Titre: Title:	From a PLC to a PFC front end programming unit
Auteurs: Authors:	PierPaolo Carini, Romano M. De Santis, & André Bazergui
Date:	1990
Type:	Rapport / Report
Référence: Citation:	Carini, P.P., De Santis, R. M., & Bazergui, A. (1990). From a PLC to a PFC front end programming unit (Rapport technique n° EPM-RT-90-02). https://publications.polymtl.ca/9879/

Institution:	École Polytechnique de Montréal
Numéro de rapport: Report number:	EPM-RT-90-02
URL officiel: Official URL:	
Mention légale: Legal notice:	

EPM/RT-90/2

FROM A PLC TO A PFC FRONT END PROGRAMMING UNIT

Par

PierPaolo Carini, Dr. Ing.

Projet dirigé par

Romano M. DeSantis, Professeur

Département de génie électrique

École Polytechnique de Montréal

Mars 1990

gratuit

Ce document a pu être publié grâce à une subvention du "Groupe de travail sur le programme spécial en contrôle de robotique" de NATO.

Je tiens à remercier le Professeur Riccardo Scattolini de l'Université de Pavia qui m'a donné la possibilité de vivre à Montréal une importante expérience dans ma vie et mes études.

Dépôt légal, 1er trimestre 1990
Bibliothèque nationale du Québec
Bibliothèque nationale du Canada

Pour se procurer une copie de ce document, s'adresser au:

Service de l'édition
École Polytechnique de Montréal
Case postale 6079, Succursale A
Montréal (Québec) H3C 3A7
(514) 340-4000

Compter 0,10 $ par page (arrondir au dollar le plus près) et ajouter 3,00 $ (Canada) pour la couverture, les frais de poste et la manutention. Régler en dollars canadiens par chèque ou mandat-poste au nom de l'École Polytechnique de Montréal. Nous n'honorerons que les commandes accompagnées d'un paiement, sauf s'il y a eu entente préalable dans le cas d'établissements d'enseignement, de sociétés ou d'organismes canadiens.

INDEX

SUMMARY

The modifications required to obtain a front end programming unit for a Programmable Fuzzy controller (PFC) from a front end programming unit of a Programmable Logic Controller (PLC) are discussed. This is done by considering PLC units using ladder logic diagrams and by restricting attention to a special class of PFCs. A simple example illustrates main concepts and procedures.

Chapter 1

INTRODUCTION

In a number of industrial applications the usage of logic controllers [1,2] is often complemented with that of fuzzy logic controllers [1,3]. In analogy to the by now well established Programmble Logic Controllers (PLC), it appears then natural to consider the development of general purpose Programmable Fuzzy Controllers (PFC). As a first step in this direction, in what follows we discuss the modifications which are required to make a PLC front end programming unit suitable to serve as a PFC front end unit.

In view of the large variety of both fuzzy controllers and PLC units available, this will be done by restricting attention to fuzzy controllers of a special class (to be defired in the next section) and to PLC front end programming units based on ladder logic diagrams. With this formalism, inputs to the controllers are represented in terms of energized/dienergized state of coils; the working of the controller is represented in terms of a diagram of contacts and coils. The programming of a given logic controller is then implemented by interpreting its function in terms of such a diagram and by communicating the result to the PLC front end unit. While for detailed discussion on ladder logic diagrams the reader is referred to [1,2], for the purposes of the present development it is sufficient to illustrate the passage from the logic description of a logic controller to a ladder logic diagram by considering a simple example. With reference to figure 1:

a) gives the boolean scheme of the logic controller; b) the representation of the physical inputs of the controller; c) the output of the controller; d) gives the program coding of the logic controller using ladder diagram language.

One observes the presence of three blocks: the first block assigns a memory location to every physical input; the second represents the logic function of the controller in terms of the ladder diagram language; the third indicates the memory locations where the outputs of the function are stored.

