A MICROPOWER LEARNING VECTOR QUANTIZER FOR PARALLEL ANALOG-TO-DIGITAL DATA COMPRESSION

Jeremy Lubkin and Gert Cauwenberghs

Department of Electrical and Computer Engineering
The Johns Hopkins University, Baltimore, MD 21218-2686
E-mail: gert@bach.ece.jhu.edu

ABSTRACT

An analog VLSI architecture for learning vector quantization (LVQ), with on-chip adaptation and dynamic storage of the analog templates, is presented. The architecture extends to Fuzzy ART and Kohonen self-organizing maps through digital programming. The analog memory and adaptive element of the LVQ cell comprise 6 MOS transistors and one capacitor, and provide for robust self-refresh of the dynamic analog storage. Total cell size including distance and adaptive computations is 80 × 70 lambda in scalable MOSIS technology. Experimental results from a fabricated 16 × 16 cell prototype in 2 μm CMOS are included.

1. INTRODUCTION

Vector quantization (VQ) [1] is widely used for digital encoding of vectorial analog signals, with applications to pattern recognition and data compression in vision, speech and beyond. Learning VQ (LVQ) augments coding with adaptive on-line construction of the template codebook, for optimal performance under changing input statistics. Neural variants on LVQ are Adaptive Resonance Theory (ART) [2] and Kohonen self-organizing maps, among others.

In its basic form, the implementation of VQ involves a search among a set of vector templates for the one which best matches the input vector, according to a given distance metric. Learning continually adjusts the best matching templates towards the input.

Several analog VLSI vector quantizers and their variants have been developed in recent years, *e.g.* [3]-[9]. The 16 × 16 VQ chip presented here is implemented in current-mode BiCMOS technology for low-power operation, and integrates learning as well as long-term dynamic capacitive storage of the analog templates using an incremental partial refresh scheme [12].

2. ARCHITECTURE

The architecture of the learning vector quantizer is shown in Figure 1. The core contains an array of 16 × 16 distance estimation cells interconnecting rows of templates with columns of input components. Each cell constructs the distance between one component x_j of the input vector $\mathbf{x}$ and the corresponding component ${y^i}_j$ of one of the template vectors $\mathbf{y}^i$,

$$d(x_j, {y^i}_j) = |x_j - {y^i}_j|^\nu\ ,\ \ i,j = 1\ldots16\ , \tag{1}$$

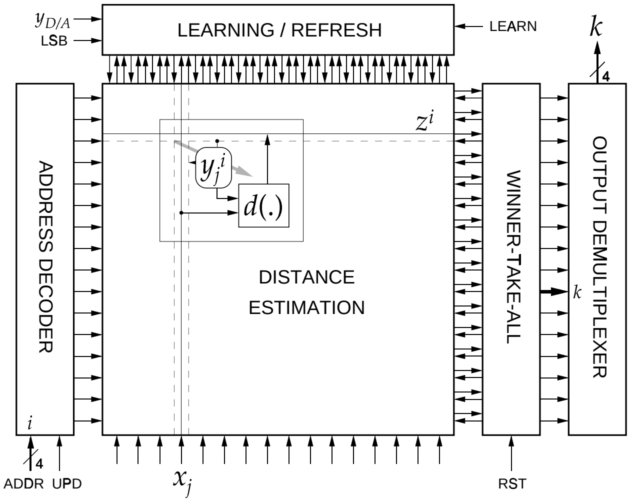

Figure 1: *Parallel VLSI architecture for analog-to-digital vector quantization (VQ), including template learning and refresh functions.*

using a mean absolute difference (MAD) distance metric, with norm $\nu = 1$. The component-wise distance is accumulated across inputs along template rows

$$z^i = \sum_{j=1}^{16} d(x_j, {y^i}_j)\ ,\ \ i = 1\ldots16 \tag{2}$$

and presented to a winner-take-all (WTA), which selects the single winner

$$k^{\text{WTA}} = \arg\min_i z_i\ . \tag{3}$$

Learning VQ is performed by selecting the winning template k and producing an incremental update $\Delta\mathbf{y}^k$ in the stored vector $\mathbf{y}^k$ towards the input vector,

$$\Delta{y^k}_j = \lambda\,(x_j - {y^k}_j). \tag{4}$$

where λ determines the rate of adaptation. In the case of Kohonen self-organizing maps [3], the neighbors of the winner, $k = k^{\text{WTA}} \pm 1$, are also adjusted according to (4) to preserve topological ordering in the digital coding.

