AN MDAC SYNAPSE FOR ANALOG NEURAL NETWORKS

Ryan J. Kier[*], *Reid R. Harrison*[*], *and Randall D. Beer*[**]

[*]University of Utah, Department of Electrical and Computer Engineering
Salt Lake City, UT
[**]Case Western Reserve University, Department of Electrical Engineering and Computer Science
Cleveland, OH

ABSTRACT

Efficient weight storage and multiplication are important design challenges which must be addressed in analog neural network implementations. Many schemes which treat storage and multiplication separately have been previously reported for implementation of synapses. We present a novel synapse circuit that integrates the weight storage and multiplication into a single, compact multiplying digital-to-analog converter (MDAC) circuit. The circuit has a small layout area (5400 μm^2 in a 1.5-µm process) and exhibits good linearity over its entire input range. We have fabricated several synapses and characterized their responses. Average maximum INL and DNL values of 0.2 LSB and 0.4 LSB, respectively, have been measured. We also report on the performance of an analog recurrent neural network which uses these new synapses.

1. INTRODUCTION

Many implementations of analog neural networks have been developed over the past 15 years. Two critical issues in analog neural networks are weight storage and multiplication, often referred to collectively as a synapse. Synapse implementations vary widely among reported analog neural networks. Weight storage can be accomplished by either analog or digital circuit techniques while multiplication is usually performed by a Gilbert multiplier.

Analog weight storage is typically implemented by storing charge on a capacitor. This charge must be refreshed periodically and therefore requires additional programming circuitry that is constantly operating. However, dynamic capacitive memories have often been favored by designers interested incorporating on-line, on-chip learning [1,2]. These analog memories have the advantage of small layout area, but they achieve this layout savings at the cost of increased power dissipation in the required refresh circuitry.

Floating-gate synapses offer an alternative to capacitive analog memory [3,4]. Floating-gate storage also boasts small layout areas, but requires special high-voltage programming circuitry on chip. Furthermore, additional feedback circuitry is usually required for accurate programming.

Digital weight storage has been less popular, but it has the advantage of a simple programming interface. Digital weights can be stored in all of the familiar digital memory structures: DRAM, SRAM, or EEPROM. Since all computation is performed in the analog domain, digital weights must be converted to analog signals through the use of DACs [5].

The Gilbert multiplier circuit is the most popular technique used in analog neural networks. Variations of the Gilbert multiplier have been proposed for various neural networks [5,6,7]. Other transconductance multiplication techniques have also been reported. In [2], an extended-range transconductance amplifier is used to multiply three quantities by each other. All of these multiplication circuits behave as expected only for limited signal swings; for large inputs, all of these circuits exhibit saturating nonlinearity.

We propose a new synapse circuit to be used in analog neural networks. Our synapse employs digital weight storage and gives a response that is linear over all possible input levels using a current-mode multiplying digital-to-analog converter (MDAC). MDAC synapses have the advantage of being able to solve the weight storage and multiplication problems using a single piece of hardware.

In Section 2 we describe the design of this MDAC synapse. In Section 3 we present linearity measurements taken for a large number of our synapses. In Section 4 we demonstrate the use of the synapse in a small analog neural network chip and compare measured to simulated results. We conclude the report in Section 5.

2. MDAC SYNAPSES

2.1. Previously reported MDAC synapses

To our knowledge, MDAC synapses have been reported twice before in the literature. Boser et al. employed a current-mode MDAC which multiplied a digital input by an analog weight [8]. The weight was stored on a

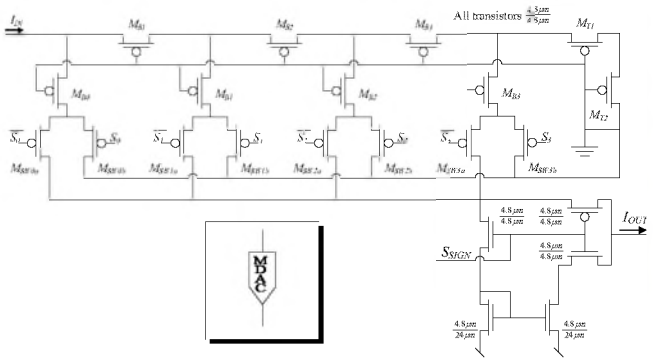

Figure 1 – Circuit diagram for a 5-bit *R*-2*R* pMOS MDAC and its symbol (inset).

