Chapter 10

DESIGN AND APPLICATION OF ANALOG FUZZY LOGIC CONTROLLERS

Carlos Dualibe[1] and Michel Verleysen[2]
[1]*Universidad Católica de Córdoba,Camino a Alta Gracia KM 10, 5016-Córdoba, Argentina.*
[2]*Université catholique de Louvain, Place du Levant 3,1348-Louvain-la-Neuve, Belgium.*

Abstract: Fuzzy logic has been successfully applied to solve problems in many fields: the classical control, power control, signal and image processing, for instance. As these kinds of application need real-time processing mode, faster, more autonomous and less power-consuming circuits, the choice of on-chip controllers becomes an interesting option. The attractiveness of analog circuits for implementing fuzzy hardware relies on its natural compatibility with most used fuzzy algorithms and the needlessness of A/D and D/A converters for interfacing sensors and actuators. This chapter deals with the implementation of programmable analog fuzzy logic controllers in CMOS technologies.

Key words: Fuzzy logic, analog CMOS VLSI circuits, mixed-signal, signal processing.

1. INTRODUCTION

A fuzzy inference arrangement intends to reproduce algorithmically the structured human knowledge by encoding it into a set of rules, which are normally expressed in the form: *"if* <antecedent> *then* <consequent>*"*. In a fuzzy controller, a relationship between its inputs and outputs is established after evaluating concurrently all rules by following three basic fuzzy operations: fuzzification, rules-evaluation (inference) and defuzzification. This is explained in Figure 10-1 that shows a 2-input 1-output Takagi-Sugeno (TS) fuzzy controller. All input variables are fuzzified by means of the so-called Fuzzy Membership Functions (FMF), that assign to the actual non-fuzzy inputs a degree of membership to the natural concept the FMFs represent (i.e. High, Close to Zero, Positive Big, etc). Fuzzy *if-then* rules are generated by associating the different FMFs defined on the inputs by means of cognitive operators (i.e. AND, OR, etc) and assigning a

rule consequent. For instance, in Figure 10-1, the first rule looks like: *"if* X1 *is* A1 *and* X2 *is* A2 *then* Y *is* Z1", where the meaning of A1 and A2 are represented by FMF-1 and FMF-2, respectively. In first-order TS models, Z1 is a linear combination of the input variables. When only the constant coefficient appears in the consequents, the controller is called zero-order or singleton controller.

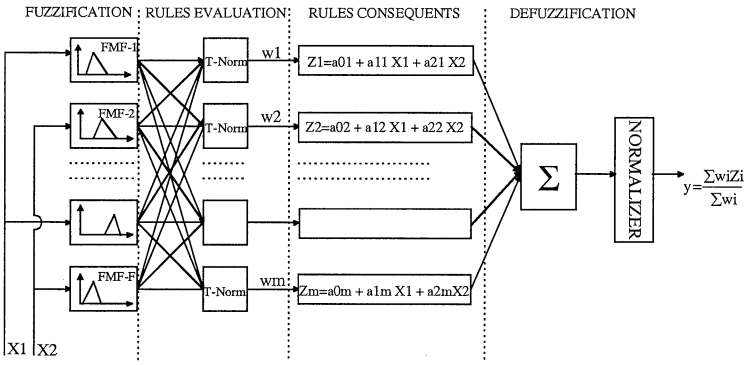

Figure 10-1. m-rule, 2-input, 1-output, Takagi-Sugeno fuzzy logic controller.

The activation degree of a rule is evaluated through the Triangular Norms (T-Norm: minimum) giving rise to the firing coefficient w1...wm that multiply (modulate) the rules consequents Z1...Zm. Rules evaluation can also be performed by means of complemented T-CoNorms (maximum) if Complementary Fuzzy Membership Functions (CFMF) are used [1]. Finally, at the defuzzifier, the actual non-fuzzy output Y results from the normalized Averaged Weighted Sum (AWS) of the modulated rules consequents.

Similar to an artificial neural network, a fuzzy controller establishes a non-linear mapping from inputs to outputs. Both networks belong to the class of the so-called model-free universal approximators that can be tuned by learning. The main difference resides in the fact that in a fuzzy controller knowledge is structured rather than scattered [2].

1.1 Perspectives and Trends of Fuzzy Logic

Applications of fuzzy logic to process control where the mathematical model of the plant is unknown, complex, ill-defined if not time varying, were first developed [3]. At present, the application of fuzzy logic exceeds the control domain. We can find fuzzy controllers in washing machines, automatic focusing for video cameras, automotive anti-skid brake [4] and many others consumer appliances. Moreover, an increasing number of applications in the domain of image processing [5], signal processing [6][7]

and power electronics [8] have been reported in the last years. Most of them need real-time processing, fast transient behavior, low-power consumption and/or autonomy. In such cases, the implementation of a fuzzy controller on an ASIC (Application Specific Integrated Circuit) is justified, even more if cost effective volume productions are desirable.

Some features of fuzzy logic that encourage its use are:

- Fuzzy logic provides a systematic framework to incorporate imprecise information from a human expert. In this way, the control strategy of an operator can be easily integrated in an automatic control system.
- A fuzzy inference system allows modeling non-linear functions of arbitrary complexity. One can create a fuzzy system to fit any training data set by means of adaptive neuro-fuzzy techniques.
- Fuzzy logic is easy to understand and can be combined with classical control techniques.

