

A Modular Programmable CMOS Analog Fuzzy Controller Chip

*Angel Rodríguez-Vázquez **, Rafael Navas *, Manuel Delgado-Restituto ** and
Fernando Vidal-Verdú **

*Dpto. de Electrónica
Universidad de Málaga
Complejo Tecnológico, Campus de Teatinos, Málaga, SPAIN
FAX: 34 5 2132782
Phone: 34 5 2133326
email: vidal@ctima.uma.es

**Instituto de Microelectrónica de Sevilla
Centro Nacional de Microelectrónica-C.S.I.C.
Edificio CICA-CNM, Avda. Reina Mercedes s/n, 41012-Sevilla, SPAIN
FAX:: 34 5 4231832
Phone:: 34 5 4239923
email: angel@imse.cnm.es

ABSTRACT: We present a highly modular fuzzy inference analog CMOS chip architecture with on-chip digital programmability. This chip consists of the interconnection of parameterized instances of two different kind of blocks, namely label blocks and rule blocks. The architecture realizes a lattice partition of the universe of discourse, which at the hardware level means that the fuzzy labels associated to every input (realized by the label blocks) are shared among the rule blocks. This reduces the area and power consumption and is the key point for chip modularity. The proposed architecture is demonstrated through a 16-rule, two-input CMOS 1 μ m prototype which features an operation speed of 2.5Mflips (2.5 x 10⁶ fuzzy inferences per second) with 8.6mW power consumption. Core area occupation of this prototype is of only 1.6mm² including the digital control and memory circuitry used for programmability. Because of the architecture modularity the number of inputs and rules can be increased with hardly design effort.

**Submitted to IEEE Trans. Circuits and Systems - II
December 1997**

Acknowledgements: The works in this paper as been partially funded by the spanish C.I.C.Y.T under contract TIC96-1392-C02-02 (SIVA).

A Modular Programmable CMOS Analog Fuzzy Controller Chip

I. INTRODUCTION

Fuzzy controllers are used to map a *multidimensional* input signal $\mathbf{x} = \{x_1, x_2, \dots, x_M\}^T$ into a scalar output y , in accordance to a well-defined *nonlinear* relationship [1],

$$y = f(\mathbf{x}) \quad (1)$$

In control applications the inputs are usually called *facts*, the output *action*, and the mapping law *surface response*. For instance, a fuzzy controller for a washing machine must univocally set the water level (action) as a nonlinear function (surface response) of the clothes' mass, the water impurity, and the time differential of impurity (facts) [2].

Fuzzy controllers employ the procedure of *fuzzy logic inference* [1] to construct the surface response. Some characteristic features of this procedure are [3][4]:

- The surface response, which is a *global* model predicting the system behavior for any input, is obtained as a composition of *local* functions, each one predicting this behavior only for inputs comprised in a limited region of the input space.
- These local functions represent *insights* on the system operation, and are described through inference *rules* of the type,

IF x_1 is A_{1k} AND x_2 is A_{2k} AND ... x_M is A_{Mk} THEN *Consequent Action*

where A_{ik} are called *fuzzy labels*, and the consequent action assigns values to y depending on the outcome of the combination of the antecedent clause statements.

- The validity of the statements “IF x_i is A_{ik} ” is continuously graded from 0 to 1; the actual grade of each statement is calculated by evaluating a nonlinear *membership function* $s_{ik}(x_i)$ which is different from zero only inside a subinterval of the whole x_i interval.

Because the statements involved in the fuzzy rules are in natural language, for instance “if the temperature is low”, this modeling technique is very well suited to capture and emulate human expertise. On the other hand, the continuous grading guarantees *generalization* of the local pieces of knowledge and hence, smooth surface responses. Finally, any change which affects only a limited region of the input space can be easily incorporated to the global model by just modifying the affected local functions – *transparency* property [3].

There are many fuzzy controller applications where the inputs and the output are *analog* signals [1][2]. The hardware required for these applications can be realized in two alternative ways. One employs analog circuitry only at A/D and D/A conversion interfaces, while the fuzzy processing is realized in digital domain by either general-purpose processors or dedicated ASICs [5]-[8]. The other realizes the fuzzy processing itself in the analog domain, and employs the digital circuitry for *programmability* and reconfigurability [9]. This paper contributes to the latter approach. Generally speaking, this approach is expected to feature larger operation speed,

lower power consumption and smaller area occupation than the other [10][11]. These expectations are confirmed by the techniques presented in this paper, which fully exploit the functional capabilities of the MOS transistor (MOST) to realize the fuzzy operators with very simple circuitry. An inherent disadvantage of analog fuzzy controllers is limited precision. However, precision levels well below 10% – sufficient for many practical applications [2][9] – can be obtained by proper modeling of the errors [12], and sound circuit design techniques.

Circuit blocks and design techniques for CMOS analog fuzzy controllers have been reported elsewhere [13]-[17]. Some of them have been demonstrated through actual *monolithic* chips, a fraction of which include programmability [15]-[17]. However, although precision is a weak point of the analog approach, most previous contributions do not consider the accuracy issue during the design phase. Whenever small transistors sizes are used – a technique inherited from digital IC design [15] – the output signal may become largely erroneous. These errors might be attenuated by post-fabrication tuning of some critical parameters, guided by learning processes [18]. Our chip overcomes this drawback by considering the accuracy issue during the design phase so that the desired performance can be anticipated inside specified error margins. Thus, our chip can be programmed in robust and transparent way.

Despite the incorporation of accuracy considerations (hence, the avoidance of minimum transistors sizes) and although the demonstration prototype in this paper implements more rules than previous analog prototypes, it features much smaller *Delay* × *Power* values: 470ns@8.6mW@16rules versus 570ns@44mW@9rules [16] and 160ns@550mW@13rules [17]. Programmability is also a quality of our proposal, which incorporates internal memory for serial digital programming of the rule consequents, and allows external analog programming of the membership functions. This is also advantageous as compared to [16], where the consequents values have to be learned using software models of the controller, and are not stored on-chip. Finally, the modular chip organization around two high level building blocks easily identified from the user and designer point of view, renders our chip architecture feasible for silicon compilation.

II. CHIP ARCHITECTURE

The chip realizes a type of fuzzy inference where the rule consequents are constant values,

$$\text{IF}(x_1 \text{ is } A_{1k})\text{AND}(x_2 \text{ is } A_{2k})\text{AND}\dots(x_M \text{ is } A_{Mk})\text{THEN } y = y_k^* \quad 1 \leq k \leq N \quad (2)$$

These values y_k^* are called *singletons*. As compared to the general case where the consequents include fuzzy labels [1], this type of fuzzy inference requires much less complex hardware [9], and, thus, less silicon area and less electrical power. Besides, it increases the transparency of the rules and, thus, eases the incorporation of programmability. On the other hand, different studies show that singleton fuzzy controllers are *universal approximators*; i.e. they are capable to approximate any surface response by properly choosing the rules and singletons [3][4].

The set of membership functions $s_{ik}(x_i)$ constitutes the elementary nonlinearities from which the surface response of a fuzzy controller is built. Fig.1(a) shows a typical membership function shape [4] – described by three parameters: *width* (2Δ), measured as the length of the interval defined by the *crossover points*; *center* (E_C), the central point of this interval; and *slope* (ζ), the absolute value of the function slope at the crossover points.

For a complete controller description, the surface response formula has to be generated from these elementary nonlinearities. Fig.1(b) illustrates the building procedure for a one-dimension, four-rules controller. Here, each rule involves only a fuzzy label, “IF x is A_k THEN $y = y_k^*$ ”, whose validity is evaluated by using the corresponding membership function $s_k(x)$. If the actual input x is at the center of the interval I_k for the k -th membership function, then $s_k(x) = 1$ and the output is given by the value of the k -th singleton $y = y_k^*$. At any point different from the centers of the membership function intervals, the output does not coincide with any of the singletons but it is interpolated by using the following formula,

$$y = y_1^*s_1^*(\mathbf{x}) + y_2^*s_2^*(\mathbf{x}) + y_3^*s_3^*(\mathbf{x}) + y_4^*s_4^*(\mathbf{x}) \tag{3}$$

where $s_k^* = s_k / \left[\sum_{k=1,4} s_k \right]^{\dagger 1}$. In this way a global response curve is built from the local data represented by the singletons, as Fig.1(b) illustrates.

In the general multi-dimensional case, the surface response is interpolated from the singletons by using multi-dimensional membership functions $w_k(\mathbf{x})$,

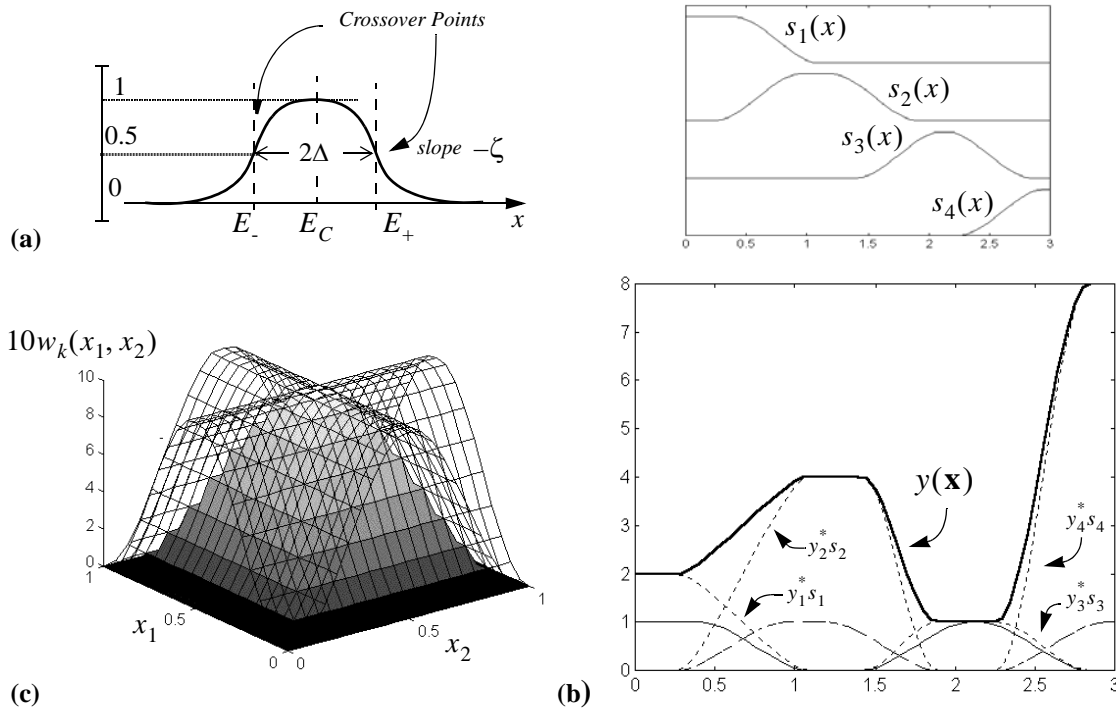


Figure 1: (a) One-dimensional membership function shape; (b) Illustrating function approximation through singleton fuzzy controllers; (c) Two-dimensional membership function.

