicon or germanium

[1] This is because the remaining benchmarks in the suite (*Maurer's Universal statistical, Linear, Random excursion* tests) require more than 10^7 bits for evaluation and it would amount to an infeasible time duration to generate the bits using simulation [19] for a TCAD-based verilog-A model for RFETs [8]

TABLE IV: P-values for the NIST benchmarks after Von-Neumann post-processing of the raw binary sequence of 110,000 bits from the proposed design. The threshold for P-value is 0.01 for a benchmark to pass [22]. **(Failed benchmark results have been highlighted in red)**

Benchmark name	**10 MHz**	**100 MHz**	**200 MHz**
Monobit Frequency	0.95	0.46	0.63
Block frequency	0.11	0.75	0.47
Runs	0.87	0.00025	0.04
Longest run	0.39	0.91	0.25
DFT	0.46	0.16	0.26
Overlap template matching	0.25	0.19	0.03
Non-overlap template matching	0.81	0.16	0.998
Cumulative sum - 1	0.44	0.20	0.74
Cumulative sum - 2	0.39	0.72	0.81
Serial - 1	0.59	0.02	0.22
Serial - 2	0.50	0.68	0.28
Approximate entropy	0.60	0.02	0.24
Binary matrix rank	0.31	0.15	0.87
Shannon Entropy	**0.9999999011**	**0.9999854835**	**0.9999937737**

nanowires, they are commercially viable without any integration barrier [16]. Hence, all measures which are applicable in case of CMOS to make RNGs robust are valid in case of RFETs. Measures such as self-calibrating feedback loops [33] that continuously monitor the output bits to detect the presence of any bias and then counter such effects are easily applicable in case of RFETs. Such methods can also tackle any adversarial attempt using temperature or other variations as RFETs are also voltage-driven device similar to CMOS.

VI. Conclusion and Future Directions

In the present work we propose a design of a metastability-based TRNG using emerging reconfigurable nanotechnology. The major focus of this work is to explore how ambipolarity can be exploited for enhanced throughput in metastable circuits. Using runtime reconfigurability, the RNG is shown to use less hardware, be compact in terms of transistor count per block (60% saving in the transistor count), consume less power (94.5% saving in leakage power and 70.7% saving in dynamic power) and has a lower critical path delay (77.3% reduction in delay) with respect to its equivalent CMOS counterpart. Statistical evaluations using our proposed proposed design were performed and results were presented. Future work directions include stochastic modeling of RFETs-based latches as proposed for CMOS-based metastable circuits in [18] and evaluations over AIS-31 benchmark suite to give more statistical guarantees [11]. Additionally, detailed robustness analysis using Monte-Carlo simulations require mature models of RFETs and is a part of our ongoing work.

References

[1] Hiroyuki Akinaga and Hisashi Shima. “Resistive random access memory (ReRAM) based on metal oxides”. In: *Proceedings of the IEEE* (2010).

[2] Swarup Bhunia and Mark Tehranipoor. “Chapter 12 - Hardware Security Primitives”. In: *Hardware Security*. Morgan Kaufmann.

[3] Y. Bi et al. “Enhancing hardware security with emerging transistor technologies”. In: *GLSVLSI*. 2016.

[4] An Chen et al. “Using emerging technologies for hardware security beyond PUFs”. In: *DATE*. 2016.

[5] Robert B Davies. “Exclusive OR (XOR) and hardware random number2 generators”. In: *Retrieved May* (2002).

[6] M. De Marchi et al. “Polarity control in double-gate, gate-all-around vertically stacked silicon nanowire FETs”. In: *IEDM*. 2012.

[7] J. D. J. Golic. “New Methods for Digital Generation and Postprocessing of Random Data”. In: *IEEE Trans. on Computers* (2006).

[8] G. Gore et al. “A Predictive Process Design Kit for Three-Independent-Gate Field-Effect Transistors”. In: *VLSI-SoC*. 2019.

[9] André Heinzig et al. “Reconfigurable Silicon Nanowire Transistors”. In: *Nano letters* (Nov. 2011).

[10] D. E. Holcomb, W. P. Burleson, and K. Fu. “Power-Up SRAM State as an Identifying Fingerprint and Source of True Random number2s”. In: *IEEE Trans. on Computers* (2009).

[11] Wolfgang Killmann and Werner Schindler. “A proposal for: Functionality classes for random number generators”. In: *ser. BDI, Bonn* (2011).

[12] Song-Ju Kim, Ken Umeno, and Akio Hasegawa. “Corrections of the NIST Statistical Test Suite for Randomness”. In: (Feb. 2004).

