

148  
2ej.



UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO

FACULTAD DE INGENIERIA

CONTROL DIFUSO PARA NIVEL DE LIQUIDOS  
IMPLEMENTADO CON MICROCONTROLADORES

**T E S I S**

QUE PARA OBTENER EL TITULO DE  
INGENIERO MECANICO ELECTRICISTA  
P R E S E N T A  
FLORICEL RAMOS VARGAS

DIRECTOR: ING. JESUS ALVAREZ CASTILLO



MEXICO, D.F.

1997

TESIS CON  
FALLA DE ORIGEN



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*A mis padres ANASTACIO Y TERESA porque siempre me apoyaron e impulsaron a concluir mi formación profesional, mi agradecimiento y todo mi cariño.*

*Con todo mi amor a JAIME, CLAUDIA Y MARIANA, por su comprensión y entusiasmo.*

*A mi familia con cariño.*

*A compañeros y amigos.*

*Con agradecimiento especial al Ing. JESUS ALVAREZ CASTILLO.*

---

---

# INDICE

CAP.		PAG.
1	INTRODUCCIÓN.	
2	MICROPROCESADORES Y MICROCONTROLADORES.	
	2.1 Introducción	4
	2.2 Microprocesadores	5
	2.2.1 Diseño C.I.S.C.	8
	2.2.2 Diseño R.I.S.C	11
	2.3 Microcontroladores	12
	2.3.1 Microcontroladores seleccionados	16
3	MULTIPROCESAMIENTO	
	3.1 Introducción	18
	3.2 Arquitectura de Multiprocesamiento	19
	3.2.1 Arquitectura acoplado flexible	19
	3.2.2 Arquitectura acoplado rigida	21
	3.3 Arquitectura de subsistemas caché.	24
4	FUNDAMENTOS DE LÓGICA DIFUSA	
	4.1 Introducción	28
	4.2 Características principales de la lógica difusa.	29
	4.3 Fundamentos matemáticos de la lógica difusa	31
	4.3.1 Conjuntos difusos	38

---

---

4.3.2	Elementos de una función miembro	45
4.3.3	Conjuntos difusos vistos de su perspectiva geométrica.	46
4.3.4	Variables lingüísticas y razonamiento aproximado	49
<b>5</b>	<b>TRANSDUCTORES Y ACTUADORES</b>	
5.1	Introducción	53
5.2	Clasificación de los transductores	53
5.3	Transductores de presión y temperatura	59
5.4	Actuadores	60
<b>6</b>	<b>CONSIDERACIONES BÁSICAS EN EL DISEÑO DEL CONTROLADOR DIFUSO</b>	
6.1	Introducción	61
6.2	Características de los algoritmos de control.	62
6.3	Estructura de un controlador difuso	63
6.4	Descripción general del proceso	65
6.5	Algoritmo difuso	66
6.5.1	Fusificación	66
6.5.2	Generación de las reglas de inferencia	69
6.5.3	Defusificación	73
6.6	Estabilidad del controlador difuso	74
<b>7</b>	<b>IMPLEMENTACIÓN DEL SISTEMA PROPUESTO</b>	
7.1	Introducción	76
7.2	Descripción general de la arquitectura del controlador.	76
7.3	Diagrama de flujo	77
7.4	Manejo del proceso desde el monitor central	82

7.5	Diagrama de bloques	85
<b>8</b>	<b>PRUEBAS, RESULTADOS Y CONCLUSIONES</b>	
8.1	Introducción.	88
8.2	Pruebas	88
8.3	Resultados	91
8.4	Conclusiones	93
<b>9</b>	<b>BIBLIOGRAFÍA</b>	<b>95</b>
	<b>ANEXO A</b>	<b>99</b>
	<b>ANEXO B</b>	<b>101</b>

## 1.- INTRODUCCIÓN

Con el desarrollo de los microprocesadores, los controladores basados en estos dispositivos son cada vez más populares y por su gran versatilidad han superado a los controladores analógicos, ya que pueden "recordar" experiencias pasadas y "predecir" las futuras como secuencias causadas por entradas presentes; además el microprocesador puede implementarse con algoritmos de control inteligentes para tratar con variaciones de parámetros físicos, esto como resultado de disturbios, procesos no lineales y cambios de parámetros, dichos algoritmos están basados en la intuición y la experiencia; además pueden implementarse fácilmente en una computadora. Ahora un controlador donde se utiliza la lógica difusa tiene la ventaja de proporcionar relativamente más rapidez y precisión en su respuesta. Otra ventaja importante del uso de la lógica difusa es su facilidad de implementación y análisis, comparada por ejemplo con las técnicas no lineales que generalmente son complejas y requieren de gran tiempo de cómputo limitando sus aplicaciones. El problema principal en este caso es formular la relación entrada-salida por medios matemáticos, que generalmente son complejos.

Como antecedente en el desarrollo de la lógica difusa, podemos mencionar los trabajos realizados por Madami y Assilian quienes presentaron un informe sobre la aplicación de la lógica difusa para controlar una pequeña máquina de vapor de laboratorio [1],[2]. El propósito fue regular la velocidad de una máquina y la presión de

vapor en la caldera. Al mismo tiempo, Van Nauta Lemke y Kickert [3],[4] examinaron la función de un controlador difuso sobre una planta de agua caliente; el éxito de éstos estudios, permitieron a King y Mamdani [5],[6] probar un control de temperatura de un reactor químico, usando algoritmos difusos. Así también Rutherford [7] y Ostergaard presentaron los resultados de experimentos con controladores difusos sobre una tolva y un intercambiador de calor respectivamente.

Los resultados de todos estos trabajos demostraron que los controladores difusos funcionan mejor, ó al menos igual que un controlador PID (Control Proporcional-Integral- Derivativo).

El objetivo de este trabajo es aplicar un algoritmo difuso en el diseño e implementación de un controlador de nivel utilizando microcontroladores.

*En el capítulo dos,* hablamos de la organización de los microprocesadores, y microcontroladores, de las características de los diseños C.I.S.C. y R.I.S.C. así como los microcontroladores utilizados en este trabajo.

*En el capítulo tres,* mencionamos una clasificación general de multiprocesamiento.

*En el capítulo cuatro,* exponemos las características de la lógica difusa, así como todo lo relacionado con los principios matemáticos de ésta.

*En el capítulo cinco,* presentamos características de los transductores y su clasificación.



*En el capítulo seis,* describimos el diseño del controlador difuso implementado.

*En el capítulo siete,* mostramos y describimos la arquitectura del controlador.

