

Hardware design of *LIF with Latency* neuron model with memristive STDP synapses

Simone Acciarito, Gian Carlo Cardarilli*, Alessandro Cristini, Luca Di Nunzio
Rocco Fazzolari, Gaurav Mani Khanal, Marco Re and Gianluca Susi

Department of Electronics Engineering, University of Rome "Tor Vergata",
Rome

Laboratory of Cognitive and Computational Neuroscience (UCM-UPM), Centre
for Biomedical Technology. Technical University of Madrid, Madrid, Spain

Abstract

In this paper, the hardware implementation of a neuromorphic system is presented. This system is composed of a Leaky Integrate-and-Fire with Latency (LIFL) neuron and a Spike-Timing Dependent Plasticity (STDP) synapse. LIFL neuron model allows to encode more information than the common Integrate-and-Fire models, typically considered for neuromorphic implementations. In our system LIFL neuron is implemented using CMOS circuits while memristor is used for the implementation of the STDP synapse. A description of the entire circuit is provided. Finally, the capabilities of the proposed architecture have been evaluated by simulating a motif composed of three neurons and two synapses. The simulation results confirm the validity of the proposed system and its suitability for the design of more complex spiking neural networks.

Keywords: *Leaky Integrate-and-Fire with Latency (LIFL), Neuron, Synapse, STDP, Memristor, Neuromorphic System, analog VLSI.*

1. Introduction

In recent years, many efforts have been done in order to reproduce the brain behaviour; this is due to a remarkable capacity of the brain itself to process data

*Corresponding author
Email address: g.cardarilli@uniroma2.it ()

in a highly distributed fashion, requiring relative low-power consumption (~ 12
5 W [1]). Moreover, real-world stimuli are transmitted and processed with high
precision within millisecond timescale. Thus, it should not be surprising that
a huge number of studies have been made over the years to understand and
reproduce the brain operations, involving several scientific application areas:
for example, real-world data classification [2, 3, 4], image recognition [5, 6, 7],
10 speech recognition [8, 9, 10], decision making [11], and rehabilitation [12] (for
an extensive review see Ponulak *et al.* [13]).

Recently, many studies have focused their attention to understand and mimic
brain-like behaviors via hardware, leading to a class of system called *neuromor-*
phic systems [14]. The growing interest in neuromorphic systems is due to wide
15 range of attractive applications, such as real-time and low power spike-based
computing system implementations, compact microelectronic brain-machine in-
terfaces, among others [15].

In our previous works [16], [17], [18], we have introduced a neuron model
called Leaky Integrate-and-Fire with Latency (LIFL). In this model, the spike
20 latency phenomenon has a primary role in the spike generation process. The
spike latency phenomenon has been studied before and used in various applica-
tions discussed and described in [19], [20], [21], [22], [23], [24], [25], [26].

In biological systems, the synapses are tens of thousands times greater in
number than neurons therefore, optimization of electronic synapse design plays
25 a key role. Scaling of synapse area and power consumption are critical issues for
designing a large scale neuromorphic circuit. However, the advancement in nan-
otechnology allows us to exploit today's nanodevices, such as *memristor*, to save
both silicon area and power consumption [27]. Despite memristor was already
theorized in 1971 by Leon O. Chua [28], its "first" physical implementation took
30 place in HP Labs only in 2008 [27]. A memristor represents a two-port passive
element with a charge-dependent resistance, also known as memristance. Under
proper conditions, this property makes the memristor able to mimic not only
the STDP (i.e., Spike-Timing-Dependent Plasticity) behavior but also to enrich
the synaptic dynamics by shaping the STDP learning window [29]. The STDP

35 mechanism is well accepted by the scientific community as a process underlying learning and memory in the brain (for a detailed overview, see [30],[31]). Thus, many studies have been made to exploit memristors with the purpose of realizing plastic synapses within neuromorphic systems [29, 32, 33, 34, 35, 36, 37, 38, 39]. In these works shape-tailored waveform pulses are often employed, limiting the
 40 choice of the neuron model to be used (usually, I&F [40] models type).

To mitigate this aspect, the authors have presented in [41] a novel circuit implementation of a memristor-based synapse able to work with any arbitrary pulse shape. Furthermore, in [42] the authors have presented a PCB circuit emulating the biological spike firing scheme that activates the memristor synapse.
 45 In this work, we present a hardware implementation of the complete LIFL neuron model. Further, an improved synapse driving circuit is presented. The paper is organized as follow: in section 2 we give the theoretical model of LIFL and synapse. A brief review of memristor device is given in section 3. In section 4 we describe the circuit implementation for the neuron and the synapse,
 50 followed by simulation results and discussion of a simple motif in section 5, and finally conclusion of the work.

2. Theoretical models

2.1. Neuron

In this section we considered spiking neural model with the aim of modeling
 55 a silicon neuron. The used model is the LIFL (Leaky Integrate-and-Fire with Latency) [17], [18]. It is characterized by two different behaviors, sub- and suprathreshold, depending if the inner state (S , i.e. the membrane potential) is under or over a spiking threshold (S_{th}), respectively. In subthreshold ($S < S_{th}$, a.k.a. passive mode) the model exhibits a leaky integrator behavior;
 60 whereas, in suprathreshold ($S \geq S_{th}$, a.k.a. active mode), a neuron does not fire instantaneously, but after a continuous-time delay called *time-to-fire*, t_f (i.e., the spike latency of the biological counterpart). The spike latency mechanism allows the strength of the input to be encoded by the spike timing, as observed

in most cortical neurons [43]. The models with a different behavior (i.e., I&F
 65 models) show a lack of information and are not bio-realistic [20].

The neuron model is described by the following equations:

$$S = S_p + P_r P_w - L_d \Delta t, \text{ if } S < S_{th} \quad (1)$$

$$S = S_p + P_r P_w + \frac{(S_p - 1)^2 \Delta t}{1 - (S_p - 1) \Delta t}, \text{ if } S \geq S_{th} \quad (2)$$

$$t_f = \frac{1}{(S - 1)} \quad (3)$$

In these equations, the state S is in the range $[0, \infty)$ (where the lower bound represents the resting state), S_p is the previous state and P_r is the “presynaptic weight” (or spike), representing the output generated by a firing neuron (ideally
 70 a Dirac delta). During the propagation towards the target neuron, this quantity is multiplied by P_w , “postsynaptic weight”, bounded in the range $[0, 1]$. The S_{th} is the spiking threshold, expressed by $1 + d$, where d is chosen in order to have a maximum finite spike latency as in biological systems [19]. Indeed, for $S = S_{th} = 1 + d$, the time-to-fire calculated through Eq. 3 is $t_{f,max} = 1/d$.
 75 Moreover, L_d is a positive quantity representing the linear subthreshold decay (note that when $L_d = 0$ the neuron behaves like a *perfect integrator*), Δt is the temporal distance between two consecutive incoming spikes. The Eq. 3 represents an approximation of the curve obtained through the simulation of a membrane patch stimulated by brief current pulses (0.001 ms of duration).
 80 This approximation was done solving the non linear and differential Hodgkin-Huxley equations [44], [45] and using NEURON simulator [46]. We called this relationship *firing equation*. In Fig. 1, a qualitative comparison between the simulated behavior of the latency and the firing equation is shown.

