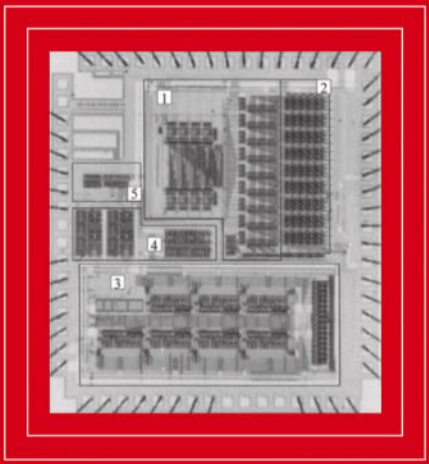
Design of Analog Fuzzy Logic Controllers in CMOS Technologies

Implementation, Test and Application

Carlos Dualibe, Michel Verleysen and
Paul G.A. Jespers



Kluwer Academic Publishers

DESIGN OF ANALOG FUZZY LOGIC CONTROLLERS IN CMOS TECHNOLOGIES

This page intentionally left blank

Design of Analog Fuzzy Logic Controllers in CMOS Technologies

Implementation, Test and Application

by

Carlos Dualibe
Universidad Católica de Córdoba, Argentina

Michel Verleysen
Université Catholique de Louvain, Belgium

and

Paul G.A. Jespers
Université Catholique de Louvain, Belgium

KLUWER ACADEMIC PUBLISHERS
NEW YORK, BOSTON, DORDRECHT, LONDON, MOSCOW

eBook ISBN: 0-306-48014-X
Print ISBN: 1-4020-7359-3

Created in the United States of America

Visit Kluwer Online at: http://kluweronline.com
and Kluwer's eBookstore at: http://ebooks.kluweronline.com

Contents

This page intentionally left blank

Contributors

Carlos Dualibe
Laboratorio de Microelectrónica
Universidad Católica de Córdoba
Camino a Alta Gracia, Km 10
5016-Córdoba, Argentina

Michel Verleysen and Paul G. A. Jespers
DICE Laboratory
Université catholique de Louvain
Place du Levant, 3
1348-Louvain-la-Neuve, Belgium

This page intentionally left blank

Acknowledgements

We would like to express our gratitude to all persons who contributed in the realization of this book. We are grateful to Prof. Denis Flandre, Prof. Piotr Sobieski and Prof. Vincent Wertz from the Université catholique de Louvain (UCL) and Prof. Joos Vandewalle from the Katholieke Universiteit Leuven (KUL), for their helpful remarks that have certainly improved this manuscript. We thank the Secrétariat à la coopération internationale of the UCL, the Belgian FNRS (Fond National pour la Recherche Scientifique) and the Universidad Católica de Córdoba (UCC) in Argentina, for their financial support. Michel Verleysen is a FNRS Senior Research Associate. We are plenty indebted with our colleagues from DICE laboratory at UCL, for the nice and friendly environment at the laboratory. We specially thank to Brigitte Dupont and her group for providing us the state-of-the-art on informatics. Our gratefulness is extended to all professors of the UCC for their encouragement. Our special thank is addressed to Prof. Carlos Marqués, Prof. Oscar Sartori, Prof. Eduardo Toselli, Prof. Carlos Diamanti and Prof. Miguel Moreno S.J., for their intense support for this research.

Finally, we thank to Catherine, Denise, Dina and Emiliano, our families. Without their endless patience, this book would have never been accomplished.

Carlos Dualibe, Michel Verleysen and Paul G. A. Jespers.
Louvain-la-Neuve, Belgium, 2002.

This page intentionally left blank

Preface

In the last years, the astonishing growth of the Japanese industry in producing a substantial number of consumer appliances using Fuzzy Controllers put Fuzzy Logic on the focus of the scientific community. At the beginning, the most popular applications of Fuzzy Logic were found in the domain of Control System. Nowadays, the application of this soft-computing technique has been extended to other fields such as Signal Processing, Image Processing and Switching Power Control, for instance. As real-time applications need ever 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 book deals with the implementation, test and application of programmable and reconfigurable Analog Fuzzy Logic Controllers in standard CMOS technologies in three fundamental stages.

In the first part, the analysis and design of basic analog building blocks have been addressed. Main topics concerning their accuracy, programmability, interfacing and VLSI compatibility for CMOS implementation have been focused. Some novel circuits are presented while others are optimized towards an improved behavior.

The second part comprises the implementation and test of programmable and reconfigurable mixed-signal architectures being capable of emulating Zero and First-Order Takagi-Sugeno algorithms. In the realized prototypes, signal processing is carried out in the analog domain whereas the system parameters and configuration are digitally programmable. The applied

testing strategy was oriented to characterize the DC and transient behaviors of the controllers as well as the statistic spreading between samples.

Finally, in the third part, a real-time application of Fuzzy Logic is undertaken in the Analog Signal Processing field: a knowledge-based technique for time-domain signal analysis is discussed. The general idea consists in building an "on-chip oscilloscope", which, based on Fuzzy Logic, could infer assertions that can be used for adaptation, testing, detection, etc. This technique has been used in a digital equalization system based on the Eye Pattern. For this purpose, a preliminary prototype comprising the Fuzzy Controller and the equalizing filter has been fabricated and tested whereas the methodology has been validated by simulations for cable equalization.

This book results from the first author's PhD thesis. It is mainly addressed to researchers, undergraduate and postgraduate students working in the field of analog VLSI implementation of Fuzzy Systems and their applications. However, the analysis and synthesis of the circuits presented herein is wide-ranging. Their use exceeds the topic of Fuzzy Logic since they can also be employed in other kind of applications in the field of Analog Signal Processing (i.e. Neural Networks, Non-Linear and Linear Adaptive Filtering, Analog Computation, etc).

Chapter 1

INTRODUCTION

Motivation and goals of this book

1. INTRODUCTION

Fuzzy Logic was originally developed in the early 1960's by Professor Lotfi Zadeh, who claimed for a new kind of computational paradigm capable of modeling the own uncertainness of human reasoning. In 1965, Zadeh published the first ideas on fuzzy sets, the key concept in Fuzzy Logic (FL).

The acceptance of this soft-computing technique by the highly "deterministic" scientific community was not immediate. At the beginning, the most popular applications of Fuzzy Logic were found in the domain of Control System. On one hand, many conservative engineers in such area claim that Fuzzy Control does not convey to better solutions than the classical ones and that Fuzzy Logic is just a marketing hype. On the other hand, several non-specialist researchers misinterpret the fact that Fuzzy Logic deals with uncertainness claiming that "*fuzzy systems reason as humans do* ", as they use to say. This misunderstanding leads some people to believe that Fuzzy Logic is a kind of *cure-all* that can solve any kind of problem.

Away from any kind of fanaticism however, Fuzzy Logic is a rigorous mathematical field [Godj97]. Fuzzy reasoning is nothing else than a straightforward formalism for encoding *human knowledge* or *common sense* in a numerical framework. In a Fuzzy Controller, human experience is codified by means of *linguistic if-then* rules that build up a so-called Fuzzy Inference System, which computes control actions upon given conditions.

Fuzzy Logic has been applied to problems that are either difficult to face mathematically or applications where the use of Fuzzy Logic provides

improved performance and/or simpler implementations. One of its main advantages lies in the fact that it offers methods to control non-linear plants, known difficult to model.

Since the first reported application of Fuzzy Logic [HoOs82], the number of industrial and commercial developments, covering a wide range of technological domains, has grown incessantly. Nowadays, countless researchers from different areas are hardly working on the subject while contributing with smart and interesting solutions for engineering. Table 1.1, which summarizes the historical development of Fuzzy Logic, highlights some of its most significant milestones as reported by [KaLa98].

Table 1.1. The historical development of Fuzzy Logic until 1993. From [KaLa98], © 1998 Kluwer Academic Publishers, reprinted with permission.

Year	Event
1961	Lotfi Zadeh claims in his paper for a new kind of "fuzzy" mathematics.
1965	Lotfi Zadeh introduces the concept of fuzzy sets.
1969	Marinos (Duke University) conducts the first research aiming at hardware implementation of Fuzzy Logic.
1972	M. Sugeno presents the idea of fuzzy measures.
1974	E. Mamdani presents a fuzzy application to control a steam machine in an academic framework.
1980	Yamakawa build the first fuzzy circuit with discrete bipolar components [Yama80].
1982	The first industrial application in a cement kiln in Denmark [HoOs82].
1984	Togai and Watanabe present the first VLSI implementation of Fuzzy Controllers [ToWa85].
1986	Hitachi puts into operation a fuzzy controlled subway system.
1987	Yamakawa presents the first analog Fuzzy Controller.
1988	Togai implements the first digital fuzzy processors.
1989	LIFE: Laboratory for International Fuzzy Engineering Research starts in Japan.
1990	Yamakawa establishes the Fuzzy Logic Systems Institute (FLSI) in Japan.
1992	The first IEEE International Conference on Fuzzy Systems.
1993	The first issue of the IEEE Transactions on Fuzzy Systems.

In the last years, the astonishing growth of the Japanese industry in producing a substantial number of consumer appliances using Fuzzy Controllers put Fuzzy Logic on the focus of the scientific community. In 1990, the market of Fuzzy Logic based products was estimated nearly equal to $2 billion [Paty92]. According to an investigation of the Market Intelligence Research Co. of California, in 1991 Japan captured 80% of the worldwide market. In 1992, the return in fuzzy products doubled with respect to the previous year, whereas companies, like OMROM, held about 700 patents at that date. Germany, India, France, Korea, Taiwan and China follow Japan in Fuzzy Logic R&D projects.

This economical success is said to be mainly due to the short time-to-market needed by the development of Fuzzy Controllers for these particular applications [ChCh95] [DrHe93]. However, this is true if sufficient expertise coming from skilled designers is available [Godj97].

In early Fuzzy Logic applications, empiricism was the main method for selecting the parameters of Fuzzy Controllers. Recent methods for automatic identification of parameters in Fuzzy Systems from training data paved the road to the application of Fuzzy Logic to systems where human knowledge is not available [BeKh92] [Jang92] [HiSu92]. Most of these are inspired from Artificial Neural Network learning techniques and lead to the development of the so-called Neuro-Fuzzy Systems [BrHa94]. The integration of fuzzy and neural techniques is nowadays an active area of research. It brings together the best features of Fuzzy Logic and Neural Networks. It provides knowledge-based systems that can be adapted or optimized according to sample data. In summary, a Neuro-Fuzzy System can codify structured knowledge as a Fuzzy System while preserving the adapting and learning capabilities of Neural Networks. To be effective, such a Neuro-Fuzzy System must hold, at least, the following basic aspects [Vida96]:

- Succinct and appropriate representation of structured knowledge.
- Tuning capabilities for parameters adaptation and identification.
- Straightforward relationship between the above parameters and the structured knowledge.

In this way, using a Neuro-Fuzzy System, the extraction of linguistic rules from the adapted parameters also gives designers the possibility to verify and/or rectify their prior knowledge [HiSu92] [Godj97]. A clever solution for this purpose was found by [Jang92] who managed the representation of a Fuzzy System by means of a multi-layer feed-forward network called ANFIS (Adaptive Network-based Fuzzy Inference System). ANFIS is able to learn from data by using the gradient descent algorithm. A remarkable feature of ANFIS is its fast learning rate when compared, for instance, with feed-forward perceptrons Neural Networks. This is due to the previous encoded knowledge before learning starts [Mend95].

Finally, let us summarize the major advantages and disadvantages of Fuzzy Logic reported in the literature. The main features of Fuzzy Logic encouraging 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, for instance [Neye97].
- In most Fuzzy Control problems, the exact model of the plant is not needed for designing the controller provided that further adjustment can be made during operation of the system [Neye97] [Godj97].
- A Fuzzy Inference System is a Universal Approximator. It 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 like ANFIS [JaGu95].
- Fuzzy Logic can be combined with classical control techniques [JaGu95]. A common trend that illustrates this is to use a Fuzzy Controller to supervise a conventional adaptive controller, whose adapting strategy is encoded into the rule base of the Fuzzy Controller.

In spite of its success and popularity, Fuzzy Control has some remarkable drawbacks too:

- In some cases, it is difficult to warrant the consistency of the rule base [Godj97]. Conflictive rules may appear, specially when the input space has a considerably large dimension.
- There is no generalized criterion to formally demonstrate the system stability. This has been addressed in particular cases only [Neye97].
- There is a lack of systematic methods to translate human knowledge into the rule base. Moreover, the expert knowledge is sometimes incomplete or vaguely defined. There is no clear-cut procedure for choosing the suitable number of rules as far as the many factors involved [Godj97].

Nonetheless, many researchers succeeded in automating the modeling and optimization processes of Fuzzy Systems. A comprehensive work in this area can be found in [SuYa93]. In this, as an extension of the classical System Identification methodology used for linear systems [GeBa94], a three-step identification procedure is proposed:

1. Variables identification: means to identify the input variables, among a set of candidates, that play an important role in the process to control.
2. Structure identification: means to identify the appropriate number of rules and membership functions per input/output and the correct fuzzy partition type (i.e. grid, tree or scatter partition).
3. Parameters identification: means to identify the proper shape and position of the membership functions.

Apart from the high computational efforts demanded by the above methodology, its success relies mainly on availability of the measured sample data.

1.1 R&D Fields of Fuzzy Logic

As already stated above, the applications of Fuzzy Logic (FL) to Control Systems and Process Control where the mathematical model of the plant is unknown, complex, ill defined if not time varying, were developed in the beginning. Nowadays, the areas where Fuzzy Control has been applied comprise a wide variety of applications, with different complexity and performances. We can find Fuzzy Controllers in washing machines, automatic focusing for video cameras, automatic TV tuner, servo motor control, automotive anti-skid brake, and many others consumer appliances. At present, the application of Fuzzy Logic exceeds the control domain since it is also employed for others knowledge-based decision making tasks. Among the latter, we can mention medical diagnosis, business forecasting, traffic control, network management, image processing, signal processing, computer vision, geology and many more [CoG195].

Table 1.2 covers a range of research areas related to Fuzzy Logic as reported in the IEEE 2001 International Conference on Fuzzy Systems. Table 1.3 refers to some reported applications in different domains, wherein the use of FL offered considerable advantages.

Table 1.2. Main R&D areas in Fuzzy Logic and their major topics (FUZZ-IEEE' 2001).

R&D Area	Main Topics
Fuzzy Mathematics:	Foundations of Fuzzy Logic, approximate reasoning, evolutionary computation, identification and learning algorithms, rule base optimization.
Control Systems:	Fuzzy Control theory and applications, process and environmental control, stability criterions issues, multi-level supervisory control.
Pattern Recognition and Image Processing:	Supervised and unsupervised learning, classifiers design and integration, signal/image processing and analysis, computer vision, multimedia applications.
Soft Computing and Hybrid Systems:	Intelligent information systems, data base systems, data mining, intelligent agents, reliability engineering, Neuro-Fuzzy Systems, Internet computing, networks traffic modeling and control.
Electronic Systems:	Fuzzy hardware implementation and embedded applications.
Robotics and Automation:	Fuzzy Logic in robotics, industrial automation and other industrial applications.

Table 1.3. Some reported applications of Fuzzy Logic in different domains.

Application	Reference
Process Control	
5MW Nuclear Reactor Control	[Bern88]
Cement Kiln control	[HoOs82]
Sake Brewing Manufacturing	[OiTo91]
Activated Sludge Wastewater Plant	[ToBe80]
Water Purification Plant	[Yalt85]
Multilayer Waste Incinerator	[SuKa86]
Environmental Emissions Control	[TaSa94]
Aero-Space Applications	
Aircraft Carrier Landing System	[Stie93]
Attitude Control of a Flexible Satellite	[DaGi89]
Flexible Wing Roll Control	[ChCh91]
Pitch Control of an Interceptor Missile	[Chri93]
Robotics and Automation	
Mobile Robot Control	[SaRu93]
Robot Arm Control	[ScMa85]
Container Crane Control	[YaHa86]
Industrial Automotive	
Car Modeling in Extreme Situations	[AlKr92]
Automotive Automatic Transmission	[SaSa93]
Automotive Anti-Skid Braking	[LaPa93]
Autonomous Vehicle Controller	[HaRe89]
Automatic Train Operation	[YaMi85]
Car Parking Modeling	[SuMu85]
Automotive Engine Idle Control	[VaFa93]
Image Processing	
Real-Time Motion Estimation	[ChJo99]
Image Color Correction Systems	[HuKe00]
Facsimile Image Fuzzy Processor	[LeLe93]
Image Processing in IQTV	[MaDa95]
Fuzzy Stack Filters for Image Processing	[YuCh96]
Signal Processing	
Non-linear Channel Equalization	[WaMe94]
AGC Control for Radio Communications	[BaDi00]
Decision Feedback Equalizer	[LiMe00]
Magnetic Recording Equalization	[Mage94]
Analog Filter Tuning	[MoPi94]
Adaptive Noise Cancellation	[PaKo00]
Real-Time Source Separation	[ToHa98]
Fuzzy Digital PLL Filter	[SiEl95]
Earthquake Precursors Detection	[ItWa92]
Power Electronics	
Control of DC/DC Converters	[CrLi98]
Control of PWM DC Converters	[FrMa98]
Embedded Low Cost Control for DC Motors	[OeGr96]
Fuzzy Control of Induction AC Motors	[ClTu92]

Notice from the latter table the increasing number of applications in the domain of Image Processing, Signal Processing and Power Electronics that have been reported in the last decade. 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.

1.2 Fuzzy Controllers Implementation Choice

A comprehensive taxonomy of the commonly used approaches for the physical implementation of the fuzzy reasoning algorithms can be found in [CoG195]. In order to situate this work within the context of the different hardware alternatives the main conclusions of the latter mentioned reference are summarized herein.

Typically, a Fuzzy Controller is composed by a set of *if-then rules* involving three basic operations:

- Fuzzification: is the operation that translates *crisp* input data into a membership degree by means of the so-called membership functions.
- Fuzzy Inference: this operation uses the membership degrees to deduce a fuzzy output for each rule and a final fuzzy output of the controller.
- Defuzzification: is the operation that translates the final fuzzy output into a *crisp* value, compatible with the deterministic external environment.

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 used as a performance parameter for comparisons. Figure 1.1 a) shows a classification of typical control problems depending on the complexity (i.e. number of rules) and the system response time.

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 structures for implementing Fuzzy Controllers.

[CoG195] identifies four classes among the different hardware implementation alternatives:

1. *General-Purpose Processors*: implementation of fuzzy algorithms on standard microcontrollers is the most widely used technique.

2. *General-Purpose Processors with dedicated fuzzy instructions:* still maintains general-purpose computation capabilities by employing general-purpose processors (i.e. CISC and RISC) with the addition of a few specialized instructions to accelerate the fuzzy operations.
3. *Special-Purpose Coprocessors:* these are special-purpose processors dedicated to fuzzy computations. They cannot implement the entire control system by themselves due to the lack of general-purpose computation capability, but still provide some flexibility and configurability features.
4. *Dedicated ASICs:* direct implementation on silicon of fuzzy algorithms by using high-level or full-custom synthesis techniques (i.e. analog, digital or mixed-signal). The FPGA-based implementations also fall in this category.

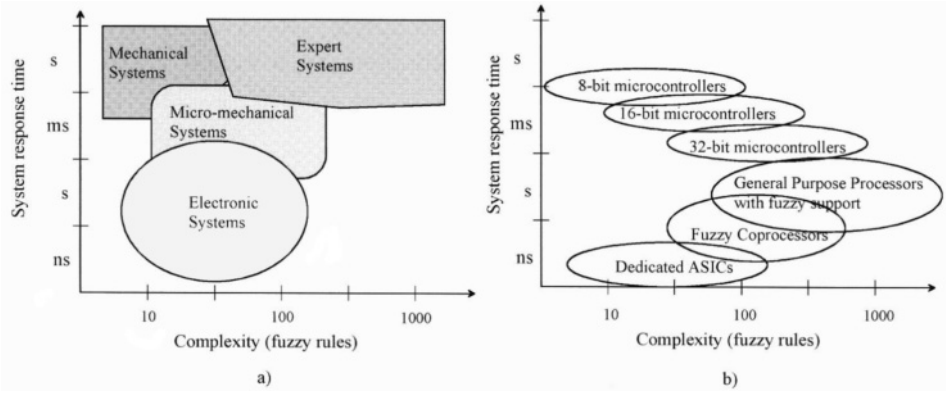


Figure 1.1. a) Allocation of the typical control problems in the complexity-time response space. b) Allocation of the possible hardware solutions in the complexity-time response space. From [CoGl95], © 1995 IEEE.

Figure 1.1 b) shows a qualitative allocation of the different classes in the space complexity-time response. Table 1.4 summarizes the main advantages and disadvantages in each alternative.

It is evident that speed, complexity, flexibility, interaction requirements, 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, but the different approaches should be considered according to the requirements of a given design.

Table 1.4. Main advantages and disadvantages for each hardware implementation option.

Alternative	Advantages	Disadvantages
1. General-Purpose Processors:	- Complete flexibility - Short prototyping time - For high complexity FLCs - Availability of development systems [Moto92] [Info94]	- Low speed (control systems with slow dynamics: 0.1-1 KHz)
2. General-Purpose Processors with Dedicated Fuzzy Instructions:	- Complete flexibility - Short prototyping time - For high complexity FLCs - Higher speed compared with 1: 1-10KHz (CISC), 1-50KHz (RISC) - Simple extension of existing CPU cores [Info94]	- Higher cost than 1 (due to small production volume) - Low performance for high-end applications
3. Special-Purpose Coprocessors:	- Higher speed than 1 and 2 (i.e. 10-100KHz) - For moderate complexity FLCs (256-rule, 16-I/16-O) - Short prototyping time - Modularity	- Need of general-purpose processor to support non-fuzzy operations - Limited flexibility [ShOs92] [PaPo92]
4. Dedicated ASICs:	- Fast processing, tuned to the application (i.e. 0.1-1MHz) - Availability of high-level synthesis tools [FrMa98] [HoHa96] - Low cost in terms of die area - Available as a subsystem block for embedded applications - Cost-effective for high production volumes	- Fixed applications - Low flexibility (minimal programmability) - Limited complexity FLCs - Only for ASIC or FPGA based systems - Large prototyping time

1.3 Objectives and Plan of this Book

The signals, which a Fuzzy Controller deals with, are analog in nature. Fuzzifiers, Inference Operators and Defuzzifiers handle continuous variables. Therefore, designing fuzzy hardware should be oriented towards the analog domain mainly. Digital Fuzzy chips provide enough potential for general applications but their speed is limited. Furthermore, the use of Analog-to-Digital and Digital-to-Analog converters becomes mandatory for interfacing sensors and actuators in a real application. Consequently a further speed deterioration together with an increased complexity can be expected. On the other hand, it has been shown in the previous sections that in the last years the use of Fuzzy Logic has been extended beyond the classical Control field. Signal Processing, Image Processing and Power Electronics seem to be

possible niches where this soft-computing technique can meet a broad range of applications. Hence, as those real-time applications need ever faster, more autonomous and less power consuming circuits, the choice of analog circuits becomes an interesting option, specially in the cases where embedded applications are being contemplated.

Motivated by the above realities this book focuses on the design, implementation and test of programmable analog Fuzzy Controllers using standard CMOS technologies for applications of medium to high-speed. For this purpose, the rest of this book is organized in five chapters as follows:

- **Chapter 2:** it presents a general overview of Fuzzy Logic and Fuzzy Systems in order to familiarize the readers with the subject. The fundamental topics of Fuzzy Logic theory are introduced: *fuzzy sets, operation with fuzzy sets, fuzzy operators, linguistic variables, fuzzy propositions, linguistic rules and approximate reasoning.* The structure of a Fuzzy Controller supporting different kinds of fuzzy algorithms is described together with a succinct review of the general guidelines for designing and optimizing Fuzzy Controllers. Finally, in the frame of supervised learning, the usefulness of ANFIS is demonstrated through an example.
- **Chapter 3:** in this chapter the analysis and design of the basic analog fuzzy building blocks is addressed. Main topics concerning systematic errors, mismatch errors, programmability and interfacing are focused. Some novel circuits are presented while others previously reported are optimized towards an improved behavior. Despite the fact that only discretely programmable demonstrators have been integrated, the electrical tuning capabilities of the fuzzifiers and consequents are also considered. This would make feasible the implementation of fully analog programmable Fuzzy Logic processors in technologies where analog storage devices are available.
- **Chapter 4:** it comprises the implementation and test of programmable as well as reconfigurable mixed-signal architectures for Zero and First-Order Takagi-Sugeno fuzzy algorithms. Two controllers have been implemented. In the first one, a small singleton controller intended for embedded applications, antecedent and consequent parameters can be programmed whereas the number of rules, of inputs and outputs is fixed. In the second general-purpose programmable MIMO (Multiple-Input Multiple-Output) controller, reconfiguring capabilities are introduced by allowing to choose the number of inputs, the number of outputs and the number of rules. In addition, a particular innovation has been implemented in the latter controller, since it is able to easily synthesize the First-Order Takagi-Sugeno algorithm.

- **Chapter 5:** in this chapter a real-time application of Fuzzy Logic is considered on the Analog Signal Processing field. A knowledge-based technique for time-domain signal analysis is discussed. This leads to the idea of the "on-chip oscilloscope" that can be used for adaptation, detection, testing, etc. As an example, this methodology is used in a self-adaptive cable equalization system based on the Eye Pattern. To get primary insight about its feasibility, a preliminary prototype comprising the Equalizing Filter and the Fuzzy Controller has been fabricated and tested. Finally, the methodology was validated by simulations of the complete system.
- **Chapter 6:** in this last chapter we summarize the main achievements and discuss future improvements and open challenges.

This page intentionally left blank

Chapter 2

FUZZY LOGIC AND FUZZY SYSTEMS

A survey

2. INTRODUCTION

Fuzzy Systems deal with human reasoning from a high level of abstraction point of view. A Fuzzy Inference arrangement intends to reproduce algorithmically the structured human knowledge by encoding it into a set of *"if-then"* rules. Each rule represents an *action* to execute when a certain *condition* is met. This remarkable feature distinguishes Fuzzy Systems from Artificial Neural Networks. The latter intends to reproduce a human brain in a low level of abstraction approach (i.e. by means of massively interconnected elemental processing units) [Verl92].

The information processing mode based on structured knowledge was widely focused a long time ago by means of Artificial Intelligence engineering. Expert Systems process inference rules, which are logic implications associating *actions* to *conditions,* by means of symbolic representations. Symbols model *thinking* or *short-term memory* whereas the relation between symbols models the *long-term memory.* Instead of learning there is programming and the logic inference replaces the non-linear dynamics of Artificial Neural Networks [Kosk92].

However, since symbols cannot be mathematically handled, hardware implementation of expert systems cannot be efficiently afforded as such. In contrast, Fuzzy Systems codify structured knowledge supported on a numerical framework, so that they can be physically implemented either with digital or analog circuits [Vida96].

Expert Systems, Artificial Neural Networks and Fuzzy Systems share the property of being model-free approximators, which means that no exact

mathematical model of the physical system to control or to approximate is needed. Table 2.1 summarizes the taxonomy of the model-free approximators [Kosk92].

The key concept that makes the numerical processing of the structured knowledge possible in Fuzzy Systems is the Fuzzy Set Theory. This chapter presents a general overview of the basic concepts of Fuzzy Logic and Fuzzy Systems. The fundamental topics of Fuzzy Set Theory are first introduced: *fuzzy sets, fuzzy operators, linguistic variables, fuzzy propositions, linguistic rules* and *approximate reasoning*. Finally, the structure of a Fuzzy Controller supporting different kinds of Fuzzy reasoning algorithms is described together with the general guidelines for designing Fuzzy Controllers.

Table 2.1. Taxonomy of model-free estimators. User need not state how system outputs explicitly depend on inputs. "Neural Networks and Fuzzy Systems by Kosko, © 1992, reprinted by permission of Pearson Education, Inc., Upper Saddle River, NJ.".

		FRAMEWORK	
		SYMBOLIC	NUMERICAL
KNOWLEDGE	STRUCTURED	AI EXPERT SYSTEMS	FUZZY SYSTEMS
	UNSTRUCTURED	----------	NEURAL SYSTEMS

2.1 From Sets to Fuzzy Sets

A fundamental concept in mathematics is the notion of set. A set is a collection of specific, discernible objects. A set can be finite, countable or uncountable and it can be described in three ways:

- By naming all its elements:

$$A = \{a, b, c, 10, \pi\} \Rightarrow \quad a, b, \ldots \pi \in A; \text{ whereas } f, \beta \notin A. \tag{2.1}$$

- By stating a property for all its elements:

$$A = \{x \mid P(x)\}, \text{ where } P \text{ defines the properties of the elements } x \text{ belonging to } A \text{ (i.e. } A = \{x \mid x>3\}). \tag{2.2}$$

- By defining a *Characteristic Function* for all the elements x of the Universe of Discourse U, the latter being a set that contains all objects related to a given context. The set U is also called the Super Set, the

Universal Set or the Referential Set. Therefore, for a set A belonging to U, the *Characteristic Function* $\mu_A: U \rightarrow [0,1]$ is defined as:

$$\mu_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A. \end{cases} \tag{2.3}$$

This function is called *membership function* in Fuzzy Set Theory. For instance, in the latter mentioned example, assume that U is the set of all positive integers denoted as E^+. Thus, the set of elements $A=\{x \mid x>3\}$, is associated to the characteristic function shown in Figure 2.1.

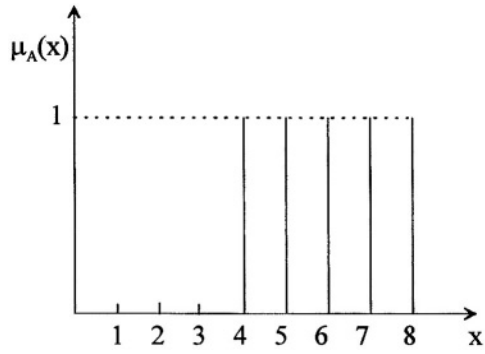


Figure 2.1. Characteristic Function for the set $A=\{x \mid x \in E^+ \text{ and } x>3\}$.

The above defined classical sets are commonly called *crisp* set to differentiate them from *fuzzy* sets. Let us summarize the main operations that can be performed on *crisp* sets. Given two sets A and B belonging to the same Universe of Discourse U:

- **Complement:** $\bar{A} = \{x \mid x \notin A\}$, $\mu_{\bar{A}}(x) = 1-\mu_A(x)$.
- **Intersection:** $A \cap B = \{x \mid x \in A \ \& \ x \in B\}$, $\mu_{A \cap B}(x) = \min(\mu_A(x),\mu_B(x))$.
- **Union:** $A \cup B = \{x \mid x \in A \text{ or } x \in B\}$, $\mu_{A \cup B}(x) = \max(\mu_A(x),\mu_B(x))$.

The symbols "$\wedge$" and "$\vee$" will also be used along this work to denote intersection and union respectively. The main properties of the above operations on Crisp Sets are: Commutative, Associative and Distributive. De Morgan Laws are sustained.

2.1.1 Fuzzy Sets

The fundamental concept in Fuzzy Set Theory is the idea of fuzzy set: *"a fuzzy set is a class of objects with a continuum of grades of membership.*

Such a set is characterized by a membership (characteristic) function which assigns to each object a grade of membership ranging between zero and one" [Zade65]. A fuzzy set F is entirely defined by the set of ordered pairs:

$$F = \{(x,\mu_F(x)) \mid x \in U\} \quad \text{and} \quad \mu_F: U \rightarrow [0, 1], \tag{2.4}$$

where x is an element of the universe of discourse U and μ_F is a membership function that assigns a degree of membership $\mu_F(x)$ to each element x of F. The membership function μ_F of a fuzzy set F corresponds to the characteristic function of a crisp set. However, while the characteristic function of a crisp set can only take values 0 or 1, the membership function of a fuzzy set can take any value from the interval [0,1]. Thus, a fuzzy set is a generalization of the crisp set.

2.1.1.1 Representations of Fuzzy Sets.

Similar to crisp sets, fuzzy sets can also be defined by extension. For instance, given the Universe of Discourse U={a, b, c, 10, π} and the fuzzy set F of U defined as:

$$F = \{(a,0.32), (b,0.15), (c,1), (10,0.8), (\pi,0)\}, \tag{2.5}$$

one must understand that "a" belongs to F with a "degree" of 0.32, "b" with 0.15, etc.

When the Universal Set U is continuous or uncountable rather than discrete, membership functions can be expressed analytically by using continuous or piece-wise explicit mathematical functions. Figure 2.2 depicts some commonly used membership function shapes.

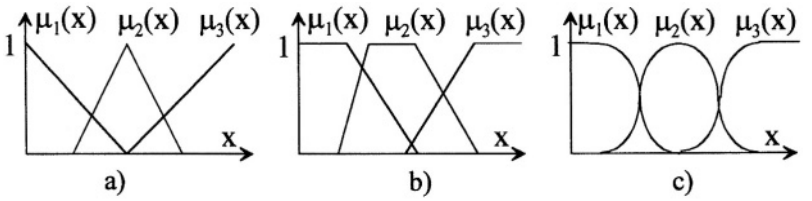


Figure 2.2. Examples of Fuzzy Membership Functions (FMF): a) Triangular. b) Trapezoidal c) Bell-shaped or gaussian-shaped.

2.1.1.2 Operation on Fuzzy Sets.

The operations on fuzzy sets are defined by means of their membership functions so that the definitions of those operations generalize their equivalents in crisp set theory. This is to say, if the membership degrees

were constrained to adopt only the values 0 and 1, the fuzzy operators would give identical results as the crisp ones. Most important operations on fuzzy sets are defined in the following.

Complement: given the fuzzy set A defined in the Universe of Discourse U, its complement is the fuzzy set:

$$\bar{A} = \{(x,\mu_{\bar{A}}(x)) \mid x \in U\} \quad \text{and} \quad \mu_{\bar{A}}(x) = \sim\neg(\mu_A(x)) = 1-\mu_A(x), \qquad (2.6)$$

where "$\sim\neg$" denotes the "fuzzy negation" of a membership function, coincident in this case with the complement operator for crisp sets. There are others complement operators "$\sim\neg$". In general, they are called C-Norms.

Intersection: given the fuzzy sets A and B, defined in U with membership functions μ_A and μ_B, their intersection is the fuzzy set:

$$A\cap B = \{(x,\mu_{A\cap B}(x)) \mid x \in U\}, \text{ with } \mu_{A\cap B}(x) = \mu_A(x) \sim\wedge \mu_B(x), \quad (2.7)$$

being "$\sim\wedge$"the symbol to denote the fuzzy intersection or "fuzzy and".

Intersection operators belong to the class of the so-called Triangular Norms or T-Norms. These are binary operators defined in the interval [0,1] satisfying the following properties:

a) $T(0,1) = T(1,0) = T(0,0) = 0$, $T(1,1) = 1$; (agreement with boolean AND).
b) $T(x,1) = x$, $\forall\ x\in[0,1]$; (identity).
c) $T(x,y) = T(y,x)$, $\forall\ x,y\in[0,1]$; (commutativity).
d) $T(x1,y) \le T(x2,y)$, $\forall\ x1,x2,y\in[0,1]$, $x1 \le x2$; (monotonicity).

Several operators have been proposed as T-Norms. We will mention only the most used ones:

- *min:* $\mu_{A\cap B}(x) = \mu_A(x)\sim\wedge\mu_B(x) = \min(\mu_A(x),\mu_B(x))$;
- *algebraic product*: $\mu_{A\cap B}(x) = \mu_A(x)\sim\wedge\mu_B(x) = \mu_A(x)\,\mu_B(x)$;
- *bounded difference*: $\mu_{A\cap B}(x) = \mu_A(x)\sim\wedge\mu_B(x) = \max(0,\mu_A(x)+\mu_B(x)-1)$.

Union: given the fuzzy sets A and B, defined in U with membership functions μ_A and μ_B, their union is the fuzzy set:

$$A\cup B = \{(x,\mu_{A\cup B}(x)) \mid x \in U\} \quad \text{and} \quad \mu_{A\cup B}(x) = \mu_A(x) \sim\vee \mu_B(x), \quad (2.8)$$

being "$\sim\vee$"the symbol to denote the fuzzy union or "fuzzy or".

Union operators belong to the class of the so-called S-Norms or Triangular CoNorms (T-CoNorms). These are binary operators defined in the interval [0,1] satisfying the following properties:

a) $S(0,1) = S(1,0) = S(1,1) = 1$, $S(0,0) = 0$; (agreement with boolean OR).
b) $S(x,0) = x$, $\forall\ x \in [0,1]$; (identity).
c) $S(x,y) = S(y,x)$, $\forall\ x,y \in [0,1]$; (commutativity).
d) $S(x1,y) \leq S(x2,y)$, $\forall\ x1,x2,y \in [0,1]$, $x1 \leq x2$; (monotonicity).

Several operators have been proposed as T-CoNorms. The most used are ones:

- *max:* $\mu_{A\cup B}(x) = \mu_A(x) \sim\vee \mu_B(x) = \max(\mu_A(x), \mu_B(x))$;
- *algebraic sum*: $\mu_{A\cup B}(x) = \mu_A(x) \sim\vee \mu_B(x) = \mu_A(x) + \mu_B(x) - \mu_A(x)\,\mu_B(x)$;
- *bounded sum*: $\mu_{A\cup B}(x) = \mu_A(x) \sim\vee \mu_B(x) = \min(1, \mu_A(x) + \mu_B(x))$.

Figure 2.3 shows a graphical representation of these three basic operations. The main properties of the above defined operations on fuzzy sets are: Commutative, Associative and Distributive. In addition, De Morgan Laws are sustained.

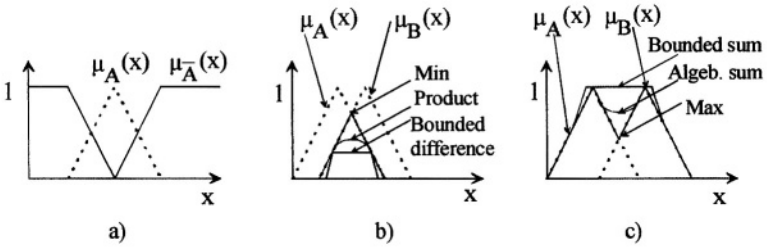


Figure 2.3. Geometrical interpretation of the operations on fuzzy sets: a) Complement. b) Intersection. c) Union. Original membership functions are in dashed line. "From "Neuro-Fuzzy Controllers Design and Application", J. Godjevac, Presses Polytechniques et Universitaires Romandes, Lausanne, Switzerland, 1997, reprinted with permission".

2.1.1.3 A Clarifying Example.

Fuzzy sets intend to model the idea of uncertainty or vagueness associated to the natural human reasoning, which is based on linguistic words and sentences rather than in mathematical expressions and relations. For instance, assume two complementary crisps sets, A and B, defined in the Universe of Discourse U={"the population of Louvain-la-Neuve"}:

A = {"people of L-L-N younger than 25 years old"},
B = {"people of L-L-N older than 25 years old"}. (2.9)

Figure 2.4 a) shows the characteristic functions for the two sets above. A "fuzzy" definition of those previous sets could be:

AF = {"the young people of L-L-N"},
BF = {"the adult people of L-L-N"}. (2.10)

Figure 2.4 b) shows the membership functions for AF and BF together with their intersection that has been performed by using the "min" operator as T-Norm. Notice that the intersection between the crisp sets A and B is the null set (i.e. {Φ}). In contrast, the intersection between AF and BF is another, not null, fuzzy set that can be interpreted as: *"the people of L-L-N that are neither young nor adult"*. This is a remarkable characteristic of fuzzy sets that strengthens the idea of "imprecision": *"the intersection between any fuzzy set and its complement is not null"*

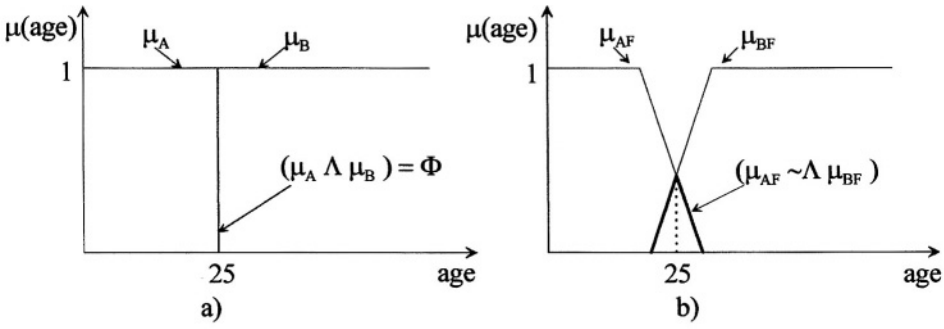


Figure 2.4. a) Characteristic Functions for the complementary crisp sets A and B. b) Membership functions for the complementary fuzzy sets AF and BF.

2.2 Approximate Reasoning

Approximate reasoning or fuzzy reasoning is a mode of reasoning which is neither exact nor inexact [Zade75]. It is supported on the Fuzzy Logic theory and it offers a realistic framework for representing human reasoning. Approximate reasoning is the fundament for Fuzzy Inference Systems. To fully understand how it is performed three basic concepts should be clarified:

- Linguistic Variable.
- Fuzzy Proposition.
- Linguistic Rule.

A rigorous mathematical explanation of the above concepts can be found in [Godj97]. Based on the latter, we present in the following a brief conceptual interpretation.

2.2.1 Linguistic Variable

A linguistic variable is a variable whose values are words or sentences in a natural or artificial language rather than numerical [Zade75]. According to [Godj97] a linguistic variable is defined by:

- Its name: "x".
- Its term set: "TS(x)", which is the set of linguistic values or *labels* of "x".
- The base variable "u", which supports the linguistic values of "x". In others words, the membership functions for the linguistic values of "x" are defined in the domain of "u".
- The universe of discourse "U" associated with the base variable "u".

"x" should not be confused with "u": "x" is the name of a linguistic variable (i.e. *distance, angle,* etc) whereas "u" is the name of the base variable giving physical sense to "x" (i.e. meters, degrees, etc). In physical applications, "x" may adopt linguistic values (i.e. *small, very large,* etc) whereas "u" may adopt numerical values (i.e. 100m, -30°, etc). However, it is very common in the literature to use the same nomenclature to make reference either to the linguistic variable or its base variable. In some part of this text, we will do it.

In a Fuzzy Controller, inputs and outputs are defined as linguistic variables. For instance, suppose that we design a controller to guide a vehicle towards a reference point in a plane. The inputs of the controller are the *distance* between the reference point and the vehicle and the *speed* of the vehicle. The output of the controller is the *power* supplied to the vehicle's engine. The term set of the linguistic variable *distance,* for instance, can be defined, as:

$$TS(distance) = \{small, medium, big\} = \{S, M, B\}. \quad (2.11)$$

Since the distance is a length, the base variable "u" associated with the linguistic variable *distance* may adopt values expressed in terms of length units. Assuming that the length measurement ranges from 0 to 1000 cm, the Universe of Discourse is U=[0,1000].

A linguistic value belonging to TS(*distance)* makes physical sense through the definition of its membership function that confines its domain in terms of the variable "u". This restriction is the "meaning" of such a

linguistic value. As an example, let us suppose that the "meanings" of the labels *small, medium* and *big* are restricted by the following bell-shaped membership functions defined in the domain of u:

$$\mu_S(u) = \frac{1}{1+\left(\frac{u}{130}\right)^2}, \mu_M(u) = \frac{1}{1+\left(\frac{u-500}{200}\right)^2}, \mu_B(u) = \frac{1}{1+\left(\frac{u-1000}{130}\right)^2} \tag{2.12}$$

By using linguistic modifiers (i.e. *very, more, close to, a sort of,* etc) and linguistic connectives (i.e. *and, or, not,* etc), the original term set TS(*distance*) can be extended to support more linguistic values such as: *very small, close to medium,* etc. One can estimate the membership function for two new linguistic values *very small* and *very big,* for instance, by halving the scaling parameter of the argument "u" in the expression of the membership functions of the labels *small* and *big,* respectively. Thus:

$$\mu_{VS}(u) = \mu_S(2u), \qquad \mu_{VB}(u) = \mu_B(2(u-1000)). \tag{2.13}$$

Any mathematical transformation being capable to produce shape compression could be used to perform the linguistic modifier "*very*". However, the choices of the distribution, the shape and the modifier operators of the membership functions depends on the application and is a matter of optimization. Figure 2.5 shows the membership functions of the extended term set TS(*distance*).

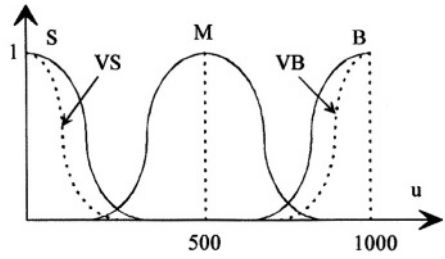


Figure 2.5. Membership functions for the labels *very small* (VS), *small* (S), *medium* (M), *big* (B) and *very big* (VB) of the linguistic variable *distance*. "From "Neuro-Fuzzy Controllers Design and Application", J. Godjevac, Presses Polytechniques et Universitaires Romandes, Lausanne, Switzerland, 1997, reprinted with permission".

2.2.2 Fuzzy Proposition

A fuzzy proposition is a statement expressed in a natural or artificial language. In contrast to classical logic propositions, a fuzzy proposition may adopt a truth-value from the interval [0,1] [Godj97]. For the former example on vehicle guidance, the following sentences are fuzzy propositions:

distance is *very small*, *distance* is *big*, (2.14)

where the meanings of these propositions are determined by the corresponding membership functions defined in Figure 2.5.

Fuzzy Controllers normally deal with several input variables defined in different Universe of Discourses. Therefore, the compound fuzzy propositions that are formed using linguistic connectives as *and, or, not,* etc, are more frequently encountered. For instance, in the former example on vehicle guidance compound fuzzy propositions should look like:

distance is *small* and *speed* is *fast*, *distance* is *big* or *speed* is *low*.(2.15)

2.2.3 Linguistic *If-Then* Rule: Syntax, Meaning and Evaluation

According to [Godj97], a linguistic *if-then* rule has two parts:

- *antecedent* part (premise), expressed by: *if* <fuzzy proposition>,
- *consequent* part, expressed by: *then* <fuzzy proposition>,

where the fuzzy propositions at the antecedent and consequent parts may be simple or compound. In a Fuzzy Controller, the antecedent part is related to the inputs of the controller whereas the consequent part is related to the outputs.

Let us reconsider the former example on vehicle guidance. Suppose that in the corresponding controller there is a fuzzy rule defined as:

R: *If distance* is S (*small*) *then power* is L (*low*). (2.16)

This rule relates the input variable *distance* to the output variable *power*. This is defined in the domain Y=[0,1] of the base variable "y". Its term set is:

TS(*power*) = {*low*, *normal*, *high*} = {L, N, H}. (2.17)

What is the "meaning" of the rule R provided that the actual input *distance* is a *crisp* value "do"? That is, if *distance* were equal to the fuzzy set

S, the rule says that *power* must be equal to the fuzzy set L. However, to what extend does the actual "inferred" output of the rule differ from L when the actual input differs from S? To solve this problem, the "meaning" of the rule is evaluated by using the so-called *fuzzy implication* (*fuzzy inference*), which can be stated as:

- Premise (actual input): *distance* is "do".
- Rule: *If distance* is S (*small*) *then power* is L (*low*).
- Consequence (inferred output): *power* is L*;

where L* is the modified *consequent* fuzzy set different from the original L. From the theory of Fuzzy Logic it can be formally demonstrated that L* can be calculated using the following two steps:

1. *Fuzzification*: determines the degree of membership to the fuzzy set S of the crisp input "do". It simply means calculate $\mu_S(u)$ for u=do. In this particular case, since there is a simple fuzzy proposition at the antecedent part of the rule, this value also represents the so-called *firing degree* or *firing strength* of the rule:

$$\textit{firing degree}(R) = \mu_S(do). \tag{2.18}$$

2. *Consequent Modulation*: according to the obtained value $\mu_S(do)$ in the previous step calculate the modified output fuzzy set L* by using a *modulation* operator. Several *modulation* operators have been reported in the literature. The two most used ones are the "min" (*clipping*) and the "scalar product" (*scaling*). Thus, the membership function of L* may be given by:

$$\begin{aligned} &\mu_{L*}(y) = \min(\mu_S(do), \mu_L(y)) && (\textit{clipping}), \text{ or} \\ &\mu_{L*}(y) = \mu_S(do)\, \mu_L(y) && (\textit{scaling}). \end{aligned} \tag{2.19}$$

The evaluation of the rule R is graphically represented in Figure 2.6 where the "min" clipping operator was used. It can be noticed that the output "inferred" by the rule (i.e. L*) is still a fuzzy set and cannot be directly used in the real world (i.e. to steer the vehicle). Therefore, it must be transformed into a crisp value by means of the so-called *defuzzification* operation. This will be presented later.

In the more general case of n-input, m-output Fuzzy Controllers (MIMO controllers), a fuzzy rule looks like:

"*If* x1 is A1 and x2 is A2 and.....xn is An
then y1 is B1, y2 is B2,.....ym is Bm". (2.20)

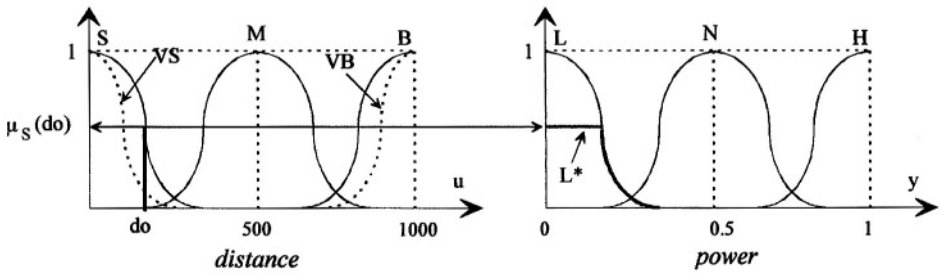


Figure 2.6. Fuzzy Inference: the "meaning" of the rule R using the clipping operator "min".

The "and" connective operators in the antecedent part of the above rule must be interpreted as the *intersection* operations. Hence, any T-Norm operator can be used to evaluate this compound fuzzy proposition. In this case using also the "min" operator as T-Norm, given the *crisp* input vector (x1o,x2o...xno), the *firing degree* of the rule is given by:

$$\mu_{A1\cap A2\cap\ldots An}(x1o,x2o\ldots xno) = \min(\mu_{A1}(x1o),\mu_{A2}(x2o)\ldots\mu_{An}(xno)). \quad (2.21)$$

The modified fuzzy set "inferred" at each output yi is calculated as in (2.19). Thus, using the *min clipping* operator the membership function for each modified output fuzzy set Bi* is given by:

$$\mu_{Bi*}(yi) = \min(\mu_{A1\cap A2\cap\ldots An}(x1o,x2o\ldots xno),\ \mu_{Bi}(yi)),\ \text{for } i = 1\ldots m. \quad (2.22)$$

For the sake of the demonstration let us include a second output variable in our example on vehicle guidance: the *steering angle* of the wheel. Let us consider the following rule in the controller:

R1: "*If distance* is *very small* (VS) and *speed* is *fast* (F)
then steering angle is *positive big* (PB); *power* is *low* (L)". (2.23)

Figure 2.7 illustrates the graphical representation of this rule together with the modified output fuzzy sets PB* and L* obtained after evaluating the rule for the actual crisp input values "do" and "so". Therefore, accordingly with (2.21), the *firing degree* of Rl is given by:

$$\textit{firing degree}\ (R1) = \min(\mu_{VS}(do),\mu_F(so)) = \mu_F(so). \qquad (2.24)$$

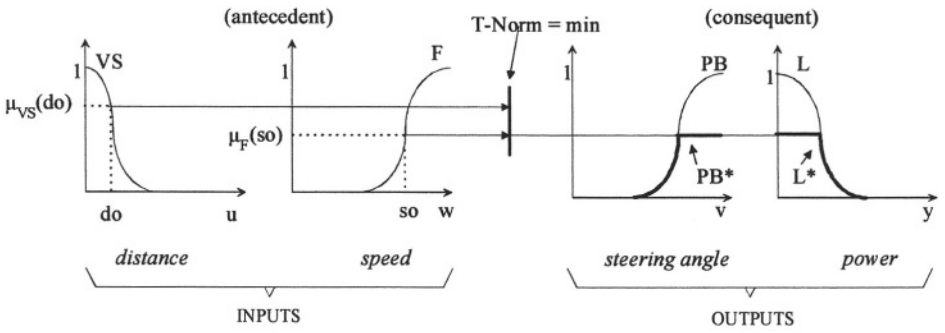


Figure 2.7. Evaluation of the rule R1.

2.2.3.1 Complementary Fuzzy Membership Function.

Frequently in Fuzzy Controllers, instead of working with the actual Fuzzy Membership Functions defined for the input variables in the antecedent parts of the rules, it is more practical to use the complemented versions of the membership functions. In this text we call the latter as Complementary Fuzzy Membership Function (CFMF). They are obtained by applying the complement operator defined in (2.6) to each predefined Fuzzy Membership Function (FMF).

However, to fully sustain a mathematical equivalence with (2.21) De Morgan's law must be applied. For this purpose, a T-CoNorm (S-Norm) followed by complementation must be used rather than a direct T-Norm. In this way, using the "max" operator as T-CoNorm and the CFMFs, equation (2.21) can be reformulated as:

$$\begin{aligned}\textit{Firing degree} &= \min(\mu_{A1}(x1o),\mu_{A2}(x2o)\ldots\mu_{An}(xno)) \\ &= 1 - \max(\mu_{\bar{A}1}(x1o),\mu_{\bar{A}2}(x2o)\ldots\mu_{\bar{A}n}(xno)),\end{aligned} \qquad (2.25)$$

where $\mu_{\bar{A}i}=(1-\mu_{Ai})$ is the complement of μ_{Ai}, for i=1…n. The application of (2.25) to the former example is graphically explained in Figure 2.8.

The main reason to use (2.25) instead of (2.21) resides on the fact that "max" operators are easier implemented than "min" operators with analog current-mode hardware.

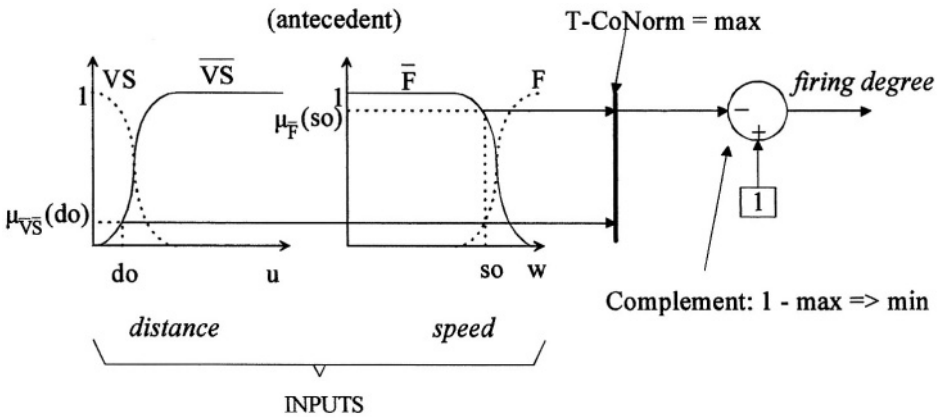


Figure 2.8. Evaluation of the firing degree of the rule R1 using Complementary Fuzzy Membership Functions (CFMF) and a T-CoNorm "max" operator. Actual Fuzzy Membership Functions are in dashed.

2.2.3.2 Takagi-Sugeno's *If-Then* Rules.

To conclude this section dedicated to Fuzzy Rules, let us introduce the Takagi-Sugeno's *if-then* rules model [SuYa93]. Because of its simplicity, it is the most used fuzzy reasoning method. It can be easily implemented using analog or digital hardware. The antecedent part of this kind of rules has fuzzy sets as in the classical Fuzzy Controllers discussed above. In contrast, the consequent part is described by a linear combination of the actual, non fuzzy, inputs of the controller. Assuming for clarity that the controller has only one output, the general form of Takagi-Sugeno's rule is:

$$\text{"}If\ x1 \text{ is } A1 \text{ and } \ldots xn \text{ is } An\ then\ y = ao + a1\ x1 + \ldots + an\ xn\text{"}, \quad (2.26)$$

where coefficient ao...an are *crisp* real constant values. Notice in this case at the consequent part of the rule that the terms x1...xn refer to the numerical base variables rather to the equally named linguistic variables. The controller defined in terms of Takagi-Sugeno's rules performs approximations by interpolating the hyperplanes represented by the linear expressions at the consequent parts of each rule.

When, with the exception of ao, all coefficients in the linear output expressions are identically zero, the controller is called "Zero-Order" or "Singleton" Takagi-Sugeno's controller. Otherwise, it is called "First-Order" Takagi-Sugeno's controller.

2.2.4 Set of Linguistic Rules, Fuzzy Reasoning Methods and Defuzzification

The design of a Fuzzy Controller involves the design of a set of linguistic rules that are evaluated concurrently [Godj97]. Consider the following set of linguistic rules for a 2-input, 1-output, n-rule controller:

R1: *If* x is A1 and y is B1 *then* z is C1 or
R2: *If* x is A2 and y is B2 *then* z is C2 or
.....
..... or
Rn: *If* x is An and y is Bn *then* z is Cn. (2.27)

If a crisp input (xo,yo) is presented at the inputs (x,y), each rule infers a clipped fuzzy set (i.e. Ck*) as shown in the previous sections. On the other hand, since the rules are connected by "or" operators it turns out that the final output inferred by the whole set of rules will be the union of the individual clipped fuzzy set inferred by each rule. Thus:

$$C = \bigcup_{k=1}^{n}(Ck^*). \tag{2.28}$$

The latter operation is also called *aggregation.* In the latter expression, C is the *aggregated* fuzzy set at the output of the controller whereas Ck* is the clipped output fuzzy set at the k-th rule. Using the "max" operator as S-Norm (or T-CoNorm) for the connectives "or" and the *clipping operator* "min", the membership function of C can be calculated as:

$$\mu_C(x,y,z) = \max_k(\min(\min(\mu_{Ak}(x),\mu_{Bk}(y)),\mu_{Ck}(z))). \tag{2.29}$$

For a given crisp 2-input value (x,y)=(xo,yo), the final fuzzy set inferred by the whole set of rules is the *aggregation* (union) of all clipped fuzzy set of each rule. Therefore, the membership function of this aggregated output fuzzy set is given by:

$$\mu_C(xo,yo,z) = \max_k(\min(\min(\mu_{Ak}(xo),\mu_{Bk}(yo)),\mu_{Ck}(z))). \tag{2.30}$$

The above detailed procedure corresponds to the Mamdani's fuzzy reasoning method. In the Takagi-Sugeno fuzzy reasoning algorithm, each linear expression at the output of each rule is multiplied by the

corresponding *firing degree* of each rule. Then, the resulting *weighted* linear expressions are *aggregated* by means of the *algebraic sum.*

Figure 2.9 gives a graphical insight of both Mamdani and Takagi-Sugeno algorithms. The antecedent part of the rules is identical for both cases. The "min" operator is used as T-Norm for the evaluation of the *firing degree* of the rules.

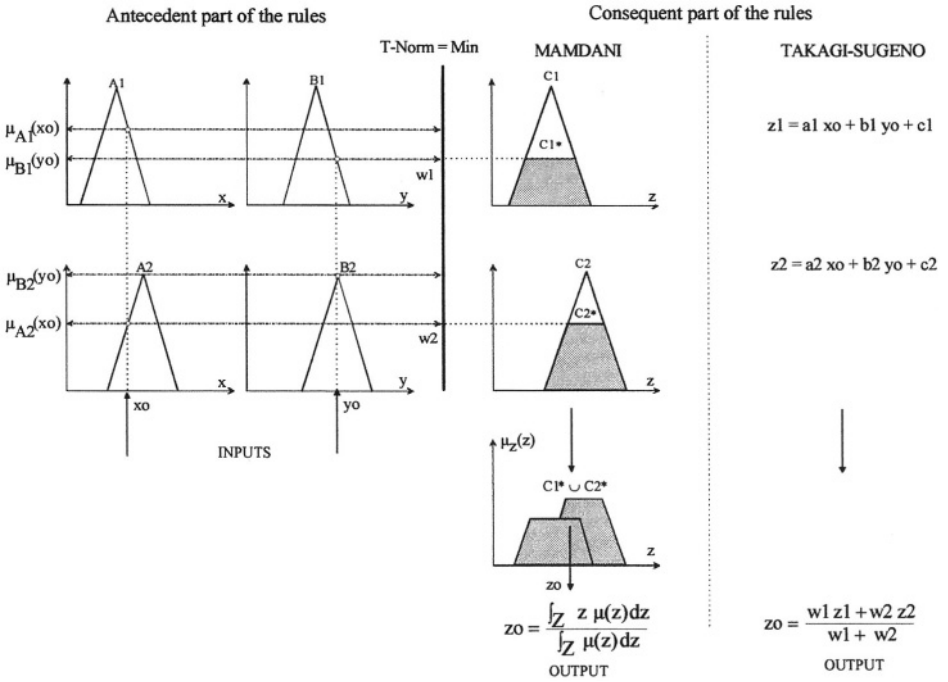


Figure 2.9. Mamdani and Takagi-Sugeno fuzzy reasoning methods. "From "Neuro-Fuzzy Controllers Design and Application", J. Godjevac, Presses Polytechniques et Universitaires Romandes, Lausanne, Switzerland, 1997, reprinted with permission".

Given the crisp inputs "xo" and "yo", after calculating the firing degrees of each rule as explained in previous section (i.e. $w1=\min(\mu_{A1}(xo),\mu_{B1}(yo))$ and $w2=\min(\mu_{A2}(xo),\mu_{B2}(yo))$), the *fuzzy inference* process is completed as follows:

1. For Mamdani's controllers, all clipped fuzzy sets at the consequent part of the rules (i.e. C1* and C2*) are aggregated by using the "max" (union) operator to generate the final *aggregated* output fuzzy set $C1^* \cup C2^*$. For Takagi-Sugeno's controllers, all weighted linear expressions at the consequent part of the rules (i.e. w1z1 and w2z2) are simply summed yielding the final expression called *weighted sum* (i.e. w1z1+w2z2).

2. The last step is the so-called *defuzzification*. It consists in generating a *crisp* output value in the domain of the output variable z that best "represents" the *aggregated* output fuzzy set in Mamdani's controllers, or the *weighted sum* in Takagi-Sugeno's controllers. For Mamdani's controllers, the crisp *defuzzified* output "zo" is calculated as the Center of Gravity (COG) of the output fuzzy set $C1^* \cup C2^*$ by using the expression shown in the Figure 2.9. For Takagi-Sugeno's controllers, the crisp *defuzzified* output "zo" is calculated as the Averaged Weighted Sum (AWS) or Normalized Weighted Sum by dividing the *weighted sum* calculated in step 1 by the sum of the weights w1, w2, as shown in the same figure.

Obviously, considering the *aggregation* and *defuzzification* processes in both kind of reasoning, Mamdani's controller demands more computational efforts than Takagi-Sugeno's controller does. This is the main reason why Takagi-Sugeno's controllers are preferred for hardware implementation. Table 2.2 summarizes the main advantages of both methods [JaGu95].

Table 2.2. Advantages of Mamdani and Takagi-Sugeno fuzzy reasoning methods [JaGu95].