1. This normalization precludes the output to take a value larger than the largest singleton at any point.

$$y = f(\mathbf{x}) = \sum_{k=1, N} y_k^* \frac{w_k(\mathbf{x})}{\sum_{k=1, N} w_k(\mathbf{x})} \quad (4)$$

where the function $w_k(\mathbf{x})$ is evaluated by choosing the *minimum*^{† 2} among the values of the uni-dimensional membership functions $s_{ik}(x_i)$ associated to the k -th rule,

$$w_k(\mathbf{x}) = \min\{s_{1k}(x_1), s_{2k}(x_2), \dots, s_{Mk}(x_M)\} \quad (5)$$

Fig.1(c) illustrates the build-up procedure and final shape of a two-dimensional membership function.

The fuzzy controller chip architecture of Fig.2 realizes (4) for a system with M inputs, L fuzzy labels per input and $N = L^M$ rules. The architecture is composed of the interconnection of blocks of two different types, namely: *label* and *rule*. Each fuzzy label, say A_{ij} (the j -th fuzzy label of the i -th input), has an associated label block which evaluates the corresponding

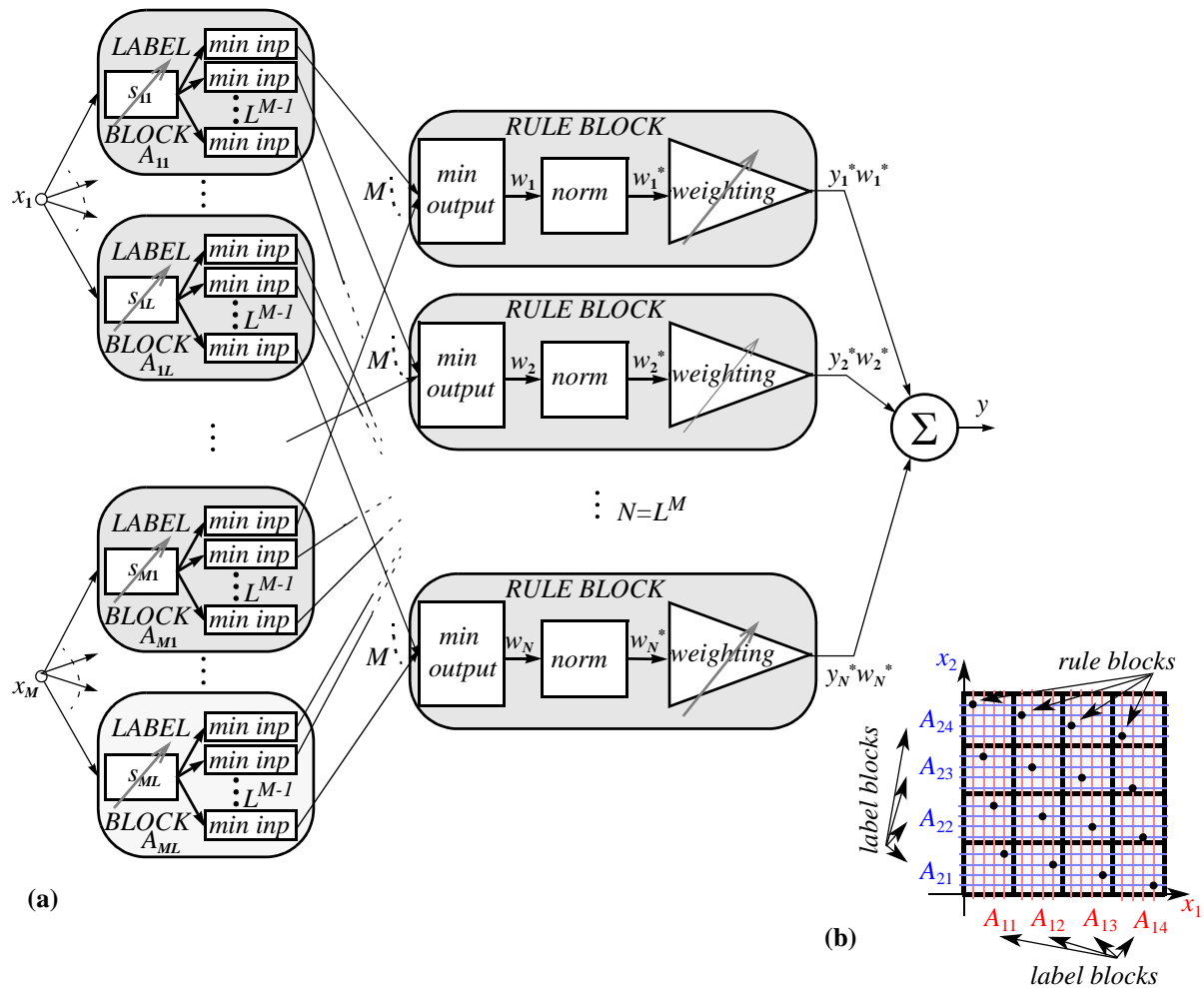


Figure 2: (a) Controller chip architecture; (b) Interconnection of label and rule blocks in the $1\mu\text{m}$ CMOS prototype.

2. This is the AND operator used in our chip. Other operators could be used as well [1]-[4].

membership function $s_{ij}(x_i)$ and generates L^{M-1} replicas of the result. These replicas are processed in the “*min inp*” sub-blocks of the label blocks to make a first step towards the realization of the minimum. Each rule blocks combines M inputs coming from the label blocks to: first, realize the second step of the minimum operation; second, evaluate the function $w_k^*(\mathbf{x})$; and, third, multiply this function by its associated singleton to obtain $y_k^* w_k^*(\mathbf{x})$. The final aggregation leading to (4) is performed at the output node.

Fig.2(b) illustrates the interconnection of label and rule blocks for a system with two inputs and four fuzzy labels per input, as it is the case for the CMOS prototype presented in Section VI of this paper. Each box in the grid corresponds to a rule, has an associated singleton value, and is defined by two labels, one per input. Each label block is shared by four different rules. Because of this membership function sharing the architecture of Fig.2(a) can only generate *lattice* partitions (see Fig.3(a)); *tree* (Fig.3(b)) and *scatter* (Fig.3(c)) partitions [4] are not allowed.

Generally speaking lattice partitions have the potential disadvantages of *curse of dimensionality* (the number of rules needed to perform a good approximation may become prohibitively large for large number of inputs) and *inappropriate generalization* (the partition granularity needed to approximate the function in a region of the input space may be inappropriate in other region). However, these potential disadvantages are not really significant for the type of problems which analog fuzzy controllers are intended for (medium-to-low complexity problems with low number of inputs and low number of rules). In this scenario, the architecture of Fig.2 features significant pros for hardware implementation, namely:

- Area and power consumption required for the implementation of the rules antecedents is smaller than in the case of scatter and/or tree partitions. This is because the replication operation is much less area- and power-demanding than the membership function evaluation itself.
- The whole architecture is highly modular and can be made to grow in very simple manner. Consequently, it is very well suited for design automation.
- Programmability can be easily incorporated.

Inputs to the chip are voltages for easier interfacing. On the other hand, the minimum and the normalization operations are realized in current domain because this requires much simpler circuitry than their voltage domain counterparts [19]. Thus, the inputs to the membership function circuits are voltages, while their outputs are currents. However, as already mentioned, the

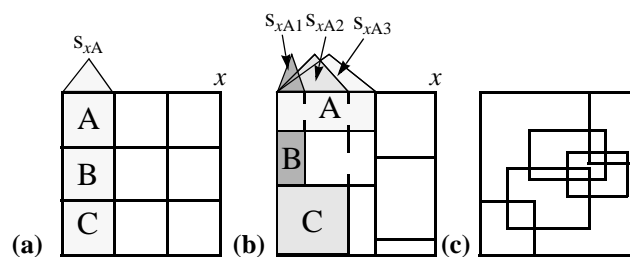


Figure 3: Examples of different types of input space partitions

label blocks do not directly deliver the membership function currents to the rule blocks; these currents are non-linearly preprocessed to produce intermediate output voltages. This simplifies the realization of the minimum operation in the rule block. Besides, transmitting these voltages (instead of the original currents) from the label to the rule blocks largely simplifies the inter-block routing as these latter blocks have only one input node (instead of M if currents were transmitted).

III. LABEL BLOCK

Each label block is driven by a component x_i of the input voltage vector to, first, obtain a membership function current $s_{ij}(x_i)$ and, second, generate L^{M-1} replicas of a voltage V_{Gij} which is a nonlinear function of this current – a preprocessing step for the realization of the minimum operator in the rule blocks. This section describes first the membership function circuitry, then the complete minimum circuitry and, finally, outlines some major design considerations to reduce systematic errors in these circuits.

III.A. Membership Function Circuitry

A few alternative realizations of the pseudo-trapezoidal function shape of Fig.1(a) have been reported in literature [15][21]-[23]. One, see Fig.4(a), consists of a cascade of a linearized transconductor, to convert the input voltage into a current, and a current-mode nonlinear block to realize the pseudo-trapezoidal shape [20]; this latter block can be realized by using the techniques proposed in [15][21][22]. A drawback of this implementation is the extra area occupation and power consumption of the linearization circuitry. Also, because the transconductor cannot be linearized in the whole input range, some of this range is wasted.

Fig.4(b) employs a slightly different strategy [23]. It uses two quasi-linear transconductance amplifiers to, at a first step, obtain monotone increasing and decreasing, respectively, currents around the crossover points; then, at a second step, these currents are first clipped and then aggregated in current domain. This strategy shares the drawbacks associated to linearization. However, as compared to Fig.4(a), it has the advantage that the centers and widths of the membership functions are controlled through voltages applied to high-input impedance nodes, which requires a simpler control circuitry and yields smaller loading errors in the application of the control signal.

