Fully Integrated Spiking Neural Network with Analog Neurons and RRAM Synapses

A. Valentian[1], F. Rummens[1], E. Vianello[1], T. Mesquida[1], C. Lecat-Mathieu de Boissac[1*], O. Bichler[2], C. Reita[1]
[1]CEA-Leti, MINATEC Campus, Grenoble, France, email: alexandre.valentian@cea.fr
[*]Now with STMicroelectronics – [2]CEA, LIST, Gif-sur-Yvette, France

Abstract— This paper presents, to the best of the authors' knowledge, the first complete integration of a Spiking Neural Network, combining analog neurons and Resistive RAM (RRAM)-based synapses. The implemented topology is a perceptron, aimed at performing MNIST classification. An existing framework was tailored for offline learning and weight quantization. The test chip, fabricated in 130nm CMOS, shows well-controlled integration of synaptic currents and no RRAM read disturb issue during inference tasks (at least 750M spikes). The classification accuracy is 84%, with a 3.6 pJ energy dissipation per spike at the synapse and neuron level (up to 5x lower vs. similar chips using formal coding).

I. Introduction

Data-centric workloads exhibited by Deep Neural Network (DNN) applications call for circuit architectures where data movement is reduced to a minimum. This has motivated architectures in which memories are spatially located near the computing elements. These memories must be very dense, preferably non-volatile and inserted into the computational dataflow, therefore RRAMs are excellent candidates to this purpose. DNN computations using classical formal coding and RRAM cells have recently made significant progress [1, 2]. Spiking Neural Networks (SNN), also called 3rd generation of NNs, operates using spikes, which are discrete events that take place at point in time, rather than continuous values. These networks are promising to further reduce the computational power [3]. To date, demonstrations of RRAM based SNNs have been limited to system level simulations calibrated on experimental data [4]. In this paper, we present for the first time a complete integration of a SNN combining analog neurons and RRAM synapses.

II. Learning Framework

One of the issue of SNNs is training (*i.e.* the definition of the synaptic weights). Although there are unsupervised biological learning methods such as Hebbian learning (*e.g.* Spike Time Dependent Plasticity), they do not offer good performances as supervised training models. For this reason, we do learning off line in the classical domain using the gradient descent algorithm, which represents the state of the art. The network is afterwards converted into a quantized, spike-based network (Fig. 3). For both learning and conversion, we used N2D2 [5], which is to our knowledge, the first public deep learning framework to integrate transcoding of spiking DNNs, simulation and dedicated hardware code generation.

A. Transcoding principle

The rate-based transcoding principle was partially formalized for Leaky-Integrate & Fire (LIF) neurons in [6]. We have, in this work, improved it and extended it for simple Integrate & Fire (IF) neurons (Fig. 4).

The base operation in classical coding is the Multiply-Accumulate (MAC) one, between input data x_i and the associated weights w_{ij} in a given receptive field (Eq. 1 in Fig.4). Omitting the non-linear function $h()$ in Eq.1 (Fig. 4), one can convert this equation into a rate-based equation using an IF neuron model with a threshold x_{th}. With n_i and n_j the number of spikes at input x_i and output y_j respectively, over an accumulation period of T_{acc}, one obtains Eq (2) in Fig. 4. In this equation, spikes can carry a sign and therefore n_i and n_j can be positive or negative. In this case, the neuron has a positive threshold $x_{th+} > 0$ and a negative threshold $x_{th-} < 0$. Note that we assume that upon reaching the threshold, the integration is reset to its value minus $x_{th\, sign(n_j)}$ and not to zero, contrary to [6], in order to avoid losing precision over multiple spikes.