Mamdani	Takagi-Sugeno
- More intuitive	- Computational efficiency
- Widespread acceptance	- Works well with linear techniques (i.e. PID control)
- Better suited to human input	- Works well with optimization and adaptive techniques
	- Guaranteed continuity of the output surface
	- Better suited to mathematical analysis

2.2.4.1 A Numerical Example.

Following with the example on vehicle guidance let us evaluate a fuzzy rule set intended to control the *power* supplied to the vehicle as a function of the actual *speed* of the vehicle and the *distance* to the reference point. The input variables are: *distance* associated with the base variable $u \in U=[0,1000]$ and *speed* associated with $w \in W=[0,100]$. The output variable is *power* associated with $y \in Y=[0,1]$. Both fuzzy reasoning methods, Mamdani and Takagi-Sugeno, will be considered.

For the Mamdani algorithm, the term set TS for each input/output variable comprises three fuzzy sets represented by triangular membership functions equally distributed along the domain of the corresponding base variable. Therefore, the linguistic values that each variable may adopt are:

$$\begin{aligned} &TS(\textit{distance}) = \{\textit{small, medium, big}\} = \{S, M, B\}, \\ &TS(\textit{speed}) = \{\textit{zero, slow, fast}\} = \{Z, S, F\}, \\ &TS(\textit{power}) = \{\textit{low, normal, high}\} = \{L, N, H\}. \end{aligned} \tag{2.31}$$

For the Takagi-Sugeno algorithm, only the term set for the output variable *power* is different. Assuming a Zero-Order controller, the consequent singletons are set to a constant value representing numerically the concept of *low, normal, high,* such as: L=0; N=0.5 and H=1. For both algorithms, the set of rules is detailed in Table 2.3.

Table 2.3. Rule set describing the fuzzy relationship *power* = f(*distance*, *speed*) for Mamdani and Zero-Order Takagi-Sugeno algorithms.

Rules	Mamdani	Takagi-Sugeno	Rules connective
R1: *if distance* is S and *speed* is Z *then power* is	N	0.5	or
R2: *if distance* is S and *speed* is S *then power* is	N	0.5	or
R3: *if distance* is S and *speed* is F *then power* is	L	0	or
R4: *if distance* is M and *speed* is Z *then power* is	H	1	or
R5: *if distance* is M and *speed* is S *then power* is	N	0.5	or
R6: *if distance* is M and *speed* is F *then power* is	N	0.5	or
R7: *if distance* is B and *speed* is Z *then power* is	H	1	or
R8: *if distance* is B and *speed* is S *then power* is	H	1	or
R9: *if distance* is B and *speed* is F *then power* is	H	1	

When the input space is bi-dimensional, another way for representing the rules set is through the so-called *rules map,* as illustrated in Figure 2.10 a). In the latter, the membership functions of the input variables are drawn along the corresponding base variable axis. Each rule (or *fuzzy cluster*) is defined by a pair of membership functions, each one belonging to one input variable. This compact representation gives a graphical idea about the partition of the input Universe of Discourse performed by the *fuzzification* of the input variables. In this case, all possible combination between the *labels* (or *fuzzifiers*) defined for the input variables have been considered and the partition is called *grid partition.* This allows sharing the membership functions by different rules, but the total number of rules needed to cover the input domain grows geometrically with the number of inputs of the controller. There are other kinds of input partitions, *tree* and *scatter partitions,* which convey to decrease substantially the number of rules [JaSu95]. In these cases, the membership functions for each input are defined after identifying the *clusters* in the output surface that can be represented by each rule. Nevertheless, sharing membership functions may become impossible and the total number of fuzzifiers may increase considerably when using the latter kinds of partitions [Vida96].

After performing step by step the reasoning algorithms explained in Figure 2.9 for all pairs (u,w) belonging to the input space a non-linear relationship y=g(u,w) is established. The results are plotted in Figure 2.11 a) and b) for each case. It can be noticed that these surfaces, also called *output control surfaces*, differ in their values at each point of the input space, but

their gaits resemble in such a way that they convey to similar control policies.

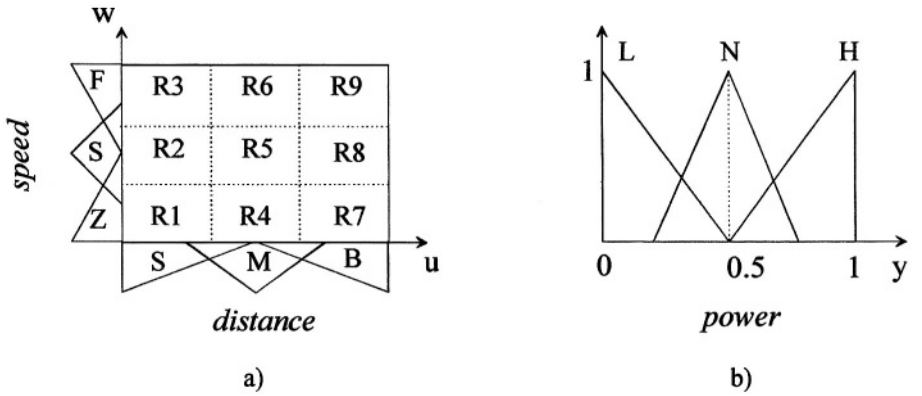


Figure 2.10. a) Fuzzy partition of the input space and rules map. b) Membership functions for the output variable *power* in the Mamdani algorithm.

First-Order Takagi-Sugeno algorithm could be also used to perform the above control task. However, the identification of the coefficients multiplying the input variables in the linear expression of the consequents of the rules is not immediate. It is difficult to establish a relation between a hyperplane (i.e. ao+a1u+a2w) and a linguistic value. Nevertheless, it has been shown that for the same output surface needed for a given control policy, First-Order controllers can realize better approximations, with smoother transitions, while demanding a much reduced rules set than the Zero-Order counterparts [GoAl01].

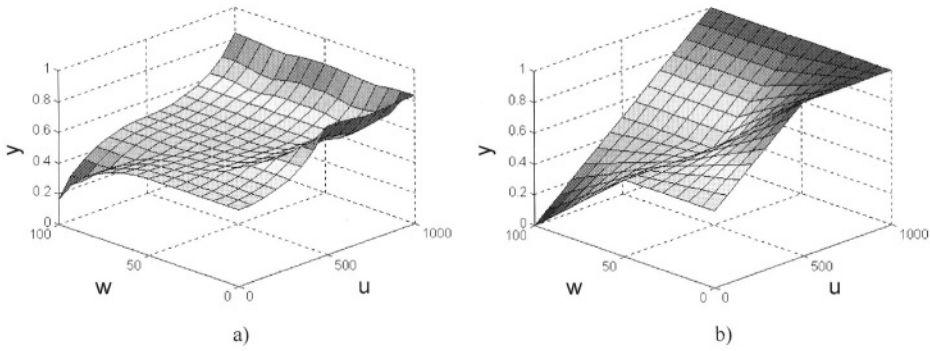


Figure 2.11. a) *Output control surface* following Mamdani algorithm. b) *Output control surface* following Takagi-Sugeno algorithm.

2.3 Design and Optimization Issues for Fuzzy Controllers

The design of a Fuzzy Controller involves the correct choice of: the linguistic variables (i.e. process states, input and output variables), the set of linguistic rules, the kind of fuzzy reasoning method and the defuzzification strategy. A design and optimization methodology suitable for Fuzzy Control is presented in [Godj97]. It is illustrated in Figure 2.12, which shows the block diagram of a Fuzzy Controller inserted in closed loop together with the Plant under control.

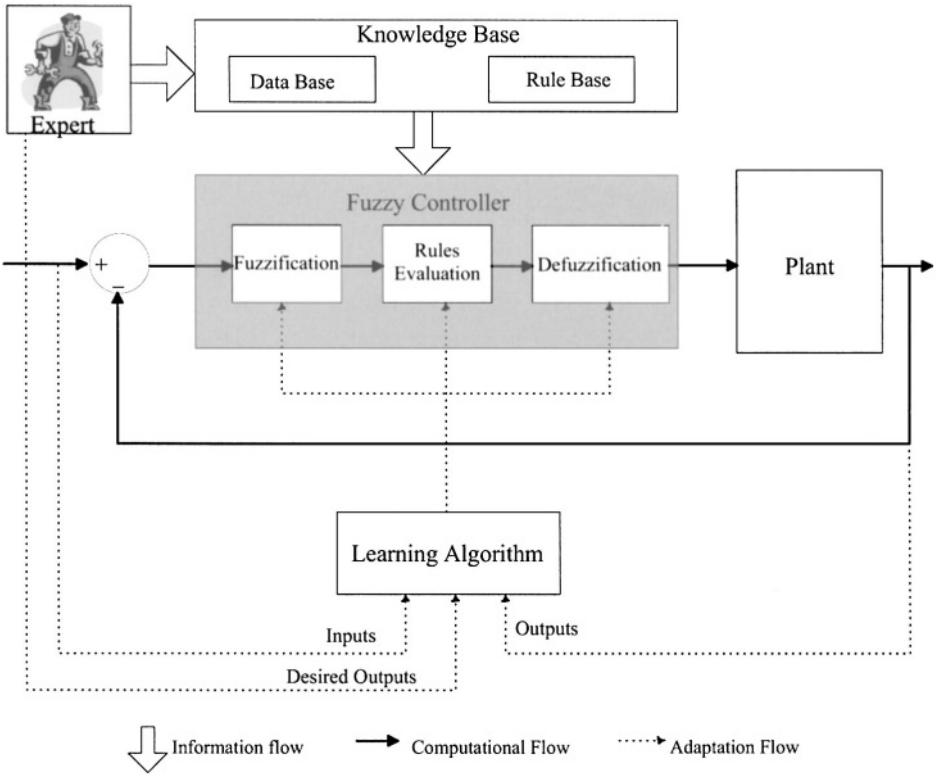


Figure 2.12. Block diagram for the design methodology for Fuzzy Controllers. "From "Neuro-Fuzzy Controllers Design and Application", J. Godjevac, Presses Polytechniques et Universitaires Romandes, Lausanne, Switzerland, 1997, reprinted with permission".

Physically, the controller itself corresponds to the shaded block, wherein the three basic fuzzy operations are highlighted, in agreement with the algorithms presented in Figure 2.9. The Knowledge Base provided by an expert is also represented in Figure 2.12. It comprises a Data Base and a Rule Base. The first provides the linguistic variables, their term sets, the

membership functions shape and position, the inference method and the defuzzification method. The Rule Base includes the source and the contents of the set of rules, which can be derived from modeling the human operator's actions or from the linguistic description of the controlled plant (i.e. a kind of "fuzzy inverse modeling"). Finally, a Learning or Tuning Algorithm adapts the parameters of the controller by minimizing, for instance, the RMSE (Root Mean Square Error) between the desired and actual outputs of the plant.

The design and optimization procedure can be summarized as follows:

1. The know-how coming from an expert is codified in a set of initial linguistic rules.
2. The rules are implemented by setting initial values of the controller's parameters (i.e. antecedent and consequent parameters).
3. The controlled plant is set operating in closed loop.
4. Following a gradient descent criterion (in order to find the minimal $\mathbf{RMSE=(\Sigma\ (actual\_output-desired\_output)^2)^{1/2}}$), the parameters of the controller are updated during plant operation as time goes on [RoMo51].
5. If the results are not satisfactory, steps 2 to 4 are repeated but changing some initial guesses, i.e., the number of rules, the number of fuzzifiers per input, the shape of the fuzzifiers, etc.
6. When the results are acceptable, more accurate linguistic rules than the ones initially assessed by the expert in step 1 can be extracted. This allows the expert to redefine and expand his knowledge about the system. In this way, steps 1 and 2 could be reformulated and a finer tuning cycle could be started.

2.3.1 Learning in Takagi-Sugeno's Controllers: ANFIS

Takagi-Sugeno fuzzy reasoning can be represented as a special kind of Neural Network called ANFIS (Adaptive Network-based Fuzzy Inference System) [Jang92]. A network structure facilitates the computation of the error gradient vector for adapting the parameters of the controller. Hence, it can be tuned by applying the gradient descent algorithm based on some collection of input-output data [Verl92].

Figure 2.13 shows an example of an ANFIS network for a 2-input, 1-output, 2-rule Zero-Order controller. We will summarize briefly the fuzzy operations performed at each layer:

– Layer 1: performs fuzzification. The outputs of the nodes $O1_{A1}...O1_{B2}$ at this layer are the degrees of membership of the inputs according to the defined input membership functions.

- Layer 2: performs the T-Norm operation evaluating the firing strength (firing degree) of each rule (i.e. w1 and w2).
- Layer 3: performs the normalization of the firing strengths of the rules. That means w1N=w1/(wl+w2) and w2N=w2/(wl+w2).
- Layer 4: Multiplies each normalized firing strength by the corresponding consequent singletons a01, a02, which have been defined for each consequent part of the rules.
- Layer 5: realizes the aggregation (sum) of the output "inferred" by each rule. It outputs the final crisp defuzzified value.

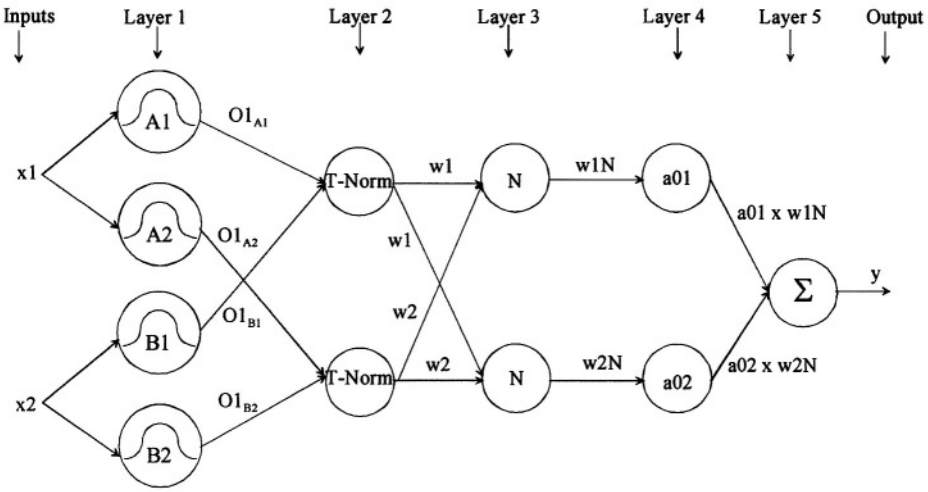


Figure 2.13. ANFIS network for a 2-input, 1-output and 2-rule Zero-Order (singleton) Takagi-Sugeno's Fuzzy Logic Controller. From [Jang92], © 1992 IEEE.

The design of controllers by using ANFIS in an iterative way is very easy. The adapting procedure [Godj97] can be summarized as follows:

1. Initialization of parameters:
   - Consequent parameters are randomly initialized.
   - Choice of antecedent parameters (membership function parameters).
2. Set the input vector and the desired output vector of the controller.
3. Computation of the actual outputs and learning errors.
4. Adaptation of the consequent parameters (i.e. a01, a02).
5. Adaptation of the antecedent parameters (i.e. A1, A2, B1, B2).
6. Evaluation of the learning RMSE.
7. Restart from step 2 until the above RMSE is smaller than a tolerable value, which is preset according to the application requirements.

If the attained performance is still poor, we can start again from step 1 but choosing now a new structure for the controller (i.e. different number of rules, different input partition, etc).

Finally, to illustrate the design philosophy using ANFIS we present in the following an example. The goal is to approximate the function:

$$Z = \frac{3}{4}\sin\left(\frac{5}{9}(X-1.5)(Y-1.5)\right) + \frac{5}{2}, \qquad (2.32)$$

in the domain: $X,Y \in [1.5, 4.5]$. The target function is drawn in Figure 2.14.

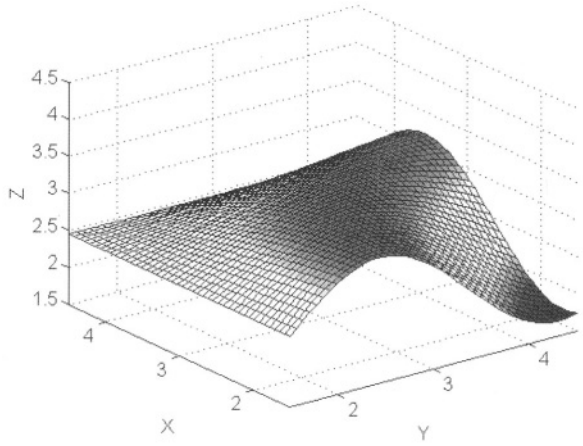


Figure 2.14. Target Function.

After a visual inspection of the function to approximate, we estimate in a first trial that the input space could be partitioned in nine clusters. So we initialize the controller with three membership functions per input equally distributed along the input domain of X and Y. We start the learning algorithm with all singletons at the consequents initialized at zero. After 60 epochs (i.e. epoch=learning period after a complete training data set is processed), the RMSE remains stable with a final value of 0.082. Figure 2.15 shows the membership functions of input X before and after learning, the evolution of the RMSE and the obtained surface after 140 epochs.

Not being satisfied with the results, we decide to repeat the former experience but increasing to four the number of FMF per input. Thus, a 16-rule Zero-Order controller is initialized in the same way as before and the experience was repeated once again. The results are shown in Figure 2.16. The final RMSE is halved (i.e. 0.042) with respect to the first experience. In this way, we could continue by modifying the structure of the controller in

terms of the number of rules and membership functions per input until reaching the desired accuracy in the approximation. Note the fast convergence (i.e. <140 epochs) of the method in both cases, making it very attractive for the iterative design of Takagi-Sugeno's Controllers.

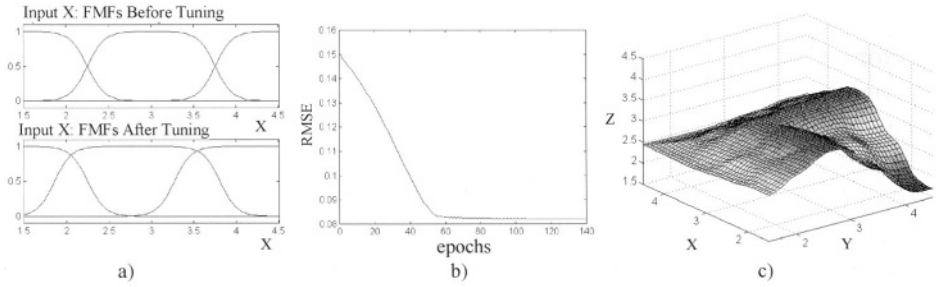


Figure 2.15. Approximation with a 9-rule, Zero-Order Takagi-Sugeno's controller. a) FMFs of input X before and after learning. b) RMSE during learning. c) Actual surface after 140 epochs. All consequent singletons were initialized to zero. After learning, the singletons adopt the following values: [a01…a09]=[2.43, 2.44, 2.6, 2.4, 3.12, 3.26, 2.6, 3.27, 1.7].

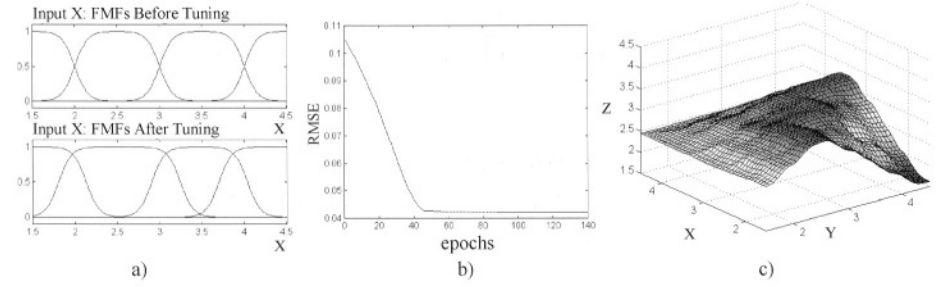


Figure 2.16. Approximation with a 16-rule, Zero-Order Takagi-Sugeno's controller. a) FMFs of input X before and after learning. b) RMSE during learning. c) Actual surface after 140 epochs. All consequent singletons were initialized to zero. After learning, the singletons adopt the following values: [a01…a016]=[2.43, 2.44, 2.45, 2.54, 2.44, 2.87, 3.2, 3.25, 2.45, 3.2, 3.3, 3.43, 2.54, 3.26, 2.42, 1.6].

2.4 Conclusions

In this chapter, the fundamental concepts of Fuzzy Logic, Approximate Reasoning and Fuzzy Controllers have been presented. The notions of Fuzzy Set, operations with Fuzzy Sets, T-Norm, T-CoNorm were first outlined. Approximate reasoning was explained through the concepts of Linguistic Variable, Fuzzy Proposition, Linguistic *If-Then* Rule and Fuzzy Inference. The meaning and the evaluation of a single rule and a set of rules were

focused. This lead to the description of the two most used Fuzzy Reasoning algorithms: Mamdani and Takagi-Sugeno.

The methodology for design and optimization of Fuzzy Systems proposed by [Godj97] has been briefly introduced. It allows finding an optimal Fuzzy Controller in an iterative way. Beginning with the know-how provided by an expert the adaptation of the controller's parameters is on-line performed under the supervision of a learning algorithm. After learning is accomplished, this methodology allows the expert to redefine or improve his knowledge regarding the process by extracting rules that are more accurate.

Finally, regarding Takagi-Sugeno's models, ANFIS [Jang92] provides a network representation of the Fuzzy System allowing the use of the gradient descent algorithm, the widespread adapting technique for Neural Networks. The use of ANFIS was demonstrated through an example for fitting a non-linear function. The usefulness of this technique can be appreciated by means of the fast convergence of the system. This is due to the previously encoded linguistic knowledge before learning starts [Mend95].

This page intentionally left blank

Chapter 3

ANALOG BASIC BUILDING BLOCKS

A comprehensive analysis of circuits performances

3. INTRODUCTION

A direct mapping of the previously presented Fuzzy algorithms into silicon requires the implementation of basic circuits for the computation of *"if-then"* rules. According to Figure 2.9 these basic blocks can be grouped as follows:

1. Fuzzy Membership Functions circuits (FMF and/or CFMF).
2. T-Norm and T-CoNorm circuits.
3. Consequent circuits (for Takagi-Sugeno's controllers).
4. Defuzzifiers circuits (Normalizers).

This chapter deals with the analysis of the above building blocks in the frame of analog realizations of Fuzzy Logic Controllers. For each circuit the main items concerning its functional behavior, systematic errors as well as mismatch errors due to technological parameter fluctuations will be discussed. The analysis of the imperfections of the circuits developed in this chapter is wide-ranging and can be applied for any degree of accuracy aimed at a given implementation. However, features concerning the transient behavior and sizing criterions will be discussed in Chapter 4 together with the implementation of the Fuzzy Controllers.

In view of a broad flexibility that allows extending the range of applications, programmable fuzzy chips are preferred. Therefore, tuning is another important issue that has been specially considered throughout the synthesis of membership functions and consequent circuits. Although the

used CMOS process allowed us to fabricate only discretely-programmable analog demonstrators, the latter mentioned blocks have been designed following a continuous electrically-tuning philosophy. Thus, they can also be used in the case where analog non-volatile memory devices are available.

Given that the defuzzifying procedure in the Mamdani algorithm demands high computational efforts, its analog implementation becomes cumbersome. For this reason only defuzzifier circuits for Takagi-Sugeno's (TS) controllers (i.e. averaged weighted sums) have been considered all along this work. However, with the exception of the particular implementation presented herein for the defuzzifier, all others circuits can be used in any algorithm implemented in a Fuzzy Controller.

Based on previous related approaches [VaVi99] [Espe94], the last section of this chapter presents a methodology for the error propagation analysis. This technique enables designers to get insight on the global accuracy attainable by a given configuration. Starting with the characterization of the error introduced by each particular operating block, a comprehensive final expression for the variance of the controller's output can be obtained. This expression involves the individual error contributions by each block and permits to identify the critical sources of imperfections along the signal path. This technique was applied to estimate the precision achievable by an architecture built with the proposed circuits. Any realistic situation that can be met during the execution of the fuzzy rules of most controllers can be modeled.

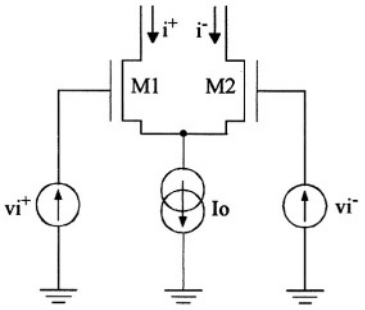
3.1 Fuzzy Membership Functions

Fuzzifying is the first step in a Fuzzy System. This is carried out through the Fuzzy Membership Function (FMF) circuits, which perform a non-linear transform from their inputs to their outputs. In the previous chapter, the possibility of working also with the complementary version of the membership functions (CFMF) has been pointed out. In this chapter, we will use sometimes the term FMF to refer either to the direct or to the complementary membership function, or simply, we will call it membership function (MF).

FMF's inputs interact with the external environment from which signals are normally supplied in voltage mode. On the other hand, it has been shown that the processing steps following the fuzzification stage perform better in current mode [BaHu97] [LePa94] [MaCo97] [MaFa94] [PaQu95] [VaVi99]. It turns out that the choice of transconductors for the synthesis of FMFs circuits offers an optimal alternative for the input interface requirement.

The most popular transconductor used for the synthesis of FMFs reported in the literature [Vida96] [SoQu98] [MaFa94] [GuPe96] [RoPe97] consists

in the simple differential pair with saturated input transistors. The circuit is shown in Figure 3.1 together with its large-signal DC transfer function equations [LaSa94].



$$Vin = vi^{+} - vi^{-}$$

$$I = i^{+} - i^{-} = \begin{cases} \sqrt{\beta Io}\, Vin \sqrt{1 - Vin^{2}\frac{\beta}{4Io}} & |Vin| \le \sqrt{\frac{2Io}{\beta}} \\ Io\, sgn(Vin) & |Vin| \ge \sqrt{\frac{2Io}{\beta}} \end{cases}$$

$$\beta = \mu\, Cox\, (W / L)_{M1,2}$$

Figure 3.1. Differential amplifier circuit and its DC large-signal equations

Because of the simplicity, its use is preferable when fixed dedicated Fuzzy Controllers are contemplated. However, for general-purpose programmable Fuzzy Controllers it presents some disadvantages, which are summarized hereafter:

a) The tail current Io is the maximum current that a differential pair can supply through one of its branches. Hence, Io represents the logical value "1" in a membership function built by a differential pair and it must be fixed to a reference level. In a Fuzzy Controller, all fuzzy operations performed by any fuzzy block are carried out with respect to that reference level (i.e. Io). Since the transconductance of the differential pair depends on Io and of the size of the transistors, the slopes of the membership functions cannot be electrically tuned. This is a substantial weakness, specially when analog storage is available for on-chip electrical tuning.
b) Consider the case where discrete programming of slopes is being carried out by switching a set of differently sized input transistors. Calling S the ratio between the maximum and the minimum desirable slopes, the ratio between the maximum and minimum transistor size becomes $((W/L)_{max}/(W/L)_{min})=S^2$. Thus, the total amount of silicon surface to build a discretely-programmable transconductor may become very large as S increases.
c) Low-power Fuzzy Controllers need relatively low tranconductance values for their membership function circuits. For instance, to map input voltages ranging from 0.3V to 3V into currents ranging from 0 to $10\mu A$

the transconductance should range from 3.3 μS to 33μS. As long as the slopes of the FMF need to be small, very long channel saturated transistors (i.e. L=250μm) with increased active area (W×L) are needed. Consequently, the input gate-to-source capacitance Cgs could result considerably large and the transient behavior of the transconductor could be degraded.

All previously mentioned shortcomings may be overcome if the input transistors are constrained to operate in the triode region performing linear (or quasi-linear) transconductors. In this way, their small signal transconductance is equal to:

$$gm = \frac{\partial Id}{\partial Vgs} = \mu Cox \frac{W}{L} Vds. \qquad (3.1)$$

Equation (3.1) attests that triode transconductors meet smartly the requirements for tunable FMFs in the following aspects:

a) Slopes could be electrically programmed by controlling the drain-to-source voltage drops of triode transistors (Vds).
b) For digital discrete programming, the ratio between the maximum and the minimum transistor size is equal to S, rather than S^2, for a given desirable slope range. Therefore, FMF circuits result smaller leading to a considerable saving of silicon area.
c) Small slopes are readily achievable by setting smaller Vds without the need for very long channel transistors.

In the next sections, two kinds of FMFs built around a regulated-cascode triode transconductor will be presented after the analysis and characterization of the transconductors themselves.

3.1.1 Differential Regulated-Cascode Triode Transconductor

The circuit of a differential triode transconductor is depicted in Figure 3.2 a). It comprises two regulated-cascode loops (M1, Mc1, DA1 - M2, Mc2, DA2). The drain-to-source voltage drops (Vds) across transistors M1 and M2 are kept constant over a wide range of the input voltage Vin. These Yds are fixed by means of the artificially increased offset voltages of the differential amplifiers DA1 and DA2 as will be explained later. Since these offsets are smaller than the saturation drain-to-source voltage $Vds_{sat}=(Vgs-V_T)/n$ of each input transistor, the latter are constrained to operate in the triode region.

Thus, the transconductances gm_{M1} and gm_{M2} are given by (3.1) and can be electrically tuned to an appropriate value provided that Vds can be controlled. This is done by means of the feedback loops around Mc1 and Mc2, which reproduce the gain-boosting circuit described in [BuGe90].

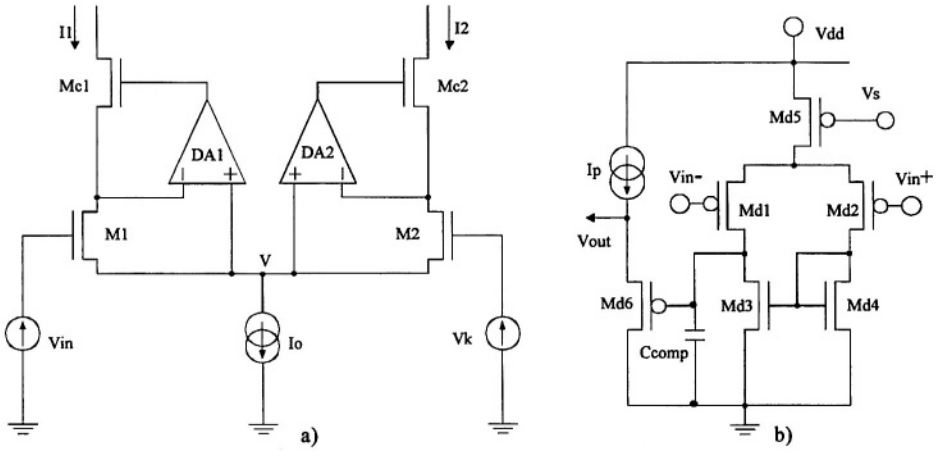


Figure 3.2. a) Differential regulated-cascode triode transconductor. b) Differential amplifiers of the regulated-cascode loops.

The loop differential amplifiers circuit is sketched in Figure 3.2 b). The choice of the PMOS differential pair rests on the fact that the input voltage range of the transconductor can be enlarged. Consequently, a level shifter (Md6, Ip) must be inserted to reach the bias voltage level required at the gate of transistors Mc1 and Mc2. However, a capacitor Ccomp is needed at the gate of Md6 in order to compensate the DA and stabilize the loop. Considering we are dealing with a tunable transconductor for a programmable membership function; the stability conditions must be guaranteed for every value of the tuning voltage source Vs. Allowing each DA to be non-symmetrical [Anne98], its input voltage offset can be linearly controlled by the voltage source Vs. Hence, choosing $(W/L)_{Md1}>(W/L)_{Md2}$ and assuming all transistors being biased in strong inversion, we have:

$$\begin{aligned} Vds &= Vin^{-} - Vin^{+} \\ &= \left(\sqrt{\frac{(W/L)_{Md5}}{2\,(W/L)_{Md2}}} - \sqrt{\frac{(W/L)_{Md5}}{2\,(W/L)_{Md1}}} \right) (Vdd - |VTp| - Vs). \end{aligned} \tag{3.2}$$

Considering in Figure 3.2 a) symmetrical branches ($\beta 1=\beta 2=\beta$; Vds1=Vds2=Vds and $gm_{M1}=gm_{M2}=gm$) and transistors Ml and M2 working

in the triode region, we can write the following expressions for the currents I1 and I2 and the common-source node voltage V:

$$I1 = \beta\left((Vin - V - VTn)Vds - \frac{n}{2}Vds^2 \right), \tag{3.3}$$

$$I2 = \beta\left((Vk - V - VTn)Vds - \frac{n}{2}Vds^2 \right), \tag{3.4}$$

$$V = \frac{1}{2}\left(Vin + Vk - 2VTn - nVds - \frac{Io}{gm} \right). \tag{3.5}$$

Considering Vin as the input variable and Vk as a parametric threshold, the latter three equations can be used to obtain the approximated DC voltage-to-current transfer characteristic of the transconductor:

$$I1 \approx \begin{cases} 0 & \text{if} \quad 0 \le Vin \le Vk - \dfrac{Io}{gm}, \\ \dfrac{gm}{2}\left(Vin - Vk + \dfrac{Io}{gm} \right) & \text{if} \quad Vk - \dfrac{Io}{gm} \le Vin \le Vk + \dfrac{Io}{gm}, \\ Io & \text{if} \quad Vk + \dfrac{Io}{gm} \le Vin\,; \end{cases} \tag{3.6}$$

$$I2 = Io - I1. \tag{3.7}$$

Figure 3.3 shows a graphical representation of (3.6) and (3.7) superimposed to a hand-made curve that intends to represent the actual behavior of the circuit. A disparity between both approaches is found when currents I1 and I2 lie near either zero or Io. In those particular points, either transistor M1 or M2 is driven through the saturation region because its gate-voltage overdrive GVO is smaller than its Vds voltage drop. This remains unchanged due to the regulation loop.

In spite of the above discussed local divergences, the approximate expressions are fairly acceptable since the transconductor will be used to build a FMF circuit, whose linearity is normally irrelevant.

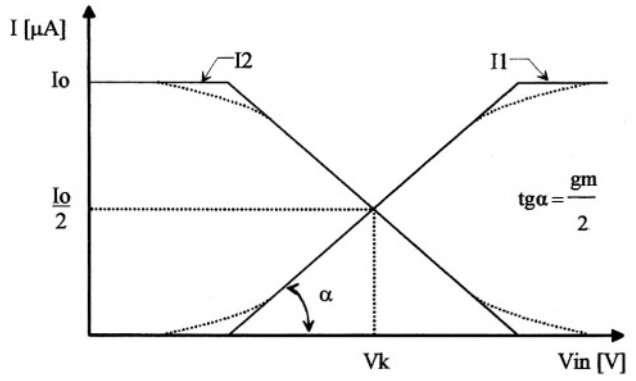


Figure 3.3. Approximate (solid) and accurate (dashed) DC transfer characteristic of the differential triode transconductor. In this case $gm=gm_{M1}=gm_{M2}$.

3.1.2 Single Regulated-Cascode Triode Transconductor

By replacing one of the regulated-cascode branches by a large size transistor in the circuit of Figure 3.2 a), another transconductor, much simpler than the previous one, is obtained. It is illustrated in Figure 3.4 a). A significant saving in silicon area and complexity is thus achievable by using this implementation.

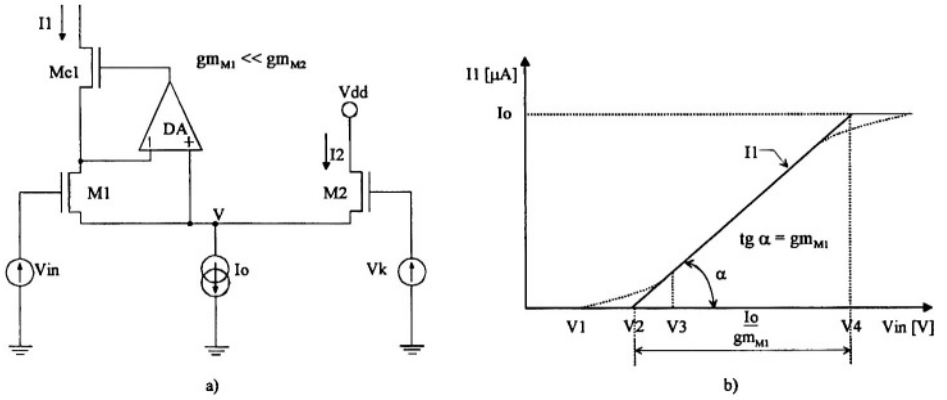


Figure 3.4. a) Single regulated-cascode triode transconductor circuit. b) Approximate (solid) and actual (dashed) DC transfer characteristic.

In Figure 3.4 a), the large size transistor M2 is aimed for settling a threshold voltage reference at the common-source node V. Considering that Vk is constant, as long as M2 is being switched on the voltage V is kept nearly invariant as the gate-voltage overdrive of transistor M2 is negligible.

This condition is sustained even when M2 conveys the maximum current (Io), provided that the latter transistor is large.

Figure 3.4 b) shows in dashed line a hand-made curve that intends to represent the actual evolution of the output current I1 against Vin. In solid line, a linear approximation is sketched. Like in the former transconductor the differences appear around the knees because of the different conduction modes that transistor Ml switches through. Particularly, between V1 and V3 M1 is saturated (i.e. $GVO_{M1}<Vds_{M1}$). This explains the parabolic trace at this interval.

Let us define the nominal threshold voltage for the transconductor as the value at the intersection of the plain line (approximate behavior) with the Vin axis, denoted V2 in Figure 3.4 b). It can be demonstrated that the value of V2 is given by:

$$V2 = Vk + \frac{n\, Vds_{M1}}{2}. \tag{3.8}$$

According to the latter equation, if Vds_{M1} is kept small and n, accounting for the Body effect, rounded to one, V2 should be rather close to Vk. This condition (i.e. $nVds_{M1}/2<Vk$) must be carefully respected during the design in order to avoid transconductors whose knee threshold may depend notably on its slope value for the same threshold voltage Vk (the slope of the transconductor is given by $gm_{M1}=\beta 1 Vds_{M1}$).

Finally, an approximate expression for I1 in Figure 3.4 a) is given by:

$$I1 \approx \begin{cases} 0 & \text{if} \quad 0 < Vin < Vk, \\ gm_{M1}(Vin - Vk) & \text{if} \quad Vk < Vin < Vk + \dfrac{Io}{gm_{M1}}, \\ Io & \text{if} \quad Vk + \dfrac{Io}{gm_{M1}} < Vin. \end{cases} \tag{3.9}$$

3.1.3 Complementary Fuzzy Membership Function Type-I

Two differential transconductors as presented in section 3.1.1 make up the circuit drawn at Figure 3.5. It corresponds to a four-parameter electrically tunable CFMF, whose shape is nearly trapezoidal. Two independent slopes can be programmed by setting the voltage sources Vs1 and Vs2 to the corresponding values. Voltages Vk1 and Vk2 take care of the placement of the fuzzifiers (labels) along the voltage range of Vin (i.e. the

Universe of Discourse). Cross-sums of the transconductors currents give rise to outputs Iout1 and Iout2. These, in turn, are sunk from the low-compliance cascoded diodes built by transistors Mp1_1-Mp2_1 and Mp1_2-Mp2_2, respectively. Mirror transistors should be employed to replicate the outputs as many times as the same CFMF is being shared by different rules of a Fuzzy Controller.

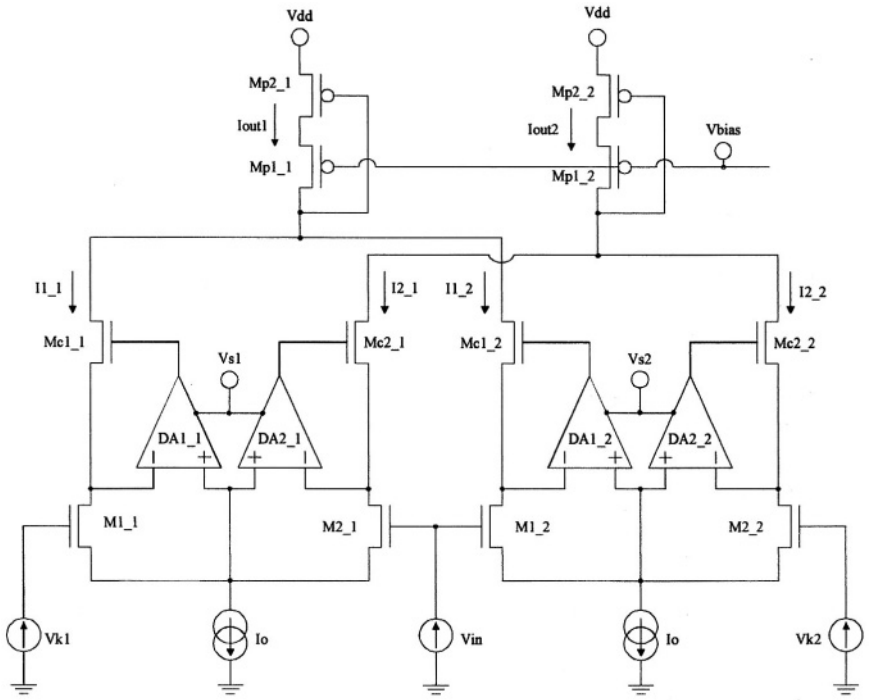


Figure 3.5. Four-parameter Complementary and Direct Fuzzy Membership Function Type-I.

DC input-output relationships are graphically represented at Figure 3.6, assuming the following settings: $gm1=gm_{M1\_1}=gm_{M2\_1}$ and $gm2=gm_{M1\_2}=gm_{M2\_2}$. It is worth noticing that the circuit offers the possibility to synthesize either Direct or Complementary Fuzzy Membership Functions, depending on which output is being considered. Despite of this, only the behavior of the CFMF will be discussed in this section. However, the results and conclusions concerning the systematic errors and the mismatch errors could be also extended to direct FMFs.

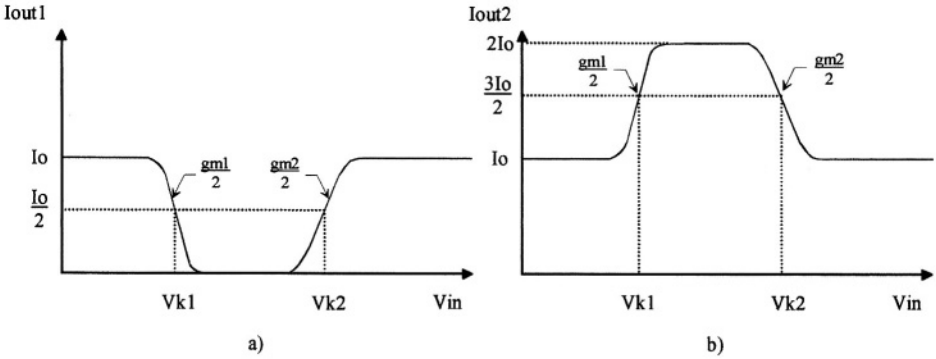


Figure 3.6. a) Complementary FMF (CFMF) b) Direct FMF.

3.1.3.1 Input Voltage Range of the CFMF Type-I.

Since a membership function is the first operational block in a fuzzy processor, its input voltage range is a major concern for design. Given the dynamic range for proper operation of the system and in order to relax noise figures, a large input voltage swing should be attained. Moreover, some applications need a fine partition of the input Universe of Discourse. Therefore, the wider the input range, the more fuzzy labels could be accommodated provided the minimum width of the fuzzifiers (FMFs or CFMFs) are also technically constrained.

Figure 3.7 shows only one branch of a differential transconductor that builds a CFMF Type-I. It illustrates the main devices conditioning both the minimum and maximum allowable input voltage Vin. Since the differential amplifier DA1 is built from a PMOS input pair, the minimum value for Vin is limited by the minimum gate-to-source voltage needed to switch M1 on. In addition, the voltage compliance demanded by the current source Io must be added. Therefore:

$$Vin \geq VTn + Vdssat_{Mn1} + Vdssat_{Mn2} . \qquad (3.10)$$

To determine the upper limit for Vin we first assume that the current Io is being steered throughout transistor M1, which is still in triode region. Expressions (3.3), (3.4) and (3.5) are used to determine the value of the common-source node voltage V. This, added to the controlled Vds1 voltage drop of M1, gives the voltage value of the gate of Md1 in DA1. From this point, there are two possible paths towards Vdd to consider. The first one, through transistor Mc1, limits the maximum voltage Vin to:

$$Vin \le Vdd + \frac{Io}{gm1} + (VTn - |VTp|) - \left(1 - \frac{n}{2}\right)Vds1 - \frac{1}{n}\sqrt{\frac{2nIo}{\beta_{Mc1}}} - \sqrt{\frac{2nIo}{\beta_{Mp2}}}. \quad (3.11)$$

The second alternative, through DA1, Vgs of transistor Md1 and Vds_{sat} of Md5, constrain Vin to:

$$Vin \le Vdd + \frac{Io}{gm1} + (VTn - |VTp|) - \left(1 - \frac{n}{2}\right)Vds1 - \sqrt{\frac{2nId_{Md1}}{\beta_{Md1}}} - \frac{1}{n}\sqrt{\frac{2nId_{Md5}}{\beta_{Md5}}}. \quad (3.12)$$

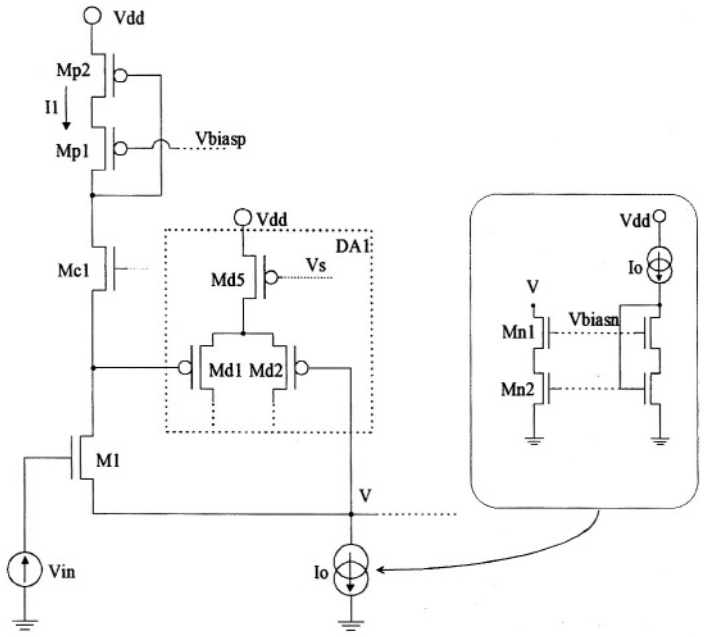


Figure 3.7. Half differential transconductor used to estimate the Vin range the CFMF Type-I.

In the actual implementations, (3.11) is a little larger than (3.12) (i.e. ≈0.4V). Therefore, the latter expression establishes the maximum allowable value for Vin.

3.1.3.2 Systematic Errors in the CFMF Type-I.

Systematic errors typify the deviations from the desired behavior a particular circuit undergoes, assuming nominal values of the technological parameters. They are due mainly to second order non idealities of the devices (i.e. Early effect, Body effect, etc) and can be modeled deterministically to some extend. Different circuit architectures accomplishing the same function might be more or less sensitive to the same non ideality. Consequently, systematic errors are also closely related to the topology of the circuit itself.

On the other hand, one of the most frequently used FMF shapes at system level simulation and modeling is the so-called *bell-shaped* FMF or CFMF [Jang92]. Its mathematical expression is given by (3.13). In order to clarify ideas about systematic errors it will be supposed in the following that we are trying to fit a real CFMF circuit to this symmetrical function [Vida96]. Thus, to perform the comparison let us relate the parameters of the ideal function in Figure 3.8 to the actual parameters of the real symmetrical CFMF Type-I:

$$\mu(x) = 1 - \frac{1}{1 + \left(\frac{x - xo}{d}\right)^{2p}}. \qquad (3.13)$$

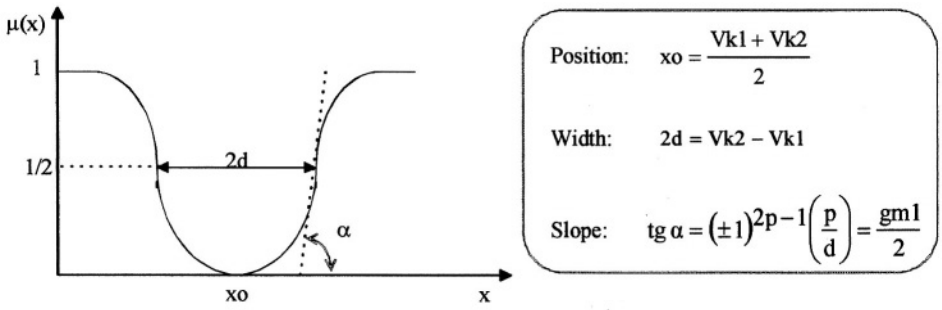


Figure 3.8. Bell-shaped symmetrical CFMF and its parameters, which are related to the electrical parameters of the CFMF Type-I.

Figure 3.9 highlights two kind of systematic errors typically present in most differential transconductors-based CFMFs [Vida96]. The ideal bend of the curves is drawn in dashed lines. Notice in Figure 3.9 a) the shift of the crossover points (i.e. when I=Io/2) for a given couple of thresholds Vk1 and Vk2. These are now reallocated at Vk1' and Vk2' respectively. The nominal width 2d and position xo of the membership function, as defined in Figure 3.8, have also changed. This is a consequence of the Early effect at

the transconductors' branches. To understand the origin of this non ideality let us assume in Figure 3.5 that we are using the simple differential pair of Figure 3.1 in place of the differential regulated-cascode transconductors. Let us just consider the left transconductor of Figure 3.5 that comprises transistors M1_1 and M2_1, which are now saturated. In that case, the Early effect in transistors M1_1 and M2_1 should not be neglected. Crossover points take place when both inputs gate voltages at the pair are equal. When this occurs, the current source Io is not equally split because the drain voltages of M1_1 and M2_1 are different as they are tied to different nodes. From Figure 3.5 and Figure 3.6 it is clear that the previous mentioned nodes voltages differ. Fortunately, thanks to the regulated-cascode configuration we have used to build our CFMF, this error becomes negligible.

The second kind of error shown in Figure 3.9 b) is a consequence of the Early effect in the transistors delivering the tail current Io. From (3.5) it is clear that the common-source voltage V in Figure 3.2 a) varies accordingly to Vin and Vk. Thus, ΔIo in Figure 3.9 b) depends not only on the input voltage but also on the position of the fuzzifier along the input range of Vin. Certainly, this current error reduces to a tolerable value as the finite output conductance go of the current source Io is being decreased by cascoding.

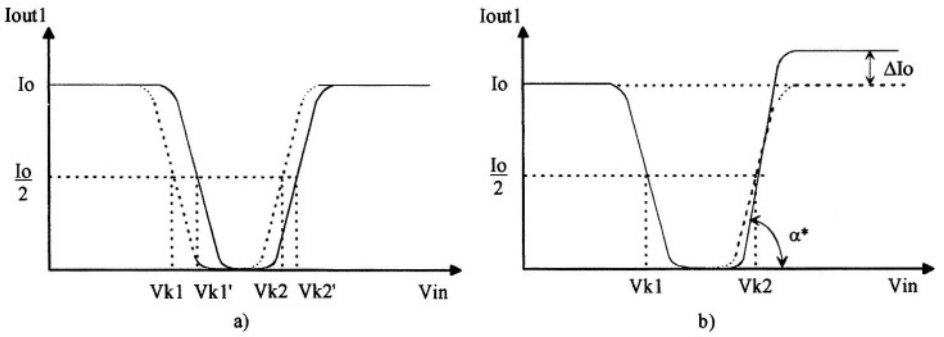


Figure 3.9. Systematic errors in a CFMF: a) Crossover points are shifted whereas the width changed due to the Early effect at the branches of a differential pair. b) Change on the maximum current level due to the Early effect at the mirror supplying the tail current Io.

In Figure 3.10, a CFMF holding the latter defect (in solid line) is compared with an ideal symmetrical one (in dashed line), which is free of Early effect at transistors supplying Io. Thus, the position xo and the width 2d spread out their nominal values as follows:

$$xo' = xo + \Delta xo = \frac{Vk1 + Vk2}{2} + \frac{go(Vk2 - Vk1)}{2gm1}, \tag{3.14}$$

$$d' = d + \Delta d = \frac{Vk2 - Vk1}{2} + go\left(\frac{\Delta V}{gm1} + \frac{Vk2 - Vk1}{2gm1}\right). \qquad (3.15)$$

In the above expressions, gm1 is the transconductance of all triode transistors in Figure 3.5 (i.e. gm1=gm1_1=gm2_1=gm1_2=gm2_2). Expressions (3.14) and (3.15) demonstrate that Δxo and Δd are magnified for wider and less raised CFMFs. Notice also the dependence of Δxo and Δd on the conductance go.

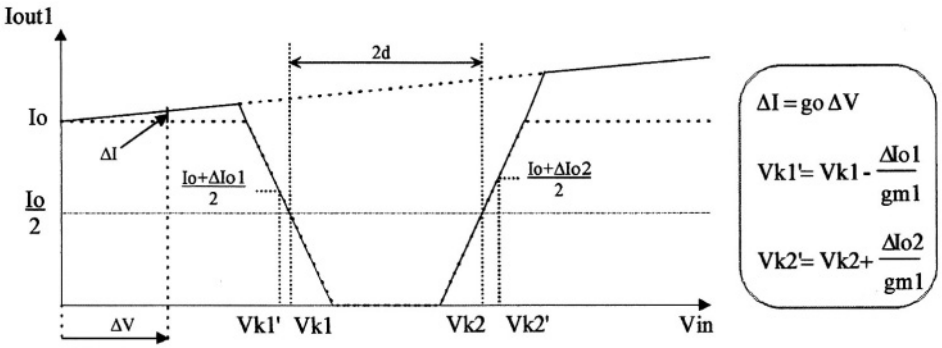


Figure 3.10. Deviations of the parameters of a symmetrical CFMF Type-I owing to the finite conductance go of the current source Io. The ideal curve shape is in dashed line. "gm1" is the transconductance of both branches of the symmetrical CFMF Type-I.

Another kind of systematic error is introduced by the loop amplifiers DA1 and DA2 in each transconductor of the CFMF. Any current difference between the branches of the DAs will cause Vds to move from its nominal value given in (3.2). Consequently, the nominal slopes of the CFMF will change. Figure 3.11 shows a branch of one of the triode transconductors of the CFMF Type-I. It can help us to get insight about this systematic error.

If the drain voltages Vd3, Vd4 of transistors Md3 and Md4 are different, a mismatch between currents Id1 and Id2 appears owing to the Early effect. It follows that a ΔVds must be added to the expression (3.2). This additional drain-to-source offset is approximately given by:

$$\Delta Vds = \left(\frac{\partial GVO_{Md2}}{\partial Id2} + \frac{\partial GVO_{Md1}}{\partial Id1}\right)\frac{\lambda n\, Id\,(Vd4 - Vd3)}{4}, \qquad (3.16)$$

where λn [V^{-1}] is the channel length modulation coefficient accounting for the Early effect of transistors Md3 and Md4. Using (3.2), (3.5) and (3.16),

assuming identical PMOS and NMOS threshold voltages and the input Vin set to the crossover point value (Vin=Vk), we have:

$$\frac{\Delta Vds}{Vds} = \frac{\lambda n}{4}\left(\frac{(\beta_{Md2})^{-\frac{1}{2}} + (\beta_{Md1})^{-\frac{1}{2}}}{(\beta_{Md2})^{-\frac{1}{2}} - (\beta_{Md1})^{-\frac{1}{2}}}\right)(Vd4 - Vd3), \tag{3.17}$$

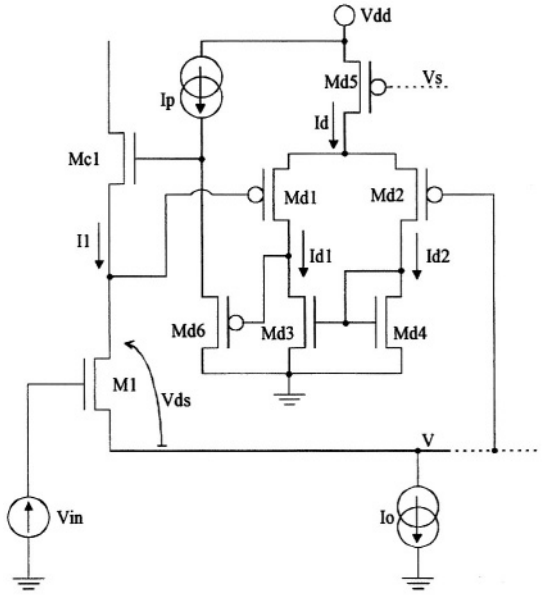


Figure 3.11. Circuit used to estimate systematic errors in the slopes of the CFMF Type-I.

where:

$$Vd4 - Vd3 = (2VTn - Vk) + \sqrt{\frac{2nIp}{\beta_{Md6}}} - \sqrt{\frac{nIo}{\beta_{Mc1}}} + \frac{Io}{2gm1} + \frac{gm1}{\beta_{M1}}\left[\left(\sqrt{\frac{\beta_{Md4}}{\beta_{Md2}}} - \sqrt{\frac{\beta_{Md4}}{\beta_{Md1}}}\right)^{-1} - \left(1 - \frac{n}{2}\right)\right]. \tag{3.18}$$

Since Vk appears in (3.18), the relative error of Vds depends on the position of the CFMF along the input range of Vin. For a given current Io, the aspect ratio of transistors Md6 and Mc1 should be chosen in order to

minimize the term (Vd4–Vd3), for reducing this error. However, in order to keep the transition frequency of the loop amplifiers DAs beyond the bandwidth of the transconductor, the current Ip cannot be constrained to very small values. Finally, Table 3.1 gives some typical values for the systematic errors of this CFMF assuming Vdd=5V.

Table 3.1. Systematic errors in the CFMF Type-I.

Δxo	Δd	ΔVds/Vds (1.5V < Vk < 4.5V)
1.1mV	3.3mV	0.037 < ΔVds/Vds < 0.065

3.1.3.3 Mismatch Errors in the CFMF Type-I.

Technological parameters fluctuations around their nominal values lead to device mismatch. Consequently, the performances of the circuits are subject to some tolerance. Robust design strategies point at shortening those tolerances as much as possible. The basis of matching properties of CMOS transistors was exhaustively studied by [PeDu89] [KiSt96] and verified by [Espe94] [Vida96]. Most relevant conclusions are summarized in the following.

In MOSFETs, it is possible to distinguish three kinds of mismatches:

a) *Mismatch of two transistors implemented on different wafers and different batches*: it gives information about the reproduction capability of the circuits and it must be considered for industrial and commercial developments. This is out of the scope of this work.

b) *Mismatch of two transistors implemented on the same wafer and far from each other:* the variance of the parameters is proportional to the square of the distance D between devices. For the threshold voltage and the current gain the following expression for their variance agree with experimental measurements:

$$\sigma^2_{\Delta VTo} = S^2_{VTo}\, D^2, \qquad \frac{\sigma^2_{\Delta\beta}}{\beta^2} = S^2_{\beta}\, D^2. \tag{3.19}$$

c) *Mismatch of two transistors implemented on the same wafer with the same size, shape, orientation and close to each other:* in this case the variances are inversely proportional to the transistor area:

$$\sigma^2_{\Delta VTo} = \frac{A^2_{VTo}}{WL}, \qquad \frac{\sigma^2_{\Delta\beta}}{\beta^2} = \frac{A^2_{\beta}}{WL}, \tag{3.20}$$

where the coefficients S_{VTo}, S_β, A_{VTo} and A_β are process-dependent constants.

In conclusion, the total variance for each class of parameter of devices that belong to the same wafer is found by adding (3.19) to (3.20). However, if small size transistors (or large size transistors split in smaller ones connected in parallel) are only considered, the terms depending on the distance D become smaller if compared to the one depending on transistors size [PeDu89]. Consequently, we will consider through this analysis the mismatch between close transistors only.

Coming back to the CFMF circuit under analysis, our goal is to find expressions for the variance of the output current Iout1 in Figure 3.5. This output current is considered now as a random variable whose mathematical expectation is given by equations (3.6) and (3.7) for any value of Vin. To facilitate the analysis we will concentrate only on symmetrical CFMFs but the results could be easily generalized. It should be considered that each edge of the CFMF in Figure 3.6 a) is yielded by a branch of one among the two differential transconductors in Figure 3.5. For this reason we will focus the analysis on one of the transconductors by using its input-output relationships given by (3.2) to (3.5).

The policy is to find difference equations for the output current Iout1 as a function of the technological variables (i.e. VT and β) of devices playing an active role. Afterward, the variance of the current is found by taking the sum of the squared terms of the expression, assuming the technological variables are themselves statistically independent. With regards to Figure 3.3 and Figure 3.5 the following expressions are satisfactory for the variance of the fraction of Iout1 supplied by only one of the transconductors:

$$\sigma^2_{Iout1} = \begin{cases} 0 & \text{if } Iout1 = 0, \\ \frac{1}{4}\left[Iout1^2 + (Io - Iout1)^2\right]\frac{\sigma^2_{\beta 1}}{\beta 1^2} + \frac{1}{2}gm1^2\sigma^2_{VTn1} + & \\ \quad + \frac{1}{2}gd1^2\sigma^2_{Vds1} + \frac{1}{4}\sigma^2_{Io} & \text{if } 0 < Iout1 < Io, \\ \sigma^2_{Io} & \text{if } Iout1 = Io, \end{cases} \tag{3.21}$$

where $\beta 1$, gm1, gd1 and Vds1 correspond to transistor M1_1 in Figure 3.5, and the variances for Io and Vds1 are given below by equations (3.22) and (3.23), respectively. These equations are also applicable for the second half of the CFMF by changing the subscripts of parameters and variables. Strictly

speaking, when Iout1 is zero, its variance is equal to the leakage current variance of the cut-off transistor M1_1 instead of zero.

$$\sigma^2_{Io} = 2\,Io^2\,\frac{\sigma^2_{\beta_{Mn2}}}{\beta^2_{Mn2}} + 8\,Io\,\beta_{Mn2}\,\sigma^2_{VTn_{Mn2}}. \tag{3.22}$$

$$\sigma^2_{Vds1} = \sigma^2_{VTp_{Md1}} + \sigma^2_{VTp_{Md2}} + \frac{1}{2}\left(\sqrt{\frac{\beta_{Md5}}{\beta_{Md2}}} - \sqrt{\frac{\beta_{Md5}}{\beta_{Md1}}}\right)^2 \sigma^2_{VTp_{Md5}} + \frac{\left[\left(\sqrt{\frac{\beta_{Md5}}{\beta_{Md2}}} - \sqrt{\frac{\beta_{Md5}}{\beta_{Md1}}}\right)^2 \frac{\sigma^2_{\beta_{Md5}}}{\beta^2_{Md5}} + \left(\frac{\beta_{Md5}}{\beta_{Md2}}\right) \frac{\sigma^2_{\beta_{Md2}}}{\beta^2_{Md2}} + \left(\frac{\beta_{Md5}}{\beta_{Md1}}\right) \frac{\sigma^2_{\beta_{Md1}}}{\beta^2_{Md1}}\right]}{2\left(\frac{gm}{Id}\right)^2_{Md5}}. \tag{3.23}$$

One must add in (3.22) the effect of parameters fluctuations of the cascode transistor Mn1 (see Figure 3.7). This effect was ignored so far because it produces a small random shift of the drain voltage of the saturated transistor Mn2. Moreover, in (3.23) we did not even consider the mismatch of the current mirror Md3, Md4. The variance of the current at this mirror, divided by the square of the transconductance of the feedback amplifier DA1 accounts for another term that must be added to σ^2_{Vds1}. Nevertheless, in the actual implementations this quantity is negligible with respect to the variance of threshold voltages VTp of the input differential pair of DA1. In addition, the influence of the terms due to the transistors' β spreading in (3.23) could be kept considerably small if Md5 is being biased in the moderate inversion region (i.e. high $(gm/Id)_{Md5}$). In this way, the major contributions to σ^2_{Vds1} are due to the PMOS threshold voltage mismatch of DA1, which in turn can be halved if the centroid layout technique is practiced [PeDu89] [Vitt94].

