Hardware spiking neurons design: analog or digital?

A. Joubert* and B. Belhadj* and O. Temam† and R. Héliot*†
* CEA-LETI, Minatec Campus, 17 rue des Martyrs, 38054 Grenoble, France.
† INRIA Saclay Ile-de-France, rue Jean Rostand, 91893 Orsay Cedex, France.
rodolphe.heliot@cea.fr

***Abstract*— Neuromorphic circuits aim at emulating biological spiking neurons in silicon hardware. Neurons can be implemented either as analog or digital components. While the respective advantages of each approach are well known, i.e., digital designs are more simple but analog neurons are more energy efficient, there exists no clear and precise quantitative comparison of both designs. In this paper, we compare the digital and analog implementations of the same Leaky Integrate-and-Fire neuron model at the same technology node (CMOS 65 nm) with the same level of performance (SNR and maximum spiking rate), in terms of area and energy. We show that the analog implementation requires 5 times less area, and consumes 20 times less energy than the digital design. As a result, the analog neuron, in spite of its greater design complexity, is a serious contender for future large-scale silicon neural systems.**

I. Introduction

Neuromorphic architectures have been proposed in the past two decades [1], [2] which aim at emulating biological spiking neurons on dedicated silicon hardware. Such neuromorphic systems can be used either to model biological systems [3], or to implement processing tasks [4]. In either case, area and power consumption of the hardware design can have a significant influence on system scalability. Many different implementations of silicon neurons have been proposed, and whatever the neuron model considered (Hodgkin-Huxley, FitzHugh-Nagumo, Integrate-and-Fire, ...), two implementation options are available: analog [5], [6], [7] or digital [8] ,[9] [10]. Depending on the required signal-to-noise ratio, the relative efficiency of a digital versus analog implementation can vary [11].

For spiking neurons, analog designs can take advantage of electronic and physical laws in order to implement the base functions of a spiking neuron, see Figure 1 for a neuron model example. Temporal integration can be realized through capacitive integration, and spatial summation through Kirchhoff's law; leakage is an intrinsic behavior of microelectronic devices. However, analog implementations of spiking neurons often suffer from two limitations. First, analog designs typically exhibit a high sensitivity to process variability, which generally requires additional area and power to guarantee a given signal-to-noise ratio. Second, spiking neurons require rather large capacitances that act as cell membrane (see Section III).

On the other hand, digital implantations are generally faster and easier to design, are not sensitive to process variability, feature inherent noise rejection, and can fully benefit from microelectronics technology scaling. However, at low signal-to-noise ratios, they usually require higher power consumption and area [11].

The goal of this paper is to compare analog and digital implementations. For that purpose, we focus on a specific neuron model at a given technology node (ST CMOS 65nm). We consider the implementation of a Leaky Integrate-and-Fire neuron, and the corresponding model is described in Section II. Analog and digital designs are respectively detailed in Sections III and IV, along with energy and area measurements. They are compared in Section V.

II. LIF Neuron Model

Our target neuron model is the popular Leaky Integrate-and-Fire model, which can be written as follows (see Figure 1):

$$C_i \ \frac{dV_i(t)}{dt} = -I_{leak} + \sum_{n}^{j=1} W_{ij} \ s_j(t) \tag{1}$$

with the additional output and reset equation:

$$\begin{aligned} &\text{if } V_i < V_{th}, \quad \text{then } s_i = 0 \\ &\text{if } V_i \geq V_{th}, \quad \text{then} \begin{cases} s_i = 1 \\ V_i = 0 \end{cases} \end{aligned} \tag{2}$$

where V_i is the internal potential for neuron i, C_i its capacitance, V_{th} is the neuron threshold, s_j is the output of neuron j, I_{leak} is the leakage current, and W_{ij} is the synaptic weight from neuron j to neuron i.

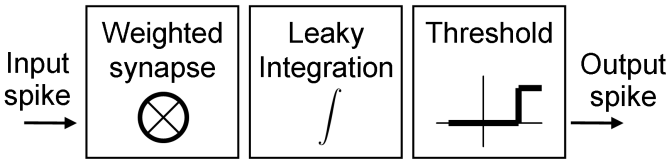

Fig. 1. Block diagram of a leaky-integrate-and-fire neuron.

