The 2004 47 th Midwest Symposium on Circuits and Systems

MWSCAS 2004

Hiroshima, Japan
July 25-28, 2004

Conference Proceedings
Volume III of III

IEEE

Sponsored by the IEEE
Circuits and Systems Society
and hosted by Hiroshima University

CIRCUITS AND SYSTEMS SOCIETY

MWSCAS 2004

The 2004 47th Midwest Symposium on Circuits and Systems

Conference Proceedings

Volume III of III

Hiroshima, Japan

July 25-28, 2004

IEEE

CIRCUITS AND SYSTEMS SOCIETY

Sponsored by the IEEE
Circuits and Systems Society
Hosted by Hiroshima University

Proceedings of The 2004 47th Midwest Symposium on Circuits and Systems

IEEE Catalog Number: 04CH37540

ISBN: 0-7803-8346-X

ISSN: 1548-3746

Additional Copies of this Publication are Available From:

IEEE Service Center
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
1-800-678-IEEE

The 47th IEEE International Midwest
Symposium on Circuits and Systems

Functionally Complete Element for Fuzzy Control Hardware Implementation

V. Varshavsky
Bar Ilan University,
Israel
vviktor@zahav.net.il

V. Marakhovsky
The University of Aizu,
965-8580, Japan
marak@u-aizu.ac.jp

I. Levin
Bar Ilan University,
Israel
i.levin@ieee.org

N. Kravchenko
Bar Ilan University,
Israel
natali_kra@list.ru

***Abstract*-It is proposed to implement fuzzy control devices as multi-valued logic functions, using directly analog input variables and forming analog values of output variables. CMOS summing amplifiers are used as basic elements for designing appropriate circuits. It has been proved that a summing amplifier is a functionally complete element in arbitrary-valued logic. In a plenty of cases this approach enables principally simplification of fuzzy logic controllers for a broad class of applications. Some fuzzy control hardware implementations using this design approach are given.**

I. Introduction

Fuzzy Logic Control is a methodology bridging Artificial Intelligence and traditional Control Theory. Fuzzy controllers have extraordinary wide application in the cases when accuracy of control is not of high necessity or importance.

On the other hand, as it is stated in [1], "Fuzzy Logic can address complex control problems, such as robotic arm movement, chemical or manufacturing process control, antiskid braking systems, or automobile transmission control with more precision and accuracy, in many cases, than traditional control techniques have.... Fuzzy Logic is a methodology for expressing operational laws of a system in linguistic terms instead of mathematical equations."

The conventional fuzzy logic methodology [2, 3] comprises three phases:

1. *Fuzzifycation:* It is transformation of analog (continuous) input variables to linguistic ones, e.g. transformation of temperature into the terms "cool", "warm", "hot" or that of speed into the terms "negative big (NB)", "negative small (NS)", "zero (Z)", "positive small (PS)", "positive big (PB)". Such transformation is realized by introduction of so-called membership functions. Both a range of linguistic values and a degree of their participation on intervals between adjacent values defines these membership functions. For linguistic variables, it is important not only which membership function a variable belongs to, but also a relative degree to which it is a member, i.e. a variable can have a weighted membership in several membership functions at the same time.

2. *Inference:* A connection between input and output variables, i.e. fuzzy controller behavior specification, is determined by the system of rules of "IF... THEN ..."-type. For instance: "IF the temperature is "worm" THEN the speed is PS" or "IF the speed is NB THEN force is ZERO", etc. Since input linguistic variables are weighted, the output variables can be obtained weighted as well.

3. *Defuzzification:* Weighted values of output linguistic variables, obtained due to fuzzy inference, have to be transformed to analog (continuous) variables. This procedure is also based on membership functions. Two major methods are used for defuzzyfication:

- maximum defuzzification method, wherein an output value is determined by the linguistic variable with the maximum weight;
- centroid calculation defuzzification method, wherein an output value is determined by the weighted influence of all the active output membership functions.

