A dissertation submitted to

UNIVERSITÀ DEGLI STUDI DI ROMA "TOR VERGATA"
FACOLTÀ DI SCIENZE MATEMATICHE FISICHE E NATURALI
for the degree of Dottore di Ricerca

and to

UNIVERSITÄT ZÜRICH
MATHEMATISCH - NATURWISSENSCHAFTLICHEN FAKULTÄT
for the degree of Naturwissenschaftlichen Doktorwürde

Co-Doctorate in Physics

Networks of spiking neurons and plastic synapses: implementation and control

Massimiliano Giulioni
Born January 25, 1979
citizen of Italy

Accepted on the recommendation of

Rome

Dr. Paolo Del Giudice
Supervisor
Department of Technologies and Health
Italian National Institute of Health

Dr. Gaetano Salina
Supervisor
Istituto Nazionale Fisica Nucleare
Department of Rome "Tor Vergata"

Prof. Lucia Zanello
Examiner
Physics Department
University of Rome "La Sapienza"

Zurich

Dr. Giacomo Indiveri
Supervisor
Institute of Neuroinformatics
University of Zurich

Prof. Rodney Douglas
Supervisor
Institute of Neuroinformatics
University of Zurich

Prof. Richard Hahnloser
Examiner
Institute of Neuroinformatics
University of Zurich

Rome March 28, 2008

Abstract

The brain is an incredible system with a computational power that goes further beyond those of our standard computer. It consists of a network of 10^{11} neurons connected by about 10^{14} synapses: a massive parallel architecture that suggests that brain performs computation according to completely new strategies which we are far from understanding.

To study the nervous system a reasonable starting point is to model its basic units, neurons and synapses, extract the key features, and try to put them together in simple controllable networks. The research group I have been working in focuses its attention on the network dynamics and chooses to model neurons and synapses at a functional level: in this work I consider network of integrate-and-fire neurons connected through synapses that are plastic and bistable. A synapses is said to be plastic when, according to some kind of internal dynamics, it is able to change the "strength", the efficacy, of the connection between the pre- and post-synaptic neuron. The adjective bistable refers to the number of stable states of efficacy that a synapse can have; we consider synapses with two stable states: potentiated (high efficacy) or depressed (low efficacy). The considered synaptic model is also endowed with a new *stop-learning* mechanism particularly relevant when dealing with highly correlated patterns.

The ability of this kind of systems of reproducing in simulation behaviors observed in biological networks, give sense to an attempt of implementing in hardware the studied network. This thesis situates at this point: the goal of this work is to design, control and test hybrid analog-digital, biologically inspired, hardware systems that behave in agreement with the theoretical and simulations predictions. This class of devices typically goes under the name of *neuromorphic* VLSI (*Very-Large-Scale Integration*). *Neuromorphic* engineering was born from the idea of designing bio-mimetic devices and represents a useful research strategy that contributes to inspire new models, stimulates the theoretical research and that proposes an effective way of implementing stand-alone power-efficient devices.

In this work I present two chips, a prototype and a larger device, that are a step towards endowing VLSI, *neuromorphic* systems with autonomous learning capabilities adequate for not too simple statistics of the stimuli to be learnt. The main novel features of these chips are the implemented type of synaptic plasticity and the configurability of the synaptic connectivity. The reported experimental results demonstrate that the circuits behave in agreement with theoretical predictions and the advantages of the *stop-learning* synaptic plasticity when highly correlated patterns have to be learnt. The high degree of flexibility of these chips in the definition of the synaptic connectivity is relevant in the perspective of using such devices as building blocks of parallel, distributed multi-chip architectures that will allow to scale up the network dimensions to systems with interesting computational abilities capable to interact with real-world stimuli.

Table of contents

List of Figures

Chapter 1

Introduction

Human brain is an incredible system, able to interact with the real world, solve problems, take decisions, learn, think, invent, create new theories or write a PhD thesis. And trying to understand it, its basic principles, its fundamental mechanisms is a fascinating challenge that moves and gathers the interests of people from different research fields as neurophysiologists, mathematicians, physicists or engineers.

The human brain works thanks to its 10^{11} neurons connected by about 10^{14} synapses, packed in a volume of 1,7 liter [Williams and Herrup, 1988] [Pakkenberg and Gundersen, 1997] [Pakkenberg et al., 2003]. Its architecture and its way of performing computation are profoundly different from those of a computer. The brain processes information on a network of tons of simple and densely inter-connected analog elements instead of exploiting one or few complex digital elaboration units. In terms of energy consumption Sarpeshkar [1998] estimates the efficiency of the brain as about $3 \cdot 10^{14}$ operations per Joule. On the other hand the supercomputer BlueGene/L, constituted of 131072 digital processors designed by IBM [Gara et al., 2005], can reach a theoretical peak of $367 \cdot 10^{15}$ floating point operations per second with a power consumption of about 10^6W. This corresponds to a performance of 10^9 operations per Joule, still 5 orders of magnitude lower than the brain efficiency. Furthermore, the inner ear by itself carries out at least the equivalent of a billion of floating-point operations per second, about the workload of a typical game console [Sarpeshkar, 2006]. The inner ear together with the brain can distinguish sounds that have intensities ranging over 120 decibels, from the roar of a jet engine to the rustle of a leaf, and it can pick out one conversation from among dozens in a crowded room. The truly amazing thing is that the game console consumes about 50W whereas the inner ear uses just 14 microwatts [Sarpeshkar, 2006]. How is this possible, or better, which is the computational strategy used by the brain is one of the engines of this kind of research.

To study the nervous system a reasonable starting point is to model its basic units, neurons and synapses, extract the key features, and try to put them together in simple controllable networks. It is somehow an attempt to rebuild a small piece of nervous tissue putting in only the known, and hopefully relevant, elements. Different levels of description can be adopted ranging from detailed to functional models. Detailed models try to describe how a system operates on the basis of known anatomy, physiology and circuitry; an example is the model of neuron proposed in 1952 by two British physiologists and biophysicists Hodgkin and Huxley [1952] who won the 1963 Nobel Prize in Physiology or Medicine for their work on the basis of nerve action potential, the electrical impulse that enables communication

among neurons. They modeled the biochemical reactions that took place on the neuron membrane and provide a mathematical formulation of the phenomena. A detailed model that successfully reproduced the single neuron behavior. To study the nervous system at another level, that of neuronal circuits or networks, some details related to the anatomy or the physiology of the nervous tissue have to be discarded, if a certain degree of analytical control is desired. Theoretical analysis of the dynamics of a network requires significant simplification in the description of the original neuron. An example of functional model is the integrate-and-fire neuron originally proposed by Lapique in 1907 [Lapique, 1907]. The model captures the essence of the non-linear process performed by a neuron: the cell integrates the inputs and when a threshold is crossed, an action potential, a spike, is generated. The alternative choice of maintaining detailed models of neurons and synapses and build with them a simulation of a large networks results in an incredible workload: in 2005 the Blue Brain Project has been launched [Markram, 2006], they planned to simulate a network of up to 10^5 highly complex, detailed neurons, to this aim they will use the Blue Gene IBM supercomputer. Chosen the models and built the network, it is possible to start studying the system dynamics and the emerging properties. Ultimately, the model to use is chosen according to research goals, and will be the ability of models of reproducing the experimental data (at the desired level of description) to legitimate the choice of that particular model.

The research group I have been working in focuses its attention on the level of description of the network dynamics and consequently neurons and synapses are modeled at a functional level: in particular in this thesis work I considered network of integrate-and-fire neurons connected through synapses that are plastic and bistable. A synapses is said to be plastic when, according to some kind of internal dynamics, it is able to change the "strength", the efficacy, of the connection between the pre- and post-synaptic neuron. The adjective bistable refers to the number of stable states of efficacy that a synapse can have; we consider synapses with two stable states: potentiated (high efficacy) or depressed (low efficacy). Network of integrate-and-fire neurons connected through plastic synapses demonstrated to have interesting learning abilities [Amit and Fusi, 1994], and to be able to reproduce some neurophysiological experimental results [Giudice et al., 2003].

Even if we are far from a comprehensive knowledge of the brain, or better, we are just at the beginning of a fascinating way, the results obtained in simulation give sense to an attempt of reproducing in hardware the studied network. This thesis situates at this point: the goal of this work is to design, control and test hybrid analog-digital, biologically inspired, hardware systems that behave in agreement with the theoretical and simulations predictions. This class of devices typically goes under the name of *neuromorphic* VLSI.

The *neuromorphic* design philosophy was originally proposed by Mead in 1989 [Mead, 1989]. *Neuromorphic* VLSI refers to those devices that directly embody, in the physics of their components, analogues of the physical processes that underlie the computation of a neural system. *Neuromorphic* VLSI is something different from a mathematical description or from a PC-based numerical simulation. Such research methods, essentially math and software, live on abstract worlds of equations or of boolean algebra. *Neuromorphic* VLSI stresses the character of the computation as a physical process pushing the parallelism between the biological and artificial systems to a lower plane, on which the attempt is to map the biochemical processes going on in the biological neuron and synapses in the physics of transistors that constitute VLSI devices [Mead, 1990]. The parallelism between biology and VLSI, inspires also the architectural level of this hardware: parallel asynchronous hybrid systems are preferred to those serial and digital architecture typical in commercial

microprocessors.

In this work I present the hardware implementation of two *neuromorphic* chip implementing configurable neural networks; two VLSI devices whose hybrid analog-digital circuits reproduce mathematical models of neurons and synapses. The first device has been thought as a test chip of 32 neurons and 64x32 synapses; it has been extensively and successfully tested. A second, larger chip has then been designed, improved under various technical aspects, it hosts a network of 128 neurons with 128 synapses for each neuron.

Does it make sense to spend energy, resources and time on custom VLSI devices when much larger networks can be simulated on standard personal computers in real-time? In the present stage of development VLSI devices are difficult to tune, unstable, absolutely unable to compete with PC-based simulations and expensive too. If just a few years ago having neural networks in hardware was the only feasible way to "simulate" them, today computers development has completely overturned the situation. Using Perseo, for instance, a simulation software written by Mattia and Del Giudice [Mattia and Giudice, 2000], it is possible to simulate on a standard personal computer, in real time, networks of few thousands of neurons connected through plastic synapses. As far as I know, custom analog VLSI devices implementing networks of analogous dimensions are still not available.

The question now is: why should you design such expensive toys? A simple answer is that if you do not start, you will never arrive; designing hardware neural networks has a meaning in perspective. If one assumes a continuous progress in the comprehension of the nervous system, in a future it will become desirable having a new class of "neural" VLSI microprocessors able to perform computation as the brain does today. *Neuromorphic* VLSI remains a useful research strategy, complementary to theoretical speculations to evolve towards a better comprehension of the mechanism at the basis of neural computation. A real device is a test-bench for the models, to verify how much robust they are in front of problems not foreseen in the theoretical models. And, in the other direction, testing the hardware should suggest new ideas and solutions to improve the models. Thus theory and hardware should grow in parallel to gather benefits from each other. The theory tries to describe something real, the brain, composed of biological tissues; having something real, even if of other kind, on which is possible to map the models, run experiments and analyze results, can help refining the theory, at least in addressing those aspects that represent strict constrains only for real devices as the power consumption of neurons and synapses or as the issue of making a huge amount of cells asynchronously communicate each other; problems to which biology found solutions that artificial systems try to reproduce. The chips described in this thesis represent the result of such a mutual interaction between the implementation and the theoretical levels. One example of this process is the chosen model of a plastic bistable synapse. Material implementations of a synapse introduced various constraints as for instance a limited maximum number of analog states stable over long time periods [Amit and Fusi, 1994]. This limit drove the theoretical research towards a synaptic model able to turn into advantage such a constrain. The result is the bistable synapse able of stochastic learning [Fusi, 2001] implemented in the chip described in this thesis. Summing up, advances of the theory and a growing experience in designing *neuromorphic* chips lead to models explicitly thought to be implemented in hardware, and to a hardware that correctly reproduces the models. The problem of dealing with small networks seems not, right now, a big issue. The point is that there is not yet a theory, except for particular cases, able to exploit the computational power that derives from a neuronal architecture, whatever the dimensions of the network are.

Besides being a research strategy in neuroscience and a source of inspiration for the theory, *neuromorphic* devices are low-power compact portable systems that process information in a way similar to biological nervous tissues. Progress in this area could eventually lead to specific-task application in the brain-machine interface or prosthetic fields where *ad hoc neuromorphic* structures can maximize computational power and could be naturally interfaced to biological nervous systems.

Chapter 2

Models for a compact VLSI implementation

One of the main goal of physicists is to extract from complex systems relevant features to create controllable models. The brain is one of those systems terribly complex and terribly interesting at the same time. A reasonable strategy to study the brain starts from the identification of the key features of its basilar components, neurons and synapses, from which to build functional models of simplified network. Many of such models, both at biological and functional level have been developed. Here we will focus on those interesting for VLSI implementations of a neural networks.

2.1 Neurons

Neurons are highly specialized for generating electrical signals in response to chemical and other inputs, and transmitting them to other cells. Many different types of neurons exist in the human brain and endless variations in neuron types of other species. Beside the cellular body, the soma, it is possible to discern two other important morphological specializations in neurons (see figure 2.1): the dendrites that receive the inputs from other neurons and the axon that carries the neuronal output to other cells. Neurons receive a large number of inputs from thousands to hundreds of thousands, as in the cerebellar Purkinje cell. On the output side axons from single neurons can traverse large fractions of the brain or in some cases, of the entire body. The soma is the 'central process unit' and it performs an important non-linear processing step: if the total input exceeds a certain threshold, then an output signal, the action potential, is generated.

The electrical relevant signal for the nervous system is the potential difference across the soma membrane. Under resting conditions the potential inside the cell membrane is about -70mV relative to that of the surrounding bath, conventionally defined to be 0mV, and the cell is said to be depolarized. To maintain such a potential difference a current has to flow. This is the activity of the ion pumps located in the cell membrane which transport ions to maintain ionic concentration gradients. Predominantly Sodium, Potassium, Calcium and Chloride are the ionic species involved. For example the Na^+ concentration is higher outside than inside a neuron while, on the contrary, K^+ is more concentrated inside the cell than in the extracellular medium. Ions flow according to their concentration gradient through

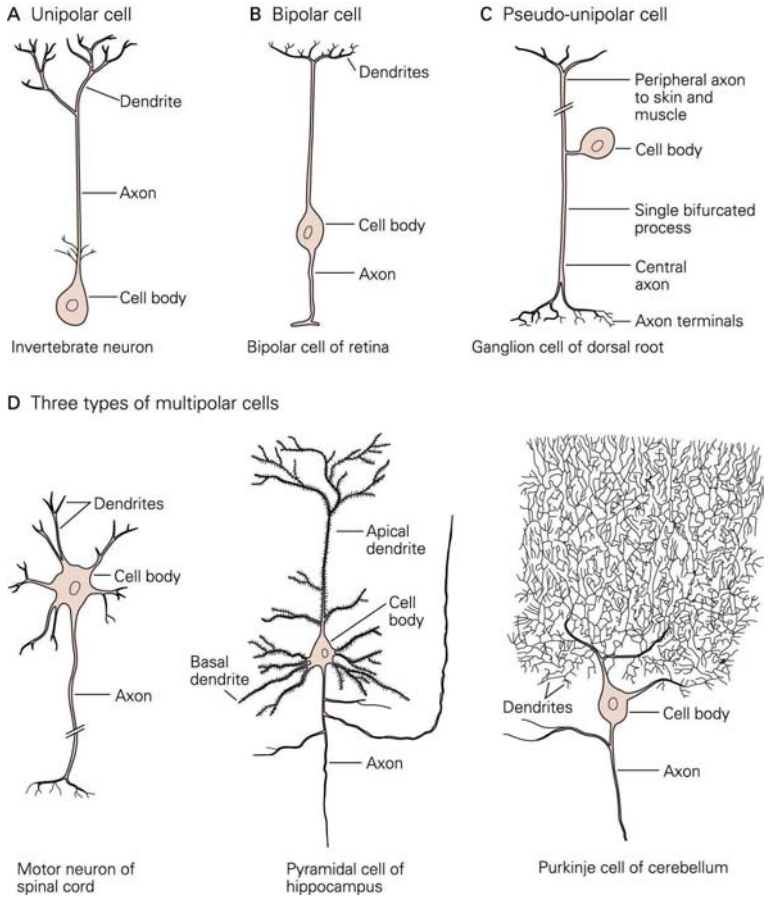


Figure 2.1: Different kinds of neurons. In a neuron is possible to distinguish three parts: the dendrites that receive the inputs, the cell body that perform a non-linear computation emitting an action potential when the input level reaches a certain threshold, and the axon which transmits the action potential.

a variety of ions channels which open and close in response to voltage changes as well as to internal or external signals. Current flowing through open channels outside the cell, makes the membrane potential more negative, a process called *hyperpolarization*. Inverse current *depolarizes* the cell. If a neuron is sufficiently depolarized, i.e. the voltage across the membrane passes a threshold, a positive feedback process is initiated and the neuron generates an action potential. It roughly is a 100mV fluctuation of the membrane potential lasting about 1ms. This signal propagates along the axon where is actively regenerated to arrive to the synaptic bouton at the end of the axonal arborization. Generation of action potentials also depends on the recent firing history of the cell. For few milliseconds after an action potential has been fired, it may be virtually impossible to initiate another spike.

This is called the *absolute refractory period*.

To get an idea of which compression level the evolution achieved in packing neurons together, it is enough to say that in one mm^3 there are about 10^5 neurons, 4 Km of axons and 450m of dendrites; the human brain hosts a total of 10^{11} neurons and 10^{14} synapses.

2.1.1 Hodgkin and Huxley model

Hodgkin and Huxley won the 1963 Nobel Prize in Physiology or Medicine for their work on the basis of nerve action potential. They proposed the first detailed neuronal model in 1952 [Hodgkin and Huxley, 1952]. Their paper summarizes the studies on the giant squid axon and proposes an analytical model of ionic currents affecting the neuronal dynamics. They showed how these currents account for important features of the neurons as the generation of action potentials and the absolute refractory period or .

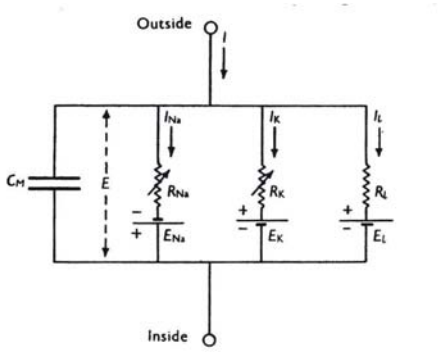


Figure 2.2: Electrical scheme for the conductance-based neuron model by Hodgkin and Huxley redraw from [Hodgkin and Huxley, 1952]. The neuronal membrane is approximated by a fixed capacitor C_m, at the outside terminal there is the voltage measured on the external face of the neuron membrane, inside is inside the neuron membrane, synaptic contributions are schematized by the current I, ionic currents through the membrane by I_{Na}, I_K and by a generic leakage current I_l mainly due to Chloride ions. I_{Na} and I_K flow through time and voltage dependent conductances, I_l through a fixed one. The three batteries account for the reversal potential of the different ionic species.

The model they proposed is based on the electrical scheme of figure 2.2. From their studies emerged that the membrane can be considered as a fix capacitor C_m affected by a set of ion currents. In particular they considered three kind of currents, one given by Sodium ions, one by Potassium ions and a third, leakage current, mainly due to Chloride ions. The current flowing in the circuit of figure 2.2 is than given by

$$I(t) = C_m \frac{dV}{dt} + I_{Na} + I_K + I_l \tag{2.1}$$

Each ionic current depends on a "driving force" and on permeability coefficients which vary according to the potential $V(t)$ across the membrane capacitor. The time-course of these coefficients affect the behaviors of the currents which in turn modify the membrane potential $V(t)$. Fitting their experimental data, Hodgkin and Huxley described analytically the permeability coefficients behavior and they derived numerically the form of the action

potential. The "driving force" is due to the differences of single ionic specie concentrations

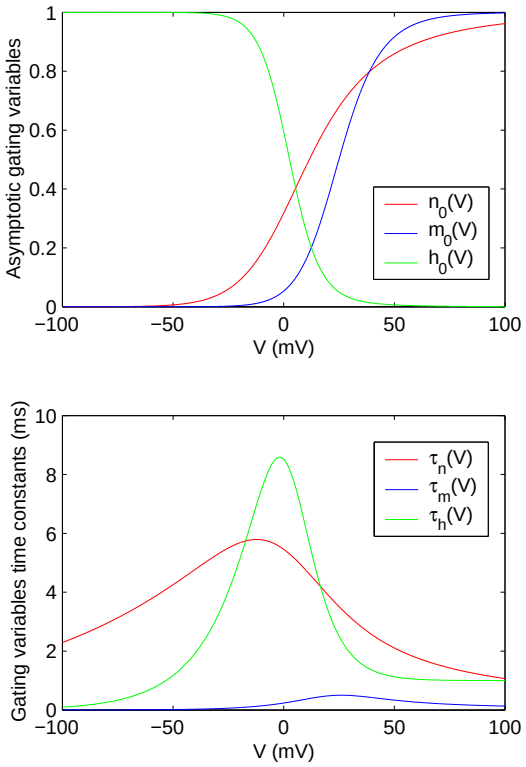


Figure 2.3: Asymptotic values (upper panel) and time constants of the Hodgkin and Huxley gating variables (n, m and h) with respect to the membrane potential V. The asymptotic values of variables n and m increase with V thus they are activation variables, while h is an inactivation variables. n, m and h controls the channel conductances of the Hodgkin and Huxley model. For the reduction to two-dimensional neuron models n_0 is considered almost equal to $1 - h_0$ and $\tau_m \ll \tau_{n,h}$ so that the time course of m will be discarded.

inside and outside the semi-permeable neuron membrane. The ionic currents can be written under the form of

$$I_k = g_k(V)(V - V_k) \tag{2.2}$$

where k stands for Na^+, K^+ and Cl^-, g_k are the conductances of the different ion channels, and V_k the reversal (equilibrium) potentials of the various ionic species. g_{Na} and g_K are modeled as time and voltage dependent conductances, while g_l is a constant value. To describe the behavior of g_{Na} and g_K, they introduced three gating variables m, n and h accounting for the probability that a channel is open. The combined action of m and h control the Na^+ channels while n controls the K^+ ones:

$$\sum_k I_k = g_{Na} m^3 h (V - V_{Na}) + g_K n^4 (V - V_K) + g_l (V - V_l) \tag{2.3}$$

Combining equation 2.3 with 2.1

$$C\frac{dV}{dt} = -g_{Na} m^3 h (V - V_{Na}) - g_K n^4 (V - V_K) - g_l (V - V_l) + I(t) \tag{2.4}$$

A physical basis for the chosen Potassium conductance is given if one supposes that the ionic specie can only cross the membrane "when four similar particles occupy a certain region of the membrane". Each, independently, will occupy its place with a probability described by n, hence the total probability will be n^4. The Sodium conductance is a little bit more complicated: the physical basis can be given supposing the channel activated by three identical independent events, each with a probability m, and blocked by another event with a probability h. The gating variables evolve according to the following differential equations:

$$\dot{x} = -\frac{1}{\tau_x(V)}[x - x_0(V)] \tag{2.5}$$

where x stands for m, n or h. If V changes suddenly to a new value, x approaches the asymptotic value $x_0(V)$ with a time constant $\tau_x(V)$. Hodgkin and Huxley were able to fit $x_0(V)$ and $\tau_x(V)$ from experimental data obtained with the *voltage-clamp* technique, which allows to measure the current needed to keep the membrane at the fixed clamp potential. Figure 2.3 reports their results: as anticipated m and n are activating variable, i.e. they increase with V, while h, having the opposite behavior, accounts for an inactivating mechanism.

So, if an external input increases the membrane potential, the Sodium channels open, Na^+ ions enter the cell and induce a further raise of the voltage V. If this positive loop is strong enough an action potential is initiated. For high values of V, h slowly approaches its asymptote closing the Sodium channels. The positive difference $\tau_h - \tau_m$ ensures the rise of V. On a timescale similar to τ_h the Potassium current sets in and, since it is in the outward direction, it lowers the membrane potential. The combined dynamics of the Sodium and Potassium currents generate a short action potential followed by a negative overshoot.

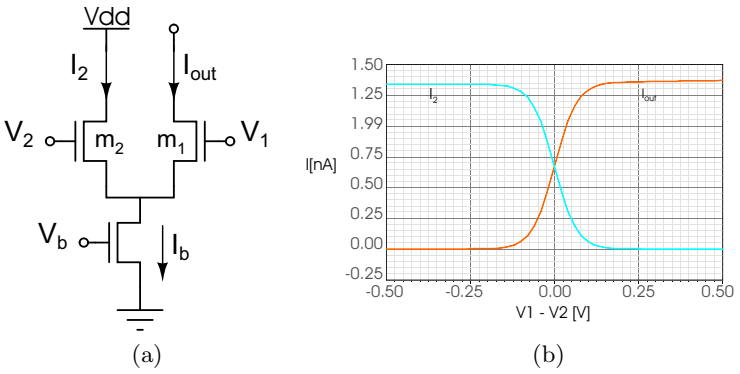
The Hodgkin and Huxley model captures the basic mechanism of generating action potentials in the giant squid axon. This mechanism is essentially preserved in higher organisms. Cortical neurons in vertebrates exhibit a much richer repertoire of electro-physiological properties due to a larger variety of different ion channels [Gerstner and Kistler, 2002]. The Hodgkin and Huxley analytical description, however, remains the landmark to model more detailed systems that will include a larger number of conductances and gating variables to describe different kind of neurons.

2.1.2 A VLSI implementation of the Hodgkin and Huxley model

The Hodgkin and Huxley model has been the starting point for the implementation of a series of neuromorphic neurons on silicon. The first authors to propose an analog VLSI circuit approximating the conductance-based behavior described by Hodgkin and Huxley were Mahowald and Douglas in 1991 [Mahowald and Douglas, 1991]. Their original idea raises from the parallelism between the sigmoidal behavior of the ions conductances and the current to voltage characteristic of the differential pair circuit [Mead, 1989] [Liu et al., 2002] reported in figure 2.4. The diff-pair circuit is a sort of source follower where the bias current I_b is shared by two MOSFETs m_1 and m_2. If all transistors operate below threshold and in saturation, the current I_{out} has the form (see figure 2.4b)

$$I_{out} = I_b \frac{e^{\kappa V_1}}{e^{\kappa V_1} + e^{\kappa V_2}}. \tag{2.6}$$

where κ is the subthreshold MOSFET factor [Mead, 1989]. The basic silicon neuron circuit



(a) (b)

Figure 2.4: Differential pair circuit (a) and I_{out} characteristic. All MOSFETs work in subthreshold regime where they have an exponential current-to-voltage characteristic. I_b is a bias current that can be tune via the voltage V_b. The sigmoidal shape of the I_{out} is used to approximate the voltage-dependent steady state behavior of the gating variables n, m and h reported in figure 2.3.

is the one proposed by Hodgkin and Huxley (see figure 2.2) where the ionic currents are the output currents of differential pair circuits. Thus the dependence of the current steady state on the voltage across the membrane capacitor follows the desired behavior. To add a time dependence, the membrane potential controls the differential pair via a low-pass filter implemented with a simple transconductance amplifier (see figure 2.5) and a capacitor [Mead, 1989]. Using these building blocks both activation and inactivation mechanisms can be designed. In figure 2.6 is shown the part of the analog circuit for the generation of the action potential as proposed in [Rasche and Douglas, 1999], which represents an improved implementation of the original design described in [Mahowald and Douglas, 1991]: the differential pairs were substituted by transconductance amplifier, a differential pair coupled with a current mirror, so that the output current follows the hyperbolic behavior:

$$I_{out} = \tanh \frac{\kappa(V_1 - V_2)}{2} \tag{2.7}$$

The analog circuit includes the Sodium and Potassium conductances as well as the passive leakage current. In the Hodgkin and Huxley model, the bell-shaped form of Sodium current is obtained as the product of an activation variable m with the inactivation variable h. In this particular silicon implementation an analogous voltage dependence is obtained summing an activation current (I_m) and a deactivation current (I_h). The circuit demonstrated its ability to qualitatively reproduce the time-course of the action potential.

Rasche and Douglas went a little bit further. Their complete circuit comprises other two blocks accounting for the Calcium current and for the Potassium conductance which depends on the Calcium concentration and which controls the so-called AHP (After Hyperpolarizing Potassium) current [Gerstner and Kistler, 2002]. The Calcium ions enter the cell via a conductance with elevated threshold that activates during the action potential. When the Ca concentration increases the AHP current starts flowing. It reduces the neuron firing rate thus realizing a frequency adaptation mechanism. This detailed conductance-based circuit has been proved to reproduce electrophysiological measurements. Such a result is obtained

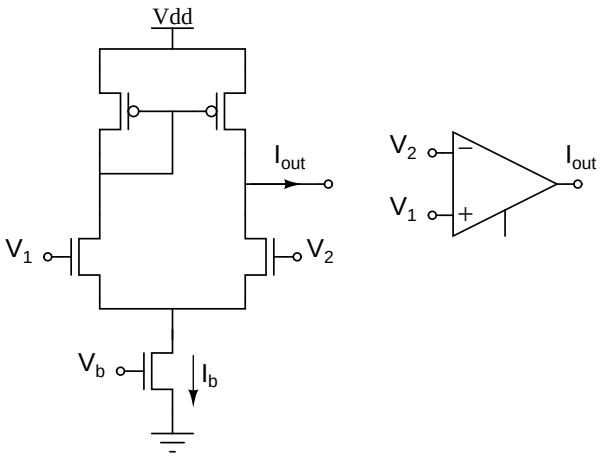


Figure 2.5: Transconductance amplifier schematic and symbol. The transistors work in subthreshold regime. This amplifier results from the coupling of a differential pair circuit and a current mirror. I_b is a bias current that can be tuned via the voltage V_b. I_{out} characteristic is the hyperbolic tangent of the difference $V_1 - V_2$. Such a sigmoidal behavior can be used to reproduce in silicon the voltage-dependent steady state behavior of the gating variables n, m and h.

in a *neuromorphic* way, mapping the ionic current onto transistor currents with a biologically plausible behavior.

The complete silicon neuron proposed consists of a series of analog blocks, each accounting for a particular conductance. Each block generates as output a current that summed with the others is then injected into the membrane capacitance. This modular way of designing conductance-based neuron gather the interests of various research groups that developed more and more detailed neuronal circuits. Simoni and his colleagues [Simoni et al., 2004] [DeWeerth et al., 2007] have proposed a sophisticated module that can be tuned to fit the biological dynamics of various conductances. Their accurate silicon neuron has been successfully interfaced to *in vitro* biological cells [Simoni et al., 2000]. Analogous hybrid biological-artificial system have been studied by Alvado and colleagues [Alvado et al., 2004]. They propose conductances designed using a custom library of analog circuits that reproduce different analytical functions. The level of accuracy that these groups reach in the respective implementation is impressive. The problem is the silicon area occupancy of those neurons: their chips realize networks of no more than 15-20 neurons. An example of conductance-based silicon neuron thought to reduce to minimum the required silicon area is reported in [Hynna and Boahen, 2007]: 8 transistors and one capacitor are sufficient to replicate in silicon the sigmoidal voltage dependence of activation or inactivation and the bell-shaped voltage-dependence of the corresponding time constants. The elegant design proposed in [Farquhar and Hasler, 2005] obtain similar results using even fewer MOSFETs.

2.1.3 Two-dimensional neuron models

The Hodgkin and Huxley model is defined by four coupled differential equations 2.4-2.5. The behavior of such an high-dimensional non-linear system is difficult to visualize or analyze.

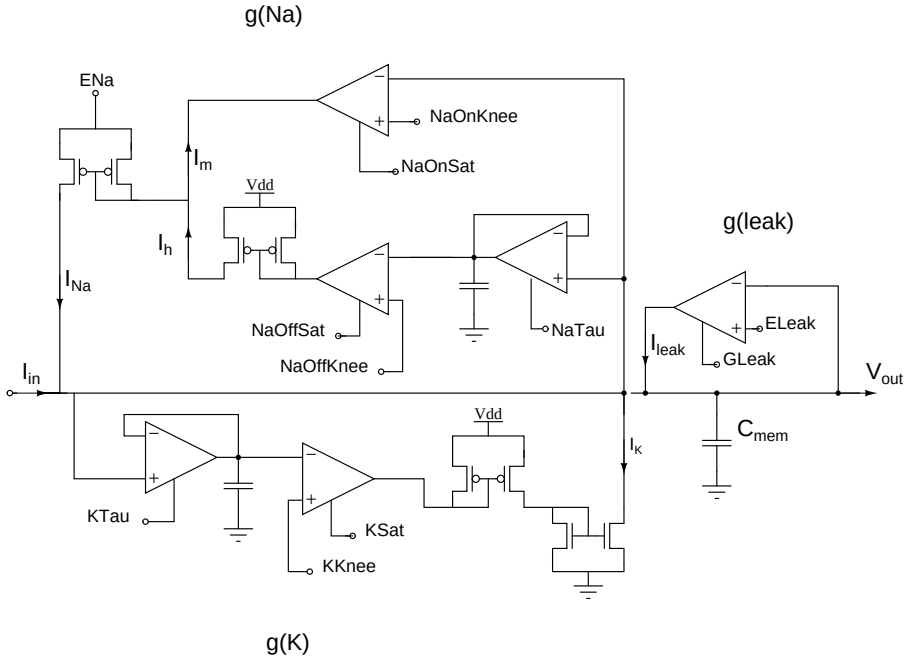


Figure 2.6: Schematic view of the neuromorphic neuron proposed by Rasche and Douglas [Rasche and Douglas, 1999]. The figure shows the part of the circuit responsible for the spike generation. I_{in}, on the left, is the synaptic current coming from the dendritic tree. The circuit reproduces on silicon the three conductances described by Hodgkin and Huxley: one for the Sodium (g_{Na}, upper part) consisting of an activation and a deactivation branch that generate respectively the currents I_m and I_h. The sum of these current is the Sodium current I_{Na} that is injected into the membrane capacitor C_{mem}. The voltage V_{out} represents the neuron membrane potential. Another silicon conductance g_K is the one for the Potassium (lower part) consisting in the activation branch which controls the current I_K. The third fixed conductance g_{leak} is for the constant leakage current I_{leak}. The time-course of the voltage dependent gating variables is obtained using a transconductance amplifier and a capacitor connected in low-pass filter configuration. The 10 parameters are adjusted to fit the behavior of the ionic currents.

Two-dimensional differential equations, however, can be studied by means of a phase plane analysis. A reduction of the four-dimensional equation of Hodgkin and Huxley to a two-variable neuron model is thus highly desirable.

The reduction is based on two qualitative observations. The first one regards the timescale of the dynamics of the four variables: m is much faster than n,h or V. This suggests to approximate the instantaneous value of $m(t)$ with its steady state value $m_0[V(t)]$. This goes under the name of the *quasi steady state approximation*. The second observation is that h and n evolve according similar time constants whatever the voltage V (see figure 2.3 lower panel), and that $n \approx 1 - h$. Thus the idea is to replace n and h with an effective variable w. To keep the formalism a slightly more general we can write $w = b - h \approx an$.

