SLOVENSKI STANDARD
SIST EN 61131-7:2002

01-september-2002

Programmable controllers - Part 7: Fuzzy control programming

Programmable controllers -- Part 7: Fuzzy control programming

Speicherprogrammierbare Steuerungen -- Teil 7: Fuzzy-Control-Programmierung

Automates programmables -- Partie 7: Programmation en logique floue

Ta slovenski standard je istoveten z: EN 61131-7:2000

ICS:

25.040.40	Merjenje in krmiljenje industrijskih postopkov	Industrial process measurement and control
35.240.50	Uporabniške rešitve IT v industriji	IT applications in industry

SIST EN 61131-7:2002 **en**

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 61131-7

December 2000

ICS 35.240.50;25.040.40

English version

Programmable controllers
Part 7: Fuzzy control programming
(IEC 61131-7:2000)

Automates programmables
Partie 7: Programmation en logique floue
(CEI 61131-7:2000)

Speicherprogrammierbare Steuerungen
Teil 7: Fuzzy-Control-Programmierung
(IEC 61131-7:2000)

This European Standard was approved by CENELEC on 2000-11-01. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the Central Secretariat or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the Central Secretariat has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Czech Republic, Denmark, Finland, France, Germany, Greece, Iceland, Ireland, Italy, Luxembourg, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland and United Kingdom.

CENELEC
European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

Central Secretariat: rue de Stassart 35, B - 1050 Brussels

Ref. No. EN 61131-7:2000 E

Foreword

The text of document 65B/406/FDIS, future edition 1 of IEC 61131-7, prepared by SC 65B, Devices, of IEC TC 65, Industrial-process measurement and control, was submitted to the IEC-CENELEC parallel vote and was approved by CENELEC as EN 61131-7 on 2000-11-01.

The following dates were fixed:

- latest date by which the EN has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2001-08-01
- latest date by which the national standards conflicting with the EN have to be withdrawn (dow) 2003-11-01

Annexes designated "normative" are part of the body of the standard.
Annexes designated "informative" are given for information only.
In this standard, annex ZA is normative and annexes A, B, C, D and E are informative.
Annex ZA has been added by CENELEC.

Endorsement notice

The text of the International Standard IEC 61131-7:2000 was approved by CENELEC as a European Standard without any modification.

CONTENTS

INTRODUCTION

The theory of fuzzy logic in the application of control is named fuzzy control. Fuzzy control is emerging as a technology that can enhance the capabilities of industrial automation, and is suitable for control level tasks generally performed in Programmable Controllers (PC).

Fuzzy control is based upon practical application knowledge represented by so-called linguistic rule bases, rather than by analytical (either empirical or theoretical) models. Fuzzy control can be used when there is an expertise that can be expressed in its formalism. That allows to take available knowledge to improve processes and perform a variety of tasks, for instance

- control (closed or open loop, single or multi-variable, for linear or non-linear systems),
- on-line or off-line setting of control systems' parameters,
- classification and pattern recognition,
- real-time decision making (send this product to machine A or B ?),
- helping operators to make decisions or tune parameters,
- detection and diagnosis of faults in systems.

Its wide range of applications and natural approach based on human experience makes fuzzy control a basic tool that should be made available to programmable controller users as a standard.

Fuzzy control can also, in a straightforward way, be combined with classical control methods.

The application of fuzzy control can be of advantage in such cases where there is no explicit process model available, or in which the analytical model is too difficult to evaluate or when the model is too complicated to evaluate in real time.

Another advantageous feature of fuzzy control is that human experience can be incorporated in a straightforward way. Also, it is not necessary to model the whole controller with fuzzy control: sometimes fuzzy control just interpolates between a series of locally linear models, or dynamically adapts the parameters of a "linear controller", thereby rendering it non-linear, or alternatively just "zoom in" onto a certain feature of an existing controller that needs to be improved.

Fuzzy control is a multi-valued control, no longer restricting the values of a control proposition to "true" or "false". This makes fuzzy control particularly useful to model empirical expertise, stating which control actions have to be taken under a given set of inputs.