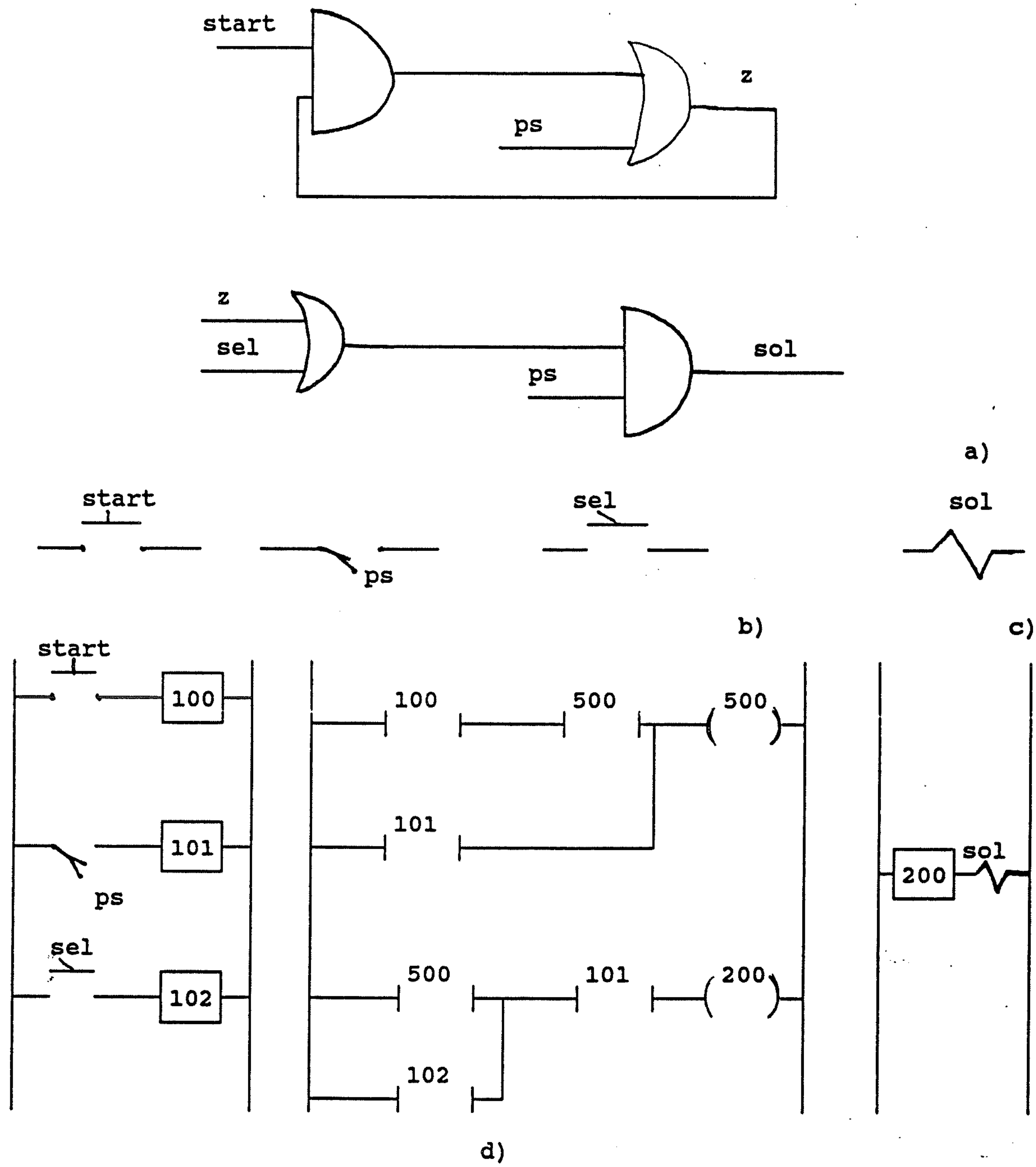


Figure 1: Program coding of a logic controller using ladder logic diagram.

Chapter 2.

A SPECIAL CLASS OF FUZZY CONTROLLERS.

2.1 Introduction

The fuzzy controllers of interest in what follows put into correspondence real valued vector inputs with real valued vector outputs. To describe these controllers, one needs to first introduce the concept of fuzzy value, fuzzy rule, truth and output value of a fuzzy rule.

2.2 Fuzzy value.

To define the fuzzy value of a real valued vector we consider the following elements (fuzzy scheme):

a) a set of attribute names:

$$\mathcal{Z} \triangleq \{Z_1, Z_2, \ldots\ldots\ldots\ldots, Z_M\}$$

b) a set of attributes domains:

$$\mathcal{D} \triangleq \{D_1, D_2, \ldots\ldots\ldots\ldots, D_M\}$$

with:

$$D_i = \{\delta_{im}\} \qquad i = 1,M; \; m = 1,p_i$$

where the elements δ_{im} denote the attributes of the domain D_i.

c) a set of membership functions:

$$\delta_{im}(\cdot): R \rightarrow [0,1]$$

These elements allow us to put in correspondence with a real valued vector:

$$\underline{z} = (z_1, z_2, \ldots\ldots\ldots\ldots\ldots, z_M)$$

a fuzzy value:

$$\underline{z}^* = (z_1^*, z_2^*, \ldots\ldots\ldots\ldots\ldots, z_M^*)$$

where:

$$z_i^* = [\delta_{im}, \delta_{im}(z_i)]$$

In linguistic terms, the fuzzy value $\underline{z}^*$ of the real valued vector $\underline{z}$ is reported by saying that:

"z_i is δ_{im} with degree of membership $\delta_{im}(z_i)$"

$$m = 1, p_i; \; i = 1, M$$

In what follows real valued vectors will represent inputs and outputs to and from a fuzzy controller.

In the case of input representation, the fuzzy scheme will use the notation:

$$\mathcal{Z} = \mathcal{X} \triangleq \{X_i\} \qquad i = 1,n$$

$$\mathcal{D} = \mathcal{A} \triangleq \{A_i\}$$

with:

$A_i = \{\sigma_{im}\}$ $\qquad i = 1,n;\ m = 1,p_i$

To the attributes σ_{im} correspond the membership functions:

$\sigma_{im}(\cdot) : R \rightarrow [0,1]$

Given as input a real valued vector:

$\underline{x} = \{x_1, x_2, \ldots\ldots\ldots\ldots, x_n\}$

we then define the fuzzy value of the input:

$\underline{x}^* = \{x_1^*, x_2^*, \ldots\ldots\ldots\ldots, x_n^*\}$

where:

$x_i^* = [\sigma_{im}, \sigma_{im}(x_i)]$

In the case of outputs representation, the fuzzy scheme will use the following notation:

$\mathcal{Z} = \mathcal{Y} \triangleq \{Y_h\}$ $\qquad h = 1,H$

$\mathcal{D} = \mathcal{B} \triangleq \{B_h\}$

with:

$B_h = \{\beta_{hl}\}$ $\qquad h = 1,H;\ l = 1,p_h$

To the attributes β_{hl} correspond the membership functions:

$\beta_{hl}(\cdot) : R \rightarrow [0,1]$

The output is given by a real valued vector:

$\underline{y} = \{y_1, y_2, \ldots\ldots\ldots\ldots, y_H\}$

2.3 Fuzzy rule.

To define a fuzzy rule we firstly introduce the concept of WFF (Well Formed Formula).

Given:

- a set of names $= \{Z_1, Z_2, \ldots\ldots\ldots, Z_M\}$
- a set of attributes domain $= \{D_1, D_2, \ldots\ldots\ldots\ldots, D_M\}$
- the set of connectives $\mathcal{C} = \{ \neg, \wedge, \vee \}$

a WFF defined over $\mathcal{Z} \times \mathcal{D}$ is recursively defined as:

- The pair (Z_i, δ_{im}), with δ_{im} attribute of the domain D_i, is a(n atomic) WFF.
- if τ_1 and τ_2 are WFF, then:
  - $(\neg\ \tau_1)$ is a WFF.
  - $(\tau_1 \wedge \tau_2)$ is a WFF.
  - $(\tau_1 \vee \tau_2)$ is a WFF.

With this concept a fuzzy rule is defined as a pair:

(Premise, Consequence)

where Premise is an element of the WFF defined over $\mathcal{X} \times \mathcal{A}$ and Consequence is an element of the atomic WFF defined over $\mathcal{Y} \times \mathcal{B}$.

2.4 Truth value and output value of a fuzzy rule.

Given:

- the fuzzy value of the input $\underline{x}^*$
- the Premise of the fuzzy rule, τ, element of WFF defined over $\mathcal{X} \times \mathcal{A}$

the Truth value of the fuzzy rule, μ, is recursively defined as follows:

- if τ is an atomic WFF of the form (X_i, σ_{ij}), then the value μ of τ is $\tau(\underline{x}^*) = \sigma_{ij}(x_i)$.
- if τ_1 and τ_2 are WFF defined over $\mathcal{X} \times \mathcal{A}$ then:
  - if $\tau = (\neg \tau_1)$ then the value μ of τ is $\tau(\underline{x}^*) = 1 - \tau_1(\underline{x}^*)$.
  - if $\tau = (\tau_1 \wedge \tau_2)$ then the value μ of τ is $\tau(\underline{x}^*) = \min(\tau_1(\underline{x}^*), \tau_2(\underline{x}^*))$.
  - if $\tau = (\tau_1 \vee \tau_2)$ then the value μ of τ is $\tau(\underline{x}^*) = \max(\tau_1(\underline{x}^*), \tau_2(\underline{x}^*))$.

The fuzzy value of a fuzzy rule is given by $y^* = [\beta, \mu]$, where β is the attribute of the domain taken in consideration in the Consequence of the rule and μ is the Truth value of the rule.
The output value of a fuzzy rule is given by $y = \beta^{-1}(\mu)$, where $\beta^{-1}(\cdot)$ denotes the pseudo-inverse of $\beta(\cdot)$.

2.5 Fuzzy controller.

A fuzzy controller is described in terms of the following three elements:

-an ENCODER, which associates to a real valued vector $\underline{x}$ of the input a fuzzy value $\underline{x}^*$, according with a given fuzzy scheme.

-a SET OF FUZZY RULES R_{hj} (j = 1,N; h =1,H)), which associates to $\underline{x}^*$ the fuzzy value of the output:

$$\underline{y}^* = (y^*_{h1}, y^*_{h2}, \ldots\ldots\ldots, y^*_{hN}) \qquad h = 1,H$$

where y_{hj} is the fuzzy value $[\beta_{hj}, \mu_{hj}]$ of the rule R_{hj} (j = 1,N; h = 1.H); β_{hj} is the attribute of the domain taken in consideration in the Consequence and μ_{hj} is the Truth value of the rule R_{hj}.

-a DECODER, which associates to a fuzzy value $\underline{y}^*$ of the output the output value $\underline{y}$. This is usually done by applying the formula:

$$y_h = \frac{\sum_{j=1}^{N} \beta_{hj}^{-1}(\mu_{hj}) * \mu_{hj}}{\sum_{j=1}^{N} \mu_{hj}} \qquad h = 1,H$$

Chapter 3.

AN EXAMPLE.

3.1 Introduction.

In what follows we will illustrate the concepts introduced in Chapter 2 by applying them to a fuzzy controller for a warm water plant (See [6] for further details). Figure 2 shows the scheme of the process: the cold water enters the tank with a fixed flow F; it is heated by a heat exchange unit in which hot water at 90 C flows with a variable flow F1. The goal is to control the temperature T_0 of the heated water produced by the plant by adjusting F1. The strategy adopted by the fuzzy controller is based on the strategy which is usually utilized by the human operator: it uses error and change in error of the temperature to affect a change of flow F1.

