[8] G. Fischer, "Switched-capacitor FIR filters," in *IEEE Proc. ISCAS*, Philadelphia, PA, May 1987, pp. 742–745.
[9] Y. S. Lee and K. W. Martin, "A switched-capacitor realization of multiple FIR filters on a single chip," *IEEE J. Solid-State Circ.*, vol. 23, pp. 536–542, April 1988.
[10] G. Fischer, "Analog FIR filters by switched-capacitor techniques," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 808–814, June 1990.
[11] ——, "Analog RRS cells-An alternative approach for sampled-data lowpass filters," *Int. J. Analog Integrated Circuits and Signal Process.*, vol. 2, pp. 53–65, 1992.
[12] P. Gillingham, "Stray-free switched-capacitor unit-delay circuit," *Electron. Lett.*, vol. 20, pp. 308–310, March 1984.
[13] J. C. Mc Creary, "Matching properties and voltage and temperature dependence of MOS capacitors," *IEEE J. Solid-State Circ.*, vol. SC-16, pp. 608–616, Dec. 1981.
[14] R. Plodeck, U. W. Brugger, D. V. Gruenigen, and G. S. Moschytz, "SCANAL—A program for the computer aided analysis of switched-capacitor networks," *IEE Proc.*, no. 6, pp. 277–285, 1981.
[15] G. Fischer, "Design of switched-capacitor filters with emphasis on the high-frequency behavior," *Diss. ETH No. 7834*, Hartung-Gorre Verlag, Konstanz, 1985.
[16] P. R. Gray and R. G. Meyer, "MOS op-amp design-An overview," *IEEE J. Solid-State Circ.* , vol. SC-19, pp. 919–925, Dec. 1984.

Fault-Tolerant Dynamic Multilevel Storage in Analog VLSI

Gert Cauwenberghs and Amnon Yariv

***Abstract*—We present an area-efficient dynamic storage technique for repetitively quantizing and refreshing the analog contents of volatile capacitive memories in VLSI, incorporating redundancy and statistical averaging to avoid sudden loss of information triggered by occasional errors in the quantization. Experimental results obtained from a CMOS implementation are included, validating the robustness of the refresh scheme for long-term analog storage in excess of 8 bit resolution.**

I. Introduction

Typical applications of analog VLSI, such as implementations of neural networks and adaptive signal processing functions, require the storage of vast arrays of adjustable analog parameters, preferably integrated locally on the circuit in direct contact with the cells using the stored information. By now, a considerable variety of technology platforms and procedures have been developed to implement integrated analog memories in VLSI, each offering certain advantages while suffering from certain deficiencies or limitations [1]. One promising class of analog memories uses an iterative procedure of quantization and refresh to dynamically store and maintain multivalued analog information in a volatile storage medium, such as a voltage on a capacitor. In essence, each iteration of the procedure identifies the discrete memory level closest to the stored analog value (*quantization*), and subsequently overwrites the stored value with the identified level (*refresh*), thereby retaining the same memory state as long as the drift in the stored analog value between consecutive refresh cycles is guaranteed smaller than the minimum separation between adjacent memory levels. Implementation of the quantization and refresh functions in VLSI basically involves activation of an A/D/A (analog-to-digital and digital-to-analog) conversion [2], which may be shared in sequence by multiple storage devices. Variants on this mode of implementation [3]–[5] provide a periodic signal which sequences through the complete set of discrete memory levels for direct comparison with the stored value in memory, identifying and copying the nearest level in a manner somewhat related to dual-slope A/D conversion. While either configuration is able to store an analog value over an extended period of time, occasional errors in the quantization and refresh, occurring in practice due to noise and analog VLSI imprecisions, have a destructive impact on the data content of the memory. Indeed, a wrong classification of the nearest discrete analog level at quantization causes a complete loss of the stored information at the successive refresh operation. Furthermore, offsets occurring at refresh, due to incorrect replication of the identified level onto the storage device, increase the likelihood of a subsequent error in the quantization. The stringent requirements on precision in the implementation severely limit the analog resolution practically attainable with the dynamic storage method. An alternative realization, which incorporates error correction at the LSB level, has been proposed recently [6]. However, the modified scheme involves extra resources for local storage and handling of one bit of information, and assumes a limited amplitude for the random quantization and refresh errors.

Manuscript received January 11, 1994; revised May 5, 1994. This work was supported by ARPA by Grant N00014-92-J-1891. This paper was recommended by Associate Editor Jacek M. Zurada.

G. Cauwenberghs was with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA. He is now with the Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218 USA.

A. Yariv is with the Department of Applied Physics, California Institute of Technology, Pasadena, CA 91125 USA.

IEEE Log Number 9405315.

