Programmable Synaptic Weights for an aVLSI Network of Spiking Neurons

Yingxue Wang and Shih-Chii Liu
Institute of Neuroinformatics
University of Zürich and ETH Zürich
Winterthurerstrasse 190, CH-8057 Zürich, Switzerland
yingxue, shih@ini.phys.ethz.ch

Abstract—**We describe a spiking neuronal network which allows local synaptic weights to be assigned to individual synapses. In previous implementations of neuronal networks, the biases that control the parameters of a particular synapse are global to all synapses of the same type regardless of the target neuron. In this new implementation, the parameters for a synapse are set by on-chip Digital-Analog-Converter (DAC) circuits, and the DACs are updated before the selected synapses are activated. Results from the fabricated chip show that the local weights are programmable and the DACs settle in the order of microseconds. These on-chip DACs allow the user to program a selected synaptic weight for connections between neurons and they can also be used for mismatch calibration.**

I. INTRODUCTION

Various aVLSI implementations of spiking multi-neuron arrays are currently being developed in several labs [1 – 7, 13]. The output spikes of the neurons are transmitted asynchronously off-chip in the form of a digital address unique to each neuron. This transmission is based on an asynchronous protocol called Address-Event-Representation (AER) [2, 3]. An AER infrastructure surrounding systems of spiking neuron chips allows the creation of connections between neurons on-chip and across chips. Questions surrounding event-based processing can be investigated using these chips [4 -7].

In many systems, parameters like the synaptic weights are controlled by global biases. Global synaptic weights, for example, will constrain the types of computation that can be studied with these spiking networks. Local synaptic settings can be added by using, for example, non-volatile technology, but this solution leads to bigger pixels and the necessity of a high-voltage power supply [8]. Another way of implementing a local synaptic weight is by using a spike burst encoding scheme where multiple spikes are transmitted to a synapse for every input spike [9]. This technique requires a large overhead in time and skews the timing of the original input spike. Yet another solution is to update an off-chip DAC that normally sets the global synaptic weight before each input spike but this method creates a finite latency (tens of microseconds to milliseconds) in the spike timing because of the settling time of the DAC.

In this work, we describe another solution for generating local synaptic weights by incorporating an on-chip global 5-bit current-mode digital-to-analog converter (DAC) for each synaptic weight parameter [10]. This circuit operates over 5 decades of current from 10pA to 1μA and produces a stable output within a few microseconds. The synaptic weight for each local synapse is set by using part of the input address space of the targeted synapse as the bits to the DAC. The on-chip DAC decodes these bits and its output sets the weight of the targeted synapse. This synapse is then stimulated after a delay which is set to the maximum settling time of the DAC.

The paper is structured as follows. Section II describes the 5-bit current-mode DAC and how it is used to drive the local synapses of each neuron. Section III describes test results from the fabricated chip.

II. CURRENT-MODE DIGITAL-TO-ANALOG CONVERTER

The global DAC and the use of it in driving the local synapses of each neuron are shown in Figure 1. The DAC itself is indicated by the special transistor symbol, MDAC (Figure 2). The input current to the DAC is generated by a mirrored version of the output of a reference current circuit (Figure 3). The DAC [10] circuit (shown in Figure 2) is based on the MOS current division technique which guarantees that the current is split by an octave at each branch of the DAC independent of the operation regime of the transistors [11]. The reference current circuit in Figure 3 comes from part of a bias generator circuit described in [12]. The reference current I_{ref} can be varied through the off-chip resistor R. By varying I_{ref} over 3 decades and by additionally changing the digital control word W = {b4 b3 b2 b1 b0} of the DAC, we can get a DAC output current that spans a total of 5 decades from about 10pA to 1μA.

0-7803-9390-2/06/$20.00 ©2006 IEEE

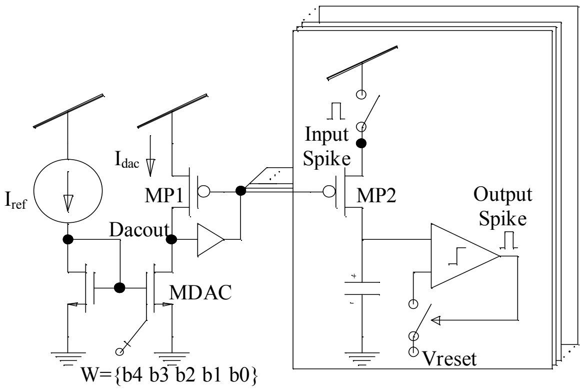

Figure 1. Global DAC driving the synaptic circuits of each of the integrate-and-fire neurons on the chip.