B. Tanh activation function equivalence

We have chosen to use the hyperbolic tangent (tanh) activation function for the classical coding neuron model. For inference only, it can be approximated with a simple saturation h_{sat} function (Eq. 3 in Fig. 4). For ensuring mathematical equivalence to the tanh function, the IF neuron model can be written as Eq. 4 in Fig. 4. Finally, in spike coding, since there is a temporal dimension, one has to decide when to stop sending input spikes, *i.e.* when there is enough spikes for the neural network to make a decision. For this, we introduce the ΔS termination parameter, which is the delta between the most spiking output neuron and the second most spiking one. When ΔS is reached, the inference task is terminated: Fig. 12 shows the activity gain and impact on accuracy.

C. Weight quantization

During the learning, the weights are clamped between [-1,1] and after the learning, they are quantized with 8 levels on the same range, then unary-coded to match the synapse implementation of the circuit.

III. Circuit Architecture

The implemented SNN topology is a single-layer, fully-connected one. Since the objective is to perform inference tasks on the MNIST database, there are 10 output neurons, one per class. To reduce the number of synapses, the images were down-scaled to 12x12 pixels (144 synapses per neuron).

978-1-7281-4032-2/19/$31.00 ©2019 IEEE

A. Synapses implementation

Synapses are implemented using Single Level Cell (SLC) RRAMs [7], i.e. only considering the low and high resistance levels. The structure is of 1T-1R type, with one Access Transistor per cell, as shown in Fig. 5. Multiple cells are put in parallel for enabling various weights [8]. Synaptic quantization experiments done on the learning framework have shown that integer values, ranging from -4 to +4, are a good compromise between classification accuracy and RRAM number. Since we aim at obtaining weighted currents, 4 RRAMs must be used for the positive weights. For the negative weights, the Sign bit could have been encoded using RRAMs as well: however, since a fault-tolerant triple redundancy would have been needed, it was preferred to use 4 additional RRAMs for implementing the negative weights. A synapse is thus composed of 8 RRAMs as shown in Fig. 5. Synapses are arranged in a matrix (Fig. 6) for sharing Word Line, Source Line and Bit Line drivers.

B. Neurons design

The "Integrate and Fire (IF)" analog neurons design was guided by the need for mathematical equivalence with the tanh activation function used in supervised offline learning. The specifications were the following: (1) a stimulation with a synaptic weight equal to ±4 must generate a spike; (2) neurons have to generate positive and negative spikes; (3) they must have a refractory period, during which they cannot emit spikes, but must continue to integrate. As illustrated in Fig. 7, neurons are architected around a MOM 200fF capacitor. Two comparators are used for comparing its voltage level to positive and negative thresholds: the various voltage levels attainable in the capacitor are illustrated in Fig. 8. Since RRAMs must be read with a voltage drop limited to 100mV between its terminals, for preventing setting the devices to LRS, the obtained currents cannot be directly integrated by the neurons: they are copied by current injectors.

C. Top architecture

Top architecture of the circuit is shown in Fig. 9. Spiking pixels addresses are sent through an SPI interface and fill an input FIFO. Addresses are decoded for reading the corresponding synapses. Output neurons spikes are read through the same SPI interface. The circuit micrograph is illustrated in Fig. 10.

IV. Measurement Results

Classification accuracy on the 10K test images of the MNIST database is measured at 84%, with the RRAM programing conditions of Fig. 2. This value must be compared to the accuracy obtained from ideal simulations of 88%, which is limited by the simple network topology (1 layer with 10 output neurons). Fig. 12 plots the network activity in terms of number of input spikes and the accuracy as a function of the difference between the most active and the second most active neuron of the network, ΔS. By changing the ΔS parameter, one can trade off accuracy against activity. The energy dissipation per synaptic event is equal to 3.6 pJ. When accounting for the circuit logic and SPI interface, it amounts to 180 pJ (it could be reduced by optimization of the communication protocol). Measurements show that an image classification needs 136 inputs spikes on average (for ΔS=10): this is less than one spike accumulated per input, leading to a 5x energy gain compared to equivalent Formal coding MAC operations in 130nm node. The synapses of the output neurons' receptive fields are shown in Fig. 11, with the corresponding RRAM matrix. The impact of the RRAM programming conditions and the corresponding percentage of correctly written synapses on classification accuracy is assessed in Table 1. The absence of read disturb to inference tasks is demonstrated in Fig. 13, with 750 million spikes sent without impact.

