Memristors in Neural Networks

Robert Lee Murrer III
College of Science and Engineering
University of West Florida
Pensacola, USA
robert.murrer@students.uwf.edu

***Abstract*—A comprehensive review of the history and functionality of memristor devices is presented. The application of this element to neuromorphic computing and neural networks is explained. The memristor is not only a memory device that can be used to store weights of neural networks, but it can also be used for logic as well. Fabrication techniques and a brief introduction on how training is performed is also included.**

Keywords—VLSI, CMOS, Memristor, Artificial Neural Networks, Large Language Models

I. Introduction

As feature sizes of CMOS technologies keep shrinking, the end of Moore's Law has been predicted and a search for ways to increase compute density has been met with the limits of physics itself. [1] In order to increase storage and computation power of integrated circuits new methods are being explored. GPU based computation has exploded in popularity lately and device RAM requirements are becoming larger and larger. With the latest trends of GPU based Machine Learning algorithms using billions of parameters which requires huge amounts of RAM to store the weights, the memristor could be a solution to the storage of these weights for deep neural networks.

A. Memristors

Originally formalized in 1971 by Leon Chua, the Memristor is described as the missing circuit element. [2] A contraction of memory and resistor the memristor allows a variable resistance to be stored in the device and stored even after power is lost. Although in the original paper it was not yet possible to fabricate such a device without an internal power source, in 2008 HP Labs was able to successfully create hardware memristors. [3]

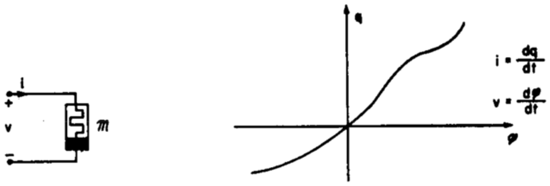


Fig. 1. Memristor symbol and Voltage/Current relationship Chua's 1971 Paper [2]

HP Labs was able to fabricate a metal/oxide/metal thin-film device with platinum electrodes and TiO2 dielectric that make up the memristor stack-up. The oxide is split into two layers with process steps to drive oxygen out of one of them to provide the switching behavior to the memristor. The states of the memristor are Low Resistance State (LRS) and High Resistance State (RHS). Through charging the device for certain periods of time through pulse width modulation, these values of resistance for these two states can be set. Since the memristor is non-volatile, it will maintain that state even with no power provided to the device.

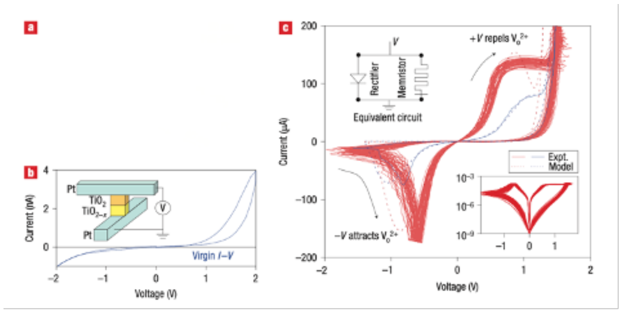


Fig. 2. Memristor cross section and measurement of hardware from [3]

B. Artificial Neural Networks

A small introduction to Machine Learning and Artificial Neural Networks (ANNs) is included to provide context for the applications of memristors. ANNs are modelled after a simplified version of neurons arranged in a graph. Each neuron acts as a gate that can either impede or pass through a signal from other neurons. The combination of the inputs to a neuron are summed and according to its activation function it will pass the signal to the next layer in the network. [4]

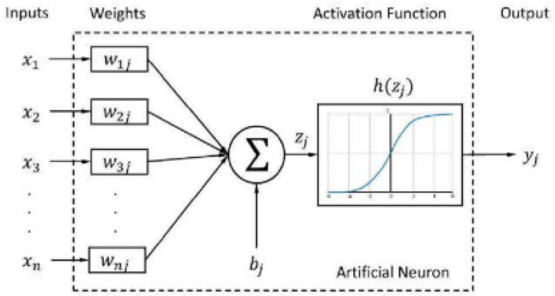


Fig. 3. A single neuron with several weights shown being summed [4]

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

The neurons are separated in three layers input, output, and hidden layers. The input to the network could be for instance a pixel value and the entire input layer would represent every pixel in the image to be classified. On the output layer a single neuron could be present. For a binary classifier of hotdog or not-hotdog the neural network could be used to identify if a hotdog is present in an image. The number of neurons in the hidden layers will determine the robustness and accuracy of the classifier.

