Neuro-Fuzzy Architecture for CMOS Implementation

Bogdan M. Wilamowski, *Senior Member, IEEE* Richard C. Jaeger, *Fellow, IEEE*, and M. Okyay Kaynak, *Senior Member, IEEE*

***Abstract*— In this paper, a nonconventional structure for a "fuzzy" controller is proposed. It does not require signal division, and it produces control surfaces similar to classical fuzzy controllers. The structure combines fuzzification, MIN operators, normalization, and weighted sum blocks. The fuzzy architecture is implemented as a VLSI chip using 2-μm n-well technology. A new fuzzification circuit, which requires only one differential pair per membership function is proposed. Eight equally spaced membership functions are used in the VLSI implementation. Simple voltage MIN circuits are used for rule selection. A modified Takagi–Sugeno approach with normalization and weighted sum is used in the defuzzification circuit. Weights in the defuzzifier are digitally programmable with 6-bits resolution.**

***Index Terms*—Control, fuzzy, VLSI.**

I. Introduction

THE analog domain is an attractive alternative for nonlinear signal processing, providing parallel processing with a speed limited only by the delay of signals through the network. However, the classical approach to fuzzy systems presented by Zadeh [14] is difficult to implement in analog hardware. Particularly difficult is the defuzzifier where signal division must be implemented. Division can be avoided through use of feedback loops [5], [6], [9], [12], [13], but this approach can lead to limited accuracy and stability problems. Also, in the case of the classical fuzzy system shown in Fig. 1, the information about the required control surface is encoded in three places: in the fuzzifier, in the defuzzifier, and in the prewired connections between MIN and MAX operators. Although the architecture is relatively simple, it is not suitable for custom programming.

The concept of the proposed circuit is shown in Fig. 2. Fuzzification is done in a traditional manner with additional normalization which leads to a linear interpolation of the output between stored points. The second stage is an array of cluster cells with fuzzy "AND" operators. Instead of classical defuzzification, simplified Takagi–Sugeno singleton inference rules [12] with normalization are used. The output is then normalized and calculated as a weighted sum of the signals approaching from all selected areas.

Manuscript received April 9, 1997; revised May 8, 1998. Abstract published on the Internet August 20, 1999.

B. M. Wilamowski is with the Department of Electrical Engineering, University of Wyoming, Laramie, WY 82071-3295 USA (e-mail: wilam@uwyo.edu).

R. C. Jaeger is with the Department of Electrical Engineering, Auburn University, Auburn, AL 36849-5201 USA (e-mail: jaeger@eng.auburn.edu).

M. O. Kaynak is with the Department of Electrical Engineering, Bogazici University, Bebek, 80815 Istanbul, Turkey.

Publisher Item Identifier S 0278-0046(99)08478-6.

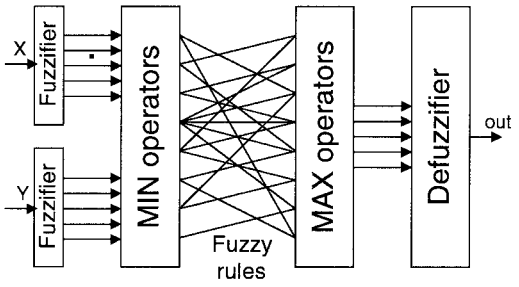


Fig. 1. Classical Zadeh-type fuzzy controller.

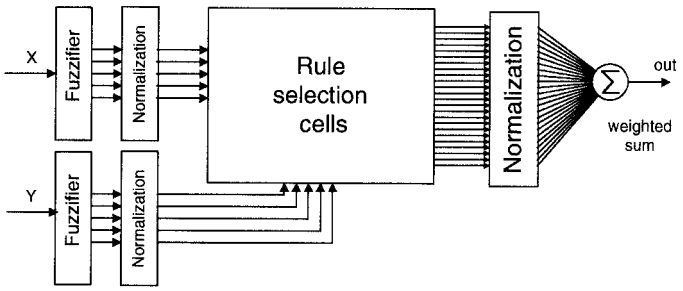


Fig. 2. Architecture of fuzzy controller suitable for VLSI implementation.

