A Spiking Neuromorphic Architecture Using Gated-RRAM for Associative Memory

ALEXANDER JONES, AARON RUEN, and RASHMI JHA, Dept. of Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, OH

This work reports a spiking neuromorphic architecture for associative memory simulated in a SPICE environment using recently reported gated-RRAM (resistive random-access memory) devices as synapses alongside neurons based on complementary metal-oxide semiconductors (CMOSs). The network utilizes a Verilog A model to capture the behavior of the gated-RRAM devices within the architecture. The model uses parameters obtained from experimental gated-RRAM devices that were fabricated and tested in this work. Using these devices in tandem with CMOS neuron circuitry, our results indicate that the proposed architecture can learn an association in real time and retrieve the learned association when incomplete information is provided. These results show the promise for gated-RRAM devices for associative memory tasks within a spiking neuromorphic architecture framework.

CCS Concepts: • **Hardware → Neural systems**;

Additional Key Words and Phrases: Associative memory, gated-RRAM, neuromorphic applications, segmented attractor network

ACM Reference format:
Alexander Jones, Aaron Ruen, and Rashmi Jha. 2021. A Spiking Neuromorphic Architecture Using Gated-RRAM for Associative Memory. *J. Emerg. Technol. Comput. Syst.* 18, 2, Article 36 (December 2021), 22 pages.
https://doi.org/10.1145/3461667

1 INTRODUCTION

Artificial intelligence is a field of science that receives more understanding with each passing day. As the field increased in complexity throughout the 20th century, demand for a method of executing artificial intelligence algorithms and structures increased. As the end of the century approached, the concept of the "neural network" became more mainstream within the field of artificial intelligence. Once research involving neural networks began to mature, a topic of study called "neuromorphic computing" emerged with an aim to execute artificial neural networks directly in hardware.

Neuromorphic computing in its history has explored many types of methods of executing neural networks. Some architectures have primarily focused on implementing fast execution of typical

This work was supported through the National Science Foundation under the awards ECCS-1926465 and #CCF-1718428.
Authors' address: A. Jones, A. Ruen, and R. Jha, Dept. of Electrical Engineering and Computer Science, University of Cincinnati, Cincinnati, 2851 Woodside Drive, Cincinnati, OH 45219; emails: {jones2a5, ruenan}@mail.uc.edu, jhari@ucmail.uc.edu.

1550-4832/2021/12-ART36 $15.00
https://doi.org/10.1145/3461667

artificial neural networks [1], while others have attempted to explore bio-inspired implementation methods [2–4]. One of the primary approaches for creating bio-inspired neuromorphic hardware is the spiking neural network (SNN). This type of design focuses on temporal learning by either using the timing between specific spikes emitted by neurons in the network or the rate at which neurons within the network spike for learning [5].

Not only has neuromorphic computing studied methods of implementing neural networks but it has also studied designing architectures to specifically implement different types of neural networks. Based on different training datasets and applications, several types of neural networks exist, including feed-forward networks, radial basis function networks, recurrent neural networks, and so forth. [6]. Each network species has its purpose and application space. One application space of interest is the area of associative memory. Associative memory is a concept that focuses on neural networks that can relate certain pre-identified objects, ideas, concepts, and actions with one another [7]. This association capability allows the network to relate complex pieces of information with one another for later use. These stored relations between information can be later used to perform tasks such as pattern completion when only partial information is available in order to form complete memories. For example, this capability could allow a robot with a sensor suite that includes a camera to potentially navigate an environment it previously explored while its camera is not operating, or could allow an object-detection system to develop a concept of object permanence when an object disappears from view but still is able to perceive other evidence that the object is nearby [8]. This type of behavior exhibited by associative memory is also often attributed to a similar concept called *content-addressable memory* in which information within a system can be recalled via partial pieces of information [9].

A primary method of implementing associative memory within a neural network is the Hopfield network [10]. This is a subspecies of recurrent networks in which information can be inserted into the network and then recalled later by giving direct input to some of the neurons within the Hopfield net [10]. Although the Hopfield network is decades old (originally defined in the 1980s), it is still a popular topic of study in a volume of contemporary work through various studies that have implemented versions of it within neuromorphic hardware [11–13]. Milo et al. focused on the concept of spike-time dependent plasticity (STDP) when implementing their network [11], while Hu et al. focused on demonstrating their network's capability for holding multiple memories simultaneously [12]. Other work, such as Yang et al., focused on reducing the number of switches and inverters needed to implement the architecture [13]. These designs vary, but all accomplish the basic goal of implementing a Hopfield network within hardware and being able to recall information from the network after it has been previously provided input.

Although the Hopfield network is still a popular topic of study, one of its major drawbacks is its memory capacity. Studies were done in the years after the Hopfield network was defined on its memory capacity that all show a memory capacity in the realm of 0.12 n to 0.15 n, where n is the number of neurons within the network [14, 15]. Recently, a new type of network like the Hopfield net but with greater memory capacity has been defined, called the Segmented Attractor Network (SAN) [16]. Not only does this network exhibit larger memory capacity than the Hopfield net (e.g., the largest memory capacity shown in [16] is 0.375 n), but the capacity of the network scales non-linearly with the size of the network [16].

Despite recent progress in the development of associative memory in neuromorphic architectures, there is still a critical need for implementing more convenient synaptic devices within these architectures for real-time associative memory formation. Although currently explored two-terminal devices can be used in architectures, they are not particularly suitable due to convoluted read and write paths. This issue creates difficulty when programming the devices during

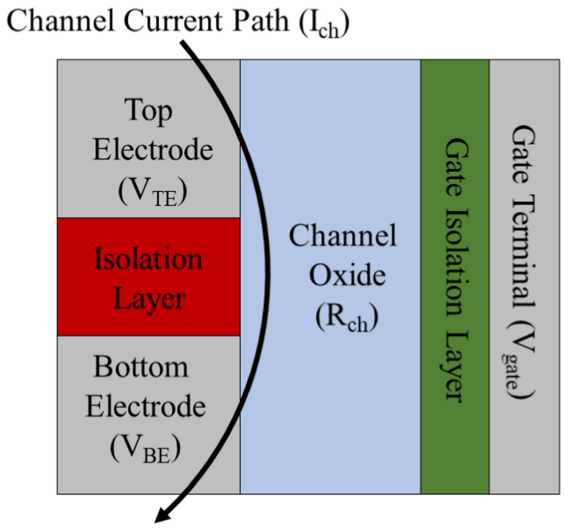

Fig. 1. Diagram showing the basic anatomy of a gated-RRAM device. Device analysis normally occurs via using the current passing from the top to bottom electrode, while the device's state is manipulated by applying voltage to the gate. The isolation and gate isolation layers can either be made of similar or different materials depending on the device design.

times when processing of information is also occurring via neurons in asynchronous, spiking architectures.

In this work, we report a novel neuromorphic architecture that implements the SAN previously mentioned. This architecture utilizes gated-RRAM (resistive random-access memory) devices for forming associative memory in real time. This work will begin by defining the gated-RRAM device and then show experimental results for such a device fabricated for this work. Next, the device will be modeled within a SPICE simulation. Finally, it will be utilized within a SPICE simulation of an SAN to form associative memories in real time.

2 GATED-RRAM DEVICES

One of the primary decisions in creating neuromorphic hardware is determining how synapses are represented within the architecture. When implementing neuromorphic synapses, one of the primary devices used to implement them is the two-terminal memristor. Originally realized in the late 2000s, the two-terminal memristor demonstrated great promise as a synaptic device due to being able to retain a resistive state after being programmed [17]. Another term for memristors often used in research literature is *resistive random-access memory* (RRAM) [18, 19]. This alternative term for memristors is often used when referring to the basic form of device that relies on the original design and functionality of the two-terminal memristor (e.g., Strukov et al. [17]). More recently, however, the design theory of RRAM has begun to expand. Within the past few years, the two-terminal RRAM concept has become more elaborate [20–23]. By adding a gate terminal onto the RRAM device, one can electrically manipulate the conductive state of the device via the gate instead of the channel terminals. Thanks to this increased capability, gated-RRAM can then be used more seamlessly in certain architectures due to the benefits of programming the device from the gate.

A gated-RRAM device (along with other devices that operate in ahighly similar manner) typically has two channel terminals, like a typical two-terminal RRAM device. The gate for the device is then capacitively coupled to the device's channel (Figure 1). Voltage can then be applied to the

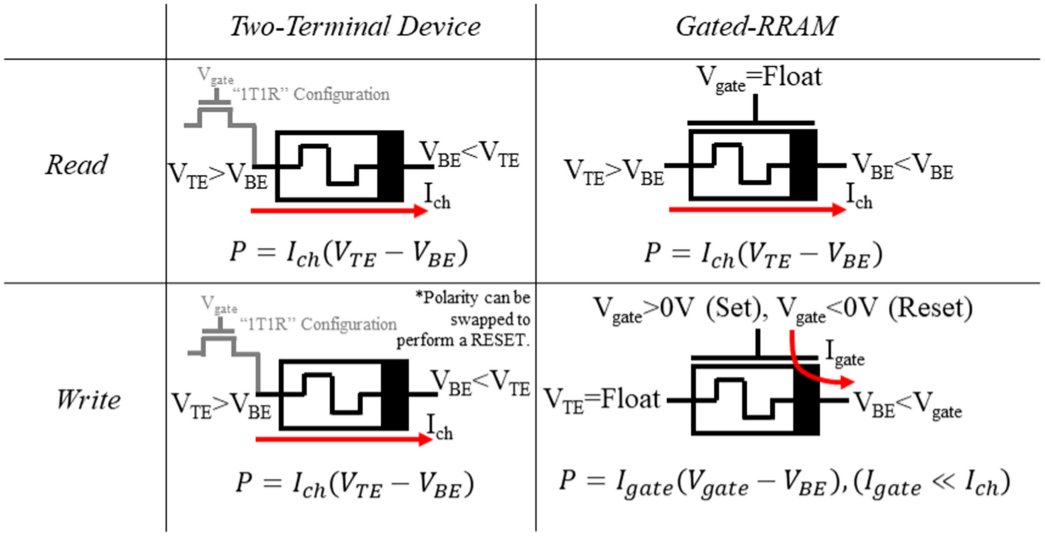

Fig. 2. Chart that depicts the differences between read and write operations in a two-terminal versus gated-RRAM device. Write operations on a two-terminal device can occur in either direction if the voltage drop across the device is sufficiently high, while write operations always occur on the gate of a gated-RRAM device. The write power dissipation of a gated-RRAM device is much more efficient than a two-terminal device due to the current from the gate to either of the other two terminals being negligible [20–23].

gate terminal to manipulate the concentration of defects, vacancies, dopants, and so on, within the device channel in order to adjust the conductance level of the device between the top and bottom electrodes. In many cases, positive bias on the gate increases the conductance of the device, while negative bias on the gate decreases conductance [20–22]. This polarity for potentiation/depression of the device's conductance can be flipped, however, if the device is designed correctly [23].