The number of neurons in the hidden layer could be in the billions for large networks. For instance, in the latest OpenAI model GPT-3 there are 175 billion parameters. [5] Each of these parameters represents a weight between neurons in the hidden layer. These weights are floating point values that take significant RAM to store and are a hinderance to running large models on consumer hardware.

These parameters are set through a training process which is iterative. Test cases of inputs are fed to the inputs of the network and the results are compared to what a known good result is. Back-propagation is then performed updating the weights throughout the network. [6] The most important part outside of the topology of neurons is these weights.

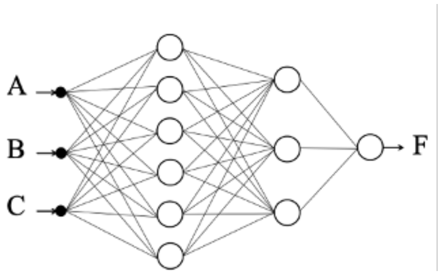


Fig. 4. A complete network with 2 hidden layers of neurons from [6]

II. Memristors

Until Chua's paper there were thought to be three basic circuit elements resistors, inductors, and capacitors. These elements provide a link and relationships between voltage, current, flux, and quality factor. From these relationships a missing link between flux and quality factor motivates the fourth basic circuit element the memristor. [2]

A. Electrical Model

This two port device acts like a resistor at any point in time, but the value of the resistance is based on the past voltage applied across the device. Thus, it has a memory-based resistance. This device exhibits non-linear behavior.

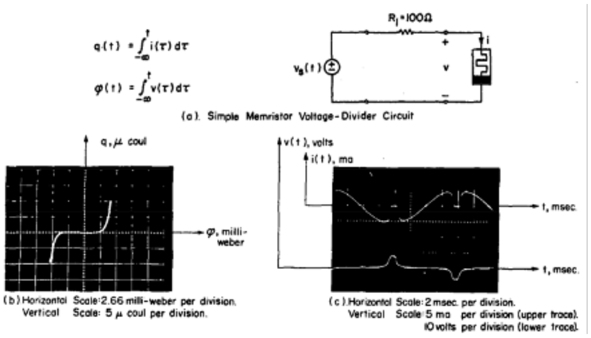

$q(t) = \int_{-\infty}^{t} i(\tau) d\tau$
$\varphi(t) = \int_{-\infty}^{t} v(\tau) d\tau$
$R_l = 100\Omega$
$v_s(t)$
i
+
v
−
(a). Simple Memristor Voltage-Divider Circuit
q, μ coul
φ, milli-weber
(b) Horizontal Scale: 2.66 milli-weber per division.
Vertical Scale: 5 μ coul per division.
v(t), volts
i(t), ma
t, msec.
t, msec.
(c) Horizontal Scale: 2 msec. per division.
Vertical Scale: 5 ma per division (upper trace).
10 volts per division (lower trace).

Fig. 5. Voltage and Current relationships of Memristor [2]

With HP Labs successfully fabricating memristors in 2008, new research and applications of memristors have increase dramatically. Because of the limited hardware available, simulation remains an important way to study applications of memristors.

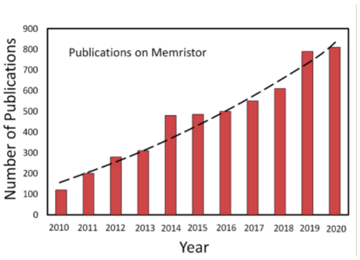
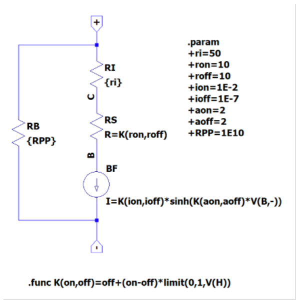


Fig. 6. The number of publications on memristors from 2010 to 2020 [9]

To model the memristors a current source is used as described in Chua's original paper. A SPICE model is listed below and its parameters are fit with measurements from memristor hardware. This enables the simulation of thousands or millions of memristors for study.

