Fifth IFSA World Congress (1993), 1394-1397

A Fuzzy Microprocessor for Real-time Control Applications

Takeshi Katashiro

Electric Devices Group, OKI Electric Industry Co., Ltd.
550-1,Higashiasakawa-cho,Hachioji-shi Tokyo 193 Japan

ABSTRACT

A Fuzzy Microprocessor(FMP) is presented, which is suitable for real-time control applications.The features include high speed inference of maximum 114K FLIPS at 20MHz system clocks, capability of up to 128-rule construction, and handling of 8 input variables with 8-bit resolution. In order to realize these features, the fuzzifier circuit and the processing element(PE) are well optimized for LSI implementation. The chip fabricated in 1.2 μm CMOS technology contains 71K transistors in 82.8 mm^2 die size and is packaged in 100-pin plastic QFP.

Key words;

Fuzzy MicroProcessor(FMP), Real-time Control, Fuzzifier Circuit, Processing Element(PE), Floating Point Arithmetic, Pipelining Operation

1. INTRODUCTION

Fig.1 shows the application area of the FMP. Fuzzy logic application to appliances is implemented with software in most cases. However, for the applications to real-time control such as motor control and engine control of automobile etc., there is a limit to software implementation because of a demand for their high-speed performance.

Such real-time control applications require more than several kilo FLIPS and more than 20 rules in many cases. LSI implementation of a fuzzy processor is essential for existing high-speed real-time control applications. Therefore, several LSI implementations are introduced[1],[2]. If all functional units to execute a fuzzy inference are incorporated into a chip, the area is too large to be applied to an actual target system ,for example,392 mm^2 consisted of 2 chips [1]. Several circuit architectures to solve the problem are proposed e.g.[3].

This paper discusses the circuit architecture to optimize the chip area.

2. CIRCUIT ARCHITECTURE

Fig.2 shows the schematic block diagram of the FMP which incorporates 7 kinds of functional units, viz., membership function memory (MF MEM), rule memory

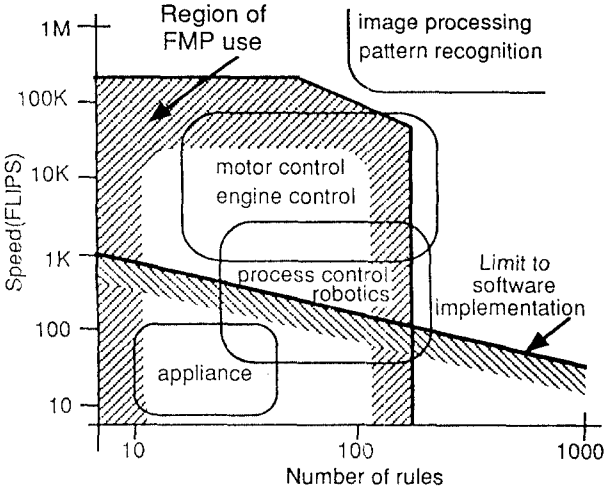


Fig.1 Application area of the FMP

(RULE MEM), fuzzifier(FUZZIFIER), processing element (PE), defuzzifier(DEFUZZIFIER), CPU interface bus , and system controller. At start-up time, the membership functions and knowledge base rules are down-loaded into the memory(MF MEM and RULE MEM) via the CPU interface bus. The FMP starts by accepting input signals, and after finishing inference, it sends the output signals (deterministic value) back to a host CPU. A processing element (PE) executes a fuzzy inference which is available for both MIN-MAX and MIN-SUM methods. Deterministic value is calculated by weighted mean arithmetic. The weight memory(WEIGHT MEM) stores "Weights" of all the rules. It aims to be used for evaluation of knowledge base rules.

The features of circuit configuration include 2 fuzzifiers and 8 processing elements(PEs) which operate in parallel. The major architectures are,

(1). In order to reduce on-chip memory capacity, the fuzzifier circuit is designed to calculate fuzziness value by using the parameters of membership function.