Through high level application simulations, we aim at implementing a neuron with synaptic weights encoded over 7 bits, plus 1 polarity bit. Membrane potential dynamics should thus feature a 7-bit dynamics, equivalent to a 35 dB SNR. The neuron should be able to operate at a high frequency in order to perform fast operations; such "faster than real-time"

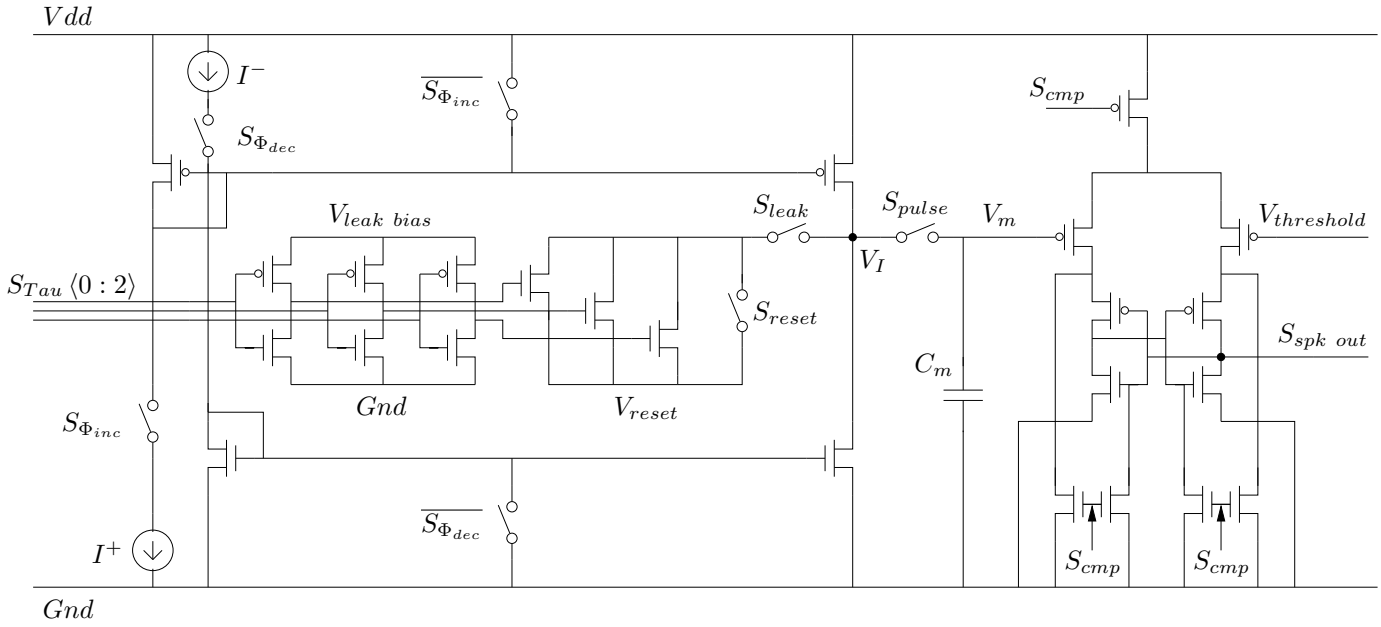

Fig. 2. Analog LIF neuron schematic

neurons are often used in neuromorphic systems to accelerate simulation time of large networks of spiking neurons. Hence, we aim at a maximum spiking frequency of $1MHz$. Overall, this neuron model and parameters are similar to the one described in [9].

III. Analog design

For the first stage of the analog LIF neuron design, we need to specify the requirements of each of the three basic functions described in Figure 1. The analog circuit that we implement is shown on Figure 2 and the main design steps were described in [12]. We briefly describe here the circuit topology, its operating mode and show an application example.

A. Topology

1) Synapse: A 7-bit synapse is realized thanks to a current injection and a digitally controlled switch S_{pulse} activated at $500\ MHz$. Such a time-based modulation injection is more accurate and less prone to variability than current-based modulation. One pulse defines the minimal weight ($1/128 = 8.10^{-3}$).

2) Neuron core: The weight is retained in the membrane potential V_m thanks to a $500\ fF$ MIM capacitance C_m, which requires $100\ \mu m^2$ silicon area. This value is determined thanks to leakage currents observed at the C_m node. With this capacitance value and at a $1\ kHz$ or higher spiking rates, the neuron can be considered as pure Integrate-and-Fire (without leakage). It should be mentioned here that MIM capacitances are fabricated using two metal layers; as a consequence, transistors can be placed on the silicon area underneath the capacitance. This area is thus not lost, but rather a lower boundary for the total neuron area. The Leaky behavior of a LIF neuron is achieved with the programmable leakage block. It is controlled by $S_{Tau}\langle 0:2\rangle$ biasing the three NMOS to a voltage bias $V_{leak\ bias}$.

