Pulse Duty Cycle Neural Processing Element Applied to Autotracking Model

G. Moon+, M. E. Zaghloul#, R. W. Newcomb* J. S. Yoo+

+Department of Electronic and
Control Engineering
Hallym University
Choonchun, Kwangwondo 200-702
Korea
e-mail: gmoon@sun.hallym.ac.kr

#Electrical Engineering and
Computer Science Department
George Washington University
Washington, DC 20052 USA
phone: (202) 994-3772

*Microsystems Laboratory
Electrical Engineering Department
University of Maryland
College Park, MD 20742 USA
phone: (301) 454-6869

Key Words: Neural Networks, Pulse Modulation, Autotracking

Abstract

Pulse Duty Cycle is introduced as a generic tool to achieve a synaptic weight multiplication. In this scheme, a simple change of the pulse duty cycle will change the synaptic junction weight. The electronic neuron is named Neural Processing Element (NPE) which uses the pulse duty cycle modulation technique for weight changing. This scheme is used in developing an Autotracking CMOS circuit. Simulations verify the pulse mode weighting operation, and the operation of the autotracking circuit. A CMOS prototype chip was designed and fabricated.

NEURAL PROCESSING ELEMENT WITH PULSE DUTY CYCLE SYNAPTIC WEIGHTING

An electronic Neural Processing Element (NPE) was developed in [1, 2] and was used for developing two Winner-Take-All models in [3]. The structure of the NPE was explained in [2]. It is composed of four functional blocks: Modified Neural Type Cell (MNTC), summation, threshold logic, and optional learning block as shown in Fig. 1. The synaptic weighting in the MNTC and the summation was processed using Pulse Duty Cycle Modulation (PDCM) technique [2], which allows us to achieve the CMOS implementation in small silicon real-estate. In the PDCM technique, the Pulse Duty Cycle (PDC) over an arbitrary time interval of t is defined as a temporal average of a pulse stream as

$$\mathrm{PDC(t)} = \frac{\int_0^t \mathrm{P(t)dt}}{\mathrm{t}} \qquad (1)$$

where P(t) is pulse stream voltage in time.

Given a pulse stream with a fixed amplitude of 5 volts being normalized as 1, we can simplify the above eq. (1) as

$$\mathrm{PDC(t)} = \frac{\sum_{j=1}^{m} \mathrm{PW(j)}}{\mathrm{t}} \qquad (2)$$

where PW(j) is the j-th pulse width in time in the stream, assumed to be of m pulses in time interval t.

We found that the PDC is monotonically controlled by either input (X in Fig. 1) or weight (W), and that the analog output (Y_A) can be controlled through the PDC [2]. This analog output (Y_A) is the post-synaptic junction output representing the weighted multiplication between the input and the weight [2]. Using the PDC as an information variable in the proposed design, we could realize both excitatory and inhibitory functions at the same synaptic junction [2].

For the threshold logic, a simple differential pair circuit in Fig. 2 was adopted, with which we could engineer the sigmoid function for comparison between the current output and the threshold value (Y_A and θ, respectively, in the Fig. 1). The dynamic behavior of the differential stage is well known and SPICE simulations for the circuit of Fig. 2 is shown in Fig. 3.

AUTOTRACKING MODEL

An autotracking model, or sometimes called a target tracking model [4], is to track a given pattern in the time domain through adaptive learning for weights. This scheme can usually be realized by adjusting the weight in proportion to the difference between the desired (target) and computed values as [4]

$$\Delta W = \alpha\,(Y_t - Y_A) \qquad (3)$$

where Y_t is the desired target value, Y_A is the current output, and α is the learning rate, typically $0<\alpha<<1$.

If the output is less than the target value, ΔW will have a positive value yielding an increase of the output. On the other hand, if the output is larger than the target value, ΔW becomes negative, which will

This work was supported by Korean Science and Engineering Foundation.

0-7803-2570-2/95 $4.00 ©1995 IEEE

force the output to be decreasing. When the output is the same as the target, ΔW sets at zero and no further changes occur. This logic enables the structure to be an autotracking mode machine. However, from a VLSI realization point of view, the above learning rule has some drawbacks in a sense that we need other auxiliaries (thus, a large real silicon estate overload): subtracter, multiplier and storage devices for storing ΔW which is to be iteratively added on the previous weight value (W).