(2). In order to minimize cell areas , a processing element(PE) is designed to function as both MAX operator and MIN operator in the same circuit.

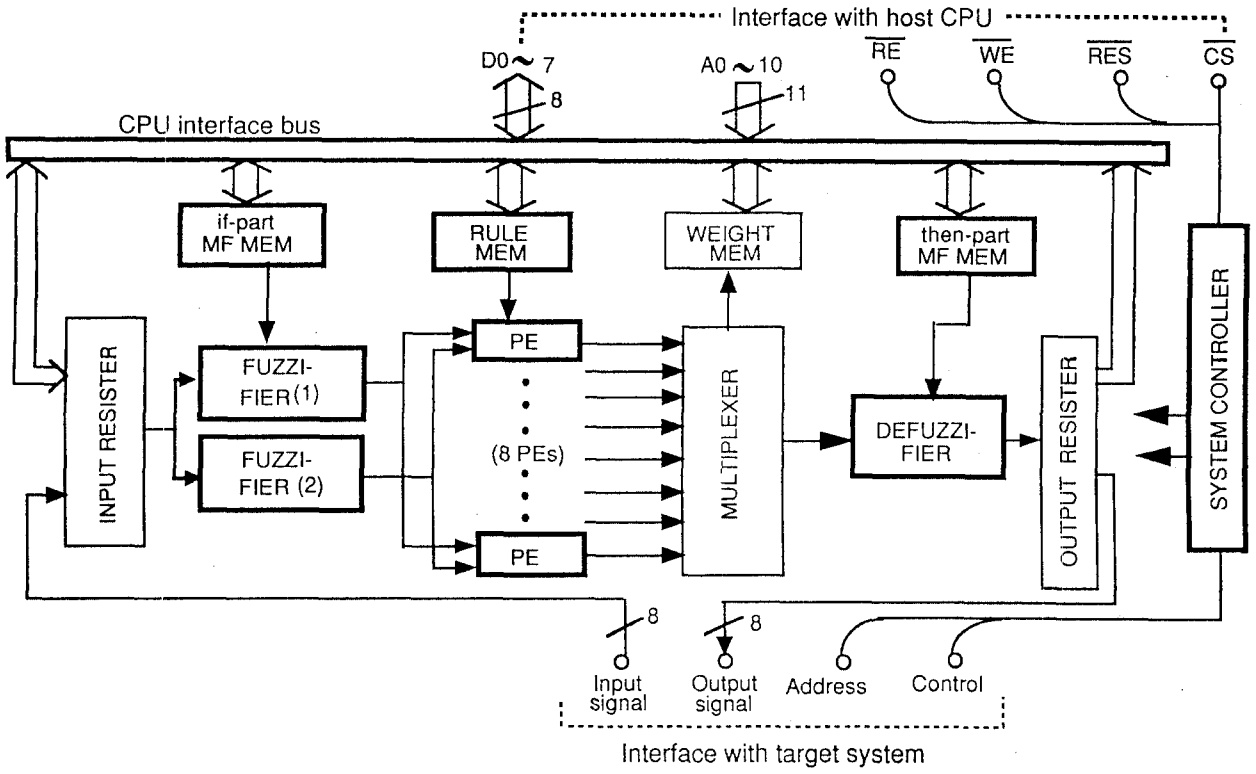


Fig.2 Schematic block diagram of the FMP

A. Fuzzifier Circuit

Fig.3 shows the membership function of input variable. The membership function{Fig.3 (A)} composed of 8 terms is divided into 2 groups{Fig.3 (B)&(C)} which are generated by "FUZZIFIER(1)" and "FUZZIFIER(2)".

