IEEE Journal on Exploratory Solid-State Computational Devices and Circuits

Received 28 October 2022; revised 22 November 2022; accepted 29 November 2022.
Date of publication 8 December 2022; date of current version 27 December 2022.

Digital Object Identifier 10.1109/JXCDC.2022.3227062

Review of Magnetic Tunnel Junctions for Stochastic Computing

BRANDON R. ZINK, YANG LV, and JIAN-PING WANG (Fellow, IEEE)

Electrical and Computer Engineering Department, University of Minnesota Twin Cities, Minneapolis, MN 55455 USA

CORRESPONDING AUTHOR: J.-P. WANG (jpwang@umn.edu)

This work was supported in part by the Center for Probabilistic Spin Logic for Low-Energy Boolean and Non-Boolean Computing (CAPSL), one of the Nanoelectronic Computing Research (nCORE) centers as task 2759.001; in part by the Semiconductor Research Corporation (SRC) Program through the National Science Foundation (NSF) under Grant 1739635; in part by the Spintronic Materials for Advanced Information Technologies (SMART), one of the seven centers of nCORE, a Semiconductor Research Corporation Program through the National Institute of Standards and Technology (NIST); in part by the SMART through the University of Minnesota (UMN) Materials Research Science Engineering Center (MRSEC) Program under Award DMR-1420013; in part by the Applications and Systems driven Center for Energy-Efficient Integrated NanoTechnologies (ASCENT), one of six centers of Joint University Microelectronics Program (JUMP), a Semiconductor Research Corporation Program through Microelectronics Advanced Research Corporation (MARCO) and Defense Advanced Research Projects Agency (DARPA); in part by DARPA through "Advanced Magnetic Tunnel Junctions (MTJs) for Computation in and Near Random Access Memory" under Grant HR001117S0056-FP-042; in part by the NIST; in part by NSF Scalable Parallelism in the Extreme (SPX) Program, under Grant 1725420; and in part by Cisco Inc.

ABSTRACT Modern computing schemes require large circuit areas and large energy consumption for neuromorphic computing applications, such as recognition, classification, and prediction. This is because these tasks require parallel processing on large datasets. Stochastic computing (SC) is a promising alternative to conventional binary computing schemes due to its low area cost, low processing power, and robustness to noise. However, the large area and energy costs for random number generation with CMOS-based circuits make SC impractical for most hardware implementations. For this reason, beyond-CMOS approaches to random number generation have been investigated in recent years. Spintronics is one of the most promising approaches due to the intrinsic stochasticity of the magnetic tunnel junction (MTJ). In this review article, we provide an overview of the literature published in recent years investigating the tunable, intrinsic stochasticity of MTJs and proposing practical methods for random number generation using spintronic hardware.

INDEX TERMS Magnetic tunnel junctions (MTJs), random number generators (RNGs), spintronic devices, stochastic-bit generators (SBGs), stochastic computing (SC).

I. INTRODUCTION

MODERN computing schemes based on binary representation have several challenges in future artificial intelligence (AI) applications [1], [2]. These challenges are mainly due to the size of the datasets and the large number of transistors required to process the data [1]. Furthermore, while modern computers excel at speed and precision, they are very inefficient at tasks involving recognition, complexity, and ambiguity [3]. This is because classical von-Neumann architectures are not very tolerant to hardware faults, noisy data, or any type of variations in the input data [4].

One technique that can address the shortcomings of modern conventional computing is stochastic computing (SC). In SC, data are represented as a stream of random binary bits, and the numeric values are the percentage of the random bits being "1" [5], [6], [7], [8], [9]. For example, given a bitstream S with N stochastic bits, the number interpreted by that bitstream (S) is the number of ones in S divided by N. Note that even though each element within the bitstream is binary, the numbers interpreted in the bitstream are analog.

There are two key advantages to SC over conventional, deterministic models. The first is that the stochastic numbers are not dependent on position of elements in the bitstream. This gives SC schemes a high degree of error tolerance and noise resilience [9], [10], [11], [12], [13]. The second is that a wide variety of arithmetic functions can be performed using a very low number of logic gates [9], [11], [14], [15], [16]. For example, multiplication of two stochastic bitstreams can be done using either a single AND gate for numbers with unipolar representation (numeric range between 0 and 1), as illustrated in Fig. 1(a), or a single XNOR gate for numbers with bipolar representation (numeric range between -1 and $+1$). Furthermore, scaled addition can be done using a 2×1 multiplexer by generating an additional input bitstream S, which has a constant, predetermined probability at the selector input, as illustrated in Fig. 1(b). Additional logic

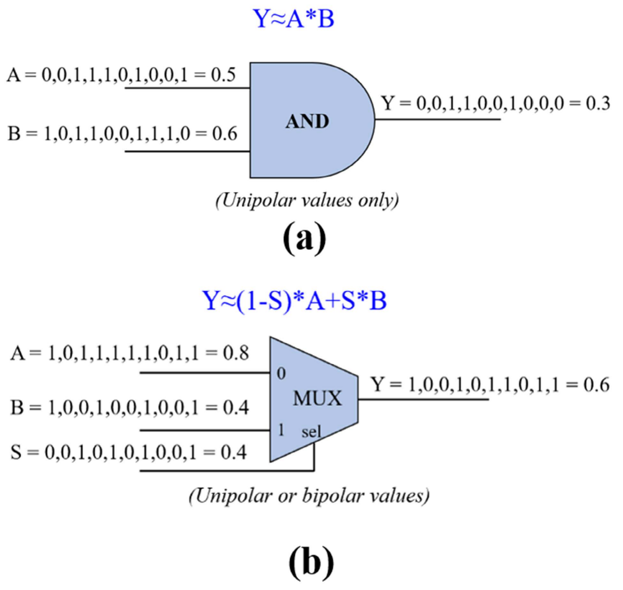
Y≈A*B
A = 0,0,1,1,1,0,1,0,0,1 = 0.5
B = 1,0,1,1,0,0,1,1,1,0 = 0.6
AND
Y = 0,0,1,1,0,0,1,0,0,0 = 0.3
(Unipolar values only)
(a)
Y≈(1-S)*A+S*B
A = 1,0,1,1,1,1,1,0,1,1 = 0.8
B = 1,0,0,1,0,0,1,0,0,1 = 0.4
S = 0,0,1,0,1,0,1,0,0,1 = 0.4
0
MUX
1
sel
Y = 1,0,0,1,0,1,1,0,1,1 = 0.6
(Unipolar or bipolar values)
(b)

FIGURE 1. Logic gates used for efficient implementation of (a) unipolar multiplication (AND gate) and (b) addition scaled by a factor of *S* (2 × 1 MUX).

gates can be introduced for highly efficient implementation of more complicated functions, such as high-order polynomials [8], [9], [10], hyperbolic tangent [16], and exponentials [11], [15], [16].

The major drawback to SC is the trade-off between numeric resolution and computation delay. When computing on bitstreams with length N, the numeric resolution is ideally $1/N$. To increase this resolution, the bitstream length could increase to $2N$, thus making the numeric resolution $1/2N$. However, this would double the number of computation cycles required to complete the task. For this reason, SC is not a very attractive solution for tasks, which require solving complex numeric problems with stringent accuracy requirements. However, there are several applications in neuromorphic computing, where high mathematical precision is not necessary, and SC is a very promising solution [9].

The error resilience and fault-tolerance inherent in SC make it a strong candidate for tasks, such as recognition, feature extraction, and classification [10], [11], [12], [13], [17]. This makes SC a promising solution for applications where approximations are used to filter noncritical information from large datasets; therefore, SC is a strong solution for these types of tasks.

One of the most common applications that have been proposed for SC is image processing. This is because various algorithms that are key for image processing tasks, such as edge detection, contrast stretching, kernel density estimation, and local image thresholding, can be efficiently performed using SC-based methods [10], [12], [13], [15], [17]. Not only does SC require less hardware usage than conventional approaches for each of these algorithms, but it is also more error tolerant. Therefore, SC-based methods can process very noisy images without significant performance degradation. For simple algorithms, such as edge detection, SC approaches consume more power than conventional approaches [15]. However, for complex algorithms, such as kernel density approximation, SC approaches consume less power [13], [15]. Furthermore, computation delay is significantly larger in SC than for conventional methods for several image processing algorithms. However, there are some applications, such as local image thresholding [12] and low-precision real-time image processing [17], where SC approaches are faster than conventional approaches.

The significant reduction in circuit area for computation in SC also makes it a promising solution in artificial neural networks [18], [19], [20], [21]. The simplified architecture allows for SC-based methods to easily increase in size, as the size of the datasets or the complexity of the tasks increases. Furthermore, SC circuits provide more design flexibility in that the trade-off between computation time and accuracy can be varied without any hardware changes. SC can also improve the circuit area, power consumption, and energy efficiency in deep neural networks, since complex activation functions and inference algorithms can be efficiently performed in SC [22], [23], [24].

SC is also a very promising solution for applications that process data with inherent randomness. For example, low-density parity check (LDPC) decoders are very powerful linear error correcting codes with decoding capabilities; however, the circuits required to implement these codes suffer from high area consumption but with low logic utilization due to the random location of ones in the parity-check matrix. The most area-efficient LDPC decoders are those based on SC [25], [26], [27], where hardware consumption is reduced by 40% on field-programmable gate arrays (FPGAs) [25] and 73.8% on the IEEE 802.15.3c decoder chips [27]. Furthermore, SC-based LDPC decoders can reduce the number of routing wires by 90% and improve the energy efficiency by 11.5% [27] while maintaining similar error floors as conventional approaches. Other examples of SC applications are Bayesian systems and weather prediction. Stochastic circuits can build Bayesian inference systems and Bayesian belief networks with high inference accuracy, fast speed, and low-power consumption for object location [28], [29] and heart disaster prediction tasks [29]. Another example is that stochastic neural networks can predict wind speed within 1-m/s accuracy with 18% reduction in circuit area [30].