*En el capítulo ocho,* desplegamos los resultados obtenidos en las pruebas hechas a nuestro controlador.

## **2.- MICROPROCESADORES Y MICROCONTROLADORES**

### **2.1.-INTRODUCCIÓN**

A través del tiempo el ser humano ha desarrollado herramientas que le permitieron conocer su medio. En el Oriente, una herramienta fue el ábaco para sumar, que fue construido con cuentas sobre barras. En 1617 Jhon Napier inventó un sistema para multiplicar y dividir basado en algoritmos. Así también en 1964 Gottfred Wilhelm Leibnitz terminó una máquina mecánica que podía sumar, restar, multiplicar, dividir y obtener raíz cuadrada. Así fueron surgiendo nuevas tecnologías, como el microprocesador, el cual transformó el desarrollo de las computadoras.

En 1971 Intel introdujo el primer microprocesador, el 8080. Poco después Motorola introdujo el 6800, Signetics 6502, Zilog y Z80, por su parte Texas Instruments introdujo el 9900 (16 bits). Para 1978 el microprocesador más común fue el de 16 bits, entre estos se encontraron el 8086 de Intel, el 68000 de Motorola y Z8000 de Zilog [10], [11]. A partir de los microprocesadores se desarrollaron los microcontroladores y en 1980 Intel da a conocer el microcontrolador 8051, con 4 Kbytes de ROM, 128 bytes de RAM, 32 líneas I/O, un puerto serial y los timers de 16 bits. Alrededor de 1985 Motorola desarrolla el microcontrolador 68HC11 que es compatible con el microprocesador 6800. Este microcontrolador es de alta densidad y

usa tecnología CMOS, haciendo éste dispositivo más pequeño, rápido y con una alta tolerancia a las señales de ruido [10],[11]. Se han desarrollado varias versiones del 68HCII, como ejemplo el MC68HCIIA8, MC68CHIIIE9 y MC68HCIIIF1. Recientemente Microchip, ha desarrollado tres líneas de microcontroladores que son el PIC16C5XX, PIC16CXX y PIC17CXX.

En este capítulo hablaremos del aspecto general y técnico de los microcontroladores y microprocesadores, haciendo una diferencia entre éstos, así como sus aplicaciones.

## **2.2.- MICROPROCESADORES**

Una computadora típica tiene en su CPU un microprocesador, este dispositivo ejecuta las funciones básicas para realizar instrucciones y procesar datos de acuerdo a un programa específico. Para realizar la operación adecuada requiere de señales de control y de tiempo, así como de los buses de dirección y de datos, y las líneas de control de lectura y escritura para comunicarse con una unidad de memoria. Una línea de control es la requisición de interrupción, en donde el microprocesador suspende la ejecución del programa para atender una rutina de servicio, al terminar regresa al programa previo [25]. Otra línea de control es la terminal reset que reposiciona ó inicia el proceso desde el principio. La requisición del bus (bus request) permite transferir información externa a la memoria sin la intervención del procesador.

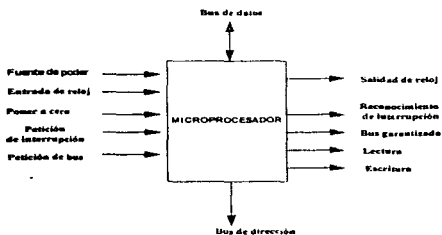


Figura. 2.1 SEÑALES DE CONTROL DE UN MICROPROCESADOR [25]

Las señales de control citadas anteriormente constituyen un conjunto mínimo de funciones de control para un microprocesador. La mayoría de los microprocesadores tienen características de control adicionales para funciones especiales [25].

**Organización Interna.-** Internamente el microprocesador tiene seis registros procesadores, un registro acumulador, un registro de tiempo y un ALU (Unidad Lógica Aritmética). Ver figura 2.2.

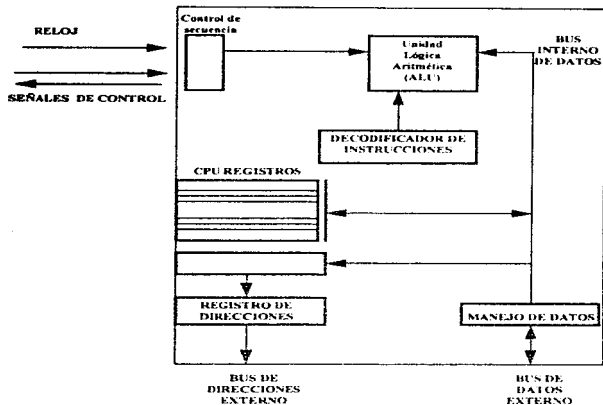


Figura 2.2 DIAGRAMA DE BLOQUES DE UN MICROPROCESADOR [10]

El ALU opera con los datos almacenados en el acumulador y el resultado se transfiere al acumulador ó a cualquiera de los registros procesadores. Durante el ciclo de búsqueda, el control lee un código de operación de la memoria y lo deposita en un registro de instrucción. La instrucción se decodifica y se traslada a actividades de procesamiento específico, las instrucciones son de nibbles, bytes, ó palabras siendo los modos de direccionamiento eficientes para operar grandes volúmenes de datos. Los separadores de direcciones reciben

la información de tres fuentes: el contador del programa (pc), el indicador de pila ó stack pointer (sp) y el registro de direcciones (ar). En cuanto al contador del programa mantiene la dirección de memoria de la instrucción corriente del programa y se incrementa después de cada búsqueda de instrucción.

### **2.2.1.- DISEÑO C.I.S.C.**

El Complex Instruction Set Computer (C.I.S.C.), es un diseño que se aplica a chips para hacer eficiente el uso de memoria. Cada instrucción en un diseño C.I.S.C. puede realizar series de operaciones en el procesador reduciendo el número de instrucciones requerido para implementar un programa dado y permite al programador aprender un pequeño y flexible grupo de instrucciones [8].

#### **- Factores que determinaron el desarrollo C.I.S.C.**

1.- **Uso de un microcódigo.**- En un sistema microprogramado, el procesador tiene memoria ROM, la cuál contiene un grupo de microcódigos de instrucciones, que corresponden con el lenguaje de máquina. Cuando una instrucción en lenguaje de máquina llega al procesador, este ejecuta las correspondientes series de microcódigos de instrucción.

