VLSI IMPLEMENTATION OF A UNIVERSAL FUZZY CONTROLLER

BOGDAN M. WILAMOWSKI
University of Wyoming
RICHARD C. JAEGER
Auburn University

ABSTRACT:
In this paper a non-conventional 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 2um n-well technology. A new fuzzification circuit, which requires only one differential pair per membership functions is proposed. Eight equally spaced membership function 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 bit resolution.

INTRODUCTION

Analog signal processing is an attractive alternative for control purposes. It combines parallel processing with a speed limited only by a delay of signals through the network. Also, in many cases, simple fuzzy controllers perform better than traditional ones, especially if systems are very nonlinear. The purpose of this work is to develop a fuzzy type controller suitable for analog VLSI implementation. The classical approach to fuzzy control [Zadeh, 65] (Fig.1) although possible [Yamakawa,93][Barturone,94][Liu,94][Colodro,95] [Ota,95][Ota,96] is difficult to implement in analog hardware. In this paper various different fuzzy architectures for VLSI circuitry are studied.

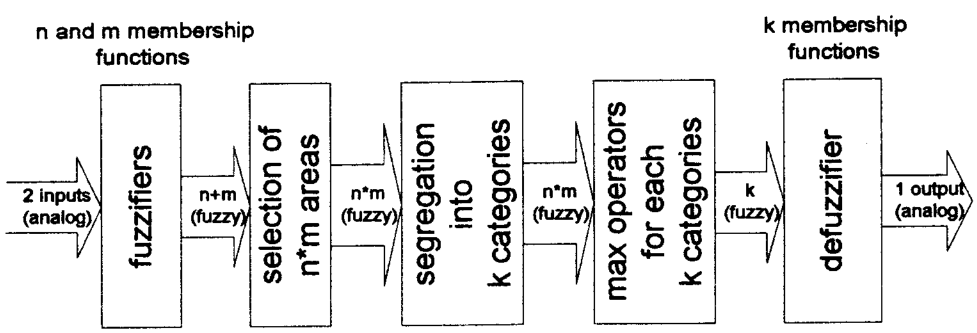

Fig. 1. Block diagram of classical fuzzy system

SYSTEM ARCHITECTURES

The most difficult block to implement into hardware is the defuzzifier. It usually requires signal division which leads to very complicated design. When a simplified singleton type of defuzzifier is used, the defuzzifier can be built using feedback [Ota, 1996] ,or it can use a normalization circuit and weighted summing [Takagi, 89][Rodriguez, 95]. The feedback approach leads to a limited accuracy. To improve accuracy large open-loop gain is required, and this can lead to a stability problem. It is much easier to implement the Takagi-Sugeno defuzzifier. If the sum of fuzzy signals coming out of the fuzzy rule table is kept constant, then the defuzzification can be done using simple weight summing (Fig.2).

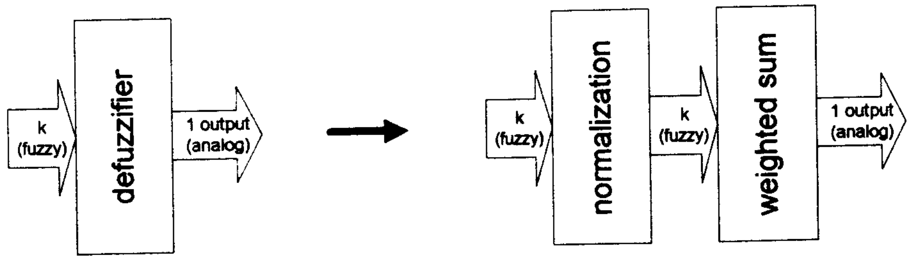

Fig. 2. Takagi-Sugeno type defuzzifier

The block for segregation into k categories (Fig. 1) is easy to implement only when the categories are specified ahead of chip fabrication. To make a programmable chip, very large and complicated circuitry is required because there are n*m inputs in the rule table and each position in the rule table can be classified into k output categories. This leads to $(n*m)^k$ possibilities. It turns out that this segregation process can be completely eliminated if all n*m fuzzy variables are used as input to the defuzzifier. This significantly reduces the fuzzy inference engine, but the number of variables supplied to the defuzzifier is much larger (Fig. 3). When the Takagi-Sugeno type defuzzifier is used, as shown in Fig. 2, the large number of defuzzifier inputs is not an issue and it can be relatively easy to implement on the VLSI chip.