According to our concept, for a broad class of fuzzy controllers, a specification is a table connecting values of input and output linguistic variables. Usually membership functions of the output variables evenly divide the ranges of their variations. (Often membership functions can be brought to even scale by increasing the number of gradations.) Therefore, specification tables represent nothing but tables determining a specific multi-valued logical function. Moreover, for a number of implementations one can possibly try to neglect weighting and determining values of input linguistic variables and to use continuous-valued variables.

The above idea was in the focus of our research. We dealt with searching for and investigating of such basic multi-valued functions, which, from the one hand, would present a complete functional basis in the multi-valued logic, and from the other hand, could be efficiently implemented by CMOS technology.

II. Summing Amplifier as a Multi-Valued Logical Element

Summing amplifier's behavior, accurate to the members of the infinitesimal order that is determined by the amplifier's gain factor in disconnected condition (Fig.1), is described as follows:

$$V_{out} = \begin{cases} V_{dd} & \text{if } \sum_{j=1}^{n} \frac{R_0}{R_j}(V_j - \frac{V_{dd}}{2}) \le -\frac{V_{dd}}{2} \\ \frac{V_{dd}}{2} - \sum_{j=1}^{n} \frac{R_0}{R_j}(V_j - \frac{V_{dd}}{2}) & \text{in other cases} \\ 0 & \text{if } \frac{V_{dd}}{2} \le \sum_{j=1}^{n} \frac{R_0}{R_j}(V_j - \frac{V_{dd}}{2}) \end{cases} \quad (1)$$

where V_{dd} - the supply voltage, V_j - the voltage on j[th] input, R_j - the resistance of j[th] input, R_0 - the feedback resistance, and $V_{dd}/2$ - the midpoint of the supply voltage.

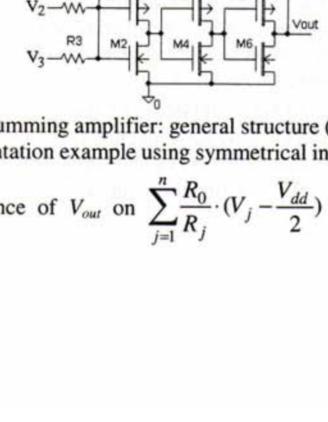

Fig. 1. Summing amplifier: general structure (a); CMOS implementation example using symmetrical invertors (b).

Dependence of V_{out} on $\sum_{j=1}^{n} \frac{R_0}{R_j} \cdot (V_j - \frac{V_{dd}}{2})$ is shown in Fig.2,a.

0-7803-8346-X/04/$20.00 ©2004 IEEE

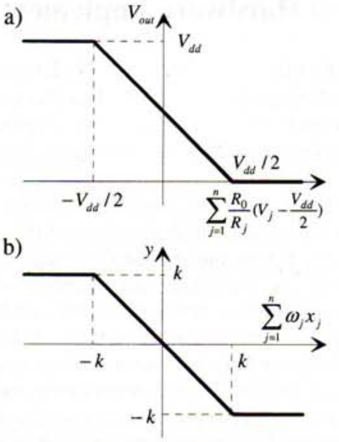

Fig. 2. Summing amplifier's behavior: within voltage coordinates (a); within multi-valued variables coordinates (b).

Let us split the source voltage V_{dd} on $m = 2k+1$ voltage levels. Then replacing the input voltages V_j by m-valued logical variables $x_j = \frac{2\cdot V_j - V_{dd}}{V_{dd}}\cdot k$, the output voltage V_{out} by m-valued variable y and designating $R_0/R_j = \omega_j$, the system (1) can be represented as (2). Graphical view of (2) is shown in Fig.2,b.

$$y=\begin{cases} +k & \text{if} \quad \sum_{j=1}^{n}\omega_j\cdot x_j \le -k \\ -\sum_{j=1}^{n}\omega_j\cdot x_j & \text{if} \quad k > \sum_{j=1}^{n}\omega_j\cdot x_j > -k \\ -k & \text{if} \quad \sum_{j=1}^{n}\omega_j\cdot x_j \ge +k \end{cases} \tag{2}$$

Later on, we will call the functional element, whose behavior is determined by the system (2), a multi-valued threshold element. In the simplest case when $\omega_j = 1,\ j = 1,2,3$, we will call it the majority element and designate as $maj(x_1, x_2, x_3)$.