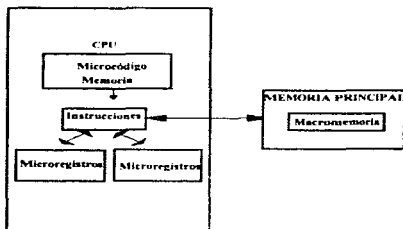


Figura 2.3 DIAGRAMA DE FLUJOS DE UN MICROPROGRAMA

**2.- Instrucciones.-** Una de las consecuencias del uso de microprograma es la funcionalidad en cada instrucción.

**3.- Lenguaje alto-nivel.-** La implementación de un lenguaje de alto nivel en las instrucciones, no sólo simplifica el trabajo de escritura del compilador si no que el compilador emite pocas instrucciones por línea de la fuente código.

**Características de un diseño C.I.S.C.**

Las instrucciones en un diseño C.I.S.C. tiene las siguientes características:

- Un formato de dos operandos, es decir, una fuente y un destino.
- Comandos de registro a registro, registro a memoria y memoria a registro.
- Múltiples modos de direccionamiento.
- Instrucciones de longitud variable dependiendo del modo de direccionamiento.
- La ejecución de instrucciones requiere múltiples ciclos de reloj.

El C.I.S.C. reduce el número de instrucciones por programa y la ecuación usual para determinar la ejecución es la suma del producto del número de ciclos por instrucción por el tiempo de ciclo de instrucción, o sea:

$$\sum_i^N \left( \begin{array}{c} \text{número de ciclos} \\ \text{por instrucción [i]} \end{array} \right) * \left( \begin{array}{c} \text{tiempo de ciclo} \end{array} \right) \quad \text{EC.2.1}$$

Los procesadores de la familia 68000 de Motorola forman la base del inicial sistema Macintosh y es representativo de un diseño C.I.S.C. Estos procesadores tienen registros de propósito general, un registro de status de propósito especial, que contiene la condición de código y un registro como stack pointer.



### **2.2.2.- DISEÑO R.I.S.C.**

Una de las mayores realizaciones de un diseño R.I.S.C. es que una sucesión de simples instrucciones produce los mismos resultados de una sucesión de instrucciones complejas. De ahí el nombre Reduced Instruction Set Computers (R.I.S.C.) [8].

#### **Características de un diseño R.I.S.C.**

- Opera con instrucciones simples.
- Misma longitud de instrucciones.
- Más instrucciones en un ciclo de máquina, esto es, permite manejar varias instrucciones al mismo tiempo. A este sistema se le conoce como Pipeline que es una técnica usada en diseños R.I.S.C.

#### **Sistema Pipeline**

La técnica Pipeline es un diseño donde el hardware procesa más de una instrucción en un tiempo y no espera una instrucción para llevarla a cabo antes de comenzar la siguiente [8].

En los sistemas C.I.S.C. existen 4 etapas (búsqueda, decodificación, ejecución y escritura) y estas mismas etapas existen en el sistema R.I.S.C., pero son ejecutadas en paralelo. Es decir, tan pronto como

una etapa finaliza, el resultado pasa a la siguiente etapa y comienza a trabajar sobre otra instrucción.

La implementación de un procesador con un diseño de instrucción simplificada tiene varias ventajas sobre la implementación de un diseño C.I.S.C., como ejemplo, mencionamos:

- **La Velocidad.**- Al utilizar instrucciones simplificadas permite implementar un sistema Pipeline.
- **Un Hardware simple.**- Como las instrucciones de un procesador R.I.S.C. son tan simples se utiliza menos espacio de memoria.

### **2.3.- MICROCONTROLADORES**

Como anteriormente mencionamos, un microprocesador es un C.P.U., en tanto un microcontrolador está formado por C.P.U., memoria y registros conectados por medio de un bus. Además externamente, tiene terminales para polarización, terminales entrada/salida (I/O) que están agrupadas dentro de unidades llamadas puertos. El C.P.U. controla la operación del microcontrolador y en la memoria se almacena datos y un código de programa (algunos microcontroladores usan memoria EPROM en lugar de ROM) [10]. Los registros son usados para almacenar información especial, y existen tres tipos básicos de registros entrada/salida (I/O): datos, control y status. Algunas veces, un microcontrolador requiere más puertos entrada/salida ó memoria que pueden ser obtenidos dentro del mismo

chip. Esto es, puede organizarse para trabajar de cualquier modo. Esta organización es llamada "modos de operación del microcontrolador".

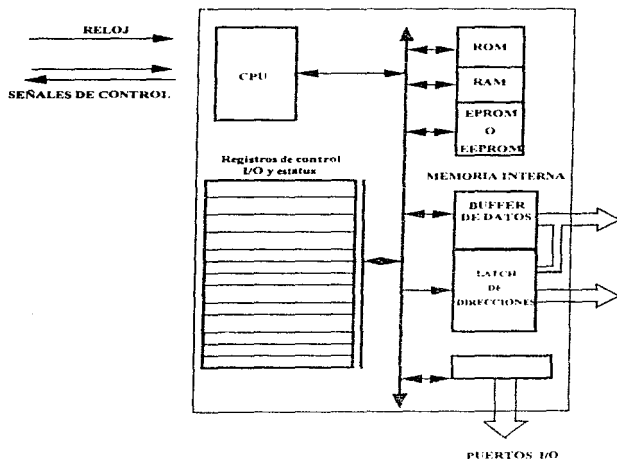


Figura 2.4 DIAGRAMA DE BLOQUES DE UN MICROCONTROLADOR [11],[12]

Actualmente existen varios fabricantes que han diseñado diferentes microprocesadores y microcontroladores con características especiales. Entre los más comunes son los siguientes [10],[11],[22]

**MICROPROCESADORES Y MICROCONTROLADORES**

Reloj	INTEL		MOTOROLA		MICROCHIP T.1		OTROS
	8088 15 MHZ	8031 12 MHZ	68000 8 MHZ	68HC11 8 MHZ	PICXX 20 MHZ	DSP56K	
Memoria Interna	-	128 Bytes de RAM	-	256 Bytes de RAM	192 Bytes de RAM 4K	256 Bytes de RAMX 256 de RAMY	1024 Bytes de RAM
Memoria Externa	1Mega	64 K	16 Megas	64k	8K	65 K/X 65K/Y	1 Mega para programar 1 Mega para datos
Puertos	PPI y USART	4 paralelos 1 serie	-	2 series 5 paralelos	33 pines	2 series 4 paralelos	2 series 6 paralelos
Tamaño de Palabra	8Bits	8Bits	16 Bits	8 Bits	8 Bits	24 Bits	16 Bits
Interrupciones	256 2 son por Hardware	5 con dos niveles de prioridad	7 niveles de prioridad	17 con prioridades fijas	12 niveles	20 con 4 niveles de prioridad	-
Temporizadores	-	2 de 16 Bits	-	1 de 16 Bits	3	-	4 de 16 Bits
Convertidor A/D	-	-	-	4 de canales de 8 Bits	8 canales de 8 Bits	-	8 canales de 16 Bits
Acumuladores	4	2	8 de datos 8 de direcciones	7	2	2 de 56 Bits	-

**Tabla 2.1 CARACTERISTICAS DE ALGUNOS MICROPROCESADORES Y MICROCONTROLADORES.**

**En donde:**

**Reloj.-** Es la frecuencia a la que opera el dispositivo, (es suministrada por medio de un dispositivo externo). Un periodo de reloj es conocido como un ciclo de máquina.