II. Fuzzifier

Various fuzzifier circuits that can be implemented in bipolar or MOS technology have already been proposed [1], [3], [7], [9]. Most approaches use two source- or emitter-coupled differential pairs for a single membership function. The approach proposed here differs from the previous techniques in two ways: 1) it is simpler—only one differential pair is required per membership function and 2) the fuzzy outputs are automatically normalized; therefore, the sum of all the signals representing the fuzzy variables of a single input is constant.

The fuzzifier circuit is presented in Fig. 3. This design requires only one differential pair for each membership function, in contrast to earlier designs, where at least two differential pairs were required [2], [4], [6]. Also, the output currents are automatically normalized because the sum of I_1 through I_6 is always equal to I_O. Thus, the normalization circuit is integrated within the fuzzifier. Using a simple Shichman–Hodges MOS transistor model [10] for strong inversion, the output currents for membership functions in midrange are given by

$$I_i = I_O\left[1 + 0.5\left(\alpha_i\sqrt{2\beta_i - \alpha_i^2\beta_i^2} - \alpha_{i+1}\sqrt{2\beta_{i+1} - \alpha_{i+1}^2\beta_{i+1}^2}\right)\right] \quad (1)$$

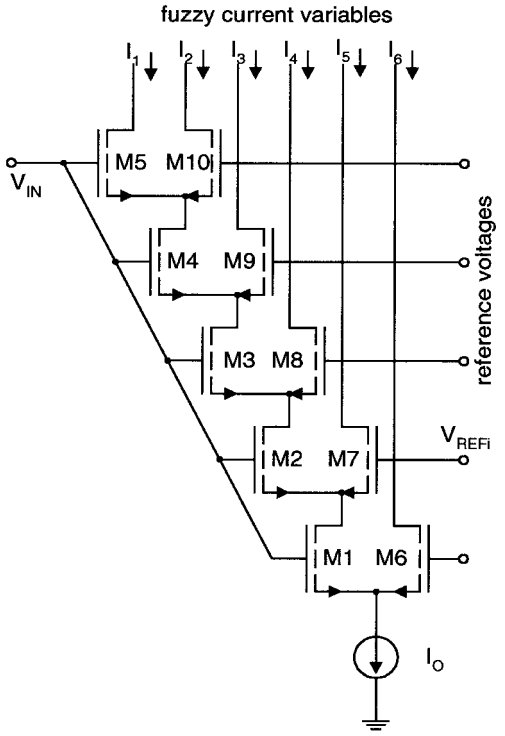


Fig. 3. Fuzzifier circuit with five differential pairs creating five membership functions: three Gaussian-like in the center and two sigmoidal types at the ends of the input range.