The existing theory and systems already realized in the area of fuzzy control differ widely in terms of terminology (definitions), features (functionalities) and implementation (tools).

Fuzzy control is used from small and simple applications up to highly sophisticated and complex projects. To cover all kinds of usage in this part of IEC 61131, the features of a compliant fuzzy control system are mapped into defined conformance classes.

The basic class defines a minimum set of features which has to be achieved by all compliant systems. This facilitates the exchange of fuzzy control programs.

Optional standard features are defined in the extension class. Fuzzy control programs applying these features can only be fully ported among systems using the same set of features, otherwise a partial exchange may be possible only. This standard does not force all compliant systems to realize all features in the extension class, but it supports the possibility of (partial) portability and the avoidance of the usage of non-standard features. Therefore, a compliant system should not offer non-standard features which can be meaningfully realized by using standard features of the basic class and the extension class.

In order not to exclude systems using their own highly sophisticated features from complying with this part of IEC 61131 and not to hinder the progress of future development, this standard permits also additional non-standard features which are not covered by the basic class and the extension class. However, these features need to be listed in a standard way to ensure that they are easily recognised as non-standard features.

The portability of fuzzy control applications depends on the different programming systems and also the charcteristics of the control systems. These dependencies are covered by the data check list to be delivered by the manufacturer.

PROGRAMMABLE CONTROLLERS –

Part 7: Fuzzy control programming

1 Scope and object

This part of IEC 61131 defines a language for the programming of Fuzzy Control applications used by programmable controllers.

The object of this part of IEC 61131 is to offer the manufacturers and the users a well-defined common understanding of the basic means to integrate fuzzy control applications in the Programmable Controller languages according to IEC 61131-3, as well as the possibility to exchange portable fuzzy control programs among different programming systems.

To achieve this, annex A gives a short introduction to the theory of fuzzy control and fuzzy logic as far as it is necessary for the understanding of this part of IEC 61131. It may be helpful for readers of this part of IEC 61131 who are not familiar with fuzzy control theory to read annex A first.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of IEC 61131. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of IEC 61131 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of IEC and ISO maintain registers of currently valid International Standards.

IEC 60050-351:1998, *International Electrotechnical Vocabulary (IEV) – Part 351: Automatic control*

IEC 61131-3:1993, *Programmable controllers – Part 3: Programming languages*

3 Definitions

For the purpose of this part of IEC 61131, the following definitions apply.

Further definitions for language elements are given in IEC 61131-3.

NOTE Terms defined in this clause are italicized where they appear in the text of definitions.

3.1
accumulation
result aggregation

combination of results of *linguistic rules* in a final result

3.2
aggregation
determination of degree of firing

combination of membership degrees of all individual subconditions in a rule to calculation of the degree of accomplishment of the *condition* of a rule

3.3
activation

process by which the degree of fulfilment of a *condition* acts on an output fuzzy set

3.4
conclusion
consequent

output of a *linguistic rule*, i.e. the actions to be taken (the THEN part of an IF..THEN fuzzy control rule)

3.5
condition
antecedent

expression comprising *subconditions* combined with *fuzzy operators* AND, OR, NOT

3.6
crisp set
special case of a *fuzzy set*, in which the *membership function* only takes two values, commonly defined as 0 and 1

3.7
defuzzification
conversion of a *fuzzy set* into a numerical value

3.8
degree of membership
membership function value

3.9
fuzzification
determination of *degrees of membership* of the crisp input value of the *linguistic terms* defined with each input *linguistic variable*

3.10
fuzzy control
type of control in which the control algorithm is based on *fuzzy logic*

[IEV 351-17-51, modified]

3.11
fuzzy logic
collection of mathematical theories based on the notion of *fuzzy set*. *Fuzzy logic* is a kind of infinite-valued logic

3.12
fuzzy operator
operator used in *fuzzy logic* theory

3.13
fuzzy set
A *fuzzy set* A is defined as the set of ordered pairs $(x,\mu A(x))$, where x is an element of the universe of discourse U and $\mu A(x)$ is the *membership function*, that attributes to each $x \in U$ a real number $\in [0,1]$, describing the degree to which x belongs to the set