V. Discussion

The analog neurons and synaptic matrix were also implemented on test scribes in 28nm FDSOI, which are under fabrication. Area of the RRAM matrix is divided by a factor of 36x, as illustrated in Fig. 14, while that of the neurons is divided by a factor 17x. The energy per synaptic event is decreased by 10x. Synaptic density can further be improved by a factor of 4x, by using RRAM as a Multiple Level Cell (MLC) [9]. The cell read current is defined by the programming current. In this implementation, we only need two cells per synapse, one MLC holding the weight and one SLC storing the sign (high and low resistance states for the positive and negative sign respectively). Fig. 15 plots an example of 5 levels per RRAM. The current ranges '1', '2', '3' and '4' store the synaptic weights. The range '0' corresponds to the absence of synaptic connection, it is mapped to the lowest current range. Since the synaptic currents are directly integrated into the analog neurons, the '2', '3' and '4' current ranges should be multiples of the first one. We sent 5 million spikes to test the impact of read disturb during inference (blue line in Fig. 15).

VI. Conclusion

We have demonstrated a fully-functional Spiking Neural Network combining analog neurons and RRAM synapses. The 130nm test chip shows a 5x energy gain compared to an equivalent chip using formal coding (3.6 pJ per spike at synapse level). Moving to the 28nm node (currently under processing) will lead to a 10x energy reduction and a 30x density gain. Synaptic density can be further be improved by a factor of 4x, by using RRAM as a Multiple Level Cells.

A live demonstration of classification of digits drawn on a touch screen interface is running, as illustrated in Fig. 16.

Acknowledgment

This research work was supported by H2020 NeuRAM3 project (687299) and by French ANR via the Carnot funding.

References

[1] S. Ambrogio et al. *Nature*, 2018, vol.558, no.778, pp.60
[2] R. Mochida et al., *VLSI Technology*, pp. 175-176, 2018
[3] P. Merolla et al., *Science*, pp. 668-673, 2014
[4] D. R. B. Ly et al.,*J. Phys. D Appl. Phys*. 2018, 51 444002
[5] https://github.com/CEA-LIST/N2D2
[6] J. Perez-Carrasco et al., *TPAMI*, pp. 2706-2719, 2013
[7] A. Grossi et al., *VLSI Journal*, pp. 2599-2607, 2018
[8] D. Garbin et al., *TED,* vol.62 issue 8, 2015
[9] B. Q. Le et al., *TED*, vol.66 issue 1, 2019

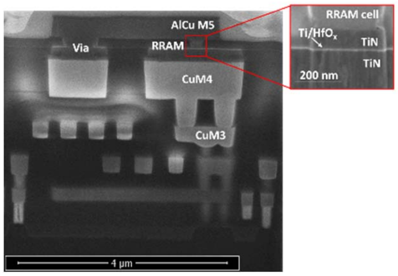
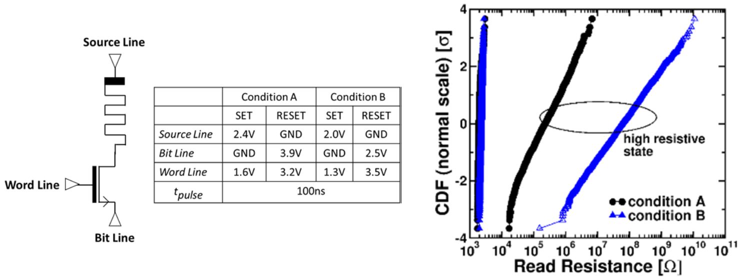


Fig. 1. SEM cross-section of the RRAM cell monolithically integrated on the top of 130nm CMOS



	Condition A		Condition B	
	SET	RESET	SET	RESET
Source Line	2.4V	GND	2.0V	GND
Bit Line	GND	3.9V	GND	2.5V
Word Line	1.6V	3.2V	1.3V	3.5V
t_{pulse}	100ns			

