Mixed Analog–Digital Fuzzy Logic Controller with Continuous-Amplitude Fuzzy Inferences and Defuzzification

Stamatis Bouras, Manousos Kotronakis, Ken Suyama, *Senior Member, IEEE*, and Yannis Tsividis, *Fellow, IEEE*

***Abstract*— A fuzzy logic controller has been realized using mixed analog–digital CMOS very large scale integration (VLSI) circuits for application in cases where the input and output variables are in analog form. It employs a new architecture where time sweeping of variables allows continuous-amplitude evaluation of fuzzy inferences and defuzzification during each evaluation cycle without having to discretize input and output variables. Direct processing of the analog input signal is used to obtain the corresponding crisp value; the digital portion is used only for programmability. No A/D and D/A converters are needed. The controller can handle three inputs, one output, and 25 programmable fuzzy rules. The test IC chips were fabricated using 0.7-μm CMOS technology. A control problem of stabilizing a ping-pong ball in a tube with a controllable air flow has been successfully demonstrated.**

***Index Terms*—Fuzzy logic controller, mixed analog–digital integrated circuits.**

I. Introduction

THERE are mainly two realization methods for fuzzy logic processing. One method is to digitize the input signals and to process the information in digital fashion throughout the system. Depending on system requirements, either a special digital very large scale integration (VLSI) circuit (for example, [1]–[8]) or a general-purpose microprocessor is used. The all-digital method is certainly appropriate in cases where the input and output are in digital form. However, in many instances inputs come from sensors and are analog in form and outputs must also be analog in order to drive several types of actuators. In applications where such input and output analog signals are many, a large number of A/D and D/A converters with sufficient resolution would be needed if an all-digital processor is to be used. This will increase chip size and power dissipation. To avoid this problem, a mixed analog–digital VLSI circuit can be used in such cases (for example, [9]–[27]). Digital logic is still used for programmability in the proposed method, but all processing maintains the signals in their original analog form. Since the output of most sensors changes slowly, sequential processing and time-sharing can be used to keep the power dissipation low.

This paper describes a new mixed analog–digital fuzzy logic controller with *continuous*-amplitude fuzzy inference and defuzzification within each evaluation cycle. Most of the critical fuzzy computation is done by analog IC techniques, which have been well developed over the last several years in the context of integrated analog continuous-time signal processors (see, for example, [28] and the many references therein). This is in contrast with many other mixed analog–digital techniques reported to date where some form of amplitude discretization of the input and output variables is used. The prototype controller has three inputs and one output with 25 possible programmable rules.

In our system, the fuzzy inference algorithm is a "min-max" type and the ith rule can be written as follows:

$$\text{If } x_1 \text{ is } A_{1i} \text{ and } x_2 \text{ is } A_{2i} \text{ and } x_3 \text{ is } A_{3i} \text{ then } y_i \text{ is } B_i \quad (1)$$

where x_1, x_2, and x_3 are the input variables, y_i is the ith output variable from the antecedent, and A_{1i}, A_{2i}, A_{3i}, and B_i are fuzzy sets for the ith rule. When the input assumes particular values a_1, a_2, and a_3, the fuzzy inference process based on the "min-max" algorithm can be formally written as follows:

$$\mu_{C_i}(y_i) = \min(\mu_{B_i}(y_i), \min[\mu_{A_{1i}}(a1), \mu_{A_{2i}}(a2), \mu_{A_{3i}}(a3)]), \quad \forall y_i \ \exists \ [y_{\text{low}}, y_{\text{high}}] \quad (2)$$

$$\mu_B(y) = \max(\mu_{C_1}(y_1), \ldots, \mu_{C_i}(y_i), \ldots, \mu_{C_{25}}(y_{25})), \quad \forall y \ \exists \ [y_{\text{low}}, y_{\text{high}}] \quad (3)$$

where $\mu_E(z)$ represents a membership function of fuzzy set E as a function of z and y_{low} and y_{high} define the output universe of discourse.

The final crisp output value y_{crisp} is obtained through the center of gravity as follows:

$$y_{\text{crisp}} = \frac{\int_{y_{\text{low}}}^{y_{\text{high}}} y \cdot \mu_B(y)\, dy}{\int_{y_{\text{low}}}^{y_{\text{high}}} \mu_B(y)\, dy}. \quad (4)$$

As explained later, the integrals are evaluated continuously without discretizing the output universe of discourse.

Two IC chips have been fabricated; one handles the fuzzy inference described above and the other contains the defuzzification circuits. In the next section, the overall architecture of the proposed controller and its operation are described. The circuit realization of some of the essential circuit blocks is then explained. The chips have been used to stabilize a ping-pong ball in a tube with a controllable air flow where the air turbulence around the ball makes the stabilization complicated.

Manuscript received January 18, 1996; revised February 17, 1997. This work was supported in part by PENED Grant 91ED122.

S. Bouras and M. Kotronakis are with the Division of Computer Science, National Technical University of Athens, Zographou, Athens, 15773 Greece.

K. Suyama and Y. Tsividis are with the Department of Electrical Engineering, Columbia University, New York, NY 10027 USA.

Publisher Item Identifier S 1063-6706(98)00851-0.

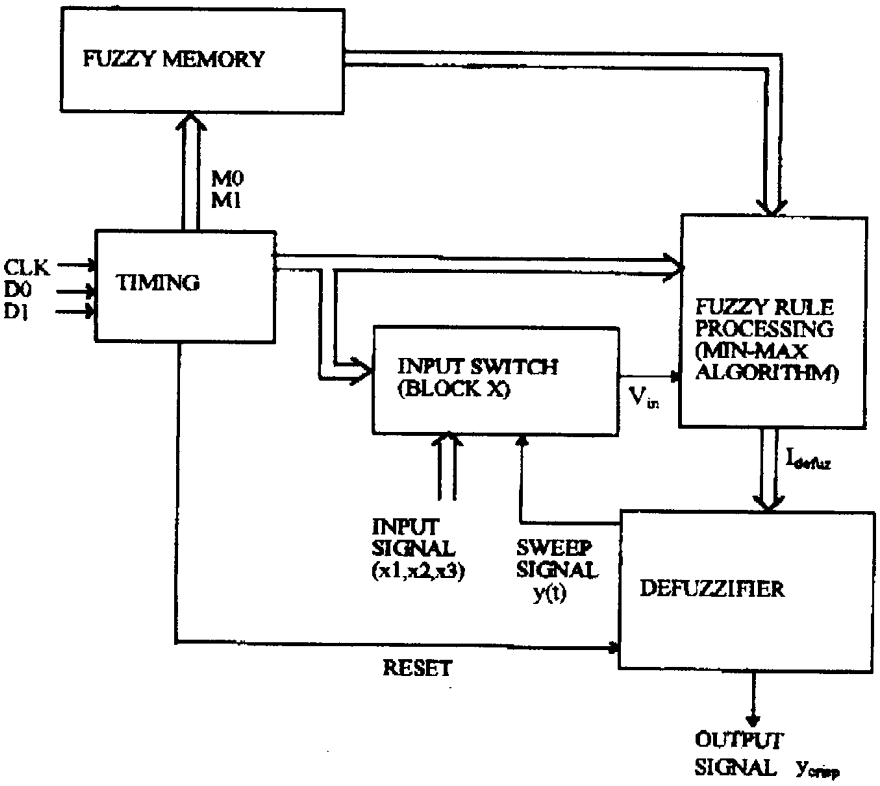


Fig. 1. Overall architecture of the fuzzy logic controller.