Introducing these approximations in equation 2.4 we have:

$$C\frac{dV}{dt} = -g_{Na}[m_0(V)]^3(b-w)(V-V_{Na}) - g_K(\frac{w}{a})^4(V-V_K) - g_l(V-V_l) + I \tag{2.8}$$

or

$$C\frac{dV}{dt} = \frac{1}{\tau}[F(V,w) + RI] \tag{2.9}$$

with $R = g_l^-1$, $\tau = RC$ and some function F. For what concerns the three equations 2.5, the one for m simply disappears because $m(t) \to m_0[V(t)]$ and the other two reduce to an effective equation for w

$$C\frac{dw}{dt} = \frac{1}{\tau_w}[G(V,w)] \tag{2.10}$$

where τ_w is a parameter and G is a function that has to be specified. Equations 2.9 and 2.10 define a general two-dimensional neuron model. For an analytical justification of the chosen approximation refer to [Gerstner and Kistler, 2002]. Two examples of two-dimensional models are the Morris-Lecar and the FitzHugh-Nagumo neurons.

2.1.4 Morris-Lecar model

Morris and Lecar [Morris and Lecar, 1981] proposed to model the neuron behavior through two equations: one for the evolution of the membrane potential V and one for a slow "recovery" variable $\hat{w}$. In dimensionless variables the Morris-Lecar equations read

$$\begin{aligned} \frac{dV}{dt} &= -g_1\hat{m}_0(V)(V-1) - g_2\hat{w}(V-V_2) - g_l(V-V_l) + I \\ \frac{d\hat{w}}{dt} &= -\frac{1}{\tau(V)}[\hat{w} - w_0(V)] \end{aligned} \tag{2.11}$$

The voltage has been scaled so that one of the reversal potentials is unity; time is measured in units of $\tau = RC$. Comparing these equations to those by Hodgkin and Huxley, we can set $\hat{w} = (w/a)^4$ and $\hat{m}_0 = [m_0(V)]^3$. The difference between equation 2.9 and 2.11 is the absence in this second model of the "blocking" term $(b-w)$. Morris and Lecar approximated the equilibrium functions shown in figure 2.3 with:

$$\begin{aligned} m_0(V) &= \frac{1}{2}[1 + tanh(\frac{V-V_1}{V_2})] \\ w_0(V) &= \frac{1}{2}[1 + tanh(\frac{V-V_3}{V_4})] \end{aligned} \tag{2.12}$$

where V_1,...,V_4 are parameters. The time constant is approximated by

$$\tau(V) = \frac{\tau_w}{cosh(\frac{V-V_3}{V_4})} \tag{2.13}$$

with a further parameter τ_w.

Morris-Lecar model 2.11-2.13 gives a phenomenological description of action potentials.

2.1.5 FitzHugh-Nagumo model

FitzHugh and Nagumo were probably the first to propose that, for a discussion of action potential generation, the four equations of Hodgkin and Huxley can be replaced by two of the form of 2.9 and 2.10. They obtained sharp pulse-like oscillations reminiscent of trains of spikes by defining the functions $F(V,w)$ and $G(V,w)$ as

$$F(V,w) = V - \frac{1}{3}V^3 - w \tag{2.14}$$

$$G(V,w) = b_0 + b_1 V - w \tag{2.15}$$

where V is the membrane potential and w is a recovery variable [FitzHugh, 1961] [Nagumo et al., 1962]. F and G are linear in w and the sole non-linearity is the cubic term in V.

2.1.6 IF model

The two-dimensional models described above still are not our models, typically too complex [Patel and DeWeerth, 1997] [Linares-Barranco et al., 1991] for our goal of a compact VLSI implementation. Starting again from the Hodgkin and Huxley model, (this process can be easily adapted to other complex detailed models) further simplifications are, clearly, possible and they lead to a one-dimensional integrate-and-fire model. The analytical tractability of this effective description together with its simple Silicon implementation will make it the favorite candidate to realize and study, both theoretically and experimentally, controllable neural networks.

The reduction of the model is based on simplifications on the output and on the input side of the Hodgkin and Huxley model.

On the output side the major consideration is that an Hodgkin and Huxley neuron will typically emit a spike whenever its membrane potential reaches a threshold value of about -55 to -50mV. The action potential produced is roughly always the same, independently from the evolution of the input currents that have triggered the spike. The spike has a stereotyped shape that seems not to convey important information, which will arise from the time of spike occurrences. This suggests that the generation of the spike can be discorporated from the equations and reduced to a pure boundary condition so that when the membrane potential $V(t)$ crosses a given threshold θ_V then $V(t)$ undergoes a pulse-like excursion, the spike, before returning to a resting value V_r. The costly numerical integration is then stopped as soon as the spike is triggered and restarted after the downstroke of the spike about 1.5-2ms later. This interval of time corresponds to an absolute refractory period (τ_{abs}) of the neuron. This reduction clearly simplifies the equations that have now to describe only the subthreshold behavior of the potential and no more the delicate equilibrium among conductances dynamics that account for the spike generation.

On the input side, we still have all the four variables V, n, m and h of the Hodgkin and Huxley model. One can distinguish the variables into two classes, those that are either fast as compared to V or slow. With a little bit of arrogance, one can replace the fast variables, in our case m, by their steady state values $m_0[V(t)]$, as already done in the two-dimensional models, and the slow variables, n and h by their averaged values n_{av} and h_{av}. Rewriting equation 2.4 with these approximations we have

$$C\frac{dV}{dt} = -g_{Na}[m_0(V(t)]^3 h_{av}(V - V_{Na}) - g_K n_{av}^4 (V - V_K) - g_l(V - V_l) + I(t) \tag{2.16}$$

or, dividing by C,

$$\frac{dV}{dt} = F(V) + \frac{I(t)}{C} \qquad (2.17)$$
$$\text{If } V(t) = \theta_V, \text{then } V \rightarrow V_r$$

Which is the generic formulation of the integrate-and-fire model. The basic IF model was originally proposed by Lapique in 1907 [Lapique, 1907], long before the mechanisms that generates the action potential were understood; an entire series of non-linear IF models have been proposed to better reproduce the subthreshold behavior of particular classes of neurons.

The function $F(V)$ could be further simplified and reduced for instance to a linear function: these is possible discarding all the active conductances and maintaining only the constant one g_l of the leakage current. This means that the membrane potential is stimulated only by the synapse contributions, and merely discharged through a fixed conductance. This version of the model is called the passive or *leaky* integrate-and-fire (LIF) neuron. For small fluctuations about the resting membrane potential, neuronal conductances are approximately constant [Dayan and Abbott, 2001]; the LIF model assumes that this constancy holds over the entire subthreshold range. For some neurons this is a reasonable approximation, and for others it is not. With these approximation the subthreshold behavior of the model corresponds to that of the circuit in figure 2.7 consisting of a fixed capacitor in parallel with a fixed resistor driven by a current $I(t)$: a simple RC circuit. $I(t)$ splits in two components,

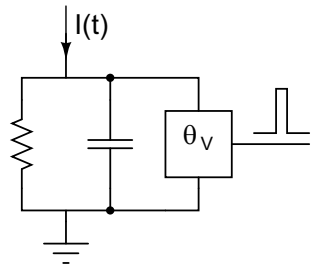


Figure 2.7: Schematic view of the *Leaky integrate-and-fire* neuron: the RC model. In this version of the one-dimensional integrate-and-fire model the membrane potential (the voltage across the capacitor) is excited by synaptic current $I(t)$ and merely discharges through a fixed resistance. The RC circuit models the subthreshold behavior of the membrane potential, the action potential is reduced to a boundary condition specified by equation 2.18. A spike under the form of a digital pulse, is generated when the voltage across the capacitor, i.e. the membrane potential, reaches the threshold Θ_V.

one across the capacitor, one across the resistor, thus:

$$\frac{dV}{dt} = -\frac{V(t)}{\tau} + \frac{I(t)}{C} \qquad (2.18)$$
$$\text{If } V(t) = \theta_V, \text{then } V \rightarrow V_r \qquad (2.19)$$

where $\tau = RC$.

The LIF neuron represents a drastic simplification of the Hodgkin and Huxley model reducing from four to one the number of equations involved. Intermediate solutions have also been proposed, among which some reduce the dynamics to a two dimensional problem. Among all these available models, the LIF neuron maintains its own appeal, both from a theoretical point of view for its analytical tractability and from an electronic point of view for its simple implementation.

If on one side, the discrete electronics implementation of the LIF model is very simple, in CMOS technology a resistor is not a desiderate component. A linear resistor is usually designed as a long and tight rectangle of a heavily doped polycrystalline Silicon, the so-called polysilicon layer, folded as a snake to maintain it compact. The problem is that the resistance coefficient of the polysilicon is low and to have a resistance of a reasonable value large Silicon area should be used. Moreover VLSI designer do not like passive resistors because together with their dimensions they bring a non negligible continuous power dissipation. Few CMOS processes make available a "high" resistance layer, explicitly thought for resistors; their layout become smaller but still they do not represent a practicable way to pack thousands of neurons together in a single chip. Active components can be connected to act as resistors, but their linear range is usually too small to be successfully used [Mead, 1989]. In short, the resistor has to be removed to design a compact, power efficient neuron.

Gernstein and Mandelbrot [Gerstein and Mandelbrot, 1964] suggested that the neuronal subthreshold activity could be described as a random walk based process towards an absorbing barrier [Ricciardi, 1977] . The depolarization dynamics then becomes:

$$\frac{dV}{dt} = \frac{I(t)}{C}$$

where the synaptic current $I(t)$, the sum of excitatory and inhibitory synaptic contributions, drives the random walk of the depolarization. This further simplification of the model ignores the decay of the biological neuron, an important feature that reduces the memory of recent interactions with the other neurons. The decay can be reintroduced as a constant leakage term:

$$\frac{dV}{dt} = -\beta + I(t) \qquad \beta \geq 0 \tag{2.20}$$

$$\text{If } V(t) = \theta, \text{then } V \to V_r \tag{2.21}$$

$$\text{If } V(t) \leq V_{min}, \text{then } V \to V_{min} \tag{2.22}$$

This model, endowed with a lower bound for the membrane potential, has been introduced by Carver Mead in 1989 [Mead, 1989]; it is an ideal solution for what concerns the VLSI implementation on Silicon. We will call this model VIF, VLSI integrate-and-fire neuron. Despite the linear decay, the model has proved to reproduce a rich phenomenology similar to that of the LIF, both at single unit and network level [Fusi and Mattia, 1999]. As motivated in next sections, the synaptic current $I(t)$ can be reduced to the form:

$$I_{\text{syn}} = I_{\text{exc}} + I_{\text{inh}}$$

$$I_{type} = \sum_{j}^{C_{type}} J_j \sum_{k} \delta(t - t_j^k - \delta_j). \tag{2.23}$$

where J is the synaptic efficacy, *type* alternatively stands for excitatory or inhibitory, t^k is the time of arrival of the k-th presynaptic action potential ($\delta(x)$ stands for the Dirac's

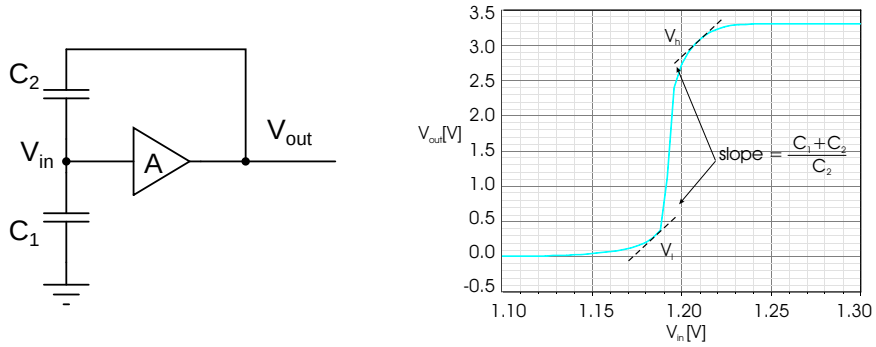
delta function), δ is the synaptic delay that account for the time needed to a spike to travel along the axon and reach the synaptic contact and C is the total number of synapses of one kind belonging to the same dendritic tree. J is a fixed parameter, positive for excitatory synapses and negative for the inhibitory ones. The membrane dynamics of the i-th neuron in a network of N IF neurons is:

$$\dot{V}_i = -\beta + \sum_j^{C_i} J_{ij} \sum_k \delta(t - t_j^k - \delta_{ij})$$

$i = 1, 2, \ldots N$.

2.1.7 IF model on Silicon

The simplest circuit implementing the VIF model, originally proposed by Mead, is reported figure 2.8.



(a) (b)

Figure 2.8: Axon-Hillock circuit schematic (a) and characteristic of the non inverting amplifier (b). This simple circuit, originally proposed by Mead [Mead, 1989] reproduces the VIF model described by eq. 2.20. The feedback through the capacitive divider C_1C_2 is responsible for the generation of a fast rising of the output voltage V_out when the input V_{in} reaches the level V_l shown in the right panel. For a complete action potential a resetting mechanism has to be added : a detailed view of the complete Axon-Hillock circuit is reported in fig. 2.9.

From the detailed Hodgkin and Huxley model it maintains the idea of the positive feedback to generate the spike. As for the Sodium a membrane depolarization induces an increase of the conductance, which, in turn further depolarize the membrane, here an augment in the voltage at node V_{in}, due to the synaptic current I_{syn}, causes an increase in the output V_{out} of the non inverting amplifier, and the positive feedback via the capacitor C_2 induces a further growth of V_{in}. A sudden raise of V_{out} is the result. The increase of V_{out} is reported in input through the capacitive voltage divider C_1C_2. Assuming that the feedback is sufficiently fast that a negligible amount of charge can flow into the capacitors, hence under the hypothesis of conservation of charge, the variation in V_{out} is reported in input multiplied by a factor $C_2/(C_1 + C_2)$:

$$\Delta V_{in} = A \frac{C_2}{C_1 + C_2} \Delta V_{out}.$$

Where A is the amplifier gain (see figure 2.8). Hence the positive feedback starts when V_{in}

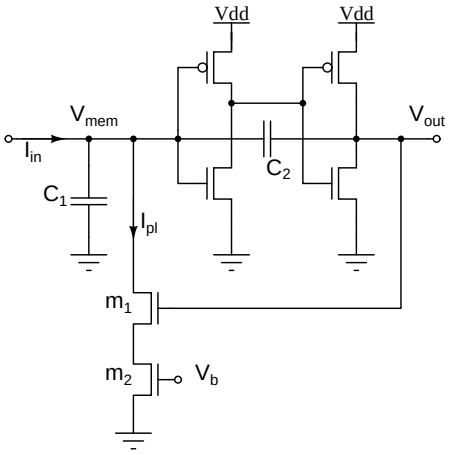


Figure 2.9: MOSFET level schematic of the Axon-Hillock circuit. The non-inverting amplifier of figure 2.8 is composed of two simple inverters in serie; I_{in} is the synaptic current and V_{mem} is the neuron membrane potential. The resetting mechanism is implemented with the two transistors m_1 and m_2. The first one acts as a switch that activates when V_{out} goes high, while m_2 controls the maximum value of I_{pl} (according to the bias V_b) and hence the duration of the spike pulse.

passes the voltage V_l for which $A = (C_1 + C_2)/C_2$ so that the gain over the loop become greater than one. The circuit makes a decision and goes for the generation of a spike, i.e. the output rises till Vdd. To complete the pulse, and to bring back the circuit to the initial situation, we need something to discharge the capacitors. Mead introduced a couple of mosfet m_1 and m_2 (see figure 2.9). The former is turned on by the output raise, the latter acts as a current controller and imposes the maximum amplitude of the current I_{pl}. When V_{out} equals Vdd, and hence the amplifier gain is null, the effect of I_{pl} starts to be important and V_{in} discharges towards ground at a rate

$$\frac{dV_{in}}{dt} = -\frac{I_{pl}}{C_1 + C_2}$$

When V_{out} passes V_h (see figure 2.9) the loop gain is again greater than one and a positive feedback starts. But now the effect is in the opposite direction, thus bringing V_{out} and V_{in} to ground. This reset mechanism has an effect analogous to the one induced by the Potassium currents in the Hodgkin and Huxley model. Injecting a constant I_{syn} current, the Axon-Hillock circuit will generate a regular train of spikes as in figure 2.10.

Starting from this circuit an entire generation of IF neurons were designed and successfully used in VLSI neural networks each one optimizing certain aspects or introducing new features [van Schaik, 2001] [Culurciello et al., 2001] [Schultz and Jabri, 1995] [Badoni and Annunziato, 1996].

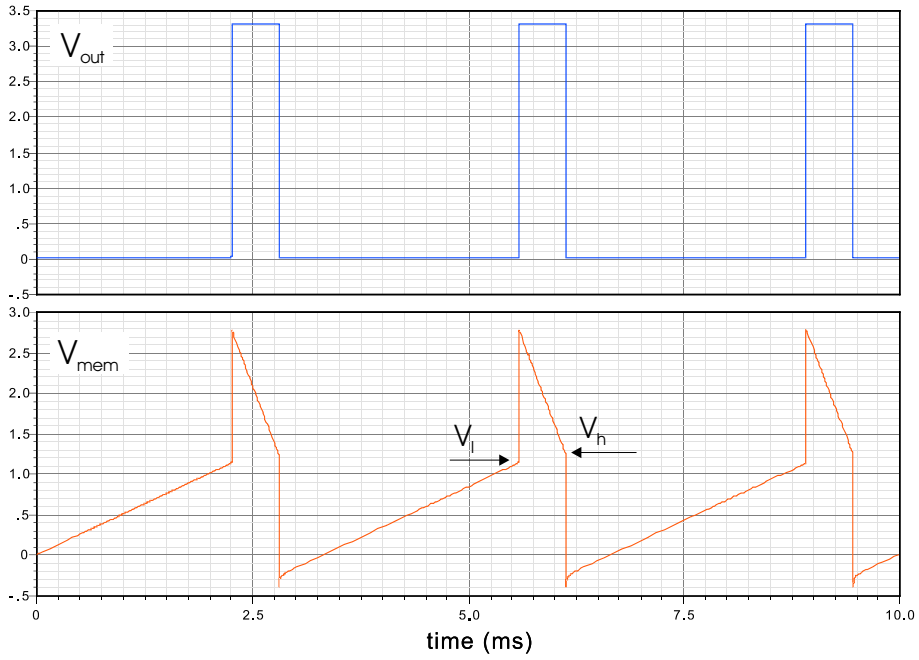


Figure 2.10: Simulation of the Axon-Hillock circuit: injecting a constant current I_{in} the circuit produces in output a regular spike train. The circuit performs an analog-to-digital conversion: it integrates the analog synaptic current at the node V_{mem} (lower panel) and when the threshold is reached it generates in output (V_{out}) a digital pulse. On the lower panel are shown the points V_l and V_h where the feedback become positive.

2.2 Synapses

Interactions between neurons are mediated by small, specialized junctions termed synapses. Their function is to control the conductance of the membrane separating the interior of the postsynaptic cell from the extracellular fluid. This conductance is controlled by the potential across the presynaptic membrane. The synapse ability of controlling current into or out of one electrical node by the potential on another node is the key ingredient that makes all information processing possible and, in this sense, a single synapse is the neural counterpart of a transistor.

A first classification of synapses can be based on the type of the transmission that take place, so that we have electrical and chemical synapses. The former type is represented by communicating junctions which connect the cytoplasm of two adjacent cells allowing the exchange of small molecules. They seem to recover a role in the synchronization of neurons activity. More interesting are the chemical synapses where the interactions between the pre and the postsynaptic cell is mediated by neurotransmitters. Triggered by the arrival of a presynaptic spike, the action of a chemical synapse involves a series of steps that, extremely simplified, reduce to this sequence (see figure 2.11): the influx of Ca^{++} ions into the presynaptic terminal starts a series of reactions that bring to the fusion of a synaptic vesicle with the plasma membrane; each vesicle contains a certain amount of transmitter molecules

which, further to the vesicle fusion, diffuse across the narrow synaptic cleft separating the pre and postsynaptic cell; the neurotransmitter acts on receptor molecules in the postsynaptic membrane leading in some cases to direct gating of conductance at an *ionotropic* receptor. This alters the corresponding ionic flux which, in turn, changes the postsynaptic membrane potential. The transmitter molecule may also activate a *metabotropic* receptor linked to a second-messenger pathway that modulates a membrane conductance or has other metabolic effects. In any case the receptor activation results in the opening (or closing) of certain ion channels and, thus, in an excitatory or inhibitory effect (EPSP of IPSP). The effect ends when the concentration of the transmitter in the synaptic clef is reduced by enzyme-mediated degradation and presynaptic uptake mechanism. We are, again, dealing with conductances

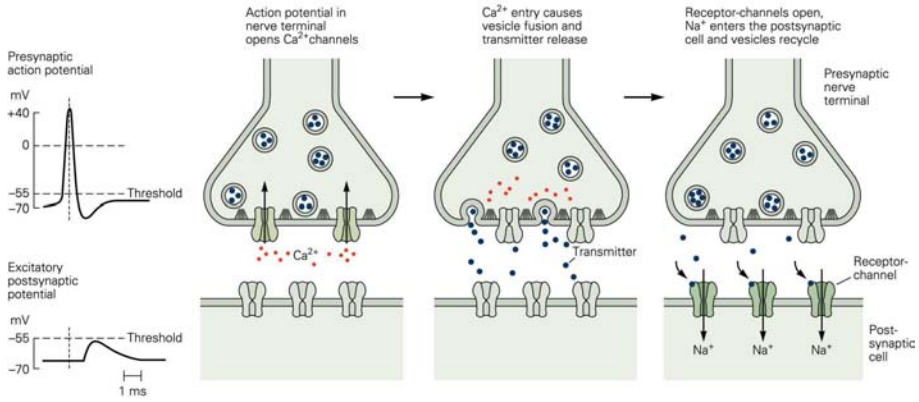


Figure 2.11: A chemical synapse at work. The arrival of a presynaptic spike triggers a series of steps that, extremely simplified, reduces to this sequence: the influx of Ca^{++} ions into the pre-synaptic terminal brings to the fusion of synaptic vesicles with the plasma membrane; vesicles contains the neurotransmitter that diffuse across the synaptic cleft; the neurotransmitter acts on receptor molecules in the postsynaptic membrane leading in some cases to direct gating of conductances at an *ionotropic* receptor. This alters the corresponding ionic flux which, in turn, changes the postsynaptic membrane potential.

and currents, and hence, as before

$$I_{syn}(t) = g_{syn}(t)(V - V_{syn})$$

where V_{syn} is the reversal potential and the function $g_{syn}(t)$ is used to characterize different kind of synapses.

The major neurotransmitters in the brain are glutamate and γ-aminobutyric acid (GABA). The former has an excitatory effect, i.e. it induces a depolarization of the postsynaptic membrane; the latter is an inhibitory transmitter which causes an hyperpolarization of the postsynaptic neuron. On the postsynaptic side the GABA receptors are associated with a reversal potential around -75mV. The $GABA_A$ receptors control the Cl^- conductance which reacts on a timescale of 5ms, while $GABA_B$ receptors produce a longer effect on the K^+ conductances which decays with a time constant of about 50 ms. For the glutamate there are the so-called AMPA and NMDA receptors, both with reversal potential around 0mV. The relevant difference between the two is the timescale on which they react: the AMPA

receptors generate a very fast rising ($\tau_{rise} \simeq 0.1ms$) and decaying current ($\tau_{rise} \simeq 0.1ms$). On the other side the NMDA receptors exhibit a slower but also richer behavior. Their state does not depends only on the the transmitter concentration but also on the postsynaptic membrane depolarization. Mg^{++} ions block the NMDA channels until the membrane potential rises above -50mV. Once unblocked, in presence of glutamate the channels open and stay open for 10-100 milliseconds. Hence the NMDA channels activate only when there is a coincidence between the arrival of a presynaptic spike and the depolarization of the postsynaptic membrane. NMDA-controlled channels are permeable both to Sodium and Potassium ions, but even more (five to tens times more) permeable to Calcium ions. Ca^{++} are known to have a relevant role in the modulation of intracellular signalling and are involved in the long-term modifications of synaptic efficacy.

Long- and short- term metabolic effects, in general, derive from the activation of second messengers by either *ionotropic* or *metabotropic* receptors and they lead to changes in synaptic transmission efficacy. In this way the synapse results a plastic device which modulates its influence on the postsynaptic cell according to its history of activity. The long sequence of steps that take place in a synapse, lends itself to different modifications offering a number of biochemical reactions on which various mechanisms can act. Short-term effects refers to those phenomena that affect the synapse dynamics and last anywhere from milliseconds to tens of seconds. The two principal types are the short-term depression and the short-term facilitation which occur after a rapid sequence of presynaptic action potential. Long-term plasticity are, on the other side, extremely persistent (hours and more): long-term potentiation (LTP) and long-term depression (LTD) are the most prominent of these effects and are studied as the basis of learning and memory.

The best known mechanism for synaptic LTP is related to NMDA-controlled channels [Kauer and Malenka, 2007]: they are not selective channels and, as mentioned above, they allow an influx of Calcium ions into the dendritic spines. The rise in postsynaptic Ca^{++} concentration is the crucial trigger for LTP. It activates complex intracellular signalling that results in an increased number of AMPA receptors in the postsynaptic plastic membrane. Furthermore there is a growing evidence that LTP is accompanied by observable enlargements of dendritic spines. These structural changes may be essential to cement the information-storage process. LTP may also be related to modifications on the presynaptic membrane. The activity-dependent increase of Calcium concentration in presynaptic terminals starts a chain of reactions that leads to a persistent increase in the amount of glutamate released upon the arrival of an action potential and, hence, potentiate the excitatory synaptic transmission.

On the other side LTD can be induced by weak activations of NMDA receptors due, for instance, to modest membrane depolarization or low stimulation frequencies. The smaller rise of the Ca^{++} concentration triggers a set of Ca^{++}-dependent intracellular signalling different from those required for LTP. The result is a removal of AMPA receptors and hence a reduced effect of the excitatory transmission. The internalization of postsynaptic AMPA receptors is also triggered by activation of metabotropic glutamate receptors. Another mechanism that in some cells leads to LTD, involves the synthesis of lipophilic molecules that travel retrogradely across the synapse and, binding to particular receptors, depress neurotransmitter release. This process appears after a brief but strong influx of Calcium ions on the postsynaptic membrane.

Besides those synapse specific LTP and LTD processes, other mechanisms appear to regulate the strength of an elevated number of synapses. The widespread effects are thought

to be homeostatic responses that maintain the neurons activity within some finite range. They take place when the activity levels are changed for prolonged periods (hours or days). Specifically, enduring decreases in activity globally increase the synaptic strength.

In short, synapses are plastic devices whose transmission efficacy can vary according to their respective history of activity. The long-term effects last over hours and act in two opposite directions: the reinforcement (LTP) or the depression (LTD) of the transmission efficacy. They are thought to be responsible for learning and memory

2.2.1 Fixed synapses in a simple VLSI network

The synapse is such a complex biological system that designing a detailed electronic counterpart seems not a practicable way. As for the neuron, the synapse could be reduced to a circuit with voltage-controlled conductances, one of the most sophisticated example of a VLSI implementation is reported in [Bartolozzi and Indiveri, 2007]. To realize a compact circuit we chose to forsake the time-course description of some processes and retain only few key features. To this end a first approximation consists in neglecting the detailed behavior of the fast PSPs mediated by the AMPA and $GABA_A$ receptors and to consider them as point effect occurring upon the arrival of a presynaptic spike. A second step is ignoring the slow PSPs as those due to NMDA and $GABA_B$ receptors. The long-term changes in the synaptic efficacy will be reintroduced in a different manner. Under this conditions the synaptic current is described by equation 2.23. This kind of transmission is based on simple digital switches rather than on time-dependent analog voltages and can be easily implemented on Silicon accepting that that the delay δ is the one that naturally comes with electronic analog circuits. Thus, the type of synaptic transmission and stereotyped nature of the action potentials suggest for spikes on Silicon the form of digital pulses.

Before reintroducing in the model the long-term plasticity, I would like to discuss a simple implementation of a network of integrate and fire neurons connected by fixed synapses. In [Fusi and Mattia, 1999] the authors studied such kind of networks and demonstrated both analytically and in simulation that the system dynamics, even in the case of a purely excitatory connectivity, has two stable fixed points, i.e. two state in which the activity is self-sustained and maintained over long timescales, one at low and one at high frequencies.

On Silicon a synapse with a fixed efficacy that follows equation 2.23 is a simple device composed of only two MOSFETs, one acting as a digital switch, the other as a current regulator. In figure 2.12 the schematic view of a dendritic tree composed of four fixed synapses connected to an Axon-Hillock circuit is reported. The first two synapses on the left are inhibitory synapses, the others are excitatory ones. The inhibitory synapses are made of n-type MOSFETs whose task is to suck current from the neuron capacitors upon the arrival of a presynaptic spike. This have an inhibitory effect on the postsynaptic neuron because it induces a decrease in the voltage V_{mem} moving it away from the spike emission threshold. The MOSFET m_1 acts as a switch; it is driven by the presynaptic spike spk_2 coming under the form of a digital active high pulse. m_1 enables current flowing from the common node V_{mem} to ground for the period of time ΔT during which spk_2 stays high. The amplitude I_{inh} of the current is limited by m_2 using a bias voltage so that the amount of charge subtracted from the neuron capacitors is $I_{inh}\Delta T$ and hence V_{mem} undergoes a downward jump

$$\Delta V = \frac{I_{inh}\Delta T}{C_1 + C_2} \tag{2.24}$$

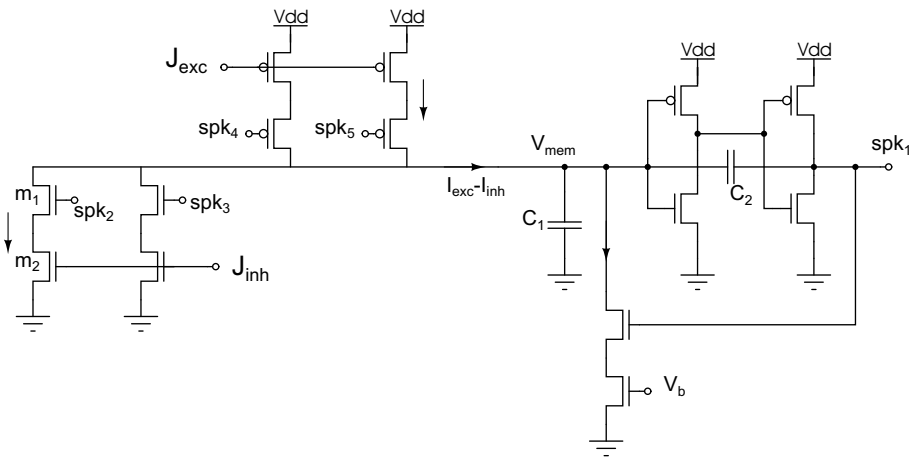


Figure 2.12: Schematic view of a simple dendritic tree composed of four fixed synapses and one postsynaptic neuron. Each synapse consists of two transistors: one receive the spike pulse ($spk_{2,3,4,5}$ signals) of a presynaptic neuron and acts as a switch which closes during the incoming spike, the second one acts as a current controller and limit the synaptic current injected (or subtracted) into (or from) the postsynaptic neuron according to the bias J_{exc} (J_{inh}). The first two synapses on the left are inhibitory, the other two are excitatory. The neuron circuit is the Axon-Hillock one discussed in section 2.1.7. The neuron output is the voltage node named spk_1.

that represents the synaptic efficacy. Excitatory synapses work in an analogous way: when they receive active-low spikes they inject current into the common node thus provoking upward jumps for the membrane potential V_{mem}.

Packing together in a single chip many dendritic trees as the one in figure 2.12 and connecting them together as shown in figure 2.13, a simple neural network of integrate-and-fire neurons can be realized. It represents the basic idea for the VLSI networks described in chapter 3 and 4. Actually it is both a very simple and compact scheme; it is completely asynchronous and no intermediate structures are required to connect the various parts; neurons emit spikes under the form of digital voltage pulses, synapses convert them into analogical current contributions that sums together on the dendritic tree to stimulate the postsynaptic neuron. All the circuits operate in parallel and, clearly, in real time.

I would like to underline that the network in figure 2.13, whatever its dimensions, does not suffer from any communication problems as insufficient bandwidth or scant address space that typically affect parallel architectures that exploit multiplexed channels to connect the various units. Two factors make this possible 1) each neuron sends spikes over its own communication channel, the axon, and 2) the various synaptic contributions are analog current pulses that just sum together according to Kirchhoff's law. It is the solution that the biological evolution adopted for the nervous system. The choice is to spend energy and space occupancy for tons of wires connecting cells together rather than developing incredibly fast multiplexed serial link.

However the simple network in figure 2.13 has many limitations: first of all an additional

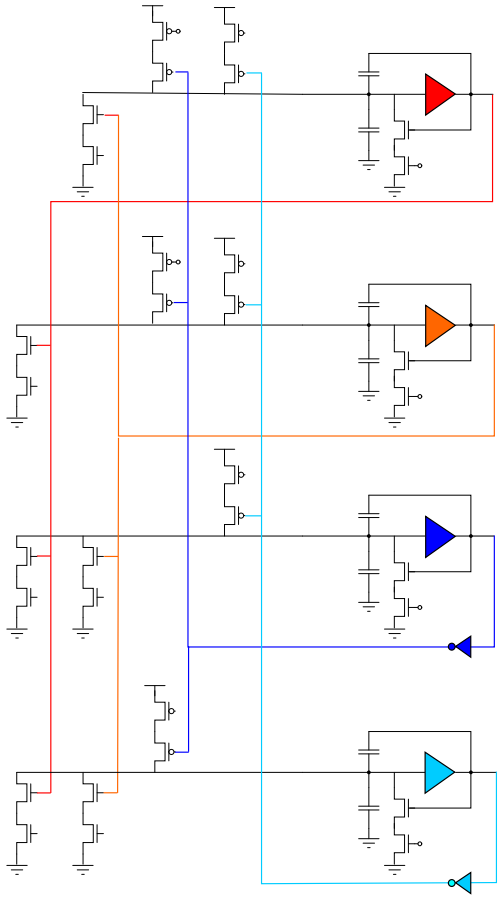


Figure 2.13: Schematic view of a simple network of four full connected neurons designed coupling together four dendritic branches as the one reported in fig. 2.12. The network comprises 12 fixed synapses among which 6 inhibitory and 6 excitatory synapses. Networks designed following this scheme, whatever the dimensions of the system, do not suffer of any communication problems as insufficient bandwidth or scant address space that typical affect parallel architectures that exploit multiplexed channels to connect the various units. Two factors make this possible 1) each neuron sends spikes over its own communication channel, the axon, and 2) the various synaptic contributions are analog current pulses that just sums together according to Kirchhoff law.

interface for reading/writing input/output signals to it is required, second some parameters are not tunable at all as for instance the firing threshold of the neurons, and some others are coupled together as for example the spike duration ΔT, that represents the absolute refractory period, takes part in the definition of the synaptic efficacy. From an experimental point of view it would be better to have a larger number of independent tunable parameters. Said that, this simple network has been successfully used as part of a larger system endowed with a set of plastic synapses [Chicca et al., 2003]. Plastic synapses, much more complex than the two-MOSFETs fixed ones, can modify their efficacy J according to some kind of

so-called *learning rule*.