The output current of the DAC is summed at the transistor MP1 and is mirrored to the synaptic weight transistor MP2 of each synapse circuit (Figure 1). This synaptic current drives an integrate-and-fire neuron circuit which was previously described in [5].

An amplifier is added between the Dacout node and the gate of MP1 so that we can speed up the DAC response by isolating the Dacout node from the large capacitive load of the synapse fanout [12].

III. Experimental Results

A test chip was fabricated in a 4-metal 0.35um CMOS technology. It has 16 integrate-and-fire neurons, and two global 5-bit DACs which control the weights of the excitatory and inhibitory synapses. Figure 4 shows the layout of the DACs and the bias current generator. The size of the transistors in the current splitter is 2.4um/1.2um.

A. Measured DAC Outputs

We first describe the measured operating range of the fabricated DAC circuit. The first thing we verified was the monotonic behavior of the DAC, that is, the output current of the DAC should increase with a corresponding increase in the digital control word W. With an off-chip resistor of R=100kOhm, we get the staircase-like data with increasing W shown in Figure 5 as measured for 4 chips. The jumps at W = 8, 16, and 24 are caused by transistor mismatch and can be reduced by increasing the transistor sizes in the current splitter. As previously mentioned, the operating range of the DAC can be increased by changing the value of the off-chip resistor R. The plot in Figure 6 of the individual branch currents against different off-chip resistor values shows an almost reciprocal relationship [12]. From both Figures 5 and 6, we show how we can operate the DAC from about 10pA to 1µA by changing R and W. The dynamic response of the DAC was also tested as shown in Figure 7. The settling time of the DAC is around 0.4µsec for an output current I_{dac}=747nA, and 6µsec for I_{dac}=26.8nA. Simulations show that the settling time will increase to about 100 µsec for I_{dac}=10pA. However, we do not normally need such low currents for the operation of the aVLSI multi-neuron network.

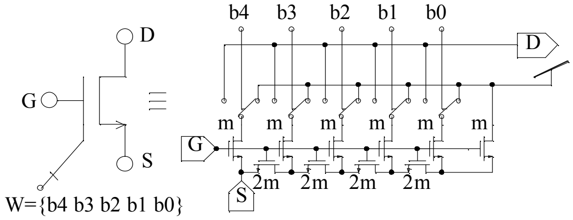

Figure 2. Voltage biased DAC based on the current splitter principle. The parameter, m, is the W/L ratio of the vertical transistors.

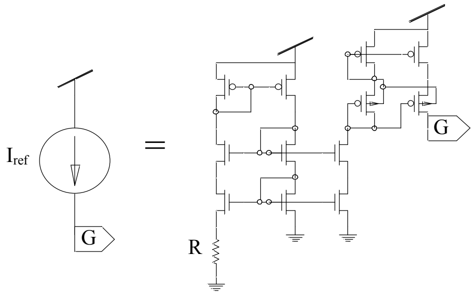

Figure 3. Circuit which creates a reference current I_{ref} that is tunable through resistor R [12].

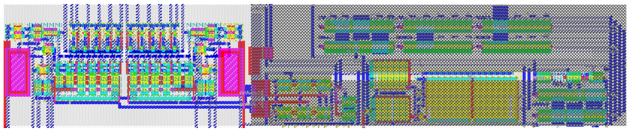

Figure 4. Layout of DACs with bias current generator

B. Mismatch Compensation

We now show how we use the DAC to compensate for mismatch in the synaptic weights. We first measured the excitatory postsynaptic potential (EPSP) at each neuron for all 32 values of the digital word W, by stimulating their corresponding synapses.