Despite the promising prospects of SC, the hardware costs for generating stochastic bits using modern CMOS technology make the realization of SC in future technology unfeasible. Generating stochastic bits using CMOS-based platforms are typically done using linear feedback shift registers (LFSRs) [31] or ring oscillator-based circuits [8]. In some cases, the circuits required for generating stochastic bitstreams take 80% of the total circuit and 80% of the total energy consumption [10]. This means that the total circuit area and total energy consumption may not be reduced in SC compared with conventional methods, when considering the hardware used for generating stochastic bits. For this reason, alternative methods and technologies should be considered for generating stochastic bitstreams in SC. One promising approach is to exploit the intrinsic stochasticity of spintronic devices for random number generation. Magnetic tunnel junctions (MTJs) are highly influenced by thermal fluctuations at the atomic level; therefore, a single MTJ is highly effective at generating tunable random numbers. By utilizing this property, the large, energy consuming circuits for CMOS-based random number generators (RNGs) can be replaced by a single, nano-sized MTJ in SC circuits. In this report, we investigate the prospects of MTJs to perform as RNGs for SC as well as MTJ-based circuits to perform stochastic computation.

II. MAGNETIC TUNNEL JUNCTIONS

Magnetic random access memory (MRAM) has been developed for commercial purposes in recent years [32], [33], [34], [35], [36], [37], [38]. The elementary storage component in state-of-the-art MRAM is the MTJ, which is a two-terminal nanomagnetic device consisting of several thin-film magnetic devices, which consists of several ultrathin layers forming a nanopillar. There are three key layers that determine the properties of the MTJ, which are the fixed layer, the tunnel barrier, and the free layer. The fixed and the free layer are both ferromagnetic layers, where the magnetization has either in-plane orientation or an out-of-plane orientation. The magnetization of the fixed layer is pinned in a predetermined direction, and the magnetization of the free layer, on the other hand, can be oriented in the same direction (parallel) or in the opposite direction (antiparallel) to the magnetization of the fixed layer. The tunneling barrier is a thin insulating layer, typically MgO, which separates the fixed and free layers. In this chapter, we discuss the most common write mechanisms of MTJs and how they can be used for both memory applications as well as tunable RNGs.

A. MTJ PHYSICS

An MTJ can exist in one of two binary states, which is determined by the magnetization direction of the free layer ($\hat{m}_f$) relative to the magnetization direction of the fixed layer ($\hat{m}_p$). If $\hat{m}_f$ is parallel to $\hat{m}_p$, then the MTJ is said to be in the P-state; however, if $\hat{m}_f$ and $\hat{m}_p$ point in opposite directions, then the MTJ is in the AP-state. The electrical resistance of the MTJ (R_{MTJ}) is larger in the AP-state than in the P-state (R_{AP} and R_{P}, respectively). The difference in R_{AP} and R_{P} is described by the tunneling magnetoresistance (TMR) ratio, which is described with the following equation:

$$\mathrm{TMR} = \frac{R_{\mathrm{AP}} - R_{\mathrm{P}}}{R_{\mathrm{P}}} * 100\%. \tag{1}$$

The reason that the orientation of $\hat{m}_f$ is limited to AP- and P-state directions is due to magnetic anisotropy, which describes the preference for the magnetization of a ferromagnetic sample to lie in a particular direction. Anisotropy is defined as the energy per unit volume required to change the magnetization from the lowest energy direction (easy axis) to its highest energy direction (hard axis). A common metric used to describe the magnetic anisotropy of an MTJs free layer is the thermal stability factor (Δ), which is defined as the ratio between the energy required for $\hat{m}_f$ to switch states (E_{B}) and the thermal energy ($k_{\mathrm{b}}T$, where k_{b} is Boltzmann's constant, and T is the temperature). The thermal stability factor will determine the retention time of the MTJs free layer (τ), as expressed in the Néel–Arrhenius equation shown in (2). In (2), τ_0 is the inverse attempt frequency, which is often assumed to be 1 ns [39]. The retention time represents the expected amount of time the MTJ will retain its present magnetization state until thermal fluctuations cause $\hat{m}_f$ to switch.

For modern MRAM applications, it is desired that $\tau >$ 10 years, which corresponds to $\Delta > 60$. In the absence of external influences on $\hat{m}_f$, such as an external magnetic field or an applied voltage, E_{B} is given by the anisotropy energy; therefore, Δ is defined using the expression in (3), where V is the volume of the MTJ's free layer. This equation also shows the expression for Δ in terms of the anisotropy field (H_{K}), which represents the hypothetical field needed to align $\hat{m}_f$ along the hard-axis direction

$$\tau = \tau_0 \exp(\Delta) \tag{2}$$

$$\Delta = \frac{K_U V}{k_b T} = \frac{H_K M_S V}{2k_b T}. \tag{3}$$

B. STT AND SOT SWITCHING

The oldest method of switching MTJ states in MRAM cells is through an external magnetic field generated by current carrying wires. However, the dimensions of a field switching MRAM cell cannot be scaled to sizes below 90 nm due to the drastic increase in current required to generate fields with decreasing wire size [40], [41]; therefore, this method of switching MTJ states is no longer pursued in modern MRAM designs. Modern MRAM technologies utilize Spin-transfer torque (STT) switching, which is a much more practical writing method for MRAM cells than field switching. In an STT-MRAM cell, the write and read cycles are performed using a voltage or current applied across the MTJ (V_{MTJ} and I_{MTJ}). The mechanism that allows for STT switching is the spin filtering effect, which creates a spin-polarized current through the MTJ [42], [43], [44]. This spin polarization generates a torque on the free layer magnetization and eventually causes the MTJ to switch resistance states.

A major shortcoming of STT-MRAM is that the MTJs are susceptible to dielectric breakdown after multiple write cycles [45]. In recent years, the spin–orbit torque (SOT) effect has been studied in order to develop MRAM cells, which utilize SOT as the primary switching mechanism [46], [47]. Unlike STT switching, SOT switching does not require a voltage or current to be applied directly across the MTJs tunneling barrier. Instead, the current is passed through a nonmagnetic heavy metal (HM) with strong spin–orbit coupling (SOC; typically Ta or W). In an SOT-MRAM cell, the MTJ is fabricated on top of an HM-based SOT channel, so that the MTJ's free layer is adjacent to the SOT channel. Note

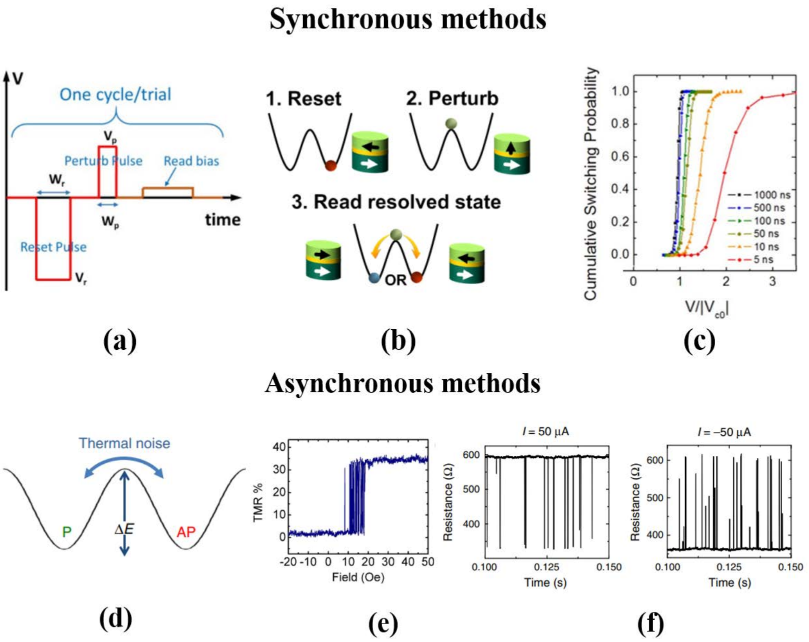

FIGURE 2. **(a) Task scheduling for synchronous probabilistic switching measurements. (b) Illustration of energy barrier during synchronous probabilistic switching measurements. (c) Examples of switching probability distribution versus perturb voltage amplitude at various pulse widths. (d) Illustration of the influence of thermal noise, (e) field switching plot, and (f) influence of a bias current on MTJs with low thermal stability. (a) Extracted from [84]. (b) Extracted from [63]; plots shown in (c) were extracted from [52]. (d) and (f) Extracted from [75]; plot shown in (e) was extracted from [55].**

that the SOT-MRAM cell has three terminals rather than two, since a small voltage is still applied across the MTJ during the read cycle. The mechanisms for SOT switching are two SOC phenomena, which are the spin-Hall effect (SHE) and the interfacial Rashba–Edelstein effect, both of which initiate spin accumulation at the HM/free layer interface. Spin-polarized electrons then diffuse into the FM layer, thus exerting a torque on the free layer magnetization. The spin current (J_{S}) generated from charge current (J_{C}) is expressed in (4), where σ is the polarization of the spin current, θ_{SH} is the spin-Hall angle, $\hbar$ is Planck's constant, and e is the charge of an electron

$$\vec{J}_{\mathrm{S}} = \frac{\hbar}{2e}\theta_{\mathrm{SH}}\left(\vec{J}_{\mathrm{C}} x \vec{\sigma}\right). \tag{4}$$

C. TUNABLE STOCHASTICITY IN MTJs

Magnetization switching in MTJs is strongly influenced by thermal fluctuations. This is because thermal fluctuations cause random nucleation to occur in the magnetic layers. These can be aided via STT or SOT effects, which lead to one of the most promising features of MTJs, tunable stochasticity. This means that, under certain conditions, the MTJ will switch states randomly with a controlled probability. In general, tunable stochasticity can be achieved in MTJs either through synchronous or asynchronous methods. Note that in either case, the switching probability can be controlled with either STT or SOT switching (recall Section II-B).

