

퍼지 컴퓨터

오 경 환

(서강대 전자계산학과 조교수)

기존의 이진논리는 애매모호한 인간의 지식을 표현하는 데 많은 어려움이 있었다. 컴퓨터의 사고를 보다 인간에 가깝게 하기 위해 0과1의 이진논리가 아닌, 0과1 사이의 실수로 애매모호함을 표현하는 Zadeh의 퍼지집합이론이 제안되었다. 이를 기초로 하여, 실제로 여러 종류의 퍼지 연산들을 수행하는 퍼지프로세서들이 개발되었으며, 퍼지 컴퓨터를 실현시키기 위한 연구가 활발히 진행되고 있다.

본고에서는 퍼지논리에 기초하여 퍼지정보처리(Fuzzy Information Processing)을 수행하는 대표적인 하드웨어 시스템인 퍼지 컴퓨터와 퍼지 컨트롤러(fuzzy controller)에 대해 알아보고, 다단계 퍼지 추론을 수행하는 퍼지 메모리 모듈(fuzzy memory module)의 기본인 퍼지 플립플롭에 대해 알아보고자 한다.

1. 퍼지 컴퓨터(Fuzzy Computer)

퍼지 논리는 0과1 사이의 실수로 표현되기 때문에 기존의 디지털 컴퓨터로는 상당한 시간이 요구된다. 즉 수백개의 fuzzy implication들의 퍼지 추론이 실시간내에 real-time-mode에서 수행되기가 쉽지 않기 때문에, 이진 신호가 아닌 퍼지신호를 처리하는 하드웨어 시스템이 요구된다.

1.1 퍼지 하드웨어 시스템의 자료구조

(1) 소속함수(Membership function)

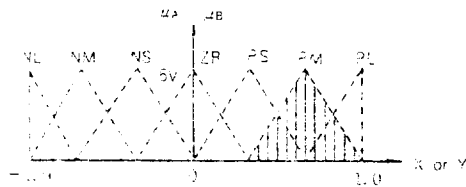


그림 1. 퍼지 하드웨어 시스템에서 사용되는 퍼지워드의 자료구조

소속함수는 본래 연속적인 함수이지만, 퍼지 하드웨어 시스템에서는 구분적 선형함수(piecewise linear function)로 표현하기로 한다. 즉, 퍼지 언어를 NL(Negative Large), NM(Negative Medium), NS(Negative Small), ZR(Approximately Zero), PS(Positive Small), PM(Positive Medium), PL(Positive Large)의 값으로 표현한다. (그림 1)

(2) 퍼지 언어(fuzzy Word)

퍼지 정보처리시스템에서 퍼지언어의 정보는 소속함수로 특징지어진다. 이 소속함수는 m개를 샘플링(Sampling)하여 얻은 m소자 벡터(vector)로, 다음과 같이 표현된다.

$$(0, 0, \dots, 0.1, 0.2, \dots, 0.9, 1.0, \dots, 0, 0, 0)$$

각 소자는 퍼지 하드웨어 시스템의 0V에서 5V의 전압을 나타낸다. 즉, 하드웨어 시스템의 샘플링할 수 있는 만큼의 신호로 나누어서 동작하게 된다. (그림 2. 참조)

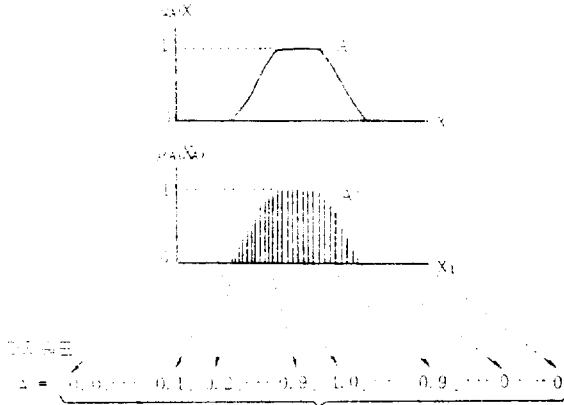


그림 2. 소속함수를 m소자 퍼지워드로 샘플링

1.2 Fuzzy Computer의 구성요소

(1) 퍼지 추론 기관

퍼지 정보처리 시스템은 다음의 Generalized Modus Ponens(GMP)나 Generalized Modus Tollens(GMT)을 사용하여 근사 추론(Approximate Reasoning 또는 퍼지추론)을 한다.

a. GMP(Forward Inference)

Implication : If x is A, then y is B

Premise : x is A'

Conclusion : y is B'

b. GMT(Backward Inference)

Implication : If x is A, then y is B

Premise : y is B'

