Hybrid Analog-Digital Learning with Differential RRAM Synapses

T. Hirtzlin, Marc Bocquet, M. Ernoult, J.-O Klein, E. Nowak, E. Vianello, J.-M Portal, D. Querlioz

Hybrid Analog-Digital Learning with Differential RRAM Synapses

T. Hirtzlin[1*], M. Bocquet[2*], M. Ernoult[1], J.-O. Klein[1], E. Nowak[3], E. Vianello[3], J.-M. Portal[2] and D. Querlioz[1]
[1]C2N, Univ Paris-Sud, CNRS, Orsay, France, email: damien.querlioz@c2n.upsaclay.fr
[2]Aix Marseille Univ, Université de Toulon, CNRS, IM2NP, Marseille, France
[3]CEA, LETI, Grenoble, France [*]These authors contributed equally to the work

***Abstract*—**Exploiting the analog properties of RRAM cells for learning is a compelling approach, but which raises important challenges in terms of CMOS overhead, impact of device imperfections and device endurance. In this work, we investigate a learning-capable architecture, based on the concept of Binarized Neural Networks, which addresses these three issues. It exploits the analog properties of the weak RESET in hafnium-oxide RRAM cells, but uses exclusively compact and low power digital CMOS. This approach requires no refresh process, is more robust to device imperfections than more conventional analog approaches, and we show that due to the reliance on weak RESETs, the devices show outstanding endurance that can withstand multiple learning processes.

I. Introduction

In recent years, the progresses of deep neural networks have led to an incredible development of artificial intelligence (AI), but which bears considerable costs in terms of energy consumption. As resistive memory cells are reminiscent of neural networks synapses, research aims at solving the AI energy challenge by designing hardware neural networks with RRAMs as synapses. A particularly attractive idea is to exploit their analog properties for learning [1], [2]: during training, following a learning rule, the synaptic weights, implemented by RRAM conductance, are adjusted sequentially until the neural network reaches maximum accuracy. This technique is compelling but faces multiple challenges. First, it requires analog CMOS circuitry or conversion between analog and digital signals, which brings high area and energy overheads [3]. Second, it is highly sensitive to RRAM imperfections such as asymmetry between SET and RESET process, non-linearity and noise [1]. Third, devices optimized for analog operation may suffer from reduced reliability and endurance [4].

In this work, we solve these three issues by proposing a learning-capable hardware Binarized Neural Network (BNN). The learning process uses the particular properties of the weak RESET process of hafnium oxide-based RRAM devices featuring high endurance and fully integrated in a commercial CMOS technology. This design exploits the analog physics of the RRAM cells, but uses exclusively digital CMOS, since the resistance value of RRAM devices is never extracted. This approach also avoids the need of often employed refresh operation [1], [5]. Moreover, based on extensive endurance measurements, we show that the devices can sustain multiple learning processes. This works extends to learning our prior work on the same RRAM technology on inference (non-learning) AI [6]. It simplifies the proposal of [5] by replacing the refresh operation by a simple programming strategy, and demonstrates the required device reliability.

II. Hybrid Analog / Digital Learning

Conventional neural networks are constituted of real valued neurons connected by synapses, characterized by their real valued weight (Fig. 1(a)). In BNNs, by contrast, both synaptic weights and neurons assume binary values (+1 and -1) [7], [8]. During inference, this makes their arithmetic extremely simple: multiplication is replaced by one-bit XNOR gates, and sum is replaced by POPCOUNT operations (counting the number of ones). Despite their simplicity, BNNs can approach state-of-the-art performance on vision tasks. During training, a hidden real weight is also associated with synapses, which is adjusted by the learning rule. The binarized weight used by the neural network in all arithmetic operations is the sign of the hidden real weight (Fig. 1(b)). Once training is finished, the hidden real weight is of no use and can be discarded. For this reason, BNNs are now vastly investigated for inference hardware, *i.e.* hardware with no learning capability [6], [9], [10]. For RRAM-based learning hardware, BNNs do not seem ideal at first sight, as synapses need to be associated with real weights. However, the BNN learning process does not need to *know* the value of the hidden real weight, as only the sign is used in arithmetic operations. The network only needs the ability to increase or decrease it. We show here that this is feasible in a simple manner employing RRAM cells with only an adapted programming sequence and purely digital CMOS.