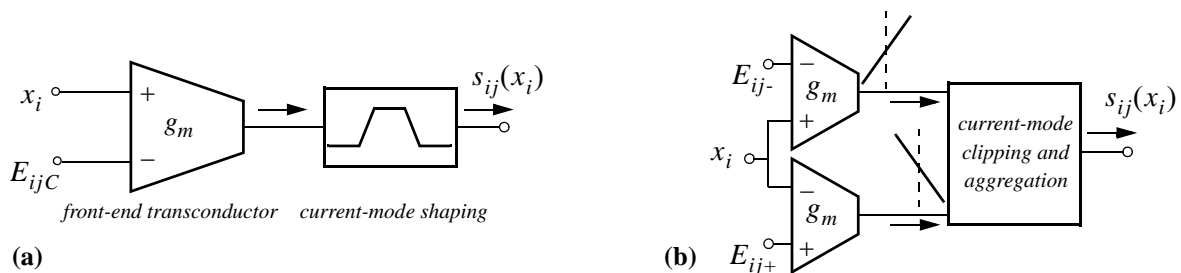


Figure 4: Concepts for the realization of a transconductance membership function by current shaping: (a) Global shapping in current-mode [20]; (b) Partial shapping in current-mode [23]

The membership function circuit used in our chip (see the shaded region at the left in Fig.5) approximates the shape of Fig.1(a) by using the nonlinear DC characteristics of a CMOS differential pair. This strategy is based on the work by Fattaruso and Meyer on CMOS function approximation [24], and was proposed for analog fuzzy design in [25]. Analysis of this circuit assuming equal differential pairs and using the square-law MOS transistor characteristics [26] obtains,

$$s_{ij} = \begin{cases} 0 & ,(x_i - E_{ij-}) < -\sqrt{\frac{I_Q}{\beta_{N1}}} \\ I_Q + \sqrt{2\beta_{N1}I_Q}(x_i - E_{ij-})\sqrt{1 - \frac{\beta_{N1}}{2I_Q}(x_i - E_{ij-})^2} & , -\sqrt{\frac{I_Q}{\beta_{N1}}} < (x_i - E_{ij-}) < \sqrt{\frac{I_Q}{\beta_{N1}}} \\ 2I_Q & , \text{otherwise} \\ I_Q - \sqrt{2\beta_{N1}I_Q}(x_i - E_{ij+})\sqrt{1 - \frac{\beta_{N1}}{2I_Q}(x_i - E_{ij+})^2} & , -\sqrt{\frac{I_Q}{\beta_{N1}}} < (x_i - E_{ij+}) < \sqrt{\frac{I_Q}{\beta_{N1}}} \\ 0 & ,(x_i - E_{ij+}) > \sqrt{\frac{I_Q}{\beta_{N1}}} \end{cases} \quad (6)$$

where β_{N1} is the large signal transconductance factor of the transistors in the differential pairs^{†3} and we assume that the membership function width is large enough to allow the output current reaching the logic unit value (I_Q) at the center.

This membership function circuit shares the advantages of Fig.4(b) regarding control of the centers and widths through voltages applied to high-input impedance terminals,

$$2\Delta_{ij} = E_{ij+} - E_{ij-} \qquad 2E_{ijC} = E_{ij+} + E_{ij-} \quad (7)$$

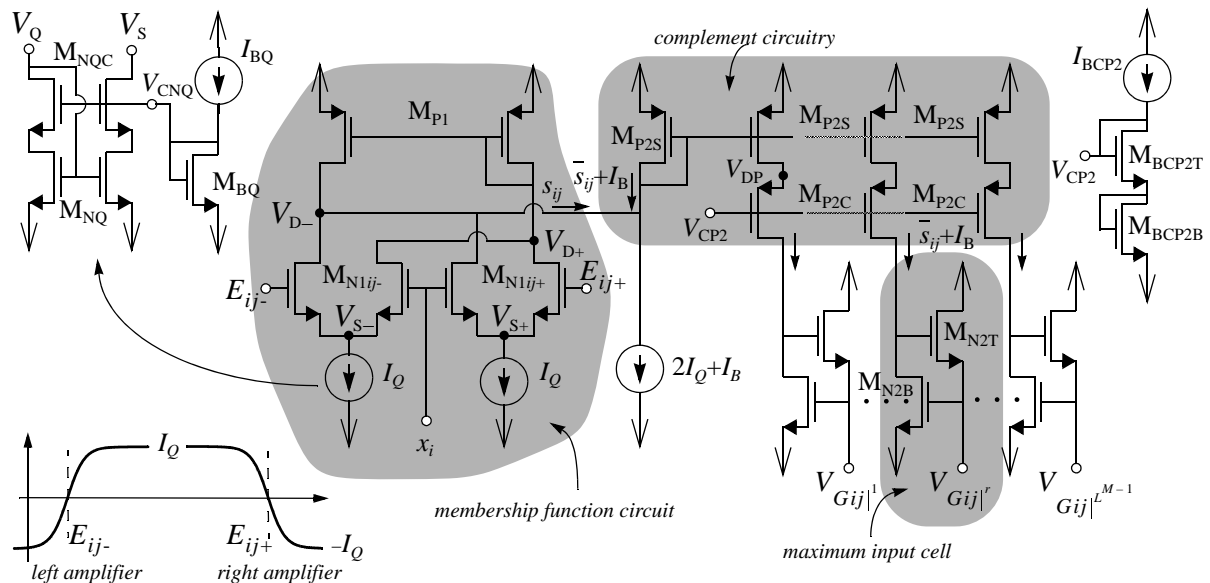


Figure 5: The Label Block.

3. This equation shows the simplest case where the positive and negative input transistors are equal.

On the other hand, the slope at the crossover points ζ_{ij} is controlled by the large signal transconductance of the MOS transistor⁴,

$$\zeta_{ij} = \sqrt{2I_Q\beta_{N1}} \quad (8)$$

The main advantage of this membership function circuit is that it does not require any linearization circuitry – why to linearize if the whole behavior is nonlinear?. Thus, it features minimum area occupation and power consumption, and full usage of the transconductor input dynamic range. On the other hand, it has been shown that the shape in (6) can actually realize the universal approximation feature, even when parasitics (systematic as well as random) are taken into account [18].

Considerations about the main non-idealities that influence the membership function circuitry, and the design strategies adopted to reduce their influence, are presented in subsequent sections. However, because they are influenced by the preprocessing circuitry used for the minimum operation, we will describe this circuitry first.

III.B. Minimum Circuitry

As mentioned in Section II, the minimum operation is realized in three steps: two in the label blocks and other in the rule block. However, for clarity, these three steps are described as a whole in this section.

The whole operation of the minimum circuit is to select-and-propagate the minimum among a set of M input currents $s_{ik}(x_i)$. However, for convenience, we do not directly select the minimum among the input currents, but the maximum among their *fuzzy complements*,

$$\overline{s_{ik}(x_i)} \equiv 2I_Q - s_{ik}(x_i) \quad (9)$$

where the current level $2I_Q$ corresponds to the logic “1”. This is based on the De Morgan’s law [39],

$$w_k(\mathbf{x}) = \min\{s_{1k}(x_1), s_{2k}(x_2), \dots, s_{Mk}(x_M)\} = \overline{\max\{s_{1k}(x_1), s_{2k}(x_2), \dots, s_{Mk}(x_M)\}} \quad (10)$$

and takes advantage of the larger simplicity of the current-mode maximum circuitry [27].

Fig.6(a) shows conceptual circuits to evaluate the fuzzy complements by KCL, for positive (entering to) and negative (leaving from) currents. Regarding the maximum circuit itself, several alternatives appear which have to be evaluated bearing in mind the following major architectural features:

- Neither constraints nor penalties should be imposed to the number of inputs since it coincides with the number of controller inputs.

4. Using the bias current to control the slope is not convenient because the bias current set the logical value “1”.

- The inter-block routing should be the smallest possible for increased modularity.

These considerations lead us to discard realizations with $O(n^2)$ complexity [28]. Realizations based on sequential binary selection trees [29] are also discarded because, although they have $O(n)$ complexity, their implementation requires $\log_2(n)$ circuit layers, and causes the errors and delays to be accumulated proportionally to the number of inputs. The maximum circuit used in our chip (see Fig.6(b)) is based on the winner-take-all circuit by Lazzaro [30] and was proposed in [25]. Its steady-state circuit operation is simple: the bottom transistor driving the maximum current will force the common voltage V_G through its associated top transistor, while the remaining bottom transistors are driven into ohmic region to comply with their input currents and, consequently, their associated top transistors are cut-off. Then, provided the out-

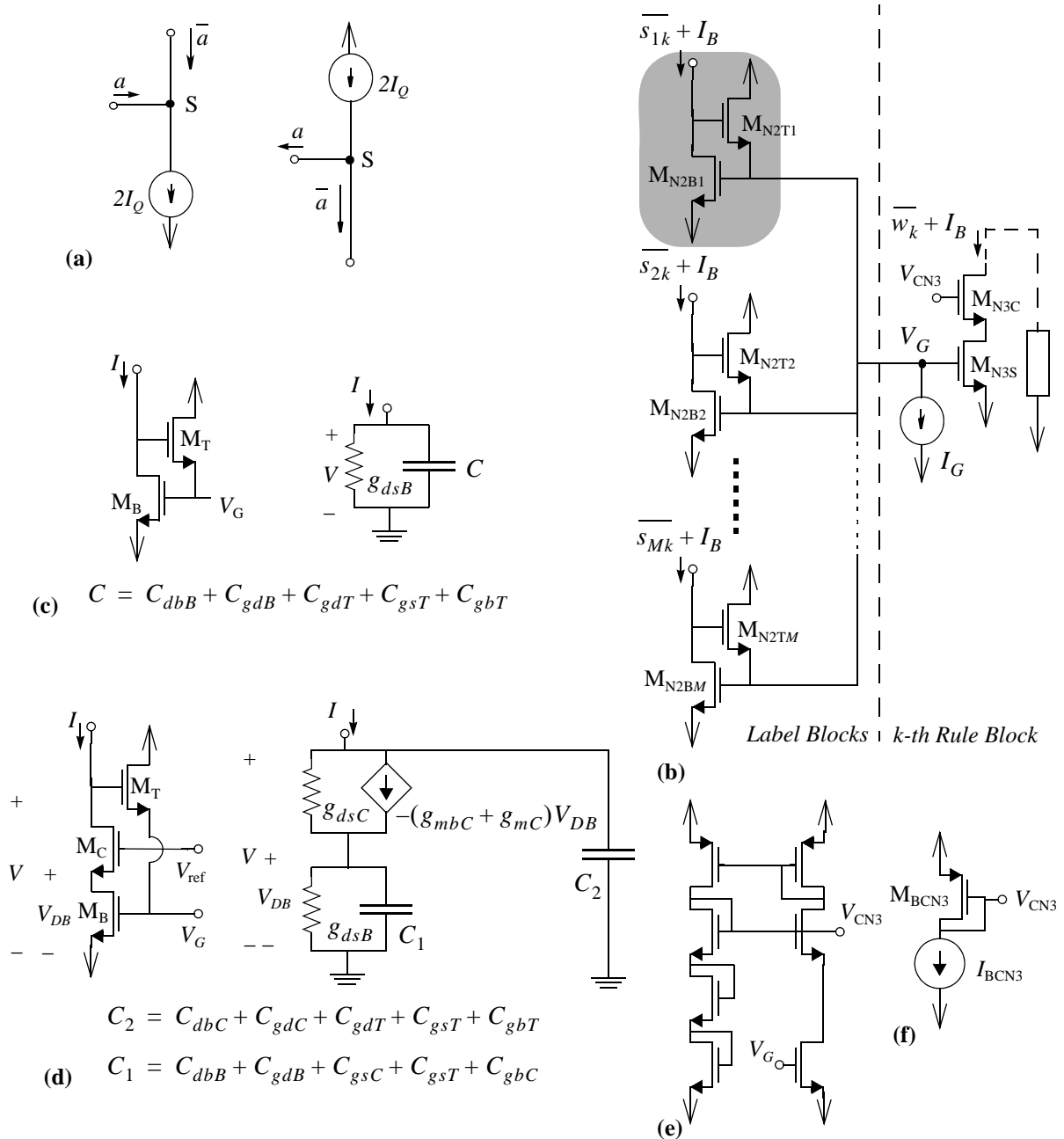


Figure 6: Circuitry for the minimum computation: (a) complement implementation; (b) maximum circuit; (c) and (d) small signal models of input unit cells; (e) adaptive bias circuit; (f) fixed bias circuit.

put transistor works in saturation region, its current coincides with the maximum one. When the maximum current is switched from one input terminal to another, a transient takes place where the difference between the new and the old maximum current is integrated in the latter terminal, thus driving this transistor into a conducting state and, eventually, changing the value of the common voltage V_G .