In order to be effective, an incoming pre-synaptic spike must occur before
 85 the spike generation. Then, the denominator of the fractional term in Eq. 2 is always greater than zero. The fractional term allows the evaluation of the inner state of neuron target when it receives further inputs during the t_f time

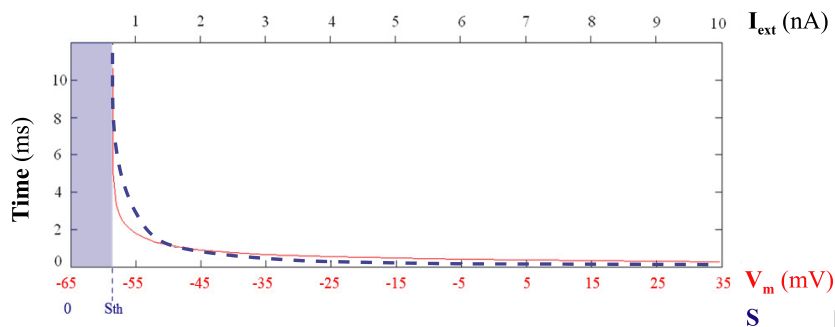


Figure 1: Comparison between Latency (red) and *firing equation* (dotted blue). Modified from [18].

window. Finally, after the firing, the neuron is reset to its resting potential (i.e., $S = 0$) for a time equal to t_{arp} (i.e., absolute refractory period), in which the
 90 neuron remains insensitive to further incoming spikes.

The continuous-time behavior of the model is suitable for event-driven simulation methods, by which it is possible to save overall information [47]. For a detailed description of the model see [18].

2.2. Synapse and STDP

95 Neurons in the network are connected by links (i.e., synapses) characterized by synaptic weights which permits to modify the amplitude of the passing pulses. Synapses can vary according to the activity of the network. Synaptic plasticity is the ability of synapses to strengthen or weaken of the weight over time, in response to increases or decreases in their activity. A well-known type of synaptic
 100 plasticity is based on the precise timings of pre- and post-synaptic spikes, influencing the magnitude and direction of change of the synaptic strength [48]. This rule is referred to as Spike-Timing-Dependent Plasticity (STDP, see [30]), found by Bi & Poo through an experimental protocol. In order to show the general synaptic change trend, a properly elaborated STDP behavior is represented
 105 in Fig. 2. The change of synaptic weight is plotted as a function of the relative timing between pre-synaptic spike arrival and post-synaptic firing. The right dashed blue curve represents the change of synaptic weight when pre-synaptic

spike arrives before the post-synaptic spike is generated; the left curve dashed blue represents the change of synaptic weight when post-synaptic spike is generated before the pre-synaptic spike is arrived. In Fig. 3 Δt is represented, with reference to the synapse $S_{j,i}$. Note that $\Delta t = t_{post} - t_{pre}$.

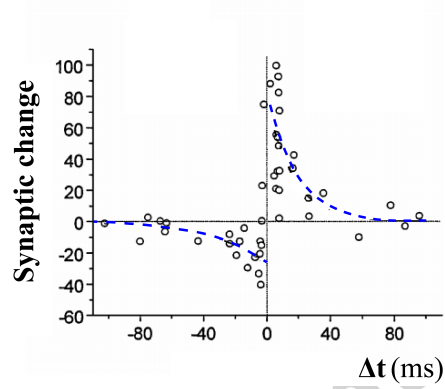


Figure 2: STDP behavior. Long Term Depression LTD ($\Delta t < 0$). Long Term Potentiation LTD ($\Delta t > 0$). Modified by [30]

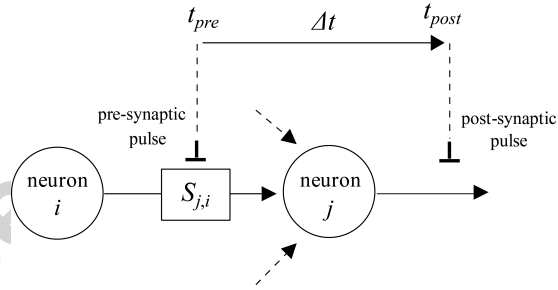


Figure 3: Δt representation.

The STDP behavior can be approximated by Eq. 4, which gives the modification in synaptic strength *i.e.* *synaptic change* for a pre-synaptic arrival time,

Table 1: standard values set for STDP.

$\tau^+ = 16.8 \text{ ms}$
$\tau^- = 33.7 \text{ ms}$
$A_+ = 0.78$
$A_- = -0.27$

t_{pre} , and a post-synaptic firing time, t_{post} :

$$\text{synaptic change} = \begin{cases} A_- \exp \frac{\Delta t}{\tau^-} & \text{for } \Delta t < 0, \\ 0 & \text{for } \Delta t = 0, \\ A_+ \exp \frac{-\Delta t}{\tau^+} & \text{for } \Delta t > 0, \end{cases} \quad (4)$$

115 In the above equation (Eq.4), τ^+ and τ^- are the time constants for potentiation and depression, respectively, and A_+ and A_- are the maximum amplitudes of potentiation and depression, respectively. All these parameters are positive, except A_- .

120 Although STDP varies tremendously across synapse types and brain regions [49], a standard choice for these parameters, obtained by fitting the real case [50], is shown in Table 1.

Note that in some variants of STDP models A_+ and A_- are represented by proper functions in order to keep the synaptic weights in a bio-plausible range of variation.

125 Although experimental studies usually report synaptic change as a fractional change (*i.e.*, $\frac{\Delta w_{j,i}}{w_{j,i}}$) resulting after some number of pairings. In many models it's assumed that changes induced by spike pairings at a particular Δt are absolute changes with units of conductance (e.g.[51]). In this work the latter choice is adopted.

130 **3. Memristor**

Memristor is a two terminal device originally postulated by L.O. Chua in 1971 [28]. The resistance of memristor (memristance) can be controlled by changing the input that can be a voltage or a current, in this manner we have voltage/flux controlled or current/charge controlled memristor. The voltage
 135 controlled memristor can be described by the following equations:

$$I_{MR} = M_G(w, V_{MR}, t)V_{MR} \quad (5)$$

$$w = f(w, V_{MR}, t) \quad (6)$$

Where, I_{MR} , V_{MR} are the current and the voltage input to the memristor, respectively. M_G , M_R represent the memconductance and the memristance, respectively [52]. The current controlled memristor can be described by the following equations:

$$V_{MR} = M_R(w, I_{MR}, t)I_{MR} \quad (7)$$

$$w = f(w, I_{MR}, t) \quad (8)$$

140 Note that w is the internal state parameter of the memristor. The most important characteristic of the memristor is the pinched hysteresis loop (i.e., hysteresis loop passed through origin) in its current-voltage ($I - V$) curve when activated by an alternating signal. This characteristic is able to modify the conductance respect to the input signal change both in magnitude and polarity, and store the
 145 last conductance state even after the device is switched off. These properties are also closely followed by the biological synapse [53], [54] and in this sense, a memristor behaves like a synapse. A synapse is a connection between two neurons in the brain with a plastic/programmable synaptic weight that can be modified to alter the efficiency/strength of signal transmission between neurons
 150 under the influence of transmission itself. For this reason, several memristor based synapse implementations have been proposed in literature [15], [30], [29], [32], [39], [55], [56].

