A Neuron- and a Synapse Chip for Artificial Neural Networks

Lansner, John; Lehmann, Torsten

Published in:
Proceedings of the 18th European Solid-State Circuits Conference

Publication date:
1992

Document Version
Publisher's PDF, also known as Version of record

A Neuron- and a Synapse Chip for Artificial Neural Networks

JOHN A. LANSNER AND TORSTEN LEHMANN

The Computational Neural Network Center,
Electronics Institute, Technical University of Denmark, Building 349,
DK-2800 Lyngby, Denmark.

Abstract — A cascadable, analog, CMOS chip set has been developed for hardware implementations of artificial neural networks (ANN's): $^{I)}$ a *neuron chip* containing an array of neurons with hyperbolic tangent activation functions and adjustable gains, and $^{II)}$ a *synapse chip* (or a matrix-vector multiplier) where the matrix is stored on-chip as differential voltages on capacitors. In principal any ANN configuration can be made using these chips.

A neuron array of 4 neurons and a 4 × 4 matrix-vector multiplier has been fabricated in a standard 2.4 μm CMOS process for test purposes. The propagation time through the synapse and neuron chips is less than 4 μs and the weight matrix has a 10 bit resolution.

I INTRODUCTION

Artificial neural network (ANN) implementations in analog VLSI technology have the advantages of great compactness and high speed, which make them suitable for real-time systems. ANN's are often modelled as $\underline{v} = \underline{g}(\underline{\underline{w}}[\underline{v}^T, \underline{u}^T]^T)$, where $\underline{v}$ is the neuron activation vector, $\underline{u}$ is the input vector, $\underline{\underline{w}}$ is the connection strength (synapse) matrix and $\underline{g}$ is a nonlinear function (squashing function) that is applied by coordinates [4, 6]. Thus, a hardware ANN could consist of a matrix-vector multiplier (synapse) chip followed by a squashing function (neuron) chip [3, 6].

For the *neuron chip* we have chosen the hyperbolic tangent, **tanh**, as the activation function. There are two reasons for this: $^{I)}$ Due to the exponential nature of bipolar transistors the **tanh** is simple to implement and hence well-defined; $^{II)}$ it has a convenient gradient function which makes the implementation of a learning algorithm for the neural network easier and more efficient.[1]

The *synapse chip* is a matrix-vector multiplier which is to be used both in the implementations of the ANN's and in future implementations of learning algorithms (eg. *Backpropagation* [4] or *Real-Time Recurrent Learning* [6]). The synaptic weights are stored as differential voltages on capacitors — refreshed by a static RAM via a D/A converter [5].

A neuron chip with 4 neurons and a synapse chip with 4 × 4 synapses have been fabricated in a 2.4 μm CMOS process for test purposes. The neuron chip has current inputs and voltage outputs. The synapse chip has voltage inputs and current outputs. Using this current-voltage scheme, the outputs from several synapse chips can be connected to one neuron input, and the output from one neuron can be distributed to several synapse chips. Thus in principal, any ANN configuration can be made with these chips.[2] This is illustrated in *fig.4*.

[1] The derivative of tanh is a function of tanh: $\tanh'(\beta x) = \beta(1 - \tanh^2(\beta x))$.

[2] In a continuous time, recurrent network, the stability has to be taken into consideration.

II THE NEURON CHIP

The neuron chip contains an array of neurons. Each neuron has three stages: An input stage controlling the *gain-term*, a transfer stage containing the *hyperbolic tangent function*, and finally an output buffer, see *fig.1*. The input current I_{in} is converted to a voltage V' by an opamp with "Double-MOSFET" feedback [1, 2], *fig.1a*. The gain-term is controlled by $\Delta V_{\text{gain}} = V_{\text{gain1}} - V_{\text{gain2}}$. The voltage V' is transfered by a hyperbolic tangent function (tanh) to the voltage V_{out}, *fig. 1b*. The **tanh** function is basically obtained from a differential pair of *lateral bipolar transistors*, LPNP [7]. $V_{\text{tanh}} = V_{\text{tanh1}} - V_{\text{tanh2}}$ and I_{bias} control the magnitude of the output range.

The transfer function for a neuron is given by

$$V_{\text{out}} = V_{\text{ref}} + \frac{\alpha I_{\text{bias}}}{K_N \frac{W_t}{L_t}(V_{\text{tanh1}} - V_{\text{tanh2}})} \tanh\left(\frac{kI_{\text{in}}}{K_N \frac{W_g}{L_g}(V_{\text{gain1}} - V_{\text{gain2}})}\right) \tag{1}$$

K_N, W, and L denote the transconductance parameter, the channel width, and the channel length of the feedback transistors, respectively. The subscripts $_g$ and $_t$ refers to the transistors of the input stage, and the transferstage, respectively. α and k are constants; $\alpha = -I_C/I_E$, where I_E are the emitter current, and I_C are the lateral collector current for a single LPNP.[3] $V_{\text{ref}} = -2\,\text{V}$ and the transistor dimensions are designed to give $V_{\text{out}} \in \{-3\,\text{V}, -1\,\text{V}\}$ which corresponds to the input-range of the synapse chip.