1.2 Approaches to Fuzzy Hardware

The computational complexity of a fuzzy rule base can be quantified by means of several parameters such as: the number of inputs, the number of outputs, the number and shape of the membership functions per input/output, the number of rules, the rule inference method, the defuzzification algorithm, and the precision needed. The system response time (i.e. input to output delay) is also a relevant parameter that should be kept in mind when evaluating the system performances. Apart from the always possible software implementation of fuzzy inference rules in a workstation, which offers the highest flexibility but the largest response time, the relative simplicity of the fuzzy algorithms makes attractive the use of hardware resources for implementing fuzzy controllers. In [9], four classes among the different hardware implementation alternatives are identified:

1. *General-purpose digital processors*: they offer complete flexibility and short prototyping time. Professional development tools are available. However, speed is limited, even decreasing with complexity. This is the widely used approach for systems with slow dynamics (up to 1KHz).
2. *General-purpose digital processors with dedicated fuzzy instructions:* they employ general-purpose digital cores with the addition of a few specialized instructions to accelerate the fuzzy operations. Higher speed than the one in the previous option is attainable (1 to 50KHz). The prototyping time remains short but the cost is increased due to the small volume production of these units. High-complexity and medium-speed systems can be implemented.
3. *Special-purpose digital coprocessors:* these are special processors dedicated to fuzzy computations. Since they cannot implement the entire

control system by themselves a general-purpose processor is needed to support non-fuzzy operations. They still provide some flexibility for moderate complexity systems with speed ranging from 10 to 100KHz.

4. *Dedicated ASICs:* direct implementation on silicon of fuzzy algorithms by means of high-level or full-custom synthesis techniques (analog or digital) is the finest solution when low-power and high-speed systems (above 1MHz) are intended. However, the prototyping time is the largest whereas the flexibility is reduced. This solution is optimal for well-targeted applications and cost-effective volume production.

It is evident that speed, complexity, flexibility, real-time constraints, prototyping and production times will strongly influence the choice of the implementation option. Certainly, an optimal solution for the whole application range does not exist [9], but the different approaches should be considered according to the requirements of a given design.

1.3 Why Analog Circuits?

The signals, which a fuzzy controller deals with, are essentially analog. Fuzzifiers, inference operators and defuzzifiers handle continuous variables. Therefore, designing fuzzy hardware can be naturally oriented towards the analog domain. Digital fuzzy chips provide enough potential for general applications but their speed is limited. Moreover, the use of analog-to-digital and digital-to-analog converters becomes mandatory for interfacing sensors and actuators. This further deteriorates the speed and increases the hardware complexity. On the other hand, it has been mentioned in the previous sections the need of embeddable fast controllers for applications in many other fields than the classical control. For these cases the use of analog circuits significantly improves performances such as power consumption and silicon die area. Nevertheless, pure analog processors suffer from the lack of suppleness owing to the wearisome implementation of analog storage devices. However, in the frame of standard CMOS technologies, a trade-off between accuracy and flexibility is achieved when a finite set of analog parameters is provided. For instance, a voltage parameter can be settled by using a binary-scaled set of currents sources yielding a discrete set of voltage drops across a linear resistor. Thus, one can use a digital memory to store a binary combination of the set of currents. This technique gives rise to the so-called mixed-signal circuits [8][10].

According to the above discussion, we present in the following the design and test of a digitally programmable analog fuzzy logic controller in a standard CMOS technology for applications of medium to high-speed.

2. A PROGRAMMABLE AND RECONFIGURABLE ANALOG FUZZY LOGIC CONTROLLER

A general architecture for Takagi-Sugeno controllers, that holds high flexibility, is presented hereafter. It features the following characteristics:

- Programmable number of inputs and FMFs per input;
- Programmable number of outputs;
- Programmable number of rules;
- Programmable antecedent and consequent parameters;
- Support of zero-order and first-order Takagi-Sugeno algorithms;
- On-chip digital storage.

It is well know that the first-order Takagi-Sugeno algorithm yields better function approximations. Its output surfaces are continuous and smooth, keeping better accuracy while resulting in lower rule-count controllers than the zero-order counterpart for the same function to approximate. This represents a good reason to investigate its feasibility.

2.1 Architecture of the Controller

Figure 10-2 illustrates the block diagram of a general mixed-signal controller with M rules, N inputs, Q zero-order outputs and one first-order output. It comprises F independent Complementary Fuzzy Membership Function circuits (CFMF) arranged in two banks, followed by a switch matrix. This distributes the membership function outputs between the different T-Norm* operators. Each T-Norm, built by complemented maximum T-CoNorm circuits, corresponds to one rule. In addition, the T-Norms must support as many inputs as the maximum number of inputs Vin is desired. The number of inputs of the T-Norms can be programmed.

The inputs of all CFMFs circuits are left available off-chip. To configure a desired number x of inputs Vin, the inputs of the used CFMFs must be grouped in x groups (by off-chip wiring) whereas the T-Norms must be programmed so as to support the same number x of inputs.