For this work, we use a hafnium oxide based OxRAM technology, integrated in the BEOL of a 130 nm CMOS logic process, on top of the fourth metal layer (Cu), and based on a TiN/HfO_2 (10 nm)/Ti (10 nm)/TiN stack (Fig. 2). Synapses are implemented using a 2T2R differential structure comprising two devices "BL" and "BLb" (Figs. 3 and 4), similar to the one used in [5], [6], and we make use of a full 1Kbit / 2048 devices array incorporating the CMOS row and column decoders as well as sense circuitry. The 2T2R synapses are used in the following way (Fig. 5): if the BL cell has higher resistance than the BLb one, the 2T2R structure implement synaptic weight +1, and otherwise -1. The synaptic weight can therefore be obtained by comparing the conductance of the two devices. In our chip, this is achieved by using the sense amplifier of Fig. 6(a) [11]. This sense amplifier can further be enhanced with an XNOR functionality by the addition of 4 transistors, allowing realizing the multiplication directly within the memory array (Fig. 6(b)).

For learning, we exploit weak RESET pulses [12], which exhibit a progressive effect (Fig. 7(a)): when such a pulse is applied to a RRAM cell, it increases the resistance of the device slightly. This effect is strongly dependent on the programming voltage and duration, and is subject to significant noise and variability due to the atomic size of filaments in the RRAM devices (Fig. 7(a)/8). The conductance vs. pulse number is also highly nonlinear. It is however seen on all devices that we measured. Fig. 9 also highlights the high progressivity level of the process. Finally, in Fig. 10, we propose a behavioral model for the effect of weak RESET, including noise. Fig. 7(b) shows comparison of the model and of measurement.

As the SET process does not feature progressive effect, we mainly exploit the RESET effect for learning. Whenever the learning rule of BNNs predicts to increase a synaptic weight, we apply a weak RESET to the BL device, whereas if the learning rule predicts a decrease in the weight, we apply a weak RESET on the BLb device (Fig. 11(a)). This technique naturally leads to a reversal of the binarized weight when the resistance of the two devices are crossing. However, it raises the issue that device resistance can saturate. [5] solved this issue by performing refresh operations, which are time and energy hungry and require complex programming circuits. Here, we propose a simpler approach based on an adapted programming sequence implemented only with pure digital circuitry (Fig. 11(b)). Before any write operation, we check the binary value of synapse. After the progressive RESET, we check if the binary value of the synapse has switched. If this has happened both devices are reprogrammed with a full SET and one with a weak RESET to recover an opportunity for progressive RESET.

III. Device Imperfections and Learning

We now check the efficiency and robustness of our technique in practical simulation. We train a neural network with 784 inputs neurons, a hidden layer of 1024 neurons and 10 output neurons on the MNIST handwritten recognition task, the RRAM cells are modeled with the experiment-matched compact model and CMOS circuits with cycle accurate modeling. Fig. 12 shows the training process if the post-programming check is not performed (Fig. 11(a)). The network does learn to recognize digits with ~98% accuracy, but the learning is not stable: when the devices start to saturate, the network accuracy starts to drop. Fig. 13 adds the post-programming check (Fig. 11(b)): accuracy is not affected, but learning is now stable. We now look at the impact of various devices issues. In Fig. 14, we simulate the neural network training for different levels of noise (η in Fig. 10) associated with the weak RESET process. We compare our BNN strategy and a control case employing the conventional approach of using RRAM devices in a fully analog fashion. We see that with our approach, unlike the conventional one, noise is tolerated to an outstanding level. The level of noise seen in our experimental devices (η=3 in Fig. 7) causes no accuracy degradation, whereas it would cause important degradation in the conventional approach. Fig. 15 shows that RESET noise is not only tolerated, in some situations it can even be useful. In this particular situation, several training processes are realized cumulatively, without resetting devices in-between the processes. We see that the re-learning processes are more effective when RESET noise is stronger. Fig. 16 looks at the impact of the nonlinearity of the RESET process (β in Fig. 10) as well as the non-symmetry between the SET and RESET process in our approach and in the conventional approach. We see an outstanding tolerance of our approach to these effects, whereas the conventional approach is highly sensitive to them.