2.2.2 Plastic synapses

To learn a stimulus its presentation has to leave some kind of traces in the synaptic connectivity. It has to modify the way the neurons interact among themselves, that is it has to modify the synaptic efficacy. The neuronal correlate of learning is a process of adaptation of the parameter J; the procedure for adjusting the synaptic weight is referred to as a *learning rule*. How does this mechanism work is one of the current major research topic. Tons of models have been developed, accounting for different kind of synaptic plasticity based on the mean firing neurons activity or on precise spike timing. Here we will focus on the model implemented in the neuromorphic chip described in the next chapter; the synaptic model, proposed in [Fusi et al., 2000a] is a bistable (only two values for J) Hebbian synapse with stochastic learning endowed with a self-regulating mechanism [Brader et al., 2007]. The various choices that lead to this model are described in what follows.

In 1949, in its 'The Organization of Behavior' Hebb stated a basic idea that is still today an important benchmark:

"When an axon of a cell A is near enough to excite B or repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cell firing B, is increased."

The class of synaptic changes that are driven by correlated activity of pre and postsynaptic neurons is usually called *Hebbian learning*: (Even though the concept of learning through correlations was originally proposed by James in 1890.) Almost 60 years ago Hebb formulated his principle as a pure postulate without any experimental evidence. He understood that strengthening the coupling of two active neurons corresponds to the stabilization of specific neuronal patterns of activity. In 1973 Bliss and Lømo [Bliss and Lømo, 1973] experimentally proved Hebb's hypothesis. Today, as described above, electrophysiological experiments show that the NMDA controlled channels could play an important role in such a correlation detection mechanism. In 1997 Markram and his colleagues [H et al., 1997] documented, studying pyramidal neurons, the fact that the coincidence of postsynaptic action potentials and excitatory postsynaptic potentials (EPSPs) induce changes in EPSPs amplitude. Markram's idea, profoundly different from the Hebbian rate-based hypothesis, stated that if a postsynaptic action potential occurs 10ms before the EPSP, than the EPSP amplitude is reduced; on the other side, if the postsynaptic spike occurs 10ms after the onset of the EPSPs, than the synaptic efficacy is increased. This idea is usually named spike timing dependent plasticity (STDP). Whether synaptic changes are driven by precise spike timing or follow a Hebbian rule is still a matter of debate. In 2001 Sjostrom et al. [Sjöström et al., 2001] combined rate-based and STDP ideas; they also introduced a dependence on the instantaneous value of the postsynaptic depolarization and obtained a model able to reproduce their experimental data. The ingredients of their model were:

1. STDP. If a presynaptic spike precedes a postsynaptic action potential within a given temporal window, the synapse is potentiated, and the modification is stable on long timescales in the absence of other stimulation. If the phase relation is reversed, the synapse is depressed.

2. Dependence on postsynaptic depolarization. The postsynaptic neuron needs to be sufficiently depolarized for LTP to occur.

3. LTP dominance at high frequencies. When both pre and postsynaptic neurons fire at elevated frequencies, LTP always dominates LTD regardless of the phase relation between the pre and postsynaptic spikes.

The synaptic model we implemented in the chip, proposed in [Brader et al., 2007] and detailed in the next section, includes the elements listed above plus an additional regulatory mechanism that prevents undesired synaptic changes. The STDP behavior can be achieved using a combination of depolarization dependence and an effective neuronal model as in [Fusi et al., 2000a] and [Fusi, 2003]: when a presynaptic spike shortly precedes a postsynaptic action potential, it is likely that the depolarization of an integrate and fire neuron is high, resulting in LTP. If the presynaptic spikes comes shortly after the postsynaptic action potential, the postsynaptic integrate and fire neuron is likely to be recovering from the reset following spike emission, and it is likely to be hyperpolarized, resulting in LTD. Only one variable $X(t)$ is necessary to model the synaptic dynamics, provided that $X(t)$ is modified on the basis of the postsynaptic depolarization $V(t)$. This synaptic model displays Hebbian behavior: LTP dominates at high postsynaptic firing rate and LTD dominates for low postsynaptic firing rate [Brader et al., 2007].

During the presentation of each stimulus a pattern of activity is imposed onto the network and each synaptic weight J is updated to encode the information carried by the pre and postsynaptic neuron. The question now is the number of accessible values for J: can J sweep a continuous analog range or should assume only a subset of discrete values? On short timescales it may be reasonable to hypothesize that a synapse can modify its efficacy in an analog way, while, on long timescales, if memory is to be maintained, it is more likely that only a small set of stable states can be preserved. A discrete set of values for J could have a biological basis in the *quantal* nature of the mechanism for neurotransmitter release which is based upon the fusion of a discrete number of vesicles with the presynaptic membrane. Pushing this reasoning to its extreme one can argue that only two different synaptic weights are possible, the synapse becomes a bistable device [Amit and Fusi, 1994] that will be potentiated if its efficacy is high, or depressed when its efficacy is low. On long timescale synapses act as digital device able to store one bit of information.

If we assume that the information to be coded is redundant, there are no disadvantages in using a discrete code. Let consider the current generated by the synapses as the relevant variable. We assume that it is the linear sum of many input neuronal activities a_i multiplied by the corresponding weights J_i. Let I_N be the current induced by N neurons that encode the same information, i.e. that are activated in the same way by a generic stimulus ($a_i = a$ for $i = 1, ..., N$):

$$I_N = \frac{1}{N}\sum_{i=1}^{N} J_i a_i = \frac{a}{N}\sum_{i=1}^{N} J_i \tag{2.25}$$

Starting from fully analog synaptic values and then clamping them to the closest stable states, the error on I_N goes as $\sim 1/(2\sqrt{N})$. If N is large enough (the code is redundant) the error become negligible and there is no relevant loss of information [Fusi, 2001].

The above constraints for the synapse severely reduce the storage capacity of the network generating the *palimpsest* property. The interference of novel stimulations with already acquired 'memories' may give rise to memory loss or irrelevant noisy inputs could modified

the constituted connectivity matrix. The result is a memory limited to a sliding window containing a certain number of stimuli; recent stimuli are best remembered, older stimuli outside the window are completely forgotten. The window width depends on how many synapses are changed following each presentation: if this number is large, the network learns quickly, but the window could be so small to compromise the functioning of the system as an associative network. On the other side slow learning means also slow forgetting and a more equal distribution of synaptic resources [Giudice et al., 2003]. It is the stability-plasticity dilemma: the memory should be stable for long periods and, at the same time, the internal state should be rapidly modified to acquire the relevant information. This dilemma becomes particularly arduous when dealing with real electronic devices that do not allow arbitrarily large time constants or fine parameter tuning. [Fusi, 2001].

Over a long timescale the preservation of two well distinguished states appear easier than the maintaining of exact analog values, so a bistable synapse intuitively helps memory retention but, on the other side, it seems not to be the smartest choice to solve the stability-plasticity dilemma. To be learnt a stimulus has to leave a sign on the matrix of connections, which means that it has to modify the state of a certain number of synapses. If this mechanism is deterministic so that the probability that a synapse is modified is 1, than all the synapses reacting to that stimulus will be used to store the information. If later the same synapses are required to learn a novel stimulus the memory of the previous will be completely disrupted.

Under the assumption that the information to be coded is redundant, i.e. a certain number of synapses will store similar information, a possible escape is based on the fact that only a subset of those synapses is necessary to retain the stimulus memory. In other words, there is no need to modify all the synapses and, if the fraction of synapses that are changed following each stimulus is small, it is possible to better redistribute the synaptic resources among the different patterns and actually recover the optimal storage capacity [Amit and Fusi, 1994]. To reduce the fraction of modified synapses, the idea is to pass from a deterministic to a stochastic learning. And if the probability that a synapse, for instance, passes from potentiated to depressed is much less than one, than only a small subset of synapses would change their state to encode the novel information. The rest of the synapses can be used to memorize other information.

The stochastic learning, under the hypothesis of a redundant code, reduces the problem of interferences between new and old memories. The question now is how to design a simple system capable of stochastic learning. Probably not by chance the required source of noise can be find in the neuron spiking activity. The noise produced by a network of coupled neurons has the great advantage to be available to each synapse which can easily detect the activity of both the pre and postsynaptic neurons. It has been shown that, in VLSI device, this noise is suitable to be the basis of a stochastic learning rule [Chicca and Fusi, 2001].

2.2.3 Effective model of a plastic bistable synapse

The model we adopted exploits the noisy neuronal spiking activity to generate low probabilities for long-term synaptic changes (LTP or LTD) [Fusi et al., 2000a]. An internal synaptic variable $X(t)$ experiences a noisy time course that consists in a series of small upward and downward jumps triggered by the arrival of a presynaptic spike. For this reason this kind of synapse is called spike driven. The upward jumps occur when, upon the arrival of a presynaptic spike the instantaneous value of the postsynaptic membrane potential $V_{\rm post}(t)$

is found above a certain threshold θ_V, the downward jumps occur when $V_{\text{post}}(t)$ is found below θ_V. The state of the synapse, and hence its efficacy, is dictated by the comparison of $X(t)$ with another threshold θ_X. If $X(t)$ is above θ_X than the synapse is potentiated, if below, the synapse is depressed. The time course of $X(t)$ depends, in this way, on the statistical properties of the activity of the two neurons. For instance, synaptic potentiation will happen when the probability of having upward jumps P_{up} is sufficiently higher than the probability of having downward jumps P_{dw}. P_{up} and P_{dw} depends on the distribution probability of the postsynaptic membrane potential and on the the relative timing of the pre an postsynaptic neurons activities. In short, the tendency of $X(t)$ of moving towards its upper or its lower bound, and consequently the transition probability, can be tuned controlling the neuronal activity. In particular this synapse acts as a frequency meter of the pre and postsynaptic neurons [Giudice and Mattia, 2001]. The choice of an upward or a downward jump comes from the detection of a coincidence of two events, one related to the presynaptic neuron (the emission of a spike) and one related to the postsynaptic neuron (its depolarization). This mechanism guarantees the possibility of having very low transition probabilities that are essential to achieve high memory capacity.

Between two presynaptic spikes a driving force attracts $X(t)$ towards its upper or lower bound depending on where the last jump left $X(t)$ respectively above or below the threshold θ_X. This refresh tends to maintain the synaptic state stable on long time period.

The complete synaptic dynamics can be summarized as:

upon the arrival of a pre-synaptic spike

$$\begin{aligned} X(t) &\rightarrow X(t) + J_{up} \quad \text{if} \quad V_{post}(t) > \theta_V \\ X(t) &\rightarrow X(t) - J_{dw} \quad \text{if} \quad V_{post}(t) \leq \theta_V \end{aligned} \tag{2.26}$$

in the absence of impinging spikes

$$\begin{aligned} X(t) > \theta_X &\rightarrow J = J_+ \quad \textit{and} \quad \dot{X}(t) = \alpha \\ X(t) < \theta_X &\rightarrow J = J_- \quad \textit{and} \quad \dot{X}(t) = -\beta \end{aligned} \tag{2.27}$$

where J_{up} and J_{dw} are respectively the upward and downward jumps amplitudes, J_+ and J_- are the potentiated and depressed synaptic efficacy, α and β are the refresh rates.

Network of linear integrate-and-fire neurons and Hebbian plastic bistable synapses are non trivial systems; they are able to reproduce the elevated spike rates observed during neurophysiological experiments throughout the delay interval between successive visual stimuli [Fusi and Mattia, 1999]. The prototypical experimental protocol to investigate delay activity and *working memory* is the delayed match to sample (DMS) task (for reviews see [Miyashita and Toshiro, 2000, Wang, 2001]): a trial starts with the presentation of one visual image; after a delay period of several seconds another image is presented and the monkey has to respond differently if the second stimulus is identical to the sample or not. The activity during the delay is triggered by specific visual stimuli and it lasts for long periods after the removal of the sensory stimulus. In the inferotemporal cortex the sustained activity is stimulus specific, i.e. each visual stimulus evokes a particular pattern of activity.

The experimental findings of DMS experiments can be interpreted as an expression of an attractor dynamics in the cortical module. A comprehensive picture which connects the pattern of delay activities to the recall of memories into active states has been proposed in [Amit, 1995]. The basic idea is that the sustained activity is not a single-cell property

but rather a cooperative effect related to a self-maintaining feedback that results from the structured synaptic connectivity. A learned stimulus leaves a synaptic engram of potentiated excitatory synapses connecting the cells driven by the stimulus. A successive presentation of the same stimulus re-activates this set of cells that cooperate to maintain elevated their firing rates even after the stimulus removal. Since many cells cooperate the delay pattern of activity is robust to stimulus error, i.e. even if few cells belonging to the self-maintaining group are not activated by the stimulus or are driven at the "wrong" frequency, following the removal of the stimulus, the network dynamics is attracted towards the "nearest" stable pattern of activity, that is the *attractor*. All the stimuli leading to the same network response are said to belong to the same *basin of attraction*. In [Giudice and Mattia, 2001, Giudice et al., 2003, Fusi, 2002] the authors show that an initially unstructured network of integrate-and-fire neurons and plastic Hebbian bistable synapses, under stimulation, autonomously develops a synaptic structure supporting both spontaneous and stimulus-specific stable activities.

2.2.4 VLSI implementation of the effective synaptic model

The synaptic model described above results to have a spike-driven, rate-based, Hebbian behavior suitable for stochastic learning and thought to be implemented in VLSI device. The time course of $X(t)$ can be thought as a random walk between the two stable states [Ricciardi, 1977] [Holden, 1976] [Gerstein and Mandelbrot, 1964] [Brunel et al., 1998]. Such a noisy behavior is the basis to implement, on real small electronic devices, the long time constants needed for LTP and LTD: in [Fusi et al., 2000a] the authors designed a VLSI circuit reproducing the theoretical model and successfully tested it in the range of plausible biological timescales. The schematic view of their VLSI synapse is reported in figure 2.14. The capacitor C_{syn} is the memory element of the synapse and the voltage across it is the internal synaptic variable $X(t)$.

Without descending in details (refer to [Fusi et al., 2000a]) three main blocks are visible: the *Hebbian block*, the *Refresh Term* and the *Dendrite*. The *Hebbian block*, upon the arrival of a presynaptic spike *preSpk*, induces the upwards or downwards jumps in $X(t)$ through the injection or subtraction of charges into or from the capacitor C_{syn}. The *Refresh Block* compares $X(t)$ with the threshold θ_X and forces $X(t)$ towards Vdd or towards ground. The *Dendrite*, according to whether $X(t)$ is above or below θ_X, excites the postsynaptic neuron with a small current I_1 or with a higher current $I_1 + I_2$. Compared to the fixed synapses of the simple network in figure 2.13, this synapse is much more complex consisting in 19 transistors in spite of two. This synapse can be seen as the juxtaposition of the two MOSFETs fixed synapses (m_2 and m_3) with a much larger circuit whose final effect is just deciding to activate or not, through the switch m_4 another branch in the *Dendrite Block*.

This synapse has been used in a first small VLSI network of 21 linear integrate-and-fire neurons connected by 60 excitatory plastic synapses and 35 inhibitory fixed ones [Fusi et al., 2000b, Chicca and Fusi, 2001]. This hybrid analog/digital VLSI system (see figure 2.15) has been named LANN21 where LANN stands for Learning Attractor Neural Network. In [Chicca et al., 2003] the authors demonstrate that following a suitable stimulation protocol the synaptic stochastic plasticity produces the expected pattern of potentiation and depression in the electronic network. The chip has been designed using a $0.6\mu m$, three metal layers, standard CMOS technology.

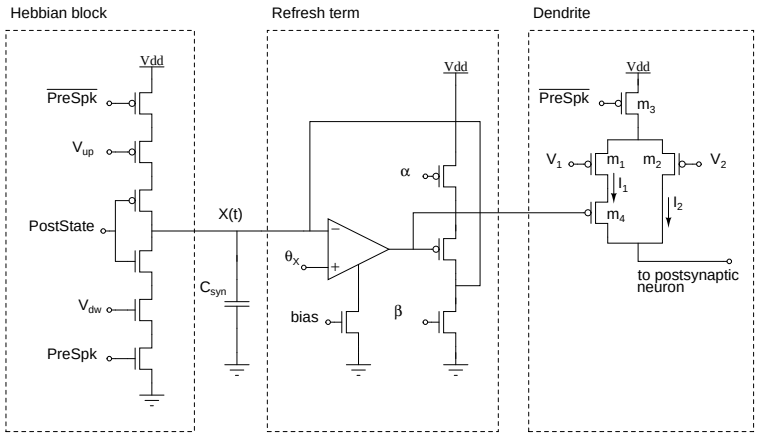


Figure 2.14: Schematic of a plastic synapse as proposed in [Fusi et al., 2000a]. The capacitor C_{syn} is the memory element of the synapse and the voltage across it is the internal synaptic variable $X(t)$. The three main blocks are the *Hebbian block*, the *Refresh Term* and the *Dendrite*. $X(t)$ evolves according to the current injected or subtracted into or from the capacitor by the *Hebbian block* and by the refresh circuit. When $X(t) < \Theta_X$ the synapse is depressed, otherwise the synapse is potentiated. In the first case a current I_2 will flow to the postsynaptic neuron, in the second case a larger current $I_1 + I_2$ will influence the postsynaptic neuron. For further details please refer to [Fusi et al., 2000a].

2.2.5 The Calcium *self-regulating* mechanism

In [Brader et al., 2007] the authors introduce a regulatory mechanism for the synapse whose aim is to improve the distribution of synaptic resources for the various stimuli to learn. The basic idea is that if a stimulus has already been learnt, no further synaptic modifications are required. Which is the reason to name this mechanism a *stop-learning* mechanism. Two issues come out: there should be something that 1) decides when the stimulus has been learnt and 2) that blocks any further synaptic transitions. Let consider a simple feed-forward architecture, a perceptron composed of an output neuron and its afferent synapses. Stimuli presentation consists in a set of trains of spikes impinging onto the synapses; different stimuli are coded in different mean firing rates assigned to the various spike trains. A simple perceptron is able to classifies linearly separable patterns of activity, so that after learning we expect that the firing rate of the output neuron will be, for instance, elevated when the active stimulus is recognize belonging to a certain class and low when the current stimulus is not belonging to that class. In this context a stimulus is learnt when the output neuron of the perceptron reacts with a sufficiently high firing rate: the solution to point 1) is then a measure of the postsynaptic neuron activity. To this end a new variable, called *Calcium variable* $C(t)$, has been introduced in the model:

$$\tau_C \frac{dC(t)}{dt} = -C(t) + J_C \sum_i \delta(t - t_i) \tag{2.28}$$

where the sum is over the postsynaptic spikes arriving at times t_i. It can be shown that the mean value of $C(t)$ is a good measure of the mean firing rate of the output neuron. The

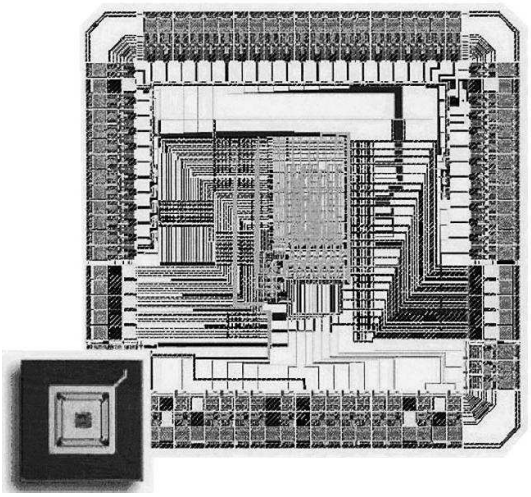


Figure 2.15: LANN21 Learning Attractor Neural Network layout view and photo. The chip has been designed using a $0.6\mu m$, three metal layers, standard CMOS technology. It implements a network of 21 integrate and fire neurons

value of $C(t)$ enters in the synaptic learning rule in this way:

upon the arrival of a pre-synaptic spike

$$\begin{aligned} X(t) &\rightarrow X(t) + J_{up} \quad \text{if} \quad V_{post}(t) > \theta_V \quad \text{and} \quad K_{up}^{low} < C(t) < K_{up}^{high} \\ X(t) &\rightarrow X(t) - J_{dw} \quad \text{if} \quad V_{post}(t) \leq \theta_V \quad \text{and} \quad K_{dw}^{low} < C(t) < K_{dw}^{high} \end{aligned} \tag{2.29}$$

in the absence of impinging spikes

$$\begin{aligned} X(t) > \theta_X &\rightarrow J = J_+ \quad and \quad \dot{X}(t) = \alpha \\ X(t) < \theta_X &\rightarrow J = J_- \quad and \quad \dot{X}(t) = -\beta \end{aligned} \tag{2.30}$$

where J_{up}, J_{dw}, J_+, J_-, θ_V, α, β and the thresholds K are all positive constants. The K_{up} define a range inside which the internal synaptic variable $X(t)$ can undergo upward jumps, while the K_{dw} define the range for the downward jumps. Upon the presentation of a stimulus, if the number of potentiated synapse is enough to drive the postsynaptic neuron at a sufficiently elevated rate, so that $C(t) > K_{up}^{high}$, than potentiation of further synapses is not necessary and upward jumps are blocked. The model is symmetric thus if the neuron is already too inactive further synaptic depotentiations are inhibited.

The *stop-learning* mechanism has a regulatory effect that tend to maintain the mean level of activity of a network in a given range. This synaptic model, together with integrate-and-fire neurons, has been tested in simulation [Brader et al., 2007] and demonstrated to improve the perceptron classification performances in particular when the stimuli to be learn have a relevant overlap, i.e. when different stimuli, coded in a set of spike trains, share a non-negligible subset of common trains. We carried on analogous experiments on a VLSI

chip, CLANN (Configurable Learning Attractor Neural Network). A detailed description of the chip and the obtained results are reported in the next chapter.

2.3 Conclusions

Summing up, the nervous system is a complex object, both at an architectural and at a single unit level. Neurons and synapses behaviors are the results of many concatenated biochemical reactions [Shepherd, 1998] [Kandel and Schwartz, 1985]. In trying to understand their functioning, a great variety of models have been investigated from very detailed to very abstract ones. Simple models of integrate-and-fire neurons and Hebbian plastic synapses represent the building blocks of networks able to reproduce some biological behaviors as the delayed activity observed in *Delayed Match to Sample* experiments [Amit, 1995]. The effective models described in this chapter have been thought for electronic VLSI implementation [Mead, 1989] [Indiveri, 2003] [Giudice et al., 2003] [Fusi et al., 2000a] which imposes strict limitations on the model complexity: limited range for the variables, small time constants, a restricted set of stable analog values on long timescale, a small number of manageable variables, noise limitations and so on. These constraints resulted as an additional drive to develop simple and effective models. Networks of integrate and fire neurons and spike-driven Hebbian plastic bistable synapses succeeded in demonstrating emerging properties of learning and memory maintenance [Giudice et al., 2003] [Fusi, 2002]. This triggered the interest towards VLSI devices implementing electronic recursive neural networks. With the previously proposed LANN21 chip the designers obtained interesting results even for a very small network [Chicca et al., 2003] [Fusi et al., 2000b]. In the next chapter a larger and more sophisticated VLSI implementation of a neural network is reported, together with the experimental results which prove that the chip behavior agrees with the theoretical models.

Chapter 3

CLANN

In this chapter I present a first version of a silicon configurable learning attractor neural network (CLANN). I describe the chip architecture and I provide a block-level description of main circuits. Results from characterization tests and from a classification experiment are also reported. Circuits details and layout choices are discussed in the apperndix.

3.1 Introduction: main ideas

CLANN stands for Configurable Learning Attractor Neural Network. It implements a network of 32 IF neurons and 32x64 configurable plastic synapses endowed with a new *stop-learning* mechanism.

Thought to be a small prototype chip to test new ideas and circuits, it has been designed to be as much flexible as possible. It allows to study and characterize in details the behavior of single circuits as well as to run network-level experiments on freely configurable architectures. Far from being competitive with PC-based simulations, at least for what concern the number of involved neurons and synapses, it represents a necessary step toward the development of stand alone neural network devices.

According to [Mead, 1989] we will use the term *neuromorphic* to refer to a class of aVLSI chips that mimics the biological nervous system in its organization and behavior. These devices typically have massive parallel architectures composed of highly interconnected analog computational elements, neurons and synapses; they are completely asynchronous and they exploits MOSFETs operating in subthreshold regime. These features make them compact, low power devices.

CLANN is a semi-neuromorphic chip: the core (neurons and synapses) is analog and asynchronous while some peripheral and "service" structures are based on standard digital cells which represented the fastest way to put new ideas into circuits. The strategy consists in testing new features using available and reliable pieces of hardware, leaving for a second development step the custom optimization of the circuits. The reliability and testability aspects rather than the Silicon occupancy and the power consumption were privileged.

The chip results a mixed signal analog-digital cross-talk free, device. Its Silicon occupancy is 5.4 by 2.6 mm. It has 143 input/output pads and has been designed in CMOS 0.35μm technology.

Here the main ideas implemented in CLANN.

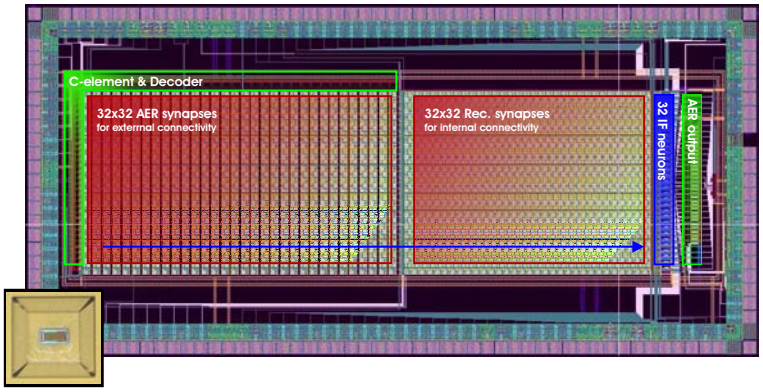


Figure 3.1: CLANN (Configurable Learning Attractor Neural Network) top layout view. In blue the array of 32 integrate-and-fire neurons is highlighted. The synaptic matrix is in red: it is composed of two parts, one of 32x32 synapses for internal recursive connections and one of 32x32 synapses accepting spikes from external devices, the so called AER (*Address Event Representation*) synapses the name of the protocol adopted. To accept in input external events a control logic and a decoder, left side in green, are required. The AER output system, in green on the right, takes care of sending spikes to external devices.

The synaptic model. An innovative feature of CLANN is the new model of synaptic plasticity implemented: the *stop-learning* synapse described in equations 2.29-2.30. The synapse stops to "learn" when the post-synaptic neuron firing rate enters in a desired range. A measure of the neuron activity is given by a Calcium variable whose dynamics approximates that described in eq. 2.28. Its generation and comparison with a set of thresholds as prescribed by inequations 2.29 required new circuits that have been simulated and designed. Thanks to standard digital circuits, the Calcium dynamics can be turned on or off at will and the performances of the network with or without the *stop-learning* mechanism have been compared.

Local and external connectivity On Silicon, one of the major issues of these massive parallel devices is connecting neurons together. The brain solves this problem in the easiest possible way: it spreads in three dimensions and it dedicates one communication channel, the axon, to each neuron. On Silicon this would correspond to designing one wire for each cell, which is the solution we adopted to create a local on-chip recurrent network. Besides a recurrent configurable internal network, CLANN is also endowed with structures to send and accept external connections. Unfortunately, the limited number of available input/output pins do not permit to exploit the "one-wire-per-neuron" strategy also for external connections (for more than 128 units). To bypass this constraint a solution is to multiplex outgoing spikes onto a single communication bus. Traffic load and channel capacity of this external bus become important characteristic to consider for designing large and distributed networks. The choice of using Silicon space for internal connections is motivated by the attempt to reduce the traffic on the external bus.

Configurability Each synaptic contact can be enabled or disabled and its excitatory or inhibitory nature decided. The reconfigurability of the synaptic matrix is an innovative feature of CLANN. This is valid for all synapses regardless they are devoted to internal or external connectivity. The network architecture can be reconfigured at will during an initial

setup phase loading a serial digital bitstream onto an on-chip shift-register: up to an all-to-all internal recurrent connectivity is supported as well as pure feed-forward architectures.

Among the 64 synapses of each dendritic tree, 32 are devoted to on-chip local connections and 32 to receive stimuli from external devices. By connecting multiple CLANN chips together it is possible to create different kinds of larger networks: for instance 4 chips can make a network of 128 neurons with uniform connectivity at 50% (i.e. each neuron is connected to any other neuron in the network with a probability of the 50%), 8 chips a network of 256 neurons uniformly connected at 25%. To make chips communicate among each other it is not sufficient to connect the output of a sender chip to the input of the receiver one. An external device working as a mapper is necessary. The mapper has to establish the external connectivity routing the events it receives from the sender chip to the right target synapses on the receiver chip.

AER compliant Connectivity from and towards external devices is performed through the Address Event Representation (AER) protocol originally described in [Mahowald, 1992] and later studied in [Boahen, 1998] [Boahen, 1999a] [Boahen, 1999b] [Culurciello and Andreou, 2003] [Dante et al., 2005]. This protocol, completely asynchronous, is now a standard for this kind of neuromorphic chips and has demonstrated to be a good way to convey information from one chip to another. The original idea is to multiplex a large number of axons on a single, digital, AER channel: when a spike is emitted, a digital address identifying the sending neuron is written on the bus. Being completely asynchronous the timing information is implicit and each address on the bus represent an event for the network (AER).

If, on one side, the AER fits well the needs of an asynchronous network [Boahen, 1999a], on the other side designers have to face problems related to asynchronous communication. One of these problems is the data conflicts: neurons trying to access the bus at the same time generate data conflicts on the AER bus. The typical neuromorphic choice to manage the conflicts is to introduce an arbiter in charge of deciding which neuron can access the bus and which neuron has to wait. This solution is optimal if the queues and the related delays are small enough not to affect the network dynamics. In other words, the arbitration is a good strategy until the traffic load is sufficiently smaller than the channel capacity [Boahen, 1999a].

CLANN is endowed with four AER structures: on the input side a C-element accepts incoming AER spikes and a decoder routes them to the correct AER synapse, on the output side an arbiter manage the bus accesses and an encoder write on the bus the correct digital address. All these circuits were provided by the Institute of NeuroInformatics in Zurich.

On the output side of the chip the internal network and external bus activities have been decoupled interposing a memory element (a FIFO with a depth of 1 bit) between neurons and arbiter. This ensures that any delays experienced on the communication pipeline due to traffic overloads or data transfer failures do not influence the internal neurons behavior. The price to pay is a possible event loss in case of AER overload.

MUX Designed to be a test chip CLANN is endowed with a digital multiplexer to sample the neurons output just before the spikes enter the memory element used as FIFO. This allows monitoring the neurons activity without turning on the AER systems which is useful during the setup debug phase. It also lets us record and compare the neurons activity before and after the AER output circuits so that we can evaluate the communication system reliability and performances under different conditions.

Designed and simulated with CADENCE In Designing the chip, Monte Carlo simulations have been performed to estimate the effects of mismatch. Due to mismatch analog

circuits designed to be identical have, on chip, different behaviors; the characteristic of two nominally identical MOSFETs can actually differ for more than the 50%: mismatch has to be considered designing the chip. It derives from Silicon inhomogeneities and imperfections introduced during the manufactory stages. Choosing particular geometries, adequate orientations and suitable dimensions for MOSFETs and capacitors it is possible to reduce mismatches among circuits; CADENCE Monte Carlo simulations is a helpful tool to evaluate designers choices.

3.2 Architecture

In Fig. 3.1 the top level layout of the chip is reported. This paragraph wants to be a guided tour around the chip layout.

This chip has 32 IF neurons arranged in a column, highlighted in blue, on the right side of Fig. 3.1. The dendritic tree of each neuron spreads horizontally, parallel to the blue arrow. Each tree comprises 32 AER synapses, accepting external spikes, and 32 recurrent synapses, for on-chip local connectivity. The Silicon occupancy of the complete synaptic matrix is evidenced in red. The matrix is divided in two parts: on the right side there are 32x32 synapses for recurrent connectivity; on the left side 32x32 AER synapses for external connectivity. The AER part is larger than the recursive one because next to each AER synapse there is a pulse shaper circuit which gives to impinging AER spikes a suitable form for successive processing. Synapses and neurons represent the core of the chip, the network. All the other circuits exploit services necessary for the core to work and for communication.

The connectivity with external devices is ensured by the AER input and output systems (in green). The AER input system, on the left side, consists of a control element and a decoder. The former handles the asynchronous handshake with external devices required to receive the AER data, the latter is in charge of routing the incoming spikes to the target synapse. The AER output system, composed of a memory element, an arbiter and an encoder, receives spikes emitted by neurons and manages the access to the external AER output bus.

On the perimeter of the chip the padframe is visible: it is composed of 143 pads for analog or digital signals, providing electrical connections with the external world.

The big amount of silicon space covered by the synaptic matrix is clearly visible in Fig. 3.1. It occupies more than the 80% of the Silicon surface. Given a Silicon area, the occupancy of a single synapse essentially imposes the dimensions of the network. Projecting reliable and compact synapses is one of the designers main goal.

3.3 Signal flow

This paragraph gives a more detailed description of the signal flow through the various blocks, from the arrival of an AER event to the emission of an outgoing spike (see Fig. 3.2).

A spike from an external device comes under the form of an AER event: a digital 10 bits address identifying one of the 1024 AER synapses of CLANN. Communication on the AER asynchronous bus uses a four-phase handshake between the sender and the receiver device [Dante et al., 2005]. The C-element [M. Shams and Elmasry, 1998] handles this phase and when data on the bus are valid it lets the decoder read the 10 bits address. The decoder, consisting of a horizontal and a vertical arm, stimulates the target synapse selecting a row

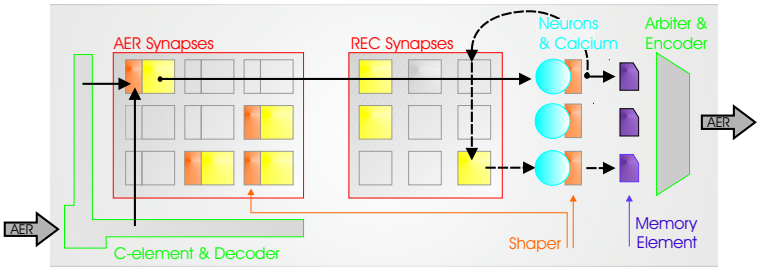


Figure 3.2: CLANN main signal flow. A spike from an external device, under the form of an AER event, is accepted by the decoder which in turn stimulates the AER shaper (in orange) next to the target synapse (yellow square). The shaper circuit extends the incoming spike pulse to make it useful for the synaptic dynamics. Active synapses (those colored in the figure) according to their nature, excite or inhibit the corresponding neurons (blue circles) on the same horizontal line. Synapses on the same horizontal line belong to the same dendritic tree, whatever they are AER or recursive synapses. A spike emitted by a neuron is extended by a shaper circuit and sent to 1) all the recursive synapses on a given column, 2) to the AER output system via a memory element (in violet) and 3) to a digital standard multiplexer (not shown) whose outputs can be directly probed.

and a column of the synaptic matrix using two identical digital pulses (X_{AER} and Y_{AER} signals). These pulses remain active till the end of the handshake. Hence their duration depends not only on CLANN circuits, but also on the characteristics of external devices as well as on the electrical implementation of the bus itself. In typical conditions, in our experimental setup, the duration is about 200ns.