Finally, the variances of the system-level parameters xo and d are given by:

$$\sigma^2_d = \sigma^2_{xo} = \frac{Io^2}{4gm1^2}\,\frac{\sigma^2_{\beta 1}}{\beta 1^2} + \frac{n^2}{4}\,\sigma^2_{Vds1} + \sigma^2_{VTn1}. \tag{3.24}$$

Table 3.2 shows the mismatch parameters used to calculate the variance of the threshold voltages and current gains of transistors. Figure 3.12 a) shows a simulation of the standard deviation of Iout1 for a CFMF Type-I. Two different slopes have been set: gm=7μS and gm=70μS. Figure 3.12 b) shows the individual contribution of each variable to the total variance of Iout1 for the case of the smaller transconductance.

Table 3.2. Mismatch parameters extrapolated from [PeDu89] [KiSt96].

A_{VTon}	A_{VTop}	$A_{\beta n}$	$A_{\beta p}$
24 mVμm	28 mVμm	2.3 %μm	3.2 %μm

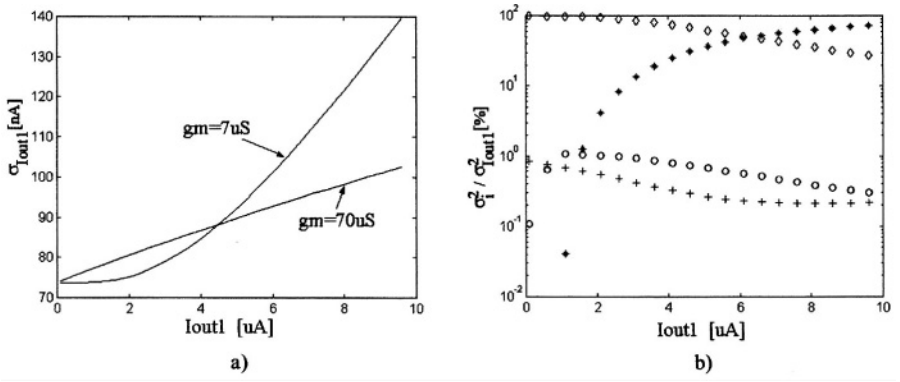


Figure 3.12. a) Standard deviation of the output current of the CFMF Type-I due to mismatch. b) Individual contribution of each random variable in (3.21) to the total variance of the output current Iout1 for gm=7μS: (+) $\sigma_{\beta 1}$; (o) σ_{VT1}; (*) σ_{Vds1} and (◊) σ_{Io}.

3.1.4 Complementary Fuzzy Membership Function Type-II

Two single triode transconductors as the one presented in section 3.1.2 build up the circuit of the CFMF depicted in Figure 3.13. This is also a four-parameter fuzzifier. With proper design, slopes and positions can be almost independently tuned.

As discussed in section 3.1.2, large size transistors M2_1 and M2_2 are required. In Figure 3.14 the generic DC transfer characteristic is represented together with some parameters definitions. For symmetrical membership functions one must set $gm_{1\_1}=gm_{1\_2}=gm$. It should be noticed that the crossover points defined in previous section (i.e. Iout1=Io/2) do not coincide with thresholds Vk1*, Vk2* like in the former CFMF (Vk1* and Vk2* are the little shifted versions of Vk1 and Vk2, due to the small GVO of M2_1 and M2_2, respectively). As a result of the new allocation of the crossover points, for the same pair Vk1 and Vk2, the width 2d of a symmetrical membership function, as defined in Figure 3.8, will depend on the slope. In

contrast, the expression for the position xo remains the same as in the CFMF Type-I.

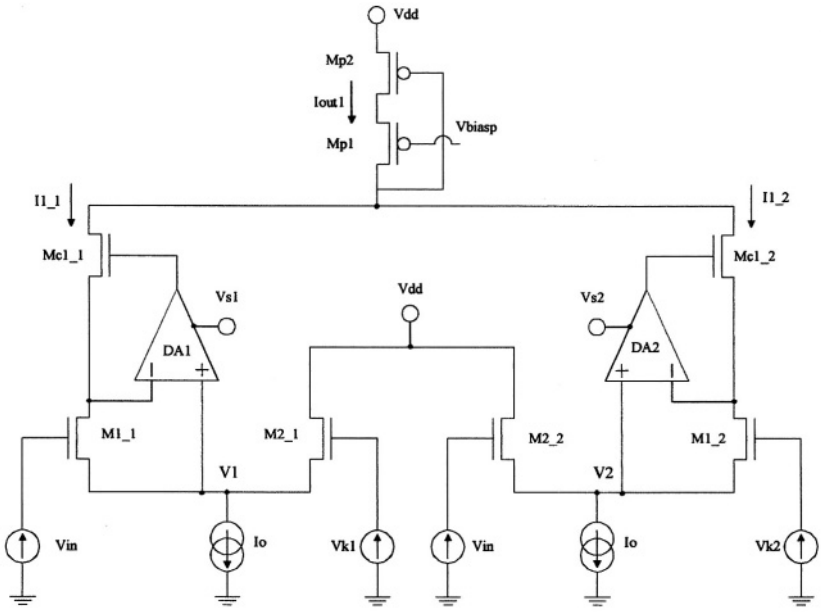


Figure 3.13. Complementary Fuzzy Membership Function Type-II circuit [DuVe00].

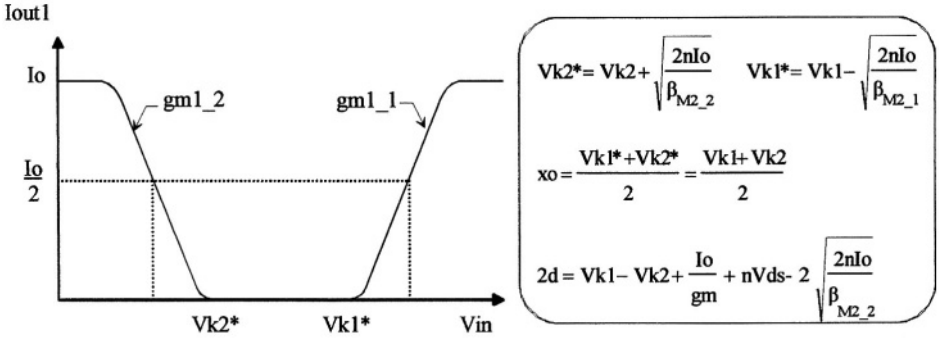


Figure 3.14. Complementary FMF Type-II: generic DC input-output transfer characteristics and parameters definitions for symmetrical CFMFs: $gm_{1\_1}=gm_{1\_2}=gm$, $Vds1\_1=Vds1\_2=Vds$, $\beta_{M1\_1}=\beta_{M1\_2}$ and $\beta_{M2\_1}=\beta_{M2\_2}$.

3.1.4.1 Input Voltage Range of the CFMF Type-II.

The minimum value for Vin is determined by the compliance voltage of the current source Io and the threshold voltage of transistor M1_1. Assuming a cascoded mirror supplying Io as in Figure 3.7:

$$\mathrm{Vin} \geq \mathrm{VTn} + \mathrm{Vdssat}_{\mathrm{Mn1}} + \mathrm{Vdssat}_{\mathrm{Mn2}} \,. \tag{3.25}$$

The maximum allowable value for Vin is given by:

$$\mathrm{Vin} \leq \mathrm{Vdd} + \frac{\mathrm{Io}}{\mathrm{gm1\_1}} + (\mathrm{VTn} - \mathrm{VTp}) - \left(1 - \frac{\mathrm{n}}{2}\right)\mathrm{Vds1\_1} - \\ - \sqrt{\frac{2\mathrm{nId}_{\mathrm{Md1}}}{\beta_{\mathrm{Md1}}}} - \frac{1}{\mathrm{n}}\sqrt{\frac{2\mathrm{nId}_{\mathrm{Md5}}}{\beta_{\mathrm{Md5}}}} \,. \tag{3.26}$$

The latter upper limit is also limited by the sum of the voltage drops along the branch corresponding to the triode-transistor M1_1 through the differential amplifier DA1 (see Figure 3.13).

3.1.4.2 Systematic Errors in the CFMF Type-II.

In contrast to the CFMF Type-I, this one is sensitive to the kind of systematic error represented in Figure 3.9 a). In Figure 3.13, due to the uncascoded branch of each single transconductor, the Early effect in M2_1 and M2_2 makes the crossover points shift. As a result, the expressions of the parameters xo and 2d defined in Figure 3.14 must be updated including the channel length modulation effect in transistors M2_1 and M2_2. Therefore, for the position of the membership function we have:

$$\mathrm{xo'} = \frac{\mathrm{Vk1} + \mathrm{Vk2}}{2} + \frac{1}{2}\left(\sqrt{\frac{\mathrm{nIo}}{\beta_{\mathrm{M2\_2}}(1 + \lambda(\mathrm{Vdd} - \mathrm{V2}))}} - \right. \\ \left. - \sqrt{\frac{\mathrm{nIo}}{\beta_{\mathrm{M2\_1}}(1 + \lambda(\mathrm{Vdd} - \mathrm{V1}))}}\right), \tag{3.27}$$

where V1 and V2, shown in Figure 3.13, are the common-source node voltages when both branches of the respective transconductors convey the same current. The modified width of the fuzzifier is given by:

$$\mathrm{2d'} = \mathrm{Vk1} - \mathrm{Vk2} + \frac{\mathrm{Io}}{\mathrm{gm1\_1}} + \mathrm{nVds1\_1} - \\ - \left(\sqrt{\frac{\mathrm{nIo}}{\beta_{\mathrm{M2\_2}}(1 + \lambda(\mathrm{Vdd} - \mathrm{V2}))}} + \sqrt{\frac{\mathrm{nIo}}{\beta_{\mathrm{M2\_1}}(1 + \lambda(\mathrm{Vdd} - \mathrm{V1}))}}\right). \tag{3.28}$$

In addition, in the above expression, Vds1_1 is the drain-to-source voltage drop of triode transistors in both transconductors.

It is difficult to get closed form expressions for V1 and V2. However, some useful conclusions by simple inspection of (3.27) and (3.28) can be drawn. Owing to the Early effect of transistors M2_1 and M2_2 the position of the CFMF is shifted left whereas its width gets smaller. However, as long as the CFMF is being allocated at higher values on the input Universe of Discourse both errors in width and position are diminished.

Concerning the other inaccuracy introduced by the Early effect at current source Io (see Figure 3.10) the consequences are the same than in the former CFMF. Expressions (3.14), (3.15) are also suitable for the correction that must be introduced on the position and the width of the CFMF. In this case however, for the same triode transistor sizes, slopes are doubled with respect to the CFMF Type-I. As a result, we have:

$$xo' = xo + \Delta xo = \frac{Vk1 + Vk2}{2} + \frac{go(Vk1 - Vk2)}{4gm1\_1}, \tag{3.29}$$

$$d' = d + \Delta d = \frac{Vk1 - Vk2}{2} + go\left(\frac{\Delta V}{2gm1\_1} + \frac{Vk1 - Vk2}{4gm1\_1}\right). \tag{3.30}$$

Therefore, the total systematic errors for the position xo and width d are calculated as the sum of the errors given by (3.27) to (3.30).

To conclude this section, let us consider the systematic errors in the slopes introduced by the loop differential amplifiers. Notice that equations (3.16) and (3.17) are also applicable in this case. However, considering the left transconductor in Figure 3.13, (Vd4–Vd3) in (3.18) is given, for this circuit, by:

$$Vd4 - Vd3 = (2VTn - Vk1) + \sqrt{\frac{2nIp}{\beta_{Md6}}} - \sqrt{\frac{nIo}{\beta_{Mc1\_1}}}\left(1 - \sqrt{\frac{\beta_{Mc1\_1}}{\beta_{M2\_1}}}\right) + \\ + \frac{gm1\_1}{\beta_{M1\_1}}\left[\left(\sqrt{\frac{\beta_{Md4}}{\beta_{Md2}}} - \sqrt{\frac{\beta_{Md4}}{\beta_{Md1}}}\right)^{-1} - 1\right]. \tag{3.31}$$

In the above equation, one must assume that Vin is set to the value corresponding to the crossover point (i.e. I1_1=Io/2) whereas the PMOST and NMOST threshold voltages values are identical. Notice the dependence of this error on the placement of the CFMF along the range of Vin. Table 3.3 shows some typical systematic errors of this circuit for Vdd=5V.

Table 3.3. Systematic errors in the CFMF Type-II.

Δxo	Δd	ΔVds/Vds (1.5V < Vk < 4.5V)
3.5mV	12mV	0.03 < ΔVds/Vds < 0.085

3.1.4.3 Mismatch Errors in the CFMF Type-II.

Following the same procedure as described in section 3.1.3.3 and appealing to Figure 3.4, the variance of the current supplied for each transconductor of this CFMF can be expressed as:

$$\sigma^2_{Iout1} = \begin{cases} 0 & \text{if } Iou1 = 0, \\ \left(1+\dfrac{gm1}{gm2}\right)^{-2}\left[Iout1^2 \dfrac{\sigma^2_{\beta 1}}{\beta 1^2} + gm1^2(\sigma^2_{VTn1}+\sigma^2_{VTn2}) + \right. & \\ \quad + gd1^2\sigma^2_{Vds1} + \dfrac{1}{4}\left(\dfrac{n\, gm1\ gm2}{\beta 2}\right)^2 \dfrac{\sigma^2_{\beta 2}}{\beta 2^2} + & \\ \quad \left. + \left(\dfrac{gm1}{gm2}\right)^2 \sigma^2_{Io} \right] & \text{if } 0 < Iout1 < Io, \\ \sigma^2_{Io} & \text{if } Iout1 = Io. \end{cases} \tag{3.32}$$

In the above equation, σ^2_{Io} and σ^2_{Vds1} are also given by (3.22) and (3.23), respectively. Simulation results concerning the mismatch errors of this fuzzifier are shown in Figure 3.15 a) and b). Finally, the variance of the position xo and width 2d are given by:

$$\sigma^2_d = \sigma^2_{xo} = \frac{1}{8}\left(\frac{1}{gm2}-\frac{1}{gm1}\right)^2 \sigma^2_{Io} + \frac{Io^2}{8gm1^2}\frac{\sigma^2_{\beta 1}}{\beta 1^2} + \frac{Io^2}{8gm2^2}\frac{\sigma^2_{\beta 2}}{\beta 2^2} + \\ + \frac{1}{8}\left(\frac{Io}{gm1\ Vds1} - n\right)^2 \sigma^2_{Vds1} + \frac{1}{2}\sigma^2_{VTn1} + \frac{1}{2}\sigma^2_{VTn2}\,. \tag{3.33}$$

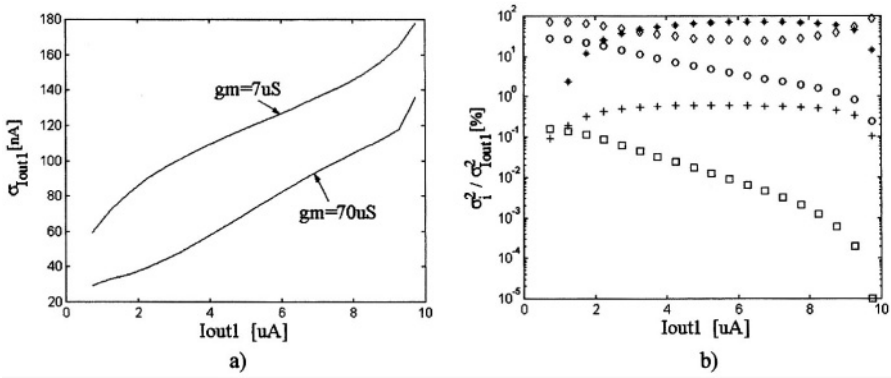


Figure 3.15. a) Standard deviation of the output current of the CFMF Type-II for two different transconductance values. b) Individual contribution of each random variable in (3.32) to the variance of Iout1 for gm=7μS: (+) $\sigma_{\beta 1}$; (o) σ_{VT1}; (*) σ_{Vds1}; (□) $\sigma_{\beta 2}$ and (◊) σ_{Io}.

3.1.5 Compact Fuzzy Partition Circuits

So far, we have discussed and analyzed FMF (or CFMF) circuits as isolated operator blocks. Normally, in a programmable Fuzzy Controller, individual FMFs (or CFMFs) are combined building up a bank of fuzzifiers. As explained in Chapter 2, each FMF in the bank can be shared by different rules. A degree of reconfigurability would be also attainable if the input of each FMF is also left independently accessible. This arrangement provides the maximum of flexibility in terms of number of inputs and fuzzifiers (FMFs or CFMFs) per input for any particular application.

Many applications require only a fixed number of inputs and FMFs per input without loosing the programmability capability of the fuzzifiers (i.e. slopes and position). In such a case, a multiple-output compact fuzzy partition circuit for each input is better suited than a bank of individual membership functions (or fuzzy labels). In this way, a considerable saving in silicon area and current consumption may be achievable.

In [CoCr99], a current-mode fuzzy partition circuit is presented. Slopes and widths of the membership function are electrically tunable. However, one main drawback is the need of voltage-to-current converters for the inputs. Moreover, the circuit exploits the exponential relationships of MOSFETs biased in weak inversion. Consequently, no high processing speed may be expected from this particular implementation.

The VLSI-oriented idea presented at [WiJa96] satisfies the input interface requirements. It consists of several stacked differential pairs each one performing a bell-shaped fuzzy label at one of its outputs (i.e. outputs are currents). However, electrical slopes tuning is not easily affordable and they must be fixed at the mask level by adequately sizing the transistors of

the stacked differential pairs. On the other hand, as a consequence of the totem architecture of the circuit, low-voltage applications may constrain the minimum size of the transistors to be used and/or the maximum number of labels per input to be allocated. Thus, degradation on the resolution of the input space partition and/or a loss of flexibility for the choice of the FMF's slopes may be expected, unless a higher voltage supply source is used. For instance, in [WiJa96] a 10V power supply has been needed for a 7-label fuzzy partition circuit.

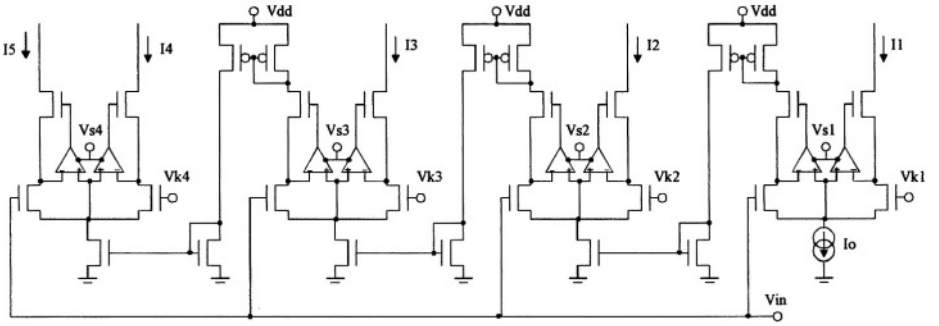


Figure 3.16. Full electrically-programmable five-label fuzzy partition circuit built from differential triode transconductors.

Based on [WiJa96], but making use of our triode transconductors, we suggest in Figure 3.16 a fully-electrically programmable 5-label fuzzy partition circuit. Each current I1…15 represents a direct fuzzy label. Voltages Vk1…Vk4 (Vk1<Vk2<…<Vk4) determine the crossover points between the contiguous fuzzy labels. In this way, all fuzzy labels I1…15 are generated when Vin is swept along its range. In addition, the slopes of the FMFs can be electrically programmed by voltages Vs1 …Vs4.

Figure 3.17 shows the simulated DC transfer characteristics of a fuzzy partition circuit with seven fuzzy labels distributed along a 3V input voltage range.

This fuzzy partition is performed by means of direct FMFs. Only six transconductors are needed. Notice that trying to synthesize this input space partition with individual FMFs, fourteen transconductors would be necessary. Furthermore, the current consumption would raise to 14Io (excluding the consumption of the loop differential amplifiers). In contrast, in the fuzzy partition circuit the current consumption depends on the value of the input voltage Vin, which range from Io to 11Io. Assuming Vin as a random variable uniformly distributed along its input range, a mean current consumption of 6Io should be attainable.

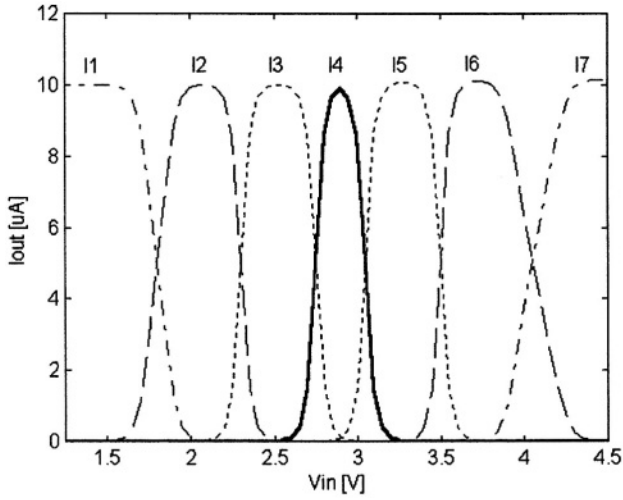


Figure 3.17. SPICE simulation of a 7-label fuzzy partition circuit built from differential triode transconductors.

Much effort must be paid however, in minimizing systematic and mismatch errors in the PMOS and NMOS mirrors conveying the bias current Io from one cell in Figure 3.16 to the adjacent one, specially when no tuning facilities are provided (i.e. dedicated controllers). This design strategy must be carefully followed when a large number of fuzzy labels is being intended, assuming one tries to minimize the cumulative mirroring error at the last cell. On the other hand, due to the cascade interconnection between the transconductors, a relative higher delay must be expected with respect to the individual fuzzifiers arrangement, specially for large input signal sweeps.

3.2 T-Norm and T-CoNorm Operators

Triangular Norms (T-Norms) are intermediate fuzzy operators taking care of the computation of the firing degree of the rules in a fuzzy inference scheme. The Triangular CoNorm (or S-Norm) is the dual operator of the T-Norm performing the opposite logical operation. Thus, by applying De Morgan's law, it is easy to transform a T-CoNorm into a T-Norm by a simple complementation of their inputs and outputs. Moreover, if the membership functions preceding this operation deliver straight complemented outputs signal (CFMFs), the implementation of the T-Norm using T-CoNorm is further simplified [VaVi99] [DuVe00] [GuPe96] [SaIn90]. In this case, one only must complement the output signal of the T-CoNorm.

In the most general case, there is one T-Norm (or T-CoNorm) per rule connecting N Fuzzy Membership Functions outputs, where N stands for the number of inputs of the controller. For this reason, circuits allowing multiple inputs are well adapted for multiple-input Fuzzy Controllers. One important issue to consider while designing multiple-input operators concerns its complexity as a function of the number of inputs N. A naive approach to design a N-input T-Norm or T-CoNorm consists in building a binary tree by cascading 2-input operators [YaMi86] [LiHu93]. However, while the complexity of the circuit is O(2N-1), the total input-output delay grows proportionally to $\log_2(N)$ [Dual94]. In [SaIn90], in an attempt to avoid binary trees, Multiple-Input T-Norm (MIN) and T-CoNorm (MAX) operators are suggested. But the proposed topologies have $O(N^2)$ complexity whereas the input capacitance at each input is proportional to the number of inputs N. Thus, when N becomes large enough, the performances in terms of circuit size, current consumption and total delay decrease, when compared with a binary tree with the same number of inputs [Dual94].

From the above discussion it turns out that the main criterions that should be taken into account while designing these operators are:

- Parallelism: cascading unit cells should be avoided, and the same input-output delay obtained from every circuit input.
- Inputs transparency: each circuit input must represent the same load to any input signal, independently of N.
- O(N) complexity: the size and the current consumption of the circuit should be proportional to the number of inputs N.

Nevertheless, in particular cases some of the above general guidelines could be ignored if the simplicity of the resulting circuit justifies it.

On one hand, the membership functions preceding the T-Norms or T-CoNorms deliver current signals. On the other hand, it will be shown later that the stage following the T-Norm or T-CoNorm (the aggregation of the rule's consequents) can be easily implemented in current domain. It turns out that current mode is a practical choice for the input-output interface requirements of these circuits. In this way, intermediate current-to-voltage or voltage-to-current converters can be avoided.

The most widely operators used for the T-Norm and the T-CoNorm are the MINIMUM and MAXIMUM functions respectively. Their electrical implementation is based on multiple inputs rectifiers-like schemes, which are easily achieved using CMOS transistors. Other T-Norm operators like those presented in Chapter 2 perhaps yield better function approximations than the "hard-switch" MAX or MIN operators do. However, they are not easy to implement following the guidelines discussed above.

In this section, two circuits performing the MAXIMUM operation are presented. The first one is a modified version of another classical MAXIMUM circuit, where accuracy and time response have been improved. In a second MAXIMUM circuit presented, input signals are voltages that are generated by the current delivered by a membership function through a diode-connected transistor. In this way, by means of a simple wire, the same signal coming from one membership function can be supplied to more MAXIMUM circuits, which belong to different rules. The use of this MAXIMUM is encouraged in the cases where the same membership function is being shared by a large number of rules. With this circuit, one can avoid the use of multiple current mirrors to distribute the output current of a membership function among the different rules.

Considering the lack of direct multiple-input MINIMUM circuits working in current mode, we have also investigated the feasibility of this kind of circuits. In this respect, two new multiple-input MINIMUM circuits are presented. One of them holds O(N) complexity [DoDu00] while the other $O(N^2)$ [DuVe01]. The latter results in a very simple circuit when a small fan-in is required (up to 3 or 4 inputs), which is frequently the case in Fuzzy Controllers.

3.2.1 Lazzaro's Winner-Take-All and MAXIMUM

Figure 3.18 a) illustrates the widely used N-input WTA-MAXIMUM Lazzaro's circuit [LaRy89]. It consists of N identical current-controlled voltage sources (M1 and M2) connected in parallel. Consider first each controlled voltage source isolated and assume its transistors working in strong inversion. The controlled voltage is the voltage at the source of transistor M1 and its value is proportional to the square root of its controlling current (i.e. I1…IN). However, in the circuit of Figure 3.18 a) the source terminal of transistors M1 are tied to a common diode-connected transistor Mo, which is equally sized to transistors M2 of the cells. In this way, all current-controlled voltage sources are connected in parallel and they fight to impose their own voltage at the common node. The one holding the largest voltage wins while switching off transistors M1 of the remaining cells. In this way, the winner cell together with the output common-diode Mo remains configured as a Wilson current mirror, which replicates the controlling current of the winner cell (i.e. the maximum input current) in Mo. As a result, transistors M2 in the loser cells remain in the triode region featuring a small drain-to-source voltage drop.

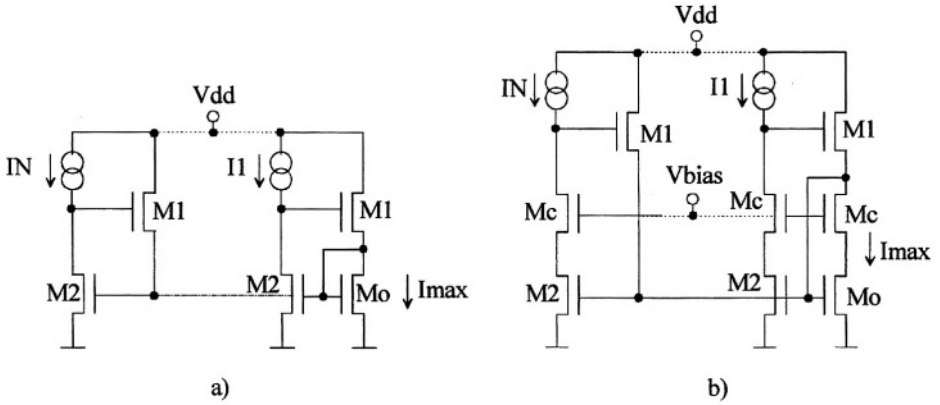


Figure 3.18. a) Lazzaro's WTA-MAXIMUM [LaRy89]. b) Improved version with cascoded branches.

Figure 3.18 b) shows a modified version of this circuit where transistors M2 have been cascoded by transistors Mc. These are properly biased with Vbias in order to warrant the saturation of transistors M2 and Mo. In this way, the winner current is more accurately mirrored to the output. The size and consumption of this circuit rise proportionally to N. From Figure 3.18 a) and b) we can realize that any input represents the same load to the signals I1…IN. However, as long as more cells are being added, the output load at the common node (the drain of Mo) is increased since more M2's Cgs capacitances are being lumped in parallel. Thus, the speed of the circuit may be considerably influenced by the number of inputs [Vida96].

3.2.1.1 Systematic Errors in the WTA-MAXIMUM.

The first kind of systematic error of this circuit is associated to its discrimination capability. For instance, for a 2-input MAXIMUM circuit, the resolution of the circuit (or discrimination error) is given by the minimum difference necessary between the two input currents that allows the circuit to output the maximum.

In order to quantify the above defined resolution a 2-input MAX circuit is assumed and is shown in Figure 3.19 b). In this figure, input I2 is held constant while I1 ranges from 0 to a value far above I2. The evolution of the output current Imax at the common-diode Mo is shown in Figure 3.19 a) as a function of I1, while I2 is a horizontal line [Vida96]. If the circuit was ideal, Imax would follow the well-defined broken-line path, being equal to I2 when I1 is the smallest current and vice versa.

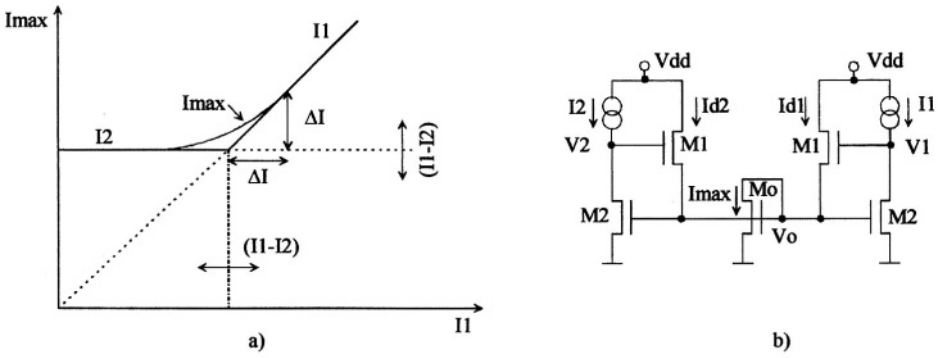


Figure 3.19. a) Graphical interpretation of the discrimination error of Lazzaro's circuit [Vida96]. b) 2-input MAX circuit used to quantify the discrimination error.

In a real circuit, Imax exhibits a different behavior when its inputs take close values as represented by the curved line in the neighborhood of the 11=12 point. For the output Imax to match exactly the maximum input (in this case I1), I1 should overpass I2 by a little quantity called ΔI in Figure 3.19 a). This ΔI is the so-called absolute discrimination error or resolution of the circuit. Figure 3.19 b) enables us to estimate the difference between I1 and I2 as follows:

$$I1 - I2 = \Delta I = \lambda n\,(V1 - V2)\,Imax = \lambda n \sqrt{\frac{2n}{\beta 1}}\left(\sqrt{Id1} - \sqrt{Id2}\right)Imax\,, \quad (3.34)$$

where λn is the inverse of the transistors M2 Early voltage and all transistors are biased in strong inversion. Therefore, an expression for ΔI can be found from (3.34) when Imax, and consequently Id1, becomes equal to I1 while Id2 is zero. Assuming $I1 \approx Imax$, the relative discrimination error is equal to:

$$\frac{\Delta I}{I1} = \frac{\Delta I}{Imax} = \lambda n \sqrt{\frac{2n}{\beta 1}}\sqrt{Imax} = 2\lambda n\left(\frac{gm}{I}\right)^{-1}_{M1}. \quad (3.35)$$

It is clear from the latter equation that either long-channel transistors M2 or transistors M1 working in moderate to weak inversion should improve this error figure. However, to get also high-speed behavior, strong inversion of M1 is recommended. Thus, we prefer to improve the discrimination of the MAXIMUM circuit by cascoding transistors M2, which reduces considerably the Early effect. For this case, equation (3.35) still remains

(approximately) valid if λn is replaced by its equivalent expression after including the cascoding effect of transistors Mc:

$$\lambda n_{cas} = \left(\frac{gd_{Mc}}{gm_{Mc}} \right) \lambda n . \tag{3.36}$$

Another kind of systematic error is the propagation error or simply relative error. This error figure intends to quantify the difference between the maximum winner input and the actual output current value. In this circuit, this error is related to the quality of the mirroring performed by transistor M2 of the winner cell and the common-diode Mo in Figure 3.18 a). Since both transistors hold different drain-to-source voltage drops, the Early effect generates a difference between input and output currents. Considering I1 as the winner, the relative value of this error can be approximated as follows:

$$\varepsilon = \frac{I1 - Imax}{I1} = \lambda n \left(VTn + \sqrt{\frac{2nImax}{\beta 1}} \right), \tag{3.37}$$

where the expression within the parenthesis represents the Vgs voltage drop of transistor M1. Therefore, this error can also be confined to a very small value by using the proposed cascode configuration.

3.2.1.2 Mismatch Errors in the WTA-MAXIMUM.

Considering the circuit in Figure 3.18 b) we now focus on the error introduced by the random mismatches between different transistors. While the circuit propagates the maximum input current, the winner cell works as a current mirror. Thus, the estimation of the mismatch error is reduced to investigate the variance of the output current Imax as a function of the variance of the threshold voltages VT and the current gains β of transistors Mc, M2 and Mo. For this purpose, we assume that VT and β of all transistors in the circuit are statistically independent random variables.

Since the mirroring is actually performed by the bottom transistors M2 and Mo, we can ignore the effect of the mismatch of the cascoding transistors Mc. This is because the latter will only produce a tiny disparity of the drain voltages of transistors M2 and Mo. This has no significant influence on the drain currents of the latter transistors since they are saturated. Thus, the relative mismatch error of the output current is given by:

$$\frac{\sigma^2_{Imax}}{Imax^2} = 2\left(\frac{\sigma^2_{\beta M2}}{\beta^2_{M2}} + \left(\frac{gm}{I}\right)^2_{M2} \sigma^2_{VTn_{M2}}\right). \quad (3.38)$$

Table 3.4 shows some typical values of the systematic and mismatch errors of this circuit for two different values of the output current Imax.

Table 3.4. Systematic and mismatch errors of the Lazzaro's WTA-MAXIMUM for Vdd=5V.

Imax	Discrimination Error	Propagation Error	Mismatch Error
5μA	0.47%	1.4%	0.85%
10μA	0.66%	1.6%	0.6%

3.2.2 Mixed-Mode Multiple-Input MAXIMUM

The circuit shown in Figure 3.20 actually performs the MINIMUM operation in voltage mode [CaRa00]. It consists of a set of N source-follower cells, each one comprising an amplifier A and a PMOS transistor M2 whose sources share the common-node C. The input signals are the voltages at the non-inverting inputs of the amplifiers A. The output signal is the voltage of the common-node C. Considering that the source followers are built by PMOS transistors, the common-node C will follow the minimum input voltage. Therefore, transistors M2 of the other cells with higher input voltages are switched off. It is easy to understand that the specially adapted scheme in Figure 3.20 can also calculate the MAXIMUM between the input currents I1…IN: the maximum current yields indeed the minimum input voltage at the gate of the corresponding transistor M1. Therefore, if we consider that transistors M1 and Mpc are equally sized and neglect the finite gain and the offset of the amplifiers A, the maximum current is mirrored in Mpc. This maximum current can also be recovered through the NMOS common transistor Mnc, which is also connected as diode.

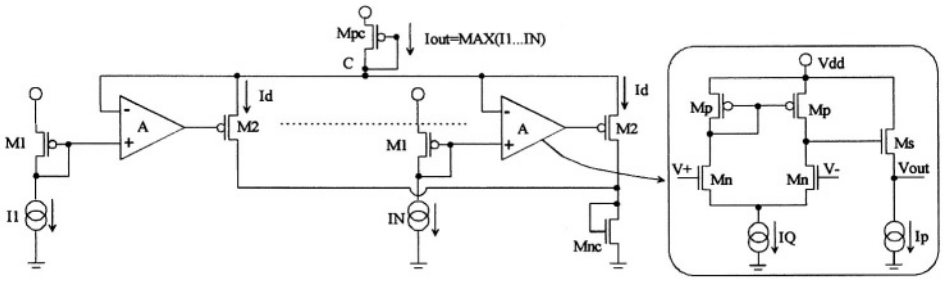


Figure 3.20. Mixed-mode N-input MAXIMUM and operational amplifier A of each cell.

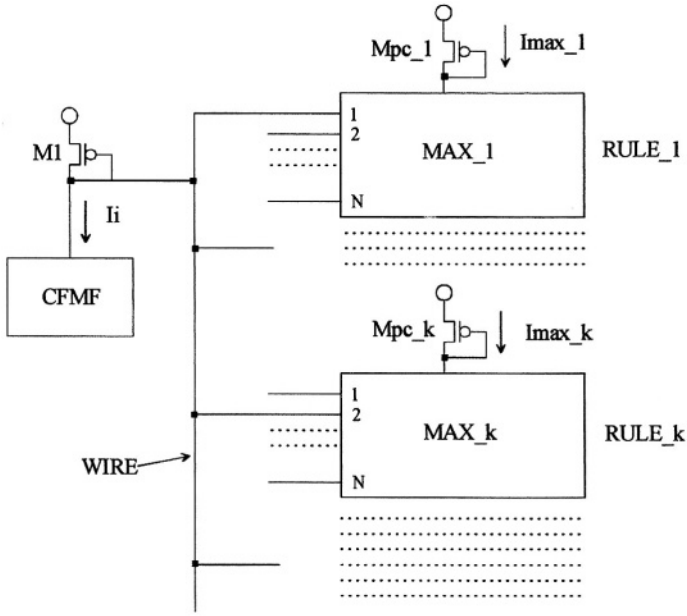


Figure 3.21. Distribution of the input signals in the Mixed-Mode MAXIMUM circuit: a CFMF is shared by several rules.

The circuit of the amplifiers A is also shown in Figure 3.20. It comprises a differential amplifier (i.e. 2×Mn, 2×Mp, IQ) followed by a NMOS level shifter (i.e. Ms, Ip). This is needed to avoid that transistor Mn at the inverting input gets out the saturation region [CaRa00].

The use of this circuit as T-CoNorm is convenient when the same input signal Ii (assumed to be the output of a Complementary Fuzzy Membership Function circuit) takes part in several MAX operators corresponding to different rules. This situation is illustrated in Figure 3.21 where each MAX block comprises the set of N amplifiers A and transistors M2 as well as the common transistor Mnc, as shown in Figure 3.20. In this way, the signal distribution at the inputs of the MAXs can be performed in the voltage domain by means of a simple wire without loosing the current-mode operation of the circuit. In addition, the use of multiple current mirrors for the distribution of the signals Ii (for i=1…N) is avoided. Finally, we will show later that the stage following these MAX operators (the defuzzifier) performs optimally in the current domain. Consequently, keeping the output of this operator working in current mode is also desirable. This is the reason why we call this circuit mixed-mode maximum.

3.2.2.1 Systematic Errors in the Mixed-Mode MAXIMUM.

To investigate the discrimination capabilities of this circuit we consider a 2-input MAXIMUM. Let us assume that the values of the input currents (I1, I2) corresponding to each cell are close enough. In such a case, transistors M2 of both cells are switched on and convey the drain currents Id1 and Id2 (see currents Id in Figure 3.20). Calling Ao the DC amplifiers gain, the difference between the two input currents is given by:

$$I1 - I2 = \Delta I = \frac{2}{Ao}\sqrt{\frac{\beta 1}{\beta 2}}\left(\sqrt{Id1} - \sqrt{Id2}\right)\sqrt{Id1 + Id2}\,. \tag{3.39}$$

From the above equation we can estimate the absolute resolution ΔI by assuming, for instance, that Iout=Id1=I1 when Id2=0. Then, the relative discrimination error is given by:

$$\frac{\Delta I}{I1} = \frac{\Delta I}{Iout} = \frac{2}{Ao}\sqrt{\frac{\beta 1}{\beta 2}}\,. \tag{3.40}$$

As should be expected the gain of the amplifiers plays an important role in the discrimination. The larger the gain, the better the resolution because the amplifiers can react to smaller voltage differences at their inputs caused by smaller input currents differences.

Let us consider now the propagation error. Suppose that the first input cell in Figure 3.20 conveys the highest current (i.e. I1). Therefore, taking into account the finite gain of the amplifier Ao and its offset Voff, this error is given by (3.41). Notice from this equation the incidence of the systematic offset of the amplifier A. An expression for Voff is derived from [LaSa94], as expressed by (3.42). In the latter, the letters n, p and s make reference to transistors Mn, Mp and Ms belonging to the amplifiers A in Figure 3.20, respectively.

$$\varepsilon = \frac{I1 - Iout}{I1} = \frac{\left(Voff + \sqrt{\frac{2nI1}{\beta 1}} - \frac{|VTp|}{Ao}\right)^2}{\frac{2nI1}{\beta 1}\left(\frac{1}{Ao} + 1\right)^2} - 1, \tag{3.41}$$

$$Voff = \frac{gd_{n}+gd_{p}}{gm_{n}}\left(\sqrt{\frac{nIQ}{\beta p}}+\sqrt{\frac{2nIp}{\beta s}}-\sqrt{2nI1}\left(\frac{1}{\sqrt{\beta n}}+\frac{1}{\sqrt{\beta 1}}\right)\right). \quad (3.42)$$

3.2.2.2 Mismatch Errors in the Mixed-Mode MAXIMUM.

If we consider a relatively high gain Ao of the amplifiers A while assuming a mismatch between transistors M1 and Mpc, the output current Iout can be expressed as a function of the winner input current I1:

$$Iout = \frac{\beta pc}{2n}\left(|VTp1|-|VTppc|+Voff+\sqrt{\frac{2nI1}{\beta 1}}\right)^{2}, \quad (3.43)$$

where VTp1 and VTppc are the threshold voltages of the above mentioned transistors. From this equation, we can find the expression for the relative variance of the output current. Considering that under nominal conditions $\sigma_{VTp1}=\sigma_{VTppc}$ (remember that $(W/L)_{M1}=(W/L)_{Mpc}$) the relative variance of Iout is given by:

$$\frac{\sigma^{2}_{Iout}}{Iout^{2}} = 2\left(\left(\frac{gm}{I}\right)_{M1}+\frac{\beta 1\, Voff}{nIout}\right)^{2}\left(\sigma^{2}_{VTp1}+\frac{1}{2}\sigma^{2}_{Voff}\right)+ \\ +\left(\frac{1}{\sqrt{\beta 1}}+\frac{Voff}{\sqrt{2nIout}}\right)^{2}\left(\left(\frac{1}{\sqrt{\beta 1}}+\frac{Voff}{\sqrt{2nIout}}\right)^{2}+\frac{1}{\beta 1}\right)\sigma^{2}_{\beta 1}, \quad (3.44)$$

where the variance of the amplifiers offset Voff is derived from [LaSa94]:

$$\sigma^{2}_{Voff} = 2\sigma^{2}_{VTn}+2\left(\frac{gm_{n}}{gm_{p}}\right)^{2}\sigma^{2}_{VTp}+\left(\frac{IQ}{gm_{p}}\right)^{2}\frac{\sigma^{2}_{\beta p}}{\beta^{2}_{p}}+\left(\frac{IQ}{gm_{n}}\right)^{2}\frac{\sigma^{2}_{\beta n}}{\beta^{2}_{n}}. \quad (3.45)$$

Notice the strong influence of σ_{Voff} on the relative standard deviation of the output current Iout. As a conclusion, biasing M1 and Mpc in strong inversion will improve the mismatch error if considered that these transistors mirror the winner current. Table 3.5 shows some typical values of the systematic and mismatch errors of this MAXIMUM.

Table 3.5. Systematic and mismatch errors of the Mixed-Mode MAXIMUM for Vdd=5V.

Iout	Discrimination Error	Propagation Error	Mismatch Error
5μA	1.7%	2.05%	3%
10μA	1.7%	2.3%	2.3%

3.2.3 New Multiple-Input O(N²) LTA-MINIMUM

In the circuit depicted in Figure 3.22 a) all transistors Mp and Mn are equally sized with respect to their homologues at each layer (row). The output branch (column) at the right comprises the PMOS transistors Mp3, Mp6 and a NMOS diode-connected transistor Mc. The circuit makes a comparison between the input currents I1 and I2 [DuVe01]. The smaller input current is mirrored to the other input branch whose corresponding transistor Mn is driven to the triode region. The minimum input current is also mirrored to the output branch and can be recovered through the NMOS diode Mc. Additionally, the drain voltages drops Vo1, Vo2 of transistors Mn1, Mn2 can be used to point out the minimum input current. Therefore, the circuit can be also used as a LTA (Loser-Take-All).

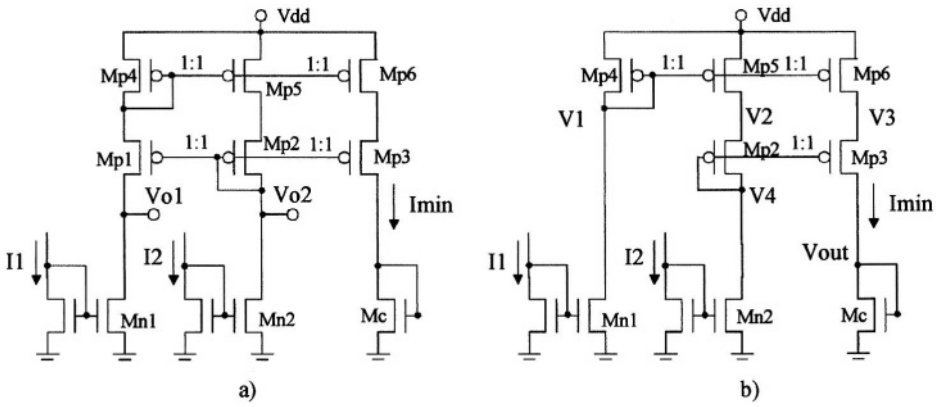


Figure 3.22. a) 2-input MINIMUM-LTA circuit with $O(N^2)$ complexity. b) 2-input MINIMUM circuit with $O(N^2/2+N)$ complexity.

From Figure 3.22 a), a multiple-input MINIMUM can be built by adding branches (i.e. columns) with stacked transistors Mp. The circuit has $O(N^2)$ complexity, however, and the maximum number of inputs is limited by the value of Vdd. In Figure 3.22 b) a simplified version is depicted. It performs only the MINIMUM operation while holding a slightly reduced complexity (i.e. $O(N^2/2+N)$).

In both circuits, owing to the lack of symmetry, there is no input transparency and the performance in terms of accuracy and speed depends

on which input is being considered. Nevertheless, since the circuit performs an exhaustive comparison between all of their inputs simultaneously, the parallel processing mode is still sustained. Furthermore, for small fan-in, which is frequently the case of Fuzzy Controllers, the use of this circuit is justified. This is due to its simplicity, when compared with other MINIMUM circuits even featuring O(N) complexity [DoDu00] [KeSc93].

3.2.3.1 Systematic Errors in the O(N²) LTA-MINIMUM.

The propagation error of this cell is also related to the accuracy within the minimum input current is being mirrored to the output branch of the circuit. As mentioned before, the asymmetry of this architecture imposes to perform an error analysis input by input. To get an idea about its accuracy we have evaluated the systematic errors in the alternative shown in Figure 3.22 b), assuming also two inputs.

With reference to the latter mentioned figure, when I1 is smaller than I2, I1 is mirrored to the central branch by Mp5 and to the output branch by Mp6. Thus, the voltages V2 and V4 fall abruptly due to the unsaturated condition of the transistor Mn2. V3 follows the same tendency since it is the source voltage of a PMOS transistor whose gate is driven by V4. Consequently, the difference between the minimum input current I1 and the actual output current Imin is caused by the Early effect in the PMOS transistor Mp6 as a result of the difference between V3 and V1. Thus, the relative error is given by:

$$\frac{I1 - Imin}{I1} = \lambda p \left(2|VTp| + 2\sqrt{\frac{2nI1}{\beta_{Mp}}} + \frac{1}{n}\sqrt{\frac{2n}{\beta_{Mn}}}\left(\sqrt{I2} - \sqrt{I2 - I1}\right) - Vdd \right)$$
$$= \lambda p\,(V3 - V1)\,. \qquad (3.46)$$

Consider now the opposite case where I2 is smaller than I1. Transistors Mp5 and Mp6 in the middle and in the output branches will be in triode region holding identical drain voltage drops. Therefore, since the mirroring of I2 (the minimum) to the output branch is actually performed by the saturated transistors Mp2 in the middle branch, the propagation error is a consequence of the difference between V4 and the drain voltage Vout of Mp3 (see Figure 3.22 b)). Consequently, we have:

$$\frac{I2 - Imin}{I2} = \lambda p\,(Vout - V4) = \lambda p\left(|VTp| + VTn + \sqrt{\frac{2nI2}{\beta_{Mc}}} + \right.$$
$$\left. + \sqrt{\frac{2nI2}{\beta_{Mp}}} + \frac{1}{n}\sqrt{\frac{2n}{\beta_{Mp}}}\left(\sqrt{I1} - \sqrt{I1 - I2}\right) - Vdd\right). \quad (3.47)$$

From the two latter equations, it can be noticed that the channel length modulation factor λp has a clear incidence on this error. Thus, larger Mp transistors lengths or low-compliance cascoded mirrors should be adopted.

The discrimination capacity of this circuit is associated to the impedance of the node V2 in Figure 3.22 b). Finding the absolute resolution ΔI necessary to discriminate two close input currents is equivalent to find the minimum currents difference ΔI needed at the central branch to produce the maximum voltage swing at the node V2. Therefore, a rough approximation for this error figure is given by:

$$\Delta I \approx \frac{Vdd - |VTp| - \sqrt{2nI2}\left(\frac{1}{\sqrt{\beta_{Mp}}} + \frac{1}{n}\frac{1}{\sqrt{\beta_{Mp}}} + \frac{1}{n}\frac{1}{\sqrt{\beta_{Mn}}}\right)}{2R_{V2}}, \quad (3.48)$$

where R_{V2} represents the small-signal resistance at the node V2. R_{V2} can be approximated as:

$$R_{V2} \approx \frac{1}{I2}\left(\frac{1}{VEAn} + \frac{1}{VEAp}\right)^{-1}, \quad (3.49)$$

where VEAn and VEAp are the Early voltages of transistors Mn2 and Mp5 at the central branch, respectively.

3.2.3.2 Mismatch Errors in the O(N²) LTA-MINIMUM.

With regards to Figure 3.22 b), when I1<I2, transistors Mp5 and Mp6 in the upper row of the circuit are saturated whereas the minimum input value (I1) is mirrored to the output. However, we have to consider also the mismatch introduced by the bottom NMOS current mirror (i.e. transistors Mn1). Hence:

$$\frac{\sigma^2_{Imin}}{Imin^2} = 2\left(\frac{\sigma^2_{\beta Mp}}{\beta^2_{Mp}} + \left(\frac{gm}{I}\right)^2_{Mp}\sigma^2_{VTp}\right) + 2\left(\frac{\sigma^2_{\beta Mn}}{\beta^2_{Mn}} + \left(\frac{gm}{I}\right)^2_{Mn}\sigma^2_{VTn}\right). \tag{3.50}$$

When I1>I2 transistors Mp5 and Mp6 in the upper row of the circuit in Figure 3.22 b) remain in the triode region. Let us call Vdp the drain-to-source voltage drop of those transistors. Thus, taking into account the mismatches of Vdp of transistors Mp5 and Mp6, an additional random component appears. Therefore, the modified expression for the relative variance of the output current is given by:

$$\frac{\sigma^2_{Imin}}{Imin^2} = 2\left(\frac{\sigma^2_{\beta Mp}}{\beta^2_{Mp}} + \left(\frac{gm}{I}\right)^2_{Mp}\left(\sigma^2_{VTp} + \sigma^2_{Vdp}\right)\right) + 2\left(\frac{\sigma^2_{\beta Mn}}{\beta^2_{Mn}} + \left(\frac{gm}{I}\right)^2_{Mn}\sigma^2_{VTn}\right). \tag{3.51}$$

However, since Vdp depends on the minimum input current I2=Imin, the difference equation used to calculate the variance of the output current Imin should be reformulated taking into account such dependence. Therefore, the relative variance of Imin is now given by:

$$\frac{\sigma^2_{Imin}}{Imin^2} = \frac{2}{\left(1+\frac{gmp}{gdp^*}\right)^2}\left[\left(1+\left(\frac{gmp}{gdp^*}\right)^2\right)\frac{\sigma^2_{\beta Mp}}{\beta^2_{Mp}} + \frac{\left(1+\left(\frac{gmp^*}{gdp^*}\right)^2\right)}{\left(\frac{gm}{I}\right)^{-2}_{Mp}}\sigma^2_{VTp}\right] + 2\left(\frac{\sigma^2_{\beta Mn}}{\beta^2_{Mn}} + \left(\frac{gm}{I}\right)^2_{Mn}\sigma^2_{VTn}\right), \tag{3.52}$$

where gdp* and gmp* are the conductance and the transconductance of the PMOS transistors in the ohmic region (i.e. Mp5, Mp6) respectively, whereas gmp is the transconductance of the other Mp transistors in saturation. To conclude, Table 3.6 shows some typical values of the systematic and mismatch errors of this circuit.

Table 3.6. Systematic and mismatch errors of the $O(N^2)$ LTA-MINIMUM for Vdd=5V.

Imin	Discrimination Error	Propagation Error	Mismatch Error
5μA	2.4%	2.3%	2.3%
10μA	2.8%	2.5%	1.8%

3.2.4 Novel Multiple-Input O(N) Complexity LTA-MINIMUM

A novel current-mode circuit for the MINIMUM-LTA computation [DoDu00] is presented in Figure 3.23 a). In this figure, the Cell-i is repeated as many times as the number of inputs is desired. Currents Ii in each Cell-i are the inputs of the circuit whereas the output Imin is retrieved at the common-cell branch (i.e. M4, Mp, Iop). Input currents Ii are illustrated by ideal current sources for clarity but, actually, they are implemented through a PMOS current-mirror depicted in Figure 3.23 c). In short, the circuit behaves as a set of current-controlled voltage sources (the Cells-i) that are connected in parallel to a common-node Nc and fight to impose their own voltages. However, in contrast to Lazzaro's circuit, due to the source-follower connected PMOS transistor M3 in each cell, the common-node Nc follows the lowest voltage source rather than the highest.

In each Cell-i the controlling loop is made up by the common-source connected transistor M2 driving the gate of transistor M3 through the source-follower transistor M1, which is biased by Ion. This stage is needed to adapt the DC level of the drain of M2 to the one required by the gate of M3.

When the input current Ii becomes small, the drain voltage of M2 falls while the gate and the source voltages of M3 follow the same trend. Thus, the cell with the smallest input current defines the voltage at the node Nc whereas transistors M3 in the other cells switch off. Consequently, the common current source Iop at the common cell will be sunk by the diode-connected transistor Mn corresponding to the Cell-i that conveys the smallest (minimum) input current Ii. As a result, either the current Ioi or the voltage Voi in Figure 3.23 a) can be used as digital output for the Looser-Take-All (LTA) operation. On the other hand, transistors Mpi (see Figure 3.23 c)) in all other cells conveying higher input currents remain in the triode region while holding a high drain voltage level. Furthermore, transistors M2 in all cells remain saturated while conveying the MINIMUM

input current. Therefore, the MINIMUM input current will be mirrored at the output by transistor M4 at the common cell.

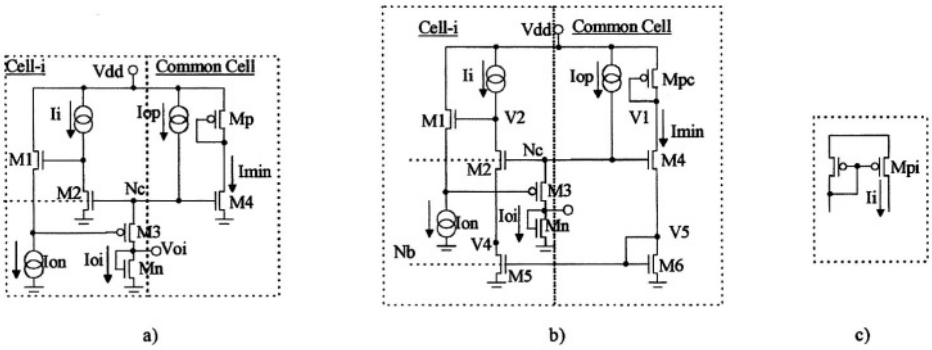


Figure 3.23. a) Multiple-input O(N) MINIMUM-LTA b) Improved version with current feedback. c) Actual input current Ii at each Cell-i. From [DoDu00], © 2000 IEEE.

In Figure 3.23 b) an improved version of this circuit is sketched. In this case, diode-connected transistor M6 at the common cell replicates the MINIMUM output current to all cells through mirrors M5. With this current feedback, the accuracy of the MINIMUM function becomes superior. This is because the common cell and the Cell-i that conveys the minimum input current remain configured as a variant of an Enhanced Wilson current mirror with cascoded output.

3.2.4.1 Systematic Errors in the O(N) Complexity LTA-MINIMUM.

Let us first considerer the propagation errors for the MINIMUM circuit of Figure 3.23 b). Let us assume that the Cell-i sketched in this figure conveys the MINIMUM input current Ii. In this case, the mirroring error is also due to the Early effect if we consider the different drain voltage V4 and V5 of transistors M5 and M6 respectively. Since the mirror is cascoded by M2 and M4, a small propagation error should be expected. Hence:

$$\frac{Ii - Imin}{Ii} = \lambda n\,(V5 - V4) = \lambda n^2 \left(\frac{gm}{I}\right)^{-1}_{M4} (V1 - V2), \qquad (3.53)$$

where (V1–V2) is equal to:

$$(V1 - V2) = Vdd - 3\,VTn + \sqrt{\frac{2nIop}{\beta_{M3}}} - \sqrt{\frac{2nIon}{\beta_{M1}}} + \\ + \sqrt{2nIi}\left(\frac{1}{\sqrt{\beta_{M6}}} - \frac{1}{\sqrt{\beta_{M4}}} - \frac{1}{\sqrt{\beta_{Mpc}}}\right). \tag{3.54}$$

According to (3.53), since λn appears squared, this error can be confined to a very small value. In order to find the discrimination error a two-input circuit is assumed. When both inputs are close enough their corresponding transistors M3 are switched on. Thus, the difference between the two input currents is given by:

$$I1 - I2 = \Delta I = Imin\ \frac{\lambda n\, gd_{M4}}{gm_{M4}}\left(\sqrt{Id_{M3\_1}} - \sqrt{Id_{M3\_2}}\right), \tag{3.55}$$

where Id_{M3_1} and Id_{M3_2} are the currents through each transistor M3. Consequently, the absolute discrimination error is given by:

$$\Delta I = 2\ \lambda n^2\ Imin\left(\frac{gm}{I}\right)^{-1}_{M4}\left(\frac{gm}{I}\right)^{-1}_{M3}. \tag{3.56}$$

Notice that the discrimination of the circuit is inversely proportional to the gm/I of transistors M3. Therefore, when a very accurate resolution is desired, biasing the latter transistor in moderate inversion is encouraged.

3.2.4.2 Mismatch Errors in the O(N) Complexity LTA-MINIMUM.

In this case the mismatch between the output current Imin and the minimum input current is also due to the mirroring mismatch errors inherent to a cascode mirror (i.e. M2, M4, M5 and M6 in Figure 3.23 b)). Hence:

$$\frac{\sigma^2_{Imin}}{Imin^2} = 2\left(\frac{\sigma^2_{\beta M5}}{\beta^2_{M5}} + \left(\frac{gm}{I}\right)^2_{M5}\sigma^2_{VTn\,M5}\right) + 2\left(\lambda n\left(\frac{gm}{I}\right)^{-1}_{M4}\right)^2 \cdot \\ \cdot\left(\frac{\sigma^2_{\beta M2}}{\beta^2_{M2}} + \left(\frac{gm}{I}\right)^2_{M2}\sigma^2_{VTn\,M2}\right). \tag{3.57}$$

Notice that the second term in (3.57) corresponds to the error due to mismatches in the cascoding transistors M2 and M4. It can thus be neglected. Table 3.7 shows the systematic and mismatch errors of this cell.

Table 3.7. Systematic and mismatch errors of the O(N) LTA-MINIMUM for Vdd=5V.

Imin	Discrimination Error	Propagation Error	Mismatch Error
5μA	0.06%	0.27%	1.7%
10μA	0.08%	0.4%	1.25%

3.3 Defuzzifiers

Several strategies have been proposed for the calculation of the defuzzified output value of Fuzzy Controllers. Analog Mamdani's defuzzifiers circuits have been reported in [BoTs97-98] [FaSh94] [PeRo93]. Because they demand high computational efforts, which make the defuzzifying stage slow compared to other implementations, they will not be considered for the implementation of our controllers. In contrast, Takagi-Sugeno's (TS) defuzzifier circuits are much easier to implement because they perform only a normalized weighted sum (or Averaged Weighted Sum). In the following discussion, we will only consider Zero-Order Takagi-Sugeno's controllers, which implies that the consequents of the rules are singletons. These consequent singletons will be generically called αi independently on how they are physically implemented. In addition, the firing degree of the rules, which can be represented by a current or a voltage signal, will be generically called Ii. According to the method used to achieve the normalization, these defuzzifiers could be classified differently.

In the first class of circuits that we call closed-loop defuzzifiers, normalization is achieved by using negative feedback. One implementation commonly used from the early stages of analog fuzzy hardware devices [TsIn95] [Land93] [MaFr96] [GuPe96] [RoPe97-b] operates in voltage mode and is illustrated in Figure 3.24 a). Assuming m rules, it comprises a set of transconductors, one per rule, sharing the output terminal Vo. From the output of the defuzzifier a negative feedback loop is ended at the inverting input of each transconductor. A reference voltage αi, representing the rule's consequent singleton, is applied to each non-inverting input. The firing degree of the rules is represented by the values of the transconductances Gi, which are controlled by signals Ii coming from the T-Norms to each corresponding transconductor. Depending on the implementation of the transconductors Gi, their controlling signal Ii may be a voltage or a current.

Another kind of closed-loop defuzzifier that works in current mode [RoPe97] [PaQu95] [SoQu98] is represented in Figure 3.24 b). In this case,

the singletons αi are scalar current multipliers (i.e. scaled current mirrors or current-mode D/A converters). The sum of the firing degrees of the rules Ii is forced to be constant and equal to the current reference Io thanks to a negative feedback loop. This feedback loop controls the gain of the previous stages, namely the T-Norms or the Fuzzy Membership Functions FMFs [RoPe96] [RoPe99]. Therefore, the currents Ii at the output of the T-Norm are yet normalized and, after being multiplied by their corresponding singletons αi, they are simply summed to yield the final defuzzified output value Iout.