3.14
inference
application of *linguistic rules* on input values in order to generate output values

3.15
linguistic rule
IF-THEN rule with *condition* and *conclusion*, one or both linguistic

3.16
linguistic term
in the context of fuzzy control, *linguistic terms* are defined by *fuzzy sets*

3.17
linguistic variable
variable that takes values in the range of *linguistic terms*

3.18
membership function
function which defines the *degree of membership* over the universe of discourse for a given *fuzzy set*
[IEV 351-17-52, modified]

3.19
singleton
fuzzy set whose *membership function* is equal to one at one point and equal to zero at all other points

3.20
subcondition
elementary expression in the form of a variable or as term "*linguistic variable* IS *linguistic term*"

3.21
rule base
collection of *linguistic rules* to attain certain objectives

3.22
weighting factor
value between 0..1, that states the degree of importance, credibility, confidence of a *linguistic rule*

4 Integration into the programmable controller

The fuzzy control applications programmed in Fuzzy Control Language FCL according to clause 5 shall be encapsulated in Function Blocks (or Programs) as defined in IEC 61131-3. The concept of Function Block Types and Function Block Instances given in IEC 61131-3 applies to this standard.

The Function Block Types defined in Fuzzy Control Language FCL shall specify the input and output parameters and the fuzzy control specific rules and declarations.

The corresponding Function Block Instances shall contain the specific data of the fuzzy control applications.

Function Blocks defined in Fuzzy Control Language FCL may be used in Programs and Function Blocks written in any of the languages of IEC 61131-3, for example, Ladder Diagram, Instruction List, etc. The data types of the input and output parameters of the Function Block or Program written in FCL shall match those of the corresponding "calling environment" as illustrated in figure 1.

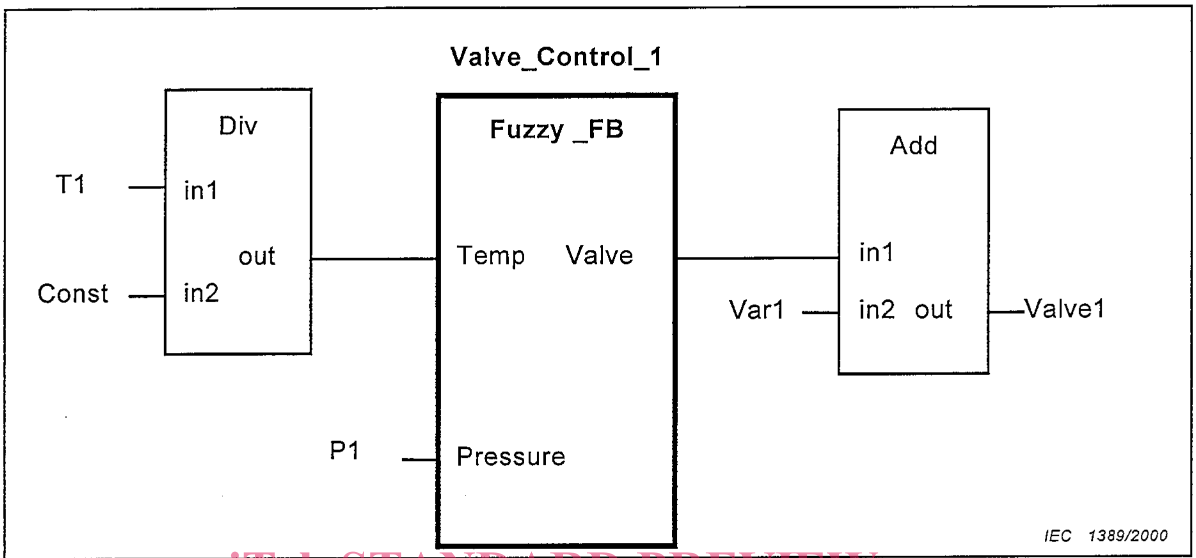


Figure 1 – Example of a fuzzy control Function Block in FBD representation

In this example, Valve_Control_1 is a user defined Function Block Instance of the Function Block Type Fuzzy_FB. The Function Block Type Fuzzy_FB may be programmed in Fuzzy Control Language FCL according to clause 5. The Function Block Fuzzy_FB is used here in a program or a Function Block which is represented in the graphical language FBD (Function Block Diagram) of IEC 61131-3.