At each synaptic site an AND gate combines the X_{AER} and Y_{AER} pulses. The AND output signal is sent to the AER synapse through a pulse shaper circuit which extends the duration of the pulse. The duration of this pulse is tunable via an analog bias supplied from outside the chip. The shaper output is connected to the synapse that needs "long" signal to work properly (see section 3.4). In typical conditions the pulse duration is tuned to about 10 μs.

The function of the shaper is 1) to equalize the duration of incoming spikes, 2) to define a time window suitable for the short-term dynamics of the synapse and 3) to decouple the AER handshake duration (about 200ns) from the length (about 10μs) of internal spike pulses accepted by synapses. Decoder output signals act just as triggers for shaper circuits.

We chose to design one shaper next to each AER synapse: 1024 shapers in total, occupying a non-negligible percentage of Silicon area. The reason behind this choice is to maintain the handshake phase as fast as possible, and hence to achieve high AER channel capacity. Let me try to clarify this point. The option of having just one shaper for all the AER synapses has been rejected because such a shaper would have represented a slow stage in a fast serial chain: an incoming AER spike would have not been processed until the previous one have been completely served. Considering 10μs per spike, the upper bound for the channel capacity reduces to 100KHz. (This can be a too strict constrain if all the 1024 AER synapse are active: imaging a Poisson activity around 20Hz for all the 1024 connected neurons, according to [Boahen, 1999a] the collision probability would result more than the

30%). On the other side, with one shaper for each AER synapse, there would be a queue (or an event loss) only if the inter spike interval between two spikes impinging on the *same* AER synapse is less than 10μs. (For a comparison, in the same situation described above, the collision probability reduces to less than 0.8%).

AER synapses implemented in CLANN are plastic bistable systems which evolve according to equations 2.29. Their short-term dynamics is spike-driven and hence is affected by the arrival of AER spikes. To update the value of the synaptic internal variable $X(t)$ (see eq. 2.29), the synaptic circuit needs to know both the state of the post-synaptic neuron potential and the state of the corresponding Calcium variable. These information are coded in two digital bits that back-propagate from the post-synaptic neuron to the synapses (both recursive and AER) belonging to the same dendritic tree. While updating $X(t)$, during the time window set by the shaper, the synapse also excites (or inhibits) the post-synaptic neuron. This is achieved through an injection (or a subtraction) of current into (or from) the neuron circuit. This current travels on an analog wire on which also the contributions from the other synapses converge. The various current contributions simply sum together according to Kirchoff law thus nor data conflicts, nor channel capacity neither complex communication structures has to be taken into account. This is a great design simplification possible in the analog domain. The number of synapses connected to the same dendritic tree is limited only by considerations on signal to noise ratio. In short, the synapse receives digital signals and produce analog output currents. A hybrid digital/analog circuit is necessary.

The IF neuron executes the opposite conversion: it receives analog signals and generates digital spikes. Its internal dynamics is driven by the synaptic analog contributions and by a tunable constant afferent current supplied from outside. Next to each neuron there are circuits handling the dynamics of the Calcium variable. Their output are two digital signals used by the synapses for their internal dynamics. The spike emitted by the neuron has the form of a fast digital pulse, lasting no more than 40ns. Spikes emitted by local neurons propagates along the neuron axon and reaches every synapse belonging to a column of the recurrent matrix (dashed line in Fig. 3.3). To become useful for recurrent synapses spike pulses have to be extended to $10\mu s$ pulses, as for AER spikes. To this end other pulse shapers have been introduced (orange elements on the right side of Fig. 3.3). The differences with the AER case, are that 1) the spike emitter is on-chip and 2) that each neuron has its own axon. The spike extension is than performed at the output of the unique sender, the neuron, instead that at the input of the various receivers, the synapses. Conflicts or queuing issues as well as the design of communication structures are here bypassed thanks to the "redundancy" of the communication channels: each neuron has its own axon.

Each column of the recurrent synaptic matrix is composed of 32 synapses, each one is part of the dendritic tree of a different neuron. According to the loaded configuration some of the synaptic contacts will be active (yellow synapses in Fig. 3.3) establishing in this way an internal connection between two local neurons. Both the shaper and the synaptic circuits remain the same in the recurrent and AER part of the chip. This ensures an homogeneous behavior throughout the entire network.

Other chip designers prefer not to provide their chips with internal connections [Mitra et al., 2006] [Indiveri et al., 2006], saving in this way Silicon space. The local connectivity is obtained connecting the chip output with its input. This solution has as advantage a significant reduction of silicon occupancy and as the major drawback an increase in the load on the external bus. The external communication bus, in its current form, is one of the main obstacles to large networks. The traffic load and the dimensions of the necessary address

space grow with the square of the number of neurons and soon exceed the channel capacity. Different approach to this problem can be chosen: one can improves the protocol, increases the bus capacity, thinks to fast serial link or start moving the load elsewhere.

I would like to underline that this problem, the communication problem, is one of the main issue limiting the growing in dimensions of the networks in HW. Diffent kind of solutions have been investigated, from an improvement in the AER protocol [Boahen, 2004a] [Boahen, 2004b] [Boahen, 2004c] [Merolla et al., 2006] to the usage of faster communication channel as for instance the serial ATA protocol. The asynchronous AER bus has a bandwidth of 10 MHz in its present stable implementation [Dante et al., 2005] [Chicca et al., 2007]. This channel capacity reduced to something about 500 KHz when one wants to monitor and map the AER event in real time. The brain choice is different, in spite of multiplying the channel capacity it multiplies the channels. This solution has as the "HW" advantage that the malfunctioning of one channel does not compromises the entire network: the redundancy increases the reliability of the system. Another advantage is the ability of transmitting a large amount of simultaneous spikes, task that a serial channel suffers to complete without introducing relevant delays.

The neuron output is connected not only to the shaper but also to a memory element which represent the first stage of the AER output system. The memory element is a column of 32 D-type flip-flops, one for each neuron, between the neurons and the arbiter. A flip-flop is set when the corresponding neuron emits a spike. The output of an active flip-flop represents a request to access the AER bus. The bus accesses are completely asynchronous and two neurons could make their requests simultaneously. The arbiter solves these conflicts according to the logic "the first wins": it chooses the winner and sends it an acknowledge signal that reset the flip-flop. Other spikes wait their turn to be served. Hence, in case of conflicts, the formation of a queue is possible and the information that a spike have been emitted has to be retained somewhere. We chose to maintain this information in the memory element which essentially is, for each neuron, a FIFO with a depth of 1 bit. In this way the neuron can be reset as soon as the flip-flop is set, regardless the state of the AER bus. Once the spike is transmitted the flip-flop is reset. This solution completely decouples the internal activity from the external AER ones. Delays or blackouts on the AER channels do not influence the network dynamics. As already said, the price to pay is that a spike is lost if the spike previously emitted by the same neuron has not been served yet. The acknowledge given by the arbiter to a chosen neuron, represent a "go" signal for the encoder that put a 5 bits code on the AER output bus. This code identifies the emitting neuron.

The AER output bus is physically different from the AER input bus. The output bus carries a 5 bits information identifying the emitter, the input bus a 10 bits address identifying the receiver. To connect the output of one CLANN chip to the input of another CLANN chip an external mapper is needed. It has to retain the information on the connectivity and to map the incoming neuron addresses onto the corresponding target synapse addresses. The channel bandwidth and the available address space are critical issues for the input bus where both the number of synapses to identify and the traffic load scale with the square of the number of neurons in the network. On the output bus these quantities scale linearly with the number of neurons.

A digital multiplexer (MUX) 32to2 has been implemented to monitor the output of the neurons during the setup debug phase. The MUX receives extended spikes and routes them to dedicated test pins. Other test pins give direct access to 2 neurons and 2 synapses and allow to continuously monitor the circuits behavior.

The signal flow can be summarized in this way: an AER spike is routed to the target synapse by the decoder. The spike passes through a pulse shaper before arriving on the synapse. The synapse translates the extended pulse in a current that affects the neuron dynamics. When the neuron emits a spike it is sent, via pulse shaper, to recursive synapses and, in parallel outside the chip via a memory element, an arbiter and an encoder.

In addition an extra signal path is present on the chip: it allows to load the synaptic configuration in order to set individually each synaptic contact as an active/inactive excitatory/inhibitory one. During an initial phase a serial bitstream is fed into a shift register composed of 4096 flip-flops: each synapse hosts two of the 4096 D-type flip-flops. By connecting the two flip-flops within each synapse in series and then connecting them in series with the flip-flops belonging to contiguous synapses, it is possible to create a unique shift register that, as a snake, extends over the entire chip touching all the synapses. The two flip-flops per synapse store two configuration bits that decide the activation/inactivation of the contact and its inhibitory or excitatory nature. The serial method was chosen because only two pins are necessary: one for the data, one for the clock. This is the only part of the chip that necessitates a clock. It is designed with standard AMS (Austria Micro Systems) cells: D-type flip-flops and clock buffers. Once the configuration is loaded the clock is stopped, the logical gates stop switching and the neural activity can start.

The signal paths described above includes digital and analog circuits. Spikes from external devices come into the form of digital 10 bits addresses, pass the AER input system and are then converted into packets of analog charges by the synapses. The neurons execute the opposite conversion: they receive analog contributions and emit spikes under the form of digital pulses. Fast switching digital signals and slow analog ones cohabits in the chip. To realize on silicon such a pathway a number of problems related to noise has been considered. In such kind of mixed digital-analog chip, a typical problem is the cross-talk effect between fast-switching digital wire and slow analog wire. The effect is induced by parasitic capacitances coupling the wires together. What happens is that when a digital circuit switches it creates a step signal on its output net. The fast components of this signal pass the parasitic capacitors and generate a peak in the voltage of the coupled line. If this second line drives digital circuits usually the noise induced is negligible compared to the large difference between the analog voltages coding for a digital 1 (3.3V) or a 0 (0V). On the other hand, if the influenced line is carrying a precise analog voltage level, the noise can become a problem. A typical situation where this could happen in CLANN is between the axons and the dendritic trees. The axons carry a fast digital pulse representing the spike, while each dendritic tree has an analog net on which the synaptic contributions sums together. As shown in Fig. 3.13 each axon crosses all the dendritic trees: in the worst, but possible, scenario the cross-talk could become important compared to the synaptic contributions, and the network dynamics completely corrupted.

Cross-talk, circuit coupling through the bulk, power bounces can cause the complete chip malfunctioning. To avoid or at least reduce this kind of effects circuits placement, signal and power routing, constrains on MOSFET geometries, separation of analog and digital power, distances between analog and digital blocks and noise barriers should be considered planning the architecture of the chip.

Building a mixed signal chip of this kind, is not just connecting pieces together. A top-down design strategy was adopted. The architecture of the chip is important as much as the single circuits. A rough version of the top level layout of CLANN has been designed before the implementation of the various blocks. Clearly, building the chip, the original idea

evolved to fit the needs of single circuits. This approach let us design a system composed of parts that, in the end, work nicely together.

3.4 Neuron and Synapse, block level

The core part of the signal path, described in the previous section, involves neurons and synapses. A block level description of these circuits is given in the next paragraphs, together with the description of the Calcium circuit. Figure 3.3 reports the a diagram of the circuits.

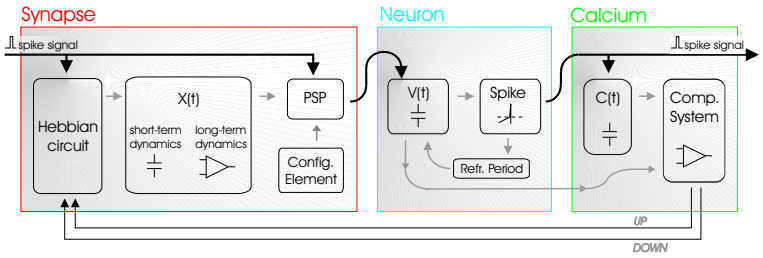


Figure 3.3: Synapse neuron and calcium circuits, block level diagram. The incoming spike, is received by the Hebbian and by the Post-Synaptic Potential (PSP) blocks. The Hebbian block is the entry point for the synaptic dynamics. It collects information from the Calcium circuit and when triggered by the incoming spike it injects (subtracts) a current into (from) the $X(t)$ block where the internal analog synaptic variable $X(t)$ is updated and maintained. The output of the $X(t)$ block is a digital flag accounting for the synaptic state: *potentiated* or *depressed*. The PSP block generates the synaptic current contribution for the postsynaptic neuron according to the synaptic state and to two bits stored in the configuration element. Each synapse can be configured as excitatory, inhibitory or inactive. The synaptic output current charges or discharges the neural membrane capacitance included in block tagged $V(t)$ in the neuron circuit. When the voltage $V(t)$ across the membrane capacitance reaches a threshold the Spike block generates an action potential and activates a feedback loop that brings $V(t)$ back to a reset potential as imposed by the theoretical model (see eq. 2.20). Integration of incoming synaptic contributions restarts only after a refractory period. The Calcium circuit reads the emitted spikes and manages the Calcium variable ($C(t)$ block). The system of comparators generates two signals to be back-propagated to synapses for implementing the learning rule of eq. 2.29.

The synapse On the the left the synapse is divided into four blocks: the Hebbian circuit elaborates the input for the X module, the core of the synapse. According to the internal state of the X module, and to the Configuration Element, the PSP block generates the the correct stimuli for the postsynaptic neuron.

The internal synaptic variable X is a voltage across a capacitor placed in the X module. As described in the previous chapter, the variable X is made eligible for an upward or a downward jump whenever a pre-synaptic spike arrives. The Hebbian circuit receives the presynaptic spike and two digital signals, UP and $DOWN$, coding the output of inequalities 2.29. If UP is active, the Hebbian circuit injects a current into the capacitor inducing an upward jump; if $DOWN$ is active a subtraction of current from the capacitor generates

a downward jump. If both digital signals are disabled, no jumps are triggered. UP and $DOWN$ signals back-propagate from the Calcium circuit connected to the postsynaptic neuron to all the synapses belonging to the dendritic tree of the neuron.

The jumps, triggered by spikes, represent the short-term dynamics (on the timescale of few microseconds) of the internal synaptic variable. In the absence of impinging spikes X is forced towards its upper or lower bound according to where the last jump left it, respectively above or below a given threshold θ_X. This refresh mechanism tends to maintain an "high" or a "low" value for X. On time scale of the order of hundreds of milliseconds, the slow internal refresh and the fast spike-induced jumps, give X the tendency to increase or decrease its value according to the statistical properties of the neural spiking activities. This is the long-term dynamics of the synapse which can comports a change in the synaptic state: the synapse is potentiated if $X(t) > \theta_X$, depressed if $X(t) < \theta_X$. The output of the X module is a digital flag that encodes the synaptic state.

PSP (post-synaptic potential) block stimulates the postsynaptic neuron either injecting or subtracting current during the time-window set by the impinging spike. These currents induce upward or downward jumps in the postsynaptic neuron potential. In this block, digital signals control the output analog currents. In input there are four digital signals: the flag from the X module, two bits from the Configuration Element and the presynaptic spike pulse. In output, instead, the PSP block communicates over an analog channel which is a node shared by all the synapses belonging to the same dendritic tree directly connected to the capacitor representing the postsynaptic neuron soma. The effect on the post-synaptic neuron potential, depends on how the synapse has been configured, excitatory, inhibitory or disconnected. If excitatory the PSP block injects current into the output node inducing an upward jump; the amplitude of the current is determined by the digital flag coding the synaptic state: the current is low (small jump) if the synapse is depressed, higher (bigger jump) if the synapse is potentiated. If inhibitory, the current is sucked from the analog node (downward jump) and its amplitude is fixed independently from the value of the digital flag. If disconnected no current is produced. The Configuration Element retains, in two flip-flops, memory of the two configuration bits loaded during an initial phase. Each bit enables the excitatory or the inhibitory part of the PSP block.

The spike duration is determined by the shaper, and typically is set to 10μs. In this interval of time, both the jumps of the synaptic variable X and those on the postsynaptic neuron potential $V(t)$ happen. The choice of few microseconds for the length of this time window derives from the constraints that have to be satisfied. 1) From a theoretical point of view, according to the models described in the previous chapter, the jumps on X and on $V(t)$ should take place in a negligible amount of time. For what concerns the synapse, the term of comparison is the long-term dynamics timescale: few hundreds of milliseconds. For the post-synaptic neuron potential the 10μs should be compared to the minimum expected inter-spike interval which is of the order of few milliseconds. 2) From an hardware point of view various factors concur to determine the spike length. The inferior limit is given by the smaller time window that allows a fine tuning of the jumps amplitude. This is determined by the amount of charge transferred to the capacitors, and hence by the product of the spike length with the amplitude of the injected currents. Moreover to reduce noise generation, the time interval should not be reduced too much so that the current steps produced on the analog lines can be kept sufficiently small. On the other side, a synapse should be able to process a spike before receiving the next one. And hence the maximum spike length is fixed by the presynaptic neuron activity. The choice of working in the range of few microseconds

satisfy all this constraints.

Neuron Figure 3.3 reports a simplified block diagram for the neuron circuit. The $V(t)$ module receives and integrates the PSP block contributions onto a capacitor. The voltage across the capacitor represents the neuron membrane potential $V(t)$. In addition to the PSPs, the capacitor is charged by a constant afferent current and discharged by a constant leakage current as dictated by the model described in the previous chapter. The voltage $V(t)$ evolves according to these stimuli and when it reaches a given threshold θ_V a spike is emitted. In VLSI this means that an inverter, in the spike block, changes its digital output. This activates a feedback loop that resets to zero the membrane potential causing the inverter to switch again. In this way a digital pulse, the spike, is generated at the output of the inverter. The refractory block, in the middle of the feedback loop, shortcut $V(t)$ to zero for a certain time interval, thus implementing a refractory period (τ_{arp}) during which incoming PSPs and afferent current do not affect the capacitor. At the end of the refractory period, $V(t)$ is released and the integration starts again. The circuit used in CLANN is the one described in [Indiveri et al., 2006] which is endowed also with frequency adaptation capability thanks to another negative feedback loop, not shown in the figure.

The circuit, as detailed in the next section, is a real neuromorphic one. It is an hybrid analog/digital, compact and low power circuit. This is the result of 17 years of evolution of VLSI IF neurons. For other parts of the chip there has not been such evolution yet. And we prefer to adopt standard circuit to test the ideas, leaving to future steps the improvement of the design.

Calcium A variable that measures the recent neuron activity is necessary for the stop learning mechanism. This information is used to stop a further synaptic potentiation or depression if the post-synaptic neuron is already too active or too inactive. This information is stored in an analog variable we will refer to as the Calcium variable: a voltage across a capacitor (clearly). Every time a spike is emitted the variable is suddenly increased of a certain amount; in absence of spikes the capacitor discharge linearly; thus, If the mean firing rate of the neuron is above a certain threshold, the calcium variable tends to increase, otherwise it tends to decrease. The Calcium block of figure 3.3 contains the circuits for the generation of this variable and the comparators block which reads the Calcium variable $C(t)$ and the neuron potential $V(t)$ and executes the comparisons described in inequalities 2.29. Its output are the UP and $DOWN$ digital signals that are broadcast to all the synapses belonging to the dendritic tree. The comparators system can be configured, thanks to digital standard cells, and it is possible to exclude the calcium dynamics from the comparison process. This allow to implement the basic learning rule described by equations 2.26.

Figure 3.4 illustrates a typical behavior of the main analog variables involved in the network dynamics. The traces, from top to bottom, represent the postsynaptic neuron potential $V_{post}(t)$, the internal synaptic variable $X(t)$, the Calcium variable $C(t)$ and the presynaptic neuron potential $V_{pre}(t)$. An increase in the afferent current forces the postsynaptic neuron to fire at increasing frequency. Consequently the Calcium variable raises and exits the range set by the K thresholds. This activates the *stop-learning* mechanism and $X(t)$ stops undergoing upward or downward jumps and is than attracted towards ground by the refresh circuit.

Previous paragraphs provide a description of CLANN main parts and explain how these blocks communicate each other. Further details on circuits are furnished in the appendix of this chapter where the schematics and layout are reported and technical choices concerning mismatch and noise reduction are discussed.

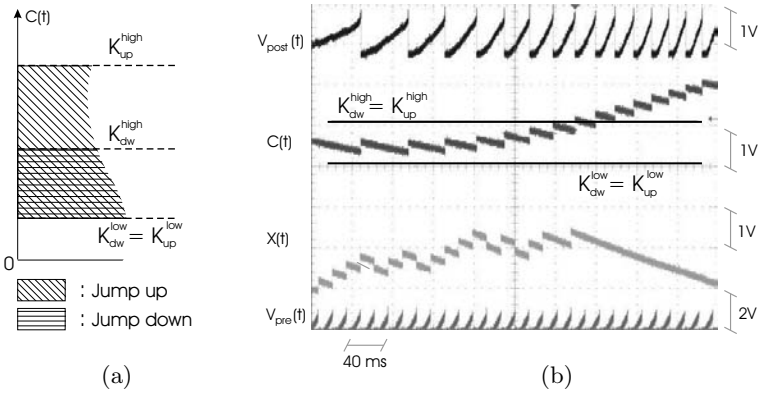


Figure 3.4: On the left (a) a typical scheme for Calcium thresholds ($K_{up}^{low} = K_{dw}^{low}$ and $K_{up}^{high} > K_{dw}^{high}$): according to 2.29 downward jumps are allowed inside the region shaded with horizontal lines, upward jumps inside the region shaded with oblique lines. On the right (b) an illustrative example of the *stop-learning* mechanism ; top to bottom: post-synaptic neuron potential V_{post}, calcium variable C, internal synaptic variable X, presynaptic neuron potential V_{pre}.

We performed extensive experiments to characterize single circuits behavior and collective neurons dynamics. An *ad-hoc* experimental setup has been build to control input signals and to monitor CLANN outputs. In input, 69 parameters, under the form of analog voltages, should be tuned; an AER external infrastructure is necessary to stimulate the network dynamics; 4 digital signals are required to configure the synaptic matrix, to enable the memory element and to select the learning rule; and, at last, a 3-bit address is needed to control the digital MUX. On the output side, the network activity can be monitored reading the AER events produced by the chip; 13 test points give information on the dynamics of two synapses and two neurons (see, for instance, traces in figure 3.4); and the MUX makes available on two output pins the digital spike pulses produced by two selected neurons. The experimental setup deals with all these signals: a PCB hosts DAC and trimmers to set the biases and a micro-controller that takes care of the digital configuration signals. The PCI-AER board [Dante et al., 2005] [Chicca et al., 2007] manages and monitor the AER traffic both in input and output. Scope probes are, from time to time, connected to the interesting test points. Another aspect that has become a necessity during the test sessions is the control of the chip temperature. The entire setup absorbs less than 150 mA, and CLANN contributes only to a small extent to this power consumption. Turning on CLANN and bringing its internal activity to full regimes, cause an increase inferior to 1 degree in the external temperature of the chip case. Problems related to temperature changes are not caused by the chip power consumption but depend on the room conditions. Considering the form of the sub-threshold mosfet characteristic [Mead, 1989] it has been necessary to build a system to fix the chip temperature. A simple but efficient approach has been adopted: a peltier cell, a heat sink and a thermostat ensure a temperature variation inferior to 1 degree, sufficient for our aims.

The lab technicians handled the implementation of the setup, I took part only to the definition of the setup specifics and to the system debug phase.

The most interesting experiment run on CLANN is a classification task, in which the chip, configured to have a perceptron-like architecture, learns to discriminate different overlapping patterns. The test has been carried out both with active and inactive *stop-learning* mechanism; the results are described in what follows. To reach this goal, a non negligible amount of time has been spent performing low-level tests, both on single circuits and on the final architecture. Their aims were to characterize the circuits behavior, to measure the effective values of parameters important for the network dynamics, to evaluate the mismatch, to figure out if the high-level behavior agrees with the predictions of the theoretical models despite the noise and the spurious effects unavoidable in a real VLSI device. For this a chip-oriented simulation has been set up and the results compared with the experimental measures. The next paragraphs shortly retrace the way from the low-level tests to the classification task.

3.5 Measuring parameters through neural and synaptic dynamics

Parameters setting is a non-trivial stage of setting up chip experiments, since mismatches and other sources of variability induce wide distributions of relevant neural and synaptic parameters. As most of them are not directly accessible, suitable stimulation protocols were devised, in order to infer the on-chip parameter distributions corresponding to given external settings.

As an example we describe here the protocol we use to measure the amplitude of J_{dw} and the value of the refresh (α) toward the upper bound (H_s) of the synaptic internal variable $X(t)$. Using the same method we measure J_{up} and the refresh (β) toward the lower bound (H_i). The *stop-learning* mechanism is switched off for this measurement.

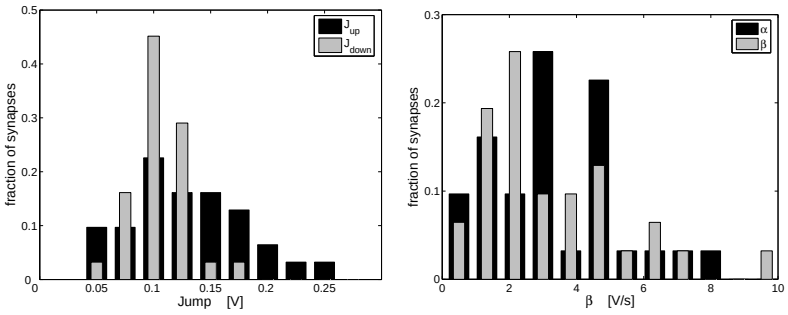
The protocol consist of stimulating the AER synapses with external spike trains and analyzing the chip response. We initialize $X(t)$ to its upper bound H_s: to do this we set the threshold θ_V on the post-synaptic neuron potential to 0, we choose a very high J_{up} and we send a pre-synaptic AER spike which certainly brings $X(t)$ above θ_X, such that it relaxes to H_s. We then set the threshold θ_V to its upper bound, so that from then on $X(t)$ will undergo only downward jumps; the synaptic efficacy J_+ for potentiated synapses is chosen large enough to guarantee one post synaptic spike for each impinging pre-synaptic spike. For depressed synapses the efficacy is set to zero. We send an AER spike train with a constant frequency ν to the synapse. The post-synaptic neuron emits spikes till the time t^* when the synapse passes from potentiated to depressed. We calculate the number of jumps $n = t^* \nu$ necessary for a synaptic downward transition. We then repeat the measure decreasing the synaptic threshold θ_X. In these particular conditions, according to the theoretical model described before, the relationship between n and θ_X follows this behavior:

$$n(\theta_X) = \frac{H_s + \alpha/\nu}{J_{dn} - \alpha/\nu} - \theta_X \frac{1}{J_{dn} - \alpha/\nu} \tag{3.1}$$

In this description we neglect the fact that, for the real circuit, the values of the jump and of α show a moderate dependence on the value $X(t)$: they decrease if $X(t)$ is near H_s or H_i. We obtain the value of α from the difference between the slopes measured for two

different ν. To accumulate the statistics on this measure we repeat the entire procedure for different pairs of frequencies.

The measures for each synapse are affected by a relative error less than 10%. The distributions over the 31 synapses of the jumps (up and down) and of the refreshes (up and down) are reported in figure 3.5; for the sake of simplicity we report in Table 3.1 only the mean values of all the measured parameters used in the experiments and simulations described in the following Sections.



Jump Up and Down distribution α and β distribution

0

Figure 3.5: Distributions over the 31 synapses of the jumps J_{up} and J_{dw} (left panel) and of the refreshes α and β (right panel) of the synaptic internal variable $X(t)$.

We would like to stress that the above strategy for estimating relevant network parameters is consistent with the *electrophysiology on silicon* approach advocated in the neuromorphic engineering, in which one takes the attitude that estimates have to be obtained from the spiking activity the neurons on the chip as the only accessible observable, for the different sets of parameters explored. More generally, the whole subject of testing the neuromorphic chip should be viewed in this light as the setup of suited *electrophysiology* experiment such that the expected network behavior is to be assessed through the emerging dynamics, typically in noisy conditions, as opposed to the usual situation, in which testing a chip amounts essentially to checking the deterministic correspondence between the design specifications and the observed behavior. In the following Section we check the statistical properties of synaptic plasticity along these lines.

3.6 LTP/LTD probabilities: measurements *vs* chip-oriented simulation

In this section we demonstrate the *stop-learning* stochastic mechanism for synaptic plasticity by estimating long-term potentiation (LTP) and long-term depression (LTD) probabilities as functions of the post-synaptic firing rate for a subset of synapses, and we compare the results with chip-oriented simulations.

For each of 31 synapses sharing the same post-synaptic neuron, we generate a pre-synaptic Poisson spike train at 70 Hz. The post synaptic neuron is forced to fire in turn a Poisson spike train by applying the superposition of an external DC current, and a Poisson distributed train of inhibitory spikes through AER. Setting to zero both the potentiated

Table 3.1: Measured parameters are divided in three categories: Neuron, Synapse and Calcium. In the column *note* we report if the parameters are measured with a suitable *test* protocol or just with the oscilloscope (*osc*). The former are different for each synapse and we report the distribution width σ.

parameter	value	error	distribution width σ	note
Neuron				
β_v	-30.00 [V/s]	± 0.70 [V/s]	/	osc
θ_{fire}	0.90 [V]	± 0.05 [V]	/	osc
J_{Inh}	0.15 [V]	± 0.01 [V]	/	osc
J_{Exc}	0.10 [V]	± 0.04 [V]	0.03 [V]	test
H_i	0.00 [V]	± 0.05 [V]	/	osc
Synapse				
α	3.71 [V/s]	± 0.04 [V/s]	1.87 [V/s]	test
β	3.63 [V/s]	± 0.04 [V/s]	2.07 [V/s]	test
J_{up}	0.14 [V]	± 0.04 [V]	0.05 [V]	test
J_{dw}	0.12 [V]	± 0.04 [V]	0.02 [V]	test
θ_X	1.50 [V]	± 0.05 [V]	/	osc
θ_V	0.30 [V]	± 0.05 [V]	/	osc
H_s^X	0.05 [V]	± 0.05 [V]	/	osc
H_i^X	3.00 [V]	± 0.05 [V]	/	osc
Calcium				
K_{up}^{high}	2.30 [V]	± 0.05 [V]	/	osc
K_{up}^{low}	0.05 [V]	± 0.05 [V]	/	osc
K_{dw}^{high}	3.00 [V]	± 0.05 [V]	/	osc
K_{dw}^{low}	0.05 [V]	± 0.05 [V]	/	osc
C_{start}	0.40 [V]	± 0.05 [V]	/	osc
β_{Ca}	12.00 [V/s]	± 0.28 [V/s]	/	osc
J_{Ca}	0.17 [V]	± 0.05 [V]	/	osc

and depressed efficacies, the activity of the post-synaptic neuron can be easily tuned by varying the amplitude of the DC current and the frequency of the inhibitory AER train. We initialize the 31 (AER) synapses to be depressed (potentiated) with the same protocol described in the previous section, and we monitor the post-synaptic neuron activity during a stimulation trial lasting 0.5 seconds. At the end of the trial we read the synaptic state using a suitable AER protocol. For each chosen value of the post-synaptic firing rate, we evaluate the probability to find synapses in a potentiated (depressed) state after the trial, repeating the test 50 times. The results reported in figure 3.6 (solid lines) represent the average LTP and LTD probabilities per trial over the 31 synapses. Tests are performed

both with active and inactive *stop-learning* mechanism. When *stop-learning* mechanism is inactive, the LTP is monotonically increasing with the post-synaptic firing rate while, when the calcium circuit is activated the LTP probability has a max for ν_{post} around 80 Hz.

Identical tests are also run in simulation (dashed curves in figure 3.6). For the purpose of a meaningful comparison with the chip behavior, we implement in the simulation the previously estimated distribution of relevant parameters affecting neural and synaptic dynamics.

Simulated and measured data are in qualitative agreement. The parameters we choose for these text are the same as those used for the classification task described in the next section.

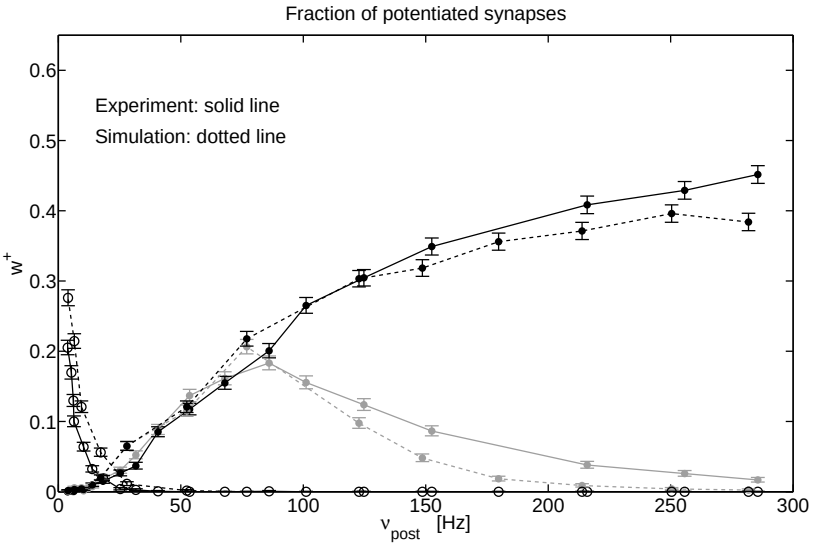


Figure 3.6: Transition probabilities. Black and gray lines with filled markers are LTP probabilities with and without calcium *stop-learning* mechanism respectively. Black lines with empty markers are LTD probabilities without *stop-learning* mechanism. The case LTD with *stop-learning* mechanism is not shown. Error bars are standard deviations over the 50 trials

We emphasize that, despite the many sources of variability and inhomogeneity in the chip, taking into account the uncertainty in a limited set of effective parameters directly mapped onto the theoretical model is enough to account for the amount of variability observed in the measurements. The agreement between measurements and simulations assures us that there are not spuriuos effects in the chip which introduce systematic deviations from the expected behaviour (e.g. correlated fluctuations in some of the parameters).

3.7 Learning overlapping patterns

We configure the synaptic matrix in order to have a subnetwork with a perceptron like architecture with 1 output and 32 inputs (32 AER synapses). 31 synapses are set as plastic excitatory ones, the 32nd is set as inhibitory and used to modulate the post-synpatic neuron

activity.

Our aim is to teach the perceptron to classify two patterns *Up* and *Down* through a semi-supervised learning strategy to be explained below. After learning we want the perceptron to respond with high output frequency for pattern *Up* and with low output frequency for pattern *Down*. The self regulating *stop-learning* mechanism is exploited to improve performances when *Up* and *Down* patterns have a significant overlap. Learning is semi-supervised: for each pattern a "teacher" input is sent to the output neuron, steering its activity to be high or low, as desired. At the end of the learning period the "teacher" is turned off and the perceptron output is driven only by the input stimuli: in this conditions its classification ability is tested.