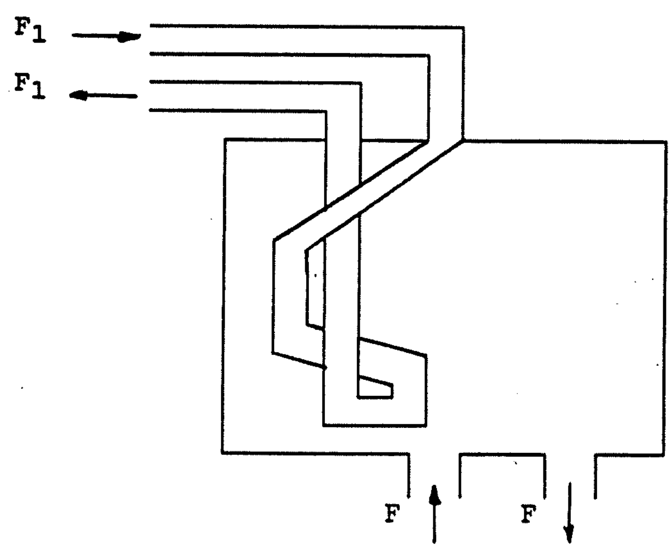


Figure 2: Scheme of a warm water plant.

3.2 Fuzzy values.

- set of input names:

$$\mathcal{X} = \{X_1, X_2\}$$

where:

- $X_1 \triangleq$ error of temperature
- $X_2 \triangleq$ change in error of temperature

- set of attributes domains:

$$\mathcal{A} = \{A_1, A_2\}$$

The attributes domain A_1 associated to X_1 is characterized by the attributes:

σ_{11} = small-negative

σ_{12} = very-small-negative

σ_{13} = small-positive

σ_{14} = very-small-positive

The attributes domain A_2 associated to X_2 is characterized by the attributes:

σ_{21} = small-positive

σ_{22} = medium-positive

σ_{23} = big-positive

σ_{24} = small-negative

σ_{25} = medium-negative

σ_{26} = big-negative

To each one of these attributes corresponds a membership function; in this example we have:

ATTRIBUTE	CORRESPONDING MEMBERSHIP FUNCTION		
σ_{11}	$\sigma_{11}(x)$ =	$small_E(x)$	$x<0$
		0	$x>0$
σ_{12}	$\sigma_{12}(x)$ =	$verysmall_E(x)$	$x<0$
		0	$x>0$
σ_{13}	$\sigma_{13}(x)$ =	$small_E(x)$	$x>0$
		0	$x<0$
σ_{14}	$\sigma_{14}(x)$ =	$verysmall_E(x)$	$x>0$
		0	$x<0$

σ_{21}	$\sigma_{21}(x) =$	$small_{CE}(x)$	$x>0$
		0	$x<0$
σ_{22}	$\sigma_{22}(x) =$	$medium_{CE}(x)$	$x>0.5$
		0	$x<0.5$
σ_{23}	$\sigma_{23}(x) =$	$big_{CE}(x)$	$x>2$
		0	$x<2$
σ_{24}	$\sigma_{24}(x) =$	$small_{CE}(x)$	$x<0$
		0	$x>0$
σ_{25}	$\sigma_{25}(x) =$	$medium_{CE}(x)$	$x<-0.5$
		0	$x>-0.5$
σ_{26}	$\sigma_{26}(x) =$	$big_{CE}(x)$	$x<-2$
		0	$x>-2$

where we have used the notation:

$$small_E(x) = (1+0.5|x|)^{-1}$$

$$very\text{-}small_E(x) = (1+x^4)^{-1}$$

$$small_{CE}(x) = (1+(3|x|)^2)^{-1}$$

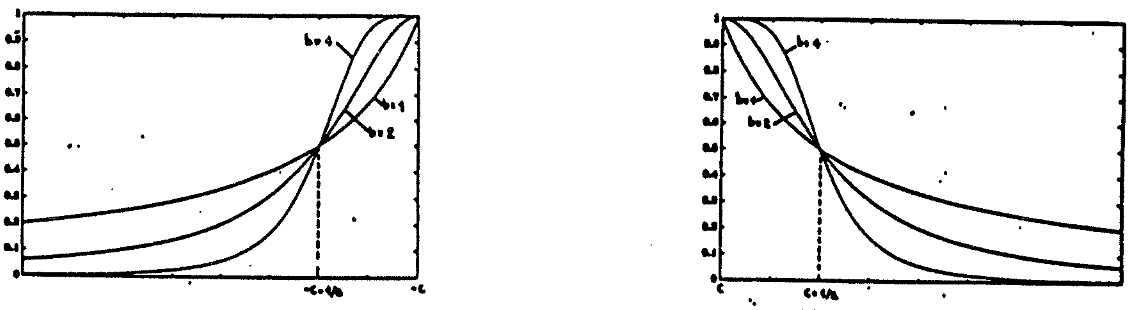
$$medium_{CE}(x) = (1+(3(|x|-0.5))^2)^{-1}$$

$$big_{CE}(x) = (1+(3(|x|-2))^2)^{-1}$$

These functions are all of the form:

$$f(x) = (1+(a(|x|-c))^{b})^{-1}$$

Figure 3.a shows the form of f(x) for x<-c, while in figure 3.b is given the form of f(x) for x>c; c alters the position of the points in which f(x) = 1, a changes the spread and b the contrast.



a) b)

Figure 3: Form of the membership functions taken into consideration in the presented front end.