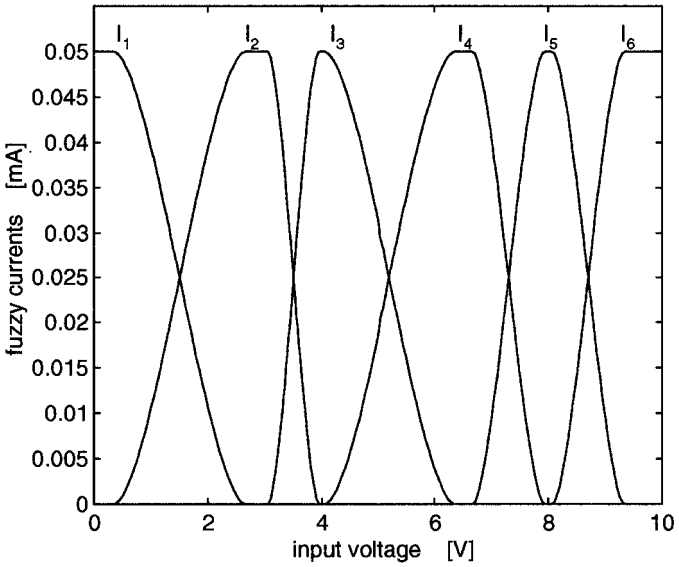


Fig. 4. An example of six output currents from a fuzzifier with five differential pairs obtained using SPICE simulation with the following parameters: $K = 24\ \mu\text{A/V}^2$, $V_{\text{TH}} = 0.75$ V, $I_O = 0.05$ mA, and W/L for transistors $M1$–$M5$ and $M6$–$M10$ are 3, 20, 3, 10, and 10.

while on the ends of the input range the above equation simplifies to

$$I_i = 0.5 I_O\left(1 + \alpha_i \sqrt{2\beta_i - \alpha_i^2 \beta_i^2}\right) \tag{2}$$

where

$$\alpha_i = \frac{V_{\text{IN}} - V_{\text{REF}i}}{V_{\text{TH}}} \quad \beta_i = \frac{KW_i V_{\text{TH}}}{2L_i I_O} \tag{3}$$

and I_O is the source current, V_{IN} is the input voltage, K is the MOS transistor transconductance parameter, V_{TH} is the MOS transistor threshold voltage, $V_{\text{REF}i}$ are reference voltages, and W_i and L_i are widths and lengths of the transistor channels. There are six output currents for a fuzzifier containing five differential pairs, as shown in Fig. 3. Examples of these currents for arbitrarily chosen W/L ratios are shown in Fig. 4. Note the wide range of flexibility in adjusting shapes and slopes by choosing proper W/L ratios in reference and differential pair transistors.

By choosing different reference voltages and different W/L ratios, any trapezoidal type of membership function can be accomplished. The circuit can also create Gaussian-type shapes of membership functions. See current I_5 in Fig. 4 for example. The corners of the trapezoids are rounded by parabolic functions if the transistors operate in the strong inversion mode, or by exponential functions if they operate in the weak inversion mode. Even though the circuit is very flexible, equally spaced symmetrical membership functions were chosen in VLSI implementation presented here.

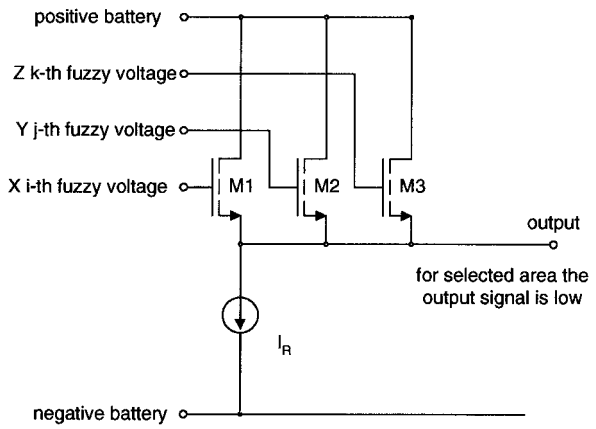


Fig. 5. Rule selection circuit—AND/MIN operator for inverted signal convention.

III. Array of Rule Selection Circuits

Each rule selection circuit is connected to one fuzzy variable from each fuzzifier. Therefore, the number of these circuits is equal to $n_1 * n_2$, where n_1 and n_2 are numbers of fuzzy variables for each fuzzifier. The rule selection circuit cell is activated only if both fuzzy inputs have nonzero values. Due to the specific shapes of the fuzzifier membership functions, where only two membership functions can overlap, a maximum of four cluster cells are active at a time. Although current mode MIN and MAX operators are possible [3], [5], it is much easier to convert currents from the fuzzifiers into voltages and use the simple rule selection circuits shown in Fig. 5. The rule selection circuit requires the fuzzy conjunction (AND) or