**Memoria.-** Con respecto a su implementación, existen dos tipos de memoria: externa e interna y es manejada mediante el bus de direcciones [11]. En el microcontrolador la memoria interna es donde se encuentran todos sus registros, además cuenta con regiones RAM, ROM y en algunos EEPROM, para utilizarlas con programas pequeños. Su memoria externa es de mayor capacidad y puede ser de lectura, escrituras ó ambas y se usa para aplicaciones que requieran de un espacio mayor en memoria.

**Puertos.-** Son los elementos que sirven de contacto entre el microprocesador y el mundo exterior. De la forma en que transmiten ó reciben datos pueden ser Paralelo y Serie.

**Tamaño de Palabra.-** Se denomina así al número de bits que conforman un dato que pueda ser manejado por el micro. Además es el espacio (a lo ancho) que tiene la memoria.

**Velocidad de Procesamiento.-** Es la rapidez para ejecutar una instrucción y estar listo para atender la siguiente [12],[10].

**Conjunto de Instrucciones.-** Es el grupo de instrucciones que es capaz de realizar un micro.

**Capacidad de Memoria.-** Es el máximo espacio disponible para almacenar información (programas y/o datos).

### 2.3.1.- MICROCONTROLADORES SELECCIONADOS

De lo anterior y por la versatilidad de sus subsistemas entrada/salida (I/O) nos permitimos utilizar los microcontroladores 68HC11A8 (miembro de la familia 68HC11) de Motorola y el PIC16C74 de Microchip para la implementación de nuestro algoritmo difuso. Las características principales de estos microcontroladores son las siguientes[22].[11]:

68HC11A8	PIC16C74
- Diseño C.I.S.C.	- Diseño R.I.S.C.
- 8K ROM	- 4K EPROM
- 256 bytes, RAM	- 192 bytes, Memoria Datos
- 512 bytes, EEPROM	- Puerto Serial
- 64K Direccionamiento	- Timer
- SPI, Interface Periférica Serial	- SPI
- SCI, Interface Comunicación Serial	- SCI
- Timer de 16 bits	- Puerto Paralelo

**68HC11A8**

- Ocho canales A/D
- Interrupción en Tiempo Real
- Watchdog
- 38 Terminales I/O

**PIC16C74**

- Ocho Canales
- Doce Fuentes de Interrupción
- 33 terminales I/O

En el *PIC16C74* además utiliza arquitectura Harvard, en el cuál el programa y datos son accedidos a memorias independientes usando bases independientes. Esto mejora el ancho de banda sobre la tradicional arquitectura Von Neumann, donde el programa y datos son llevados a la misma memoria usando el mismo bus [22]. También el direccionamiento a los registros o a la memoria de datos puede ser directo ó indirecto.

En cuánto al 68HC11A8 una de las desventajas es que no dispone de suficiente memoria para diseñar nuestro algoritmo por lo que es necesario utilizarse en modo expandido. Por esta razón fue necesario dividir nuestro algoritmo en dos partes fundamentales: Fusificación y Defusificación, que corresponden respectivamente al PIC16C74 y 68HC11A8 [11]. Una de las ventajas de el 68HC11A8 es que utiliza el proceso semiconductor metal-óxido complementario alta densidad (HCMOS), que lo hace más pequeño y rápido. Además consume menos potencia y tiene alta tolerancia con señales de ruido.

### **3.- MULTIPROCESAMIENTO**

#### **3.1.- INTRODUCCIÓN**

En los últimos 12 años, la tecnología se ha dirigido al desarrollo de redes locales (LAN), aplicaciones de servidores-clientes y distribución de base de datos como herramientas para mejorar el acceso y manipulación de envío de datos. Este giro en tecnología, junto con el incremento de manipulación de datos, ha dado lugar a la alta demanda, sobre la función de los servidores utilizados en la corporación de redes. Un método para incrementar el desempeño a los servidores es el Multiprocesamiento.

Compaq introdujo a la industria el desarrollo e implementación del avanzado sistema de multiprocesamiento. En 1994, Compaq logró un 29% en la distribución de mercado de servidores PC en el mundo, 50% más grande que IBM, el siguiente competidor. Compaq también logró un 71% en la distribución de mercado de superservidores más de seis veces que cualquiera de sus competidores [27]. Compaq trabaja fuertemente con sistemas de operación estratégicas y base de datos para dar todo el soporte a la arquitectura de multiprocesamiento de servidores Compaq.



### **3.2.-ARQUITECTURA DE MULTIPROCESAMIENTO**

El multiprocesamiento requiere avanzado hardware (tales como alta velocidad caché, controladores inteligentes I/O y avanzada arquitectura de bus) para la gran cantidad de datos que transfiere entre procesadores, memoria principal y subsistemas I/O. También el multiprocesamiento requiere sistemas de operación que puedan distribuir la carga de trabajo a través de un procesador y aplicaciones que puedan dividir su tarea individual hacia una línea de códigos que puedan ser previstos para la ejecución por el sistema de operación. Hay dos importantes métodos utilizados para implementar múltiples procesadores en un servidor [9]:

- Arquitectura de acoplado flexible.
- Arquitectura de acoplado rígido.

#### **3.2.1.- ARQUITECTURA ACOPLADO FLEXIBLE**

En este diseño, algunas veces referido como un grupo, cada procesador tiene su propia memoria local, sistema de bus, y componentes periféricos. Este grupo ha sido popular en las minicomputadoras y en el computador central por algún tiempo, típicamente incluye la conexión de varias computadoras flexiblemente acopladas (ó nodos de procesadores) con algún tipo de interconexión. Esta interconexión puede ser un cable conectado entre los nodos del procesador ó puede ser un bus interno de una computadora, y es usada para transferir mensajes entre los procesadores. Ya que no hay

memoria compartida, se requiere el software adicional para mantener los datos coherentes entre los nodos. El sistema de operación es generalmente construido para formar un nodo individual presentando un sistema monolítico.