Synchronous methods use clocked cycles of sequential reset, perturb, and read voltage (or current) pulses, which is illustrated in Fig. 2(a) and (b) [48], [49], [50], [51], [52], [53]. These measurements can be collected for both AP-to-P and P-to-AP switching directions. The sign, amplitude, and duration of the reset pulse are set to switch the MTJ to the initial state with 100% probability. The perturb pulse then switches the MTJ with a probability determined by the amplitude and duration of the pulse (V_P and t_P, respectively). The last step of each cycle reads the final state of the MTJ with a voltage pulse small enough, so that it does not influence the state of the MTJ. The switching probability (P_{SW}) is then determined by repeating this cycle multiple times and finding the percentage of cycles where the MTJ switched states. Examples of P_{SW} distribution curves with respect to V_P at various t_P values are shown in Fig. 2(c).

For STT switching, voltage required for STT to exceed the damping torque is intrinsic critical switching voltage, or V_{C0}. Thermal activated switching occurs when $V_P \leq 0.8 \times V_{C0}$ and $t_P \geq 10$ ns, where thermal fluctuations cause the MTJ to switch with a probability of $P_{\mathrm{SW}} = 1 - \exp(-t_P/\tau)$ [48]. The main purpose of measurements collected in this regime is to determine Δ and V_{C0}. Equation (5) shows a modified version of (2), which includes the influence of V_P on τ. This equation shows that V_P reduces the switching energy of the MTJ by a factor of $1 - V_P/V_{C0}$, thus increasing the probability that thermal fluctuations will cause the MTJ to switch states. The critical switching voltage (V_C) is expressed using (6) and is defined as the voltage when $\tau = t_P$. Extrapolation from the P_{SW} distribution data can be used to determine V_C by finding V_P when $P_{\mathrm{SW}} = 1 - \exp(-1)$, or $P_{\mathrm{SW}} \approx 0.63$. The values for Δ and V_{C0} can be calculated through by linearly fitting the data for V_C and $\ln(t_P/\tau_0)$, where the y-intercept and slope of the fit line are equal to V_{C0} and $-V_{C0}/\Delta$, respectively

$$\tau = \tau_0 \exp\left(\Delta\left[1 - \frac{V_P}{V_{C0}}\right]\right) \tag{5}$$

$$V_C\left(t_P\right) = V_{C0}\left(1 - \frac{1}{\Delta}\ln\left(\frac{t_P}{\tau_0}\right)\right). \qquad (6)$$

The two other switching regimes are called the precessional and dynamic reversal. Precessional switching occurs when $t_P \leq 1$ ns, where magnetization switching is dependent on initial thermal distribution rather than thermal agitation [48]. In this regime, $\tau \propto (V - V_{C0})^{-1}\ln(\pi/2\theta)$, where θ is the initial angle between the magnetization and the easy axis [48], and P_{SW} versus t_P behavior is expressed using (7) and (8). The dynamic reversal regime occurs when $t_P > 1$ ns and $t_P < 10$ ns, where the mechanism for magnetization switching is a combination of both the initial thermal distribution and thermal agitation

$$P_{\mathrm{SW}}\left(t_P\right) \propto \exp\left(\frac{H_K M_S \mathrm{Vol}}{2_{k_b T}}\left(1-\cos^2 \emptyset\right)\right)\left(V-V_{C0}\right)\sin^2 \emptyset \qquad (7)$$

$$\emptyset = \frac{\pi}{2}\exp\left(-\frac{\eta\mu_B}{eM_S t_F}\left(J - J_{C0}\right)t_P\right). \qquad (8)$$

Asynchronous measurements are typically obtained for thermally unstable MTJs with $\Delta < 20$, where thermal fluctuations drive random switching between states, as illustrated in Fig. 2(d) [54], [55], [56], [57], [58]. These devices typically show random fluctuations near the switching fields, as illustrated in Fig. 2(e). The average switching rates of these thermal fluctuations can range from 1 kHz to above 1 GHz [59], [60], [61], [62], depending on Δ. For these types of measurements, time-domain data are obtained at multiple bias voltages, examples of which are shown in Fig. 2(f). These figures show that the percentage of time that the MTJ spends in each state can be tuned with the bias voltage. Therefore, the time domain data can be analyzed in terms of the average resistance state and the average AP-state and P-state dwell times (τ_{AP} and τ_{P}, respectively). The AP- and P-state components of Δ can be determined from τ_{AP} and τ_{P} using the Néel–Arrhenius equation [recall (2) and (5)].

It should be noted that both synchronous and asynchronous switching experiments are typically used as methods of obtaining intrinsic properties of the MTJ, such as Δ and V_{C0}. However, in Section III, we will discuss recent studies, which have demonstrated that these techniques can be used as hardware-efficient approaches of generating stochastic bitstreams.

III. MTJs FOR SC

In recent years, several experimental and modeling studies have proposed MTJs as key components in RNGs or stochastic-bit generators (SBGs) for SC using the methods described in Section II-C. Some studies have even proposed that the MTJ-based circuits can not only generate the randomness required to produce stochastic bitstreams, but they can also be used to perform the necessary arithmetic functions via stochastic methods. In Sections III-A–III-E, we will review the various theories and experimental works that have been performed in recent years, where MTJs have been proposed in SC circuits.

A. TRUE RNGs THROUGH SYNCHRONOUS METHODS

General circuits for stochastic-bit generation consist of a comparator with an RNG and an analog value connected at the inputs. The output of the RNG should be centered at 0.5, meaning that the probability of a bit being 1 in the stochastic bitstream generated at the output of the comparator will be equal to the analog value at the input of the comparator. As mentioned in Section I, CMOS-based RNGs are very expensive in terms of their hardware usage and energy consumption. Furthermore, they can only generate pseudorandom numbers, meaning that computing the stochastic bitstreams generated by CMOS-based RNGs may be susceptible to errors caused by a high degree of correlation between the bitstreams. On the other hand, magnetization switching in MTJs is driven by thermal fluctuations; therefore, it is purely random. By using the synchronous method described in Section II-C and by setting V_P and t_P, so that $P_{\mathrm{SW}} = 0.5$, a single MTJ can be used as a true RNG (TRNG).

One way to ensure that $P_{\mathrm{SW}} = 0.5$ from an MTJ is through real-time output probability tracking, which is illustrated in Fig. 3(a) and (b) [63]. In this method, no reset pulse is needed. Instead, V_P at $P_{\mathrm{SW}} = 0.5$ for both AP-to-P and P-to-AP switching (V_{P+} and V_{P-}, respectively) is determined prior to TRNG implementation. The resistance state of the MTJ is continuously monitored, and the polarity of V_P changes, as the MTJ switches states. This method can be expanded even further by introducing a feedback loop that continuously adjusts t_{P-} based on the number of ones in a given segment of the bitstream, as shown in Fig. 3(c). If a string of ones is too long or too short, t_{P-} is adjusted. Similarly, a digitally controlled probability locked loop (DCPLL) can be used to fix P_{SW} at 0.5, as shown in Fig. 3(c) and (d) [64]. In this circuit, the switching probability of the MTJ is controlled with a current pulse (I_{SW}) that has a magnitude that corresponds to a 50% switching probability. The output bitstream generated is measured, and the correction logic circuit adjusts I_{SW} whenever P_{SW} deviates from 0.5.

B. STOCHASTIC BIT GENERATION THROUGH SYNCHRONOUS METHODS

By exploiting the capability of tuning P_{SW} over the entire 0–1 range, MTJs can be used as a single device, SBGs. This means that the comparator is not required to generate stochastic bitstreams, and a single MTJ can behave as an analog-to-stochastic bit converter. The most straightforward method of achieving this is to use the synchronous approach described in Section II-C, and then to control P_{SW} of the MTJ with either the perturb pulse amplitude or duration. Fig. 4(a) shows a circuit for an analog-to-digital converter (ADC) based on the switching probability of an MTJ [65]. In this particular study, the stochastic bitstreams generated from the analog voltage pulses were simply used to produce a digital output, and no stochastic computation was performed. However, it does demonstrate that the perturb pulse can be tuned to generate stochastic bitstreams with a controllable probability.