II. Partial Incremental Refresh

The alternative technique for dynamic multilevel storage presented here avoids the sensitivity to errors in the quantization by specifying partial updates for the analog value, with small fixed size increments in the direction of the nearest discrete level, rather than completely substituting the current analog value at refresh [7]. Hence occasional errors in the quantization merely cause small increments in the wrong direction, which are likely removed in subsequent cycles unless errors occur rather frequently. In addition, since the refresh operation does not attempt to replicate the identified discrete level onto the memory, potential offset errors at refresh are avoided, and resources to physically provide an analog signal representing the identified level for use in the memory cells may be eliminated. Also, only binary information is needed from the quantization to obtain the direction towards the nearest discrete level. This binary quantity, defining the polarity of the refresh increments, depends on the position of the analog value relative to the discrete levels. The relationship between the analog value and the corresponding binary value can therefore be formulated in terms of a binary quantization function, of which the characteristics relate to the position of the discrete levels. An example of the binary quantization function $Q(.)$, mapping an analog value V_m to a corresponding binary value $Q(V_m)$ for subsequent refresh, is illustrated in Fig. 1, with the arrows indicating the corresponding direction of the refresh increments. The resulting procedure is somewhat related to the technique of delta-sigma modulation, although different in scope and formulation. Formally, with V_m the analog level defining the state of the memory, and δ the size of the partial increments, the procedure combines the

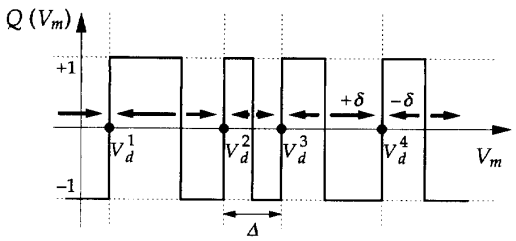

Fig. 1. Example illustrating the binary quantization function $Q(.)$

consecutive steps of [7]

$$\textit{Binary Quantization:}\quad Q(.)\colon \mathcal{R} \rightarrow \{-1, +1\};\ \text{and}$$
$$\textit{Incremental Refresh:}\quad V_m := V_m + \delta.\ Q(V_m). \tag{1}$$

Iteration of this procedure yields a stored memory value V_m maintained in the vicinity of one of the discrete levels $V_d{}^i$, as indicated on Fig. 1, provided the amplitude of the iterative corrections δ is sufficiently small compared to the worst-case separation between consecutive levels, $\Delta = \min_i(V_d{}^{i+1} - V_d{}^i)$. In particular, the degree of fault-tolerance provided by the repetitive procedure (1) drastically improves as the increment size δ is reduced, since then basically a longer sequence of consecutive errors is required to trip the analog value towards a neighboring faulty memory state over a distance Δ, an event becoming exponentially less likely with the length of the error sequence assuming errors occur randomly. The lower bound on δ is given by the nominal drift of the storage medium over one refresh cycle, $|r_d|T$ with r_d the worst-case storage droop rate and T the refresh time interval. Hence the choice of δ needs to satisfy

$$|r_d|T < \delta \ll \Delta. \tag{2}$$

Clearly, the increased fault-tolerance of the modified refresh procedure comes at the expense of an increase in the minimum refresh rate required to control the drift of the volatile medium, with the amount of increase corresponding to the degree of redundancy Δ/δ. Nevertheless, in practical analog VLSI situations with modest leakage rates of capacitive storage, the requirement (2) is fairly easy to accommodate.

III. IMPLEMENTATION

Two architectures for implementing the above procedure are given in Fig. 2. Both configurations comprise the same functional elements, a binary quantizer Q and an increment/decrement device I/D, which interface with a capacitive storage device C to generate a binary quantization value and to produce the corresponding refresh increment, respectively. The first configuration of Fig. 2(a) employs dedicated instances of elements Q and I/D for every individual storage device C. In the second configuration of Fig. 2(b), one binary quantizer Q is shared among several memory cells, each containing a storage device C and an I/D element, in a multiplexed arrangement for sequential refresh. The latter configuration allows a significant reduction in the cell complexity, but requires a quantizer of sufficient bandwidth to support the entire array of memory cells at the desired refresh rate.