Our approach can function with extremely low energy. To anticipate its power consumption in a more modern technology than the 130 nm technology of our experiments, we designed the whole system (with digital CMOS) in a commercial 28 nm process, and evaluated its energy requirements using the Cadence encounter tool. The whole learning process, with programming conditions of Fig. 7 would consume only ~50 mJ. Once learning is finished, recognizing a digit requires only ~25 nJ, orders of magnitude below CPUs and GPUs. Fig. 17 compares the energy requirement with the one of an in-memory ASIC with conventional neural network and 8-bits precision. Our approach typically saves a factor ten in energy.

IV. High Device Endurance and Learning

A very important concern is endurance, and the reliance of our technique on weak RESET pulses has tremendous benefits in that regards. Fig. 18 shows cycled measurements alternating strong SET and weak RESET pulses. Outstanding endurance of 55 Billion cycles is seen on the two devices under test. Fig. 19 shows measurements on 32 synapses/64 devices. After 10^5 pulses, all devices are still functional and device variability is not increased. In our simulations, training a neural network on MNIST typically involved ~10^4 cycles. This result shows that a chip with our approach could allow multiple learning processes.

V. Conclusion

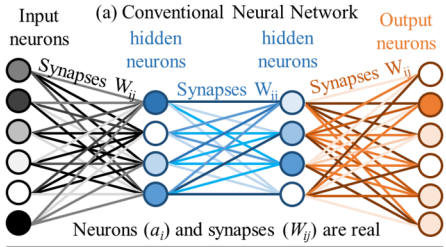
This works presents a hybrid road for implementing learning with RRAM devices, which exploits the analog physics of RRAM but relies entirely on digital CMOS. Table I compares our approach to the literature. The approach is simpler and more resilient to device imperfection than analog approaches, which allows relying on weak RESETs. In these conditions, the devices show a lot of noise but outstanding endurance, allowing performing multiple learning processes.

Acknowledgment

This work is supported by ERC grant NANOINFER (715872) and ANR grant NEURONIC (ANR-18-CE24-0009).

References

[1] S. Ambrogio *et al.*, *Nature*, vol. 558, no. 7708, pp. 60–67, Jun. 2018.
[2] F. Alibart *et al*, *Nat. Commun.*, vol. 4, Jun. 2013.
[3] F. Cai *et al.*, *Nat. Electron.*, p. 1, Jul. 2019.
[4] A. Serb *et al.*, *Nat. Commun.*, vol. 7, p. 12611, Sep. 2016.
[5] Z. Zhou *et al.*, in *IEDM Tech Dig.*, 2018, pp. 20.7.1-20.7.4.
[6] M. Bocquet *et al.*, in *IEDM Tech. Dig*, 2018, pp. 20–6.
[7] I. Hubara *et al.*, in *Proc. NIPS*, 2016, pp. 4107–4115.
[8] M. Rastegari *et al.*, in *Proc. ECCV*, 2016, pp. 525–542.
[9] T. Hirtzlin *et al.*, *IEEE Access*, vol. 7, pp. 76394–76403, 2019.
[10] S. Yu *et al.*, *Proc. IEEE*, vol. 106, no. 2, pp. 260–285, Feb. 2018.
[11] W. Zhao *et al.*, *IEEE TCAS I*, vol. 61, no. 2, pp. 443–454, Feb. 2014.
[12] G. Piccolboni *et al.*, in *IEDM Tech. Dig.*, 2015, pp. 17.2.1-17.2.4.
[13] V. Sze *et al.*, Proc. IEEE, vol. 105, no 12, p. 2295□2329, déc. 2017.4.

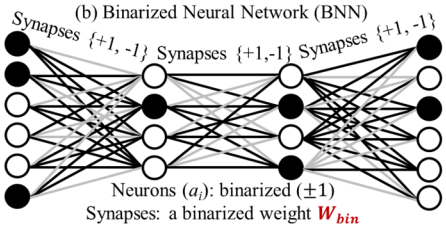


Inference :

$$a_i = f\left(\sum_j W_{ij} \cdot a_j\right)$$

Learning :

$$W_{ij} \leftarrow W_{ij} + LearningRule(W_{ij})$$



Inference :

$$a_i = sign\left(popcount\ XNOR(W_{bin,ij}, a_j) - T\right)$$

Learning :

$$W_{real,ij} \leftarrow W_{real,ij} + LearningRule(W_{bin,ij})$$

Fig. 1. Principle of Binarized Neural Networks

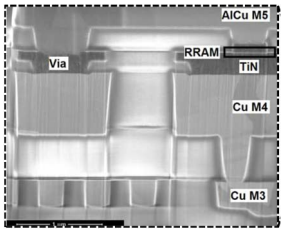


Fig. 2. SEM image of an RRAM device integrated in the BEOL of our technology.