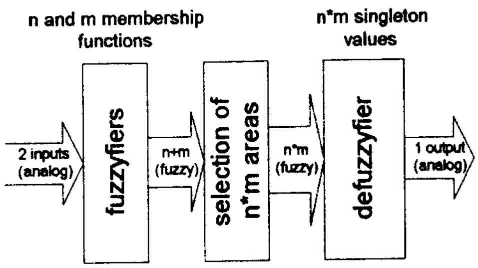

Fig.3. Block diagram of modified fuzzy system

The circuit for selection of the m*n areas is usually implemented using MIN circuits, which are relatively difficult to implement if current mode circuitry is used [Ota, 95]. The area selection circuit can also be implemented using neuron circuits. The neuron circuit performs summing with a threshold of the incoming signals. The threshold is set to the peak value of the membership functions. In this way, a single fuzzy variable is not able to activate the area. Only areas where the sum of two fuzzy variables is larger than the threshold are activated. In traditional fuzzifiers, the sum of the membership functions usually does not exceed the peak value of the single membership function (Fig. 5(a)). In the case of the neuron circuit (Fig. 4 and Fig. 8(b)), the membership functions must overlap, and the sum of its values must be larger - presumably 1.2-1.8 times of the peak value (Fig.5(b)). Larger overlap leads to a smoother, but less accurate, profile of the control surface.