In this work we use a particular memristor model presented in [57] for simulating the proposed system. The Spice model of the memristor used in this experiment is given in [57]. The model is very general and has good flexibility to accommodate a different type of memristor dynamics. Here, we briefly discuss the core equations used by the authors in [57] to developed the model. Values of all the variables used in our experiment are listed in Table 2

The $I - V$ equation for the used memristor is given in Eq. 9

$$\begin{cases} a_1 \times x(t) \times \sinh(bV(t)) & V(t) \geq \theta \\ a_2 \times x(t) \times \sinh(bV(t)) & V(t) < 0 \end{cases} \quad (9)$$

where, $x(t)$ is memristor state variable, which defines the change in resistive state of the memristor device based on the dynamics of the particular device. In the used memristor model the state variable can have any value between 0 and 1. Where,

$$\begin{cases} \text{Lowresistivestate} & x(t) = 1 \\ \text{Highresistivestate} & x(t) = 0 \end{cases}$$

The parameters a_1 , a_2 and b are fitting parameters and their respective values used in our work is given in the Table 2. The rate of change of the state $x(t)$ is given in Eq.10. The change in the state variable depends on two different functions, first $g(V(t))$ and second $f(x(t))$. Function $g(V(t))$ implements the threshold voltage limit for the model used. The second function $f(x(t))$ implements the non linear ion motion also commonly known as non linear dopant drift in the state variable motion. η represents the direction of motion of the state variable with respect to the polarity of input voltage. η could be 1 or -1 , when $\eta=1$ positive voltage above the threshold will increase the value of state variable whereas when $\eta=-1$ positive voltage will decrease the value of state variable.

$$\frac{d(x)}{d(t)} = \eta * g(V(t)) * f(x(t)) \quad (10)$$

Voltage threshold function for the used model is given by the following Eq.3.

$$g(V(t)) = \begin{cases} A_p(\exp V(t) - \exp V_p) & V(t) > V_p \\ -A_n(\exp(-V(t)) - \exp V_n) & V(t) < -V_n \\ 0 & -V_n \leq V(t) \leq V_p \end{cases} \quad (11)$$

Where, V_p and V_n are positive and negative voltage threshold values and A_p and A_n are parameters that defines how fast the device changes its states after the threshold is reached. The values of V_p , V_n , A_p and A_n are listed in the Table

180 2.

The non linear dopant drift function $f(x(t))$ for the used model is given by the Eq.12. Variables x_p and x_n are the points up until which the state variable motion is linear/constant. After this point the state variable motion is govern by a decaying exponential function with decaying rate of u_p or u_n respectively for point after x_p or x_n . The values of these parameters used in our experiment are listed in the Table 2. The variables w_p and w_n are the window functions used to ensure that the function $f(x(t))$ remains within the valid interval [1-0]. The windowing functions are given in Eq.13 and Eq. 14.

$$f(x(t)) = \begin{cases} \exp(-u_p)(x - x_p) * (w_p(x, x_p)) & x \geq x_p \\ 1 & x < x_p \\ \exp(u_n)(x + x_n - 1) * (w_n(x, x_n)) & x \leq -x_n \\ 1 & x > 1 - x_n \end{cases} \quad (12)$$

$$w_p(x, x_p) = \frac{(x_p - x)}{(1 - x_p)} + 1 \quad (13)$$

$$w_n(x, x_n) = \frac{x}{(1 - x_n)} \quad (14)$$

4. Circuit implementation

190 In this section, we start to describe the overall system, composed of the neuron and synapse models. All the schematics and the simulations are performed

Table 2: Parameters use in this model.

Parameter	Value
$A_p = A_n$	4000
x_p	0.3
x_n	0.5
α_p	1
α_n	5
$a_1 = a_2$	0.17
b	0.05
V_p	0.16 [V]
V_n	-0.15 [V]

in PSpice environment.

4.1. Neuron

Here we illustrate the circuit implementation of the neuron model. As shown
 195 in Fig. 4, it is composed of a certain number of logical sub-systems:

1. Integrator circuit (I): composed of the input stage of an OTA.
2. Internal State (IS): it consists of RC group.
3. Minimum Threshold (mT): an input stage of a differential amplifier.
4. Non Linear Element (NLE): common-source amplifiers, peak detectors
 200 (diode and capacitor), shunt, and voltage translators.
5. Latency Generation (LG): a ramp generator, a monostable and an adder.
6. Maximum Threshold (MT): input stage of a differential amplifier.
7. Pulse Generator (PG): a monostable and an output buffer.
8. Refractory (R): a peak detector with loss (diode and capacitor and resis-
 205 tor) and a buffer.

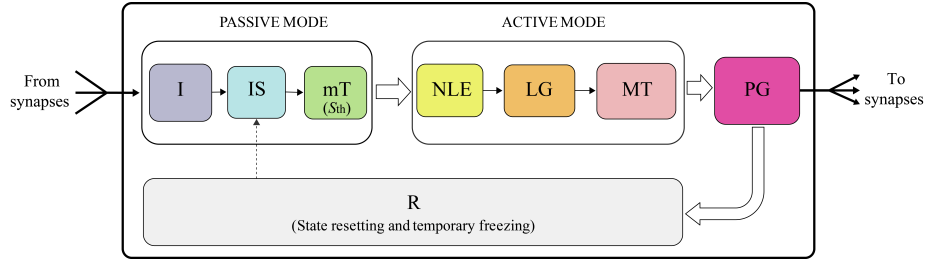


Figure 4: Logical Block of the overall circuit. I: integrator block; IS: internal state block; mT: minimum threshold block; NLE: non-linear element block; LG: latency generation block; MT: maximum threshold block; PG: pulse generator block; R: refractory block.

The functionality of the circuit is described the case of a pre-synaptic spike (or pulse) able (or not) to trigger a post-synaptic spike.

In this case, when the input pulse arrives, the value of voltage of the *internal state* (light blue block 4) instantaneously changes. If this value is over the threshold (i.e., over the minimum threshold), the *latency generation* block is triggered. When the latter reaches the maximum value, a one shot component (dark pink block in Fig. 4) generates a pulse (i.e., the post-synaptic spike). Moreover, at the same time the *refractory* block (gray in Fig. 4) will be activated and other pulses will have no effect to the internal state. Whereas, if the internal state is under the threshold, it will discharge and no post-synaptic spike will be generated. As described in section 2.1, the neuron has two behavioral modes, passive and active described by Eq. 1 and Eq. 2 respectively. The minimum internal state threshold (S_{th}) divides the active mode from passive mode and this is the first threshold in our model (green block in Fig.4). When in active mode, the neuron does not fire instantaneously, but waits for an certain amount of time called time to fire (t_f) (see Eq. 3) before to fire. This is the second threshold in our model (pink block in Fig.4). Therefore, these two thresholds were used in our model to accommodate S_{th} and t_f .

In Fig.5 is shown the latency/input voltage characteristic of the simulated neuron, compared with the ideal one (i.e., *firing equation*, see sect. 2.1). Note that we have considered the effects of latency in the input-output response for

different input voltage values. From the simulation result shown in Fig.5 , the trend approximately follows the ideal curve (i.e., a branch of rectangular hyperbola).