In this work, a new and simple technique for the autotracking model is proposed. Instead of adopting the iterative learning rule of eq. (3), we devise a configuration such that the weight value is directly manipulated from the error signal, and, consequently, we call this technique as Direct Autotracking Technique (DAT). To be short, we use 'the other output' of the differential amplifier for the adaptive weight signal. This signal (W in Fig. 2) will be fed back to the MNTC and accomplishes the autotracking dynamics. Analytical verification for DAT is given in the following section.

DIRECT AUTOTRACKING TECHNIQUE (DAT)

Let W be the weight parameter associated with the neural network model shown in Fig. 1. The weight W in the CMOS circuit diagram of Fig. 2 is expressed in terms of the variable V_D as [2]

$$W = Vdd - Vtp - \frac{2}{K_3}\left[\frac{I_{ss}}{2} + \frac{I_{ss}}{2}\left(\frac{K_1 V_D^2}{I_{ss}} - \frac{K_1 V_D^4}{4 I_{ss}}\right)^{\frac{1}{2}}\right] \quad (4)$$

where $V_D = Y_A - Y_t$, which is a signed difference between the present output (Y_A) and the desired target output (Y_t). Vdd and Vtp are power supply (5V) and the threshold voltage of the P-type transistor, respectively, and Ki is i-th transistor gain factor in the differential amplifier [2]. Iss is a current source.

Differentiating the above eq. (4) with respect to V_D will yield

$$\frac{\partial W}{\partial V_D} = \left(-\frac{K_1}{K_3}V_D + \frac{K_1^2}{I_{ss}}V_D^3\right)\left(\frac{K_1 V_D^2}{I_{ss}} - \frac{K_1^2 V_D^4}{4 I_{ss}}\right)^{-\frac{1}{2}} \quad \text{for } V_D > 0 \quad (5)$$

$$\frac{\partial W}{\partial V_D} = \left(\frac{K_1}{K_3}V_D - \frac{K_1^2}{I_{ss}}V_D^3\right)\left(\frac{K_1 V_D^2}{I_{ss}} - \frac{K_1^2 V_D^4}{4 I_{ss}}\right)^{-\frac{1}{2}} \quad \text{for } V_D < 0 \quad (6)$$

And if V_D=0, from the above eq. (4), we have

$$\frac{\partial W}{\partial V_D} = 0 \quad (7)$$

\

Using the MOSIS parameters along with the circuit sizes in [2], the gradient of the weight can then figuratively be drawn as in Fig. 4. The dark line shows the valid lines. As can be seen, in the possible range of V_D, -5V ~ 5V, with a 5V power supply, the gradient is always negative except V_D=0 where the gradient is zero. Thus, summarizing the eqs. (4) - (7), we have

$$\left\{\begin{array}{l} \dfrac{\partial W}{\partial V_D} < o, \text{ for } V_D \neq 0 \\ \dfrac{\partial W}{\partial V_D} = o, \text{ for } V_D = 0 \end{array}\right\} \quad (8)$$

Rewriting the eq. (8) in incremental form will yield

$$\left\{\begin{array}{l} \dfrac{\Delta W}{\Delta(Y_A - Y_t)} < o, \text{ for } Y_A \neq Y_t \\ \dfrac{\Delta W}{\Delta(Y_A - Y_t)} = o, \text{ for } Y_A = Y_t \end{array}\right\} \quad (9)$$

If the system is at $Y_A = Y_t$, then $\Delta W = 0$, and no changes of the weight will occur. The model will stay in this stable state unless other external changes are given. Suppose the target Y_t is now increasing from the stable point, above the present output value Y_A, the denominator will become negative (see eq. (9)) and this will 'force' the nominator to become a positive value, $\Delta W > 0$, under the inequality constraint above. Thus, the weight will be increasing, which then will result in the increase of the output Y_A. If, on the other hand, the target is decreasing, the same logic above can be applied only with opposite directions, and the output will also be decreasing. Here at this moment, it should be pointed out that the reasoning above is a necessary condition for the autotracking neural model but is not yet a sufficient one, which is currently being under investigations. However, with this simple configuration, we could succeed to witness the autotracking dynamics as explained in the next section.

SIMULATION RESULTS AND VLSI DESIGN

Figure 5 shows a CMOS circuit diagram for the autotracking model and simulation results are given in Fig. 6. As seen, the output (Y_A) adaptively tracks the varying target (Y_t) with ±0.3V error. Notice that the weight (W in Fig. 2) is adaptively controlled and also that this DAT has 'fully asynchronous and analog' behavior (there is no system clocks). Thus, the maximum speed will be restricted only by design parameters as well as device physics.