The real valued vector of the input is constituted by the elements:

- x_1 = E
- x_2 = CE

It is then possible to define the fuzzy value of the output as:

$$\underline{x} = (x_1,\ x_2)$$

where:

- $$x_1 = \begin{bmatrix} [\sigma_{11},\ \sigma_{11}(x_1)] \\ [\sigma_{12},\ \sigma_{12}(x_1)] \\ [\sigma_{13},\ \sigma_{13}(x_1)] \\ [\sigma_{14},\ \sigma_{14}(x_1)] \end{bmatrix} \qquad x_2 = \begin{bmatrix} [\sigma_{21},\ \sigma_{21}(x_2)] \\ [\sigma_{22},\ \sigma_{22}(x_2)] \\ [\sigma_{23},\ \sigma_{23}(x_2)] \\ [\sigma_{24},\ \sigma_{24}(x_2)] \\ [\sigma_{25},\ \sigma_{25}(x_2)] \\ [\sigma_{26},\ \sigma_{26}(x_2)] \end{bmatrix}$$

In the case of the output we have:

- a real scalar value:

$$y \triangleq CF1$$

- an output name:

$$Y \triangleq \text{change in flow F1}$$

- an attributes domain B.

Note: in this example the output value is scalar; with reference to the given fuzzy controller scheme this means that $H = 1$; for simplicity in what follows we thus will not use the index h ($h= 1,H$).

The attributes domain B associated to Y is characterized by the set of attributes:

β_1 = very-big-positive

β_2 = very-small-positive

β_3 = small-negative

β_4 = medium-negative

β_5 = big-negative

β_6 = very-big-negative

β_7 = very-small-negative

β_8 = small-positive

β_9 = medium-positive

β_{10} = big-positive

To each one of these attributes corresponds a membership function; we thus have:

ATTRIBUTE NAME	CORRESPONDING MEMBERSHIP FUNCTION			
β_1	$\beta_1(x)$	=	very-big$_{CF1}(x)$	$x>12$
			0	$x<12$
β_2	$\beta_2(x)$	=	very-small$_{CF1}(x)$	$x>0$
			0	$x<0$
β_3	$\beta_3(x)$	=	small$_{CF1}(x)$	$x<-0.2$
			0	$x>-0.2$
β_4	$\beta_4(x)$	=	medium$_{CF1}(x)$	$x<-1$
			0	$x>-1$
β_5	$\beta_5(x)$	=	big$_{CF1}(x)$	$x<-3$
			0	$x>-3$

β_6	$\beta_6(x) = \text{very-big}_{CF1}(x)$	$x>12$
	0	$x<12$
β_7	$\beta_7(x) = \text{very-small}_{CF1}(x)$	$x>0$
	0	$x<0$
β_8	$\beta_8(x) = \text{small}_{CF1}(x)$	$x>0.2$
	0	$x<0.2$
β_9	$\beta_9(x) = \text{medium}_{CF1}(x)$	$x>1$
	0	$x<1$
β_{10}	$\beta_{10}(x) = \text{big}_{CF1}(x)$	$x>3$
	0	$x<3$

where:

$$\text{very-big}_{CF1}(x) = (1+(2(|x|-12))^2)^{-1}$$
$$\text{very-small}_{CF1}(x) = (1+2x^2)^{-1}$$
$$\text{small}_{CF1}(x) = (1+(2(|x|-0.2))^2)^{-1}$$
$$\text{medium}_{CF1}(x) = (1+(2(|x|-1))^2)^{-1}$$
$$\text{big}_{CF1}(x) = (1+(|x|-3)^2)^{-1}$$

3.3 Fuzzy rules.

The set of rules is given by:

R_1 : To this rule corresponds the linguistic expression:

IF (error of temperature, not small-negative)

THEN (change of flow, very-big-positive)

Premise : $(\neg X_1, \sigma_{11})$

Consequence: (Y, β_1)

R_2 : IF (error of temperature, small-negative)

THEN (change of flow, very-small-positive)

Premise : (X_1, σ_{11})

Consequence: (Y, β_2)

R_3 : IF (error of temperature, small-negative)

AND (change in error of temperature, small-positive)

THEN (change of flow, small-negative)

Premise : $((X_1, \sigma_{12}) \wedge (X_2, \sigma_{21}))$

Consequence: (Y, β_3)

R_4 : IF (error of temperature, very-small-negative)

AND (change in error of temperature, medium-positive)

THEN (change of flow, medium-negative)

Premise : $((X_1, \sigma_{12}) \wedge (X_2, \sigma_{22}))$

Consequence: (Y, β_4)

R_5 : IF (error of temperature, very-small-negative)

AND (change in error of temperature, big-positive)

THEN (change of flow, big-negative)

Premise : $((X_1, \sigma_{12}) \wedge (X_2, \sigma_{23}))$

Consequence: (Y, β_5)

R_6 : IF (error of temperature, very-small-negative)