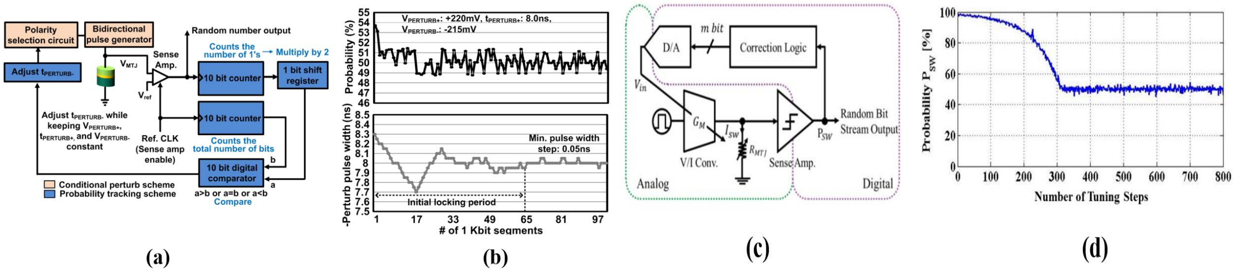
Polarity selection circuit
Bidirectional pulse generator
Adjust tPERTURB-
VMTJ
Sense Amp.
Vref
Random number output
Counts the number of 1's
Multiply by 2
10 bit counter
1 bit shift register
10 bit counter
Adjust tPERTURB- while keeping VPERTURB+, tPERTURB+, and VPERTURB- constant
Ref. CLK (Sense amp enable)
Counts the total number of bits
10 bit digital comparator
Conditional perturb scheme
Probability tracking scheme
a>b or a=b or a<b
Compare
(a)
VPERTURB+: +220mV, tPERTURB+: 8.0ns, VPERTURB-: -215mV
Probability (%)
-Perturb pulse width (ns)
Min. pulse width step: 0.05ns
Initial locking period
# of 1 Kbit segments
(b)
D/A
m bit
Correction Logic
Vin
GM
ISW
RMTJ
V/I Conv.
Sense Amp.
PSW
Random Bit Stream Output
Analog
Digital
(c)
Probability PSW [%]
Number of Tuning Steps
(d)

FIGURE 3. Examples of MTJ-based approaches for true random number generation including (a) and (b) conditional perturb and real-time output probability tracking and (c) and (d) digitally controlled probability-locked loop. (a) and (b) Extracted from [63]. (c) and (d) Extracted from [64].

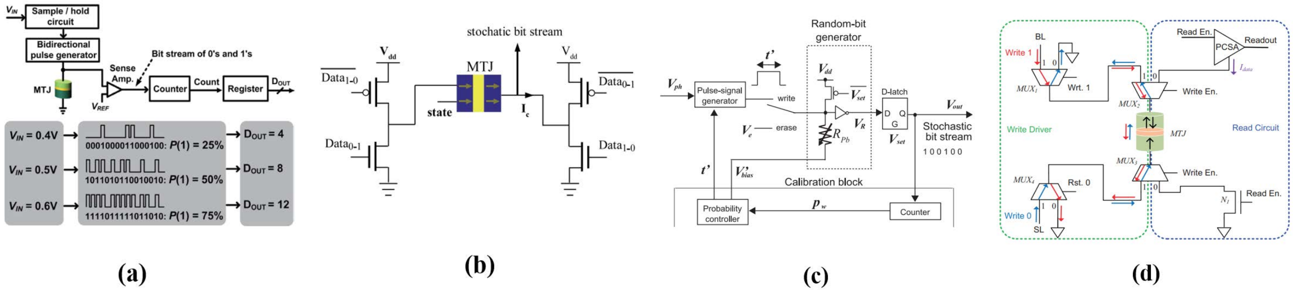

FIGURE 4. Examples of MTJ-based analog-to-stochastic bitstream conversion circuits using synchronous methods. (a) Extracted from [65]. (b) Extracted from [66]. (c) Extracted from [67]. (d) Extracted from [29].

Other examples of MTJ-based stochastic bitstream generators are shown in Fig. 4(b)–(d). Fig. 4(b) shows four-transistor, one-MTJ circuit used to generate stochastic bits by applying a current slightly below the threshold current [66]. In this circuit, no reset pulse is required; instead, the transistors are used to alternate current directions every clock cycle. The switching probability is controlled through the frequency of the clock cycle. Fig. 4(c) converts an analog signal from an image sensor to generate stochastic bitstreams with a probability determined by the amplitude of the input analog signal [67]. Unlike the approach in Fig. 4(b), this method resets the MTJ back to the original state each cycle through the erase line. Note that this approach can correct for variability in the MTJs by introducing a calibration block with a digital counter and a probability controller, which varies the perturb pulse duration to achieve the desired probability. Fig. 4(d) shows an MTJ-based, tunable SBG, where the write and reset paths are separated using multiplexers [29]. In this circuit, the write 0 path acts as the reset path by setting the MTJ back to the P-state with 100% probability every cycle. The write 1 path acts as the perturb path and sets the MTJ to the AP-state with a probability determined by the voltage amplitude.

C. SINGLE-BIASED sMTJs FOR ASYNCHRONOUS METHODS

Superparamagnetic MTJs (sMTJs) are a promising solution for generating stochastic bitstreams via asynchronous methods. They have the same behavior as the MTJs described in Fig. 2(d)–(f), except to be considered superparamagnetic; their Δ values are ideally near 0. Two options in using sMTJs for generating stochastic bits are as follows: 1) to tune the output probability with a current or voltage to behave as an SBG and 2) to generate signals with a fixed probability of 0.5, so that the sMTJ behaves as a TRNG.

The most straightforward approach for using sMTJs as a TRNG for is to tune the output signal to an average value of 0.5 using a bias voltage or current, and then reading the MTJ state in recurring time increments. This strategy has been tested experimentally and demonstrated [68], [69] that the signals generated passed all of the NIST statistical test suite randomness quality tests [70], which are used to determine if a random signal satisfies cryptographic quality requirements. Furthermore, sMTJ-based TRNGs have been demonstrated in a Bayesian inference circuit for calculating e-mail spam probability at nearly 100% accuracy rates [68].

TRNG circuits based on sMTJs consume less energy than current CMOS-based circuits by nearly one order of magnitude [68]. However, the energy consumption can be reduced even further by implementing a precharge sense amplifier (PCSA) circuit [71], as shown in Fig. 5(a). In this design, there are two PCSA circuits, one to control the clocking signal (left) and the other to control the readout (right). Fig. 5(b) illustrates how the signals generated from the sMTJ and the clocking signal correspond to the output signal.

In addition to asynchronous RNGs, sMTJs can also be used as key components for probabilistic bits (p-bits). Networks of interconnected p-bits have been proposed to perform novel computation functions, such as invertible Boolean logic [72], [73], integer factorization [74], solving a set of expansion functions [75], Bayesian inference [76], and combinatorial optimization [77]. It should be noted that the circuits in these studies are examples of Boltzmann's machines, which have different algorithms than SC-based circuits. However, the functionality of the MTJ device is very similar to those of asynchronous SC-based approaches, so they should be mentioned.

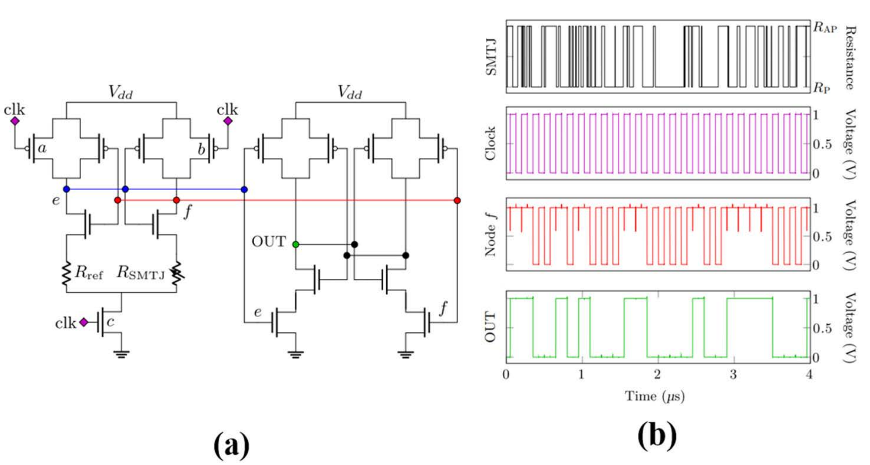

FIGURE 5. **(a) sMTJ TRNG with two PCSA circuits for clocking and readout. (b) Time-domain signals generated from the sMTJ, the clocking signal, node *f* in the circuit shown in (a), and the final output signal. Images were extracted from [71].**

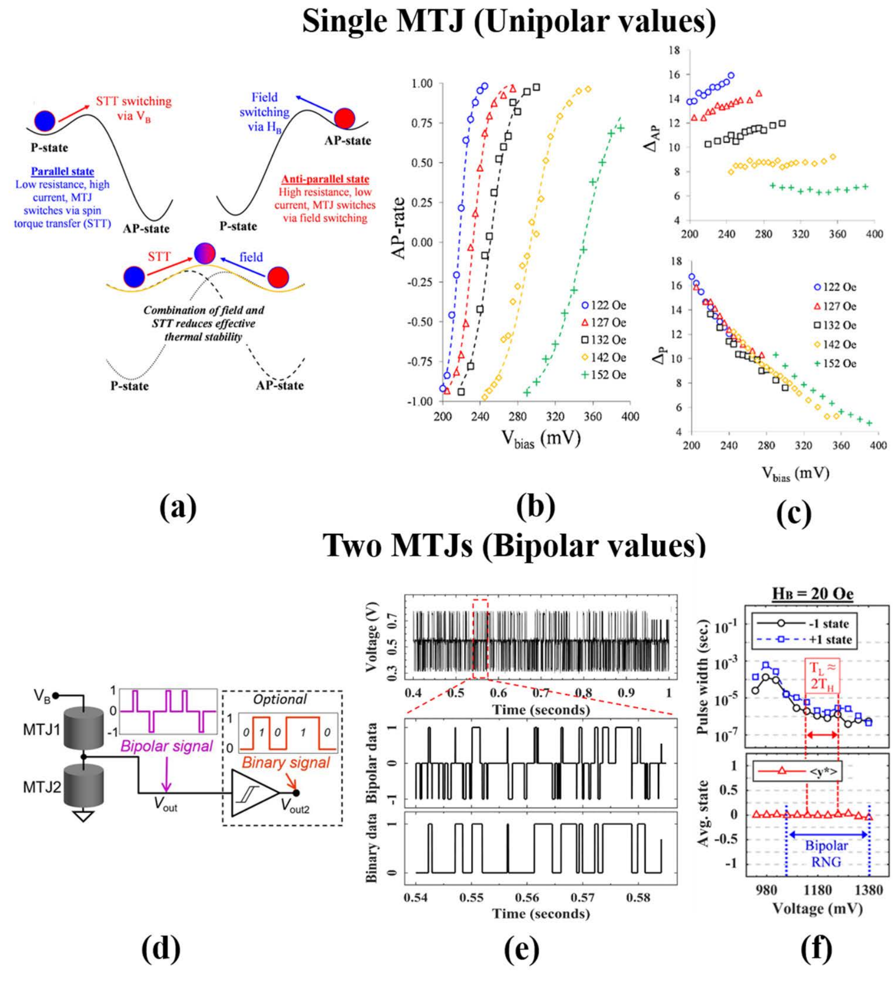

FIGURE 6. **(a) Illustration of the evolution of the energy barrier between the AP- and P-states during the dual-biasing process. (b) Transfer properties of dual-biased MTJs. (c) AP- and P-state components of the effective thermal stability factors calculated from the average AP- and P-state dwell times. (d) General schematic for dual biasing on two MTJs connected in series for bipolar random number generation. (e) Time-domain signals generated. (f) Average dwell times for +1 and −1 pulses and average bipolar value (average state) versus bias voltage for bipolar random signals. Plots shown in (b) and (c) were extracted from [81], and plots shown in (d)–(f) were extracted from [83].**