Analogous experiments on a similar device are described in [Mitra et al., 2007].

We present learning performances for input patterns with increasing overlap, and demonstrate the effect of the stop learning mechanism (overlap ranging from 6 to 14). Together with the overlap the coding level (i.e. the fraction of perceptron inputs affected by the stimulus) also increases (from 0.5 to 0.7). This allows to use all the 31 AER synapses in all the experiments.

Upon stimulation, active pre-synaptic inputs are Poisson distributed spike trains at 70 Hz, while inactive inputs are Poisson spike trains at 20 Hz. Each trial lasts half a second. *Up* and *Down* patterns are randomly interleaved with equal probability. The teaching signal, a combination of an excitatory constant current and of an inhibitory AER spike train, forces the output firing rate either to 50 or to 0.5 Hz. One run includes 150 trials, which is sufficient to stabilize the output frequencies. At the end of each trial we turn off the teaching signal, freeze the synaptic dynamics by setting the refresh to a high value and read the state of each synapse using a suitable AER protocol. In these conditions we perform a 5 seconds test ("Checking Phase") to measure the perceptron frequencies when pattern *Up* or pattern *Down* are presented. Each experiment includes 50 runs. For each run we change: a) the "definition" of patterns *Up* and *Down*: inputs activated by pattern *Up* and *Down* are chosen randomly at the beginning of each run; b) the initial synaptic state, with the constraint that only about 30 % of the synapses are potentiated; c) the stimulation sequence. Results are described in figures 3.7 to 3.9.

We carry out learning experiments with different overlaps between the two patterns to be learnt (ranging from 0 to 10), comparing the performance when the *stop-learning* mechanism is inactive/active. In the last case only the threshold K_{up}^{high} is active (the threshold above which jumps up are inhibited). The Calcium circuit parameters are such that the Ca variable passes K_{up}^{high} when the mean firing rate of the post-synaptic neuron is around 80 Hz. For orthogonal stimuli (zero overlap) the perceptron was able to correctly learn the stimuli, and activating the *stop-learning* mechanism does not imply a qualitative difference in performances (left column in figure 3.9).

We then studied the case of patterns with fixed overlap (10) with active and inactive *stop-learning* mechanism.

In figure 3.7 we report the distributions of perceptron frequencies over 50 runs at four different stages along the run for overlap 10 with inactive (upper panels) and active (lower panels) *stop-learning* mechanism. In black the distributions corresponding to the perceptron output frequencies when the Pattern *Down* is active, in grey the distributions corresponding to pattern *Up* active. The frequencies have been monitored during the checking phases. At the beginning of learning the grey and black distributions are completely overlapped, regardless the *stop-learning* mechanism. The separation of the two distributions increases

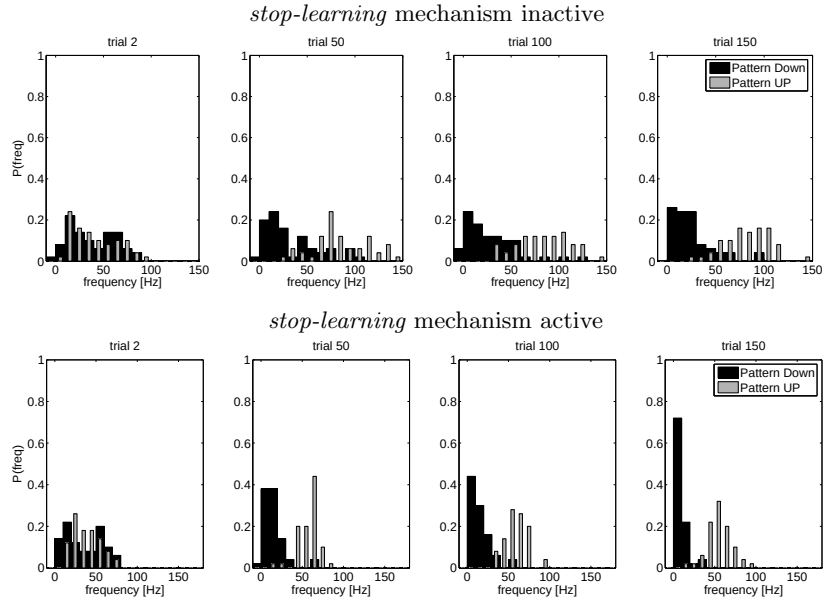


Figure 3.7: Distribution of perceptron output frequencies at four different stages of the learning process. The number of inputs belonging to both patterns is 10.

towards the end of the run. The improvement due to the *stop-learning* mechanism is clearly visible in the much better separation between the frequency distributions.

We show in figure 3.8 the distributions of the fraction of potentiated synapses over the 50 runs during the same experiment of figure 3.7. We divide synapses in three subgroups: *Up* (grey) synapses with pre-synaptic input activated solely by the *Up* pattern, *Down* (black) synapses with pre-synaptic inputs activated only by the *Down* pattern, and *overlap* (white) synapses with pre-synaptic inputs activated by both patterns *Up* and *Down*. The state of the synapses is probed and recorded after every learning step. Accumulating statistics over the 50 runs we obtain the distributions reported in figure 3.8. The fraction of potentiated synapses is calculated over the number of synapses belonging to each subgroup.

When the *stop-learning* mechanism is inactive, at the end of the experiment the white distribution of *overlap* synapses is broad, while when the *stop-learning* mechanism is active *overlap* synapses tend to be depotentiated. This is the "microscopic" effect of the *stop-learning* mechanism since *overlap* synapses are pushed half of the times to the potentiated state and half of the times to the depressed state, and it is more likely for the *Up* synapses to reach earlier the potentiated state. When the *stop-learning* mechanism is active, once the potentiated synapses are enough to drive the output neuron about 80 Hz, further potentiation is inhibited for all synapses so that *overlap* synapses get depressed on average. This happens for sufficiently small transition probabilities.

Final distributions of the output frequencies for increasing overlap is illustrated in figure 3.9 (*stop-learning* mechanism inactive in the upper panels, active for the lower panels).

The frequencies are recorded during the "checking phase". In black the histograms of the output frequency ν_{dw} for the *Down* pattern, in grey those for *Up* pattern ν_{up}. It is clear from

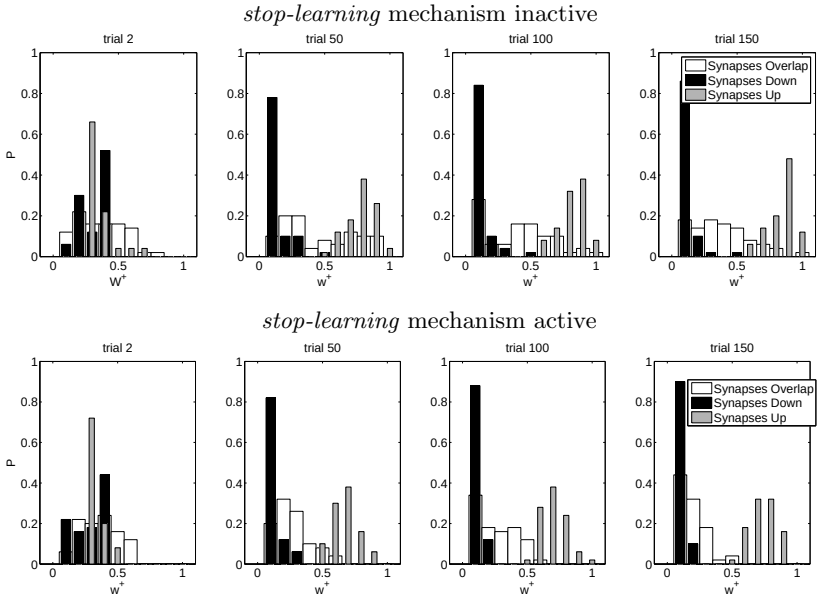


Figure 3.8: Distributions of the fraction of potentiated synapses (w^+) for three subgroups of synapses. Synapses that receive active input only during the presentation of Pattern *Up* are in the subgroups named *Up*. Synapses that receive active inputs only during Pattern *Down* presentations are in *Down* group. *Overlap* synapses are those that receive active inputs both during Pattern *Up* and Pattern *Down* presentation. The number of inputs belonging to both patterns is 10.

the figure that the output frequency distributions remain well separated even for high overlap when the *stop-learning* mechanism is active. The increase in the coding level together with the increase in the overlap could in principle push the output frequencies towards higher or lower values depending on the chosen parameters. In our conditions, as shown in figure 3.9, we have a shift towards higher frequencies when the *stop-learning* mechanism is inactive. The *stop-learning* mechanism is very effective in stabilizing the output rate distributions, as shown by the histograms in the second row of figure 3.9.

To provide a quantitative measure of how the perceptron performances are affected by the *stop-learning* mechanism, we report in Fig.3.10 the fraction of correct responses in the two cases, for different values of the overlap between the *Up* and *Down* patterns. An *Up* (*Down*) pattern is taken to be correctly classified if the perceptron's output firing rate is above (below) a pre-determined threshold. In order to choose an appropriate choice for the threshold (for each value of the overlap) a reasonable criterion is to approximately equalize the performances for the two patterns. The left panel in Fig.3.10 shows the resulting performances, where the choice of the optimal threshold has been made separately for the *stop-learning*-ON and the *stop-learning*-OFF cases. It is seen that, as expected, performances worsen with increasing overlap. When the *stop-learning* mechanism is active, the performance stays above 90% for all the values of the overlap explored, while performances decrease quickly with increasing overlap when the *stop-learning* mechanism is switched off.

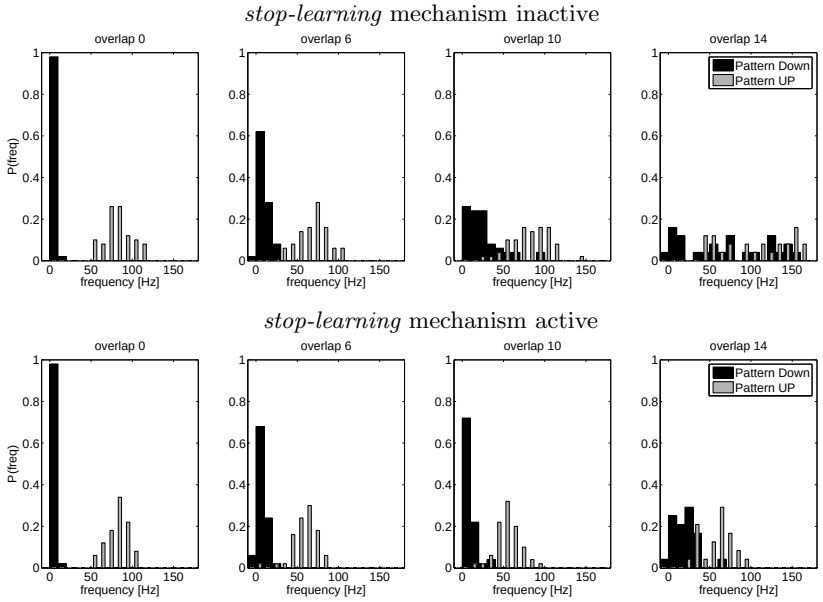


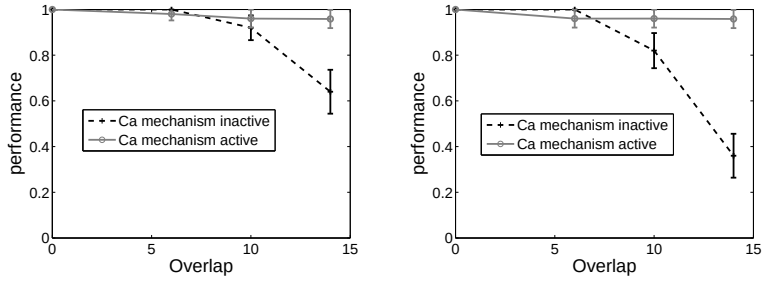
Figure 3.9: Distributions of perceptron frequencies after learning two overlapped patterns. Black bars refer to pattern *Down* stimulation, grey bars refers to pattern *Up*. Each panel refers to a different overlap.

For the *stop-learning*-OFF case the threshold varies between 25 and 95 Hz, while for the *stop-learning*-ON case it spans a much smaller range (40 - 45 Hz), consistently with the fact that the *stop-learning* mechanism tempers the variability of the output rate distributions at later stages of learning. Adopting an *ad hoc* mechanism for the optimization of the threshold for each overlap might appear questionable in view of the operation of the network in more "natural" conditions, in which the overlap between the patterns to be learnt is not pre-determined, and can vary[1]. For this reason we also checked the performances of the network for a fixed value (45 Hz) of the threshold (right panel in Fig.3.10). In this more 'realistic' case, as expected, a greater divergence between the two cases is seen, and the benefit of the *stop-learning* mechanism shows up also for moderate overlap.

3.8 Summary and Discussion

In this chapter I presented the design and tests of a semi-*neuromorphic* chip implementing a network of 32 integrate-and-fire neurons (see eq. 2.20) and 32x64 Hebbian plastic bistable synapses capable of stochastic learning and endowed with a self-regulating mechanism (see eq. 2.29 - 2.30). The synaptic matrix is completely reconfigurable: each synaptic contact can be set as active or inactive and its excitatory/inhibitory nature can be decided during an initial setup phase. The chip is endowed with structures that support AER-based (*Address-Event Representation*) communication with external devices.

[1]Though one might imagine, for stationary statistics of the patterns to be classified, a 'bootstrap' stage in which the network can self-tune its threshold to meet the given optimality requirement.



(a) (b)

Figure 3.10: Perceptron performances obtained for different values of the overlap between the *Up* and *Down* patterns with *stop-learning* mechanism active (solid line) and inactive (dashed line). (a): the threshold used is fixed at 45 Hz, see the text. (b): the threshold used is optimized for each value of the overlap. The error bars refer to the measure over the 50 runs.

A custom hardware-software setup has been designed and debugged. It allows to control the chip under repeatable conditions and to perform both low-level hardware tests and network-level experiments. Single circuits have been characterized under various conditions. Acquired data have been analyzed to extract parameters relevant for the network dynamics as for instance the values of the synaptic efficacy or the amplitudes of synaptic jumps (see tab. 3.1). Measures of synaptic transition probabilities were in agreement with results from a chip-oriented simulation. This demonstrates that the hardware implementation behaves as theoretical predictions despite all the unavoidable dissimilarities between analytical models and behaviors of analog circuits. A classification experiment was then performed: the network, configured as a simple perceptron-like architecture with 32 input and 1 output units, learned to classify two different correlated patterns. The results summarized in Fig. 3.10 show an improvement in classification performances when the self-regulating synaptic mechanism is activated, as theoretically predicted in [Brader et al., 2007].

The chip described in this chapter is only one of the hybrid analog-digital VLSI implementations of neural networks that have been developed in these last few years, each one optimized for a particular research goal. Some non-*neuromorphic* chips are thought as stand-alone simulation tools (as the powerful implementation described in [J. Schemmel and Ostendorf, 2007]) or as analog co-processors for numerical pc-based simulations [Alvado et al., 2004]. Some VLSI systems implement detailed conductance-based neurons designed to be easily interfaced with biological tissues [G. Le Masson and Bal, 2002] [Simoni et al., 2000]; *Neuromorphic* chips range from sensory devices such as silicon retinas [P. Lichtsteiner and Delbruck, 2006] [T. Y. W. Choi, 2005] and cochleas [van Schaik and Liu, 2007] to reconfigurable arrays of integrate-and-fire neurons [U. Mallick and Cauwenberghs, 2005] [Liu and Douglas, 2004] [Merolla and Boahen, 2006], to learning chips implementing models of spike-based synaptic plasticity as in [Indiveri et al., 2006] [Arthur and Boahen, 2006] [Riis and Hafliger, 2007] [Petit and Murray, 2003] or as in our case. A VLSI system very similar to CLANN has been described in [Mitra et al., 2006] and experimental results are reported in [Mitra et al., 2007].

One innovative characteristic of CLANN is its completely reconfigurable matrix of on-chip recurrent connections. Some of the VLSI system cited above realize connections among on-chip neurons via off-chip programmable channels [U. Mallick and Cauwenberghs, 2005] [Indiveri et al., 2006] while other implementations are endowed with hardwired internal connections [Liu and Douglas, 2004] [Merolla and Boahen, 2006] [E. Chicca and Douglas]. The connectivity matrix of CLANN easily allows to create local and reconfigurable connections without adding workload on external buses.

CLANN, whose capabilities have not been completely explored yet, has been conceived as a flexible test chip; it proved to be a reliable piece of hardware behaving in agreement with theoretical predictions; its circuits and architecture have been chosen as the starting point for a larger and more sophisticated chip described in the next chapter.

Appendix C

C.1 Circuits details and layout

C.1.1 Synapse

In figure 3.11 the synapse schematics is reported, it is arranged in the same blocks of figure 3.3. Both AER and recursive synapses follow this scheme.

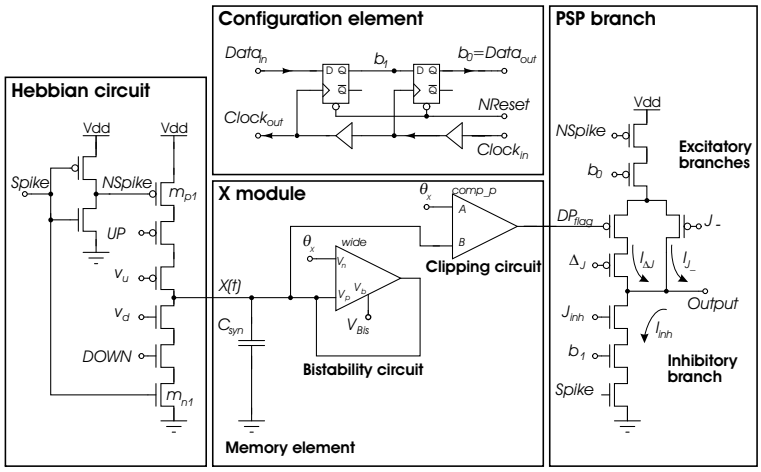


Figure 3.11: Synapse schematics. The block level diagram of the synapse is reported in fig. 3.3. The *Hebbian* block receives the *Spike* signal from the presynaptic neuron. *UP* and *DOWN* signals are two digital bits coming from the Calcium circuit measuring the activity of the postsynaptic neuron. V_{up} and V_{down} signals are two biases voltages to tune the maximum current which influences the evolution of the synaptic variable $X(t)$ (the voltage across the synaptic capacitor C_{syn}). The *Bistability Circuit* compares $X(t)$ with a bias voltage Θ_X and implements the refresh mechanism of eq. 2.30. The comparator named *Clipping Circuit* generates in output the digital flag DP_{flag} accounting for the the synaptic state (potentiated or depressed). *PSP* (Post-Synaptic Potential) is activated by the *Spike* signal and its negated version *NSpike*. J_-, ΔJ and J_{inh} are three analog biases. b_0 and b_1 two digital bits stored in two flip-flops in the *Configuration Element*. *Clock*, *DATA* and *NReset* digital lines controls the flip-flops.

The spike arrives on the terminal *Spike* of the Hebbian circuit under the form of an "long" pulse coming from the shaper circuit. Its arrival closes the switches m_{n1} and, through the inverter, also the switch m_{p1}. This activates the output section of the Hebbian circuit which is divided in two branches: a p one composed of three p-type MOSFETs and a complementary n branch consisting in 3 n-type transistors. During the spike pulse, according to the values of the digital signals UP (an active-low signal) and $DOWN$ (an active-high signal), either the p branch or the n branch is enabled. This respectively causes an injection

or a subtraction of current into or from the capacitor C_{syn} (about 0.5pF) inducing an upward (J_{up}) or a downward (J_{dw}) jump in the internal synaptic variable $X(t)$, which is the voltage across C_{syn}. The amplitude of the injected current, can be tuned with the bias v_u, the amplitude of the subtracted current with v_d; both are typically in the order of few nA. The digital signals UP and $DOWN$ are the outputs of the comparators system described below. Clearly, the MOSFETs used as switches occupy the minimum space allowed by the 0.35μm technology; those used as current regulators are more than double in size. To implement the refresh mechanism, C_{syn}, in the X module, is connected to a wide-range transconductance amplifier [Liu et al., 2002] with positive feed-back. Its positive input is connected to $X(t)$ and the negative input is connected to the bias voltage θ_X. In this configuration if $X(t) > \theta_X$ the amplifier injects a small current I_{bis} into C_{syn}, otherwise the current I_{bis} is sucked from the capacitor: this forces $X(t)$ to drift towards its upper bound (H_s) or towards its lower bound (H_i). Low amplitudes of I_{bis} and corresponding slow dynamics of $X(t)$ (about few Volts per second) are obtained by keeping the MOSFETs of the amplifier in the weak-inversion regime. The bias V_{bis} is used to tune the amplitude of I_{bis} in the range of pico-Ampere. The Clipping circuit, a transconductance open-loop amplifier, compares the instantaneous value of $X(t)$ to θ_X. The output is a digital signal DP_{flag} encoding the synaptic state: potentiated ($DP_{\text{flag}} = 0$) if $X(t) > \theta_X$, or depressed ($DP_{\text{flag}} = 1$) if $X(t) < \theta_X$. The excitatory or inhibitory nature of the synapse is determined by the configuration bits b_0 and b_1 stored in two D-type flip-flops of the Configuration Element. If $b_0 = b_1 = 1$ than the inhibitory branch (see figure 3.11) of the PSP block is enabled and a current I_{inh} flows from the *Output* node to ground during the *Spike* pulse. The inhibitory branch is composed of three n-type MOSFETs, two acting as switches and one as a current regulator (as in the Hebbian circuit): I_{inh} amplitude can be adjusted with the bias J_{inh}. If $b_0 = b_1 = 0$ the synapse is set excitatory and the excitatory branch, composed of five p-type transistors, is active: during the pre-synaptic spike pulse, a current is injected into the *Output* node. The amplitude of this current will be $I_- = I_{J_-}$ if the synapse is depressed ($DP_{flag} = 1$) or $I_+ = I_{J_-} + I_{\Delta J}$ if the synapse is potentiated ($DP_{flag} = 0$). The amplitude of both I_{J_-} and $I_{\Delta J}$ can be tuned using the biases J_- and ΔJ respectively. If $b_0 = 1$ and $b_1 = 0$ both branches are disabled and the synaptic contact is inactive.

The two flip-flops of a Configuration Element are connected in series as well as the Configuration Elements of different synapses: they globally constitute a 4032-bit shift register into which the synaptic configuration is serially fed during an initial phase; only two signals are required, a clock and a data line. The design of a symmetric clock distribution tree, for the entire shift-register, seemed not to be an "economic" solution for the geometry of the synaptic matrix. We decided to provide the flip-flops with the clock using an unique line which runs along the whole shift-register. Relevant delays on this line, which is endowed with one clock buffer for each flip-flop, could cause errors in the bit transfer. Specifically, errors occur if the input-to-output clock buffer delay is similar to the input-to-output flip-flop delay. In this condition a flip-flop could receive the clock rising edge after that the previous flip-flop has changed its output state (and not before as it should be). To prevent this problem data and clock signals propagate in opposite directions so that the first flip-flop to be updated is the last one in the chain. This ensures that each element of the shift register correctly updates its output before its input is changed by the previous flip-flop, which will be updated on the following step.

In figure 3.12, the layout of the synapse is reported; all the four metal layer and two polysilicon ones were used. More than one third of the space is occupied by AMS standard

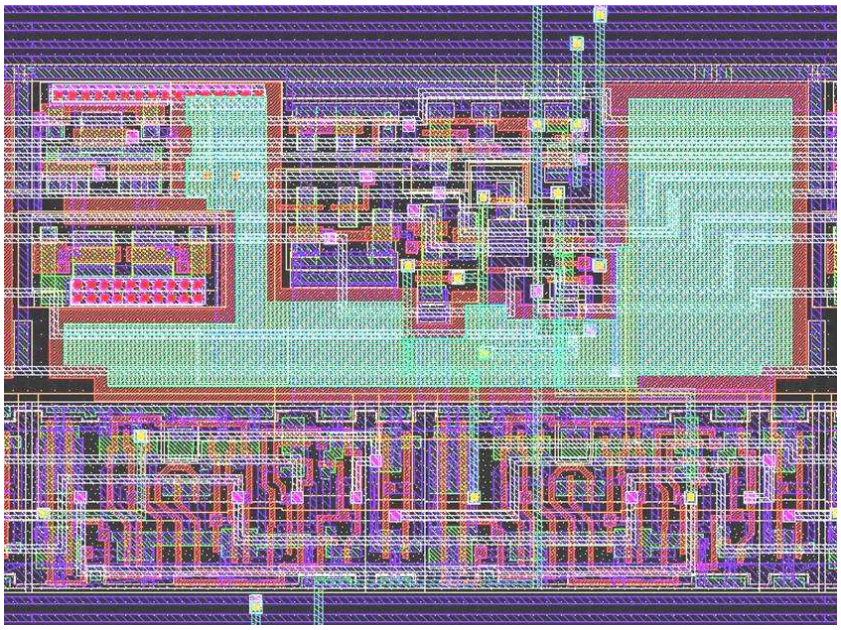


Figure 3.12: Synapse layout. The lower part is constituted by AMS (Austria Micro Systems) standard cells: two flip-flops and two clock buffers form the *Configuration Element* of fig. 3.11. The upper part comprises the analog custom circuits: the *Hebbian*, *X(t)* and *PSP* blocks. In green is clearly visible the capacitor (C_{syn} in fig. 3.11) shaped to shield sensitive analog circuits from noisy digital ones. Blue horizontal paths on the top carry the UP, $DOWN$ and *Spike* signals. White paths over the custom circuits are analog lines for input bias voltages and output synaptic currents.

cells: flip-flops and clock buffers. The custom circuits of the Hebbian, X and PSP blocks are visible in the upper part of the figure. The green area, is occupied by the capacitor realized with the two available polysilicon layer. It is shaped to shield the sensitive analog circuits from the noisy digital ones. As the capacitor, all the synaptic components have been placed, shaped and connected with a particular care to cross-talk reduction. In the upper part three horizontal blue nets are visible, they carry the UP and $DOWN$ signals and the spike pulse. They travel along a digital channel bordered by two guarding bars (only one of the two guarding bars is visible, the other one is the ground net of the standard cell belonging to the next synapse). Far from these digital nets the analog ones run, on the second metal layer (white), above the custom cells. One of them is the analog *Output* node of the synapse which is a really delicate net directly connected to the neuron capacitor. Noise on this line strongly affects the entire network behavior; the other analog lines carry the bias voltages for the synaptic circuits. As shown in figure 3.13, each digital net delivering the spike pulse, coming from the neurons or from the decoder, crosses all the analog lines. This digital net travel horizontally on metal one, in vertical on metal four (in green). This allow shielding the analog lines with a power plane of metal three (not shown in figure 3.12) that covers the

entire synapse. This become a sort of safe box inside which all the analog signals are kept. The digital signals travel outside the boxes entering only when necessary. Thanks to this architecture we did not experienced any cross-talk problems during the experimental tests.

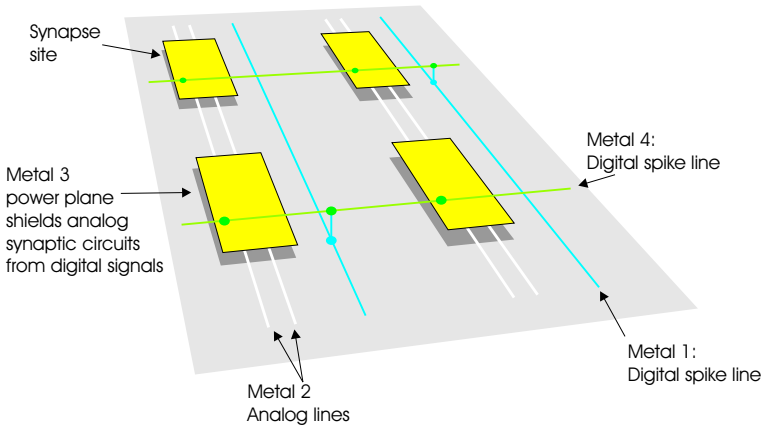


Figure 3.13: Synaptic matrix paths: in blue (metal 1) and green (metal 4) the digital paths, in white (metal 2) the analog ones. Metal 3 (yellow) constitutes power planes that shield delicate analog lines and circuits from digital nets.

C.1.2 Neuron

The implemented neuron is the IF neuron proposed in [Indiveri et al., 2006] with constant leakage term and a lower bound for the membrane potential $V(t)$. The author of this circuit specifically addresses the problem of power consumption, reducing it to less than 1.5μW for typical working conditions. The circuit schematics, in figure 3.14, is the results of an evolution of IF neuron circuits started with the Axon-Hillock circuit proposed in [Mead, 1989] in 1989. The neuron circuit is endowed with elements to implement spike frequency adaptation, to set an arbitrary refractory period and to modulate the neural threshold voltage. In what follows a short description of the circuit schematics is given (see figure 3.14), for further details please refer to [Indiveri et al., 2006]. Four blocks can be considered: the soma, the spike, the refractory period and the frequency adaptation block. The first comprises the capacitor, the MOSFET m_{20} that explicitly implements the leakage current and MOSFET m_{21} for the constant afferent current. Onto the common node of these elements, synaptic contributions arrive inducing jumps in the "membrane" voltage $V(t)$ across the capacitor C_{mem}. When $V(t)$, modulated by the source follower $m_1 - m_2$, approaches the threshold of the inverter $m_4 - m_5$, it activates the spike emission process. A positive feedback loop makes the inverter switch very rapidly, saving power; thus the node V_1 becomes 0 and node V_2 suddenly switches to Vdd. This closes m_{12} which shortcuts the

capacitor C_{mem} to ground. This feedback loop brings $V(t)$ back to its initial value. The mosfet from m_8 to m_{12} implements the absolute refractory period of the neuron. Once $V(t)$ is reset, the node V_1 switches back to Vdd and V_2 is then discharged at the rate imposed by $V_{\rm rf}$ and by the parasitic capacitance on node V_2. At the end of this process, that implements the refractory period, C_{mem} is released and the integration starts again. In the meanwhile, the inverter $m_{13}-m_{14}$ reads in input the voltage V_1 and produces at its output an active-high digital pulse representing the spike. The spike frequency adaptation is implemented through a current-mirror integrator (from m_{15} to m_{19}) which dictates the amplitude of the current $I_{\rm adap}$ subtracted from the capacitor. This system works thanks to the parasitic capacitance at node V_{ca}: a current set by $V_{\rm adap}$ charges the capacitance upon the emission of a spike, while the dark current discharge the capacitance. The result is a current $I_{\rm adap}$ increasing with the neuron firing rate (see [Indiveri et al., 2006] for further details). The original schematics and layout, provided by the Institute of Neuroninformatics (INI) in Zurich, have been slightly modified to optimize communication with the other circuits.

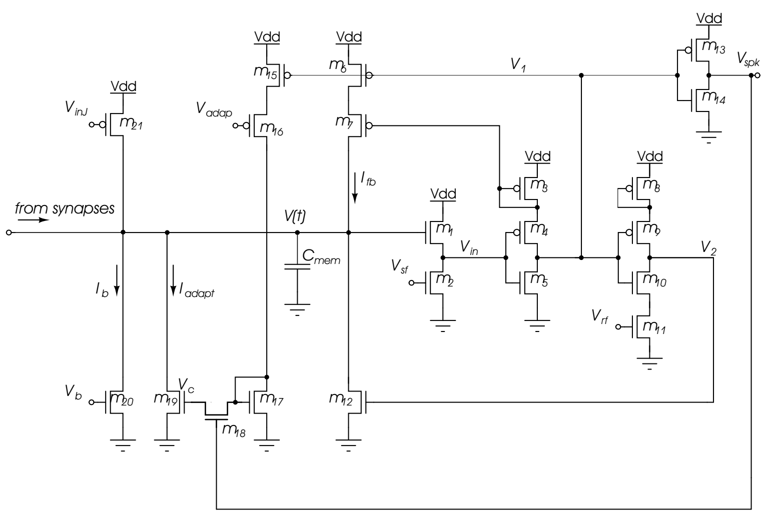


Figure 3.14: Neuron schematics. $V(t)$ is the neuron membrane potential. A series of analog biases control the neurons parameters: V_{inj} tunes a constant afferent current charging the capacitor C_{mem}, V_b controls the constant leakage current that discharge C_{mem}, V_{adap} regulates the efficiency of the frequency adaptation mechanism, V_{sf} decides the firing threshold and V_{rf} sets the duration of the refractory period. V_{spk} is the digital output where a pulse is generated when $V(t)$ crosses the firing threshold. The current I_{fb} activates during the action potential and speeds up the increase of $V(t)$ towards Vdd during the raising phase.

C.1.3 Calcium

When a spike is generated, the signal $NSpike$ (figure 3.15a) triggers a current tuned by the v_{ca_i} bias, which charges the capacitance C_{ca} so that the voltage $C(t)$ undergoes an upward jump J_{Ca}. In the absence of spikes a constant current set by v_{ca_t} linearly discharges C_{ca} so that $C(t)$ decays towards ground. The instantaneous value of $C(t)$, together with the

post-synaptic neuron potential $V_{post}(t)$ are than compared to a set of threshold to generate the UP and $DOWN$ signals. Three kinds of comparators based on the two stages open-loop transconductance amplifier are used in the comparators system shown on the right side of figure 3.15. The digital output of the comparators is *high* or *low* depending on the difference of voltages applied to the positive input A and negative input B. Comparators tagged *comp_p* are two stages open-loop p transconductance amplifier: if $V_A > V_B$ the output is *high*, otherwise the output is *low*. The comparator tagged *comp_p_EN* is a two stages open-loop p transconductance amplifier accepting two enable input signals En_1 and En_2: when both En_1 and En_2 are low than the comparator works as a normal p comparator, otherwise the output is forced to be *high*. The *comp_n_EN* comparator is analogous to the *comp_p_EN* one but designed with complementary mosfet. This system of comparators implement the conditions described in (2.29).

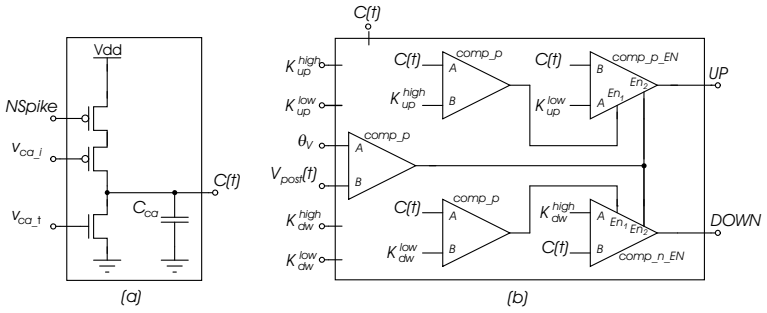


Figure 3.15: Calcium circuit schematics: on the left the circuit that generates the Calcium variable $C(t)$, i.e. the voltage across the capacitor C_{ca}. $Nspike$ is the digital spike pulse from the neuron. V_{ca_i} tunes the amplitude of the upward jumps that $C(t)$ undergoes upon the arriving of a spike. V_{ca_t} sets the leakage current that discharges C_{ca}. On the right side the system of comparators implementing eq. 2.29. K and Θ_V are analog voltages.