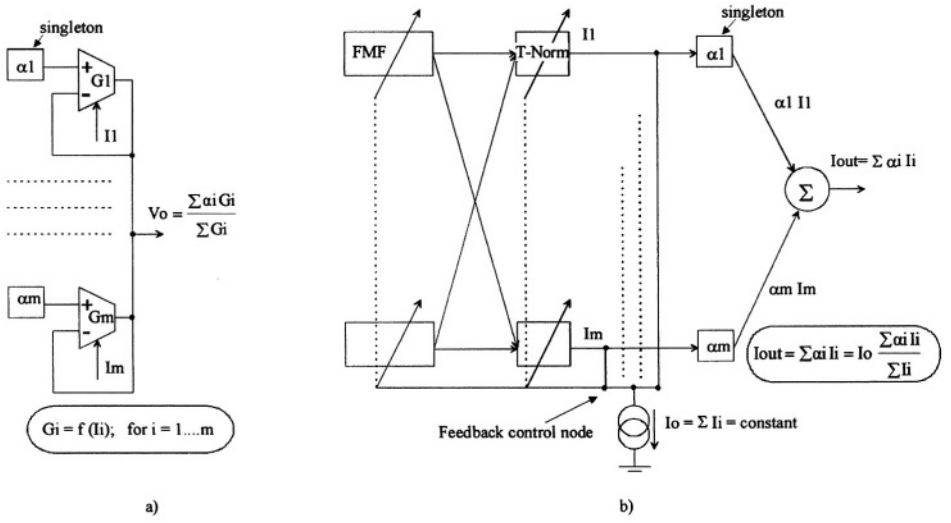


Figure 3.24. Closed-loop defuzzifiers [VaVi99]: a) Voltage mode. b) Current mode.

The second class of defuzzifiers performs normalization without the need for feedback; therefore, we call them open-loop defuzzifiers. In Figure 3.25 a) normalization is achieved by using a pseudo-normalizer [WiJa96] [VaVi99]. It comprises a set of pseudo-resistances Ri whose values are controlled by signals Ii that could be currents or voltages. The sum of the normalized signals IiN across Ri is then constrained to be constant and equal to the current reference Io. Assuming that signals IiN are currents, they are scaled by means of current mirrors whose scaling ratio represents the value of the singletons αi. After this, the weighted signals $\{\ldots \alpha i \times IiN \ldots\}$ are summed in the output common node to yield the defuzzified output value of the controller. This kind of normalization is actually non ideal. In a true normalizer the normalized signals IiN are equal to $IiN=Ii/(\sum Ii)$. In pseudo-normalizers however, those normalized signals are given by IiN=Ii/H(I1,...Ii,...Im), where H is a non-linear function of all unnormalized signals Ii. In order to be close to $(\sum Ii)$, the function H must be monotonously

increasing concurrently with any of its inputs I1,...Ii,...Im [VaVi99]. However, this non ideality does not represent a major drawback considering that the non-lineal functions that Fuzzy Controllers normally fit can also be approximated by means of non-linear operators.

The last kind of open-loop defuzzifier explicitly uses an analog divider as normalizer. This is illustrated in Figure 3.25 b) [KeSc93] [BaHu97] [BoTs97] [DuVe00]. In this way the firing signal Ii of each rule is replicated twice. The first replica is summed in a common node, which becomes the denominator input of the divider. The other replica of Ii, after being scaled by the singleton αi is summed in another common node, which becomes the numerator of the divider.

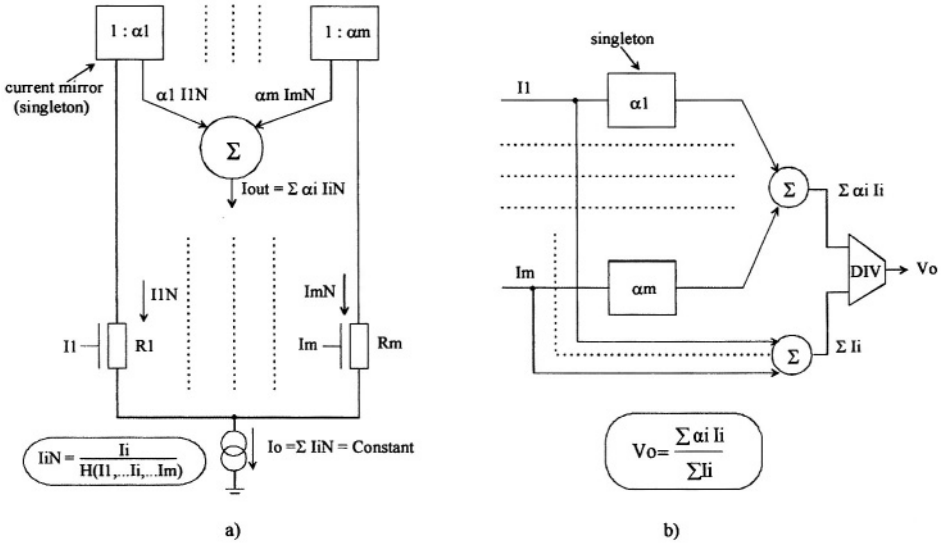


Figure 3.25. Open-loop defuzzifiers [VaVi99]: a) With pseudo-normalizer. b) With divider.

Due to their inherent feedback, closed-loop normalizers present the major disadvantage of needing frequency compensation techniques to stabilize the output signal. For instance, in Figure 3.24 b) a relatively large capacitor must be placed at the feedback control node. Therefore, a slower transient behavior of the defuzzifier circuit, and consequently, of the controller itself may be expected. Moreover, in the case shown in Figure 3.24 a), the silicon surface and current consumption of the normalizer circuit increase with the number of rules m of the controller. This is evident since one transconductor per rule is required. In the case of the scheme in Figure 3.24 b), either the membership function circuits or the T-Norm operators must be provided with an additional input controlling the gain of these blocks. This makes such implementation troublesome.

Open-loop pseudo-normalizers can overcome the stability problem since they do not require compensation. However, depending on how the controlled resistors Ri are implemented their complexity may also become very large when a Fuzzy Controller with a large number of rules is contemplated. Furthermore, the non-linear relationship between the output of the defuzzifier and its inputs Ii [VaVi99] turns the on-chip implementation of a learning (tuning) circuit into a difficult challenge.

Some authors suggest avoiding dividers for the purpose of normalizers. One of the main reasons lies on the fact that in pseudo-normalizers or in closed-loop normalizers, the computation of the defuzzified output value is carried out by means of a massive number of identical circuits. Therefore, individual deviations due to technological spreading of the elemental circuit should statistically cancel out at the output node provided the defuzzifier circuit is large. However, the latter assertion is correct only if the whole normalizer circuit remains active all the time. This situation never happens in a Fuzzy Controller since mostly three or four rules only are fired simultaneously while the cells related to the inactive rules are set off.

Another argument found in the literature that prevents the use of dividers, is supported by the fact that in most reported MOS dividers circuits, transistors operate in weak inversion. Consequently, those dividers exhibit poor transient performances. On the other hand, some implementations of dividers in strong inversion feature a complex signal interface. For instance, in [HuBa97] a current-input voltage-output divider is presented. An additional fully differential current-to-voltage converter is needed, however, at one of the divider inputs. In [BuWa87], another current-input current-output divider is proposed. In this device, one input current must be supplied twice to two different nodes of the circuit, while the other input is a floating current source.

In the light of the above discussions, we preferred to implement our defuzzifier according to the open-loop policy and using a simple divider circuit. The latter has been designed to operate in strong inversion while allowing easy signal interfaces [DuVe98]. In this way, the complexity of the normalization stage remains almost independent of the number of rules of the controller. Only one extra mirroring transistor per rule is needed to compute the denominator of the normalized weighted sum. On the other hand, since normalization is taken care by only one circuit, special attention can be paid during the design for optimizing its performance. In addition, as it will be explained below, when the singletons αi are supposed to be digitally programmable, using a divider together with a "common-weighting" strategy leads to a further simplification of defuzzifying stage.

3.3.1 Open-Loop Takagi-Sugeno's Defuzzifier with Divider and Digitally-Programmable Consequent Singletons

In digitally programmable Zero-Order TS's controllers, a typical approach for implementing the consequent singletons of the rules consists of using current-mode D/A converters like the one illustrated in Figure 3.26. In order to replicate exactly the scheme in Figure 3.25 b) one D/A per rule is required. We will call this strategy the "local-weighting" approach [HuBa97] [VaVi99] [WiJa96]. In this case, the value of the singleton is programmed according to the state of the switches $C_0 \ldots C_{n-1}$ in Figure 3.26. However, the number of transistors needed per singleton reaches 2^n. Therefore, the total number of transistors demanded by the whole set of singletons of the controller increases considerably with the number of rules m and the resolution n of the D/As. In addition, the load capacitance that each singleton represents for each signal Ii raises to $2^n Cgs$, Cgs being the gate-to-source capacitance of each elemental transistors in the D/A. Thus, the speed of the weighting operation could be substantially diminished, as the resolution n of the D/A gets high.

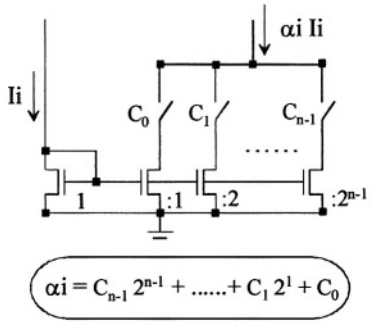


Figure 3.26. Current-mode D/A converter used as digitally-programmable singleton.

We discuss hereafter a strategy that minimizes the number of transistors per singleton as well as the input capacitance of each singleton. We call this strategy the "common-weighting" approach [Land96]. Figure 3.27 shows the architecture of the defuzzifier stage we have implemented. The singletons αi are now represented by n unit-gain current mirrors. One extra mirror is required at each row for computing the sum of the firing degrees of the rules (i.e. ΣIi). However, the weighting operation is now performed by one D/A only, which is a modified version of the circuit presented in Figure 3.26. Each scaled mirror of the common D/A is independently accessible, as shown in Figure 3.27. In addition, in order to minimize the current consumption, the values of the singletons were defined smaller than one.

This also conveys the current inputs of the divider (i.e. $IN=\Sigma\alpha i Ii$, $ID=\Sigma Ii$) to range evenly. It will be shown in Chapter 4 that the divider can be sized and biased properly upon such input range conditions.

In the following we will show that this "common-weighting" strategy leads exactly to the same defuzzified output than the one yielded by controllers that make use of the "local-weighting" counterpart.

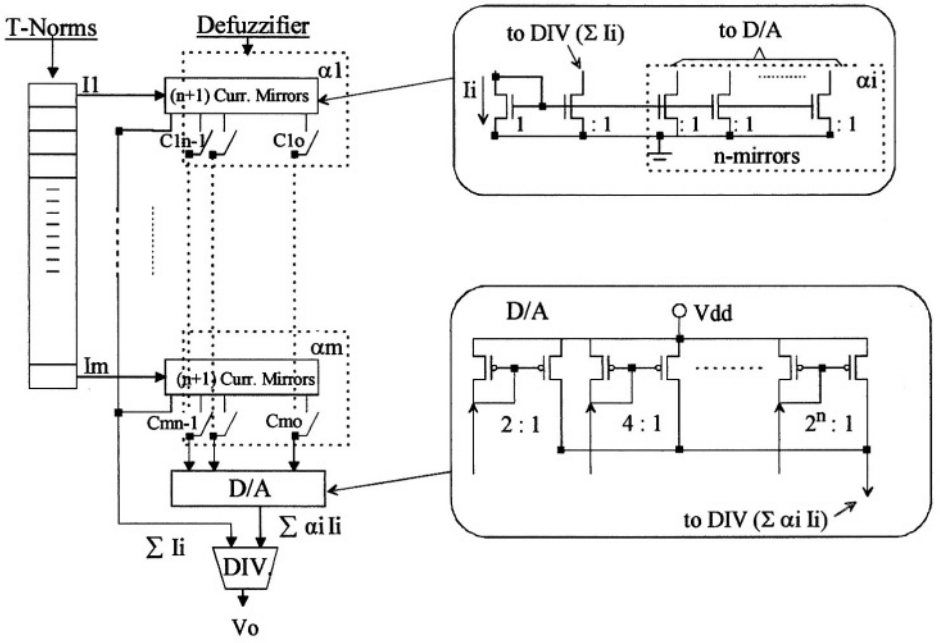


Figure 3.27. Open-loop defuzzifier with a divider and the "common-weighting" strategy.

For the consequent of each rule a discrete-singleton αi smaller than 1 is given by:

$$\alpha i = Ci_{n-1}\,2^{-1} + Ci_{n-2}\,2^{-2} + \ldots + Ci_0\,2^{-n}, \tag{3.58}$$

where n stands for the resolution of the D/A, m is the number of rules and i ranges from 1 to m. The coefficients $(Ci_{n-1}\ldots Ci_0)$ adopt binary values according to the state of their corresponding switches in Figure 3.27. In the same figure, the outputs of the (n+1) current mirrors of the whole m-singleton set are column-wise summed to give the following (n+1) values:

$$\left(\sum_{i=1}^{i=m} Ii\right);\ \left(\sum_{i=1}^{i=m} Ci_{n-1}\,Ii\right);\ldots;\ \left(\sum_{i=1}^{i=m} Ci_0\,Ii\right). \tag{3.59}$$

The first term above is the denominator of the divider that follows the D/A. All other terms in (3.59) are independently weighted and summed in the common D/A before becoming the numerator input of the divider. Hence, the output current of the D/A is given by:

$$\begin{aligned} Iout_{D/A} &= \left(\sum_{i=1}^{i=m} Ci_{n-1}\, 2^{-1}\, Ii\right) + \ldots + \left(\sum_{i=1}^{i=m} Ci_0\, 2^{-n}\, Ii\right) \\ &= \sum_{i=1}^{i=m} \left(\left(Ci_{n-1}\, 2^{-1} + \ldots + Ci_0\, 2^{-n}\right) Ii\right) = \sum_{i=1}^{i=m} \alpha i\ Ii. \end{aligned} \tag{3.60}$$

In conclusion, with the common D/A the surface occupied by one singleton as well as its input capacitance are reduced by a factor $(2^n/n)$ when compared to the "local-weighting" approach. Moreover, in order to improve the matching properties of the common D/A converter, this can be built using non-minimum size transistors. Furthermore, it will be shown in next sections that each bit of the D/A converter has its own mismatch figure, which depends on the bit weight. Since our D/A is a set of n independent binary-scaled current mirrors, each mirror can be sized independently according to a desired accuracy specification for the total weighting process. Finally, since the layout of the whole defuzzifier gets smaller, routing capacitances are considerably diminished.

Previous idea on global weighting can be found in [Land96]. In the latter, weighting is achieved in a pseudo-normalizer by exploiting a current division technique by means of encoding splitters (one per singleton) and a R/2R network. However, for discrete-programming singletons, splitters need two switch transistors per bit and the availability of both complementary digital programming signals (i.e. Ci and /Ci) applied at the gate of the switch transistors. Thus, splitters demand 33% more silicon area and 50% more routing space for the digital programming signals. Moreover, as a consequence of the current splitting, the defuzzified output value results n times more attenuated than in our case. Consequently, for a given dynamic range, the signal-to-noise figure needed at the R/2R network results more compromised. In addition, the defuzzified output is a current that requires an extra current-to-voltage converter if the controller is inserted within a voltage-mode controlled environment.

3.3.1.1 Novel Current-Input Voltage-Output Analog Divider.

A novel transresistive divider [DuVe98] was specially designed to carry out the normalization operation in the defuzzifier. The circuit is shown in Figure 3.28. Transistors in the right column are k times scaled with respect to their homologues at each row, as shown at the bottom of the figure. The

division is performed by means of transistors M1, M2, M3 at the bottom layer, all of them being constrained to operate in the triode region. The drain-to-source voltage drops Vds of those transistors are matched thanks to the common-gate transistors M4, M5, M6, whose gate-voltage overdrives (GVO) are identical. This is guaranteed by the upper PMOS mirrors (M7 to M9), which replicate the current I2 to the left branch and the scaled current k×I2 to the right branch.

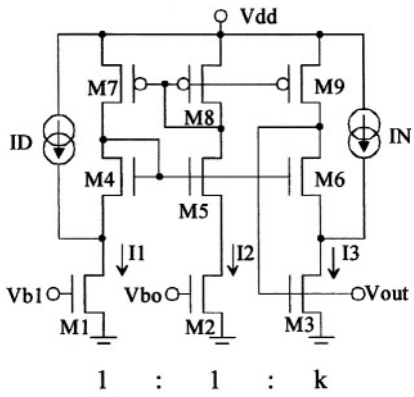


Figure 3.28. Novel current-input voltage-output analog divider. From [DuVe98], © 1998 IEE, reprinted with permission.

While Vb1 and Vbo are fixed bias voltages, the gate voltage Vout of transistor M3 is self-adjusted so that the drain current of M6 matches the current imposed by M9. In this way, the following relations hold for the drain currents of the triode transistors:

$$I1 = \beta\, Vds\left(Vb1 - VTn - \frac{n}{2}Vds\right) = ID + I2, \tag{3.61}$$

$$I2 = \beta\, Vds\left(Vbo - VTn - \frac{n}{2}Vds\right), \tag{3.62}$$

$$I3 = k\,\beta\, Vds\left(Vout - VTn - \frac{n}{2}Vds\right) = IN + k\,I2, \tag{3.63}$$

where Vds is the common drain-to-source voltage drop for the three bottom transistors while β is the current gain ratio of M1 and M2. Therefore, upon ideal matching conditions, from the latter three equations we can write:

$$(\text{Vout} - \text{Vbo}) = \frac{(\text{Vb1} - \text{Vbo})}{\text{k}} \frac{\text{IN}}{\text{ID}}. \tag{3.64}$$

Thus, if Vout is referred to Vbo, we obtain a two-quadrant divider. Given the desired current ranges of IN and ID, the maximum output voltage swing is defined by the difference (Vb1–Vbo) and the scaling factor k, which must be chosen accordingly.

Whenever a voltage-mode external output inteface is required, the transresistive property of this divider makes it suitable for Fuzzy Controllers performing internally current-mode analog computations. Hence, there is no need for extra interface converter circuits neither at its inputs [HuBa97] [LiCh95] [VlSi99] nor at the output [LiHu94] [Wang91] [MaCo97].

3.3.1.2 Systematic Errors in the Analog Divider.

Like in any translinear network, the mobility reduction is the main source of systematic errors in this circuit. Since the division is performed by means of transistors operating in the triode mode, which have different gate voltages, the mismatch between their mobility is likely to impair the performances. Calling θ the mobility reduction coefficient, equations (3.61), (3.62) and (3.63) must be rewritten taking into account this effect:

$$\text{I1} = \frac{\beta}{1+\theta(\text{Vb1} - \text{VTn})} \text{Vds} \left(\text{Vb1} - \text{VTn} - \frac{\text{n}}{2} \text{Vds} \right), \tag{3.65}$$

$$\text{I2} = \frac{\beta}{1+\theta(\text{Vbo} - \text{VTn})} \text{Vds} \left(\text{Vbo} - \text{VTn} - \frac{\text{n}}{2} \text{Vds} \right), \tag{3.66}$$

$$\text{I3} = \frac{\text{k}\beta}{1+\theta(\text{Vout} - \text{VTn})} \text{Vds} \left(\text{Vout} - \text{VTn} - \frac{\text{n}}{2} \text{Vds} \right). \tag{3.67}$$

By introducing in the avobe equations the first order Taylor aproximation of the denominators:

$$\frac{1}{1+\theta\,(V-VTn)} \approx 1-\theta\,(V-VTn), \quad \text{with } V \in \{Vbo, Vb1, Vout\}, \tag{3.68}$$

after some manipulations we obtain the following approximated equation for the divider output:

$$(Vout-Vbo) \approx (Vb1-Vbo)\frac{IN}{k\,ID} + \theta\,(Vb1-Vbo)^2\left(\frac{IN}{k\,ID}\right). \left[\left(\frac{IN}{k\,ID}\right)-1\right]. \tag{3.69}$$

Considering that the first term at the second member of (3.69) is the error-free output expression, we conclude that the relative error due to mobility reduction is given by:

$$(Vout-Vbo) = (1+\varepsilon_\theta)(Vb1-Vbo)\frac{IN}{kID}, \text{ with } \varepsilon_\theta \approx \theta\,(Vb1-Vbo)\left(\frac{IN}{k\,ID}-1\right). \tag{3.70}$$

As may be expected, this error is proportional to the coefficient θ and also to (Vb1-Vbo). Since the output swing is also proportional to the former voltage difference, a trade-off between accuracy and output range appears.

Another kind of systematic error concerns the quality of the mirroring operation performed by the upper PMOS transistors layer in Figure 3.28. When the Early effect is being considered in transistors M7 and M9, the expression for the currents IN and ID should be modified as follows:

$$ID = \beta\,Vds\,(Vb1-Vbo) - \lambda p\,I2\,(Vdd-Vd7), \tag{3.71}$$

$$IN = k\,\beta\,Vds\,(Vout-Vbo) - k\,\lambda p\,I2\,(Vdd-Vout), \tag{3.72}$$

where Vd7 is the drain voltage of M7 whereas I2 is the current in the central branch given by (3.62). Considering this non ideality, we arrive to an expression for (Vout–Vbo) that shows a gain error ε_G and an offset $Voff_{div}$:

$$(Vout-Vbo) = \frac{(Vb1-Vbo)}{k}\,\frac{IN}{ID}\,(1+\varepsilon_G) + Voff_{div}, \tag{3.73}$$

where:

$$\varepsilon_G \approx \frac{\lambda p\left(Vbo - VTn - \frac{n}{2}Vds\right)}{1 - \lambda p\left(Vbo - VTn - \frac{n}{2}Vds\right)}\left(1 + \frac{Vdd - Vd7}{Vb1 - Vbo}\right), \qquad (3.74)$$

$$Voff_{div} \approx \frac{\lambda p\left(Vbo - VTn - \frac{n}{2}Vds\right)}{1 - \lambda p\left(Vbo - VTn - \frac{n}{2}Vds\right)}\,(Vdd - Vbo). \qquad (3.75)$$

Since the gain error and offset are proportional to the channel length modulation coefficient λp the impact of Early effect can be further diminished if the upper PMOS mirrors are cascoded. Table 3.8 shows some typical values of the systematic errors of the divider.

Table 3.8. Systematic errors of the divider for Vdd=5V, Vbo=1.7V, Vb1=2.7V and k=1. Voff and ε_G values result after cascoding the PMOS upper mirrors.

Input Range	Output Range [V]	ε_θ [%]	Max. Vout Deviation [Wieg93]	ε_G	Voff
$0 \le \frac{IN}{ID} \le 1$	$1.7 \le Vout \le 2.7$	$0 \le \varepsilon_\theta \le 3.5$	8.8mV (0.88%)	0.001	1.12mV

3.3.1.3 Mismatch Errors in the Analog Divider.

It is very difficult to obtain a complete expression for the divider output voltage spreading as a function of the variances of the threshold voltages VT and current gains β of each transistor in Figure 3.28. Therefore, let us introduce some simplifications. Considering first that the performance of the division operation is mainly related to the triode operating transistors M1, M2 and M3, we must consider the spreading of the technological parameters of these transistors (i.e. σ_{VT} and σ_β). However, if we consider these technological parameters as independent random variables, their effects are interrelated not only by equations (3.61) to (3.64) but also by the set of relationships linking the Vds of the bottom triode transistors. These can be expressed as:

$$Vds1 + VT4 + \sqrt{\frac{2\,n\,(I1 - ID)}{\beta 4}} = Vds2 + VT5 + \sqrt{\frac{2\,n\,I2}{\beta 5}}, \tag{3.76}$$

$$Vds3 + VT6 + \sqrt{\frac{2\,n\,(I3 - IN)}{\beta 6}} = Vds2 + VT5 + \sqrt{\frac{2\,n\,I2}{\beta 5}}, \tag{3.77}$$

$$Vds3 + VT6 + \sqrt{\frac{2\,n\,(I3 - IN)}{\beta 6}} = Vds1 + VT4 + \sqrt{\frac{2\,n\,(I1 - ID)}{\beta 4}}. \tag{3.78}$$

On the other hand, the relative variance of the current through a triode transistor is given by:

$$\frac{\sigma_I^2}{I^2} = \frac{\sigma_\beta^2}{\beta^2} + \left(\frac{gm}{I}\right)^2 \sigma_{VT}^2 + \left(\frac{gd}{I}\right)^2 \sigma_{Vds}^2 . \tag{3.79}$$

Notice that an extra term due to the variance of the Vds drop appears owing to the fact that the drain output conductance gd of the triode transistor is not negligible. Consequently, the mismatches between Vds1, Vds2 and Vds3 should be also introduced for the computation of the total output variance of the divider. However, for low current level at the columns of the divider, the main sources provoking the mentioned Vds disparities are the threshold voltage fluctuations of transistors M4, M5 and M6 in the middle layer (row) in Figure 3.28. Therefore, the root-squared terms in equations (3.76) to (3.78), which represent the GVOs of transistors M4 to M6, can be disregarded during the estimation of variances. As a result, the variance of the output voltage of the divider can be expressed as:

$$\sigma_{(Vout-Vbo)}^2 = \alpha_{VT1}\sigma_{VT1}^2 + \alpha_{VT2}\sigma_{VT2}^2 + \alpha_{VT3}\sigma_{VT3}^2 + \alpha_{VT4}\sigma_{VT4}^2 + \\ + \alpha_{VT5}\sigma_{VT5}^2 + \alpha_{VT6}\sigma_{VT6}^2 + \alpha_{\beta 1}\sigma_{\beta 1}^2 + \alpha_{\beta 2}\sigma_{\beta 2}^2 + \alpha_{\beta 3}\sigma_{\beta 3}^2 . \tag{3.80}$$

Assuming the nominal conditions VT1=VT2=VT3=VT4=VT5=VT6, Vds1=Vds2=Vds3=Vds and $\beta 1=\beta 2=(\beta 3/k)=\beta$, the analytical values for the coefficients α above are given in Table 3.9.

Table 3.9. Analytical expressions for the coefficients α in (3.80).

Coeff. α	Value
α_{VT1}	$\left(\frac{IN}{k\,ID}\right)^2$
α_{VT2}	$\left(\frac{IN}{k\,ID}-k\right)^2$
α_{VT3}	1
α_{VT4}	$\left(\frac{IN}{k\,ID}\right)^2\left(\frac{\beta\,(Vb1-VT1)\,(Vb1-Vbo)}{ID}-n\right)^2$
α_{VT5}	$\left[\left(\frac{IN}{k\,ID}\right)\left(\frac{\beta\,(Vb1-VT1)\,(Vb1-Vbo)}{ID}-n\right)+n-\left(\frac{\beta\,(Vb1-Vbo)}{ID}\right)\left(\frac{IN}{k\,ID}(Vb1-Vbo)-Vbo+VT1\right)\right]^2$
α_{VT6}	$\left[n-\left(\frac{\beta\,(Vb1-Vbo)}{ID}\right)\left(\frac{IN}{k\,ID}(Vb1-Vbo)-Vbo+VT1\right)\right]^2$
$\alpha_{\beta1}$	$\left(\frac{IN}{k\,ID}\right)^2\left(\frac{1}{\beta}\right)^2\left(Vb1-VT1-\frac{n}{2}\frac{ID}{\beta\,(Vb1-Vbo)}\right)^2$
$\alpha_{\beta2}$	$\left(1-\frac{IN}{k\,ID}\right)^2\left(\frac{1}{\beta}\right)^2\left(Vb1-VT1-\frac{n}{2}\frac{ID}{\beta\,(Vb1-Vbo)}\right)^2$
$\alpha_{\beta3}$	$\left(\frac{1}{k\beta}\right)^2\left(\frac{IN\,(Vb1-Vbo)}{k\,ID}+Vbo-VT2-\frac{n}{2}\frac{ID}{\beta\,(Vb1-Vbo)}\right)^2$

Figure 3.29 a) illustrates the standard deviation of the output voltage of the divider for different input current conditions satisfying IN≤ID. In all cases k=1, Vbo=1.7V and Vb1=2.7V. It is evident that largest mismatch errors take place when currents IN and ID (specially ID) get small.

Figure 3.29 b) shows the total output variance composition from the individual contributions of the different parameters. Notice from the latter figure the strong incidence of the mismatch of the threshold voltages of transistors M4, M5 and M6 when the current ID adopts low values.

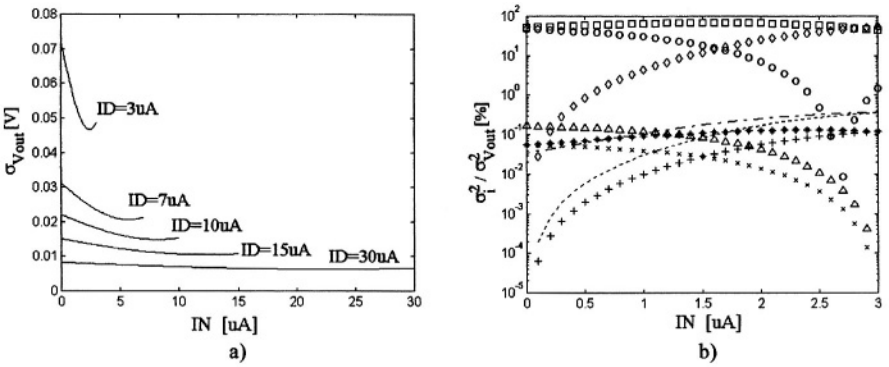


Figure 3.29. a) Standard deviation of the voltage Vout of the divider due to mismatch. b) Relative incidence on the total output variance of each parameter mismatch for ID=3μA: (+) σ_{VT1}; (x) σ_{VT2}; (*) σ_{VT3}; (◊) σ_{VT4}; (o) σ_{VT5}; (□) σ_{VT6}; (-) $\sigma_{\beta 1}$; (△) $\sigma_{\beta 2}$ and (-.) $\sigma_{\beta 3}$.

For given β, Vb1 and Vbo, it can be demonstrated that the drain-to-source voltage drops Vds of the triode-transistors M1, M2 and M3 are controlled by the denominator current. As long as the denominator current gets small, Vds follows the same trend. Therefore, for small ID the mismatch of the threshold voltages of transistors M4, M5 and M6 may impact strongly on the mismatch of the Vds drops of the triode transistors.

3.3.1.4 Systematic Errors in the Consequent and the Common D/A.

According to Figure 3.27 systematic errors for the arrangement consequent+D/A are due to the Early effect in the NMOS unit mirrors of the singleton and the PMOS scaled mirrors of the D/A. Let us assume only one fired rule in the controller whose corresponding singleton value is set to the maximum allowed ($\alpha i=1$). In order to compute the total systematic error at the output of the D/A the individual contribution of each bit is first addressed.

Figure 3.30 represents the connection between a consequent singleton and the D/A, for the generic bit (N-j), N* being the resolution of the D/A whereas j ranges from 1 to N. In the same figure Iout is the output of the D/A that corresponds to the signal ($\Sigma \alpha i \times Ii$) in Figure 3.27. In Figure 3.30, it

* To avoid misunderstanding with the coefficient n accounting for the Body effect, here we prefer to denote N the number of bits.

is clear that as long as V2 differs from V1, I2 offsets Iin. In a similar way, $(2^j)\times I1$ diverges from I2 due to the difference between voltages V2 and Vout. Thus, the input-output relationship for this bit is given by:

$$I1 = \frac{1}{2^j}\left(\frac{1+\lambda p\,(Vdd - Vout)}{1+\lambda p\,(Vdd - V2)}\right)\left(\frac{1+\lambda n\,V2}{1+\lambda n\,V1}\right) Iin, \tag{3.81}$$

where λn and λp are the channel length modulation coefficients of NMOS and PMOS transistors, repectivelly, whereas V1 and V2 are given by:

$$V1 = VTn + \sqrt{\frac{2\,n\,Iin}{\beta n}}, \qquad V2 \approx Vdd - |VTp| - \sqrt{\frac{2\,n\,Iin}{2^j\,\beta p}}. \tag{3.82}$$

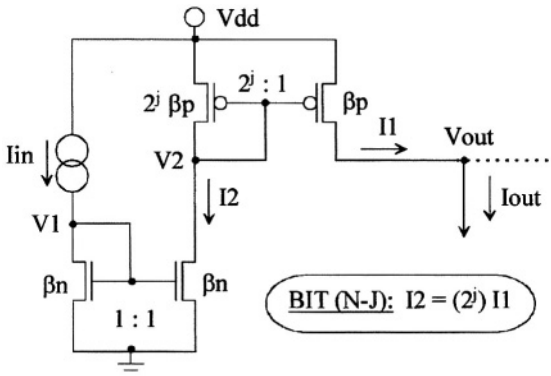


Figure 3.30. Connection between the bits (N-j) of the consequent singleton and the D/A.

$$Iout = Iin\sum_{j=1}^{j=N}\left[\frac{(1+\lambda p\,(Vdd - Vout))}{2^j\left(1+\lambda p\left(|VTp| + \sqrt{2^{(1-j)}\,n\,Iin\,\beta p^{-1}}\right)\right)}\cdot \frac{\left(1+\lambda n\left(Vdd - |VTp| - \sqrt{2^{(1-j)}\,n\,Iin\,\beta p^{-1}}\right)\right)}{\left(1+\lambda n\left(VTn + \sqrt{2\,n\,Iin\,\beta n^{-1}}\right)\right)}\right]. \tag{3.83}$$

Since the output current Iout of the D/A is the sum of the individual bit currents, it turns out that Iout is given by (3.83) In the above equations βn and βp are the current gains of the NMOS and PMOS unit transistors respectivelly. Notice from the latter cumulative error expression the importance of using cascoded mirrors at the consequent singletons and the D/A.

3.3.1.5 Mismatch Errors in the Consequent and the Common D/A.

With regard to Figure 3.30, we assume the threshold voltages VT and current gains β of all transistors at bit (N-j) to be statistically independent random variables. Thus, the relative variance for the current I1 at the output node of each bit is given by:

$$\frac{\sigma_{I1}^2}{I1^2} = \left(\frac{\sigma_{\beta p}^2}{\beta p^2} + \left(\frac{gm_p}{I1} \right)^2 \sigma_{VTp}^2 \right) \left(1 + \frac{1}{2^j} \right) + \frac{\sigma_{I2}^2}{I2^2}. \qquad (3.84)$$

The first term in the second member of the above equation represents the mismatch introduced by the down-scaled PMOS mirror in Figure 3.30. Notice that it is affected by the factor $(1+2^{-j})$ that increases with the weight of the bit (j=1 for the most significant bit). Hence, the least significant bit contributes with the smallest fraction to the total variance.

The second term in the latter expression corresponds to the relative variance of current I2 owing to the mismatch introduced by the bottom NMOS mirror. Thus:

$$\frac{\sigma_{I2}^2}{I2^2} = 2 \left(\frac{\sigma_{\beta n}^2}{\beta n^2} + \left(\frac{gm_n}{I2} \right)^2 \sigma_{VTn}^2 \right). \qquad (3.85)$$

Assume again only one active rule with its corresponding singleton set to the maximum value. Therefore, under nominal conditions we have:

$$I2 = 2^j \, I1 = Iin \qquad \text{and} \qquad Iout = \sum_{j=1}^{j=N} 2^{-j} \, Iin \approx Iin. \qquad (3.86)$$

Consequently, the final expression for the relative variance of the output Iout of the D/A is given by:

$$\frac{\sigma^2_{Iout}}{Iout^2} = \sum_{j=1}^{j=N}\left(\frac{1}{2^{2j}}\right)\left[\left(1+\frac{1}{2^j}\right)\left(\frac{\sigma^2_{\beta p}}{\beta p^2}+\left(\frac{gm_p}{I1}\right)^2\sigma^2_{VTp}\right)+ 2\left(\frac{\sigma^2_{\beta n}}{\beta n^2}+\left(\frac{gm_n}{I2}\right)^2\sigma^2_{VTn}\right)\right]. \quad (3.87)$$

In this expression, gm_p and gm_n stand for the transconductance of the unit PMOS and NMOS transistors in Figure 3.30. However, in the latter procedure we have considered, for clarity, that currents I2 at each bit are independent, which is not true. Since in the real implementation of the circuit all those currents are mirrors of the same current Iin, they are correlated (see in Figure 3.27 the circuit of a singleton). Therefore, the cross-correlation coefficient between current I2 at bit (N-j) and current I2 at bit (N-i) should be calculated for all pairs of currents $(I2_{N-i}, I2_{N-j})$ with $i \neq j$.

Since the bottom NMOS mirror in Figure 3.30 is identical for all bits, these correlation coefficients are identical for any pair of currents under consideration. Thus, the amount to add to equation (3.87) due to the cross-correlation between currents I2 at each bit is given by:

$$\frac{\sigma^2_{corr}}{Iout^2} = \sum_{i,j=1;\, i\neq j}^{N} \frac{1}{2^j}\frac{1}{2^i}\frac{\sigma^2_{I2_{N-j},\, I2_{N-i}}}{Iout^2} = 2\left(\frac{\sigma^2_{\beta n}}{\beta n^2}+\left(\frac{gm_n}{I2}\right)^2\sigma^2_{VTn}\right)\left(\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\frac{1}{2^j}\frac{1}{2^i}\right). \quad (3.88)$$

Figure 3.31 a) shows simulation results for the standard deviation of Iout at the D/A. Figure 3.31 b) shows the contribution of each individual bit.

As predicted, the most significant bit yields the largest portion of the total variance. However, we can take advantage of the fact that all inputs of the weighting D/A are independent (see Figure 3.27). Thus, their corresponding transistors size can be chosen independently one of each other while keeping their own binary scaling ratio. In this way, the high order bits can be built by transistors with increased active area (W×L) with respect to transistors building the low order bits.

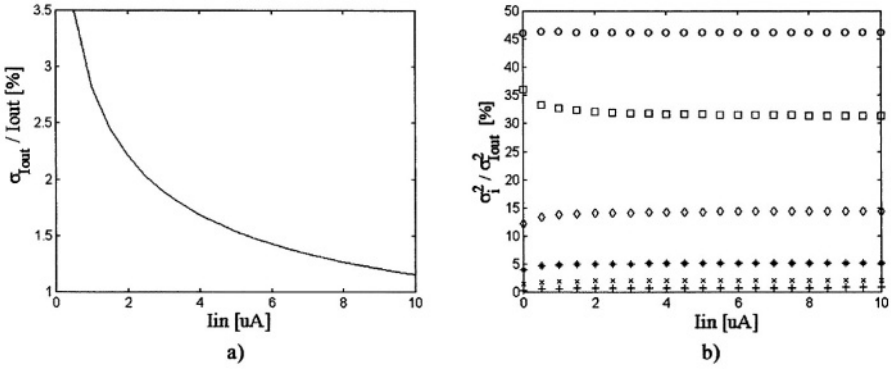


Figure 3.31. a) Relative standard deviation of the output current Iout of the D/A. b) Incidence of the variance of each bit and the correlation term given by (3.88) on the total output variance: (o) $\sigma_{I1\text{-}4}$; (◊) $\sigma_{I1\text{-}3}$; (*) $\sigma_{I1\text{-}2}$; (x) $\sigma_{I1\text{-}1}$; (+) $\sigma_{I1\text{-}0}$ and (□) σ_{corr}.

3.3.2 Open-Loop Defuzzifier with Analog Programmable Singletons

For the sake of completeness, we have also investigated the possibility of integrating analog programmable circuits in a technology providing analog non-volatile storage capabilities (i.e. floating-gate programmable devices). In such a case the structure of the Defuzzifier shown in Figure 3.25 b) remains intact but the consequent singletons should be electrically programmable.

Figure 3.32 a) shows a possible realization of an electrically tunable current mirror that can be used for this purpose. The same circuit was also used as a fully electrically-controllable transconductor to build a gm-C filter. Its working principle and non idealities will be explained in details in Chapter 5.

Briefly, the circuit comes from the divider previously presented but a little modification has been introduced. Assuming the NMOS transistors at the bottom layer working in the triode region, the former numerator current (IN) is now controlled by a negative feedback loop. This is performed by transistor M10 whose current is in turn B times up-scaled by mirror M11 that delivers the output current Iout. Therefore, the set of equations defined in section 3.3.1.1 for the divider still holds. From these equations and assuming ideal matching conditions, we can write:

$$Iout = B\,\frac{(Vb2 - Vbo)}{(Vb1 - Vbo)}\,Iin\,. \qquad (3.89)$$

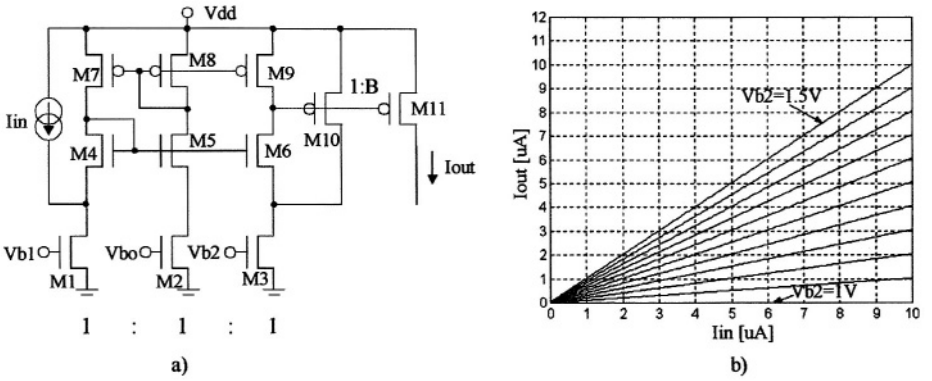


Figure 3.32. a) Electrically-programmable current mirror. b) SPICE simulation result for Vbo=1V, Vb1=1.5V, B=1 and Vb2 ranging from 1 to 1.5V by 50mV steps.

Notice from the latter expression that not only the relationship between input and output current is linear but also the current mirror gain can be linearly tunable. For this purpose, the bias voltage Vb2, eventually derived from an analog memory, can be used as the gain-controlling variable. Figure 3.32 b) shows a SPICE simulation wherein the linearity of the mirror can be appreciated all along a 10μA input-output range.

3.4 An Estimation of the Accuracy of a Fuzzy Controller

After a close look to the non idealities present in each building block, an evaluation of their incidence on the global accuracy of the controller they build is now addressed. Based on the approaches presented by [Vida96] [Espe94], a method for estimating the propagation of the errors of the different fuzzy operators along the signal path is discussed hereafter.

On one hand, systematic errors can be minimized by following proper design strategies. On the other hand, since they are to some extend deterministic, they can be incorporated into the ideal input-output relationship of each block. This gives rise to a more accurate model that can be directly used to fit a desired target function at the system level design step. For those reasons systematic errors will not be considered for this analysis. In contrast, mismatch errors can only be statistically estimated. An investigation about the spreading of the controller behavior becomes mandatory, specially when dedicated Fuzzy Controllers without tuning capabilities are intended.

Figure 3.33 shows a block diagram of the controller involving the basic operators taking part in a rule. In a first simulation, we will consider only one fired rule at the controller. Some assumptions are to be made however:

- The singleton of the consequent of the rule under consideration is set to its maximum value (i.e. $\alpha i=1$). It has been proven in the former section that this situation corresponds to the maximum standard deviation at the output of the consequent+D/A grouped blocks in Figure 3.33.
- Since the T-Norm plays the role of a switch selecting the minimum value among those provided by the set of membership functions in a rule, we will consider this operator as the identity function. Therefore, the T-Norm outputs the minimum value delivered by only one membership function. This assumption remains valid even if a T-CoNorm (maximum) followed by complementation is used.

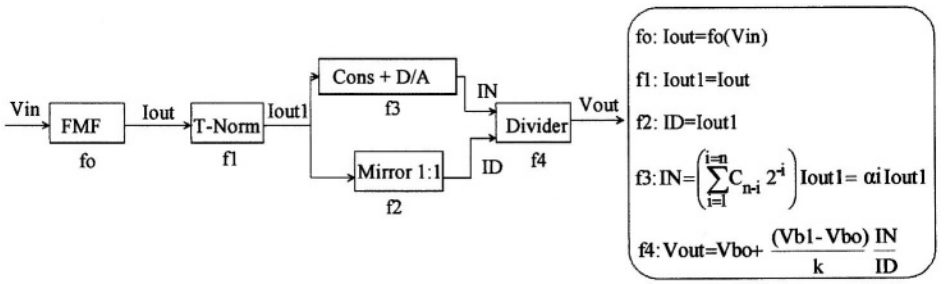


Figure 3.33. Simplified block diagram of the controller for the estimation of the Standard Deviation of its output Vout due to mismatch. Only one fired rule is considered.

Bearing in mind the above considerations we can start at the end of the signal path by writing the variance of the output signal Vout as:

$$\sigma^2_{Vout} = \sigma^2_{div} + \left(\frac{\partial f4}{\partial ID}\right)^2 \sigma^2_{ID} + \left(\frac{\partial f4}{\partial IN}\right)^2 \sigma^2_{IN} + 2\left(\frac{\partial f4}{\partial ID}\right)\left(\frac{\partial f4}{\partial IN}\right)\sigma^2_{ID,IN} \cdot \tag{3.90}$$

The first term σ_{div} in the above equation corresponds to the mismatch error introduced by the divider itself calculated in section 3.3.1.3. The second and third terms correspond to the errors carried by signals ID and IN and propagated through the divider. The last term accounts for the cross-correlation between the latter two signals, considering they are generated from a common node at the output of the T-Norm (Iout1). Therefore, we can write:

$$\sigma^2_{IN} = \sigma^2_{cons+D/A} + \left(\frac{\partial f3}{\partial Iout1}\right)^2 \sigma^2_{Iout1}, \tag{3.91}$$

$$\sigma^2_{ID} = \sigma^2_{mirror} + \left(\frac{\partial f2}{\partial Iout1}\right)^2 \sigma^2_{Iout1}, \tag{3.92}$$

$$\sigma^2_{IN,ID} = \left(\frac{\partial f2}{\partial Iout1}\right)\left(\frac{\partial f3}{\partial Iout1}\right) \sigma^2_{Iout1}, \tag{3.93}$$

where the expressions for the standard deviations $\sigma_{cons+D/A}$ can be found in section 3.3.1.5, and σ_{mirror} is the standard deviation of the current of the mirror before the denominator ID in Figure 3.33. Finally, the variance of the error for the signal Iout1 can be calculated as the sum of the variance of the error introduced by the T-Norm plus the one related to the membership function FMF, which is propagated through the triangular operator. Hence:

$$\sigma^2_{Iout1} = \sigma^2_{T\text{-}Norm} + T\,\sigma^2_{Io} + \left(\frac{\partial f1}{\partial Iout}\right)^2 \sigma^2_{FMF}, \tag{3.94}$$

where the expressions for $\sigma_{T\text{-}Norm}$ and σ_{FMF} can be found in sections 3.2.1.2 and 3.1.3.3 respectively. In the latter equation, the variable T adopts the value of zero when the T-Norm is a direct triangular norm (i.e. MINIMUM) whereas it becomes one if a T-CoNorm (i.e. MAXIMUM) is used. In this case, not explicited in Figure 3.33, we must add the variance of the current Io complementing the output of the T-CoNorm.

From (3.90) to (3.94) we can derive a clear-cut relationship that illustrates clearly the contribution of the individual mismatch error from each operator on the variance of the output of the controller. In this way, we can write:

$$\begin{aligned}\sigma^2_{Vout} = {} & \sigma^2_{div} + a_{cons+D/A}\,\sigma^2_{cons+D/A} + a_{mirror}\,\sigma^2_{mirror} + \\ & + a_{T\text{-}Norm}\,\sigma^2_{T\text{-}Norm} + a_{Io}\,\sigma^2_{Io} + a_{FMF}\,\sigma^2_{FMF},\end{aligned} \tag{3.95}$$

where the analytical expression for the coefficients a are given in Table 3.10.

Simulation results that illustrate the standard deviation of the output Vout are shown in Figure 3.34 a) and b) as a function of the firing degree of the only rule considered. The adopted membership function circuit corresponds to the CFMF Type-I whose slopes were set to the minimum value. The T-Norm comprises a Lazzaro's WTA circuit followed by complementation. For the divider, the following settings have been assumed: Vb1=2.7V, Vbo=1.7V and k=1. Some remarks can be made as follows:

- As shown in Figure 3.34 a) the standard deviation of the output of the controller decreases from 80mV to 20mV as the rule is being increasingly activated. However, in any Fuzzy Controller, when only one rule is fired, its firing degree turns to be high (near 1). Therefore, to be more realistic, one should consider that the standard deviation of Vout ranges from 20mV to 30mV (i.e. 2% to 3% of the output voltage swing).
- From Figure 3.34 b) we see that the most important contribution to the total variance of the controller output is due to the divider, followed by the consequent+D/A and the mirror before the denominator ID. On the contrary, the mismatch errors of the membership function and the T-Norm circuits are not propagated towards the output when only one rule is fired. This is due to the normalization performed by the divider that cancels out any deviation simultaneously present at both inputs.

Table 3.10. Analytical expressions of the coefficients "a" in the expression of σ_{Vout}.

Coeff. a	Value
$a_{cons+D/A}$	$\left(\frac{Vb1 - Vbo}{k\,ID}\right)^2$
a_{mirror}	$\left(\frac{(Vb1 - Vbo)}{k}\frac{IN}{ID^2}\right)^2$
$a_{T\text{-}Norm}$	$\alpha i^2\left(\frac{Vb1 - Vbo}{k\,ID}\right)^2 + \left(\frac{(Vb1 - Vbo)\,IN}{k\,ID^2}\right)^2 - 2\,\alpha i\left(\frac{IN}{ID}\right)\left(\frac{(Vb1 - Vbo)}{k\,ID}\right)^2$
a_{Io}	$T\,a_{T\text{-}Norm}$
a_{FMF}	$a_{T\text{-}Norm}$

Considering that the former simulation did not allow us to find out how the errors generated by the antecedent part of the rules are propagated we carried out a second simulation. We stress this aspect by considering now that two rules operate in a complementary way, i.e. the more one of the rules is fired, the less the other is. This condition is frequently met in controllers whose membership functions overlap considerably. However, in order to facilitate the analysis some simplifications were introduced. In Figure 3.33 the mirror before the denominator ID and the consequent+D/A circuits are now supposed to be error-free blocks.

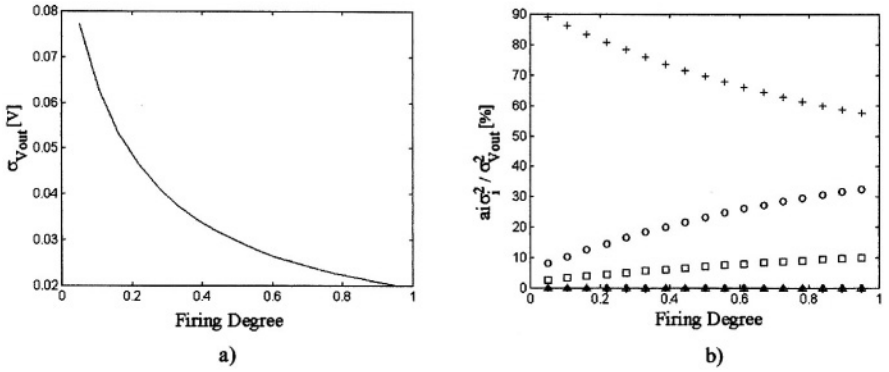


Figure 3.34. a) σ_{Vout} over the firing degree of the only active rule. b) Individual contribution of each building block to the total output variance: (+) divider; (o) consequent+D/A; (□) mirror before ID; (◊) T-Norm; (△) CFMF Type-I and (*) Io.

Consequently, the block diagram in Figure 3.33 was slightly modified by adding a second membership function FMF cascaded with another T-Norm. The output of this T-Norm is summed with the signal coming from the other rule (FMF+T-Norm) block at the common node yielding the signal Iout1 in Figure 3.33. The circuits used are the same as in the previous setup. However, while the consequent singleton was fixed to 0.9 in one rule ($\alpha 2$), it was now allowed to range from 0.1 to 0.9 in the other ($\alpha 1$). Simulations results are shown in Figure 3.35 a) and b). The main conclusions are summarized below:

- As the difference between the singleton values of both rules gets large (for instance $\alpha 1=0.1$ and $\alpha 2=0.9$), the mismatch errors of the antecedent part of the rules (FMFs + T-Norms) are more propagated to the output. This can be seen in Figure 3.35 a) and b). In contrast, the latter mismatch errors are not propagated when both singletons adopt the same values (i.e. $\alpha 1=\alpha 2=0.9$). When this happens, numerator and denominator of the divider remain constant. As in the former case, the normalization eliminates the deviations coming from previous stages when $\alpha 1=\alpha 2$.
- Notice in Figure 3.35 a) that the standard deviation of Vout ranges from 15mV (1.5%) to 25mV (2.5%). Nevertheless, these figures should be incremented by 1% (at least) when the mismatch errors introduced by the consequent+D/A and the mirror before the denominator ID (neglected for this simulation) are also considered.
- Figure 3.35 b) attests that the main error contribution is again caused by the divider. It has been demonstrated in section 3.3.1.3 that maximum deviations of the divider take place when its denominator input current

gets small. This explains the larger values of σ_{Vout} for the smaller firing degrees (<0.4) in Figure 3.34 a). In this second simulation, owing to the complementary rule activation mode, the denominator current remains constant and equal to Io=10μA. Thus, for a given pair of singletons α1 and α2 in Figure 3.35 a), σ_{Vout} remains almost independent of the firing degree of a rule and smaller than in Figure 3.34 a). Therefore, a good rule of thumb for keeping small the mismatch errors introduced by the divider is to avoid small denominator currents. This condition is ensured as long as the membership functions overlap considerably to allow a complementary activation of the neighboring rules. This is normally the case in a Fuzzy Controller.

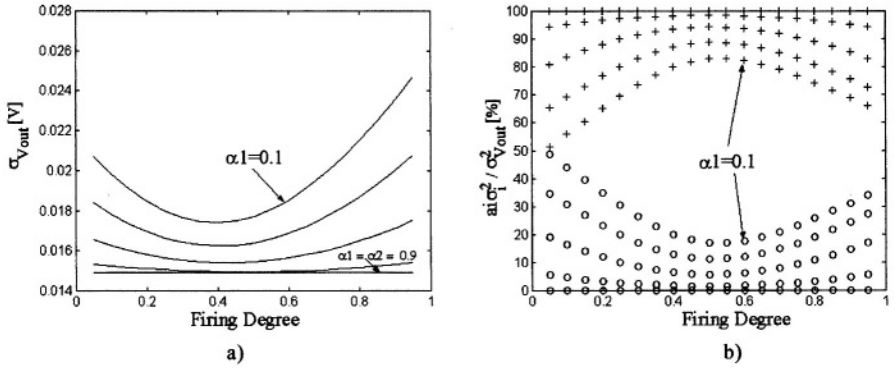


Figure 3.35. a) σ_{Vout} over the firing degree of a rule assuming two rules fired in a complementary way (α2=0.9 while α1 ranges from 0.1 to 0.9 in 0.1 steps). b) Individual contributions to the total output variance: (o) CFMF+T-Norm+Io and (+) divider.

In conclusion, the error analysis technique presented above allows us to get insight about the attainable accuracy in a Fuzzy Controller. It can be applied without taking care neither of the number of rules that take part in the process nor of their firing conditions. Certainly, each situation must be characterized with its own block diagram clarifying the corresponding signal path.

3.5 Conclusions

In this chapter, an analog framework for the hardware implementation of Fuzzy Logic Controllers has been established. It comprises mainly the analysis of the basic functional blocks. Significant issues concerning their performances can be drawn and used for the synthesis. The major items regarding the circuit accuracy, programmability, interfacing and complexity

have been focused. Some new circuits have been presented while others have been optimized in view of an improved behavior.

Two Complementary Fuzzy Membership Functions circuits built around regulated-cascode triode transconductors have been proposed. The main improvements with respect to classical approaches rely on the fact that an increased slope range can be obtained. Small slopes can be easily set without the need for very long channel saturated transistors. As a byproduct, slopes can be electrically tuned. Thus, these CFMF circuits are appropriate for controllers implemented in technologies allowing analog storage.

Two multiple-input MAXIMUM circuits have been found to suit optimally the interface requirements for implementing the T-CoNorm operators when the current-domain processing mode is adopted. The first one is the classical Lazzaro's WTA circuit. In this one, some modifications have been introduced in order to reduce the propagation errors. In the second circuit, while the maximum computation is performed between currents, the distribution of input signals is simply carried out by wiring. This avoids the use of repetitive current mirrors for the distribution of the input signals, leading to a considerably saving in routing space.

Two new multiple-input MINIMUM-LTA circuits have been presented. The first one realizes an exhaustive comparison between all inputs while delivering at its output a replica of the minimum input current. Despite its $O(N^2)$ complexity, its simplicity turns it attractive for being used in small fan-in cases. The second proposed MINIMUM-LTA circuit shows O(N) complexity and is more suitable for large number of inputs.

The choice of an open-loop architecture for the defuzzifier eliminates stability problems that could arise in the closed-loop counterparts. In contrast to open-loop pseudo-normalizers whose complexity may increase with the number of rules of the controller, the use of a divider permits to concentrate the optimization efforts in only one circuit. Furthermore, considering the linear operation implicated in a D/A conversion the adopted configuration permits the use of a single weighting operator. This defuzzifying scheme benefits from the savings in silicon surface, improved speed and better accuracy if compared with the approaches with a local D/A per rule.

A novel transresistive divider has been designed. It satisfies straightforwardly the interface requirement for voltage-mode output controllers. The division operation is actually performed by matched triode transistors operating in strong inversion. Therefore, a better transient behavior can be expected from this circuit.

With little modifications, the circuit of the divider can be adapted to implement an electrically tunable linear current mirror. This fits the requirements needed for the singletons of an analog programmable Fuzzy Controller. Programmability is easily achievable by linear setting of a bias

voltage. In addition, upon ideal matching conditions, the input-output relationship does not depend on technological parameters. This adds to enhanced robustness. Simulations predict good linearity and low offsets.

Finally, the explained methodology for the estimation of the global accuracy attainable in a given controller becomes a useful tool for optimization purposes at the design step. The contribution of each fuzzy operator to the total output error can be identified. Moreover, all rule activation situations commonly presented in a Fuzzy Controller can be modeled with reduced mathematical manipulations.

Chapter 4

MIXED-SIGNAL PROGRAMMABLE FUZZY LOGIC CONTROLLERS

Design, fabrication and test of demonstrators

4. INTRODUCTION

By employing the basic analog processing blocks studied in the previous chapter, the design of programmable analog Fuzzy Logic Controllers is addressed hereafter. An improved flexibility of the controllers allows using them in wide range of applications. Therefore, main topics concerning their architectures and programmability are discussed.

Lower power consumption, higher speed and reduced silicon die distinguish analog circuits from their digital counterparts [EiKü96] [WaDe90], for a given complexity of the intended controller. However, the parameters defining the controller's input-output relationships are also analog values, which must be stored and adapted during programming and/or tuning. From this point of view, analog controllers are more disadvantageous than the digital ones if considered the troublesome implementation of analog memories.

Nevertheless, a trade-off between accuracy and complexity is achieved if only a finite discrete set of analog parameters is provided. For instance, a voltage parameter can be settled by using a binary scaled set of currents sources that yields a discrete set of voltage drops in a linear resistor. Thus, we can use a digital memory to store the binary representation of each current within the set. This technique gives rise to the so-called mixed-signal analog computation circuits, which have been widely used for the implementation of Artificial Neural Networks and Fuzzy Systems [BaHu97-98] [MaHo94] [BoTs98] [Espe94] [LePa94] [MaRo98] [VaVi99]. It benefits

from the above mentioned advantages of analog circuits for signal processing together with the straight viability of digital circuits for storing purpose only. Certainly, the maximum achievable precision will be limited when using a finite set of analog parameters. Nonetheless, accuracy is not a major concern in a considerable number of reported Fuzzy Logic applications. Moreover, as long as learning procedures are being practiced the precision of the controller can be further improved.

We are intending programmable controllers with medium accuracy (i.e. 2-4%) and delays below the microsecond. Several Fuzzy Logic applications requiring such performance figures have been reported. For instance, in [FrMa98] a PWM DC/DC converter is controlled by using non-linear laws implemented in a Fuzzy Controller. The controller demands a small number of rules (i.e. 5 or 6) and fast settling time. This timing constraint makes an analog implementation of the controller very attractive, as it allows avoiding the additional delays due to A/D and D/A conversion. Another interesting application whose requirements can be fulfilled by our controllers is reported in [MoPi94]. In this, a Fuzzy Controller is used to tune a filter by attempting to place its frequency response curve within a typical window tolerance. The frequency response curve does not have to match exactly a given shape as long as it is found within the window. Therefore, the tolerance of the window makes relaxing the accuracy needed for the controller. In Chapter 5, we present another real-time Fuzzy Logic application in the domain of signal processing: the automatic channel equalization after digital transmission. We will show that even using a controller with a RMSE rounding 4% the equalization is achieved. Finally, let us recall that not only performance constraints justify the implementation of fuzzy algorithms on silicon. For instance, the need of cost-effective volume productions [OeGr96] [NaVi00] could also convey to ASIC implementations like the ones presented herein.

This chapter is divided in two sections. In the first one, a small Zero-Order Takagi-Sugeno's controller is presented. Only antecedent and consequent parameters can be programmed whereas the number of rules, inputs and outputs are fixed. The obtained performance measured in terms of power consumption, speed and area turns this controller attractive to be used as an on-chip subsystem. Besides its simplicity, this controller fulfils the requirements needed for several real-time applications like the ones reported in [FrMa98] [MoPi94] [BaDi00].

The second part is devoted to the design and test of a more complex programmable and reconfigurable architecture. In this case, not only antecedent and consequent parameters are programmable but also the number of inputs, outputs and rules can be chosen from a maximum limit. Moreover, with few manipulations of the inputs and outputs signals, First-Order Takagi-Sugeno's controllers can also be afforded.

In both implemented demonstrators, signal processing is performed in the analog domain, but their programming interfaces behave as static RAMs. These are easily accessible through a PC parallel port, which allows loading and verifying the controllers' parameters.

4.1 A 9-Rule, 2-Input, 1-Output Fuzzy Logic Controller

It has been shown that current-mode processing lends to simple rule-evaluation and aggregation circuits that can work at a reasonable speed [BaHu97] [VaVi99] [LePa94]. On the other hand, most reported real-time applications work in a voltage-mode controlled environment. If some of the unwanted current-to-voltage and/or voltage-to-current intermediate converters can be avoided, the delay through cascaded operators may be even shortened and higher speeds and accuracy achieved. This comment is interesting to take into consideration when fuzzifier and defuzzifier circuits are being designed, as these blocks are found at the controller's input/output interfaces. Fortunately, the circuits developed for the fuzzifiers (transconductor) and the output divider (transresistance) of the defuzzifier do not need extra signal conversion. Thus, they are able to directly interface the current-mode core of the controller with the voltage-mode environment.

In order to reduce die silicon area and power consumption some building blocks can be shared without altering their functionality. This is the case of the membership function circuit, whose output can be distributed through mirroring and can be used in several rules. On the other hand, the strategy adopted for the defuzzifying algorithm benefits from the use of only one divider and one weighting circuit for all rules, as explained in Chapter 3. This also leads to a low-complexity layout that contributes to improve the speed of the controller.