capacitor which set the V_{GS} for a set of transistors with binary-weighted width-to-length ratios. The digital input was used to select which currents would sum to produce the output. More recently, a voltage-mode MDAC synapse that multiplies an analog differential voltage by a digital weight has been reported [9]. Unfortunately, this scheme occupies a prohibitively large amount of layout area (1.35-mm^2 in a 1.2-µm process).

2.2. Proposed MDAC synapse

Our proposed MDAC synapse is a compact current-mode circuit which multiplies an input current by a digital weight. The circuit diagram for our proposed synapse is shown in Fig. 1. The operation of the circuit is based on the familiar *R*-2*R* resistive current divider. Here, however, pMOS transistors are used in place of polysilicon resistors to save chip area.

In [10], Vittoz and Arreguit introduced the concept of a pseudo Ohm's law for MOSFETs. Simply stated, a network of MOSFETs sharing the same gate voltage is linear with respect to currents but not voltages. Further, the current through each transistor is determined only by its geometry. This allows one to borrow resistive current division networks and incorporate them directly into VLSI circuits without using large resistors.

Each transistor in Fig. 1 is drawn with identical width and length. The pseudo-resistance of a MOS transistor is determined only by its width-to-length ratio. We can denote the pseudo-resistance of each transistor as *R*. The specific value of *R* is not important. The 2*R* 'resistance' is provided by series combination of the switching transistor (M_{SWxa} *or* M_{SWxb}) and the branch transistor (M_{Bx}). Note that in each branching section, only one switching transistor is on at a time because the pair is driven with complementary signals. Therefore, each downward branch of the ladder provides a 'resistance' of 2*R* to ground. The current in each branch is switched to ground or onto the output node, which may not be at the same potential as the circuit ground. However, all that is required for MOS pseudo-resistor circuits is a large network voltage drop to keep the switch transistors M_{SWxy} in saturation. This

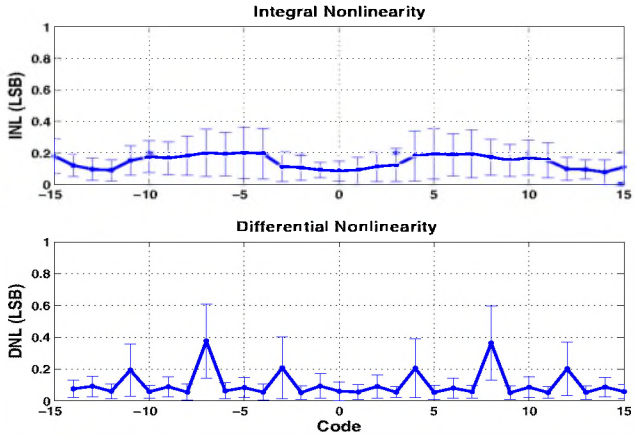

Figure 2 – Mean nonlinearity measures for 32 MDACs. Differential nonlinearity peaks at code changes from 7 to 8. This is caused by uneven current splitting at each stage of the *R*-2*R* structure.

concept is referred to as a 'pseudo-ground' [10]. This feature of MOS pseudo-resistive networks removes the need for an op-amp providing a virtual ground (which is required in resistive implementations).

Negative weights are realized by directing the output current of the MDAC into an nMOS current mirror, reversing the current flow. An additional set of switching transistors controlled by a digital input, S_{SIGN}, is used to direct the output of the MDAC either through an nMOS mirror or directly to the output node. The transfer characteristic for the 5-bit MDAC in Fig. 1 is given by

$$I_{OUT} = D \cdot I_{IN} \tag{1}$$

where

$$D = (-1)^{S_{SIGN}} \sum_{i=0}^{3} \frac{S_i}{2^i} \tag{2}$$

is the stored weight. It can be seen from (2) that weight magnitudes are always less than one. If larger weights are desired, additional current gain may be added before or after the MDAC circuit. This is most easily accomplished by increasing the size of the current mirror supplying I_{IN}.