VLSI design was done with given a set of MOSIS parameter and layout of the chip is shown in Fig. 7.

CONCLUSIONS

Analog CMOS design for an Autotracking model with the proposed NPE is introduced. Instead of employing the conventional iterative learning rule, which necessitates many other auxiliary, we adopt, in this work, a simple direct adaptive autotracking technique (DAT). In the proposed configuration, since the other output of the differential amplifier in the threshold block is used, the overall structure becomes very simple with no other auxiliaries. Besides its simplicity, the dynamics of the proposed design are based on the two most peculiar features of the biological neural systems: Analog and Asynchronous. Simulations verify the proposed design.

Future works are on to widen the dynamic tracking range and to enhance the accuracy as well as the speed.

REFERENCES

[1] G. Moon, M. E. Zaghloul, and R. W. Newcomb, " An Improved Neural Processing Element Using Pulse-Coded Weights," Proc. IEEE ISCAS, Chicago, pp. 2760-2763, May 1993.

[2] G. Moon, "VLSI Design of Neural Networks Using Pulse Coded Weights With On-Chip Learning Capability," Ph.D. Thesis, Department of Electrical Engineering and Computer Science, The George Washington University, Washington, DC, May 1993.

[3] G. Moon, M. E. Zaghloul, and R. Newcomb, " CMOS Design of Two Winner-Take-All Circuits Using Pulse Duty Cycle Synaptic Weighting," Proc. IEEE ISCAS, London, pp. 6.379-6.382, June 1994.

[4] P. K. Simpson, Artificial Neural Systems: Foundations, Paradigms, Applications, and Implementations, Pergamon Press, San Diego, 1990.

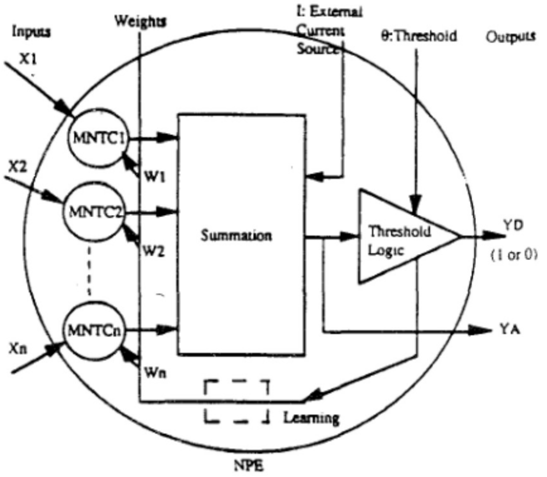


Fig. 1. Functional block diagram of a Neural Processing Element (NPE).

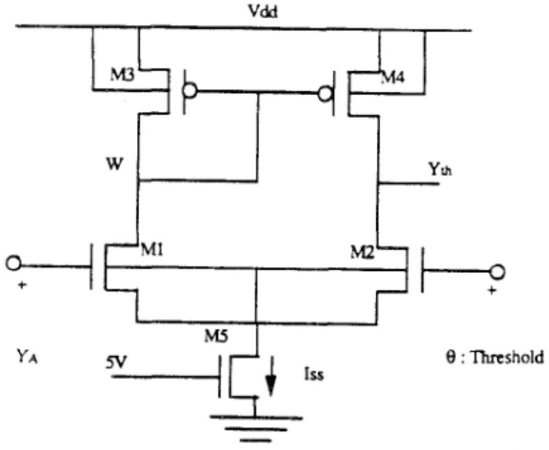
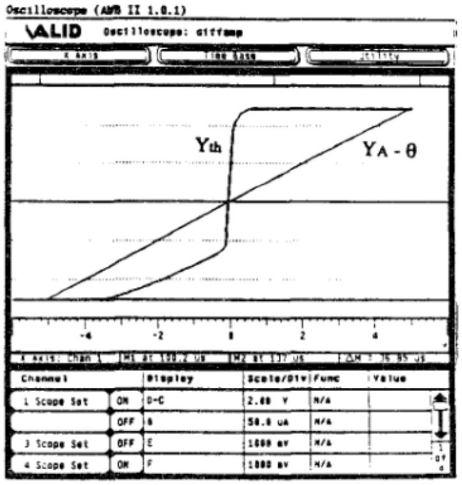


Fig. 2. CMOS differential amplifier for a sigmoidal threshold function.