Various facets of gated-RRAM devices were studied in previous work [20–23]. However, one aspect that has not been studied extensively is the transient decay of a device's conductive state with time once potentiated. This phenomenon is shown in previous device work [22, 23], but it is not sufficiently explored in how it could be utilized. Even though state volatility in a device might initially be considered detrimental, there are some application spaces where it could be useful. Specifically, this feature can be beneficial in forming associative memory within the segmented attractor network discussed in this work.

Gated-RRAM devices provide two critical benefits over typical two-terminal devices typically used within neuromorphic architectures: simultaneous read and write capabilities and more power-efficient write operations. Both benefits are due to the extra terminal on the device and how it is connected to the device channel.

The first benefit that gated-RRAM devices provide is one that appears at the system level. Being able to simultaneously read through the top to bottom electrode and write to the device via the gate allows for the architecture to never worry about swapping between read and write operation modes while processing information. In many two-terminal RRAM architecture implementations, the devices are used in a layout called 1T1R (1-Transistor 1-Resistor/RRAM) [24–26]. This layout is shown in Figure 2. It has the RRAM device placed on either the source or drain of the transistor. The other two terminals of the transistor then act as select lines (e.g., word line and bit line) to access the RRAM for read and write operations. To perform write actions such as SET or RESET, the voltage applied to the transistor gate should be ON and a voltage bias is applied in one of two

directions across the RRAM channel path (see Figure 2). Read operations (also shown in Figure 2) require the gate to be ON while a different read voltage is applied to the RRAM device. In this configuration, the gate terminal of the transistor simply acts as an enable signal to access the two-terminal RRAM device. In order to physically read or write to the RRAM device, both of the other terminals on the cell must be used. If a write operation is being performed using those two terminals, a read cannot be performed at the same time. This same rule can be applied to other methods using two-terminal RRAM devices in which a single operation requires usage of both terminals, leaving no space for the other operation to occur simultaneously [27, 28].

In a gated-RRAM device, the read and write processes occur in a different manner. The gate of the device now acts as the sole write terminal for the device instead of an enable signal. Read operations are always conducted via the two device channel terminals (top and bottom electrodes). Examples of these two processes can be seen in Figure 2. In this setup, the write process is separate from the read process thanks to the extra device terminal. This separation allows both reads and writes to the device to occur in parallel, which is of great benefit to neuromorphic architectures in which asynchronous operations happen constantly within the network, such as SNNs [29, 30].

Since the gate of the RRAM device is separated from the device channel, this naturally leads to the second benefit of the gated-RRAM device: more power-efficient write operations. Typical two-terminal memristors are limited to using the same conduction channel for write operations as reads. In gated-RRAM devices, however, write operations take place via the gate terminal, which can be designed to be electrically isolated from the memristive device channel [22]. This isolation reduces the power consumption of write operations in gated-RRAM by reducing the write current and making it independent of the resistive state of the device, which is not possible in two-terminal devices. The current that does flow from the gate to either device channel terminal is often insignificant enough for previously published gated-RRAM devices to not report them [20, 21, 23] or, in the rare case, they are reported be a small fraction with respect to the device channel current (i.e., one order of magnitude or greater difference) [22]. This means that the write power to a gated-RRAM device relies on I_{gate} and not I_{ch}, making the write operations much more efficient than what is often required in a write operation for a two-terminal device (see Figure 2). If one were to plot the write power consumed in a two-terminal versus a gated device over time, the efficiency of the gated-RRAM becomes apparent (Figure 3). Based on the work of Herrmann et al. [22], one can see that as voltage is applied to the device over time to increase the conductivity between top and bottom electrodes, the current from the gate to the channel remains low. This observation can be translated into comparing write energies between two-terminal and gated-RRAM. This behavior shows that as conductance increases, the amount of energy saved during write operations scales to greater than an order of magnitude in difference between the two devices.

3 GATED-RRAM DEVICE EXPERIMENTAL DEMONSTRATION

The gated-RRAM device fabricated for this work used the basic structure shown in Figure 1 with niobium oxide as the channel material. The device was fabricated by first growing a 100-nm SiO_2 layer via dry oxidation on the surface of a 4" p-silicon wafer. Next, a 50-nm bottom electrode (BE) of titanium nitride (TiN) was deposited via RF sputtering of a Ti target in an Ar/N_2 environment at room temperature. Then, a 20-nm insulator layer of Si_3N_4 was deposited via plasma-enhanced chemical vapor deposition at 250°C. This is then capped with a 40-nm W TE via RF sputtering of a W target in an Ar environment at room temperature. The TE (top electrode)/Insulator/BE is then coated with AZ 1518 photoresist (PR) and spin-coated and patterned with an EVG 620 mask aligner to pattern the TE probe pads and expose the global BE. The W TE is then etched via wet etching with tungsten etchant. Next, the Si_3N_4 layer is etched via reactive ion etching (RIE) with a CF_4:O_2 plasma at low power (75 W). The PR is stripped, and a new pattern is applied to create the channel.

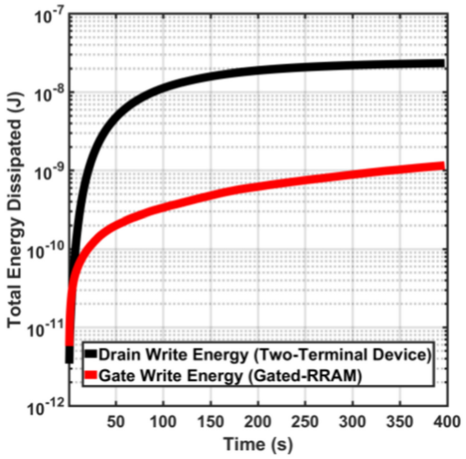

Fig. 3. Plot that depicts a general comparison between write energy consumption of a gated-RRAM and a two-terminal memristive device. The numbers here are extrapolated from current and voltage values reported in Herrmann et al. [22], where a gate voltage of 1V (to program the device) and drain voltage of 1V (to read the device) was used during a write process. These values can change from device to device for both gated-RRAM and two-terminal devices but portrays the general benefit that gated-RRAM provides via its gate write operation.

To etch the channel, the W wet etch and Si_3N_4 dry etch are repeated, followed by an additional RIE process using CF_4:O_2 plasma at high power (300 W). The PR is stripped again and a new deposition of 40 nm of Nb_2O_5 is deposited via DC sputtering of an Nb target in an Ar/O_2 environment at 400°C. Next, a 20-nm Si_3N_4 layer is deposited via RF sputtering of a Si target in an Ar/N_2 environment at room temperature. Finally, a 40-nm W gate electrode is deposited via DC sputtering of a W target in an Ar environment at room temperature. The new stack is patterned to mask the channel region of the device. A final set of W wet etching followed by Si_3N_4 RIE with CF_4:O_2 plasma (75 W) and wet etching of Nb_2O_5 with Ti TFT etchant finishes the pattern. Devices were electrically tested via a Cascade probe station using a Keithley 4200 SCS fitted with a 4225 Pulse Measurement unit. There were variance in the devices; thus, some devices were more conductive, with high-resistance state read currents of around 70 pA and maximum read currents of around 4 mA and some with high-resistance state and maximum read currents of 10 pA and 0.7 mA, respectively. The read current in the high-resistance state for the tested device was approximately 50 pA while the read current in the device's lowest resistance state was almost 1 mA. Figure 4 shows top-view scanning electron microscopy (SEM) of gated-RRAM device arrays with a cross-section schematic diagram showing various layers.

To study the gate-controlled changes in the conductance between TE and BE, write (also referred to as *potentiating*) voltage pulses are applied on gate (G) while TE and BE are grounded. Thereafter, the conductance between TE and BE was measured by applying read bias on TE and grounding the BE. The G was left floating during a read. Figure 5(a) shows read current between the TE and BE (TE-BE I_{max}) immediately after removal of write pulses on the G versus number of write pulses on the G. Figure 5(b) shows current-voltage (I-V) sweep characteristics between the TE and BE by sweeping read bias on the TE between −0.5 V to 0.5 V while grounding the BE after potentiating the device by applying write bias on the G. In Figure 5(a), more than five orders of change in conductance can be observed after potentiating the gate with 2 V after 10,000 pulses, which is remarkable. The same changes in conductance between the TE and BE can be obtained with a

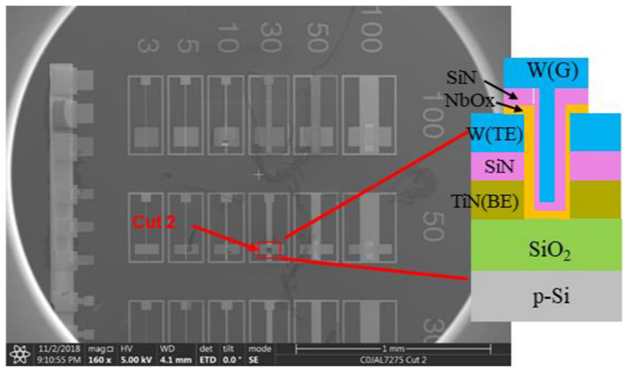

Fig. 4. Top-view scanning electron microcopy (SEM) of gated-RRAM devices and cross-section schematic showing the material stack.

