

Robust and Programmable Logic-In-Memory Devices Exploiting Skyrmion Confinement and Channeling Using Local Energy Barriers

Naveen Sisodia^{✉,*}, Johan Pelloux-Prayer[✉], Liliana D. Buda-Prejbeanu, Lorena Anghel, Gilles Gaudin, and Olivier Boulle^{✉,†}

Univ. Grenoble Alpes, CNRS, CEA, SPINTEC, Grenoble F-38000, France



(Received 18 March 2022; revised 4 May 2022; accepted 12 May 2022; published 12 July 2022)

Magnetic skyrmions are promising candidates for logic-in-memory applications, intrinsically merging high-density nonvolatile data storage with computing capabilities, owing to their nanoscale size, fast motion, and mutual repulsions. However, concepts proposed so far suffer from reliability issues as well as inefficient conversion of magnetic information to electrical signals. In this paper, we propose a logic-in-memory device, which exploits skyrmion confinement and channeling using anisotropy energy barriers to achieve reliable data storage and synchronous shift in racetracks combined with cascadable and reprogrammable logics relying purely on magnetic interactions. The device combines a racetrack shift register based on skyrmions confined in nanodots with full-adder (FA) gates. The designed FA is reprogrammable and cascadable and can also be used to perform simple logic operations such as AND, OR, NOT, NAND, XOR, and NXOR. The monolithic design of the logic gate and the absence of any complex electrical contacts makes the device ideal for integration with conventional CMOS circuitry.

DOI: [10.1103/PhysRevApplied.18.014025](https://doi.org/10.1103/PhysRevApplied.18.014025)

I. INTRODUCTION

“Logic-in-memory” architectures have recently emerged as an alternative to von Neumann architectures, where the constant shuttle of data between logic and memory units leads to a critical rise in energy consumption and delay when downscaling. This has led to the search for technologies that combine memory and logic functionalities. Recently, magnetic skyrmions have been proposed as the building block of such logic-in-memory technologies. Magnetic skyrmions are local chiral whirling of the magnetization. Their small lateral dimensions, down to the nanometer size and topological stability grant them particlelike properties, which, combined with the possibility to be manipulated via electrical currents, can be exploited to code data and achieve computation at the nanoscale. These textures appear promising for logic-in-memory applications since they intrinsically merge high-density nonvolatile storage and logic capabilities. However, while a number of memory and logic concepts have been proposed based on skyrmions [1–12], several major issues have hindered their technological advancement. In a skyrmion racetrack [13], data is encoded through the distance between neighboring skyrmions and the memory operation relies on the synchronous shift of skyrmion trains, where the skyrmion interdistance remains constant. However, the latter can be easily perturbed by thermal

activation or small variation in the skyrmion velocities, induced for instance by local changes in the magnetic properties in the material or process variations. Different pathways have been proposed to solve this issue such as double lanes [14] or local gate control [15]. However, these solutions appear difficult to implement. For instance, local gate control requires numerous gate contacts leading to increased periphery area, limited density, and increasing complexity. Regarding skyrmion-based logic, the fundamental building blocks need to fulfill a set of criteria, including cascading, fan out, logic level restoration, immunity to noise, mitigation of potential loss of information, and input-to-output isolation. However, concepts proposed so far required complex operations and faced serious limitations. These include skyrmion-charge interconversion, which limits the advantage of a full skyrmionic signal [6,7,10,11], large skyrmion circuits requiring complex synchronization of data [4], multiple gate voltages with complex implementation [3,5], etc. In addition, several proposals do not fulfill the aforementioned basic requirements of logic, namely cascading, input and clock synchronization, etc. . . . Thus, there is currently no concept for a full skyrmionic computer, incorporating logic, memory, and interconnects using exclusively magnetic signals without the need for intermediate conversion to charge signals.

Besides the aforementioned limitations, moving skyrmions are subject to the skyrmion Hall effect (SKHE), namely a motion transverse to the driving force. To address this problem, topological spin textures with vanishing topological charges and thus SKHE have been proposed

*naveensisodia2@gmail.com

†olivier.boulle@cea.fr

as alternatives of skyrmions [16–23]. Of particular interest are skyrmions in synthetic antiferromagnets (SAF), which have been recently demonstrated to be stable at room temperature [20–22]. These skyrmions are resilient against external stray magnetic fields and possess velocities, which are much higher than their ferromagnetic counterparts, making them good candidates for elementary building blocks in storage and logic. In addition to using magnetic configurations with zero topological charge, SKHE can also be suppressed with different confinement techniques [24–29]. In particular, we have shown that skyrmion channels can be defined by a local modification of the magnetic properties using light ion irradiation. This can be exploited to guide the skyrmion dynamics and suppress the skyrmion Hall effect [28]. This technique can also be leveraged for a reliable control of the skyrmion position in racetracks as well as to guide their motion in complex geometries to achieve logic operations.

In this work, using micromagnetic simulations, we propose a logic-in-memory device based on SAF skyrmions, which leverages skyrmion confinement using local variation of the anisotropy. It combines two main innovations: Firstly, a concept of racetrack shift register where the skyrmion position is defined by low anisotropy dots and moved reliably using current induced spin-orbit torques. This provides an elegant solution to the issue of reliability of data retention and shift operation in racetracks. We develop elementary protocols for basic “nucleate” and “shift” operations on a train of skyrmions. Secondly, a compact full-adder (FA) logic gate extendable to n -bit FA by cascading is designed. The designed FA can be reprogrammed to perform different logic operations (e.g., NOT, BUFFER, AND, OR, XOR, NXOR, NAND). Our proposal allows a simple and intuitive synchronization scheme, which in turn enables us to seamlessly cascade logic design to implement large-scale networks without any additional electronic circuitry. The designed logic architecture can also tolerate deviations in the amplitude or width of the current pulses, which may arise from the electrical part of the design.

II. SKYRMION NUCLEATION AND CONFINEMENT

As shown in our previous work [28], it is possible to artificially create an energy barrier for a skyrmion using focused He^+ ion irradiation, which locally modifies the material properties [anisotropy and Dzyaloshinskii-Moriya interaction (DMI)] of the ferromagnet. Using this feature, we can engineer regions of different anisotropy values specifically tuned for two different tasks, namely nucleation and confinement of skyrmions.

For the micromagnetic simulations, a SAF structure is assumed composed of two Co layers with a thickness of 0.9-nm antiferromagnetically coupled by

Ruderman-Kittel-Kasuya-Yosida (RKKY) interaction and separated by a thin spacer [see Appendix A for details regarding the micromagnetic model and parameters]. The magnetic parameters for the irradiated and nonirradiated areas are close to the ones of the Pt/Co/MgO stacks measured experimentally in Ref. [28]. The He^+ ion irradiation leads to a decrease of the perpendicular magnetic anisotropy as well as the DMI.