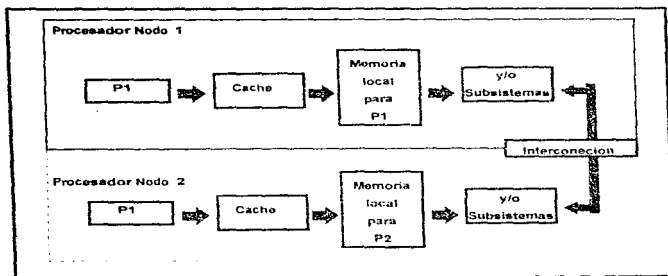


Figura 3.1 MULTIPROCESAMIENTO ACOPLADO FLEXIBLE [27]

### 3.2.2.- ARQUITECTURA ACOPLADA RÍGIDA

Este diseño, es encontrado en la mayoría de los multiprocesamientos comerciales, en donde los procesadores comparten la memoria principal, sistema bus y subsistemas entrada/salida (I/O) [9]. La arquitectura acoplada rígida puede ser arquitectura de multiprocesamiento simétrico ó asimétrico.

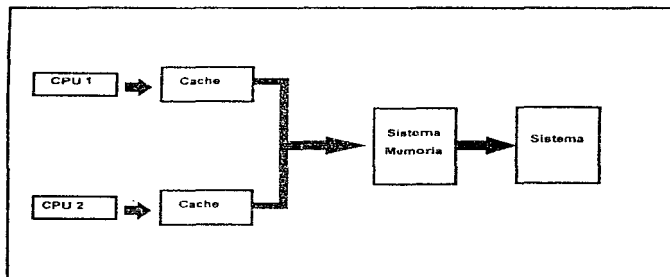


Figura 3.2 MULTIPROCESAMIENTO ACOPLADO RÍGIDO [27]

#### Multiprocesamiento Asimétrico

En un multiprocesamiento asimétrico, los procesadores no son idénticos, es decir, no tienen acceso idéntico a todos los sistemas

La asimetría presenta diferencias en mapas de memoria, restricciones en los diferentes direccionamiento entrada/salida (I/O), ó diferentes capacidades de interrupción. Además permite extender la carga de trabajo a todos los procesadores, pero la asimetría puede llegar a veces en un procesador sobrecargado, mientras que a otro poco cargado.

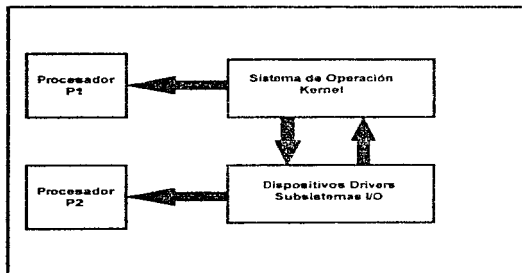


Figura 3.3 MULTIPROCESAMIENTO ASIMÉTRICO [27]

**Multiprocesamiento Simétrico (SMP)**

Este diseño simplifica el problema del balance de la carga de los sistemas de operación, permitiendo desarrollar equitativamente todas las funciones para cualquier procesador. Las interrupciones de diferentes orígenes pueden ser enviadas a cualquier procesador y manejarlas ahí. Los registros de control específicos, tales como los registros de control de interrupciones ó registros de identificación deben estar en la misma dirección física de todos los procesadores.

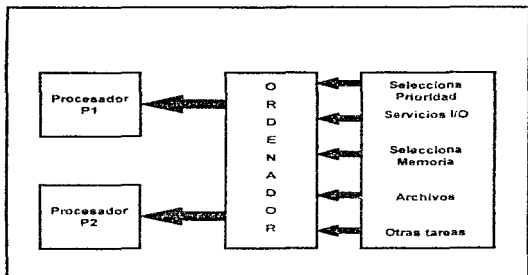


Figura 3.4 MULTIPROCESAMIENTO SIMETRICO [27]

### **3.3.- ARQUITECTURA DE SUBSISTEMAS CACHÉ**

En el multiprocesamiento los sistemas caché reducen el tráfico de bus que en un sistema de bus compartido, incrementando la cantidad de caché entre el procesador y el bus. En un procesador la memoria caché es usada para dar más rapidez de acceso a la memoria principal.

Las tres clases generales de una estructura multiprocesamiento caché son:

- Procesamiento dual-caché compartido.
- Procesamiento dual, doble caché.
- Multiprocesamiento-multichaché.

#### **Procesamiento Dual-Caché Compartido.**

En este sistema el procesador comparte un segundo nivel de la memoria caché. Por lo que son necesarios controladores caché óptimos para minimizar el tiempo de acceso a el segundo nivel. La escalabilidad es limitada por el caché compartido, pero sobre todo por el costo del diseño, que es bajo.

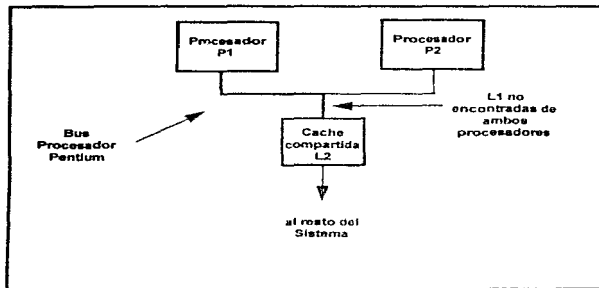


Figura 3.5 PROCESAMIENTO DUAL CACHE-COMPARTIDA [27]

### Procesamiento Dual- DobleCaché

Este diseño se distinguirá del procesamiento dual-cache compartido, por la adición de un segundo nivel caché. Con ambos procesadores operando por separado el segundo nivel de caché externo, la mayoría de los primeros niveles de caché (L1) tienen la necesidad de ser aislados en el segundo nivel externo, por eso se minimiza la frecuencia de acceso a la más lenta memoria principal.

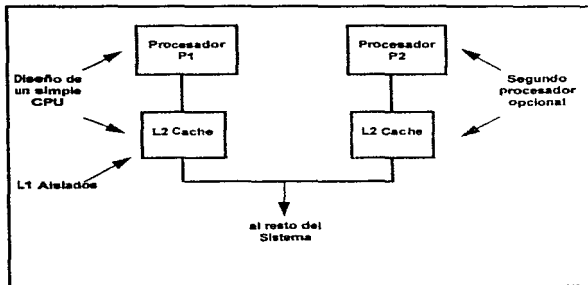


Figura 3.6 PROCESAMIENTO DUAL-CACHE DUAL [27]

### Multiprocesamiento Multi-Caché

En este diseño cada procesador tiene un segundo nivel caché externo por separado. Con todos los procesadores operando por separado sus segundos niveles caché externos, la mayoría de los primeros niveles (L1) tienen la necesidad de ser aislados en el segundo caché, por eso minimiza la frecuencia de acceso a el bus compartido y memoria principal. El objetivo de un diseño multi-caché es dar un alto desempeño y confiabilidad con una eficiente escalabilidad.