The shape of terms consists of 4 kinds; trapezoid, triangle, Z, and S shape,as shown in Fig.4(A). The trapezoid shape is basic, and the others are based on the trapezoid shape. The trapezoid shape has 4 kinds of parameter ; a_n, b_n, μ_n, and β_n (n;term number), as shown in Fig.4(B). The a_n and b_n represent inflection positions of the trapezoid shape, and the μ_n and β_n represent the slopes. Given the input signal x, fuzziness value G(x) is calculated by,

$$G(x) = \begin{cases} \mu_n(x-a_n), & a_n \le x < b_n, \text{ and} \\ 1+\beta_n(x-b_n), & b_n \le x \end{cases}$$

The results are limited to 1 from 0 because of the definition of membership function. The calculation is executed by floating point arithmetic to adjust precisely the slope of the membership function. It aims at accurate tuning-up of target controllers.

Triangle, Z, and S shape are obtained by setting $\mu_n(b_n-a_n)$ to 1, by setting μ_n to the maximum value (actually equal to 255), and by setting β_n to 1 , respectively.

Fig.5 show the fuzzifier unit. The fuzzifier operates in sequence,viz.,

1st stage: calculation of $x-a_n$ and $x-b_n$,
2nd stage: calculation of $\mu_n(x-a_n)$ and $1+\beta_n(x-b_n)$,
final stage: floating point arithmetic.

One of the 2 fuzzifiers performs the above equation against n=1,3,5,7, and the other performs against n=2,4,6,8. "FULL ADD" in combination with "2s' COMPL" functions as a comparator. It compares x with a_n(or b_n) sequentially against n=1,2,3 ~ 8. When the output of "FULL ADD" changes from positive to negative level, calculation of $(x-a_n)$ and $(x-b_n)$ finishes on the 1st stage. And "COUNTER" outputs m which means the term number to be used for fuzzification of x. On the 2nd stage, calculation of $\mu_n(x-a_n)$ and $1+(x-b_n)$ are executed against n=m on "16-bit sift type MULTIPLIER". After fioating point operation, finally the fuzzifier transfers the fuzziness value Gm(x) to the next circuits(PEs). "LIMITER" in combination with "2s' COMPLE" limits Gm(x) to 1 from 0.