Fig. 2. LRS and HRS cumulative distributions (with programming conditions) measured on 4kbit cells

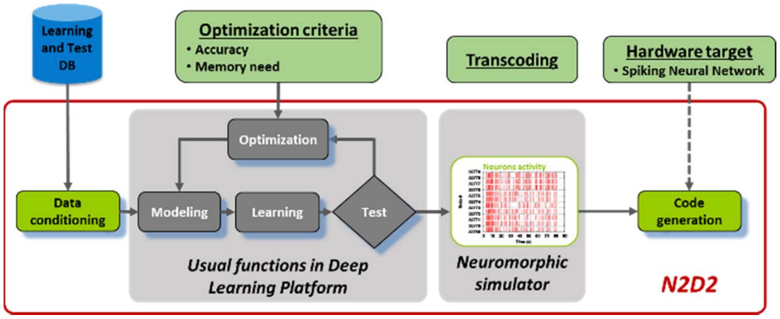
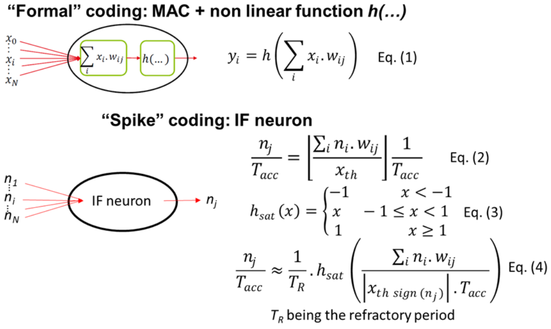


Fig. 3. N2D2 (Neural Network Design and Deployment) learning framework, aimed at (1) doing supervised learning using back-prop algorithm and (2) transcoding into the spike domain



“Formal” coding: MAC + non linear function *h(…)*

$$y_i = h\left(\sum_i x_i . w_{ij}\right) \quad \text{Eq. (1)}$$

“Spike” coding: IF neuron

$$\frac{n_j}{T_{acc}} = \left\lfloor \frac{\sum_i n_i . w_{ij}}{x_{th}} \right\rfloor \frac{1}{T_{acc}} \quad \text{Eq. (2)}$$

$$h_{sat}(x) = \begin{cases} -1 & x < -1 \\ x & -1 \leq x < 1 \\ 1 & x \geq 1 \end{cases} \quad \text{Eq. (3)}$$

$$\frac{n_j}{T_{acc}} \approx \frac{1}{T_R} . h_{sat}\left(\frac{\sum_i n_i . w_{ij}}{\left|x_{th\ sign\,(n_j)}\right| . T_{acc}}\right) \quad \text{Eq. (4)}$$