The implemented architecture is also capable of performing fuzzy adaptive resonance [2] classification and learning, for stable

This work was supported by ARPA/ONR MURI N00014-95-1-0409, and NSF Career Award MIP-9702346. Chip fabrication was provided through MOSIS.

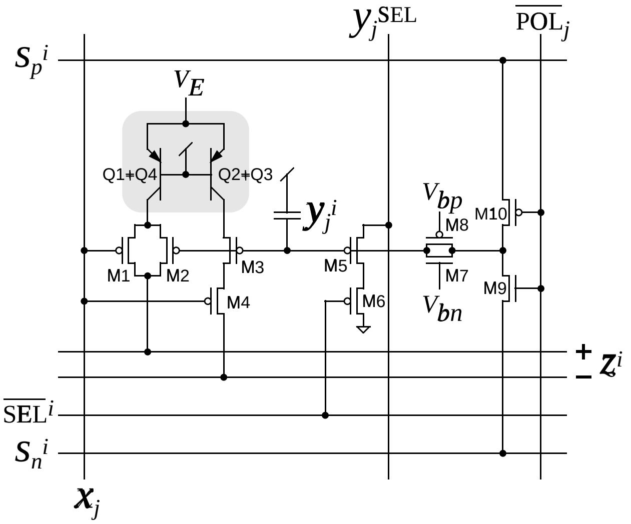

Figure 2: *Circuit schematic of the VQ distance estimation cell, with integrated learning and template refresh. The dashed inset indicates a matched double pair of lateral bipolar transistors.*

coding with control over the granularity of the template classes, by reconfiguring the chip. This amounts to modifying (1) into a fuzzy minimum, and corresponding changes to the learning updates (4).

3. VLSI IMPLEMENTATION

The circuits are implemented in current-mode BiCMOS technology, with MOS transistors operated in subthreshold for low-power dissipation. Lateral bipolar transistors offer full compatibility with standard CMOS processes.

3.1. Distance Estimation

The circuit diagram of the distance estimation cell is shown in Figure 2. The principle to construct the mean absolute error (1) measure between x_j and ${y^i}_j$ is similar to the approach in [7], except here operated continuous-time and in current-mode. The measure (1) is decomposed as the difference between the maximum and the minimum of x_j and ${y^i}_j$, accumulated separately onto two wires z^i+ and z^i- and differentially combined outside of the array. The differential distance computation is performed by modulating the Early effect (collector conductance) of a matched (double) pair of bipolar transistors Q1-Q4 and Q2-Q3, by means of MOS transistors M1, M2, M3 and M4 connected as source followers. Parallel and series connections in the source followers yield the maximum and minimum of x_j and ${y^i}_j$, respectively, in the output currents.

A centroid geometry, shown in Figure 3, is used for improved matching between the bipolar transistor currents that supply the differential output. By combining collector outputs in pairs C_1+C_4 and C_2+C_3, systematic variations and gradients in geometry are cancelled to first order. Matching is important, since the collector conductance is relatively small. The Early effect is maximized by using a minimum length geometry for the base, equaling the minimum length of an MOS transistor.

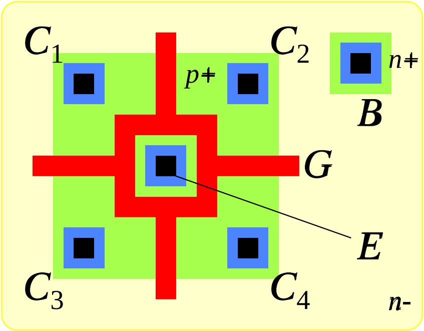

Figure 3: *Centroid geometry of the matched double pair of lateral bipolar transistors, in conventional n-well CMOS technology.*

The winner-take-all (WTA) is implemented as a variation on the standard current-mode design in [10, 11] with the addition of a triggered voltage-mode output stage for improved settling accuracy and speed [7].

3.2. Learning and Refresh

Transistors M5 through M10 implement the incremental update $\Delta {y^i}_j$, when the cell is selected either for refresh, or for learning (when $k^{\mathrm{WTA}} \equiv i$). The update $\Delta {y^i}_j$ is constant amplitude, variable polarity. A constant-amplitude discrete update is easier to implement than a continuous update, and gives superior results in the presence of analog imprecisions in the implementation [13]. The polarity $\overline{\mathrm{POL}}_j$ of the update $\Delta {y^i}_j$ is precisely implemented by means of a binary controlled charge pump [12]. The charge pump is free of switch charge injection parasitics, by avoiding clock signals on the MOS gates that couple capacitively into the storage capacitor. V_{bn} and V_{bn} are biased deep in subthreshold for precisely controlled increments and decrements as small as 10 μV. The timing of the update (and selection of the template) is performed by means of signals S_n^i and S_p^i [13].