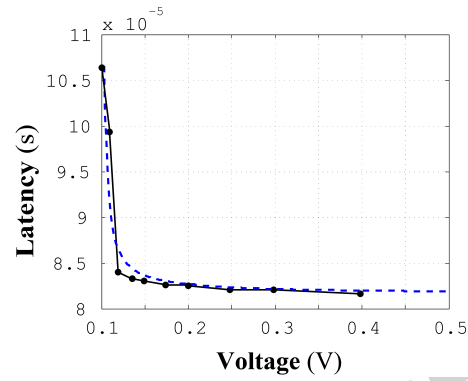


Figure 5: Latency behavior simulated using PSpice (black) compared with the ideal one (dotted blue). The latter is a rectangular hyperbola which parameters are properly tuned to that of LIFL neuron model [17].

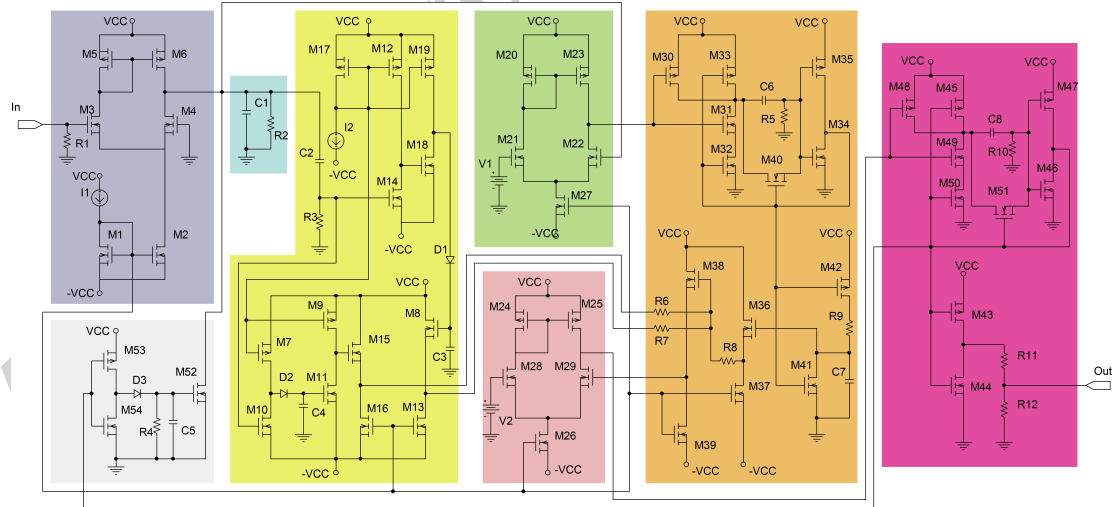


Figure 6: Expanded scheme of overall neuron circuit of Fig. 4. The color used in this figure are coherent with those of Fig. 4.

230 In Fig. 6 is shown the complete PSpice model of the neuron, realized cascading the blocks shown in Fig. 4.

Table 3: Circuit complexity in terms of used devices for neuron.

Devices	Number
MOS transistors	54
Resistors	12
Capacitors	8
Current sources	2
Supply voltages	2

Finally, in Tab. 3 the number of devices used in the circuit are listed.

The resulting circuit shows low complexity associate with good bio-plausibility (for instance, like FitzHugh Nagumo model [43], [58]). The number of transistors is instead comparable with other implementations [59], [60], [61], [62] that do not present the spike latency generation. Moreover, thanks to the implementation of subthreshold integration, refractoriness and spike latency behavior, this model mimics quite well the biological counterpart.

4.2. Synapse and STDP

The logic diagram of the STDP system is illustrated in Fig. 7. While more details about the circuit implementation are given in Fig. 9.

The system of Fig. 7 is composed of two main logic blocks. The first one (dashed- dot black) is the synapse (i.e., the memristor) and the second one (dashed gray) is the synapse weight update system (i.e., the system that changes the memristor conductance according to Eq. 4). In addition there are two modes of operation.

1. Normal Mode: In this mode, the system works as a synapse. The signal (pre-pulse) enters through terminal Sin2 and exits out through S1 (close) and memristor. In this case, the switches S2, S3, S4, S5, S6 are open and the other blocks are off.
2. Synapse Weight Update: In this mode, the system updates the synaptic weight (memristor conductance) according to the Spike-Timing-Dependent

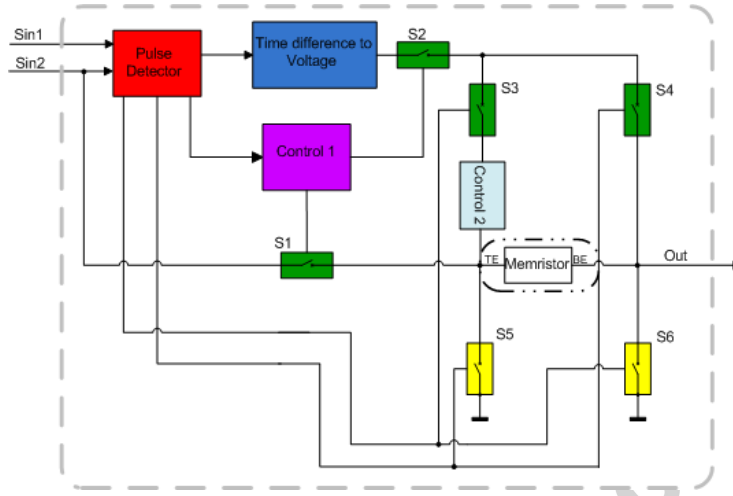


Figure 7: Logic diagram of overall system for STDP. Pulse Detector (red): it detects the pulse time order, Time difference to voltage (blue); it converts the time difference between the two pulses in an equivalent voltage, Control 1 (fuchsia); it allows the synapse weight update, S1, S2, S3, S4, switches (green), Control 2 (light blue): it mitigates the asymmetry of the memristor, S5, S6 switches (yellow).

Plasticity rule. The pulse detector closes S3 and S6 (i.e., it connects TE and BE terminal to the switch S2 and ground, respectively) and opens S4 and S5 allowing the increasing of memristor conductance (pre pulse arrives before post pulse). In the other case (post pulse arrives before the pre pulse), it closes S4 and S5 (i.e., it connects BE and TE terminal to the switch S2 and ground, respectively) and opens S3 and S6, allowing the decreasing of memristor conductance. The "Time difference to Voltage" block, converts the difference of time between the pre pulse and post pulse in an equivalent voltage. "Control 1" block, opens S1 and closes S2 allowing the changing of memristor conductance. Finally, "Control 2" block, mitigates the asymmetry of I-V curve of memristor.

In Fig. 9, the circuit implementation of the above system is shown. Its simulation behavior is shown in Fig. 10. The circuit is composed of two basic blocks:

1. Time Sensing Block-Control (TSB-C): This block detects the order of arrival of the two pulses (pre and post-pulses) and generates the signals for the control of the change of the mem-conductance.
- 270 2. Time Sensing Block (TSB): This block generates a voltage proportional to the time difference of pre and post-pulses.