3) Threshold: Several topologies have been used in neuron circuits to perform threshold comparison like current starved inverter in [3] or basic OTA [6]. A basic OTA is not suitable because of power consumption considerations when V_m is just below $V_{threshold}$. Since the neuron is designed for fast operation, we choose to compare the membrane capacitance and the threshold voltage using a clocked comparator [13].

4) Operating mode: When an input spike is sent to the neuron, a current mirror is activated according to the sign of the weight. This action is performed by $S_{\Phi_{inc}}$ or $S_{\Phi_{dec}}$; both channels respectively contains the signal and its inverse, e.g., $S_{\Phi_{inc}}$ and $S_{\Phi_{inc_{not}}}$. Once the current mirror is stable, the reset switch S_{leak} is opened and the injection begins through the switch S_{pulse}. After a number of pulses corresponding to the synaptic weight, a bias voltage equal to V_{reset} is applied to the V_I node thanks to S_{leak} and S_{reset}. If the neuron operates in its leaky mode, both S_{pulse} and S_{leak} are closed and the capacitance is discharged through one to three transistor(s) while S_{reset} remains open. At the same time, the comparator is activated and outputs a logical event if $V_M > V_{threshold}$. In that case, a request is sent to a digital routing mechanism, which sends back an acknowledgment, which, in turn, activates S_{pulse} - and S_{reset} in the leaky mode -, resetting V_M to V_{reset}. Threshold comparisons are only performed after current injections corresponding to positive weights, since the membrane potential cannot cross the threshold after injection of a negative weight.

B. Simulation

Figure 3 shows a functional example of the designed neuron and its control signals. The synaptic weight is equal to 0.7; the neuron is configured first in IF mode ($t < 5\ \mu s$) and next ($t > 5\ \mu s$) with a leakage time constant equal to $\tau = 2\mu s$. A spike is emitted when the neuron membrane potential crossed

the threshold voltage $V_{threshold}$.

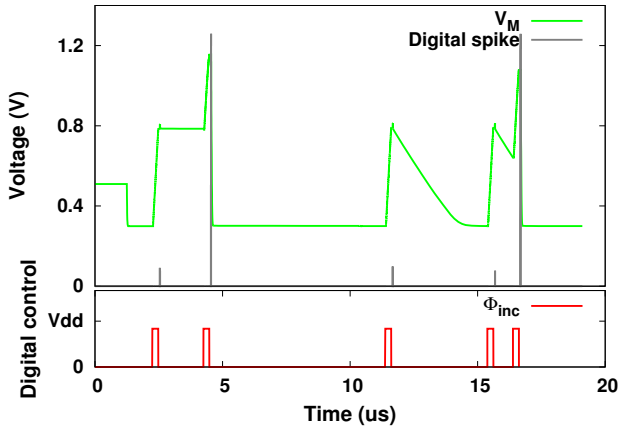

Fig. 3. Analog simulation: Evolution of membrane potential, neuron output, and control signals.

C. Results

Spike injection can be performed using a digital counter generating a number of pulses corresponding to the weight. This digital weight injection block requires about 1300 μm^2 at $65nm$ CMOS technology, and can be shared across multiple neurons. The analog neuron (core + comparator) layout implementation is full-custom and compact (120 μm^2; see figure 4). The neuron can handle injection of positive and negative weights, and has a 35 dB signal-to-noise ratio. Through simulations, we characterized the analog neuron power consumption, which was measured at 2 pJ per spike.