Memristors can change their resistance based on the voltage applied in the past to the device. This is accomplished by a split dielectric with regions that are doped differently that cause the resistance to change. [7] This change is persisted until another pulsed charge is applied to the device. There are two states Low Resistance State (LRS) and High Resistance State (HRS) these are toggled through SET and RESET lines like a flip-flop. To distinguish between a read and a set, the amount of voltage applied in the pulse width modulation is much higher than what is used in reading, similar to a breakdown voltage of a diode



```
.param
+ri=50
+ron=10
+roff=10
+ion=1E-2
+ioff=1E-7
+aon=2
+aoff=2
+RPP=1E10
```

```
.func K(on,off)=off+(on-off)*limit(0,1,V(H))
```

Fig. 7. Memristor SPICE Electrical Model [7]

B. Fabrication

Several methods are available to realize memristor hardware. The topology of fabrication normally places the memristor in a crossbar pattern. This crossbar patterns allows hundreds or thousands of memristors to be fabricated in a single structure. Traditional CMOS methods of fabrication are available currently from TSMC 40nm RRAM Process. [8]

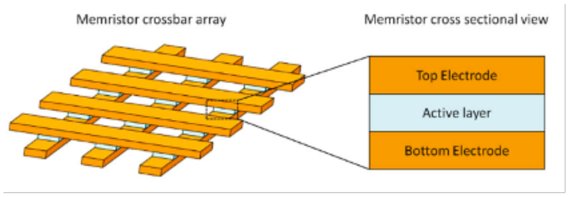


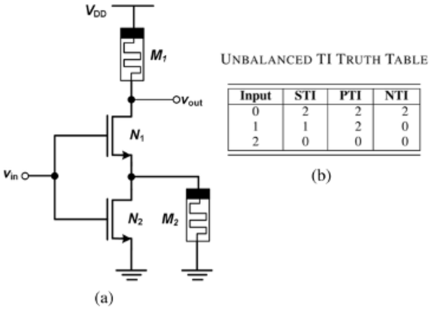
Fig. 8. Crossbar structure for memristors [9]

Material used for the active layer or dielectric in all the fabrications mentioned in a survey of techniques is TiO_2. [9] While the electrodes are often platinum, they can also be other materials like TiN. The active area is dielectric that is processed with various doping techniques to create the memristor behavior. In some cases, Hafnium Oxide is used for the dielectric or active area. [10]

C. Computational Methods

Although the memristor up this point has been alluded to as a memory device, it also possesses the ability for Boolean logic. NAND, NOR, and other basic gates can be constructed with memristors. In the quest for more density the memristor provides a tradeoff of reduced space used versus noise and accuracy requirements. Sometimes it is more important for data density, and this is where memristors provide an order of a magnitude improvement. [8]

With the usage of memristors as both a storage device and logic device von-Neumann architecture that is traditionally used in CMOS could be replaced or augmented. Because memristor crossbar arrays are field reconfigurable they can act more like a Programmable Logic Array (PLA) or Field Programmable Gate Array (FPGA)



UNBALANCED TI TRUTH TABLE

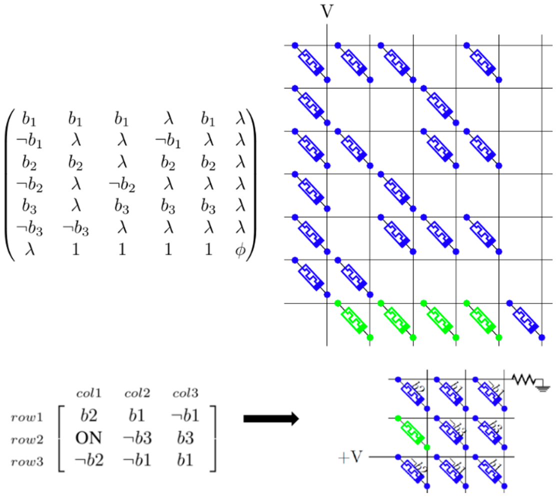
Input	STI	PTI	NTI
0	2	2	2
1	1	2	0
2	0	0	0

(b)

Fig. 9. Proposed ternary inverter gate - NMOS transistors and memristors [8]

While Electronic Design Automation (EDA) tools are lacking for memristors and these new architectures, the benefits of memristors may become more mainstream. To aid in the design of logic units with memristors, automated formal methods of generating crossbars have been explored. [11]

Further minimizing area by reducing effects of "Sneak Currents" Velasquez can realize Boolean logic with automated tools. These circuits can become much smaller than designing by hand because of the elimination of many disabled memristor that are removed to prevent unintended currents. This method is used to create circuits like in Figure 10 of a 3-bit parity check. The results of the computation are being stored in bottom right memristor. The green memristor is in LRS or ON.