D. DUAL-BIASED MTJs FOR ASYNCHRONOUS METHODS

A major challenge for single-biased sMTJs in asynchronous SC circuits is that their switching rate and transfer curves are extremely sensitive to variations in device dimensions [78]. Previous studies have proposed one method of overcoming these effects is via a ‘‘dual-biasing’’ method where tunable

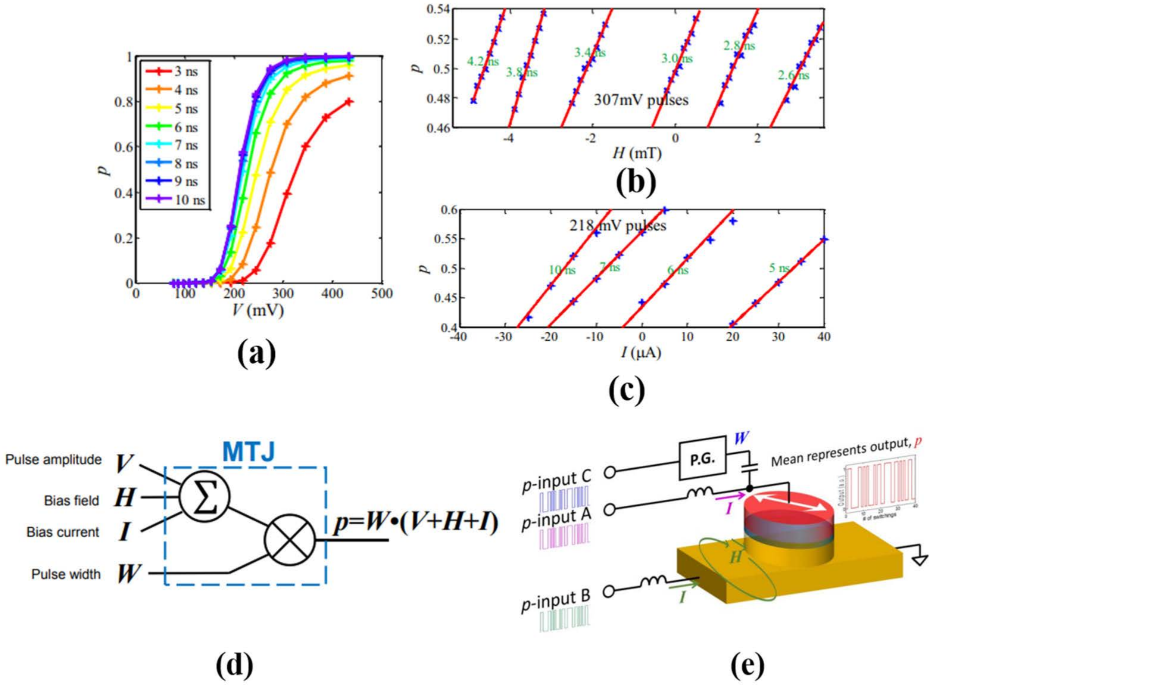

FIGURE 7. Switching probability versus (a) perturb pulse amplitude, (b) external magnetic field, and (c) pulsewidth, all showing a linear relation for switching probabilities around 0.5. (d) Diagram of the arithmetic functions performed for a single-MTJ SC unit with V, H, I, and W as inputs. (e) Example of a device level implementation with I, H, and W being determined by inputs A–C, respectively. All figures were extracted from [84].

random switching signals are generated in MTJs by utilizing the effects of two biases. These studies have demonstrated three key features of the dual-biasing method that make it a promising solution for overcoming the effects of device variations. One is that it can be used to generate stochastic switching signals in thermally stable MTJs as well as sMTJs [79]. Second is that the average switching rate of the signals generated can be tuned by over four orders of magnitude and reach switching rates above 1 MHz [79], [80], [81]. The third, and most unique, feature of dual biasing is that the two biases can control the AP- and P-state dwell times separately, which we refer to as two degrees of tunability, and implies that the average output and average switching rate can be tuned independently. The adverse impact of device variations can be eliminated through dual biasing by either using thermally stable MTJs, which are more robust to device variations, or by using sMTJs and adjusting the average switching rates in the slower MTJs.

One way to implement the dual-biasing method is via an external magnetic field (H_{bias}) and a dc voltage (V_{bias}), where H_{bias} is set to favor the P-state and V_{bias} is set to favor the AP-state, as shown in Fig. 6(a). This figure also illustrates the mechanism for generating telegraphic signals, which is the evolution of the energy barrier between the AP- and P-states. In the P-state, the current through MTJ from V_{bias} is larger than the current in the AP-state, meaning that the influence of the STT effect is more significant in the P-state. There are certain combinations of magnitudes of V_{bias} and H_{bias}, where the MTJ is never in an energetically favorable state, thus causing the MTJ to continuously toggle between the two states. As with single-biased sMTJs, the transfer properties of dual-biased MTJs are determined by the average resistance state versus V_{bias}, as shown in Fig. 6(b). The difference is that the center and the width of the transfer curve can be controlled with H_{bias} for dual-biased MTJs. Fig. 6(c) demonstrates the two degrees of tunability capability of dual biasing, where Δ_{AP} is much more dependent on H_{bias} than V_{bias}, whereas Δ_{P} is almost entirely dependent on V_{bias} and not H_{bias}. This method of dual biasing has been used in the first hardware demonstration of a p-bit-based invertible AND gate [82].

A dual-biased MTJ behaves as a single, asynchronous stochastic unit with a tunable output probability. However, an asynchronous, TRNG can also be built through dual biasing on two MTJs connected in series [83], as shown in Fig. 6(d). Fig. 6(e) shows that this configuration generates signals with three logic states, meaning that the numeric output has bipolar representation (within a $[-1, +1]$ range) rather than unipolar representation (within a $[0, +1]$ range). Note that an output probability of 50% corresponds to a bipolar value of 0.

Fig. 6(f) shows that at sufficiently large H_{bias} (20 Oe in this example), the average dwell times for $+1$ and -1 pulses (T_H and T_L, respectively) are never equal for the entire range of V_{bias} values tested. This shows that there are some slight differences in the intrinsic properties of the two MTJs. Despite these differences, Fig. 6(f) demonstrates that the average bipolar output, $\langle y^* \rangle$, becomes fixed at zero over the nearly entire V_{bias} range. Furthermore, nearly all the datasets passed the standardized National Institute of Standards and Technology Statistical Test Suite (NIST STS [70]). These sets are labeled as "Bipolar RNG" in Fig. 6(f). The reason that $\langle y^* \rangle$ becomes fixed at zero over a large range of V_{bias} values and sufficiently large H_{bias} is due to the two degrees

TABLE 1. **Overview of all designs, methods, and applications for MTJ-based SC TRNGs or SBGs discussed in this article.**

SBG method	MTJ type	Experiment/ Simulation	Application	Tunable or fixed	Reference	Year Published
Synchronous	Thermally stable	Experiment	NA	STT Tunable	[48 – 52]	2007 - 2012
Asynchronous	Low – Δ	Experiment	NA	STT Tunable	[54-55, 57-58, 62]	2016 – 2022
Asynchronous	Low – Δ	Experiment	NA	SOT Tunable	[56]	2019
Asynchronous	High – Δ (Dual-biasing)	Experiment	NA	STT + Field Tunable	[59, 61, 81]	2004 – 2019
Asynchronous	Low – Δ (Dual-biasing)	Experiment	NA	STT + Field Tunable	[60]	2011
Asynchronous	Low or High – Δ (Dual-biasing)	Experiment	NA	STT + Field Tunable	[79]	2021
Synchronous	Thermally stable	Experiment	True random number generator	STT Fixed	[53, 63]	2015
Synchronous	Thermally stable	Simulation	True random number generator	STT Fixed	[64]	2015
Synchronous	Thermally stable	Experiment	Tunable random number generator	STT Tunable	[65]	2015
Synchronous	Thermally stable	Simulation	Tunable random number generator	STT Tunable	[66-67]	2015 – 2016
Asynchronous	Low – Δ	Experiment	True random number generator	STT Tunable	[68]	2017
Asynchronous	Low – Δ	Experiment	True random number generator	STT Fixed	[69]	2017
Asynchronous	Low – Δ	Experiment	SBG for asynchronous SC	STT Tunable	[71]	2020
Asynchronous	Thermally stable (Dual-biasing)	Experiment	Neural spiking unit with high information capacity	STT + Field Tunable	[80]	2018
Asynchronous	Thermally stable (Dual-biasing	Experiment	Tunable bipolar random number generator	STT + Field Tunable	[83]	2022
Synchronous	Thermally stable	Experiment	Single SC unit	STT + Field + pulse width tunable	[84]	2017
Asynchronous	Low – Δ	Simulation	Invertible Boolean Logic	STT + SOT Tunable/Fixed	[72 – 73]	2017 – 2019
Asynchronous	Thermally stable (DB)	Experiment	Invertible Boolean Logic	STT + Field Tunable/Fixed	[82]	2019
Asynchronous	Low – Δ	Simulation	Integer factorization	STT Tunable	[74]	2019
Asynchronous	Low – Δ	Simulation	Solve set of expansion functions	STT Tunable	[75]	2018
Synchronous	Low – Δ	Simulation	Bayesian Inference	STT Tunable	[29]	2018
Asynchronous	Low – Δ	Simulation	Bayesian Inference	STT Tunable	[76]	2018
Asynchronous	Low – Δ	Simulation	Combinatorial Optimization	STT or SOT Tunable	[77]	2017

of tunability feature of dual biasing [83].