II.1 EXPERIMENTAL RESULTS OF THE NEURON CHIP

The neuron output voltage V_{out} has been measured as a function of the input current I_{in} with $\Delta V_{\text{gain}} \in \{0.25\,\text{V}, 0.5\,\text{V}, 1\,\text{V}, 2\,\text{V}\}$, see *fig.5a*. The maximum deviation to the desired tanh functions is about 2% of the output range. The gain is adjustable with a range of 1:30 ($0.1\,\text{V} < \Delta V_{\text{gain}} < 3\,\text{V}$). The derivative of V_{out} with respect to I_{in} has been compared with $\beta(1 - V_{\text{out}}^2)$, where β is a constant. It appears that the deviation between the quantities of $dV_{\text{out}}/dI_{\text{in}}$ and $\beta(1 - V_{\text{out}}^2)$ is less than 10% of the maximum value of $dV_{\text{out}}/dI_{\text{in}}$. Input "current offsets" of $10\,\mu\text{A}$ were measured. The reason could be that the input opamp has a low gain ($< 60\,\text{dB}$), which together with an opamp offset voltage of 2 mV would give the measured "current offset". α has been measured to about 0.55, and the output range was adjusted by ΔV_{tanh}. The V_{ref} was adjusted (4 mV) to get the desired center value of the output range. The delay times (t_{HL},t_{LH}) were measured to be in the range from 400ns to 800ns.

III THE SYNAPSE CHIP

The synapse chip is a parallel, cascadable, analog, CMOS *matrix-vector multiplier* (MVM) with an analog stored matrix. The $(m \times n)$ MVM consists of m *inner product vector multipliers* (IPM's) as shown in *fig.2* [1, 2]. (The MOS transistors are working in the linear region.)

It can be shown [1] that the output voltage of the opamp is given by:

$$v_{OA} - V_{O\text{ref}} = \frac{1}{(W/L)_0(v_{C1} - v_{C2})} \sum_{i=1}^{n} (W/L)_i (x_{i1} - x_{i2})(y_{i1} - y_{i2}) \tag{2}$$

having quite good linearity. Setting $y_{i1} - y_{i2} = [\underline{v}^T, \underline{u}^T]_i^T$ for all the IPM's and $x_{i1} - x_{i2} = \underline{\underline{w}}_{ji}$ for the j'th IPM gives the matrix-vector multiplier. To save pins, single-ended signals were selected on the chip; that is $x_{i2} = 2\,\text{V}$ and $y_{i2} = V_{O\text{ref}} = -2\,\text{V}$.

As the high impedance x inputs are used as inputs for the matrix elements, these elements can be stored on the chip as charges on capacitors [5]. Using this scheme, only four transistors and two capacitors are essentially needed for each matrix element, thus making the potential dimensions ($(m \times n)_{\text{max}}$) of the matrix large. The price for the analog storage of $\underline{\underline{w}}$ is that the capacitors must be *refreshed* from an external, digital RAM with regular intervals (in a serial manner to save pins on the chip) as indicated in *fig.4*. This is justifiable as digital RAM is very cheap.

The matrix *unit element* (a synapse) is shown in *fig.3*. The *nand gate* and the sample switches do not take up much space as minimum transistors are used. To reduce the effect of charge injection [8] and leakage currents, a *differential sampling scheme* is used to write the matrix elements on the capacitors [5].

[3] Because of the (vertical) substrat collector current we have $\alpha \approx \frac{1}{2}$.

III.1 Experimental Results of the Synapse Chip

To ensure good resolution and high noise rejection (at the cost of linearity), large input voltage levels were selected on the test chip: $|x_{i1} - x_{i2}|_{max} = |y_{i1} - y_{i2}|_{max} = v_{C1} - v_{C2} = 1\,V$. The transconductor was implemented such that $1\,V \times 1\,V \sim 30\,\mu A \rightarrow 100\,\mu A$.

The *transfer characteristics* of a multiplier element (synapse) has been measured (*fig.5b*) and showed a quite good linearity — with the exception of the case with negative $x_1 - x_2$ values and positive $y_1 - y_2$ values. This is due to the fact that it was necessary to lower V_{SS} to ensure a reasonable output current swing. The problem can be solved by improving the opamp and the transconductor. The addition of two synaptic terms has been measured and the effect of a limited output current was observed; otherwise, the linearity is as would be expected in the light of the multiplication linearity.