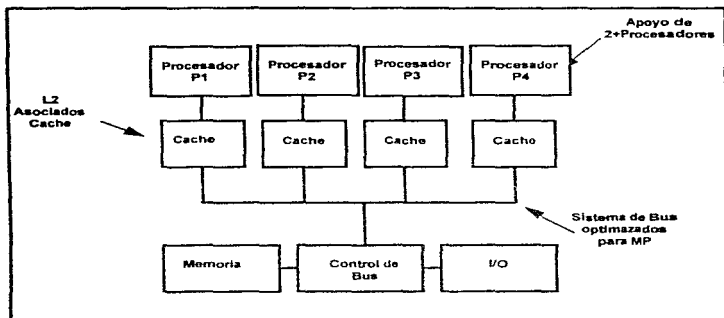


Figura 3.7 MULTIPROCESAMIENTO MULTI-CACHE [27]

## 4. FUNDAMENTOS DE LÓGICA DIFUSA

### 4.1.- INTRODUCCIÓN

La lógica difusa (fuzzy logic) la cual nace en los Estados Unidos alrededor de 1965, en la Universidad de California, en Berkeley y creada por el profesor Lofti A. Zadeh, llega después de la lógica multivaluada propuesta por Jon Lukasiewicz (1930) que propone 3 niveles lógicos [15]: verdadero (1), falso (0) y neutro (1/2), que representa la mitad verdadera o la mitad falsa. Sin embargo, la lógica difusa abarca una infinidad de valores los cuales estarán comprendidos entre "0" (completamente falso) y "1" (completamente verdadero). Las aplicaciones de esta lógica se dividen en tres áreas que son las siguientes[14]:

- a) Propósitos Numéricos: Donde se aplican cálculos y análisis numéricos, como solución de problemas matemáticos referidos a diferentes áreas del conocimiento humano.
- b) Propósitos de Bases de Datos: Donde se almacenan grandes cantidades de información.
- c) Ingeniería de Conocimiento: Donde se encuentran los sistemas expertos, siendo esta el área más actual.

Precisamente la lógica difusa tiene aplicaciones tanto en sistemas expertos, como en sistemas de control, siendo mayor en el área numérica, y la de

Ingeniería. Como una primera aplicación industrial fue en 1977 a la industria del cemento en Dinamarca [17]. Por otra parte en la lógica predictiva, las reglas de subjetividad u objetividad, así como los conocimientos que se deberán tener para simular la experiencia adquirida por aprendizaje de un experto para resolver un determinado problema son expresados por sentencias o un conjunto de estas. Las proposiciones son modelos escritos, así que podemos representar valores ambiguos. **En términos generales los sistemas difusos son convenientes cuando existe incertidumbre o razonamiento aproximado ó cuando los modelos matemáticos que describen el comportamiento del sistema son demasiados complejos para modelar** [16].

#### 4.2.- CARACTERÍSTICAS PRINCIPALES DE LA LÓGICA DIFUSA.

Dentro de los principales parámetros que caracterizan a la lógica difusa podemos mencionar los siguientes:

-Capacidad de Aprendizaje. Donde se añaden ciertos datos para configurar un mejor control.

-Reglas de Inferencia. Las cuales, están hechas en base a un determinado conocimiento del problema y a la experiencia para resolverlo.

Es decir la lógica difusa está basada en razonamientos aproximados por manipulaciones simbólicas y cálculos numéricos y esto a su vez por conceptos ambiguos, como caliente, muy caliente, viejo, joven etc. llamadas

variables lingüísticas. Mientras que la teoría de Probabilidad esta basada en un razonamiento exacto, por manipulación simbólica y cálculos numéricos y esto a su vez por predicciones. De esta forma podemos formar una estructura constituida por varios pasos para desarrollar un sistema difuso como es lo siguiente [14], [15], [16]:

- Determinar si la estructura del problema a resolver amerita el uso de lógica difusa.
- Si es así ,determinar el rango en donde se manejan las variables de entrada y de salida.
- Definir las funciones que constituyen las salidas y las entradas, siendo estas las posibles opciones que tomarán las entradas y salidas.
- Construir las reglas que relacionarán las entradas y salidas.
- La determinación de dichas reglas requiere de pruebas exhaustivas para poder verificar su correcto funcionamiento.

Los sistemas de Control Difuso en términos generales son estables, aunque la estabilidad de las reglas nos dan una estabilidad parcial con respecto a cada regla, para que el sistema sea estable, todas sus reglas en conjunto deberán ser estables.

### 4.3.- FUNDAMENTOS MATEMÁTICOS DE LA LÓGICA DIFUSA

Empezaremos describiendo los elementos involucrados en la teoría de conjuntos convencionales y posteriormente los de la teoría de conjuntos difusos [15], [17].

**Conjuntos Convencionales.** Con respecto a los conjuntos convencionales se puede mencionar que los valores o están dentro del conjunto o no lo están, de acuerdo al siguiente modelo .

$$X(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases} \quad \text{ec.4.1}$$

Como ejemplo, si hablamos de divisiones que caractericen la estatura de una persona bajo este esquema, el conjunto estatura se puede dividir, por ejemplo, en 3 subconjuntos:

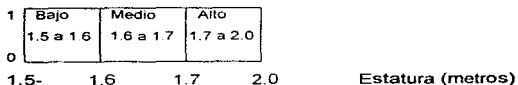


Figura 4.1 CONJUNTO ESTATURA

A dichas divisiones se les llamará conjuntos crisp. En donde cada conjunto crisp contiene un determinado rango de valores, teniendo en común un solo valor.