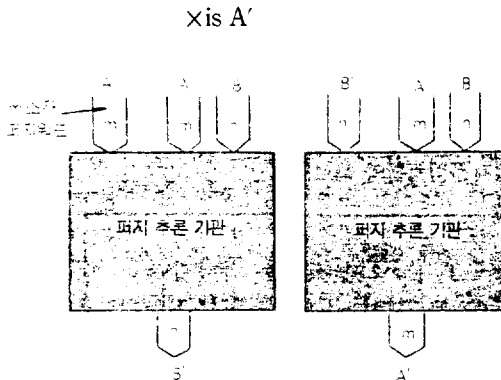


그림 3. 퍼지추론기관

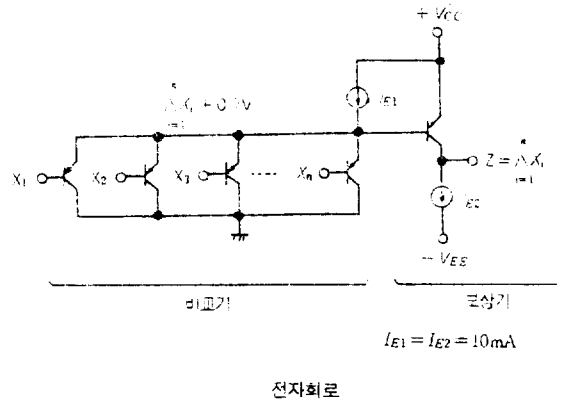


그림 4. 에미터 결합 퍼지 논리 게이트(MIN)

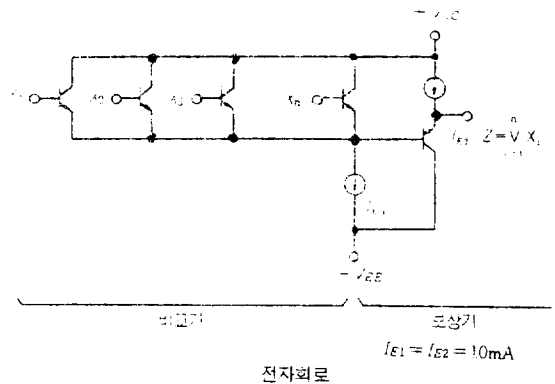


그림 5. 에미터 결합 퍼지 논리 게이트(MAX)

(A, A', B, B' : Fuzzy linguistic values)

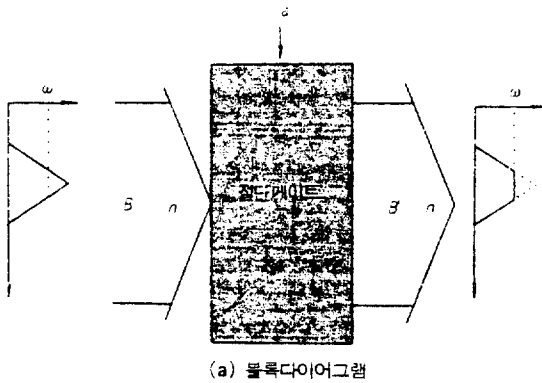
(그림 3)은 퍼지 추론 기관을 보여주고 있다.

퍼지 추론 기관의 기본 퍼지 논리 cell은 일반적으로 MIN, MAX회로로 (그림 4, 5, 6)과 같다.

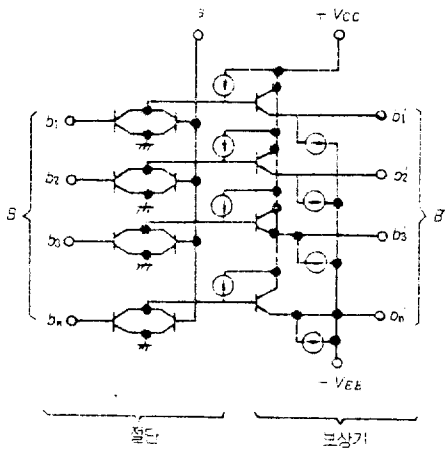
(그림 7)은 퍼지 추론 기관의 구조로, C-MIN은 두 입력 배열의 MIN회로에 해당되고, E-MAX는 MAX회로에 해당된다. 실제 prototype fuzzy computer YFC-1의 구조는 (그림 8)과 같으며, 응답시간이 100nsec보다 작아 10MFIPS(Mega Fuzzy Inferences Per Second)로 수행된다.

(2) 퍼지 메모리(fuzzy Memory)

퍼지 메모리는 fuzzy linguistic value의 소속함수를 표현하는 퍼지 워드들로 구성되며, $m=n=25$ 의 퍼지 워드를 구현하려는 경우에는 25개의 샘플링 신호로 7개의 소속함수를 기억하게 된다.