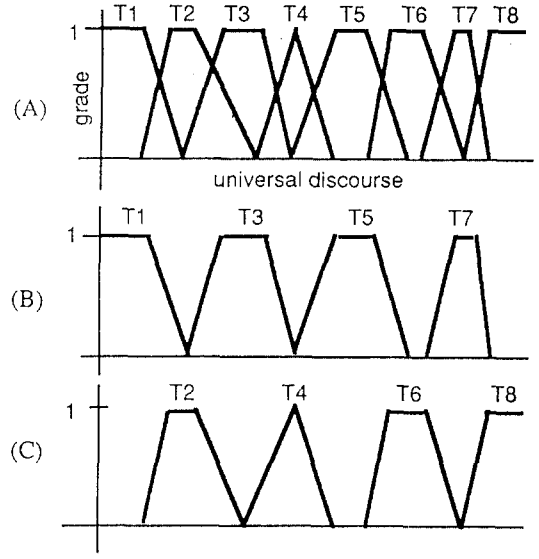


Fig. 3 Membership function of input variable

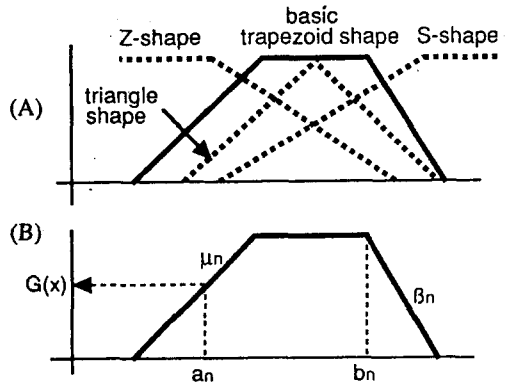


Fig.4 Shape of term

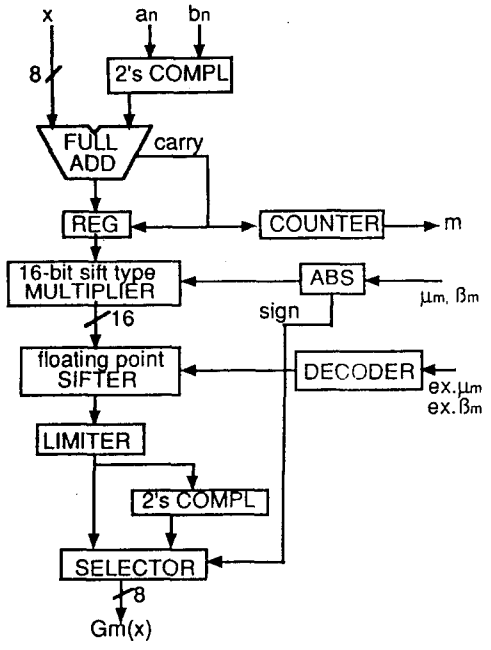


Fig.5 Fuzzifier unit

B. Processing Element

The FMP employs 8 processing elements (PEs), each of which processes a rule-set with 4 rules. Fig 6 shows the circuit configuration of the PE which functions as both MIN and MAX operator.The solid lines and the dotted lines make the circuit configuration into MIN and MAX operator, respectively. The "FULL ADD" in combination with the "2s' COMPLEMENT" functions as a comparator, and the carry-out " C " and " $\overline{C}$ " are valid at MAX and MIN operation respectively.

The "n" transferred from the rule memory(RULE MEM) is the term number defined by the knowledge base rules. Given the fuzziness value Gm(xi) (i =1,2,3∼ 8), n is compared with m. Then fuzziness value Gn(xi) is obtained by,

$$Gn(xi) = \begin{cases} Gm(xi), & n=m \\ 0, & n \neq m \end{cases} \quad \text{and}$$

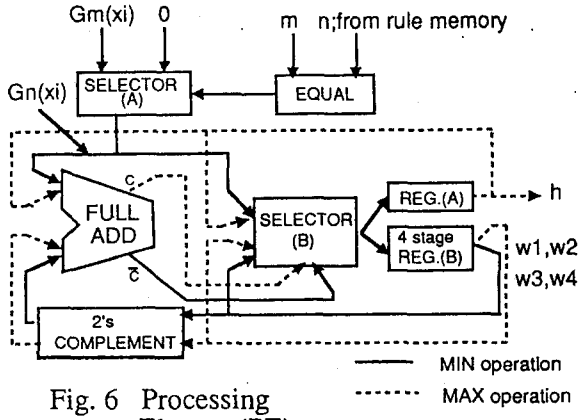


Fig. 6 Processing Element(PE)

MIN and MAX operations are sequentially repeated. Fig 7 shows the sequence of MIN and MAX operation in the case of a rule-set with 4 rules. The result "h" (=w1∨ w2∨ w3∨ w4) is transferred to the "DEFUZZIFIER" via the "MULTIPLEXER". The "h" decides level of the singleton function, which is adopted as membership function of output variable at the FMP.

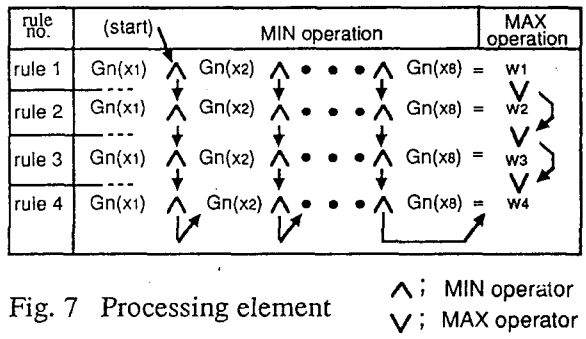


Fig. 7 Processing element

3. SUMMARY

Table 1 lists up the memory capacity. The improvement of the fuzzifier allows 75% reduction of the membership function memory capacity in comparison with our previous prototype.