The update polarity is computed externally to the array, by circuitry common for all cells on the same column, shown at the top of Figure 1. This arrangement is most space efficient since only one row of cells needs to be updated at once. A global LEARN signal selects the mode of operation, learning or refresh.

Figure 4 shows the simplified schematic of the learning cell, one per column of VQ distance cells. The circuit receives the selected analog template value ${y^i}_j$, which is used to generate the update polarity $\overline{\mathrm{POL}}_j$ and supply it to the selected distance cell. When a distance cell is selected, switch M6 is closed and transistors M5-M6 along with M11-M13 implement a comparator. The update is performed in the cell according to $\overline{\mathrm{POL}}_j$ by activating the signals S_n^i and S_p^i for the entire row of selected cells.

In learning mode (LEARN≡1), the polarity $\overline{\mathrm{POL}}_j$ is computed by comparing ${y^k}_j$ with the input x_j, yielding a modified update (4)

$$\Delta {y^k}_j = \mu\, \mathrm{sgn}(x_j - {y^k}_j). \qquad (5)$$

where μ is the constant-amplitude adaptation rate. In refresh mode (LEARN≡0), ${y^k}_j$ is compared with an external reference signal $y_{\mathrm{D/A}}$ to construct a binary quantization function used for partial incremental refresh [12]. As in [14], the binary quantization Q of ${y^k}_j$ is obtained by retaining the least significant bit (LSB) of analog-to-digital (A/D) conversion of ${y^k}_j$. A single-slope sequential A/D

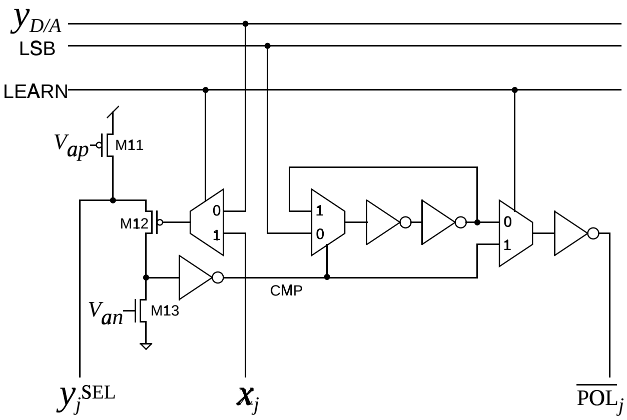

Figure 4: *Simplified schematic of the learning and refresh circuitry, in common for a column of VQ cells. Analog multiplexers are implemented with complementary CMOS switches.*

is implemented for simplicity, using a D/A signal on $y_{\mathrm{D/A}}$, ramped up in discrete steps synchronously with the alternating LSB. When the comparator flips sign, the instantaneous LSB value is sampled and latched to generate the update polarity $Q(y^k{}_j)$, producing an update [12]

$$\Delta y^k{}_j = -\mu\, Q(y^k{}_j). \tag{6}$$

4. EXPERIMENTAL RESULTS

A micrograph of the 16×16 learning vector quantizer, implemented in 2 μm CMOS technology, is shown in Figure 5. The chip is fully functional, and dissipates 2 mW from a 5 V supply at a 10 ksample/s parallel data rate.

The experimental distance metric is illustrated in Figure 6, obtained by sweeping one of the 16 inputs while fixing the other inputs to the template values. The inverting offset of the metric is intended to interface the current output with the winner-take-all, which searches for maximum current $-z^i$. A fuzzy ART metric [2] is obtained by bypassing one of the two differential currents from the array.

Results for dynamic refresh of the templates at 32-level quantization are shown in Figure 7, obtained by observing the drift in stored voltage level on one of the cells over 1000 refresh cycles, for 300 different initial values of the voltage $y^i{}_j$. The corresponding drift without refresh would have been tens of volts. Refresh tests across the entire array demonstrated long-term storage with minor level drifts over more than 1000 cycles. Further improvements in resolution and stability can be achieved, at some expense in silicon area, by using the A/D/A quantizer in [14].

Learning tests which validate the functionality of the learning vector quantizer with adaptive updates according to (5) are illustrated in Figure 8. The asymmetry in charging rate for upward and downward transitions is caused by transistor mismatch. Further experiments are currently underway to characterize system-level performance, for applications in speech coding and reconstruction.