(a) 블록다이어그램



(b) 전자회로 수행

그림 6. 절단게이트

퍼지 추론 기관의 구조

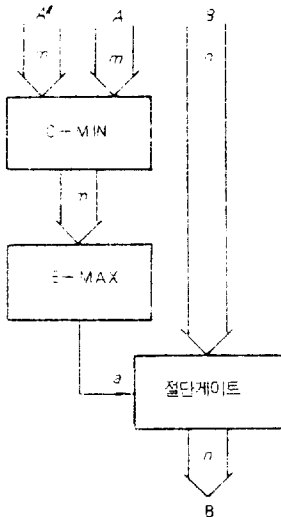


그림 7. 퍼지 추론 기관의 구조

퍼지컴퓨터 YFC-1의 구조

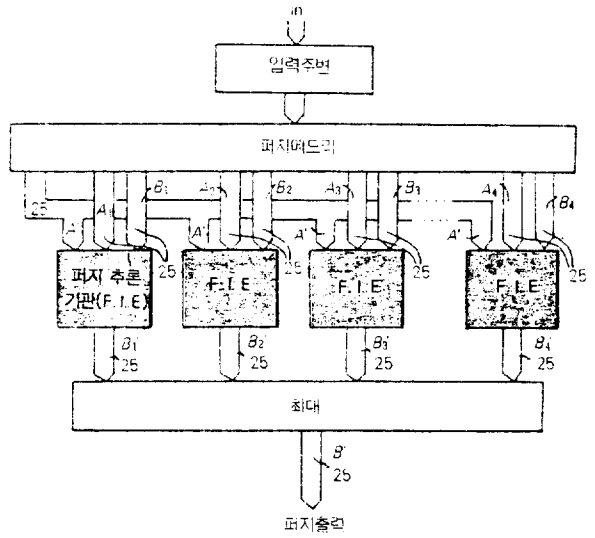


그림 8. 퍼지컴퓨터 YFC-1의 구조

(3) 퍼지 컴퓨터의 구조

퍼지 컴퓨터는 GMP나 GMT를 수행하는 퍼지추론기관들을 병렬로 수행하는 선형회로로, rule갯수만큼의 퍼지추론기관이 필요하게 된다. 그 구조는 (그림 8)로, r개의 퍼지 추론 기관을 병렬로 수행하는 퍼지 컴퓨터이다.

2. 퍼지 컨트롤러(Fuzzy Controller)

2.1 정의 및 구조

퍼지 컨트롤러는 퍼지 컴퓨터의 대표적인 예로, 입력과 출력이 fuzzy Word가 아닌 신호(signal)라는 점에서 앞에서 언급한 퍼지 컴퓨터와 다르다. 즉, 퍼지 컨트롤 시스템은 현재 상태 Y에서 외부의 입력 X를 받아들여 Z를 출력하게 된다. (그림 9)는 퍼지 컨트롤 시스템의 블록 다이어그램이다.

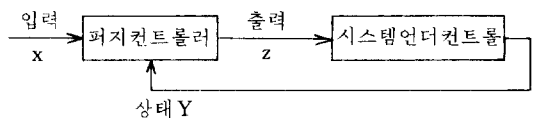


그림 9. 퍼지 컨트롤 시스템의 블록다이어그램

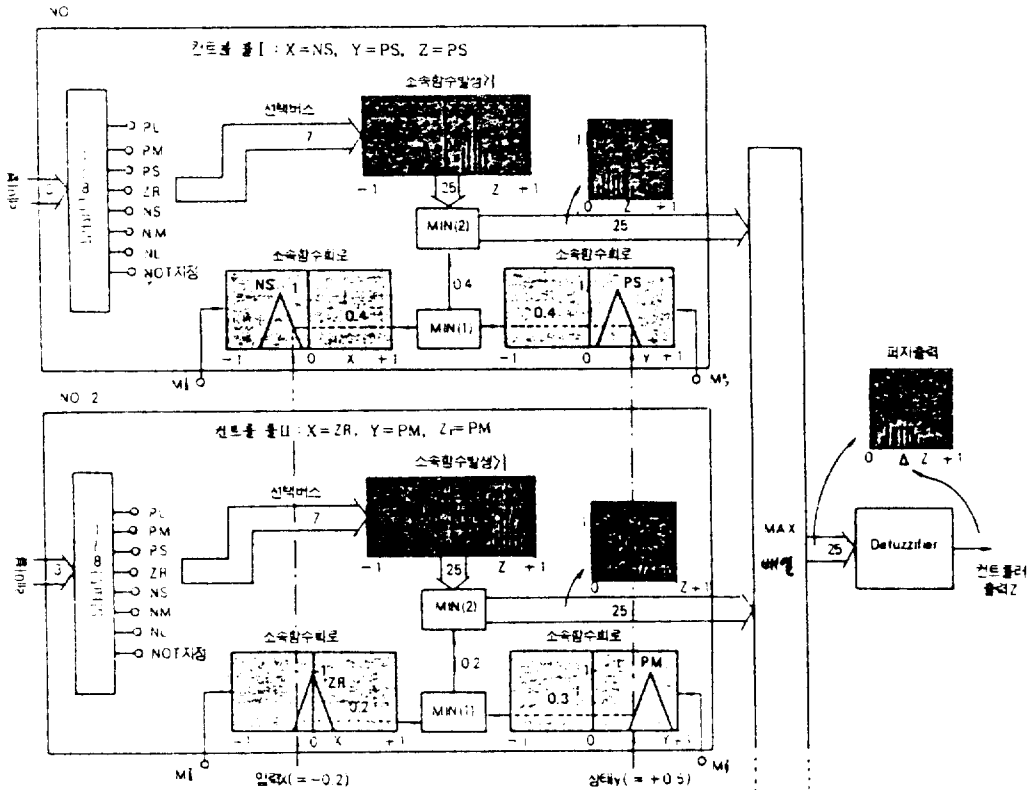


그림 10. 퍼지 컨트롤러 하드웨어 시스템의 구성

효과적인 제어(Control)을 하기 위해서, 전문가의 지식을 IF-AND-THEN의 rule들로 표현되는 Control rule들로 퍼지 컨트롤 시스템을 구성한다. Control rule은 다음과 같이 표현된다.