T_R being the refractory period

Fig. 4. Mathematical equivalence between Formal and Spiking neuron models

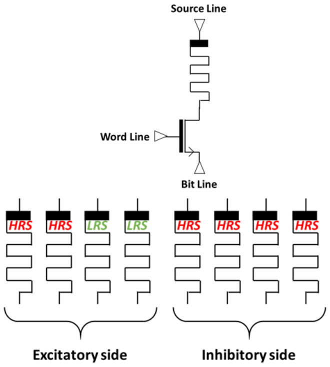


Fig. 5. RRAM-based synapse design

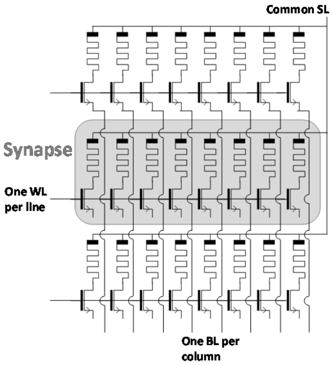


Fig. 6. Schematic of synaptic matrix

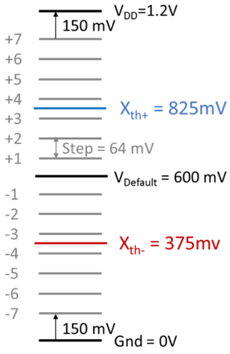


Fig. 8. Voltage levels in membrane

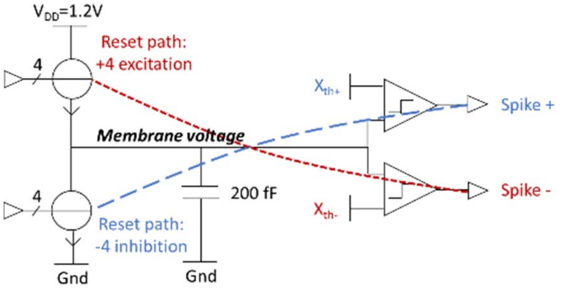


Fig. 7. Neuron schematic, with reset paths for ensuring model equivalence

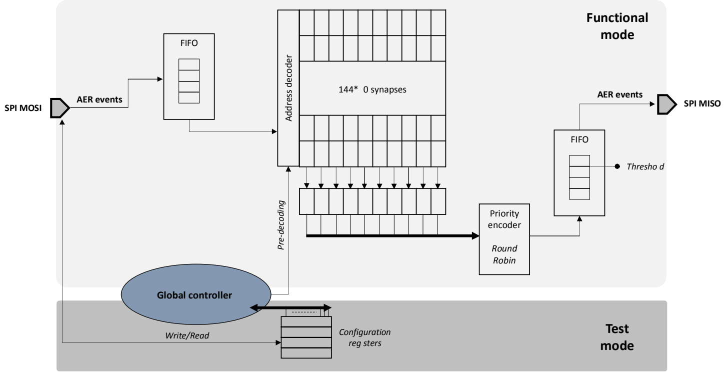
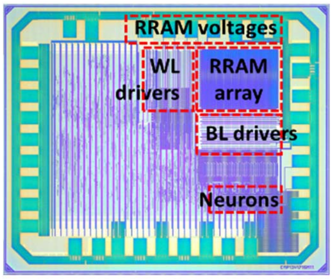


Fig. 9. SNN fully-connected circuit architecture



CMOS process	130 nm
Cu interconnect	5
RRAM number	13,5 K
RRAM configuration	1T-1R
RRAM size	0.5µm x 0.5µm

Fig. 10. Chip micrograph

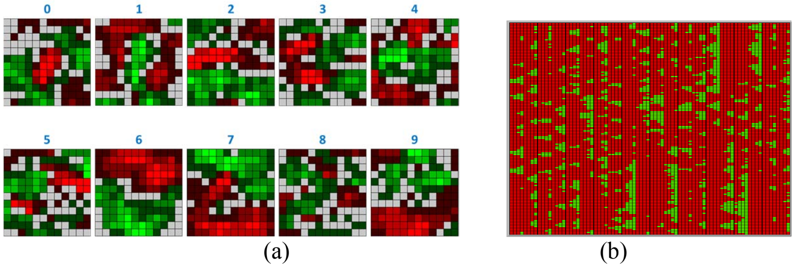


Fig. 11. (a) Learned synapses of the 10 output neurons (Green: excitatory; Red: inhibitory); (b) Corresponding RRAM matrix (Green: LRS state; Red: HRS state)

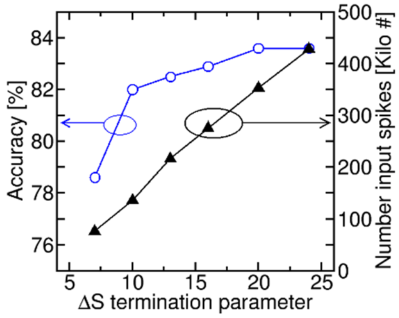


Fig. 12. Accuracy / Activity tradeoff

Condition	Soft Reset (Condition A of Fig. 2)	Strong Reset (Condition B of Fig. 2)	Theoretical
Correct synapses	73,5 %	98,6 %	100 %
Classification accuracy	62.7 %	85,6 %	88 %