C.1.4 Shaper and other circuits

The shaper circuit used in CLANN, in figure 3.16, exploits a switch capacitor configuration to transfer a certain amount of charge from C_1 to C_2 upon the arrival of a spike. Both the AER and recursive shapers, the former placed next to each AER synapse, the latter soon after each neuron, follow the same schematics; the only difference is that the AER one is preceded by an AND gate combining the X and Y pulses generated by the decoder, while the recursive circuit receives the trigger directly from the V_{spk} line of the neuron. When a pulse on V_{in} opens the mosfet m_1 and closes m_2, the node V_s undergoes an upward jump whose amplitude, given the initial conditions $V_1 = 0$ and $V_s = Vdd$, is $\Delta V = Vdd \cdot \frac{C_2}{C_1+C_2}$. The jump amplitude depends only on the capacitors values and on the initial conditions. A sufficiently high jump makes the inverter $m_4 - m_6$ switch. Once V_{in} is released, C_1, linearly charges with a rate tuned by V_{bias}, V_1 increases towards Vdd and the inverter switches back. An extended pulse of tunable amplitude is generated. To increase the slope of its edges, a second inverter $m_7 - m_9$ is added. Both the inverter are limited in current to reduce power

consumption. The circuits described above constitute the core of the chip, the network. To

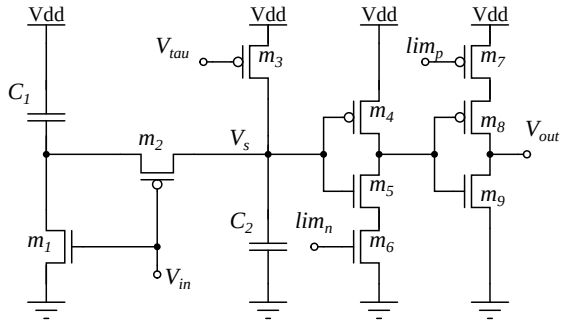


Figure 3.16: Pulse shaper schematics. m_1 and m_2 act as switches and create a switch capacitor scheme with C_1 and C_2. This is the core of the circuit. Node V_{in} is a digital input line which receives the incoming spike pulse. lim_n and lim_p are two analog voltages that limits the currents flowing on the corresponding inverters.

make them work and communicate with external devices other few structures are required. On the input side the C-element and the decoder, on the output side the memory element, the encoder and the arbiter; moreover the padframe is necessary. The memory element has been implemented using standard D-type flip-flops. The AER structures are built from custom cells [Boahen, 1999b] using automatic *ad-hoc* compilers written by various people at the INI [Bartolozzi, 2007] [Chicca, 2006]. On the input side the C-element follows the point-to-point AER protocol, while on the output side the multi-sender protocol is implemented [Dante et al., 2005]. The custom pads (from INI) were placed and connected to the core by hands: no critical analog nets crosses noisy digital ones without being shielded by large power lines.

Chapter 4

FLANN

In this chapter a new chip named FLANN, the big brother of CLANN, is presented: it implements a network of 128 integrate-and-fire neurons and 16384 spike-driven rate-based Hebbian plastic bistable synapses. Directly derived from CLANN, FLANN is the Final Learning Attractor Neural Network designed within the European ALAVLSI project and it is the result of a growing experience accumulated designing and testing CLANN. Besides being much larger, FLANN is endowed with a series of technical improvements aiming to increase the reliability, the computational power and the testability of the system. The shaper and calcium circuits have been completely redesigned, single synapses can be configured as AER or recursive, can be initially set as potentiated or depressed and their internal states monitored on-line without interfering with the network dynamics. FLANN has been realized in 0.35 μm AMS CMOS technology, it occupies 68.9 mm^2 and is hosted in a 256 PGA package. In figure 4.1 the layout of the chip and its id picture are shown.

4.1 Architecture

The main blocks of FLANN are the same of CLANN, there are the neurons array, the synaptic matrix, the AER input and output systems. The first notable difference is in the structure of the synaptic matrix. Here there is no more the differentiation between an AER block hosting the synapses for external connections and a recursive part for those synapses accepting spike from local neurons. Each synapse can be set as AER or recursive through a configuration bit. In a large variety of architectures this degree of configurability allow to reduce Silicon waste. If it is possible to exploit all the synapses, for instance in a network composed of four CLANN chips with neurons uniformly connected at 50%, than there are no advantages; but if, on the other side, the network architecture requires only AER synapses, as the case of a set of externally controlled independent perceptrons, than, half of the synapses of CLANN, the recursive ones, would do nothing else than occupy precious Silicon area. A similar situation there would be if the chip is used as a pure recursive block: the AER synapses, in this case, would be useless. The AER/recursive configurability, represents a key feature to reduce the fraction of unused Silicon. Moreover, besides improving the architecture flexibility, having only one kind of synapse throughout the entire matrix greatly simplifies the layout design.

As visible in figure 4.1 the synaptic matrix is divided in four submatrixes. This comes

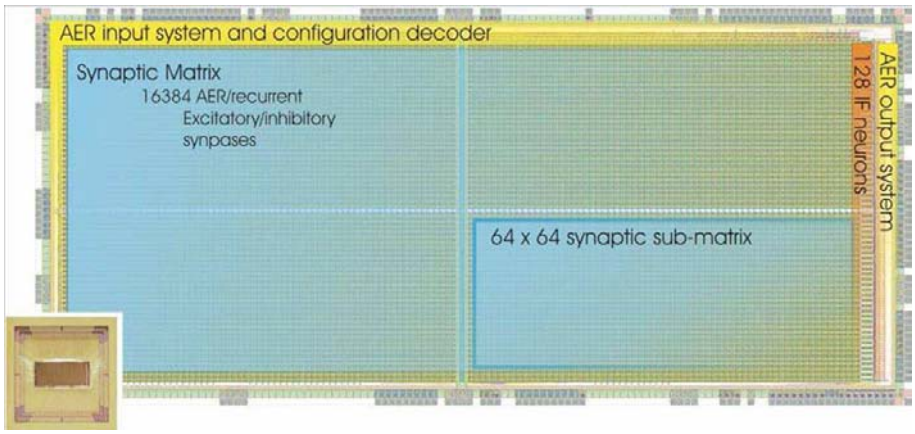


Figure 4.1: FLANN, Final Learning Attractor Neural Network, top layout view. Visible blocks are the array of 128 integrate-and-fire neurons, the synaptic matrix of 128x128 configurable AER/recursive, excitatory/inhibitory matrix, AER (*Address Event Representation*) input and output systems. FLANN has been realized in 0.35 μm AMS CMOS technology, it occupies 68.9 mm^2 and is hosted in a 256 PGA package.

from the requirement of buffering digital signals to be transmitted along the entire length of the chip 11mm long. The capacitive loads of the digital lines, as for instance those carrying the spike pulses, have been considered non negligible; to maintain a decent slew rate of the edges of the digital signals, we split the matrix in four smaller parts and we buffered the digital signals in input to each block. The choice of having four submatrixes come form quantitative analysis of the loads, considerations on noise produced by the buffers, and criteria of layout simplicity. Moreover, to respond to experimental demands faced testing CLANN chip, there are two independent sets of parameters, one for the submatrixes on the left side, one for those on the right side. And in addition the first column of synapses on the left has independent biases to tune the synaptic efficacy. Having different parameters for different subset of synapses is useful when complex, non homogeneous architecture should be tested; in particular the first column of synapses has been thought to be used as fixed excitatory AER synapses when this is required to control from outside the behavior of the neurons, as for instance to implement a spiking teaching signal.

The neuron array is highlighted in orange on the right side of figure 4.1. It consists of 128 integrate-and-fire neurons identical to those designed for CLANN. Next to each neuron the new Calcium circuit comprises a differential pair integrator (DPI) configured to act as a low pass filter, winner-take-all circuits, and current conveyors. The spikes generated by the neurons are sent to the synapses configured as recursive, to the Calcium circuits and to the AER output system (in yellow) composed as in CLANN of an array of memory elements, an arbiter and an encoder.

The digital MUX is present too; it reads the 128 neurons output and report them on 8 digital output pins.

On the input side the AER system (in yellow), is composed of an element that handles the

AER handshake and of an X-Y decoder. The entire system has been completely rethought and the AER handshake streamlined for multi-chip systems. Standard digital cells have been used and decoders have been designed exploiting automatic placement and routing tools.

The configuration of the synaptic matrix is no more serially fed: each synapse can be independently addressed using XY decoders placed next to the AER ones. Creating a giant shift register composed of 32768 cells seemed not a great idea, at least from a reliability point of view: it is sufficient a non-working cell to compromise the system configuration ability. An XY strategy requires not only a decoder to address a specific synapse but also a more complicated signal protocol that have to deal both with the configuration data and with the synapse address data. We try to simplify as much as possible the system and we decided to switch to this method also to introduce the possibility to read, through a dedicated hardware, the state of a specific synapse. The configuration decoder, compared to a serial scanner, allow to select more easily the synapses to monitor. Another configuration novelty of FLANN is the possibility to set the initial state of each synapse at hardware level, allowing a faster (no AER protocol and bias tuning required) and more reliable "download" of the initial state of the connectivity weights.

The padframe of the chip is composed of 200 pads, derived from those of CLANN, they were modified to reduce voltage drops on the power lines and noise generation on the input, digital pads. In CLANN the padframe determines the chip area, in this chip, on the contrary, the Silicon area necessary for the core imply such a long perimeter that roughly only half of it is occupied by the padframe (see figure 4.1).

4.2 Signal flow

The signal flow is analogous to the one described in the previous chapter for CLANN. In short: an external spike comes under the form of an AER event encoding the address of the target synapse. This event is accepted by the new AER input logic element and transformed by the decoders in a couple of X and Y pulses each lasting about 100. These pulses simultaneously stimulate respectively the line and the column of the synaptic matrix on which the target synapse resides. Next to each synapse, a shaper circuit receives the X-Y signals and generates in output a single digital pulse lasting about $10us$ that triggers 1) jumps on the synaptic internal variable $X(t)$ and 2) jumps on the postsynaptic neuron potential $V(t)$ according to the models described in chapter 2. If the neuron membrane potential reaches a given threshold θ_V, a spike is emitted and sent to those synapses connected to the neuron axon and configured as recursive contacts. Differently from what happens in CLANN, here the spike is not extended by a shaper placed next to the neuron but by the shaper placed next to the target synapse. At each synaptic site the same shaper accepts AER or recursive spikes: a digital MUX, according to a configuration bit, connects the shaper either to external or local lines, making the synapse AER or recursive. As in CLANN the local spike is sent not only to the synapses but also to 1) the Calcium circuit which generates the UP and $DOWN$ signals controlling the synaptic dynamics, 2) to the AER output system that encodes the spike in an AER event then sent to external devices, and 3) to a digital MUX, useful during a debug phase, which directly reports the spike on an output test pad. The AER output system is composed of a memory element that decouples internal network activity from the external AER bus one: as in CLANN it is an array of D-type flip-flops, 128 in this case, one for each neuron. A flip-flop is set by the neuron and

reset by the arbiter. When a flip-flop is set, a request of accessing the AER bus is forward to the arbiter which, if the bus is free, accepts the request and sends a "go" signal to the encoder which in turn puts on the external bus the address corresponding to the neuron that fired the spike. If the bus is busy the spike has to wait before being served: the neuron anyhow resets immediately after the spike emission, while the flip-flop remains set as long as the arbiter gives it an acknowledge signal once the spike has been transmitted.

4.3 Block level description

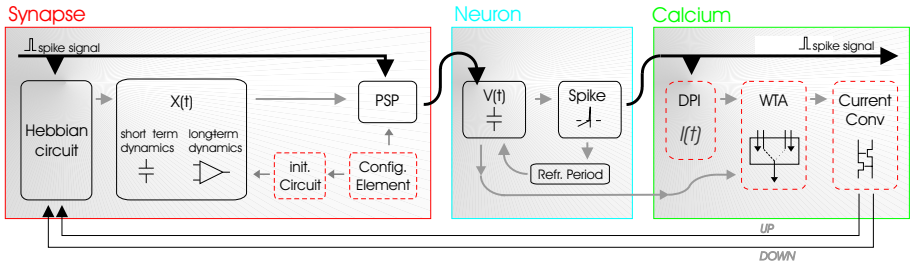


Figure 4.2: Synapse, neuron and calcium circuits: block level diagram. In red are highlighted new or modified circuits compared to CLANN ones (see fig. 3.3). The synapse have the new *initialization circuit* which is the active element that sets the initial state of the synapse. The *configuration element* has been completely modified to accept the new protocol for loading the synaptic configuration, to handle the new AER/Recursive configurability, and to set/read the synaptic state. On the right side of the figure there is a completely new current-mode Calcium circuit: the calcium variable in encoded in a current, and not in a voltage as was in CLANN, generated by a linear integrator circuit; consequently a new system of comparators has been designed and a current-to-voltage converter introduced in the chain.

For the sake of clarity, to visualize where, at circuit level, the new features of FLANN comport the introduction of new circuits and modifications of existing elements, in figure 4.2 is depicted a block level diagram of the synapse, neuron and calcium circuits: the new and modified blocks are highlighted in red. The synapse have the new *initialization circuit* which is the active element that sets the initial state of the synapse. The *configuration element* has clearly been completely modified to accept the new protocol for loading the synaptic configuration, to handle the new AER/Recursive configurability, and to set/read the synaptic state. On the right side of the figure there is a completely new current-mode Calcium circuit: the calcium variable in encoded in a current generated by a linear integrator circuit; consequently a new system of comparators has been designed implementing in currents the inequalities 2.29. The synaptic internal variable $X(t)$ then undergoes jumps according to

$$\begin{aligned} X(t) &\rightarrow X(t) + J_{up} \quad \text{if} \quad V_{post}(t) > \theta_V \quad \text{and} \quad I_{up}^{low} < I_{Ca} < I_{up}^{high} \\ X(t) &\rightarrow X(t) - J_{dw} \quad \text{if} \quad V_{post}(t) \leq \theta_V \quad \text{and} \quad I_{dw}^{low} < I_{Ca} < I_{dw}^{high} \end{aligned} \tag{4.1}$$

where J_{up}, J_{dw} and the thresholds $I_{up,dw}$ are all positive constants. The output of these comparisons is encoded in two currents that the Current Conveyors block converts to voltages to maintain the compatibility with the synapses.

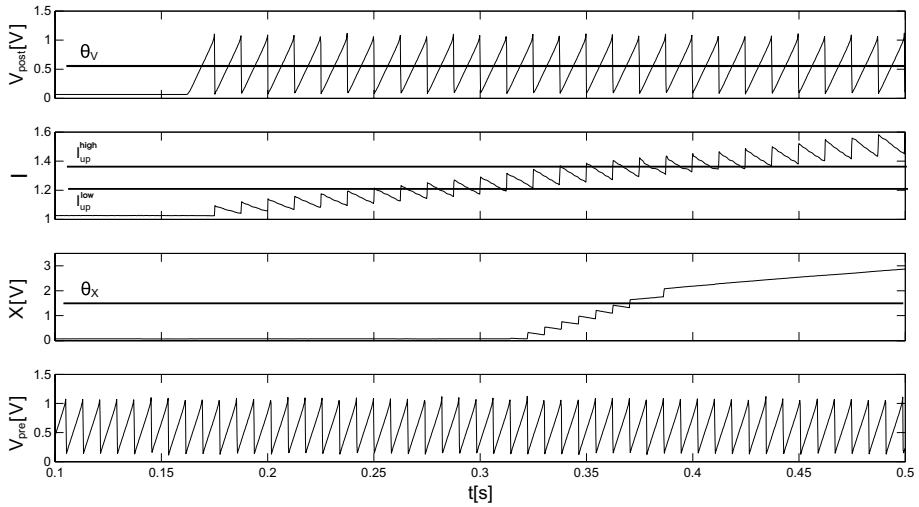


Figure 4.3: FLANN synapse at work: an illustrative example of the *stop-learning* mechanism; top to bottom: postsynaptic neuron potential V_{post}, calcium variable $I = \frac{I_{Ca}}{I_0}$, internal synaptic variable X, presynaptic neuron potential V_{pre}. The chosen thresholds values for the Calcium circuit allow only upward jumps. The synapse is initially set as depressed; at $t = 0,16s$ a constant current induces a periodic firing activity for the postsynaptic neuron and the calcium variable I undergoes upward jumps moving to a new asymptotic average value. When $I_{up}^{low} < I < I_{up}^{high}$, $X(t)$ jumps are allowed

Figure 4.3 illustrates the behaviors of the main analog variable involved in the network dynamics. The bottom trace is the presynaptic neuron potential: upon the arrival of one spike, the internal synaptic variable $X(t)$, second trace from bottom, updates its value according to inequalities 4.1. The thresholds to be compared with the current $I_{\rm ca}$ were set to have only upward jumps: $I_{dw}^{low} = I_{dw}^{high}$ and $I_{up}^{low} < I_{up}^{high}$. The synapse is initially set depressed and then a constant current is injected into the postsynaptic neuron (top trace). The neuron activity begins and the calcium variable I_{Ca} undergoes upward jumps moving to a new asymptotic average value. The trace labeled I corresponds to $\frac{I_{Ca}}{I_0}$ where I_0 is a constant value; it is derived from the measured V_{Ca} and computed as $I = e^{(V_{dd} - \kappa V_{Ca})/U_T}$, where κ is the subthreshold factor, taken equal 0.7, and U_T is the *thermal voltage* which, at room temperature, is equal to 25mV. When I_{Ca} is smaller than I_{up}^{low}, transitions of $X(t)$ are disallowed. In the intermediate regime between I_{up}^{low} and I_{up}^{high}, up jumps are allowed. When I_{Ca} is larger than I_{up}^{high}, jumps of X are once again disallowed. Note that the refresh mechanism attracts $X(t)$ towards its upper bound when $X(t) > \theta_X$ (i.e. when $t > 0.37s$) and towards its lower bound when $X(t) < \theta_X$ ($t > 0.37s$).

4.4 Synapse and shaper: circuits and layout

In this section a detailed description of the new parts is provided

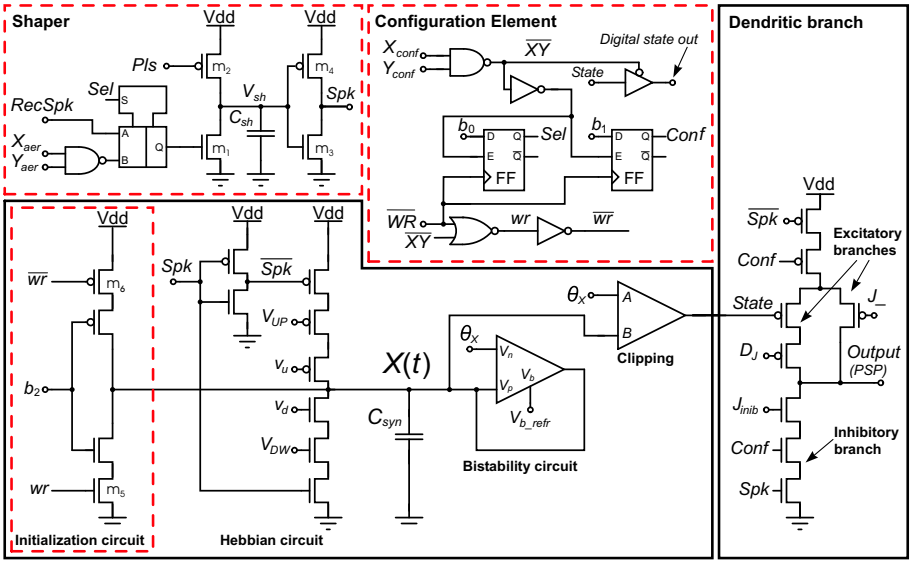


Figure 4.4: FLANN synapse and shaper, schematic view. New or modified parts are highlighted in red (to compare to fig. 3.11). On the left the *Initialization Circuit* activates when the write (wr) signal stays high: it shorts the synaptic internal variable $X(t)$ to Vdd or to ground accordingly to bit b_2. This imposes the synaptic state: potentiated or depressed. The write signal, from the modified *Configuration Element*, is high only when $W\bar{I}R$ signal is active and the synapse is selected using the X_{conf} and Y_{conf} signals. When a synapse is selected, the three-state buffer in the *Configuration Element* reports on an accessible pin the digital state of the synapse. The modified *Shaper* accepts either recurrent or AER spikes thanks to a standard digital MUX in input. The pulse extension is based on the slow charge of capacitor C_{sh}, charge that depends on the current flowing through m_2 tuned by the analog bias pls

.

4.4.1 Synapse

The schematic view of the synaptic circuit is reported in figure 4.4. The *configuration element* have been designed using standard AMS digital cells, it consists of two flip-flops to store the configuration bits b_0 and b_1, a three-state buffer for the synaptic state readout, and four logic ports to combine the incoming digital signals. These signals comprise an horizontal and a vertical selection line, X_{conf} and Y_{conf} coming from the configuration decoder, a write signal WR, and the bits to be memorized: bit b_0 chooses between an AER or recursive synapse, the output of the corresponding flip-flop is the signal sel that controls the digital MUX in input to the shaper; bit b_1 allows to configure the synaptic contact as excitatory or inhibitory setting the value of the digital line $conf$ which enables either the excitatory

($conf = 0$) or the inhibitory ($conf = 1$) branch of the *PSP* block. Another bit b_2 is required to set the initial state of the synapse. We chose not to allocate hardware resources to store it: b_2 directly impinges onto the synaptic *initialization circuit* which activates during the write signal. if $b_2 = 0$ the capacitor C_{syn} is shortcut to Vdd and hence the synapse is forced to be potentiated; if $b_2 = 1$ the synapse will be depressed. The three bits $b_{1,2,3}$ together with the *WR* signals are broadcast to all the synapses, the selection lines X_{conf} and Y_{conf} take care of enabling one synapse at time. Thus, to configure a synapse the protocol is the following: 1) the data $b_{1,2,3}$ lines are driven at the right values, 2) the address of the synapse is presented to the decoders, 3) an active-high pulse on the write signal is sent: it triggers the bits memorization on the flip-flops and the initialization of the synaptic state. Clearly the write pulse has to start after the synaptic address has been completely processed by the decoder, and has to last the time required by the *initialization circuit* to bring the voltage $X(t)$ to Vdd or ground.

To read the state of the synapse it is sufficient to select it: the three-state buffer, whose input is connected to the *state* line, activates and drives an output digital pin, common to all the synapses in the same submatrix, making available to external devices the digital state of the synapse. In CLANN reading or setting the synaptic state involved parameters tuning, AER spikes stimulation and an off-line analysis of the chip reaction. In FLANN the introduction of three-state buffers let monitoring, on-line, the evolution of the synaptic weights without affecting the network dynamics, scanning continuously the entire matrix or just a chosen subset of synapses.

The *initialization circuit* consists of two n-type and two p-type MOSFETs; the transistors driven by b_2 simply form an inverter that is enabled when the signal wr and its negative version $\overline{wr}$ close the switches m_5 and m_6 connecting the inverter to the power supply so that the node $X(t)$ is forced to ground or to Vdd.

4.4.2 Shaper

The analog part of the shaper is much simpler compared to the circuit used in CLANN. There are only one capacitor C and four MOSFETs. When the output of the digital MUX undergoes an active-high pulse, the capacitor is shortcut to ground through the transistor m_1. After the pulse duration Δt_{in}, the line is released and the capacitor is slowly recharged by the current tuned by the analog bias *pls* flowing through the p-type MOSFET m_2. The inverter composed of m_3 and m_4 initially switches when the voltage across the capacitor V_{sh} goes to zero, and switches back after an interval of time

$$\Delta t_{\text{pls}} = \frac{\theta_{\text{inv}} C}{I_{\text{pls}}} \tag{4.2}$$

where θ_{inv} is the threshold of the inverter. On the output line tagged Spk, the pulse will last $\Delta t_{\text{out}} = \Delta t_{\text{in}} + \Delta t_{\text{pls}}$. In typical conditions Δt_{out} is set to about 10μs so that the difference in the length between AER ($\Delta t_{\text{in}} = 100ns$ and recursive ($\Delta t_{\text{in}} = 20ns$) spikes do not affect the network dynamics: the amplitudes of the induced jumps differ for about the 1%, both for the internal synaptic variable and for the postsynaptic neuron potential. To obtain a 10μs spike pulse.

The advantage of the circuit in CLANN is that the amplitude of the jump induced in the voltage across the capacitor is less affected by the mismatch because in CLANN it depends on the capacitance values and not on the current flowing through a MOSFET. The effect of

this mismatch is relevant when two spikes with an ISI minor than Δt_{out} arrive on the shaper so that the desired output pulse should double its length. To do this a fine control on the jump amplitude is required and the mismatch could severely compromise this mechanism. As already said ISI minor than 10μs are not typical in our network. In FLANN the MOSFET m_1 acts as a switch and ensures that the amplitude of the jump equals, in every shaper, Vdd. Thus the mismatch affecting the jump aplitude is not a problem. In FLANN the differences in Δt_{out} are mostly due to mismatch on m_2 which tune the I_{pls} current. To reduce this problem m_2 is three times larger than the minimum size MOSFET.

The digital MUX is a "huge" standard cell that, controlled by the *sel* signal coming from the *configuration element*, selects the incoming signal: if $sel = 0$ the input digital pulse comes from a local neuron, if $sel = 1$ the stimulus is the NAND composition of the X_{aer} and Y_{aer} signals generated by the AER decoders.

4.4.3 Synapse layout

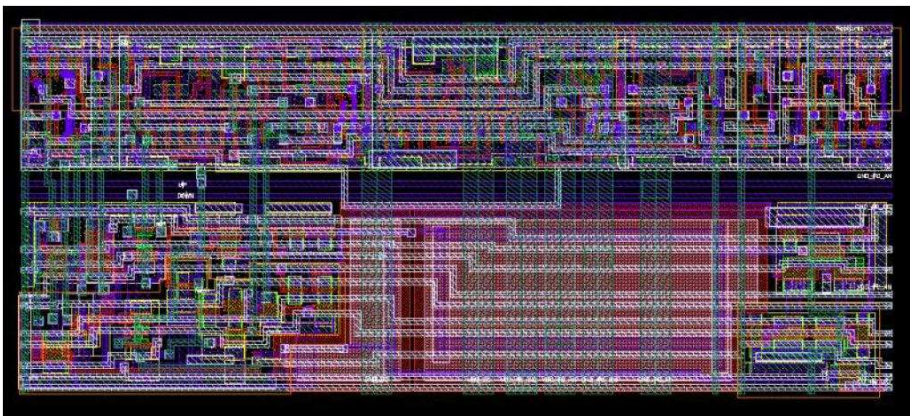


Figure 4.5: Synapse and shaper, layout view. Standard cells are positioned in the upper part, custom cells in the lower part. About the 40% of the area is occupied by standard cells. The recurrent spike signal travels on the white path on the top of the figure, while the "pseudo-digital" V_{up} and V_{dw} signals are brought to the synapse by the two blue paths between the standard and custom circuits. In pink are visible the two capacitances, one for the shaper and one for the synapse. They shield delicate analog MOSFETs, as that in which flows the synaptic refresh current, from those transistors used as switches. The white horizontal paths over the custom parts carry the analog biases.

The layout of the synapse and of the shaper are reported in figure 4.5. The block is 87μm long and 37μm high: $3219\mu m^2$, not much larger than the CLANN one ($3011\mu m^2$). About the 40% of the area is occupied by the standard cells. As in CLANN the analog and digital parts are kept apart, standard cells are in the upper part, while full-custom cells are in the lower part. Moreover the delicate analog transistors of full custom cells, as

those of the refresh circuit, are on the right side of the capacitors, while the one used as switches are on the left side. The white horizontal lines are made of metal two: those passing over the custom cells carry nine analog biases, those on the standard cells carry six digital signals. The node $V_{\text{post}}(t)$ shared by all the synapses belonging to the same dendritic tree is another horizontal net on metal two in the middle of the full-custom section. The standard and full-custom area are separated by a channel on which the pseudo-digital V_{up} and V_{dw} signals travel. The spike from an on-chip neuron comes from the right side on the topmost horizontal white line; this net goes through the matrix and connects with a vertical line on metal four (in green), when it reaches its target column of synapses, in a scheme analogous to that used in CLANN (see figure 3.13). The Y_{aer} signal travels horizontally (it selects a line) on metal two, while X_{aer} is on a vertical green line.

Three couples of power supplies are brought to each synapse, one for the digital standard cell, one for digital custom transistors, one for analog custom MOSFETs. They all arrive both on vertical (metal four) and on horizontal (metal one, in blue, and metal two) large paths such that each of them is part of a network spreading all over the matrix. This makes them low impedance and high capacitance nodes reducing voltage drops and ground bounces. The supply tracks on metal three shield the analog blocks from the local and global digital signals creating a sort of roof for the custom block. The synapse proved to work properly, and no appreciable phenomenons related to noise or cross-talk were experienced during the tests.

4.5 Calcium circuit

As shown in figure 4.2 three elements compose the Calcium block: the spikes impinge into the differential pair integrator (DPI) circuit that generates the calcium variable under the form of a current $I_{\text{ca}}(t)$. This current is compared, according to the theoretical model (eq. 4.1), with the thresholds $I_{\text{up,low}}$ by the comparators system made of a voltage-mode and of a current-mode part. The output of the comparison are two currents which are converted into two voltages in the *Current conveyor* block.

4.5.1 Differential pair integrator

The DPI circuit designed for FLANN is the current-mode low pass filter proposed in [Bartolozzi and Indiveri, 2007]. It acts as a linear RC circuit even if it is composed of MOSFETs working in the subthreshold regime where the voltage-to-current characteristic is exponential. The circuit exactly implements the model described in the first chapter (see eq. 2.28). For the p-type transistor m_{out} we can write:

$$I_{\text{Ca}} = I_0 e^{-\frac{\kappa(V_G - Vdd)}{U_T}} \tag{4.3}$$

where I_0 is the leakage current, κ is the subthreshold slope factor, and U_T is the thermal voltage [Mead, 1989]. The current I_d flowing in the right branch of the differential pair formed by the m_1-m_4 MOSFETs can be expressed as:

$$I_d = I_{\text{in}} \frac{e^{\frac{\kappa V_C}{U_T}}}{e^{\frac{\kappa V_C}{U_T}} + e^{\frac{\kappa V_{\text{thr}}}{U_T}}} \tag{4.4}$$

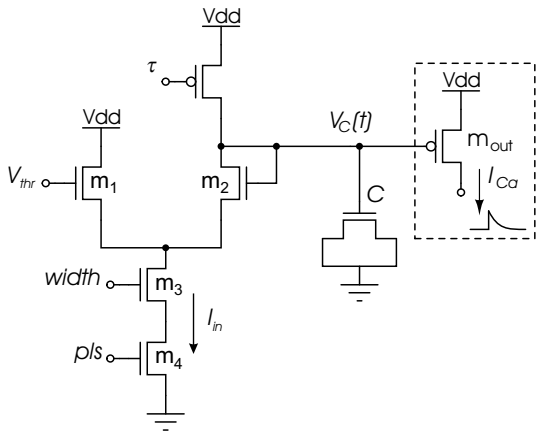


Figure 4.6: Differential Pair Integrator (DPI) circuit configured as a current-mode low-pass filter. The Calcium variable is the current I_{Ca} controlled by the voltage $V_C(t)$ across the capacitor C. τ, V_{thr} and $width$ are three analog bias controlling the behavior of the circuit (see eq. 4.7). The pls signal is the digital input that receives the spike pulse.

assuming that the subthreshold slope factor κ is the same for n- and p-type MOSFETs. The current I_{in} is limited by m_3 and activated during the presence of the spike pulse which is input to the gate terminal of m_4. Multiplying numerator and denominator of equation 4.4 by $e^{-\frac{\kappa V dd}{U_T}}$ and considering equation 4.3, I_d can be rewritten as:

$$I_d = \frac{I_{\text{in}}}{1 + (\frac{I_{\text{out}}}{I_g})} \tag{4.5}$$

where the current $I_g = I_0 e^{-\frac{(\kappa V_g - V dd)}{U_T}}$ represents a virtual p-type subthreshold current that is not generated by any p-FET in the circuit. We can now differentiate the equation 4.3 with respect to V_C and combine it with the equation for the capacitor $C\frac{d}{dt}V_C = -(I_d - I_\tau)$ to obtain:

$$\tau \frac{d}{dt} I_{\text{out}} = -I_{\text{out}}(1 - \frac{I_d}{I_\tau}) \tag{4.6}$$

where $\tau = \frac{C U_T}{\kappa I_\tau}$. Replacing I_d from equation 4.5 into equation 4.6 we obtain:

$$\tau \frac{d}{dt} I_{\text{out}} + I_{\text{out}} = I_{\text{in}} \frac{I_{\text{out}}/I_\tau}{1 + (I_{\text{out}}/I_g)}. \tag{4.7}$$

This is a first-order non-linear differential equation. Its steady state can be easily derived however:

$$I_{\text{out}} = \frac{I_g}{I_\tau}(I_{\text{in}} - I_\tau). \tag{4.8}$$

If the DC component of the input signal I_{in} is much grater than I_τ, then $I_{\text{out}} \gg I_g$ and in this conditions equation 4.7 reduces to

$$\tau \frac{d}{dt} I_{\text{out}} + I_{\text{out}} = I_{\text{in}} \frac{I_g}{I_\tau} \tag{4.9}$$

In the Laplace domain the DPI transfer function is therefore:

$$\frac{I_{\text{out}}}{I_{\text{in}}} = \frac{I_g}{I_\tau}\frac{1}{1+\tau s} \tag{4.10}$$

which is the transfer function of a linear filter with a tunable gain I_g/I_τ.

This circuit, derived from the classical *log-domain* integrator described by Frey [Frey, 2000], has been proposed and studied in [Bartolozzi and Indiveri, 2007]. The mean value of the output current is proportional to the mean firing rate of the neuron. This is the advantage of using this circuit in spite of the one in CLANN. With a linear low-pass filter it is possible to have a measure of the recent mean activity of the corresponding neuron, and this is what we need. The typical behavior of the calcium current can be seen in figure 4.3.

4.5.2 Comparators

Circuits described in this section, first proposed in [Indiveri and Fusi, 2007], produce the I_{LTP} and I_{LTD} signals that once converted in voltages back-propagate to all the synapses belonging to the same dendritic tree.

The Calcium current I_{Ca} and the neuron potential V_{post} are in input to the comparators system composed of three winner-take-all (WTA) blocks and of a voltage comparator implemented with an open loop transconductance amplifier. In figure 4.7 the schematic view is reported. The voltage-mode comparison is between the postsynaptic membrane potential $V_{post}(t)$ and the corresponding threshold θ_V. The result enables the right or the left branch of the current comparators. Each WTA block of the current comparators, has in input the current I_{ca} and one of three current thresholds I_{TH1}, I_{TH2} or I_{TH3}: The threshold $I_{\text{up}}^{\text{low}}$ and $I_{\text{up}}^{\text{low}}$ of equation 4.1 are fixed to equal values, a typical choice, and both correspond to I_{TH1}; $I_{\text{TH2}} = I_{\text{dw}}^{\text{high}}$ and $I_{\text{TH3}} = I_{\text{up}}^{\text{high}}$. The output are the two currents I_{LTD} and I_{LTP} that converted in voltages, are broadcast to the synapses.