This circuit exhibits the architectural features mentioned above: a) it has $O(n)$ complexity; b) the different inputs share only the node V_G . This latter feature allows us to partition the circuit as Fig.6(b) shows, so that the rule block has only one input. Another current-mode maximum circuit based also on Lazzaro's was proposed in [31] and used in [14]. It connects the output transistor as a diode, removes the current source I_G , and connects the drains of the top transistors M_{N2Tk} to a common node which is the output node. Thus, the inputs share two nodes instead of one. Besides, the removal of the current I_G renders the resolution of this circuit dependent of the output current level and, specifically, small for large currents [27]. Finally, because the output node load increases with the input count, this circuit performs poorer than Fig.6(b) when the number of inputs increases.

Let us now describe the realization of the two first steps for the minimum in the label block. The first (complementation) is realized by KCL at the input node of the right-top current mirror in Fig.5. Its input current is $\overline{s_{ij}} + I_B$ where the current I_B is added to preclude the transistors entering in subthreshold, where the operation speed would become significantly degraded. Note, on the other hand, that this current mirror has with L^{M-1} output branches to generate the membership function output replicas $s_{ij}|^r$ for $1 \leq r \leq L^{M-1}$ for the different rules. The second step is realized also in the label block and consists of the generation of a set of intermediate voltages $V_{Gij|}^r$ as nonlinear functions of the currents $s_{ij}|^r + I_B$. Each of these voltages is generated in the right-bottom shaded area of Fig.5 by a two transistor circuit (see also Fig.6(b)); for proper operation of this two transistor circuit, some artifact must be added to discharge the node V_G – provided by the current source I_G included in the rule block (see Fig.6(b)).

The next step for the minimum operation is realized in the rule block (bear in mind, Fig.2(a), that this block has one input and one output). To that purpose the set of voltages V_{Gik} for the M membership function values associated to the k -th rule are routed and tied together at the input node of the rule block (see the left-hand part of Fig.6(b)). Thus, a collective computation is performed at this common node such that the maximum among the set of voltages prevails. From this maximum voltage the corresponding maximum current $\overline{s_{ik}(x_i)}|_{max} + I_B$ is generated by the transistor M_{N3S} in Fig.6(b). According to (10) this corresponds to the fuzzy complement $\overline{w_k(\mathbf{x})}$ of the multidimensional membership value shifted by I_B .

III.C. Design Considerations in the Label Block

A thorough analysis of the static (systematic and random) and dynamic errors of Fig.5 and Fig.6(b), as well as that of other alternative circuit implementations, is found in [27]. This sec-

tion summarizes some main results regarding systematic errors due to the finite output impedance of the MOST's ^{†5} which are relevant to design purposes. Random errors are covered for the whole controller in Section V.

III.C.1. Membership Function Circuit

A first consideration refers to the common-mode input range of the differential amplifiers of Fig.5. It is calculated by constraining the transistors to operate in saturation region,

$$V_{C_\Omega} + V_{TN1} + \sqrt{\frac{I_Q}{2\beta_{N1}}} \leq x_i \leq V_{DD} - \sqrt{\frac{2I_Q + I_B}{\beta_{P1}}} \quad (11)$$

where V_{C_Ω} is the compliance voltage of the current I_Q , β_{N1} is the large signal transconductance of the input NMOST's, V_{TN1} their threshold voltage and β_{P1} is the transconductance of the top PMOST's. A strategy to improve the common-mode range is using bias current circuits with the smallest possible value of V_{C_Ω} , such as that attached to Fig.5 where,

$$V_{C_\Omega} = \sqrt{\frac{I_Q}{\beta_{NQ}}} + \sqrt{\frac{I_Q}{\beta_{NQC}}} \quad (12)$$

Biasing of the current mirror that generates I_Q is then carried out by the circuit at the left of Fig.5, where I_{BQ} is a reference current and the geometry of M_{BQ} is obtained from

$$S_{BQ} \equiv \frac{\beta_{BQ}}{\beta_{0N}^*} = \frac{I_{BQ}}{\beta_{0N}^* (\sqrt{I_Q/\beta_{NQC}} + V_{TNQC} + V_{C_\Omega})^2} \quad (13)$$

where β_{0N}^* is the large-signal transconductance density of the NMOST. Typical input range values are around 3.25V by following this approach with a 1 μ m CMOS standard technology and 5V of voltage supply.

Another error source is DC voltage mismatching between the drains of the input transistors (nodes V_{D+} and V_{D-} in Fig.5) which might cause offset and distortion of the membership output current for finite MOST Early voltages. However, because these two nodes are both of low-impedance type, the voltage excursions are largely attenuated by the transconductance of the PMOST's and the error is, hence, negligible.

The last error is due to DC voltage mismatching between the input and output nodes of the PMOS current mirror driving the minimum input cell,

$$\varepsilon \approx 1 + \frac{V_{DP} - V_{D-}}{V_{AP}} \quad (14)$$

where V_{AP} is the equivalent Early voltage of the PMOSTs. This error can be attenuated by

5. They will be modelled through an equivalent Early voltage V_A which is a quasi-linear function of the channel length [26].

proper setting of the bias voltage V_{CP2} of the cascode transistor M_{P2C} . For optimum attenuation, this voltage should be different for different input values. However, system-level considerations [27] show that it suffices to obtain the largest possible attenuation at the crossover points. The corresponding voltage is calculated to annul the following expression of the absolute current error,

$$\begin{aligned} \varepsilon^* \cong & \left(\frac{3}{V_{AP}} + \frac{1}{V_{AP2C}} \right) \frac{I_Q}{2\sqrt{\beta_P}} \left(\sqrt{I_Q + I_B} - \sqrt{\frac{3I_Q}{2}} \right) + \\ & + \frac{1}{V_{AP}} (I_Q + I_B) \left[V_{DD} - 2V_{T0P} - \gamma_{P2C} [\sqrt{V_{DP} + \phi_B} - \sqrt{\phi_B}] - \sqrt{\frac{I_Q + I_B}{\beta_{P2C}}} - \sqrt{\frac{I_Q + I_B}{\beta_P}} - V_{CP2} \right] \end{aligned} \quad (15)$$

where we assume (as it happens in practice) that all PMOS signal transistors in Fig.5 are equal ($\beta_{P1} = \beta_{P2S} \equiv \beta_P$, $V_{AP1} = V_{AP2S} \equiv V_{AP}$); V_{T0P} is the threshold voltage of the PMOST zero bias and γ_P and ϕ_B are technological parameters [26]. This optimum voltage can be generated by the circuit at the right in Fig.5, where we assume that the two diode-connected transistors have the same aspect ratio,

$$S_{BCP2} \equiv \frac{\beta_{BCP2}}{\beta_{0N}^*} = \frac{4I_{BCP2}}{\beta_{0N}^* (V_{CP2} - V_{TBCP2T} - V_{TBCP2B})^2} \quad (16)$$

and I_{BCP2} is a reference current. This choice reduces the relative error in (14) to around 0.5% – negligible at the system level.

III.C.2. Static and Dynamic Errors in the Maximum Circuit Operation

Two major features related to the DC operation are the *discrimination* (the circuit ability to distinguish two close input values), and the error due to DC voltage mismatching between the input node sinking the maximum current and the drain of the output transistor. The discrimination of Fig.6(b) is calculated as [27],

$$\frac{\Delta I}{I} \approx \frac{1}{V_{AN2B}} \sqrt{\frac{I_G}{\beta_{N2T}}} \quad (17)$$

where ΔI is the minimum current increment that can be detected by the circuit, and V_{AN2B} is the equivalent Early voltage of the bottom MOST. This equation shows that the discrimination improves for I_G decreasing, β_{N2T} increasing and V_{AN2B} increasing. The $1\mu\text{m}$ CMOS controller demonstrator in this paper obtains ΔI values as small as 8nA, for input currents around $10\mu\text{A}$, with $I_G = 0.5\mu\text{A}$, and transistors sizes $W = 10\mu\text{m}$ and $L = 5\mu\text{m}$.

On the other hand, the current gain error due to input-output DC voltage mismatching is given by,

$$\varepsilon \approx 1 + \frac{V_D|_{M_{N2Bmax}} - V_D|_{M_{N3S}}}{V_{AN}} \quad (18)$$

where we have assumed equal Early voltages for the input and output transistors. Calculation of this error for Fig.6(b) and a maximum current level I obtains,

$$\varepsilon \approx \frac{\left[V_{T0N} + V_{TN2T} + V_{TN3C} + \sqrt{\frac{I}{\beta_{N3C}}} + \sqrt{\frac{I}{\beta_N}} + \sqrt{\frac{I_G}{\beta_{N2T}}} - V_{CN3} \right]}{V_{AN}} \quad (19)$$

$$1 + \frac{\left[V_{T0N} + V_{TN2T} + \sqrt{\frac{I}{\beta_N}} + \sqrt{\frac{I_G}{\beta_{N2T}}} \right]}{V_{AN}}$$

where we assume $\beta_{N2B} = \beta_{N3S} \equiv \beta_N$. This expression shows that V_{CN3} can be chosen to annul the error for a given current level. Because the compensation value depends on the current, the adaptive biasing stage of Fig.6(e) [27] can be used to obtain V_{CN3} varying with the current level. In the $1\mu\text{m}$ CMOS technology used in the paper's prototype, this adaptive biasing obtains errors as low as 0.3% for input currents up to $20\mu\text{A}$ – a precision larger than needed for most practical fuzzy logic applications. In practice a simpler biasing stage (see Fig.6(f)) providing a constant V_{CN3} value is enough. This voltage V_{CN3} can be obtained by making $\varepsilon = 0$ in (19) for I corresponding to the middle of the range. The size of M_{BCN3} in Fig.6(f) is then determined by the equation,

$$S_{BCN3} \equiv \frac{\beta_{BCN3}}{\beta_{0P}^*} = \frac{4I_{BCN3}}{\beta_{0P}^*(V_{DD} - V_{CN3} - V_{TBCN3})^2} \quad (20)$$

Another strategy to attenuate this error is by adding cascode transistors (see Fig.6(d)) to equalize both drain voltages in (18). However, this slows down the transient following an inter-terminal switching of the maximum input current. This transient has two phases: during the first the voltage V_G remains quasi-constant while the voltage at the new winning input terminal is built-up (henceforth called *switching transient*); during the second phase the voltage V_G is updated to conform to the new current (*propagation transient*). Differences between Fig.6(c) and (d) arises mostly at the switching transient and can be assessed by comparing the time constants of the first-order models attached to the figures,

$$\tau|_{simple} = \frac{C}{g_{dsB}} \quad (21)$$

$$\tau|_{cascode} \approx \frac{C_2}{g_{dsB}} \times \frac{g_{mC} + g_{mbC}}{g_{dsC}}$$

Assuming equal transistor sizes so that $C \approx C_2$, and because $g_m \gg g_{ds}$, we obtain

$\tau|_{cascode} \gg \tau|_{simple}$ – the reason leading us to discard cascode input transistors.

