

Low Power, Low Voltage Conductance-Mode CMOS Analog Neuron

V. Fabbriozio, F. Raynal, X. Mariaud, A. Kramer, G. Colli

Neural Network Design Group - Central R&D
SGS-Thomson Microelectronics
Agrate Brianza (Milan), ITALY

Abstract

Analog implementations of neural networks have been used for a wide variety of tasks especially in the area of image processing. Typically, implementations of analog neural networks have been based on the use of either current or charge as the variable of computation. This work introduces a new class of analog neural network circuits based on the concept of conductance-mode computation. In this class of circuits, accumulated weighted inputs are represented as conductances, and a conductance-mode neuron is used to apply nonlinearity and produce an output. The advantages of this class of circuits are twofold: firstly, conductance-mode computation is fast - we have developed circuits based on these principles which compute at 5-10 MHz; secondly, because conductance-mode computation requires the minimum charge necessary to compare two conductances, its energy-consumption is self-scaling depending on the difficulty of the decision to be made - we have a working prototype which consumes 166fJ per connection. The computing precision of these circuits is high: test results on a small test structure indicate an intrinsic precision of 8-9 bits. We have developed a larger test circuit which is able to perform computation with 1056 binary-valued inputs. Initial measurements in this large test structure indicate a more limited computing precision of 6+ - 8+ bits depending on the common mode of the input signal.

1: Introduction

Artificial neural networks have been successfully applied to problems such as speech or character recognition, and texture analysis [1, 2, 4, 6]. In the classification phase the network parameters are fixed and the network executes the recognition or the analysis starting from the information contained in the topology and in the weights. The base

operation executed by the single neuron during classification is the application of a non-linearity function to the weighted sum of the inputs:

$$out = f\left(\sum_{i=1,n} W_i * X_i\right)$$

where X are the inputs to the neuron, W the weights, and out is the result of the "activation function." In the hardware implementation of neural networks it is important to consider flexibility and power consumption in order to satisfy a wide range of applications. Our approach has been to focus on circuits which consume very little power per connection allowing for a high number of connections (> 1k) per neuron.

Analog implementations of Neural Network Architectures provide a framework for computation which is more efficient than standard digital techniques for certain problems. The purpose of this work is to explore the viability of this approach on a large scale using novel techniques based on Flash-EEPROM technology. One of the most attractive features of our circuit is that it implements synapses by a simple circuit based on a pair of floating-gate transistors, providing both analog multiplication and weight storage with low power consumption and high density (16µm x 4.4µm per synapses - 0.7µm CMOS technology). Both the weight storage and analog multiplication are implemented concurrently in a pair of floating gate of transistors.

The neuron consists of a conductance comparator which senses the difference between the synapses separated in positive and negative weights; this approach allows to achieve a computation with low power consumption (166 fJ for each connection), high precision (8 bit) and high speed (5 - 10MHz). We have implemented in a CMOS testchip our conductance mode neuron circuit. This novel conductance mode neuron is a suitable building block for large-scale array-based analog neural network

implementations and has been designed using a mixture of analog and digital subcircuits (mixed-mode).

The neuron circuit uses analog weighting and analog computation internally to reduce silicon area and power consumption while the data inputs and outputs are digital; a chip in which I/O is digital greatly simplifies integration at the system level.

2: Neuron Chip and Synapses

The use of floating gate technology for efficient long-term analog storage is well explored, especially in neural network implementations [3, 4, 5]. In this work we use a single pair of Flash-EEPROM devices for both analog storage and analog computation. Essentially, the core computational concept we are exploiting is to make use of a floating-gate device as a programmable switched conductance. By storing one analog value as the threshold of a floating gate device and applying a second digital value on the gate of the device, the conductance of the devices can be either zero (off) or a pre-programmed analog value (fig.1).

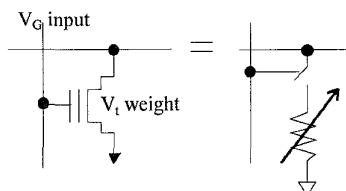


Figure 1. Ideal schematic concepts of Flash device functionality as a programmable switched conductance.