At the consequent parts of the rules, the singletons are staked in Q columns. Each singletons column comprises M independent singletons in accordance with the number of rules M. The outputs of the singletons in a column are column-wise summed. On the other hand, each MAX circuit provides Q identical current signals that after being complemented (not shown in Figure 10-2) represent the firing degree of the same rule. The Q identical firing degrees of each rule are distributed between the Q singletons

* For simplicity, along this section we will reference the maximum circuits as the T-Norms instead of complemented T-CoNorms.

common to the same rule. For each column of singletons there is one weighting D/A and one divider that perform weighting and normalization, respectively. Each divider delivers the defuzzified value of each zero-order output of the controller.

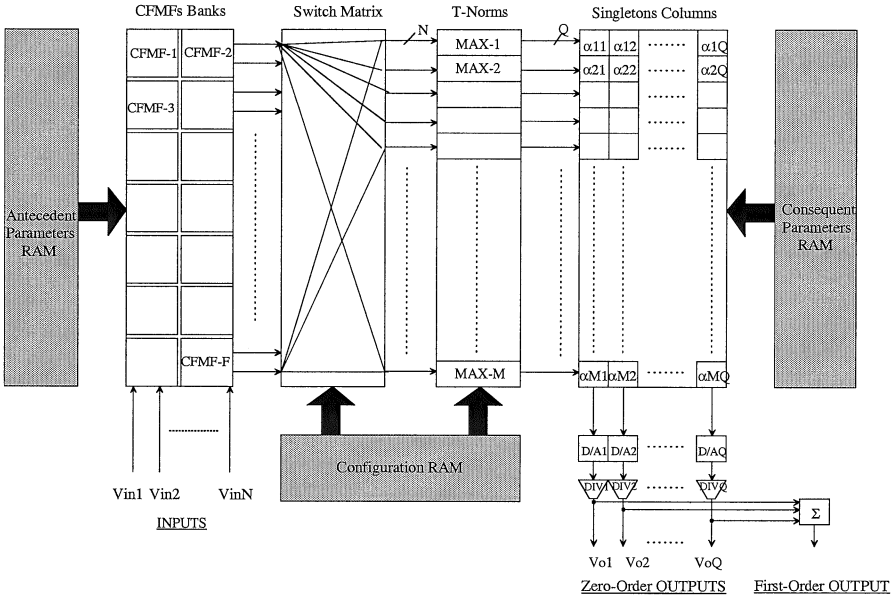

Figure 10-2. Block diagram of the fuzzy controller (From [1], © 2003 KAP).

To configure the first-order output all zero-order outputs are summed in a common Q-input voltage adder. At the same time, each active input Vin of the controller is connected to the reference input Vb1 of each divider. This is not shown in Figure 10-2 and will be detailed later.

The parameters of the membership functions (antecedent parameters), the values of the singletons (consequent parameters) and the configuration of the controller (the number of inputs of the T-Norms) are independently programmable by means of three independent distributed RAMs, represented by shaded blocks in Figure 10-2. The digital memories are built by D flip-flops arranged in different shift registers that facilitate the distribution of the stored bits among the different blocks. In the fabricated prototype, N was set to 5, F to 16, Q to 3, M to 27 and the singletons resolution n to 5.

2.1.1 Fuzzy Membership Function (FMF) Circuit

Two differential regulated-cascode triode-transconductors [1] make up the circuit drawn at Figure 10-3 a). It corresponds to a 4-parameter

electrically tunable FMF, whose shape is nearly trapezoidal, offering the possibility to synthesize either Direct or Complementary Fuzzy Membership Functions (CFMF). Cross-sums of the transconductors currents give rise to the current-mode outputs Iout1 and Iout2 and to the voltage-mode outputs Vout1 and Vout2. Voltages Vk1 and Vk2 take care of the placement of the fuzzifiers (labels) along the voltage range of Vin (the universe of discourse). Electrical programmability of the membership functions slopes is feasible by constraining transistors M1_1, M2_1, M1_2 and M2_2 to operate in the triode region. This is achieved by settling their drain-to-source voltage drop Vds to a small constant value (100 to 300 mV) throughout the regulated-cascode loop. Hence, their transconductance is given by:

$$gm = \frac{\partial Id}{\partial Vgs} = \mu Cox\left(\frac{W}{L}\right)_{M1\_1} Vds. \qquad (10.1)$$

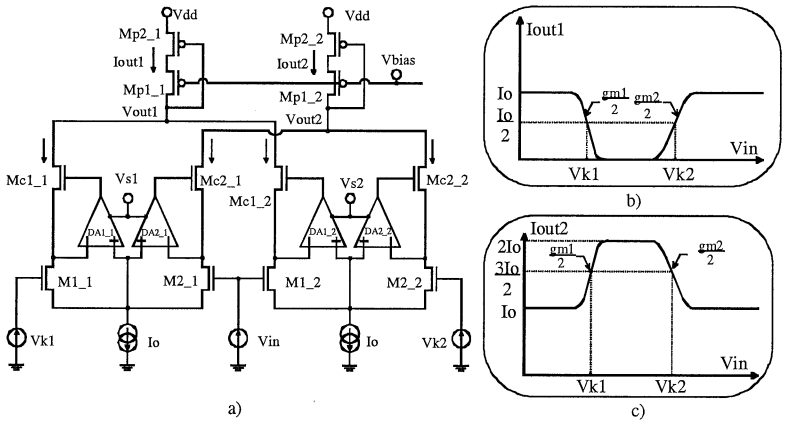

Figure 10-3. Fuzzy membership function: a) Circuit. b) CFMF. c) Direct FMF (From [1], © 2003 KAP).