III. Functional Completeness of a Threshold Element in Multi-Valued Logic

The basic operation (or set of basic operations) is called functionally completed in arbitrary-valued logic, if any function of this logic can be represented as superposition of basic operations.

There are some known functionally completed sets of functions. It is clear, that for proving functional completeness of some new function it is sufficient to show that the functions of the known functionally completed set can be represented as superposition of the considered function. One of functionally completed functions in m-valued logic is the Webb's function [4]:

$$w(x,y) = [\max(x,y)+1]_{\bmod m}. \tag{3}$$

Therefore, for proving functional completeness of threshold operation in multi-valued logic it is sufficient to show how the Webb's function can be represented through this operation.

First, let us represent the function $\max(x_1, x_2)$ by threshold functions. To do this let us consider the function $f_a(x)$ diagram, such as

$$f_a(x) = \max(x,a) = \begin{cases} a & \text{if} \quad a \ge x \\ x & \text{if} \quad x > a \end{cases}. \tag{4}$$

This function diagram is shown in Fig.3,a.

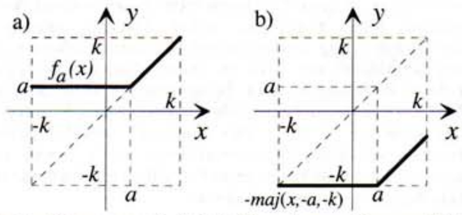

Fig. 3. Diagrams of $f_a(x)$ (a) and $-maj(x,-a,-k)$ (b) functions.

The $-maj(x,-a,-k)$ function diagram is shown in Fig.3,b. Actually, as far as $x < a$ $\quad x-a-k < -k$ and $-maj(x,-a,-k) = -k$. Note that for all x values, $f_a(x) - maj(x,-a,-k) = a-k$ as it follows from Fig.2, hence

$$f_a(x) = -maj[maj(x,-a,-k), a, -k]. \tag{5}$$

Taking into consideration that $-maj(a,b,c) = maj(-a,-b,-c)$, it follows from (5) that

$$\max(x_1,x_2) = maj(maj(-x_1,x_2,k),-x_2,k). \tag{6}$$

Now let us consider the function $(x+1)_{\bmod m}$ representation by threshold functions. To make it clear let us turn to the sequence of pictures Fig.4.

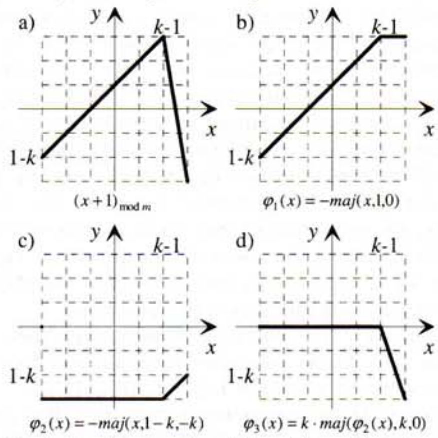

Fig. 4. Implementation of $(x+1)_{\bmod m}$ function.

From Fig.4 it is easy to see that

$$(x+1)_{\bmod m} = \varphi_1(x) + 2\varphi_3(x)$$

and obviously, this function can be implemented on threshold elements too. Hence, the functional completeness of the summing amplifier in arbitrary-valued logic is shown.

It is clearly seen from the Fig.4 that the above constructions should be considered as only the way to prove functional completeness, but in no circumstances as a synthesis method. The methods of synthesizing circuits in the proposed base are to be developed in future. However, as it will be shown below, for a number of real circuits the proposed base allows designing simple and efficient circuits.

IV. Fuzzy Devices as Multi-Valued and Analog Circuits

Fuzzy devices or fuzzy controllers implementing fuzzy control functions usually have the structure shown in Fig.5.