5 Fuzzy Control Language FCL

5.1 Exchange of fuzzy control programs

The definition of the Fuzzy Control Language FCL is based on the definitions of the programming languages in IEC 61131-3. The interaction of the fuzzy control algorithm with its program environment causes it to be "hidden" from the program. The fuzzy control algorithm is therefore externally represented as a Function Block according to IEC 61131-3. The necessary elements for describing the internal linguistic parts of the fuzzy control Function Block like membership functions, rules, operators and methods have to be defined according to this clause.

The language elements of FCL standardize a common representation for data exchange among fuzzy control configuration tools of different manufacturers shown in figure 2. Using this common representation, every manufacturer of programmable controllers may keep his hardware, software editors and compilers. The manufacturer has only to implement the data interface into his specific editor. The customer would be able to exchange fuzzy control projects between different manufacturers.

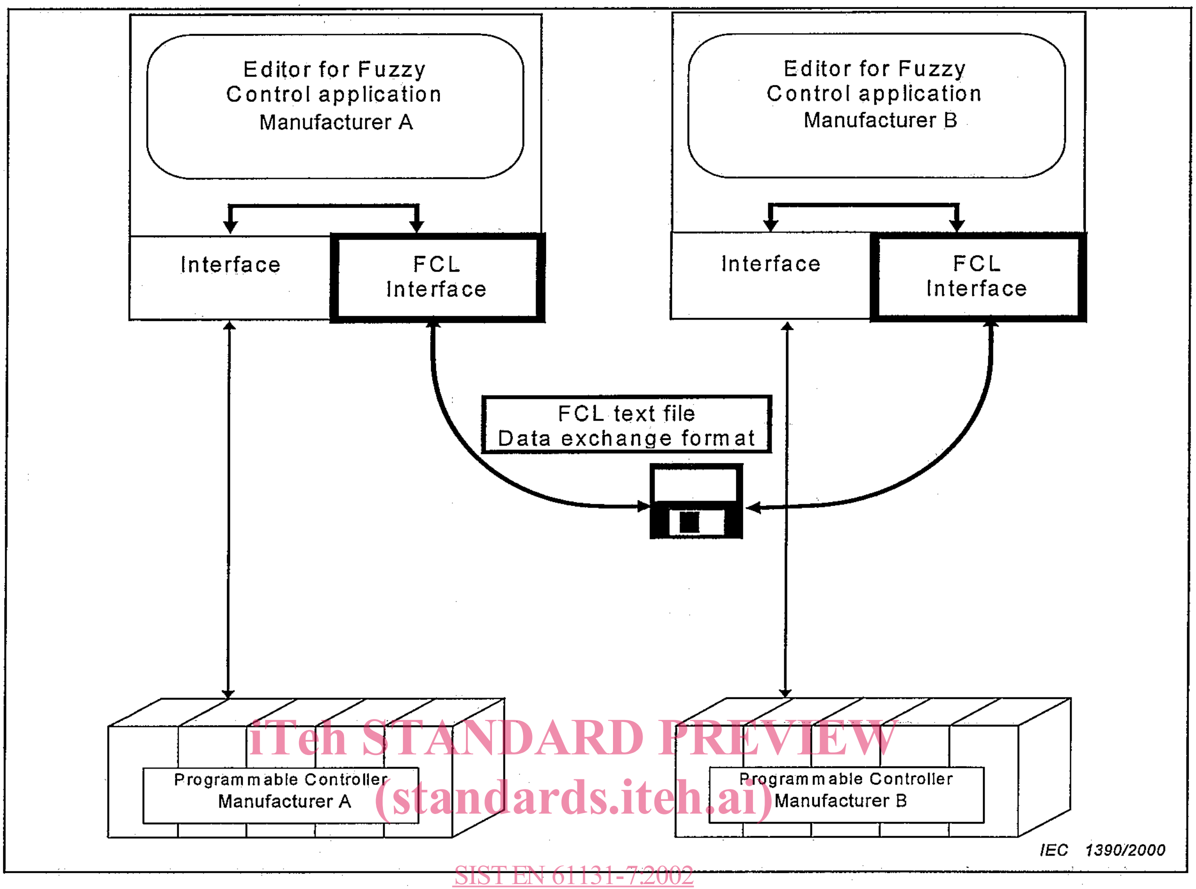


Figure 2 – Data exchange of Programs in Fuzzy Control Language (FCL)

5.2 Fuzzy Control Language elements

Fuzzy control language elements in this subclause are described using examples. The detailed production rule is given in 5.4.

5.2.1 Function Block interface

According to clause 4, the external view of the fuzzy Function Block requires that the following standard language elements of IEC 61131-3 be used:

FUNCTION_BLOCK *function_block_name*	Function block
VAR_INPUT *variable_name*: *data_type*; .... END_VAR	Input parameter declaration
VAR_OUTPUT variable_name: *data_type*; .... END_VAR	Output parameter declaration
VAR *variable_name: data_type*; END_VAR END_FUNCTION_BLOCK	Local *variables*

With these language elements, it is possible to describe a function block interface. The function block interface is defined with parameters which are passed into and out of the function block. The data types of these parameters shall be defined according to IEC 61131-3.