In Fig. 1(a), we show the schematic of a skyrmion generator device coupled with a memory (bit) cell, which can physically confine and hold a skyrmion inside of it. There are three different regions of anisotropy. The larger 150-nm square-shaped cell (R_1) with a very small effective anisotropy of $B_{K,\text{eff}} = 5$ mT is used to nucleate the skyrmion. The anisotropy of this region is intentionally kept smaller to reduce the current density required to nucleate the skyrmion. The second region is composed of the 80-nm bit cells (R_2 and R_3) with $B_{K,\text{eff}} = 100$ mT, which are surrounded by high anisotropy regions of $B_{K,\text{eff}} = 165$ mT. These act as a barrier for the skyrmion, effectively trapping the skyrmion inside the bit cells. In Fig. 1(b), we show the energy of the skyrmion as a function of its position x as it passes from region R_2 to R_3 . The energy is highest in the middle of regions R_2 and R_3 . The energy barrier is approximately $33k_B T$ providing reasonable stability to the device against thermal noise. More information on the calculation and optimization of the energy barrier is given in Appendix B.

To nucleate and shift the skyrmions, three separate contacts C_1 , C_2 , and C_3 are used [Fig. 1(a)]. To nucleate a skyrmion in R_1 [first panel of Fig. 1(c)], a short (0.3 ns) current pulse ($J = 15$ MA cm^{-2}) is first injected in between contacts $C_1 - C_2$. A second current pulse with a higher current density ($J = 35$ MA cm^{-2} for 0.55 ns) allows the skyrmion to overcome the barrier and move from R_1 towards R_2 . Simultaneously, due to the high current density, another skyrmion nucleates in the region R_1 . This whole operation can thus be dubbed as “fire+nucleate” as one skyrmion is nucleated and another skyrmion is simultaneously fired [panel 2 of Fig. 1(c)]. Once nucleated, the shift operation is performed by injecting a current between contacts C_2 and C_3 such that the skyrmions move between $R_2 - R_3$ without influencing the nucleation region R_1 [third row panel in Fig. 1(c)]. Overall, to encode a full data stream, the following protocol is adopted: (1) Nucleate in region R_1 . (2) If bit 1 is required, use “fire+nucleate” followed by “shift.” (3) If bit 0 is required, use only “shift.” We show in Fig. 1(d) and in Video S1 within the Supplemental Material [30] the encoding of an 8-bit binary number “10001011” in a racetrack composed of nine cells using this protocol. After nucleation, the skyrmion train shifts synchronously along the cell track when injecting the current pulses. Thus, our proposed racetrack design based on confinement cells and current-induced tunneling provides large bit stability

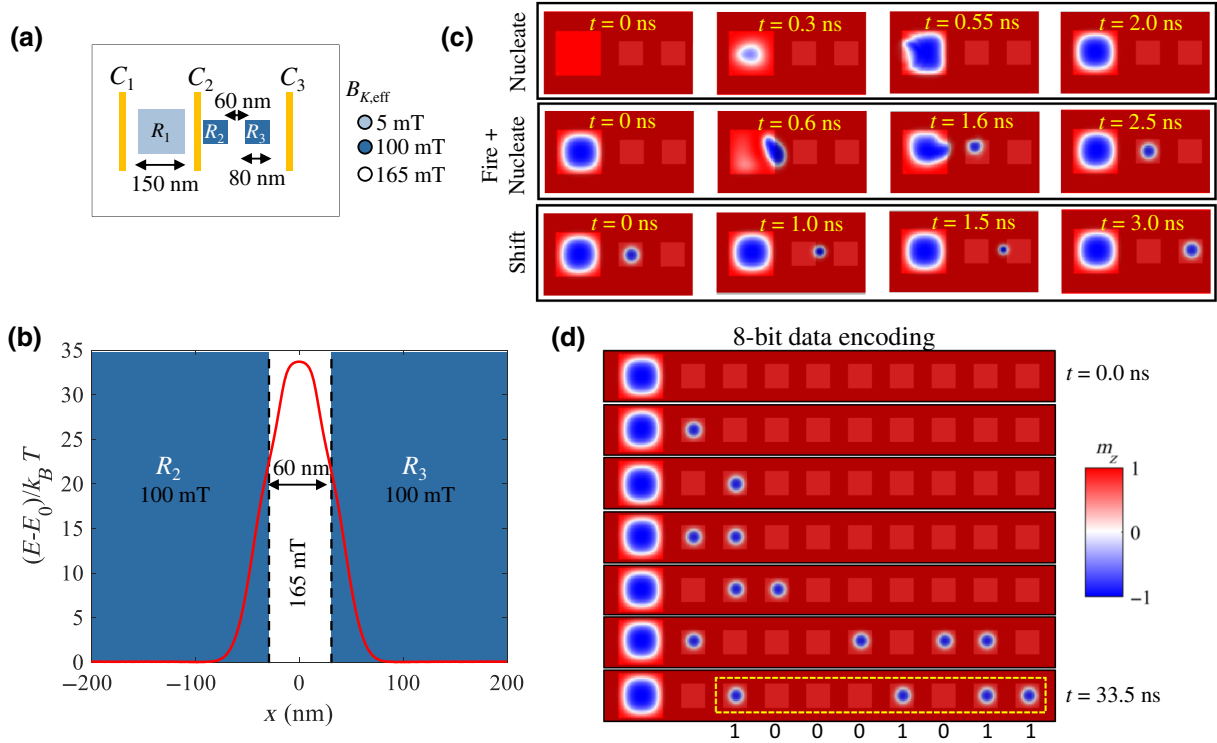


FIG. 1. (a) Schematic of a skyrmion generator with regions of different anisotropy values. Current can be sent in between $C_1 - C_2$ or $C_2 - C_3$. (b) Energy as the skyrmion moves between two bit cells R_2 and R_3 . E_0 is the energy of the skyrmions inside the bit cell. The anisotropy fields are indicated in the different regions. (c) Protocol for data encoding: a skyrmion is first nucleated in the low anisotropy region (row 1) by sending current between $C_1 - C_2$ ($J = 15 \text{ MA cm}^{-2}$ for 0.3 ns) and then pushed (“fired”) to the 80-nm cells (R_2), which work as memory bits ($J = 35 \text{ MA cm}^{-2}$ for 0.55 ns). A shift operation is carried out by sending current between $C_2 - C_3$ (row 3) ($J = 15 \text{ MA cm}^{-2}$ for 0.9 ns). We show only the top layer magnetization of the SAF as the bottom layer is magnetized symmetrically opposite to the top layer due to strong RKKY coupling. (d) Full 8-bit skyrmion data encoding of an 8-bit binary number “10001011” using operations given in (c). Note that the data is encoded starting first from the least significant digit.

and natural synchronization of the bit train, solving an important issue of initial skyrmion racetrack design.

III. FULL-ADDER DESIGN

A compact FA logic gate can also be designed using anisotropy energy barriers in irradiated SAF stacks. A full adder is considered as the basic unit in arithmetic logic units (ALUs) performing bitwise addition of two binary numbers. It has three inputs, A , B , and $carry\ in$ (C_{in}) and produces two outputs, Sum and $carry\ out$ (C_{out}). The logic gate is divided into three parts [see Fig. 2(a)]. The left part of the device is made of three bit cells, which are the inputs A , B , and C_{in} . Similarly, at the right-hand side of the logic gate, three other bit cells hold the two outputs Sum and C_{out} of the FA. The middle region of the design represents the part of the device that performs the logic operation. All three regions and the individual bit cells are separated from each other by anisotropy energy barriers.