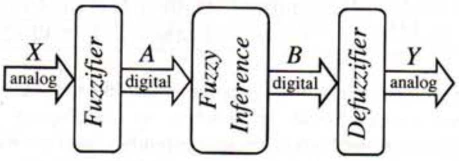

Fig. 5. Fuzzy device structure.

Analog variables $X = \{x_1, x_2, ..., x_n\}$ go the fuzzy device input. Fuzzifier converts a set of analog variables x_j into that of weighted linguistic (digital) variables $A = \{a_1, a_2, ..., a_n\}$ of the type *"variable x_j has an average positive value with the weight of 0.3 and a large positive value with the weight of 0.4"*.

Fuzzy Inference block, based on the fuzzy rules like *"if a_j has an average positive value and a_i has a small negative value, b_k has a small positive value"*, generates a set of weighted linguistic variables values $B = \{b_1, b_2, ..., b_k\}$.

Defuzzifier converts a set of weighted linguistic (digital) variables $B = \{b_1, b_2, ..., b_k\}$ into a set of output analog variables $Y = \{y_1, y_2, ..., y_k\}$.

As a rule, Fuzzifier and Defuzzifier are implemented as AD and DA (analog-digital and digital-analog) converters, i.e. by hardware implementation. Fuzzy inference is usually implemented as microprocessor software.

On the other hand, there is the set of output analog variables, which values unambiguously corresponds to each set of input analog variable values; hence a Fuzzy Device could be specified as a functional analog of signal converter

$$Y(X) = \{y_1(X), y_2(X), ..., y_k(X)\}$$

and its output Y determines a system of n-dimensional surfaces. Let us consider the possibility of specification of these surfaces as that of their piecewise-linear approximation.

Let $m = 2k + 1$ linguistic variable a_j values[1] correspond to analog variable x_j. Then basing on fuzzy rules system, we can specify a system of m-valued logic functions, as follows:

$$B(a_1, a_2, ... a_n) = \{b_1(A), b_2(A), ..., b_k(A)\}. \quad (7)$$

Note that most publications describing fuzzy controllers contain the tables, specifying fuzzy controllers' behavior as (7) and a plenty of publications contain piecewise-linear approximations of the corresponding surfaces.

The apparent conclusion can be made from the things mentioned above: if a fuzzy controller is represented as (7), it can be implemented as superposition of multi-valued threshold elements. In this case, owing to linear behavior of the threshold element in the zone between the saturation levels ((2) and Fig.2,b) natural linear approximation appears between the discrete points of specification.

[1] An odd number of linguistic variable values do not break the generality of consideration.

V. Implementation of Fuzzy Controllers as Threshold Elements Circuits

A. *Example 1*

This example is taken from [5]: *"Design of a Rule-Based Fuzzy Controller for the Pitch Axis of an Unmanned Research Vehicle"*.

The fuzzy control rules for the considered device depend on the error value $e = ref - output$ and changing of error $ce = \dfrac{old\ e - new\ e}{sampling\ period}$. Fuzzifier brings seven levels of linguistic variables for each (NB – negative big (-3); NM – negative middle (-2); NS – negative small (-1); ZO – zero (0); PS – positive small (1); PM – positive middle (2); PB – positive big (3)) in correspondence with error and changing of error values. The output has the same seven gradations. At that defuzzification of NB value generates $V_{out} = 0\text{V}$, that of ZO value generates $V_{out} = 1.75\text{V}$ and that of PB value generates $V_{out} = 3.5\text{V}$, if the supply voltage $V_{dd} = 3.5\text{V}$. The corresponding 49 fuzzy rules are represented in Table 1.

TABLE 1
TABLE OF FUZZY RULES

		Error (e)						
Change of error (ce)		NB	NM	NS	ZO	PS	PM	PB
	NB	ZO	PS	PM	PB	PB	PB	PB
	NM	NS	ZO	PS	PM	PB	PB	PB
	NS	NM	NS	ZO	PS	PM	PB	PB
	ZO	NB	NM	NS	ZO	PS	PM	PB
	PS	NB	NB	NM	NS	ZO	PS	PM
	PM	NB	NB	NB	NM	NS	ZO	PS
	PB	NB	NB	NB	NB	NM	NS	ZO

Table 2 represents Table 1 as function of 7-valued logic.