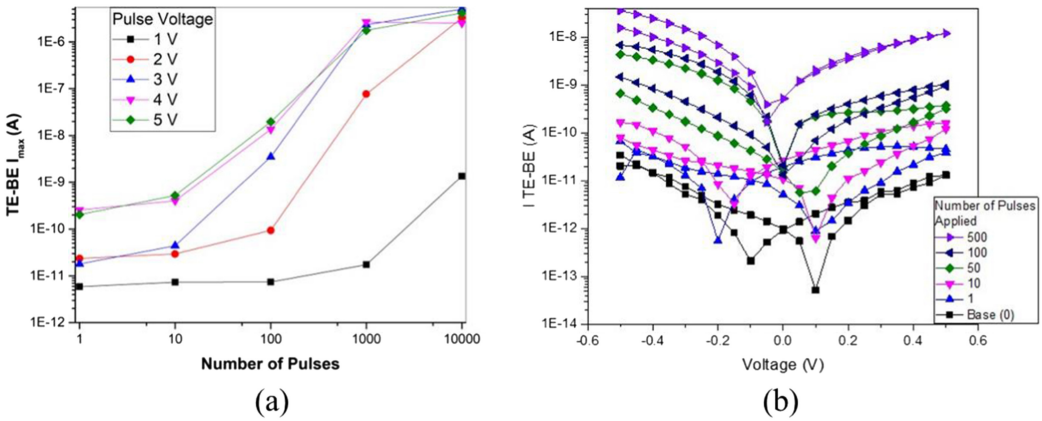

Fig. 5. (a) Read current from top electrode (TE) to bottom electrode (BE) immediately after device is potentiated by applying a certain number of write pulses on gate electrode (G). Legend shows the amplitude write pulses on the gate. The pulses applied to the gate had a width of 1 ms, a period of 1.2 ms, a frequency of 833.3 Hz, and a duty cycle of 83.3%. The read bias was a constant 0.5 V applied to the TE. (b) Dual voltage sweeps as a device is potentiated. The voltage sweep began at −0.5 V, went to 0.5 V, and returned to 0.5 V in 0.05-V steps and was applied to the TE while the BE was grounded and the G was left floating. The voltage sweeps were done after 5-V 1-ms pulses were applied to the gate with the TE and BE grounded.

relatively lower number of potentiating pulses on the G if the write pulse amplitude is increased. The gradual change in conductance with each potentiating pulse (as shown in Figure 5(a)) is beneficial; thus, the potentiation of gated-RRAM devices does not occur via spurious spikes or noise within the programming signal to the device. Only signals that persist for a period of significant time should program the device. The gradual change in conductance could also be utilized in improving the accuracy of neural networks, as slower learning rates (i.e., conductance changes) have been shown to improve a network's accuracy [31]. One can also observe the analog nature of conductance changes between the TE and BE as a function of the number of potentiating pulses on the G, which is another merit of this device. After a change of approximately five orders of magnitude in the device's conductance (with respect to the virgin state for all potentiating conditions), the conductance saturates. This behavior demonstrates a self-limiting nature for these devices that is not seen in many two-terminal devices. Filamentary, two-terminal RRAM devices require a

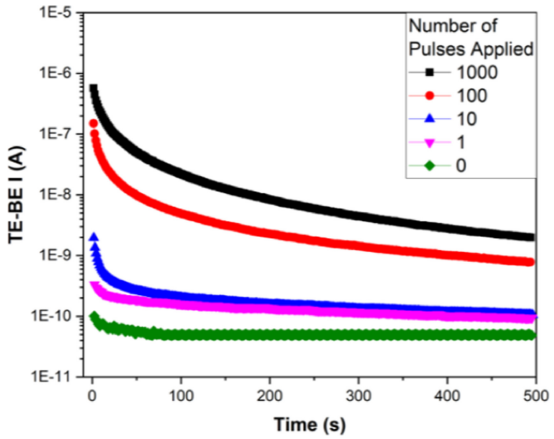

Fig. 6. Top to bottom electrode read current after the device is potentiated by applying write pulses on gate. The write pulses had a width of 1 ms, a period of 1.2 ms, a frequency of 833.3 Hz, and a duty cycle of 83.3%. The read bias was a constant 0.5 V applied to the TE.

reset event (e.g., by choosing the amplitude of applied voltage or pulse-width) or set event (e.g., by choosing the compliance current in 1T1R structures) to achieve conductance modulation. These events in two-terminal devices can be sudden and lead to shorted devices.

Since analog changes in conductive states were observed, it was important to characterize the retention of these states. Figure 6 shows changes in read current (TE-BE I) measured over time after the devices were potentiated by applying different write pulses on the G. Clearly, some decay in state could be observed initially though states were distinct even after 8 minutes. One can observe that retention of state is a function of the number of pulses and amplitude of pulses that was used to potentiate. More conductive states (obtained using a greater number of write pulses or higher amplitude of write pulses) tend to retain longer while intermediate states tend to decay faster.

Understanding the underlying cause of these device-level observations is very important and will require further studies. At this juncture, we propose two hypotheses that can explain these observations. Our first hypothesis involves resorting back to oxygen vacancies (V_o^{2+}) transport in NbOx. When positive bias is applied to the G terminal with the TE and BE grounded, V_o^{2+} can migrate from the NbOx/SiN/G interface to the TE/NbOx and BE/NbOx interfaces. An increase in the concentration of V_o^{2+} at these interfaces can reduce the contact resistivity between TE/NbOx/BE, leading to an increase in the conduction between the TE and BE via NbOx. The decay in conductance can be explained by back diffusion of V_o^{2+} towards the NbOx/SiN/G interface when no potentiating bias is applied on the G [32].

Our second hypothesis is based on V_o^{2+}-based trap states at the TE/NbOx and BE/NbOx interfaces. The charge state of V_o^{2+} based on trapping or de-trapping of electrons can govern the effective doping density at the TE/NbOx and BE/NbOx interfaces [33]. When no bias is applied on the G. then V_o^{2+} at these interfaces trap electrons, leading to a net lower doping concentration and higher contact resistivity. When positive bias is applied to the G terminal with the TE and BE grounded, then V_o^{2+} can de-trap electrons, leading to an increase in effective doping density at these interfaces that results in a net lower contact resistivity. The lowering of contact resistivity governs the increase in the conductance between the TE and BE via NbOx when potentiated with write bias on the G. Note that when gate bias is removed, these V_o^{2+} traps will re-trap electrons based on the trapping constant, which leads to a decay in the conductive state over time (shown in Figure 6). A residual increase in conductive state that does not decay even after 8 minutes can

Table 1. Model Hyperparameter Values Used to Obtain Figures 7 to 9

	Description	Figure 7 Values	Figure 8 Values	Figure 9 Values
g_{min}	Minimum conductance (S)	1e-11	3e-11	2e-10
g_{max}	Maximum conductance (S)	8e-6	8e-8	4e-6
t_{set}	Ideal set time of device (s)	5	2	5
v_t	Gate threshold voltage (V)	0.999	0.999	0.999
b_{rev}	Reverse-bias diode constant	0.00	0.00	0.00
r_{stp}	Rate of short-term decay	5e-3	5e-3	5e-3
g_c	Transient conductance evolution constant	0.95	0.95	0.95
n_{amp}	Depression amplification constant	1	1	1
o_c	Channel voltage influence constant	0	0	0
t_c	Soft/hard threshold control constant	1	1	1
q_{ltp}	Strength of long-term potentiation	0.02	0.02	0.02
r_{ltp}	Rate of long-term decay	1e-7	1e-7	1e-7

be explained by the creation of new V_o^{2+} or generation of other deep traps with higher time constants when the device is stressed with more aggressive write pulses (i.e., higher amplitude and/or higher number of pulses). While further device-level studies are needed to develop a better understanding of these mechanisms, we report that these device-level dynamics are important and need to be modeled in computer-aided-design (CAD) compatible languages that can be utilized to develop diverse neuromorphic networks.

4 GATED-RRAM MODEL DEVELOPMENT

With the experimental verification of the gated-RRAM device, it can have its behavior extracted and placed into a model that is used within a simulation framework. The behavior of this model can be replicated by a behavioral device model specifically designed to encapsulate behaviors observed by gated devices intended for synaptic use (including gated-RRAM devices) [34]. With this behavioral model, many of the device's conductive characteristics can be described by the list of user-defined parameters that capture many higher-level behaviors of the device with respect to time.

The list of user-defined parameters within the model can be found in Table 1. In order to fit the model to the previously demonstrated experimental results of the gated-RRAM device, three different sets of slightly varying parameters can also be found in Table 1. These three sets of parameters will fit to the previously shown Figures 5(a), 5(b), and 6. The fitting process focused on matching as many critical points from each experiment as possible while also attempting to capture the general behavior of each device.

The basic premise of the model described in [34] centers around a few key equations that dictate the device's channel current, conductance, degree of potentiation, and rate of decay at any given time.

The first equation from the model describes the channel current of the device at any given time as a piecewise equation

$$I_{ch} = \begin{cases} \Delta V \geq 0, \; g_{syn}\Delta V \\ \Delta V < 0, \; b_{rev} g_{ch} \Delta V + (1 - b_{rev})\, g_{ch} \left(e^{\Delta V} - 1\right) \end{cases} \tag{1}$$

[34], where ΔV is the voltage difference between the channel nodes (top/bottom electrodes), g_{ch} is the conductance of the channel, and b_{rev} controls whether the behavior of the channel is that

of a linear resistor ($b_{rev} = 1$), a diode ($b_{rev} = 0$), or anything between during reverse bias. The conductance of the channel, g_{ch}, is defined by the equation

$$g_{ch} = \max(1 - 2g_c, 0)\left(g_{range}\left(1 - e^{-px}\right)\right) + (-abs(2g_c - 1) + 1)(g_{range}x + g_{min}) + \max(2g_c - 1, 0)\frac{g_{max}}{1 + e^{-mx+s}} \quad (2)$$

[34], where g_c controls the general shape of the conductance curve with respect to time and other parameters such as s, m, and p are pre-calculated fitting parameters [34] that have their values based off the values given in the parameter list shown in Table 1. The variable x within Equation (2) describes the current conductive state of the device, and can be changed at any timestep by Δx (if the gate voltage is $> v_t$) given by

$$\Delta x = x_{scale}\left(V_{eff} - sign(V_{eff})t_c v_t\right) - d_{stp} \quad (3)$$

[34], where x_{scale} dictates how much x can change by at any given time (dictated by the timestep) and V_{eff} is the effective voltage applied to the gate of the device. The parameters v_t and t_c are the gate threshold voltage and threshold influence constant, respectively. The purpose of the t_c term is to allow the model to distinguish between hard and soft threshold values that can be seen within devices [34].