The 5-bit MDAC described above occupies an area 90 µm by 60 µm (5400 µm^2 in a 1.5-µm process). The total synapse area depends on the choice of SRAM cells used to store the weight. Our initial implementation uses a SRAM cell based on a resettable D-latch. Consequently, the layout area required for weight storage is roughly five times the layout area for the MDAC. The area impact of weight storage circuitry will be reduced in future implementations by using a simpler SRAM cell.

This MDAC synapse has been incorporated into a four-neuron analog recurrent neural network chip. Each chip contains 20 MDAC synapses implemented in AMI's 1.5-µm two metal, two poly CMOS process through MOSIS. The design and operation of the neural network chip are described in Section 4.

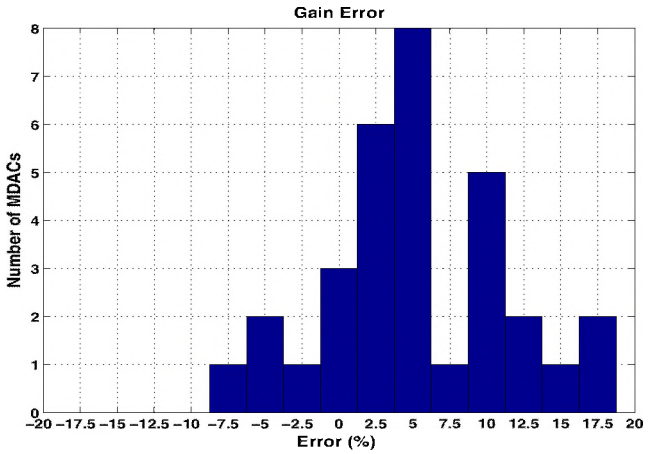

Figure 3 – Histogram of gain errors for 32 MDACs. The mean error is 5% and the maximum is 18.2%.

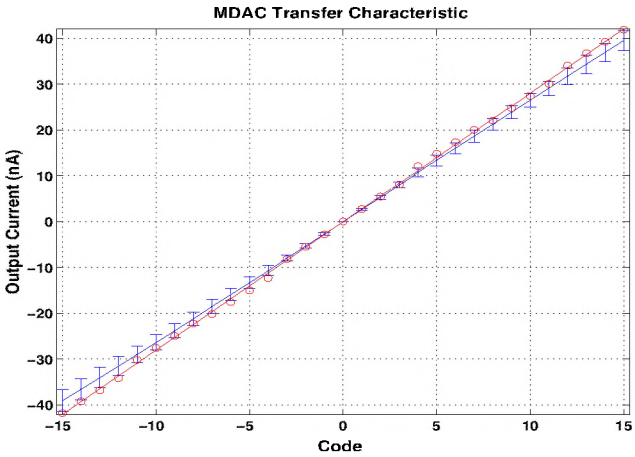

Figure 4 – Measured response for a randomly-chosen 5-bit MDAC. Also shown is the mean response of 32 MDACs.

3. CIRCUIT LINEARTIY

The response characteristics for 32 MDAC synapses were measured on two chips (16 synapses per chip). Standard linearity measures—integral nonlinearity (INL) and differential nonlinearity (DNL)—were computed for each MDAC. Fig. 2 shows plots of the average of the absolute value of the INL and DNL at each weight value. The error bars in Fig. 2 denote one standard deviation above and below the mean.

INL was computed as the difference between the least-squares linear fit and the measured data. Data was not normalized before computing INL; instead, a histogram of the gain errors is shown in Fig. 3.

The gain errors shown in Fig. 3 are not the result of the *R*-2*R* structure. Rather, the gain errors reflect device mismatch in three cascaded current mirrors through which input current must pass. Only the last current mirror, where gain is added, should be considered part of the MDAC circuit, but it was not possible to isolate every MDAC input during testing. Instead, a single master bias supplied the input current to all MDACs. As a result, large gain errors, which are not typical of the MDAC circuit, were measured.

