CMOS FUZZY CONTROLLERS IMPLEMENTED AS MIXED-SIGNAL ICS

I. Baturone, S. Sánchez Solano, A. Barriga, J. L. Huertas

Instituto de Microelectrónica de Sevilla - Centro Nacional de Microelectrónica
Avda. Reina Mercedes s/n, (Edif. CICA)
E-41012, Sevilla, Spain

1996 IEEE International Symposium on Circuits and Systems (ISCAS'96), Vol. 3, pp. 422-425, Atlanta, May 12-15, 1996.

CMOS Fuzzy Controllers Implemented as Mixed-Signal ICs

I. Baturone, S. Sánchez-Solano, A. Barriga, and J. L. Huertas.

Dept. of Design of Analog Circuits. Centro Nacional de Microelectrónica.
Edificio CICA, Avda. Reina Mercedes s/n, 41012-Sevilla (Spain). e-mail: lumi@cnm.us.es

Abstract

This paper describes a programmable fuzzy controller chip designed with mixed-signal IC techniques. Its input and output signals are analog to directly interact with the information from the real world. The programmability interface is digital and the output signal is also given in digital format to allow easy embedding into digital processing environments. Experimental results from a prototype integrated in a 2.4-μm CMOS process are included.

I. Introduction

Fuzzy logic has raised a great interest especially in the field of heuristic control. Those applications which require real-time control and/or low area occupation and power consumption demand hardware realizations of fuzzy controllers (fuzzy chips). The kind of operation involved in a fuzzy inference system (addition, bounded difference, scaling, etc.) and the inherent low precision of approximate reasoning make analog circuits very suited for fuzzy chips, in particular those circuits based on the current-mode techniques [1-3].

The control surface provided by a fuzzy controller depends on the way of covering the input and output universes of discourse with the membership functions of fuzzy sets and the way of relating input (antecedents) with output (consequents) via "IF-THEN" rules. Some reported analog realizations provide fixed control surfaces [3-4], while in other proposals these surfaces are programmable [5-6]. In the last group, the analog circuitry performs the computing operations and allows a direct communication with the signals of the real world. The digital part facilitates the programming interface and provides a simple and safe mechanism to storage the programmable parameters.

A significant advantage of a programmable fuzzy chip is its capability to solve different control problems. It can be adapted to provide a static control surface or can be dynamically tuned in complex applications which require time-varying control surfaces. In these cases, one of the following tuning mechanisms, illustrated in Figure 1, can be employed: (a) off-chip manual tuning, (b) off-chip automatic tuning, governed by a standard computer (in this case, the output of the fuzzy controller is usually considered by the tuning algorithm, so that a digital format of the output is also convenient), and (c) on-chip automatic tuning.

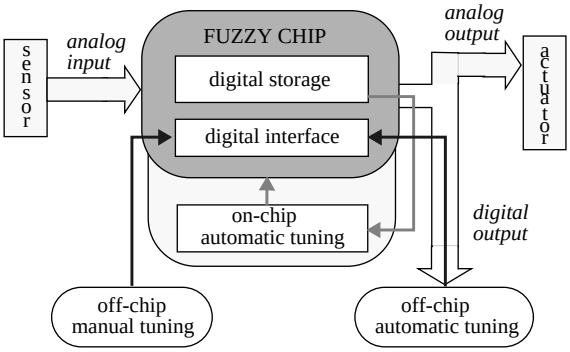

Fig. 1: Work environment of a programmable fuzzy chip.

This paper describes the architecture and building blocks of a CMOS programmable fuzzy chip which implements the Singleton or zero-order Sugeno´s Method [5]. It accepts analog input signals and digital words to program the control surface, and provides the output in either a digital or analog format. These features make it possible off-chip manual or automatic techniques and allow the microcontroller to interact with analog or digital subsequent systems. The mixed-signal circuits to implement the different building blocks are based on current-mode data converters and digital registers. To illustrate these ideas, experimental results of a two-input one-output prototype with nine rules are included.

To appear in the 1996 *Proceedings o f the IEEE Int. Conference on Circuits and Systems* (ISCAS'96)

II. Architecture

The architecture of a fuzzy microcontroller greatly depends on the fuzzy inference method. Since the circuitry required is proportional to the complexity of the method, the trend is to implement simplified methods that define the membership functions of the consequent fuzzy sets with a few parameters. Among these simplified methods, the chip described in this paper implements the Singleton or zero-order Sugeno´s Method. In this method, a consequent fuzzy set is represented by a single value, c_i, that is related with the position of the consequent membership function. Each of these singletons is weighted by the sum of the activation degrees, h_i, of the rules where they appear. The overall output results from a weighted average extended to the number of rules, r :

$$\text{output} = \frac{\Sigma_{i=1}^{r} h_i c_i}{\Sigma_{i=1}^{r} h_i} \qquad (1)$$

The building blocks required are the following: (a) membership function circuits (MFCs) to provide the membership degree of the input signals to the fuzzy sets of the antecedents; (b) connective operators (MIN, MAX) to obtain the activation degrees of the rules, h_i; (c) circuits to define the singletons of the consequents (CONSs), which also implement the fuzzy implication operation (the product $h_i c_i$); and (d) a divider (DIV) to calculate the output. Since internal signals in the chip are represented by currents, addition is reduced to connecting wires.