We show the operation of the designed full-adder gate in Figs. 2(b)–2(f) and in Video S2 within the Supplemental Material [30]: the skyrmions which are present

inside the bit cells inputs A , B , and C_{in} on the left region are pushed inside the middle region after passing over the energy barrier by sending a current pulse uniformly through the device ($J_1 = 15 \text{ MA cm}^{-2}$). When skyrmions are inside the middle region, different possibilities arise depending on the total number of skyrmions inside the middle region. For only one skyrmion [Figs. 2(b) and 2(c)], the most stable state for the skyrmion is to stay at the center of the middle region. However, when there are two skyrmions [Figs. 2(d) and 2(e)], they repel each other and would stay near the top and bottom edge of the middle region to minimize their mutual interaction. When the total number of skyrmions in the middle region is three [Fig. 2(f)], the skyrmions will maximize the individual distances between them and will acquire a position at the top, middle, and bottom parts. To achieve the relaxation of the interacting skyrmions in the middle part, two pulses with smaller amplitude $J_2 = 5 \text{ MA cm}^{-2}$ are then injected. The skyrmions are then pushed into the rightmost region by sending a current pulse with a larger amplitude J_1 . At the output end, the middle cell will only have a skyrmion if the total number of skyrmions on the input side is either

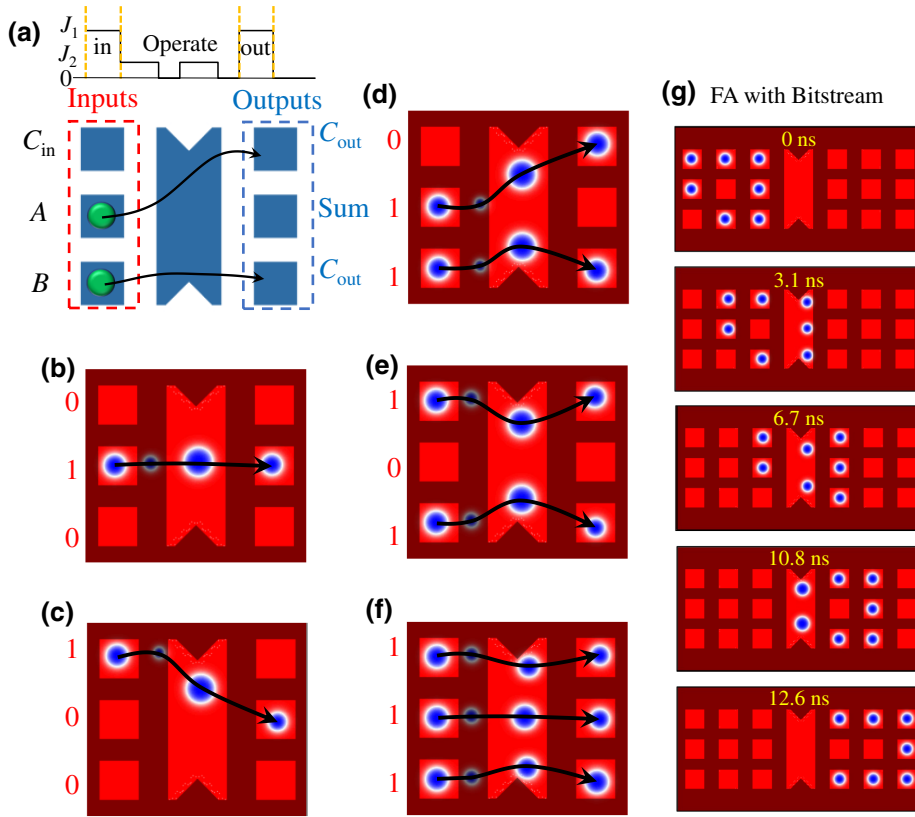


FIG. 2. Design and operation of a 1-bit full adder (FA) based on irradiated SAF. (a) Schematic of the FA and the current pulses, which are designed to move the skyrmions from their respective bits on the left to the middle region where the operation takes place and then to the output bits on the right. The current pulses J_1 and J_2 have an amplitude and duration of $J_1 = 15 \text{ MA cm}^{-2}$ and $t_1 = 0.8 \text{ ns}$ and $J_2 = 5 \text{ MA cm}^{-2}$ and $t_2 = 0.9 \text{ ns}$. (b)–(f) Micromagnetic simulation of the 1-bit FA gate for various test cases. Snapshot of the magnetization at different times are overlaid to show the complete trajectories of the skyrmions (black solid curves). (g) Serial FA operation on a stream of C_{in} , A , and B inputs.

one or three. This represents the *Sum* output of the full adder. Similarly, the top and bottom outputs will only have a skyrmion if the total number of input skyrmions is either two or three. This is the same as the C_{out} (carry out) output of the full adder. We thus achieve an entire full-adder operation with a single logic gate.

We have specifically designed our FA logic gate in Fig. 2(a) keeping in mind an easy integration with racetrack storage such as the one in Fig. 1(a). Using this architecture, the FA logic gate can be easily extended to perform logic operations with a continuous stream of data to serially compute *Sum* and C_{out} outputs for each input set of data bits. Such an operation is shown in Fig. 2(g), where we show serial 1-bit FA calculation on streams of input A , B , and C_{in} (see also Video S3 within the Supplemental Material [30]). The current pulses, in general, follow the same protocol and can be easily extended to perform n operations, where n is the total number of bits in the bit stream [31].

IV. FULL-ADDER MULTIBIT OPERATION

One limitation of the FA gate is that it only operates on 1-bit inputs. For instance, the actual operation on the bitstreams shown in Fig. 2(g) is a 1-bit FA operation done serially on each arriving set of input bits, since the carry bit (C_{out}) is not taken forward in each successive computation. We show here that N -bit FA gates can be easily designed

by properly cascading 1-bit FA gates. As an example, we present in Fig. 3(a) the design of a 3-bit FA gate composed of three cascaded FA gates where the carry bit from each computation is used in the next computation. The bottom C_{out} from the FA1 is connected to the top bit cell of FA2, such that the output C_{out} from FA1 operation is used as one of the inputs of FA2 operation. To ensure that the two other inputs of FA2 reach at the same time as the output C_{out} of FA1, we add some delay gates Sync In (green boxes) before FA2. These synchronization gates can be cascaded to vary the delay time and provide on-demand synchronization between various segments of logic operations. Similar gates are also added to synchronize the outputs (Sync Out, black).

In Fig. 3(b) and Video S4 within the Supplemental Material [30], we show the operation of a 3-bit FA using cascaded 1-bit FAs. The calculation performed is $(a_3 \ a_2 \ a_1) + (b_3 \ b_2 \ b_1) = (c_4 \ c_3 \ c_2 \ c_1)$, where $(a_3 \ a_2 \ a_1) = (101)$ and $(b_3 \ b_2 \ b_1) = (111)$. The expected output is $(c_4 \ c_3 \ c_2 \ c_1) = (1100)$, which is correctly obtained by bit cells c_4, c_3, c_2 , and c_1 shown by dotted yellow boxes.