$$\begin{pmatrix} b_1 & b_1 & b_1 & \lambda & b_1 & \lambda \\ \neg b_1 & \lambda & \lambda & \neg b_1 & \lambda & \lambda \\ b_2 & b_2 & \lambda & b_2 & b_2 & \lambda \\ \neg b_2 & \lambda & \neg b_2 & \lambda & \lambda & \lambda \\ b_3 & \lambda & b_3 & b_3 & b_3 & \lambda \\ \neg b_3 & \neg b_3 & \lambda & \lambda & \lambda & \lambda \\ \lambda & 1 & 1 & 1 & 1 & \phi \end{pmatrix}$$

$$\begin{array}{cc} & \begin{array}{ccc} col1 & col2 & col3 \end{array} \\ \begin{array}{c} row1 \\ row2 \\ row3 \end{array} & \begin{bmatrix} b2 & b1 & \neg b1 \\ \text{ON} & \neg b3 & b3 \\ \neg b2 & \neg b1 & b1 \end{bmatrix} \end{array}$$

Fig. 10. 3-bit Parity Construction with green memristors in LRS. Top is manually drawn with removed memristors to prevent sneak currents. Bottom is generated circuit through automation with formal methods [11]

III. NEUROMORPHIC COMPUTING WITH MEMRISTORS

Neuromorphic computing is defined as using biologically inspired architecture rather than traditional von-Neumann architecture. What is known as the von-Neumann bottleneck, computational and logic elements must communicate with and receive/write back data to memory elements. Memristors provide an opportunity to break this paradigm.

Artificial Neural Networks used for large language models are dominated currently by GPU manufacturer NVIDIA. [5] The large language models [LLM] described by Touvron in the LLaMA paper have successfully captivated the consumer in paying for access to these AI models which can be used to generate text and visual products. The AI generated text could be said to be good enough to pass as a human. [12]

The size of the networks and the storage requirements of the weights for LLM are pushing the boundaries of the biggest GPUs available and with the supply constraints of NVIDIA A100 GPUs with 80gb of RAM the time to train takes 21 days and million dollars for 13B parameter model. [13] The costs to train the GPT-3.5 and 4 with over a trillion parameters have not been published.

A. *Neural Networks with Memristors*

The storage and energy costs of GPU based Neural Networks [NN] approaching uneconomical heights, memristors have been studied for application. Smaller NNs have been realized at the hardware level with memristors. [14] Because large memristor arrays are not generally available, most of the research must be done with combination of measurement and approximate simulations.

In 2021 an article published presents a fully memristive neural network that was trained completely in hardware. [14] Kiani et al used analog components (opamps) to connected through printed circuit boards to their memristor arrays to create ReLu activation units for a fully hardware-based network of neurons.

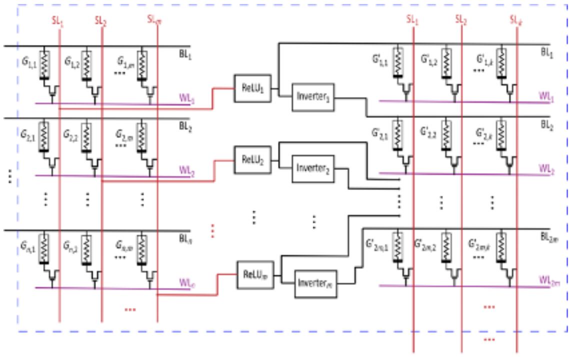


Fig. 11. Schematic of hardware neuron with activation ReLu [14]

The fully hardware based neural network was able to classify digits in the MNIST dataset to the accuracy of 95%. Although this is considered a simple task by modern standards it is a good proof of concept that on device training can be accomplished with memristor neural networks. The estimation of savings of die space to be 5x in the 65nm CMOS technology node and a power savings of 32x. [14]

Performing the training with back-propagation in explained in more detail in Hasan's 2014 paper. The training is done fully in hardware and at the end of each iteration it adjusts the weights through setting memristors HRS minimizing the error for each layer. [15]

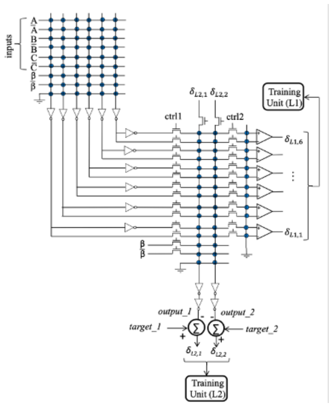


Fig. 12. Schematic of crossbar training with back-propagation [15]