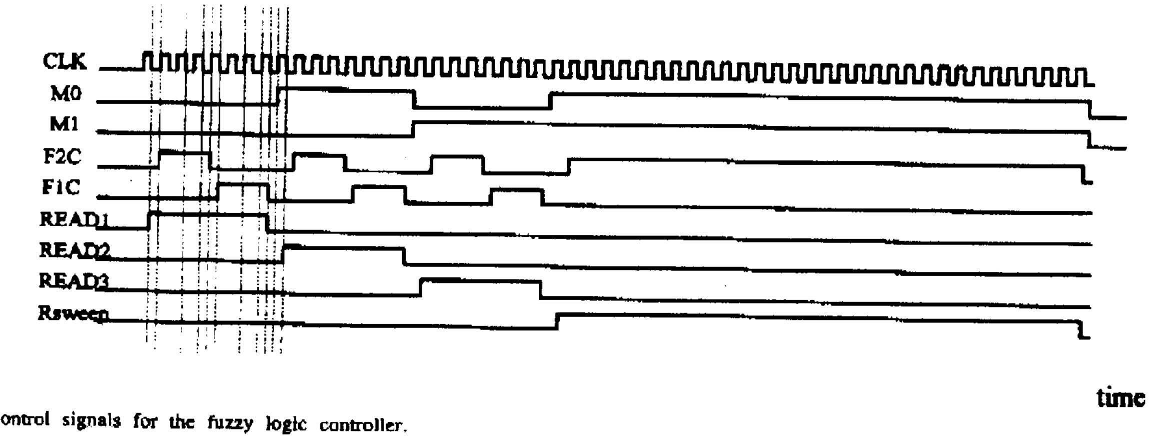


Fig. 2. Control signals for the fuzzy logic controller.

II. Fuzzy Controller Architecture

The architecture of the fuzzy controller is shown in Fig. 1. The three inputs and the output of the system are denoted as x_1, x_2, x_3, and y_{crisp}, respectively. The input switch, fuzzy rule processing, and defuzzifier blocks consist of analog circuits. The timing block is implemented using digital circuits and the memory block has a mixture of both analog and digital circuits. The following subsections explain each block.

A. Timing Block

The whole controller operates in a sequential fashion where each input is sampled and processed one at a time. After three inputs (if all three inputs are present) are processed, the defuzzification is done by time sweeping as it will be shown in Section II-D. Therefore, it is essential that the timing block broadcasts the state of operation to all blocks. The user needs to specify the number of inputs to the timing block through D0 and D1. Then, the block generates appropriate control signals to other blocks. Bits M0 and M1 are used to indicate to the memory block, which one of the inputs (the three inputs and the sweep signal) is being processed so that the memory block can generate appropriate fuzzy labels.

The timing block is also responsible for generating control signals to the input switch, fuzzy rule processing, and defuzzifier blocks, as shown in Fig. 2. The role of the control signals

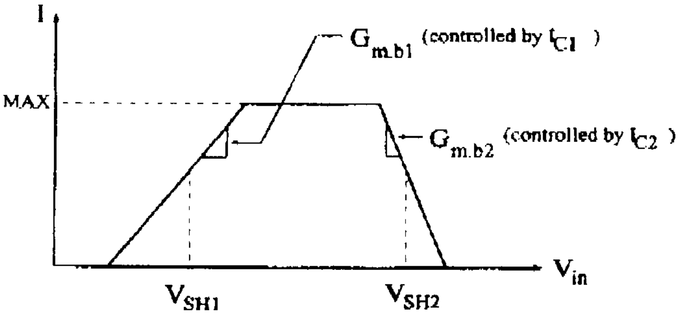


Fig. 3. Definition of a membership function by five quantities.

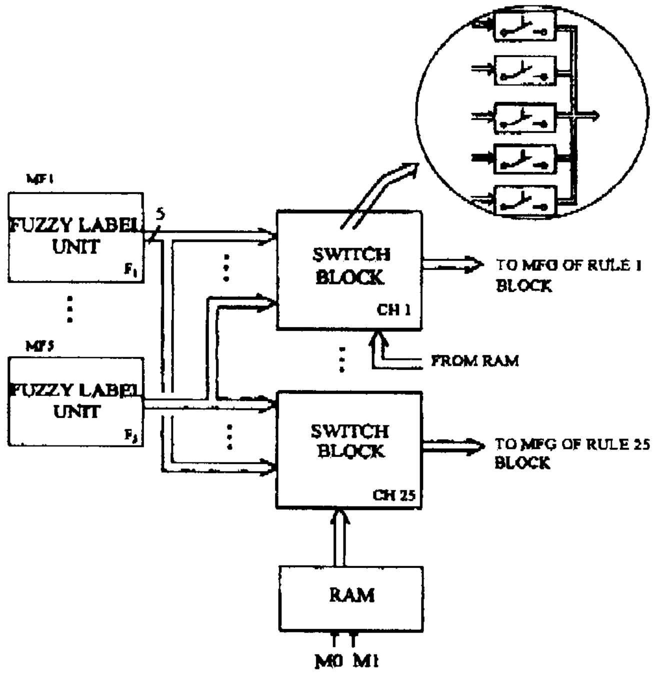


Fig. 4. Memory.

F1C, F2C, READ1, READ2, READ3, and Rsweep will be explained in Section II-C.

B. Memory Block

Depending on the state of operation given by M0 and M1, this block generates appropriate quantities which define membership functions being used in the fuzzy rule processing block. The proposed controller can generate fully programmable triangular and trapezoidal membership functions. Each fuzzy label corresponds to five quantities as shown in Fig. 3. Two voltages V_{SH1} and V_{SH2} are used to shift each segment independently, two currents I_{C1} and I_{C2} control the slope of each segment independently, and I_{MAX} determines the height.

Fig. 4 shows the memory. Since we chose to employ five membership functions to represent the input and output universe of discourse, there are five fuzzy label units (MF1 to MF5); each unit produces five analog quantities (V_{SH1}, V_{SH2}, I_{C1}, I_{C2}, and I_{MAX}) to specify a fuzzy label. The current values I_{C1}, I_{C2}, and I_{MAX} are represented by corresponding voltage values in fuzzy label units and the voltage values are simply converted to appropriate current values in each membership function generator. The fuzzy label units are connected to the membership function generator circuits in the fuzzy rule processing block (Fig. 1) via 25 switch blocks so that the fuzzy label units can be multiplexed (see the inset of Fig. 4). Recall that there are 25 possible programmable rules in the controller. This multiplexing of fuzzy label units through the switch boxes allows a modular way to add more membership functions. Since CMOS switches are small, the increase in area due to additional membership functions is minimal.

The realization of the fuzzy label unit depends on the extent of programmability desired. The units may be just analog memories (capacitors) if appropriate voltages are supplied externally. It is also possible to generate the voltages internally by having D/A converters fed by digital programming signals. In our experimental chip, we have employed the former approach. If some constraints are allowed for the extent of programmability, then significant simplification in the design of fuzzy label units can be done. For example, if all membership functions have the same shape [i.e., I_{C1}, I_{C2}, and I_{MAX} are the same for all medium frequencies (MF's)] and their positions are shifted by the constant amount within universe of discourse, then fuzzy label units can be designed by having only five analog values and two strings of voltage dividers, which generate appropriate V_{SH1} and V_{SH2} for all fuzzy label units.