The variable x also has a term d_{stp} that is removed from x at every timestep of the model simulation. The term d_{stp} represents state decay of the device's conductance and is state based in nature. This term is defined by

$$d_{stp} = r_{stp}t_{set}(x - x_{min})\Delta t \quad (4)$$

[34], where Δt is how much time has passed in the simulation since the device was last updated. The parameters r_{stp} and t_{set} are the degree of short-term decay and expected ideal set time of the device when not considering decay, respectively. The variable x_{min} is the lowest value that x can drop to at any given time. Its default value is zero but can be increased/decreased if the term q_{ltp} is defined as greater than zero.

The first fit to the gated-RRAM device is matching I_{ch} to the number of programming pulses applied to the gate over time at different voltage levels as previously shown in Figure 5(a). Voltage pulses of identical width and period were applied to the device model in order to properly fit the model to the measured results. The results of this fit can be seen in Figure 7. One of the key takeaways from Figure 7 is that the model potentiates much more quickly and consistently than what is shown in the experimental results in the 1 to 10 pulse range. After this earlier range of pulses, the model and results begin to more closely align. The exclusive lack of potentiation when applying 1V potentiation pulses to the device's gate suggests that a gate threshold voltage exists that is close in proximity to the 1V pulse magnitude. In order to implement this behavior, a $v_t = 0.999$V was used for the device.

The second fit to the gated-RRAM device is against the I_{ch} versus V_{TE} curves previously shown in Figure 5(b) (while V_{BE} is grounded). In Figure 8, the second set of parameters from Table 1 can be seen fitting the model to the device's measured results.

The final results being fit to the experimental model is how the device retains its state with respect to time as previously shown in Figure 6. Just as in the previous experiment, the device is programmed via the gate using identical voltage pulses with identical periods. The device then has its channel current observed with respect to time to gauge the rate of conductance decay. Multiple terms within the model play a role in dictating the shape of this curve, including r_{stp} (short-term decay), r_{ltp} (long-term decay), and q_{ltp} (quality of long-term potentiation). The results of the model versus the experiment can be seen in Figure 9. The model does predict slightly faster state decay

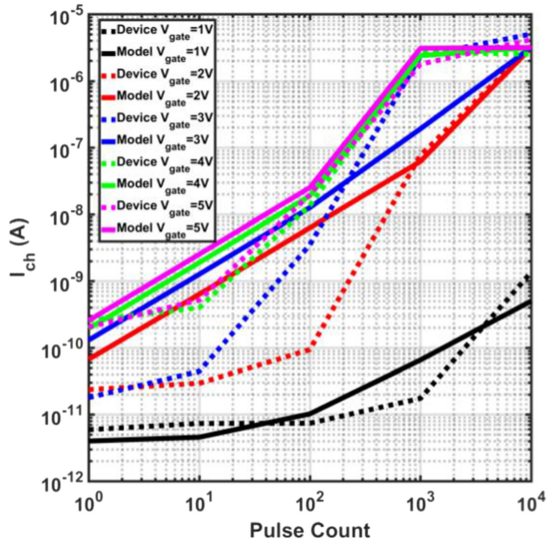

Fig. 7. Fit of the channel current versus pulse count using the model from [34] to the gated-RRAM device demonstrated in Figure 5(a). Logarithmically intermediate pulse count values shown here vary to a degree for mid-range gate voltage values. Later pulse counts do not show this mismatch, however.

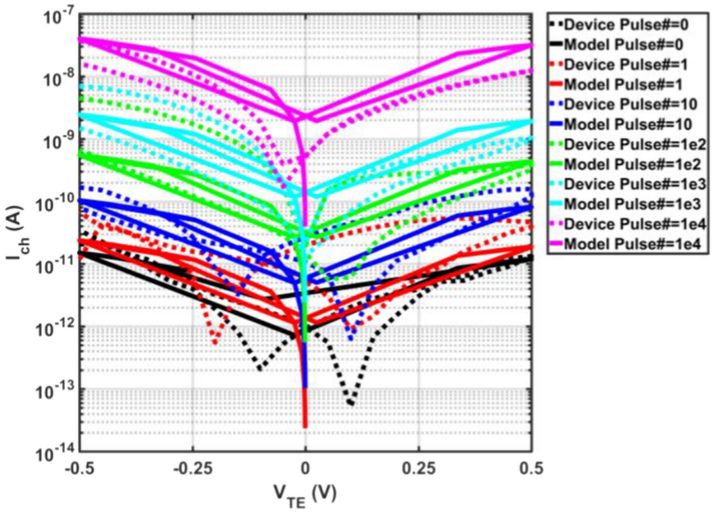

Fig. 8. Fit of the IV sweep at different pulse counts to the device's gate using the model from [34] to the gated-RRAM device demonstrated in Figure 5(b).

than what is seen in the experiment but to a minor degree. The experiment also showed very slight decay of the device's conduction after no pulses were applied, which does not happen in the model.

5 NEUROMORPHIC SAN ARCHITECTURE

A gated-RRAM device such as the one shown here could be useful in a variety of neuromorphic environments. An example of one of these neuromorphic environments would be in the realm of the segmented attractor network as shown in Jones et al. [16]. The segmented attractor network

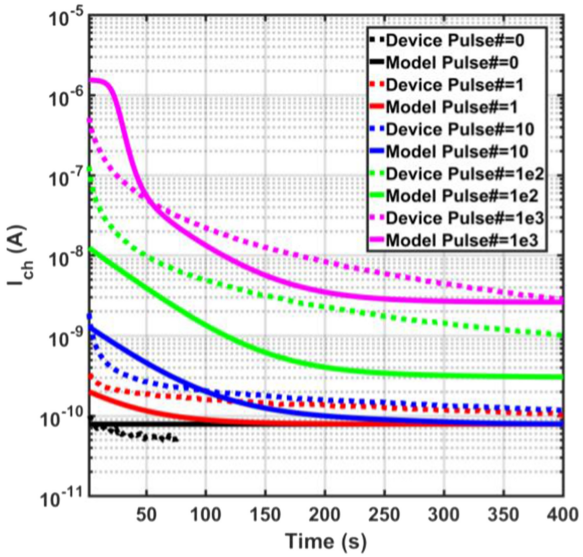

Fig. 9. Fit of the decay over time curves at various pulse counts using the model from [34] to the gated-RRAM device demonstrated in Figure 6. The intermediate pulse count of 100 is off by 1 order of magnitude. However, the model properly captures the other remaining decay rates.

(SAN) was designed to be implemented in a spiking neural network framework that uses recurrent connections along with basic concepts from content-addressable memory in order to form a neuromorphic architecture capable of creating associative memory [16].

The design of the SAN makes it an ideal associative memory architecture for applications in which various forms of categorized input are given to a neural network that need to be associated with one another. The application space for the SAN spans any situation in which finding relationships between categorical data assist in further understanding a problem. One example of such a situation is a system that relies on a sensor suite for navigation purposes to solve a problem such as simultaneous localization and mapping (SLAM) [35]. When solving the SLAM problem, a map of an entity's environment needs to be formed by associating coordinates or location data obtained via sensors with landmark data derived from camera or audio information.

In order to accomplish the task of forming associative memory, the SAN operates using an array of neurons that are segmented into different categorical sets [16]. One set of neurons might be allocated to coordinates, for example, while another set is used for tracking another category of data, such as observed landmarks or time (Figure 10). Within each set of neurons, each neuron is specified to track a certain feature or value of that set. For example, each neuron within the coordinate set could be designated as a certain coordinate value or each neuron for the landmark set could represent a different observed object.

To create an entire SAN in hardware using the gated-RRAM device, an architecture was designed as shown in Figure 11. This demonstration of an architecture consists of an array of six neurons, 24 gated-RRAM devices, and a dozen AND gates. Each neuron also possesses a resistive diode on its external input line. The neuron circuits within the architecture are divided into three distinct sets, where each set possesses two features (or neurons). The SPICE simulations conducted in this work consider these specific device components but not their geometric VLSI layout.

Each neuron receives input for its integration and fire operation from the post-synaptic connections of synapses within the architecture along with its external input line. Each synapse within

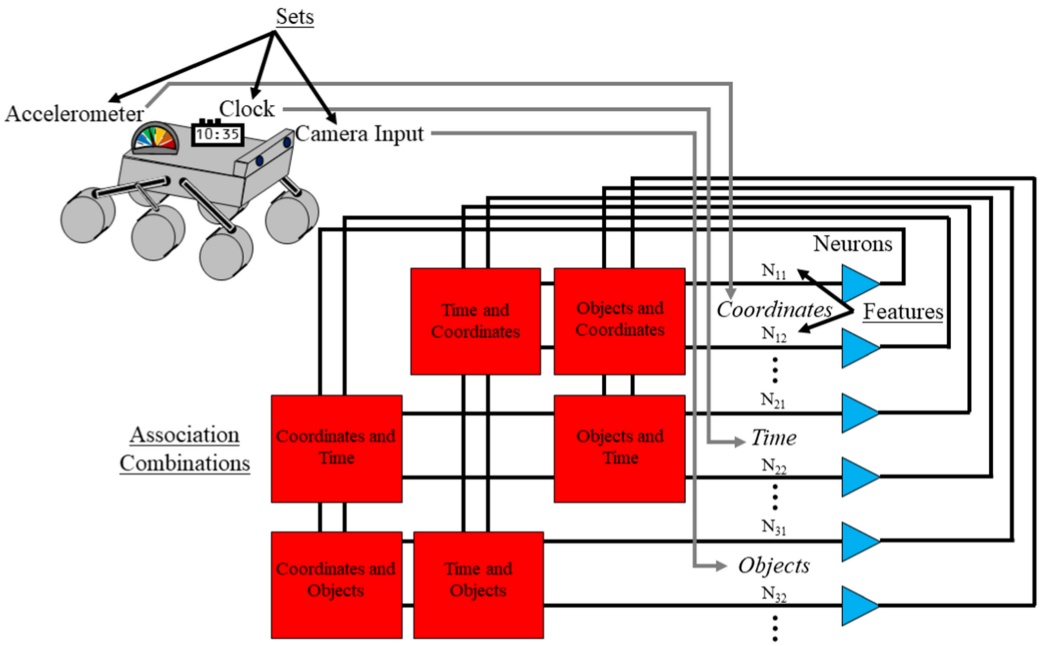

Fig. 10. A schematic diagram of how a SAN is structured within a simple application example of SLAM. Sensor data such as temperature, air pressure, and light intensity can have their values binned and assigned to neurons within the SAN. The network then uses a matrix of associations to create relations between simultaneously occurring values across different sets.

the SAN's synaptic grid is responsible for forming a relationship between two external input values provided to the SAN via the AND gates within the architecture. The AND gates have their output connected to the gate terminal of a gated-RRAM device to program/potentiate the device when both given external inputs are provided simultaneously.