- Control Rule I = IF x=NS AND y=PS,
THEN z=PS ALSO
- Control Rule II = IF x = ZR AND y = PM,
THEN z=MP ALSO
- Control Rule III = IF... AND... THEN
ALSO
- ⋮

(그림 10)은 Control rule를 바탕으로 하드웨어 시스템을 구성한 것으로, 각 fuzzy Control rule을 Control rule board로 구현하고 두 개의 아날로그 입력 터미널(입력 x와 시스템의 상태 y)과 두 개의 소속 추론 터미널, 3비트 레이블 터미널, 25개의 신호 출력 터미널을 가진다. 여기서 각 Control rule board는 앞에서 언급한 Control rule들의 집합에 의

해 이미 프로그램되어있다.

2.2 구성요소

- (1) 소속 함수 발생기: 앞에서 언급한 7가지 소속 함수중의 하나를 25소자의 fuzzy word로 만들어내는 회로이다. 소속 함수 회로는 한개의 입력과 출력을 가지지만 소속 함수 발생은 25개의 출력 신호만 가질뿐 입력이 필요없다. (그림 11 참조)
- (2) 소속함수회로: 임의의 신호에 해당하는 소속 함수를 출력하는 회로이다. (그림 12 참조)
- (3) MIN회로: m개의 입력을 가지는 MIN회로를 (그림 13)과 같이 구성할 수 있다.
- (4) MAX회로: (그림 10)의 MAX배열은 25n-입력 MAX회로로 구성되어진다. (여기서 n은 시스템에서 사용하는 Control rule board의 수이다) 일반적으로, m-입력 MAX회로는 (그림 14)와 같이 구성된다.
- (5) Defuzzifier: Defuzzifier는 fuzzy word를 Cen-