Besides of dynamic aspects involved in the switching process, we have to take into account that the dynamic response of these implementations depend on the number of inputs, since the parasitic capacitance at the common gate increases. Possible solutions for a high number of inputs are using bias currents and trees with complemented PMOS and NMOS circuits [32].

IV. Rule Block

The k -th rule block is intended to, first, calculate the current w_k and, second, generate an output current given by,

$$y_k = y_k^* \frac{w_k}{\sum_{k=1,N} w_k} \quad (22)$$

these currents are then routed to a common node to implement (4) through KCL.

There are three main approaches for the analog implementation of (22) and/or (4): using an extension of the Mead's [33] follower-aggregation circuit with weighting capability [16][37]; using weighting-plus-division circuits [14][22][35][36]; using normalization-plus-weighting circuits [9] [25] [28]. The first uses an elegant circuit concept, see Fig.7(a), to implement a non-linear version of (4) with voltage output. However, because of the feedback, its transient response is not optimum; also, because a large signal current w_k is applied at the TA bias terminal, the linear operation range and the transient response, are largely non-homogeneous over the university of discourse; finally, additional MDACs are required to incorporate digital programmability of the singletons. Fig.7(b) and (c) show the concepts of the other two approaches. Both permit transparent digital programmability of the singletons. However, different reasons lead us to using the normalization-plus-weighting approach. First, the weighting-plus-division approach requires replication of the input currents and wide-range linear current-mode divid-

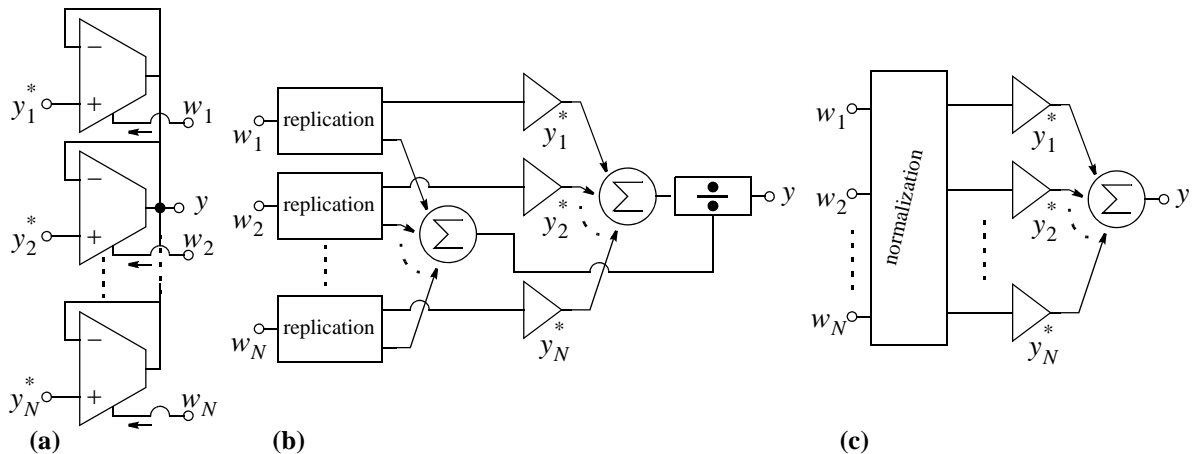


Figure 7: Singleton defuzzification strategies: (a) follower-aggregation; (b) weighting-plus-division; (c) normalization-plus-weighting.

ers, while the normalization can be realized through a collective computation circuit with only two transistors per input; the chosen approach results, hence, in simpler circuits. Second, because the transmission path for the numerator and the denominator of (4) are not the same in Fig.7(b) this approach is more sensitive to mismatching. Third, the transient response of Fig.7(b) is largely-dependent on the signal level. Fourth, there is no simple way to compensate for the errors in the divider - the only way is using very accurate dividers.

Fig.8 shows the schematics of the rule block where four different operations are realized: first, the current $\overline{w_k} + I_B$ is generated as explained in Section III.B; second, this current is complemented and shifted to obtain $w_{ks} = w_k + I_{OS}$; third, a collective computation is carried out by all the rule blocks (they share the global nodes A_{NOR} and B_{NOR}) to realize the normalization operation; fourth, the resulting current is weighted by a digitally-controlled current mirror to obtain the shifted version of the k -th rule output current.

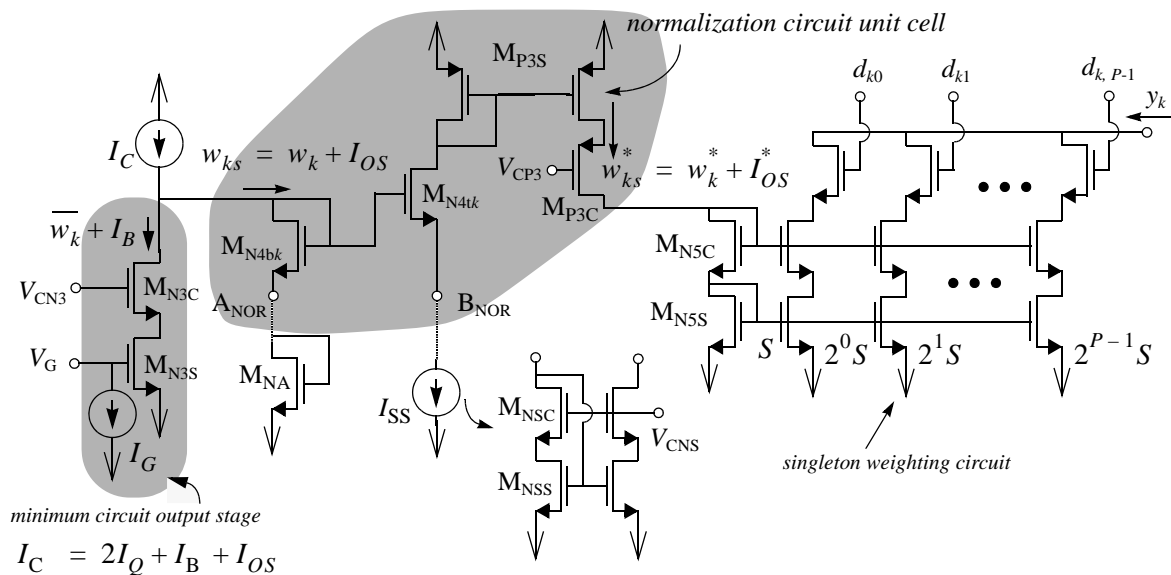


Figure 8: Rule Block

IV.A. Normalization Circuitry

Fig.9 shows the CMOS normalizer circuit used in our chip – based on a translinear BJT cir-

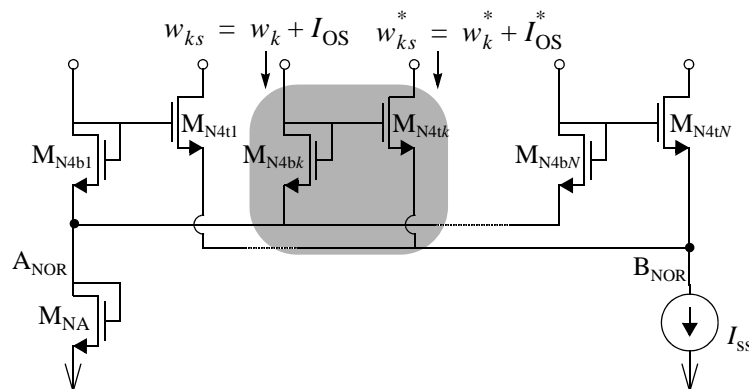


Figure 9: Normalization circuit schematics

cuit by Gilbert [34]. As a difference to the normalizers used in [9] [28], Fig.9 does not involve any feedback loop and, hence, features much faster dynamic response.

Note that Fig.9 can be split into N cells, one per each input-output pair, plus a little common circuitry consisting of the transistor N_{NA} and the current source I_{SS} . Fig.8 exploits this modularity by incorporating one of these cells at each rule block.

Assuming that the transistors operate in strong inversion, where the BJT translinear principle does not hold, the circuit is found to realize the following nonlinear transformation,

$$w_{ks}^* = \frac{\beta_{N4t}}{\beta_{N4b}} w_{ks} \left[1 + \frac{\eta(\mathbf{w}_s)}{\sqrt{w_{ks}}} \right]^2 \quad (23)$$

where the function $\eta(\mathbf{w}_s)$ is,

$$\eta(\mathbf{w}_s) = \frac{\sum_k \sqrt{w_{ks}}}{N} \left(\sqrt{1 + \frac{N \left(I_{SS}' - \sum_k w_{ks} \right)}{\left(\sum_k \sqrt{w_{ks}} \right)^2} - 1} \right) \quad I_{SS}' = \frac{\beta_{N4b}}{\beta_{N4t}} I_{SS} \quad (24)$$

and,

$$w_k^* = w_{ks}^* - I_{OS}^* \quad I_{OS}^* = w_{ks}^*(I_{OS}) \quad (25)$$

The offset current I_{OS} is added to improve the dynamic behavior. Note from Fig.8 that it is related to the bias currents in the rule antecedent by $I_{OS} = I_C - 2I_Q - I_B$. Thus, it can be introduced by just increasing the current I_C without additional area cost, although it will be preserved in figures and equations to gain clarity.

The circuit in Fig.9 exhibits the following features: a) the sum of all output currents is constant and equal to I_{SS} ; b) for each input, the input-output transformation is a soft monotonic one, i.e, the higher an input current, the higher the corresponding output current. Thus, the relative strengths of the different rule antecedents are preserved at the outputs – as required for defuzzification [1]-[4]. Hence, although this circuit does not realize the ideal normalization operation, it keeps the essential features needed for defuzzification; non-linearity is not problematic because the whole controller chip is highly nonlinear. Actually, system-level analysis shows that, despite this non-linearity, the normalization-plus-weighting defuzzification approach features smaller deviations from the linear interpolation than the ideal weighting-plus-division structure [27].