A summary of the most *important properties* of the chip is given below. $1\,LSB_8$ is one *least significant bit* for an 8 bit resolution of the appropriate signal. Matrix offset: $\lesssim 16\,mV$ ($2\,LSB_8$), matrix resolution: $\lesssim 2\,mV$ ($\frac{1}{4}\,LSB_8$), synapse non-linearity: $\lesssim 16\%$ ($21\,LSB_8$)[4], output offset: $\lesssim 16\,\mu A$ ($7\,LSB_8$)[5], input offset: $\lesssim 6\,mV$ ($1\,LSB_8$), propagation delay: $\lesssim 2.5\,\mu s$ (to $\frac{1}{2}\,LSB_8$), matrix write time: $\lesssim 150\,ns$, matrix (weight) drift: $\lesssim 0.5\,mV/s$ ($0.07\,LSB_8/s$). It should be noted that the offset errors are (mostly) non-systematic and are of magnitudes compatible with ANN applications [6].

IV Conclusions

In this paper we have presented two cascadable, analog CMOS chips: a neuron chip and a synapse chip. Neurons on the neuron chips can be interconnected at random via synapses on the synapse chips, thus implementing an artificial neural network with arbitrary topologi. The synapse chip can also be used as a part of a hardware implementation of a *learning algorithm* for a neural network. The chips have been tested independently and have shown excellent properties with respect to ANN applications:

The neuron function is well-defined, and the derivative can be calculated directly from the output voltage. The adjustable gain ensures that the numbers of connected synapse inputs can be variable within a wide range. LPNP-transistors work well as a differential pair.

The synapse matrix resolution is better than 8 bits and is high enough for many ANN and learning applications [6]. The leakage currents in the capacitors holding the matrix elements are extremely small. For this reason, a serial refreshing scheme of the matrix elements in a 100×100 elements chip would be no problem. Actually, in a real-time system with learning, it might be possible altogether to omit the digital RAM that is used as back-up memory for the matrix elements.

The output offset currents from the synapse chip and the "current offsets" at the neuron chip inputs are quite large. But for an ANN application, this is no major problem (provided that the network is trained and used using the same chips) as the offset currents just displaces the neuron biases [4]. For a learning application this might be a problem, though [6].

The propagation time through the synapse and neuron chips is rather small ($< 4\,\mu s$), even though the opamps are quite slow. And as the propagation time is essentially independent of the number of devices cascaded, it is possible to get a very high throughput using these chips.

The neuron area is less than $4 \cdot 10^5\,\mu m^2$. A future implementation of a neuron chip with 100 neurons will have a chip area of approximately $50\,mm^2$. In the present implementation the area of a synapse is $33280\,\mu m^2$ but this can easily be reduced to about $15000\,\mu m^2$. For a $1\,cm^2$ chip this gives $(m \times n)_{max} \approx 100^2$, which is also the pin limitation of available packages if a fully parallel solution is sought.

In a *conclusion*, large, fast, analog *neural networks* with arbitrary topologies can be implemented by using full size neuron chips and synapse chips.

Acknowledgement: Thanks are due to Thomas Kaulberg for designing the operational amplifiers and the transconductance amplifiers.

[4] A non-linearity of $\lesssim 3\%$ ($4\,LSB_8$) is estimated if a better opamp is used.

[5] The offset can be reduced with an improved opamp.

REFERENCES

[1] Bibyk, Steven & Mohammed Ismail (1989): "Issues in Analog VLSI and MOS Techniques for Neural Computing". In: Carver Mead & Mohammed Ismail, eds.: "Analog VLSI Implementation of Neural Systems", pp. 103–133. Norwell: Kluwer Academic Publishers.

[2] Czarnul, Zdzislaw (1986): "Novel MOS Resistive Circuit for Synthesis of Fully Integrated Continous-Time Filters". *IEEE Transactions on Circuits and Systems, vol.* **33**, *no. 7*, pp. 718–721.

[3] Eberhardt, Silvio, Tuan Duong & Anil Thakoor (1989): "Design of Parallel Hardware Neural Network Systems from Custom Analog VLSI 'Building Block' Chips" (Washington 1989). *IEEE International Joint Conference on Neural Networks*, pp. II-183–II-190.

[4] Hertz, John, Anders Krogh & Richard G. Palmer (1991): "Introduction to the Theory of Neural Computation". Redwood City: Addison-Wesley Publishing Company.

[5] Kub, Francis J., Keith K. Moon, Ingham A. Mack & Francis M.Long (1990): "Programmable Analog Vector-Matrix Multipliers". *IEEE Journal of Solid-State Circuits, vol.* **25**, *no. 1*, pp. 207–214.

