# A CAPACITIVE NEURAL NETWORK FOR ASSOCIATIVE MEMORY

M.Verleysen, D.Martin, P.Jespers

Neural networks show interesting properties in vision problems, such as pattern or character recognition. However, the advantages of such networks (speed, convergence,...) are only fully exploited with dedicated processors. This paper presents a VLSI architecture for Hopfield-like fully interconnected networks. Its particularity is to use capacitors as synaptic connections in place of resistors or current sources. This network is programmed by a learning rule adapted to such networks where the dynamics of the connection weights is restricted to some discrete values. The feasibility of this architecture has been proved by a 8-neurons network built with discrete components.

## I. INTRODUCTION

Artificial neural networks have already been theoretically studied for several years. They show interesting properties such as their speed or their ability to converge to stable states in many perception or optimization problems, and are used in pattern and character recognition. Most of the actual results are obtained by simulations on conventional computers; however, some advantages of neural networks are lost during simulation: speed, parallelism, fault-tolerance,... Dedicated VLSI processors are thus necessary for specific applications, where these advantages will make neural networks more interesting than conventional computers.

Neural networks are generally characterized by a very regular structure. For the implementation, several designs can be used. The simplest one is to use resistors as synaptic connections; the value of the resistor will then determine the synaptic weight. However, resistors are difficult to implement in standard VLSI technology, and take a large area on the chip [1]. Current

---

The authors are with the Laboratory of Microelectronics, Université Catholique de Louvain, 3 pl. du Levant, 1348 Louvain-la-Neuve, Belgium.

Michel Verleysen is working under an IRSIA fellowship.

sources can also be used, but the matching between the different components contained in the chip is not easy to obtain [2].

The architecture proposed in this paper uses capacitors in place of resistors and current sources. Capacitors are more easy to integrate in VLSI technology, and matching techniques used for example in switched-capacitors filters can be applied to enhance the precision.

## II. HOPFIELD'S FULLY INTERCONNECTED NETWORK

For recognition or vision tasks, several neural architectures can be used: multi-layer perceptrons, Hopfield's fully interconnected networks,... This paper will not discuss about the various possibilities, nor about their particularities and their ability to solve specific problems [3]. For the simplicity, we will consider in this paper only fully interconnected networks: the regularity of their structure make them the best candidates for an efficient solution.

An Hopfield fully interconnected network consists in N neurons, each one being connected to each other by one of the $N^2$ synapses (figure 1).

The output value of neuron j, $V_j$, can be either +1 or -1. In this model, we limit the possible connection values $T_{ij}$ to +1, 0 and -1 (learning algorithms coping with this restriction gave about the same results than with continuous synaptic weight values [4]). The function of each synapse is to multiply the synaptic weight Tij by the output value $V_j$ of the neuron to which the synapse is connected. The output values $V_i$ of the synapses in the same column are then summed by the neuron, and the results go through a non-linearity so that the output of the neuron is +1 if its input is positive, and -1 in the opposite case. The output of each neuron is then fed back into the synapses.

Although the architecture described here is an Hopfield network, the design used to implement the circuit is general and can be adapted for example to perceptrons; the only differences will be in the connections between neurons and synapses, while their design will remain identical.