AND (change in error of temperature, small-negative)

THEN (change of flow, small-positive)

Premise : $((X_1, \sigma_{12}) \wedge (X_2, \sigma_{24}))$

Consequence: (Y, β_8)

R_7 : IF (error of temperature, very-small-negative)

AND (change in error of temperature, medium-negative)

THEN (change of flow, medium-positive)

Premise : $((X_1, \sigma_{12}) \wedge (X_2, \sigma_{25}))$

Consequence: (Y, β_9)

R_8 : IF (error of temperature, very-small-negative)

AND (change in error of temperature, big-negative)

THEN (change of flow, big-positive)

Premise : $((X_1, \sigma_{12}) \wedge (X_2, \sigma_{26}))$

Consequence: (Y, β_{10})

R_9 : IF (error of temperature, not small-positive)

THEN (change of flow, very-big-negative)

Premise : $(\neg X_1, \sigma_{13})$

Consequence: (Y, β_6)

R_{10} : IF (error of temperature, small-positive)

THEN (change of flow, very-small-negative)

Premise : (X_1, σ_{13})

Consequence: (Y, β_7)

R_{11} : IF (error of temperature, very-small-positive)

AND (change in error of temperature, small-negative)

THEN (change of flow, small-positive)

Premise : $((X_1, \sigma_{14}) \wedge (X_2, \sigma_{25}))$

Consequence: (Y, β_8)

R_{12} : IF (error of temperature, very-small-positive)

AND (change in error of temperature, medium-negative)

THEN (change of flow, medium-positive)

Premise : $((X_1, \sigma_{14}) \wedge (X_2, \sigma_{26}))$

Consequence: (Y, β_9)

R_{13} : IF (error of temperature, very-small-positive)

AND (change in error of temperature, big-negative)

THEN (change of flow, big-positive)

Premise : $((X_1, \sigma_{14}) \wedge (X_2, \sigma_{26}))$

Consequence: (Y, β_{10})

R_{14} : IF (error of temperature, very-small-positive)

AND (change in error of temperature, small-positive)

THEN (change of flow, small-negative)

Premise : $((X_1, \sigma_{14}) \wedge (X_2, \sigma_{21}))$

Consequence: (Y, β_3)

R_{15} : IF (error of temperature, very-small-positive)

AND (change in error of temperature, medium-positive)

THEN (change of flow, medium-negative)

Premise : $((X_1, \sigma_{14}) \wedge (X_2, \sigma_{22}))$

Consequence: (Y, β_4)

R_{16} : IF (error of temperature, very-small-positive)

AND (change in error of temperature, big-positive)

THEN (change of flow, big-negative)

Premise : $((X_1, \sigma_{14}) \wedge (X_2, \sigma_{23}))$

Consequence: (Y, β_5)

3.4 Output value and Truth value of the rules.

In this example $\underline{y}^* = [y_j^*]$ (j = 1,16) where:

$y_1^* = [\beta_1 \; ,\mu_1]$

$y_2^* = [\beta_2 \; ,\mu_2]$

$y_3^* = [\beta_3 \; ,\mu_3]$

$y_4^* = [\beta_4 \; ,\mu_4]$

$y_5^* = [\beta_5 \; ,\mu_5]$

$y_6^* = [\beta_8 \; ,\mu_6]$

$y_7^* = [\beta_9 \; ,\mu_7]$

$y_8^* = [\beta_{10} \; ,\mu_8]$

$y_9^* = [\beta_6 \; ,\mu_9]$

$y_{10}^* = [\beta_7 \; ,\mu_{10}]$

$y_{11}^* = [\beta_8 \; ,\mu_{11}]$

$y_{12}^* = [\beta_9 \; ,\mu_{12}]$

$y_{13}^* = [\beta_{10}, \; \mu_{13}]$

$y_{14}^* = [\beta_3 \; ,\mu_{14}]$

$y_{15}^* = [\beta_4 \; ,\mu_{15}]$

$y_{16}^* = [\beta_5 \; ,\mu_{16}]$

and:

$\mu_1 = 1 - \sigma_{11}(x_1)$

$\mu_2 = \sigma_{11}((x_1)$

$\mu_3 = \min(\sigma_{12}(x_1), \; \sigma_{21}(x_2))$

$\mu_4 = \min(\sigma_{12}(x_1), \sigma_{22}(x_2))$

$\mu_5 = \min(\sigma_{12}(x_1), \sigma_{23}(x_2))$

$\mu_6 = \min(\sigma_{12}(x_1), \sigma_{24}(x_2))$

$\mu_7 = \min(\sigma_{12}(x_1), \sigma_{25}(x_2))$

$\mu_8 = \min(\sigma_{12}(x_1), \sigma_{26}(x_2))$

$\mu_9 = 1 - \sigma_{13}(x_1)$

$\mu_{10} = \sigma_{13}((x_1)$

$\mu_{11} = \min(\sigma_{14}(x_1), \sigma_{24}(x_2))$

$\mu_{12} = \min(\sigma_{14}(x_1), \sigma_{25}(x_2))$

$\mu_{13} = \min(\sigma_{14}(x_1), \sigma_{26}(x_2))$

$\mu_{14} = \min(\sigma_{14}(x_1), \sigma_{21}(x_2))$

$\mu_{15} = \min(\sigma_{14}(x_1), \sigma_{22}(x_2))$

$\mu_{16} = \min(\sigma_{14}(x_1), \sigma_{23}(x_2))$

The pseudo-inverse of the functions $\mu = \beta_i(y)$ $(i = 1,10)$ are defined as:

$\beta^{-1}(\mu) =$	12	$\mu = 0$
	$12 + \sqrt{(1 - \mu)/(2\mu)}$	for other values of μ
$\beta^{-1}(\mu) =$	0	$\mu = 0$
	$\sqrt{(1 - \mu)/\mu}$	for other values of μ
$\beta^{-1}(\mu) =$	-0.2	$\mu = 0$
	$-(0.5\sqrt{(1 - \mu)/\mu} + 0.2)$	for other values of μ
$\beta^{-1}(\mu) =$	-1	$\mu = 0$
	$-(0.5\sqrt{(1 - \mu)/\mu} + 1)$	for other values of μ

$$\beta^{-1}(\mu) = \begin{cases} -3 & \mu = 0 \\ -(0.5\sqrt{(1-\mu)/\mu} + 3) & \text{for other values of } \mu \end{cases}$$

$$\beta^{-1}(\mu) = \begin{cases} -12 & \mu = 0 \\ -(12 + \sqrt{(1-\mu)/(2\mu)}) & \text{for other values of } \mu \end{cases}$$

$$\beta^{-1}(\mu) = \begin{cases} 0 & \mu = 0 \\ -\sqrt{(1-\mu)/\mu} & \text{for other values of } \mu \end{cases}$$

$$\beta^{-1}(\mu) = \begin{cases} 0.2 & \mu = 0 \\ 0.2 + 0.5\sqrt{(1-\mu)/\mu} & \text{for other values of } \mu \end{cases}$$

$$\beta^{-1}(\mu) = \begin{cases} 1 & \mu = 0 \\ 1 + 0.5\sqrt{(1-\mu)/\mu} & \text{for other values of } \mu \end{cases}$$

$$\beta^{-1}(\mu) = \begin{cases} 3 & \mu = 0 \\ 3 + \sqrt{(1-\mu)/\mu} & \text{for other values of } \mu \end{cases}$$

Chapter 4.

FRONT END PROGRAMMING UNIT FOR A PROGRAMMABLE FUZZY CONTROLLER.

The task of a front end programming unit for a PFC is to communicate structure and function of a fuzzy controller. To do this it is necessary to give the description of the membership functions and the description of the fuzzy rules. A simple solution to describe membership functions is given in the case where the membership functions belong to a certain class and are specified in terms of a limited number of parameters; in this case one describes them by giving the specific values assumed by this parameters.
In the example of the previous Chapter, for instance, membership functions have the form:

$$\mu(x) = \begin{cases} (1+(a(x-c))^{b})^{-1} & x>c \\ 0 & x<c \end{cases} \qquad (p=1)$$

$$\mu(x) = \begin{cases} (1+(a(|x|-c))^{b})^{-1} & x<-c \\ 0 & x>-c \end{cases} \qquad (p=0)$$

Therefore they are completly defined by assigning values a, b, c and p. The following figures usa this example to illustrate the type of interface which is required to do this.

INPUT

-NAME:
Error of temperature
-CODE:
X_1

ATTRIBUTES	A	B	C	P
NAME: small-negative CODE: 1_1	0.5	1	0	0
NAME: very-small-negative CODE: 12	1	4	0	0
NAME: small-positive CODE: 13	0.5	1	0	1
NAME: very-small-positive CODE: 14	1	4	0	1

-NAME:
Change in error of temperature
-CODE:
X_2

ATTRIBUTES	A	B	C	P
NAME: small-positive CODE: 21	3	2	0	1
NAME: medium-positive CODE: 22	3	2	0.5	1
NAME: big-positive CODE: 23	1	2	2	1
NAME: small-negative CODE: 24	3	2	0	0
NAME: medium-negative CODE: 25	3	2	0.5	0
NAME: big-negative CODE: 26	1	2	2	0

Figure 4: Table to introduce the parameters characterizing the parameters of the input.