The design of Fig. 3(a) is very promising for fast n -bit FA operation and for further modifications using cascading and synchronization. However, a more compact and low-power design can be implemented by directly computing the n -bit FA operation using streams of skyrmions in two racetracks and a single FA gate. This would be similar to the operation of FA shown in Fig. 2(g), however,

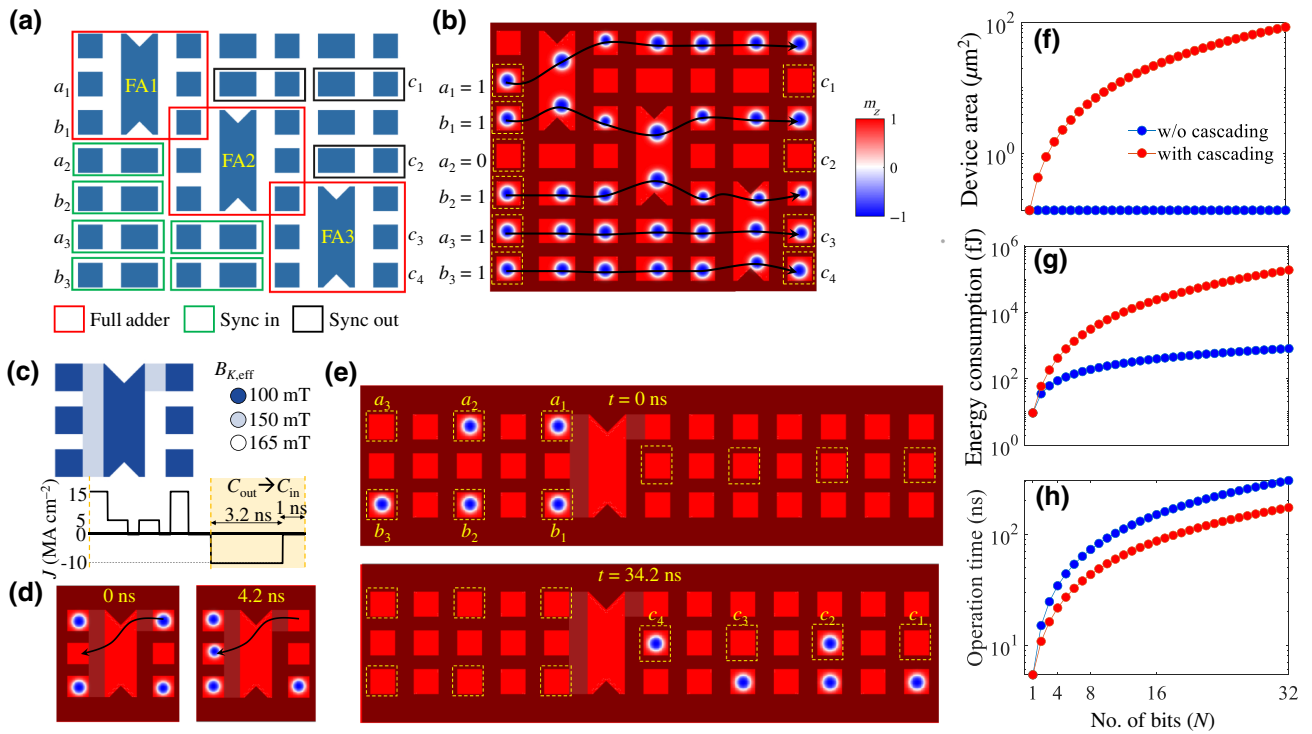


FIG. 3. (a) Design of a 3-bit FA by cascading three 1-bit FAs. Additional components are added to synchronize inputs' application and the entire operation (b) shows the micromagnetic simulation for the case of $(101 + 111 = 1100)$. The current pulses used to perform the logic operation are exactly the same as the 1-bit FA operation repeated 3 times. (c),(d) The design and operation of a FA where the carry bit can be sent back using the current of negative polarity with density $J = -10 \text{ MA cm}^{-2}$ for $t = 3.2 \text{ ns}$ followed by an off pulse ($J = 0 \text{ MA cm}^{-2}$) of 1 ns . (e) Operation of a 3-bit FA using the modified FA shown in (c). The operation corresponds to the case $(011 + 111 = 1010)$. (f)–(h) Comparison of both approaches for 3-bit addition in terms of device area, energy consumption, and operation time, respectively.

with bit streams replaced with N -bit numbers. For this purpose, we modify our FA design to include another region of different anisotropy value as shown in Fig. 3(c). This region (light blue), which covers the area joining the input bits to the middle region and the area to the left of the top C_{out} bit has an anisotropy in between the anisotropy of the regular bit cells and the barrier region. This enables us to send a negative polarity current pulse, which is enough to move only the C_{out} bit while keeping the other skyrmions at their respective places. This pulse is shown alongside in Fig. 3(c) (highlighted part). In Fig. 3(d), we show how the C_{out} bit is moved back to the input end by this current pulse of negative polarity. During the entire operation, none of the other skyrmions are affected. Using this, we can now compute the addition operation of two N -bit numbers by successively sending them through the top and bottom inputs of the FA (middle input is left empty). The design and operation for a 3-bit FA using this strategy is shown in Fig. 3(e) and Video S5 within the Supplemental Material [30]. It may be noted that we need to add an empty bit between each successive bit as no operation can take place during the movement of C_{out} bit using negative polarity current. The snapshots

in Fig. 3(e) correspond to the initial and final stage of the addition operation $(a_3 a_2 a_1) + (b_3 b_2 b_1) = (c_4 c_3 c_2 c_1)$, where, $(a_3 a_2 a_1) = (011)$, $(b_3 b_2 b_1) = (111)$ and the expected output is $(c_4 c_3 c_2 c_1) = (1010)$, which is correctly computed with our implementation. The inputs and outputs are shown in dotted yellow boxes.

The performance of the two methods for N -bit FA computation described above can be compared on the basis of three important characteristic metrics: (i) device area, (ii) energy consumption, and (iii) operation time. In Fig. 3(f), we show the variation of the device area with the number of bits (N). For the cascaded FA shown in Fig. 3(a), both dimensions in the x - y plane increase linearly with N leading to an $\mathcal{O}(N^2)$ variation of device area. However, since only one FA gate is needed for the alternative approach (without cascading) shown in Figs. 3(c)–3(e), the device area remains the same as “ N ” increases. Note that in Fig. 3(e), the increase in the length along x axis is not included in the calculation as the extended regions are considered as a part of the racetrack storage rather than logic. In Fig. 3(g), we show the total energy consumption of the FA logic gate as a function of the number of bits. For a 1-bit operation, the energy of the operation is 9

fJ. In the case of cascaded FA operation, the energy dramatically increases with the number of bits as $\mathcal{O}(N^3)$ due to linear increase in operation time and quadratic increase in the device area. However, when using the FA design shown in Figs. 3(c)–3(e), the total energy only increases linearly with the number of bits. For a 32-bit operation, the total energy consumption is approximately 0.8 pJ, which is more than 2 orders lower than the cascaded FA operation. However, this design suffers from a larger operation time compared to cascaded FA gate since each operation is delayed by 4.2 ns to move the carry-out bit back to the input position [Fig. 3(h)]. Due to the Joule dissipation in the metallic conductors, the proposed device consumes 1–2 orders of magnitude more energy for individual logic operations compared to state-of-art CMOS full adder [32]. However, owing to its intrinsic logic-in-memory approach, a large gain in energy and delay is expected at the device level, since it minimizes the energy dissipated in interconnects and the stand-by power. Note that the speed of both designs can be further increased by using skyrmions with higher velocity (> 1000 m/s is expected for SAF [33], which is 6 times the maximum velocity of skyrmions in this work).