Bipolar encoding in SC enables efficient processing of a broader range of arithmetic functions, such as scaled subtraction and hyperbolic tangent function [15]. Typically, bipolar algorithms are still encoded using binary bits, where a binary bit ‘‘0’’ represents a bipolar bit ‘‘-1.’’ This means that bipolar encoding reduces the numeric resolution of the stochastic bitstreams by one half [15]. The resolution can be improved by doubling the size of the bitstreams; however, this would unfortunately double the computation delay. The method described in Fig. 6(f) and (g) can eliminate the need for this trade-off between numeric resolution and computation delay, since bipolar numbers are represented with three bipolar bits rather than with binary bits. Furthermore, the two degrees of tunability feature of dual biasing make this method robust against device variations and ensure a high quality, random signal under the proper biasing conditions.

E. MTJ SC UNIT

In Sections III-A–III-D, we illustrated how MTJs can be used for TRNGs or SBGs for SC. In this section, we will describe MTJ-based circuits can also perform SC functions [84], [85]. First, we will describe an approach based on the synchronous method described in Section II-C, where a single MTJ can be used to perform stochastic multiplication and addition. In this approach, the perturb pulsewidth (W), perturb pulse amplitude (V), an external magnetic field (H), and a dc bias current (I) are considered to be inputs, and P_{SW} is the output.

Fig. 7(a)–(c) shows that P_{SW} is linear with V, I, and H, respectively, when P_{SW} is around 0.5. Furthermore, these results showed that H at $P_{\mathrm{SW}} = 0.5$ and I at $P_{\mathrm{SW}} = 0.5$ were linear. Note that V and W were kept constant. A similar analysis was done for V and I, where V at $P_{\mathrm{SW}} = 0.5$ was proportional to I at $P_{\mathrm{SW}} = 0.5$, while H and W were kept constant. This implies that for switching probabilities around 0.5, P_{SW} is proportional to $V + H + I$. These figures also show that W affects the slope of P_{SW} with V, H, and I, implying that P_{SW} is proportional to $W \times (V + H + I)$, in the form of a mean value, digital sequence, as illustrated in Fig. 7(d).

Fig. 7(e) shows an example of a device level implementation of this approach with input bitstreams A–C and output bitstream P. Both bitstreams A and B are low-pass filtered; however, A is fed directly to the MTJ in the form of a dc bias current, whereas B is fed to the bottom bus line in the form of an external bias field. Bitstream C is fed to a pulse generator and controls either the pulse amplitude or pulsewidth applied to the MTJ for each perturb cycle. The output bitstream would have a probability that is proportional to the summation of A–C multiplied by a fixed value determined by the pulsewidth or it is proportional $C \times (A + B)$ with an additional fixed term included in the sum determined by the pulse amplitude.

Alternatively, stochastic computation can be performed within the computational random access memory (CRAM) array [85]. In this approach, simple Boolean logic is performed in the CRAM array to perform arithmetic functions on stochastic bitstreams [recall Fig. 1(a) and (b)]. Furthermore, this approach allows for stochastic bitstreams to be generated within the CRAM array by adding a perturb step where the input MTJs switch probabilistically prior to each logic step.

IV. SUMMARY

In this article, we reviewed the literature published in recent years where practical methods for random number generation and stochastic-bit generation via MTJ-based hardware were examined. Table 1 provides an overview of all the designs, methods, and applications that were discussed in this article. We discussed how true randomness can be achieved in MTJs via synchronous or asynchronous methods. In either method, MTJ-based circuits can be used to generate true random numbers where mean value is centered around 0.5 or 0 for random numbers with bipolar representation, or to generate stochastic bitstreams with a tunable output. While CMOS-based circuits for random number generation are very expensive in terms of hardware usage and energy consumption, MTJ-based circuits provide a low-cost, energy-efficient solution for future hardware implementation in SC algorithms.

REFERENCES

[1] C. Li, X. Zhang, J. Li, T. Fang, and X. Dong, "The challenges of modern computing and new opportunities for optics," *PhotoniX*, vol. 2, no. 1, pp. 2–4, Sep. 2021, doi: 10.1186/s43074-021-00042-0.

[2] A. McAfee, E. Brynjolfsson, T. H. Davenport, D. J. Patil, and D. Barton, "Big data needs a hardware revolution," *Nature*, vol. 554, pp. 145–146, doi: 10.1038/d41586-018-01683-1.

[3] S. Furber, "Large-scale neuromorphic computing systems," *J. Neural Eng.*, vol. 13, no. 5, 2016, Art. no. 051001, doi: 10.1088/1741-2560/13/5/051001.

[4] G. Indiveri and S.-C. Liu, "Memory and information processing in neuromorphic systems," *Proc. IEEE*, vol. 103, no. 8, pp. 1379–1397, Aug. 2015, doi: 10.1109/JPROC.2015.2444094.

[5] B. R. Gaines, "Stochastic computing," in *Proc. Comput. Conf. (Spring)*, 1967, pp. 149–156, doi: 10.1145/1465482.1465505.

[6] W. J. Poppelbaum, C. Afuso, and J. W. Esch, "Stochastic computing elements and systems," in *Proc. Comput. Conf. AFIPS (Fall)*, 1967, pp. 635–644, doi: 10.1145/1465611.1465696.

[7] B. R. Gaines, "Stochastic computing systems," *Adv. Inf. Syst.*, vol. 2, pp. 37–172. Mar. 1969, doi: 10.1007/978-1-4899-5841-9_2.

[8] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Trans. Embedded Comput. Syst.*, vol. 12, no. 2s, pp. 1–19, May 2013, doi: 10.1145/2465787.2465794.

[9] A. Alaghi, W. Qian, and J. P. Hayes, "The promise and challenge of stochastic computing," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 37, no. 8, pp. 1515–1531, Aug. 2018, doi: 10.1109/TCAD.2017.2778107.

[10] W. Qian, X. Li, M. D. Riedel, K. Bazargan, and D. J. Lilja, "An architecture for fault-tolerant computation with stochastic logic," *IEEE Trans. Comput.*, vol. 60, no. 1, pp. 93–105, Jan. 2011, doi: 10.1109.TC.2010.202.

[11] P. Li, D. J. Lilja, W. Qian, K. Bazargan, and M. D. Riedel, "Computation on stochastic bit streams digital image processing case studies," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 3, pp. 449–462, Mar. 2014, doi: 10.1109/TVLSI.2013.2247429.

[12] M. H. Najafi and M. E. Salehi, "A fast fault-tolerant architecture for Sauvola local image thresholding algorithm using stochastic computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 2, pp. 808–812, Feb. 2016, doi: 10.1109/TVLSI.2015.2415932.

[13] P. Li and D. J. Lilja, "A low power fault-tolerant architecture for the kernel density estimation based image segmentation algorithm," in *Proc. IEEE Int. Conf. Appl.-Specific Syst. Archit. Process*, Sep. 2011, pp. 161–168, doi: 10.1109/ASAP.2011.6043264.

[14] J. M. de Aguiar and S. P. Khatri, "Exploring the viability of stochastic computing," in *Proc. 33rd IEEE Int. Conf. Comput. Design (ICCD)*, Oct. 2015, pp. 391–394, doi: 10.1109/ICCD.2015.7357131.

[15] P. Li, W. Qian, D. J. Lilja, K. Bazargan, and M. D. Riedel, "Case studies of logical computation on stochastic bit streams," in *Integrated Circuit and System Design. Power and Timing Modeling, Optimization and Simulation* (Lecture Notes in Computer Science), vol. 7606. Heidelberg, Germany: Springer, 2013, pp. 235–244, doi: 10.1007/978-3-642-36157-9_24.

[16] Y. Liu and K. K. Parhi, "Computing hyperbolic tangent and sigmoid functions using stochastic logic," in *Proc. 50th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2016, pp. 1580–1585, doi: 10.1109/ACSSC.2016.7869645.

[17] A. Alaghi, C. Li, and J. P. Hayes, "Stochastic circuits for real-time image-processing applications," in *Proc. 50th Annu. Design Autom. Conf.*, 2013, pp. 1–6, doi: 10.1145/2463209.2488901.

[18] B. D. Brown and H. C. Card, "Stochastic neural computation I: Computational elements," *IEEE Trans. Comput.*, vol. 50, no. 9, pp. 891–905, Sep. 2001, doi: 10.1109/12.954505.

[19] B. D. Brown and H. C. Card, "Stochastic neural computation II: Soft competitive learning," *IEEE Trans. Comput.*, vol. 50, no. 9, pp. 920–960, Sep. 2001, doi: 10.1109/12.954596.