B. *Improving Robustness of Memristors*

With the storage of weights there are a few challenges using memristors. First the resolution of the floating-point values that are stored is approximately 4bits. It is possible to use as little as 8bits with some neural networks, but accuracy and stability of the network starts degrading. [5]

To improve the accuracy, Das swept line widths of the crossbars in a Cadence SPECTRE simulation. An improvement of nearly 20% in read resolution of memristor values was found with an optimal design rule and current being used. [10]

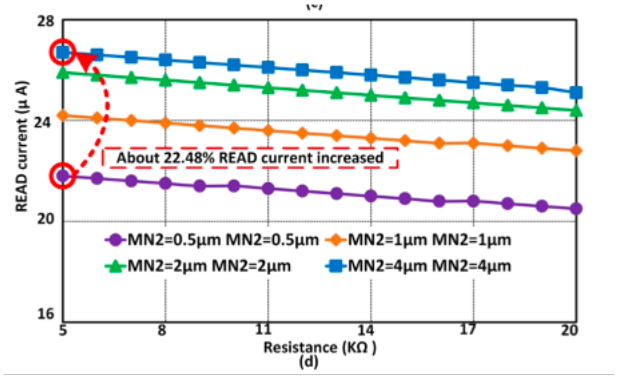


Fig. 13. SPECTRE simulation sweeping geometries of memristors [10]

IV. Commercialization

There are not many mass market devices that include memristors currently. In 2015 a company KNOWM was formed and started producing development kits for researchers.

Currently you can purchase a 16x1 array of memristors of various material types from KNOWM website. The cost of these devices is 139$ and come in a bare die package as well as bond-wire development PCBs. [16]

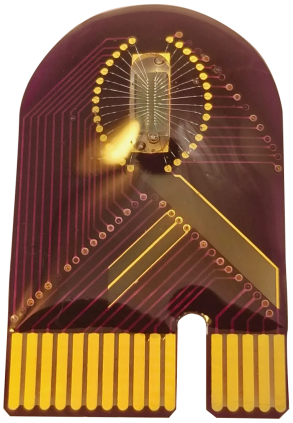
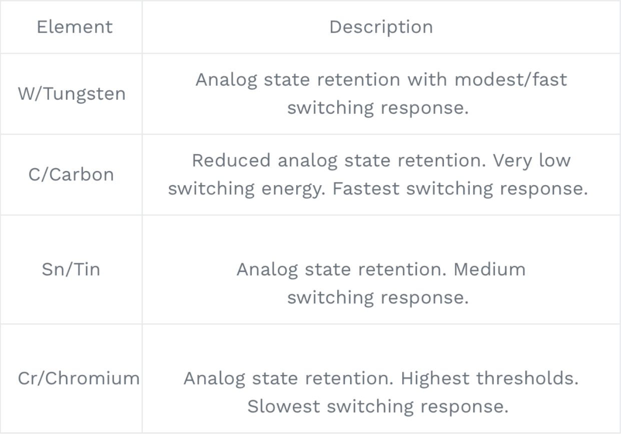


Fig. 14. KNOWM 16x1 Memristor Development PCB M+SDC

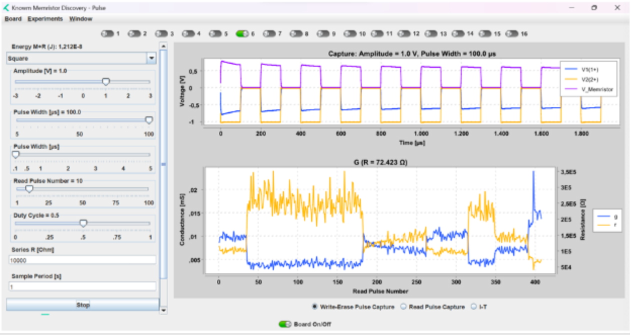
This development board kit contains software for doing neuromorphic experiments and has been used to fit models that are then simulated in SPICE. [7] The various materials available from KNOWM provide various tradeoffs. Switching speed, stability, and long-term non-volatility can be selected in the various material dopings.



Element	Description
W/Tungsten	Analog state retention with modest/fast switching response.
C/Carbon	Reduced analog state retention. Very low switching energy. Fastest switching response.
Sn/Tin	Analog state retention. Medium switching response.
Cr/Chromium	Analog state retention. Highest thresholds. Slowest switching response.

Fig. 15. Materials available for specific features from KNOWM

The programming of the KNOWM Memrisor PCB is done through the Digilent Discovery USB device. Pulse widths and ramp times as well as voltages can be configured to experiment with the memristors.