The description of the circuit in Fig. 9 can be done taking into account the considerations developed for the scheme of Fig. 7. NMOS transistors M8, M9, M10, M11, M14, M15 M16 and M17 of TSB are off while the pulse enters in
 275 “IN” and goes through the memristor (MEM,) to “OUT”. The increasing of the conductance of MEM is obtained by connecting the “BE” terminal to ground and “TE” to V(I) via M10. On the other hand, a decreasing of the conductance can be obtained connecting “TE” to ground and “BE” to V(I) via M17. In other words, V(B) allows the voltage stored to be transferred to the memristor. The
 280 couple M10 and M17 connects the MEM to C3 and are controlled by the voltages V(H) and V(J). The couple M8 and M11 allows the grounding of “TE” terminal of MEM, whereas M16 and M15 allows the grounding of “BE” terminal. V(A) is the signal for the change of mem-conductance. It is generated in TSB-C block. In this block the DB (i.e., a differential amplifier) is able to detect the arriving
 285 time of the two pulses, charging or discharging the capacitor C1, turning on M1, M4 and turning off M2, M3, when the presynaptic pulse arrives before post synaptic pulse (i.e., potentiation case). In this case, when the post pulse arrives, TRIG block (R2, C1, U5) triggers the MONO (i.e, monostable) that generates the signal V(A) (see Fig.8) opening the Buffer I/O and allowing the change of
 290 memristor conductance.

Due to the asymmetry of memristor curve, the positive conductance change (LTP) is higher than the negative one (LTD) for the same value of voltage applied to the device, Fig. 10. The device set (R4, R5, R7, C2, D1, Vref) is the mitigation circuit that controls the value of voltage applied to the device for
 295 the LTP case, needed to obtain a more bio-realistic curve as in [30]. Moreover, it is possible to do a tuning of the conductance value via the voltage generator

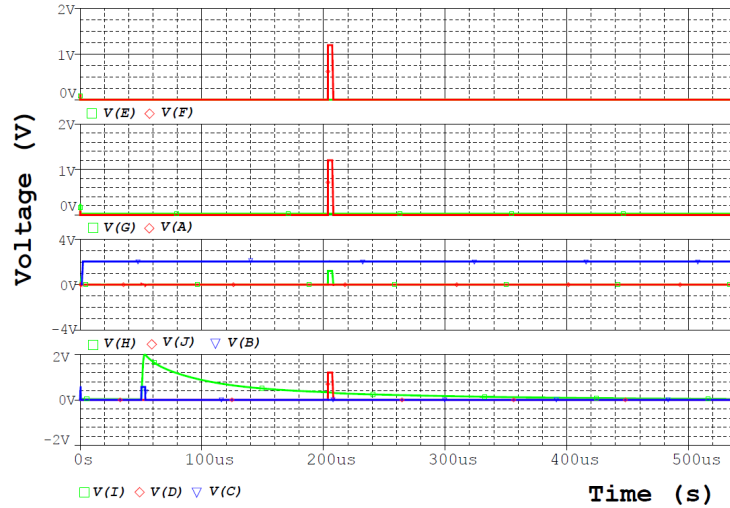


Figure 8: Voltage-Time diagram for driving the TSB circuit show in Fig. 9, for “potentiation” case.

V_{ref} and supply voltage V_{cc} applied to transistors M5, M7. The MEM is the memristor-based synapse. The third terminal of memristor (“XSV” in Fig. 9) is only used for plotting the internal state variable [57].

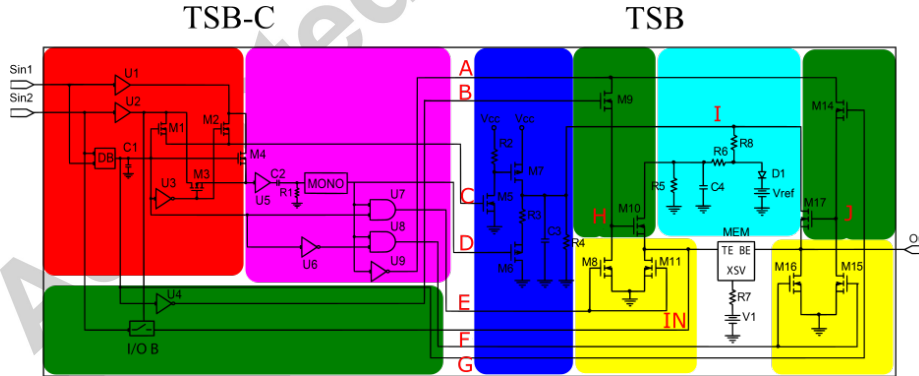


Figure 9: Overall circuit scheme using the colors of Fig. 7. Sin1 and Sin2 are the input of the circuit, Out is the output of the system.

300 We remark that this circuit does not depend on the waveform of the pulses as in [32], [39], [55], [63], but it is thought to realize a spike timing dependent

Table 4: Circuit complexity in terms of used devices for synapse.

Devices	Number
MOS transistors	68
Resistors	13
Capacitors	6
Supply voltages	2

circuit based on memristors, allowing a timing-sensitive behavior. The STDP behavior obtained by PSpice simulations of the whole circuit is shown in Fig. 10. For further details see [41].

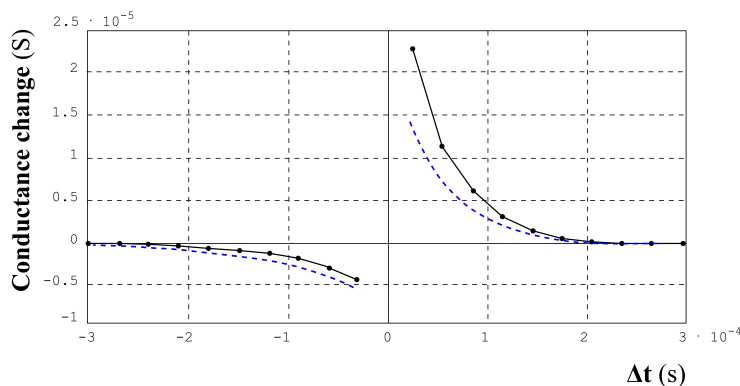


Figure 10: STDP-like behavior simulated using PSpice (black) compared with the ideal one (dotted blue, see Fig. 2) . The latter has been traced using $\frac{A_+}{A_-}$ and $\frac{\tau_{au}^+}{\tau_{au}^-}$ ratios obtained from standard STDP parameters, listed in table 1.

305 Tab. 4 gives the number of devices used in the circuit.

5. Simulation results and discussion

In order to show the behavior of the overall circuits a motif system is presented. It is a neural network with two input neurons for sensory and one output neuron for the association and the decision, as shown in Fig. 11A. This three
 310 neurons and two synapses system is widely used for describing and analyzing

the behaviour of neural circuits [39], [64],[65]. Moreover, the low complexity of the system allows to perform an electrical simulation.

All simulations are performed using Pspice. For both circuits we have used a CMOS 90nm 6 Metal Copper low-K level 3 technology.