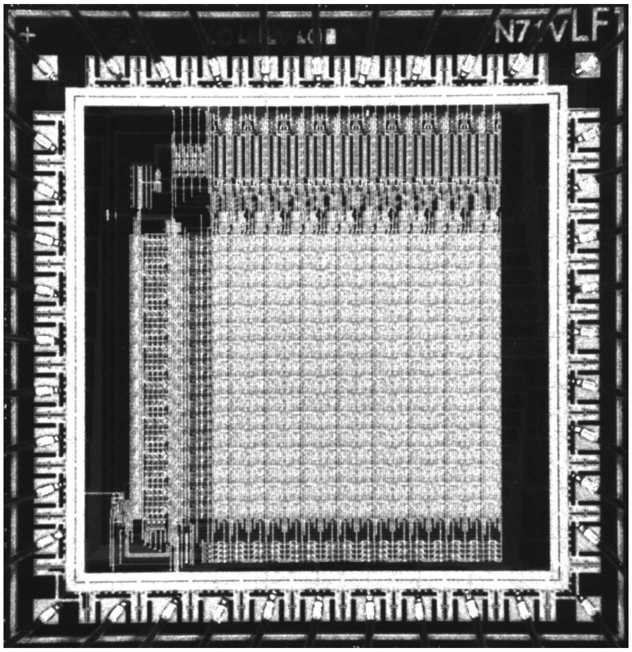

Figure 5: *Chip micrograph of the* 16×16 *array, parallel learning vector quantizer. The die size is* 2.2×2.25 mm^2 *in 2 μm CMOS technology.*

5. CONCLUSIONS

We presented a parallel architecture and corresponding analog VLSI implementation of a learning vector quantizer, fabricated in a CMOS-compatible BiCMOS process. The chip is fully functional, and can be configured for neural variants on VQ such as Fuzzy ART and Kohonen self-organizing maps. The chip incorporates analog storage of the templates, sharing the same circuitry used for learning. With a dense cell size of 70×80 λ units in scalable MOSIS technology, the integration of a 200-input, 1000-category classifier is feasible in a 0.35 μm CMOS process.

6. REFERENCES

[1] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression,* Norwell, MA: Kluwer, 1992.

[2] Gail A. Carpenter, Stephen Grossberg and David B. Rosen, “Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System,” *Neural Networks*, vol. **4**, pp 759-771, 1991.

[3] B. Hochet, V. Peiris, S. Abdot, and M.J. Declercq, “Implementation of a Learning Kohonen Neuron Based on a New Multilevel Storage Technique," *IEEE J. Solid-State Circuits,* vol. **26**, pp 262-267, 1991.

[4] W.C. Fang, B.J. Sheu, O.T.C. Chen and J. Choi, “A VLSI Neural Processor for Image Data-Compression Using Self-Organization Networks," *IEEE Transactions on Neural Networks,* vol. **3** (3), pp 506-518, 1992.

[5] Y. He and U. Cilingiroglu, “A Charge-Based On-Chip Adaptation Kohonen Neural Network," *IEEE Transactions on Neural Networks*, vol. **4** (3), pp 462-469, 1993.

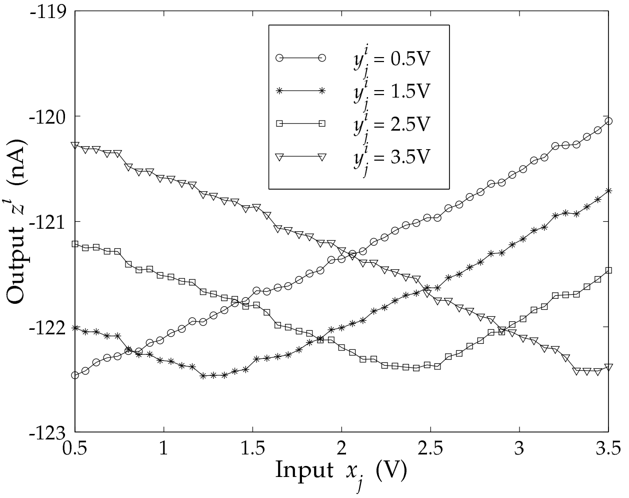

Figure 6: *Measured distance output for one row of cells, sweeping one input component while fixing the other 15 inputs.*

[6] G.T. Tuttle, S. Fallahi, and A.A. Abidi, "An 8b CMOS Vector A/D Converter," in *ISSCC Technical Digest,* IEEE Press, vol. **36**, pp 38-39, 1993.