Energy M+R (J): 1,212E-8

Square

Amplitude [V] = 1.0

Pulse Width [µs] = 100.0

Pulse Width [µs]

Read Pulse Number = 10

Duty Cycle = 0.5

Series R [Ohm]: 10000

Sample Period [s]: 1

● Write-Erase Pulse Capture ○ Read Pulse Capture ○ I-T

Fig. 16. KNOWM program for programming memristors

V. Conclusions

The memristor is a fascinating device. From theory of its existence to its actual realization in 2008, the applications continue to grow. Memristors can be used to store information within a much smaller region of die than traditional CMOS NAND gates.

The power consumption is significantly less than traditional CMOS as well. With these advantages researchers have simulated and fabricated memristors to be applied to neuromorphic computing.

On device training has been successful in training small neural networks for classification tasks. The usage of the memristor to store the weights of the neural networks as well as the ability to perform logic switching offers the option eliminate the von-Neumann bottleneck between computation and memory.

While the EDA tools are lacking for large scale designs with memristors, some researchers are developing boutique tools for automation. With these automations and the fact that TSMC supports Memristor devices in their design rule manual for their RRAM process, it is just a matter of time before we see a mass market product using this technology originally postulated by Leon Chua in 1971.

References

[1] Kumar, Suhas. (2015). Fundamental Limits to Moore's Law.

[2] L. Chua, "Memristor-The missing circuit element," in IEEE Transactions on Circuit Theory, vol. 18, no. 5, pp. 507-519, September 1971, doi: 10.1109/TCT.1971.1083337.

[3] Yang, J., Pickett, M., Li, X. et al. Memristive switching mechanism for metal/oxide/metal nanodevices. Nature Nanotech 3, 429–433 (2008). https://doi.org/10.1038/nnano.2008.160

[4] Singh, Saurabh & Hussain, Shoeb & Bazaz, Abid. (2017). Short term load forecasting using artificial neural network. 1-5. 10.1109/ICIIP.2017.8313703.

[5] H. Touvron et al., LLaMA: Open and Efficient Foundation Language Models. 2023.

[6] Enabling Back Propagation Training of Memristor Crossbar Neuromorphic Processors

[7] P. A. Lázaro, I. J. Gallo, J. R. Aranda, A. del Barrio García, G. B. Juan, and F. J. Molinos, Design and simulation of memristor-based neural networks. 2023.

[8] X. -Y. Wang et al., "High-Density Memristor-CMOS Ternary Logic Family," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 68, no. 1, pp. 264-274, Jan. 2021, doi: 10.1109/TCSI.2020.3027693.

[9] S. Ali, S. Khan, A. Khan and A. Bermak, "Memristor Fabrication Through Printing Technologies: A Review," in IEEE Access, vol. 9, pp. 95970-95985, 2021, doi: 10.1109/ACCESS.2021.3094027.

[10] H. Das, N. N. Chakraborty, C. Schuman, and G. S. Rose, Enhanced Read Resolution in Reconfigurable Memristive Synapses for Spiking Neural Networks. 2023.

[11] A. Velasquez and S. K. Jha, "Automated synthesis of crossbars for nanoscale computing using formal methods," Proceedings of the 2015 IEEE/ACM International Symposium on Nanoscale Architectures (NANOARCH´15), Boston, MA, USA, 2015, pp. 130-136, doi: 10.1109/NANOARCH.2015.7180599.

[12] Elkins, Katherine & Chun, Jon. (2020). Can GPT-3 Pass a Writer's Turing Test?. Journal of Cultural Analytics. 5. 10.22148/001c.17212.

[13] Ray-Project, "Ray-project/LLM-numbers: Numbers every LLM developer should know," GitHub, https://github.com/ray-project/llm-numbers (accessed Jul. 12, 2023).

[14] Fatemeh Kiani et al. ‚A fully hardware-based memristive multilayer neural network.Sci. Adv.7,eabj4801(2021).DOI:10.1126/sciadv.abj4801

[15] R. Hasan and T. M. Taha, "Enabling back propagation training of memristor crossbar neuromorphic processors," 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, 2014, pp. 21-28, doi: 10.1109/IJCNN.2014.6889893.

[16] [1] "M+SDC 16 discrete memristor encapsulated edge," Knowm Inc, https://knowm.com/collections/frontpage/products/m-sdc-16-discrete-memristor-encapsulated-edge (accessed Jul. 12, 2023).