From these measurements (shown in Figure 8), we describe how we compensate for the mismatch using the DAC. For each neuron and each value of W, we find a new W that will result in an EPSP that is closest to the mean EPSP value of the original W for all neurons. This information is stored in a look-up table and is used to set the proper W every time a particular neuron is stimulated. In this way, the programming and the calibration are combined using one DAC. Figure 9 shows the distribution of EPSPs after mismatch calibration, and Figure 10 compares the coefficient of variation (CV) of EPSPs before and after calibration. The high CV at low W is due to the power supply noise which has comparable magnitude to the EPSP

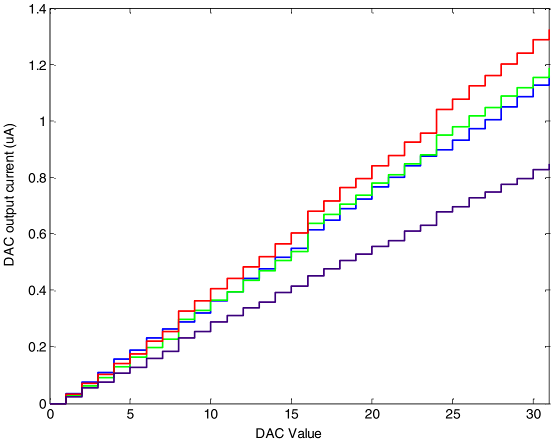

Figure 5. Measured current output of the DAC for different chips with an off-chip resistor R=100kOhm.

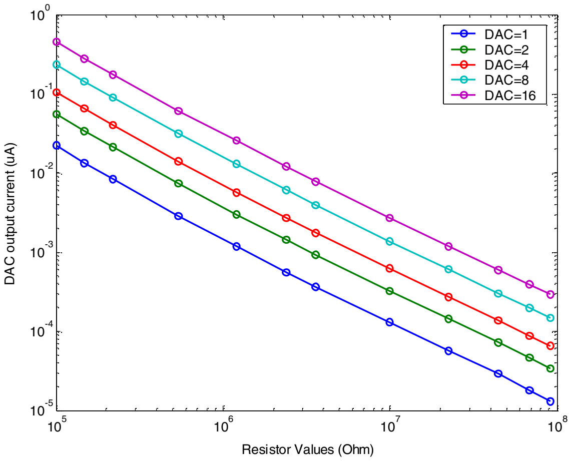

Figure 6. Current through each branch of current splitter of the DAC with resistor R sweeping from 100kOhm to 100MOhm.

itself. As can be seen in Figures 9 and 10, the variance of the EPSP drops by a factor of 10 for intermediate values of W. The increase in variance for high values of W can be explained by the dependence between the desired EPSP and the synaptic mismatch. When the desired EPSP is higher than the highest EPSP that a synapse can put out for W=31, we cannot compensate for the mismatch of that synapse through the DAC. Thus, the EPSP variance increases after the mismatch calibration. This problem can be mitigated by using two DACs for controlling the synaptic weight; one for programming and the second for calibration [13]. With the additional DAC, the calibration will not be restricted by the W from the programmed EPSP, and a better CV can be expected in this case.

C. *Synaptic Programming*

We show a simple example of how we used the DACs to program a particular synaptic weight profile. Figure 11(a) shows a predefined weight vector that follows a normal distribution. The spatial connectivity of an external neuron to the neurons on-chip is programmed with this weight vector. The input from the external neuron is a spike train with a regular spike rate of 100Hz. The spiking activity of the neurons (shown in Figure 11(b)) shows a similar profile to that of the weight vector.

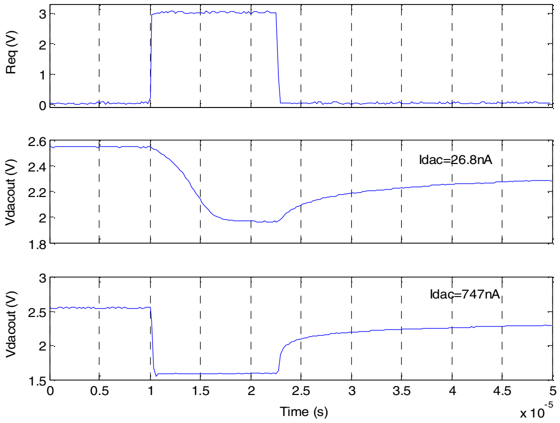

Figure 7. Settling time of the DAC for different I_{dac}. The top figure shows the timing of the signal (Req) that indicates the validity of the addresses to the DAC. The middle and bottom figures show the settling time of the Dacout node in Figure 3 for two values of I_{dac} (26.8nA, 747nA).