In principle, any device constructing a binary output value from an analog input value in fairly consistent manner can perform the function of binary quantizer Q, provided the transfer characteristic contains sufficient and uniformly distributed bit transitions to accommodate the desired analog resolution of the memory. Practical realizations can be obtained as direct extensions on the previously used A/D/A quantizers for identifying the discrete level closest to the analog value [2], [1]. The binary value, encoding the direction towards the closest level, is then simply obtained through comparison

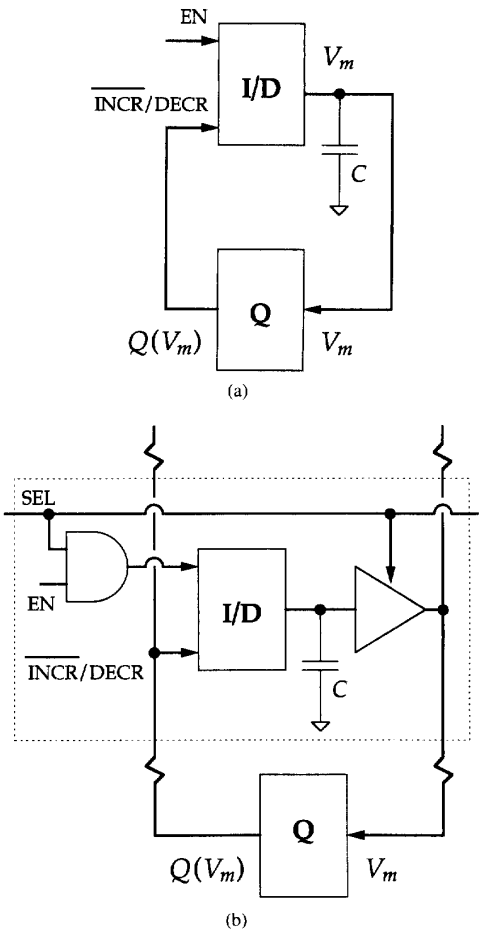

Fig. 2. Architectures implementing the partial incremental refresh method. (a) Standard configuration of the analog memory cell; (b) array configuration of analog memory cells with time-multiplexing of a common quantizer.

of the analog value with the identified level. Simple extensions on the dual slope variant [3]–[5] can be devised as well, e.g., comparison of the analog value with a reference ramp or staircase signal, sampling the instance of a periodically alternating binary clock signal at the time when the reference signal exceeds the analog value, to yield the binary quantization value.

A charge-pump implementation of the increment/decrement device in CMOS technology, in contact with a capacitive storage device C, is given in Fig. 3. Fixed charge increments or decrements on C are obtained by selectively activating one of two supplied constant currents of opposite polarity, over a fixed time interval. The switched currents are provided by two complementary transistors MN and MP, biased at constant gate voltages $V_{b\mathrm{DECR}}$ and $V_{b\mathrm{INCR}}$ in the sub-threshold region, thereby allowing for exponential control of the update currents down to sub-*pA* levels. The particular transistor providing the update current, for either increment or decrement action, is selected and activated by means of the input binary signal $\overline{\mathrm{INCR}}$/DECR, obtained from the quantizer, and the complementary enable control signals EN and $\overline{\mathrm{EN}}$, respectively. Switch injection noise on the storage capacitor at deactivation of the supplied current is mostly avoided, since switching is achieved by driving the source voltage of either transistor MN or MP while maintaining both gate voltages at constant levels, thereby directing the excess channel charge of the active

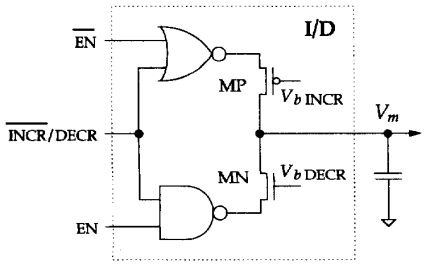

Fig. 3. A CMOS charge-pump implementation of the I/D device.

transistor during deactivation to the source rather than to the drain connected to the storage capacitor. The clean switching transients allow for precise control of the increment/decrement size δ down to small levels. Furthermore, the subthreshold update currents and the reverse diode leakage currents, largely responsible for the drift, exhibit approximately the same Boltzman temperature dependence, such that the critical requirement for memory stability in (2) can be met over a wide temperature range.