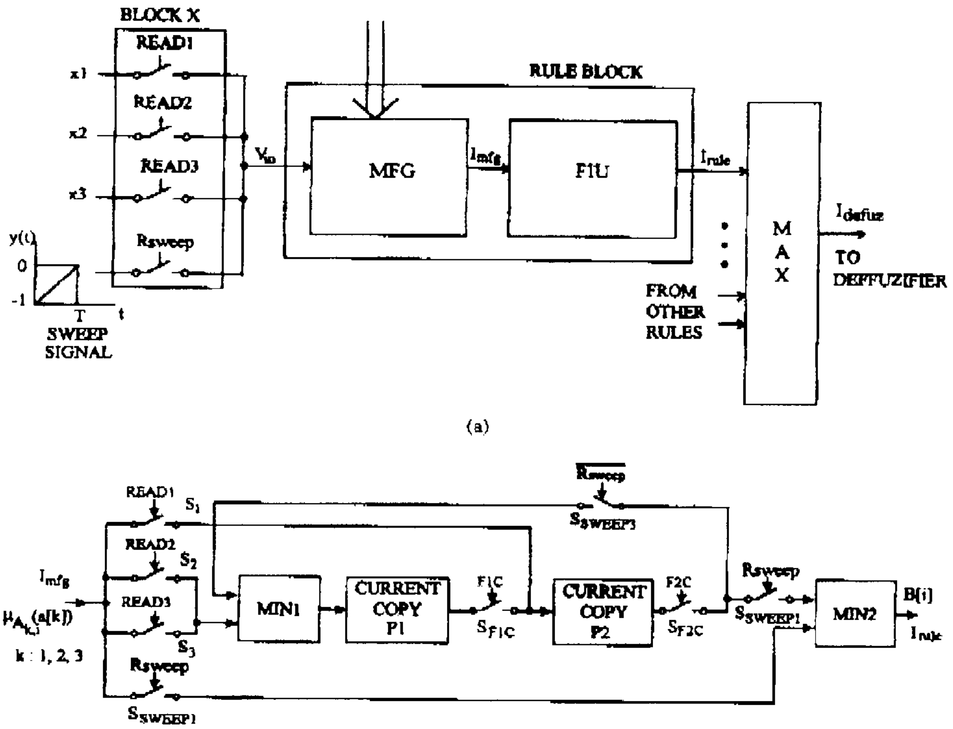
The analog multiplexer is controlled by M0 and M1 through binary RAM. The memory described above is similar in operation to the fuzzy memory device proposed in [29].

C. Fuzzy Rule Processing Block

Fig. 5(a) shows the fuzzy rule processing and input switch blocks. Note that only one input switch block is necessary for all 25 rules but 25 rule blocks are needed. As explained in Section II-A, the operation of the controller is based on sequential sampling of the three inputs. The input switch block connects the inputs one at a time to the membership function generator according to the control signals shown in Fig. 2. Another terminal in the block is used to sweep a voltage signal for a period of T (see Fig. 2). During the sweep phase, the MAX and defuzzifying operations are done (see Section II-D).

The membership function generator [Fig. 5(a)] receives an appropriate fuzzy label (the five analog values described in Section II-B) from the memory for the particular input and fuzzy rule it is processing. The membership function generator generates a continuous current, which represents the degree of the membership function within the closed universe of [0, 20 μA]. Each input is represented by a continuous voltage in the universe of [−1, 0 V], chosen to provide adequate linearity of various internal blocks with the process and power supply voltages used. The output current from the membership function generator is fed to the fuzzy inference unit.

The block diagram of the fuzzy inference unit is shown in Fig. 5(b). It consists of two MIN circuits, two current copiers,



(a)

(b)

Fig. 5. (a) Input switch and fuzzy rule processing blocks. (b) Fuzzy inference unit.

and several analog switches. The current copiers are used as analog memory (see below). Sequential MIN operations are first carried out on three fuzzy values, which corresponds to three inputs.

For the first input $a1$ (READ1 is "ON") in the ith rule, the degree of membership (i.e., $\mu_{A_{1i}}(a1)$) is calculated. This current is stored in current copier P2 [see Figs. 2 and 5(b)]. Then, the second input [i.e., $\mu_{A_{2i}}(a2)$] is connected to the MIN1 (READ2 is "ON"). In the MIN1, this current is compared with the previous current that has been stored in P2 (the switch F2C is "ON" in this phase). The smaller of the two currents is stored in current copier P1. once the MIN operation is done, the resultant current is copied into P2 (F1C is "ON"). This "pipeline " approach eliminates the need for having a sample-and-hold circuit. The same procedure is repeated for the third input, if any. At the end of this process, P2 has a stored current, which represents the result of the antecedent of the ith rule. Here, the inputs are assumed to change slowly compared to the pipelined inference process. It is simple to add more inputs to the controller by going through the above procedure for the additional inputs. Only the number of switches in the Block X and each rule block is increased by the number of additional inputs. Of course, there is a tradeoff involved between the number of inputs used and the required duration of one cycle.

The consequent ("then" clause) of the ith rule is done by generating the output membership function for the rule. The corresponding fuzzy label is given by the memory block and the resulting continuous current $[\mu_{B_i}(y_i)]$ from the membership function generator is available at the input of the fuzzy inference unit. During this phase, Rsweep is "ON." The consequent is completed when this current is compared with the stored current in P2 through MIN2, which represents the result of the antecedent of this rule. It should be noted that a change of variable has occurred from voltage to time. Therefore, it is possible to observe the resultant membership functions in an oscilloscope, as shown in Section V.

The currents from all rule blocks are applied to the MAX block as shown in Fig. 5(a). The output of the MAX block is the current that corresponds to the overall fuzzy inference. The current is then applied to the defuzzifier.

D. Defuzzifier Block

A time-sweeping approach to defuzzification was recently reported elsewhere [27], but it is not suitable for implementation using conventional IC technologies. Here we propose instead a method using time sweeping, which can be implemented in standard CMOS. The defuzzification is based on (4) and is implemented by a multiplier, a divider, and three integrators (Fig. 6). We also use a current-to-voltage converter, an attenuator, and a division control unit. The inputs are the current $I_{\rm in}(t)$ that represents the overall fuzzy inference and the sawtooth voltage $V_{\rm sweep}(t)$. The output is $y_{\rm crisp}$, which is a voltage signal. The sawtooth voltage $V_{\rm sweep}(t)$ is generated by applying a dc voltage V_S to the integrator I1 in Fig. 6 for a period of T [Fig. 5(a)] and resetting the integrator until the next sweep period.