4.1.1 Architecture of the Controller

Figure 4.1 shows the block diagram of the 2-input, 1-output, 9-rule singleton controller and highlights the three well-known basic fuzzy operations: fuzzification, rules evaluation (inference) and defuzzification. These operations are being performed concurrently. The choice of a Zero-Order Takagi-Sugeno's architecture is sustained on the good trade-off between simplicity and accuracy that this simple fuzzy algorithm holds.

The fuzzifying step is performed by means of the complementary membership function circuit called CFMF Type-II in Chapter 3. Since it delivers complementary membership degrees, the MIN inference method should be reformulated by applying De-Morgan laws and using complemented T-CoNorms (MAX) operators. Thus:

$$\mathrm{MIN}(\mu_{A1}(x1),\mu_{A2}(x2),\ldots) = 1 - \mathrm{MAX}(1-\mu_{A1}(x1),1-\mu_{A2}(x2),\ldots). \tag{4.1}$$

A set of three Complementary Fuzzy Membership Functions (CFMF) per input, being shared by several rules, performs the fuzzification operation. This allows setting up a grid partition that yields nine rules. Next, nine 2-input complemented MAXIMUM operators perform the rules evaluation stage. The outputs of the MAXs are complemented to the current Io, which represents the logical value "one" in (4.1). This current is also the maximum value that a CFMF delivers at its output (i.e. the tail current of the transconductors that build the CFMF). After complementing the outputs of the MAX operators the firing degree of each rule is provided in the form of a current-signal Ii, for i=1…9.

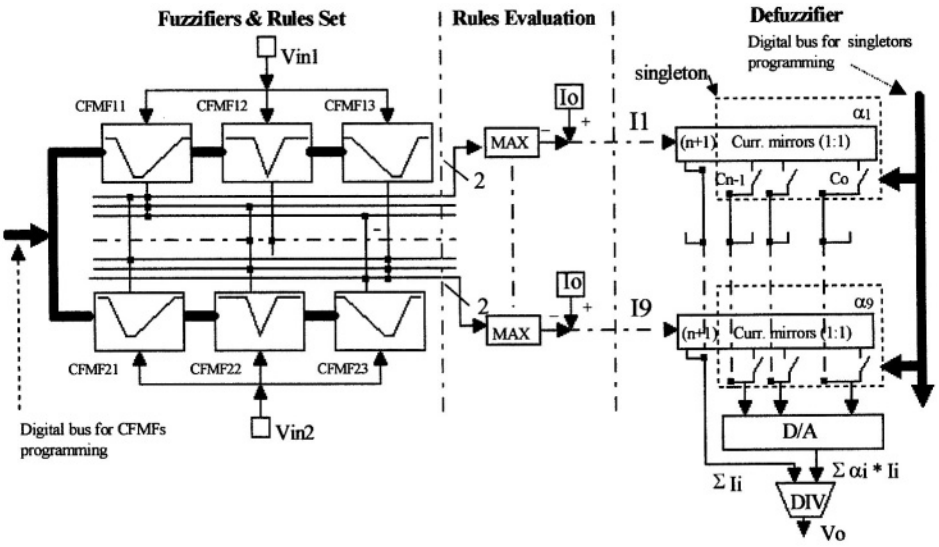


Figure 4.1. Block diagram of the 2-input, 1-output, 9-rule Fuzzy Controller. In this case n=5. From [DuVe00], © 2000 IEEE.

At the defuzzifier, each current Ii is replicated (n+1) times via unit-gain mirrors, where n stands for the resolution of the discrete-value αi of the consequent singletons of the rules (i.e. n=5). This value αi is codified according to the state of the switches $C_{n-1}\ldots C_0$. Finally, a shared current-mode digital-to-analog converter (D/A), used as a weighting operator, together with an analog divider take care of the computation of the Averaged Weighted Sum (AWS). Hence, the defuzzified output value Vo is:

$$Vo = k \frac{\sum_{i=1}^{9} \alpha i \times Ii}{\sum_{i=1}^{9} Ii}, \quad (4.2)$$

where k is a voltage-dimension constant defined by the transfer function of the divider itself.

4.1.2 Building Blocks: Sizing and Programming Strategies

In our designs, the sizing criterion for the circuits is mainly oriented towards the achievement of controllers with delays below 1μs, demanding an acceptable power consumption and die silicon area. Thus, it is desired that each fuzzy operator in the signal path presents delays ranging in the order of one or two hundreds of nanoseconds. Moreover, in view of an improved transient behavior, some circuits were modified from their previously presented version. These modifications will be explained and justified later.

Since programmability is provided to the antecedent and consequent circuits, the sizing strategy based on strictly minimizing the mismatch errors [VaVi99] can be relaxed to some extent. This strategy normally conveys to very large size transistors with increased stray capacitances and, consequently, with poor transient behavior. Therefore, we mainly prefer to use medium/small size transistors. This leads to controllers of medium accuracy where the mismatch errors could be further improved (to some extent) by subsequent tuning. However, whenever it was possible to reduce mismatch errors without compromising too much other performance figures of the circuits (i.e. time behavior, circuits size); or even when reducing mismatch errors became mandatory for the proper operation of some circuits, we used larger transistors area.

A common parameter for the whole circuit that must be firstly defined is the value of the logical "one", corresponding to the tail current Io of the membership function circuits. The choice of the value of this current is conditioned by several factors like power consumption, speed, circuit size, and noise. This is discussed in the following.

Using a large current Io becomes necessary when small delays are being intended. On one hand, the upper limit for Io is defined by the maximum admissible power consumption. On the other hand, as long as Io augments the gate-voltage overdrive of transistors follows the same trend and the available voltage swing for the signal is reduced, for a given power supply

value. This effect can be counteracted if larger transistors sizes were adopted but the silicon area and the stray capacitances would result enlarged.

In Chapter 3 we have demonstrated that the mismatch errors of the currents delivered by the membership function circuits, are proportional to the relative mismatch error carried by Io itself. The latter error is, in turn, inversely proportional to Io. When a small current Io is being set, mismatch errors could be attenuated by using larger transistors size [PeDu89]. However, this would not only demand more silicon area (even for medium accuracy controllers) but also the stray capacitances may result considerably large, which compromises the transient behavior of the circuit.

Another issue that precludes from adopting a very small current Io arises when noisy devices are contemplated. The rms thermal noise voltage referred to the gate of transistors is given by:

$$V_g^2 = 4KT\left(\frac{2}{3}\right)\frac{1}{gm}\, f_{max}\ (a),\quad V_g^2 \approx 4KT\,\frac{\beta\,(Vgs - Vt)}{gm^2}\, f_{max}\ (b),\quad (4.3)$$

where expressions (a) and (b) are valid in the saturation and triode region respectively, K is the Boltzmann's constant, T is the absolute temperature and f_{max} is the bandwidth of the circuit where the transistor is being used [JoMa97]. For a given transistor size its transconductance gm gets smaller as the biasing current is reduced. Let us consider the case of the input transistors of the membership function circuit. Suppose we want to uphold a given noise figure, which upon (4.3) is related to a given gm of the input transistors of the fuzzifier. However, working with smaller currents would imply decreasing the input voltage swing of the fuzzifiers if we want to sustain the same gm and consequently the same noise figure. In such a case, the input voltage swing would result smaller and a degradation of the input dynamic range of the membership function could be expected.

In the light of the above discussion and after performing some rough estimations on current consumption and transient response, we fixed Io to $10\mu A$.

4.1.2.1 Membership Function Circuit.

Because of its simplicity, the fuzzifier circuit called CFMF Type-II in Chapter 3 was used. Figure 4.2 illustrates the half of the circuit while showing in details the schemes adopted for discrete programming of slopes and knees.

The main specifications to meet are: input range, settling time (i.e. accordingly we stipulate the bandwidth BW of the fuzzifier), programmability range (slopes and positions), current consumption and

silicon area. We summarize hereafter the main items concerning the sizing, biasing and programming of this fuzzy block.

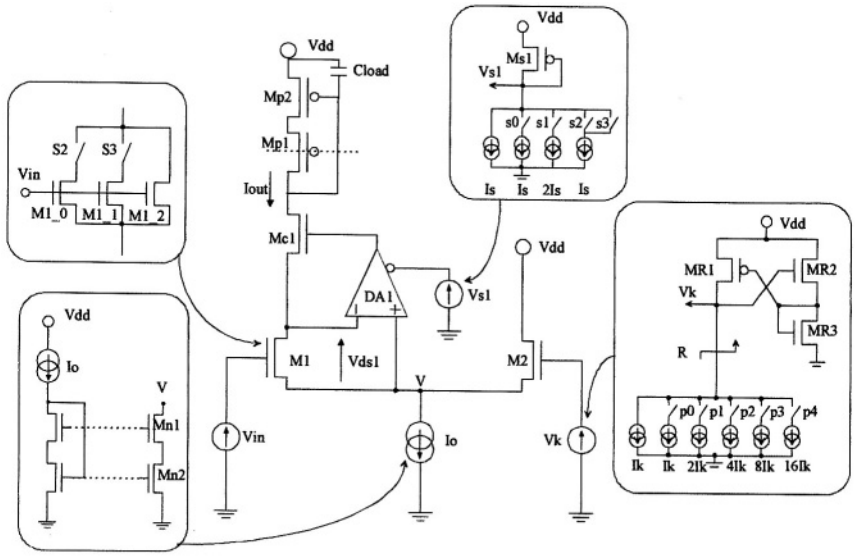


Figure 4.2. Half CFMF Type-II: biasing and discrete programming of slopes and knees.

The input voltage range of the fuzzifiers defines the input voltage range of the controller. A large input swing allows a large input dynamic range and, on the other hand, the possibility of performing a fine fuzzy partition (i.e. large number of fuzzifiers per input). From previous estimations carried out in Chapter 3 we concluded that, for Vdd=5V, an input voltage range $Vin_{Range}=3V$ (i.e. from 1.5V to 4.5V) is achievable with this circuit. This allows a wide range of slopes and positions for the CFMFs.

To maximize the flexibility of the circuit, the ratio between the maximum and the minimum available slopes was fixed to $S\approx 10$. Assuming the minimum slope corresponding to a non-symmetrical membership function ("shoulder type") extended all over the input range, the minimum transconductance needed is $gm1_{min}\approx Io/Vin_{Range}$. Therefore, the maximum transconductance is given by $gm1_{max}=Sgm1_{min}$. Trying to cover the mentioned slopes range by changing only $(W/L)_{M1}$ or Vs1 is not practical. For this reason we used in combination a set of differently sized transistors M1 together with a set of discrete values of Vs1 (or equivalently: a set of currents Id_{Md5} in the DA1, see Figure 3.2 b)). This, in turn, corresponds to a set of Vds1 values, as shown in Figure 4.2.

Since Ml must work in the triode region over a wide range of the input Vin, the maximum value allowed for Vds1 is limited by the input range Vin_{Range} (i.e. $Vds1_{max}=5\%$ to 10% of Vin_{Range}). The minimum value for Vds1

is limited by the minimum artificial offset that the non-symmetrical DA1 is able to set with a tolerable error at the drain-source terminals of M1. We have taken as a figure of error a typical random offset of a differential amplifier (Voff≈3 to 5 mV). Thus, $Vds1_{min}$ must be much greater than Voff.

From the previous calculated limits for gm1 and Vds1, we are able to estimate the maximum and minimum size for M1. The choice of the channel length of transistors M1 is a matter of further optimization taking into account other performance figures like the desired transient behavior (i.e. parasitic capacitances).

Transistor M2 is aimed to introduce the "knee" of conduction of the transconductor at approximately Vin=Vk. Therefore, its gate-voltage overdrive $GVO_{M2}=(2\ n\ Io/\beta_2)^{1/2}$ must be negligible.

Transistors Mn1 and Mn2 have been sized to supply the tail current Io with the compliance needed for allowing the minimum desired input voltage Vin at the membership function (see section 3.1.4.1 in Chapter 3).

Transistors Mc1, Mp1 and Mp2 have been sized in order to maximize the signal swing. This has been achieved by choosing for those transistors appropriate drain-to-source saturation voltages without yielding very large aspect ratios with increased parasitic capacitances. Cascoded-transistors Mp1 and Mp2 mirror the output of the CFMF towards the different MAX blocks where the fuzzifier takes part. For the desired transient response of the fuzzifier (i.e. settling time of 100ns to 200ns), it should be taken into account that transistor Mp2 will be loaded by several mirrors (i.e. Cload) as the membership function is shared by several rules. For this design, we assumed Cload corresponding to a fan-out of ten, approximately.

The sizing and biasing of the differential amplifier DA1 is conditioned by the desired transient behavior for the transconductor (i.e. settling time, bandwidth) and, on the other hand, by the DC specifications (i.e. artificial offsets). A comprehensive study of the time and frequency response of the regulated-cascode amplifier can be found in [FlVi97]. Their main conclusion is that the transition frequency of the loop amplifier must be larger than the bandwidth of the regulated-cascode amplifier itself. In this way, the dynamic behavior of the regulated-cascode loop does not tamper with the dynamic behavior of the main amplifier. However, in our case the constraints for sizing and biasing the loop amplifier DA1 are more relaxed. On one hand, since the transconductor is loaded with a diode it features small gain if not attenuation. On the other hand, the loop amplifier output is buffered with a source follower. Finally, the transconductor works in open loop.

The choice of the size of transistors Md1, Md2 and the range of the tail current Id_{Md5} (see Figure 3.2 b)) are conditioned by AC and DC constraints. On one hand, the transition frequency of DA1 must be greater than the bandwidth of the transconductor, for every value of the tail current of DA1,

Id_{Md5}, which changes upon programming. It should also be considered the compensation capacitance Ccomp needed at the gate of Md6 for stabilizing the loop. On the other hand, in order to yield the demanded artificial offsets (i.e. $Vds1_{min}<Vds1=GVO_{Md2}(_{IdMd5})-GVO_{Md1}(_{IdMd5})<Vds1_{max}$, for $Id_{Md5min}< Id_{Md5} <Id_{Md5max}$) the size of transistors Md1 and Md2 should be sharply discernible one from each other (i.e. $(W/L)_{Md1}=2(W/L)_{Md2}$) allowing clearly the generation of those artificial offsets. Md3 and Md4 should be sized in agreement with the signal swing needed at the gate of Md6. This mirror also introduces a non-dominant pole that must be located beyond the transition frequency of the differential amplifier [LaSa94]. Current Ip and transistor Md6 in Figure 3.2 b) must be estimated so as to shift the output level of the differential amplifier to the voltage level required at the gate of Mc1 in Figure 4.2. The choice of Ip is related to the bandwidth needed for the source follower that must be larger than the bandwidth of the differential amplifier. After estimating Id_{Md5max} the size of Md5 is derived in order to minimize the compliance of the tail current source of DA1 and maximize the signal swing.

In view of a uniform allocation of the membership function along the input Universe of Discourse, a discrete set of uniformly spaced voltages Vk is required. For this purpose, a binary-scaled set of currents yields an identically scaled set of voltage drops across a self-biased linear resistor R. This comprises transistors MR1, MR2 and MR3 in Figure 4.2 [GrTe86]. The particular connection between these transistors guarantees the triode operation of MR1. In this way, the value of R is given by [GrTe86]:

$$R = \frac{1}{\beta_{MR1}\left((1-n/2)Vdd + (n-1)VTn - |VTp|\right)}, \text{ if } \frac{\beta_{MR3}}{\beta_{MR2}} = \left(\frac{2}{n} - 1\right)^2. \tag{4.4}$$

The choice of a relatively large R allows minimizing the value of the current Ik in view of a minimum current consumption. Figure 4.3 shows some membership function shapes measured with a HP4145 equipment. Table 4.1 summarizes the features of this circuit.

Concerning the transient behavior of the circuit two step response simulations (large signal swing) have been carried out with SPICE by setting the maximum and the minimum transconductance in each one. In both cases the output current swing was allowed to reach $10\mu A$. The output of the circuit has been loaded with a capacitor that represents the load corresponding to the desired fan-out. Notice from Figure 4.4 a) and b) that the settling time (i.e. to reach the 90% of the steady state value) ranges from 100ns to 150ns. In Figure 4.4 a), the overshoot at the beginning of the falling edge is due to a transmission zero present in the transfer function of this

circuit. Since transistor M1 in Figure 4.2 works in the triode region, its gate-to-drain capacitance Cgd is not negligible. The magnitude of the transmission zero is given by the ratio gm1/Cgd, gm1 being the transconductance of the triode-transistor M1. As gm1 becomes smaller, the position of the zero is shifted to the low frequencies while conveying the circuit to get derivative properties. This is the case corresponding to Figure 4.4 a) where the value of gm1 is the smallest. The occurrence of this overshoot must not be considered as a drawback. On one hand, in this simulation the circuit has been driven by an ideal input step with zero rise and fall times. In a realistic situation, input signals are band-limited. Rise and fall time are non zero and a smaller overshoot can thus be expected. On the other hand, the circuits following the membership functions (i.e. MINIMUM or complemented MAXIMUM) impel the propagation of this overshoot towards the subsequent stages.

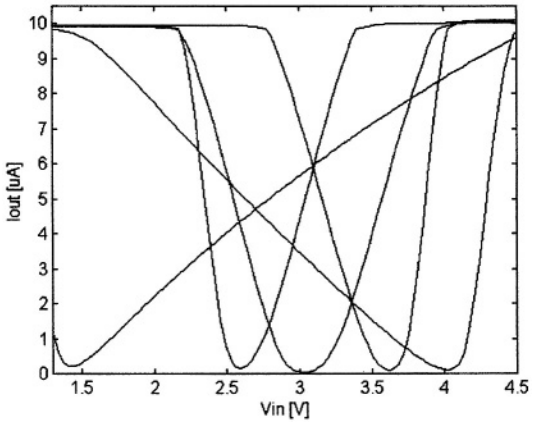


Figure 4.3. Measured characteristics of the CFMF Type-II: symmetrical and non-symmetrical shapes with small and large slopes. From [DuVe00], © 2000 IEEE.

Table 4.1. CFMF Type-II: Transistors sizing (in [µ/µ]), biasing and global performance.

Transconductor (#2)								
Input Range	Io	M1_0	M1_1	M1_2	Mp1 Mp2	Mn1 Mn2	Mc1	M2
1.5V<Vin<4.5V	10µA	12/12	20/4	12/12	24/3	45/3	22/3	45/3

Differential Amplifier DA (#2) (see figure 3.2 b))						Slopes Programming (#2)		
Ip	Md1	Md2	Md3,4	Md5	Md6	Is	Ms1	Msw_s0-s3
10µA	2 x 10/8	10/8	35/4	40/4	30/2.4	4µA	40/4	4.8/2.4

Knees Programming (#2)					Performance of the CFMF	
Ik	MR1	MR2	MR3	Msw_p0-p4	Current Consumpt.	Active Area
0.5µA	4/25	18/12	7/12	4.8/2.4	220µA (max)	3800μ^2

According to Figure 4.4, rise and fall times limit the maximum achievable speed of the CFMF. These delays are associated to the time constant of the CFMF output node whose large capacitive load Cload is due to the multiple mirrors distributing the current delivered by the fuzzifier.

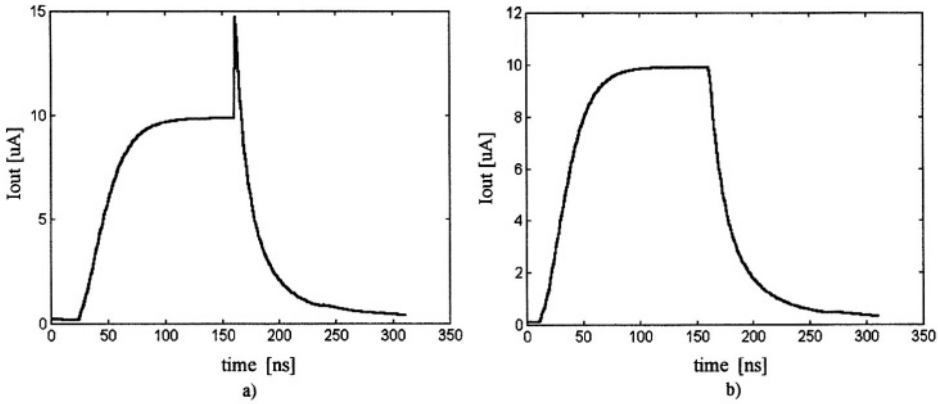


Figure 4.4. Transient response of the CFMF Type-II for a 10μA output swing and Cload=700fF. a) Minimum M1 transconductance. b) Maximum M1 transconductance.

4.1.2.2 MAXIMUM Circuit.

The Lazzaro's WTA-MAX circuit presented in Chapter 3 was adopted for the T-CoNorm operator. However, some modifications have been introduced in order to speed up the transient response of this circuit.

Figure 4.5 a) shows the complete schematic of this operator including the circuit that complements its output. The latter comprises Mio1, Mio2, which convey Io, and replicas of transistors M4 and M5 that convey Imax. This is needed for transforming the maximum into the minimum according to (4.1). It can be noticed that two extra diode-connected transistors M2 and M3 have been connected in series at each input. Without these diodes, the gate voltage of transistor M1 would fall more deeply under the loser condition of the corresponding cell. This is because, in a loser cell, transistors M4 and M5 are kept in the triode region whereas transistor M1 is switched off. To avoid this effect, those diodes introduce a supplementary voltage drop of at least 2VTn that boosts up the voltage level at the gate of transistors M1 of the loser cell. In this way, the recovery time of the cell (when the cell passes from loser to winner) is improved. This is now possible because the voltage swing needed to reach the voltage level that switch M1 on is considerably reduced.

The improvement introduced in this circuit can be appreciated in Figure 4.6 c). It shows several transient SPICE simulations that have been performed by replacing the diodes by an ideal voltage source E whose value

is allowed to change. Notice that a reduction of 60ns in the recovery delay together with a smaller undershoot can be achieved if E is set high enough.

A practical limit for the maximum tolerable E value is determined by the threshold voltage VTn of M1, taking also into account the spreading of the latter parameter. As shown in Figure 4.6 b), in the loser cells, the gate-to-source voltage drop of M1 must remain smaller than VTn.

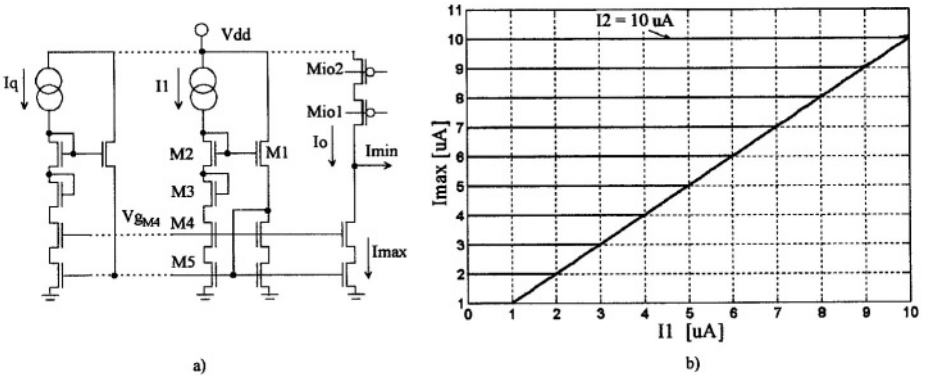


Figure 4.5. a) q-input complemented MAXIMUM circuit [DuVe00]. b) DC nested-sweeps simulation for a two-input circuit: I1 ranges from 0 to 10μA while I2 from 1 to 10μA by 1μA steps.

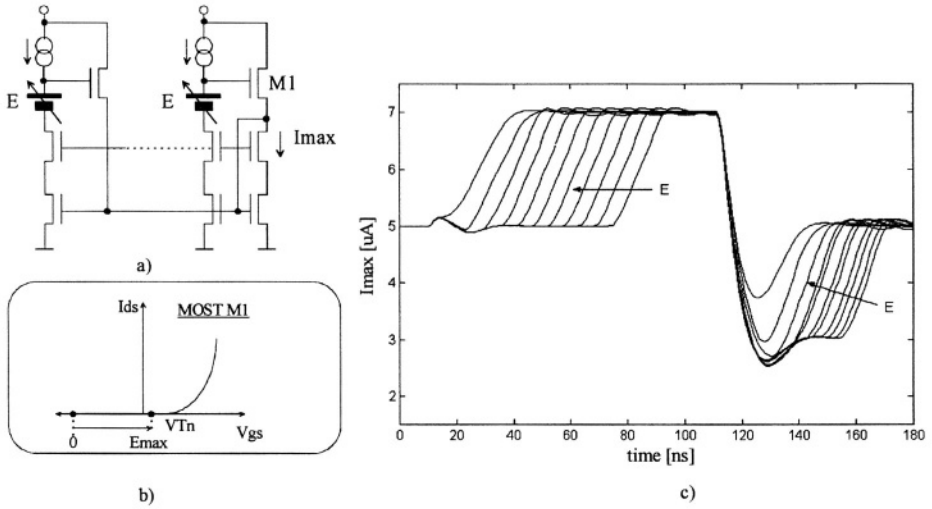


Figure 4.6. Delay improvement in the WTA-MAXIMUM circuit: a) Circuit schematic. b) Vgs range allowed for the loser transistors M1. c) Transient SPICE simulation for 0<E<2V.

Transistors size and bias should be correctly set in view of the proper operation of the circuit over the whole range of the input currents. This is accomplished if the following relation is held for any value of the input current Iin ($0 \le Iin \in \{I1 \ldots Iq\} \le Iin_{max}$):

$$\left(n + \frac{1}{n} - 1\right)\sqrt{\frac{2\,n\,Iin}{\beta 4}} \le (n-1)(Vg_{M4} + VTn) + \sqrt{2\,n\,Iin}\left(\frac{1}{\sqrt{\beta 1}} - \frac{1}{\sqrt{\beta 2}}\right), \tag{4.5}$$

where n accounts for the Body effect whereas all transistors are assumed being biased in strong inversion.

Notice that M5 does not take part in the above relation. Its aspect ratio can be chosen so as to keep its active area (W×L) considerably large. This warrants small mirroring mismatch errors. However, an upper limit for the size of M5 is imposed by the transient performance desired for the cell. The output common node of the MAX circuit is loaded with the gate-to-source capacitance of transistors M5 (Cgs_{M5}). When the number of inputs of the circuit is large (i.e. N-input MAX), this node results considerably loaded and the transient behavior of the circuit deteriorated.

Transistor M4 must hold a relatively small saturation drain-to-source voltage (Vds_{satM4}) that allows maximizing the signal swing. This transistor is aimed for cascoding in order to improve the systematic mirroring error.

Diode transistors M2 and M3 should be sized as large as possible in order to minimize their gate-voltage overdrive. This ensures the desired signal swing at the input branches in accordance with the drain-to-source voltage drops demanded by transistors M4 and M5. However, an upper limit for the size of those diode transistors is given by their time constant (the ratio between their gate-to-source capacitance and their transconductance) that must remain smaller than the maximum delay expected for the cell.

Finally, the size of transistor M1 must be chosen in order to satisfy (4.5) for every value of the input signal Iin within its range.

For a 2-input MAXIMUM circuit transient simulations performed for different input conditions reveal total delays shorter than 50ns. Figure 4.5 b) shows a DC simulation of a 2-input MAX circuit for the whole range of the input currents. Table 4.2 summarizes the performance of this circuit.

Table 4.2. 2-Input MAXIMUM: sizing (in [μ/μ]) and performance for 0≤Iin≤10μA.

2-Input MAXIMUM circuit (#9)								
M1	M2,3	M4	M5	Mio1,2	Io	Delay	Current Consump.	Active Area
12/4	24/2.4	12/2.4	10/10	15/4	10μA	~50ns	40μA (max)	773μ^2

4.1.2.3 Defuzzifier: Singletons, Weighting D/A and Divider.

Figure 4.7 a) shows an array of 5+1 unit-gain cascoded current mirrors corresponding to a discrete singleton in Figure 4.1. At each mirror, cascode-transistor Mc1 ensures low systematic errors.

Each singleton in Figure 4.7 a) represents to the previous MAXIMUM circuit, which furnishes the current Ii, a load capacitance of $7Cgs_{Mc2}$ in parallel with a diode resistance ($i.e. \approx 1/gm_{Mc2}$). Thus, the size of Mc2 results after trading low mismatch errors (i.e. large Mc2 area) for short delays. With the adopted size for Mc2, the simulated settling time for a singleton reaches to 20ns for an input step ranging from $1\mu A$ to $10\mu A$.

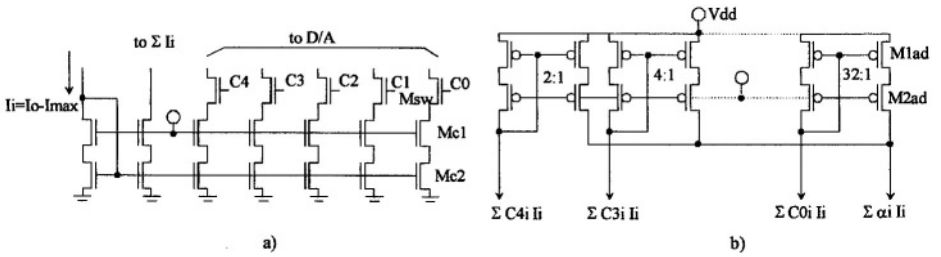


Figure 4.7. a) 5+1 unit-gain current mirrors conveying the firing degree of a rule "Ii". The value of the singleton αi is set through C0…C4. b) Weighting D/A circuit [DuVe00].

Figure 4.7 b) shows the circuit of the weighting D/A, which comprises a set of five independent binary-scaled current mirrors that sum their currents in a common output node. In Chapter 3, it has been demonstrated that the mismatch errors introduced by each bit of the D/A depend on the weight of the bit. We profited from this property for sizing each mirror of the D/A independently by assigning larger transistor areas to the bits that introduce the larger fraction of the total mismatch error. However, the maximum size for the transistors of the D/A is constrained by the desired transient response according to the time constant of each mirror. The simulated D/A settling time reaches to 60ns for an input step Ii ranging from $1\mu A$ to $10\mu A$.

The choice of consequent singletons αi smaller than one is sustained on the saving on current consumption that can be achieved. On the other hand, the divider can be optimally sized and biased because its inputs ID and IN can, consequently, range evenly. In the opposite case with singletons $\alpha i>1$, for a 5-bit resolution and a 1V output swing, the difference (Vb1–Vbo) should be set to 32.2mV to manage representing the maximum quotient IN/ID=31. Thus, the biasing of the divider would become delicate.

Concerning the divider shown in Figure 4.8 a), it has been sized in order to allow the input currents IN, ID to range from 0 to $30\mu A$. The maximum input current ($30\mu A$) was estimated under the assumption of three

simultaneous fired rules with the maximum firing degree (i.e. $Ii=10\mu A$). The choice of (Vb1–Vbo)=1V gives rise to a 1V output swing. For the given input/output ranges, the size of transistors M1,2,3 and the maximum drain-to-source voltage drops Vds_{max} that warrants their triode operation are mutually constrained by the following relationship:

$$\beta_{M1,2,3} Vds_{max} = \frac{ID_{max}}{Vb1 - Vbo}. \tag{4.6}$$

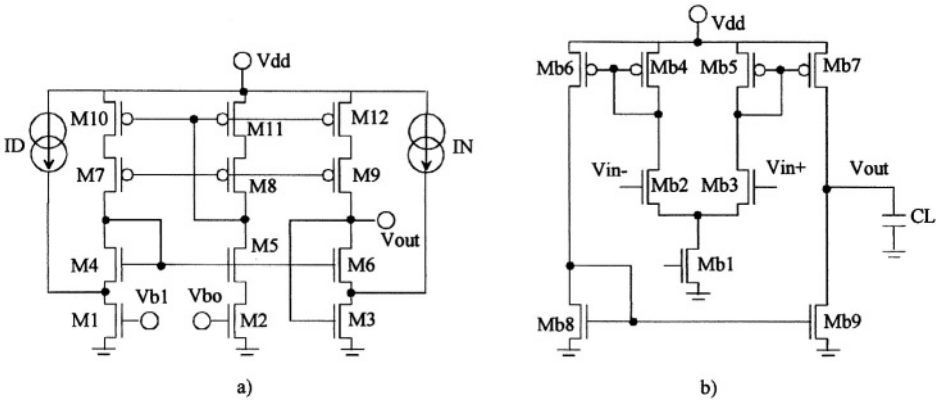


Figure 4.8. a) Current-input voltage-output analog divider. From [DuVe00], © 2000 IEEE. b) Output buffer.

A high Vds_{max} yields smaller transistors M1,2,3 aspect ratio with reduced parasitic capacitances. The relative incidence of the mismatch of transistors M4,5,6 on the triode-transistors Vds mismatch is also reduced. However, in this case, Vbo, and consequently Vb1 should be increased in order to sustain the triode condition of M1,2,3. Therefore, the mobility reduction in the latter transistors is stressed. In contrast, a low Vds_{max} allows smaller M1,2,3 gate voltages while relaxing the mobility reduction. Nonetheless, the aspect ratio of triode transistors grows leading to increased stray capacitances. In addition, for a low Vds_{max} the mismatch of the threshold voltages of transistors in the middle layer (M4,5,6) strongly affects the matching properties of the drain-to-source voltage drops of the triode transistors.

The transient behavior of the divider is considerably affected by the stray capacitances of the triode transistors. Increasing their aspect ratio makes proportionally increase their stray capacitances. However, the current at the central branch does not increase in the same proportion. Thus, large size triode transistors yield larger delays. On the other hand, the use of large area triode transistors is encouraged for the mismatch and noise reduction. In

summary, the actual size of transistors M1,2,3 is defined after trading speed, systematic errors and mismatch errors.

As commented in Chapter 3, by cascoding the upper PMOS mirror of the divider its systematic gain error and offset are minimized. This is achieved by means of the low-compliance cascode configuration shown in Figure 4.8 a) (M7 to M12). On the other hand, it is desirable that the time constant introduced by this mirror remains lower than the expected settling time for the divider. Thus, M7-M12 must be kept operating in strong inversion. This also ensures small mirroring errors.

Transistors M4,5,6 become responsible of the equalization of the drain-to-source voltage drops Vds of the triode-transistors M1,2,3. Bearing in mind that the correct operation of this translinear circuit lies mainly on the good matching conditions of triode-transistors M1,2,3, biasing M4,5,6 in strong inversion while keeping their areas relatively large is also recommended.

Figure 4.9 a) shows some measured and calculated characteristic curves of the divider. In Figure 4.9 b) the relative errors are shown.

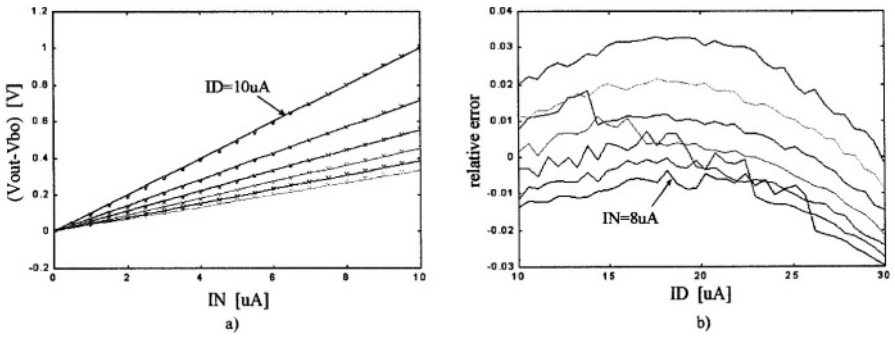


Figure 4.9. Performance of the divider: a) (Vout–Vbo) over IN, for 0≤IN≤10μA and ID from 10 to 30μA by 4μA steps: measured (x), calculated (-). b) Relative errors over ID, for 10 μA≤ID≤30μA and IN from 2μA to 8μA by 1μA steps. From [DuVe98], © 1998 IEE, reprinted with permission.

The settling time of the divider is obviously affected by the capacitive load at its output. Since this circuit is the last stage of the controller, the use of a buffer is recommended for dealing with large capacitive loads. The circuit of the buffer, which corresponds to a single-stage OTA, is shown in Figure 4.8 b). It has been calculated in order to yield a 1 Vpp step on 5pF in a 20ns settling time, approximately. Figure 4.10 shows a SPICE transient simulation of the cascade divider-buffer. It can be noticed that the settling time of the divider itself (Vout-div.) results near to 60 ns.

Table 4.3 summarizes the performance of the set of circuits belonging to the defuzzifier stage, including the output buffer.

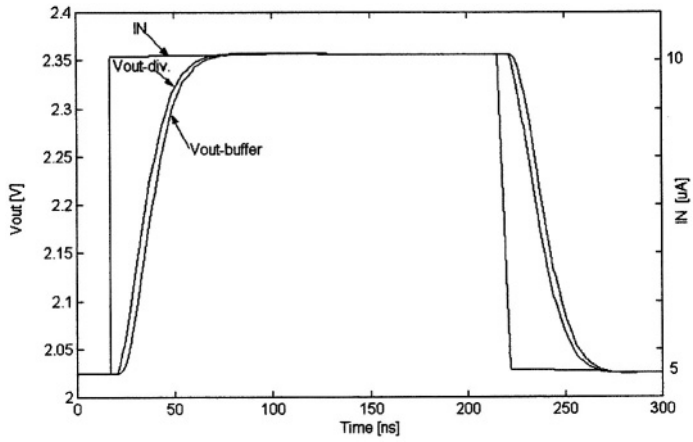


Figure 4.10. Simulated transient response of the divider followed by the buffer (@CL=5pF). ID is held constant to 15µA whereas IN is a pulse from 5µA to 10µA.

Table 4.3. Performance of the circuits of the Defuzzifier. Transistor sizes are given in [µ/µ].

Singletons (#9)					
Mc1,2	Msw	Resolution	Current Consumption		Active Area
10/4	6/3	5 bits	70µA (max)		650µ^2
Weighting D/A (#1)					
Mad1,2 bit 0,1	Mad1,2 bit 2	Mad1,2 bit 3	Mad1,2 bit 4	Current Consumption	Active Area
5/5	10/5	20/5	28/5	30µA (max)	4672µ^2
Divider (#1)					
M1-3	M4-6	M7-12	Input Range	Vb1	Vbo
10/10	20/8	28/4	0≤ IN, ID ≤30µA	2.7V	1.7V
Output Swing		Current Consumption		Active Area	
0≤(Vout-Vbo)≤1V		30µA (max)		1452µ^2	
OTA Output Buffer (#1)					
Mb1	Mb2,3	Mb4-7	Mb8,9	Adc	fT (@CL=5pF)
1200/4	375/3	60/6	60/9	47dB	50MHz
Phase Margin		Current Consumption		Active Area	
62°		1mA		9570µ^2	

4.1.3 Test Results of the 9-Rule Controller

Figure 4.11 shows a block diagram corresponding to the measurement set-up used for testing the prototype. The DC characterization has been

carried out with the HP4145B equipment that furnishes the voltage sweeps for the inputs Vin1, Vin2 of the controller while measuring its output Vout. The controller is programmed from a PC through a parallel port that outputs the antecedent and consequent digital codes. Stored data can also be read from the PC in order to verify the correct working of the internal shift registers of the chip. The measured analog data is retrieved through the HPIB bus to the PC. The transient characterization of the controller was performed by measuring the settling time of the controller's output in response to an input voltage step supplied by the pulse generator HP8112A.

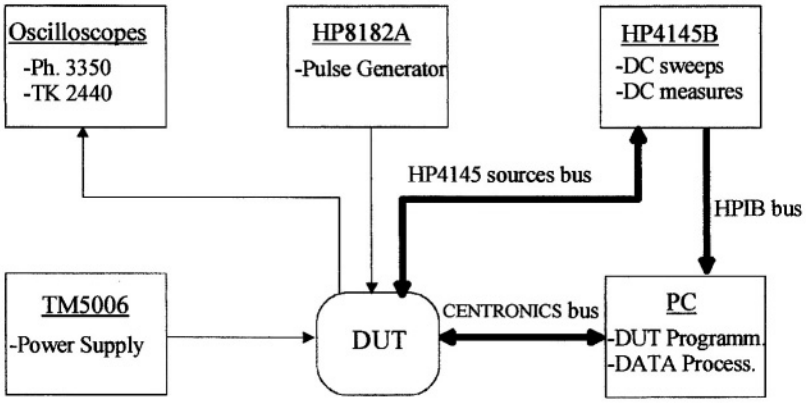
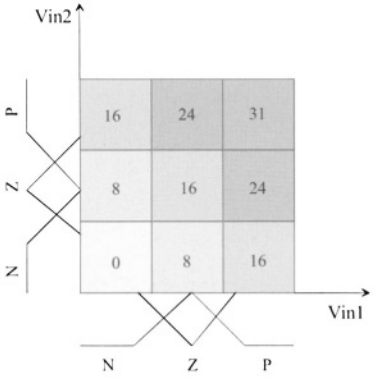


Figure 4.11. Block diagram of the measurement set-up.

Figure 4.12 a) shows the rules map and the digital codes of the consequent singletons that correspond to a particular non-linear transfer function, which is widely used in Control applications. Fuzzy labels defined for each input variable are sketched around the rules map. For clarity, direct membership functions are shown but the controller actually works with the corresponding complementary ones (CFMF). Figure 4.12 b) shows the values of relevant electrical parameters of the controller as well as the loaded digital codes of the membership functions parameters for both inputs.

Figure 4.13 a) illustrates the target output surface, which has been approximated after further manipulation of the nominal digital values of some parameters previously defined in Figure 4.12 a) and b). This task has been accomplished by simple inspection, without following a formal adapting strategy. Figure 4.13 b) shows the measured output surface. The relative errors between target and measured surfaces are shown in Figure 4.13 c). These errors vary along the input domain but the magnitude of most of them is found within a ±4% tolerance. This is shown by the histogram in Figure 4.13 d). However, error peaks near 7% can also be found. Taking a close look at Figure 4.13 c) one can realize that most error

peaks are found at the boundaries of the fuzzy clusters (rules). This is normally due to misplacement of the fuzzifiers or mismatch between their slopes.



a)

Electrical Settings:				
Vdd	Io	Vbo	Vb1	Singleton Resolution
5V	10 A	1.7 V	2.7V	LSB=(1/32) (Vb1-Vbo)=31.25mV
Input Range:				Output Range
1.5V < Vin1, Vin2 < 4.5V				1.7V < Vout < 2.7V

Inputs Fuzzy Labels: Parameters Digital Codes

	pr	sr	pl	sl
N	0	15	14	2
Z	11	15	17	15
P	11	1	31	15

- pr, pl: right and left knees
- sr, sl : right and left slopes

b)

Figure 4.12. a) Rules map, input fuzzy partitions and digital codes of the consequent singletons of each rule. b) Electrical settings of the chip and digital codes of the CFMFs parameters of both inputs.

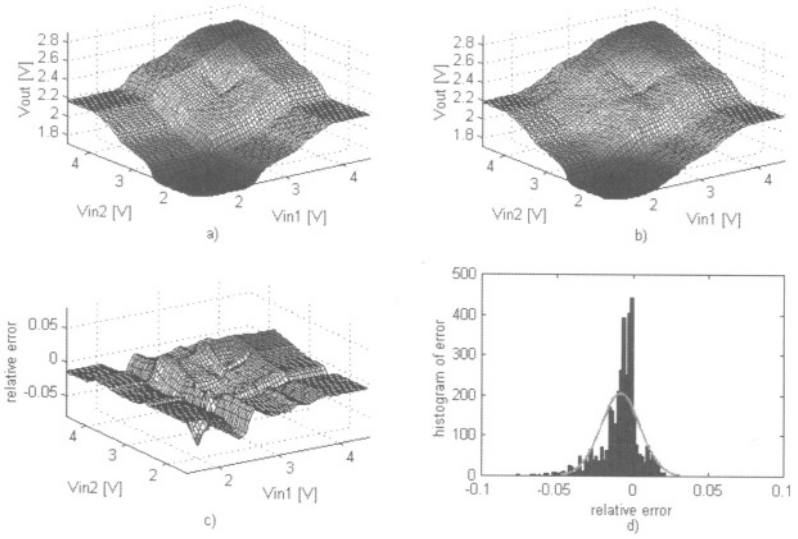


Figure 4.13. DC test results: a) Target output surface. b) Measured output surface. c) Relative error surface. d) Relative error distribution. From [DuVc00], © 2000 IEEE.

In a wide range of applications, instead of estimating local deviations, the precision of the controller is typified by means of the Root Mean Square Error (RMSE). This error figure gives an idea on the global accuracy of the controller. Calling N the total number of measured points at the output surface, the RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{i=N} \left(Vout_{measured} - Vout_{target}\right)_i^2}{N}}. \qquad (4.7)$$

For the case of Figure 4.13 a) and b) the RMSE reaches to 27mV that referred to the full-scale output swing gives rise to a 2.7% "global" accuracy. To get an idea about the mismatch errors due to the technological parameters spreading, the characterization of the statistical deviation between samples has also been addressed. Several prototypes holding the same settings have been measured. At each point of the input space the mean value and the standard deviation have been calculated as follows:

$$\overline{Vout(x,y)} = \frac{\sum_{1}^{P} Vout(x,y)}{P}, \sigma_{Vout}(x,y) = \sqrt{\frac{\sum_{1}^{P}\left(Vout(x,y) - \overline{Vout(x,y)}\right)^2}{P}}, \qquad (4.8)$$

where x and y stand for Vin1 and Vin2, respectively, whereas P is the number of tested samples. Since the above figures depend on the value of the inputs, the distribution of the standard deviation has been characterized. While most values of σ_{Vout} are smaller than 0.055V (5.5%), the standard deviation features a peak of 0.062V (6.2%) and a mean of 0.035V (3.5%).

Finally, the transient behavior of the controller has been characterized by measuring its large and small swing step responses. In Figure 4.14 a) a 500mVpp step applied to one input yields a 100mVpp step at the output. The other input is held constant. The measured delay reaches to 70ns and the rise time (i.e. to reach the 90% of the steady state value) to 120ns. Thus, the total settling time reaches to 190ns (90% of the steady state value). In Figure 4.14 b), a 3Vpp step was applied at the input whereas the controller outputs a 500mVpp step in 450ns (90% of the steady state value), which encompass 180ns of delay plus 270ns of rise time. The measured current consumption of this controller rises to 2.68mA.

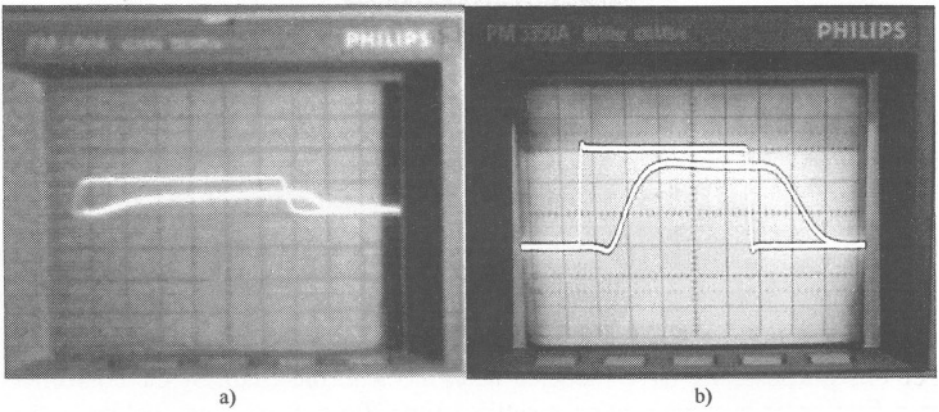


Figure 4.14. Step response of the controller: a) Small swing step response. Vertical scales: input=0.5V/div - output=0.2V/div. Time scale: 100ns/div. b) Large swing step response. Vertical scales: input=1V/div - output=0.2V/div. Time scale: 200ns/div.

4.1.3.1 Discussion.

Let us discuss the differences between the measured spreading figures and the ones estimated in section 3.4 of Chapter 3. Several aspects must be taken into account for a comparison. Firstly, the controller model used in section 3.4 is a simplified version. It does not take into account, for instance, the analog circuits used for programming and biasing the fuzzy blocks. Moreover, we must also consider a certain amount of mismatch due to the long distance between devices that has been neglected for the simulations. Therefore, the non-modeled mismatches do certainly add to a larger dispersion figure than the one estimated in that section.

In order to detect the critical blocks that yield the larger error fractions let us now consider the measured RMSE. One can assume that after tuning the remaining errors are due to mismatch only. Under this assumption, we can use the results of the simulations performed in section 3.4 in order to distribute the total error between the different blocks. In this way, the contribution of each block to the total measured RMSE (2.7%) is: CFMF(0.27%)→MAX(0.27%)→Cons+D/A(0.9%) →Divider(1.26%).

The divider becomes thus the critical block affecting the precision of the controller. Increasing the accuracy of the divider (i.e. by working with higher denominator currents and/or using larger transistors active area) will certainly improve all error figures. Fortunately, since it is only one circuit, the impact on the total size and consumption of the controller will be quite low if the proposed solutions are introduced.

In general, RMSE and relative errors can be further improved by increasing the resolution of the antecedent and consequent parameters (i.e. in view of a fine tuning) and/or using transistors with larger active area for

minimizing mismatch errors. Certainly, those solutions impact strongly on the size of the controller and the total storage capacity needed in a mixed-signal design philosophy. Anyhow, the measured error figures of this controller fulfill the requirements for the kind of applications quoted at the beginning of this chapter. However, if more accurate controllers are needed, one should orient the design towards the minimization of the deviations, specially in the case when the controller works in open loop (i.e. non-linearities correction of sensors using fuzzy networks [HuKe95]).

Considering the transient behavior of the controller we perform in the following an intuitive analysis in order to identify critical sources of delay. For this purpose, we have modeled each block in the chain of the controller as a first-order system holding a settling time (large signal) equivalent to the one obtained by simulations for each block in the previous sections. After a step response MATLAB simulation, we could notice that the output of the controller reaches 90% of the steady state value in approximately 330ns. One can assign the difference between measured and simulated total delays (large signal) to several issues that have not been modeled for the simulation. On one hand, the extra delays introduced by the routing stray capacitances and other stray capacitances present in the actual layout may represent a considerable fraction of the total measured delay (i.e. just consider the 2.4μ technology that was used). On the other hand, particularly in non-linear circuits, the delay of each block depends on the working conditions and excitation of the block (i.e. signals swing, biasing, etc). It is highly probable that the blocks working conditions assumed for the transient simulation do not match the actual working conditions of the same blocks in the chip, which are hardly difficult to identify. Finally, the actual load capacitance at the output buffer during measurements may have been larger than the one estimated for the simulation.

According to the simulation result, the distribution of the total delay among the cascaded fuzzy blocks gives rise to the following individual contributions in percent: CFMF(32.5%)→MAX(13.5%)→Cons+D/A (21.6%)→Divider(21.6%)→Buffer(10.8%).

Since the CFMFs are multiple-output, they are considerably loaded, as analyzed in the corresponding previous section. Thus, these circuits are the critical sources of delay. A solution that may improve the transient behavior of the fuzzifiers consists in using active mirrors (i.e. buffered mirrors) for distributing the output current of the CFMF. In this way, the total current consumption and area will be increased by the extra amount demanded by the active mirrors only (one per fuzzifier).

4.1.4 Summary and Comparison with Other Approaches

Figure 4.15 shows the microphotograph of the controller. Its core occupies $4.5\ mm^2$ including the digital circuits, which represent almost 50% of the total area. Since digital data is stored in serially loading shift registers (one for the antecedents and another for the consequents), the external digital interface of the chip comprises only two bits. Table 4.4 summarizes the main measured features of this demonstrator.

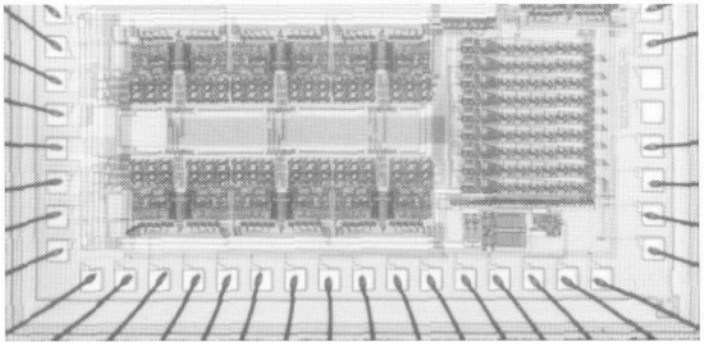


Figure 4.15. Microphotograph of the 9-rule Fuzzy Controller. From [DuVe00], © 2000 IEEE.

Table 4.4. Measured performance of the fabricated 9-rule Fuzzy Logic Controller.

9-rule Fuzzy Logic Controller		
Technology:	CMOS-2.4μ	
Complexity:	9-rule@2-input@1-output	
Interface (inputs@outputs):	voltages@voltages	
Power Supply:	Vdd=5V	
Power Consumption:	Core:	8.4mW
	Buffer:	5mW
	Total:	13.4mW
Area:	Analog:	$2.3mm^2$
	Digital:	$2.2mm^2$
	Total:	$4.5mm^2$
Accuracy:	RMSE:	27mV (2.7% output swing)
Programmability (on-chip):	CFMF Slopes:	2x4 bits
	CFMF Knees :	2x5 bits
	Singletons:	5 bits
Total storage capacity needed:	153 bits	
Input/Output Delay:	Small swing:	190ns
(90% of steady state value)	Large swing:	450ns
Standard deviation between samples:	Maximum:	62.5mV (6.25%)
	Mean:	35mV (3.5%)

According to the latter table, the processing speed of this controller ranges from 2.22 to 5.26 MFLIPs (Mega Fuzzy Logic Inference per Second). This feature must be taken as a proof of the optimal strategy adopted for the design, namely the avoidance of intermediate current-to-voltage and/or voltage-to-current converters, the reduced complexity of the defuzzifier and the simplicity of the divider. Other relevant figures could be mentioned: 1.5 mW of power consumption per rule, a relatively low power-delay product of 5.6 nWs and 153 bits of memory needed for completely programming the chip. These measured performances render this controller attractive for being used as embedded subsystem for medium-speed, medium-accuracy and low-power applications.

Table 4.5 reveals the main features of other approaches found in the literature. Before performing any comparison however, it should be taken into account the different number of rules, inputs, outputs and, above all, the different technologies that have been employed for implementing the demonstrators in each case. In addition, the lack of key information in some of the cited references should be noticed.

Table 4.5. Main features of other analog implementations of Fuzzy Controllers.

	[KeSc93]	[MaFr96]	[GuPe96]	[BaHu98]	[VaVi99]
Complexity:	9-rule @2-input @1-output	9-rule @2-input @2-output	13-rule @3-input @1-output	9-rule @2-input @1-output	16-rule @2-input @1-output
Technology:	CMOS-3μ	CMOS-0.7μ	CMOS-2.4μ	CMOS-2.4μ	CMOS-1μ
Power Supply:	Vdd=10V	Vdd=5V	Vdd=10V	Vdd=5V	Vdd=5V
Power Consumption:	no data	44mW	550mW	20mW	8.6mW
Input/Output Delay:	no data	570ns	160ns	500ns	471ns
Precision:	no data	no data	no data	no data	6.5%(3σ)
Interface:	currents @currents	voltages @voltages	voltages @voltages	voltages @voltages	voltages @currents
Area:	13.75mm^2	1.9mm^2	16.2mm^2	1mm^2	1.6mm^2
Programmab. MF Knees:	no data	on-chip	no data	on-chip(6b)	off-chip
MF Slopes:	no data	fixed	no data	on-chip(2b)	fixed
Consequents:	no data	on-chip	no data	on-chip(4b)	on-chip(4b)

4.2 A 27-Rule General-Purpose Programmable and Reconfigurable Analog Fuzzy Logic Controller

So far, we have demonstrated the functionality and modularity of the discrete-programming fuzzy blocks for a low-complexity non-reconfigurable Zero-Order controller. One step ahead, a more general architecture that overcomes, to some extent, the mentioned lack of 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.

First-Order Takagi-Sugeno algorithm yields better function approximations: the output surfaces are continuous and smooth while keeping better accuracy, as required for some applications. Moreover, given a function to approximate, the number of rules and the total number of parameters are further reduced compared to the needs of a Zero-Order controller. To our knowledge, no analog implementation of First-Order algorithms has been reported in the literature. This represents a good reason to investigate its feasibility. It will be shown later that with little manipulations of signals, the circuits already used in the previous controller can also be employed to configure a First-Order TS's controller.

Much effort has been put in achieving full programmability while minimizing the digital storing capacity needed. This represents in most cases a significant amount of silicon area devoted only for programming rather than for signal processing. In our case, an acceptable flexibility has been achieved with a relatively small size memory, if compared with others approaches. This is the result of the architecture that has been employed, which allows an optimal interface between the functional blocks and exploits the operator-sharing philosophy.

However, the long-channel technology that has been used constrains the capacity of the controller for a given tolerable chip area and power consumption. Nonetheless, the physical design was oriented to be easily scalable into a most modern CMOS technology that permits a larger integration density.

4.2.1 Architecture of the General-Purpose Fuzzy Logic Controller

Figure 4.16 illustrates the block diagram of a general mixed-signal architecture corresponding to a controller with M rules, N inputs, Q Zero-Order outputs and one First-Order output. It comprises mainly F independent Complementary Fuzzy Membership Function circuits (CFMF) arranged in two banks (for layout optimization), 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 contemplated for the controller (see Figure 2.9 in Chapter 2). The number of inputs of the T-Norms can be programmed, as will be explained later.

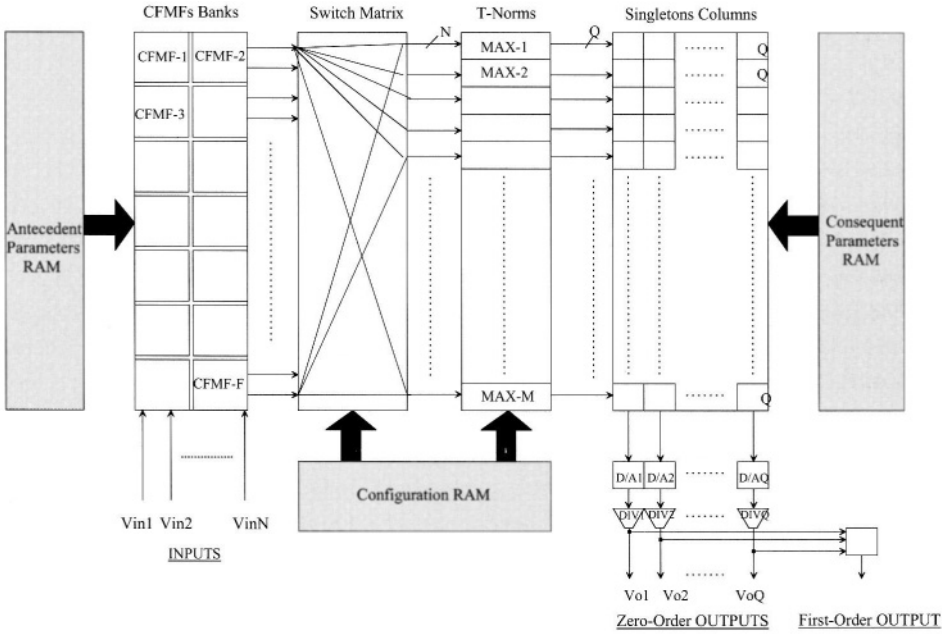


Figure 4.16. Block diagram of the general-purpose programmable and reconfigurable mixed-signal Fuzzy Controller: F shared CFMFs, N inputs Vin, M rules, Q Zero-Order outputs and one First-Order Output.

Considering that normally more than one membership function per input Vin is needed, it turns out that F must be greater than N. The inputs of all CFMFs circuits are left available off-chip. In this way, for a given number of

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

inputs Vin desired in the controller, the number of membership functions per input Vin can be externally defined by wiring. In short, 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. The number of singleton columns matches the number of Zero-Order outputs (Q). Each singletons column comprises M independent singletons in accordance with the number of rules M. Like in the former controller, 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 represent the firing degree of the same rule. The complementing operation is not shown in Figure 4.16. The Q identical firing degrees of each rule are distributed between the Q singletons common to the same rule but belonging to the Q different columns (or belonging to the Q different Zero-Order outputs). 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.

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 (see Figure 4.8 a)). This is not represented in Figure 4.16. It is intended to avoid the use of multipliers and it will be explained later.

Similar to the 9-rule controller the external signal interface in this architecture remains voltage-input voltage-output. The internal analog processing is carried out in current mode, with the exemption of the distribution of the membership functions' outputs among the several rules, which is now accomplished in voltage mode. The reasons of this innovation will be explained later in the section dedicated to the MAXIMUM operators.

The parameters of the membership functions (antecedent parameters), the values of the singletons (consequent parameters) and the configuration of the controller (the rule base and the number of inputs of the T-Norms) are independently programmable by means of three independent distributed RAMs. They are represented by shaded blocks in Figure 4.16. The digital memories are built by D flip-flops arranged in different shift registers that make easy the distribution of the stored bits among the different programmable blocks. Moreover, the use of shift registers prevents to dedicate a large number of input/output pads that would be necessary if classical matrix-arranged RAMs were used.

Aside the technological constraints that limit the capability of the implemented demonstrator, our controller holds almost all the characteristics

needed for a flexible mixed-signal architecture. 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.

Considering that we are dealing with a general-purpose controller, the optimization of power consumption and silicon area led the design. Since some of the building blocks and programming circuits have been already used for the former 9-rule controller we will mainly address the novelties that have been introduced in this one.

4.2.1.1 Membership Function Circuit.

In this case, we have chosen the CFMF Type-I detailed in Chapter 3. Although it occupies a larger silicon area, owing to its symmetry the systematic errors are smaller than in the CFMF Type-II. Similar considerations than in sections 4.1.2 and 4.1.2.1 have been taken into account for sizing, biasing and programming the circuit.

Nevertheless, in contrast to the former controller the output current is not replicated with mirrors as many times as the same circuit is being shared by different rules. With reference to Figure 4.2, the gate voltage of transistor Mp2 is now the signal representing the output of the CFMF. This signal is routed using a single wire towards the different T-Norms operators (the different rules) where the corresponding CFMF takes part. In this way, the routing space requirements are minimized. Moreover, with the avoidance of current mirrors for distributing the output of the CFMFs, the accuracy of the controller is improved because mirroring errors are eliminated.

Figure 4.17 shows some measured curves corresponding to this membership function. Notice that the attainable slope range remains large. The transient behavior was also characterized by simulations and in all cases the settling time (90% of the steady state value) remains in the order of 150ns.

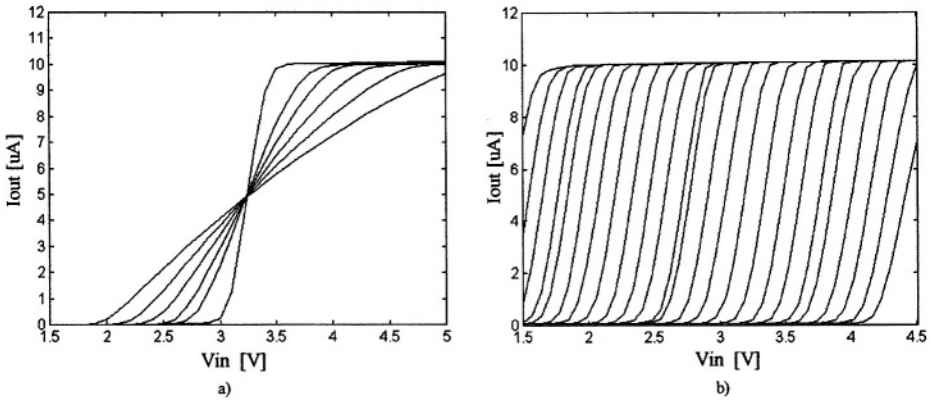


Figure 4.17. Measured CFMF Type-I for Io=10μA: a) Slopes. b) Right edge positions.