Unlike other analog realizations, where each rule has its own MFCs, MIN, and CONS blocks, the architecture here selected shares the MFCs and CONS blocks by all the rules. Figure 2 illustrates this matrix-like architecture for the case of two input signals. This solution is advantageous when the number of rules

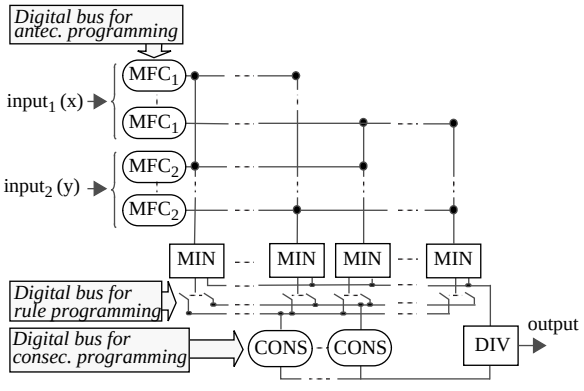

Fig. 2: Matrix-like architecture with rule and membership functions programmability.

y \ x	N	Z	P
N	P	P	Z
Z	P	Z	N
P	Z	N	N

Fig. 3: Typical set of rules used by many fuzzy controllers. P, Z, and N design input and output fuzzy labels.

is large compared to the number of input fuzzy sets and output singletons. This is the case when the complete fuzzy rule set is implemented on-chip, as happens in the prototype we have designed. This prototype has two inputs that are covered by three fuzzy sets, thus giving nine possible rules. The position and shape of each input membership function are programmed by an 8-bit digital word. The output universe of discourse is also covered by three consequents, that is, three singletons, which are programmed by a 4-bit word. Sharing consequents avoids the possible repetition of CONS blocks but requires additional information to define the consequent of each rule, which is the purpose of the digital bus for rule programming shown in Figure 2. In the prototype that has been integrated, this information is not required because the rule set was fixed to the typical disposition shown in Figure 3.

III. Building Blocks

Membership function circuits:

Input fuzzification is realized according to the expression:

$$MFC(I_{in}) = I_{ref} \ \Theta\ m(\mid I_{in} - I_{aux} \mid \Theta\ I_{sat}) \qquad (2)$$

where Θ is a rectification or bounded difference operator.

The resulting triangular or trapezoidal membership functions are defined by the parameters I_{aux} (central point of the fuzzy label), I_{sat} (saturation of the trapezoidal function), and m (the slope of the membership function). I_{ref} defines the maximum degree of pertenence and I_{in} is the input signal. The MFC used to implement equation (2) (Figure 4) is described in [7]. It contains three current-mode D/A converters to program the position (I_{aux}) and shape (I_{sat}, m) of the membership functions. To also accept input signals represented by voltages, the V-to-I converter reported in [8] has been employed. Figure 5 shows experimental results of the fuzzification stage in our prototype. They illustrate different ways of covering the input universe of discourse by changing the slope of the central triangular label and the position of the extreme trapezoidal labels.

Connective circuits:

The activation degrees of the rules are calculated by current-mode MAX operators based on the structure

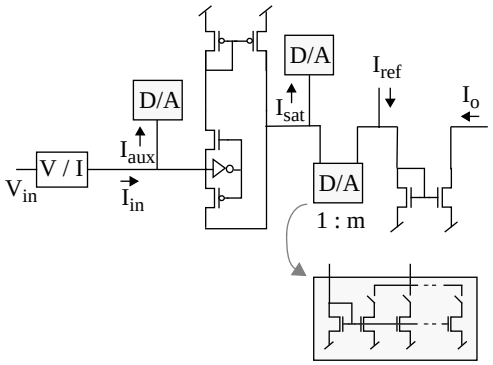

Fig. 4: Schematic of a digitally programmable MFC.

proposed by the authors and also described in [7] (Figure 6a). They perform a minimum operation, according to De Morgan´s law. Figure 6b shows experimental results that illustrate the minimum operation calculated between a central label and different extreme labels.