The above Vds constant value (Vds=V- - V+, in DA1_1, DA2_1, DA1_2 and DA2_2) is yielded by the artificial offset created at the inputs of the loop Differential Amplifiers (DA), which are built by a non-symmetrical PMOS input differential pair [1]. The tail current of the DAs input pair is controlled by voltages Vs1 and Vs2, and thus the Vds of triode transistors. Finally, Vs1, Vs2, Vk1 and Vk2 are settled by means of a programmable discrete set of currents that bias MOS linear resistors.

Assuming that: $gm1=gm_{M1\_1}=gm_{M2\_1}$ and $gm2=gm_{M1\_2}=gm_{M2\_2}$, DC input-output relationships are graphically represented at Figure 10-3 b) and c). In our case, the gate voltage of transistor Mp2_1 (Vout1) in

Figure 10-3 a) is the signal representing the output of the CFMF. This signal is routed using a single wire towards the different T-Norms operators where the corresponding CFMF takes part. In this way, the routing space requirements are minimized and the accuracy of the controller is improved due to the avoidance of current mirrors for conveying signals.

2.1.2 Mixed-Mode Maximum Circuit and Switch Matrix

The maximum circuit is shown in Figure10-4. It consists of a set of N source-follower cells, each one comprising an amplifier A and a PMOS transistor M1 whose sources share the common-node C. Input currents I1...IN are supposed to be the output currents Iout1 of the CFMFs taking part in a rule. Consider the voltages at the non-inverting inputs of the amplifiers A. Since the source-followers are built by PMOS transistors, the common-node C follows the minimum non-inverting input voltage, corresponding to the maximum input current. This is replicated by transistors Mpc1 and Mpc2, which are equally sized to transistors Mp1_j and Mp2_j.

The inputs to be processed can be selected by means of the switches sw2_1...sw2_N. In this way, the number of inputs of the maximum can be adapted to the number of inputs Vin configured in the controller. By simply wiring, the same CFMF can be connected to different MAX operators in different rules. Since this circuit can handle voltage inputs while delivering current outputs, the interface to the subsequent defuzzification stage, performed in current mode, is straightforward fulfilled.

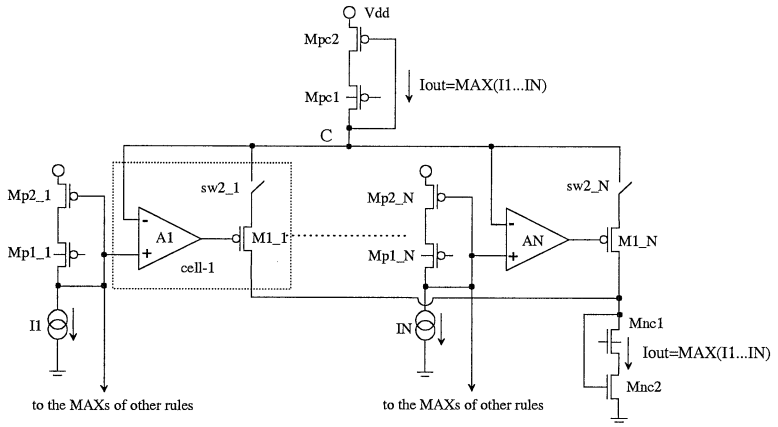

Figure 10-4. Mixed-mode N-input maximum circuit (From [1], © 2003 KAP).

For distributing adequately the outputs of the membership functions between the different rules several approaches could be considered. They are

extensively explained in [1]. In a first option, T-Norms have N inputs. An analog switch matrix with F inputs and M×N outputs allows connecting any membership function output to any T-Norm input of any rule. A second option consists in building T-Norms operators supporting up to F inputs whereas all CFMFs outputs in the banks are directly connected to one input of all T-Norms of all rules. Thus, each T-Norm would require F cells. Since the above options exceed the requirements of practical applications, we preferred using a wired switch matrix with programmable N-input T-Norms, with N<F. The distribution of the CFMF's outputs between the different T-Norms is performed randomly. However, an equilibrated participation of each CFMF in the whole set of rules must be warranted. In this way, our prototype can support grid fuzzy partitions for:

- 2-input, 3-zero order output, 5-CFMF per input, 25-rule controllers,
- 2-input, 1-first order output, 5-CFMF per input, 25-rule controllers,
- 3-input, 3-zero order output, 3-CFMF per input, 27-rule controllers.

Controllers supporting tree and scatter partitions are also affordable [1].