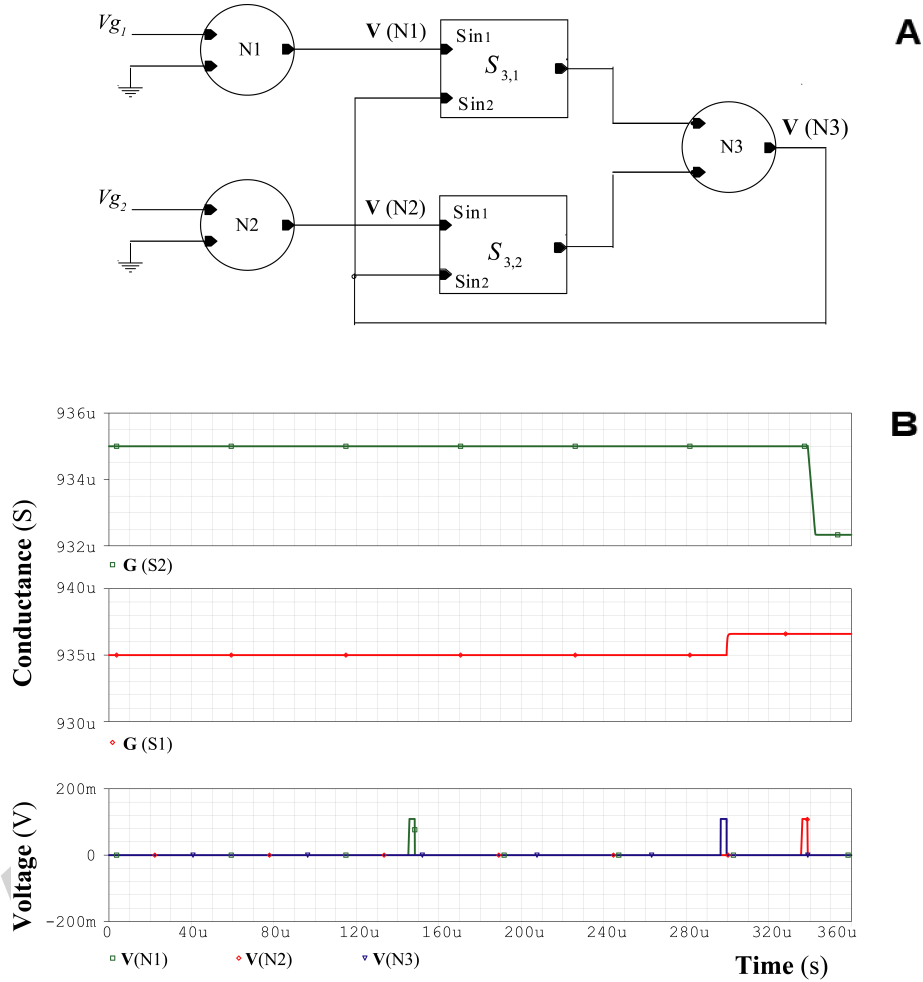


Figure 11: (A) Simple simulated circuit composed of 3 neurons (N1, N2, N3) and 2 synapses (S1, S2). In order to trigger the neurons N1 and N2, two impulsive generators are necessary (i.e., V_{g1} and V_{g2}). (B) bottom: voltage pulses; middle: LTP behavior for S1; top: LTD behavior for S2.

315

The Fig. 11(A) shows a simplified diagram of the network motif used for

the simulation. The result of simulation is shown in Fig. 11(B). The obtained results showed that the circuit is able to recognize the spike time order (i.e., pre-synaptic spike before post-synaptic spike, and vice versa). Further, it is able to process the time difference in an equivalent voltage, and apply this voltage value in order to change the state of memristor synapse. In this way, the LTP (i.e., Long Term Potentiation) and LTD (i.e., Long Term Depression) obtained behaviors are similar to those shown in Bi & Poo [30].

As described in section "Synapse and STDP" (4.2), the proposed implementation mitigates the main limitations present in the literature: the use of specific spike shapes [32], [39], [55], the presence of a relative low excursion for the conductance change [32], [63].

In case of spiking neural networks composed of a large number of neurons, the very high number of synapses makes them the main source of power consumption. For this reason, the synapse driving circuit presented in [41] has been improved taking into account the specific characteristics of our application. The most significant changes are the following.

1. DB: In the original block (Fig.12A) the main source of power consumption is the current mirror (M5, M6) for the biasing of differential amplifier. Since the output can assume only two states (binary output), we can reduce this bias current using a transistor in diode configuration (M5), as depicted in Fig.12B.
2. TRIG: this circuit drives the MONO block in Fig.9. It represents the main source of power consumption is the circuit developed in [41]. In particular, the biasing of level shifter (M5, M6) Fig.12C requires large current. Since it drives a monostable circuit, its output assumes two levels (trigger on, trigger off). As a consequence we can substitute the circuit formed of R,C and a level shifter (M5, M6) with a voltage divider (M5, M6) and capacitor (C) Fig.12D.

The overall results are shown in table 5.

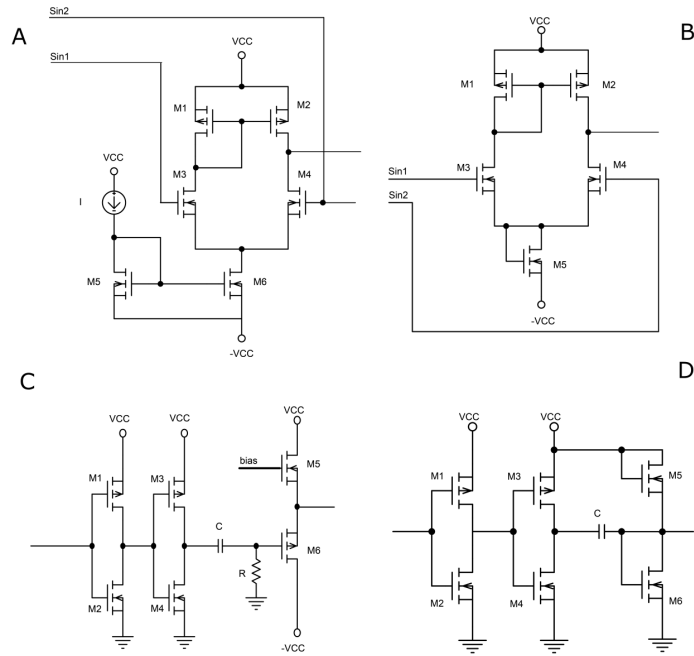


Figure 12: Circuit for power reduction. Subfigure a), c) are the previous version presented in [41]. The other subfigure b), d) are the new ones proposed in this work.

Table 5: Comparison of a synapse driving circuit power consumption with previous work.

Work	Power	Area
[41]	9.58 mW	500 μm^2
Present paper	0.54 mW	480 μm^2

345 6. Conclusion

In this work, we have described the implementation of a neuromorphic system able to mimic some relevant bio-inspired neuron features, such as integration, refractoriness, spike latency, and STDP-like behavior. The number of transistors used per single neuron in our design is comparable with other implementations, maintaining the basic bio-inspired behaviors. Moreover, we have shown a new method for driving memristor as synapse, independently of the spike pulse shape. Also, this driving system together with the memristor guar-

350

antees a good approximation of the synapse STDP-like behavior. Even though we have considered a simple network due to the heaviness of PSpice simulations, we have obtained good results in terms of bio-plausible characteristics. In view of an on-chip implementation of a quite large neuromorphic system, we are currently improving the circuits related to both neuron and synapse driving circuit in order to further reduce silicon area and power consumption as well.