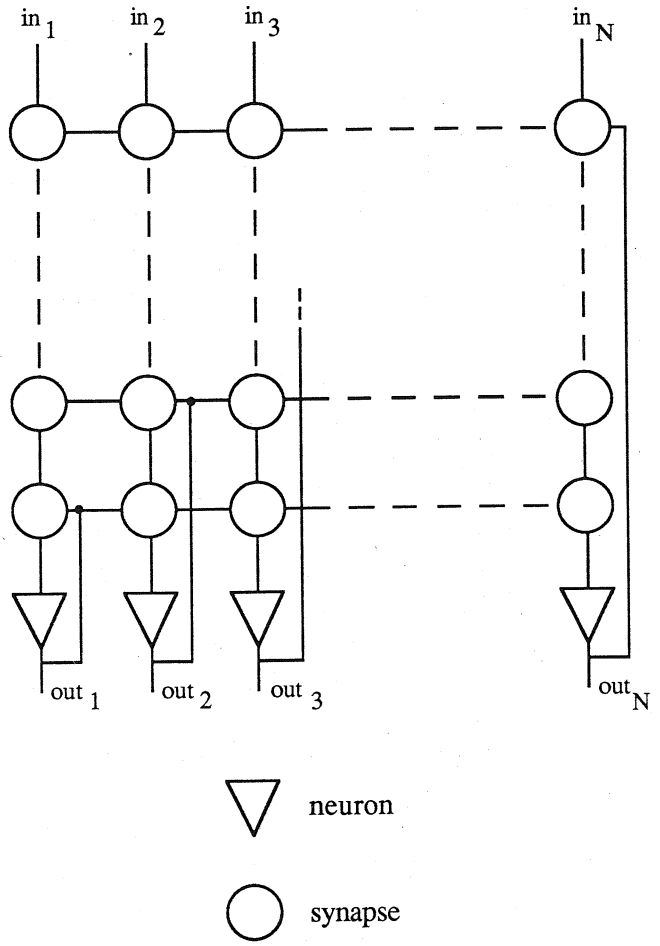


Figure 1: Hopfield's fully interconnected network

## III. CAPACITIVE VLSI ARCHITECTURE

The new architecture proposed in this paper is shown in figure 2. In this fully interconnected network, the neurons (at the bottom of the figure) sum the contribution of each synapse. The result of each sum is then compared to a threshold, latched and finally fed back into the square plane formed by the synapses. The reason to clock the whole circuit is to examine the intermediate states before the stable output is reached.

Each neuron consists in three separate parts: an integrator, a comparator and a latch.

The integrator collects all the charges coming from the $C_{ij}$ capacitors (for all j) and sums them into $C_i$. This scheme of integrator is in fact independent of any parasitic capacitor. Indeed, possible parasitic capacitors are shown in figure 3 ($C_{p1}$ ... $C_{p3}$). The capacitors $C_{p1}$ and $C_{p2}$ have no influence on the integration because they are connected to low-impedance nodes. The case of $C_{p3}$ is different. Before any integration the whole system is reset and therefore the potential of node A will be Vref. During the integration period, the potential on this node will vary but will finally converge to the same potential than the non-inverting input of the operational amplifier: $V_{ref}$. This comes from the fact that the operational amplifier built with a feedback loop will minimize the difference between its two inputs. Since $C_{p3}$ will not see different potentials between the beginning and the end of integration, this capacitor will have no influence on the result. The main speed limitation will come from the settling time of the operational amplifier charged by a capacitor.

The output of the integrator is then compared to $V_{ref}$ and the result is latched by $\Phi$. The latch provides two complementary logic outputs $V_i$ and $\overline{V}_i$ which are fed back into the synapse plane.

As described in §II, each synapse has to perform the product $T_{ij}.V_j$ . This is implemented by transmission gates which will route to $C_{ij}$ either $V_j$, $\overline{V}_j$ or $V_{ref}$ respectively corresponding to the weights +1, -1 and 0.

The weight of each synapse is stored on the memory points $M_{ij0}$ and $M_{ij1}$. During initialization, the network has to be programmed by storing the weights in every synapse. Before each evaluation, all the capacitors have to be discharged by the reset command. Only then the inputs can be injected