V. PROGRAMMABLE FA LOGIC

Full-adder logic gates can be easily modified in order to build a number of basic logic functions, which in turn can be combined to perform complex logic operations. The output of the full-adder gate is given as $Sum = C_{in} \oplus (A \oplus B)$; $C_{out} = AB + BC_{in} + AC_{in}$. By fixing one or more of the inputs, different logic gates can be achieved. Note that the inputs of a FA logic gate are all interchangeable, so it does

not matter which of the inputs are fixed. Figures 4(a)–4(c) show the modified FA gates, which can perform COPY, NOT, AND, XOR, OR and NXOR operations.

The universal NAND gate can also be designed by cascading two modified FA gates [Fig. 4(d)]. The first gate is modified by setting one of the inputs to “0” such that the C_{out} represents the output of an AND gate. The second FA gate is modified to perform an inversion operation by setting two of the inputs to “0” and “1.” The operation of this gate for a test case of $A = 1$ and $B = 1$ is shown in Fig. 4(e). The output $\overline{A \cdot B} = 0$ corresponding to the NAND gate is highlighted. It may be noted that we also obtain AND and XOR outputs along with the NAND output.

VI. ELECTRICAL TOLERANCES AND FAILURE MECHANISMS

For practical application of our design, it is useful to probe its tolerance against variations in the current amplitude or pulse width. In Fig. 5(a), we show the set of pulses used for the 1-bit full-adder operation. There are two different current density values used in the circuit: $J_1 = 15 \text{ MA cm}^{-2}$ and $J_2 = 5 \text{ MA cm}^{-2}$ with pulse widths of $t_1 = 0.8 \text{ ns}$ and $t_2 = 0.9 \text{ ns}$, respectively. There is also an off pulse of width $t_0 = 0.5 \text{ ns}$. We individually vary each of these parameters by fixing other parameters at their reference values. We also try to vary J_1 , J_2 or t_0 , t_1 , and t_2 , simultaneously. The results are shown in Fig. 5(b) depicting the acceptable level of variation for each case for which the logic operations are executed without any errors. Note that these results are also valid for the NAND as well as other derived logic gates. These tolerance

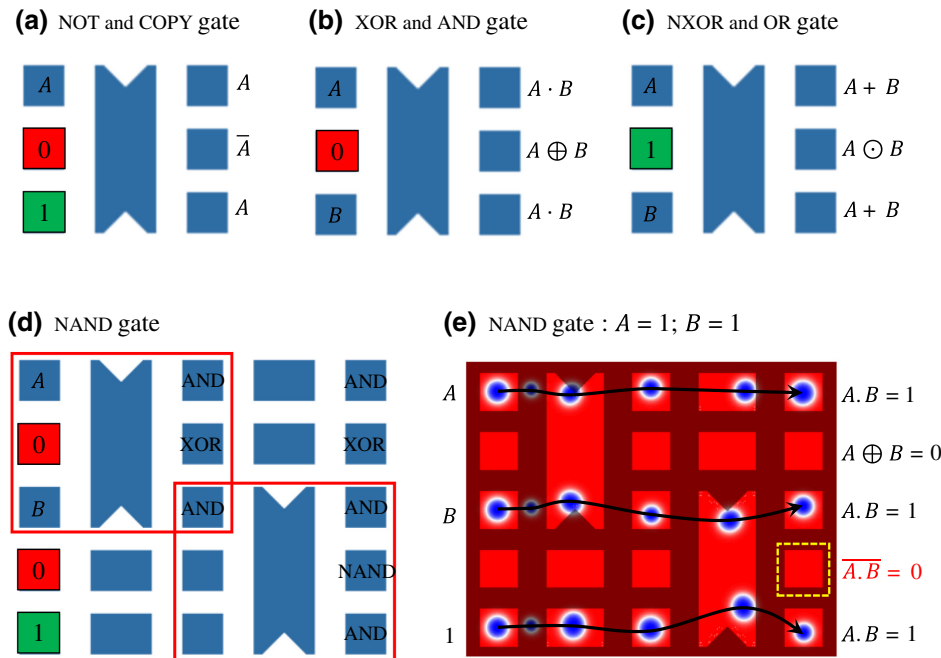


FIG. 4. Different logic gate designs derived from the FA. (a) A NOT/COPY gate design. Some input bits are fixed to either “0” or “1.” Similarly, (b),(c) show XOR/AND, and NXOR/OR gates derived from FA. (d) shows a universal NAND gate designed by cascading AND and NOT gates, both of which are modified versions of FA. (e) The operation of NAND for both inputs as 1.

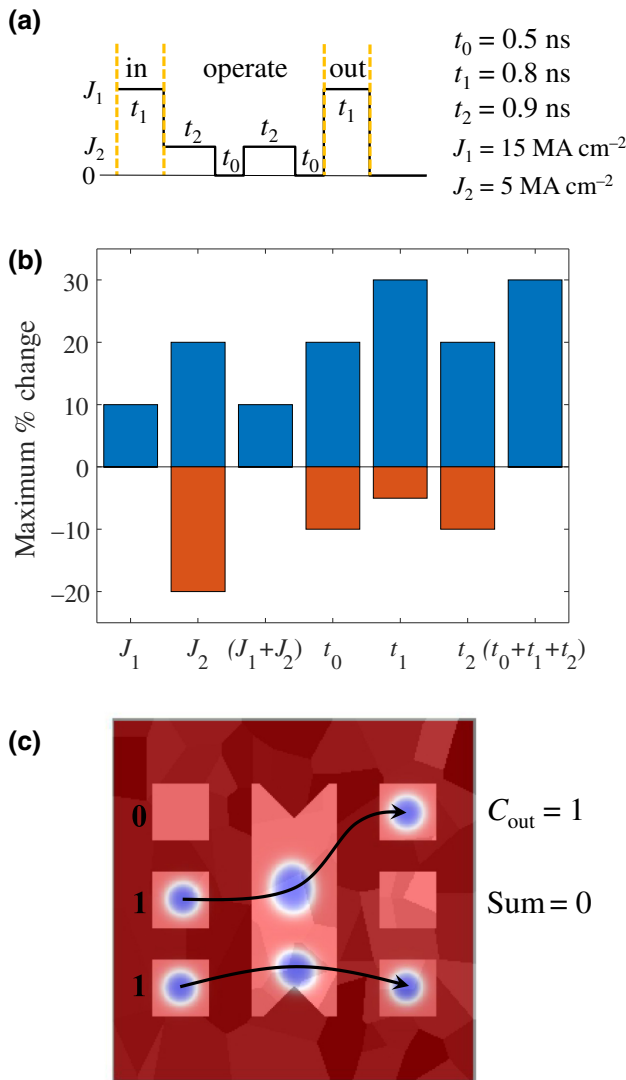


FIG. 5. (a) Current pulses used for FA operation in Fig. 2. (b) Acceptable change (%) in different parameter values in the current pulse. (c) Test case with inputs (0,1,1) for FA operation with random anisotropy grains.

values provide valuable input for further modeling with large-scale electrical circuit designing tools.