The use of a differential input scheme consisting of two conductance summing lines allows weights (conductance) to be either “positive” or “negative.” A conductance comparison neuron can then compare total “positive” conductance to total “negative” conductance and make a decision on the polarity of the total weighted inputs (conductance) as shown in fig. 2. The conductance or weight of these synapses are determined by the threshold (V_t) programmed on the device and the precision to which this threshold can be controlled gives the effective bit-equivalent precision of the synapse weight. It is possible to program the threshold of a floating gate device to a precision of 64mV[3]. This corresponds to 5 bits (32 levels) over our 2V input dynamic. The use of two devices adds a sign bit for a total of 6 bits per weight. Fig.3 shows the design of a standard Flash-EEPROM used as a conductance-mode synapse; each synapse occupies a $16\mu\text{m} \times 4.4\mu\text{m}$ area rendering very compact the area occupied and allowing high density of computational elements.

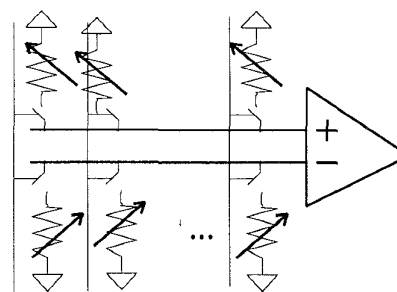


Figure 2. Electrical representation of Neuron functionality .

We use a conductance comparator as a neuron to apply the “activation function” of the neuron because comparing the two conductances allows us to compute with very little energy, reducing the overall power. The circuit that performs the neuron computation is shown in fig.4. The circuit is a conductance comparator that consists of three principal blocks: a current buffer to de-couple the synapses from the neuron, the neuron to perform the comparison, and a latch to digitise the output. The current buffer is shown in fig. 4.a.

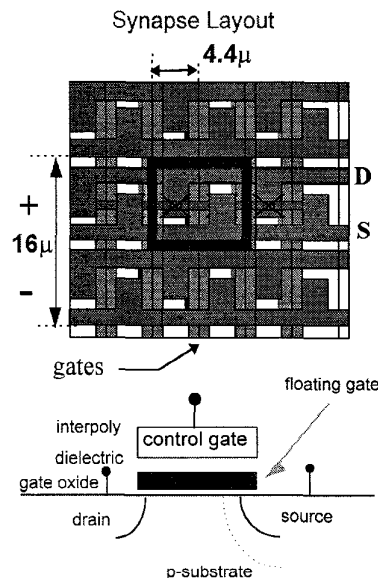


Figure 3. Conductance-mode Synapse

Each synapse is a conductance element implemented by a floating gate device; this presents several design constraints to the current buffer: firstly to minimise disturb programming it is important keep the drain voltage as low as possible ($< 100\text{mV}$); secondly as we intended to use

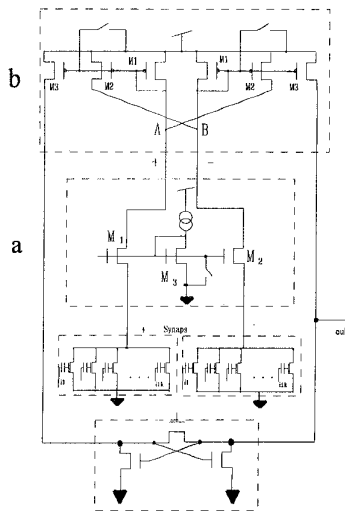


Figure 4. Neuron Conductance Comparator

many synapses ($O(1k)$) and each has parasitic drain capacitance on the order of 2fF, the drain node will respond slowly if it must undergo large voltage swings. To have a neuron with a large dynamic range output voltage and high speed, it is necessary to de-couple the drain node from the neuron.