2.1.3 Consequent Singleton Columns and Defuzzifier

Figure 10-5 shows the architecture of the defuzzifier stage. The singletons αi stacked in a column are represented by n unit-gain current mirrors, whose outputs are column-wise summed. One extra mirror is required at each row for computing the sum of the firing degrees of the rules ($ID=\Sigma Ii$). The weighting operation is performed by one D/A only whose output delivers the weighting sum ($IN=\Sigma\alpha iIi$). It can be shown [11][1] that this strategy leads exactly to the same defuzzified output than the one yielded by using one weighting circuit per rule. The normalization operation is performed by the divider (DIV), whose transistors M1, M2, M3 at the bottom layer work in the triode region. While Vb1 and Vbo are fixed bias voltages, the gate voltage Vout of transistor M3 is self-adjusted so that the drain currents of M6 and M9 match. Hence, it can be shown [1] that:

$$Vo = (Vout - Vbo) = (Vb1 - Vbo)\frac{IN}{ID}. \qquad (10.2)$$

In first-order Takagi-Sugeno inference systems the consequents of the rules are linear combinations of the controller's inputs. In our implementation, it is possible to configure a 2-input, 1-output, 27-rule controller of this type. Since there are three independent singletons columns we can use one of them for the constant terms of the consequents linear expressions. The two others columns can be used for the coefficients multiplying the inputs variables Vin in the consequents linear expression.

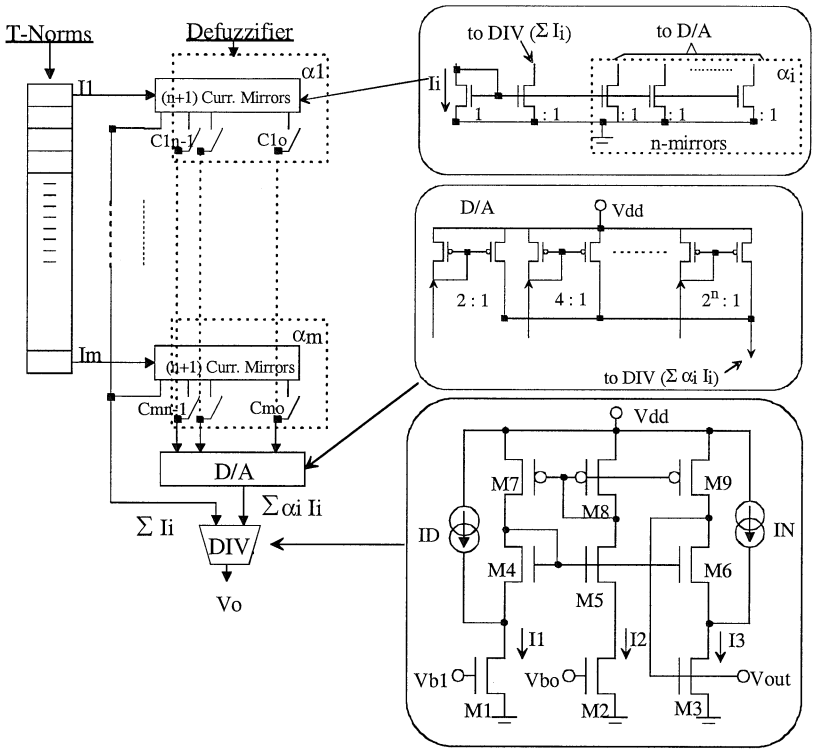

Figure 10-5. Singleton column and defuzzifier (From [1], © 2003 KAP).

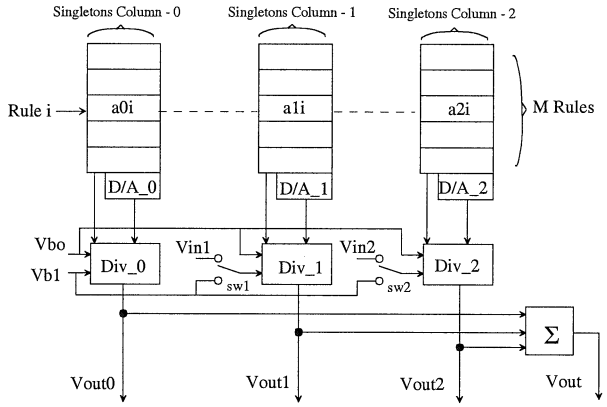

Figure 10-6. Zero and first-order configurations (From [1], © 2003 KAP).

This situation is clarified in Figure 10-6. When the switches sw1 and sw2 are set to Vb1, the controller yields three independent zero-order outputs

(Vout0, Vout1 and Vout2). In the opposite case, voltages Vin1 and Vin2 multiply their corresponding averaged linear coefficients and the first-order output is retrieved after the voltage adder. This sums the outputs of the dividers to deliver the signal Vout. In this way, by exploiting the entire divider relationship given by (10.2), the use of multipliers is avoided. For the voltage adder, a new high input impedance circuit that impels loading the dividers output was derived from the circuit of the divider itself [1].

2.1.4 Experimental Results

A 25-rule, 2-input, 5-label per input, 1-output, zero-order TS controller has been first configured in the chip. Figure 10-7 shows the settings and the measured results. The RMSE (Root Mean Square Error) between target and measured is 80mV (4% of full scale). The standard deviation between samples is kept below 180mV. Following, a 2-input, 1-output, 4-rule first-order TS controller has also been configured and tested. Each input variable Vin1, Vin2 has been fuzzified by means of two CFMFs, {P, N}. Figure 10-8 depicts the measured CFMFs and DC transfer function, wherein the smooth transitions between clusters can be appreciated.