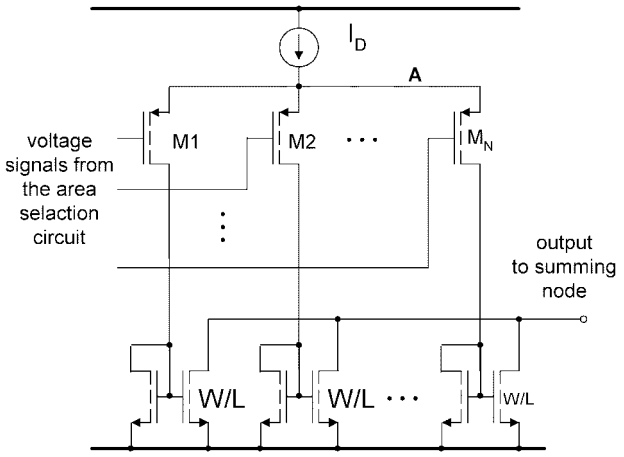


Fig. 6. Normalization circuit.

fuzzy MIN operator. It is implemented using source follower circuits shown in Fig. 5. The voltage on the common node of all sources always follows the highest potential of any of the transistor gates, so it operates as a MAX/OR circuit. However, using the negative signal convention (lower voltages for higher signals), this circuit performs the MIN/AND function. This means that the output signal is low only when all inputs are low. A cluster is selected when all fuzzy signals are significantly lower than the positive battery voltage. Selectivity of the circuit increases with larger W/L ratios. Transistor $M3$ would be required only if three fuzzifier circuits were used with three inputs.

IV. Normalization Circuit

In order to obtain proper outputs, it is essential that normalization occurs before weights are applied to the summed currents. The normalization circuit can be implemented using the same concept as the rule selection circuit. For the negative signal convention, PMOS transistors supplied by a common current source are required. The normalization circuit is shown in Fig. 5. The voltage on the common node *A* follows the lowest input potential. The normalization circuit consists of transistors $M1, M2, \cdots, M_N$ connected to a single common current source. This means that the sum of all drain currents of transistors $M1, M2, \cdots, M_N$ is always constant and equal to I_D. The W/L ratios in current mirrors can determine the output value for each cluster. Currents from all cluster cells are summed to form the output current.

V. Weight Circuit

The weights for different clusters can be assigned by setting proper W/L ratios for current sources on Fig. 6. This simple approach can be used only when a dedicated fuzzy controller is designed on the mask level. The purpose of this study was to develop a digitally programmable fuzzy controller for any required control surface. This task can be accomplished by introducing a digitally programmable current mirror shown in Fig. 7.

The cascode current mirror ($M1$–$M4$) splits its input current with a 4-to-1 ratio using different W/L ratios for the two

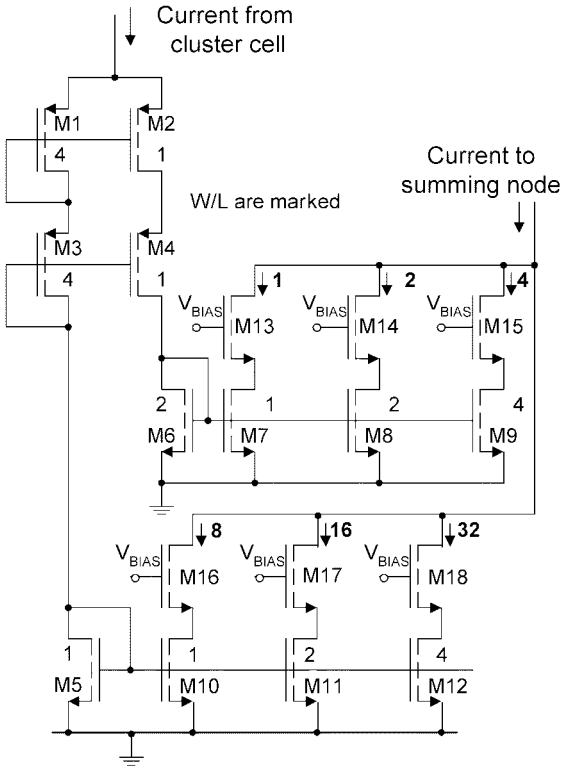


Fig. 7. Weight circuit—digitally programmable current source with 6-bits accuracy.