The attenuator block with a gain of 1/2 is inserted since the nonlinearity of the multiplier becomes significant for signals

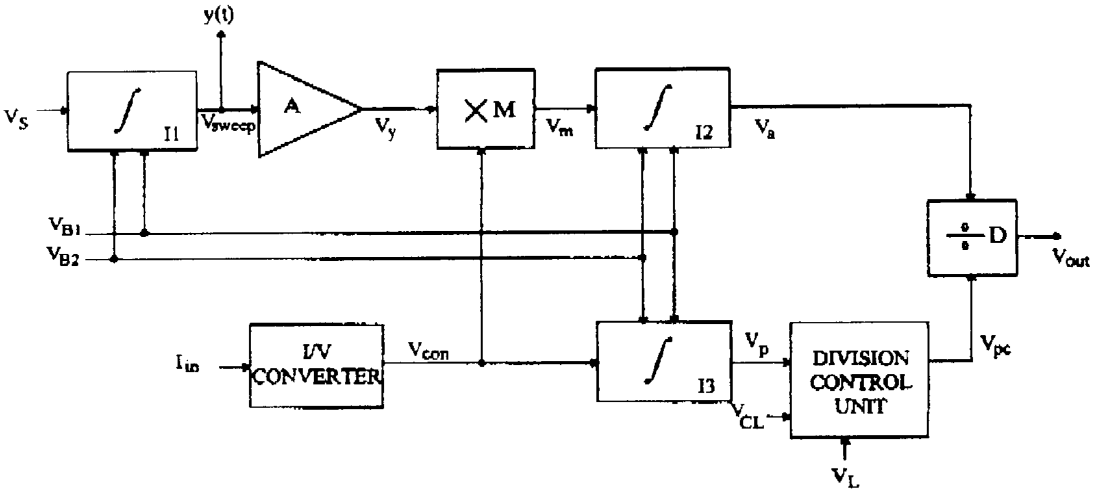


Fig. 6. Block diagram of defuzzifier.

greater than 0.5 V (−1 to −0.5 V). The current-to-voltage (I/V) converter is needed since the multiplier block is based on a voltage-mode integrator. The division control unit is used to avoid very small signals applied to the denominator of the divider. When the value V_p becomes smaller than a certain voltage V_L, the output voltage of the unit V_{pc} is kept to V_{CL}. Otherwise, $V_{pc} = V_p$. The defuzzified output voltage is

$$y_{crisp} = V_{out}(T) = M \cdot A \cdot D \cdot \frac{\int_0^T I_{in} \cdot V_{sweep}(t)\, dt}{\int_0^T I_{in}(t)\, dt} \qquad (5)$$

where M is the multiplication factor, D is the division factor, A is the attenuator factor, and T the sweep duration.

III. BUILDING BLOCKS

Various analog IC techniques are employed in the fuzzy rule processing and defuzzifier blocks. The current copiers used as an analog memory in the fuzzy inference unit (Fig. 5(b)) are based on [30]. The MIN and MAX circuits are similar to techniques previously reported (for example, [9]–[11], [13], [16]). The following sections briefly describe the programmable membership function generator and defuzzifier and the current copier.

A. Membership Function Generator

The principle of the programmable membership function generator circuit is shown in Fig. 7(a). The input is a voltage signal V_{in} The output of the membership function generator is I_{mfg} This circuit can produce fully programmable triangular and trapezoidal membership functions. The transconductance $G_{m,b1}$ and $G_{m,b2}$ are controlled by currents I_{c1} and I_{c2}, respectively. The basic circuit to determine the slope and position of membership functions consists of a constant transconductance cell $G_{m,a}$ [31] and a current mirror with current controlled gain [32], as shown in Fig. 7(b). As shown in Fig. 3 and Section II-B, a membership function can be specified by five quantities: two voltages V_{SH1} and V_{SH2} position each segment independently, two currents I_{C1} and I_{C2} control the slope of each segment independently, and I_{MAX} determines the height. Note that V_{SH1}, V_{SH2}, I_{C1}, I_{C2}, and I_{MAX} are all programmed by the memory block.

B. Defuzzifier

The key components in this block are the integrator, divider, multiplier, I–V converter, and the division control unit. The integrator and divider are explained here. The multiplier and the I–V converter are based on a design methodology similar to that of the integrator. The division control unit consists of a comparator and several switches. In order to achieve a high degree of linearity for each component and to limit the disturbance from spurious signals, a fully balanced architecture is used [28].

The integrators are designed using triode-operated MOSFET's [33]. Fig. 8(a) shows the integrator with reset switches. The circuit implements the following equation [33]:

$$\begin{aligned} &V_{o1} - V_{o2} \\ &\quad \simeq -\frac{K \cdot (W/L)_I \cdot (V_{B1} - V_{B2})}{C} \cdot \int_0^T (V_{a1} - V_{a2})\, dt \qquad (6) \end{aligned}$$

where $(W/L)_I$ is the ratio of the MOSFET's used in the integrator, K is a process-dependent parameter, and T is the integration duration. The op amp is a fully balanced design [34] in which the common-mode reference level is −1 V. The gate voltages V_{B1} and V_{B2} can be used to "tune" the integrator to eliminate the effect of tolerances, temperature, and aging. This can be done automatically [28]; however, for flexibility during testing we chose manual control of these voltages.

The divider circuit is shown in Fig. 8(b). It uses the same four-MOSFET cell as the integrator [33]. The op amps used in the circuit are of the same type as the one in the integrator. R_1, R_2, and R_3 are polysilicon resistors. The circuit realizes the following equation.

$$V_{out1} - V_{out2} = -\frac{R_2}{R_3} \cdot \frac{1}{R_1 \cdot K \cdot (W/L)_D} \frac{V_{a1} - V_{a2}}{V_{pc1} - V_{pc2}} \qquad (7)$$

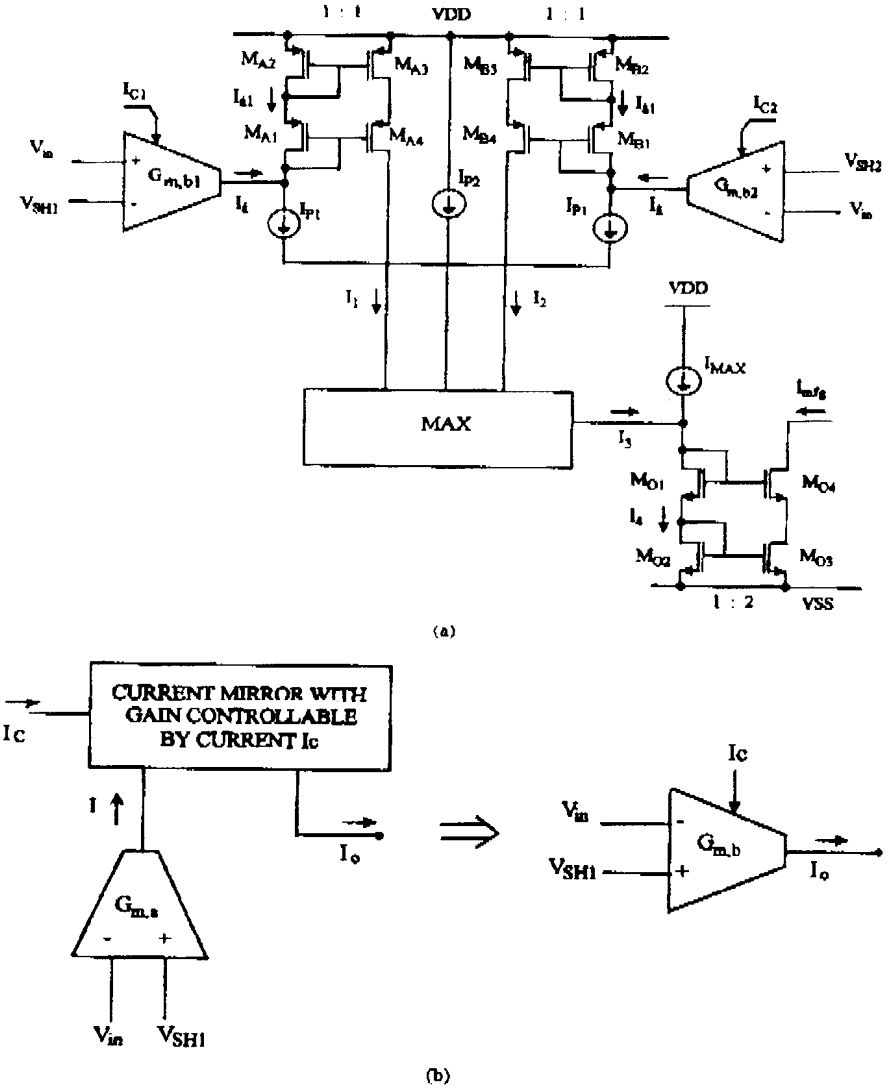


Fig. 7. Programmable membership function generation circuit.

where $(W/L)_D$ is the ratio of the MOSFET's used in the divider. It should be noted that a full multiplier is used in the feedback as opposed to only four transistors in order to avoid positive feedback in certain cases.