[20] Y. Ji, F. Ran, C. Ma, and D. J. Lilja, "A hardware implementation of a radial basis function neural network using stochastic logic," in *Proc. Design, Automat. Test Eur. Conf. Exhib. (DATE)*, Mar. 2015, pp. 880–883, doi: 10.5555/2755753.2757016.

[21] V. Canals, A. Morro, A. Oliver, M. L. Alomar, and J. L. Rosselè, "A new stochastic computing methodology for efficient neural network implementation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 3, pp. 551–564, Mar. 2016, doi: 10.1109/TNNLS.2015.2413754.

[22] J. Li et al., "Towards acceleration of deep convolutional neural networks using stochastic computing," in *Proc. 22nd Asia South Pacific Design Automat. Conf. (ASP-DAC)*, Feb. 2017, pp. 115–120, doi: 10.1109/ASP-DAC.2017.785306.

[23] A. Ren et al., "SC-DCNN: Highly-scalable deep convolutional neural network using stochastic computing," *ACM SIGOPS Operating Syst. Rev.*, vol. 51, no. 2, pp. 405–418, Jul. 2017, doi: 10.1145/3093336.3037746.

[24] S. Bodiwala and N. Nanavati, "An efficient stochastic computing based deep neural network accelerator with optimized activation functions," *Int. J. Inf. Technol.*, vol. 13, no. 3, pp. 1179–1192, Jun. 2021, doi: 10.1007/s41870-021-00682-2.

[25] S. S. Tehrani, S. Mannor, and W. J. Gross, "Fully parallel stochastic LDPC decoders," *IEEE Trans. Signal Process.*, vol. 56, no. 11, pp. 5692–5703, Nov. 2008, doi: 10.1109/TSP.2008.929671.

[26] S. S. Tehrani, A. Naderi, G.-A. Kamendje, S. Hemati, S. Mannor, and W. J. Gross, "Majority-based tracking forecast memories for stochastic LDPC decoding," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4883–4896, Sep. 2010, doi: 10.1109/TSP.2010.2051434.

[27] X. R. Lee, C. L. Chen, H. C. Chang, and C. Y. Lee, "A 7.92 Gb/s 437.2 mW stochastic LDPC decoder chip for IEEE 802.15.3c applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 2, pp. 507–516, Feb. 2015, doi: 10.1109/TCSI.2014.2360331.

[28] A. Coninx et al., "Bayesian sensor fusion with fast and low power stochastic circuits," in *Proc. IEEE Int. Conf. Rebooting Comput. (ICRC)*, Nov. 2016, pp. 1–8, doi: 10.1109/ICRC.2016.7738672.

[29] X. Jia, J. Yang, Z. Wang, Y. Chen, H. H. Li, and W. Zhao, "Spintronics based stochastic computing for efficient Bayesian inference system," in *Proc. 23rd Asia South Pacific Design Autom. Conf. (ASP-DAC)*, Jan. 2018, pp. 580–585, doi: 10.1109/ASPDAC.2018.8297385.

[30] D. Zhang, H. Li, and S. Y. Foo, "A simplified FPGA implementation of neural network algorithms integrated with stochastic theory for power electronics applications," in *Proc. 31st Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Jan. 2005, pp. 1018–1023, doi: 10.1109/IECON.2005.1569044.

[31] H. Hsiao, J. Anderson, and Y. Hara-Azumi, "Generating stochastic bitstreams," in *Stochastic Computing: Techniques and Applications*. Cham, Switzerland: Springer, 2019, pp. 137–152, doi: 10.1007/978-3-030-03730-7_7.

[32] D. Apalkov, B. Dieny, and J. M. Slaughter, "Magnetoresistive random access memory," *Proc. IEEE*, vol. 104, no. 10, pp. 1796–1830, Oct. 2016, doi: 10.1109/JPROC.2016.2590142.

[33] J.-P. Wang et al., "A pathway to enable exponential scaling for the beyond-CMOS era: Invited," in *Proc. Des. Autom. Conf. (DAC)*, vol. 16, Jun. 2017, pp. 1–6, doi: 10.1145/3061639.3072942.

[34] B. Dieny et al., "Opportunities and challenges for spintronics in the microelectronics industry," *Nat. Electron.*, vol. 3, pp. 446–459, Aug. 2020, doi: 10.1038/s41928-020-0461-5.

[35] S. Ikegawa, F. B. Mancoff, J. Janesky, and S. Aggarwal, "Magnetoresistive random access memory: Present and future," *IEEE Trans. Electron Devices*, vol. 67, no. 4, pp. 1407–1419, Apr. 2020, doi: 10.1109/TED.2020.2965403.

[36] A. Hirohata et al., "Review on spintronics: Principles and device applications," *J. Magn. Magn. Mater.*, vol. 509, Apr. 2020, Art. no. 166711, doi: 10.1016/j.jmmm.2020.166711.

[37] Y.-C. Liao, C. Pan, and A. Naeemi, "Benchmarking and optimization of spintronic memory arrays," *IEEE J. Explor. Solid-State Comput. Devices Circuits*, vol. 6, no. 1, pp. 9–17, Jun. 2020, doi: 10.1109/JXCDC.2020.2999270.

[38] P. Barla, V. K. Joshi, and S. Bhat, "Spintronic devices: A promising alternative to CMOS devices," *J. Comput. Electron.*, vol. 20, no. 2, pp. 805–837, Apr. 2021, doi: 10.1007/s10825-020-01648-6.

[39] L. Lopez-Diaz, L. Torres, and E. Moro, "Transition from ferromagnetism to superparamagnetism on the nanosecond time scale," *Phys. Rev. B, Condens. Matter*, vol. 65, May 2002, Art. no. 224406, doi: 10.1103/PhysRevB.65.224406.

[40] J.-G. Zhu, "Magnetoresistive random access memory: The path to competitiveness and scalability," *Proc. IEEE*, vol. 96, no. 11, pp. 1786–1798, Nov. 2008, doi: 10.1109/JPROC.2008.2004313.

[41] S. Bhatti, R. Sbiaa, A. Hirohata, H. Ohno, S. Fukami, and S. N. Piramanayagam, "Spintronics based random access memory: A review," *Mater. Today*, vol. 20, no. 9, pp. 530–548, Nov. 2017, doi: 10.1016/j.mattod.2017.07.007.

[42] J. C. Slonczewski, "Current-driven excitation of magnetic multilayers," *J. Magn. Magn. Matls.*, vol. 159, pp. L1–L7, Jun. 1996, doi: 10.1016/0304-8853(96)00062-5.

[43] W. H. Butler, X. G. Zhang, T. C. Schulthess, and J. M. Maclaren, "Spin-dependent tunneling conductance of Fe|MgO|Fe sandwiches," *Phys. Rev. B, Condens. Matter*, vol. 63, Jan. 2001, Art. no. 054416, doi: 10.1103/PhysRevB.63.054416.

[44] A. Brataas, A. D. Kent, and H. Ohno, "Current-induced torques in magnetic materials," *Nat. Mater.*, vol. 11, pp. 372–381, May 2012, doi: 10.1038/nmat311.

[45] S. Amara, H. Bea, R. C. Sousa, and B. Dieny, "Barrier breakdown mechanisms in MgO-based magnetic tunnel junctions under pulsed conditions," in *Proc. 4th IEEE Int. Memory Workshop*, Jun. 2012, pp. 1–4, doi: 10.1109/IMW.2012.6213653.

[46] X. Han, X. Wang, C. Wan, G. Yu, and X. Lv, "Spin-orbit torques: Materials, physics, and devices," *Appl. Phys. Lett.*, vol. 118, no. 12, Mar. 2021, Art. no. 120502, doi: 10.1063/5.0039147.

[47] Q. Shao et al., "Roadmap of spin-orbit torques," *IEEE Trans. Magn.*, vol. 57, no. 7, pp. 1–39, May 2021, doi: 10.1109/TMAG.2021.3078583.

[48] Z. Diao et al., "Spin-transfer torque switching in magnetic tunnel junctions and spin-transfer torque random access memory," *J. Phys., Condens. Matter.*, vol. 19, no. 16, Apr. 2007, Art. no. 165209, doi: 10.1088/0953-8984/19/16/165209.

[49] T. Aoki, Y. Ando, M. Oogane, and H. Naganuma, "Reproducible trajectory on subnanosecond spin-torque magnetization switching under a zero-bias field for MgO-based ferromagnetic tunnel junctions," *Appl. Phys. Lett.*, vol. 96, Apr. 2010, Art. no. 142502, doi: 10.1063/1.3380595.

[50] H. Zhao et al., "Low writing energy and sub nanosecond spin torque transfer switching of in-plane magnetic tunnel junction for spin torque transfer random access memory," *J. Appl. Phys.*, vol. 109, Mar. 2011, Art. no. 07C720, doi: 10.1063/1.3556784.

[51] R. Heindl, W. H. Rippard, S. E. Russek, M. R. Pufall, and A. B. Kos, "Validity of the thermal activation model for spin-transfer torque switching in magnetic tunnel junctions," *J. Appl. Phys.*, vol. 109, Apr. 2011, Art. no. 073910, doi: 10.1063/1.3562136.

[52] H. Zhao et al., "Spin-torque driven switching probability density function asymmetry," *IEEE Trans. Magn.*, vol. 48, no. 11, pp. 3818–3820, Nov. 2012, doi: 10.1109/TMAG.2012.2197815.

[53] A. Fukushima et al., "Spin dice: A scalable truly random number generator based on spintronics," *Appl. Phys. Exp.*, vol. 7, Jul. 2014, Art. no. 083001, doi: 10.7567/APEX.7.083001.