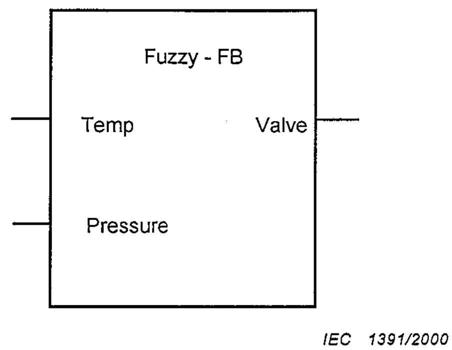
Figure 3 shows an example of a Function Block declaration in Structured Text (ST) and Function Block Diagram (FBD) languages.

```
FUNCTION_BLOCK Fuzzy-FB
  VAR_INPUT
     Temp: REAL;
     Pressure: REAL;
  END_VAR

  VAR_OUTPUT
     Valve: REAL;
  END_VAR

END_FUNCTION_BLOCK
```

Structured Text (ST)



Function Block Diagram (FBD)

Figure 3 – Example of a Function Block interface declaration in ST and FBD languages

5.2.2 Fuzzification

The values of the input variables have to be converted into *degrees of membership* for *the membership functions* defined on the variable. This conversion is described between the keywords FUZZIFY and END_FUZZIFY.


```
FUZZIFY variable_name
        TERM term_name:= membership_function;
        ....
END_FUZZIFY
```

After the keyword FUZZIFY, the name of a variable which is used for the fuzzification shall be named. This is the name of a previously defined variable in the VAR_INPUT section. This *linguistic variable* shall be described by one or more *linguistic terms*. The *linguistic terms* introduced by the keyword TERM described by *membership functions* in order to fuzzify the variable. A *membership function* is a piece-wise linear function. It is defined by a table of points.

```
membership_function ::= (point i), (point j), ...
```

Every point is a pair of the values of the variable and the membership degree of that value separated by a comma. The pairs are enclosed in parentheses and separated by commas.

```
point i ::= value of input i | variable_name of input i , value i of membership degree
```

With this definition, all simple elements, for example ramp and triangle, may be defined. The points shall be given in ascending order of variable value. The membership function is linear between successive points. The degree of membership for each term is therefore calculated from the crisp input value by the linear interpolation between the two relevant adjacent membership function points.

The minimum number of points is two. The maximum number is restricted according to clause 6 conformance classes.