The WTA circuit used in FLANN is the one originally proposed in [Lazzaro et al., 1988] and reported in figure 4.8. Two current conveyors $m_1 - m_2$ and $m_3 - m_4$ receive two input currents I_{in1} and I_{in2} and compete for the bias current I_{bias} via the common node V_c. If the two input currents are equal the bias current is equally split between the two branches and the two output currents are equal. When one of the two input currents, increases with respect to the other, the two current conveyors begin to compete and the node receiving the highest input suppresses the other. The output current of the winning cell, in the final state, will equal I_{bias} while the output current of the loosing cell will be zero. The output of the WTA can be then read as a binary variable encoding the output of the comparison between I_{in1} and I_{in2}.

The behavior of the WTA can be analyzed in two different regimes: when one input is much greater than the other, or when the inputs differ of a small amount. In what follows we will consider only the static response of the circuit; an extensive discussion of the circuit static and dynamics can be found in [Bartolozzi, 2007]. The current of a MOSFET can be divided into a *forward* component, I_f, and a *reverse* component, I_r: when the transistor source voltage V_s is approximately equal to its drain voltage V_d, I_r becomes comparable to I_f [Liu et al., 2002]. Let consider the case in which $I_{\text{in1}} \gg I_{\text{in2}}$. If m_1 is in saturation ($V_{d1} > 4U_T$), the dominant component of its drain current will be in the *forward* direction and its gate voltage V_c will increase such that

$$I_{\text{in1}} = I_{f1} = I_0 e^{\kappa \frac{V_c}{U_T}}. \tag{4.11}$$

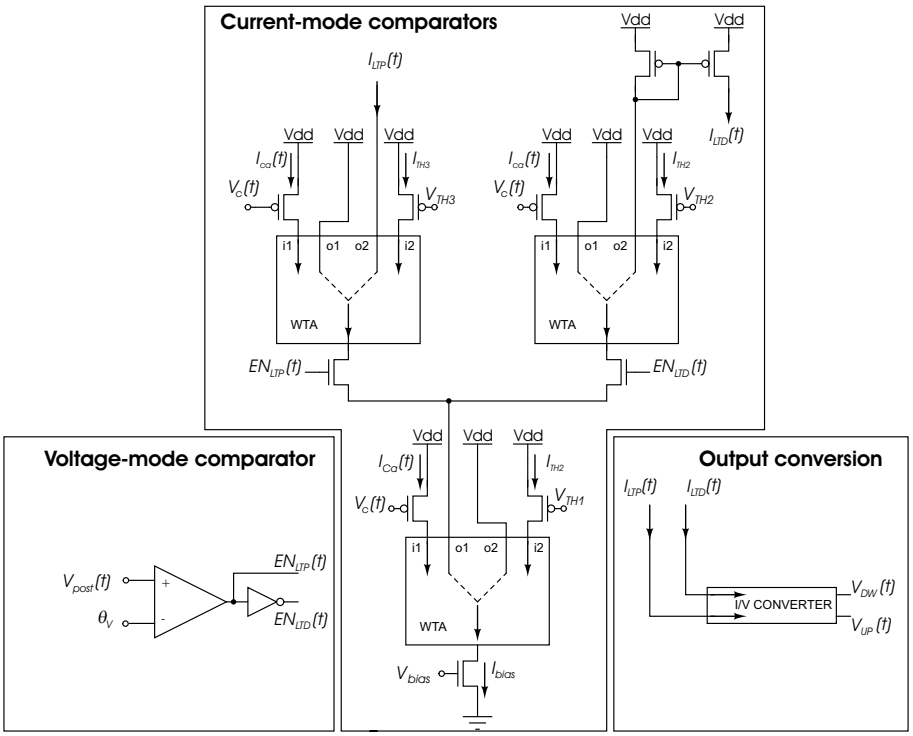


Figure 4.7: Comparators system schematics: on the left an open-loop transconductance amplifier compares the neuron potential V_{post} with the analog threshold Θ_V. The output are two opposite signals which enable either the left or the right branch of the current-mode system based on WTA (*Winner-Take-All*) circuits. This system compares the Calcium current I_{Ca} with three threshold currents I_{TH1}, I_{TH2} and I_{TH3} set by three corresponding analog biases V_{TH1}, V_{TH2} and V_{TH3}. The bias current I_{bias} for the WTA circuits is tuned by V_{bias} and set the maximum amplitude of I_{LTD} and I_{LTP}. The output current-to-voltage conversion is performed by current conveyors shown in figure 4.9.

Although the two input currents are different, the two *forward* component of m_1 and m_2 are equal because the two transistors share the same gate voltage V_c and both their sources are tied to ground, hence $I_{f1} = I_{f2}$. The drain current I_{d2} of m_2 is forced to equal $I_{\text{in}2}$ so that:

$$I_{f1} - I_{r2} = I_{\text{in}2} \tag{4.12}$$

which, combined with equation 4.11, implies that

$$I_{r2} = I_{f1} - I_{\text{in}2} = I_{\text{in}1} - I_{\text{in}2} \gg 0. \tag{4.13}$$

This means that the reverse component of I_{d2} is significant, which is possible only if m_2 operate in its ohmic region, i.e. $V_{d2} \leq 4U_T$. In this case, the output transistor m_4 is effectively switched off, and $I_{\text{out}2} = 0$. Consequently, m_3 sources all the bias current and $I_{\text{out}1} = I_{\text{bias}}$.

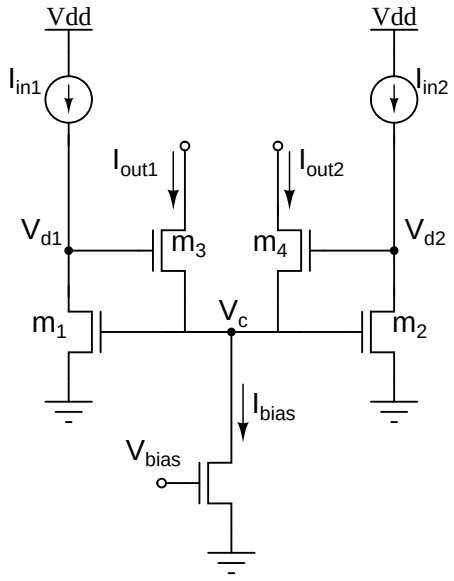


Figure 4.8: Winner-Take-All (WTA), schematic view. I_{in1} and I_{in2} are two input currents for two coupled current-conveyor $m_1 - m_3$ and $m_2 - m_4$ competing for the bias I_{bias} tuned by the voltage V_{bias}. The dynamics of the circuit is such that if $I_{in1} > I_{in2}$ then $I_{out1} = I_{bias}$ and $I_{out2} = 0$ otherwise if $I_{in1} < I_{in2}$ then $I_{out1} = 0$ and $I_{out2} = I_{bias}$.

In the other case, when the two inputs differ by a small amount, we have to consider the Early effect of the transistor operating in the saturation regime [Liu et al., 2002]:

$$I_{ds} = I_{\text{sat}}(1 + \frac{V_{ds}}{V_e}) \tag{4.14}$$

where V_e is the Early voltage. Assume the initial condition $I_{\text{in1}} = I_{\text{in2}}$: the two transistors m_1 and m_2 will operate in the saturation regime and the output currents will both be equal to $I_{\text{bias}}/2$. If now one of the input current increases of a small amount δ_I, applying eq. 4.14 to M_1, then the drain voltage V_{d1} will increase by

$$\delta_V = \frac{\delta_I}{I_{\text{sat}}} V_e. \tag{4.15}$$

As V_{d1} is also the gate voltage of m_3, I_{out1} will be amplified by an amount proportional to e^{δ_V}. Consequently I_{out2} decreases by the same amount implying a reduction of δ_V in the drain voltage of m_4.

So, starting from two equals input currents and hence two equal output currents, if one of the two input increases, the current of the losing branch initially decreases due to the Early effect, then, for increasing difference, the input MOSFET of the losing part is brought out of its saturation regime and the corresponding output transistor is shut off. Thus the output current of the losing branch goes to zero, while the one of the winning branch equals I_{bias}.

4.5.3 Current conveyors

The two currents $I_{\rm LTP}$ and $I_{\rm LTD}$ are reported in voltages through two independent current conveyors (see figure 4.9) of the type used in the WTA circuit.

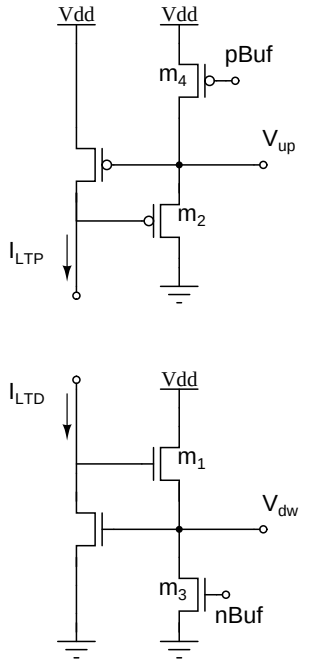
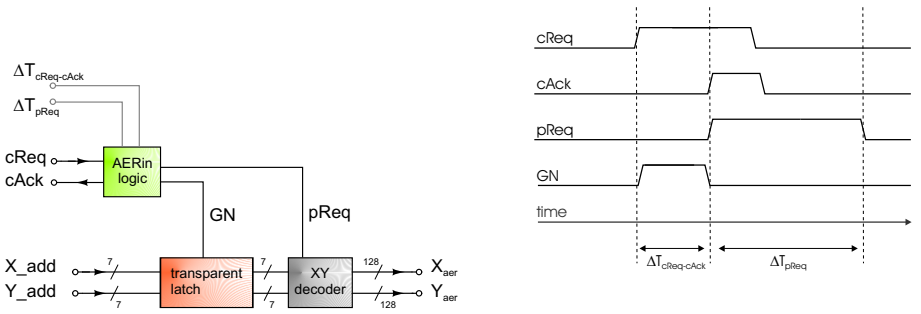


Figure 4.9: Current conveyor schematics. I_{LTP} and I_{LTD} are two input currents, V_{up} and V_{dw} two output voltages, $nBuf$ and $pBuf$ two analog biases.

In this configuration the circuit decouples the output voltage from the input one. If the input current $I_{\rm LTD}$ ($I_{\rm LTP}$) assumes the value of $I_{\rm bias}$ than the output voltage for the n-type current conveyor will be fixed at $V_{dw} = \frac{U_T}{\kappa} \log \frac{I_{\rm bias}}{I_0}$ ($V_{up} = \frac{1}{\kappa}(Vdd - U_T \log \frac{I_{\rm bias}}{I_0})$. If the input current is null, than the input transistor of the current conveyor will not be in saturation anymore, i.e $V_{ds} < 4U_T$; m_1 (m_2) would be shut off and the output voltages would be attracted to the power supply by m_3 (m_4). The logarithm transduction generates two digital signals whose dynamic range is less than the 3.3V given by the power supply and is determined by the bias current of the WTA. The voltages $V_{\rm up}$ and $V_{\rm dw}$ are common to all the synapses belonging to the same dendritic tree.

4.6 New AER input circuit

Testing CLANN we faced some problems due to its AER input system. Under certain conditions the communication stop until all the AER signals involved in the handshake were forced to reset from outside. We completely redesigned the circuits streamlining the handshake for multi-chip architectures. Both the decoders and the circuit handling the handshake phase (the AERin logic) have been implemented using standard cells. The relevant new features



(a) (b)

Figure 4.10: (a) AERin complete system. It consists of 1) a logic block (AERin logic) that handles the four-phase handshake, 2) of an array of transparent latches for incoming address-events and 3) of an XY decoder. $\Delta T_{cReq-cAck}$ and ΔT_{pReq} are two tunable parameters. (b) Time courses of involved signals are shown.

are essentially two: 1) the chip-acknowledge signal (*cAck*) is released soon after the request signal goes down [Dante et al., 2005], without waiting any pixel-acknowledge from the target synapse as happens in CLANN, and 2) the pixel-acknowledge signal path back from the target synapse to the AERin logic block has been removed. Two tunable delays ensure that the handshake phase, the AER address decoding and the stimulation of the synapses are successfully performed. Essentially two circuital components have been added: an array of 14 latches to store the AER address data and a cell to delay the rising edge of a digital pulse. The various blocks are organized as show in figure 4.10a. A zoom on the AERin logic block is reported on the left side of figure 4.11, while on the right side of the same figure the schematic view of the *rising delay* cell is shown.

The protocol implemented works as follows (see figure 4.10b):

1. An external device requests to communicate driving the chip request signal *cReq* high, and pushing the address lines, directly connected to the latch buffers, to the desired levels.

2. FLANN answers bringing high the chip-acknowledge signal *cAck* after a delay $\Delta T_{\text{req-ack}}$ which gives time to the data lines to stabilize to the correct values and allows to store the address bits on the latch buffers. When the *cAck* goes high the input of the latches is closed (signal *GN* goes down) and their outputs will remain stable till the next transaction.

3. The external device sees a rising edge on the acknowledge line and replies removing the request.

4. FLANN immediately releases the *cAck* and triggers a pulse on the pixel-request *pReq* line lasting ΔT_{pReq} which enables the address decoding. At this point, even if the address, stored in the latch has not been decoded yet, the bus is free and can be used to deliver a new AER event to a different chip. ΔT_{pReq} is adjusted so that the outputs of the decoder, the X_{aer} Y_{aer} pulses, have time to propagate and to correctly stimulate

the target synapse. To avoid wrong decoding, a new request signal is processed only after the end of the *pReq* digital pulse.

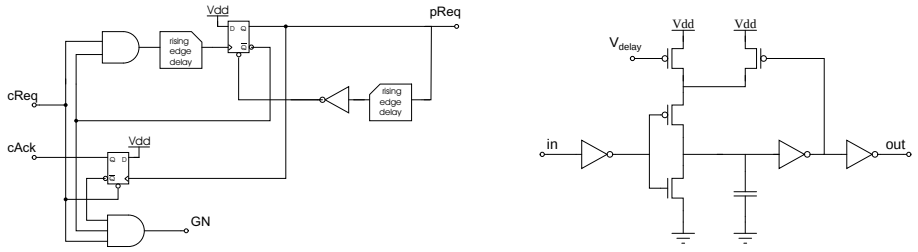


Figure 4.11: (a) Schematic view of the AERin logic circuit. It consists of two D-Type flip-flops, two AND and one NOT gates, and two *rising edge delay* custom cell. The circuit is designed to have two loops, one controlling the $cReq - cAck$ delay, and one for the *pReq* pulse duration. Each loop comprises one *rising edge delay* cell (b) tunable via the V_{delay} bias.

Adjusting the two delays $\Delta T_{\text{req-ack}}$ and ΔT_{pReq} to the right values, the AER communications system works fine, without the annoying problem faced during CLANN tests.

4.7 Preliminary characterization tests: synaptic efficacy

In this section I present preliminary results obtained from characterization tests performed on FLANN. Such tests aim to measure the synaptic efficacy under various working conditions estimating the mean values of this relevant parameter as well as its relative dispersions due to circuits mismatch and inhomogeneities in analog biases. Looking forward to create multi-chip architectures, I present data accumulated testing different FLANN chips.

Consider a system composed of a pre-synaptic neuron, an excitatory synapse and a post-synaptic neuron. Upon the arrival of a pre-synaptic spike, the excitatory synapse injects a current I_{syn} (see eq. 2.23) into the post-synaptic neuron capacitor. The amplitude of this current depends on the synaptic state: it will be small (I^{-}_{syn}) if the synapse is depressed, larger (I^{+}_{syn}) if the synapse is potentiated. This current remains active for the entire duration of the spike pulse, duration set by the pulse shaper circuit (see section 4.4.2). The post-synaptic neuron receives the synaptic current and consequently its potential $V(t)$ undergoes an upward jump. The synaptic efficacy is the amplitude of this jump and will be measured as a fraction of the subthreshold dynamic range of $V(t)$ i.e. as a fraction of the value $\Theta - V_r$, where Θ is the firing threshold, and V_r the reset potential (in our case $V_r = V_{min}$, see equation 2.20). A simple method to measure the synaptic efficacy consists in counting the number of jumps that $V(t)$ undergoes before a spike is emitted. The protocol I adopted comprises the following steps:

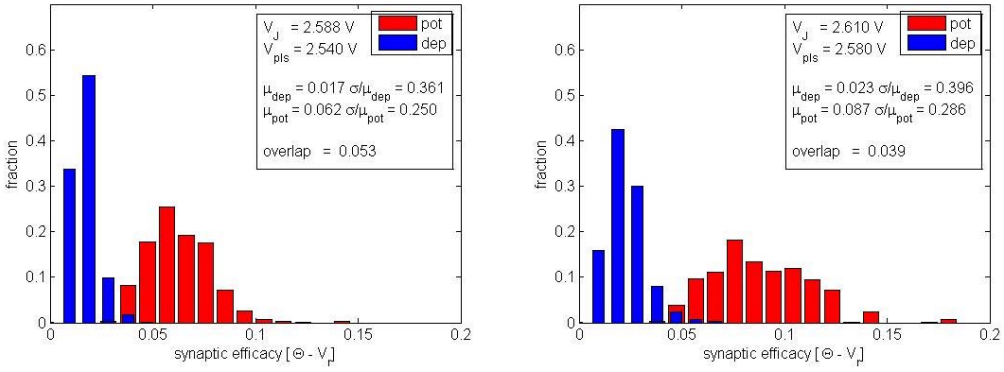
1. the considered synapse is set depressed (potentiated) and the internal synaptic dynamics is turned off;

2. the parameters of the post-synaptic neuron are tuned such that the constant leakage current equals the constant afferent current; in this way they do not induce or inhibit any firing activity; (see Fig. 3.14)

3. a regular AER spike train at 1KHz is sent to the post-synaptic neuron via the synapse to be characterized; this stimulation lasts 1 second;

4. during the stimulation, the firing activity of the post-synaptic neuron is monitored;

5. the synaptic efficacy is evaluated as ν_{post}/ν_{pre}, where $\nu_{pre} = 1$KHz is the AER spike train frequency and ν_{post} is the mean firing rate of the post-synaptic neuron.



(a) (b)

Figure 4.12: Synaptic efficacy distributions over 128 synapses. The blue histograms are obtained when synapses are set depressed, the red ones when synapses are set potentiated. Blue histograms partially overlap the red ones. The level of overlap could be seen as an estimate of the signal-to-noise ratio on the dendritic tree. Left and right panels refer to two different choices of chip biases.

The second point of this protocol is an annoying fine tuning step that could be bypassed implementing a more sophisticated method compensating for the neuron "spontaneous" activity. Anyhow the chosen method guarantees relative errors inferior to 5%, a value compatible with the aims of these measures. I repeated the test for all the 128 synapses belonging to the dendritic tree of a neuron, results are shown in Fig. 4.12 composed of two panels corresponding to two different choices of chip biases. Blue histograms refer to the synaptic efficacy of the 128 synapses all set to be depressed, red histograms refer to the same 128 synapses all set to be potentiated.

Consider separately blue and red histograms: from such plots the mean values and the standard deviations can be evaluated (see Fig. 4.12) and used for instance to reproduce the network behavior with a numerical simulation as done for CLANN (see section 3.6). Furthermore, these histograms are benchmarks for our future VLSI systems. CADENCE, the software we use to design chips, provides a powerful tool for Monte Carlo simulations to run on analog circuits schematics; adjusting the simulation parameters as the temperature or the correlation level among MOSFETs, it will be possible to fit histograms obtained from experimental data, to modify the circuits and check the improvements of new designs.

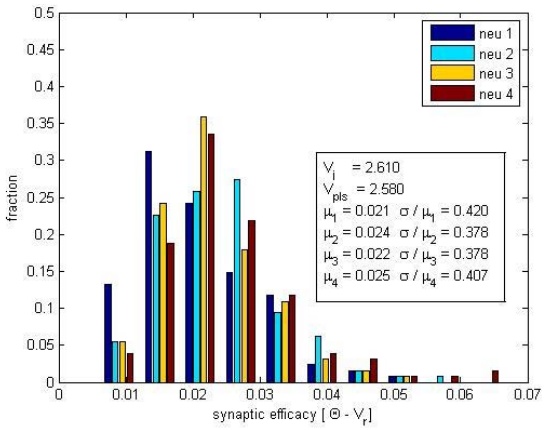


Figure 4.13: Four distributions of the synaptic efficacy measured on depressed synapses. Each distribution accounts for 128 synapses connected to a different neuron. The distributions are qualitatively similar. 128 synapses are a good statistical sample to represent $4 \cdot 128$ synapses.

In figure 4.12, left or right panel, blue histograms overlap the red ones. The level of overlap depends on the distance between the mean values of the two distributions. Unfortunately these mean values are not completely free to move, they should stay within a certain range typically determined by the network architecture and by experimental requirements. For instance it could be not desirable to have synapses with a zero or a huge efficacy; such constraints together with the width of the distributions define the limits for the range of the mean values. If the two histograms had been completely overlapped, learning would have been impossible: indeed, for the post-synaptic neuron there would be no difference, statistically, between a potentiated or a depressed synapse. The situation of figure 4.12 is not so dramatic, but we still have some synapses whose depressed efficacy is equal or even larger than the efficacy of some other potentiated synapses. Mismatch generates a certain kind of noise in the synaptic communication; the level of overlap of the two distributions is somehow a measure of the signal-to-noise ratio for the dendritic tree.

Looking forward to configure FLANN as a homogeneous recurrent network, a relevant parameter to know is the value of the mean efficacy calculated over all the active synapses, a value to use, for instance, in mean-field equations or numerical simulations. In the next paragraphs I discuss measures concerning 512 synapses belonging to dendritic trees of four different neurons. FLANN neurons are arranged in a vertical array as shown in figure 4.1, those considered are placed one at the top, one at the bottom and two in the middle of the array. In this way the 512 synapses of the four dendritic trees are placed in different regions of the silicon surface.

In figure 4.13 are shown four different histograms, each accounting for the synaptic efficacy distribution of synapses connected to one of the four neurons, for a typical choice of chip biases. The four distributions are incredibly similar. The blue one is shifted a little bit on the left, but this is due to a non perfect fine tuning of the neuron parameters as prescribed by the second point of the test protocol described above. One possible reason behind this similarity is that 128 synapses are a good statistical sample for the set of 512 synapses, i.e all the effects of the mismatch are already evident in a subset of 128 elements. The four

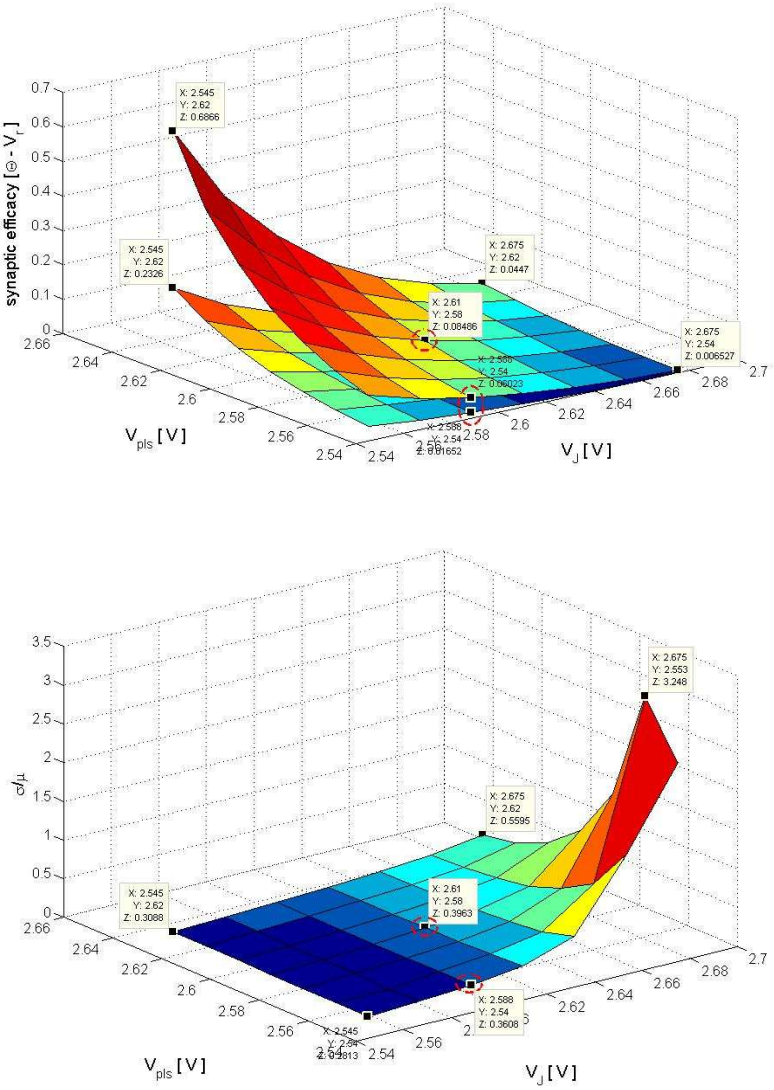


Figure 4.14: Synaptic efficacy surfaces, mean values and relative standard deviation. *Upper panel*: lower graph reports mean values (μ) of the synaptic efficacy over 512 depressed synapses. The upper surface shows mean efficacy values for potentiated synapses. The amplitude of the synaptic efficacy depends on two biases V_{pls} (on the x-axis) and V_J (on the y-axis). These parameters control respectively the duration of the spike pulse (ΔT_{spk}) and the amplitude of the synaptic current I_{syn}. In the *Lower panel* corresponding values of the relative standard deviation σ/μ are reported. The surface accounts only for the case of depressed synapses. Highlighted points correspond to data reported in figure 4.12.

subgroups of synapses have similar statistical characteristics, they form a homogeneous set that will be considered as a whole from now on, discarding the information on which synapse

is connected to which neuron. At this point it is reasonable also to affirm that 512 synapses are representative of all the 16384 synapses in FLANN.

The synaptic efficacy depends on the amount of charge injected in the neuron capacitor; this charge is the product of the amplitude of the synaptic current (I_{syn}) times the spike pulse duration (ΔT_{spk}). FLANN is endowed with three independent analog biases to tune the synaptic efficacy (see Fig. 4.4): V_{pls} sets ΔT_{spk}; V_J adjusts I^-_{syn}; and $V_{\Delta J}$ controls a current $I_{\Delta J}$ that activates when a synapse potentiates so that the current injected into the neuron becomes $I^+_{syn} = I^-_{syn} + I_{\Delta J}$. In these tests $V_{\Delta J} = V_J$ so that, in theory, the efficacy of potentiated synapses should double the efficacy of depressed synapses.

I performed the measurement of the synaptic efficacy for all the 512 synapses. For each synapse, I repeated the test 49 times, each time choosing a different couple of values for V_{pls} and V_J. For each of the 49 points of the (V_{pls},V_J) plane, I calculated the mean efficacy value (μ) over the 512 synapses, and the corresponding relative standard deviation σ/μ. Results are reported in figure 4.14.

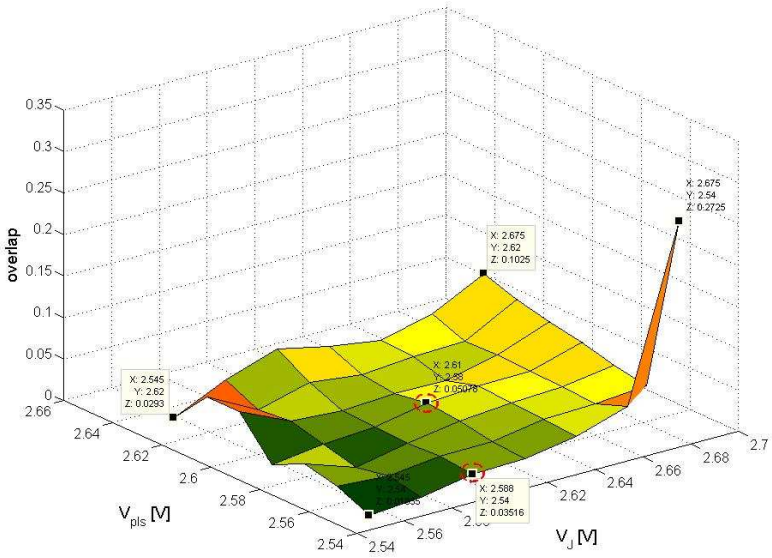


Figure 4.15: Level of overlap between synaptic efficacy distributions obtained from potentiated and depressed synapses. Considering the left side of the surface the overlap spans within 2-5 percent. Highlighted points correspond to the data reported in figure 4.12

Unfortunately on the x and y axis I could report only the values in volt of the biases, and not simply ΔT_{spk} and I_{syn} that are not directly accessible and would require dedicated characterization tests. To read the graph is sufficient to know that, as expected, the mean values of the synaptic efficacy is maximum when ΔT_{spk} and I_{syn} are at their maximum values (on the left side of the graph); the synaptic efficacy is at its minimum (right side of the graph) when ΔT_{spk} and I_{syn} assume their lowest values[1]. From the lower panel of figure 4.14 one can see that the dependence of σ/μ on I_{syn} variation is stronger than on ΔT_{spk}

[1] ΔT_{spk} increases with V_{pls} and varies in the range of few microseconds (from a rough estimation, its mean values over the 512 synapses are between 2 and 15 μs). I_{syn} is in the range of few nA and increases when V_J decreases (see figure 4.4)

variation. The minimum of σ/μ is in the corner next to the reader, i.e. at $V_{pls} = V_J = 2.54V$. These are the values for which both the currents flowing in the MOSFETs (m_{pls} and m_{syn}) controlled by V_{pls} and V_J reach a maximum. The σ/μ graph suggests to use small ΔT_{spk} and high amplitudes of I_{syn}: the inferior limit of ΔT_{spk} and the upper limit of I_{syn} should be chosen considering the amount of noise that the activation or deactivation of the currents flowing in m_{pls} and m_{syn} generates in contiguous circuits.

Our next chip would be probably endowed with the configuration ability described in [Mitra et al., 2006]: the ability to discard some neurons and connect their synapses to the dendritic trees of other neurons. Thus it would be possible, for instance, to have one neuron connected to 512 synapses. To get an idea of the amount of overlap that for various choices of the biases we could have in this situation, I plotted the surface reported in figure 4.15. For reasonable values of the parameters the overlap ranges from 2 to 5 percent.

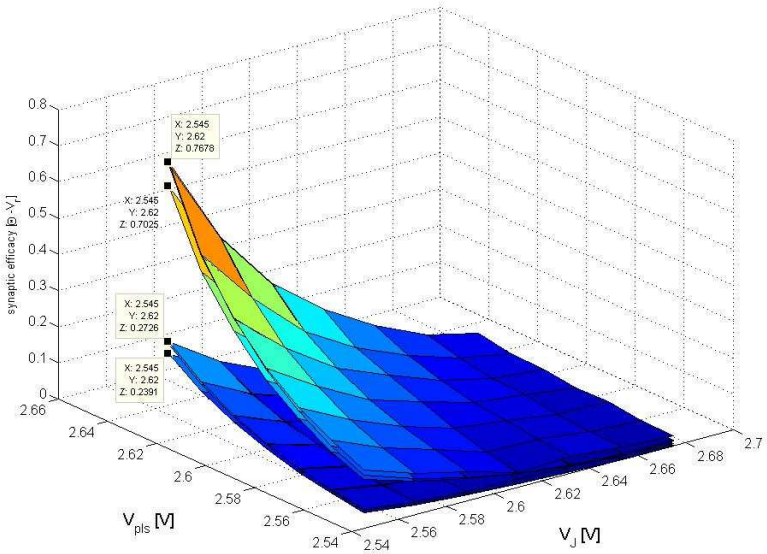
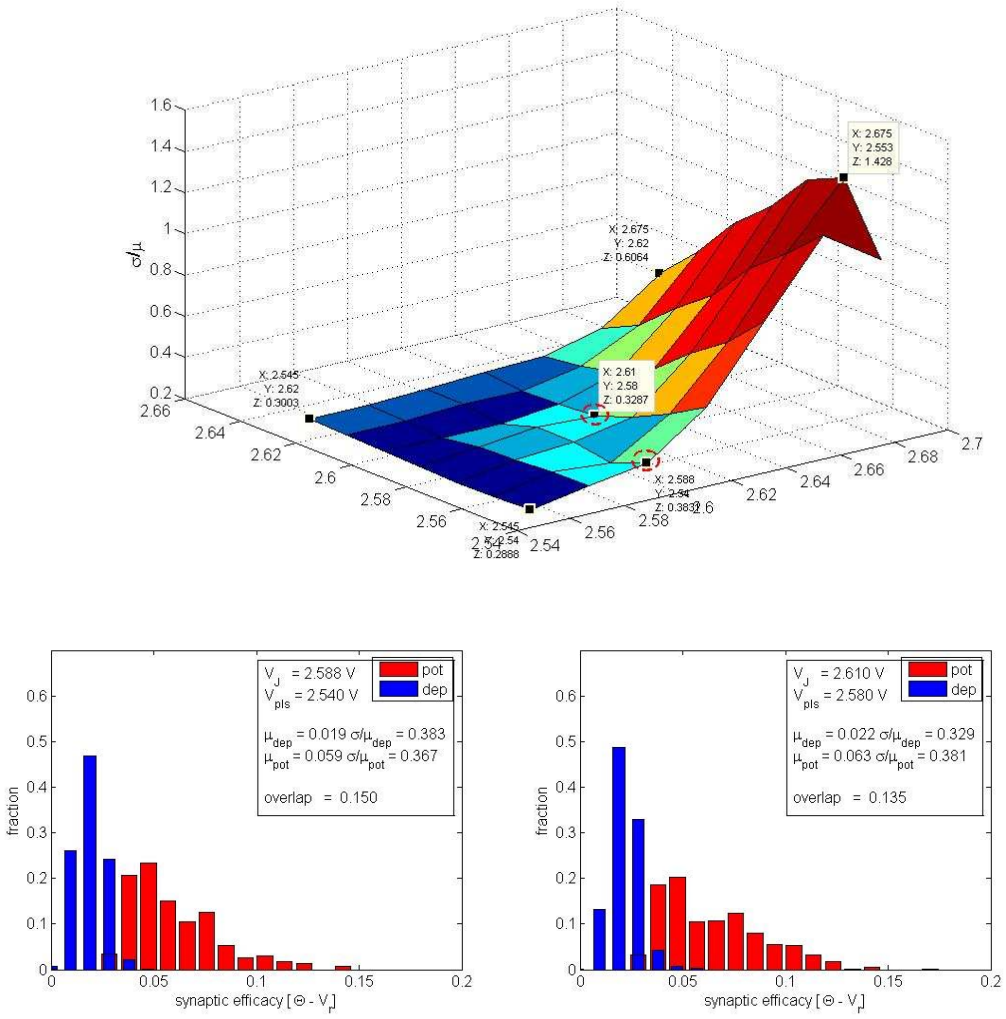


Figure 4.16: Synaptic efficacy mean values for potentiated and depressed. This figure provides a qualitative idea of the situation. Data are obtained from three different chips to be connected in a multi-chip network.