IV. Conclusion

In this paper, a 5-bit current-mode DAC is used to set the local synaptic weights on an aVLSI array of integrate-and-fire neurons. We show how we can use this global DAC for simultaneous mismatch compensation and unique weight programming for each local synapse.

Acknowledgment

The authors would like to thank M. Oster for help with setting up the AER infrastructure, S. Zahnd for the PCB design, and T. Delbrück for discussions on the bias current generator circuits. This work is partially supported by ETH Research Grant TH-20/04-2.

References

[1] S.-C. Liu, "Analog VLSI circuits for short-term dynamic synapses", EURASIP Journal on Applied Signal Processing, Special Issue on Neuromorphic Signal Processing and Applications, editors A. van Schaik and S. Shamma, vol. 7, pp. 1-9, 2003.

[2] K. A. Boahen, "Point-to-point connectivity between neuromorphic chips using address events", IEEE Transactions on Circuits and Systems II, vol. 47, pp. 416-434, 2000.

[3] S. R. Deiss, R. J. Douglas, and A. M. Whatley, "A pulse-coded communications infrastructure for neuromorphic systems", in "Pulsed Neural Networks", W. Maass, C. M. Bishop, Eds. Boston, MA: MIT Press, Ch. 6, pp. 157-178, ISBN 0-262-13350-4, 1999.

[4] S.-C. Liu, and R. Douglas, "Temporal coding in a network of silicon integrate-and-fire neurons", IEEE Transactions on Neural Networks: Special Issue on Temporal Coding for Neural Information Processing, vol 15, no 5, Sep., pp. 1305-1314, 2004.

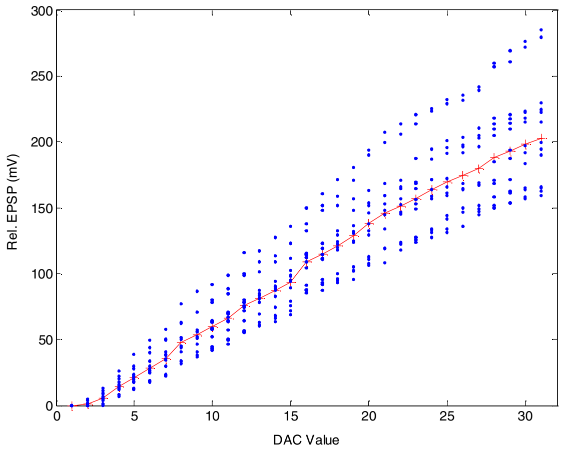

Figure 8. Measured EPSP (blue dots) of each neuron and the mean of EPSPs for each W (red crosses) for every DAC value from 0 to 31.

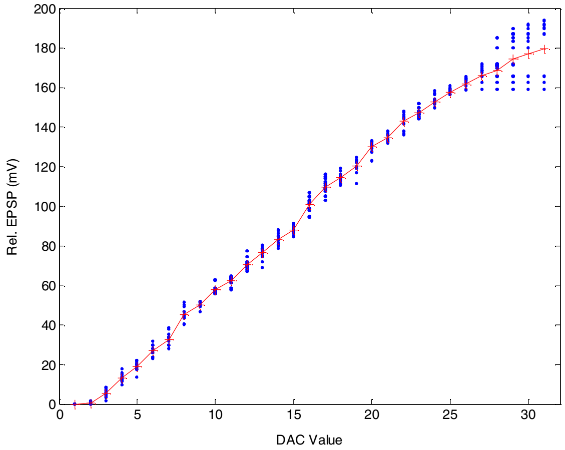

Figure 9. Measured EPSP (blue dots) of each neuron and the new mean of EPSPs for each W (red crosses) after mismatch calibration .