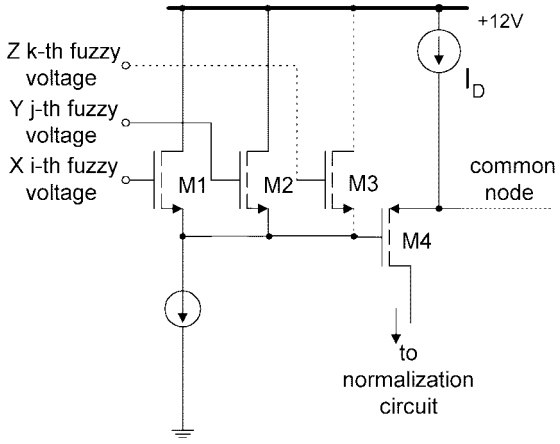


Fig. 8. The cluster cell with rule selection (transistors $M1$–$M4$) and normalization (source I_0 and transistors $M4$–$M6$).

current branches. Transistors $M5$–$M12$ form two multiple output current mirrors with current multiplication factors being 1, 2, 4, 8, 16, and 32 dependent on the W/L ratio chosen. Transistors $M13$–$M16$ work as digitally controlled current switches.

VI. VLSI Implementation

The rule selection and normalization circuits shown in Figs. 5 and 6 would require a prohibitive number of interconnections, and they are not practically realizable on a VLSI

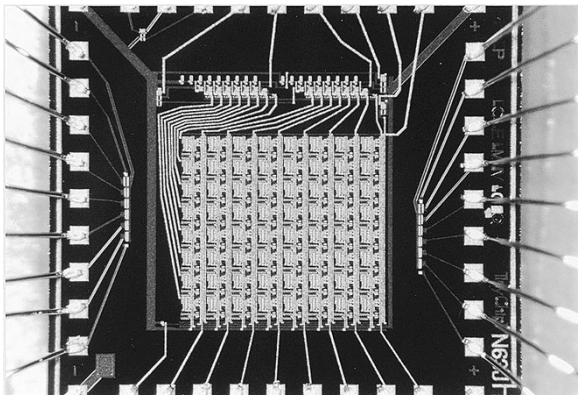
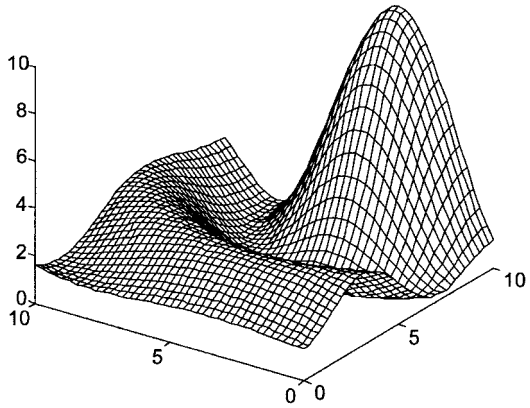
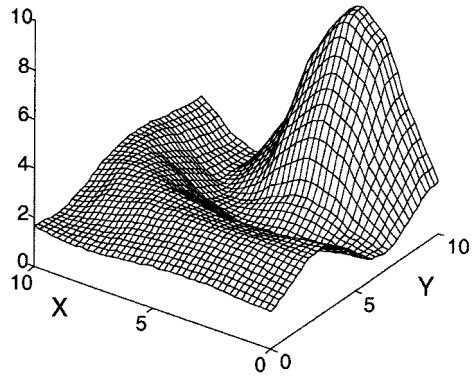


Fig. 9. Microphotograph of fabricated VLSI chip.