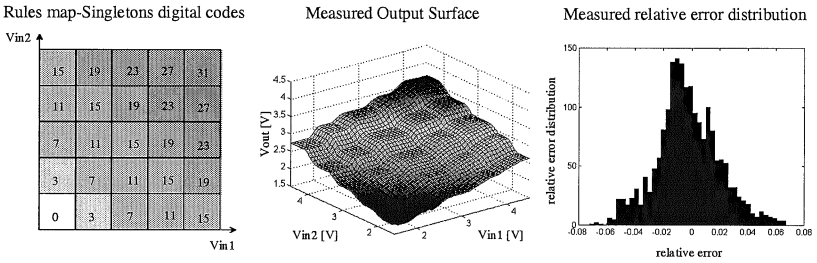

Figure 10-7. Zero-order controller: DC performance (From [1], © 2003 KAP).

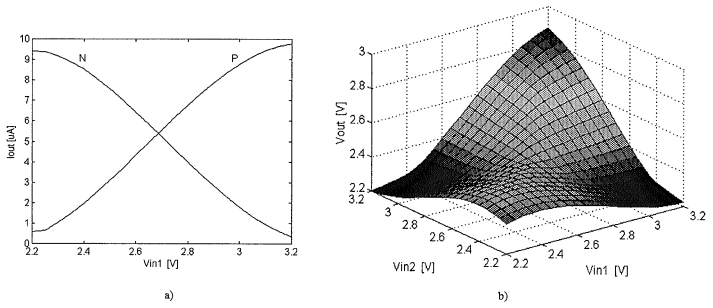

Figure 10-8. First-order TS controller: a) Measured CFMF for the labels P and N. b) Measured output surface (From [1], © 2003 KAP).

The transient behavior of this prototype has been addressed by means of large and small swing step responses. The settling times resulted in 1100ns and 570ns, respectively, to reach 90% of the steady state value. Therefore, the controller speed ranges from 0.9 to 1.75 MFLIPS (Mega Fuzzy Logic Inferences per Second). Figure 10-9 shows the microphotograph of the chip fabricated in CMOS-2.4μ technology. Table 10-1 summarizes the main features of our demonstrator.

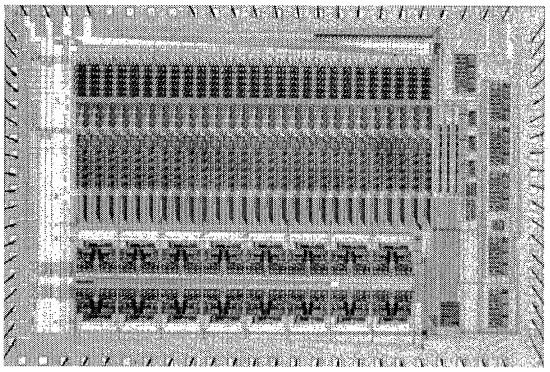

Figure 10-9. Chip microphotograph (From [1], © 2003 KAP).

Table 10-1. Main Features of the fabricated controller (From [1], © 2003 KAP).

27-rule General-Purpose Fuzzy Logic Controller			
Technology:	CMOS-2.4μ		
Processing Mode:	Continuous-Time		
Complexity:	a) Zero-order TS: 27-rule@5-input@3-output (max)		
	b) First-order TS: 27-rule@2-input@1-output (max)		
Power Supply:	Vdd=5V		
Chip Interface:	voltage-input@voltage-output		
Input/Output Voltage Range:	$1.5V \leq Vin \leq 4.5V$ /	$1.7V \leq Vout \leq 3.7V$	
Accuracy:	RMSE < 4.7%		
Power Consumption:	Core: 45mW		
	Buffers: 18mW		
	Total: 63mW	(2.33mW/rule)	
Area:	Analog: 22.8mm^2		
	Digital: 11.2mm^2		
	Core: 34mm^2		
	Pads: 5.5mm^2		
	Total: 39.5mm^2	(1.46 mm^2/rule)	
Parameters Programming:	Membership Functions:	Slopes:	2x4 bits
		Crossover Points:	2x5 bits
	Consequents:	Singletons:	5 bits
Total storage capacity needed:	909 bits		
Input/Output Delay:	Small swing: 570ns -	Large swing:	1.100ns

The chip digital interface comprises an 8-bit word: two bits for loading the CFMF's parameters in both banks, one bit for the T-Norms configuration and five bits for the consequent singletons. Involving a total number of 21000 transistors, our prototype demands 39.5mm^2 of silicon area. Estimations performed in [1] foretell a 60% of area reduction if this controller was implemented in a sub micron technology (i.e. CMOS-0.8μm).

2.2 Others Approaches on Mixed-Signal Controllers

In [12] a fuzzy coprocessor for digital controllers and DSPs is presented. Internal computation is carried out in the analog domain whereas inputs and outputs variables are digital. Allowing up to eight inputs and four outputs, a 32-rule programmable and reconfigurable fuzzy inference system has been fabricated in a 0.8μ CMOS technology. The processing mode is performed in sampled-time by means of switched-capacitor techniques. On the other hand, the implemented defuzzifying algorithm consists of a variant of Mamdani algorithm, which demands more computational efforts.