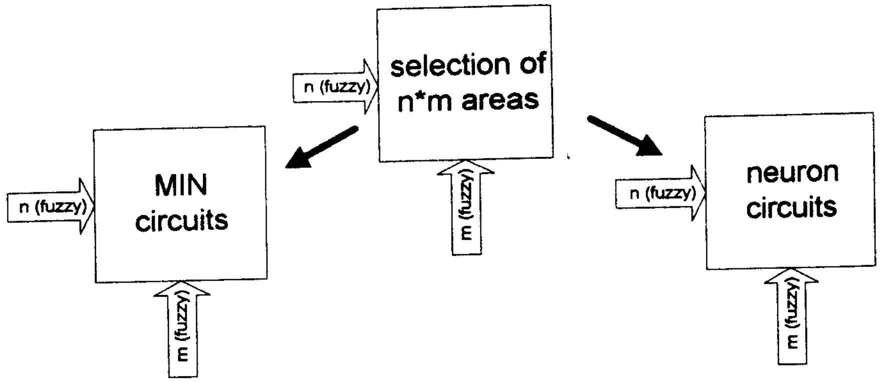

Fig. 4. Two ways of implementing area selection circuits

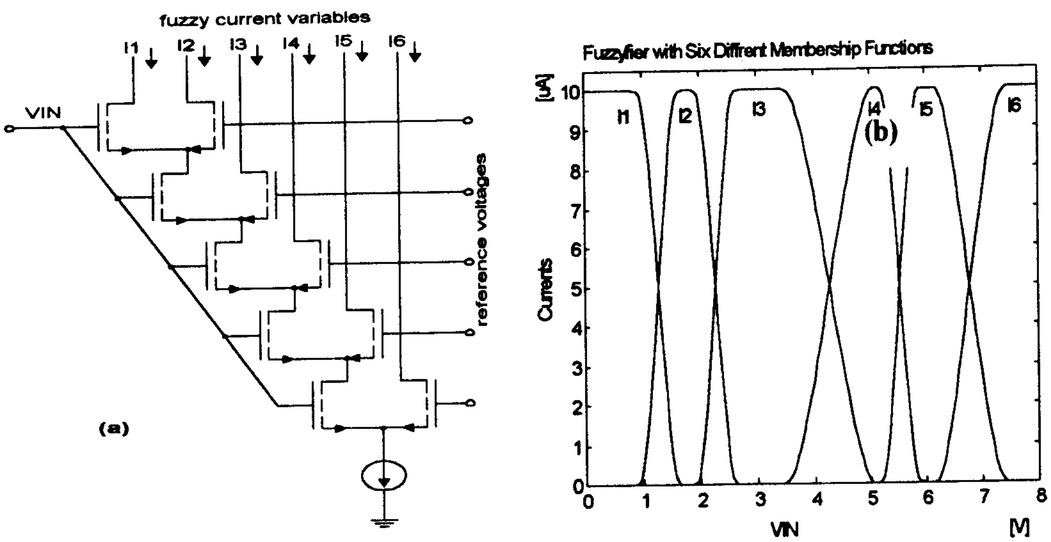

Fig. 5. Fuzzyfier (a) circuit diagram of fuzzyfier, (b) example of the SPICE simulation.

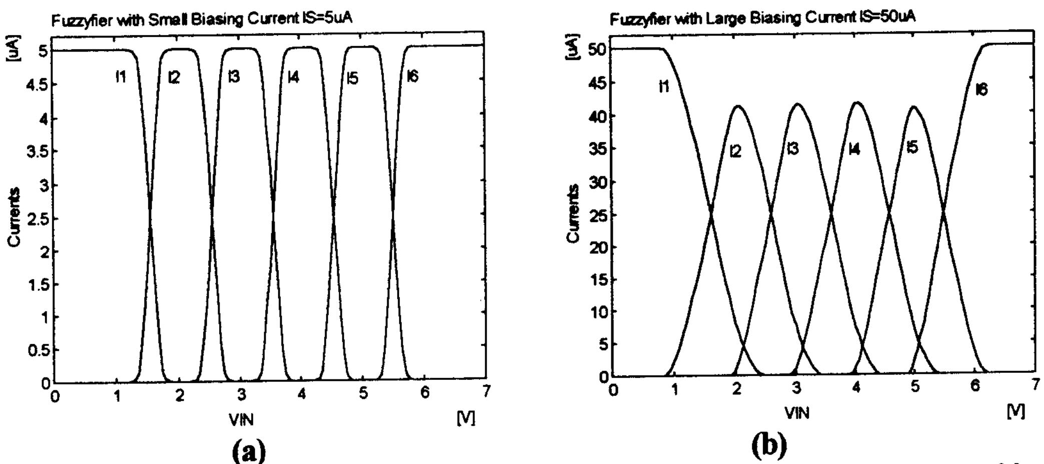

Fig. 6. Equally spaced membership functions of the fuzzifier for (a) small and (b) large biasing currents

CIRCUIT IMPLEMENTATION

The fuzzifier circuit is presented in Fig. 5(a). This unique design require only one differential pair for each membership function, in the contrast to earlier designs where at least two differential pairs were required [Angulo,95][Choi,94][Ota,95]. An example of SPICE simulation is shown in Fig. 5(b). Note the flexibility of choosing shapes for the membership functions. Because the circuit is supplied from a single current source, the sum of all fuzzy variables (currents) in this case is always constant. By choosing different reference voltages and different W/L ratios, any trapezoidal type of membership function can be accomplished. 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.

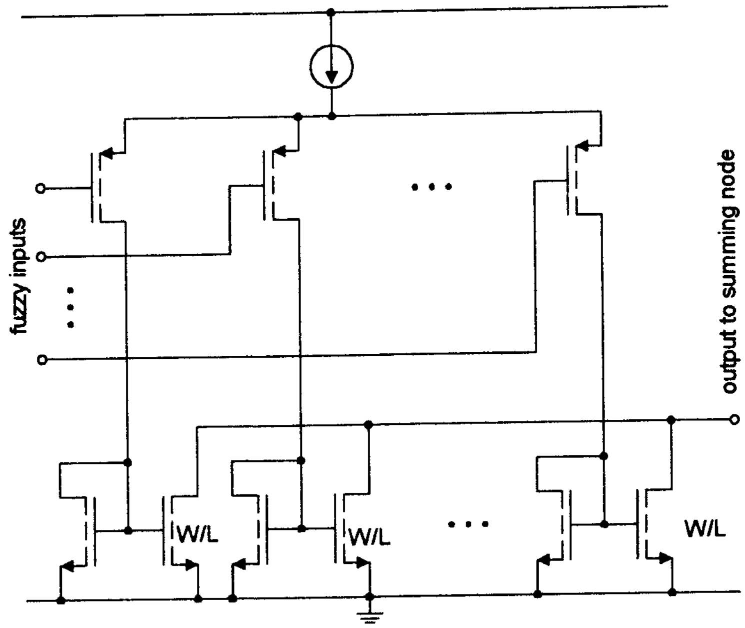

Fig. 7. defuzzifier using normalization and weighted sum.