Consequent circuits:

According to equation (1), all the activation degrees, h_i, associated with the same consequent are scaled by the same singleton, c_i. The circuits (CONS) to implement this scaling are current-mode D/A converters that perform as programmable multipliers (like those shown in the dashed box of Figure 4). The wires carrying the current-mode values h_i are connected to the input of its corresponding CONS block, so that the addition operation is directly performed. Figure 7 shows experimental results that illustrate how a particular value of Σh_i is scaled by different values of c_i.

Divider circuit:

The final division in the current domain is performed by a current-mode A/D converter. The input to the converter is the numerator current, $\Sigma h_i c_i$, while the reference current is the denominator value, Σh_i [9]. This makes the output digital word represent the analog division:

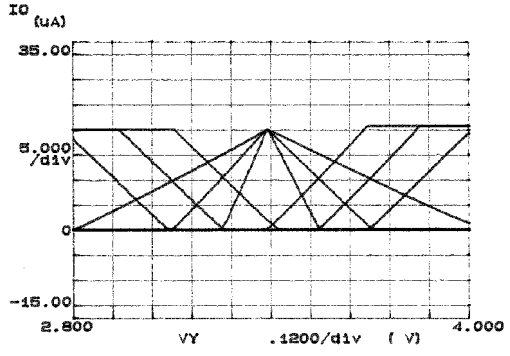

Fig. 5: Experimental results showing the programmability of the fuzzification stage.

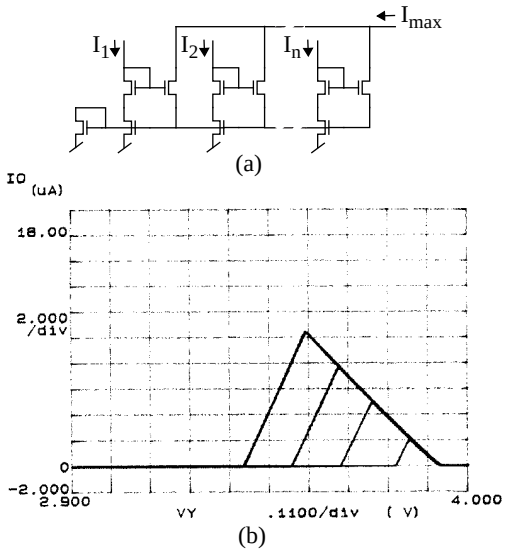

Fig. 6: (a) Schematic of a current-mode multiple-input MAX circuit. (b) Experimental results of the connective stage.

$$w_o = (\Sigma_{i=1}^{n} b_i 2^{-i}) = \frac{I_{num}}{I_{den}} = \frac{\Sigma h_i c_i}{\Sigma h_i} \qquad (3)$$

An algorithmic A/D converter has been selected. It consists of a cascade of identical alternating bit cells, as shown in Figure 8. The current comparator proposed in [10] is used with a double functionality: to compare the input current to the bit cell, I_i, with the reference, $I_{den}/2^i$, and to convey the current $I_i - I_{den}/2^i$ (if I_i exceeds $I_{den}/2^i$) to the following cell.

A 5-bit A/D converter has been included in our prototype, together with a current-mode D/A converter to also provide an analog output. Figure 9 shows the quantized analog signals at the microcontroller output (a resistive load was connected to convert current to voltage). These results are obtained by sweeping one of the input signals and fixing the other to three different values. The singletons were not superior to 0.5

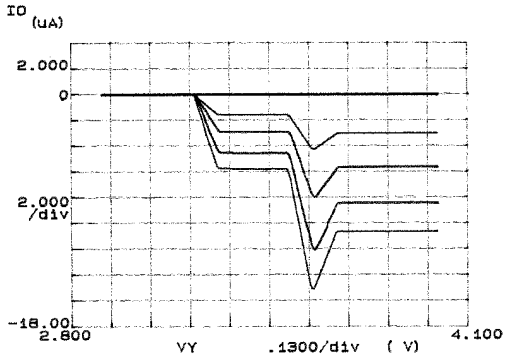

Fig. 7: Experimental results showing the programmability of the singleton in a consequent.

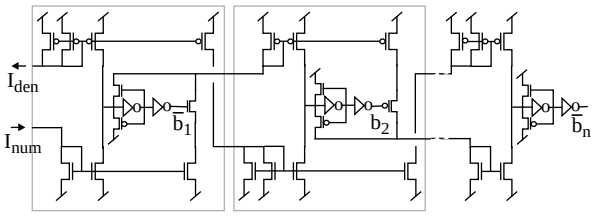

Fig. 8: Schematic of the current-mode algorithmic A/D converter used to implement the division.

($I_{num}/I_{den} \leq 0.5$), so that the whole output range is quantized in 17 steps (from 00000 to 10000 code).