IV. Experimental Results

A simple test structure, containing the above charge-pump I/D element, a 1 pF capacitor, and a triggered latching comparator on a CMOS integrated circuit, was used to validate the above principles. The circuit conforms to the arrangement of Fig. 2(b), with a memory cell size of 0.01 mm^2 in 2 μm CMOS technology. The comparator, with the positive input in contact with the capacitor and the negative input connected to a constant threshold voltage level, serves to symbolize the binary quantizer. Hence the voltage on capacitor C is incremented by δ if that voltage is below the threshold, and is decremented by the same amount otherwise. While obviously the single-state analog memory thus implemented is of little practical significance, the example chosen is instructive as to assess the performance of an analog memory operating near one particular memory level, with regard to the impact of random errors in the binary quantization. The relative inaccuracy of the implemented comparator, representative of quantization errors, is evident from Fig. 4, showing an uncertainty region for the comparator binary output ranging over 6 mV of the analog input. Nevertheless, the excursion in the voltage on the memory capacitor observed under repetitive partial refresh, illustrated in Fig. 4 for different values of δ, was confined within a narrow region near the threshold level of the comparator, with a spread mainly limited by the increment size δ (and the resolution of the voltmeter used in the observations). Clearly, small values for δ, in the μV range, offer a resolution of the confined voltage beyond the precision of the quantizer. Extensive observation of the capacitor voltage under continued refresh with $\delta = 200$ μV, over an interval exceeding 10^9 cycles, has identified only a few cases of excursion beyond a distance 5 mV from the center of the distribution. Consequently, a noisy binary quantizer with characteristics matching those of the above comparator can comfortably accommodate a reliable analog memory of level separation say $\Delta = 12$ mV. With the dynamic range of the capacitor and I/D element exceeding 3 V from a single 5 V supply, and with a capacitor droop rate $|r_d|$ much less than 10 mV/s, 8 bit of analog resolution is thus supported reliably at a 100 Hz refresh rate.

V. Conclusion

A simple fault-tolerant refresh method for dynamic multilevel analog storage, with corresponding VLSI implementation, was presented and verified experimentally. Unlike related approaches basically replacing the stored analog value with its nearest quantized level

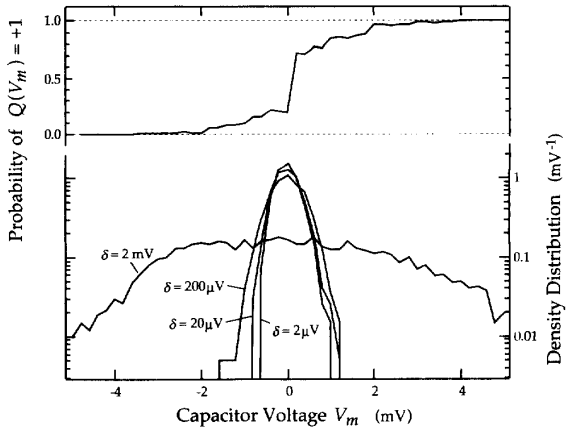

Fig. 4. Experimental observation of quantization and refresh. *Top:* Measured probability of comparator output "+1" state. *Bottom:* Density distribution of the capacitor voltage observed under periodic partial incremental refresh.

at refresh, the method specifies small partial fixed size incremental updates in the direction of the nearest level, as determined from a binary quantization of the stored value. Owing to the redundancy in the incremental updates, the memory retention under refresh using the method is significantly less sensitive to random errors in the quantization than related methods, effectively trading available refresh bandwidth for higher resolution through a mechanism similar to delta-sigma modulation. Consequently, the precision of the implemented binary quantizer is not a major factor in the analog resolution and retention ability of the memory. Furthermore, the incremental nature of the refresh mechanism obliterates the need of accurately replicating the analog value of the identified discrete level onto the storage device, thereby simplifying the implementation and avoiding potential offsets induced at the refresh stage due to device mismatch. Instead, the accuracy of analog memory operation is mainly determined by the relative size and direction of the implemented incremental updates. We proposed a charge-pump CMOS I/D device supplying reliable small size increments, and performed experiments on a fabricated test structure including the I/D device, which indicated stable memory operation at 8 bit analog resolution and beyond.

Acknowledgment

Fabrication of the CMOS chip used in the experiments was provided by the ARPA/NSF MOSIS service.

References

[1] Y. Horio and S. Nakamura, "Analog memories for VLSI neurocomputing," in *Artificial Neural Networks: Paradigms, Applications, and Hardware Implementations*. Piscataway, NJ: IEEE Press, 1992, pp. 344–363

[2] L. M. Terman, Y. S. Yee, R. B. Merrill, L. G. Heller, and M. B. Pettigrew, "CCD memory using multilevel storage," *IEEE J. Solid-State Circ.*, vol. 16, no. 5, pp. 472–478, 1981.

[3] B. Hochet, "Multivalued MOS memory for variable-synapse neural networks," *Electron. Lett.*, vol. 25, no. 10, pp. 669–670, 1989.

[4] 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 Circ.*, vol. 26, no. 3, pp. 262–267, 1991.

[5] E. Vittoz, *et al.*, "Analog storage of adjustable synaptic weights," in *VLSI Design of Neural Networks*. Norwell MA: Kluwer, 1991, pp. 47–63.

[6] E. K. F. Lee and P. G. Gulak, "Error correction technique for multi-valued MOS memory," *Electron. Lett.*, vol. 27, no. 15, pp. 1321–1323, 1991.

[7] G. Cauwenberghs and A. Yariv, "Method and apparatus for long-term multi-valued storage in dynamic analog memory," U.S. patent pending, filed 1993.