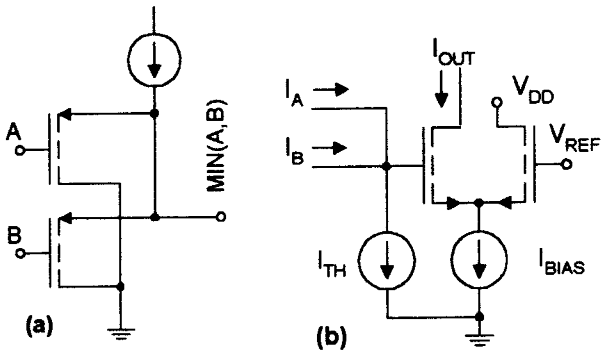

Fig. 8. Selection circuits (a) MIN circuit in voltage mode (b) neuron circuit with threshold in the current mode

A MIN circuit operating in the voltage mode is shown in Fig. 8. Its equivalent operating in the current mode was already described by [Ota, 95].

CIRCUIT SIMULATION TOOLS

The typical size of the array of fuzzy rules is large, ranging from 30 to 100 cells, which leads to the 1000 to 3000 transistors on the VLSI chip. All transistors operate in the analog mode, and a SPICE type program would be required for simulation. Some subcircuits can be easily tested with SPICE, but it would not be possible to use the SPICE program to test entire circuits, namely for verification of the shape of the control surface. Although analytical formulas exist for the fuzzifier circuits, proper design requires repetitive simulation of the circuits. The computation time can be reduced by two or more orders of magnitude using a specially developed simulator with derived analytical equations. Fortunately the architectures do not use feedback loops, and all signals are feedforward. Therefore, it was possible to develop a special simulation tool using the MATLAB environment and simple transistor models. The control surface depends not only on shape of the membership locations and singleton values of the defuzzifier but also depends on the biasing currents and W/L ratio of transistors. Different program codes were written for each proposed architecture.

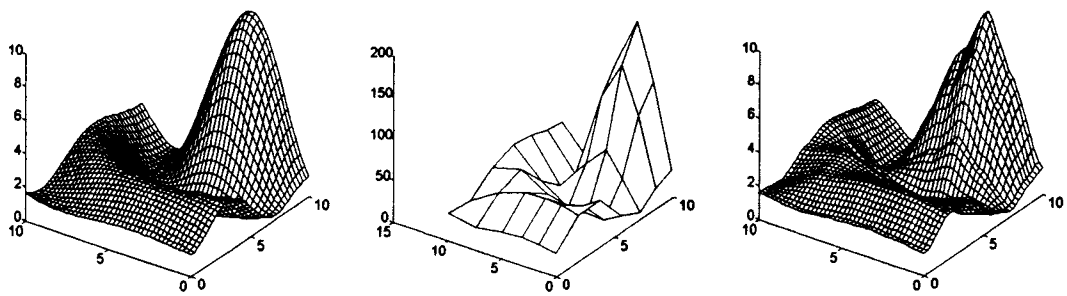

Fig. 9. Control surfaces: (a) desired control surface, (b) information stored in defuzzifier as weights, and (c) example of a control surface obtained from fuzzy system using weights of Fig. 9(b).

Figure 9 provides an example of an approximator designed using this software. Figure 9(a) presents the desired function. The design uses six X, and seven Y, unequally spaced, membership functions. Only the information defining the nodes of Fig. 9(b) is stored in the system. Fig. 9(c) shows the control surface obtained from the approximator. The shape of control surface is smoothed by choosing proper source currents in the fuzzifier circuits. Large currents lead to overlapping of membership functions and to spatial averaging of the output variable. Small values of fuzzifier currents lead to step-like of responses.