References

- [1] R. Sarpeshkar, Analog versus digital: extrapolating from electronics to neurobiology, *Neural Comput.* 10 (7) (1998) 1601–1638.
- [2] S. M. Bohte, J. N. Kok, H. La Poutré, Error-backpropagation in temporally encoded networks of spiking neurons, *Neurocomputing* 48 (1–4) (2002) 17–37.
- [3] S. M. Bohte, H. La Poutré, J. N. Kok, Unsupervised clustering with spiking neurons by sparse temporal coding and multilayer RBF networks 13 (2) (2002) 426–435.
- [4] A. Belatreche, L. P. Maguire, M. McGinnity, Q. X. Wu, A method for supervised training of spiking neural networks, in: *IEEE Conference on Cybernetics Intelligence – Challenges and Advances, CICA’2003*, 2003, pp. 39–44.
- [5] S. J. Thorpe, A. Delorme, R. VanRullen, Spike-based strategies for rapid processing, *Neural Netw.* 14 (6–7) (2001) 715–726.
- [6] R. Guyonneau, R. VanRullen, S. J. Thorpe, Temporal codes and sparse representations: a key to understanding rapid processing in visual system, *J. Physiol., Paris* 98 (4–6) (2004) 487–497.
- [7] L. Perrinet, M. Samuelides, S. J. Thorpe, Sparse spike coding in an asynchronous feed-forward multi-layer neural network using matching pursuit, *Neurocomputing* 57 (2004) 125–134.

- 380 [8] J. J. Hopfield, C. D. Brody, What is a moment? “Cortical” sensory integration over a brief interval, *Proc. Natl. Acad. Sci. USA* 97 (25) (2000) 13919–13924.
- [9] D. Verstraeten, B. Schrauwen, D. Stroobandt, J. Van Campenhout, Isolated word recognition with the Liquid State Machine: a case study, *Infor. Process. Lett.* 95 (6) (2005) 521–528.
- 385 [10] R. Gütig, H. Sompolinsky, Time-warp-invariant neuronal processing, *PLoS Biol.* 7 (7).
- [11] C. Glackin, L. McDaid, L. Maguire, H. Sayers, Implementing fuzzy reasoning on a spiking neural network, in: *Artificial Neural Networks - ICANN 2008*, Vol. 5164 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2008, pp. 258–267.
- 390 [12] R. Rom, J. Erel, M. Glikson, R. A. Lieberman, K. Rosenblum, O. Binah, R. Ginosar, D. L. Hayes, Adaptive cardiac resynchronization therapy device based on spiking neurons architecture reinforcement learning scheme 18 (2) (2007) 542–550.
- 395 [13] F. Ponulak, A. Kasiński, Introduction to spiking neural networks: Information processing, learning and applications, *Acta Neurobiol. Exp.* 71 (4) (2011) 409–433.
- [14] C. Maed, *Analog VLSI and Neural Systems*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.
- 400 [15] G. Indiveri, et al., Neuromorphic silicon neuron circuits, *Front. Neurosci.* 5 (73).
- [16] M. Salerno, G. Susi, A. Cristini, Accurate latency characterization for very large asynchronous spiking neural networks, in: *BIOINFORMATICS 2011 - Proceedings of the International Conference on Bioinformatics Models, Methods and Algorithms*, SciTePress, Rome, IT, 2011, pp. 116–124.
- 405

- [17] G. C. Cardarilli, A. Cristini, L. Di Nunzio, M. Re, M. Salerno, G. Susi, Spiking neural networks based on LIF with latency: Simulation synchronization effects, in: 2013 Asilomar Conference on Signals, Systems and Computers, IEEE, Pacific Grove, CA, USA, 2013, pp. 1838–1842.
- [18] A. Cristini, M. Salerno, G. Susi, A continuous-time spiking neural network paradigm, in: Advances in Neural Networks: Computational and Theoretical Issues, Springer International Publishing, 2015, pp. 49–60. doi:10.1007/978-3-319-18164-6_6.
- [19] R. FitzHugh, Mathematical models of threshold phenomena in the nerve membrane, *Bull. Math. Biol.* 17 (4) (1955) 257–278.
- [20] E. M. Izhikevich, Dynamical systems in neuroscience: the geometry of excitability and bursting, Computational neuroscience, MIT Press, Cambridge, Mass., London, 2007.
- [21] G. Zheng, A. Tonnelier, Chaotic solutions in the quadratic integrate-and-fire with adaptation, *Cogn. Neurodyn.* 3 (3) (2009) 197–204.
- [22] E. M. Izhikevich, Hybrid spiking models, *Phil. Trans. R. Soc. A.* 368 (1930).
- [23] H. T. Chen, K. T. Ng, A. Bermak, M. K. Law, D. Martinez, Spike latency coding in biologically inspired microelectronic nose, *IEEE Transactions on Biomedical Circuits and Systems* 5 (2) (2011) 160–168. doi:10.1109/TBCAS.2010.2075928.
- [24] J. Al Yamani, F. Boussaid, A. Bermak, D. Martinez, Glomerular latency coding in artificial olfaction, *Frontiers in Neuroengineering* 4 (18). doi:10.3389/fneng.2011.00018.
URL <http://www.frontiersin.org/neuroengineering/10.3389/fneng.2011.00018/abstract>
- [25] T. Gollisch, M. Meister, Rapid neural coding in the retina with relative spike latencies, *Science* 319 (5866) (2008) 1108–1111. arXiv:http://

- science.sciencemag.org/content/319/5866/1108.full.pdf, doi:10.
435 1126/science.1149639.
URL <http://science.sciencemag.org/content/319/5866/1108>
- [26] B. Fontaine, H. Peremans, Bat echolocation processing using first-spike latency coding, *Neural Networks* 22 (10) (2009) 1372 – 1382. doi:<http://dx.doi.org/10.1016/j.neunet.2009.05.002>.
440 URL <http://www.sciencedirect.com/science/article/pii/S0893608009000914>
- [27] D. B. Strukov, G. S. Snider, D. R. Stewart, R. S. Williams, The missing memristor found, *Nature* 453 (2008) 80–83.
- [28] L. O. Chua, Memristor – the missing circuit element 18 (5) (1971) 507–519.
- 445 [29] T. Serrano-Gotarredona, T. Masquelier, T. Prodromakis, G. Indiveri, B. Linares-Barranco, STDP and STDP variations with memristors for spiking neuromorphic learning systems, *Front. Neurosci.* 7 (2).
- [30] G.-Q. Bi, M.-M. Poo, Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type, *J. Neurosci.* 18 (24) (1998) 10464–10472.
450
- [31] H. Markram, W. Gerstner, P. J. Sjöström, Spike-timing-dependent plasticity: a comprehensive overview, *Front. Synaptic Neurosci.* 4 (2).
- [32] C. Zamarreño Ramos, L. A. Camuñas Mesa, J. A. Pérez-Carrasco, T. Masquelier, T. Serrano-Gotarredona, B. Linares-Barranco, On spike-timing-dependent plasticity, memristive devices, and building a self-learning visual cortex, *Front. Neurosci.* 5 (26).
455
- [33] G. Indiveri, B. Linares-Barranco, R. Legenstein, G. Deligeorgis, T. Prodromakis, Integration of nanoscale memristor synapses in neuromorphic computing architectures, *Nanotechnology* 24.
460 URL <http://dx.doi.org/10.1088/0957-4484/24/38/384010>