To reduce the power consumption in the computation it is better that all the devices implementing synapses work in the triode region; for this reason we need a reference voltage to fix the device V_{ds} under the overdrive. The buffer function is realised by the M1 and M2 devices connected with a common gate, while the device M3 in diode configuration fixes the V_{ds} of the synapse devices. The fig.4.b shows the design of the neuron. The neuron is a conductance sensing circuit that performs a comparison between two different conductances coming from the synapse devices. The aim of the design is to solve the problem of the common mode range from 1 to 1000 synapses on during a computation with a solution that guarantees low power consumption, small silicon area, and high precision.

Consider an input to the neuron consisting of two currents (+ and -), the circuit is able to subtract the common mode current and discriminate the line of synapses with the minimum conductance using a cross mirror and positive feedback: this allows the circuit to achieve high precision over a wide range of input current. The speed of the circuit changes with the overall current: to optimise the speed performance and to digitise the output we use a standard latch structure.

3: Experimental results

An implementation of the neuron can be seen in fig.5 which shows a layout of a test structure. The neuron measure $200\mu m \times 32\mu m$ and it is fabricated in $0.7\mu m$ CMOS technology double poly, double metal, and contains 18 transistor total. The aim of the precision test on this structure is to estimate the computing precision of the comparator. The test structure contains two pairs of transistors connected to the neuron (fig. 6). One pair of the transistors are very big ($B=800/2$) and represent the common mode signal for the positive and negative synapses; the other pair of transistors are small ($S=0.8/2 = B/2^{10}$) like the minimum flash devices and represents the variable input signal. After offset compensation between the two large "common mode" transistors we apply a common mode voltage to them and find the minimum input signal needed to control the output via the pair of small transistors.

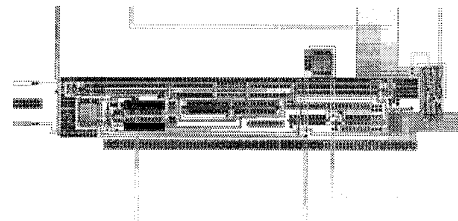


Figure 5. Layout of the test structure

A measurement was made to characterise the percentage of times the output was correctly controlled by the input over 1000 cycles using a "001100..." input pattern. With the maximum V_{common} fixed to 0.9V (100-200 mV above V_t) we measured the statistical output as a function of ΔV_{input} between the two "small" transistors. Fig. 8 shows a loss of precision starting at $V_{input} = 120mV$; the

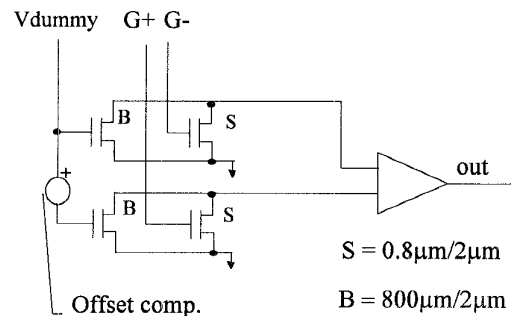


Figure 6. Schematic of the test structure

correspondence between this measurement and precision depends critically on the precise threshold of the devices: for our process we know V_t is between 0.7V and 0.8V which in the worst case means 8 bits and in the best case 9 bits of precision. This analysis is based on drawn transistor dimensions making exact precision characterisation following fabrication difficult.

We have realised another test structure (fig.7) to estimate the computing precision of the neuron comparator. The test structure contains 1k flash devices ($0.8\mu\text{m}/2\mu\text{m}$) for common mode input whose inputs are controlled by a shift register and 32 flash devices for differential input whose inputs are controlled by a latch. The circuit also contains programming and erasing drivers for the flash devices. The entire 1k flash represents the common mode signal for the positive and the negative synapses; 16 compensation flash are able to compensate the programming error and the remaining 16 input flash are used to apply the input signal to measure the overall precision. After all the flash devices have been programmed to $V_t = 2.5\text{V}$ we fill the shift register with "1"s. The gate voltage is fixed for all the flash to $V_t + \text{LSB}$ (we tested the precision for different LSB voltages). Applying a $V_t + \text{LSB}$ voltage to the flash gate means fixing a common mode signal for the positive and the negative synapses (as has done by the big transistors of the previous test structure). After compensation we apply a sequence of "00110011..." to 1 to 16 of the inputs testing for the correct output over 1 million cycles.