To implement the SAN within a spiking neuromorphic architecture hardware simulation in SPICE, devices intended for synaptic use (such as gated-RRAM devices) are often used in combination with circuits designed to mimic the behavior of biological neurons. Most of these circuits often operate on an "integrate and fire" principle in which the circuit integrates the current provided to the input of the circuit [36, 37]. If the integration operation that the circuit is performing ever crosses a threshold, the circuit emits a voltage spike on its output node and resets its current integration process. One such circuit is the *self-resetting octopus retina neuron circuit* from Jones et al. [35]. This neuron circuit can be constructed using the 180-nm technology node from TSMC within a SPICE simulation framework. A diagram of the neuron circuit can be seen in Figure 12(a). If this circuit is provided $V_{DD} = 1.8V$ and $V_{b1} = V_{b2} = 0.4V$ (bias voltages for the circuit), a frequency and duty cycle profile can be obtained for the circuit based upon the amount of input current provided to its input node. This profile can be seen in Figure 12(b) and will be a useful tool when implementing the gated-RRAM device into a spiking neuromorphic architecture within a SPICE simulation.

Each feature neuron within the SAN receives input from an array of gated-RRAM devices acting as synapses that correspond to the features within every other set within the network. In the example shown in Figure 11, this means that neuron N_{11} receives input from four synapses that relate N_{11} to the four neurons within the other two sets within the network (N_{21}, N_{22}, N_{31}, N_{32}). N_{11} does not receive any input from a synapse that relates it to N_{12} since they are a part of the same set, and the SAN architecture assumes that features within a single set cannot be

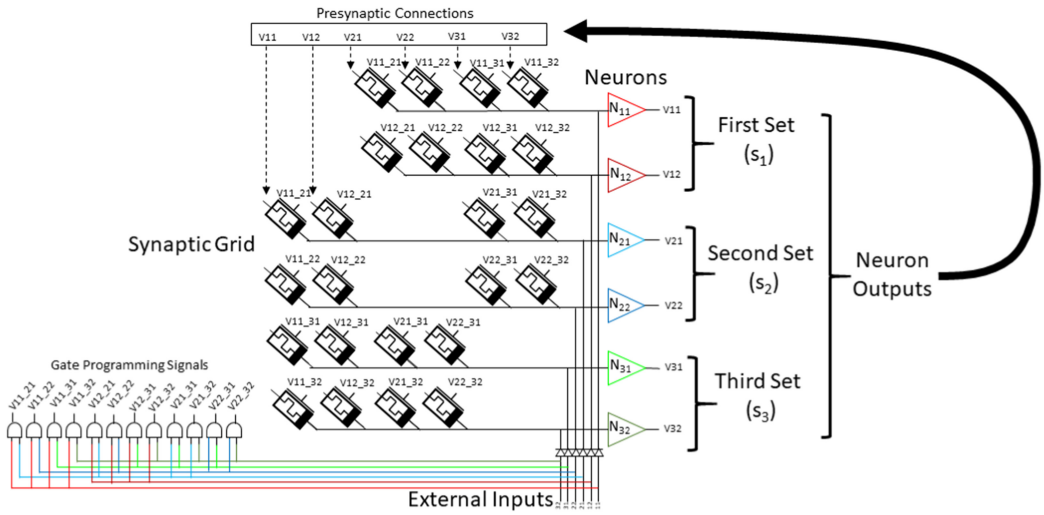

Fig. 11. Diagram depicting an example of the SAN defined in [16] constructed in hardware. All external inputs are routed through resistive diodes to their respective neuron circuits within the architecture. These external inputs are also sent in a combinatorial fashion to an array of digital AND gates to program the gated-RRAM devices acting as synapses within the architecture by applying the output of the AND gates to the gate terminals of each synaptic device. Each AND gate has its output run to two synaptic devices per the labels in the diagram. The neuron outputs are recurrently connected to the pre-synaptic terminals of the synapses (the top electrodes) in the fashion described at the top of the diagram. The network's decisions are determined by measuring the maximum frequency from the output nodes of the neurons within each set.

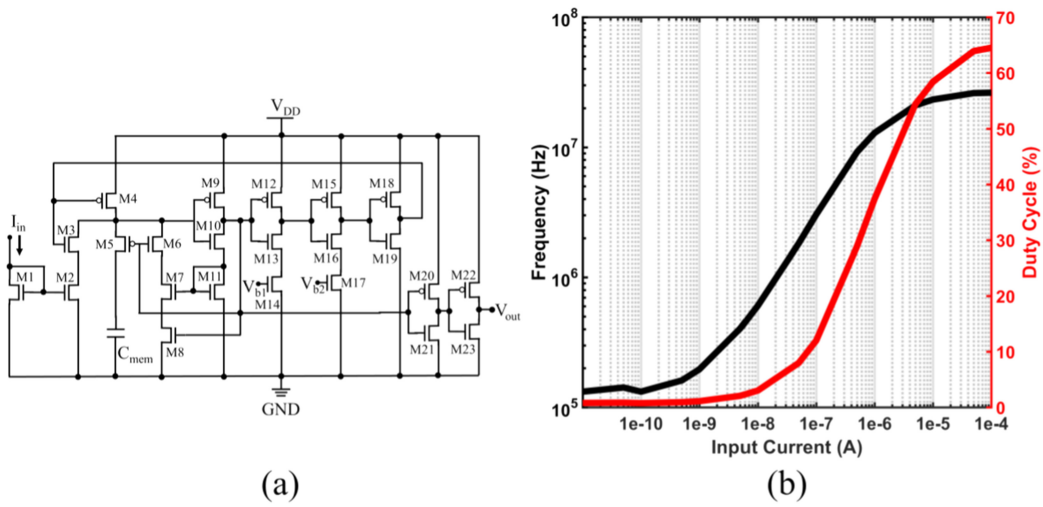

Fig. 12. (a) Schematic of the neuron circuit used. V_{b1} and V_{b2} both were set to 0.4 V for all situations in this work (although other values can be used to manipulate the circuit's behavior and reset speed). (b) Frequency and duty cycle profile of the self-resetting octopus retina neuron defined in [34] when built using the 180-nm TSMC technology node. The neuron will continue to spike at a low-frequency rate even at very low values of current, while at higher values of input current, the frequency eventually saturates.

related to one another [16]. Every other neuron within the network receives a similar sequence of synaptic input from four synapses in the example in Figure 11.

The pre-synaptic connections to each gated-RRAM synapse receive input from the output of the neuron array within the SAN [16]. This connection scheme defines the SAN as a recurrently connected network since it uses its own output as a form of input for analysis. The pre-synaptic connections within the SAN are determined by the external input provided to the gate terminal of the RRAM device that does not already have its corresponding feature neuron connected to its post-synaptic terminal. For example, since the synapse at the top of Figure 11 relating N_{11} and N_{21} has its post-synaptic terminal connected to the input of the N_{11} neuron, the pre-synaptic terminal of that synapse is connected to the output of neuron N_{21}. This form of connection scheme is common among other types of attractor networks [38–40].

The gated-RRAM device is ideal for use in an architecture such as the SAN. Not only will write operations be more power efficient in a gated-RRAM version of this architecture, but the capability of simultaneous read/write operations allows the network to not constantly swap between read and write modes while in use. The recurrent connections within the SAN lend it more towards asynchronous operation when performing analysis on each input set provided. Analyzing data asynchronously means that swapping between read and write modes is not ideal and performing both simultaneously would be optimal (which the gated-RRAM device allows). Simultaneous read/write capability also means that not only can the architecture be used from the typical neural network perspective of training the network via programming a set of associations into its synapses and then testing the recall capability of the network afterwards, but it can also be used for lifelong learning applications. In lifelong learning networks, the network does not go through a strict process of being trained and then being tested but instead learns continuously as it receives input throughout its lifespan [41]. This behavior means that every time the network receives external input to analyze, that occurrence is both a training and evaluation point. If this network were to be implemented using two-terminal technology, the more consistent potentiation of synapses over time (i.e., write operations) would require increased power.

6 RESULTS AND DISCUSSION

To demonstrate the operation of the network, simulation-based studies were conducted in which the network received three "memories" during an association or training phase. The network within the demonstration will utilize the aforementioned TSMC 180-nm neurons and AND gates along with gated-RRAM devices that use the Figure 9 column of hyperparameters from Table 1. After the memories were placed into the network, they were recalled afterward by giving only partial information back to the SAN in the form of external input. This method of recall is useful when trying to identify relationships between pieces of information that the network has previously seen to perform tasks such as pattern completion [42, 43].