VLSI IMPLEMENTATION

Current mode operation for MIN operators or neuron circuits (Fig. 8(b)) requires a prohibitive number of current sources, and they are not practically realizable on a VLSI chip. Therefore the voltage mode MIN operators and fuzzy architecture shown in Fig.10 were actually implemented. In order to make the chip universal, each fuzzifier consist seven differential pairs with seven equally spaced reference voltages. This results in eight membership function for each input and 64 rules in the rule table. Sixty-four adjustable current mirrors for setting an output signal for each rule (Fig. 7) are programmed with a six bit accuracy (this part of the circuit is not shown in the Figure). For an arbitrary two-dimensional function only 6*64=384 bits are required for programming. The circuit is being implemented in the 2 μm low noise n-well MOSIS process using more than 2000 transistors to perform the analog signal processing.

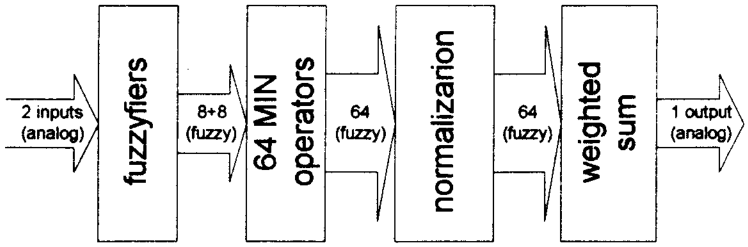

Fig. 10. Block diagram of the fuzzy approximator implemented on the VLSI chip.

CONCLUSION

The proposed approach represents a further simplification of fuzzy controllers. By setting an additional requirement that the sum of all membership functions for each input variable must equal one leads to significant simplification of the rule selection process. In this paper the technique was used for VLSI implementation, but the proposed approach is more general and can be used to simplify microprocessor based systems.

REFERENCES

Angulo J.R., (1995) A BiCMOS Universal Membership Function Circuit with Fully Independent, Adjustable Parameters, *Proc. IEEE International Symposium on Circuits and Systems*, pp. 275-278.

Choi J., B.J.Sheu, and J.C.F. Chang, (1994) A Gaussian Synapse Circuit for Analog VLSI Neural Networks. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. **2**, no. 1, pp. 129-133.

Colodro F., A. Torralba, and L.G.Franquelo, (1995) A Circuit for Learning in Fuzzy Logic-based Controllers, *Proc. IEEE International Symposium on Circuits and Systems*, pp. 279-282.

Kosko B., (1993) *Fuzzy Thinking, The New Science of Fuzzy Logic*. Hyperion, New York.

Liu B.D., C.Y. Huang, H.Y. Wu, (1994) Modular Current Mode defuzzification Circuit for Fuzzy Logic Controlers, *Proc. 3rd International Conference on Fuzzy Logic*, Neural Networks, and Soft Computing, Iizuka, Japan.

Ota Y., B.M.Wilamowski, (1995) CMOS Implementation of a Current-Mode Fuzzy Min-Max Controller, *Proc. of World Congress on Neural Networks*, Vol. **2**, pp. 480-483, July 17-21.

Ota Y., B.M.Wilamowski, (1996) Analog Hardware Implementation of a Voltage-Mode Fuzzy Min-Max Controller, *Journal of Circuits, Systems, and Computers*, Vol. **6**, No.2, pp. 171-184,

Ramirez-Angulo J., (1995) A BiCMOS Universal Membership Function Circuit with Fully independent, Adjustable Parameters, *IEEE International Symposium on Circuits and Systems*, Seattle WA, vol. **1**, pp.275-278, April 30-May 3.

Rodriguez-Vazquez A. and F. Vidal-Verdu, (1995) Learning in Neuro/Fuzzy Analog Chips, *IEEE International Symposium on Circuits and Systems*, Seattle WA, vol. **3**, pp.2325-2328, April 30-May 3.

Takagi T. and Sugeno, (1989) Derivation of Fuzzy Control Rules from Human Operator's Control Action. *Proc. of the IFAC Symp. on Fuzzy Inf. Knowledge Representation and Decision Analysis*, pp. 55-60, July.

Zadeh L. A.,(1965) Fuzzy sets. *Information and Control*, New York, Academic Press vol **8**, pp.338-353.