IV.B. Design Considerations in the Normalization Circuit

A first consideration refers to the input range of the normalization circuit when embedded into Fig.8. Consider first the common-mode range, where all input currents are equal. If they increase, transistors M_{N4bk} evolve towards the ohmic region; on the other hand, if they decrease,

the transistors used in the current source I_{SS} evolve towards the ohmic region. Thus, the common-mode input range is given by,

$$\left[\frac{V_{\text{Bnorm}_\Omega} - V_{T0N} + \sqrt{\frac{I_{SS}}{N\beta_{N4t}}}}{\sqrt{\frac{1}{\beta_{N4b}} + \frac{N}{\beta_{NA}}}} \right]^2 - I_{OS} \leq w_k \leq \left[\frac{V_{DD} - V_{T0P} - V_{T0N} - \sqrt{\frac{I_{SS}}{N\beta_{N4b}}}}{\sqrt{\frac{1}{\beta_{N4b}} + \frac{N}{\beta_{NA}}}} \right]^2 - I_{OS} \quad (26)$$

where V_{Bnorm_Ω} is the compliance limit for the current source I_{SS} and we have assumed that the threshold voltages of top (M_{N4tk}) and bottom (M_{N4bk}) transistors are approximately equal, because their sources are at similar voltage. The bottom limit in (26) is valid whenever $V_{\text{Bnorm}_\Omega} \geq V_{T0N} - \sqrt{I_{SS}/(N\beta_{N4t})}$, otherwise the real condition limit is zero. The wide range cascode current mirror enclosed in Fig.8 allows us to obtain a good common mode range (given by $V_{\text{Bnorm}_\Omega} = \sqrt{I_{SS}/\beta_{NSS}} + \sqrt{I_{SS}/\beta_{NSC}}$) as well as good precision.

Consider now the differential range; if one input current changes while the others are kept constants, the top transistor for the changing current will eventually drive all the current I_{SS} , and the other top transistors will be cut-off. The differential range is given by,

$$0 \leq w_k \leq \beta_{N4b} \left(\sqrt{\frac{I_{OS}}{\beta_{N4b}}} + \sqrt{\frac{I_{SS}}{\beta_{N4t}}} \right)^2 - I_{OS} \quad (27)$$

where we have considered that the set of fuzzy rules is *consistent* [39], i.e., when an input is maximum the remaining are zero.

There are three main sources of systematic errors in Fig.8: the finite impedance of I_{SS} , the DC voltage mismatching among output nodes of the circuit core (shaded in Fig.9), and the DC voltage mismatching between input and output nodes of the output PMOS mirrors. The adopted cascode realization of I_{SS} makes the first negligible. On the other hand, because the top transistors are connected to low-impedance nodes, the second error is largely attenuated by the transconductances of the PMOSTs used at these nodes. Concerning the third error source, it can be minimized by inserting cascode transistors, as Fig.8 shows. The error is then given by,

$$\varepsilon(w_{ks}^*) = \frac{1}{V_{AP3S}} w_{ks}^* \left(V_{DD} - 2V_{T0P} - \sqrt{\frac{w_{ks}^*}{\beta_{P3S}}} - \sqrt{\frac{w_{ks}^*}{\beta_{P3C}}} - V_{CP3} \right) \quad (28)$$

which is minimized by proper choice of V_{CP3} . Again, a particular signal value has to be selected to guide the choice of V_{CP3} . Because most output branches drive a current value I_{OS}^* , such current level defines a good choice. Thus, V_{CP3} is obtained from (28) for $\varepsilon = 0$ and $w_{ks}^* = I_{OS}^*$, and it is generated in similar as already explained for Fig.6(f).

With regard to the dynamic response, analysis recommends to scale the width of M_{NA} as well as the value of I_{SS} proportionally to the number of normalizer inputs, i.e. rules in the controller,

in order to preserve the dynamic response as the complexity increases.

IV.C. Singleton Weighting and Output Layer

Fig.8 employs a digitally-controlled current-mirror (represented at the conceptual level in Fig.10(a)) to implement a programmable singleton value y_k^* . As compared to analog-programmed current mirrors [38][40], the digital approach is preferred because it is more robust and accurate, compatible with standard memory circuits and directly controllable through conventional computers.

Regarding the mirror circuitry itself, and because the normalization circuit output stage does not impose major range limitations, a stacked (self-biased) cascode structure is used to minimize errors due to DC mismatching. On the other hand, parallel-connected unit transistors are used to realize the binary weighting and, thus, reduce systematic errors caused by the lack of symmetry. The bias current depicted with dashed lines in Fig.10(a) is added to reduce speed degradations due to the increase of the parasitic capacitance for large singleton values.

After singleton weighting the rule block outputs are wired up to the output node where a current $-\sum_{1,N} y_k^* I_{OS}^*$ is added to remove the offset and, thus, obtain (4).

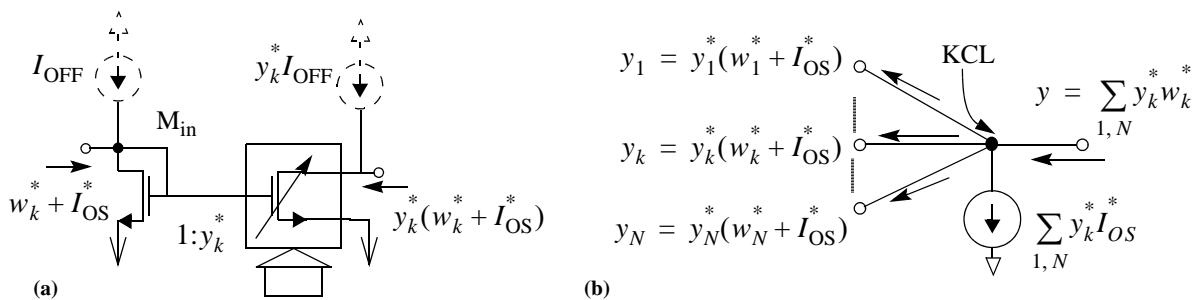


Figure 10: (a) Singleton weighting concept; (b) controller output node.

V. Global Considerations

V.A. Dependence on Temperature

All the building blocks except the membership function circuit have temperature independent transfer functions. The temperature dependence of the latter is caused by the large signal transconductance β in (6). However, because of the differential pair symmetry, the location and width defined in (7) are not affected by temperature changes. Electrical values of the logical zero and one are neither affected, provided the current reference is temperature-independent, because these values are associated to, respectively, logical states of the transistors in the differential pairs. The only parameter which is affected by temperature changes is the membership function slope, see (8). From a global point of view, this means that the slope of the generated function between interpolation points changes with temperature, as Fig.11 illustrates for a controller with four rules. Thus, the interpolation smoothness changes with temperature, but the

interpolation points are not affected if membership functions are wide enough to saturate in the whole temperature range.

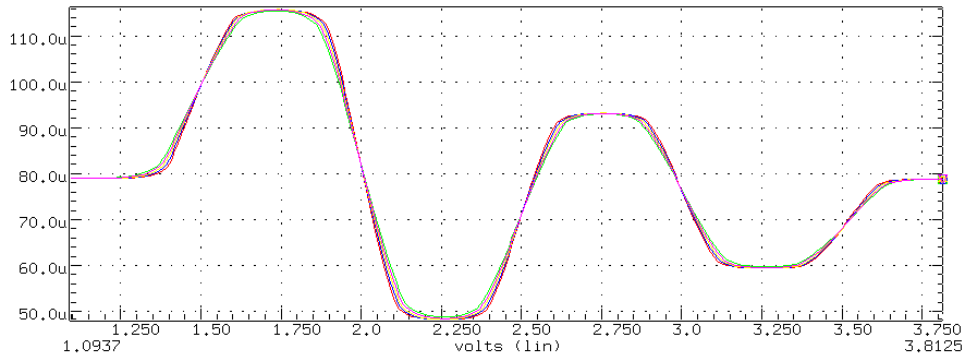


Figure 11: Illustrating dependence on temperature.

V.B. Power Estimation

Let us consider a controller with M inputs, N rules and $L = N^{1/M}$ fuzzy labels whose maximum singleton value in the associated rule base is y_{kmax}^* . The maximum static power consumption is calculated as,

$$P_w = \{M \times [N \times (I_U + I_B) + L \times (2I_U + I_B)] + N \times (I_C + I_G) + 2I_{SS}\} V_{DD} + V_{load} I_{SS} y_{kmax}^* \quad (29)$$

where $I_U = 2I_Q$ and the currents I_B , I_G and I_C are defined in Figs. 5, 6 and 8.

V.C. Mismatching Errors

Random variations of the transistor parameters V_{T0} , β and γ can be modelled as normal distributions whose mean values are the nominal parameter values. For close and small enough transistors the variances depend mostly on the device area [12],

$$\sigma^2(V_{T0}) = \frac{A_{V_{T0}}^2/2}{WL} \quad \sigma^2(\gamma) = \frac{A_\gamma^2/2}{WL} \quad \frac{\sigma^2(\beta)}{\beta^2} = \frac{A_\beta^2/2}{WL} \quad (30)$$

where W and L are the transistor channel width and length, and $A_{V_{T0}}^2$, A_γ^2 and A_β^2 are technology-dependent.

Based on , we can obtain expressions for the errors in the fuzzy controller blocks. The detailed explanation of these errors is beyond the scope of this paper; thus, only those resulting in important design equations will be outlined.

Consider the membership function circuit first. Analysis shows that the most significant error corresponds to the case where the rule output is maximum [27]. The variance of the complement of the membership function current (its mean value is I_B) is given by,

$$\begin{aligned} \sigma^2(\bar{s}_{ij}) = & 4I_Q^2 \frac{\sigma^2(\beta_{NQ})}{\beta_{NQ}^2} + 16\beta_{NQ}I_Q\sigma^2(V_{T0NQ}) + 8I_Q^2 \left(\frac{\sigma^2(\beta_P)}{\beta_P^2} + \frac{4\beta_P\sigma^2 V_{TOP}}{2I_Q} \right) + \\ & + I_B^2 \frac{\sigma^2(\beta_{NB})}{\beta_{NB}^2} + 4\beta_{NB}I_B\sigma^2(V_{T0NB}) + 2I_B^2 \left(\frac{\sigma^2(\beta_P)}{\beta_P^2} + \frac{4\beta_P\sigma^2 V_{TOP}}{I_B} \right) \end{aligned} \quad (31)$$

where we assume that the PMOSTs MP1 and MP2S are equal, and cascode transistors mismatching is not computed because their influence is negligible as compared to signal transistors. This expression includes the errors due to the NMOS transistors (parameters β_{NB} and V_{T0NB}) of the current mirror used to provide I_B in Fig.5^{†6}.