(a)



(b)

Fig. 10. Control surfaces. (a) Desired function and (b) control surface measured from the fabricated VLSI chip using an HP4155 semiconductor parameter analyzer.

chip. This problem can be solved by using a single cell cluster shown in Fig. 8 where both rule selection (transistors $M1$–$M4$) and normalization (source I_D and transistors $M4$–$M6$) are implemented. The normalization circuit consists of the $M4$ transistors from all cells connected to a single common current source. The sum of all $M4$ transistors currents is, therefore, always constant.

A universal fuzzy approximator has been designed and fabricated (Fig. 9). In order to make the chip universal, each fuzzifier consists of seven differential pairs with seven equally spaced reference voltages. This results in eight membership functions for each input and $8*8 = 64$ cluster cells. Sixty-four adjustable current mirrors for setting weights of output signals (Fig. 7) are programmed with 6-b accuracy. For an arbitrary two-dimensional function, only $6*64 = 384$ bits are required for programming. A test chip has been implemented in the 2-μm n-well MOSIS process using more than 2000 transistors to perform the analog signal processing. To simplify the test chip implementation, current sources were programmed at the contact mask level. Fig. 10 shows a comparison between the desired and the actually measured control surface from the fuzzy chip.

VII. Conclusion

An universal nonlinear VLSI approximator using the concepts of fuzzy logic was developed. The current chip is designed with two analog inputs, but it is possible to increase the number of inputs. The proposed architecture is a further simplification of fuzzy controllers. With this architecture, any required control surface can be digitally programmed. While the architecture was developed for VLSI implementation, the proposed approach is more general, and it can also be used in a microprocessor-based fuzzy system.

A new fuzzifier circuit with fewer transistors has also been presented. This circuit is very flexible, and any desired trapezoidal or Gaussian-type shape can be implemented. It also has a built-in normalization feature. A simple voltage-type MIN circuit was used for fuzzy inference. Instead of a classical defuzzifier, which requires signal division, normalization and weighted sum circuits were used.

References

[1] S. Ahmadi, L. Sellami, and R. W. Newcomb, "A CMOS PWL fuzzy membership function," *IEEE Int. Symp. Circuits and Systems*, Seattle, WA, Apr. 30–May 3, 1995, vol. 3, pp. 2321–2324.

[2] J. R. Angulo, "A BiCMOS universal membership function circuit with fully independent, adjustable parameters," in *Proc. IEEE Int. Symp. Circuits and Systems*, 1995, pp. 275–278.

[3] I. Barturone, A. Barriga, and J. L. Huertas, "Multi-input voltage and current-mode Min/Max circuits," in *Proc. 3rd Int. Conf. Fuzzy Logic, Neural Networks and Soft Computing*, Iizuka, Japan, 1994, pp. 120–126.

[4] J. Choi, B. J. Sheu, and J. C. F. Chang, "A Gaussian synapse circuit for analog VLSI neural networks," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 129–133, Mar. 1994.

[5] Y. Ota and B. Wilamowski, "Current-mode CMOS implementation of a fuzzy Min-Max network," in *Proc. World Congr. Neural Nerworks*, 1995, vol. II, pp. 480–483.

[6] Y. Ota and B. M. Wilamowski, "Analog hardware implementation of a voltage-mode fuzzy Min-Max controller," *J. Circuits, Syst., Comput.*, vol. 6, no. 2, pp. 171–184, 1996.

[7] J. Ramirez-Angulo, "A BiCMOS universal membership function circuit with fully independent, adjustable parameters," in *Proc. IEEE Int. Symp. Circuits and Systems*, Seattle, WA, Apr. 30–May 3, 1995, vol. 1, pp. 275–278.

[8] A. Rodriguez-Vazquez and F. Vidal-Verdu, "Learning in neuro/fuzzy analog chips," in *Proc. IEEE Int. Symp. Circuits and Systems*, Seattle, WA, Apr. 30–May 3, 1995, vol. 3, pp. 2325–2328.

[9] M. Sasaki, N. Ishikawa, F. Ueno, and T. Inoue, "Current mode analog fuzzy hardware with voltage input interface and normalization locked loop," *IEICE Trans. Fundam.*, vol. E57-A, no. 6, pp. 650–654, 1992.