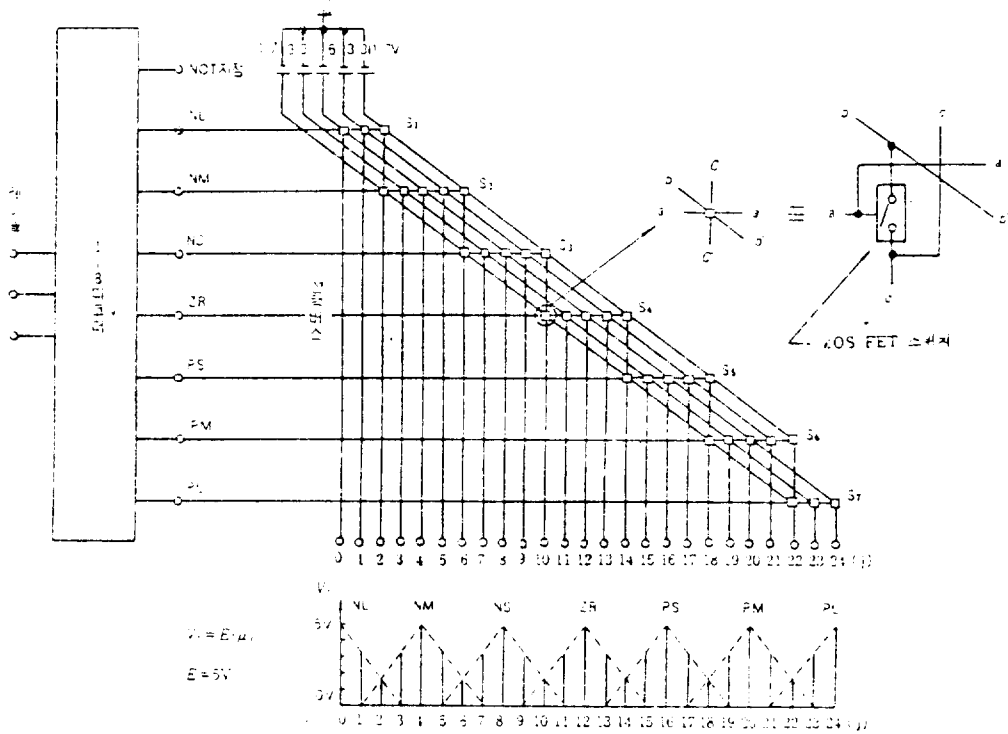


그림 11. 소속함수발생기와 1-8 디코더

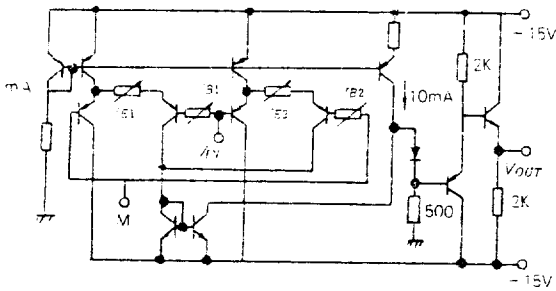


그림 12. 소속 함수 회로

ter of gravity 방법으로 분석하여 출력을 신호로 바꾸어주는 회로이다. 즉, 25개의 병렬 신호선을 하나의 신호로 바꾸어주고, 컨트롤러의 출력을 추출한다. 지금까지 여러 종류의 defuzzification algorithm이 제안되었지만 모두 고유의 Semantic error를 가진다. (그림 15)는 Center of Gravity 방법으로 defuzzifier를 구현한 것이다.

2.3 프로그래밍

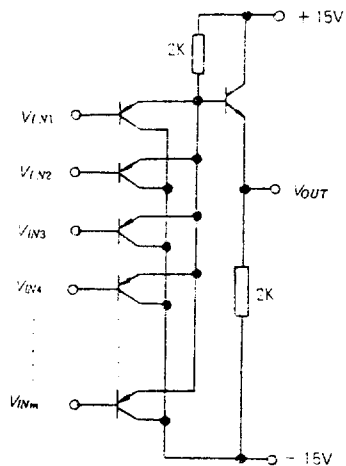


그림 13. MIN회로

퍼지 컨트롤러 하드웨어 시스템을 동작시키기 위해 전문가의 linguistic control rule로 구성된 control rule board의 터미널 전압을 지정해야 한다. 컨트롤러는 이 전압값에 따라 여러 용도로 사용될 수

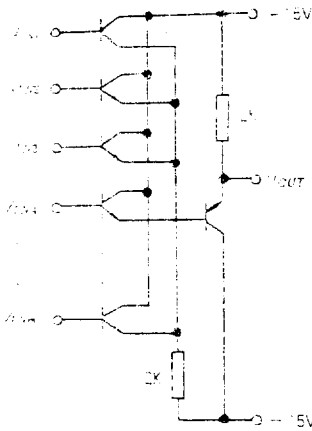


그림 14. MAX회로

표 1. 하드웨어 시스템의 구현

출력비트	M(V)	입력워드 레이블
PL	+5	111
PM	+3.3	110
PS	+1.7	101
ZR	0	100
NS	-1.7	011
NM	-3.3	010
NL	-5	

있기 때문에 이 과정을 프로그래밍(programming)이라 한다.

프로그래밍은 linguistic control rules에 일치하는 소속함수를 할당해주는 것으로, 소속함수의 레이블과 하드웨어 시스템에서의 구현은 <표 1>에 나타나 있다.

다음의 Control rule들을 사용하여 프로그래밍 과정을 살펴보면 다음과 같다.

Control rule I : $Mx^1 = -1.7V, My^1 = +1.7V,$
label=101

Control rule II : $Mx^2 = 0V, My^2 = +3.3V, \quad =110$

Control rule III : $Mx^3 = \text{don't care}, My^3 = \text{don't care}, \quad =000$

Control rule IV : $Mx^4 = \text{don't care}, My^4 = \text{don't care}, \quad =000$

Control rule I의 조건부분은 Control rule board상의 소속함수의 인수가 된다. Control rule I의 결론부분은 control rule board(label =101)상의 3bit word label을 주게된다. 주어진 Control rule수가 Control rule board보다 작다면, 남은 Control rule board를 label=000로 하여, 입력과 상태신호에 영향이 미치지 않게 해야한다.

2.4 구현결과

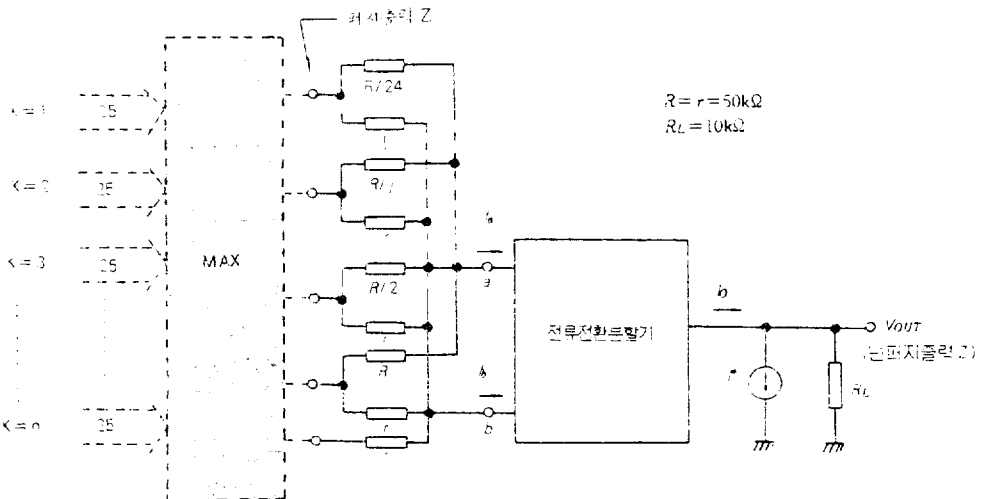


그림 15. Center of gravity 방법을 사용한 defuzzifier회로

여기서 구현한 컨트롤러의 응답시간은 Control rule 수에 관계없이 670nsec이고, 퍼지 추론이 초당 1,000,000배 빨리 수행됨을 알 수 있다. 이러한 MFIPS 기계를 여러 용도로 프로그래밍하여, 좀더 강력한 퍼지 컴퓨터를 구현할 수 있다.