The error at the rule output is calculated by adding the error caused by the minimum circuit to the previous one. The variance for the worst case (only one antecedent active in the rule and non sharing of the membership function circuits) is,

$$\sigma^2(w_{ks}) = 2I_B^2 \left(\frac{\sigma^2(\beta_{N3S})}{\beta_{N3S}^2} + \frac{4\beta_{N3S}\sigma^2 V_{T0N3S}}{I_B} \right) + I_C^2 \frac{\sigma^2(\beta_{PC})}{\beta_{PC}^2} + 4\beta_{PC}I_C\sigma^2(V_{T0PC}) + \sigma^2(\bar{s}_{ij}) \quad (32)$$

and the corresponding mean value is $I_C - I_B$. The bracketed terms correspond to the maximum circuit; the others to the complement and membership function circuits. The mismatch is smaller for any other case, although the expression of the variance is difficult to obtain because of correlations between variables. Parameters β_{PC} and V_{T0PC} in (32) correspond to the large signal transconductance value and the zero bias threshold voltage respectively, of the non-cascode output PMOS transistor in a current mirror that provides I_C in Fig.5.

The errors due to the normalization circuit are characterized by the following approximate variance expression,

$$\begin{aligned} \sigma^2(w_{ks}^*) \approx & 4\beta_{N4t}w_{ks}^*(1-\Gamma)^2 [\sigma^2(V_{T0N4b}) + \sigma^2(\gamma_{N4b})(\sqrt{V_{ANOR} + \phi_B} - \sqrt{\phi_B})^2] + \\ & + 4\beta_{N4t}w_{ks}^*(1-\Gamma)^2 [\sigma^2(V_{T0N4t}) + \sigma^2(\gamma_{N4t})(\sqrt{V_{BNOR} + \phi_B} - \sqrt{\phi_B})^2] + \\ & + (w_{ks}^*)^2 (1-\Gamma)^2 \frac{\sigma^2(\beta_{N4t})}{\beta_{N4t}^2} + \frac{\beta_{N4t}}{\beta_{N4b}} w_{ks}^* w_{ks} (1-\Gamma)^2 \frac{\sigma^2(\beta_{N4b})}{\beta_{N4b}^2} + \\ & + 2(w_{ks}^*)^2 \left(\frac{\sigma^2(\beta_{P3S})}{\beta_{P3S}^2} + \frac{4\beta_{P3S}\sigma^2 V_{TOP3S}}{w_{ks}^*} \right) + \left(\frac{N-1}{N} \frac{\beta_{N4t}}{\beta_{N4b}} \sqrt{\frac{I_{SS}}{Nw_{ks}}} \right)^2 \sigma^2(w_{ks}) \end{aligned} \quad (33)$$

where the mean value of w_{ks}^* is given by (23),

6. This mirror was omitted there in behalf of clarity and because its design is not critical for other performance parameters.

$$\Gamma = \frac{\sqrt{w_{ks}^*}}{\sum_{k=1, N} \sqrt{w_{ks}^*}} \quad (34)$$

and $\sigma^2(w_{ks})$ is given by (32) for a maximum rule antecedent output current.

The approximation used to calculate (33) consists of neglecting the mismatching in those normalizer inputs others than the k -th. These terms contribute only around 3% of the variance for the 16 rules CMOS prototype in this paper, and their contribution decreases as the rule count increases. This highlights an interesting feature of Fig.9 which is not shared by other approaches to the normalization operation; namely, the mismatching errors of the different rules are nearly independent. Thus, they are not mixed in the output node and manifest as off-sets (easy to correct) at the points were the rule outputs are maximum (the most significant to design purposes).

The global error at the rule block output includes also the influence of the weighting circuit,

$$\sigma^2(y_k) = (y_k^*)^2 \sigma^2(w_{ks}^*) + \left[y_k^2 \frac{\sigma^2(\beta_{in})}{\beta_{in}^2} + 4\beta_{in} y_k \sigma^2(V_{T0in}) \right] \left(1 + \frac{1}{y_k^*} \right) \quad (35)$$

where $\sigma^2(w_{ks}^*)$ is given by (33), and β_{in} and V_{T0in} refers to the non-cascode input transistor in the weighting circuit (see Fig.10(a)). The first term at the right in (35) corresponds to the error transmitted by the weighting circuit from previous stages, while the second term corresponds to the error introduced by itself. Note that the latter decreases when the singleton value grows.

While residual systematic errors may be filtered out by the normalizer [27], the only way to attenuate the random errors is solving the design equations (30)-(35) to obtain proper transistor sizes, which is more conveniently performed with the help of an iterative optimizer.

VI. Experimental Results

Fig.12(a) shows the microphotograph of a chip that performs the processing tasks involved in (4) and Fig.2(a). It is a lattice controller with two inputs and four labels per input (see Fig.2(b)). Thus, eight label blocks, four per chip input, are needed, as well as sixteen rule blocks. The label blocks outputs are connected to inputs of rule blocks through a “ring bus”. Bias circuitry, as well as one diode connected transistor and one current mirror, which complete the normalization circuit in Fig.9, are implemented in the “biasing box”. Table 1 shows the most relevant transistor sizes in this chip.

Table 1: Transistor sizes (W/L) in $\mu\text{m}/\mu\text{m}$ in the prototype chip

M_{N1ij-}	M_{N1ij+}	M_{P1}	M_{P2S}	M_{P2C}	M_{N2B}	M_{N2T}	M_{NQ}	M_{NQC}	M_{N3S}	M_{N3C}	M_{N4bk}	M_{N4tk}	M_{P3S}	M_{P3C}	M_{N5S}	M_{N5C}
40/10	40/10	50/5	50/5	50/2	20/10	20/10	10/10	10/1	20/10	20/1	30/5	50/5	50/5	50/1	20/10	20/5

Digital values to program the output current mirror and hence the singleton values are stored in a “shift register” which is the chip internal memory element and is serially programmed through two pads. Apart from digital programmability of the singleton values, width and location of membership functions are also analogically programmable by setting the voltages E_{ij+} and E_{ij-} (see (7) and Fig.5).

Figs. 13(a) and (b) show two output surfaces generated by the chip. The bias signals are $V_{DD} = 5V$, $V_{SS} = 0V$, $I_Q = 7,5\mu A$, $I_B = 10\mu A$, $I_G = 0,5\mu A$, $I_C = 35\mu A$ and $I_{SS} = 37\mu A$, while the voltages E_{ij} are fixed to obtain a uniform lattice partition of the input space. The circuit was loaded with a constant voltage source of 2.5V and a current source to remove the offset introduced in the normalization circuit. Singletons are set to decimal values 1 and 15 in Fig.13(a), which highlights the locality of the fuzzy basis functions, while Fig.13(b) illustrates an exemplary surface obtained with different singleton values. Finally, Fig.13(c) depicts a set of sections from Fig.13(b) which show the output when it reaches their local maximum values, thus the singleton values.

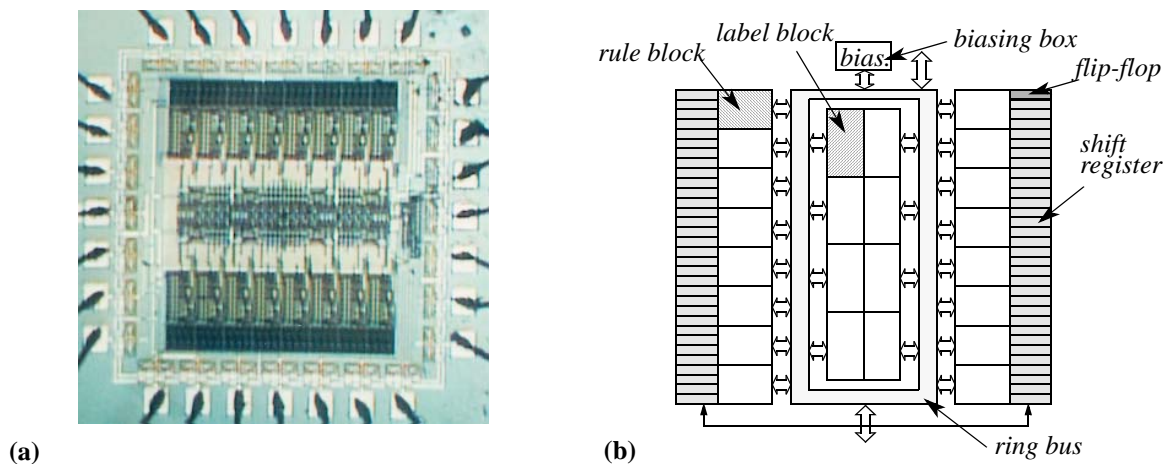


Figure 12: (a) Chip microphotograph; (b) internal architecture.

Maximum circuit delay is 471ns (90% of the full scale output current) for a step input, while power consumption is 8.6mW and resolution is around 6.5%. The latter was obtained through Monte Carlo simulations (30 iterations) which take into account parameter mismatching among transistors, with 3σ ($\pm 1.5\sigma$) as error figure. Finally, input voltage range is over 3.25V and the area of the chip without pads is 1.6mm². It is possible to achieve faster designs by introducing bias currents at input and output branches of the current mirror that replicates membership function output, and in the output mirror that implements singleton weighting. It is also possible to achieve a higher precision by inserting the chip in a learning loop with a computer and using the hardware-compatible learning algorithms presented in [18].

Table 2 compares the performance of this prototype to that of previous analog monolithic controllers. It features much smaller *Delay* × *Power* values than these previous circuits, and similar programmability levels than the controller in [16]. However, in this latter controller the consequents values have to be learned using software models, the programming signals are

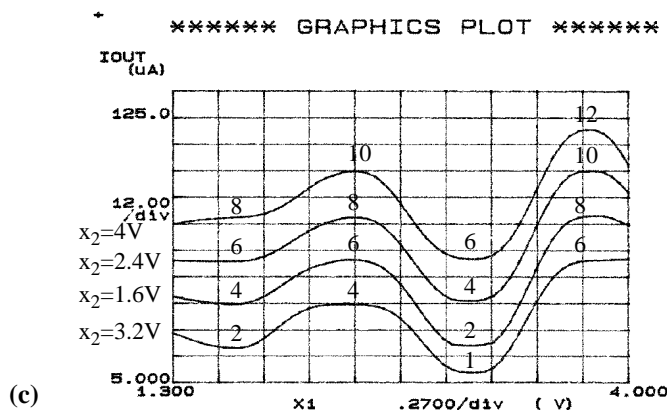
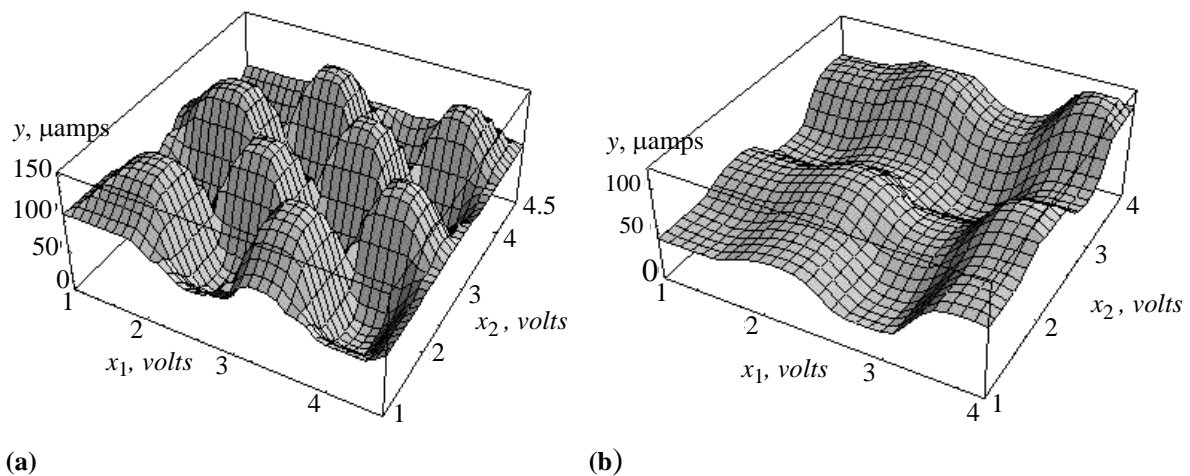


Figure 13: (a) and (b) Controller output for two different sets of singleton values: (c) and sections from (b) at maximum local points.

analog (more difficult interface) and are not stored on-chip.