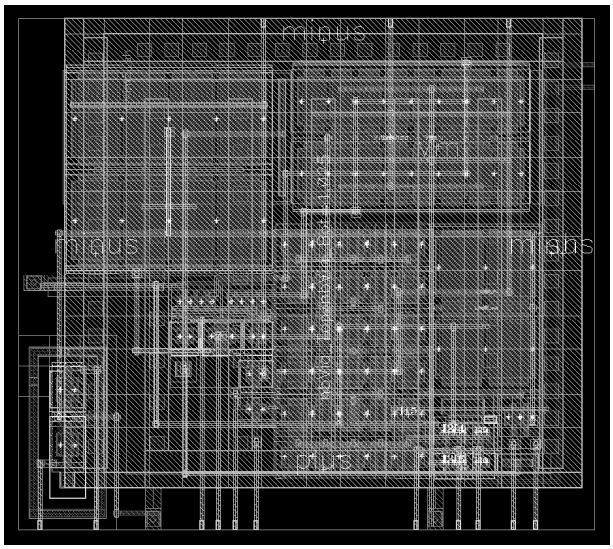

Fig. 4. Analog LIF neuron layout

IV. Digital design

The LIF digital design has the same key components as the analog design described in the previous section. For the sake of a fair comparison, we separately implemented the synapse, the neuron core and the membrane potential comparator so that the behavior of each block matches the behavior of the desired LIF neuron (see Figure 1), and thus, of the analog neuron. The neuron digital implementation is designed to be as compact and as power efficient as possible. Figure 5 shows the block diagram of the digital neuron.

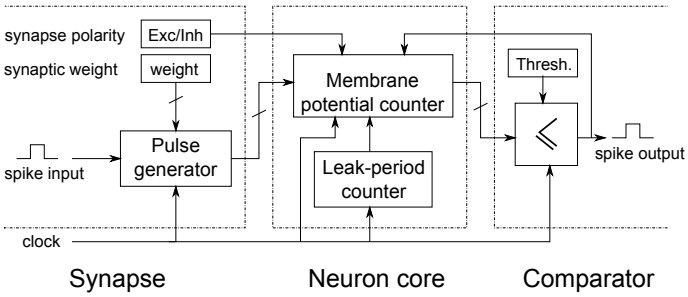

Fig. 5. Digital implementation of the LIF neuron

A. Topology

1) Synapse: This block is triggered by the arrival of external spikes. The block generates a digital pulse whose duration represents the synaptic weight. The heart of the circuit is a 7-bit counter which is reset to the synaptic weight value whenever a spike arrives. The pulse signal is, therefore, held to logic value "1" until the counter reaches zero. The circuit operation supposes that there are no arriving spikes during the pulse generation process. The synapse may have an excitatory or inhibitory impact. The 1-bit polarity parameter is configured to set the synapse type. This parameter will be sent to the neuron core block along with the synaptic weight-coded pulse.

2) Neuron core: This block performs two main functions: the synaptic weight-coded pulse integration and the leak function. The integration process starts with the reception of a pulse from the synapse module. The pulse width indicates the magnitude that has to be added or subtracted to/from the membrane potential according to the synapse type. The membrane potential is represented by a 7-bit up/down counter initially set to zero. The counter is incremented whenever it receives pulses from excitatory synapses, and is decremented whenever it receives pulses from inhibitory synapses. The counter (membrane potential) is bounded and cannot become negative.

The leak function operates when there is no synapse pulse applied to the neuron core (pulse set to logic value "0"). It consists of periodically decrementing the counter of the membrane potential. The leak period may be configured to accelerate or slow down the membrane potential decrease according the neuron model definition.

3) Threshold: The comparator takes the membrane potential counter value as input. It is activated whenever the membrane potential is incremented. The membrane potential is then compared to a threshold value (a configurable variable ranging from 1 to 255). If the membrane potential exceeds the threshold, one spike is generated as a digital pulse. At the

same time, a reset signal feeds back to the neuron core module in order to set the membrane potential to its initial rest value.

B. Simulation

Figure 6 shows a simulation example of the digital LIF neuron. We applied the same spike train inputs as in analog simulation (see Figure 3). During the first two spikes ($t < 5\ \mu s$), the leak function is deactivated. Synaptic weights are then accumulated and the membrane potential reaches the threshold value (set to 126): an output spike is generated. In the remainder of the simulation ($t > 5\ \mu s$), the leak function is turned on. The membrane potential starts to be decremented when its value is greater than zero.

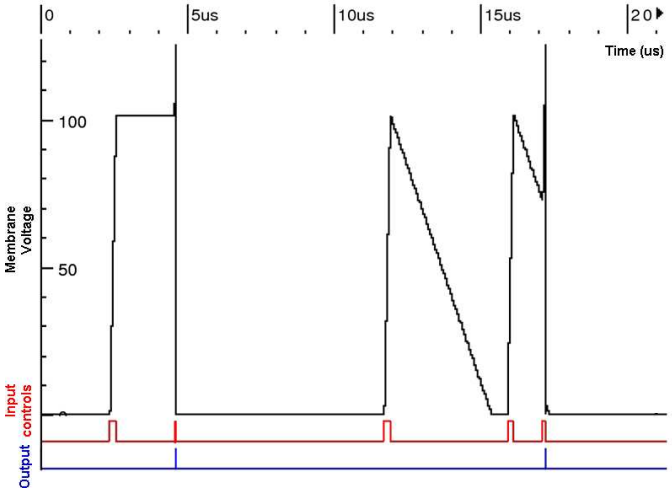

Fig. 6. Simulation of the digital LIF implementation: neuron inputs, output, and membrane potential.