[54] M. Bapna, S. K. Piotrowski, S. D. Oberdick, M. Li, C.-L. Chien, and S. A. Majetich, "Magnetostatic effects on switching in small magnetic tunnel junctions," *Appl. Phys. Lett.*, vol. 108, no. 2, Jan. 2016, Art. no. 022406, doi: 10.1063/1.4939911.

[55] M. Bapna and S. A. Majetich, "Current control of time-averaged magnetization in superparamagnetic tunnel junctions," *Appl. Phys. Lett.*, vol. 111, no. 24, Dec. 2017, Art. no. 243107, doi: 10.1063/1.5012091.

[56] V. Ostwal and J. Appenzeller, "Spin-orbit torque-controlled magnetic tunnel junction with low thermal stability for tunable random number generation," *IEEE Magn. Lett.*, vol. 10, 2019, Art. no. 4503305, doi: 10.1109/LMAG.2019.2912971.

[57] B. Parks, A. Abdelgawad, T. Wong, R. F. L. Evans, and S. A. Majetich, "Magnetoresistance dynamics in superparamagnetic Co-Fe-B nanodots," *Phys. Rev. A, Gen. Phys.*, vol. 13, Jan. 2020, Art. no. 014063, doi: 10.1103/PhysRevApplied.13.014063.

[58] T. Funatsu, S. Kanai, J. Ieda, S. Fukami, and H. Ohno, "Local Bifurcation with spin-transfer torque in superparamagnetic tunnel junctions," *Nature Commun.*, vol. 13, Jul. 2022, Art. no. 4079, doi: 10.1038/s41467-022-31788-1.

[59] M. R. Pufall, W. H. Rippard, S. Kaka, S. E. Russek, and T. J. Silva, "Large-angle, gigahertz-rate random telegraph switching induced by spin-momentum transfer," *Phys. Rev. B, Condens. Matter*, vol. 69, Jun. 2004, Art. no. 214409, doi: 10.1103/PhysRevB.69.214409.

[60] W. Rippard, R. Heindl, M. Pufall, S. Russek, and A. Kos, "Thermal relaxation rates of magnetic nanoparticles in the presence of magnetic fields and spin-transfer effects," *Phys. Rev. B, Condens. Matter*, vol. 84, Aug. 2011, Art. no. 064439, doi: 10.1103/PhysRevB.84.064439.

[61] S. K. Piotrowski, M. Bapna, S. D. Oberdick, and S. A. Majetich, "Size and voltage dependence of effective anisotropy in sub-100-nm perpendicular magnetic tunnel junctions," *Phys. Rev. B, Condens. Matter*, vol. 94, Jul. 2016, Art. no. 014404, doi: 10.1103/PhysRevB.94.014404.

[62] K. Hayakawa et al., "Nanosecond random telegraph noise in in-plane magnetic tunnel junctions," *Phys. Rev. Lett.*, vol. 126, Mar. 2021, Art. no. 117202, doi: 10.1103/PhysRevLett.126.117202.

[63] W. H. Choi et al., "A magnetic tunnel junction based true random number generator with conditional perturb and real-time output probability tracking," in *IEDM Tech. Dig.*, Feb. 2015, pp. 1–4, doi: 10.1109/IEDM.2014.7047039.

[64] S. Oosawa, T. Konishi, N. Onizawa, and T. Hanyu, "Design of an STT-MTJ based true random number generator using digitally controlled probability-locked loop," in *Proc. IEEE 13th Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2015, pp. 3–6, doi: 10.1109/NEWCAS.2015.7182089.

[65] W. H. Choi, Y. Lv, H. Kim, J.-P. Wang, and C. H. Kim, "An 8-bit analog-to-digital converter based on the voltage-dependent switching probability of a magnetic tunnel junction," in *Proc. Symp. VLSI Technol. (VLSI Technol.)*, Jun. 2015, pp. T162–T163, doi: 10.1109/VLSIT.2015.7223662.

[66] L. A. de Barros Naviner, H. Cai, Y. Wang, W. Zhao, and A. Ben Dhia, "Stochastic computation with spin torque transfer magnetic tunnel junction," in *Proc. IEEE 13th Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2015, pp. 1–4, doi: 10.1109/NEWCAS.2015.7182031.

[67] N. Onizawa, D. Katagiri, W. J. Gross, and T. Hanyu, "Analog-to-stochastic converter using magnetic tunnel junction devices for vision chips," *IEEE Trans. Nanotechnol.*, vol. 15, no. 5, pp. 705–714, Sep. 2016, doi: 10.1109/TNANO.2015.2511151.

[68] D. Vodenicarevic et al., "Low-energy truly random number generation with superparamagnetic tunnel junctions for unconventional computing," *Phys. Rev. Appl.*, vol. 8, Nov. 2017, Art. no. 054045, doi: 10.1103/PhysRevApplied.8.054045.

[69] B. Parks, M. Bapna, J. Igbokwe, H. Almasi, W. Wang, and S. A. Majetich, "Superparamagnetic perpendicular magnetic tunnel junctions for true random number generators," *AIP Adv.*, vol. 8, no. 5, May 2018, Art. no. 055903, doi: 10.1063/1.5006422.

[70] L. E. Bassham et al., "A statistical test suite for random and pseudo-random number generators for cryptographic applications," NIST Special Publication, NIST, Gaithersburg, MD, USA, Tech. Rep. 800-22 Rev 1a, 2010. [Online]. Available: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=906762

[71] M. W. Daniels, A. Madhavan, P. Talatchian, A. Mizrahi, and M. D. Stiles, "Energy-efficient stochastic computing with superparamagnetic tunnel junctions," *Phys. Rev. A, Gen. Phys.*, vol. 13, no. 3, Mar. 2020, Art. no. 034016, doi: 10.1103/PhysRevApplied.13.034016.

[72] K. Y. Camsari, R. Faria, B. M. Sutton, and S. Datta, "Stochastic p-bits for invertible logic," *Phys. Rev. X*, vol. 7, no. 3, p. 31014, Jul. 2017, doi: 10.1103/PhysRevX.7.031014.

[73] K. Y. Camsari, B. M. Sutton, and S. Datta, "P-bits for probabilistic spin logic," *Appl. Phys. Rev.*, vol. 6, no. 1, Mar. 2019, Art. no. 011305, doi: 10.1063/1.5055860.

[74] W. A. Borders, A. Z. Pervaiz, S. Fukami, K. Y. Camsari, H. Ohno, and S. Datta, "Integer factorization using stochastic magnetic tunnel junctions," *Nature*, vol. 573, no. 7774, pp. 390–393, Sep. 2019, doi: 10.1038/s41586-019-1557-9.

[75] A. Mizrahi et al., "Neural-like computing with populations of superparamagnetic basis functions," *Nature Commun.*, vol. 9, p. 1533, Apr. 2018, doi: 10.1038/s41467-018-03963-w.

[76] R. Faria, K. Y. Camsari, and S. Datta, "Implementing Bayesian networks with embedded stochastic MRAM," *AIP Adv.*, vol. 8, no. 4, Apr. 2018, Art. no. 045101, doi: 10.1063/1.5021332.

[77] B. Sutton, K. Y. Camsari, B. Behin-Aein, and S. Datta, "Intrinsic optimization using stochastic nanomagnets," *Sci. Rep.*, vol. 7, p. 44370, Mar. 2017, doi: 10.1038/srep.44370.

[78] M. A. Abeed and S. Bandyopadhyay, "Sensitivity of the power spectra of thermal magnetization fluctuations in low barrier nanomagnets proposed for stochastic computing to in-plane barrier height variations and structural defects," *Spin*, vol. 10, no. 1, Nov. 2019, Art. no. 2050001, doi: 10.1142/S2010324720500010.

[79] B. R. Zink and J.-P. Wang, "Influence of intrinsic thermal stability of switching rate and tunability of dual-biased magnetic tunnel junctions for probabilistic bits," *IEEE Magn. Lett.*, vol. 12, pp. 1–5, 2021, doi: 10.1109/LMAG.2021.3084901.

[80] B. R. Zink, Y. Lv, and J.-P. Wang, "Telegraphic switching signals by magnet tunnel junctions for neural spiking signals with high information capacity," *J. Appl. Phys.*, vol. 124, no. 15, Oct. 2018, Art. no. 152121, doi: 10.1063/1.5042444.

[81] B. R. Zink, Y. Lv, and J.-P. Wang, "Independent control Antiparallel- and parallel-state thermal stability factors in magnetic tunnel junctions for telegraphic signals with two degrees of tunability," *IEEE Trans. Electron Devices*, vol. 66, no. 12, pp. 1582–1587, Dec. 2019, doi: 10.1109.TED.2019.2948218.

[82] Y. Lv, R. Bloom, and J.-P. Wang, "Experimental demonstration of probabilistic spin logic by magnetic tunnel junctions," *IEEE Magn. Lett.*, vol. 10, 2019, Art. no. 4510905, doi: 10.1109/LMAG.2019.2957258.

[83] Y. Lv, B. R. Zink, and J.-P. Wang, "Bipolar random spike and bipolar random number generation by two magnetic tunnel junctions," *IEEE Trans. Electron Devices*, vol. 69, no. 3, pp. 1582–1587, Mar. 2022, doi: 10.1109/TED.2022.3144117.

[84] Y. Lv and J.-P. Wang, "A single magnetic-tunnel-junction stochastic computing unit," in *IEDM Tech. Dig.*, Dec. 2017, p. 36, doi: 10.1109/IEDM.2017.8268504.

[85] B. R. Zink et al., "A stochastic computing scheme of embedding random bit generation and processing in computational random access memory (SC-CRAM)," *IEEE J. Explor. Solid-State Comput. Devices Circuits*, submitted for publication.