An analog continuous-time fuzzy controller fabricated in a 0.7μ CMOS technology is reported in [8]. It supports up to 15 rules, 3 inputs and 1 output. Only the zero-order TS algorithm can be configured. Each rule comprises an independent set of three FMF circuits. The antecedent and consequent parameters are discretely programmable whereas the FMF's slopes are defined at the mask level. All programmable parameters are on-chip generated and distributed through a programmable cross-matrix. The chip requires a 16-Kbit digital memory to be configured.

Table 10-2 shows the characteristics of these analog fuzzy processors.

Table 10-2. Main features of others analog fuzzy processors (From [1], © 2003 KAP).

	[12]	[8]
Technology:	CMOS-0.8μ	CMOS-0.7μ
Processing Mode:	Sampled-Time	Continuous-Time
Complexity:	32-rule@8-input @4-output (max)	15-rule@3-input @1-output (max)
Power Supply:	no data	Vdd=5V
Power Consumption:	no data	45mW (3mW/rule)
Input/Output Delay:	~2μs	~600ns
Accuracy:	no data	RMSE < 3%
Interface (inputs@outputs):	digital@digital	voltages@voltages
Input-Range@Output-Range	8-bit@6-bit	2V@2V
Area (total):	70mm^2	32mm^2
Programmability		
FMF Position:	on-chip	on-chip
FMF Slope:	on-chip	fixed
Consequents:	on-chip	on-chip
Total storage capacity needed:	no data	16-Kbit

3. APPLICATIONS

Low-power consumption, autonomy and fast time response are needed for real-time processing applications. Therefore, the implementation of embedded fuzzy controllers becomes an attractive option, if not the unique in some cases. In this section, we describe briefly some reported application whose requirements are fairly fulfilled by analog implementations as the ones presented in the previous sections.

In [8], a PWM DC/DC converter is controlled by means of nonlinear laws implemented with an analog fuzzy controller. The block diagram and the controlling laws are shown in Figure 10-10. Current IL and voltage error E=Vout-Vref are fed back through the fuzzy surfaces DP and DI in order to regulate the PWM duty cycle D for a desired voltage at the output Vout. For a switching frequency of 500KHz, the system demands a fast settling time with reduced switching current spikes. These timing constraints favor the implementation of the controller towards the analog domain, as it avoids the additional A/D or D/A conversion delays present in digital controllers [8].

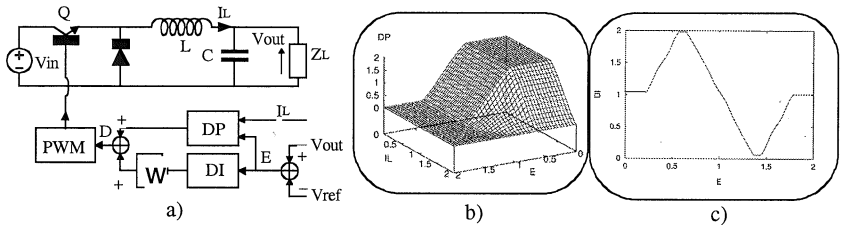

Figure 10-10. a) PWM DC/DC converter. b), c) Control surfaces (From [8], © 1998 IEEE).

Applications of fuzzy logic to signal processing are being increasingly reported. A method for analog filter tuning in frequency-domain is described [6]. This is shown in Figure 10-11, where the fuzzy controller attempts to place the filter frequency response |H(w)| within a typical window specification. Upon measuring the filter response to a discrete set of different frequencies sinusoidal waves (on-chip generated as Test Signals), the output of the controller updates some filters parameters according to the desired response. The area tolerance in the window makes relaxing the accuracy needed for the controller. Thus, the control policy can be easily synthesized through fuzzy statements.

In [1], a time-domain signal analysis technique employing fuzzy logic is discussed. The general setup is shown in Figure 10-12 a). It consists in exploring the signal over time and converting it into a geometrical figure, as in any oscilloscope. Clusterizing the space [Time, Signal-Value] by means of fuzzy partitions, signal attributes can be identified. In this way, a fuzzy

decision making arrangement can be settled and after a comparison with a reference pattern, the fuzzy rule-set may infer meaningful assertions that could be used for adaptation, detection, testing, etc. This kind of "on-chip oscilloscope" can also be used for analog filter tuning, as shown in Figure 10-12 b). The test signal consists in a square wave exciting the filter and generating the linear ramp. This, together with the actual filter output, is input to the controller that encodes the desired step response and the control strategy upon deviations. Apart from this example, this technique could also be exploited for building analog Built-In Self-Test (BIST) units [1].

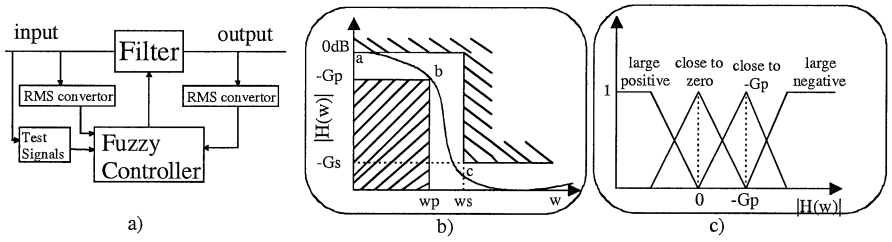

Figure 10-11. Frequency-domain filter tuning: a) Block Diagram. b) Window specifications. c) Membership functions to quantify magnitude in passband (From [6], © 1994 IEE).