3. 퍼지 플립플롭(Fuzzy flip-flop)

3.1 퍼지 플립플롭의 정의

퍼지 플립플롭은 이진 J-K플립플롭을 확장한 것이다. 단단계 퍼지추론을 하는 기존의 퍼지프로세서에서 다단계 퍼지추론을 수행하기 위해서는 퍼지 플립

플롭으로 구성된 퍼지 메모리 모듈이 필요하다.

이러한 퍼지 플립플롭은 fuzzy negation, t-norm, s-norm을 바탕으로 제안되어왔다. 일반적으로 tuzzy negation으로 Complementation을, t-norm으로 MIN을, s-norm으로 MAX를 사용한다.

기존의 이진 논리 J-K 플립플롭의 기본식은 아래와 같고,

$$\text{Minterm 표현식} = Q(t+1) = J\bar{Q} + \bar{K}Q \dots (1)$$

$$\text{Maxterm 표현식} = Q(t+1) = (J+Q)(\bar{K}+\bar{Q}) \dots (2)$$

이 식 (1)과 (2)를 퍼지화한 것이 각각 (3)과 (4)에 해당한다.

$$Q_R(t+1) = [J \wedge (1-Q)] \vee [(1-K) \wedge Q] \dots (3)$$

$$Q_S(t+1) = (J \vee Q) \wedge [(1-K) \vee (1-Q)] \dots (4)$$

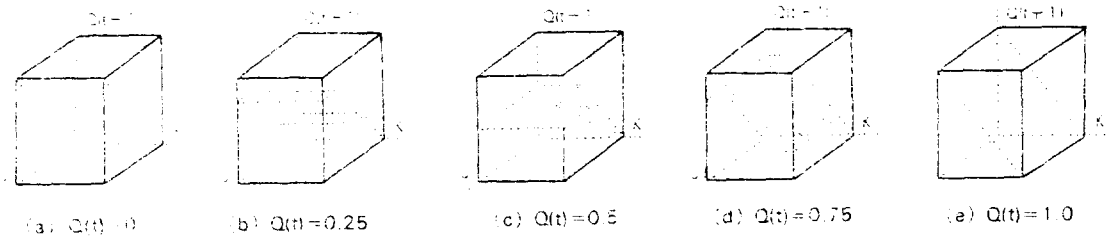


그림 16. MIN, MAX reset형 퍼지플립플롭의 특성

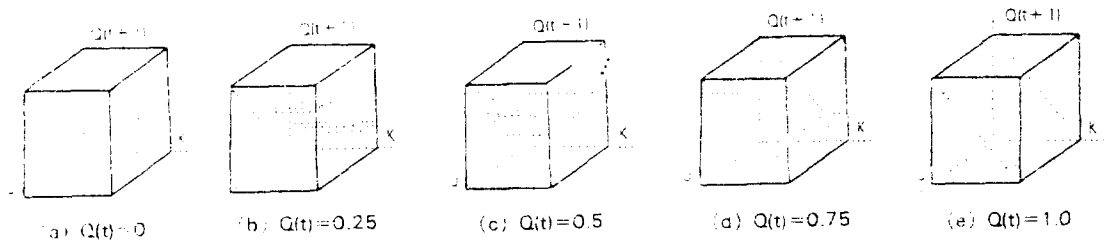


그림 17. MIN, MAX set형 퍼지플립플롭의 특성

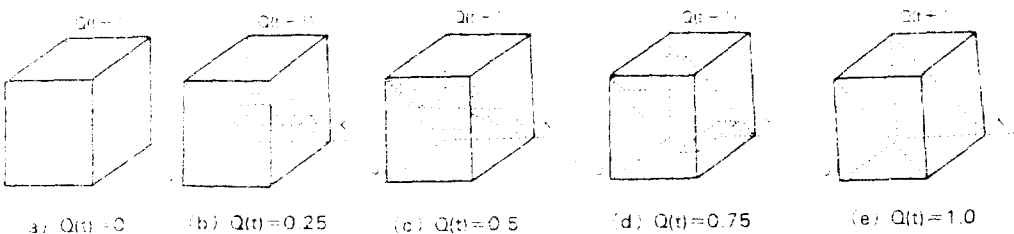


그림 18. MIN, MAX 퍼지플립플롭의 특성

여기서, 식(1)과 (2)는 항상 같지만, 퍼지논리의 식(3)과 (4)는 항상 같지가 않음을 주의해야 한다. 즉,

$$Q_R(t+1) \leq Q_S(t+1) \dots\dots\dots(5)$$

임을 쉽게 유도할 수 있다.