In the fabricated controller, there are two CFMF banks, each one comprising eight membership function circuits. Each bank is independently programmable by two 144-bit shift registers wherein digital values of the CFMF slopes and crossover points, codified through 18 bits, are serially loaded and stored. Thus, for programming purpose the CFMFs block demands only two external pins for data loading.

4.2.1.2 MAXIMUM Operator Circuit.

The mixed-mode MAXIMUM circuit explained in section 3.2.2 of Chapter 3 has been chosen for this implementation. In Figure 4.18 input currents I1…IN are supposed to be the outputs of the CFMFs taking part in a rule. In this circuit, one can select the inputs to be processed by the MAXIMUM operator 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 (Vin1... VinN in Figure 4.16).

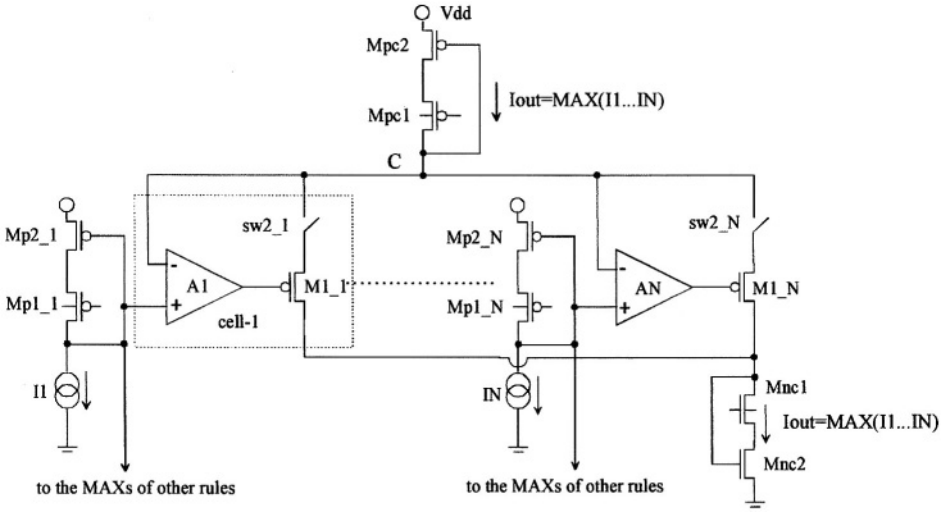


Figure 4.18. Mixed-mode N-input MAXIMUM circuit comprising N identical cells. The inputs to be processed (up to N) are set by switches sw2_1 to sw2_N.

The most important feature of this circuit resides in the fact that it can handle voltage inputs while delivering a current output. In this way, the interface with the membership functions is simplified. Each CFMF in the banks of the controller (see Figure 4.16) conveys its output current to its own PMOS cascoded-diode (Mp1_i, Mp2_i). Then, by simply wiring, the voltage drop at the latter mentioned diode can be distributed among the several rules where the same CFMF takes part. In addition, this MAXIMUM circuit also fulfils the interface requirements with the subsequent defuzzification stage.

In the latter, the aggregation and weighting of the singletons are performed in current mode, as already explained in section 3.3.1.

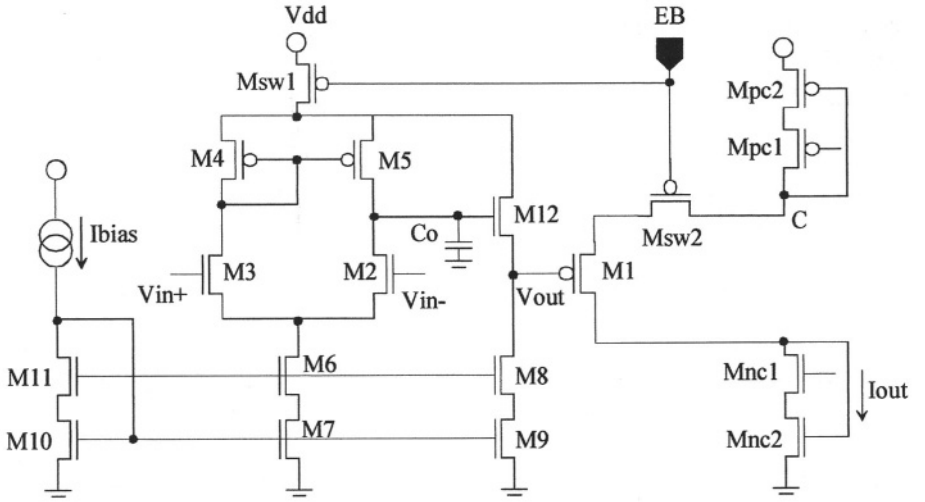


Figure 4.19. Detailed schematic of each cell of the N-input MAXIMUM operator.

Figure 4.19 depicts the detailed schematic of an input cell of the N-input MAXIMUM circuit. It also includes the PMOS and NMOS cascoded diodes, common to all cells in the MAXIMUM circuit (Mpc1, Mpc2 and Mnc1, Mnc2 in Figure 4.18). For the correct operation of this circuit the following issues must be considered. Firstly, the cascoded-diodes (Mp1_i, Mp2_i) (one diode per membership function) and (Mpc1, Mpc2) (one diode per MAX) must be correctly matched. Therefore, in the layout, they must be placed one close to each other. Secondly, the offset of the amplifiers A1…AN must be minimized so as to keep the propagation error as small as possible. Finally, a relatively high gain for the amplifiers A1…AN warrants a good discrimination capability for this circuit.

Figure 4.20 a) shows a DC nested-sweeps simulation for a two-input MAXIMUM circuit. Figure 4.20 b) illustrates its transient behavior. For the latter simulation, the common-node C was loaded with a capacitor that represents the load of a 5-input MAXIMUM circuit. The settling time remains in the order of 150ns. Table 4.6 summarizes the performance of the MAXIMUM operator.

With reference to Figure 4.18, Figure 4.19 shows the schematic of an amplifier A (M2 to M12), the PMOS source-follower transistor (M1, or M1_i in Figure 4.18), the switch transistor Msw2 (sw2_i in Figure 4.18) and an extra switch transistor per cell Msw1 (not shown in Figure 4.18). Since the number of active cells in the MAXIMUM operator must be adapted to

match the number of desired inputs Vin at the Fuzzy Controller, each cell is connected or disconnected to the common-node C through the switch transistor Msw2. In order to save current consumption, when a particular cell does not take part to the MAXIMUM it is also disconnected from Vdd by means of the switch Msw1. Both switches are driven by the binary signal EB stored in memory.

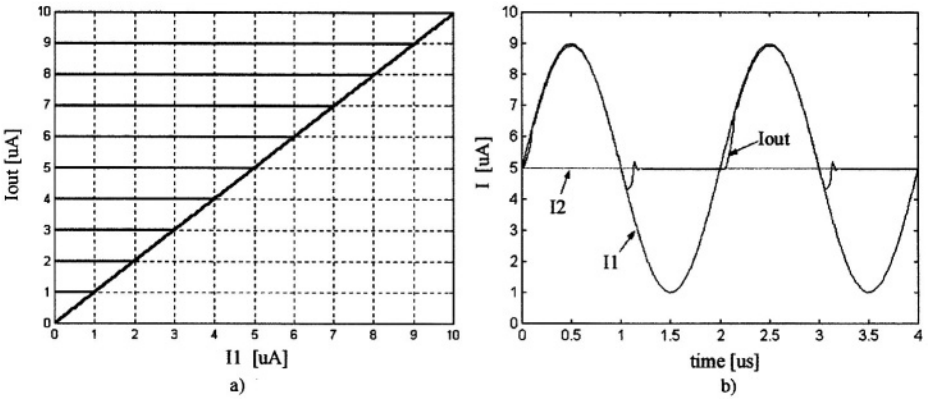


Figure 4.20. SPICE simulations of the MAXIMUM circuit: a) DC nested-sweeps: I1 ranges from 0 to 10μA while I2 from 1μA to 10μA by 1μA steps. b) Transient behavior: I1 is a 500KHz sine wave ranging from 1μA to 9μA while I2 is held to 5μA.

Considering that the non-inverting inputs of the amplifiers A are connected to the gate of a PMOS cascoded diode tied to Vdd (Mp2_1...Mp2_N in Figure 4.18), these input voltages result relatively high. Thus, the use NMOS input differential pairs for A1...AN is preferred, as shown in Figure 4.19. On the other hand, since the common-node C in Figure 4.18 is connected to the inverting inputs of the amplifiers, the DC output level of A1...AN must be shifted down. This is accomplished by the additional NMOS level shifter M12 in Figure 4.19. This strategy prevents M2, in Figure 4.19, to get out the saturation region [CaRa00]. Finally, a compensation capacitor at the drain of M2 (Co≈100fF) guarantees a first-order behavior of the cell without ringing.

Table 4.6. Sizing (in [μ/μ]) and performances of the mixed-mode MAXIMUM circuit cell.

MAXIMUM input-cell								
M1	M2,3	M4,5	M6-11	M12	Mpc1 Mpc2	Mnc1 Mnc2	Msw1	Msw2
16/3.2	15/4.8	16/4.8	18/3.2	16/3.2	24/3.2	18/3.2	7.2/2.4	4.8/2.4

Discrimin. Error	Propag. Error	Ibias	Input/Output Range	Delay	Current Consump.	Active Area
~1.7 %	~2 %	20μA	0≤ Ii, Imax≤10μA	~150ns	50μA	1043μ^2

4.2.1.3 Switch Matrix and Configuration of the Controller.

Let us assume a controller with a total number of F membership function circuits a maximum of N inputs Vin and M rules. In order to distribute adequately the outputs of the shared membership functions between the different rules several approaches could be considered. Figure 4.21 shows three possible realizations of the switch matrix that are explained in the following.

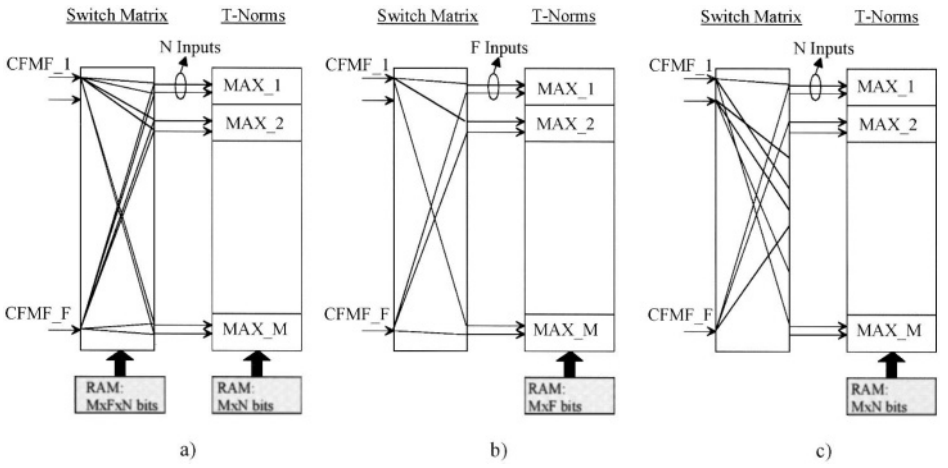


Figure 4.21. Different alternatives for the Switch Matrix implementation in a controller with N inputs Vin (not shown in the figure), F shared membership functions and M rules.

In Figure 4.21 a) T-Norms have N inputs. An analog switch matrix with F inputs and M×N outputs gives rise to a maximum of flexibility. It allows connecting any membership function's output to any T-Norm's input of any rule. The total number of bits required to configure the matrix rises to M×F×N plus M×N additional bits to configure the T-Norms in accordance to the number of inputs Vin (not shown in the figure) intended for the controller. For instance, in our case we have F=16, N=5 and M=27. Thus, the storage capacity needed for the configuration would be equal to 2295 bits while requiring 2160 analog pass gates in the switch matrix (an analog pass gate comprises a pair of NMOS and PMOS transistors connected in parallel). However, this situation is quite unrealistic because in practical applications no membership function takes part in all rules defined in the controller. Moreover, if this option was implemented, each membership function circuit would be excessively loaded with a total capacitance corresponding to Ct=M×N×Co. In the latter expression, Co represents the mean capacitance value of the routing wires connecting each membership function circuit to each corresponding pass gate of the cross-matrix. For example, let us

consider that Co=50fF. In our case (M=27, N=5), Ct would rise to 6.75pF. Therefore, the membership functions output should be buffered with the subsequent increment of the power consumption and silicon area.

A second option is shown in Figure 4.21 b). It 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 like the one shown in Figure 4.19. In this case, the total number of bits needed for configuration is equal to M×F (432 in our case). No pass gate would be needed in the matrix (wired). The participation or not of a CFMF in a rule is decided by programming the corresponding bit EB (see Figure 4.19) at the corresponding F-input T-Norm operator. The load capacitance for each CFMF output is now Ct=M×Co=1.35pF. Hence, the size and current consumption demanded by the CFMF's buffers, if used, are diminished with respect to the former option. Nevertheless, this solution remains sub-optimal. On one hand, the F-input T-Norm becomes a large circuit, with an impoverished transient behavior, while demanding increased surface. On the other hand, as stated above, the participation of a given CFMF in all rules of the controller never happens in practical applications. In conclusion, a T-Norm handling F inputs results over sized and inefficient.

Figure 4.21 c) illustrates the third alternative. This is a more practical variant of the two previously explained ones. It consists in adopting a wired switch matrix while using programmable N-input T-Norm operators, with N<F. In this case the distribution of the CFMF's outputs between the different T-Norms of the rules is performed randomly but assuring a uniform and equilibrated participation of each CFMF in the whole set of rules. On the other hand, the random distribution of the CFMF's outputs must warrant the possibility of configuring some complete set of rules covering the whole input Universe of Discourse (two and three-input controllers supporting full grid partitions, for instance). This is the actual implemented option in our controller. However, the maximum number of inputs Vin of the controller is now limited by the maximum number of cells in the T-Norm (N). In our case, after fixing N=5 and M=27, the storing capacity required for configuration becomes equal to 135 bits. On the other hand, the random wired matrix ensures grid 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,

and tree and scatter partitions for:

– 4-input, 3 Zero-Order output controllers,
– 5-input, 3 Zero-Order output controllers.

4.2.1.4 Consequent Singleton Columns and Defuzzifier.

Like in the former controller, we implemented the defuzzifier following the "common-weighting" strategy. Each singleton comprises (5+1) mirror transistors. Let us recall that an extra mirror must be provided at each singleton in order to compute the denominator current of each divider. The circuits used for the singletons, the weighting D/As, the dividers and the output buffers are similar to the ones used in the previous controller. Some of them have been resized. Particularly, the dividers and the output buffers have been modified in order to attain a 2V output voltage swing.

In the divider shown in Figure 4.8 a), the size of transistors belonging to the right column has been halved with respect to the corresponding transistors in the central and left columns. In this way, according to the general expression developed in Chapter 3 (k=1/2), we can double the output voltage swing while holding the same bias voltages values Vb1 and Vbo.

Finally, an enhanced version of the OTA buffer following the divider was sized to allow an input common-mode range and output swing of 2V. The differences with respect to the scheme in Figure 4.8 b) rest on the fact that transistors Mb4 to Mb9 and Mb1 have been cascoded in order to improve the systematic offset and the finite-gain error. Concluding, Table 4.7 summarizes the performance of the circuits of the defuzzifier stage.

Table 4.7. Defuzzifier's circuits: sizing (in [μ/μ]), biasing and performance.

Singletons (#3x27)						
Mc1,2	Msw	Resolution		Current Consump.		Active Area
12/4	5/5	5 bits		70μA (max)		847μ^2
Weighting D/A (#3)						
Mad1,2 bit 0,1	Mad1,2 bit 2	Mad1,2 bit 3	Mad1,2 bit 4	Current Consump.		Active Area
5/5	10/5	20/5	28/5	40μA (max)		5240μ^2
Divider (#3)						
M1-2	M4-5	M7,8,10, 11	M3	M6		M9,12
24/6	48/6	74/4	12/6	24/6		37/4
Vbo	Vb1	Output Swing	Input Range	Delay	Current Consump.	Active Area
1.7V	2.7V	2V	0 ≤ IN,ID ≤ 40μA	~150ns	75μA (max)	1452μ^2
OTA Output Buffer (#4)						
M1,2	M3,4,5,6	M7,8,15,16	M9,10,11,12,13,14			M17,18
300/3.2	225/2.4	450/2.4	630/2.4			63/2.4
Ipol	Adc	SR(CL=10pF)	fT (CL=10pF)	Phase Margin	Current Consump.	Active Area
30μA	75dB	40 V/μs	23MHz	67°	0.95mA	17774μ^2

4.2.1.5 Implementation of the First-Order Output.

Let us remind from Chapter 2 that in First-Order Takagi-Sugeno's inference systems the consequents of the rules are linear combinations of the inputs of the controller. In our implementation, it is possible to configure a 2-input, 1-output, 25-rule programmable controller of this type. Since there are three independent singletons columns we can use one of them to program the constant terms of the linear expressions of the consequents. The two others columns can be used for programming the coefficients multiplying the inputs variables Vin in the linear expression of the consequents. This situation is clarified in Figure 4.22 that illustrates the three singletons columns with their corresponding weighting D/A and divider. When the switches sw1 and sw2 are set to Vb1, the controller yields three independent Zero-Order outputs, namely Vout0, Vout1 and Vout2. In the opposite case, when the switches are set to the controllers' inputs Vin1 and Vin2, the First-Order output is retrieved after the adder. This sums the outputs of the dividers to deliver the signal Vout in Figure 4.22.

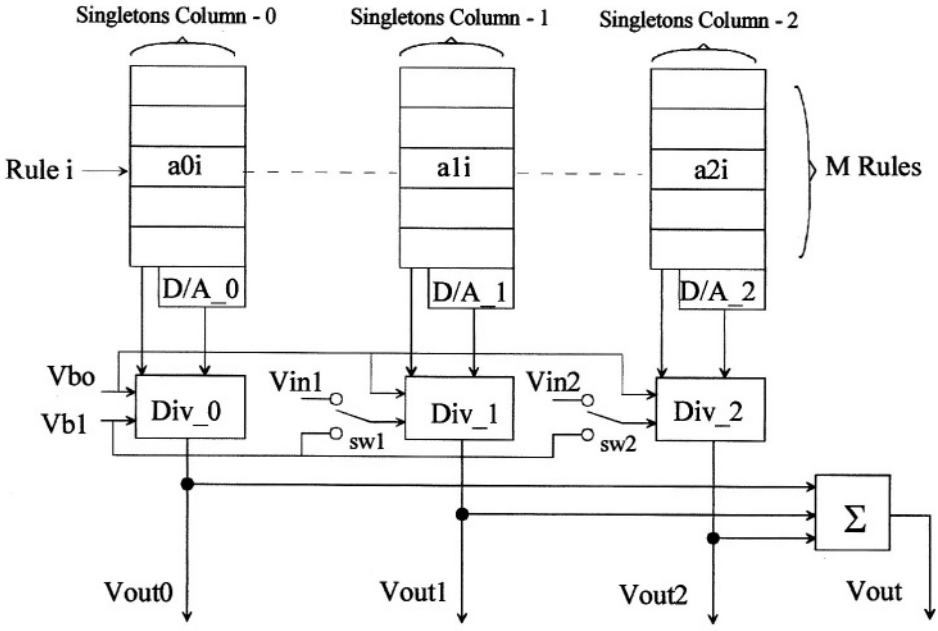


Figure 4.22. Zero and First-Order output configuration of the controller.

Let us assume a First-Order controller with two inputs, Vin1 and Vin2, and M rules. The linear expression for the consequent of the rule i can be written as:

$$Ci = a0i + a1i\,Vin1 + a2i\,Vin2, \qquad \text{for } i = 1...M. \tag{4.9}$$

It turns out that the defuzzified output value of the M-rule First-Order controller is given by:

$$\text{Vout} = \frac{\sum_{1}^{M} \text{Ci Ii}}{\sum_{1}^{M} \text{Ii}} = \left(\frac{\sum_{1}^{M} \text{a0i Ii}}{\sum_{1}^{M} \text{Ii}} \right) + \left(\frac{\sum_{1}^{M} \text{a1i Ii}}{\sum_{1}^{M} \text{Ii}} \right) \times \text{Vin1} + \left(\frac{\sum_{1}^{M} \text{a2i Ii}}{\sum_{1}^{M} \text{Ii}} \right) \times \text{Vin2}, \tag{4.10}$$

where Ii is the firing degree of rule i. It can be noticed from (4.10) that this operation implies the use of analog multipliers to multiply the controllers' inputs by their corresponding averaged coefficients of the consequent linear expressions. On the other hand, a 3-input voltage-mode adder is also needed to get the final defuzzified value Vout for the First-Order controller configuration.

However, multipliers in (4.10) can be straightforwardly avoided by exploiting the entire transfer function of the divider itself (see section 3.3.1.1). In fact, if the inputs Vb1 of the dividers in the central and right columns in Figure 4.22 are modulated by inputs Vin1, Vin2 respectively, the multiplication operation is naturally performed (if input and output signals are referred to Vbo). Thus, the output of these two dividers becomes now equal to:

$$(\text{Voutj} - \text{Vbo}) = (\text{Vinj} - \text{Vbo}) \times \frac{\sum_{i=1}^{i=M} \text{aji Ii}}{\sum_{i=1}^{i=M} \text{Ii}}, \quad \text{for } j = 1, 2. \tag{4.11}$$

On the other hand, trying to perform in voltage mode the addition operation requested by (4.10) requires additional input buffers, if the typical inverting configuration with an operational amplifier is used. This is shown in Figure 4.23 a). Buffering the inputs of the adder becomes necessary in order to prevent loading the dividers' output. Indeed, even if the resistors R were implemented by using the typical four-transistors transresistive array, the adder's inputs would demand a DC current component. If no buffers were provided these DC currents should be supplied by the dividers. Consequently, the translinear relationships in (4.11) would be no longer accomplished.

4.2.1.6 Novel MOST-Only Multiple-Input Voltage Adder.

To overcome the problem explained above and based on the circuit of our divider [DuVe98], a new high input impedance multiple-input adder was designed. It is a much simpler circuit than the above mentioned classical approach and it is illustrated in Figure 4.23 b). In the latter, Q independent input voltages V1...VQ are applied to the gates of the Q triode-operating transistors Mi1...MiQ at the left column. The latter transistors are equally sized to the triode-transistors M2 and M3 in the other two columns. Transistors M4, M7 and M10 are also Q times up-scaled with respect to their homologues at the same layer. Upon ideal matching conditions, calling β the current gain of any triode transistor and Vds its equalized drain-to-source voltage drop, we can write:

$$
\begin{aligned}
Id_{M2} &= \beta\, Vds\,(Vbo - VTn - \frac{n}{2} Vds) && \text{(a)},\\
ID &= -Q\, Id_{M2} + \sum_{i=1}^{Q} \beta\, Vds\,(Vi - VTn - \frac{n}{2} Vds) && \text{(b)},\\
IN &= -Id_{M2} + \beta\, Vds\,(Vout - VTn - \frac{n}{2} Vds) && \text{(c)},
\end{aligned}
\qquad (4.12)
$$

where Id_{M2} is the drain current of transistor M2 and n accounts for the Body effect. After replacing expression (4.12) (a) in (4.12) (b) and (c) we have:

$$
(Vout - Vbo) = \frac{IN}{ID} \sum_{i=1}^{Q} (Vi - Vbo). \qquad (4.13)
$$

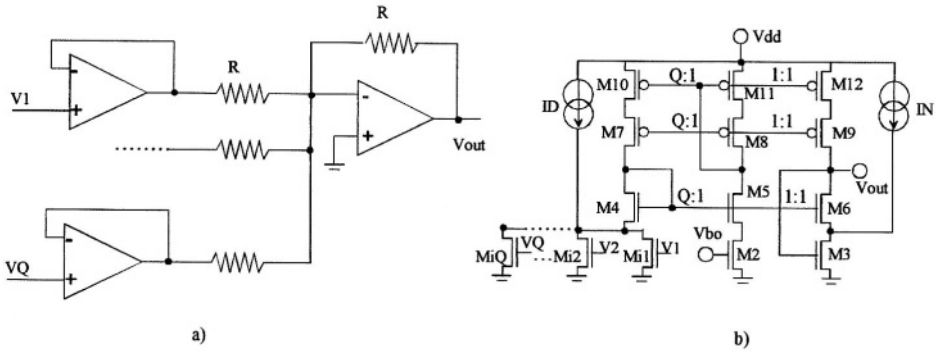


Figure 4.23. a) Classical voltage-mode multiple-input adder. Input buffers are needed to avoid loading the dividers' outputs in Figure 4.22. b) Novel MOST-only multiple-input voltage adder holding high input impedance.

Therefore, if input and output voltages are referred to Vbo, a Q-input voltage adder is obtained. Additionally, the values of the currents IN, ID can be set according to the desired output swing for the adder.

A three-input version of this circuit has been integrated in the controller and measured. Figure 4.24 a) and b) show the measured DC characteristic and the relative errors for several input conditions, respectively. Currents IN and ID have been set to 4μA and 12μA, respectively, whereas Vbo has been set to 1.5V. Therefore, a 2V output voltage swing has been reached for Vdd=5V. Notice from Figure 4.24 a) the relatively high swing allowed to the input voltages (~2V). Figure 4.24 b) attests that the relative errors can be kept below ±2%. Table 4.8 summarizes the main features of this circuit.

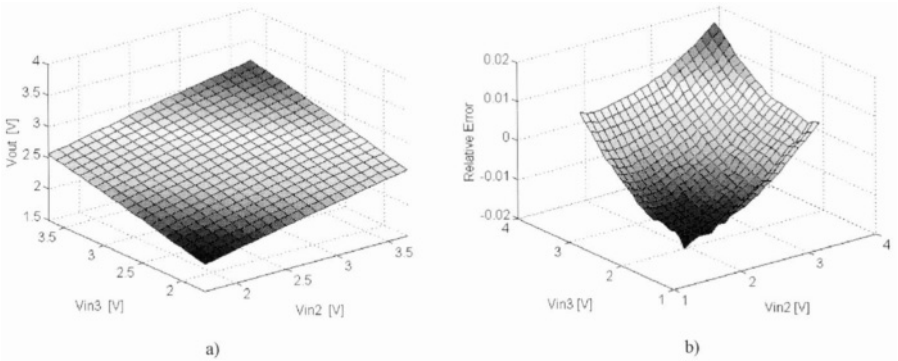


Figure 4.24. a) Measured DC characteristics of a 3-input adder for: IN=4μA, ID=12μA, Vbo=1.5V, Vin1=2.3V whereas 1.7V≤Vin2, Vin3≤3.7V b) Relative errors between measured and calculated expressions.

Table 4.8. Main features of the 3-input voltage-mode adder. Transistors sizes are in [μ/μ].

3-input adder							
Mi1-Mi3, M2, M3		M4	M7	M10	M5,6	M8,9	M11,12
10/15		30/10	120/4	120/4	10/10	40/4	40/4
IN	ID	Vbo	Relative Error	Input/Output Range		Current Consump.	Active Area
4μA	12μA	1.5V	±2%	1.7V≤ V1…V3 ≤3.7V 1.7V≤ Vout ≤3.7V		52μA (max)	2850μ²

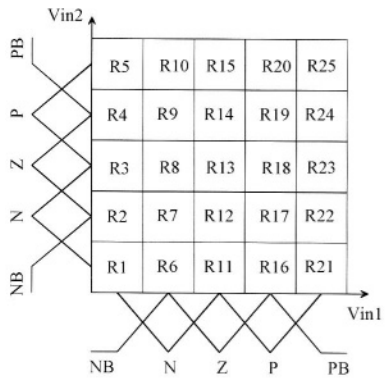
4.2.2 Test Results of the General-Purpose 27-Rule Fuzzy Controller

The measurement set-up for testing this prototype is the same as used for the previous controller (see Figure 4.11). This controller has been first configured as a Zero-Order type. The DC and statistical deviation characterizations have been performed. Then, a First-Order Takagi-Sugeno's

controller has been configured, measured and compared against a Zero-Order type that attempts to perform the same input-output relationship. Closing up this section the measured transient response of the controller is presented.

4.2.2.1 Zero-Order Controller: DC Test.

A 25-rule 2-input, 5-label per input 1-output, Zero-Order TS's controller has been first configured in the chip. Two different sets of singletons corresponding to the two different test functions mentioned above have been loaded. Figure 4.25 a) shows the fuzzy grid partition of the input space together with the rules map, which are kept identical for both testing functions. Figure 4.25 b) shows the electrical settings of the controller as well as the digital codes that define the membership shapes of the five fuzzifiers {NB, N, Z, P, PB} of each input. A uniform fuzzy partition for both input variables has been achieved by placing the fuzzy labels equally spaced along the variables input range. All membership functions have identical slopes.



a)

Electrical Settings:

Vdd	Io	Vbo	Vb1	Singleton Resolution
5V	10 µA	1.7 V	2.7V	LSB=(2/32) (Vb1-Vbo)=62.5mV

Input Range:	Output Range
1.5V < Vin1, Vin2 < 4.5V	1.7V < Vout < 3.7V

Inputs Fuzzy Labels: Parameters Digital Codes

	pr	sr	pl	sl
NB	22	15	30	15
N	16	15	22	15
Z	11	15	17	15
P	6	15	12	15
PB	0	15	6	15

- pr, pl: right and left crossover
- sr, sl : right and left slopes

b)

Figure 4.25. 25-rule Zero-Order Takagi-Sugeno configuration: a) Rules map and input fuzzy partition. b) Electrical settings and digital codes of the CFMFs parameters.

For each testing function, Figure 4.26 shows the digital codes of the rules consequent singletons, the measured output surfaces and the relative error distribution after a comparison with the target output surfaces. As in the previous controller, the nominal values of some parameters have been modified in order to approach the target function as much as possible.

The first testing surface a), corresponds to a classical control law used in Fuzzy P+D Controllers. This surface presents relatively gradual transitions between fuzzy clusters (rules); it was defined in order to test the controller's

output over its maximum swing (i.e. 2V@Vdd=5V). The measured RMSE for this function is 80mV (4% of full scale). Most relative errors between target and measured surfaces are found within a ±6% tolerance with a reduced number of outliers close to ±7.5%. The spread between chips was estimated by measuring eight prototypes programmed with the same surface. The maximum standard deviation σ_{max} raises to 180mV (9%) whereas the mean reaches to 64mV (3.2%).

The results of the test of the second programmed transfer function are shown in Figure 4.26 b). In this case, considerable level differences between contiguous regions have been programmed in order to provoke abrupt slopes at the rules boundaries, wherein the larger deviations are normally found. The RMSE in this case raises to 94mV (4.7%). Most relative errors are kept within ±8% but they can rise up to 15%. The maximum standard deviation is 240mV (12%) while the mean reaches to 75mV (3.75%).

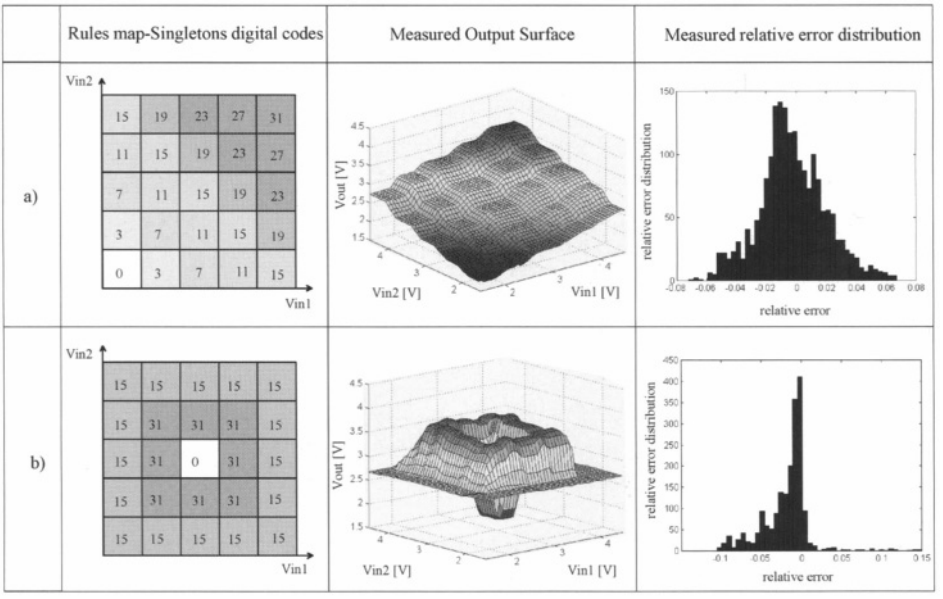


Figure 4.26. DC performance obtained for each testing function: rules consequent settings, measured output surface and distribution of the relative errors between target and measured surfaces.

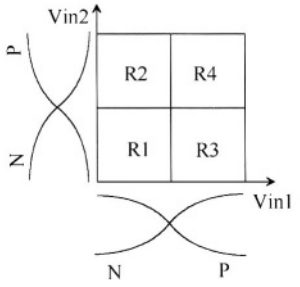
4.2.2.2 First-Order Controller: DC Test.

Concerning the characterization of the First-Order Takagi-Sugeno's controller, a 2-input, 1-output, 4-rule controller has been configured and its DC behavior has been measured. Each input variable Vin1, Vin2 has been fuzzified by means of two complementary membership functions, {P, N}. These have been defined over a 1V input range (2.2V<Vin1,Vin2< 3.2V).

Figure 4.27 shows the input partition, the electrical settings and the loaded digital codes of the antecedent and consequent parameters. This particular set-up performs an interpolation between two different planes defined by the linear expressions of consequents of the rules. It can be noticed from Figure 4.27 b) (in the table of Coefficients of the Consequent Linear Expressions) that each plane has been defined twice in two different rules. Accordingly, the 4-rule set can be linguistically expressed as follows:

- R1(R4): "*if* Vin1 is N(P) and Vin2 is N(P) *then* C1(C4) = a01 + a11 × Vin1 + a21×Vin2";
- R2(R3): "*if* Vin1 is N(P) and Vin2 is P(N) *then* C2(C3) = a02 + a12 × Vin1 + a22×Vin2";

where Vin1 and Vin2 are the inputs of the controller.



a)

Electrical Settings:

Vdd	Io	Vbo	Vb1	Singleton Resolution
5V	10 μA	1.7 V	2.7V	LSB=(2/32) (Vb1-Vbo)=62.5mV

Input Range:	Output Range
2.2V < Vin1, Vin2 < 3.2V	1.7V < Vout < 3.7V

Inputs Fuzzy Labels: Parameters Digital Codes

	pr	sr	pl	sl
P	17	7	31	15
N	0	15	17	7

- pr, pl: right and left crossover
- sr, sl : right and left slopes

Coefficients of the Consequents Linear Expressions

	a0	a1	a2
R1	15	20	20
R2	15	3	3
R3	15	3	3
R4	15	20	20

b)

Figure 4.27. 2-input, 1-output, 4-rule First-Order Takagi-Sugeno's controller configuration: a) Input partitions and rules map. b) Electrical settings, membership functions parameters and coefficients of the linear expression of consequents of the rules.

Figure 4.28 a) depicts the measured membership functions (CFMFs) extended along the 1V input range of each variable Vin1, Vin2. Notice the versatile characteristic of the implemented membership function circuits built by triode transconductors, which allow setting up small slopes. This difficulty is affordable if saturated transistors transconductors were used, as in most reported approaches. Figure 4.28 b) shows the measured DC transfer function for this configuration. In the latter, the continuous and smooth transitions between clusters can be appreciated. This is due to the direct

correlation between the inputs and the output of the controller through the linear expressions defined at each rule's consequent.

By using the ANFIS learning method, the measured surface in Figure 4.28 b) has been fitted by means of three controllers configurations and the results are illustrated in Figure 4.29. In the case of Figure 4.29 a) the same configuration and type of algorithm like the one loaded in the chip has been employed. With a RMSE of 0.8%, it corresponds to the best fitting among the three configurations used. In Figure 4.29 b) a 4-rule, 2-membership function per input, Zero-Order controller has been configured. Its RMSE results in 2%. By increasing to nine the number of rules in the former configuration while using now three membership functions per input a better RMSE figure (1.4%) could be obtained. The resulting surface is shown in Figure 4.29 c).

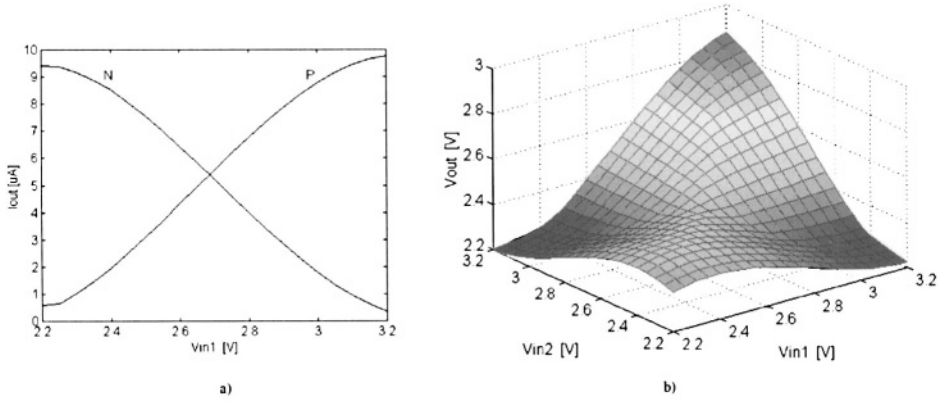


Figure 4.28. a) Measured Complementary Fuzzy Membership Functions for the labels P and N. b) Measured output surface of the First-Order Takagi-Sugeno's controller.

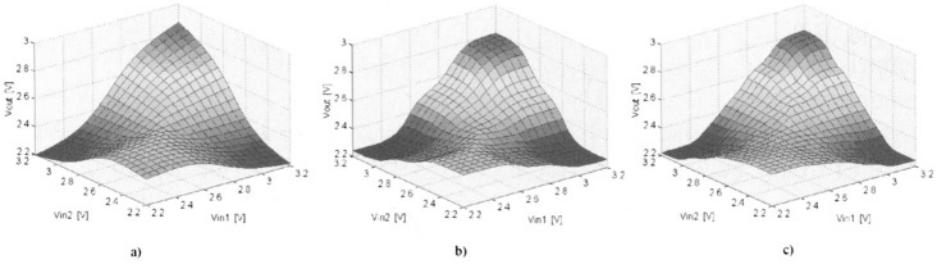


Figure 4.29. ANFIS fitting of the measured surface in Figure 4.28 b) with different controllers configurations: a) 4-rule, 2-FMF per input, First-Order controller. b) 4-rule, 2-FMF per input, Zero-Order controller. c) 9-rule, 3-FMF per input, Zero-Order controller.

Notice from the latter experience that even increasing the number of rules in the Zero-Order configuration, the First-Order type remains always superior in terms of the attained RMSEs. Moreover, geometrically speaking, the First-Order approximation in Figure 4.29 a) fits the measured surface in Figure 4.28 b) quite better than the others approximations in Figure 4.29 b) and c). Furthermore, let us assume that in terms of the obtained RMSE the Zero-Order controller configured in Figure 4.29 c) remains competitive with respect to the First-Order type in Figure 4.29 a). However, the Zero-Order type demands nine rules and a total number of 33 parameters (antecedents + consequents) whereas the First-Order type demands only 4 rules and 28 parameters.

A real-time application of First-Order controllers in the domain of Power Electronics is reported by [GoAl01]. In the latter, a switching regulator is controlled by means of a 6-rule First-Order controller following a piecewise sliding global control policy. The resulting Fuzzy Control scheme provides soft-switching interpolation among the different control laws and a much reduced complexity, if compared against other implementation that makes use of a 25-rule Zero-Order controller for a similar control task. [GoAl01] emphasize on the resulting low-count rule base that makes the design suitable for being implemented on an application-specific high-speed analog First-Order Fuzzy Controller, as the one presented above.

4.2.2.3 Zero-Order Controller: Transient Test.

Closing up the characterization of this prototype its transient behavior has been addressed by means of the large and small swing step responses. For this purpose the chip has been configured with the set-up corresponding to the surface shown in Figure 4.26 a). Figure 4.30 a) shows the measured small swing step response. In this case, one input is held constant whereas an 800mVpp step is applied to the other. The output of the controller delivers a 200mVpp step (10% of the full output range). For this case, the measured settling time (for 90% of the steady state value) reaches to 570ns that comprises 200ns of delay plus 370ns of rise time. For the large swing step response shown in Figure 4.30 b) the input step ranges from 1.5V to 4.5V (full input range) whereas a 1Vpp step is delivered at the output in 1.1μs (90% of the steady state value). The latter encompasses a delay of 300ns and a rise time of 800ns. Accordingly, the processing speed of this controller ranges from 0.9 to 1.75 MFLIPS.

Obviously, the large dimensions of this controller affect strongly its transient behavior. If we perform the same analysis than in the previous controller, the sum of the delay introduced by each operator (without considering routing capacitances) in the cascaded chain differs considerably from the actual measured delay. Undoubtedly, this difference is mainly due

to the increased routing stray capacitances. On one hand, the CFMFs are considerably loaded by the long wires needed to distribute their outputs between the different MAXIMUM and by the input capacitances of the MAXIMUM circuits themselves (each CFMF is involved in 10 to 15 rules, approximately). On the other hand, the mixed-mode MAXIMUM circuit used for this controller is slower than the one used in the previous one. Apart from the fact that the two MAXIMUM circuits are different, this one handles more inputs than the previous one. Thus, it holds larger dimensions with increased parasitic capacitances. Finally, the consequents+D/A block becomes also considerably loaded by the routing capacitances of the wires that sum the current of the singletons columns.

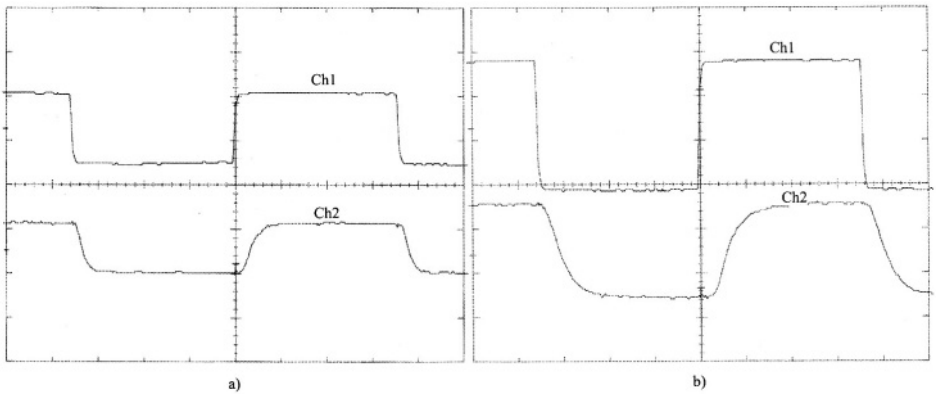


Figure 4.30. Transient behavior: a) Small swing step response to a 800mVpp input step. Ch1=500mV/div, Ch2=200mV/div, time base=1 s/div. b) Large swing step response to a 3Vpp input step. Ch1=1V/div, Ch2=500mV/div, time base=1 s/div.

4.2.3 Summary and Comparison with Other Approaches

Figure 4.31 shows the microphotograph of the chip fabricated in CMOS-2.4 MIETEC technology and encapsulated in a JLCC-84 package. Involving a total number of 21000 transistors, its core occupies 34 mm^2, 67% for the analog circuits and 33% for the digital. The total silicon area including pads raises to 39.5mm^2 (i.e. 1.46 mm^2/rule). For a 5V power supply, the total measured power consumption reaches to 63mW (i.e. 2.33mW/rule), 18mW (28.5%) accounts for the buffers whereas 45mW (71,5%) for the analog core.

The crossover points and the slopes of the CFMFs are 5-bit and 4-bit programmable, respectively. Since each T-Norm is 5-bit programmable the maximum number of inputs Vin of the controller is also limited to five. The consequent singletons are 5-bit programmable. The digital interface of the

chip comprises an 8-bit word: two bits are used for loading the parameters of the membership functions in both banks, one bit for the T-Norms configuration and the remaining five bits for the consequent singletons. Only 909 bits of memory are required for full programming and configuring of this controller.

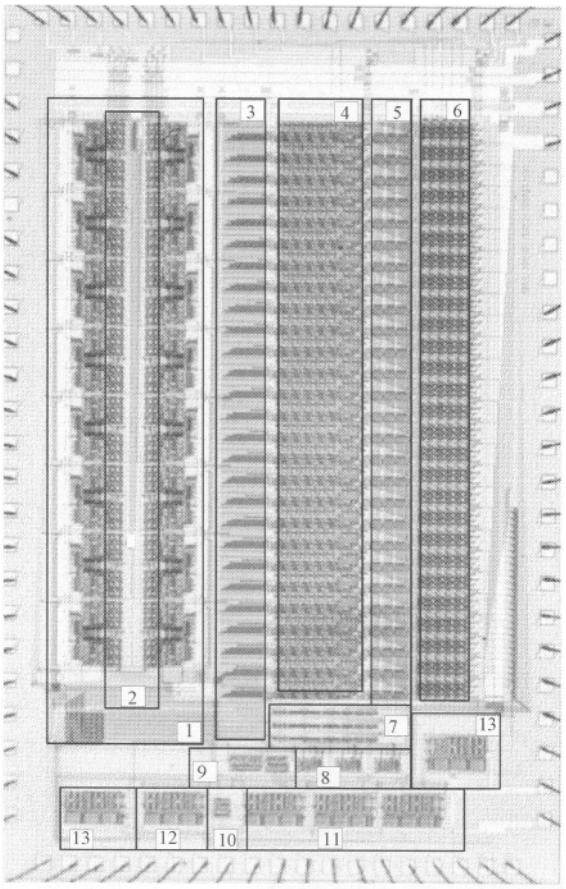


Figure 4.31. Chip microphotograph: 1) CFMFs banks (#2x8). 2) Antecedent Parameters RAM. 3) Wired Switch Matrix. 4) 5-Input T-Norms (#27) and distributed Configuration RAM. 5) Singletons Columns (#3). 6) Consequent Parameters RAM. 7) Weighting D/As (#3). 8) Dividers (#3). 9) Bias circuits. 10) 3-input Adder. 11),12) Output Buffers (#4). 13) Testing Buffers (#2).

Table 4.9 summarizes the main measured features of our 27-rule general-purpose programmable and reconfigurable controller. Table 4.10 shows the characteristics of two reported analog fuzzy processors that hold a complexity similar to ours. We will briefly summarize their features in the following.

In [FaMa94] 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.

Table 4.9. Main features of the 27-rule programmable and reconfigurable Fuzzy Controller.

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: (inputs@outputs)	voltages@voltages		
Input Range:	$1.5V \leq Vin \leq 4.5V$		
Output Range:	$1.7V \leq Vout \leq 3.7V$		
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: (90% of steady state value)	Small swing:	570ns	
	Large swing:	1.100ns	

[FrMa98] reports an analog continuous-time Fuzzy Controller. It has been implemented in a 0.7μ CMOS technology and it can be programmed to support 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 membership function circuits. With the exception of the slopes of the membership functions, which are fixed at the mask level, the antecedent and consequent parameters are programmable. For this purpose, a discrete set of voltage values are generated only once on-chip from a voltage reference that bias a resistive ladder attenuator. From the ladder, the parameters values are distributed to all programmable circuits in the chip (antecedents and

consequents) through a programmable analog cross-matrix. The chip requires a 16-Kbit digital memory to be fully configured.

Table 4.10. Main features of other programmable and reconfigurable Fuzzy Processors.

	[FaMa94]	[FrMa98]
Technology:	CMOS-0.8μ	CMOS-0.7μ
Processing Mode:	Sampled-Time	Continuous-Time
Complexity:	Mamdani: 32-rule@8-input @4-output (max)	Zero-Order TS: 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:	8-bit	2V
Output Range:	6-bit	2V
Area (total):	70mm^2	32mm^2
Programmability		
MF Position:	on-chip	on-chip
MF Slope:	on-chip	fixed
Consequents:	on-chip	on-chip
Total storage capacity needed:	no data	16-Kbit

4.3 Conclusions

The design and test of programmable analog Fuzzy Logic Controllers built from the basic analog processing blocks studied in the previous chapter has been addressed in this one. Issues concerning the architectures of the controllers, the blocks interfacing and the programming strategies have been specially focused in view of the efficient implementation of flexible Fuzzy Controllers. The attained flexibility and processing speeds render our demonstrators useful for a broaden domain of applications that may range from Control to Signal Processing.

A trade-off between accuracy and complexity has been accomplished by means of discrete sets of analog parameters that configure the controllers. The use of widespread digital memory circuits to store the digital representation of those parameters simplified extremely the on-chip programming strategy implemented. We have exploited a mixed-signal design framework that holds the advantages of the analog circuits for massive and fast computation together with the feasibility of digital circuits for storing and programming.

Sharing functional operators and avoiding the use of intermediate signal converters, have played an important role during the design step. Practiced in depth these general guidelines led to an improved modularity, which is

reflected in smaller silicon area, lower power consumption, reduced storage capacity and even shortened internal delays.

In the first implemented 9-rule, 2-input, 1-output Fuzzy Controller [DuVe00], only antecedent and consequent parameters can be programmed with a resolution ranging from four to five bits. The obtained performances in terms of power (8.4mW), speed (<5.26MFLIPS) and area ($4.5mm^2$) render this controller attractive to be used as an on-chip programmable subsystem embedded with a specific application. The use of fast controllers with small number of rules has been reported in several real-time applications [MoPi94] [FrMa98] [BaDi00], whose requirements are fulfilled by this prototype.

In the second general-purpose Fuzzy Controller the reconfiguration capability has been introduced. This controller can support up to 5 inputs, 3 outputs and 27 rules in Zero-Order mode. Antecedent and consequent parameters remain programmable with identical resolution than in the previous small prototype. Furthermore, as a merit of the flexible architecture and circuits that build the controller, a 2-input, 1-output, 25-rule First-Order Takagi-Sugeno's controller can be efficiently afforded. This only requires little manipulation of the input-output signals and the use of an extra voltage-mode adder, which has been specially designed for this purpose. It has been demonstrated that, with smaller number of rules and parameters, First-Order controllers can yield high-quality function approximations. In summary, this controller demands $39,5\ mm^2$ of silicon area in a CMOS-2.4μ technology and consumes 63mW for a 5V power supply. Its processing speed ranges from 0.9 and 1.75 MFLIPS. Only 909 bits of memory are needed for programming.

Unfortunately, the restrictions imposed by the technology we have used did not allow us to further increase the capacity and the accuracy of the latter presented fuzzy processor. Obviously, one can improve all error figures by increasing the resolution of the controllers parameters. This solution, however, impact strongly on the total silicon area, power consumption and digital memory demanded. Nevertheless, in view of an easily scalable architecture to be remapped onto a much modern technology, special attention has been paid to the layout design for placing, routing and interfacing the elemental blocks. Let us consider the general-purpose controller being now implemented in a 0.8μ CMOS technology, for instance. For the same silicon area and power consumption demanded by the actual implementation, rough estimations let us to foretell a more powerful controller supporting three times more of rules, inputs and outputs. Furthermore, short-channel CMOS technologies normally features better matching properties [KiSt96] than the long-channel ones. Thus, improved spreading figures should also be expected.

Chapter 5

TIME-DOMAIN SIGNAL ANALYSIS USING FUZZY LOGIC

An on-chip real-time oscilloscope and its application to self-adaptive channel equalization

5. INTRODUCTION

In the last years the application of Fuzzy Logic has been extended beyond classical Process Control. Signal Processing, Image Processing and Switching Power Control are being increasingly considered as potential fields where this technique brings interesting solutions. Some implementations in the latter mentioned domains are reported in [MaDa95] [BaDi00] [HuKe95] [LeLe93] [LiMe00] [PaKo00] [MoPi94] [OeGr96] [ChJo99] [KaLa98] [YuCh96] [ToHa98] [CrLi98]. Most of these applications claim for: low-power consumption, autonomy and fast time response for real-time processing. Therefore, the implementation of embedded Fuzzy Controllers becomes an attractive option, if not the unique in some cases.

For the sake of the demonstration of the usefulness of analog Fuzzy Controllers as embedded subsystems, we introduce in this chapter an investigation concerning the employment of Fuzzy Logic for the analysis of signals in real-time. The basic idea consists in exploring the signal over time and converting it into a geometrical figure, as in any oscilloscope. On the other hand, clusterizing the input Universe of Discourse [Time, Signal_Value] by means of fuzzy partitions, any attribute can be graphically identified in the resulting figure. Supported on the mentioned input partition, a fuzzy decision-making arrangement can be settled and after a comparison

with a reference pattern, the fuzzy rules set may infer meaningful assertions. According to the application, those assertions can be used for adaptation, detection, testing, etc. This technique is illustrated in Figure 5.1.

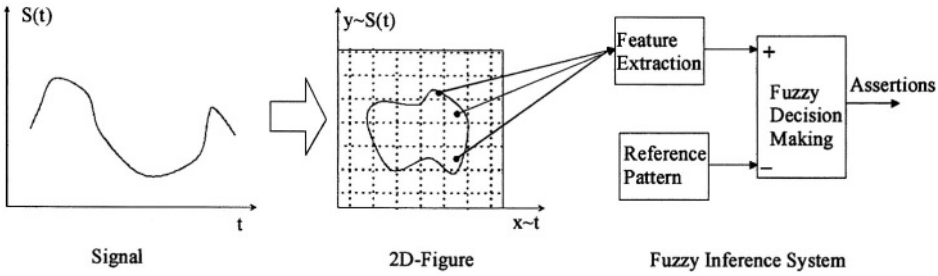


Figure 5.1. Time-Domain Signal Analysis using Fuzzy Logic: general set-up.

The above general idea suggests a straightforward methodology for automatic channel equalization after digital transmissions. It can be derived as follows: Inter-Symbol Interference (ISI) at the received signal is recognized by scanning the input stream over time at the data clock frequency. The resulting 2D-figure is compared against an ideal opened Eye Pattern encoded into a two-input one-output analog Fuzzy Logic Controller. Any deviation from the reference eye results in an error signal used to properly locate the symmetric zeros of an analog amplitude-equalizer gm-C filter, which is intended for the inversion of the transfer function of the channel. In this way, the adaptation can work on-line during transmission.

In view of the future implementation of the whole system suiting the specific equalization requirements for a particular case, a first prototype consisting of the Fuzzy Controller and the equalizer was fabricated and tested in a CMOS technology. However, the methodology was validated by simulations for cable equalization wherein the controller as well as the filter were modeled with some of their actual measured features. The system shows self-adapting capabilities for diverse cable length settings while the ISI is removed in all cases.

This chapter is divided in five sections: firstly, a brief introduction to adaptive channel equalization after digital transmissions is presented. Following, a system-level description of the self-adaptive equalization scheme based on the Eye Pattern and Fuzzy Control is detailed. After that, the design issues and the test results of the fabricated prototype are discussed. Next, the results of the simulations of the entire system for the case of cable equalization are presented and compared against other methods for adaptive equalization based on geometrical information. At the conclusions of the chapter, future works and an extension of the proposed technique to others applications are commented.

5.1 Adaptive Equalization of Channels after Digital Transmissions: a Review

In digital transmission, channel dispersion produces interference between successive symbols making difficult a reliable reception of signals. For this reason equalization is needed. The degree of success of the equalization process can be evaluated through the degree to which the ISI is eliminated and the Bit-Error-Rate (BER) improved. These, in turn, depend on the accuracy of the equalizer when fitting the inverse of the channel frequency response. Automatic adjustment of the equalizer parameters provides flexibility and robustness whenever the channel characteristics are time-variant and/or they depend on variables inherent to the communication system set-up (i.e. the cable length in wired transmissions). Furthermore, adaptive channel equalization allows transmitting signals at higher baud rates than the actual channel bandwidth [WiSt85].

Without loosing generality, we will focus the transmission of NRZ (Non Return to Zero) binary symbols in base-band; this is to say that no carrier signal is modulated by the digital information to be transmitted. This is the case of several digital systems like: magneto-optical disk drive read channels [LaGr93] [BrHu95] [LeLa96], magnetic recording [MeDa87], HDTV cable transmissions [Bake96] and LAN Ethernet transceivers [TaFr98].

5.1.1 The Need of Adaptive Equalization

In real communication systems, channels are non-ideal and their characteristics change slowly over time. Even for the same kind of transmission system, each channel characteristic may differ considerably from each other. Similar to the impulse response of an ideal channel (i.e. the "sinc" function), the non-ideal channel impulse response tends normally to decay over time in both directions. Therefore, inter-symbol interference could be removed by leaving enough space between symbols i.e. by transmitting more slowly than the Nyquist rate. However, greater speed, reliability and flexibility for digital transmissions can be achieved by means of adaptive equalization techniques [WiSt85].

Figure 5.2 illustrates a general scheme of a digital transmission system with adaptive equalization. As depicted in Figure 5.3, the adaptive equalizer performs the inverse transfer function of the channel within the desired passband BW. Outside the passband its gain is ideally zero or smaller. In this way, the resulting magnitude gain of the cascade channel-equalizer is nearly constant inside the passband. In addition, the cascade phase over frequency curve must be linear in order to avoid phase distortion. As long as these requirements are fulfilled, for a given transmission rate $fs<2BW$, the ISI can

be eliminated and the set channel-equalizer represents just a bulk delay. Nonetheless, adaptation of the equalizer becomes mandatory for dealing with the unknown and/or time-variant characteristic of the channel.

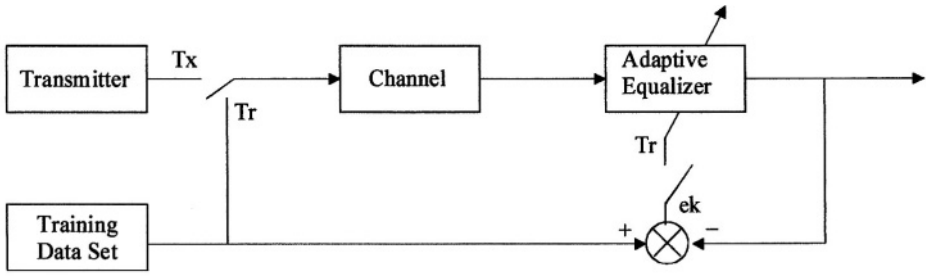


Figure 5.2. Overall communication system with adaptive channel equalization.

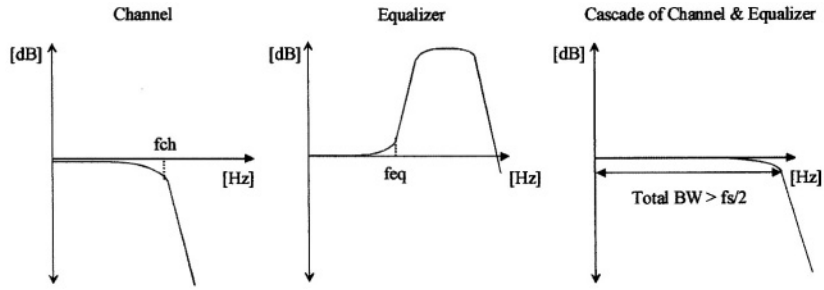


Figure 5.3. Frequency response of the channel, the equalizer and the cascade of both.

In order to adapt the equalizer for tracking channel changes, a training set or reference signal is needed as in most classical adaptive system. Assuming in Figure 5.2 the training data sequence to be also known at the receiver stage, adaptation is carried out when the switches are set to Tr. In this way, the training data set is sent through the channel and compared at the receiver against the same, previously known, data set. Therefore, an error signal ek results from the comparison between the reference training set and the actual output of the equalizer. By using a criterion leading to minimize ek, the parameters of the adaptive equalizer are updated in such a way that feq in Figure 5.3 could be placed as close as possible to fch.

However, in this case the adaptation process must be periodically repeated by interrupting the transmission of information while sending the previously known adapting sequence. For this reason, the implementation of this methodology is worrisome and becomes more and more obsolete, giving place to the so-called self-adapting equalization techniques.

5.1.2 Discrete-Time Self-Adaptive Equalization Based on Geometrical Information of Signals

Self-adaptive equalization methods do not make use of a reference training data set to adapt the equalizer. One of the most popular methods for self-adaptive equalization is the "decision-directed" learning method presented in [WiSt85]. It utilizes the equalizer's own output to yield the adapting signal without the need for a training data set. In Figure 5.4, a reference signal dk=sign(yk) is obtained by quantizing the output of the equalizing filter. Comparing the actual filter output against its binary version yields the error signal ek, which is used for adapting the parameters of the equalizer. The strategy is based on the hypothesis that the quantizer decision is true most of the time what is highly probable when the channel is relatively noise-free and there is no severe phase distortion.

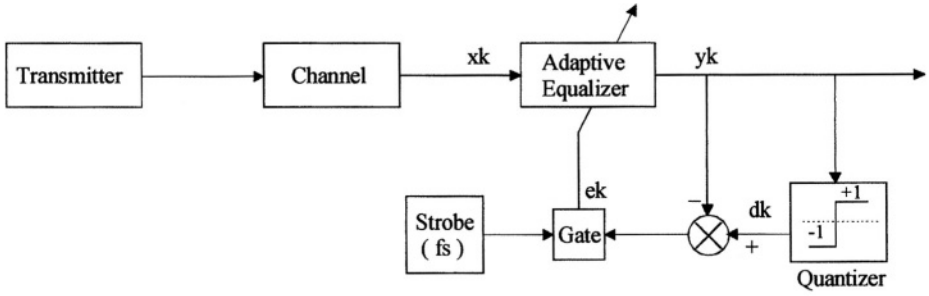


Figure 5.4. Self-adaptive equalization by using the "decision-directed" learning rule [WiSt85].

However, adaptation is enabled only at the center of the incoming "sinc" pulse, where the signal level reaches the best estimation of the actual logical value encoded in such a pulse. Therefore, the error signal ek must be sampled with strobe pulses synchronized to the transmission rate fs.

Apart from the simplicity of this method that makes it attractive for hardware implementation, it is sensible to noise, ringing or other kind of distortion altering the ideal pulse shape of the incoming signal. These non idealities may convey the adapting controller to incorrectly assess at the decision strobe time that, in turn, impels the parameters of the filter to converge towards their appropriate values.

In [Mage94], another self-adaptive method also based on a geometrical criterion is presented. Figure 5.5 illustrates this technique. It consists in sampling the input stream at twice the baud rate fs. Therefore, two samples per bit are obtained. The adapting strategy, performed by a Fuzzy Decision Making unit, consists in comparing these two samples. For instance, assume

the present sample at the peak of the pulse and the previous one near the zero crossing, as shown in Figure 5.5. Thus, one can assert: *"If S(k) is considerably greater than S(k-1) then the pulse is well equalized"*. Otherwise: *"the amount of high frequency boosting must be increased"*. In others words, the decisions for the adaptation are taken accordingly with the quality of the pulse edges rather than with the quality of the peak level of the pulses, as in the former method.