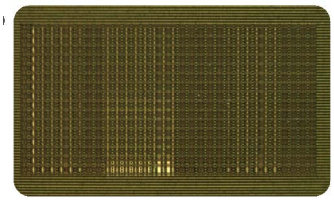


Fig. 3. Photograph of the die of our test chip with 1K synapses / 2K RRAM cells

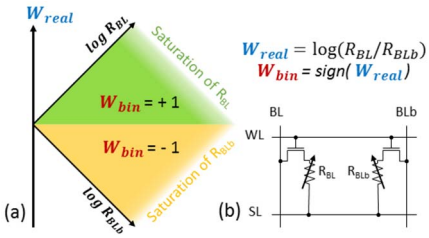


Fig. 5. Artificial synapse using a 2T2R structure.

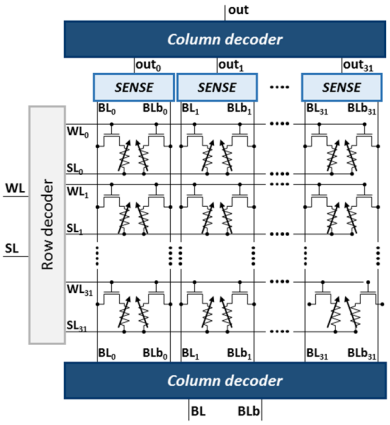


Fig. 4. Schematic of the 1K synapses / 2K RRAM cells test chip used in this work

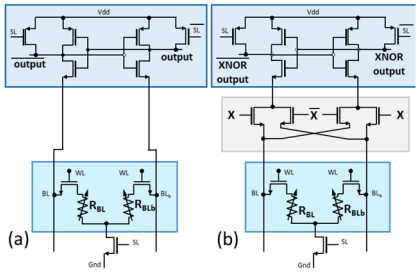


Fig. 6. (a) Schematic of the sense amplifier used to extract the binary weight from a 2T2R synapse. (b) Version augmented with an XNOR feature.

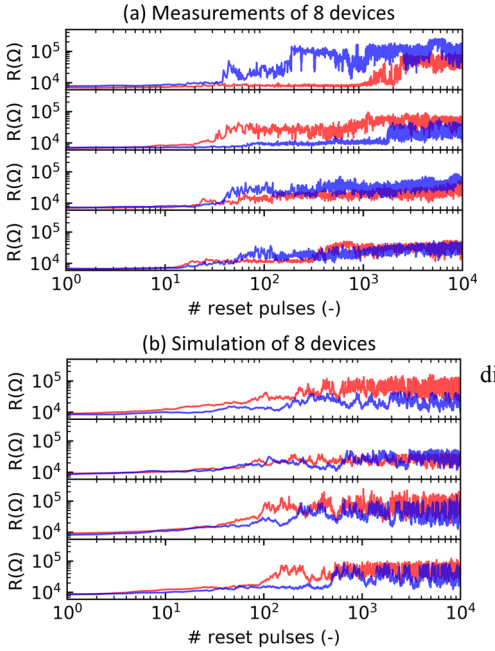


Fig. 7 (a) Measurements of the weak RESET process on 8 randomly chosen devices (b) Simulations of the weak RESET process using eqs. of Fig. 10. V_{RESET} = 1.2 V, t_{RESET}=100 ns

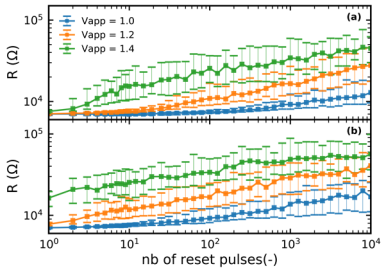


Fig. 8. Statistical measurements of the weak RESET process over 64 devices. Error bar is one standard deviation, for different RESET voltages. (a) t_{RESET}=1 µs. (b) t_{RESET}=100 ns.