〈표 2〉는 J,K를 (0,1) 두 값에 한정시킨 식(3)과 (4)의 결과이다. J=0, K=1(reset 입력)이면, 식(3)의 다음상태는 0이 되나 식(4)는 “ $Q \wedge (1-Q)$ ” (≥ 0)이 된다. 또한, J=1, K=0(set 입력)이면, 식(4)는 1이 되나 식(3)은 “ $(1-Q) \vee Q$ ” (≤ 1)이 된다. 따라서, 식(3)을 MIN, MAX reset형 퍼지 플립플롭, 식(4)을 MIN, MAX set형 퍼지 플립플롭이라고 부른다. ($Q \wedge (1-Q)$ 와 $(1-Q) \vee Q$ 는 각각 항상 0과 1이 아님을 주의해야 한다).

$Q(t)=0, 0.25, 0.5, 0.75, 1.0$ 인 경우, MIN, MAX reset형 퍼지 플립플롭의 $Q_R(t+1)$ 는 각각 〈그림 16〉에 나타나 있다. 마찬가지로 〈그림 17〉는 MIN, MAX set형 퍼지 플립플롭의 $Q_S(t+1)$ 는 〈그림 17〉에 있다.

두가지 형의 $Q(t+1)$ 의 값은 $J=K$ 라인 세그먼트 상에 연속적으로 연결됨을 $J=K < Q$ 인 경우 $J=K=Q$ 인 경우, $J=K > Q$ 인 경우에 나누어 증명하면, 식(3)은 $J \wedge (1-Q)$ 이 되고 식(4)는 $J \wedge (1-Q)$ 가 된다. 따라서, 이진 J-K플립플롭을 퍼지 플립플롭으로 확장시키기 위해서 다음의 함수를 정의해야 한다.

$$Q(t+1) = \begin{cases} (J \vee Q) \wedge \{(1-K) \vee (1-Q)\} & (J \geq K) \\ (J \wedge (1-Q)) \vee \{(1-K) \wedge Q\} & (J \leq K) \end{cases} \dots\dots\dots(6)$$

위의 식(6)를 MIN, MAX형 퍼지 플립플롭의 기본식이라 하고, 〈그림 18〉은 $Q(t)=0, 0.25, 0.5, 0.75, 1.0$ 인 경우의 식(5)의 값을 나타낸다.

표 2. 식(3)과 식(4)의 값. $\{J, K \in (0,1)\}$

J	K	$Q_R(t+1)$	$Q_S(t+1)$	F
0	0	Q	Q	Q
0	1	0	$Q \cap (1-Q)$	0
1	0	$(1-Q) \cup Q$	1	1
1	1	$1-Q$	$1-Q$	\bar{Q}

3.2 퍼지 플립플롭의 구현

퍼지 플립플롭회로는 식(3)과 (4)에 의해 〈그림 19〉와 같다. Complementation, MIN, MAX 회로의 실제적인 구현은 〈그림 20〉과 같이 구성하여, 퍼지값 [0,1]을 전압 레벨 [0V, 5V]로 나타내고 있다.

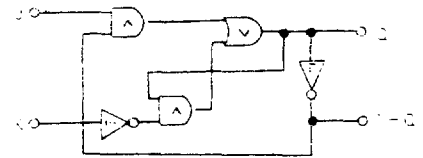
식(6)을 실제로 구현하려면 추가적으로 비교기와 계전기회로가 필요하게 되므로, 다음의 식을 고려해야 한다.

$$Q(t+1) = \{J \vee (1-K)\} \wedge \{J \vee Q\} \wedge \{(1-K) \vee (1-Q)\} \dots\dots\dots(7)$$

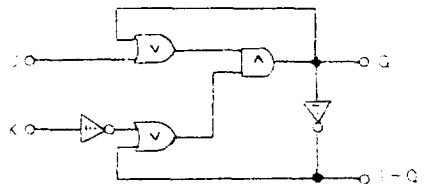
식(7)은 reset형과 set형의 특성함수가 된다. 이를 바탕으로 Complementation, MIN, MAX 게이트회로를 사용한 퍼지플립플롭 회로는 〈그림 21〉에 제시되어 있다.

현재 입력 J(t)와 K(t)는 두개의 MIN회로에 있는 동기화된 클럭에 의해 구동된다. 출력 Q(t)는 두개의 표본에 의해 기억되고, 다음 상태에 정보를 사용하기 위해서 회로를 유효하게 한다. 이회로의 타이밍도는 〈그림 21〉의 (b)와 같다. 〈그림 22〉는 일반적인 회로도이다.