Channel		Display	Scale/Div	Func	Value
1 Scope Set	ON	D-C	2.00 V	N/A	
	OFF	B	50.0 uA	N/A	
3 Scope Set	OFF	E	1000 mV	N/A	
4 Scope Set	ON	F	1000 mV	N/A	

Fig. 3. SPICE simulation result of Fig. 2 (2V/ vert.)

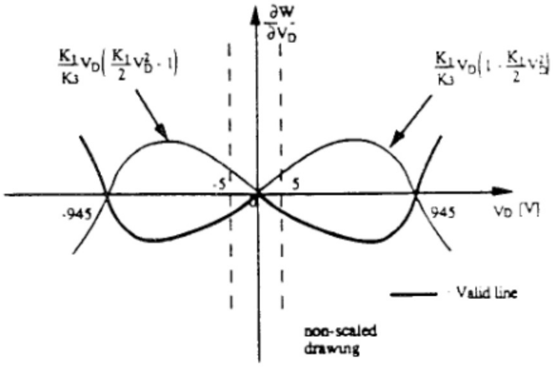


Fig. 4. The gradient of the weight with respect to the error signal (V_D).

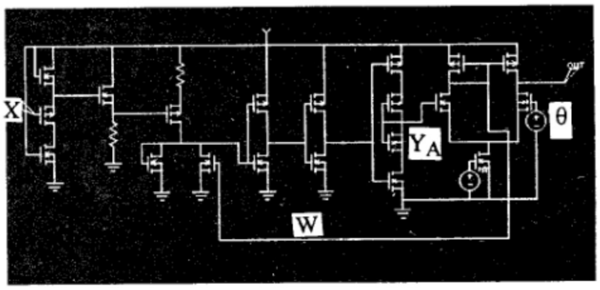


Fig. 5. CMOS circuit diagram for the autotracking model.

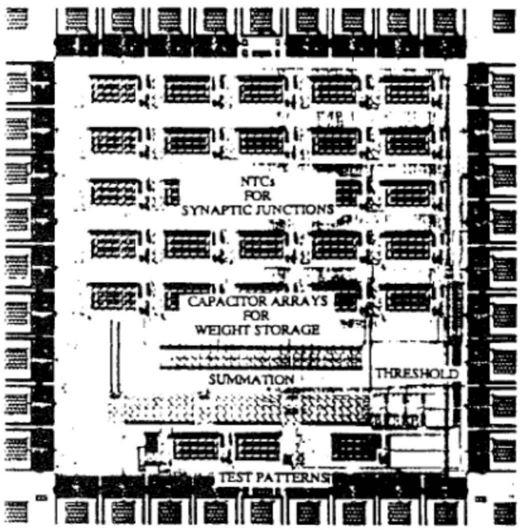
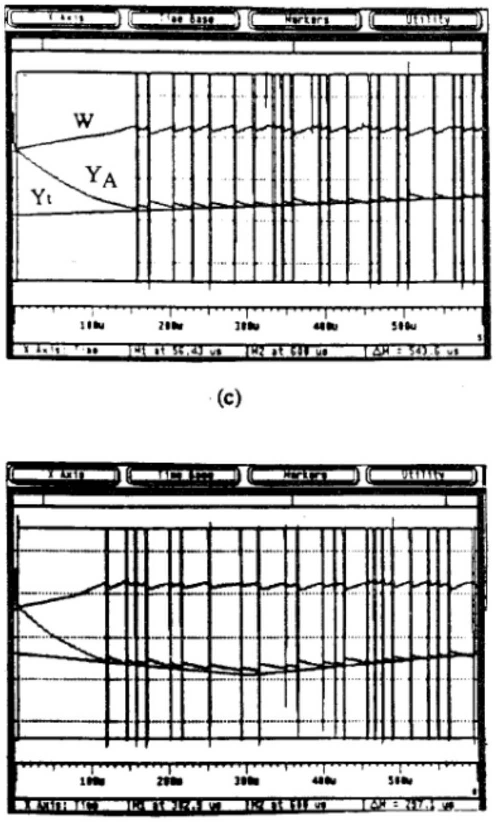


Fig. 7. Chip Layout (1.7 x 1.7 mm^2).



(c)

(d)

1V/vert.

Fig. 6. Simulation results for the autotracking model.
(a) with a ramp function target (1.5V to 2V)
(b) with a V-shape target