We also perform simulations to check the robustness of device against variations in anisotropy. We vary the effective anisotropy field of the barrier region and find that the full adder works correctly for the range of values 155–167 mT. Further, we also perform simulations by adding grains of size 15 nm with random 0.5% variation of anisotropy constant (equivalent to approximately 5%–8% change in effective anisotropy field), similar to previous works [34,35] and show that the full adder works correctly. The motion of skyrmions for a test case with inputs (0,1,1) is shown in Fig. 5(c). However, we note that a stronger anisotropy variation may lead to incorrect operations. In our device, this happens as the skyrmions are stabilized for

the parameter values, which are close to in-plane to out-of-plane transition regime. Therefore, a small change in the anisotropy constant rapidly changes the skyrmion energy and hence its trajectory. A possible strategy to mitigate this is to use higher anisotropy thin films as the pristine sample and then irradiating them to obtain the tracks and barriers with the same anisotropy difference as used in this work.

A part of the logic gate design involves device testing against a possible failure during operation. To investigate this point, we include thermal noise corresponding to $T = 350$ K in our simulations [36]. We simulate 20 instances for each of the possible eight test cases for the full adder (total 160 attempts). Despite high thermal noise, the full adder worked correctly for approximately 85% of the total attempts. For the cases in which it failed, three different failure mechanisms were found. The first one is caused by the annihilation of input skyrmions during the operation. We predict that a stronger RKKY coupling could counter the thermal noise and improve the overall stability of the SAF skyrmions. The second failure mechanism is due to the presence of skyrmions at the wrong output after the operation has ended. This happens when the skyrmion diffusion due to thermal noise significantly perturbs the intended trajectory of the skyrmions. Higher Gilbert damping is expected to minimize this effect. However, the trade-off will be an increased energy consumption as higher current densities will be required to move the skyrmions. The third possibility is a skyrmion getting stuck in between two bit cells. This occurs when the skyrmion finds a local minima in between two memory bits. Our barrier has already been optimized to avoid this situation (see Appendix B), however, further optimization is needed to comply with high-temperature conditions. This involves increasing the difference of anisotropy between the low and high anisotropy regions, which will increase the barrier height.

VII. CONCLUSION

To conclude, we propose a logic-in-memory device based on skyrmion confinement and channeling by anisotropy energy barriers in synthetic antiferromagnets. We first design a racetrack shift register memory based on skyrmions confined by anisotropy energy barriers as memory bits, allowing reliable data storage with large stability and robust synchronous shift operations on skyrmion train. This design naturally solves the stability and synchronization issues of the initial racetrack concept without introducing additional gates or complex geometries. We then combine the racetrack storage with our designed 1-bit full-adder gate extendable to N -bits FA by cascading. The designed FA is reprogrammable and can also be used to perform AND, OR, NOT, NAND, XOR, and NXOR operations. The device is expected to perform well even with some fluctuations in the amplitude and width of the injected

current pulses and in the presence of thermal noise. The simplicity and compactness of the design combined with the minimal use of electrical circuitry brings the proposed device to the same level as the current technological capabilities of fabrication and manufacturing processes, thus providing significant advance for the development of skyrmion-based logic-in-memory technology.

ACKNOWLEDGMENTS

The authors acknowledge financial support from the French National Research Agency (ANR) (Grants No. ANR-15-CE24-0015-01 and No. ANR-17-CE24-0045) and the American Defense Advanced Research Project Agency (DARPA) TEE program (Grant No. MIPR HR0011831554). This work is partially supported by MIAI@Grenoble Alpes (Grant No. ANR-19-P3IA-0003).

APPENDIX A: MICROMAGNETIC SIMULATION MODEL

We use the micromagnetic package *Mumax*³ [37] to simulate the magnetization dynamics of a synthetic anti-ferromagnet having a heavy metal (HM)/ferromagnet (FM)/spacer (S)/ferromagnet (FM) structure by solving Landau-Lifshitz-Gilbert (LLG) equation with an additional spin-orbit torque (SOT) term.

$$\begin{aligned} \frac{d\mathbf{m}_i}{dt} = & -\gamma(\mathbf{m}_i \times \mathbf{B}_{\text{eff},i}) + \alpha \left(\mathbf{m}_i \times \frac{d\mathbf{m}_i}{dt} \right) \\ & - \eta \left(\mathbf{m}_i \times (\mathbf{m}_i \times \mathbf{e}_p) + \epsilon' (\mathbf{m}_i \times \mathbf{e}_p) \right) \\ \eta = & \frac{\hbar \theta_H J}{2|e|M_S d}. \end{aligned} \quad (\text{A1})$$

Here \mathbf{m}_i is the normalized magnetic moment of the i th ferromagnetic layer. $B_{\text{eff},i}$ is the effective field on each of the layers, which contain contributions from the anisotropy, exchange, DMI, and magnetostatic interactions along with the additional field to account for RKKY coupling between layers. We assume that SOT acts only on the top ferromagnetic layer of the SAF, which is in contact with a heavy metal, which generates a spin-orbit torque. The spin Hall angle is assumed to be $\theta_H = 0.4$. The ratio of field-like torque to dampinglike torque is $\epsilon' = 0.01$. The RKKY coupling strength between FM layers is -0.2 mJ/m^2 . The thickness of each layer in the FM/S/FM trilayer structure is 0.9 nm (HM not included in micromagnetic simulations). The simulation region is divided into micromagnetic cells of size $2 \times 2 \times 0.9 \text{ nm}^3$. Unless specified explicitly, we use the two sets of parameters corresponding to irradiated and nonirradiated regions of the SAF, given in Table I. These parameters are similar to our previous experimental work on He⁺-ion irradiated Co-based FM [28]. However, we assume higher anisotropy values to stabilize the skyrmions

TABLE I. Simulation parameters.

Parameter	Irradiated	Nonirradiated
M_s (MA/m)	1.32	1.32
$B_{K,\text{eff}}$ (mT)	100	165
$ D $ (mJ/m ²)	1.082	1.122
A (pJ/m)	16	16
α	0.05	0.05

at zero field, while keeping the difference between the anisotropy values of the irradiated and nonirradiated region the same. We also note that for the He-ion dosage ($< 2 \times 10^{14} \text{ ions/cm}^2$) required to create the anisotropy barriers used in this work, the irradiation is expected to change the resistivity of heavy metal, which generates the SOT (Pt in the reference experiments [28]) by $< 1\%$. Therefore, we do not expect any changes in the dynamics of the skyrmions due to nonuniform current flow.