Table 1. Impact of RRAM Reset conditions on classification accuracy (batch of 1000 test images)

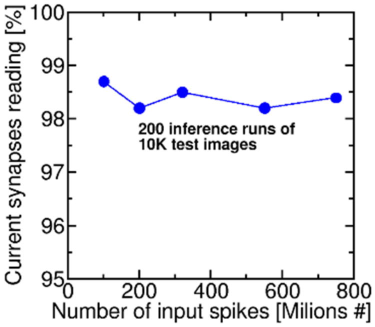


Fig. 13. Read disturb assessment on inference runs

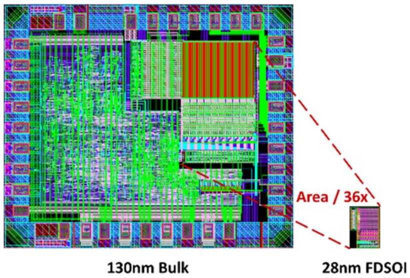


Fig. 14. Layout scaling from 130nm to 28nm node

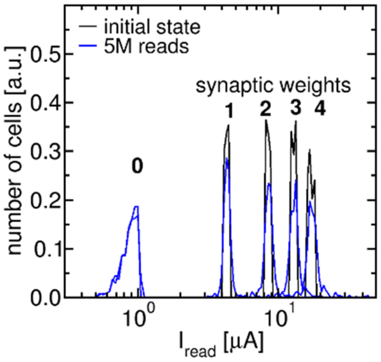


Fig. 15. Multiple valued synapse

	Science 2014 [4]	Micro 2018 [5]	VLSI 2018 [6]	VLSI 2018 [6]	This work	This work scaled	This work scaled + multivalued
Technology	28nm	14nm	40nm	180nm	130nm	28nm	28nm
Coding	Spike	Spike	Formal	Formal	Spike	Spike	Spike
Weight storage	SRAM	SRAM	RRAM	RRAM	RRAM	RRAM	RRAM
Synapses	256M	130M	4M	2M	13.5K	13.5K	-
Synapses/mm²	195K	2000K	1480K	160K	16K	575K	**2300K**
Power	63mW	-	9.9mW	15.8mW	1.5mW	-	-
Energy/syn. event	27pJ	105pJ	N/A	N/A	180pJ	**17,1pJ**	-

Table 2: Comparison to the state-of-the-art

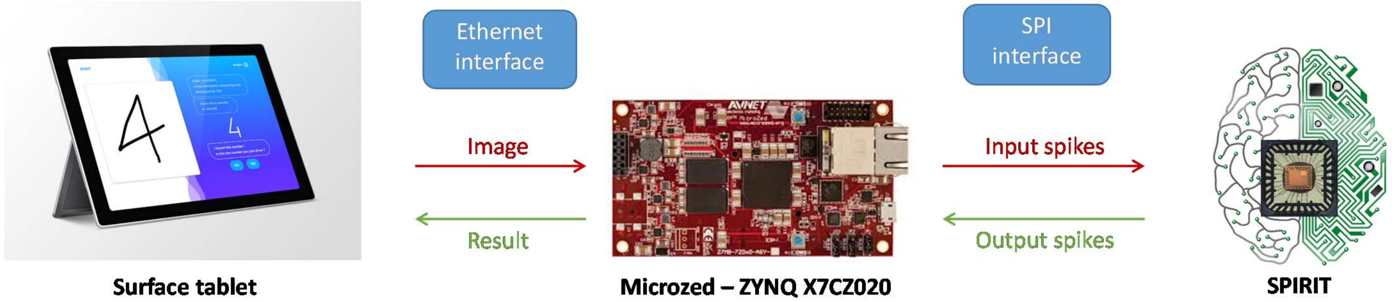


Fig. 16. Demonstration of live handwritten digits classification