C. Results

The LIF neuron has been designed, simulated and synthesized using ST CMOS 65nm standard-cells based design flow. Table I summarizes area and timing statistics of the neuron core and its comparator. The neuron controller can also be shared across multiple neurons and is compact (less than 300 μm^2). The system runs at 256 MHz which allows a maximum spiking rate of $1.9M$ spikes per second. The power required to generate one spike is estimated at 41.3 pJ.

TABLE I
65NM DIGITAL LIF NEURON STATISTICS

Area	$538\mu m^2$
General clock frequency	256 MHz
Max. spike rate	1.9 $Mspike/s$
Power	78.16 μW
Energy per spike	41.3 pJ

V. COMPARISON & DISCUSSION

Table II summarizes the main results of the analog and digital implementations in terms of area and power. It can be seen that, despite the large capacitance required by the analog implementation, it is almost 5 times more compact than the digital one. It is also 20 times more energy efficient because the analog neuron does not require the high signal-to-noise ratio of a digital implementations. While, in theory, the noise of analog neurons could cumulate when analog neurons are cascaded in large-scale neural systems, the noise is actually suppressed at each neuron because the neuron generates spikes [11], implementing a form of signal regeneration.

TABLE II
ANALOG VS DIGITAL LIF NEURON IMPLEMENTATIONS: SUMMARY

	Analog neuron	Digital neuron
Core + comparator area (μm^2)	120	538
Core + comparator energy (pJ/spike)	2	41
Max. spike rate (Mspike/s)	1.9	1.9

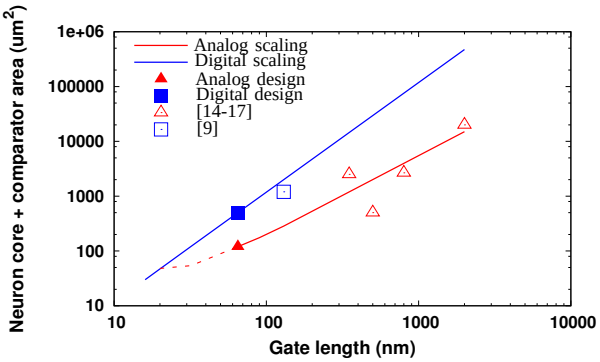

Fig. 7. LIF neuron implementations: analog vs digital neuron evolution according to critical length. A crossover point exists at 22 nm node

Let us now study in more details the impact of technology scaling on the digital vs. analog neuron comparison. Digital implementations directly benefit from technology scaling; this is not the case of analog designs, in part because of the required capacitance. However, the current area ratio between the two designs is about 5x at 65nm. Assuming an optimistic area scaling of 2x every technology node, the analog design would retain its area advantage over the digital design through the 45nm and 32nm nodes until the 22nm node at least, see Figure 7. In this figure, analog scaling takes in consideration several LIF designs implemented by various teams during the last decades [14], [15], [16], [17]. The dotted line shows the area estimation of the neuron described in this paper in advanced nodes. We used projected data from ITRS [18] where parameters such as Vdd or $\sigma_{V_{th}}$ are given until 20 nm node for a similar process type (planar bulk, low standby power logic CMOS process). The 7-bit precision for a neuron and variability among a population of neurons are kept constant, leading to a slowdown in terms of area reduction for analog design.

At 22nm, the analog design would still retain an energy advantage of about 3x over the digital design, again assuming an optimistic (and most likely unrealistic) energy scaling of 2x

every technology node for the digital design. So, in summary, while the capacitance of the analog design is, ultimately, an intrinsic scalability limitation, it shall retain a significant area and energy advantage over the digital design until the 22nm technology node, and probably beyond, assuming more realistic area and scaling of the digital design.

Moreover, novel, advanced technologies are envisioned that will disrupt estimations based on CMOS-scaling only. First, memristors are well suited for synapses implementation, and their co-integration with CMOS to implement a complete neuromorphic system (synapses, core, and comparator) will extend the advantage of analog neurons over their digital counterpart when large networks are considered [19]. Also, very dense capacitances are envisioned in technology roadmaps [20], that may greatly reduce the area cost of analog neurons. Altogether, these advanced devices will keep the analog vs. digital neuron debate open for many years.