APPENDIX B: ENERGY BARRIER CALCULATION

The energy of the skyrmion crossing the anisotropy barrier is dependent on the position of the skyrmion as well as the skyrmion radius: $E' = f(x, R)$. From our simulations, we calculate the total micromagnetic energy for all values of x and R for a particular barrier width W . We then reduce the map E' using $E = \min(E') = \min[f(x, R)]$ taking the minimum along the R dimension. We show E' scans along x with varying R in Fig. 6(a). The curve corresponding to minima of E' is shown by solid black circles. This corresponds to Fig. 1(b) in the main text for $W = 60 \text{ nm}$. In Fig. 6(b), we show the anisotropy barrier as a function of the width of the barrier. For low barrier width $W = 10 \text{ nm}$, the barrier has lower peak value ($< 10k_B T$). Moreover, the barrier also has a local minima at the center. This is not desirable for skyrmion confinement as there is a possibility of the skyrmion getting stuck in the middle of the barrier during logic operation. For high barrier width ($W = 100 \text{ nm}$), while the barrier height is sufficient (approximately $33k_B T$), the shape of the barrier is flat around the center region. This is also an unfavorable situation as in the absence of current, there will be no driving force on the skyrmion to push it towards the regions of confinement. In comparison, $W = 60 \text{ nm}$ (which is used in main text) has an ideal shape. Due to the sharp peak at the center, the barrier is always forcing the skyrmions to move towards lower anisotropy regions where they can be confined.

APPENDIX C: 4-INPUT LOGIC DEVICE

The FA design shown in the main text can also be used as a template to design logic gates with more than three inputs. We show one such example in Fig. 7, which supports four skyrmion inputs. Using the same protocol as for

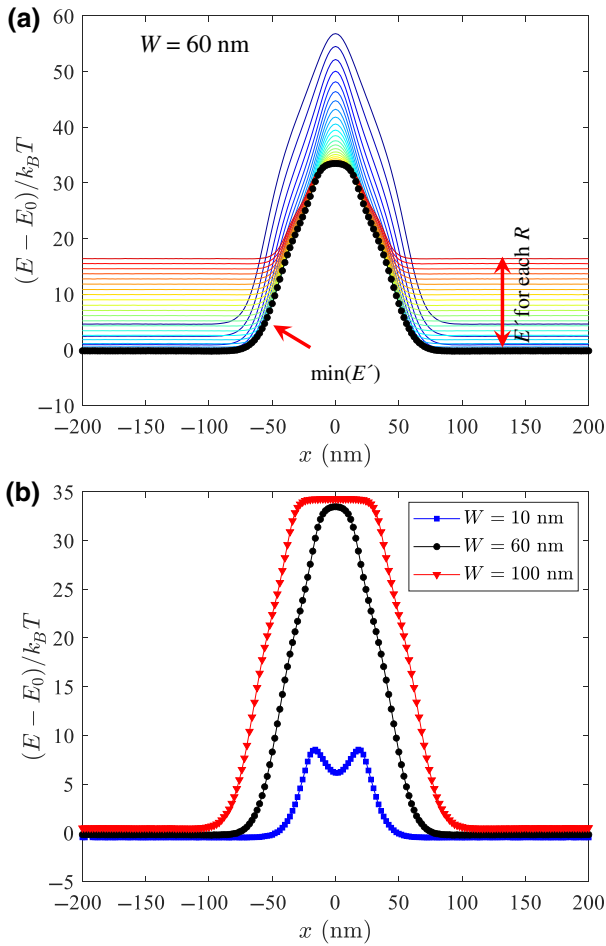


FIG. 6. (a) Energy of a rigid skyrmion passing through the barrier as a function of its position for different values of skyrmion radius. (b) Barrier energy as a function of skyrmion position for different values of barrier widths, W .

the FA gate, we perform full micromagnetic simulations to obtain the truth table corresponding to the logic functionality of the 4-input gate [Table II]. This truth table can be used to customize the logic gate to calculate different logic functions. For example, if the input A is set to 0, the output of O_2 represents the operation $B \oplus (C \cdot D)$.

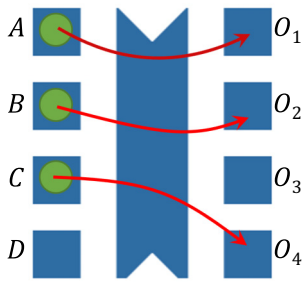


FIG. 7. Illustration of a 4-input logic gate.

TABLE II. Truth table for the 4-input logic gate.

A	B	C	D	O_1	O_2	O_3	O_4
0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0
0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0
0	0	0	1	0	0	1	0
0	0	1	1	0	1	0	1
0	1	1	0	0	1	1	0
1	1	0	0	1	0	1	0
1	0	1	0	0	1	1	0
0	1	0	1	0	1	1	0
1	0	0	1	0	1	1	0
0	1	1	1	1	0	1	1
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	0	1	1	0	1
1	1	1	1	1	1	1	1

- [1] M. Song, M. G. Park, S. Ko, S. K. Jang, M. Je, and K.-J. Kim, Logic device based on skyrmion annihilation, *IEEE Trans. Electron Devices* **68**, 1939 (2021).
- [2] L. Gnoli, F. Riente, M. Vacca, M. Ruo Roch, and M. Graziano, Skyrmion logic-in-memory architecture for maximum/minimum search, *Electronics* **10**, 155 (2021).
- [3] H. Zhang, D. Zhu, W. Kang, Y. Zhang, and W. Zhao, Stochastic Computing Implemented by Skyrmionic Logic Devices, *Phys. Rev. Appl.* **13**, 054049 (2020).
- [4] M. Chauwin, X. Hu, F. Garcia-Sanchez, N. Betrabet, A. Paler, C. Moutafis, and J. S. Friedman, Skyrmion Logic System for Large-Scale Reversible Computation, *Phys. Rev. Appl.* **12**, 064053 (2019).
- [5] Z. Zhang, Y. Zhu, Y. Zhang, K. Zhang, J. Nan, Z. Zheng, Y. Zhang, and W. Zhao, Skyrmion-based ultra-low power electric-field-controlled reconfigurable (super) logic gate, *IEEE Electron Device Lett.* **40**, 1984 (2019).
- [6] Z. He, S. Angizi, and D. Fan, Current-induced dynamics of multiple skyrmions with domain-wall pair and skyrmion-based majority gate design, *IEEE Magn. Lett.* **8**, 1 (2017).
- [7] X. Xing, P. W. Pong, and Y. Zhou, Skyrmion domain wall collision and domain wall-gated skyrmion logic, *Phys. Rev. B* **94**, 054408 (2016).
- [8] X. Zhang, M. Ezawa, and Y. Zhou, Magnetic skyrmion logic gates: Conversion, duplication and merging of skyrmions, *Sci. Rep.* **5**, 9400 (2015).
- [9] S. Luo, M. Song, X. Li, Y. Zhang, J. Hong, X. Yang, X. Zou, N. Xu, and L. You, Reconfigurable skyrmion logic gates, *Nano Lett.* **18**, 1180 (2018).
- [10] Z. Yan, Y. Liu, Y. Guang, K. Yue, J. Feng, R. Lake, G. Yu, and X. Han, Skyrmion-Based Programmable Logic Device with Complete Boolean Logic Functions, *Phys. Rev. Appl.* **15**, 064004 (2021).
- [11] M. G. Mankalale, Z. Zhao, J.-P. Wang, and S. S. Sapatnekar, Skylogic—a proposal for a skyrmion-based logic device, *IEEE Trans. Electron Devices* **66**, 1990 (2019).