The maximum INL was measured to be 0.7 LSB and the maximum DNL was 1.1 LSB. Only two MDACs had a maximum DNL value greater than 0.7 LSB. These linearity figures are not impressive when compared with those measured from general purpose DAC circuits. However, the layout area occupied by our DAC is only a small fraction of what more general purpose DACs require.

The mean response (with error bars denoting one standard deviation) of all 32 MDACs tested is shown in Fig. 4. Also shown in Fig. 4 is the response of a randomly-chosen MDAC along with its least-squares fit. The effects of the gain errors are clearly evident in this plot. The standard deviation grows large at the extremes of the input.

4. RECURRENT NEURAL NETWORK APPLICATION

We have fabricated a four-neuron chip which implements a continuous-time recurrent neural network (CTRNN). Each CTRNN neuron's behavior is described by the nonlinear differential equation

$$\tau_i \cdot \frac{dy_i}{dt} = -y_i + \sum_{j=1}^{N} w_{ij} \cdot \sigma\left(y_j + \theta_j\right) \tag{3}$$

where

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{4}$$

is the logistic sigmoid function [11].

The circuit diagram for a single CTRNN neuron is shown in Fig. 5. In this implementation, a subthreshold differential pair is used to realize the logistic sigmoid function. Its transfer function is described by

$$I_{out} = \frac{I_B}{1+e^{-\kappa V_w / U_T}} \tag{5}$$

where $U_T = kT/q \approx 26$ mV is the thermal voltage, $\kappa \approx 0.7$ is the gate coupling coefficient, and I_B is the bias current [12]. Our 5-bit MDAC synapses provide programmable connections between neurons. External *RC* networks give the neurons the dynamic behavior described by (3). The time constant, τ_i, in (3) is set by the product of *R* and *C*. The value of the resistance *R*, is fixed and given by

$$R = 4 \cdot \frac{U_T}{\kappa I_B} \tag{6}$$

This value corrects for the scaling introduced by (5) and provides an additional gain. The value of *C* is selected to give each neuron the desired time constant.

In this CTRNN implementation, we wish to use the network as a pattern *generator* as opposed to a pattern

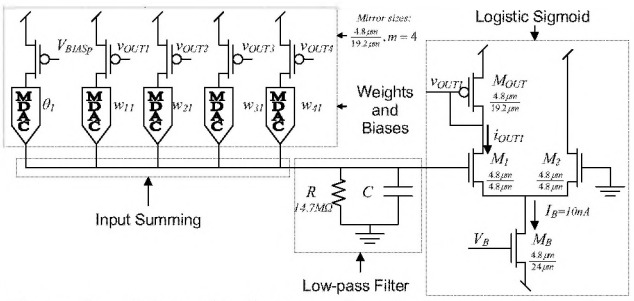

Figure 5 – Schematic diagram of a single neuron in a four-neuron CTRNN.

recognizer. Therefore, no external input connections to the neurons are available. The outputs of the neurons model the mean firing rate of motor neurons found most animal spinal cords. In animals, such temporal patterns are produced by central pattern generators [13], interconnected networks of nerve cells that autonomously generate rhythmic activity for behaviors such as walking and breathing.

4.1. CTRNN results

Fig. 6 shows a single cycle of the CTRNN chip output plotted on the same set of axes as the output of the simulated CTRNN. The network weights and biases were rounded to the nearest integer in the simulation to facilitate comparison. Time constants were selected using the nearest standard capacitance values.

The network shows good matching on neurons 1, 2, and 4. The output of neuron 3 does not match simulation as well, but the shape of the waveform is comparable to simulation. All of the on-chip neurons have amplitudes that differ from the simulation values, with neuron 3 showing the largest difference. This difference in amplitude is due to transistor mismatch on the output current mirrors.