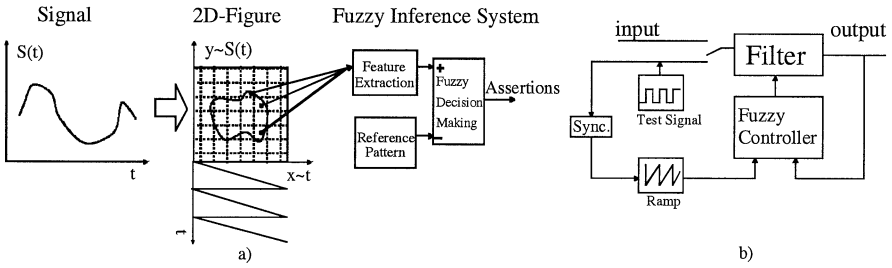

Figure 10-12. a) Time-domain signal analysis setup. b) Time-domain filter tuning (From [1], © 2003 KAP).

4. CONCLUSIONS

We addressed the design, test and application of analog fuzzy controllers in CMOS technologies for embedded applications of medium to high speed. From the analysis perspective of the different fuzzy algorithms, the implementation of Takagi-Sugeno controllers has been found to be highly feasible because their good trade-off between simplicity and accuracy. Our demonstrator features an acceptable flexibility and requires a relatively small

size memory. This is the result of the chosen architecture, which allows an optimal interface between the functional blocks.

Away from the traditional control field, others potential applications of fuzzy logic have been briefly described. For these, the use of embeddable analog controllers is justified. Considering the synthesis of analog non-linear systems, fuzzy logic provides straightforward solutions in a higher abstraction level framework than classical device physic-based approaches.

ACKNOWLEDGEMENTS

The authors want to thank the Universidad Católica de Córdoba and the Belgian FNRS for their support. Michel Verleysen is a FNRS Senior Research Associate.

REFERENCES

1. Dualibe C., Verleysen M. and Jespers P., "Design of Analog Fuzzy Logic Controllers in CMOS Technologies: Implementation, Test and Applications", *Kluwer Academic Publishers (KAP)*, ISBN: 1-4020-7359-3, The Netherlands, January 2003.
2. Kosko B., "Neural Networks and Fuzzy Systems", Chapter 1, pp. 24-26, *Prentice-Hall, Inc.*, Englewood Cliffs, ISBN: 0-13-611435-0, New Jersey 07632, USA, 1992.
3. Holmblad P. and Ostergaard J., "Fuzzy Information and Decision Processes, Chapter: Control of a Cement Kiln by Fuzzy Logic", pp. 389-399, *North-Holland*, 1982.
4. Layne J. Passino K. and Yurkovich S., "Fuzzy learning control for anti-skid braking systems", *IEEE Transactions on Control Systems Technologies*, Vol. 1, N° 2, pp. 122-129, Feb. 1993.
5. Huang C.Y., Ker J.S., Liu B.D. and Kuo Y.H., "FPGA Implementation of Fuzzy Color Correction System", in *Proc. of the IEEE 1995 Custom Integrated Circuit Conference*, pp. 503-506, USA, 1995.
6. Moreira-Tamayo O., Pineda J. and Sanchez-Sinencio E., "Filter Tuning System Using Fuzzy Logic", *Electronics Letters*, Vol. 30, N° 30, pp. 846-847, May 1994.
7. Park H. and Kong S., "Neuro-Fuzzy Control System for Adaptive Noise Cancellation", *IEEE International Fuzzy Systems Conference*, Seoul, Korea, Aug. 22-25, 1999.
8. Franchi E., Manaresi N., Rovatti R., Bellini A. and Baccarani G., "Analog Synthesis of Nonlinear Functions Based on Fuzzy Logic", *IEEE Journal of Solid State Circuits*, Vol. 33, N°6, pp. 885-895, June 1998.
9. Costa A., De Gloria A., Faraboshi P., Pagni A. and Rizzotto G., "Hardware Solutions for Fuzzy Control", *IEEE Proceedings*, Vol. 83, N° 3, pp. 422-434, March 1995.
10. Rodríguez-Vázquez A., Navas R., Delgado-Restituto M. and Vidal-Verdú F., "A Modular Programmable CMOS Analog Fuzzy Controller Chip", *IEEE Trans. on Circuits and Systems-II: Analog and Digital S. Processing* Vol. 46, N°3, pp. 251-265, March 1999.
11. Landolt O., "Low-power analog fuzzy rule implementation based on a linear MOS transistor network", *Proceedings of MicroNeuro '96*, pp. 86-93, Lausanne, 1996.
12. Fattaruso J., Mahant-Shetti S. and Barton B., "A Fuzzy Logic Inference Processor", *IEEE Journal of Solid State Circuits*, Vol. 29, N°4, pp. 397-402, April 1994.