FLANN has been designed to be part of a multi-chip system. Imaging to configure a system of 3 FLANN chips as a recurrent network with a uniform level of connectivity at 30% (i.e. each neuron has a 0.3 probability of being connected to any other neuron in the network), what would be the mean value of the synaptic efficacy over the entire network? What I have done to answer this question was to substitute the FLANN chip on the experimental setup with another FLANN chip, re-run the characterization tests and analyze the results.

To get a qualitative idea of what kind of results I found, I report in figure 4.16 the surfaces accounting for the mean efficacy values measured on three different chips when the synapses are set depressed. Figure 4.16 was encouraging: for each case, potentiated/depressed synapses, only two of the three surfaces are clearly visible, the third is only barely visible in some points. To obtain a system implementing a network as much homogeneous as



(a) (b)

Figure 4.17: Relative standard deviation (σ/μ) of synaptic efficacy distributions obtained from a set of $3 \cdot 512$ synapses placed on 3 different chips. Data refer to depressed synapses. As expected, in comparison to distribution width of synapses on a single chip, the level of dispersion is increased. Considering the left region of the surface it roughly stays between 30 and 50%. Highlighted points correspond to the data reported in the lower part of the figure, data to be compared to histograms in figure 4.12.

possible, one should choose for each chip, a different point in the (V_{pls},V_J) plane, such that the three mean efficacy values would match. This will correspond to an initial tuning phase

of the multi-chip system. Found the various biases for the chips, I plotted the graph in figure 4.17 which reports the values of σ/μ for our "matched" multi-chip architecture. Given the fact that biases do not changes too much from chip to chip it is reasonable to maintain on the x and y axis the values of V_{pls} and V_J of one of the three devices. Comparing figure 4.17 with lower panel of figure 4.14 one can see a slight increase of the dispersions width that, for typical choices of the parameters (left side of the surfaces), roughly increases only of few percentage points. The histograms reported in the lower part of figure 4.17 corresponds to histograms shown in figure 4.12. Relevant parameters to evaluate in this case are the mean values and the dispersion of the synaptic efficacy of potentiated or depressed synapses considered separately. Given the fact that the synapses do not belong to the same dendritic tree and that they are placed on different chips, the level of overlap looses a part of the sense that it has for graphs in figure 4.12. Anyhow it remains a useful parameter to evaluate the dispersions characteristics.

The preliminary results reported in this section represent useful benchmarks for the design of future chips and are also encouraging looking forward to the realization of controllable multi-chip systems.

4.8 Conclusions

The chip I presented in this chapter is a semi-*neuromorphic* implementation of a reconfigurable network of 128 integrate-and-fire neurons and 128x128 Hebbian plastic bistable synapses. FLANN is the big brother of CLANN; it has been designed starting from the circuits already tested in the previous chip and it has been endowed with a series of technical improvements suggested by the experience accumulated designing and testing CLANN.

Improvements range from circuital to architectural modifications. The new current-mode calcium circuit is capable of an exponential decay for the Calcium variable [Bartolozzi and Indiveri, 2007] [Bartolozzi, 2007] [Indiveri and Fusi, 2007] so that it exactly reproduces the theoretical model described in eq. 2.28. The new shaper circuit is a more compact one; the pads have been modified to reduce voltage drops on power lines and noise production at the input digital ports; the first column of synapses can now be tuned via dedicated parameters, a useful feature when a teaching signal is required. A new *initialization circuit* (see fig. 4.4) allows to set the initial state of the synaptic internal variable while three-state buffers permit to directly monitor the synaptic state evolution, suitable AER protocols are no longer required (see section 3.7). The possibility to set and monitor synaptic states greatly simplify experimenters work especially during network-level tests. A completely new AER input system has been designed and the communication handshake with external devices has been streamlined for multi-chip systems. From an architectural point of view two novelties have been introduced: 1) the new configuration system is based on an X-Y selection of the various synapses and no more on a serial bit-stream as in CLANN and 2) each synapse can be configured as AER or recursive one, a relevant feature for multi-chip systems.

Connecting "simple" building blocks seem to be a reasonable way to create large modular networks in hardware. Various communication strategies are being explored ranging from the standard asynchronous parallel AER systems [Dante et al., 2005], to serial burst-mode versions of the AER protocol [Boahen, 2004a] [Boahen, 2004b] and [Boahen, 2004c], to time-stamped codes for fast serial links [J. Schemmel and Ostendorf, 2007]. Different multi-chip systems have already been successfully tested: silicon retinas and networks of spiking

neurons have been connected [Lin et al., 2006] to implement orientation hypercolumns [T. Y. W. Choi, 2005]; in [U. Mallick and Cauwenberghs, 2005] the authors present a system of four chips for a real-time general-purpose processing platform; a multi-layered AER vision system is described in [et al., 2006] and a multi-chip system to model orientation selectivity is reported in [Chicca et al., 2007]: it combines a *neuromorphic* retina and a winner-take-all chip using the same external digital apparatus [Dante et al., 2005] exploited to test CLANN and FLANN.

FLANN has been though to be a reconfigurable building block for a multi-chip system. Its AER/recursive configuration ability adds a degree of freedom not present in other *neuromorphic* implementations. The communication issue is today the major constraint in designing networks of thousands of neurons. FLANN configurability appears as a relevant feature to help solving the problem: given a suitable network to be mapped on a multi-chip hardware, FLANN allows to choose the adequate fraction of synaptic resources to dedicate to external connections. In this way the desired network can be implemented in hardware reducing the AER workload (exploiting on-chip networks for local connections) and minimizing the waste of precious silicon area (no unused synapses).

Chapter 5

Conclusions

The goal of this PhD work was to implement and control hardware spiking neural networks. My work consisted in designing and testing two *neuromorphic* VLSI systems hosting networks of integrate-and-fire neurons and Hebbian plastic bistable synapses. Both chips were designed and tested in collaboration with the staff members of the Complex Systems Unit of the Italian National Institute of Health in Rome and of the Institute of NeuroInformatics in Zurich. My personal contribution to the work consisted in defining the architectures of the chips and the top-level schematic and layout; I designed and simulated some circuits and drew the layout of some structures. In the test phase I worked on the hardware setup, realized by technical personnel, to characterize the chip. We brought the hardware-software experimental setup to a high level of reliability, not a trivial task for such kind of full-custom devices. In the end a good control of VLSI devices was reached, it was possible to work under repeatable conditions and, hence, to run experiments and accumulate statistics. We performed high level experiments on CLANN and the results demonstrated the chip works in agreement with the theoretical models from which this adventure began; this motivated the design of a larger chip named FLANN endowed with a series of technical improvements that greatly facilitate the test phase. Preliminary results on FLANN proved the chip works properly in all its parts.

These chips are a step forward towards endowing VLSI, *neuromorphic* devices with autonomous learning capabilities adequate for not too simple statistics of the stimuli to be learnt.

Learning chips implementing models of spike-based synaptic plasticity as been proposed in [Indiveri et al., 2006] [Arthur and Boahen, 2006] [Riis and Hafliger, 2007] [Petit and Murray, 2003]. The main novel features of the chips presented in this work are the implemented type of synaptic plasticity and the configurability of the synaptic connectivity. Experiments performed on CLANN (see chapter 3) are meant primarily to demonstrate the first, while a dedicated setup suited to host multiple chips is being completed and will be the stage for fully illustrating the flexibility offered by the configuration features. For the CLANN experiments we take advantage of the configuration features to choose a simple perceptron architecture, and illustrate the chip working and the advantages of the stop-learning plasticity rule when highly correlated patterns are to be learnt. A VLSI system with a similar synaptic plasticity has been described in [Mitra et al., 2006] and relative experimental results are reported in [Mitra et al., 2007].

CLANN and FLANN are interesting devices in the perspective of *neuromorphic* chips

as building blocks of parallel, distributed systems with non-trivial computational abilities. Indeed, in [Brader et al., 2007] a perceptron trained with the same learning rule was shown in simulation to be able to correctly classify noise-corrupted LateX character, with performances comparable to state-of-the-art methods. Though much larger than the one implemented in FLANN, the network used in [Brader et al., 2007] is within reach of a multi-chip system of affordable complexity, foreseen in the near future as an improved version of the one presently under development. Distribute multi-chip networks of various kind have already been successfully tested: silicon retinas and networks of spiking neurons have been connected [Lin et al., 2006] to implement orientation hypercolumns [T. Y. W. Choi, 2005]; in [U. Mallick and Cauwenberghs, 2005] the authors present a system of four chips for a real-time general-purpose processing platform; a multi-layered AER vision system is described in [et al., 2006] and a two-chips system to model orientation selectivity is reported in [Chicca et al., 2007].

The value of the high degree of flexibility of CLANN and FLANN in the definition of the synaptic connectivity will be best appreciated when such chips will be embedded as components of complex, multi-chip architectures. Different architectures, such as the implementation of a set of visual receptive fields or a homogeneous recurrent network, can impose very different constraints on the relative number of excitatory and inhibitory synapses and on the relative weight of recurrent *vs* external input for each chip. On this aspect FLANN architecture represents an improvement in comparison with CLANN. In FLANN each synapse can be configured as AER or recursive, thus it is possible to choose the fraction of synaptic resources to dedicate to external inputs. Considering for instance a system of four FLANN chips, to have a homogeneous network of 512 neurons with a uniform level of connectivity at 25%, 3/4 of the synapses in each chip should be configured as AER. The remaining synapses will form local connections. In this case 1/4 of the spike traffic remains inside the chips and does not weights on the external communication system. This fraction can greatly increase when the network to implement has structured topology suitable for modular hardware systems. The AER/recursive configurability helps reducing the AER workload and at the same time minimize the number of unused synapses saving precious silicon area. Other recently proposed architectures offer alternative approaches to flexible synaptic structure, differently trading off complexity of design and load on the AER bus [Liu and Douglas, 2004] [Mitra et al., 2006] [Lin et al., 2006] [Chicca et al., 2007].

These VLSI networks are small, unfortunately not in term of Silicon occupancy, but in term of the number of neurons and synapses they host. CLANN is composed of 32 neurons and 2048 synapses, FLANN of 128 neurons and 16384 synapses. Such numbers are small not only compared to biological networks with identified computational role [Pakkenberg et al., 2003] [Williams and Herrup, 1988], but also in view of applications. *Neuromorphic* VLSI will have to scale up significantly to face not only simplified and stereotyped tasks as those proposed in our experiments, but also real-world problems.

Multi-chip architectures seem to be a reasonable step to take to reach large numbers of neurons and synapses. Even considering the best technology nowadays available, the number of transistors in a chip would not allows to design networks comparable to 1 mm^3 of the cerebral cortex which contains about 10^5 neurons. In these days the Intel is launching its new series of microprocessors designed in 45nm technology, hosting 800 millions of transistors. Both CLANN and FLANN have been designed in 0,35μm standard CMOS technology. Each neurons consists of about 30 MOSFETs and each synapse needs, redesigning the digital part, at least 50 transistors. Thus, in a rough estimate, with 800 millions of transistors we

could design a network of 8000 neurons uniformly connected at about the 20%. A multi-chip architecture, maybe a 3D stack of these chips, would constitute a non-trivial real-time device.

Despite the small dimensions of hardware neural networks there are some reasons for working with them. These small networks are good candidates to provide primitives for a future biologically inspired computational paradigm. Designing this kind of asynchronous hybrid analog-digital devices means to test new analog circuits as well as to research solutions to a number of technical problems ranging from cross-talk effects between analog and digital lines, to unavoidable circuits imperfections, to the issue of a massive asynchronous parallel communication. Constraints that naturally come with real systems and that do not appear as tight limits in the abstract world of analytical formulations or of numerical simulations. Challenges we should meet if we expect to have, in a future, working and reliable pieces of bio-mimetic hardware.

A number of commercial applications, loosely inspired to neural networks approach, has already proven to be effective: applications span from credit card fraud detection to railway maintenance, from space robot self-tuning, to pattern recognition [Polycarpou, 1997]. Advances in potential of this kind of applications is expected when a closer match between commercial devices and theoretical models will be reached. Different techniques and methods are exploited to implement neural networks in hardware: it is possible to configure an FPGA (Field-Programmable Gate Array), to program a DSP (Digital Signal Processing), to design a digital ASIC (Application Specific Integrated Circuits) or a mixed synchronous/asynchronous digital system, to use floating gates, CMOS technology or hybrid analog-digital *neuromorphic* circuits [Polycarpou, 2003]. This last possibility is our choice. The aims of *neuromorphic* engineering go beyond the production of hardware neural networks. *Neuromorphic* VLSI was born from the idea of designing biologically inspired circuits that can represent a useful research strategy, inspire new models [Amit and Fusi, 1994], be an engine for the theoretical research and a way of implementing stand-alone power-efficient (useful) devices [Mead, 1990]. The present VLSI networks are just toys to experiment new solutions and ideas, and they demonstrated to be functional to this goal. But even if we had a huge VLSI system with all the desired neurons and synapses there is not, today, a theory able to exploit the dynamics of the network to perform computation if not in a limited number of simplified situations. On the other side, waiting for a complete comprehension of the nervous system before beginning to design *neuromorphic* systems, would be senseless. Weather or not the theory, and the *neuromorphic* VLSI will reach their maturity age, this you should ask in a bit.

How Much Shall We Bet?

From Cosmicomics, by Italo Calvino.

Yes, but at the beginning nobody knew it, -Qfwfq explained- I mean, you could foretell it perhaps, but instinctively, by ear, guessing. I don't want to boast, but from the start I was willing to bet that there was going to be a universe, and I hit the nail on the head; on the question of its nature, too, I won plenty of bets, with old Dean (k)yK. When we started betting there wasn't anything yet that might lead you to foresee anything, except for a few particles spinning around, some electrons scattered here and there at random, and protons all more or less on their own. I started feeling a bit strange, as if there was going to be a change of weather (in fact, it had grown slightly cold), and so I said: "You want to bet we're

heading for atoms today?" And Dean (k)yK said: "Oh, cut it out. Atoms! Nothing of the sort, and I'll bet you anything you say." [...]

We were always betting, the Dean and I, because there was really nothing else to do, and also because the only proof I existed was that I bet with him, and the only proof he existed was that he bet with me. We bet on what events would or would not take place; the choice was virtually unlimited, because up till then absolutely nothing had happened. But since there wasn't even a way to imagine how an event might be, we designated it in a kind of code: Event A, Event B, Event C, and so on, just to distinguish one from the other. What I mean is: since there were no alphabets in existence then or any other series of accepted signs, first we bet on how a series of signs might be and then we matched these possible signs with various possible events, in order to identify with sufficient precision matters that we still didn't know a thing about. [...]

And so, from the data I had at my disposal, I tried mentally to deduce other data, and from them still others, until I succeeded in suggesting eventualities that had no apparent connection with what we were arguing about. And I just let them fall, casually, into our conversation. For example, we were making predictions about the curve of the galactic spirals, and all of a sudden I came out with: "Now listen a minute, (k)yK, what do you think? Will the Assyrians invade Mesopotamia?" He laughed, confused. "Meso- what? When?" [...]

At the point we had reached, we needed reference libraries, subscriptions to specialized magazines, as well as a complex of electronic computers for our calculations: everything, as you know, was furnished us by a Research Foundation, to which, when we settled on this planet, we appealed for funds to finance our research. [...]

Acknowledgements

I really want to thank my supervisors who guided me in such a fascinating research field. I'm grateful to Paolo Del Giudice for all the scientific overviews and to Giacomo Indiveri for his compact low-power attitude. I would like to thank radio Vittorio for the explosive knowledge of electronic and funny stuff he taught me, Erminio Petetti for his incredible mechanical solutions and Maurizio Mattia who has always found time to answer my questions. And a special thanks to Guido Gigante and Mario Pannunzi for their help and their ideas on all the scientific and/or foolish things we worked on together.

Curriculum Vitae

Personal Information

Name	GIULIONI MASSIMILIANO
Date and place of birth	25/01/1979, Ancona (Italy)
Nationality	Italian
Languages	Italian (mother tongue), English (fluent)
e-mail	massimiliano.giulioni@roma2.infn.it

Research Fields

October 2004 to present	I am responsible for the design and tests of hybrid analog/digital VLSI chips implementing neuromorphic neural networks of spiking neurons and plastic synapses and models of selective visual attention. I am working on new solutions for scalable VLSI multi-chip architectures and inter-chip communication systems.

Education

October 2004 to present	International Co-PhD in physics at the university of Rome "Tor Vergata " and at the Institute of Neuroinformatics of the University of Zurich. Thesis title: "Networks of Spiking Neurons and plastic synapses, implementation and control". Final PhD defence to be held in March 2008.
February 2004	Master degree in physics at the university of Rome "La Sapienza". Thesis title: "Analysis and project of neuromorphic visual sensors". Specialization: cybernetics Final grade: 110/110 cum laude.
June 2003	Cambridge First Certificate in English Final grade: B
July 1998	School-leaving scientific certificate Final grade: 60/60

Scientific Collaborations and workshops

March 2004 to February 2007	Involved in ALAVLSI european project regarding VLSI neuromorphic chips, I collaborated with the Institute of Neuroinformatics in Zurich (INI), the Department of Technologies and Health of the National Institute of Health (ISS) in Rome, and the Otto-von-Guericke-Universitt in Magdeburg, Germany.
June 2004	Telluride NSF Workshop on Neuromorphic Engineering.

Publications

December 2007	Giulioni, Pannunzi, Badoni, Dante, Del Giudice *A configurable analog VLSI neural network with spiking neurons and self-regulating plastic synapses which classifies overlapping patterns* Proceedings of Neural information Processing Systems 2007 Refereed Conference Paper
September 2007	Camilleri, Giulioni, Dante, Badoni, Indiveri, Michaelis, Braun, Del Giudice *A Neuromorphic aVLSI network chip with configurable plastic synapses* Proceedings of the Hybrid Intelligent Systems 2007 Refereed Conference Paper
June 2007	Giulioni, Pannunzi, Badoni, Dante, Del Giudice *Classification of overlapping patterns with a configurable analog VLSI neural network of spiking neurons and self-regulating plastic synapses* Neural Computation (submitted) Journal Paper
January 2006	Badoni, Giulioni, Dante, Del Giudice *An aVLSI recurrent network of spiking neurons with reconfigurable and plastic synapses* Proc. of the IEEE International Symposium on Circuits and Systems 2006 Refereed Conference Paper

Fellowships and Contracts

November 2007 to present	Post-Doc fellowship at the second university of Rome, Electronic engineering department. *Scalable architectures, multi-chip infrastructures*
August 2004 to June 2006	Co.Co.Co. contract with the Italian National institute of Health: researcher role within the European project ALAVLSI. *Design and test of neuromorphic chips*
October 2004 to October 2007	PhD fellowship at the second university of Rome, Physics department. *Implementation and control of hardware neural networks.*

References

L. Alvado, J. Tomas, S. Saighi, S. Reneaud, T. Bal, A. Destexhe, and G. Le Masson. Hardware computation of conductance-based neuron models. *Neurocomputing*, 58-60: 109–115, 2004.

D. J. Amit. The hebbian paradigm reintegrated: local reverberations as internal representations. *Behavioral and Brain Science*, 18:617 – 657, 1995.

D.J. Amit and S. Fusi. Learning in neural networks with material synapses. *Neural Computation*, 6:957–982, 1994.

J. Arthur and K. Boahen. Learning in silicon: Timing is everything. In B. Scholkpf Y. Weiss and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2006.

D. Badoni and M. Annunziato. avlsi design of a learning attractor network of spiking neurons. In *Workshop on Neural Networks: From Biology to Hardware Implementations Chia, Sardinia, 1996*, 1996.

C. Bartolozzi. *Selective attention in silicon: from the design of an analog VLSI synapse to the implementation of a multi–chip system.* PhD thesis, ETH Zürich, Zürich, Switzerland, May 2007.

C. Bartolozzi and G. Indiveri. Synaptic dynamics in analog VLSI. *Neural Computation*, 19 (10):2581–2603, Oct 2007.

T. V. Bliss and T. Lømo. Long lasting potentiation of synaptic transmission in the dendate area of anaesthetized rabbit following stimulation of the perforant path. *Journal of Physilogy*, 232:331 – 356, 1973.

K. Boahen. Communicating neuronal ensambles between neuromorphic chips. In T. S. Lande, editor, *Neuromorphic Systems Engineering*, pages 229–259. Kluwer Academic, Norwell, MA, 1998.

K. A. Boahen. Point-to-point connectivity between neuromorphic chips using address-events. *IEEE transactions on circuits and systems*, 20:100–117, 1999a.

K. A. Boahen. A throughput on-demand address-event transmitter for neuromorphic chips. In *Advanced research in VLSI*. IEEE computer Society, 1999b.

K. A. Boahen. A burst-mode word-serial address-event link i: Transmitter design. *IEEE transactions on circuits and systems*, 51:1269–1280, 2004a.

K. A. Boahen. A burst-mode word-serial address-event link ii: Receiver design. *IEEE transactions on circuits and systems*, 51:1281–1291, 2004b.

K. A. Boahen. A burst-mode word-serial address-event link iii: Analysis and test results. *IEEE transactions on circuits and systems*, 51:1292–1300, 2004c.

J.M. Brader, W. Senn, and S. Fusi. Learning real world stimuli in a neural network with spike driven synaptic dynamics. *Neural Computation*, 19:2881–2912, 2007.

N. Brunel, F Carusi, and S Fusi. Slow stochastic hebbian learning of classes of stimuli in a recurrent neural network. *Network*, 9:123 – 152, 1998.

E. Chicca. *A Neuromorphic VLSI System for Modeling Spike–Based Cooperative Competitive Neural Networks*. PhD thesis, ETH Zürich, Zürich, Switzerland, April 2006.

E. Chicca and S. Fusi. Stochastic synaptic plasticity in deterministic aVLSI networks of spiking neurons. In Frank Rattay, editor, *Proceedings of the World Congress on Neuroinformatics*, ARGESIM Reports, pages 468–477, Vienna, 2001. ARGESIM/ASIM Verlag.

E. Chicca, D. Badoni, V. Dante, M. D'Andreagiovanni, G. Salina, S. Fusi, and P. Del Giudice. A VLSI recurrent network of integrate–and–fire neurons connected by plastic synapses with long term memory. *IEEE Transactions on Neural Networks*, 14(5):1297–1307, September 2003.

E. Chicca, A. M. Whatley, V. Dante, P. Lichtsteiner, T. Delbrück, P. Del Giudice, R. J. Douglas, and G. Indiveri. A multi-chip pulse-based neuromorphic infrastructure and its application to a model of orientation selectivity. *IEEE Transactions on Circuits and Systems I, Regular Papers*, 5(54):981–993, 2007.

E. Culurciello and A. G. Andreou. A comparative study of access topologies for chip-level address-eventt communication channels. *IEEE Transactions on Neural Networks*, 14: 1266–1277, 2003.

E. Culurciello, R. Etienne-Cummings, and K. Boahen. Arbitrated address-event representation digital image sensor. *Electronics Letters*, 37:1443–1445, 2001.

V. Dante, P. Del Giudice, and A. M. Whatley. Hardware and software for interfacing to address-event based neuromorphic systems. *The Neuromorphic Engineer Newsletter*, 2 (1), 2005.

P. Dayan and L. F. Abbott. *Theoretical Neuroscience Computational and Mathematical Modeling of Neural Systems*. MIT press, 2001.

S. P. DeWeerth, M. S. Reid, E. A. Brown, and R. J. Butera. A comparative analysis of multi-conductance neuronal models in silico. *Biological Cybernetics*, 96:181 – 194, 2007.

G. Indiveri E. Chicca and R. Douglas. An event based vlsi network of integrate-and-fire neurons. In *Proc. of 2004 IEEE International Symposium on Circuits and Systems*, volume V.

Serrano-Gotarredona et al. Aer building blocks for multi-layer multi-chip neuromorphic vision systems. In *Advances in neural information processing systems*, volume 18, pages 1217–1224, 2006.

E. Farquhar and P. Hasler. A bio-physically inspired silicon neuron. *IEEE Transactions on Circuits and Systems*, 52:477–488, 2005.

R. FitzHugh. Impulses and physiological states in models of nerve membrane. *Biophys. J.*, 1:445–446, 1961.

D. Frey. Future implications of the log-domain paradigm. In *IEE proceeding Circuits, DEvices Systems*, volume 147, pages 65–72, 2000.

S. Fusi. Long term memory: encoding and storing strategies of the brain. *Neurocomputing*, 38:1223–1228, 2001.

S. Fusi. Hebbian spike-driven synaptic plasticity for learning patterns of mean firing rates. *Biological Cybernetics*, 87:459–470, 2002.

S. Fusi. Spike-driven synaptic plasticity for learning correlated pattern of mean firing rates. *Rev. Neuroscience*, 14:73–84, 2003.

S. Fusi and M. Mattia. Collective behavior of networks with linear (vlsi) integrate and fire neurons. *Neural Computation*, 11:633–652, 1999.

S. Fusi, M. Annunziato, D. Badoni, A. Salamon, and D.J. Amit. Spike-driven synaptic plasticity: theory, simulation, vlsi implementation. *Neural Computation*, 12(10):2227–2258, 2000a.

S. Fusi, P. Del Giudice, and D.J. Amit. Neurophysiology of a vlsi spiking neural network: Lann21. In *Proc. IEEE-INNS-ENNS International Joint Conference on Neural Networks*, volume 3, pages 121 – 126, 2000b.

D. Debay G. Le Masson, S. Reneaud-Le Masson and T. Bal. Feedback inhibition controls spike transfer in hybrid thalamic circuits. *Nature*, 14:854–858, 2002.

A. Gara, M. A. Blumrich, D. Chen, G. L.-T. Chiu, P. Coetus, M. E. Gianpapa, R. A. Haring, P. Heildelberger, D. Hoenicke, G. V. Kopcsay, T. A. Liebsch, M. Ohmacht, B. D. Steinmacher-Burow, T. Takken, and P. Vranas. Overview of the blue gene/l system architecture. *IBM Journal of research and development*, 49:195–212, 2005.

G. Gerstein and B. Mandelbrot. Random walk models for the spike activity of a single neuron. *Biophysical Journal*, 4:41–68, 1964.

W. Gerstner and Werner M. Kistler. *Spiking Neuron Models, Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.

P. Del Giudice and M. Mattia. Long and short-term synaptic plasticity and the formation of working memory: a case study. *Neurocomputing*, 38:1175 – 1180, 2001.

P. Del Giudice, S. Fusi, and M. Mattia. Modeling the formation of working memory with networks of integrate-and-fire neurons connected by plastic synapses. *Journal of Physiology Paris*, 97:659–681, 2003.

Markram H, Lubke J., Frotscher M., and Sackmann B. Regulation of synaptic efficacy by conicidence of postsynaptic aps and epsps. *Science*, 275:213–215, 1997.

A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117:500 – 544, 1952.

A. V. Holden. *Lecture notes in biomathematics*. Springer, 1976.

K. M . Hynna and K. Boahen. Thermodynamically equivalent silicon models of voltage-dependent ion channels. *Neural Computation*, 19:327–350, 2007.

G. Indiveri. A low-power adaptive integrate-and-fire neuron circuit. In *Proc. IEEE International Symposium on Circuits and Systems*, pages IV–820–IV–823. IEEE, May 2003.

G. Indiveri and S. Fusi. Spike-based learning in VLSI networks of integrate-and-fire neurons. In *Proc. IEEE International Symposium on Circuits and Systems, ISCAS 2007*, pages 3371–3374, 2007.

G. Indiveri, E. Chicca, and R. Douglas. A vlsi array of low-power spiking neurons and bistable synapses with spiketiming dependent plasticity. *IEEE Transactions on Neural Networks*, 17(2):211–221, 2006.

K. Meier J. Schemmel, D. Bruderle and B. Ostendorf. Modeling synaptic plasticity within networks of highly accelerated if neurons. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 3367 – 3370. IEEE, 2007.

E. R. Kandel and J. H. Schwartz. *Principles of neural science*. Elsevier science publishing co., 1985.

J. A. Kauer and R. C. Malenka. Synaptic plasticity and addiction. *Nature Reviews Neuroscience*, 8:844 – 858, 2007.

L. Lapique. Reserches quantatives sur l'excitation lectrique des nerfs traite comme une polarization. *J. Physiol. Pathol. Gen.*, 9:620–635, 1907.

J. Lazzaro, S. Ryckebusch, M.A. Mahowald, and C.A. Mead. Winner-take-all networks of o(n) complexity. *Advances in neural information processing systems*, 1:703, 1988.

J. Lin, P. Merolla, J. Arthur, and K. Boahen. Programmable connections in neuromorphic grid. In *Proc. of 2006 IEEE International Symposium on Circuits and Systems*, volume 1, pages 80–84, 2006.

B. Linares-Barranco, E. Shnchez-Sinencio, A. Rodriguez-Vbzquez, and J. L. Huertas. A cmos implementation of fitzhugh nagumo neuron model. *Journal of Solid State Circuits*, 26:956 – 965, 1991.

S. Liu and R. Douglas. Temporal coding in a silicon network of integrate-and-fire neurons. *IEEE Transactions on Neural Networks*, 15:1305–1314, 2004.

S. C. Liu, J. Kramer, G. Indiveri, T. Delbruck, and R. Douglas. *Analog VLSI: circuits and principles*. MIT press, 2002.

J.C. Ebergen M. Shams and M. I. Elmasry. Modeling and comparing cmos implementations of the c-element. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 6: 563–567, 1998.

M. Mahowald. Vlsi analogs of neuronal visual processing: a synthesis of form and function. *Ph.D. dissertation, Dep. Comput. Neur. Syst. California Institute of Technology*, 1992.

M. Mahowald and R. Douglas. A silicon neuron. *Nature*, 354:515–518, 1991.

H. Markram. The blue brain project. *Nature Reviews Neuroscience*, 7:153–160, 2006.

M. Mattia and P. Del Giudice. efficient event-driven simulation of large networks of spiking neurons and dynamical synapses. *Neural Computation*, 12:2305–2330, 2000.

C. Mead. Neuromorphic electronic systems. *Proceedings of the IEEE*, 78(10):1629–36, October 1990.

C.A. Mead. *Analog VLSI and Neural Systems*. Addison-Wesley, Reading, MA, 1989.

P. A. Merolla and K. A. Boahen. Dynamic computation in a recurrent network of heterogeneous silicon neurons. In *Proc. of IEEE International Symposium on Circuit and Systems*, pages 4539 – 4542, 2006.

P. A. Merolla, J. V. Arthur, B. E. Shi, and K. A. Boahen. Expandable networks for neuromorphic chips. *IEEE Transactions on Circuits and systems*, 54:301–311, 2006.

S. Mitra, S. Fusi, and G. Indiveri. A VLSI spike-driven dynamic synapse which learns only when necessary. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 2777–2780. IEEE, May 2006.

S. Mitra, G. Indiveri, and S. Fusi. Learning to classify complex patterns using a VLSI network of spiking neurons. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, Cambridge (MA), 2007. MIT Press. (In Press).

Y. Miyashita and H Toshiro. Neural representation of visual objects: encoding and top-down activation. *Current opinion in Neurobiology*, 10:187 – 194, 2000.

C. Morris and H. Lecar. Voltage oscillations in the barnacle giant muscle. *ber. Biophys. J.*, pages 193–213, 1981.

J. S. Nagumo, S. Arimoto, and S. Yoshizawa. An active pulse transmission line simulating the nerve axon. In *Proc. IRE*, volume 50, pages 2061–2070, 1962.

C. Posch P. Lichtsteiner and T. Delbruck. A 128x128 120db 30mw asynchronous vision sensor that respond to relative intensity changes. *2006 IEEE ISSCC Digest of Technical Papers*, pages 508 – 509, 2006.

B. Pakkenberg and H. J. Gundersen. Neocortical neuron number in humans: Effect of sex and age. *Journal of Comparative Neurology*, 348:312–320, 1997.

B. Pakkenberg, D. Pelvig, L. Marner, M. J. BundGaard, H. J. G. Gundersen, J. R. Nyengaard, and L. Regeur. Aging and the human neocortex. *Exp. Gerontology*, 38:95–99, 2003.

G. N. Patel and S. P. DeWeerth. Analogue vlsi morris-lecar neuron. *Electronics Letters*, 33: 997–998, 1997.

A. Petit and A. Murray. Learning temporal correlations in biologically-inspired avlsi. In *IEEE International Symposium on Circuits and Systems*, volume V, pages 817 – 820. IEEE, 2003.

Marios M. Polycarpou, editor. *IEEE Transaction on Neural Networks*, volume 14. IEEE Computational Intelligence Society, 2003.

Marios M. Polycarpou, editor. *IEEE Transaction on Neural Networks*, volume 8. IEEE Computational Intelligence Society, 1997.

C. Rasche and R.J. Douglas. Silicon synaptic conductances. *Journal of Computational Neuroscience*, 7:33–39, 1999.

L. M. Ricciardi. *Lecture notes in biomathematics*. Springer, 1977.

H. Riis and P. Hafliger. Spike based learningwith weak multi-level static memory. In *IEEE International Symposium on Circuits and Systems*, pages 393 – 396. IEEE, 2007.

R. Sarpeshkar. Brain power - borrowing from biology makes for low power computing - bionic ear. *IEEE spectrum*, 43:24–29, 2006.

R. Sarpeshkar. Analog versus digital: Extrapolating from electronics to neurobiology. *Neural Computation*, 10:1601–1638, 1998.

S. R. Schultz and M. A. Jabri. Analogue vlsi integrateand-fire neuron with frequency adaptation. *Electronic Letters*, 16:1357–1358, 1995.

Gordon M. Shepherd, editor. *The Synaptic Organization of the Brain*. Oxford University Press, 1998.

M. F. Simoni, G. S. Cymbalyuk, M. Q. Sorensen, R. L. Calabrese, and S. P. DeWeerth. Development of hybrid systems: Interfacing a silicon neuron to a leech heart interneuron. In *Advances in neural information processing systems*, volume 13, 2000.

M. F. Simoni, G. S. Cymbalyuk, M. E. Sorensen, R. L. Calabrese, and S. P. DeWeerth. A multiconductance silicon neuron with biologically matched dynamics. *IEEE Transactions on Biomedical Engineering*, 51:342 – 354, 2004.

P. J. Sjöström, G. Turrigiano, and S. B. Nelson. Rate, timin and coopearativity jointly determine cortical synaptic plasticity. *Neuron*, 32:1149–1164, 2001.

J. V. Arthur K. A. Boahen B. E. Shi T. Y. W. Choi, P. A. Merolla. Neuromorphic implementation of orientation hypercolumns. *IEEE Transactions on Circuits and Systems*, 52: 1049 – 1060, 2005.

E. Culurciello R. Etienne-Cummings U. Mallick, R. J. Volgelstein and G. Cauwenberghs. A real-time spike-domain sensory information processing system. In *Proc. of 2005 International Symposium on Circuits and Systems*, volume 3, pages 1919–1922, 2005.

A. van Schaik. Building blocks for electronic spiking neural networks. *Neural Networks*, 14: 617–621, 2001.

A. van Schaik and S. Liu. Aer ear: A matched silicon cochlea pair with address event representation interface. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume V, pages 4213–4216. IEEE, 2007.

X. J. Wang. Synaptic reverberation underlying mnemonic persistent activity. *Trends in Neuroscience*, 24:455 – 463, 2001.

R. W. Williams and K. Herrup. The control of neuron number. *The Annual Review of Neuroscience*, 11:423–453, 1988.