5. CONCLUSIONS

We have presented a new synapse circuit design for use in analog neural networks. The synapse utilizes a current-mode multiplying DAC based on familiar *R*-2*R* circuits. The synapse is compact, requiring only 5400-μm^2 in a standard 1.5-μm CMOS process, and is linear over its entire input range. We have also shown that the synapse circuit is suitable for implementing continuous time recurrent neural networks and the behavior of such networks matches simulation well.

ACKNOWLEDGEMENTS

This work was supported by an NSF BITS award (EIA-0130773).

REFERENCES

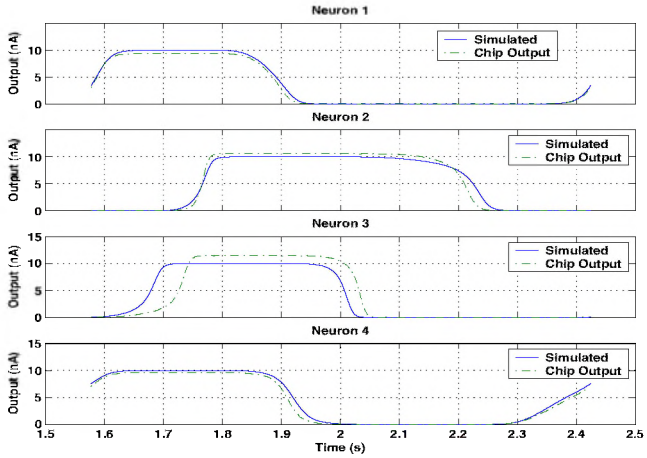

Figure 6 – Simulated vs. measured CTRNN patterns.

[1] A.J. Montalvo; R.S. Gyurcsik; J.J. Paulos, "Toward a general-purpose analog VLSI neural network with on-chip learning," *IEEE Trans. Neural Networks*, **8:**413-423, Mar. 1997.

[2] G. Cauwenburghs, "An analog VLSI recurrent neural network learning a continuous-time trajectory," *IEEE Trans. Neural Networks,* **7:**346:361, Mar. 1997.

[3] B.W. Lee, B.J. Sheu, and H. Yang, "Analog floating-gate synapses for general-purpose VLSI neural computation," *IEEE Trans. on Circuits and Systems* **38:**654 -658, 1991.

[4] D. A. Durfee and F. S. Shoucair, "Comparison of floating-gate neural network memory cells in standard VLSI CMOS technology," *IEEE Trans. Neural Networks*, **3:**347-353, May. 1992.

[5] S. M. Gowda; B. J. Sheu; J. Choi; C. G. Hwang; and J. S. Cable, "Design and characterization analog VLSI neural network modules," *IEEE J. Solid-state Circuits*, **28:**301-313, Mar. 1993

[6] B. Linares-Barranco; E. Sanchez-Sinencio; A. Rodriguez-Vazquez; and J. L. Huertas, "A modular T-mode design approach for analog neural network hardware implementations," *IEEE J. Solid-state Circuits*, **27:**701-713, May 1992.

[7] D. Coue and G. Wilson, "A four-quadrant subthreshold mode multiplier for analog neural-network applications," *IEEE Trans. on Neural Networks*, **7:**1212-1217, Sept. 1996.

[8] B. E. Boser; E. Sackinger; J. Bromley; Y. L. Cun; and L. D. Jackel, "An analog neural network processor with programmable topology," *IEEE J. Solid-state Circuits*, **26:**2017-2025, Dec. 1991.

[9] M. Al-Nsour and H. Abdel-Aty-Zohdy, "ANN digitally programmable analog synapse," *42nd Midwest Symp. On Circuits and Systems*, **1:**489-492, Aug. 1999.

[10] E. A. Vittoz and X. Arreguit, "Linear networks based on transistors,"*Electron. Lett.*, **20:** 297–299, 1993.

[11] R. D. Beer, "On the dynamics of small continuous-time recurrent neural networks," *Adaptive Behavior*, **3:**471-511, 1995

[12] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.

[13] F. Delcomyn, "Neural basis of rhythmic behavior in animals," *Science* **210:**492-498, 1980.