[6] Lehmann, Torsten (1991): "A Hardware Implementation of the Real-Time Recurrent Learning Algorithm". *10'th European Conference on Circuit Theory and Design, vol.* **2**, pp. 431–440.

[7] Vittoz, Eric A. (1983): "MOS Transistors Operated in the Lateral Bipolar Mode and Their Application CMOS Technology". *IEEE Journal of Solid-State Circuits, vol. hf sc-18, no. 3* pp. 273–279.

[8] Wegmann, George, Eric A. Vittoz & Fouad Rahali (1987): "Charge Injection in Analog MOS Switches". *IEEE Journal of Solid-State Circuits, vol.* **22**, *no. 6*, pp. 1091–1097.

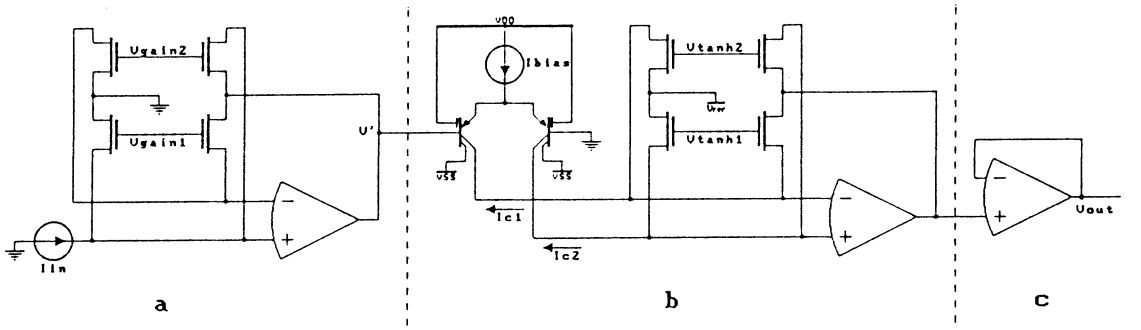

Figure 1. a) Input stage of a neuron, the adjustable current/voltage converter. b) Transfer stage, the hyperbolic tangent function. c) Output buffer.

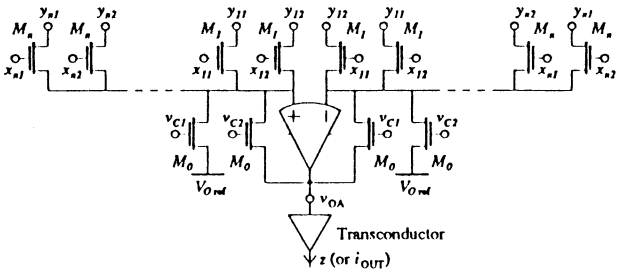

Figure 2. Inner product vector multiplier. The x_{ik}'s and y_{ik}'s are voltages.

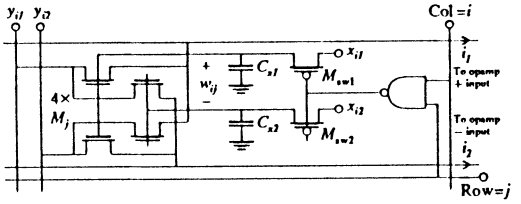

Figure 3. Matrix unit element (synapse) that calculates $(x_{i1} - x_{i2})(y_{i1} - y_{i2})$.

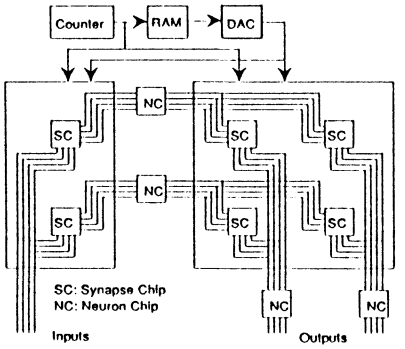

Figure 4. An implementation of a 4–8–8 feedforward network fully connected between the layers. A matrix updating scheme is also indicated.

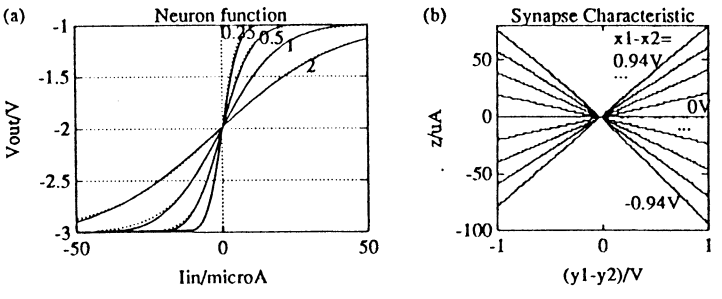

Figure 5. a) Measured neuron characteristics. b) Measured synapse characteristics.