[13] Siew-Hwee Kwok et al. “A comparison of post-processing techniques for biased random number generators”. In: *IFIP International Workshop on Information Security Theory and Practices*. Springer. 2011, pp. 175–190.

[14] Sanu K Mathew et al. “2.4 Gbps, 7 mW all-digital PVT-variation tolerant true random number generator for 45 nm CMOS high-performance microprocessors”. In: *IEEE Journal of Solid-State Circuits* (2012).

[15] Sanu K Mathew et al. “μ RNG: A 300–950 mV, 323 Gbps/W all-digital full-entropy true random number generator in 14 nm FinFET CMOS”. In: *IEEE Journal of Solid-State Circuits* (2016).

[16] Thomas Mikolajick et al. “The RFET - A reconfigurable nanowire transistor and its application to novel electronic circuits and systems”. In: *Semiconductor Science and Technology* (Dec. 2016).

[17] Shu Nakaharai et al. “Electrostatically Reversible Polarity of Ambipolar α-MoTe2 Transistors”. In: *ACS Nano* (2015). PMID: 25988597.

[18] R. J. Parker. “Entropy justification for metastability based nondeterministic random bit generator”. In: *2017 IEEE IVSW*. 2017.

[19] B. Perach and s. kvatinsky. “An Asynchronous and Low-Power True Random number2 Generator Using STT-MTJ”. In: *IEEE TVLSI* (2019).

[20] S. Rai et al. “A physical synthesis flow for early technology evaluation of silicon nanowire based reconfigurable FETs”. In: *2018 Design, Automation Test in Europe Conference Exhibition (DATE)*. 2018.

[21] S. Rai et al. “Designing Efficient Circuits Based on Runtime-Reconfigurable Field-Effect Transistors”. In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (2019).

[22] Andrew Rukhin et al. “NIST Special Publication 800-22: A Statistical Test Suite for the Validation of Random number2 Generators and Pseudo Random number2 Generators for Cryptographic Applications”. In: *NIST Special Publication 800-22* (Apr. 2010).

[23] M. Simon et al. “A wired-AND transistor: Polarity controllable FET with multiple inputs”. In: *DRC*. 2018.

[24] M. Simon et al. “Bringing reconfigurable nanowire FETs to a logic circuits compatible process platform”. In: *NMDC*. 2016.

[25] M. Simon et al. “Top-Down Technology for Reconfigurable Nanowire FETs With Symmetric On-Currents”. In: *IEEE Trans. Nanotech.* (2017).

[26] X. Tang et al. “TSPC Flip-Flop circuit design with three-independent-gate silicon nanowire FETs”. In: *ISCAS*. 2014.

[27] Naoya Torii et al. “ASIC implementation of random number2 generators using SR latches and its evaluation”. In: *EJIS* (2016).

[28] J. Trommer et al. “Functionality-Enhanced Logic Gate Design Enabled by Symmetrical Reconfigurable Silicon Nanowire Transistors”. In: *IEEE Transactions on Nanotechnology* (2015).

[29] Jens Trommer et al. “Material Prospects of Reconfigurable Transistor (RFETs)–From Silicon to Germanium Nanowires”. In: *MRS* (2014).

[30] E. I. Vatajelu and G. Di Natale. “High-Entropy STT-MTJ-Based TRNG”. In: *IEEE TVLSI* (2019).

[31] Mengxing Wang et al. “Current-induced magnetization switching in atom-thick tungsten engineered perpendicular magnetic tunnel junctions with large tunnel magnetoresistance”. In: *Nature communications* (2018).

[32] Wei Zhao and Yu Cao. “New generation of predictive technology model for sub-45nm design exploration”. In: *ISQED*. 2006.

[33] Kaiyuan Yang, David Blaauw, and Dennis Sylvester. “A robust- 40 to 120° C all-digital true random number generator in 40nm CMOS”. In: *Symposium on VLSI Circuits*. IEEE. 2015.

[34] Peide Ye, Thomas Ernst, and Mukesh V Khare. “The last silicon transistor: Nanosheet devices could be the final evolutionary step for Moore’s Law”. In: *IEEE Spectrum* (2019).

[35] J. Yuan and C. Svensson. “High-speed CMOS circuit technique”. In: *JSSC* (1989).

[36] J. Zhang et al. “Configurable Circuits Featuring Dual-Threshold-Voltage Design With Three-Independent-Gate Silicon Nanowire FETs”. In: *IEEE TCAS-I* (2014).