MIN, MAX형 퍼지플립플롭으로 퍼지메모리 모듈을 실제 VLSI로 구현하기 위해서는 액세스 타임, 온도등의 특성들이 고려되어야만 한다. 이러한 퍼지



(a) 리세트형



(b) 세트형

그림 19. Complementation, MIN, MAX 게이트를 사용한 리세트형과 세트형 퍼지플립플롭

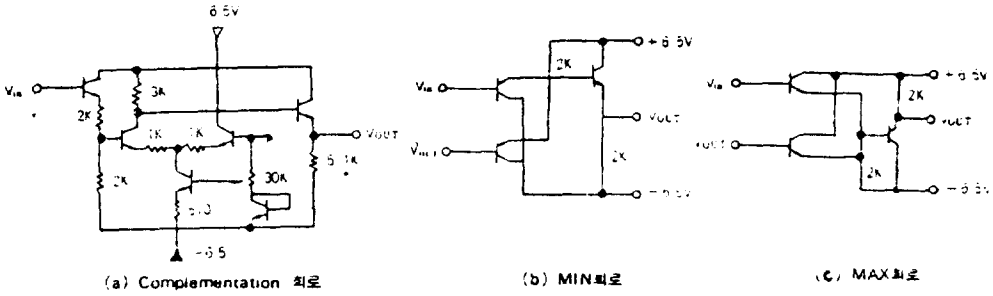


그림 20. 전압레벨에서의 Complementation, MIN, MAX회로의 구현

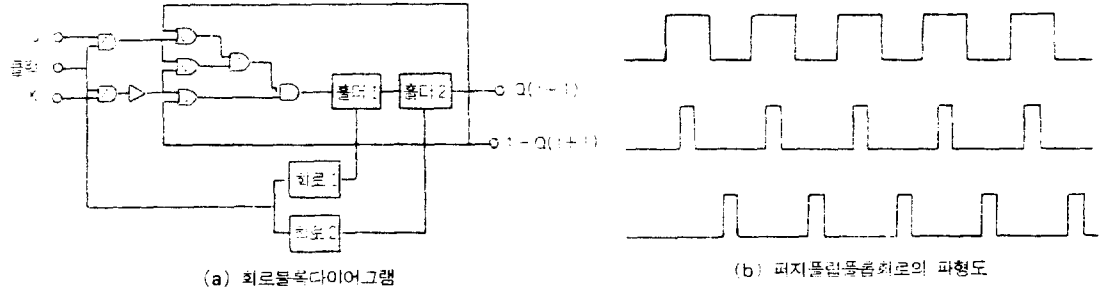


그림 21. Complementation, MIN, MAX게이트를 사용한 퍼지플립플롭회로

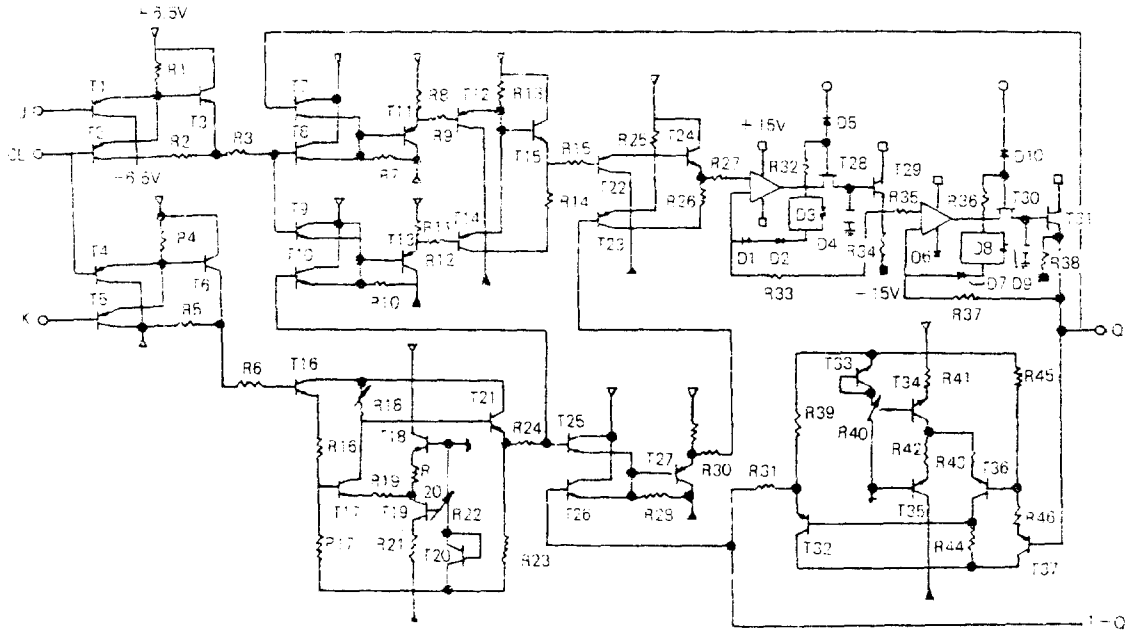


그림 22. MIN, MAX형 퍼지플립플롭의 회로도

플립플롭 회로를 사용하면, 1digit 퍼지 정보를 저장할 수 있다. 여러개의 퍼지 플립플롭을 퍼지추론

chip과 같이 사용하면 다단계 퍼지 추론 기계의 구성이 가능해진다.