IV. Concluding Results

The die photograph of the integrated microcontroller is given in Figure 10. It occupies an active area of 1mm x 1.1mm in a 2.4-µm CMOS process. Input membership functions and consequent singletons can be programmed to provide different control surfaces. As an example, Figure 11 illustrates one of them obtained from experimental results. In this case, input currents to the fuzzification stages (I_x, I_y) range from 0 to 32µA, the output current (I_o) ranges from 0 to 40.5µA, and the time response was estimated about 2µs.

V. References

[1] T. Yamakawa, T. Miki, F. Ueno, "The Design and Fabrication of the Current Mode Fuzzy Logic Semi-Custom IC in the Standard CMOS IC Technology", *Proc. IEEE Int. Symp. Multiple-Valued Logic*, 1985, pp. 76-82

[2] I. Baturone, S. Sánchez-Solano, J. L Huertas, "Current-Mode Singleton Fuzzy Controller", *Proc. 3rd. Int. Conf. on Fuzzy Logic, Neural Networks and Soft Computing*, Iizuka, 1994, pp. 647-648.

[3] L. Lemaitre, M. J. Patyra, D. Mlynek, "Analysis and Design of CMOS Fuzzy Logic Controller in Current Mode", *IEEE J. Solid-State Circuits*, Vol. 29 (3), March 1994, pp. 317-322.

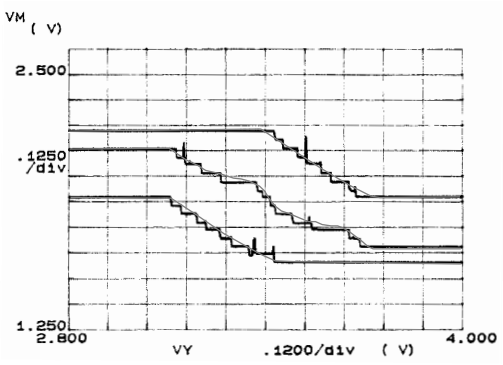

Fig. 9: Quantized analog signals measured at the microcontroller output. The dashed lines represent the ideal division values.

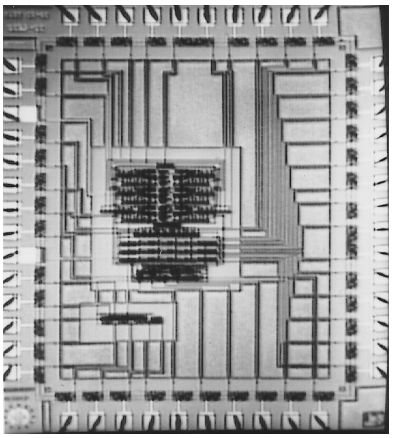

Fig. 10: Microphotograph of the CMOS microcontroller.

[4] N. Manaresi, E. Franchi, R. Guerrieri, G. Baccarani, "A Modular Analog Architecture for Fuzzy Controllers", *Proc. ESSCIRC*, 1994, pp. 288-291.

[5] T. Yamakawa, "A Fuzzy Inference Engine in Nonlinear Analog Mode and its Application to a Fuzzy Logic Control", *IEEE Trans. on Neural Networks*, Vol. 4 (3), May 1993, pp. 496-522.

[6] J. W. Fattaruso, S. S. Mahant-Shetti, J. B. Barton, "A Fuzzy Logic Inference Processor", *IEEE J. Solid-State Circuits*, Vol. 29 (4), April 1994, pp. 397-402.

[7] I. Baturone, S. Sánchez-Solano, A. Barriga, and J. L. Huertas, "Implementation of CMOS Fuzzy Controllers as Mixed-Signal Integrated Circuits", to appear in the Special Issue on Hardware Implementations of the IEEE Trans. on Fuzzy Systems.

[8] K. Bult, H. Wallinga, "A Class of Analog CMOS Circuits Based on the Square-Law Characteristic of an MOS Transistor in Saturation", *IEEE J. of Solid-State Circuits*, Vol. 22 (3), June 1987, pp. 357-365.

[9] S. Sánchez-Solano, I. Baturone, A. Barriga, J. L. Huertas, "A Current-Mode Memory and Computing Block for Advanced Fuzzy Chips", *VI IFSA World Congress*, 1995, pp. 181-184.

[10] R. Domínguez-Castro, A. Rodríguez-Vázquez, F. Medeiro, J. L. Huertas, "High Resolution CMOS Current Comparators", *Proc. ESSCIRC*, 1992, pp. 242-245.

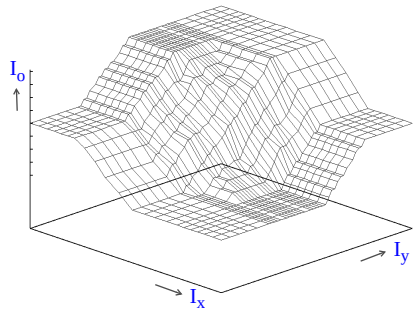

Fig. 11:Experimental control surface for a particular set of programmable parameters.