The three memories that are placed into the SAN in this example are defined as an array of three features, where each feature is from a unique set. From the neurons in the network in Figure 11, N_{11} and N_{12} belong to the first set, N_{21} and N_{22} belong to the second set, and N_{31} and N_{32} belong to the third set. The three memories will be placed into the network in sequence as follows:

- Memory #1: N_{11}, N_{21}, N_{31}
- Memory #2: N_{12}, N_{22}, N_{32}
- Memory #3: N_{11}, N_{22}, N_{31}

During the association process, external input in the form of a DC voltage is provided to each neuron's input terminal through a diode, which converts the input voltage into a current that the neuron begins to integrate. The external input ensures that a neuron spikes near its maximum

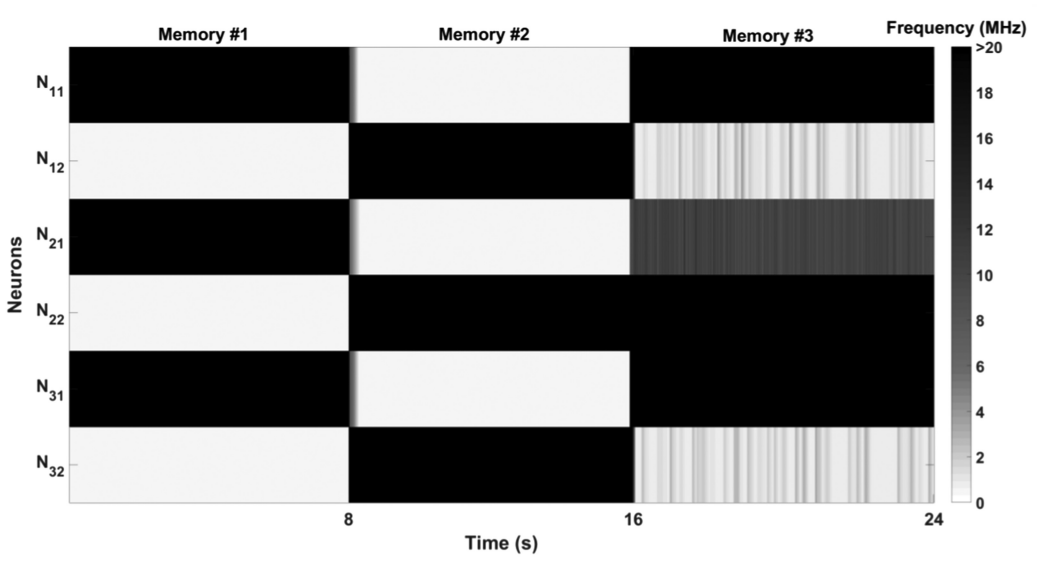

Fig. 13. Running frequency response of the network shown in Figure 11 when having the three previously mentioned memories programmed into its synapses. The slight variances in the timescale are due to how the frequency is calculated during simulation.

frequency (as previously shown in Figure 12) as the output frequency of each neuron is the metric used to determine the network's response. Forcing external input to make neurons fire at maximum frequency ensures that if a neuron within a set is receiving external input, its output in comparison to all other neurons within the set is strongest so that it is always picked as the recalled value from that set. The external stimulus via DC voltage via the AND gate connections to the gate terminals of the synaptic devices potentiates the desired synapses to their high conductance value. Once enough time has passed to allow the synapses to potentiate, the memory has been successfully provided to the network. The next memory can then be presented.

The output response of the SAN architecture during training can be seen in Figure 13, which shows a running frequency measurement of each neuron's output (i.e., the frequency measured in between each spike and the previous spike) versus time. It can be seen throughout Figure 13 that whenever a neuron is receiving external input, it spikes at a very high frequency. When a neuron receives no external input, it spikes at a low frequency or not at all. The exclusion that appears to break this rule in Figure 13 is when the third memory is programmed into the network, where N_{12}, N_{21}, and N_{32} begin to fire at a higher rate. This increase in fire rate is due to the neurons receiving input via their synaptic connections during the third memory. N_{21} is receiving synaptic input from two synapses that previously were potentiated during the first memory (the two synapses that relate it to N_{11} and N_{31}), which makes it spike at a higher rate. N_{12} and N_{32} are receiving synaptic input from one synapse each that associates them with N_{22}. This input causes the neurons to spike but at a lower rate with respect to N_{22} (since N_{22} is getting more input current). Occurrences of this phenomenon increase as more memories are placed into the network as the likelihood for overlapping features between memories increases.

To take a closer look at the evolution of the potentiation of the gated-RRAM devices during training, Figure 14 shows a series of snapshot maps of the gated-RRAM potentiation values as the memories are introduced. One observation that can be made from Figure 14 is the effect of time on

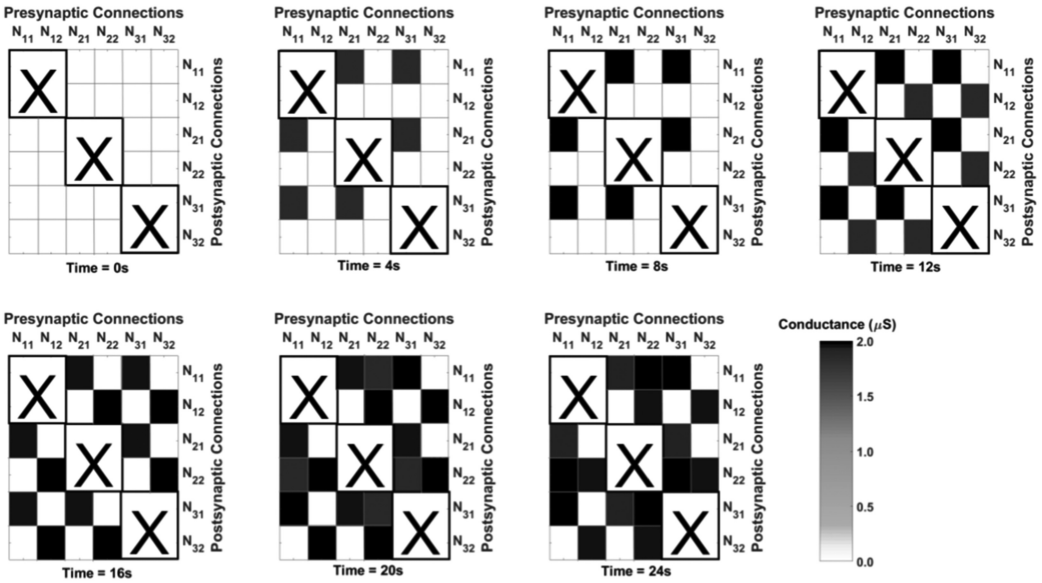

Fig. 14. Snapshot sequence of the conductive states of all gated-RRAM devices within the network as the network is being programmed with memories. The areas covered by crosses are association combinations not made by the synapses (as previously described and shown in Figure 11). As the association phase progresses, certain synapses begin to see decay in their conductive state if they have not received a programming voltage applied to their gate recently.

the potentiation of devices within the network. Once a memory is introduced and external input is removed, the synapses begin to decay due to the state decay behavior seen in the gated-RRAM devices. This might initially appear to be a detriment to the performance of the network in long-term applications. However, if the decay is not too severe for the application space in which it is used, it does not necessarily pose a problem to the network's performance. Also, state decay of the synapses within an application such as lifelong learning can be beneficial instead of detrimental. In a lifelong learning environment, learning never ceases [41]. Networks will always have some physical level of capacity and, given enough time, the network could become saturated with information once it has seen enough information [16]. With the introduction of decay into synapses, it allows the network to slowly forget things it does not normally see over time, making space for new potentially more important memories to be introduced in the future. This phenomenon additionally reduces the possibility of false associations being made by the network if too many memories were to be introduced to the network over time.

If recall is performed on the network after the three memories have been introduced, the network attempts to complete the memories it has previously seen given the input it is provided. Figure 15 shows such a recall process in which each neuron within the SAN is given external input in a sequential process (starting at N_{11} and ending at N_{32}). As each external input is provided, the network responds in a specific manner to the memories it has previously seen. The neuron with the highest average running frequency among each set during the period that each external input is provided can be determined to be the recalled feature from each set to form a completely recalled memory. Table 2 shows these results, with the dominant frequencies within each set during each external input provided to the network in bold print. Those dominant frequencies are declared as the recalled features and form the memories previously shown to the network. The

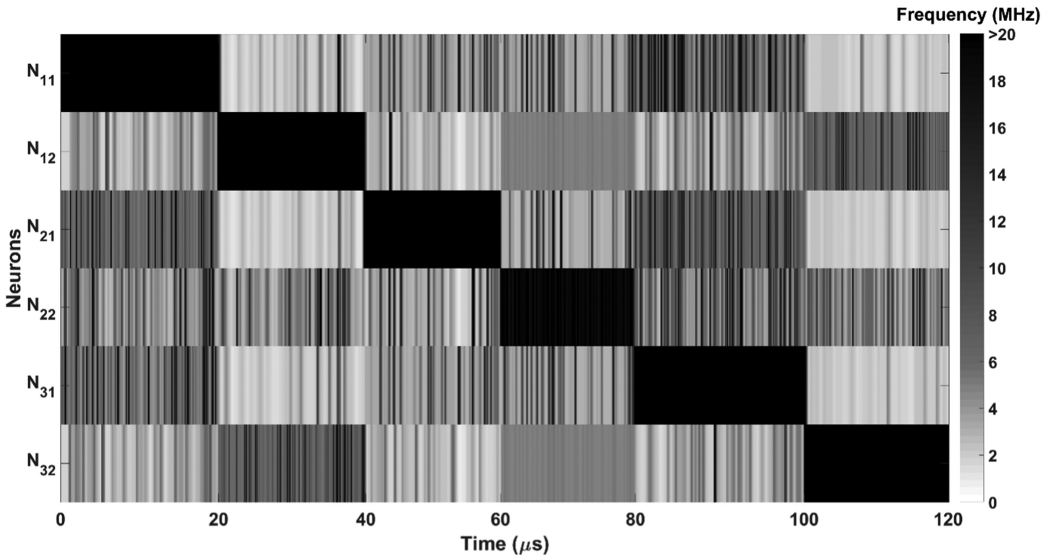

Fig. 15. Running frequency response of the network shown in Figure 11 during recall when programmed with the three memories shown in Figure 13. The slight variances in the timescale are due to how the running frequency is calculated during simulation.

Table 2. Recall Results of the Segmented Attractor Network

External Input			Recalled Frequencies (MHz)						Recalled Memory				Pwr
s_1	s_2	s_3	N_{11}	N_{12}	N_{21}	N_{22}	N_{31}	N_{32}	s_1	s_2	s_3	#	(μW)
N_{11}	-	-	**26.8**	3.65	**8.36**	6.21	**8.12**	3.67	N_{11}	N_{21}	N_{31}	*1*	688
N_{12}	-	-	3.23	**29.2**	2.24	**6.27**	2.69	**8.88**	N_{12}	N_{22}	N_{32}	*2*	645
-	N_{21}	-	**5.65**	3.35	**28.0**	5.15	**5.51**	3.07	N_{11}	N_{21}	N_{31}	*1*	627
-	N_{22}	-	**5.69**	5.01	5.26	**23.0**	**5.53**	5.00	N_{11}	N_{22}	N_{31}	*3*	615
-	-	N_{31}	**9.94**	4.30	**9.38**	8.22	**27.7**	3.99	N_{11}	N_{21}	N_{31}	*1*	755
-	-	N_{32}	2.26	**8.49**	2.15	**6.08**	2.62	**28.8**	N_{12}	N_{22}	N_{32}	*2*	624

number of each memory (as previously defined) is shown on the right side of Table 2. Also on the right side of the table is the network's average power consumption during each recall point provided to the network. This power consumption includes both V_{DD} and the bias voltages provided to the neurons (i.e., V_{b1} and V_{b2}). However, the power consumption of the bias voltage within the network is so small that it is negligible in the results presented.