C. Current Copier

The current copier of Fig. 9 operates as follows [30]: Devices M_6 and M_5 form a cascode stage, fed from source follower M_7, which is biased by current source M_8. These four devices form a transconductor with compatible input and output levels. When a current I_{in} is to be memorized, the three switches on the left are closed. I_{in} is mirrored through the p-channel current mirror and the feedback from the output to the input of the transconductor forces V_{G7} to attain whatever value is needed to make the current in M_6 equal to the mirrored current I_{in}. This value is now stored on C_1 and remains there even after the switches on the left are opened. Therefore, the current in M_6 is still equal to I_{in} and can be fed into a load by closing the switch on the right. Capacitors C_2 and C_3, along with the switches connected to them and the particular switching waveforms indicated on the figure, are used to reduce clock feedthrough [35].

IV. IC Realization

A standard 0.7-μm CMOS process was used to fabricate the input switch and fuzzy rule processing blocks on one chip (called the "inference" chip) and the defuzzifier on another chip to show the principle (see Figs. 10 and 11). The fuzzy memory and timing blocks were not fabricated at this time and they were built externally. The inference chip is capable of handling three inputs, one output, and five rules. However, the MAX circuit in Fig. 5(a) can accept 25 rules. Therefore, it is possible to extend the current system to handle 25 rules using multiple inference chips. It should be emphasized that

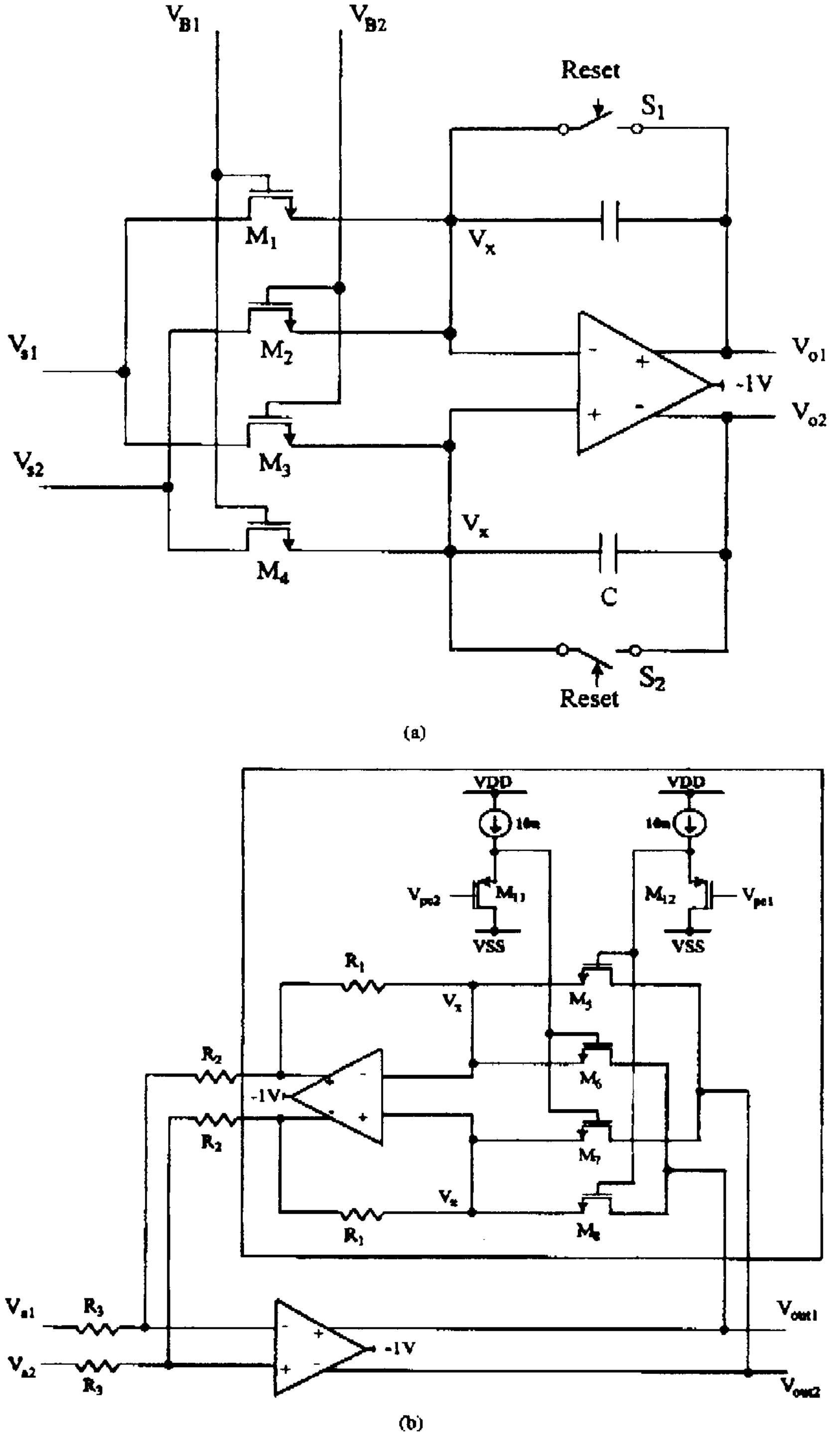


Fig. 8. Defuzzifier circuit blocks. (a) Integrator with reset. (b) Divider.

the chip set was designed to show the principle and that no effort was done to optimize the area and power in this case. Components from a standard cell library were used for some portion of the controller. The sweeping period is 24 μs and the whole cycle is 48 μs. Important design parameters are summarized in Table I.