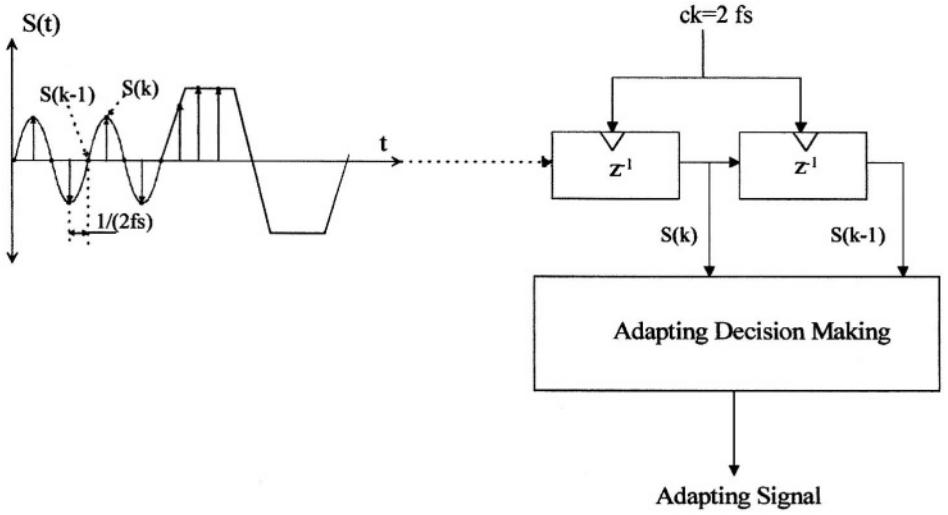


Figure 5.5. Self-adaptive equalization technique by sampling the signal at twice the baud rate.

However, two extra sample and hold circuits are needed. Moreover, since the sampling frequency must be twice the transmission rate fs it compromises the power consumption of the circuits that should perform the adaptation. These shortcomings turn the implementation of the system on a chip a little more complicated, specially for high transmission rates.

In summary, adaptive equalization eliminates inter-symbol interference and makes possible to broaden the channel bandwidth while achieving higher transmission rates. Using this approach in a telephone channel, for instance, the bit-error-rate (BER) can be reduced from 10^{-1} down to 10^{-6} or even lower [WiSt85].

In digital transmissions, the quality of any equalization can be evaluated by means of the well-known Eye Pattern. This is obtained by observing the incoming stream through an oscilloscope whose time base is synchronized to the data clock fs. Therefore, the cycles of the signal are overlaid on the screen as time goes on. Figure 5.6 a) shows the resulted pattern before equalization. Severe ISI distortion can be noticed. When equalization is achieved, the Eye Pattern is open and clear as shown in Figure 5.6 b). In this

situation, the probability of a wrong discrimination of the pulses true value is completely reduced.

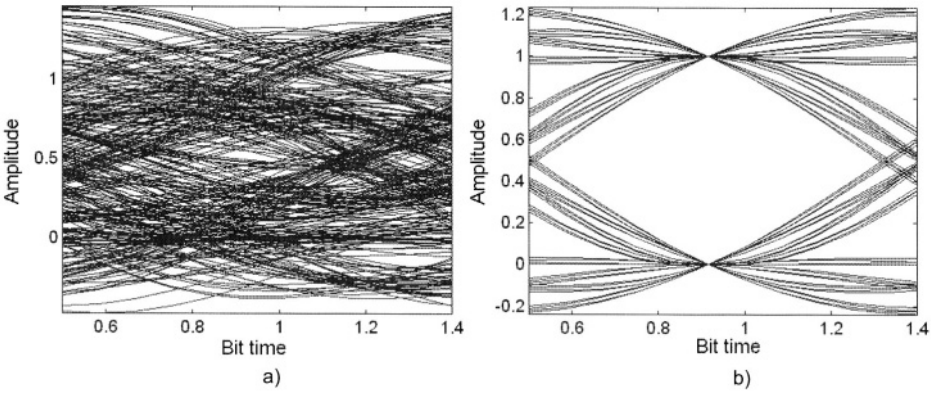


Figure 5.6. Eye Pattern: a) Before equalization. b) After equalization.

5.2 Continuous-Time Self-Adaptive Equalization Based on the Eye Pattern

The Eye Pattern presented in the last section provides enough geometrical information that can be exploited for a knowledge-based approach to self-adaptive equalization [DuVe01]. If the incoming digital stream after the equalizer is swept periodically at the baud-rate frequency, like in any oscilloscope, it should resemble an ideal eye, if well equalized. Otherwise, by comparing the actual eye with an encoded reference eye, a correction signal can be generated and used to update the parameters of the equalizing filter in closed loop. Considering that the adjusting procedure can be easily described through common-sense rules, one can make use of Fuzzy Logic to efficiently encode the non-linear adaptation strategy into a servo controller.

Figure 5.7 shows the block diagram of the continuous-time self-adapting equalization system for the transmission of data symbols at fs bauds. The channel is assumed presenting low-pass characteristics. The first block at the receiver stage is an adaptive high-frequency boost equalizing filter that will be described later. It inputs the equalized signal Vs into a Fuzzy Controller. The second input of the Fuzzy Controller, Vt, is fed from a resetable free-running ramp oscillator whose period Tr equals the inverse of the symbol rate fs. It furnishes a true time base to the Fuzzy System in order to explore the signal over time, like in any oscilloscope. Synchronization is achieved by detecting every rising edge of the signal after the equalizer. As a function of the actual values of Vs and Vt, the Fuzzy Controller bears continuously the

adaptation step ΔKz for the adjustable parameter Kz that results after integrating such step. On the other hand, the magnitude of two symmetrical real zeros belonging to the transfer function of the equalizer is proportional to Kz. Since both zeros have opposite signs, they boost-up the magnitude of the incoming signal without phase shift. In short, the smaller Kz the bigger is the high-frequency boost.

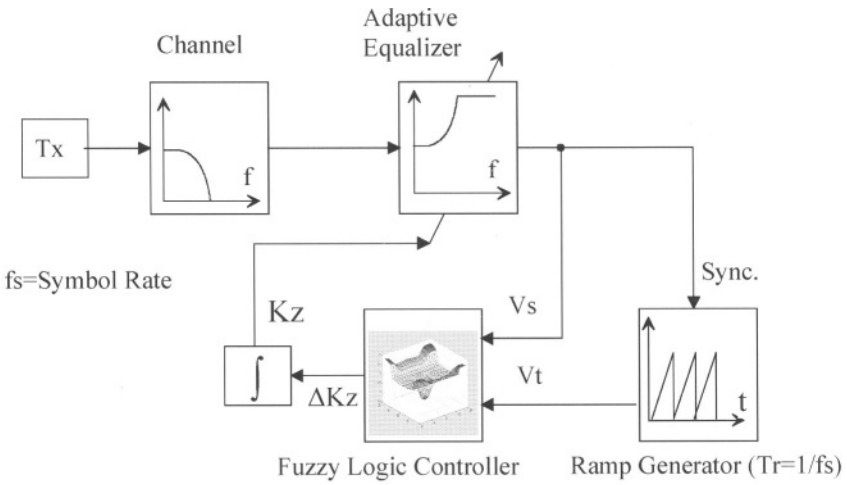


Figure 5.7. Equalizing system based on the Eye Pattern. From [DuVe01], © 2001 IEEE.

With regards to the Fuzzy System, each input variable is fuzzified by means of an odd number of labels: {NB,...Z,...PB}={*negative big,...zero,...positive big*}. Consequently, the input plane [Vs×Vt] is also partitioned in an odd number clusters, each one corresponding to one rule. This is shown in Figure 5.8 a). Considering that the goal is to keep the figure described by the set of the successive points (Vs(t),Vt(t)) as close as possible of an opened eye, the strategy for the adaptation may be easily formulated in terms of fuzzy statements. For example, if the signal is poorly equalized, the 2D-figure will concentrate at the center of the input plane as a consequence of the severe remaining ISI distortion. It turns out that the central rule should look like: *"if the signal is found at the center of the input plane [Vs × Vt], then Kz must decrease"*. After executing this rule, the zeros of the equalizer will move towards lower frequencies attempting to open the actual eye by increasing the amount of boosting.

5.2.1 Fuzzy Controller's Rule Base and Output Control Surface

In order to clarify the control policy needed for the adaptation task, let us take a close look at the rules implemented in the Fuzzy Controller.

Figure 5.8 a) shows the partition (grid partition) of the input plane [Vs×Vt] that has been chosen to encode the Eye Pattern. Figure 5.8 b) illustrates the corresponding control surface for the output variable ΔKz as a function of the inputs Vs and Vt.

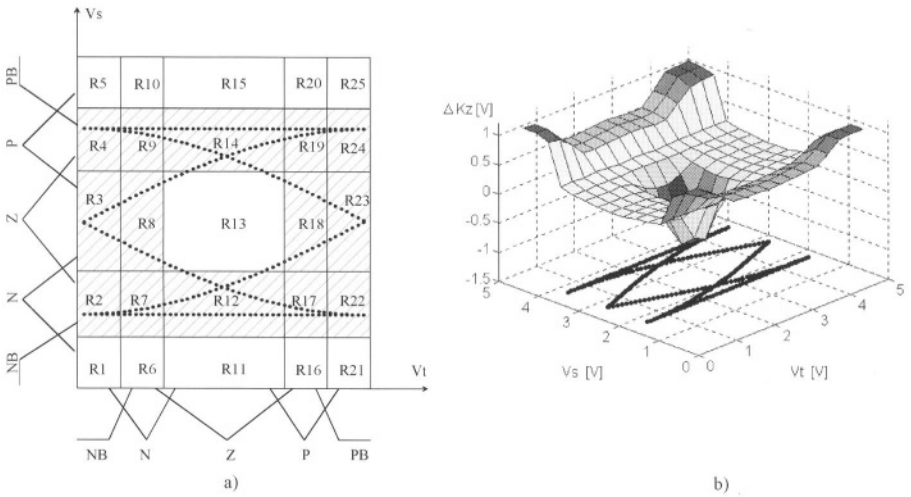


Figure 5.8. a) Fuzzy partition of the input plane [Vs×Vt] and rules map. b) Output surface of the controller related to the Eye Pattern: ΔKz=A(Vs,Vt). From [DuVe01], © 2001 IEEE.

Assuming a Zero-Order Takagi-Sugeno's controller each input variable has been fuzzified by means of five labels {NB, N, Z, P, PB} whereas only four singletons values have been defined for the output space: {Negative, Zero, MediumPositive, Positive}. With regards to the Figure 5.8 a), the 25 rules building the decision-making of the controller are explained as follows:

- Rule 13: this is the central rule that has been mentioned previously. When the digital stream is bad equalized the high frequency components of the signal concentrate in this region. Then, Kz should be decreased for increasing the high-frequency boost. The rule can be expressed as:

  "if Vs is Z and Vt is Z *then* ΔKz is Negative*"*.

- Rules 6, 10, 11, 15, 16 and 20: these rules are intended to control the amplitude of the positive and negative pulses after the eye is opened. In this way, over-equalization is avoided. This control action impels to increase the total bandwidth of the system unnecessarily. Indeed, together with the central rule R13, these ones try to keep the signal inside the

dashed area in Figure 5.8 a). Thus, for these rules we have the following format:

"if Vs is Y and Vt is W *then* ΔKz is Medium Positive",

where Y can take the values {PB, NB} whereas W can take the values {N, Z, P}, in accordance to Figure 5.8 a).

- Rules 1, 5, 21 and 25: the four corner rules have a larger consequent singleton value than the one of the previous rules, as shown in Figure 5.8 b). The goal is to detect and to weight more strongly the probable appearance of overshoots after or before the peak of the pulses to equalize. In this manner, the amplitude of these overshoots is limited because the excess of boosting is strongly reduced. Thus, these rules have the format:

  "if Vs is (NB or PB) and Vt is (NB or PB) *then* ΔKz is Positive".

- The rest of the rules in dashed define the region wherein the actual eye must be kept when equalization is achieved. Thus, the inferred control action must be null when the signal lies inside this region. Consequently:

  "if Vs is U and Vt is T *then* ΔKz is Zero",

  where U can take the values {N, Z, P} whereas T can take the values {NB, N, Z, P, PB}.

It should be noticed that employing this widened rectangular area rather than a fine-shaped Eye Pattern is advantageous. By sizing it properly, noise and/or ringing superimposed to the signal could be filtered out. This prevents the controller to infer wrong decisions for adapting Kz.

5.2.2 Transfer Function of the Adaptive Equalizing Filter

Equation (5.1) suggests one possible realization of the transfer function for the equalizer [ScGh90]. It represents a low pass filter comprising two symmetrical real zeros that introduce amplitude boosting at high frequencies without phase shift. It will be shown later that it can be synthesized starting from a general transconductor-capacitance (gm-C) analog biquad scheme. Actually, pure zeros transfer functions are not physically realizable. For this reason a pair of poles (normally complex) appears at the denominator.

However, these poles must be located beyond the desired transmission bandwidth to avoid counteractions over the boosting frequencies range.

$$E(s) = \frac{-(s/Kz)^2 + wo^2}{s^2 + 2\,\varepsilon\, wo\, s + wo^2}. \qquad (5.1)$$

From the above equation, the position of the zeros and the poles are given by:

$$z1, z2 = \pm Kz\, wo, \qquad p1, p2 = wo\left(\varepsilon \pm j\sqrt{1-\varepsilon^2}\right), \qquad (5.2)$$

where wo is the resonant frequency of the complex poles and ε is their transient damping ratio. Assuming ε=0.707 and wo=$2\pi 10^7$ [rad/s], Figure 5.9 displays the frequency response of the filter for several values of Kz from 0.1 to 1. It is worth noticing that the phase shift of this equalizing filter is kept essentially independent of Kz. On the other hand, the position of the poles defines the maximum amount of boosting for each Kz. Hence, for a given Kz value, the higher wo the larger is the attainable boosting peak.

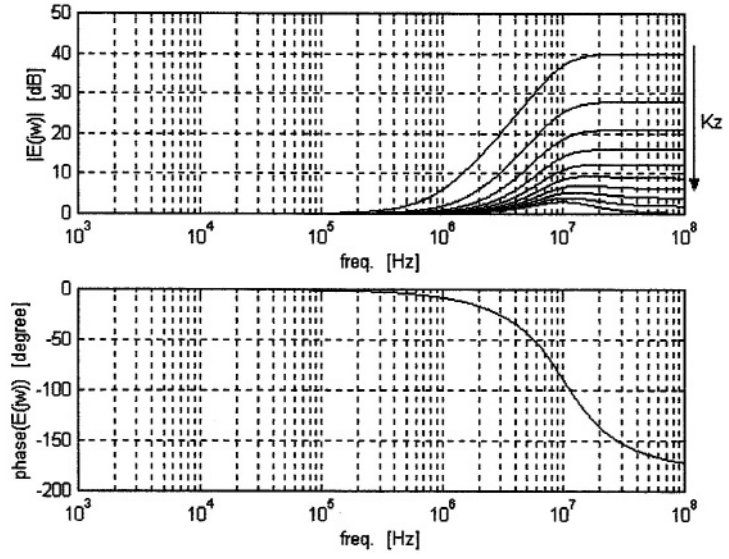


Figure 5.9. Magnitude and phase plots of the ideal frequency response of the equalizer.

Since the poles of the filters define its transient response, small values of the damping ratio yields a poor transient response with ringing. An optimal

value for ε guarantying a fast settling time and tolerable overshoot with no ringing was found to be 0.707 [LaSa94]. As a rule of thumb, wo must be approximately equal to the desired system passband. Therefore, if Kz is always kept smaller than one the zeros z1, z2 remain inside the passband of the system while yielding the desired boosting effect.

5.2.3 Synchronization and Time Base Generation

Despite the ramp generator in Figure 5.7 was not fabricated, only for the completeness of this work we present herein a simple realization that corresponds to a classic time base generation scheme available in any oscilloscope. It has been used for simulating the whole equalizing system.

In Figure 5.10, the ramp voltage Vt results after charging the capacitor C with a constant current source Ir. There are two ways to discharge C via transistor M1. The first is carried out by comparing the actual amplitude of Vt against its maximum peak value Vtmax allowed by the Fuzzy Controller at this input. When the ramp reaches the peak, a positive step is yielded at the output of the comparator Cp2. The cascade of the derivative block and the half-wave rectifier detects the positive edge of the mentioned step by delivering a short positive pulse. This triggers a monostable that, in turn, delivers a pulse with a controlled width τ. The pulse after the monostable enables M1 to discharge C to ground, and a new cycle starts. In order to synchronize the ramp with the binary data stream, a second triggering mechanism is provided. It resets the oscillator by detecting all positive edges of the signal Vs. The comparator Cp1 and a second set of derivative-rectifier blocks take care of this process.

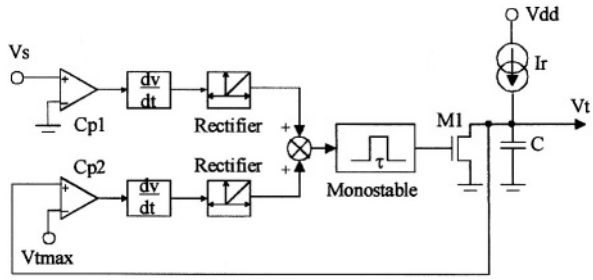


Figure 5.10. Block diagram of the triggered free-running ramp oscillator.

On one hand, the time constant τ of the monostable must be suitably large to fully discharge the capacitor. On the other hand, transistor M1 must be correctly sized in such a way that the discharge time does not exceed a

small fraction of the ramp period Tr=1/fs, as a fast retrace is desired. Given the transmission rate fs, the following relation must hold:

$$Ir = fs\,C\,Vtmax. \tag{5.3}$$

Considering the tolerances along the horizontal axis Vt in Figure 5.8 a) together with the synchronization capabilities of the ramp generator, the accuracy and stability requirements for Ir can be relaxed.

5.3 A Preliminary CMOS Testing Prototype

In view of the future implementation of the whole system suiting a specific application requirement, a first prototype consisting of the Fuzzy Controller and the equalizer has been fabricated and tested in a CMOS-2.4μ technology. We considered that we could get good insight about the system behavior if at least those two fundamental circuits were implemented. In this way, the full characterization of these blocks would allow us to perform simulations including their actual measured performances.

In this section, issues concerning the design of the Fuzzy Logic Controller, as an embedded subsystem, and the equalizer will be addressed. Main topics related to their architectures, sizing criterion, and power consumption are highlighted. With regards to the Fuzzy Controller, some of its building blocks are identical to the ones used in the previous configurations in Chapter 4. Therefore, only the innovations will be focused herein.

5.3.1 Implementation of the Analog Fuzzy Logic Controller

Understanding the controller as a subsystem, simplicity and low-power consumption are primary requirements to be fulfilled. Bearing in mind the kind of application being intended, together with the fact that the controller works in closed loop followed by an integrator, the accuracy needed is not high. It will be demonstrated later that by choosing the integrator bandwidth appropriately the resolution needed for the controller can be considerably relaxed. Even, the exact shape of the output surface is not relevant.

Figure 5.11 illustrates the block diagram of the Zero-Order TS's controller, which is close to the architecture of the first controller presented in the previous chapter. Nevertheless, several fuzzy operators have been changed. According to the low-complexity and the low-power specifications, two five-label compact fuzzy partition circuits has been conceived. The currents delivered at their outputs represent the direct fuzzified values of the inputs Vs and Vt. These currents, after being mirroring, are made available

to the 25 two-input T-Norm circuits to set up the rule base. Considering that the adopted membership functions are not complementary as before, the MIN operator was used as T-Norm. From the T-Norms, the firing degrees of the rules are conveyed by currents I_1-I25 towards the Defuzzifier. The structure and the circuits used for the latter stage are the same as used for the previous controllers, which have been already explained.

Owing to the experimental purposes of this first prototype, some degree of programmability has been considered. Namely, the placement of the membership functions along the input range of the variables Vs and Vt is set via external voltages. Additionally, the consequent singletons of the rules are 5-bit discretely programmable by an internal digital FIFO memory. However, whenever a well-defined target application is contemplated, programmability resources should be eliminated whereas fixed circuits used. In such a case, these fixed circuits must be properly sized and biased following a minimum statistical deviation criterion [PeDu89] [VaVi99].

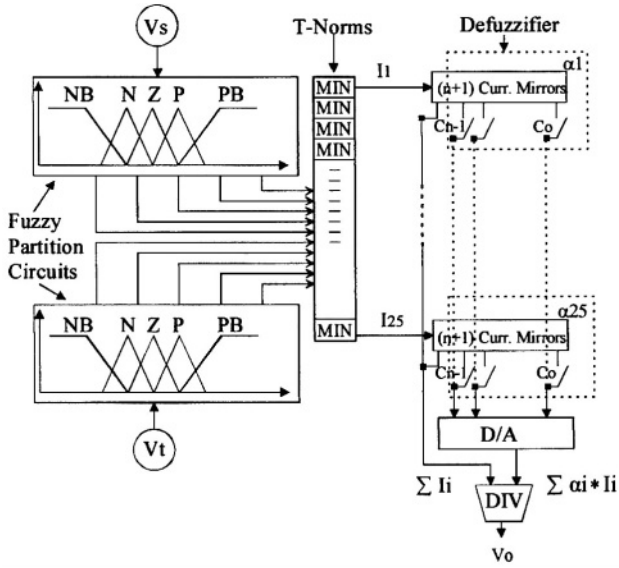


Figure 5.11. 25-rule Takagi-Sugeno's controller. From [DuVe01], © 2001 IEEE.

5.3.1.1 Fuzzy Partition Circuit.

Figure 5.12 depicts a compact circuit generating all fuzzy labels needed for each input variable. It is similar to the one presented in Chapter 3 at section 3.1.5. However, for this embedded application we do not need full programmability. Therefore, in order to reduce the complexity of the controller we can fix some parameters of the fuzzifiers at the mask level. In this particular implementation, four differential pairs of transistors Ma are

interconnected by the upper and bottom cascoded current mirrors. These are built with transistors Mb1-2 and Mc1-2 respectively. The only difference with the scheme in section 3.1.5 resides in the differential pairs whose transistors are saturated [WiJa96]. Thus, the slopes of the membership functions are fixed by the size of Ma and the value of the current Io according to:

$$\text{slope}\ \ [A/V] = \frac{1}{2} gm_{Ma(I=Io/2)} = \frac{1}{2}\sqrt{\mu_n Cox\left(\frac{W}{L}\right)_{Ma}\frac{Io}{n}}. \qquad (5.4)$$

Voltages Vk1<Vk2<Vk3<Vk4 define the 50%-overlapping crossover points between contiguous membership functions. According to the relative values between these voltage references, the position and the width of the fuzzifiers can be modified. In this manner the five current labels {INB,... IPB} are spawned when Vin is swept along its range. The latter currents are supplied to the different MIN operators by cascoded mirrors not shown at Figure 5.12. Figure 5.13 shows a simulation of the DC circuit behavior.

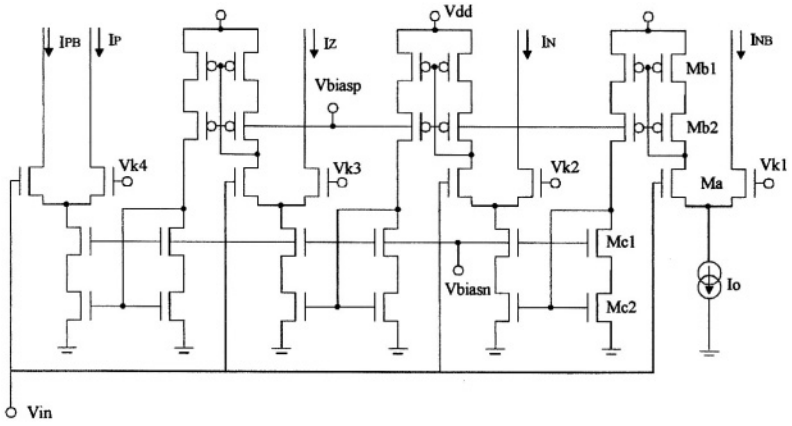


Figure 5.12. Schematic of the five-label compact fuzzy partition circuit [DuVe01].

The total current consumption of this block depends on the actual value of the input Vin. It can range from Io up to 7Io. Assuming the input Vin to be a uniformly-distributed random variable, the mean current consumption is equal to 4Io. This represents a considerable improvement with respect to the use of individual fuzzifiers per label [DuVe00] [VaVi99] [MaFr94], where the total consumption is constant and rises to 10Io for the same number of membership functions per input.

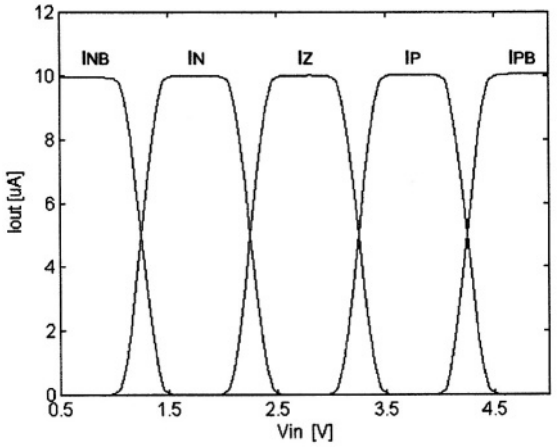


Figure 5.13. DC SPICE simulation of the fuzzy partition circuit for Io=10μA, Vk1=1.25V, Vk2=2.25V, Vk3=3.25V and Vk4=4.25V [DuVe01].

5.3.1.2 MINIMUM T-Norm Circuit.

Figure 5.14 a) shows the circuit used for the 2-input MINIMUM T-Norm operator that has been already explained in Chapter 3. Despite its $O(N^2)$ complexity, the circuit results very simple when a small number of inputs is being handled. Therefore, its use is justified in this case.

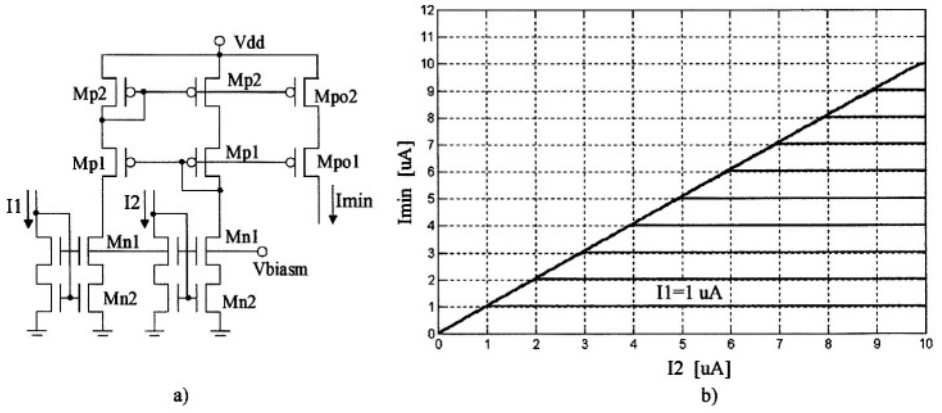


Figure 5.14. a) Two-input $O(N^2)$ MINIMUM circuit [DuVe01]. b) DC nested-sweeps SPICE simulation: I2 ranges from 0 to 10μA while I1 from 1μA to 10μA by 1μA steps.

For the same number of inputs this circuit exhibits superior performance in terms of power consumption and complexity with respect to the other O(N) minimum circuit presented in the Chapter 3 [DoDu00]. Figure 5.14 b) shows a nested-sweeps DC simulation performed with SPICE. Note the

sharp definition of knees at each switching level what can be taken as a qualitative measure of the discrimination capabilities of the circuit.

5.3.1.3 Fuzzy Controller: Sizing and Biasing.

Table 5.1 shows a detailed summary of the sizes and bias of all circuits of this controller including the total current consumption and active silicon area occupied. At the fuzzifiers in Figure 5.12, for the same reasons as argued in Chapter 4, the logical "one" Io was set to 10μA. Using a 5V power supply, the size adopted for transistors Ma gives rise to FMF slopes of 23μA/V. This allows allocating the five labels along the input range of Vs and Vt.

Table 5.1. 25-rule Fuzzy Logic Controller: summary of transistor sizes (in [μ/μ]), bias, current consumption and active area of the circuits. (*) Worst case with only four fired rules.

Fuzzy Partition Circuit (#2)					
Ma	Mb1-2	Mc1-2	Io	Vbiasn	Vbiasp
26/5	20/4	50/4	10μA	1.3V	3V
Input Range		Current Consumpt.		Active Area	
1.5≤ Vin ≤ 4.5V		70μA(max)		9600μ^2	

MINIMUM Circuit (#25)						
Mp1-2	Mn1-2	Mpo1-2	Vbiasm	Input/Output Range	Current Consumpt.	Active Area
24/8	18/4	24/8	1.6V	0 to 10μA	40μA(max)	738μ^2

Consequent Circuit (#25)					
Ms1-2	Mswc	Vaggcas	Input/Output Range	Current Consumption	Active Area
12/4	5/5	1.75V	0 to 10μA	70μA (max)	847μ^2

Weighting D/A Circuit (#1)					
Mbo, Mb1	Mb2	Mb3	Mb4	Resolution	Bit Input Range
5/5	10/5	20/5	28/5	5 bits	0 to 40μA
Output Range		Current Consumption		Active Area	
0 to 40μA		40μA (max)		5240μ^2	

Divider Circuit (#1)							
M1,2	M4,5	M7,8 10,11	M3	M6	M9,12	Current Consumpt.	Active Area
24/6	48/6	74/4	12/6	24/6	37/4	75μA (max)	2412μ^2
Vo	V1	Vdivcas	Input Range (IN,ID)		Output Range		
1.7V	2.7V	2.8V	0 to 40μA		1.7 ≤ Vout ≤ 3.7V		

Buffer Circuit (#1)							
M1,2	M3,4,5,6	M7,8,15,16	M9,10,11,12,13,14			M17,18	
300/3.2	225/2.4	450/2.4	630/2.4			63/2.4	
Avo	fT (CL=10pF)	Phase Margin	Ipol	Vn	Vp	Current Consumpt.	Active Area
72dB	47MHz	65°	60μA	1.5V	2.7V	1.86mA	17774μ^2

Global Performance of the Controller	
Total Current Consumption (*): 2.55mA	Total Active Area: 84251μ^2

Due to the low-compliance cascoded mirrors schemes in Figure 5.12 (PMOS and NMOS) the drain-to-source saturation voltage of their transistors can be kept large enough without compromising neither the input voltage range of the fuzzy partition nor the value of the voltage supply needed. Consequently, the resulting transistor sizes at the mirrors ensure low mirroring mismatch errors, as those transistors remain biased in strong inversion. This working condition also conveys to transistors with relative small size that guarantees small stray capacitances at the internal nodes.

In Figure 5.12 the values of bias voltages Vbiasn and Vbiasp have been set so as to guarantee the saturation of transistors Mb1 and Mc2. Those bias voltages values also contemplate a tolerance to cover the statistical fluctuation of the pinch-off voltages of Mb1 and Mc2 and the threshold voltages of Mb2 and Mc1.

Since the operation of the MINIMUM circuit in Figure 5.14 a) is also based on current mirroring, the same design strategy commented above for sizing and biasing the low-compliance mirrors has been practiced in this circuit. From the simulation results in Figure 5.14 b) the propagation errors were found around 2%. However, due to unavoidable mismatches the latter errors figure is expected to increase at least 1% more.

The rest of the circuits at the defuzzifier stage, namely the consequent singletons, the weighting D/A and the divider have been identically sized and biased as in the 27-rule controller presented in Chapter 4 (i.e. 5 bits of resolution for the singletons and 2V of output swing). The output of the divider was also buffered for testing purposes.

5.3.2 Implementation of the Adaptive Equalizing Filter

Nowadays, a technique frequently used for the implementation of continuous-time adaptive filters makes use of operational transconductance amplifiers (OTAs) and capacitors. These are the so-called gm-C filters. Implemented in CMOS technology, they have shown superior performance than other continuous-time methods like MOSFET-C, for instance, specially in the high-frequency range (i.e. above 2MHz).

Starting from the general structure of a gm-C biquad section, the amplitude boosting transfer function given by (5.1) can be easily synthesized. A single-ended version of the gm-C equalizer is depicted in Figure 5.15 [WySc93].

According to [ScGh90], transmission zeros can be created by injecting a current Iin proportional to Vin into any internal node of the circuit. This is carried out by the transconductor gm5 in Figure 5.15 and does not affect the position of the poles of the original biquad. Applying Kirchhoff's currents law at each node and solving for Vout we have:

$$\frac{Vout}{Vin} = \frac{gm5}{gm4} \frac{-s^2 + \frac{gm1\ gm2\ gm3}{C1C2\ gm5}}{s^2 + \frac{gm2\ gm3}{gm4\ C2}s + \frac{gm1\ gm2\ gm3}{gm4\ C1C2}}. \tag{5.5}$$

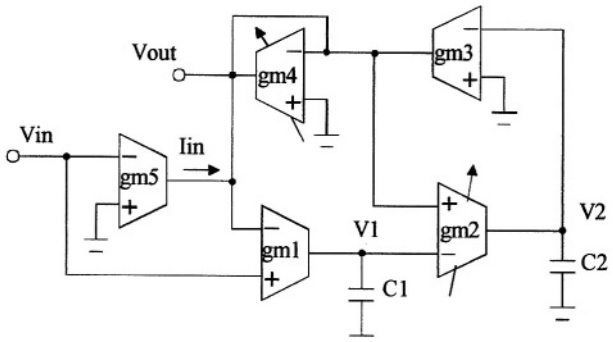


Figure 5.15. Single-ended amplitude-boosting biquad gm-C filter. From [WySc93], © 1993 Kluwer Academic Publishers, reprinted with permission.

Adaptability is accomplished by fixing gm1, gm3, gm5 to a maximum value called **gmmax** while allowing gm2=gm4 to be tunable. Thus, without loading the output node, the transfer function of the filter becomes equal to:

$$\frac{Vout}{Vin} = \frac{-\frac{gmmax}{gm4}s^2 + \frac{gmmax^2}{C1C2}}{s^2 + \frac{gmmax}{C2}s + \frac{gmmax^2}{C1C2}}. \tag{5.6}$$

According to (5.6), (5.1) and (5.2), Kz, wo, ε, z1, and z2 are equal to:

$$Kz = \sqrt{\frac{gm4}{gmmax}},\ wo = \frac{gmmax}{\sqrt{C1C2}},\ \varepsilon = \frac{1}{2}\sqrt{\frac{C1}{C2}},\ z1, z2 = \pm\sqrt{\frac{gm4\ gmmax}{C1C2}}. \tag{5.7}$$

In order to improve noise and distortion figures, power supply and common-mode rejection ratios, a fully balanced version of the filter in Figure 5.15 has been implemented. This also gives the possibility of using floating capacitors (C1, C2) with half the capacitance value needed for a

single-ended realization of the same filter. However, for implementations on double-poly CMOS technologies, grounded capacitors are favored because their parasitic bottom-plate capacitances can be eliminated [WySc93]. Furthermore, one can use the own grounded capacitors of the filter for stabilizing the common-mode loop of the transconductors driving those capacitors (i.e. floating capacitors do not represent a load to common-mode signals). This common-mode loop is performed between the active load of the transconductor and the Common-Mode Feedback circuit (CMFB), as it will be explained in the next section.

5.3.2.1 Novel Full Electrically-Tunable Triode Transconductor.

Based on the circuit of the divider shown in Figure 5.16 b), a novel pseudo-differential triode transconductor is presented in Figure 5.16 a). For this case, the gate voltage of transistor M3 in the divider is now released to become the input $\mathbf{Vin}^+$ of the transconductor. However, the current I^+ through M10 (former IN at the divider) is now controlled by the negative feedback loop performed via the latter mentioned transistor. Additionally, the up-scaled mirror M11 delivers $\mathbf{Iout}^+ = \mathbf{B}I^+$ at the transconductor output. Similarly, transistors M12 to M16 set up the complementary branch for the input $\mathbf{Vin}^-$.

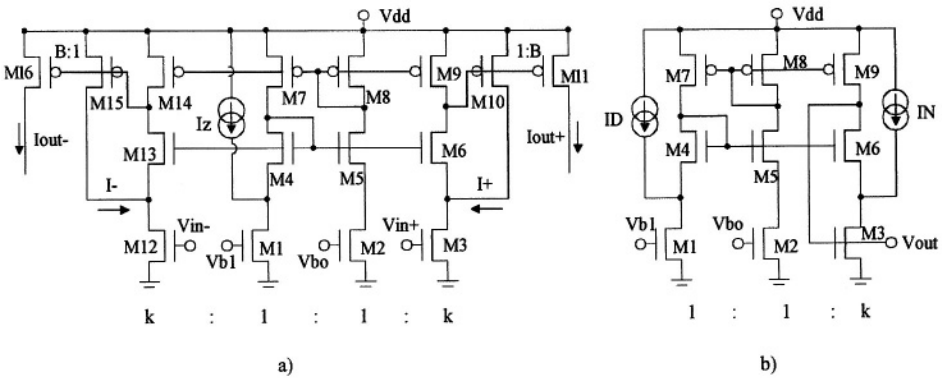


Figure 5.16. a) Electrically-tunable pseudo-differential triode transconductor based on the transresistive divider shown in b). Current Iz in the transconductor is equivalent to current ID in the divider. From [DuVe01], © 2001 IEEE.

Upon ideal matching conditions and provided that M1, M2, M3 and M12 in Figure 5.16 a) are being biased in ohmic region, the whole set of equations developed for the divider in Chapter 3 still holds. Therefore, assuming a floating load at the differential output of the transconductor (see Figure 5.17 a)) the differential current I_L through such load can be expressed as:

$$I_L = \frac{1}{2}\frac{k\,B\,Iz}{(Vb1 - Vbo)}(Vin^+ - Vin^-) = gm\,(Vin^+ - Vin^-). \qquad (5.8)$$

Notice in Figure 5.16 a) that the section of the circuit comprising transistors M1, M2, M4, M5, M7 and M8 plays the role of a biasing scheme only. In addition, the conditions Iz>0 and Vb1>Vbo must be always sustained for the correct operation of the transconductor.

Bearing in mind that in the latter equation k and B are transistors size ratios, the transconductance gm becomes fully electrically-controllable by Iz and (Vb1–Vbo). This is to say, upon ideal short-distance matching conditions, the transconductor features complete independence on technological parameters **(μCox)**. In this way, one can disregard the long-distance matching conditions between the transconductors gm building the filter. This characteristic turns this circuit attractive for implementing gm-C filters, which are typically built by an arrangement of identical transconductors gm. On the other hand, as long as the electrical variables controlling gm are being supplied by reference bias circuits, stability against temperature and power supply drifts are guaranteed.

Figure 5.17 a) shows the complete transconductor gm including the cascoded active load and the CMFB circuit. The latter is detailed in Figure 5.17 b).

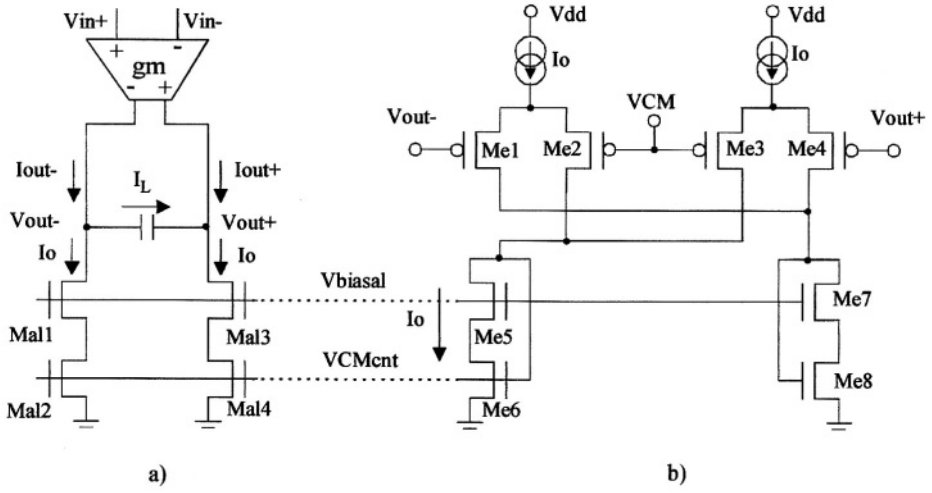


Figure 5.17. a) Complete schematic of the fully differential transconductor. b) "Current steering" CMFB circuit.

The CMFB circuit allows setting the DC common-mode voltage VCM (i.e. normally set to half the value of the rail-to-rail voltage supply) at the

balanced outputs of the transconductor. Its working principle is explained in [MaJo97]. In [DuCa93] the good performance of this circuit has been demonstrated, namely high processing speed for both the differential and common-mode signals, and small THD induced. The latter property is due to the high common-mode loop gain that this CMFB circuit holds, which attenuates the non-linearities introduced by saturated transistors Me1-Me4 [DeDu00-b]. In addition, the common-mode feedback loop can be stabilized with the own load capacitors of the transconductor, if those capacitors are grounded.

However, the main drawback of this CMFB circuit arises when the quiescent currents at the transconductor branches ($Iout^+_Q$,$Iout^-_Q$) change during tuning. In such a case, assuming in Figure 5.17 b) fixed currents Io biasing the CMFB circuit, the only way this circuit could supply the suitable quiescent currents to the transconductor is by a non-symmetric current splitting at each transistors of the differential pairs of the CMFB (Me1, Me2 and Me3, Me4). Consequently, the actual DC output voltages of the transconductor, $Vout^+$ and $Vout^-$, offset the intended reference VCM. In short, the outputs DC level shift as long as the transconductor is being tuned.

This problem has been overcome in [DeDu00] by means of the use of adaptive currents Io at the CMFB circuit instead of fixed ones. An adaptive bias generator circuit provides to all CMFBs in the filter the current Io, which is exactly matched to the quiescent currents $Iout^+_Q$ and $Iout^-_Q$, changing upon tuning. In our case, however, a simpler way to generate the adaptive current Io for the CMFBs is by mirroring the transconductor tuning current Iz while accomplishing with:

$$Io = Iout^+_Q = Iout^-_Q = \left[k\,B\,\frac{(VCM - Vbo)}{(Vb1 - Vbo)} \right] Iz. \qquad (5.9)$$

As a by-product of adapting biasing, the current consumption of the CMFB is optimized if compared to classical CMFB circuits with fixed bias Io. In the latter, Io must be set to the maximum quiescent current demanded by the transconductor.

Another important issue to consider for the design of the transconductor concerns its frequency response. Figure 5.18 illustrates the small-signal equivalent circuit of one side of the transconductor (the branch corresponding to Vin+ in Figure 5.16 a)) together with the parasitic capacitors identification. The gate voltage of transistors M6 and M9 are fixed by the section of the circuit consisting of transistors M1, M2, M4, M5, M7, and M8. As stated above, this section of the circuit plays the role of a

biasing scheme. Thus, after some simplifications, the small-signal transfer function of the transconductor is given by:

$$\frac{Vout}{Vin} = \frac{gm11\,gms6}{C2\,(C3+C1)} \cdot \frac{(-s\,C3+gm3)}{s\,CL+(gL+gd11)} \cdot \frac{1}{s^2+\frac{gms6}{(C3+C1)}s+\frac{gm10\;gms6}{C2\,(C3+C1)}} \cdot \tag{5.10}$$

Note that it comprises three poles and a right half-plane zero. The latter appears as a consequence of the increased gate-to-drain stray capacitance Cgd of transistor M3, provided this transistor is biased in the triode region.

Two conditions are needed for an acceptable behavior of the transconductor in the frequency domain. The first concerns the absolute stability that imposes to get a first order system. This is guaranteed if the circuit presents only one low-frequency pole. In fact, this is the case of this circuit and the dominant pole is found at the output node Vout in Figure 5.18. It is related to the time constant product between the high output impedance and the large load capacitor CL. At first glance, the node V2 in Figure 5.18 seems also to be a high-impedance node. Nevertheless, due to the negative voltage feedback performed by transistor M10, the actual impedance seen at such node results in two orders of magnitude much smaller. Consequently, the pole associated to this node is located at high frequencies, far beyond the transition frequency of the transconductor.

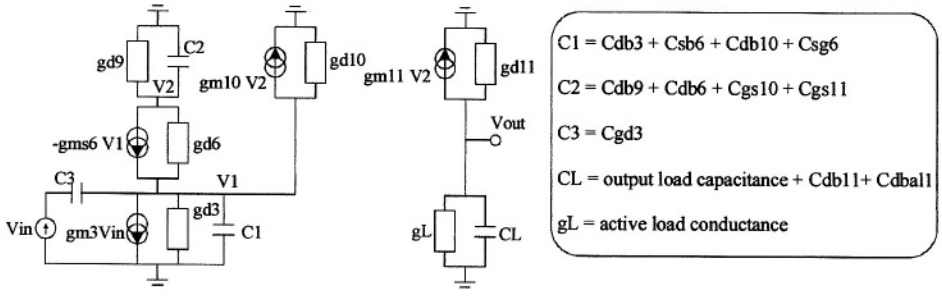


Figure 5.18. Small-signal equivalent circuit of the transconductor and stray capacitors identification.

The second condition to accomplish is related to the phase errors that all singularities of the transfer function of the transconductor may introduce. The position of the dominant pole of the transconductor with respect to its transition frequency causes "lead" phase errors. The relative position of the

others singularities outside the passband produces "excess" phase errors. Both kinds of phase errors are measured with respect to the 90° at 0dB gain of an ideal integrator (in fact, a transconductor loaded with a capacitor is just a gm-C integrator). While large "lead" errors in the transconductors of a filter may considerably deteriorate its quality factor Q, large "excess" errors may yield magnitude peaking at the corner frequency of the filter [KhGr84] [Kard92]. The latter phase error may also provoke parasitic oscillations. Therefore, in order to avoid these shortcomings all singularities of the transfer function in (5.10) must be kept at least one decade above and/or below the transconductor transition frequency. Accordingly, the following relationships should be accomplished:

$$\text{for } wT = \frac{B\,gm3}{CL},\ |wpo| < 0.1\,wT,\ \ |wp1|, |wp2|, wz > 10\,wT, \quad (5.11)$$

where wT is the transconductor transition frequency ($2\pi fT$), wpo is the dominant pole, wp1 and wp2 are the non-dominant poles and wz is the right half-plane zero, all expressed in [rad/s].

5.3.2.2 Filter and Transconductors Sizing.

The design parameters for the equalizing filter (see Figure 5.15) were chosen to achieve a 10MHz passband. On the other hand, a transient damping ratio of 0.707 implies (C1/C2)=2, in accordance with (5.7). Consequently, considering the values of the grounded capacitors C1=6pF and C2=3pF, gm_{max} in equation (5.6) and (5.7) must be set around $130\mu S$ for attaining the desired bandwidth. Following the filter specifications the sizing of the transconductor for a +5V power supply is considered in the following.

The input-output voltage range of the transconductor (see Figure 5.16 and Figure 5.17) is conditioned by the bias voltage Vbo, on one hand, and by the common-mode DC voltage VCM, on the other hand. The latter has been set to half the rail-to-rail power supply. For proper operation of the transconductor the following relation must hold:

$$Vbo < Vin^{+}, Vin^{-} < (2\,VCM - Vbo). \quad (5.12)$$

According to the above equation, the smaller Vbo the larger the input-output swing and the larger the attainable dynamic range for a given noise floor. However, as long as Vbo decreases, the aspect ratio of the triode transistors of the transconductors should be increased for keeping these transistors unsaturated. On the other hand, to move the zero wz at high frequencies the use of relative small size triode transistors is encouraged, as

their gate-to-drain capacitances would result diminished. A final aspect to consider is that the transconductance given by (5.8) can also be increased if (Vb1-Vbo) is kept small. In such a case however, the quiescent current given by (5.9) will also result high. In conclusion, for a given gm_{max} needed in (5.6), the value of Vb1, Vbo and the size of transistor M3 must be carefully chosen so as to trade input-output swing, current consumption and the correct placement of the right half-plane zero wz. At the same time tolerable values for k, B and Iz should be adopted.

Consider now transistors M6 and M9 in Figure 5.16 a) and the node V2 in Figure 5.18. Small phase errors have been targeted by adopting a suitable geometry for those transistors for minimizing stray capacitance effects. On the other hand, according to (5.8) the transconductance is also proportional to k, which can be set greater than one. Therefore, for a given gm_{max}, increasing k allows decreasing B. This, in turn, reduces the stray capacitances at node V2 because transistor M11 (Figure 5.16 a)) becomes smaller, with reduced Cgs capacitance. Finally, the maximum drain-to-source saturation voltage Vds_{sat} of M6 and M9 should be set in accordance to the power supply value and the signal swing desired at each node.

Taking into account all above commented items the transconductor was sized and biased in such a way that equations (5.11) are sustained over the whole transconductor tuning range. Table 5.2 summarizes the filter and transconductor characteristics.

Table 5.2. Filter and transconductor: transistors size (in [μ/μ]), biasing and performance.

Adaptive Equalizing Filter							
wo [Mrad/s]	ε	Kz	z1, z2 [Mrad/s]	p1, p2 [Mrad/s]	C1 [pF]	C2 [pF]	gm_{max} [μS]
62.83	0.7	0.1<Kz<1	±62.83 Kz	44.4 ± j 44.4	6	3	130
Amplitude Boost (theoretical)			Current Consumption		Active Area		
40dB (Kz=0.1)			5.25mA (max)		100807μ²		
Pseudo-Differential Triode Transconductor							
k	B	M1,2	M3,12	M4,5	M6,13	M7,8 (cascoded)	M9,14 (cascoded)
2	3	10/10	20/10	15/5	30/5	56/3.2	112/3.2
M10,15 (cascoded)		M11,16 (cascoded)	Mal1,2,3,4	Me5,6,7,8	Me1,2,3,4	M_{Io} (cascoded)	
25/3.2		75/3.2	55/5	55/5	40/3.2	160/3.2	
Iz		VCM	Vbo	Vb1	Vbiasp	Vbiasal	
0<Iz<22μA		2.5V	1.75V	2.25V	3.4V	2.8V	
Input Range		Current Consumption	Phase Errors (CL=2pF)	fT Rate (CL=2pF)	Active Area		
1.75V<Vin+, Vin-<3.25V		ICC<750μA	< 2°(excess)	0.41MHz/μA	9240μ²		

Simulations predict a linear dependence of the transition frequency $wT=2\pi fT$ on the tuning current Iz at a $0.41MHz/\mu A$ tuning rate. Phase errors result smaller than 2° (in excess) in all cases. The transconductor outputs DC level is kept within a ±10mV tolerance band around VCM over the whole tuning range. The maximum current consumption of the transconductor, including its CMFB circuit, is lower than $750\mu A$.

Table 5.3 performs a comparison of our transconductor against a classical approach based on transconductance linearization by source degeneration [KrJo88]. To conclude, let us remark that owing to the possibility of controlling the transconductance value by means of electrical references, this transconductor becomes an interesting alternative for applications with strict stability requirements (i.e. gm/C oscillators, accurate gain gm-based amplifiers, etc).

Table 5.3. Comparison of the new transconductor against other classical approach.

	[KrJo88]	Our Approach
Technology:	CMOS-3μ	CMOS-2.4μ
Power Supply:	±2.5V	5V
Current Consumption:	400μA@4MHz	750μA@10MHz
Maximum Differential Input Swing: (ΔVin_{max})	1.8V@gm=150μS	3V@gm=100μS
Transconductance Linearity: ($\Delta gm/gm$)×100	±0.8% (@ΔVin=1.8V)	±0.6% (@ΔVin=3V)
Phase Errors:	no data	~ -2°

5.3.3 Testing the Preliminary Prototype of the Equalizing System

Due to the different nature of the two circuits implemented in this first prototype, each one has been separately submitted to its corresponding static and dynamic testing strategy. Extracting the maximum ratings of each circuit (i.e. input-output voltage ranges, current consumption, processing speed, bandwidth, etc) allows performing a computer simulation of the whole equalizing system including some of the measured features.

5.3.3.1 Fuzzy Controller: Test Results.

The DC characterization of the controller, its accuracy and the statistical fluctuation between samples have been addressed in the same manner as in the previous controllers in Chapter 4. The measurement set-up corresponds to the one shown in Figure 4.11.

Figure 5.19 a) shows the measured output surface resulting after loading and setting the corresponding parameters. The values of the singletons have been chosen intending to approximate the control surface shown in Figure 5.19 b). The RMSE between these two surfaces reaches to 90mV, which

represents 4.5% of the maximum output swing (2V). Most relative errors are comprised within ±5%. Nevertheless, they can reach up to 14% in some points. These outliers normally take place at the transition between contiguous sub-regions (rules cluster) characterized by a considerable difference between their associated singleton values.

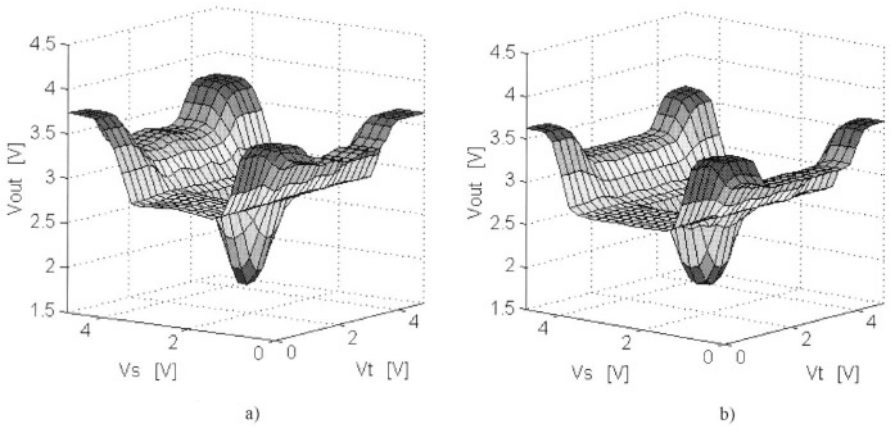


Figure 5.19. a) Measured output surface. b) Target output surface.

To perform the statistical characterization of the circuit six prototypes holding the same set-up have been measured and the resulting surfaces are shown superimposed in Figure 5.20 a). The standard deviation at each point of the control surface has been calculated and its distribution is drawn in Figure 5.20 b). The maximum standard deviation is close to 140mV (7%) whereas the mean reaches to 75mV (3.75%).

It should be mentioned that the errors and spreading figures strongly depend on the function programmed in the controller. However, for this kind of application the exact shape of the control surface is meaningless. Even with the measured RMSE the control task is fairly accomplished. Moreover, we have also performed system-level simulations by using controllers with similar shapes but different RMSE. In all cases, no significant change could be observed. When higher accuracy is needed but no tuning capabilities are being provided, a major emphasis in transistors sizing aiming at minimizing the random errors should be focused [VaVi99]. However, if this design strategy is put in practice, increased transistor sizes should be expected. Therefore, stray capacitances would result larger [VaVi99] whereas higher biasing currents should be used if the same processing speed is desired.

The transient behavior of the circuit was typified by measuring its step response. For the large swing step response, the input Vt was set to a

constant value whereas a 2.5Vpp step was applied to the input Vs. This gives rise to a 1Vpp step at the output. The measured input-output delay reaches to 150ns whereas the fall time resulted in 750ns (for 90% of the steady state value reached). This results in a total settling time of 900ns. For the small swing step response, the amplitude of the pulse applied to the input Vs was reduced to 380mVpp. This yields a 50mVpp step at the output. In this case, the delay and fall time result in 150ns and 230ns, respectively. Thus, the settling time reaches to 380ns (90% of the steady state value). Certainly, as it is difficult to identify the inputs biasing condition for the largest delay, these results give only an idea about the transient behavior of the controller, which may change from one point to another in the input space. However, in this controller the delay introduced by the fuzzifiers represents an important fraction of the total delay. This is due to the cascade interconnection of the differential pairs of the fuzzy partitions. Improved delay figure should thus be obtained if the intermediate mirrors of the circuit in Figure 5.12 were biased (i.e. Mb1, Mb2, Mc1, Mc2).

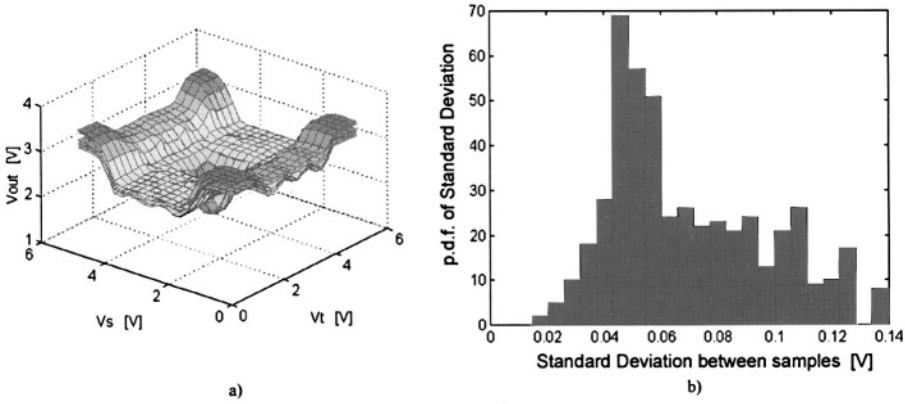


Figure 5.20. a) DC Test results in six different chips. b) Distribution of the standard deviation between samples.

5.3.3.2 Equalizing Filter: Test Results.

The testing strategy for the gm-C equalizer filter comprises its frequency response characterization. Magnitude and phase over frequency were measured with the HP4195 Network Analyzer.

Before discussing the measurement results, let us consider an additional effect. As long as stray capacitors (i.e. compensation capacitor of the CMFB circuit and routing wires) are connected at the outputs of the filter, a third pole appears in the filter transfer function. This makes the actual frequency response of the filter differ from the expected one explained in section 5.3.2. This effect can be observed in Figure 5.21, which shows the measured

magnitude and phase curves for different Iz values. Notice from Figure 5.21 that the additional pole defines now the peak values and the peak positions in the magnitude over frequency response curve.

We could measure the AC response of the filter for Iz ranging from 15μA to 35μA (@Vcc=5.5V). The boosting peaks are found at 7MHz, nearly. As expected, the amount of boosting increases as the current Iz decreases. However, for lower Iz values the phase characteristic of the filter results considerably degraded bearing small damping ratios. This conveys to impoverished transient behavior. Therefore, the effective tuning range of the equalizer gets smaller than the expected one.

The attractiveness of this filter lies mainly in its relative small complexity for implementing symmetrical zeros if compared with other approaches [TaFr98]. Nonetheless, it has been shown that its frequency response is sensitive to the capacitive output loads that can modify the filter transfer function while limiting its tuning range. One can minimize this effect by using other kind of CMFB circuit whose compensation capacitor does not load the transconductor (i.e. at least for the transconductor driving the output of the filter) [MaNe89].

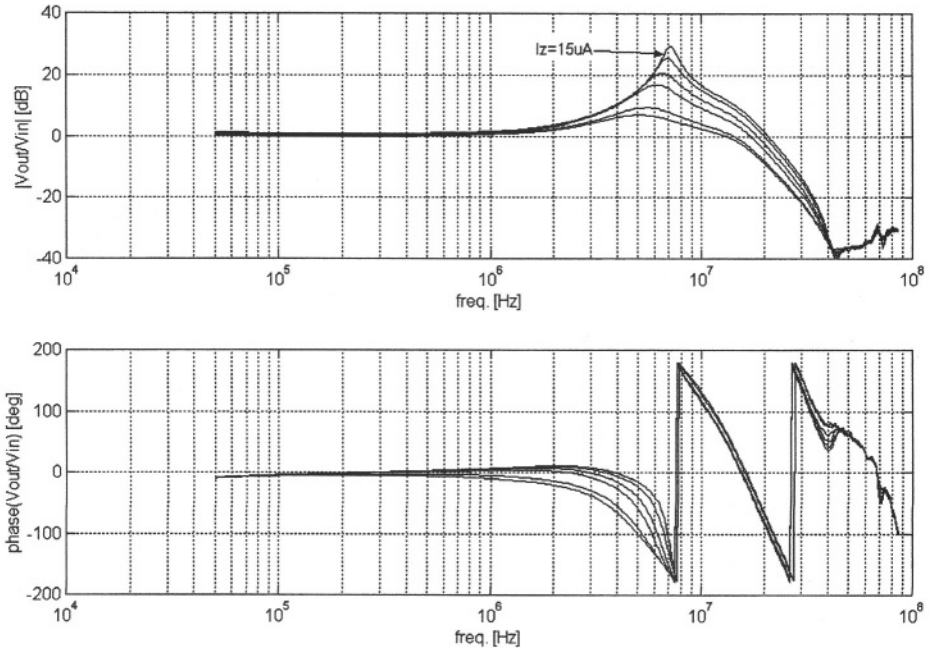


Figure 5.21. Measured magnitude and phase response of the equalizer for 15μA< Iz<35μA [DuVe01].

5.3.3.3 Summary and Conclusions on the Testing Prototype.

Figure 5.22 shows the microphotograph of the testing prototype encapsulated in a JLCC-68pins package. Table 5.4 summarizes the main measured features of the controller and the equalizer.

In a full-ended prototype supporting the whole equalizing system, the buffer at the output of the Fuzzy Controller and the internal digital FIFO can be eliminated. Upon this condition, the controller represents the 35% of the total area and the 11.5% of the total power consumption. These figures encourage the use of this Fuzzy Controller as an embedded subsystem for this kind of application.

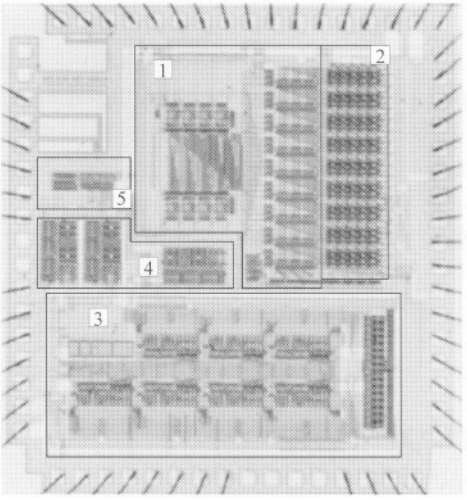


Figure 5.22. Microphotograph of the prototype for adaptive equalization: 1) Analog Fuzzy Logic Controller. 2) 5x25-bits FIFO memory. 3) Analog gm-C equalizing filter. 4) Output buffers. 5) Bias circuits. From [DuVe01], © 2001 IEEE.

The minimum bandwidth that a channel must hold for being properly equalized is defined by the frequency wherein the magnitude response of the equalizer in Figure 5.21 attains 3dB (before the boosting peak). Hence, this equalizer can boost channels with a minimum bandwidth of 1.5MHz while extending the cascade channel-equalizer bandwidth up to 7MHz. This limit is defined by the frequency wherein the equalizer attains the boosting peaks. In agreement with the Nyquist rate [Proa95], a maximum (theoretical) transmission rate of 14Mbauds should be achievable. However, according to the measurements, the speed of Fuzzy Controller becomes the limiting factor for the maximum transmission rate allowed through this preliminary prototype.

Table 5.4. Main measured features of the Fuzzy Controller and the Equalizer.

Fuzzy Logic Controller		
Technology:	CMOS-2.4μ	
Complexity:	25-rule@2-input@1-output	
Power Supply:	Vdd=5V	
Power Consumption:	Core:	4.4mW
	Buffer:	9.5mW
	Total:	13.9mW
Area:	Analog:	$2.9mm^2$
	FIFO:	$1.1mm^2$
	Total:	$4mm^2$
Accuracy:	RMSE:	90mV (4.5%)
Standard deviation between samples (#6):	Maximum:	140mV (7%)
	Mean:	75mV (3.75%)
Input/Output Delay:	Small swing:	380ns
(90% of steady state value)	Large swing:	900ns
Amplitude-boosting gm-C filter		
Technology:	CMOS-2.4μ	
Complexity:	biquad@2-zero@2-pole	
Power Supply:	Vdd=5.5V	
Power Consumption (Nominal: Iz=25μA):	Core:	27mW
	Buffers:	19mW
	Total:	46mW
Area:	$5.3mm^2$	
Bandwidth:	7MHz@Iz=35μA.	