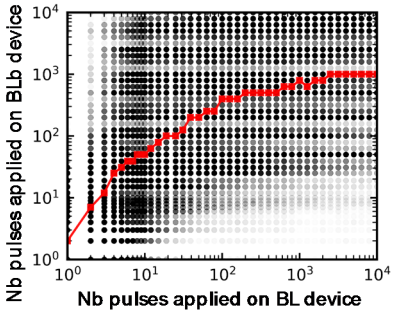


Fig. 9. Statistical measurements on 2T2R cells. Black points: number of weak RESET pulses on the BLb device to reverse W_{bin}, as a function of number of weak RESET pulses applied on the BL device. Red point: mean values, over 36,800 synapses tested. V_{RESET} = 1.2 V, t_{RESET}=100 ns

$$\frac{dG}{dt} = -C\exp\left(-\beta\frac{G_{max} - G(t)}{G_{max} - G_{min}}\right) + \eta Noise(t)$$

G : *Device conductance*	η : *Noise factor*
β : *Nonlinearity of the device*	C : *RESET strength*
G_{max}/G_{min} : *Max/Min device conductance*	

Fig. 10. Analytical equations used to model the weak RESET process in our RRAM cells. Noise(t): Gaussian noise.

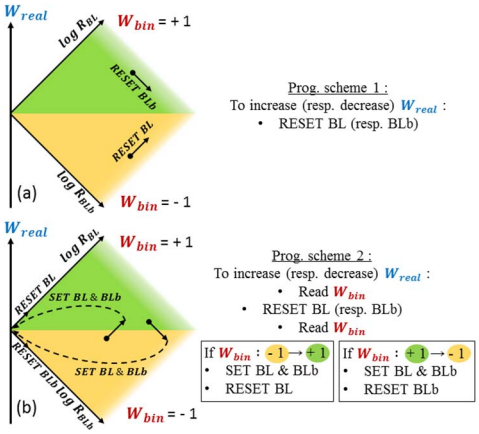


Fig. 11. Learning strategy

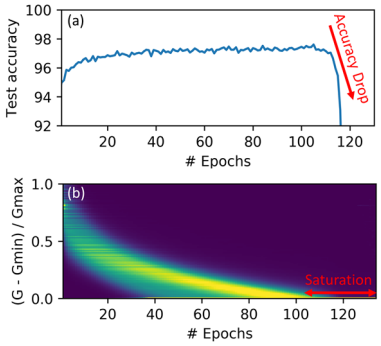


Fig. 12. (a) Test accuracy on the MNIST task and (b) distribution of the conductance of the RRAM cells, in the neural network trained **without** post-RESET check (Fig. 11(a)).

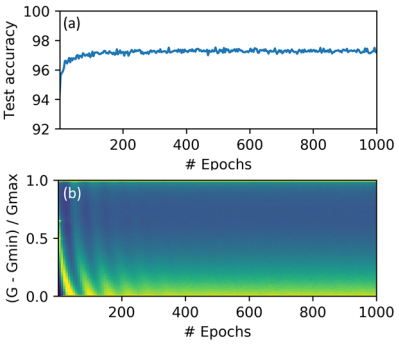


Fig. 13. (a) Test accuracy on the MNIST task and (b) distribution of the conductance of the RRAM cells, in the neural network trained **with** post-RESET check (Fig. 11(b)).

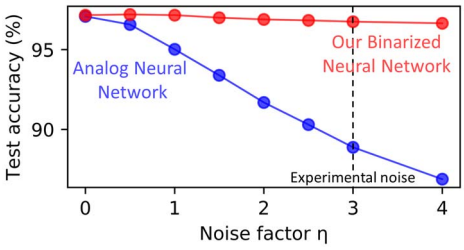


Fig. 14. Test accuracy of the neural network on MNIST as a function of the noise parameter in Fig. 10, for our BNN and an analog neural network (conventional approach).

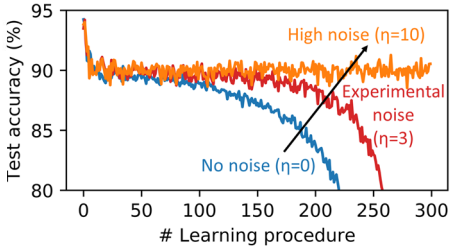


Fig. 15. Test accuracy of the neural network retrained on variations on the task without reinitialization of the devices, for different noise levels.