En donde se pueden aplicar las operaciones básicas y para cada una de ellas se tendrá su representación gráfica, en forma general será lo siguiente [13]:

Unión:  $\{x/x \in A \text{ or } x \in B\} = A \cup B$

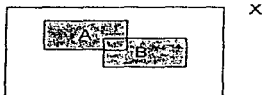


Figura. 4.2 UNIÓN DE DOS CONJUNTOS

Intersección:  $\{x/x \in A \text{ and } x \in B\} = A \cap B$

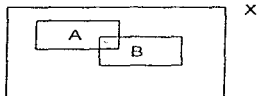


Figura 4.3 INTERSECCIÓN

Complemento:  $\{x/x \notin A, x \in X\} = \bar{A}$

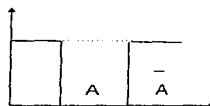


Figura.4.4 COMPLEMENTO DE A

Diferencia:  $A/B = \{x/x \in A \text{ y } x \notin B\}$

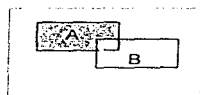


Figura 4.5a DIFERENCIA A/B

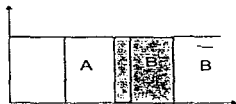


Figura 4.5b DIFERENCIA A/B

Donde se tienen las siguientes propiedades:

Commutatividad =  $AuB = BuA$

$$AnB = BnA \dots$$

ecs.4.2

Asociatividad:  $Au(BuC) = (AuB)uC$

$$An(BnC) = (AnB)nC \dots$$

ecs.4.3

Distributividad:  $Au(BnC) = (AuB)n(AuC)$

$$An(BuC) = (AnB)u(AnC) \dots$$

ecs.4.4

Idempotencia:  $AuA = A$

$$AnA = A \dots$$

ecs.4.5

Identidad:  $Au\emptyset = A$

$$AnX = A$$

$$An\emptyset = \emptyset$$

$$AuX = X \dots$$

ecs.4.6

Transitividad: Si  $A \subseteq B \subseteq C$ , entonces  $A \subseteq C$



Involución:  $A = A'$ .....

ec.4.7

Ley de la exclusión media:

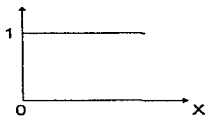


Figura 4.6  $A \cup A' = X$

Ley de la Contradicción:



Figura 4.7  $A \cap A' = \emptyset$

Leyes de Morgan:  $(\overline{A \cap B}) = \overline{A} \cup \overline{B} = \{x/x \notin A \text{ or } x \notin B\}$

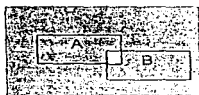


Figura 4.8  $\overline{A \cap B}$

$(\overline{A \cup B}) = \overline{A} \cap \overline{B} = \{x/x \notin A \text{ y } x \notin B\}$

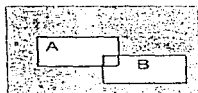


Figura 4.9  $\overline{A \cup B}$

Para ver la relación entre dos conjuntos crisp  $F$  y  $G$ , la unión y la intersección será representada por:

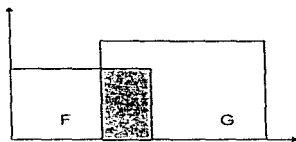


Figura 4.10 INTERSECCIÓN DE F y G

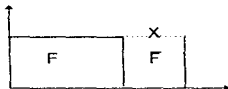


Figura 4.11 COMPLEMENTO DE F

Definiendo el criterio máximo y el criterio mínimo por las siguientes ecuaciones:

$$A \cup B = X_{A \cup B}(x) = X_A(x) \vee X_B(x) = \max(X_A(x), X_B(x))$$

ecs 4.8

$$A \cap B = X_{A \cap B}(x) = X_A(x) \wedge X_B(x) = \min(X_A(x), X_B(x))$$

#### 4.3.1.-CONJUNTOS DIFUSOS.

Su representación es por medio de la siguiente expresión, donde podrán existir una infinidad de valores entre 0 Y 1 [13]:

$$\mu(x) \in [0,1] \quad \text{ec. 4.9}$$

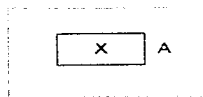
Los conjuntos difusos tendrán los siguientes elementos [12]:

- Función de pertenencia. Es el grado en el cual el elemento  $\mu(x)$  pertenece al sub-conjunto A y estará dividido en varios atributos (Fuzzy sets).

$$\mu_A(x) = \text{grado}(x \in A) \dots \dots \quad \text{ec.4.10}$$

-Conjunto (set) X, es el universo, el cual representa todos los valores posibles del sistema.

-Subconjunto (subset) A son valores que se encuentran dentro del conjunto X.



**Figura 4.12 SUBCONJUNTO A**

Donde  $\mu: X \rightarrow [0,1]$  podrá tener valores entre 0 y 1

Si hablamos de divisiones que caractericen por ejemplo la edad, como en el caso anterior, el subconjunto edad puede ser dividido en 3 subconjuntos, teniendo estos subconjuntos la forma triangular, trapezoidal o bien de campana [20].

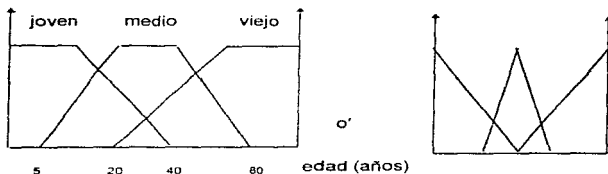


Figura 4.13 SUBCONJUNTO EDAD

A dichas divisiones se les llamara conjuntos difusos. Por otra parte definiendo convexo a un conjunto difuso tendrá que cumplir lo siguiente [29]:

$$\mu(X) \geq \min [\mu_A(X), \mu_A(Z)] \dots \dots \dots \text{ec.4.11}$$

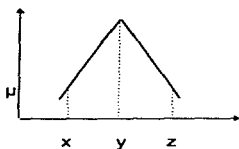


Figura 4.14 CONVEXA

De lo contrario será no convexo:

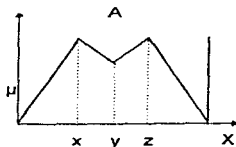


Figura 4.14 NO CONVEXA

Para ejemplificar la relación de los conjuntos difusos, se tienen los conjuntos difusos A, B, C, cuyas operaciones básicas estarán dadas de la siguiente forma [15]:

Unión:

$$\mu(x) = \mu_A(x) \vee \mu_B(x) \dots$$

ec.4.12

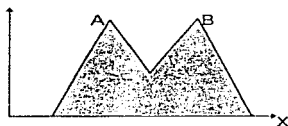


Figura 4.16 UNIÓN DE CONJUNTOS DIFUSOS

Intersección:

$$\mu_{A \wedge B}(x) = \mu_A(x) \wedge \mu_B(x) \dots$$

ec.4.13

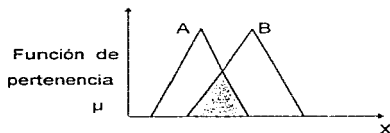


Figura 4.17 INTERSECCIÓN DE CONJUNTOS DIFUSOS

Complemento :