Table 2: CMOS Analog Implementations of Fuzzy Controllers

CMOS	Manaresi [16]	Guo [17]	Proposed
Complexity	9rules@2input @2output	13rules@3input @1output	16rules@2input@1output
Technology	0.7 μ m CMOS	2.4 μ m CMOS	1 μ m CMOS
Power Consumption	44mW@5V	550mW@10V	8.6mW@5V
Input to Output Delay	570 ns	160 ns	471 ns
Precision	No data	No data	6.5% (3σ)
Interface (inputs@outputs)	voltages@ voltages	voltages@ voltages	voltages@ currents
Programmability	high	low	high
Area	1.9 mm ²	16.2 mm ²	1.6 mm ²

VII. References

- [1] J.M. Mendel, "Fuzzy Logic Systems for Engineering: A Tutorial". *Proceedings of the IEEE*, Vol. 83, pp. 345-377, March 1995.
- [2] H. Takagi, "Applications of Neural Networks and Fuzzy Logic to Consumer Products". pp. 8-12 in *Fuzzy Logic Technologies and Applications*, New-York: IEEE Press 1994.
- [3] M. Brown and C. Harris, *Neuro-Fuzzy Adaptive Modeling and Control*. Englewood Cliffs: Prentice Hall 1994.
- [4] J.S.R. Jang and C.T. Sun, "Neuro-Fuzzy Modeling and Control". *Proceedings of the IEEE*, Vol. 83, pp. 378-406, March 1995.
- [5] 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, 1990.
- [6] K. Nakamura, N. Sakashita, Y. Nitta, K. Shimomura and T. Tokuda, "Fuzzy Inference and Fuzzy Inference Processor". *IEEE Micro*, Vol. 13, pp. 37-48, October. 1993.
- [7] H. Eichfeld, M. Klimke, M. Menke, J. Nolles and T. Künemund, "A General-Purpose Fuzzy Inference Processor". *IEEE Micro*, Vol. 15, pp. 12-17, June 1995.
- [8] A. Costa, A. de Gloria, P. Faraboschi, A. Pagni and G. Rizzotto, "Hardware Solutions for Fuzzy Control". *Proceedings of the IEEE*, Vol. 83, pp. 422-434, March 1995.
- [9] 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, pp. 496-522, May 1993.
- [10] E.A. Vittoz, "The Future of Analog in the VLSI Environment". *Proceedings of the 1990 IEEE Int. Symp. on Circuits and Systems*, pp. 1372-1375, 1990.
- [11] K.A. Nishimura, *Optimum Partitioning of Analog and Digital Circuitry in Mixed-Signal Circuits for Signal Processing*. Ph Dissertation, U.C. Berkeley, 1993.
- [12] M.J.M. Pelgrom, A.C.J. Duinmaijer, and A.P.G. Welbers, "Matching Properties of MOS Transistors". *IEEE Journal of Solid-State Circuits*, Vol. 24, pp. 1433-1440, October 1989.
- [13] J.W. Fattaruso, S.S. Mahant-Shetti, and J.B. Barton, "A Fuzzy Logic Inference Processor". *IEEE Journal of Solid-State Circuits*, Vol. 29, pp. 397-402, April 1994.
- [14] J. L. Huertas, S. Sánchez-Solano, I. Baturone, and A. Barriga, "Integrated Circuit Implementation of Fuzzy Controllers". *IEEE Journal of Solid-State Circuits*, Vol. 31, pp. 1051-1058, July 1996.
- [15] L. Lemaitre, M. J. Patyra, and D. Mlynek, "Analysis and Design of CMOS Fuzzy Logic Controller in Current Mode". *IEEE Journal of Solid-State Circuits*, Vol. 29, pp. 317-322, March 1994.
- [16] N. Manaresi, R. Rovatti, E. Franchi, R. Guerrieri, and G. Baccarani, "A Silicon Compiler of Analog Fuzzy

- Controllers: From Behavioral Specifications to Layout". *IEEE Trans. on Fuzzy Systems*, Vol. 4, pp. 418-428, November 1996.
- [17] S. Guo, L. Peters, and H. Surmann, "Design and Application of an Analog Fuzzy Logic Controller". *IEEE Trans. on Fuzzy Systems*, Vol. 4, pp. 429-438, November 1996.
- [18] F. Vidal-Verdú and A. Rodríguez-Vázquez, "Learning under Hardware Restrictions in CMOS Fuzzy Controllers able to Extract Rules from Examples". *Proc of IFSA '95*, pp. 189-192, Sao Paulo, Brazil, July 1995.
- [19] A. Rodríguez-Vázquez, M. Delgado-Restituto and F. Vidal, "Synthesis and Design of Nonlinear Circuits", Chapter 32 in the *The Circuits and Filters Handbook* (edited by Wai-Kai Chen), pp. 935-972, CRC Press 1996.
- [20] A. Rodríguez-Vázquez and M. Delgado-Restituto, "CMOS Design of Chaotic Oscillators using State Variables: A Monolithic Chua's Circuit". *IEEE Transactions on Circuits and Systems-II*, Vol. 40, pp. 596-613, October 1993.
- [21] A. Rodríguez-Vázquez and M. Delgado-Restituto, "Generation of Chaotic Signals using Current-Mode Techniques". *Journal of Intelligent and Fuzzy Systems*, Vol. 2, pp. 15-37, 1994.
- [22] T. Kettner, C. Heite, and K. Schumacher, "Analog CMOS Realization of Fuzzy Logic Membership Functions". *IEEE Journal of Solid-State Circuits*, Vol. 28, pp. 857-861, July 1993.
- [23] M. Sasaki, N. Ishikawa, F. Ueno and T. Inoue, "Current-Mode Analog Fuzzy Hardware with Voltage Input Interface and Normalization Locked Loop". *IEICE Trans. Fundamentals*, Vol. E75-A, pp. 650-654, June 1992.
- [24] J.W. Fattaruso, and R.G. Meyer, "MOS Analog Function Synthesis". *IEEE Journal of Solid-State Circuits*, Vol. 22, pp. 1059-1063. Dec. 1987.
- [25] A. Rodríguez-Vázquez and F. Vidal, "Analog CMOS Design of Singletonm Fuzzy Controllers". *The Third International Conference on Industrial Fuzzy Control Intelligent Systems*, December 1993.
- [26] Y. Tsividis, *Mixed Analog-Digital VLSI Devices and Technology*. New-York: McGraw-Hill 1996.
- [27] F. Vidal, *Design of Mixed-Signal CMOS Neuro-Fuzzy Controllers*. PhD Dissertation, University of Málaga, 1996.
- [28] M. Sasaki, T. Inoue, Y. Shirai and F. Ueno, "Fuzzy Multiple-Input Maximum and Minimum Circuits in Current Mode and Their Analyses Using Bounded Difference Equations". *IEEE Transactions on Computers*, Vol. 39, pp. 768-774, June 1990.
- [29] T. Yamakawa and T. Miki, "The Current Mode Fuzzy Logic Integrated Circuits Fabricated by the Standard CMOS Process". *IEEE Transactions on Computers*. Vol. C-35, pp. 161-167, February 1986.
- [30] J. Lazzaro, R. Ryckebusch, M. A. Mahowald, and C. A. Mead, "Winner-take-all networks of O(n) complexity". *Advances in Neural Information Processing Systems*, (D. S. Touretzky, Ed.), Vol. 1, Los Altos, CA: Morgan Kaufmann, 1989.
- [31] C. Y. Huang and B.D. Liu, "Current-Mode Multiple-Input Maximum Circuit for Fuzzy Logic Controllers". *Electronics Letters*, Vol. 30, pp. 1924-1925, 1994.
- [32] K.D. Peterson and R.L. Geiger, "Area/Bandwidth Tradeoffs for CMOS Current Mirrors". *IEEE Transactions on Circuits and Systems*, Vol. CAS-33, No. 7, pp. 667-669, July 1986.
- [33] C. Mead, *Analog VLSI and Neural Systems*. Addison Wesley 1989.
- [34] B. Gilbert, "Current-Mode Circuits from a Translinear View Point: A Tutorial". in *Analogue IC Design: The Current-Mode Approach*, C. Toumazou, F. J. Lidgley, and D. G. Haigh, (Eds.), London: Peter Peregrinus Ltd., 1990.
- [35] T. Miki, H. Matsumoto, K. Ohto and T. Yamakawa, "Silicon Implementation for a Novel High-Speed Fuzzy Inference Engine: Mega-Flips Analog Fuzzy Processor". *Journal of Intelligent and Fuzzy Systems*, Vol. 1, No. 1, pp. 27-42, 1993.
- [36] V. Catania, A. Puliafito, and L. Vita, "A VLSI Fuzzy Inference Processor Based on a Discrete Analog Approach". *IEEE Transactions on Fuzzy Systems*, Vol. 2, No. 2, pp. 93-106, May 1994.
- [37] K. Tsukano and T. Inoue, "Synthesis of Operational Transconductance Amplifier-Based Analog Fuzzy Functional Blocks and Its Application". *IEEE Transactions on Fuzzy Systems*, Vol. 3, pp. 61-68, Feb. 1995.
- [38] M. Sasaki and F. Ueno, "A VLSI Implementation of Fuzzy Logic Controller using Current Mode CMOS Circuits". *The Third International Conference on Industrial Fuzzy Control Intelligent Systems*, pp. 215-220, December 1993.
- [39] Li-Xin Wang, *A Course in Fuzzy Systems and Control*. Prebtice-Hall 1997.
- [40] A. Rodríguez-Vázquez, S. Espejo, R. Domínguez-Castro and J.L. Huertas, "Current Mode Techniques for the Implementation of Continuous and Discrete-Time Cellular Neural Networks". *IEEE Transactions on Circuits and Systems-II*, Vol. 40, pp. 132-146, IEEE March 1993.