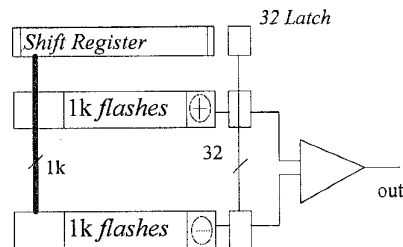


Figure 7. Block Diagram of new test structure

Figure 9 shown the percentage of correct output as a function of the number of inputs applied (1 to 16). It is clear that as the number of inputs increases the percentage of correct outputs increases, approaching 100% for 6 inputs with a $\text{LSB} = 32\text{mV}$. This correspond to a precision of 7+ bits [$\log_2(1k/6) = 7+$]. Figure 11 shows the precision in function of the LSB voltage: increasing the LSB to 128mV we loose 1 bit precision (6+ bits). The effective precision of the circuit depends critically on the common mode signal. Figure 10 shows the same test using only 0.5k flash devices (reduced common mode); in this case the input needed for 100% correct output is 2 and the effective precision is close to 8 bits [$\log_2(0.5k/2) = 8\text{bits}$].

Fig. 12 shows a measurement at 10MHz clock of the neuron functionality; limitations comes from individual PAD drivers which have a maximum speed of 10 MHz (simulation shows a peak speed 30MHz of for the neuron computation).

Figure 13 shows power consumption of the neuron over one computation: the consumption is equal to 166pJ at 5 V of power supply at a frequency of 1.7MHz. This correspond to 166fJ per input multiply-accumulate operation (1k inputs).

4: Conclusion

We have designed and characterised a conductance-mode analog neural circuit for the implementation of artificial neural networks. The circuit is based on a dense implementation of multiplying synapses which consist of a single pair of flash-EEPROM devices for storage of a 6-bit fixed weight and multiplication by a 1 bit input. A small test circuit has been characterised and demonstrates an inherent neuron precision of 8-9 bits. A large test structure based on this circuit and suitable for neural network computation with up to 1k inputs has been developed and preliminary testing results indicate a computing precision of 8 bits. This conductance-mode computing circuit is small ($200 \times 32\mu\text{m}/\text{neuron}$, $4.4\mu\text{m} \times 16\mu\text{m}/\text{synapse}$ - $0.7\mu\text{m}$ process), fast (5-10MHz) and very power efficient (166pJ/computation).

Acknowledgement

The authors express their thanks to M. Onorato for his contribution in the Shift Register and L. Fumagalli for his boards and testing equipment set up.

References

- [1] H. P. Graf et al. "A Reconfigurable CMOS Neural Network" ISSCC pp.144-145, feb., 1990.
- [2] B. E. Boser et al. "An Analog Neural Network Processor with Programmable Topology" IEEE Journal of Solid-State circuits, Vol. 26, no. 12, Dec., 1991.
- [3] A. Kramer et al. "Flash-Based Programmable Nonlinear Capacitor for Switched-Capacitor Implementations for Neural Network" IEDM Tech.Dig., pp. 17.6.1-17.6.4, Dec. 1993.
- [4] J. Lazzaro et al. "System Technologies for Silicon Auditory Models" IEEE Micro Vol. 3, no. 3, June 1994, pp. 7-15.
- [5] Holler, S. Tam, H. Castro, and R. Benson, "An Electrically Trainable Neural Network Chip (ETANN) with 1024 'Floating Gate' Synapses," in Proc. IJCNN, June 1989, pp 2.191-2.196.
- [6] A. Konig et al. "Massively Parallel VLSI-Implementation of a Dedicated Neural Network for Anomaly Detection in Automated Visual Quality Control" in Proc. Microneuro September 1994, pp.354-364.