Table 2 lists up the number of system clock cycles. The FMP performs pipelining operation, among 3 stages in "FUZZIFIER" every 8 clock cycles, and between "PE" and "DEFUZZIFIER" every 72 clock cycles. The FMP processes a rule-set with 32 rules by using 72 clock cycles. As it can process up to 4 rule-sets , the maximum number of rules is 128.

The FMP has an interface compatible with existing computer systems , IBM-PC/AT, intel 80 series/ Motorola 68 series microcomputers. For the FMP applications, we have also developed a fuzzy developing system running on IBM-PC/AT. Now the system is successfully at work on a controller of a sheet glass production line.

The specifications are listed up in Table 3, which are useful in numerous areas of real-time fuzzy controllers. The FMP matches well the applications to servo motor control, engine control, robotics, process control, motion control, navigation system, information processing, and communication control.

Rule	4096-bit SRAM
Membership function input output	3072-bit SRAM 256-bit SRAM
Weight	1024-bit SRAM
The others	128-bit resister

Table 1. Memory capacity

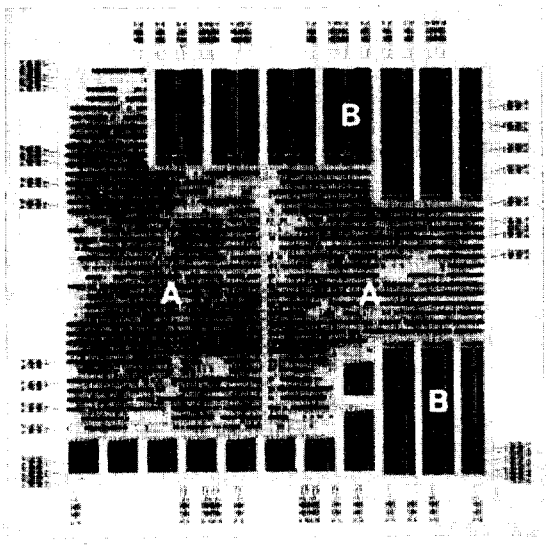
(unit; Hz)

FUZZIFIER			PE		DEFUZZIFIER	
1st stage	2nd stage	final stage	MIN	MAX	mult./add.	division
8	8	8	64	8	64	8
24			72		72	

Table 2. Number of clock cycles

Floating point:	8-bit with 3-bit exponent for membership function of input variable
Resolution:	8-bit
Inference speed:	114K FLIPS with 32 rules 51K FLIPS with 128 rules
Number of rules:	up to 128
Number of variables:	8 (input), 4 (output)
Process technology:	1.2μm double metal CMOS
Number of transistors:	71K
Die size:	9.1mm × 9.1 mm
Power supply:	3 to 6V
Power consumption:	200mW at 5V
Operating temperature:	− 40 to 85 °C
Package:	100-PIN plastic QFP

Table 3. Specifications of the FMP



A; Cell and Random logic circuit area
B, SRAM area

Fig. 8 Microphotograph of the FMP

ACKNOWLEDGMENTS

The author would like to thank Dr. Ueno and Dr. Sasaki of Kumamoto University for the idea of circuit architectures. The CAD was supported by ZENIC corporation, Kyoto.

REFERENCES

[1]. Sasaki,M.et.al.;"Implementation of A Fuzzy Inference Processor" Proc.of the 6th FUZZY System Symposium,pp.117-120, Tokyo 1990

[2]. Ikeda,H.et.al.; "A Fuzzy Inference Coprocessor Using a Flexible Active-Rule Driven Architecture", Proc. of the 1st int. Conf. on Fuzzy Systems FUZZ-IEEE '92,pp537-544

[3]. Sasaki,M.et.al. ;"Implementation of A Fuzzy Controller with Realization of High Inference Speed and Reduction of Circuitries", Proc. of the 7th Fuzzy System Symposium, pp.153-156, Tokyo 1991

ADDRESS

Takeshi Katashiro
Electric Devices Group
OKI Electric Industry Co.,Ltd.
Phone +81-426-63-1111,ext.3451
Fax +81-426-65-6536