V. Measurements

Because of the change of variable inherent in the time-sweeping process explained in Section II-C, it is possible to display membership functions easily using a digital oscilloscope. The top curve in Fig. 12(a) shows the voltage sweep

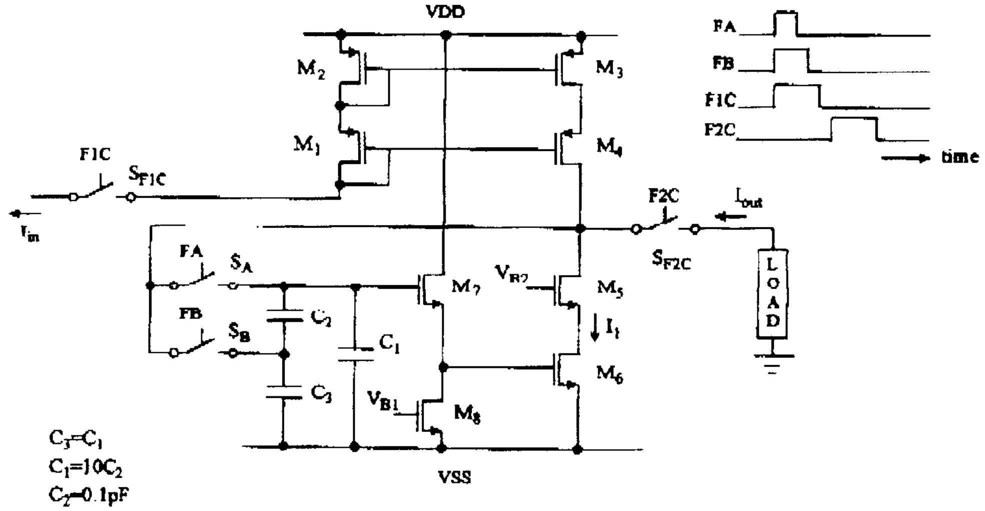


Fig. 9. Current copier circuit.

Fig. 10. Chip microphotograph of the inference chip.

Fig. 11. Chip microphotograph of the defuzzifier.

signal that ramps from −1 V to 0 V. The sweeping time was purposely slowed down for clear displaying in the presence of parasitics due to external connections during testing. The membership functions, which are in the form of current, were converted to voltage signals by external resistors and displayed in Fig. 12(b) and (c). Note that the membership functions look "inverted" due to the way the current signals are converted to voltage signals. Both triangular and trapezoidal functions can be obtained as shown. Fig. 13 demonstrates the programmability of membership functions by the membership function generator circuit (Fig. 7). Their slopes, heights, and positions can be easily programmed.

Defuzzifier signals can also be observed as shown in Fig. 14. All signals are balanced around −1 V. As discussed in Section III-B, the defuzzifier is implemented differentially. For example, Fig. 15 shows the differential sawtooth waveform at the output of the integrator O1 in Fig. 6. In Fig. 14, only the positive side of differential signals is shown. Referring to Fig. 6 for the block diagram of the defuzzifier, the plots from the top of Fig. 14 are the output of the multiplier, the integrators, and the divider, respectively. At the end of the sweep period, the defuzzification computation is completed. In Fig. 14, the divider value "H" is marked to show that it is the crisp value that should be sampled.

A control problem of stabilizing the position of a ping-pong ball in a tube with a controllable air flow has been used to demonstrate the functionality of the controller. Fig. 16 shows the experimental setup. A voltage-controlled fan with variable air flow is used at the bottom of a 1.5-m long tube to blow air, which pushes the ball upwards. Ultrasound transducers mounted at the top of the tube are used to sense the ball's position and, through differentiation, its velocity. Level conditioners (interface circuits) are used to connect the control loop that consists of the transducers, the fuzzy processor, and the motor driver since each component requires a different range of input and output values. Five rules, two inputs (the position of the ball and its derivative), and three triangular membership functions (negative, zero, and positive) are used. Let e be the distance error from a target position of the ball, de/dt be its derivative, and z be the strength control of the air flow in the tube. A set of simple rules are used as follows:

1) if e is negative and de/dt is zero, then z is positive;
2) if e is zero and de/dt is negative, then z is positive;
3) if e is zero and de/dt is zero, then z is zero;
4) if e is zero and de/dt is positive, then z is negative;
5) if e is positive and de/dt is zero, then z is negative.

The timing block was implemented externally by simple CMOS logic gates fed by a master clock. The fuzzy memory was also realized externally by providing a set of analog voltages for each label and an array of analog CMOS switches.

TABLE I
DESIGN SPECIFICATIONS OF THE FUZZY CONTROLLER

No. of inputs	3 (programmable)
No. of outputs	1
No. of membership functions	5 (programmable)
Membership function shape	Triangle & Trapezoid (programmable)
No. of rules	25 (programmable). 5 in a chip.
Fuzzy inference method	MIN-MAX
Defuzzification	Center of gravity
Power Supply	± 2.5 V
Input and output range	[-1, 0] V
Process time incl. defuzzification	48 μs
Clock Speed	1.3 MHz
Technology	0.7 μm CMOS
Chip area (inference)	1.68 mm^2
Chip area (defuzzifier)	2.4 mm^2

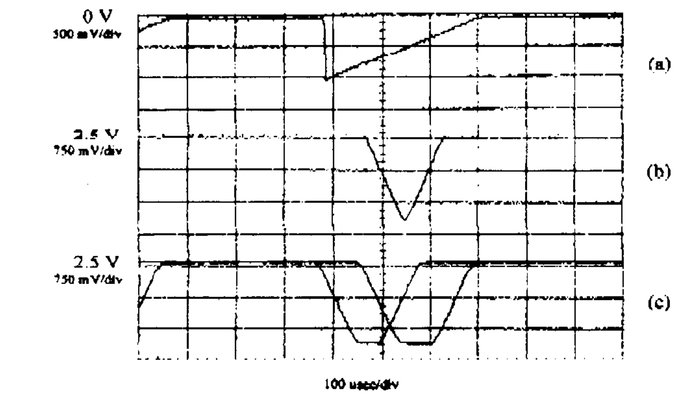


Fig. 12. Membership function signals. (a) Sweep signal. (b) Triangular membership function. (c) Trapezoidal membership function.

As already mentioned, an alternate way would be to generate these voltages through D/A converters with a few bits each, fed by digital programming signals.

VI. Demonstration

The system successfully stabilized the position of the ball even when some disturbances were purposely introduced (e.g., inhibiting part of the air flow by partially covering the top of the tube). As expected, we found that the amount of overlap between the domains of the triangular membership functions influenced the performance of the control system. For example, when the membership functions were made to overlap 25%, some ringing occurred although the ball did eventually settle to the desired position. Using 50% overlap, this problem was eliminated and the ball settled to the desired position within 3 s after a slight overshoot.

VII. Conclusion

A fuzzy logic controller has been designed, built, tested, and demonstrated in a control system. The controller has two custom-made mixed analog–digital chips using a standard 0.7 μm CMOS fabrication process. Through a time-sweeping

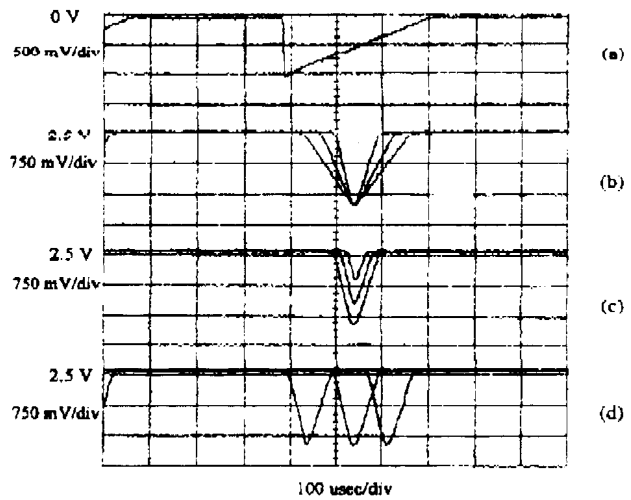


Fig. 13. Programmability of membership functions. Curves are as follows. (a) Sweep voltage. (b) Triangular functions with different slopes. (c) Triangular functions with different heights. (d) Triangular functions with different locations.