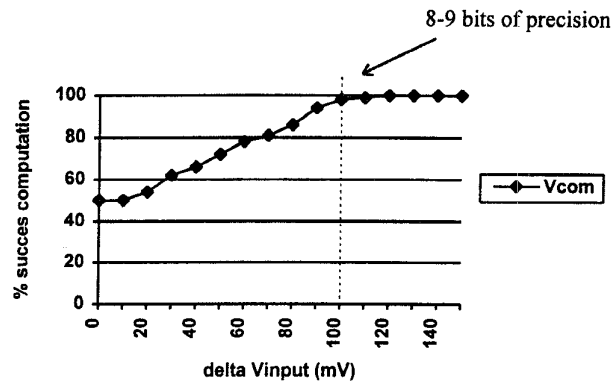


Figure 8. Intrinsic Precision curve of test structure.

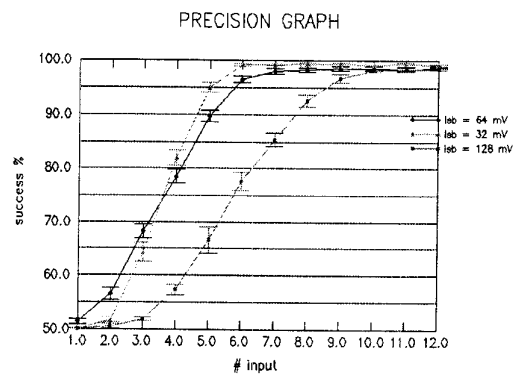


Figure 9. Precision curve with 1k devices of common mode.

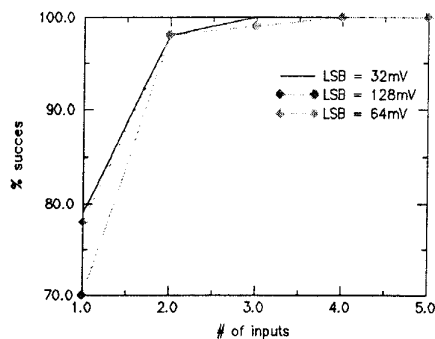


Figure 10. Precision curve with 0.5k devices of common mode.

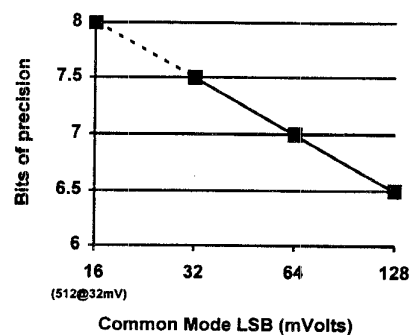


Figure 11. Bit precision curve versus LSB for 1k devices of common mode.

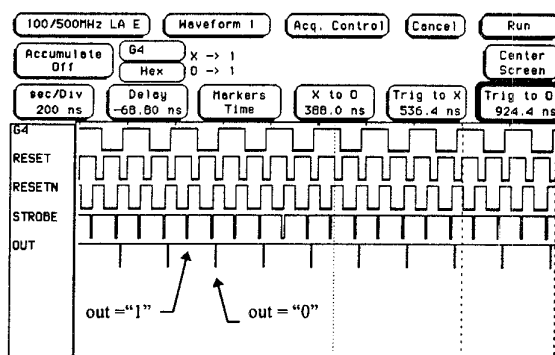


Fig 12. 10 Mhz clock diagram of neuron functionality.

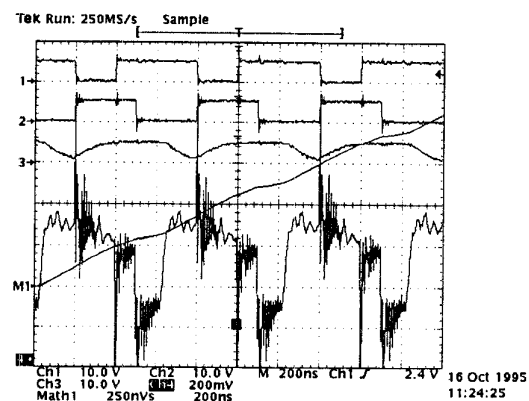


Figure 13. Power Consumption measurement at 5V op power supply. Current through 9k ohm resistor integrated.