ACKNOWLEDGMENT

This work was supported by a ANR grant ANR-09-RPDOC-002-01.

REFERENCES

[1] C. Mead, *Analog VLSI and Neural Systems*. Addison-Wesley, 1989.

[2] W. Maass and C. Bishop, Eds., *Pulsed Neural Networks*. MIT Press, 1999.

[3] R. Vogelstein, U. Mallik, J. Vogelstein, and G. Cauwenberghs, "Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses." *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 253–265, 2007.

[4] S. Mitra, S. Fusi, and G. Indiveri, "Real-Time Classification of Complex Patterns Using Spike-Based Learning in Neuromorphic VLSI," *IEEE transactions on biomedical circuits and systems*, vol. 3, no. 1, pp. 32–42, 2009.

[5] M. Mahowald and R. Douglas, "A silicon neuron," *Nature*, vol. 354, no. 6354, pp. 515–8, 1991. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/1661852

[6] A. van Schaik, "Building blocks for electronic spiking neural networks." *Neural networks*, vol. 14, no. 6-7, pp. 617–28, 2001.

[7] J. Arthur and K. Boahen, "Silicon-Neuron Design: A Dynamical Systems Approach," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 58, no. 99, pp. 1–1, 2011.

[8] L. Maguire, T. Mcginnity, B. Glackin, A. Ghani, A. Belatreche, and J. Harkin, "Challenges for large-scale implementations of spiking neural networks on FPGAs," *Neurocomputing*, vol. 71, no. 1-3, pp. 13–29, 2007.

[9] R. Emery, A. Yakovlev, and G. Chester, "Connection-centric network for spiking neural networks," in *3rd ACM/IEEE International Symposium on Networks-on-Chip*. Ieee, mai 2009, pp. 144–152.

[10] P. Merolla, J. Arthur, F. Akopyan, N. Imam, R. Manohar, and D. Modha, "A Digital Neurosynaptic Core Using Embedded Crossbar Memory with 45pJ per Spike in 45nm," in *IEEE Custom Integrated Circuits (CICC)*, San Jose, CA, USA, 2011.

[11] R. Sarpeshkar, "Analog versus digital: extrapolating from electronics to neurobiology." *Neural computation*, vol. 10, no. 7, pp. 1601–38, octobre 1998.

[12] A. Joubert, B. Belhadj, and R. Heliot, "A robust and compact 65 nm LIF analog neuron for computational purposes," in *2011 IEEE 9th International New Circuits and systems conference*. IEEE, Jun. 2011, pp. 9–12. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5981206

[13] P. M. Figueiredo and J. C. Vital, *Offset Reduction Techniques in High-Speed Analog-to-Digital Converters: Analysis, Design and Tradeoffs*. Springer, 2009.

[14] G. Cauwenberghs, "An analog VLSI recurrent neural network learning a continuous-time trajectory," *IEEE Transactions on Neural Networks*, vol. 7, no. 2, pp. 346–361, 1996.

[15] D. Goldberg, G. Cauwenberghs, and A. Andreou, "Probabilistic synaptic weighting in a reconfigurable network of VLSI integrate-and-fire neurons." *Neural networks*, vol. 14, no. 6-7, pp. 781–793, 2001. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/11665770

[16] G. Indiveri, E. Chicca, and R. Douglas, "A VLSI array of low-power spiking neurons and bistable synapses with spike-timing dependent plasticity." *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 211–21, janvier 2006. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/16526488

[17] E. Chicca, G. Indiveri, and R. Douglas, "An event based VLSI network of integrate-and-fire neurons," in *Proc. IEEE Int. Symp. Circuits Syst*, vol. pp. Citeseer, 2004, pp. 357–360.

[18] International technology roadmap for semiconductors, 2011 edition. [Online]. Available: http://www.itrs.net/Links/2011ITRS/Home2011.htm

[19] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale memristor device as synapse in neuromorphic systems." *Nano letters*, vol. 10, no. 4, pp. 1297–301, 2010.

[20] M. Budnik, A. Raychowdhury, A. Bansal, and K. Roy, "A high density, carbon nanotube capacitor for decoupling applications," in *43rd annual conference on Design automation - DAC '06*. New York, New York, USA: ACM Press, 2006, p. 935.