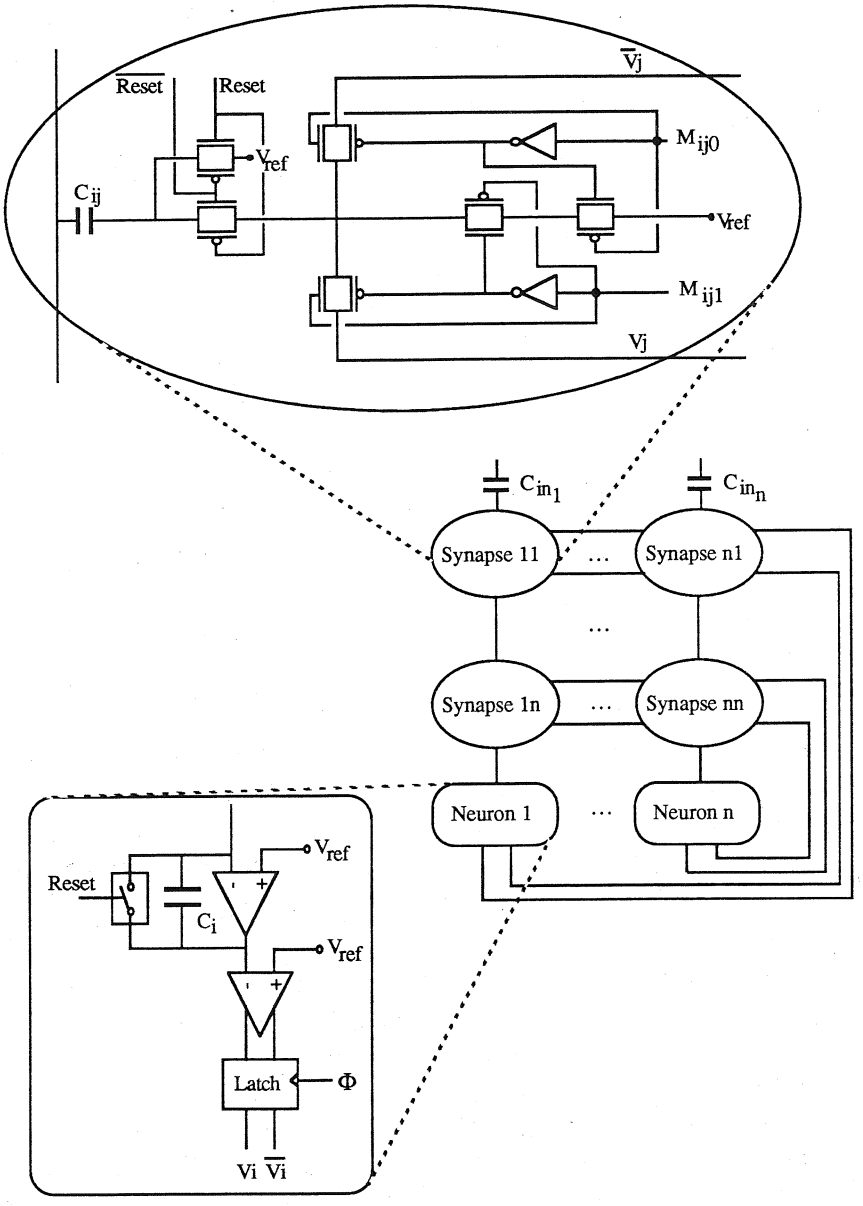


Figure 2: Capacitive architecture

through the capacitors $C_{in_i}$. The convergence process can then begin: each clock pulse will correspond to a new computation of the neuron values, and the network will have converged when all the neuron values are stable. This usually occurs after 10-50 cycles for networks with 10-100 fully interconnected neurons.

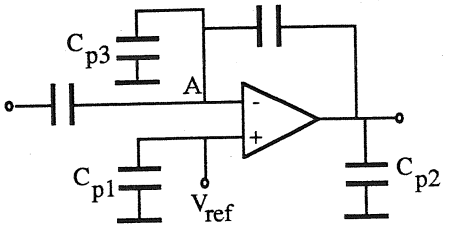


Figure 3: Parasitic capacitors in integrator

## IV. CONCLUSION

A VLSI architecture for artificial neural networks has been described. The example shown here presents an Hopfield's fully interconnected network, but the same design can be adapted to any neural architecture. In comparison with designs using resistors or current sources to realize the synapses, this architecture offers several advantages: the accuracy that can be reached with capacitors is increased, and the number of synapses that can be connected to the same neuron is greater. Furthermore, only a few transistors are imbedded in each synapse; and since only the relative values of the capacitors are important, their size can be reduced to very small values. The total area occupied by one synapse is thus small too, and a great number of them can be integrated on the same chip. While the minimum values for the capacitors to avoid interference by parasitic ones are still to be determined, estimations show that a density of about 40-50 synapses/mm2 in a 3-microns technology can be considered. A 8-neurons network has been realized with discrete components to prove the feasibility of such design.

## V. BIBLIOGRAPHY

[1] "An associative memory based on an electronic neural network architecture", R.E.Howard and al., IEEE Transactions on Electron Devices, vol.ED-34, n°7, July 1987.

[2] "A new VLSI architecture for large Hopfield's neural networks", M.Verleysen, B.Sirletti, P.Jespers, Proceedings of ESSCIRC 88, Manchester (U.K.), September 1988.

[3] "An introduction to computing with neural nets",R.Lippman, IEEE Acoustic, Speech and Signal Processing Magazine, April 1987.

[4] "Neural networks for high-storage content-addressable memories: VLSI circuit and learning algorithm", M.Verleysen, B.Sirletti, A.Vandemeulebroecke, P.Jespers, to be published in IEEE Journal of Solid-State Circuits, June 1989.