- [34] Y. V. Perchin, M. Di Ventra, Memcapacitive neural networks, *Electron. Lett.* 50 (3) (2014) 141–143.
- [35] J. Bill, R. Legenstein, A compound memristive synapse for statistical learning through STDP in spiking neural networks, *Front. Neurosci.* 8 (414).
- 465 [36] B. Guo, Y. Cai, Y. Pan, Z. Zhang, Associative learning based on symmetric spike time dependent plasticity, in: 12th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), IEEE, Guilin, China, 2014, pp. 1–3.
- [37] W. He, K. Huang, N. Ning, K. Ramanathan, G. Li, Y. Jiang, Y. Sze, L. Shi,
470 R. Zhao, J. Pei, Enabling an integrated rate-temporal learning scheme on memristor, *Sci. Rep.* 4 (4755).
- [38] G. Lecerf, J. Tomas, S. Boyn, S. Girod, A. Mangalore, J. Grollier, S. Saighi, Silicon neuron dedicated to memristive spiking neural networks, in: 2014 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE, Melbourne, Australia, 2014, pp. 1568–1571.
475
- [39] X. Wu, V. Saxena, K. Zhu, Homogeneous spiking neuromorphic system for real-world pattern recognition 5 (2) (2015) 254–266.
- [40] L. F. Abbott, Lapique’s introduction of the integrate-and-fire neuron model, *Brain Research Bulletin* 50 (5–6) (1999) 303–304.
- 480 [41] S. Acciarito, A. Cristini, L. D. Nunzio, G. M. Khanal, G. Susi, An a vlsi driving circuit for memristor-based stdp, in: 2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME), 2016, pp. 1–4. doi: 10.1109/PRIME.2016.7519503.
- [42] G. Khanal, Synaptic behaviour in zno-rgo composites thin film memristor,
485 *Electronics Letters*.
URL <http://digital-library.theiet.org/content/journals/10.1049/el.2016.3655>

- [43] E. M. Izhikevich, Which model to use for cortical spiking neurons? 15 (5) (2004) 1063–1070.
- 490 [44] A. L. Hodgkin, A. F. Huxley, Action potentials recorded from inside a nerve fibre, *Nature* 144 (1939) 710–711.
- [45] A. L. Hodgkin, A. F. Huxley, A quantitative description of membrane current and application to conduction and excitation in nerve, *J. Physiol.* 117 (4) (1952) 500–544.
- 495 [46] M. L. Hines, N. T. Carnevale, The neuron simulation environment, *Neural computation* 9 (6) (1997) 1179–1209.
- [47] R. Brette, et al., Simulation of networks of spiking neurons: A review of tools and strategies, *J. Comput. Neurosci.* 23 (3) (2007) 349–398.
- [48] D. B. Sinha, N. M. Ledbetter, D. L. Barbour, Spike-timing computation properties of a feed-forward neural network model, *Frontiers in Computational Neuroscience* 8 (5). doi:10.3389/fncom.2014.00005.
URL http://www.frontiersin.org/computational_neuroscience/10.3389/fncom.2014.00005/abstract
- 500 [49] L. F. Abbott, S. B. Nelson, Synaptic plasticity: taming the beast, *Nat. Neurosci.* 3 (2000) 1178–1183.
- 505 [50] A. Viriyopase, I. Bojak, M. Zeitler, S. Gielen, When long-range zero-lag synchronization is feasible in cortical networks, *Frontiers in Computational Neuroscience* 6 (49). doi:10.3389/fncom.2012.00049.
URL http://www.frontiersin.org/computational_neuroscience/10.3389/fncom.2012.00049/abstract
- 510 [51] M. A. Farries, A. L. Fairhall, Reinforcement learning with modulated spike-timing-dependent synaptic plasticity, *J. of Neurophysiology* 98 (6) (2007) 3648–3665.

- [52] L. O. Chua, S. M. Kang, Memristive devices and systems, in: Proceedings of IEEE, IEEE, 1976, pp. 209 – 223.
- [53] Y. Li, et al., Activity-dependent synaptic plasticity of a chalcogenide electronic synapse for neuromorphic systems, *Sci. Rep.* 4.
- [54] T. Ohno, et al., Short-term plasticity and long-term potentiation mimicked in single inorganic synapses, *Nat. Mater.* 10 (2011) 591–595.
- [55] T. Serrano-Gotarredona, B. Linares-Barranco, Design of adaptive nano/cmos neural architectures, in: 19th IEEE International Conference on Electronics, Circuits and Systems (ICECS), 2012, pp. 949–952.
- [56] G. S. Snider, Spike-timing-dependent learning in memristive nanodevices, in: IEEE International Symposium on Nanoscale Architectures, NANOARCH 2008, 2008, pp. 85–92.
- [57] C. Yakopcic, T. M. Taha, G. Subramanyam, R. E. Pino, S. Rogers, A memristor device model, *IEEE Electron Device Letters* 32 (10) (2011) 1436–1438.
- [58] R. FitzHugh, Impulses and physiological states in theoretical models of nerve membrane, *Biophysical journal* 1 (6) (1961) 445.
- [59] V. Rangan, A. Ghosh, V. Aparin, G. Cauwenberghs, A subthreshold avlsi implementation of the izhikevich simple neuron model, in: 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology, IEEE, 2010, pp. 4164–4167.
- [60] A. S. Demirkol, S. Ozoguz, A low power vlsi implementation of the izhikevich neuron model, in: New Circuits and Systems Conference (NEWCAS), 2011 IEEE 9th International, IEEE, 2011, pp. 169–172.
- [61] S. Binczak, A. S. T. Nguetcho, S. Jacquir, J. M. Bilbault, V. B. Kazantsev, Active spike responses of analog electrical neuron: Theory and experiments,

- 540 in: Proceedings of 2010 IEEE International Symposium on Circuits and
Systems, 2010, pp. 2550–2553. doi:10.1109/ISCAS.2010.5537112.
- [62] A. Petrovas, S. Lisauskas, A. Iepikas, Investigation of microcontroller based
model of fitzhugh-nagumo neuron, in: MECHATRONIKA, 2012 15th In-
ternational Symposium, 2012, pp. 1–4.
- 545 [63] W. Xinyu, V. Saxena, K. Zhu, S. Balagopal, A cmos spiking neuron for
brain-inspired neural networks with resistive synapses and in situ learn-
ing, IEEE Transactions on Circuits and Systems II: Express Briefs 62 (11)
(2015) 1088–1092.
- [64] Y. V. Pershin, M. Di Ventra, Experimental demonstration of associative
550 memory with memristive neural networks, Neural Networks 23 (7) (2010)
881–886.
- [65] X. Wu, V. Saxena, K. Zhu, S. Balagopal, A cmos spiking neuron for brain-
inspired neural networks with resistive synapses and in situ learning, IEEE
Transactions on Circuits and Systems II: Express Briefs 62 (11) (2015)
555 1088–1092. doi:10.1109/TCSII.2015.2456372.