With respect to the decay rate in the gated-RRAM, the read period to perform recall shown in Figure 15 is short. The value during this period is virtually unchanged from start to finish of the operation. The association period shown in Figure 13, however, occurs over a much longer period. As can be seen in the progression of frames in Figure 14, state decay of synapses occurs between training points during association. If the decay rate of synapses was increased, recall after the association phase would be altered. To increase decay in the network, r_{stp} was increased from 5e-3 (initial value) to 1e-2 to demonstrate how state decay affects recall. The tabularized results from this decay simulation can be seen in Table 3, which shows how as decay increases, the network begins to forget memories it hasnot seen in a while (such as the first memory).

Forgetting memories for the network does not happen instantaneously since decay of each gated-RRAM device's conductance happens gradually over time. As shown in Table 3, the first

Table 3. Recall Results of the Segmented Attractor Network with Higher Decay

External Input			Recalled Frequencies (MHz)						Recalled Memory				Pwr
s_1	s_2	s_3	N_{11}	N_{12}	N_{21}	N_{22}	N_{31}	N_{32}	s_1	s_2	s_3	#	(μW)
N_{11}	-	-	**27.9**	3.78	6.12	**7.72**	**10.0**	3.76	N_{11}	N_{22}	N_{31}	*3*	713
N_{12}	-	-	2.67	**29.2**	1.25	**7.02**	2.60	**9.49**	N_{12}	N_{22}	N_{32}	*2*	642
-	N_{21}	-	**3.77**	2.99	**28.9**	7.18	**3.79**	1.79	N_{11}	N_{21}	N_{31}	*1*	604
-	N_{22}	-	4.48	**4.92**	4.28	**22.0**	4.48	**4.88**	N_{12}	N_{22}	N_{32}	*2*	570
-	-	N_{31}	**9.70**	3.84	5.79	**7.60**	**27.7**	3.97	N_{11}	N_{22}	N_{31}	*3*	706
-	-	N_{32}	2.90	**9.30**	1.88	**6.94**	3.39	**29.1**	N_{12}	N_{22}	N_{32}	*2*	655

Table 4. Average Power Consumption Comparison to Other Associative Memory Architectures

	Category	Architecture	CMOS Node	Hit Power Consumption (Per Item)	Miss Power Consumption (Per Item)
[44]	Traditional	Analog HAM	45 nm	5.56 mW	5.56 mW
[45]	Traditional	Resistive Ternary CAM	45 nm	6.94 μW	2.61 mW
This Work	Neural Network	Gated-RRAM SNN	180 nm	109 μW	109 μW

memory is not completely forgotten when the decay rate of the devices in the network is increased; instead, it is recalled in fewer cases. In addition to the first memory being recalled less, N_{22} in Table 3 recalls the second memory instead of the third (as was recalled in Table 2). This answer was given by the network once decay was increased due to other synapses associated with the first memory feeding N_{11} and N_{31} being more decayed than before, hindering the overall feedback provided to them. Previously, the higher feedback to N_{11} and N_{31} caused the network to recall the third memory, but since the feedback to these neurons is overall lower for the results in Table 3, N_{12} and N_{32} have superior frequency output.

As the network evolves over time, memories will slowly rotate into and out of the network based upon what the network observes. The rate at which memories fade from the network can be controlled via the decay rate of each device's conductive state, which can be tuned via device engineering (e.g., introduction of extra defects/dopants during fabrication, selecting different channel bulk materials, etc.).

When comparing the average power consumption of the simulated architecture to other contemporary associative memory implementations that also use memristive devices, the spiking neural network implementation demonstrated here is superior. Table 4 shows the average power consumption per item across each implementation for hits and misses. Imani et al. [44] compare a few different associative memory approaches, of which the most optimal was an analog, hyperdimensional associative memory (HAM). The power consumption values reported do not differentiate between hits and misses. The architecture divides its bits into classes (i.e., items). The other architecture proposed by Imani et al. [45] does have a large power consumptive difference between a hit and miss. This architecture, known as *ternary content addressable memory* (CAM), defines items as lines that are selectively activated to recall memories. The architecture shown in this work represents each item as a neuron, and the power consumption does not vary between hits and misses due to the innate behavior of a neural network. Given the more consistent power consumptive behavior of the SNN demonstrated here and that it could be further improved by using a smaller complementary metal-oxide semiconductor (CMOS) node, it has clear advantages over other memristive designs.

7 CONCLUSIONS

In this work, we report an SAN neuromorphic architecture that can be used for forming dense associative memory between various inputs. Designed with gated-RRAM devices as synapses and CMOS-based spiking neurons, the simulated network was able to learn and recall memories via pattern completion. It was shown how an often deemed undesirable feature of the gated-RRAM device, decay of its conductive state, could prove beneficial in the application of associative memory within an SAN. In tandem with CMOS-based neuron circuits, a gated-RRAM device provides an extremely promising system for future spiking neuromorphic processing architectures. Within the devices studied, further research is needed to understand the details of underlying mechanisms and interplay between trapping/de-trapping, diffusion of defects, and transport mechanisms. Furthermore, extensive study should be done on device-to-device and cycle-to-cycle variability and how they affect the performance of the demonstrated associative memory architecture. Finally, the architecture studied here should be scaled to a larger size and compared with other contemporary solutions for associative memory in neuromorphic architectures.

REFERENCES

[1] Lucas Antón Pastur-Romay, Francisco Cedrón, Alejandro Pazos, and Ana Belén Porto-Pazos. 2016. Deep artificial neural networks and neuromorphic chips for big data analysis: Pharmaceutical and bioinformatics applications. *Int. J. Mol. Sci* 17, 8 (2016), 1313. DOI: https://doi.org/10.3390/ijms17081313

[2] Alexander Neckar, Sam Fok, Ben V. Benjamin, Terrence C. Stewart, Nick N. Oza, Aaron R. Voelker, Chris Eliasmith, Rajit Manohar, and Kwabena Boahen. 2019. Braindrop: A mixed-signal neuromorphic architecture with a dynamical systems-based programming model. *Proceedings of the IEEE* 107, 1 (2019), 144–164. DOI: https://doi.org/10.1109/JPROC.2018.2881432

[3] Chit-Kwan Lin, Andreas Wild, Gautham N. Chinya, Tsung-Han Lin, Mike Davies, and Hong Wang. 2018. Mapping spiking neural networks onto a manycore neuromorphic architecture. *ACM SIGPLAN Notices* 53, 4 (2018), 78–89. DOI: https://doi.org/10.1145/3296979.3192371

[4] Jianshi Tang, Fang Yuan, Xinke Shen, Zhongrui Wang, Mingyi Rao, Yuanyuan He, Yuhao Sun, Xinyi Li, Wenbin Zhang, Yijun Li, Bin Gao, He Qian, Guoqiang Bi, Sen Song, J. Joshua Yang, and Huaqiang Wu. 2019. Bridging biological and artificial neural networks with emerging neuromorphic devices: Fundamentals, progress, and challenges. *Adv. Mat* 31, 49 (2019). DOI: https://doi.org/10.1002/adma.201902761

[5] Catherine D. Schuman, Thomas E. Potok, Robert M. Patton, J. Douglas Birdwell, Mark E. Dean, Garrett S. Rose, and James S. Plank. 2017. A survey of neuromorphic computing and neural networks in hardware. arXiv:1705.06963. Retrieved December 14, 2021 from https://arxiv.org/abs/1705.06963.

[6] Mohammad Reza Mohammadi, Sayed Alireza Sadrossadat, Mir Gholamreza Mortazavi, and Behzad Nouri. 2017. A brief review over neural network modeling techniques. In *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI'17)*. 54–57. DOI: https://doi.org/10.1109/ICPCSI.2017.8391781

[7] Laura E. Matzen, Michael C. Trumbo, Ryan C. Leach, and Eric D. Leshikar. 2015. Effects of non-invasive brain stimulation on associative memory. *Brain Research* 1624 (2015), 286–296. DOI: https://doi.org/10.1016/j.brainres.2015.07.036

[8] Renée Baillargeon, Elizabeth S. Spelke, and Stanley Wasserman. 1985. Object permanence in five-month-old infants. *Cognition* 20, 3 (1985), 191–208. DOI: https://doi.org/10.1016/0010-0277(85)90008-3

[9] Telajala Venkata Mahendra, Sandeep Mishra, and Anup Dandapat. 2017. Self-controlled high-performance precharge-free content-addressable memory. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 25, 8 (2017). DOI: https://doi.org/10.1109/TVLSI.2017.2685427

[10] J. J. Hopfield. 1982. Neural networks and physical systems with emergent collective computational abilities. In *Proc. Natl. Acad. Sci. USA* 79 (1982). 2554–2558. DOI: https://doi.org/10.1073/pnas.79.8.2554

[11] V. Milo, D. Ielmini, and E. Chicca. 2017. Attractor networks and associative memories with STDP learning in RRAM synapses. In *IEEE International Electron Devices Meeting (IEDM'17)*. IEEE, San Francisco, CA. DOI: https://doi.org/10.1109/IEDM.2017.8268369

[12] S. G. Hu, Y. Liu, Z. Liu, T. P. Chen, J. J. Wang, Q. Yu, L. J. Deng, Y. Yin, and Sumio Hosaka. 2015. Associative memory realized by a reconfigurable memristive Hopfield neural network. *Nature Communications* 6, 7522 (2015). DOI: https://doi.org/10.1038/ncomms8522

[13] Jiu Yang, Lidan Wang, Yan Wang, and Tengteng Guo. 2017. A novel memristive Hopfield neural network with application in associative memory. *Neurocomputing* 277 (2017), 142–148. DOI: https://doi.org/10.1016/j.neucom.2016.07.065

[14] N. Davey and S. P. Hunt. 1999. The capacity and attractor basins of associative memory models. In *International Work-Conference on Artificial Neural Networks*. Alicante, Spain. DOI: https://doi.org/10.1007/BFb0098189

[15] E. Gardner. 1987. Maximum storage capacity in neural networks. *Europhys. Lett* 4, 4 (1987), 481–485. Retrieved December 14, 2021 from https://iopscience.iop.org/article/10.1209/0295-5075/4/4/016.