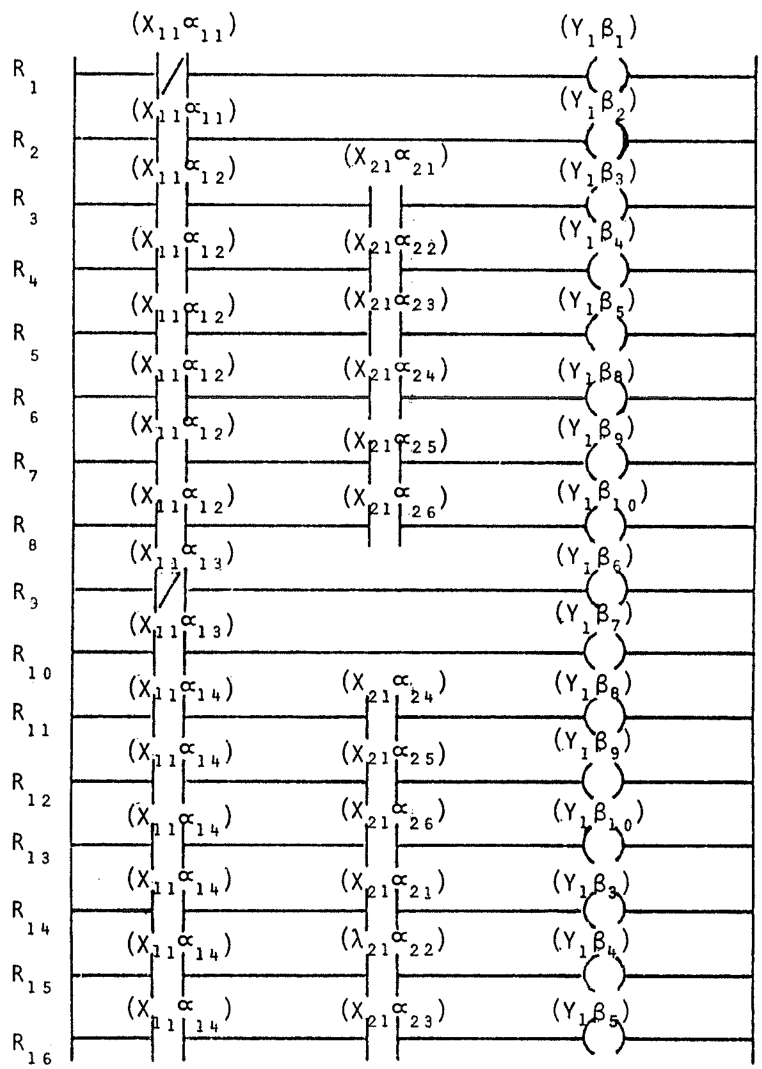
OUTPUT

-NAME:
Change in flow F1
-CODE:
Y

ATTRIBUTES	PARAMETERS			
	A	B	C	P
NAME: very-big-positive CODE: 1	2	2	12	1
NAME: very-small-positive CODE: 2	2	2	0	1
NAME: small-negative CODE: 3	2	2	0.2	0
NAME: medium-negative CODE: 4	2	2	1	0
NAME: big-negative CODE: 5	1	2	3	0
NAME: very-big-negative CODE: 6	2	2	12	0
NAME: very-small-negative CODE: 7	2	2	0	0
NAME: small-positive CODE: 8	2	2	0.2	1
NAME: medium-positive CODE: 9	2	2	1	1
NAME: big-positive CODE: 10	1	2	3	1

Figure 5: Table to introduce the parameters characterizing the parameters of the output.

To communicate the fuzzy rules we notice that the formalism used to express them is identical to that used to express logic rules for standard PLC; we may therefore describe them in terms of the same ladder logic diagrams used for the PLC. Figure 5 gives the ladder diagram representation of the sets of fuzzy rules used in the example of Chapter 3.



Chapter 5.

CLOSURE.

The interest of the PFC front end programming unit that we have presented is that it allows to transfer into the contest of fuzzy controllers the portability, familiarity and portability which characterize the well established Programmable Logic Controllers.

An additional interesting feature is in its potential of easily accomodate fuzzy controllers with a structure considerably more complex the that illustrated by our simple example. This includes for istance a fuzzy controller based on a compositional rule of inference [5, 9], multilevel rules systems [9, 10], fuzzy controllers of the predictive form [7, 8].

List of Figures.

Figure

REFERENCES

[1] Jones, C.T. and Bryan, L.A.: "Programmable Controllers" - IPC/ASTEC Publication (1985).

[2] Webb, J.W.: "Programmable Controllers" - Merril Publishing Company (1988).

[3] Zadeh, L.A.: "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes" - IEEE Trans. on SMC-3, No. 1; 28-44 (1973).

[4] Mamdani, B.E.: "Application of fuzzy algorithms for control of simple dynamic plant" - IEE Proc. Control & Science, Vol. 121, No 12; 1585-1588 (1974).

[5] Sugeno, M.: "An Introductory Survey of Fuzzy Control" - Information Sciences, 36; 59-83 (1985).

[6] Kickert, W.J.M. and Van Nauta Lemke, H.R.: "Application of a Fuzzy Controller in a Warm Water Plant" - Automatica, 36; 301-308 (1976).

[7] Miyamoto, M. and Yasunobu, S.: "Predictive fuzzy control and its application to automatic train operation systems" - First International Conference on Fuzzy Information Processing, (1984).

[8] Yasunobu, S. and Hasegawa,T: "Evaluation of an automatic container crane operation system based on predictive fuzzy control" - in "Control theory and advanced technology", Vol. 2, No. 3; 419-432 (1986).

[9] Kickert, W.J.M. and Mamdani, E.H.: "Analysis of a Fuzzy Logic Controller" - in Fuzzy Sets and Systems, No. 1; 29-44 (1978).

[10] Jensen, P.M.: "Industrial application of fuzzy logic control" Int.J.Man-Machine Studies, 12, 3-10; (1980).

[11] Ostergaard, J.J. and Holmsblad, L.P.: "Control of a cement kiln by fuzzy logic" - in "Fuzzy Information and Decision Processes" (M.M. Gupta and Sanchez, E., Eds.), North-Holland, Amsterdam; (1982).