5.4 An Investigation on Self-Adaptive Cable Equalization

Loss and phase dispersion presented in a long cable must be compensated when high-speed digital cable transmissions are intended. Moreover, for applications where the cable length may vary, the use of adaptive equalization becomes mandatory. An adaptive cable equalizer comprises mainly an analog adaptive equalizing filter, which must adapt its transfer function according to the cable characteristics [Bake96].

For this particular experiment, a CAT5 UTP (Unshielded Twisted Pair) cable type has been chosen. It corresponds to the 100Base-TX Fast Ethernet transceiver application for digital transmission rates up to 125Mb/s. The equalization requirements can be drawn from the characteristic of the cable propagation loss, which is given by:

$$\left|C(f)\right|_{dB} = -10^{-2}\, L\, \left(2\sqrt{f} + 0.023f + 0.05/\sqrt{f}\right), \qquad (5.13)$$

where the cable length, L, is expressed in meters and the frequency, f, in MHz [TaFr98].

Despite that this transmission norm contemplates only a maximum cable length of 120m, we will consider also transmission over longer cable lengths. This is due to the fact that our circuits work at lower frequencies than those required for a transmission at 125Mb/s. Consequently, to produce a significant attenuation at the maximum working frequency of our prototype, the cable length be considerably increased. It must be kept in mind that this experiment intends validating an idea rather than solving a particular well-defined equalization problem.

5.4.1 System Stability and Parameter Convergence Considerations

Starting with the block diagram of Figure 5.7 and incorporating some measured performances of the filter and the controller, the stability analysis focuses mainly on the behavior of the two later mentioned blocks working in closed loop. They are redrawn in Figure 5.23 together with the differential equations describing the system behavior in time-domain. The most important requirement to fulfill is the absolute stability of the system that must be ensured for every value of the adapting parameter Kz.

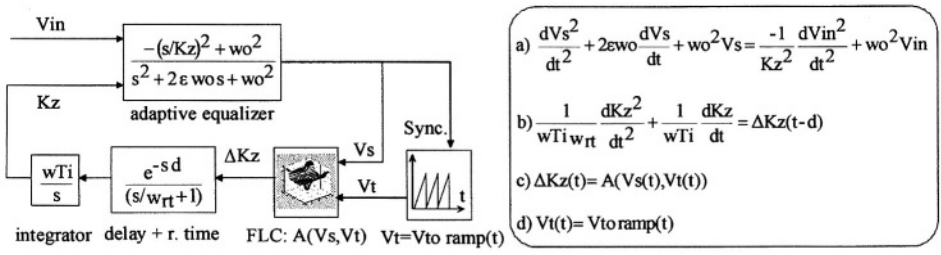


Figure 5.23. Closed-loop block diagram and differential equations system of the filter and the Fuzzy Logic Controller (FLC). Some measured parameters have been included (i.e. d, w_{rt}, wo and A(Vs,Vt)).

In the above equations, A(Vs,Vt) represents the non-linear DC transfer function of the Fuzzy Controller. Its model was obtained by fitting with ANFIS (RMSE=2.8%) the actual measured surface shown in Figure 5.19 a). A new block consisting of a transport delay d and a real negative pole w_{rt} was added in order to model the measured response of the controller (delay + rise time). The value of w_{rt} was chosen so that its associated time constant matches the measured rise of the controller. A band-limited integrator with a transition frequency of $wTi=2\pi fTi$ is assumed. For simplicity, only the measured wo, the maximum attainable amplitude boost

and the ideal transfer function given by (5.1) have been considered for modeling the filter.

By solving the system in Figure 5.23 one could find the final transfer function from Vin to Vs from which the stability conditions could be drawn. Unfortunately, equation a) is time variant (i.e. Kz(t)) whereas c) is highly non-linear. Thus, the system cannot be solved in closed form. However, a linear approximation of the transfer function A(Vs,Vt) of the Fuzzy Controller around a particular point conveys to a numerical small-signal solution. This can be done for different initial values of the parameter Kz, for different values of the linearized gain A(Vs,Vt) and for different values of the transition frequency of the integrator wTi. It will be shown below that the latter parameter plays an important role on the asymptotic convergence of the filter adapting parameter Kz.

In a linearized small signal analysis, the actual value of Vt only alters the value of A(Vs,Vt) according to the point of the input plane [Vs×Vt] wherein the linearization is being carried out. Therefore, Vt can be eliminated from equation c). Moreover, taking a close look at the block diagram in Figure 5.23, one can realize that the integrator and the controller are cascaded in the same signal path. Hence, the gain A(Vs,Vt) is multiplied by the parameter wTi. Consequently, instead of performing transient simulations for a range of linearized gains A(Vs,Vt) we can fix it to its maximum value while allowing to change wTi. In this way, we are able to find out the stability conditions for different values of wTi only.

The stability criteria is thus reduced to investigate numerically if a bounded-input Vin yields a bounded-output Vs for several initial conditions of the parameter Kz and for different values of the integrator transition frequency. In this direction, the integrator transition frequency was set to 7.5KHz and 75KHz in two different simulations, assuming in each one three different initial conditions for Kz. In all cases the gain A(Vs,Vt) was fixed to 2, its maximum value extracted from Figure 5.19 a), whereas a very short pulse was applied to Vin. By using a numerical method of MATLAB for dealing with differential equations the above system was resolved and the results are displayed in Figure 5.24 for the different cases. Several remarks can be stated from this figure:

- Despite that, for small values of Kz, Vs presents a large initial undershoot followed by an overshoot, the signal energy of the response is finite with zero steady-state value. Since Vs is bounded the system is stable over the whole range of Kz.
- The appearance of the above mentioned undershoot is due to the derivative nature the equalizer acquires when its symmetrical zeros

approach to the zero frequency as Kz decreases. With real band-limited input signals these undershoots result further diminished.

- The peak values of Vs are not affected by the integrator transition frequency wTi whereas the total settling time of Vs, rounding to 0.1μs, is almost independent on the initial value of Kz.
- The response of Kz is also bounded. However, as expected, the steady-state value of Kz strongly depends on wTi: it increases as a larger integrator transition frequency is being used.

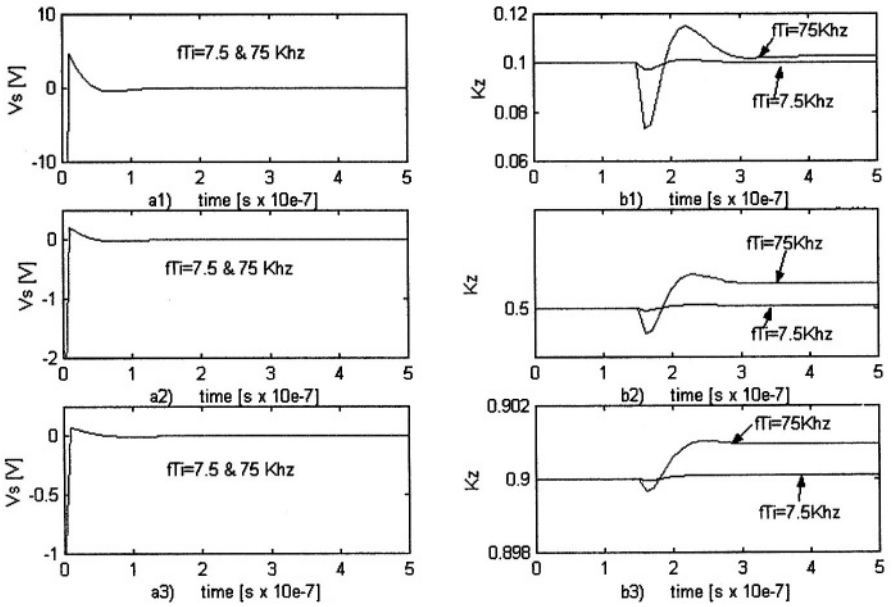


Figure 5.24. a1), a2), a3) Vs impulse response for $Kz_{(t=0)}$=0.1, 0.5 and 0.9 respectively. b1), b2), b3) Kz impulse response for $Kz_{(t=0)}$=0.1, 0.5 and 0.9 respectively. Two values of the integrator transition frequency were used for each simulation: fTi=7.5KHz and 75KHz.

Concerning the latter remark above, the fact that the incremental step of Kz depends on wTi does not represent a drawback from the stability point of view. Nevertheless, the inappropriate choice of the integrator transition frequency makes difficult the convergence of the parameter Kz. Indeed, as in most adaptive system, if the incremental steps of the adapting parameters ("learning rate") are not properly settled such parameters could not converge asymptotically [WiSt85] [Hayk91]. In such a case, the parameters final values are not stable. They oscillate with increasing amplitude as a larger updating step is used. In our case, the learning rate of Kz is given by the bandwidth of the integrator: the faster the integration, the faster the adaptation but the less accurate the final value of Kz. This effect is

illustrated in Figure 5.25 a) and b), which show the evolution of Kz along the adapting period for several values of wTi. Notice that as long as wTi increases the time required for adaptation is shortened but the accuracy of the final value of Kz is impoverished.

However, by no way the oscillations shown in Figure 5.25 b) can be interpreted as a stability problem in the frame of the usual meaning of stability. Upon the simulations presented in Figure 5.24 we can state that there are no poles with positive real part in the closed loop transfer function of the system. The periodicity of the oscillation shown in Figure 5.25 b) is due to the fact that the digital input stream used for such simulation is a repetitive sequence of 64 bits of length. If a true random binary stream with an infinite sequence length is used, the periodicity in Figure 5.25 b) will disappear whereas the fluctuations of Kz around its steady state value will also be random. In summary: the lack of convergence of Kz as wTi increases is a problem associated with the accuracy of the adaptation process rather with the stability of the system (from the usual sense of stability point of view).

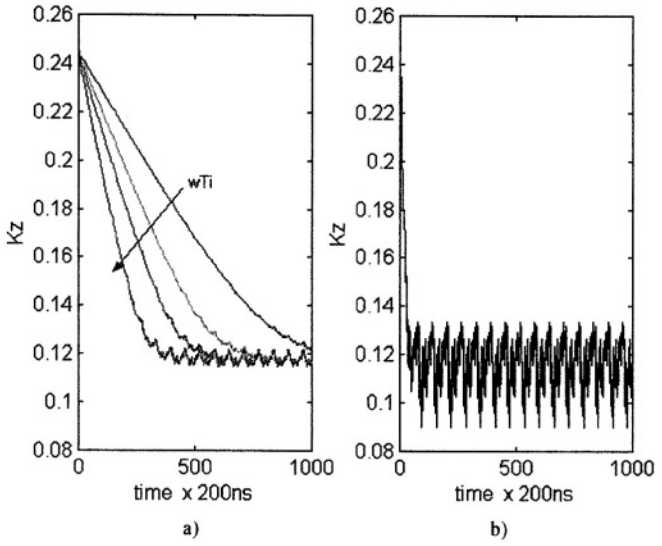


Figure 5.25. Kz evolution during the adaptive equalization of a CAT5 UTP cable of 360m of length for different fTi values: a) 2.5, 3.75, 5 and 7.5KHz. b) 75KHz. The transmission rate was set to 5Mb/s.

One can improve the accuracy of the convergence by reducing the output swing of the Fuzzy Controller. In this way, the error signal provided by the controller to the integrator would be attenuated. As a result, the integration would be carried out slowly. In our case, in order to perform a realistic

simulation as much as possible, we have employed the actual measured swing of the controller (2V@Vdd=5V). However, if the values of the loaded consequent singletons were redefined, we would be able to reduce the output swing of the controller up to a tenth of the actual measured swing (0.2V@Vdd=5V). In this case, the same control surface would result down scaled onto a much reduced output range. If a further reduced output swing is still needed, one should increase the resolution of the consequent singletons above the 5 bits that has been defined in our implementation. For this reason, accounting with wTi as a design parameter to control the adaptation rate of Kz represents an advantage: just consider that the resolution needed for the Fuzzy Controller can be relaxed if the value of wTi is appropriately set.

5.4.2 Channel Equalization for Different Cable Lengths

In order to prove the self-adapting capabilities of the system to different cable settings, three lengths of cable were adopted for the transmission of a bipolar NRZ random sequence at 5Mb/s.

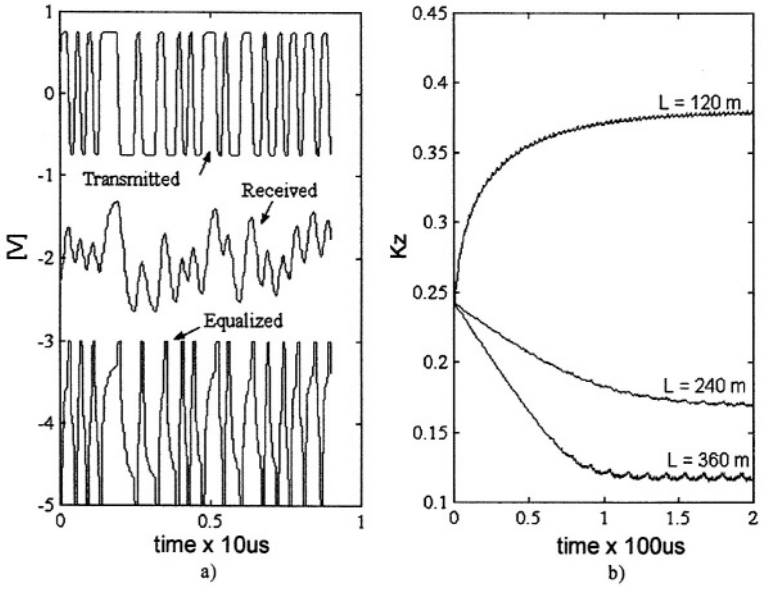


Figure 5.26. a) Transmitted, received and equalized signals for L=360m. b) Kz evolution for several cable length: L=120, 240 and 360m. In all cases fs=5Mb/s. From [DuVe01], © 2001 IEEE.

Figure 5.26 a) illustrates the transmitted, received and equalized signal after adaptation is achieved for the case with the most severe ISI that

corresponds to L=360m. For the same case, Figure 5.27 a) and b) shows the Eye Patterns before and after the equalizer, respectively. For these simulations, the transmission rate has been chosen according to the speed of the controller. Since the maximum baud rate tolerable by our prototype is relatively low (i.e. ~5Mb/s) and intending to produce a considerable ISI at such a low frequency, the cable length L was set to 120, 240 and 360m in each respective simulation.

Figure 5.26 b) shows the evolution of Kz during adaptation for the three different cable settings. Starting with the same initial value, it is worth noticing the asymptotic convergence of the parameter Kz in both directions according with each case. This is accomplished in less than 150μs that represents the total time required for the transmission of 750 bits at 5Mb/s.

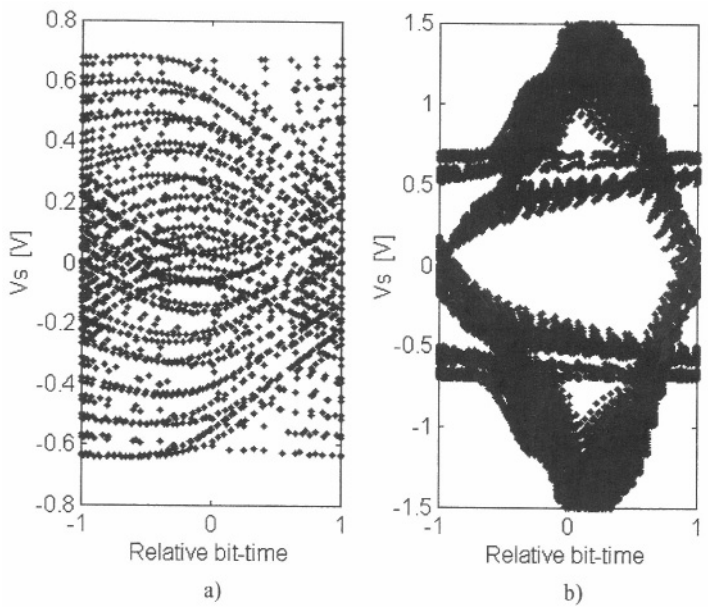


Figure 5.27. Eye diagrams for a transmission rate of 5Mb/s and L=360m: a) Before the equalizer. b) After the equalizer.

5.4.3 Adaptation Performance for Noisy Channels

Additive gaussian noise with zero mean and 0.03 of variance has been added to the signal transmitted at 5Mb/s through the 360m length cable. All others set-up parameters remain unchanged as in the former simulations. Figure 5.28 a) illustrates the convergence of Kz, which is attained in almost the same time than in the case of a noiseless channel. Small amplitude oscillations can be observed around the steady state value.

In order to perform a comparison, the same channel under the same conditions was attempted to be equalized using the "decision-directed" learning rule explained in section 5.1.2. The evolution of Kz is shown in Figure 5.28 b) wherein no convergence is observed. As it was previously commented, besides the simplicity, the high sensibility to noise and to the shape of the signal represents the main drawback of the "decision-directed" learning method. Simulations, not shown herein, confirm that even without noise the adapting parameter Kz converges but with considerable fluctuations around the asymptotic value. This could be a sign that some bad decisions are taken by the adapting mechanism, probably due to non-ideal shape of the signal being equalized. Moreover, under the same conditions, the time demanded to attain the convergence doubles the one demanded in our approach. This is probably due to the fact that only one decision per bit is taken in contrast to our method where Kz is updated continuously around the whole bit cycle.

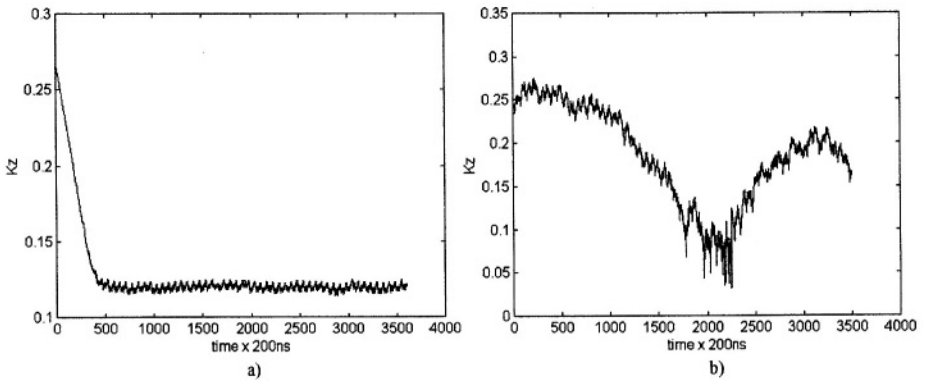


Figure 5.28. Kz evolution in a noisy channel: a) Using our approach based on the Eye Pattern. b) Using the "decision-directed" learning rule presented in [Wist85]. In both cases the noise is additive and gaussian, with 0 mean and 0.03 of variance.

Another simulation in a noisy environment by using the method presented in [Mage94] has been performed. The evolution of Kz is illustrated in Figure 5.29.

The cable length, the transmission rate, the integrator transition frequency and the noise properties were identically set as in the previous simulations. Notice from Figure 5.29 that despite the convergence, the steady state of Kz is more affected by the noise than in our case.

Finally, let us remark that in our case the system works in continuous time and we can account on the integrity of the signal concentrated in the Eye Pattern rather than on individual samples.

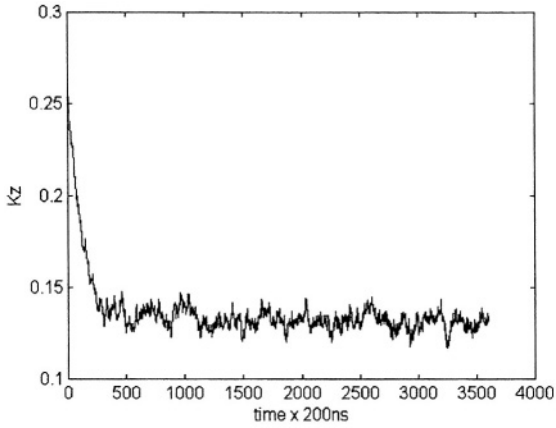


Figure 5.29. Kz evolution for a noisy channel using the approach of [Mage94].

5.5 Conclusions

This chapter intends to demonstrate the potential application of Fuzzy Logic in other domains beyond the classical Process Control. The Signal Processing field has been chosen for testing the performance of this soft-computing technique applied to a real-time, high-speed and low-power application.

A methodology to perform self-adaptive equalization of digital signal carrying ISI distortion has been derived from the general set-up for time-domain signal analysis. After a comparison of the actual eye diagram with an ideal one encoded in a Fuzzy Inference System, amplitude equalization is achieved by controlling in closed loop the adapting parameter of the equalizing filter. To get primary insight on the system feasibility, the design, fabrication and test of an embeddable Fuzzy Controller and an analog adaptive equalizer have been addressed.

According to the power consumption and the area occupied, the implemented analog Fuzzy Controller suits optimal as embedded subsystem in well-targeted applications. This is the merit of the compactness of both the fuzzy partition circuits and the defuzzifying scheme employed. A RMSE of 4.5% between target and measured surfaces has been reached. For the case of self-adaptive equalization, the achieved precision fulfills the requirements for an asymptotic and smooth convergence of the adapting parameter Kz of the equalizer. This is possible if the choice of the bandwidth of the integrator wTi is properly made.

In order to implement the analog gm-C adaptive equalizer, a novel full electrically-tunable triode transconductor has been designed [DuVe01]. Upon ideal local matching conditions, the transconductor features absolute

independence on technological parameters: the transconductance can be controlled by the ratio between a reference current and a reference voltage. This permits to disregard the long-distance matching conditions between the technological parameters of the transconductors building the filter. Simulations carried out on a gm-C integrator predict also small phase errors and linear tuning.

Automatic cable equalization has been simulated for different cable lengths. Simulations confirm that even for noisy channels adaptation is achieved. Noise, ringing and other kind of distortions that make the actual pulses to look different from the ideal ones are filtered out by means of a null-action region foreseen at the non-linear control surface. This could be straightforwardly defined by taking advantage of the ability of Fuzzy Logic for dealing with non-linear functions.

Scanning signals over time permits to identify signals patterns, like in any oscilloscope. After a comparison against an encoded reference, the result can be used for several purposes. Therefore, the applicability of this technique exceeds the topic of channel equalization. From an explicit 2D signal representation, Fuzzy Logic allows to infer meaningful assertions that can be used for adaptation, detection, testing, etc.

As an open path, we propose to exploit the time-domain signal analysis technique based on Fuzzy Logic for other kind of analog applications. For instance, in [MoPi94] on-chip analog filter tuning is addressed by using Fuzzy Logic too. However, in the mentioned reference a kind of discrete frequency-domain signal analysis is performed by measuring the filter response on a finite set of frequencies. The control strategy intends to fit the curve of the actual frequency response of the filter within a window specification. Certainly, the same task could be accomplished in time-domain by means of a built-in "oscilloscope". According to the set-up in Figure 5.1, it is possible to encode into the Fuzzy Controller the pattern corresponding to the response of a well-tuned filter to a reference square wave, for instance. Thus, tuning can be achieved by comparing against the encoded pattern the actual time response of the filter to the same reference signal. In this way, the tuning system may result more simplified than in [MoPi94], which needs the on-chip generation of several sinusoidal testing signals at different frequencies.

Time-domain signal analysis could also be employed in the field of Analog Integrated Circuit Testing, which becomes nowadays a challenging domain of research. For instance, it is possible to implement analog BIST units (Built-In Self-Test units) supporting an "on-chip oscilloscope". This should be capable to analyze in time-domain the signal delivered by the analog circuit under test and verify its quality.

Chapter 6

GENERAL CONCLUSIONS

All along this book, we addressed the design, implementation and test of programmable analog Fuzzy Controllers using standard CMOS technologies for applications of medium to high speed. From the analysis perspective of the different fuzzy algorithms proposed in the literature, the implementation of Takagi-Sugeno's controllers has been found to be highly feasible considering the good trade-off between simplicity and accuracy that these models feature. The book has been organized in two fundamental stages: a comprehensive study of the basic blocks implementing a Fuzzy Controller and the design and test of programmable architectures. Moreover, in order to demonstrate the usefulness of approximate reasoning algorithms in areas beyond the classical Control, a preliminary study of a real-time embedded Fuzzy Logic application in the Signal Processing domain has been undertaken. Let us summarize the main achievements and discuss future improvements and open challenges.

In Chapter 3, an analog framework for hardware implementation of Fuzzy Logic Controllers has been established. The study encompasses the analysis of the elementary fuzzy blocks: Fuzzifiers, Inference Operators (T-Norms and T-CoNorms) and Defuzzifiers. Major topics regarding circuit accuracy, interfacing and complexity have been focused. Some novel circuits have been presented while others were optimized in view of an improved behavior. After having characterized most circuit non-idealities, an estimation of the global accuracy attainable for a given configuration has been carried out by means of an error propagation analysis technique that allows identifying critical sources of imperfections.

In Chapter 4, the design and test of programmable analog Fuzzy Logic Controllers has been focused by making use of the basic analog processing blocks previously studied. Issues concerning controller architectures and the programming strategy have been specially focused towards the

implementation of front-end Fuzzy Processors. The objective is to feature enough flexibility, acceptable delays and power consumption that make them useful for a broad domain of applications. Sharing functional operators, as well as optimizing the interfaces between blocks by avoiding intermediate signal converters, has played an important role during the design step. These general guidelines led to architectures with improved modularity. However, technology limitations favored mixed-signal design, which holds the advantages of analog circuits for massive and fast computation together with the feasibility of digital circuits for storage. In this way, a trade-off between accuracy and complexity has been chosen to configure the controllers by means of a discrete set of analog parameters.

Two programmable Fuzzy Controllers have been designed and tested. Compared to reported works, they achieve good results for low to medium-complexity Fuzzy Systems, like those implemented by means of analog circuits. On one hand, the maximum precision is limited by the discrete set of analog values allowed for the parameters. In our circuits, the RMSE errors range from 2.7% to 4.7% and the speed from 0.9 to 5.26 MFLIPs. This may be considered an acceptable result for several kinds of applications ranging from Control to Signal Processing [FrMa98] [BaDi00] [OeGr96] [GoAl01] [MoPi94] [DuVe01] [NaVi00]. On the other hand, the tolerances between samples due to transistor mismatch become a weakness in our implementations. However, if larger transistors are used, the mismatch errors decrease. Unfortunately, in order to keep the total die of the chip reasonably small (with small routing stray capacitances) we were forced to use relatively small size transistors for building the demonstrators. Nevertheless, it is well known that shorter channel technologies present better transistor matching properties. Indeed, for the same transistor sizes, a rough estimation allows us to foresee a substantial mismatch improvement if the controllers were implemented in such technologies. Certainly, working with a short-channel technology also leads to improve the speed of the controller because of the reduced stray capacitances that may be expected. Therefore, future improvements should be endeavored by implementing our circuits in a more modern CMOS process for applications demanding increased accuracy and/or speed.

In Chapter 5, the potential use of Fuzzy Logic has been demonstrated for Signal Processing, in an application where the design of a dedicated analog Fuzzy Controller is justified. A knowledge-based methodology to perform signal analysis in the time domain has been discussed. The general idea consists in building a "kind" of built-in oscilloscope transforming the signal into a figure in the Universe of Discourse [Signal-value , Time] that can be "examined" through a Fuzzy Inference System. In this way, the Decision Making unit of the latter could be settled to infer meaningful assertions that

could be used for adaptation, detection, testing, etc. Self-adaptive equalization of digital channels has been chosen as the target application. Amplitude equalization is achieved by tuning an analog adaptive equalizing filter after the comparison of the actual Eye Pattern against an ideal one, which is encoded in the Fuzzy Controller. To get insight about the feasibility of the system and its behavior, a preliminary prototype consisting of the equalizing filter and the controller has been fabricated and tested. Some measured features extracted from this prototype have been used for modeling the equalizer and the controller in a system-level simulation of the whole equalizing system.

Scanning signals over time permits to identify signal patterns, like in any oscilloscope. Therefore, the applicability of this technique should be further exploited for other kinds of analog applications. For instance, we have discussed at the end of Chapter 5 the possibility of performing on-chip analog filter tuning in time domain. We have also proposed this technique for building analog BIST units (Built-In Self-Test units) supporting an "on-chip oscilloscope" based on Fuzzy Logic. In this way, the knowledge-based BIST units can perform a time-domain analysis of the signal delivered by the analog circuit under test in order to verify its quality.

Finally, through the application discussed in the previous chapter, we hope that we succeeded in demonstrating the need of non-linear transfer functions for optimally performing special tasks in an electronic system. However, in the Analog Design field, no great efforts has been put in developing straightforward techniques for the synthesis of non-linear circuits. In earlier years, some non-linear synthesis solutions have been frequently met by the use of translinear networks [ToLi90], for instance. On one hand, the implementation of complex functions with translinear networks becomes a cumbersome task, specially when MIMO systems (Multiple-Input Multiple-Output) are being considered. Moreover, the structure of a translinear network does not follow a general arrangement criterion [Wieg93] and may be quite different for two functions to synthesize. On the other hand, translinear relationships are usually dependent on the physical characteristics of devices. Hence, a translinear network is not easily transportable from one technology to another. In contrast, the fuzzy approach to the non-linear synthesis is performed from a higher abstraction level. In any Fuzzy System, without concerning the kind and the complexity of the function to approximate, one can find only three basic components: Fuzzifiers, Inference Operators and Defuzzifiers. These operators are always arranged in an invariant network structure to compute the *if-then* rules. As technologies moves on and newer devices appear we only need to care about readapting those three basic operators but never the entire network, whose structure remains essentially the same. Moreover, we have demonstrated

through this book the possibility of building general-purpose programmable Fuzzy Networks useful for a wide range of applications. These properties permit us, to some extent, to consider Fuzzy Logic as a technology-independent systematic approach for the analog synthesis of non-linear functions. Therefore, it is our belief that this soft-computing technique must take an important place in the toolbox of analog designers.

Bibliography

[AlKr92] Von Altrock et al., "Advanced fuzzy logic control of a model car in extreme situations", *Fuzzy Sets and Systems,* Vol. 48, N° 1, pp. 41-52, 1992.

[Anne98] Annema A., "Low-Power Bandgap References Featuring DTMOSTs", *on proc. of ESSCIRC'98,* pp. 116-119, The Netherlands, September 1998.

[BaCa94] Baschirotto A., Rezzi F. and Castello R., "Low-Voltage Balanced Transconductor with High Input Common-Mode Rejection", *Electronics Letters,* Vol. 30, N° 20, pp. 1669-1671, 1994.

[BaDi00] Ramírez-Barajas J., Dieck-Assad G. and Soto R., "A Fuzzy Logic Based AGC Algorithm for a Radio Communication System", *in Proc .of the IEEE International Conference on Fuzzy Systems,* pp. 997-980, Alaska, 2000.

[BaHu97] Baturone I., Sánchez-Solano S., Barriga A. and Huertas J., "Implementation of CMOS Fuzzy Controllers as Mixed-Signal Integrated Circuits", *IEEE Transactions on Fuzzy Systems,* vol. 5, N° 1, pp. 1-19, February 1998.

[BaHu98] Baturone I., Sánchez-Solano S. and Huertas J., "Towards the IC Implementation of Adaptive Fuzzy Systems", *IEICE Trans. on Fundamentals of Electronics. Communications and Computer Sciences,* Vol. E81-A, N° 9, pp. 1877-1885, Sept. 1998.

[Bake96] Baker A. J., "An Adaptive Cable Equalizer for Serial Digital Video Rates to 400Mb/s", *IEEE ISSCC96 Digest of Technical Papers,* pp. 174-175, 1996.

[BeKh92] Berenji H. & Khedkar P., "Learning and tuning Fuzzy Logic Controllers through reinforcements", *IEEE Trans. on Neural Networks,* Vol.3, N° 5, pp. 724-740, Sept. 1992.

[Bern88] Bernard J., "Use of a rule-based system for process control", IEEE Control Systems Magazine, Vol. 8, N° 5, pp. 3-13, 1988.

[BoTs97] Bouras S. and Tsividis Y., "Center-of-Gravity Defuzzification without Multiplication", *IEICE Transaction Fundamentals,* Vol. E80 A, N° 4, pp. 769-770, April 1997.

[BoTs98] Bouras A. et al., "Mixed Analog-Digital Fuzzy Logic Controller with Continuous-Amplitude Fuzzy Inferences and Defuzzification", *IEEE Transactions on Fuzzy Systems,* vol. 6, N° 2, pp. 205-215, May 1998.

[BrHa94] Brown M. and Harris C., "Neurofuzzy Adaptive Modeling and Control", *Prentice Hall,* 1994.

[BrHu95] Brown J., Hurst P. and Der L., "A 35Mb/s Mixed-Signal Decision-Feedback Equalizer for Disk-Drive Applications", *in Proc. of the IEEE 1995 Custom Integrated Circuit Conference,* pp. 25.6.1-26.6.4, USA, 1995.

[BuGe90] Bult K., Geelen G., "A Fast-Settling CMOS Operational Amplifier for SC Circuits with 90-dB DC Gain", *IEEE JSSC,* Vol. 25, No 6, pp. 1379-1384, 1990.

[BuWa87] Bult K., Wallinga H., "A Class of Analog CMOS Circuits Based on the Square Law Characteristic of a MOS Transistor in Saturation", *IEEE Journal on Solid State Circuits,* Vol. 22, N°3, pp. 357-365, 1987.

[CaRa00] Caravajal R., Ramirez-Angulo J. and Tombs J., "High-Speed High-Precision Voltage-mode M1N/MAX Circuits in CMOS Technology", *Proceedings of The 2000 IEEE International Symposium on Circuits and Systems ISCAS'2000,* Vol. V, pp. 13-16, Geneva, Switzerland, May 2000.

[ChCh91] Chiu S. et al., "Fuzzy logic for control of roll and moment for a flexible wing aircraft", *IEEE Control System Magazine,* Vol. 11, N° 4, pp. 42-48, June 1991.

[ChCh95] Chand S. and Chiu S., "Special Issue on Fuzzy Logic with Engineering Applications", *IEEE Proceedings,* Vol. 83, N° 3, pp. 343-344, March 1995.

[ChJo99] Chen P. and Jou J., "A Fast-Search Motion Estimation Method and its VLSI Architecture", *IEEE Trans. on Circuits and Systems-II Analog and Digital Processing,* Vol. 46, N°9, pp. 1233-1240, Sept. 1999.

[Chri93] Christian P., "Fuzzy logic control of a hypervelocity interceptor", in *2nd Int. Conference on Fuzzy Systems,* pp. 877-882, 1993.

[ClTu92] Cleland J. et al., "Fuzzy logic control of ac induction motors" in *1st Int. Conference on Fuzzy Systems,* pp. 843-850, 1992.

[CoCr99] Conti M. et al., "A Current-Mode Circuit for Fuzzy Partition Membership Functions", in *Proceedings of The 1999 IEEE International Symposium on Circuits and Systems ISCAS'99,* Vol. V, pp. 391-394, Orlando, FL., USA, May 1999.

[CoGl95] Costa A., De Gloria A., Faraboshi P., Pagni A. and Rizzotto G., "Hardware Solutions for Fuzzy Control", *Proceedings of the IEEE,* Vol. 83, N° 3, pp. 422-434, March 1995.

[CrLi98] Criscione M., Lionetto A., Nunnari G. and Occhipinti L., "Embedded Fuzzy Control on Monolithic DC/DC Converters", in *Proc. of the 1998 IEEE International Symposium on Circuits and Systems ISCAS'98,* Monterrey, CA., USA, May 1998.

[DaGi89] Daley S. and Gill K., "Comparison of a fuzzy logic controller with P+D control law", *Trans. ASME,* Vol. 111, pp. 128-137, June 1989.

[DeDu00] De Lima J. and Dualibe C., "On Designing Linearly-Tunable Ultra-Low Voltage CMOS gm-C Filters", in *Proceedings of The 2000 IEEE International Symposium on Circuits and Systems ISCAS'2000,* Vol. II, pp. 140-143, Geneva, Switzerland, May 2000.

[DeDu00-b] De Lima J. and Dualibe C., "A Linearly-Tunable Low-Voltage CMOS Transconductor with Improved Common-Mode Stability and its Application to gm-C Filters", *IEEE Trans. on Circuits and Systems-II Analog and Digital Processing,* Vol. 48, N°7, pp. 649-660, June 2001.

[DeDu99] De Lima J. and Dualibe C., "A Tunable Triode-Mosfet Transconductor and its Application to gm-C Filters", in *Proceedings of The 1999 IEEE International Symposium on Circuits and Systems ISCAS'99,* Vol. II, pp. 640-643, Orlando, FL., USA, May 1999.

[DoDu00] Donckers N., Dualibe C. and Verleysen M. "A Current-Mode CMOS Loser-Take-all with Minimum Function for Neural Computations", in *Proceedings of The 2000 IEEE International Symposium on Circuits and Systems ISCAS'2000,* Vol. I, pp. 415-418, Geneva, Switzerland, May 2000.

[DoDu99] Donckers N., Dualibe C. and Verleysen M. "Design of Complementary Low-Power CMOS Architectures for Loser-take-all and Winner-take-all", in *Proceedings of The Seventh International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems, MicroNeuro'99,* pp. 415-418, Granada, Spain, pp. 360-365, April 1999.

[DrHe93] Driankov D., Helledoorn H. and Reinfrank M., "An introduction to Fuzzy Control", *Springer,* 1993.

[Dual94] Dualibe C., "Logique Floue: de la Théorie au Matériel", *TFE Licence en Sciences Appliquées,* Université catholique de Louvain, Louvain-la-Neuve, Belgium, Sept. 1994 .

[Dual96] Dualibe C., "Electrically Tunable Current-Mode Analog Fuzzy Membership Functions", *2nd IBERCHIP Workshop,* pp. 102-111, Sao Pablo, Brazil, February 1996.

[DuCa93] Duque-Carrillo J., "Control of the common-mode in CMOS Continuous-Time Full Differential Signal Processing", *Analog Integrated Circuits and Signal Processing,* N° 4, pp. 131-140, Kluwer Academic Publishers, September 1993.

[DuVe00] Dualibe C., Jespers P. and Verleysen M., "A 5.26 Mflips Programmable Analogue Fuzzy Logic Controller in a Standard CMOS 2.4µ Technology", in *Proceedings of The 2000 IEEE International Symposium on Circuits and Systems ISCAS'2000,* Vol. V, pp. 377-380, Geneva, Switzerland, May 2000.

[DuVe01] Dualibe C., Jespers P. and Verleysen M., "Embedded Fuzzy Control for Automatic Channel Equalization After Digital transmissions", in *Proceedings of The 2001 IEEE International Symposium on Circuits and Systems ISCAS'2001,* Vol. III, pp. 173-176, Sidney, Australia, May 2001.

[DuVe01-b] Dualibe C., Jespers P. and Verleysen M., "On Designing Mixed-Signal Programmable Fuzzy Logic Controllers as Embedded Subsystems in Standard CMOS Technologies", *14th Symposium on Integrated Circuits and System Design SBCCI'2001,* pp. 194-200, Pirenopolis, Brasil, 10-15 September, 2001.

[DuVe98] Dualibe C., Verleysen M. and Jespers P., "Two-Quadrant CMOS Analogue Divider", *IEE Electronics Letters,* Vol. 34, N° 12, pp. 1164-1165, June 1998.

[Egge98] Eggermont J. P., "Study and Realization of SOI Operational Amplifiers for High-Temperature or High-Frequency Applications", *Ph.D. Dissertation,* Université catholique de Louvain, Belgium, June 1998.

[EiKü96] Eichfeld H., Künemund T. and Menke M., "A 12b General-Purpose Fuzzy Logic Controller Chip", *IEEE Transactions on Fuzzy Systems,* Vol. 4, N° 4, pp. 460-475, November 1996.

[Espe94] Espejo S., "Cellular Neural Networks: Modeling and Monolithic Design", *Ph.D. Dissertation,* Sevilla, Spain, February 1994.

[FaMa94] 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.

[FlVi97] Flandre D., Viviani A., Eggermont J.P., Gentinne B. and Jespers P., "Improved Synthesis of Gain-Boosted Regulated-Cascode CMOS Stages Using Symbolic Analysis and gm/ID Methodology", *IEEE Journal of Solid State Circuits,* Vol. 32, N° 7, pp. 1006-1012, July 1997.

[FrMa98] 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.

[GeBa94] Gevers M. and Bastin G., "Identification des systèmes", *Notes de cours,* Université catholique de Louvain, Louvain-la-Neuve, 1994.

[GoAl01] Gomariz S. et al., "Piecewise PWM-Sliding Global Control of a Boost Switching Regulator by Means of First-Order Takagi-Sugeno Fuzzy Control", Proceedings *of The*

2001 *IEEE International Symposium on Circuits and Systems ISCAS'2001,* Sidney, Australia, May 2001.

[Godj97] Godjevac J., "Neuro-Fuzzy Controllers Design and Application", *Presses Polytechniques et Universitaires Romandes,* Lausanne, Switzerland, 1997.

[GrTe86] Gregorian R. and Temes G., "Analog MOS Integrated Circuits for Signal Processing", *John Willey & Sons, Inc.,* 1986.

[GuPe96] Guo S., Peters L. and Surmann H., "Design and Application of an Analog Fuzzy Logic Controller", *IEEE Transaction on Fuzzy Systems,* Vol. 4, pp. 429-438, Nov. 1996.

[HaRe89] Harris J. & Read A., "Knowledge based fuzzy motion control of autonomous vehicle: Artificial Intelligence in real time control", *NY: Pergamon,* 1989, pp. 139-144.

[Hayk91] Haykin S., "Adaptive Filter Theory", 2nd edition, Chapter I and XIII, *Prentice Hall Information and System Sciences Series,* 1991.

[HiSu92] Hideyuki T., Suzuki N., Koda T. and Kojima Y., "Neural Networks Designed on Approximate Reasoning Architecture and their Applications", *IEEE Transactions on Neural Networks,* Vol. 3, N°5, pp. 752-760, September 1992.

[HoHa96] Hollstein T. et al., "Computer-Aided Design of Fuzzy Systems Based on Generic VHDL Specifications", *IEEE Trans. on Fuzzy Systems,* Vol. 4, N° 4, November 1996.

[HoOs82] Holmblad P. and Ostergaard J., "Fuzzy Information and Decision Processes, ch. Control of a Cement Kiln by Fuzzy Logic", pp. 389-399, *North-Holland,* 1982.

[HuBa96] Huertas, J.L., Sánchez - Solano, S., Barriga, A., Baturone, I., "Integrated Circuit Implementation of Fuzzy Controllers", *IEEE Journal of Solid State Circuits,* Vol. 31, N°7, pp. 1051-1058, July 1996.

[HuKe95] 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.

[Info94] INFORM Inc., Fuzzy-166 Processor: The chip for Flexible High Performance Fuzzy solutions. *INFORM Gmbh,* Aachen, Germany, 1994.

[ItWa92] Ito M. and Wakayama A., "Detection of Earthquake Precursors by Fuzzy Theoretical Methods", in *2nd Int. conference on Fuzzy Logic and Neural Networks,* Izuka, 1992, pp. 389-392.

[JaGu95] Jang R. and Gulley N., "Fuzzy Logic TOOLBOX for use with MATLAB", *The MATHWORKS, Inc.,* Natick, Mass. 01760-1500, January 1995.

[Jang92] Jang R., "Self-Learning Fuzzy Controllers Based on Temporal Back Propagation", *IEEE Transactions on Neural Networks,* Vol. 3, N° 5, pp. 714-723, September 1992.

[JaSu95] Jang R. and Sun C., "Neuro-Fuzzy Model and Control", *Proceedings of the IEEE,* Vol. 83, N° 1, pp. 378-406, March 1995.

[JoMa97] Johns D. and Martin K., "Analog Integrated Circuit Design", *John Wiley and Sons, Inc.,* 1997, pp. 287-291.

[KaLa98] Kandel A. and Langholz G., "Fuzzy Hardware, Architectures and Applications", *Kluwer Academic Publishers,* 1998.

[Kard92] Kardontchik J., "Introduction to the Design of Transconductor-Capacitor Filters", *Kluwer Academic Publishers,* Boston, USA, 1992.

[KeSc93] Kettner T., Schumacher K. and Goser K., "Realization of a Monolithic Analog Fuzzy Logic Controller", in *Proceedings ESSCIRC'93,* pp. 66-69, September 1993.

[KhGr84] Khorramabadi H. and Gray P., "High-Frequency CMOS Continuous-Time Filters", *IEEE Journal of Solid State Circuits,* Vol. SC-19, N°6, pp. 939-948, Dec. 1984.

[KiSt96] Kinget P. and Steyaert M., "Impact of transistor mismatch on the speed-accuracy-power trade-off of analog CMOS circuits", in *proceedings of the IEEE Custom Integrated Circuits Conference,* pp. 333-335, 1996.

[Kosk92] Kosko B., "Neural Networks and Fuzzy Systems", Chapter 1, pp. 24-26, *Prentice-Hall, Inc.,* Englewood Cliffs, New Jersey 07632, USA, 1992.

[KrJo88] Krummenacher F. and Joehl N., "A 4-MHz CMOS Continuous-Time Filter with On-Chip Automatic Tuning", *IEEE JSSC,* Vol.23, N° 3, pp. 750-758, June 1988.

[LaGr93] Laber C. and Gray P., "A 20-Mhz Sixth-Order BiCMOS Parasitic-Insensitive Continuous-Time Filter and Second-Order Equalizer Optimized for Disk-Drive Read Channels", *IEEE Journal of Solid State Circuits,* Vol. 28, N° 4, pp. 462-470, April 1993.

[Land93] Landolt O., "Efficient Analog CMOS Implementation of Fuzzy Rules by Direct Synthesis of Multidimentional Fuzzy Subspaces", *2nd International Conference on Fuzzy Systems (FUZZ-IEEE 1993),* San Francisco CA, USA, pp. 453-458, March 1993.

[Land96] Landolt O., "Low-power analog fuzzy implementation based on a linear MOS transistor network", *Proceedings of MicroNeuro '96,* 1996, pp. 86-93.

[LaPa93] 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.

[LaRy89] Lazzaro J., Ryckebusch S., Mahowald M.A. and Mead C., "Winner-take-all networks of O(n) complexity", in *Advances in Neural Information Processing Systems,* Vol. 1, D. S. Touretzky, Ed. Los Altos, CA: Morgan Kaufmann, 1989, pp. 703-711.

[LaSa94] Laker K. and Sansen W., "Design of Analog Integrated Circuits and Systems", *Mc Graw-Hill, Inc.,* New York, USA, 1994.

[LeLa96] Lee S.S. and Laber C., "A 3.5 in 230 Mbytes Read-Channel Chip Set for Magneto-Optical Disk Drives", *IEEE Trans. on Very Large Scale Integration Systems,* Vol. 4, N° 4, pp. 455-463, December 1996.

[LeLe93] Lee B. et al., "Programmable Facsimile Image Processor Including Fuzzy-Based Decision", in *Proc. of the IEEE 1993 Custom Integrated Circuit Conference,* pp. 15.6.1-15.6.4, USA, 1993.

[LePa94] Lemaitre L., Patyra M., and Mlynek D., "Analysis and Design of CMOS Fuzzy Logic Controller in Current Mode", *IEEE Journal of Solid State Circuits,* Vol. 29, N°3, pp. 1051-1058, March 1994.

[LiCh95] Liu S. and Chang C., "CMOS Analogue Divider and Four-Quadrant Multiplier Using Pool Circuits", *IEEE JSSC,* Vol. 30, N° 9, pp. 1025-1029, September 1995.

[LiHu93] Liu S., Hwang S. and Tsay J., "CCII-Based Fuzzy Membership Function and MAX/MIN Circuits", *IEE Electronics Letters,* Vol. 29, N° 1, pp. 116-118, January 1993.

[LiHu94] Liu D., Huang Y., and Wu Y., "Modular Current-Mode Defuzzification Circuit for Fuzzy Logic Controllers", *Electronics Letters,* Vol. 30, pp. 1287-1288, August 1994.

[LiMe00] Liang Q. and Mendel J., "Decision Feedback Equalizer for Nonlinear Time-Varying Channels Using Type-2 Fuzzy Adaptive Filters", in *Proc. of the IEEE International Conference on Fuzzy Systems,* pp. 883-888, Alaska, 2000.

[MaCo97] Marshall G. and Collins S., "Fuzzy Logic Architecture Using Subthreshold Analogue Floating-Gate Devices", *IEEE Transactions on Fuzzy Systems,* Vol. 5, N° 1, pp. 32-43, February 1997.

[MaDa95] Mancuso M., D' Alto V., De Luca R., Poluzzi R. and Rizzotto G., "Fuzzy Logic Based Image Processing in IQTV Environment", *IEEE Trans. on Consumer Electronics,* Vol. 41, N° 3, pp. 917-923, 1995.

[MaFr96] Manaresi N. et al., "A silicon compiler of analog fuzzy controllers: from behavioral specifications to layout", *IEEE Trans. on Fuzzy Systems,* Vol. 4, N° 4, pp. 418-428, 1996.

[Mage94] Mager K., "Equalizer Adaptation with a Geometrical Criterion and Fuzzy Logic", *IEEE Transaction on Consumer Electronics,* Vol. 40, N° 3, pp. 372-376, August 1994.

[MaHo94] Masa P., Hoen K. and Wallinga H., "A High-Speed Analog Neural Processor", *IEEE Micro,* pp. 40-50, June 1994.

[MaNe89] Mallya S. and Nevin J., "Design Procedures for a Fully Differential Folded-Cascode CMOS Operational Amplifier", *IEEE JSSC,* Vol. 24, N° 6, pp. 1737-1740, Dec. 1989.

[MaRo98] Manaresi N., Rovatti R., Franchi E. and Baccarani G., "A Current-Mode Piecewise-Linear Function Approximation Circuit Based on Fuzzy-Logic", in *Proceedings of The 1998 IEEE International Symposium on Circuits and Systems ISCAS'98,* Monterrey, CA, USA, May 1998.

[MeDa87] Mee C. and Daniel E., "Magnetic Recording, Vol. II: Computer Data Storage. Chapter 4: Data Storage on Tape", *McGraw-Hill, Inc.,* 1987.

[Mend95] Mendel J., "Fuzzy Logic for Engineering: A tutorial", *Proceedings of the IEEE,* Vol. 83, N° 3, pp. 345-377, March 1995.

[MoPi94] Moreira-Tamayo O., Pineda J. and Sanchez-Sinencio E., "Filter Tuning System Using Fuzzy Logic", *Electronics Letters,* Vol. 30, N° 30, pp. 846-847.

[Moto92] Motorola Inc., *Motorola Semiconductor* Technical Data, 1992.

[NaVi00] Navas-Gonzalez R., Vidal Verdu F. and Rodriguez-Vazquez A., "A Mixed-Signal Fuzzy Controller and its Application to Soft Start of DC Motors", *Proc. of the 2000 IEEE International Fuzzy Systems Conference,* pp. 128-133, Alaska, 2000.

[Neye97] De Neyer M., "Contributions to Fuzzy Controller Analysis and Design", *Ph.D. Dissertation,* Université catholique de Louvain, Belgium, Mars 1997.

[OeGr96] Oehm J., Grafe M., Kettner T. and Schumacher K., "Universal Low Cost Controller for Electric Motors with Programmable Characteristic Curves", *IEEE Journal of Solid State Circuits,* Vol. 31, N° 7, pp. 1041-1045, July 1996.

[OiTo91] Oishi K. et al., "Application of fuzzy control theory to the sake brewing process", *J. on Fermentation and Bioengineering.,* Vol. 72, N° 2, pp. 115-121, 1991.

[PaKo00] Park H.K. and Kong S.G., "Neuro-Fuzzy Control System for Adaptive Noise Cancellation", *in Proc. of the IEEE International Fuzzy Systems Conference,* Seoul, Korea, August 22-25, 1999.

[PaPo92] Pagni A., Poluzzi R. and Rizzotto G., "W.A.R.P.: weight associative rule processor, an innovative fuzzy controller", in *Proc. of the 2nd Int. Conference on Fuzzy Logic and Neural Networks,* pp. 543-546, Japan, 1992.

[PaQu95] Pammu S. and Quigley S., "Novel Analogue CMOS Defuzzification Circuit", *IEE Proceedings Circuits Devices and Systems,* Vol. 142, N° 3, pp. 173-178, June 1995.

[Paty92] Patyra M., "Lecture Notes in Fuzzy Logic", *CAST,* EPFL, June 1992.

[PeDu89] Pelgrom M., Duinmaijer A. and Welbers A., "Matching Properties of MOS Transistors", *IEEE J. of Solid-State Circuits,* Vol. 24, N° 5, pp. 1433-1440, October 1989.

[PeRo93] Pelayo F. J., Rojas I., Ortega J. and Prieto A., "Current-mode analogue defuzzifier", *Electronics Letters,* Vol. 29, N° 9, pp. 743-744, April 1993.

[Proa95] Proakis J.G., "Digital Communications", 3rd edition, *Mc Graw-Hill International Editions,* Electrical Engineering series, 1995.

[RoMo51] Robbins H. and Monro S., "A Stochastic Approximation Method", *Annals of Mathematical Statistics 22,* 400-407.

[RoPe96] Rojas I., Pelayo F. J., Ortega J. and Prieto A., "A CMOS implementation of fuzzy controllers based on adaptive membership function range", *Microneuro'96,* pp. 317-321, 1996.

[RoPe97] Rojas I., Pelayo F.J.; Ortega J. and Prieto A., "Compact CMOS fuzzy controllers using the normalized product of adaptive membership functions", *Electronics Letters,* Vol. 33, N° 3, pp. 221-223, January 1997.

[RoPe97-b] Rojas I., Pelayo F.J., Anguita M. and Prieto A., "Continuous-time analog defuzzifier for product-sum based implementations", *Microneuro '94,* pp. 324-330, Torino, September 1994.

[RoPe99] Rojas I., Pelayo F.J., Anguita M., Bernier J.L. and Prieto A., "Implementation of adaptable and hierarchical fuzzy T-Norm", *Electronics Letters,* Vol. 35, N° 24, pp. 2150-2152, Nov. 1999.

[SaIn90] Sasaki M., Inoue T., Shirai Y. and Ueno F., "Fuzzy Multiple-Input Maximum and Minimum Circuits in Current Mode and their Analysis Using Bounded-Difference Equations", *IEEE Trans. on Computers,* Vol. 39, N° 6, pp. 768-774, June 1990.

[SaRu93] Saffiotti A. et al., "Blending reactivity and goal-directedness in a fuzzy controller", in *2nd Int. Conference on Fuzzy Systems,* pp. 134-139, 1993.

[SaSa93] Sakaguchi S. et al., "Application of Fuzzy Logic to shift scheduling method for automatic transmission", in *2nd Int. Conference on Fuzzy Systems",* pp. 52-58, 1993.

[ScMa85] Scharf E. et al., "The application of a fuzzy controller to the control of a multi-degree-of freedom robot arm", in *Industrial Applications of Fuzzy Control,* M. Sugeno Ed., Amsterdam: North Holland, pp. 41-61, 1985.

[ScGh90] Schaumann R., Ghausi M, and Laker K., "Design of Analog Filters: Passive, Active and Switched Capacitor", *Prentice-Hall, Inc.,* Englewood Cliffs, NJ, 1990.

[ShOs92] Shimizu K. et al, "The digital fuzzy processor FP-5000", in *Proc. of the 2nd Int. Conference on Fuzzy Logic and Neural Networks,* pp. 539-542, Japan, 1992.

[SiEl95] Simon D. and El-Sherief H., "Fuzzy Logic for Digital Phase-Locked Loop Filter Design", *IEEE Transactions on Fuzzy Systems,* Vol. 3, N° 2, pp. 211-218, May 1995.

[SoQu98] Song C., Quigley S. and Pammu S., "A Novel CMOS Analogue Fuzzy Inference Processor", in *Proceedings of The 1998 IEEE International Symposium on Circuits and Systems ISCAS'98,* Monterrey, CA, USA, May 1998.

[SoTs96] So W., Tse C. and Lee Y., "Development of a Fuzzy Logic Controller for DC/DC Converters: Design, Computer Simulation, and Experimental Evaluation", *IEEE Trans. on Power Electronics,* Vol. 11, N° 1, pp. 24-32, January 1996.

[Stie93] "Development and simulation of an F/A-18 fuzzy logic automatic carrier landing system", in *2nd Int. Conference on Fuzzy Systems,* pp. 797-802, 1993.

[SuKa86] Sugeno M. et al., "Fuzzy modeling and control of multilayer incinerator", *Fuzzy Sets and Systems,* Vol. 18, N° 3, pp. 329-345, 1986.

[SuMu85] Sugeno M. et al., "An experimental study on fuzzy parking control using a model car", in *Industrial Applications of Fuzzy Control,* M. Sugeno Ed., Amsterdam: North Holland, pp. 125-138, 1985.

[SuYa93] Sugeno M. and Yasukawa T., "A Fuzzy Logic Based Approach to Qualitative Modeling", *IEEE Trans. on Fuzzy System,* Vol. 1, N°1, pp. 7-31, February 1993.

[TaFr98] Tang Z. and Frey D., "A 100Mhz Programmable CMOS Gm-C Analog Equalizer for 100Base-TX Application", *1998 ESSCIRC Conference,* pp. 372-375.

[TaSa94] Tanaka K et al., "Identification and analysis of fuzzy model for air pollution: and approach to self-learning control of CO concentration", *Fuzzy Logic Technology and Applications,* Piscataway, NJ: IEEE, 1994, pp. 261-266.

[This96] Thissen P., "Arquitectures de Circuits Massivement Parallèles pour la Classification par Méthodes Neuronales", *Ph.D. Dissertation,* UCL, Belgium, Novembre 1996.

[ToBe80] R. Tong et al., "Fuzzy control on the activated sludge wastewater plant", *Automatica,* Vol. 16, N° 6, pp. 695-701, 1980.

[ToHa98] Tolias Y., Hadjileontiadis L. and Panas S., "Real-Time Separation of Discontinuous Adventitious Sounds from Vesicular Sounds Using a Fuzzy Rule-Based Filter", *IEEE Trans. on Information Technology in Biomedicine,* Vol. 2, N° 3, pp. 204-215, Sept. 1998.

[ToLi90] Toumazou, C., Ligey, F.J. and Haigh, D.G., "Analogue IC Design: The Current-Mode Approach", *Peregrinus Cop.,* Stevenage, Herts, U.K., , 1990.

[ToWa85] Togai M. and Watanabe H., "A VLSI Implementation of a Fuzzy Inference Engine: Toward an Expert System on Chip", *Information Sciences,* Vol. 38, N° 2, pp. 147-163, 1985.

[TsIn95] Tsukano K. and Inoue T., "Synthesis of Operational Transconductance Amplifier-Based Analog Fuzzy Functional Blocks and its Application", *IEEE Transactions on Fuzzy Systems,* Vol. 3, N° 1, pp. 61-68, February 1995.

[VaFa93] Vachtsevanos G. et al., "Fuzzy logic control of an automotive engine", *IEEE Control System Magazine,* Vol. 13, N° 3, pp. 62-86, Feb. 1993.

[VaVi99] 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. Process.,* Vol. 46, N°3, pp. 251-265, March 1999.

[Verl92] Verleysen M., "Neural Networks and Content-Addressable Memories for Vision: from Theory to VLSI.", *Ph.D. Dissertation,* UCL, L-L-N, Belgique, February 1992.

[Vida96] Vidal-Verdú F., "Design of Mixed-Signal CMOS Neuro-Fuzzy Controllers", *Ph.D. Dissertation,* Málaga, Spain, March 1996.

[Vitt94] Vittoz E., "Micropower Techniques", in *Design of VLSI Circuits for Telecommunications and Signal Processing, Ed. J. Franca and Y. Tsividis, Prentice Hall, Inc.,* Englewood Cliffs, 1994.

[VlSi99] Vlassis S. and Siskos S., "Analog CMOS Four-Quadrant Multiplier and Divider", in *Proceedings of The 1999 IEEE International Symposium on Circuits and Systems ISCAS'99,* Orlando, FL., USA, May 1999.

[WaDe90] Watanabe H., Detlof W. and Yount K., "A VLSI fuzzy logic controller with reconfigurable, cascadable architecture.", *IEEE Journal of Solid State Circuits,* Vol. 25, N°2, pp. 376-382, April 1990.

[WaMe94] Wang L. and Mendel J., "An RLS adaptive filter, with application to nonlinear channel equalization", in *Fuzzy Logic Technology and Applications,* Piscataway: IEEE, pp. 389-394, 1994.

[Wang91] Wang Z., "Current-Mode CMOS Integrated Circuits for Analog Computation and Signal Processing: a Tutorial", *Analog Integrated Circuits and Signal Processing I,* pp. 287-295, Kluwer Academic Publishers, Boston, 1991.

[Wieg93] Wiegerink R., "Analysis and Synthesis of MOS Translinear Circuits", *Kluwer Academic Publishers,* 1993.

[WiJa96] Wilamowsky B. et al., "VLSI Implementation of a Universal Fuzzy Controller", *ANNIE'96: Artificial Neural Networks in Engineering,* St. Louis, Missourri, USA, November 1996.

[WiSt85] Widrow B. and Stearns S. D., "Adaptive Signal Processing", Chapter 10, pp. 244-249, *Prentice-Hall, Inc.,* Englewood Cliffs, New Jersey 07632, USA, 1985.

[WySc93] Wyszynski A. and Schaumann R., "A Current-Mode Biquadratic Amplitude Equalizer", *Analog Integrated Circuits and Signal Processing,* N°4, pp. 161-166, Kluwer Academic Publishers, Boston, 1993.

[YaHa86] Yasunobu S. and Hasegawa T., "Evaluation of an automatic container crane operation system based on predictive fuzzy control", *Control Theory and Advanced Applications,* Vol. 2, N° 3, pp. 419-432, 1986.

[YaIt85] Yagjshita O. et al., "Application of fuzzy reasoning to the water purification process", in *Industrial Applications of Fuzzy Control,* M. Sugeno Ed., Amsterdam: North Holland, pp. 19-39, 1985.

[Yama80] Yamakawa T. et al., "Implementation of Fuzzy Logic Hardware Systems-Three Fundamental arithmetic Circuits", *Trans. IECE,* Vol. 63-C, N° 10, pp. 720-721, 1980.

[YaMi85] Yasunobu S. and Miyamoto S., "Automatic train operation system by predictive fuzzy control", in *Industrial Applications of Fuzzy Control,* M. Sugeno Ed., Amsterdam: North Holland, pp. 115-128, 1985.

[YaMi86] Yamakawa T. and Miki T., "The Current Mode Fuzzy Logic Integrated Circuits Fabricated by the Standard CMOS Process", *IEEE Trans. on Computers,* Vol. c-35, N° 2, pp. 161-167, February 1986.

[YuCh96] Yu P.T. and Chen R.C., "Fuzzy Stack Filters: their Definitions, Fundamental Properties, and Application in Image Processing", *IEEE Trans. on Image Processing,* Vol. 5, N° 6, pp. 838-854, June 1996.

[Zade65] Zadeh, L., "Fuzzy sets", *Information and Control 8,* pp. 338-353.

[Zade73] Zadeh, L., "Outline of a new approach to the analysis of complex systems and decision processes", *IEEE Trans. on System, Man and Cybernetics,* Vol. 3, N° 1, pp. 28-44, January 1973.

This page intentionally left blank

Index