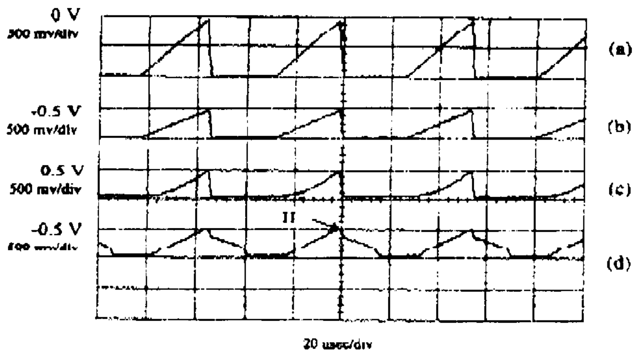


Fig. 14. Defuzzifier signals. Referring to Fig. 6, plots are as follows. (a) Multiplier output. (b) The output of the integrator I3. (c) The output of the integrator I2. (d) The output of the divider.

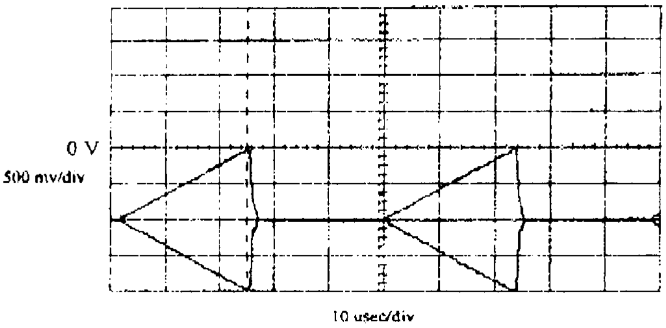


Fig. 15. Differential voltage sweep signal at the output of integrator I1 in Fig. 6.

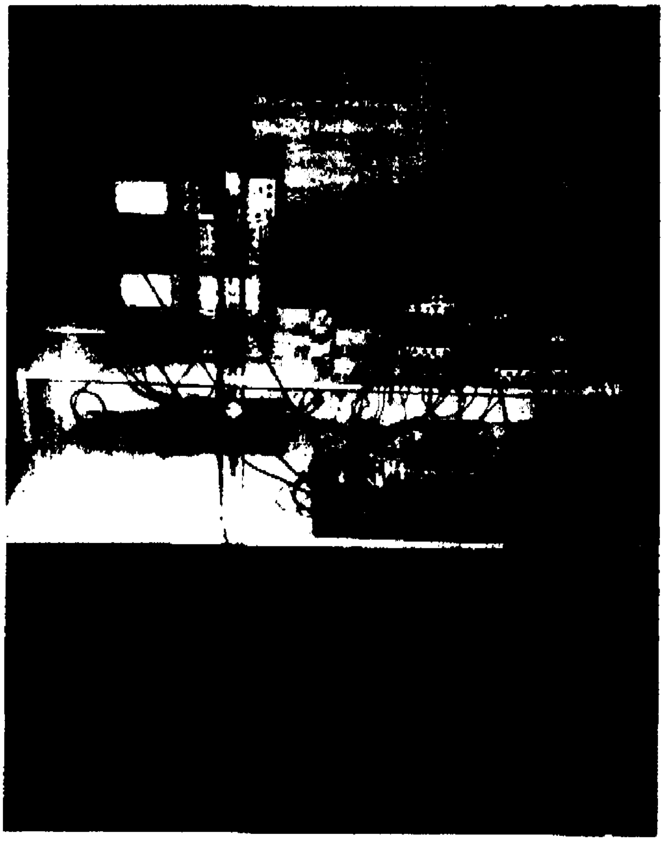


Fig. 16. Demonstration setup for stabilizing a ping-pong ball in a tube.

feature that allows a change of variable from voltages and currents to time, continuous fuzzy inferences and defuzzification were employed. This eliminates the need for any quantization in the signal path and, thus, no data converters are needed. Taking advantage of various state-of-the-art analog IC techniques (e.g., variable transconductance elements, current copiers, continuous-time integrators, etc.), analog circuits for programmable membership functions and continuous defuzzification have been designed. Although the controller presented in this paper has only three inputs, one output, and 25 possible rules, the modular design of the architecture can potentially allow more input and output variables as well as a larger number of rules. The continuous-time analog processing feature of this controller will allow it to coexist with integrated sensors within one IC chip where the controller could be used without having to discretize the sensor output.

Acknowledgment

The authors would like to thank Y. Avrithis for his help with the building of the ball-stabilizing demonstration setup and Dr. Y. Papananos for useful discussions and providing the layout of the op amp in [34].

References

[1] M. Togai and H. Watanabe, "Expert system on a chip: An engine for real-time approximate reasoning," *IEEE Expert*, vol. 1, pp. 55–62, Aug. 1986.

[2] H. Watanabe, W. D. Dettloff, and K. E. Yount, "A VLSI fuzzy logic controller with reconfigurable cascadable architecture," *IEEE J. Solid-State Circuits*, vol. 25, pp. 376–382, Apr. 1990.

[3] H. Eichfeld, M. Löhner, and M. Muller, "Architecture of a CMOS fuzzy logic controller with optimized memory organization and operator design," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, San Diego, CA, Mar. 1992, pp. 1317–1323.

[4] K. Nakamura, N. Sakashita, Y. Nitta, ~~K. Shimomura, and T. Tokuda~~, *IEICE Trans. Electron.*, vol. E76-C, no. 7, pp. 1102–1111, July 1993.

[5] Y. Harata, N. Ohya, K. Hayakawa, T. Shigematsu, and Y. Kita, "A fuzzy inference LSI for an automotive control," *IEICE Trans. Electron.*, vol. E76-C, no. 12, pp. 1780–1787, Dec. 1993.

[6] M. Sasaki and F. Ueno, "7.5 MFLIPS fuzzy microprocessor using SIMD and logic-in-memory structure," *IEICE Trans. Electron.*, vol. E77-C, no. 7, pp. 1075–1082, July 1994.

[7] M. Menke, R. Brunner, H. Eichfeld, and T. Künemund, "A general-purpose 12 bit fuzzy coprocessor," in *Proc. Eur. Solid-State Circuits Conf. (ESSCIRC)*, Lille, France, Sept. 1995, pp. 118–121.

[8] H. Surmann and A. P. Ungering, "Fuzzy rule-based systems on general-purpose processors," *IEEE Micro*, vol. 15, pp. 40–48, Aug. 1995.

[9] T. Yamakawa, T. Miki, and F. Ueno, "The design and fabrication of the current mode fuzzy logic semi-custom IC in the standard CMOS IC technology," in *Proc. IEEE Int. Symp. Multiple-Valued Logic*, Ontario, Canada, May 1985, pp. 76–82.

[10] T. Yamakawa and T. Miki, "The current mode fuzzy logic integrated circuits fabricated by the standard CMOS process," *IEEE Trans. Comput.*, vol. C-35, pp. 161–167, Feb. 1986.

[11] K. Tsukano, T. Inoue, and F. Ueno, "A design of current-mode analog circuits for fuzzy inference hardware systems," in *Proc. IEEE Int. Symp. Circuits Syst.*, Chicago, IL, May 1993, pp. 1385–1388.