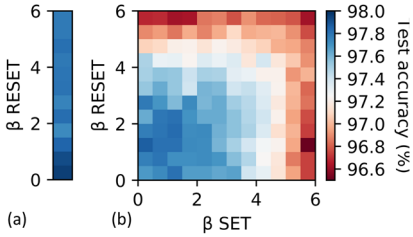


Fig. 16 Test accuracy of the neural network on MNIST as a function of non-linearity of the SET and RESET process, for (a) our BNN and (b) an analog neural network (conventional approach).

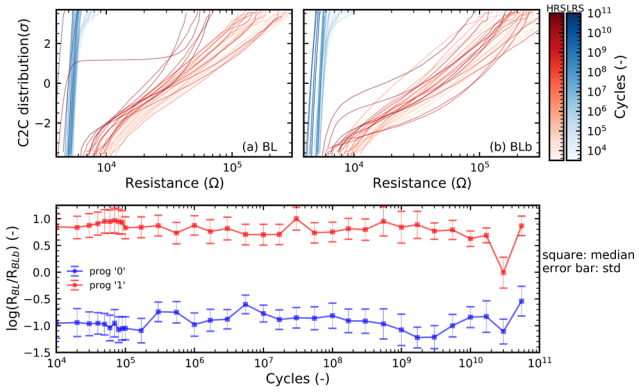


Fig. 18 Endurance measurement on two complementary devices programmed with weak RESET (V_{RESET}=1.5V), pulse width of 1 µs and SET compliance current of 200 µA. (a-b) Cycle-to-cycle (C2C) distribution of resistance values for 10k cycle. (c) median value resistance ratio (R_{BL}/R_{BLb}), extracted over 10k cycles.

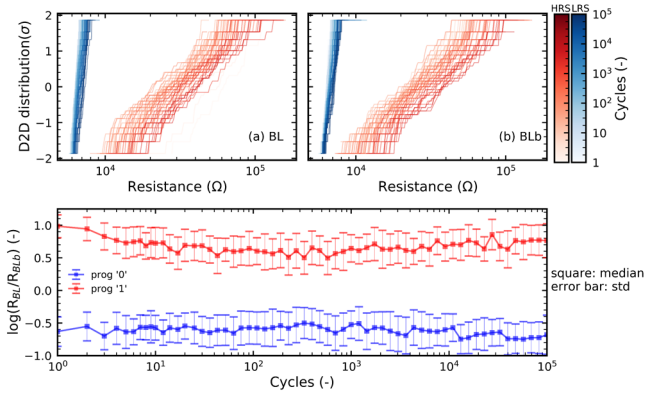


Fig. 19 Endurance measurement on 64 devices, same programming conditions as Fig. 18. (a-b) Device-to-device (D2D) distribution of resistance values. (c) median value resistance ratio (R_{BL}/R_{BLb}), extracted over 10k cycles.

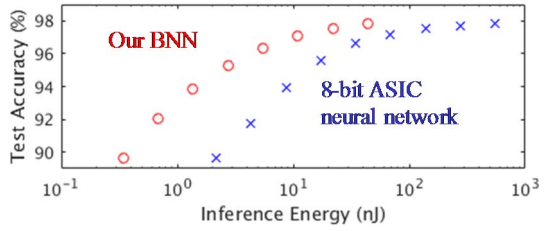


Fig. 17. Red circles: energy consumption of our BNN to recognize one digit, as a function of targeted test accuracy (simulation in a 28 nm technology). Blue: same for an in-memory ASIC with 8 bits precision

	Devices /synapse	CMOS overhead	Resilience noise	Resilience nonlinearity	Resilience asymmetry	Requires refresh
Analog RRAM [2,3]	1	High	Low	No	No	No
Analog PCM [1]	2	High	Low	No	Yes	Yes
Fully Digital [13]	8 or more	Medium	Medium	Yes	Yes	No
[5]	2	Low	High	Yes	Yes	Yes
This work	**2**	**Low**	**High**	**Yes**	**Yes**	**No**

Table I. Comparison of our approach with approaches of the literature **for learning** with resistive memories.