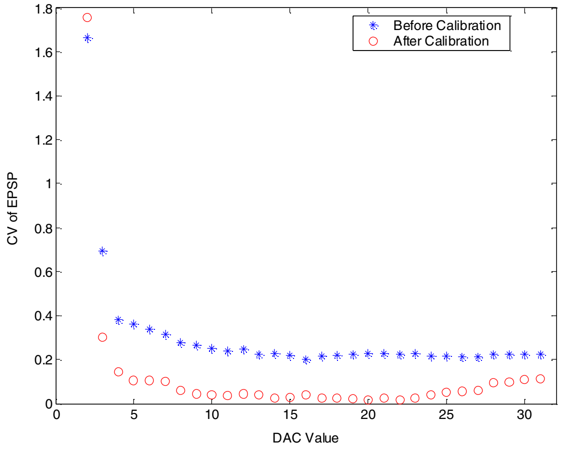

Figure 10. Coefficient of variation (CV) of EPSP before (blue asterisks) and after (red circles) mismatch calibration.

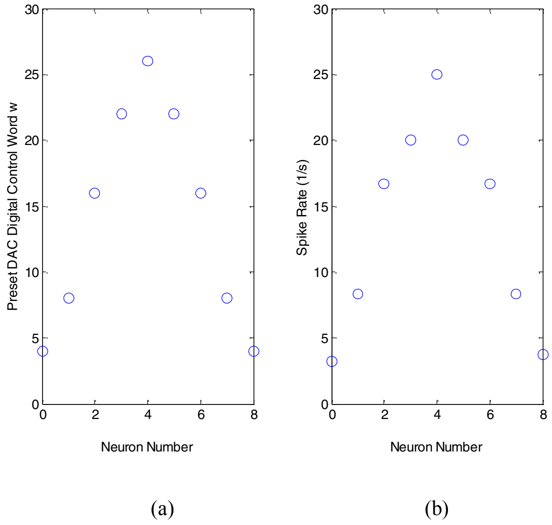

Figure 11. Programming the DACs for a specific spatial firing pattern. (a) Predefined digital control DAC word W which follows a normal distribution is used to set the spatial connectivity of an external neuron to the neurons on-chip. (b) The spike rate of the neurons when a regular spike train with 100Hz is fed to each neuron by the external neuron.

[5] S.-C. Liu, J. Kramer, G. Indiveri, T. Delbrück, T. Burg, and R. Douglas, “Orientation-selective aVLSI spiking neurons”, Neural Networks: Special Issue on Spiking Neurons in Neuroscience and Technology, vol. 14, pp. 629-643, 2001.

[6] G. Indiveri, T. Horiuchi, E. Niebur, and R. Douglas, “A competitive network of spiking VLSI neurons”, World Congress on Neuroinformatics, Vienna, Austria, 2001.

[7] Goldberg, D. H. and Cauwenberghs, G. and Andreou, A., “Analog VLSI spiking neural network with address domain probabilistic synapses”, Proceedings of the 2001 IEEE International Symposium on Circuits and Systems, vol II, pp. 241-244, May. 2001.

[8] R. R. Harrison, J. A. Bragg, P. Hasler, B. A. Minch, and S. P. Deweerth, “A CMOS programmable analog memory cell array using floating-gate circuits”, IEEE Trans. Circuits Syst. II, vol. 48, pp. 4–11, Jan. 2001.

[9] M. Oster and S-C Liu, “A Winner-Take-All Spiking Network With Spiking Inputs”, 11th IEEE International Conference on Electronics, Circuits and Systems, 2004.

[10] B. Linares-Barranco, T. Serrano-Gotarredona , and R. Serrano-Gotarredona, “Compact Low-Power Calibration Mini-DACs for Neural Massive Arrays with Programmable Weights”, IEEE Trans. on Neural Networks, vol. 14, No. 4, pp. 1207-1216, Sep.2003.

[11] K. Bult and GJGM Geelen, “An inherently linear and compact MOST-only current division technique”, IEEE J. Solid-State Circuits, vol. 27, pp. 1730–1735, Dec. 1992.

[12] T. Delbrück and A. van Schaik, “Bias Current Generators with wide dynamic range”, Analog Integrated Circuits and Signal Processing, vol . 43, pp. 247-268, 2005.

[13] B. Linares-Barranco, T. Serrano-Gotarredona, and R. Serrano-Gotarredona, “A New Charge-Packet Driven Mismatch-Calibrated Integrate-and-Fire Neuron for Processing Positive and Negative Signals in AER based Systems”, Proccedings of the 2004 IEEE International Symposium on Circuits and Systems, vol V, pp. 23-26, May.2004.