[10] H. Shichman and D. A. Hodges, "Modeling and simulation of insulated-gate field-effect transistor switching circuits," *IEEE J. Solid-State Circuits*, vol. SC-3, pp. 285–289, Sept. 1968.

[11] T. Takagi and M. Sugeno, "Derivation of fuzzy control rules from human operator's control action," in *Proc. IFAC Symp. Fuzzy Information Knowledge Representation and Decision Analysis*, July 1989, pp. 55–60.

[12] M. Tartagani and P. Perona, "Computing centroids in current-mode technique," *Electron. Lett.*, vol. 29, no. 11, pp. 1811–1813, 1993.
[13] T. Yamakawa, "A fuzzy inference engine in nonlinear analog mode and its application to a fuzzy logic control," *IEEE Trans. Neural Networks*, vol. 4, pp. 496–522, May 1993.
[14] L. A. Zadeh, *Fuzzy Sets, Information and Control*. New York: Academic, 1965, vol. 8, pp. 338–353.

Bogdan M. Wilamowski (SM'83) received the M.S. degree in computer engineering, the Ph.D. degree in neural computing, and the D.Sc. degree in integrated circuit design from the Technical University of Gdansk, Gdansk, Poland, in 1966, 1970, and 1977, respectively.

He joined the Technical University of Gdansk in 1966 and became an Associate Professor in 1978 and a Professor in 1987. He was the Director of the Institute of Electronics from 1979 to 1981 and the Head of the Solid State Electronics. He was with the Nishizawa Laboratory, Tohoku University, Japan, from 1968 to 1970, and he spent one year at the Semiconductor Research Institute, Sendai, Japan, as a JSPS Fellow during 1975–1976. He was a Visiting Scholar at Auburn University, Auburn, AL, during 1981–1982, a Visiting Professor at the University of Arizona, Tucson, during 1982–1984, and a Visiting Scholar at the Alabama Microelectronics, Science, and Technology Center, Auburn University, during 1995–1996. Since 1989, he has been with the Electrical Engineering Department, University of Wyoming, Laramie.

Dr. Wilamowski has served on the Organizing Committees for several IEEE conferences. Currently, he is an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS, IEEE TRANSACTIONS ON EDUCATION, and IEEE Industrial Electronics Society Newsletter. He is also Treasurer of the IEEE Industrial Electronics Society and a member of the IEEE Neural Networks Council.

Richard C. Jaeger (S'68–M'69–SM'78–F'86) received the B.S. and M.E. degrees in electrical engineering in 1966 and the Ph.D. degree in 1969 from the University of Florida, Gainesville.

From 1969 to 1974, he was with IBM Corporation, Boca Raton, FL. In 1974, he became a Research Staff Member at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY. In 1979, he joined Auburn University, Auburn, AL, where he is currently a Distinguished University Professor in the Electrical Engineering Department. He was instrumental in founding the Alabama Microelectronics, Science, and Technology Center in 1984 and has led its development since that time. He was Program Chairman for the 1993 International Solid-State Circuits Conference and Chairman of the 1990 International VLSI Circuits Symposium.

Dr. Jaeger was a member of the IEEE Solid-State Circuits Council from 1984 to 1991, serving as its President during 1990–1991, and is currently Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS. From 1980 to 1982, he was the founding Editor-in-Chief of *IEEE Micro*, and subsequently received an Outstanding Contribution Award from the IEEE Computer Society for development of that magazine. He later became a member of the Governing Board of the IEEE Computer Society. He received the Birdsong Merit Teaching Award from the College of Engineering, Auburn University, in 1991, and he was selected as the Outstanding Electrical Engineering Faculty Member by the undergraduate students in 1993.

M. Okyay Kaynak (M'80–SM'90), for a photograph and biography, see p. 1018 of the October 1999 issue of this TRANSACTIONS.