TABLE 2
TABLE OF THE 7-VALUED FUNCTION

		Error (e)						
Change of error (ce)		-3	-2	-1	0	1	2	3
	-3	0	1	2	3	3	3	3
	-2	-1	0	1	2	3	3	3
	-1	-2	-1	0	1	2	3	3
	0	-3	-2	-1	0	1	2	3
	1	-3	-3	-2	-1	0	1	2
	2	-3	-3	-3	-2	-1	0	1
	3	-3	-3	-3	-3	-2	-1	0

It is seen from Table 1 and Table 2 that the function is symmetric with respect to "North-West – South-East" diagonal and depends on $e - ce$. This kind of dependence is shown in Fig.6.

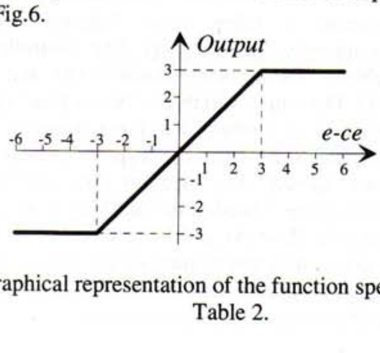

Fig. 6. Graphical representation of the function specified by Table 2.

It apparently follows from comparison of Fig.2 and Fig.6 that in order to reproduce the function specified by Table 2 it is sufficient to have one two-input summing amplifier and one inverter. Note that inversion of logic variables lying within $-k \div +k$ interval is the operation of diametric negation $\bar{x} = -x$; the operation $\overline{V_{out}} = V_{dd} - V_{in}$ corresponds to it in the space of summing amplifiers' output voltages. Thus CMOS circuit containing 12 transistors and 5 resistors, which implements our function, is shown in Fig.7.

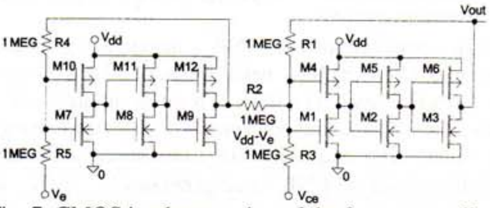

Fig. 7. CMOS implementation of the fuzzy controller specified by Table 2.

In this circuit, transistors M7-M12 compose the main body of the summing amplifier used as an inverter; transistors M1-M6 compose the main body of the summing amplifier.

B. Example 2

This example is taken from [6]: "Manipulator for Man-Robot Cooperation (Control Method of Manipulator/Vehicle System with Fuzzy Inference)".

The manipulator has two force/torque sensors. One of them (F_h) is the "operational force sensor"; the other (ω) is "the environmental force sensor". Both of the variables represented the sensors and the output variable has three linguistic values – *S* (small), *M* (middle) and *B* (big). Table 3 defines fuzzy rules and corresponding ternary logical function.

TABLE 3
TABLE OF TERNARY FUNCTION

		F_h		
		S (-1)	*M* (0)	*B* (1)
ω	*S* (-1)	*B* (1)	*B* (1)	*B* (1)
	M (0)	*S* (-1)	*M* (0)	*B* (1)
	B (1)	*S* (-1)	*S* (-1)	*S* (-1)

It can be simply proved by trivial substitution that $Output = maj(2\omega, -F_h, 0)$ and CMOS implementation coincides with the circuit of Fig.7, therein $V_e = V_{F_h}$, $V_{ce} = V_{\omega}$ and $R3 = 0.5\,\text{MEG}$.

C. Fuzzy Controller for Washing Machine

This example is taken from Aptronix Incorporated (http://www.aptronix.com/fuzzynet). The controller has two input variables: "Dirtiness of clothes" (*X*) and "Type of dirtiness" (*Y*). The output variable is "Wash time" (*Z*).