[7] G. Cauwenberghs and V. Pedroni, "A charge-based CMOS parallel analog vector quantizer," in *Advances in Neural Information Processing Systems,* Cambridge, MA: MIT Press, vol. 7, pp 779-786, 1995.

[8] M. Konda, T. Shibata, T. Ohmi, "Neuron-MOS Correlator Based on Manhattan Distance Computation for Event Recognition Hardware," in *Dig. International Symposium on Circuits and Systems,* Atlanta, GA, 1996.

[9] T. Serrano-Gotarredona, B. Linares-Barranco and J. L. Huertas, "A Real Time Clustering CMOS Neural Engine", *Advances in Neural Information Processing Systems,* vol. **7**, Ed: D. S. Touretzky, Morgan Kauffmann, San Mateo, CA, 1995.

[10] J. Lazzaro, S. Ryckebusch, M.A. Mahowald, and C.A. Mead, "Winner-Take-All Networks of O(n) Complexity," in *Advances in Neural Information Processing Systems,* San Mateo, CA: Morgan Kaufman, vol. **1**, pp 703-711, 1989.

[11] A.G. Andreou, K.A. Boahen, P.O. Pouliquen, A. Pavasovic, R.E. Jenkins, and K. Strohbehn, "Current-Mode Subthreshold MOS Circuits for Analog VLSI Neural Systems," *IEEE Transactions on Neural Networks,* vol. **2** (2), pp 205-213, 1991.

[12] G. Cauwenberghs and A. Yariv, "Fault-tolerant dynamic multi-level storage in analog VLSI," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing,* vol. **41** (12), pp 827-829, 1994.

[13] G. Cauwenberghs, Analog VLSI Stochastic Perturbative Learning Architectures," *Int. J. Analog Integrated Circuits and Signal Processing,* vol. **13** (1/2), pp 195-209, 1997.

[14] G. Cauwenberghs, "A Micropower CMOS Algorithmic A/D/A Converter," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications,* vol. **42** (11), pp 913-919, 1995.

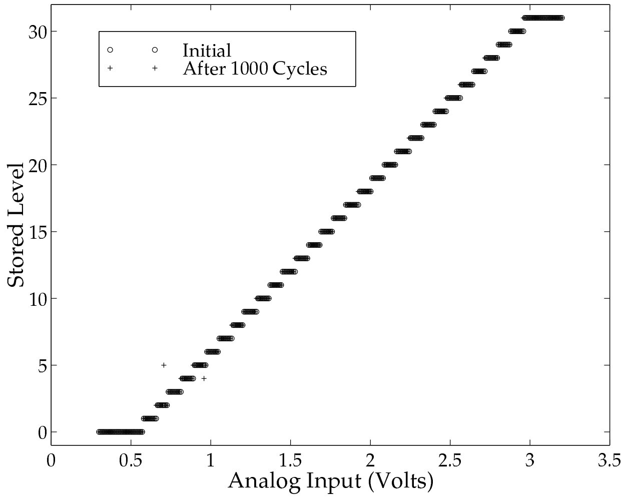

Figure 7: *Stability of the analog memory array. Drift over 1000 self-refresh cycles, for 300 different initial conditions.*

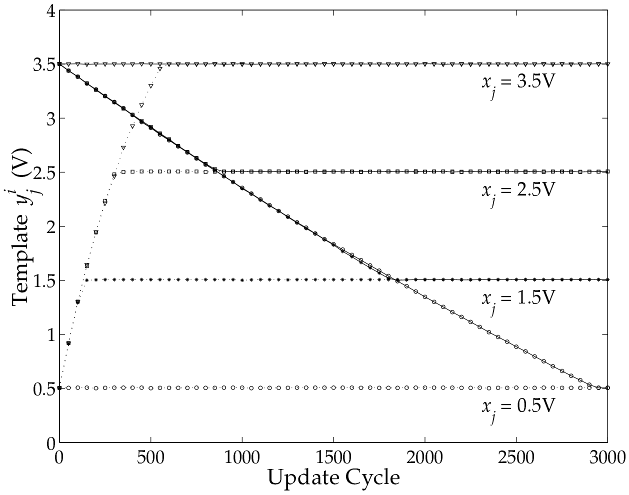

Figure 8: *Template adaptation of VQ in learning mode, under fixed inputs, from low and high template initial conditions.*