- [12] M. Fattouhi, K. Y. Mak, Y. Zhou, X. Zhang, X. Liu, and M. El Hafidi, Logic Gates Based on Synthetic Antiferromagnetic Bilayer Skyrmions, *Phys. Rev. Appl.* **16**, 014040 (2021).
- [13] A. Fert, V. Cros, and J. Sampaio, Skyrmions on the track, *Nat. Nano* **8**, 152 (2013).
- [14] J. Müller, Magnetic skyrmions on a two-lane racetrack, *New J. Phys.* **19**, 025002 (2017).
- [15] W. Kang, C. Zheng, Y. Huang, X. Zhang, Y. Zhou, W. Lv, and W. Zhao, Complementary skyrmion racetrack memory with voltage manipulation, *IEEE Electron Device Lett.* **37**, 924 (2016).
- [16] X. Zhang, Y. Zhou, and M. Ezawa, Antiferromagnetic skyrmion: Stability, creation and manipulation, *Sci. Rep.* **6**, 24795 (2016).
- [17] C. Jin, C. Song, J. Wang, and Q. Liu, Dynamics of antiferromagnetic skyrmion driven by the spin Hall effect, *Appl. Phys. Lett.* **109**, 182404 (2016).
- [18] A. G. Kolesnikov, M. E. Stebliy, A. S. Samardak, and A. V. Ognev, Skyrmionium – high velocity without the skyrmion Hall effect, *Sci. Rep.* **8**, 16966 (2018).
- [19] N. Sisodia, P. K. Muduli, N. Papanicolaou, and S. Komineas, Chiral droplets and current-driven motion in ferromagnets, *Phys. Rev. B* **103**, 024431 (2021).
- [20] T. Dohi, S. DuttaGupta, S. Fukami, and H. Ohno, Formation and current-induced motion of synthetic antiferromagnetic skyrmion bubbles, *Nat. Commun.* **10**, 5153 (2019).
- [21] W. Legrand, D. Maccariello, F. Ajejas, S. Collin, A. Vecchiola, K. Bouzehouane, N. Reyren, V. Cros, and A. Fert, Room-temperature stabilization of antiferromagnetic skyrmions in synthetic antiferromagnets, *Nat. Mater.* **19**, 34 (2020).
- [22] R. Juge *et al.*, Skyrmions in synthetic antiferromagnets and their nucleation via electrical current and ultrafast laser illumination, arXiv e-prints, [arXiv:2111.11878](https://arxiv.org/abs/2111.11878) (2021).
- [23] X. Zhang, Y. Zhou, and M. Ezawa, Magnetic bilayer-skyrmions without skyrmion Hall effect, *Nat. Commun.* **7**, 10293 (2016).
- [24] S. A. Pathak and R. Hertel, Geometrically constrained skyrmions, *Magnetochemistry* **7**, 26 (2021).
- [25] E. Albisetti, D. Petti, M. Pancaldi, M. Madami, S. Tacchi, J. Curtis, W. King, A. Papp, G. Csaba, W. Porod, P. Vavassori, E. Riedo, and R. Bertacco, Nanopatterning reconfigurable magnetic landscapes via thermally assisted scanning probe lithography, *Nat. Nano.* **11**, 545 (2016).
- [26] E. Albisetti, D. Petti, M. Madami, S. Tacchi, P. Vavassori, E. Riedo, and R. Bertacco, Nanopatterning spin-textures: A route to reconfigurable magnonics, *AIP Adv.* **7**, 055601 (2017).
- [27] Y. Guang, I. Bykova, Y. Liu, G. Yu, E. Goering, M. Weigand, J. Gräfe, S. K. Kim, J. Zhang, H. Zhang, Z. Yan, C. Wan, J. Feng, X. Wang, C. Guo, H. Wei, Y. Peng, Y. Tserkovnyak, X. Han, and G. Schütz, Creating zero-field skyrmions in exchange-biased multilayers through *x*-ray illumination, *Nat. Commun.* **11**, 949 (2020).
- [28] R. Juge, K. Bairagi, K. G. Rana, J. Vogel, M. Sall, D. Mailly, V. T. Pham, Q. Zhang, N. Sisodia, M. Foerster, L. Aballe, M. Belmeguenai, Y. Roussigné, S. Auffret, L. D. Buda-Prejbeanu, G. Gaudin, D. Ravelosona, and O. Boulle, Helium ions put magnetic skyrmions on the track, *Nano Lett.* **21**, 2989 (2021).
- [29] X. Xing, J. Åkerman, and Y. Zhou, Enhanced skyrmion motion via strip domain wall, *Phys. Rev. B* **101**, 214432 (2020).
- [30] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevApplied.18.014025> for Videos S1–S5 showing 8-bit data encoding using “Fire+Nucleate” and “Shift” operations (Video S1), operation of 1-bit FA for all test cases (Video S2), 1-bit FA operating on a stream of inputs A , B , and C_{in} (Video S3), 3-bit addition using three cascaded FA gates (Video S4), and 3-bit addition using a single modified FA gate (Video S5).
- [31] The only modification needed in the current pulses is the merging of J_1 pulse of outgoing operation with the J_1 pulse of incoming operation as both of these operations can be carried out simultaneously without affecting the logic operation.
- [32] D. E. Nikonov and I. A. Young, Benchmarking of beyond-CMOS exploratory devices for logic integrated circuits, *IEEE J. Explor. Solid-State Comput. Devices Circuits* **1**, 3 (2015).
- [33] R. Tomasello, V. Puliafito, E. Martinez, A. Manchon, M. Ricci, M. Carpentieri, and G. Finocchio, Performance of synthetic antiferromagnetic racetrack memory: Domain wall versus skyrmion, *J. Phys. D : Appl. Phys.* **50**, 325302 (2017).
- [34] R. Juge, S.-G. Je, D. de Souza Chaves, S. Pizzini, L. D. Buda-Prejbeanu, L. Aballe, M. Foerster, A. Locatelli, T. O. Menteş, A. Sala, F. Maccherozzi, S. S. Dhesi, S. Auffret, E. Gautier, G. Gaudin, J. Vogel, and O. Boulle, Magnetic skyrmions in confined geometries: Effect of the magnetic field and the disorder, *J. Magn. Magn. Mater.* **455**, 3 (2018).
- [35] I. Gross, W. Akhtar, A. Hrabec, J. Sampaio, L. J. Martínez, S. Chouaieb, B. J. Shields, P. Maletinsky, A. Thiaville, S. Rohart, and V. Jacques, Skyrmion morphology in ultrathin magnetic films, *Phys. Rev. Mater.* **2**, 024406 (2018).
- [36] As we are already doing simulations with experimental parameters extracted at $T = 300$ K, we include only thermal noise corresponding to $T = 50$ K to avoid overestimating the effect of temperature.
- [37] A. Vansteenkiste, J. Leliaert, M. Dvornik, M. Helsen, F. Garcia-Sanchez, and B. Van Waeyenberge, The design and verification of MuMax3, *AIP Adv.* **4**, 107133 (2014).