[12] 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.

[13] M. Sasaki, K. Taniguchi, Y. Ogata, F. Ueno, and T. Inoue, "An implementation of multiple-valued logic and fuzzy logic circuits using 1.5 V Bi-CMOS current-mode circuit," *IEICE Trans. Inform. Syst.*, vol. E76-D, no. 5, pp. 571–576, May 1993.

[14] T. Kettner, C. Heite, and K. Schumacher, "Analog CMOS realization of fuzzy logic membership functions," *IEEE J. Solid-State Circuits*, vol. 28, pp. 857–861, July 1993.

[15] B. D. Liu and C. Y. Huang, "Array based fuzzy inference mechanism implemented with current-mode CMOS circuits," in *Proc. Eur. Solid-State Circuits Conf. (ESSCIRC)*, Seville, Spain, Sept. 1993, pp. 537–540.

[16] L. Lemaitre, M. J. Patyra, and D. Mlynek, "Analysis and design of CMOS fuzzy logic controller in current mode," *IEEE J. Solid-State Circuits*, vol. 29, pp. 317–321, Mar. 1994.

[17] J. Fattaruso, S. S. Mahant-Shetti, and J. B. Barton, "A fuzzy logic inference processor," *IEEE J. Solid-State Circuits*, vol. 29, pp. 397–402, Apr. 1994.

[18] I. Baturone, J. L. Huertas, A. Barriga, and S. Sánchez-Solano, "Current-mode multiple-input MAX circuit," *Electron. Lett.*, vol. 30, no. 9, pp. 678–680, Apr. 28, 1994.

[19] I. Baturone, S. Sánchez-Solano, A. Barriga, and J. L. Huertas, "A switched-capacitor based singleton fuzzy controller," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Orlando, FL, June 1994, pp. 1303–1307.

[20] S. Guo and L. Peters, "A reconfigurable analog fuzzy logic controller," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Orlando, FL, June 1994, pp. 124–128

[21] A. P. Ungering, D. Herbst, A. Weyergraf, and K. Goser, "Architecture of a mixed mode fuzzy controller," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Yokohama, Japan, Mar. 1995, pp. 1191–1196.

[22] J. Ramirez-Angulo, K. Treece, P. Andrews, and T. Choi, "Current-mode and voltage-mode VLSI fuzzy processor architecture," in *Proc. IEEE Int. Symp. Circuits Syst.*, Seattle, WA, May 1995, pp. 1156–1159.

[23] A. Rodríguez-Vázquez and F. Vidal, "Modular design of adaptive analog CMOS fuzzy controller chips," in *Proc. Eur. Solid-State Circuits Conf (ESSCIRC)*, Lille, France, Sept. 1995, pp. 122–125.

[24] J. Oehm, M. Gräfe, T. Kettner, and K. Schumacher, "Analog CMOS drilling machine controller with programmable characteristic curves generated from fuzzy-rule-sets," in *Proc. ESSCIRC*, Lille, France, Sept. 1995, pp. 114–117.

[25] F. Dülger, Z. Sezgin Günay, B. Pamir, and U. Çilingiroğlu, "ASIC implementation of fuzzy controllers: A sampled-analog approach," in *Proc. ESSCIRC*, Lille, France, Sept. 1995, pp. 450–453.

[26] J. L. Huertas, S. Sánchez-Solano, I. Baturone, and A. Barriga, "Integrated circuit implementation of fuzzy controllers," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1051–1058, July 1996.

[27] H. Tang and H. C. Lin, "Defuzzifier circuits using resonant tunneling diodes," in *Proc. IEEE Symp. Circuits Syst.*, Seattle, WA, May 1995, pp. 981–984.

[28] Y. P. Tsividis and J. O. Voorman, *Integrated Continuous-Time Filters*. New York: IEEE Press, 1993.

[29] T. Yamakawa and K. Sasaki, "Fuzzy memory device," *2nd Int. Fuzzy Syst. Assoc. (IFSA) Congress*, Tokyo, Japan, July 1987, pp. 551–555 (preprints).

[30] S. J. Daubert, D. Vallancourt, and Y. P. Tsividis, "Current copier cells," *Elect. Lett.*, vol. 24, no. 25, pp. 1560–1562, Dec. 1988.

[31] E. Klumperink, E. V. D. Zwan, and E. Seevinck, "CMOS variable transconductance circuit with constant bandwidth," *Electron. Lett.*, vol. 25, no. 10, May 11, 1989.

[32] E. A. M. Klumperink and E. Seevinck, "MOS current gain cells with electronically variable gain and constant bandwidth," *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1465–1467, Oct. 1989.

[33] Z. Czarnul, "Modification of the Banu-Tsividis continuous-time integrator structure," *IEEE Trans. Circuits Syst.*, vol. CAS-33, no. 7, pp. 714–716, July 1986.

[34] Y. Papananos and Y. Tsividis, "Design and implementation of a CMOS operational amplifier architecture with dual common-mode feedback loop," in *Proc. 3rd Int. Conf. Electron., Circuits, Syst.*, Rodos, Greece, Oct. 1996, pp. 904–907.

[35] H. P. Lie, "Switched capacitor feedback sample-and-hold circuit," U.S. Patent 4 585 956, 1986.

Stamatis Bouras was born in Athens, Greece, in 1968. He received the diploma degree in electrical engineering, in 1990, and the Ph.D. degree in electrical and computer engineering, in 1996, both from National University of Athens, Greece.

His research interests are in the area of analog VLSI circuit design and hardware implementation of fuzzy logic systems.

Manousos Kotronakis was born in Kozani, Greece, on January 31, 1971. He received his diploma degree in electrical and computer engineering from the National Technical University of Athens, Greece, in 1994.

His scientific interests are in the areas of analog VLSI circuit design and fuzzy logic.

Ken Suyama (S'81–M'82–SM'96) received the B.S. degree from the University of California, Davis, in 1980, and the M.S. and Ph.D. degrees from Columbia University, New York, NY, in 1982 and 1989, respectively.

He is currently a Research Scientist and an Adjunct Associate Professor in the Microelectronic Circuits and Systems Laboratory at Columbia University. He is also an Associate Director of that laboratory. His current research interests include analog and RF integrated circuits, mixed micromechanical systems and integrated circuits, MOSFET device modeling for analog IC design, computer-aided design and analysis of integrated circuits, and chaotic neural networks.

Dr. Suyama was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II from 1995 to 1997. He is also a member of the Board of Directors and a Vice President of the Catalyst Foundation.

Yannis Tsividis (S'71–M'74–SM'75–F'86) received the B.S. degree from the University of Minnesota, Minneapolis, and the M.S. and Ph.D. degrees from the University of California, Berkeley.

He is with the Department of Electrical Engineering, Columbia University, New York, as Professor and Director of the Microelectronic Circuits and Systems Laboratory. He has worked for Motorola Semiconductor, Phoenix, AZ, and AT&T Bell Laboratories, Murray Hill, NJ, and has taught at the University of California, Berkeley, Massachusetts Institute of Technology, Cambridge, and the National Technical University of Athens, Greece.

Dr. Tsividis is the recipient of the 1984 IEEE Baker Best Paper Award and the 1986 European Solid-State Circuits Conference Best Paper Award. He is co-recipient of the 1987 IEEE Circuits and Systems Society Darlington Best Paper Award. He also received the Great Teacher Award at Columbia University.