[16] Alexander Jones, Rashmi Jha, Ajey P. Jacob, and Cory Merkel. 2019. A segmented attractor network for neuromorphic associative learning. In *Proceedings of the International Conference on Neuromorphic Systems (ICONS'19)*. ACM, Knoxville, TN, 1–8. DOI: https://doi.org/10.1145/3354265.3354284

[17] Dmitri B. Strukov, Gregory S. Snider, Duncan R. Stewart, and R. Stanley Williams. 2008. The missing memristor found. *Nature* 453 (2008), 80–83. DOI: https://doi.org/10.1038/nature06932

[18] Wenchao Lu, Wenbo Chen, Yibo Li, and Rashmi Jha. 2016. Self current limiting MgO ReRAM devices for low-power non-volatile memory applications. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 6, 2 (2016), 163–170. DOI: https://doi.org/10.1109/JETCAS.2016.2547758

[19] Sung Hyun Jo, Tanmay Kumar, Sundar Narayanan, and Hagop Nazarian. 2015. Cross-point resistive RAM based on field-assisted superlinear threshold selector. *IEEE Transactions on Electron Devices* 62, 11 (2015), 3477–3481. DOI: https://doi.org/10.1109/TED.2015.2426717

[20] Jianshi Tang, Douglas Bishop, Seyoung Kim, Matt Copel, Tayfun Gokmen, Teodor Todorov, SangHoon Shin, Ko-Tao Lee, Paul Solomon, Kevin Chan, Wilfried Haensch, and John Rozen. 2018. ECRAM as scalable synaptic cell for high-speed, low-power neuromorphic computing. In *IEEE International Electron Devices Meeting (IEDM'18)*. IEEE, San Francisco, CA, 13.1.1–13.1.4. DOI: https://doi.org/10.1109/IEDM.2018.8614551

[21] Suhwan Lim, Jong-Ho Bae, Jai-Ho Eum, Sungtae Lee, Chul-Heung Kim, Dongseok Kwon, and Jong-Ho Lee. 2018. Hardware-based neural networks using a gated-Schottky diode as a synapse device. In *IEEE International Symposium on Circuits and Systems (ISCAS'18)*. IEEE, Florence, Italy, 1–5. DOI: https://doi.org/10.1109/ISCAS.2018.8351152

[22] Eric Herrmann, Andrew Rush, Tony Bailey, and Rashmi Jha. 2018. Gate controlled three-terminal metal oxide memristor. *IEEE Electron Device Letters* 39, 4 (2018), 500–503. DOI: https://doi.org/10.1109/LED.2018.2806188

[23] Yoeri van de Burgt, Ewout Lubberman, Elliot J. Fuller, Scott T. Keene, Grégorio C. Faria, Sapan Agarwal, Matthew J. Marinella, A. Alec Talin, and Alberto Salleo. 2017. A non-volatile organic electrochemical device as a low-voltage artificial synapse for neuromorphic computing. *Nature Materials* 16, 4 (2017), 414–418. DOI: https://doi.org/10.1038/nmat4856

[24] Tzu-Ying Lin, Yong-Xiao Chen, Jin-Fu Li, Chih-Yen Lo, Ding-Ming Kwai, and Yung-Fa Chou. 2016. A test method for finding boundary currents of 1T1R memristor memories. In *IEEE 25th Asian Test Symposium (ATS'16)*. Hiroshima, Japan, 281–286. DOI: https://doi.org/10.1109/ATS.2016.44

[25] Emmanuelle J. Merced-Grafals, Noraica Dávila, Ning Ge, R. Stanley Williams, and John Paul Strachan. 2016. Repeatable, accurate, and high speed multi-level programming of memristor 1T1R arrays for power efficient analog computing applications. *Nanotechnology* 27, 36 (2016), 1–9. DOI: https://doi.org/10.1088/0957-4484/27/36/365202

[26] Mahmoud Zangeneh and Ajay Joshi. 2014. Design and optimization of nonvolatile multibit 1T1R resistive RAM. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 22, 8 (2014), 1815–1828. DOI: https://doi.org/10.1109/TVLSI.2013.2277715

[27] Max M. Shulaker, Tony F. Wu, Asish Pal, Liang Zhao, Yoshio Nishi, Krishna Saraswat, H.-S. Philip Wong, and Subhasish Mitra. 2014. Monolithic 3D integration of logic and memory: Carbon nanotube FETs, resistive RAM, and silicon FETs. In *IEEE International Electron Devices Meeting (IEDM'14)*. IEEE, San Francisco, CA, 27.4.1–27.4.4. DOI: https://doi.org/10.1109/IEDM.2014.7047120

[28] Georgios Papandroulidakis, Ioannis Vourkas, Angel Abusleme, Georgios Ch. Sirakoulis, and Antonio Rubio. 2017. Crossbar-based memristive logic-in-memory architecture. *IEEE Transactions on Nanotechnology* 16, 3 (2017), 491–501. DOI: https://doi.org/10.1109/TNANO.2017.2691713

[29] Ming Zhang, Zonghua Gu, Nenggan Zheng, De Ma, and Gang Pan. 2020. Efficient spiking neural networks with logarithmic temporal coding. *IEEE Access* 8 (2020), 98156–98167. DOI: https://doi.org/10.1109/ACCESS.2020.2994360

[30] Amirhossein Tavanei, Masoud Ghodrati, Saeed Reza Kheradpisheh, Timothée Masquelier, and Anthony Maida. 2019. Deep learning in spiking neural networks. *Neural Networks* 111 (2019), 47–63. DOI: https://doi.org/10.1016/j.neunet.2018.12.002

[31] D. R. Wilson and T. R. Martinez. 2001. The need for small learning rates on large problems. *IEEE International Joint Conference on Neural Networks (IJCNN'01)*. Washington, DC. DOI: https://doi.org/10.1109/IJCNN.2001.939002

[32] Tony J. Bailey and Rashmi Jha. 2018. Understanding synaptic mechanisms in SrTiO3 devices. *IEEE Transactions on Electron Devices* 65, 8 (2018). 3514–3520. DOI: https://doi.org/10.1109/TED.2018.2847413

[33] Branden Long, Jorhan Ordosgoitti, Rashmi Jha, and Christopher Melkonian. 2011. Understanding the charge transport mechanism in VRS and BRS states of transition metal oxide nanoelectronic memristor devices. *IEEE Transactions on Electron Devices* 58, 11 (2011), 3912–3919. DOI: https://doi.org/10.1109/TED.2011.2165845

[34] Alexander Jones and Rashmi Jha. 2020. A compact gated-synapse model for neuromorphic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, (2020). DOI: https://doi.org/10.1109/TCAD.2020.3028534

[35] Alexander Jones, Andrew Rush, Cory Merkel, Eric Herrmann, Ajey P. Jacob, Clare Thiem, and Rashmi Jha. 2020. A neuromorphic SLAM architecture using gated-memristive synapses. *Neurocomputing* 381 (2020), 89–104. DOI: https://doi.org/10.1016/j.neucom.2019.09.098

[36] Carver Mead. 1989. *Analog VLSI and Neural Systems* (1st ed.). Addison-Wesley, Reading, MA.

[37] Giacomo Indiveri, Bernabé Linares-Barranco, Tara Julia Hamilton, André van Schaik, Ralph Etienne-Cummings, Tobi Delbruck, Shih-Chii Liu, Piotr Dudek, Philipp Häfliger, Sylvie Renaud, Johannes Schemmel, Gert Cauwenberghs, John Arthur, Kai Hynna, Fopefolu Folowosele, Sylvain Saighi, Teresa Serrano-Gotarredona, Jayawan Wijekoon, Yingxue Wang, and Kwabena Boahen. 2011. Neuromorphic silicon neuron circuits. *Front. Neurosci.* 5, 73 (2011). DOI: https://doi.org/10.3389/fnins.2011.00073

[38] Yanghao Wang, Liutao Yu, Si Wu, Ru Huang, and Yuchao Yang. 2020. Memristor-based biologically plausible memory based on discrete and continuous attractor networks for neuromorphic systems. *Advanced Intelligent Systems* 2, 3 (2020), 1–7. DOI: https://doi.org/10.1002/aisy.202000001

[39] V. Milo, D. Ielmini, and E. Chicca. 2017. Attractor networks and associative memories with STDP learning in RRAM synapses. In *IEEE International Electron Devices Meeting (IEDM'17)*. IEEE, San Francisco, CA, 11.2.1–11.2.4. DOI: https://doi.org/10.1109/IEDM.2017.8268369

[40] S. G. Hu, Y. Liu, Z. Liu, T. P. Chen, J. J. Wang, Q. Yu, L. J. Deng, Y. Yin, and Sumio Hosaka. 2015. Associative memory realized by a reconfigurable memristive Hopfield neural network. *Nature Communications* 6, 7522 (2015), 1–8. DOI: https://doi.org/10.1038/ncomms8522

[41] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. 2019. Continual lifelong learning with neural networks: A review. *Neural Networks* 113 (2019), 54–71. DOI: https://doi.org/10.1016/j.neunet.2019.01.012

[42] Carina Curto and Katherine Morrison. 2016. Pattern completion in symmetric threshold-linear networks. *Neural Computation* 28, 12 (2016), 2825–2852. DOI: https://doi.org/10.1162/NECO_a_00869

[43] Segundo Jose Guzman, Alois Schlögl, Michael Frotscher, and Peter Jonas. 2016. Synaptic mechanisms of pattern completion in the hippocampal CA3 network. *Science* 353, 6304 (2016), 1117–1123. DOI: https://doi.org/10.1126/science.aaf1836

[44] Mohsen Imani, Abbas Rahimi, Deqian Kong, Tajana Rosing, and Jan M. Rabaey. 2017. Exploring hyperdimensional associative memory. *IEEE International Symposium on High Performance Computer Architecture (HPCA'17)*. IEEE, Austin, TX. DOI: https://doi.org/10.1109/HPCA.2017.28

[45] Mohsen Imani, Abbas Rahimi, Pietro Mercati, and Tajana Simunic Rosing. 2018. Multi-stage tunable approximate search in resistive associative memory. *IEEE Transactions on Multi-Scale Computing Systems* 4, 1 (2018), 17–29. DOI: https://doi.org/10.1109/TMSCS.2017.2665462

Received August 2020; revised March 2021; accepted April 2021