The "Dirtiness of clothes" has three degrees: Large (*L*), Medium (*M*), and Small (*S*). The "Type of dirtiness" also has three degrees: Greasy (*G*), Medium (*M*), and Not Greasy (*NG*). Degrees of the "Wash time" are Very Long (*VL*), Long (*L*), Medium (*M*), Short (*S*), and Very Short (*VS*).

Table 4 determines functioning of the fuzzy controller in term of linguistic variables and multi-valued logic variables at the same time.

TABLE 4
MATRIX OF LINGUISTIC AND MULTI-VALUED VARIABLES

Wash time (*Z*)		Dirtiness of clothes (*X*)		
		S (-1)	*M* (0)	*L* (1)
Type of dirt. (*Y*)	*NG* (-1)	*VS* (-2)	*S* (-1)	*M* (0)
	M (0)	*M* (0)	*M* (0)	*L* (1)
	G (1)	*L* (1)	*L* (1)	*VL* (2)

This matrix cannot be implemented directly as in the previous cases. Some approaches to developing matrix decomposition methods were suggested but they are the topic for the next paper. Here we only stated that this controller could be built using fore multi-valued threshold elements, one of which is inverter.

VI. CONCLUSION

In the above examples of controllers, push-pull summing amplifiers are used. The summing amplifier however does not have to be push-pull type. It may be differential type or any other types of operational amplifiers.

The proof of functional completeness shown in section 2 provides the possibility of implementing arbitrary function of multi-valued logic in the base of summing amplifiers. However, none mentioned above answers the question concerning the efficiency of such implementation and the techniques of synthesizing circuits in the offered base. Though the given examples show possible high efficiency and effectiveness of the implementation offered, this is true only in regard to specific examples, chosen in special way.

Techniques of synthesizing fuzzy devices in the offered base and the problems of implementability under the conditions of real production should be resolved on further work stages.

REFERENCES

[1] "*Implementation of Fuzzy Logic*", Texas Instruments, SPRA028, January 1993.

[2] "*An Introduction to Fuzzy Logic Applications in Intelligent Systems*" (Kluwer International Series in Engineering and Computer Science, 165) by Ronald R. Yager, Lotfi A. Zadeh (Editor), Kluwer Academic Publishers, January 1992, 356 p.

[3] "*Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems*": Selected Papers by Lotfi A. Zadeh (Advances in Fuzzy Systems - Applications and Theory), by George J. Klir (Editor), Bo Yuan (Editor), Lotfi Asker Zadeh, World Scientific Pub Co; Vol. 6, June 1996, 826 p.

[4] Post E., "Introduction to a general theory of elementary propositions", *Amer. J. Math.*, 43, 1921, pp. 163-185.

[5] Deepak Sabharwal and Kuldip S. Rattan, "Design of a Rule-Based Fuzzy Controller for the Pitch Axis of an Unmanned Research Vehicle", Department of Electrical Engineering, Wright State University, Dayton, OH 45435, CH3158-3/92/0000-0449©1992 IEEE, *Fuzzy Control Systems*, CRC Press LLC, Abraham Kandel (Edited by), Lotfi A. Zadeh (Foreword by), September 1993, pp.81-86.

[6] Yoshio Fujisawa, Toshio Fukuda, Kazushiro Kosuge, Fumihito Arai, Eiji Muro, Haruo Hoshino, Takashi Miyazaki, Kazuhiko Ohtsubo, Kazuo Uehara, "Manipulator for Man-Robot Cooperation (Control Method of Manipulator/Vehicle System with Fuzzy Inference)", 0-7801-0582-5/92 ©1992 IEEE", *Fuzzy Control Systems*, CRC Press LLC, Abraham Kandel (Edited by), Lotfi A. Zadeh (Foreword by), September 1993, pp.168-172.

[7] V. Varshavsky, V. Marakhovsky, I. Levin, and N. Kravchenko, "*Fuzzy Device*", the application number is 2003-190073, filed to Japan's Patent Office, Feb. 6, 2003.