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \dots\dots$$

ec.4.14



Figura 4.18 COMPLEMENTO DE A

Diferencia:

$$\mu_{A-B}(x) = \mu_A(x) \wedge \mu_{\bar{B}}(x) \dots\dots$$

ec.4.15

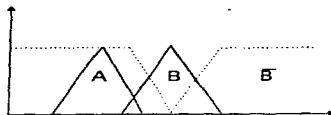


Figura 4.19 DIFERENCIA DE CONJUNTOS DIFUSOS

Donde se tienen las siguientes propiedades:



**Conmutatividad**

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \mu_{B \cup A}(x) = \mu_B(x) \vee \mu_A(x)$$

ecs.4.16

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \mu_{B \cap A}(x) = \mu_B(x) \wedge \mu_A(x)$$

Las propiedades de Asociatividad, Distributividad, Idempotencia, Identidad, Transmitividad e Involución son exactamente igual para los conjuntos crisp que para los difusos.

Ley de la exclusión media:

$$A \cup \bar{A} = X \dots\dots\dots$$

ec.4.17

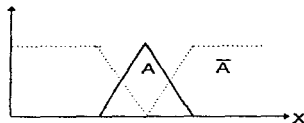


Figura 4.20 LEY DE LA EXCLUSIÓN MEDIA

Ley de la contradicción :

$$A \cap \bar{A} = \varnothing \dots\dots\dots$$

ec.4.18

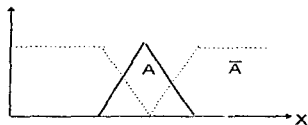


FIG.4.21 LEY DE LA CONTRADICCIÓN

Ley de Morgan:  $\mu_{A \rightarrow B}(x) = \mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x)$ ..

ec.4.19



Figura 4.22 LEY DE MORGAN

$\mu_{A \cap B}(x) = \mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x)$ .....

.ec.4.20

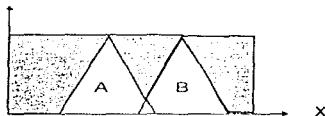


Figura 4.23 LEY DE MORGAN

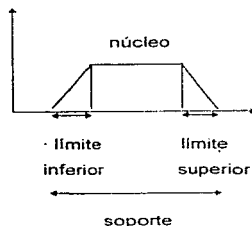
Definiendo  $\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) = \max [\mu_A(x), \mu_B(x)]$

ec.4.21

$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) = \min [\mu_A(x), \mu_B(x)]$

### 4.3.2.- ELEMENTOS DE UNA FUNCION MIEMBRO.

De acuerdo a la siguiente figura se tendrán los siguientes elementos[13]:



**Figura 4.24** ELEMENTOS DE UNA FUNCIÓN MIEMBRO

Donde el subconjunto A del conjunto X estará definido por:

$$A = \mu_A(x) / X_1 + \mu_A(x) / X_2 + \dots + \mu_A(x) / X_n = \sum_{i=1}^n \mu_A(x) / X_i = \int_k \mu_A(x) / X$$

ec.4.22

Aquí los elementos de soporte están en el numerador y el grado de pertenencia en el denominador. Usando esta última ecuación podemos expresar la unión, la intersección y el complemento de la siguiente forma [12]:

$$A \cup B = \int_x [ (\mu_A(X) \vee \mu_B(X)) / X ] \dots\dots\dots \text{ec.4.23}$$

$$A \cap B = \int_x [ \mu_A(X) \wedge \mu_B(X) ] / X \dots\dots\dots \text{ec.4.24}$$

$$\overline{A} = \int_x [ 1 - \mu_A(X) ] / X \dots\dots\dots \text{ec.4.25}$$

#### 4.3.3.-CONJUNTOS DIFUSOS VISTOS DESDE SU PERSPECTIVA GEOMÉTRICA.

Dados todos los puntos pertenecientes a un conjunto de vectores tales que se encuentran dentro de un cubo que representan un subconjunto A del set X del conjunto total de tal forma que "0" indica la ausencia de X en el vector de bits. Considere un conjunto difuso de dimensión 2 donde F(2) corresponde a la unidad cuadrada donde  $X = \{X_1, X_2\}$ , y los valores límite que son los esquinas de los cuales se les denominará conjuntos no difusos [16].

$$\begin{aligned} \phi &= (0,0) \\ \{X_1\} &= (1,0) \\ \{X_2\} &= (0,1) \\ X \dots &= (1,1) \end{aligned} \qquad \text{ecs.4.26}$$

El punto A será un subconjunto A correspondiente a los valores siguientes:

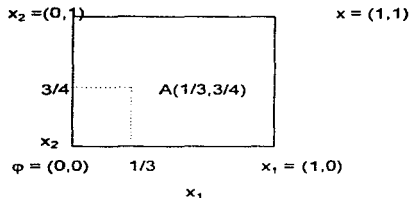


Figura 4.25 SUBCONJUNTO A

A partir de este esquema podemos extender de una dimensión 2 a una dimensión 3 donde se tendrá un subconjunto local, donde este mide el grado en el cual el conjunto pertenece al subconjunto B, que es el grado en el cual A es subconjunto de B. Denotaremos esta medida como  $S(A,B)$  [16]:

$$S(A,B) = \text{grado}(A \subset B) \quad \text{ec.4.27}$$

Entonces el subconjunto local de  $\{X\}$  en B igual al miembro o valor de elemento  $m(X)$  será [20]:

$$S(\{X\}, B) = m_B(X_i) \quad \text{ec.4.28}$$

El subconjunto local muestra la relación entre ambigüedad y la aleatoriedad. La aleatoriedad no depende de ambigüedad de un evento, esto depende de la incertidumbre entre eventos ciertos. Es decir la aleatoriedad se iguala a la incertidumbre cuando un subconjunto no difuso B es parcialmente contenido en

uno de los subconjuntos no difusos A. Si  $S(A,B) = 1$ , significa que A es un subconjunto de B aunque podrá variar entre cero y uno. ( $0 < S(B,A) < 1$ ).

En cierta forma la teoría difusa se deriva de los axiomas de la medición de la probabilidad condicional [16].

$$P(B/A) = P(A \cap B) / P(A) \qquad \text{ec.4.29}$$

La probabilidad que B ocurra dado que A ocurra, desde las propiedades de el subset local mide  $S(A,B)$ . Donde X define el "simple espacio" de todos los elementos de un experimento, entonces X es un evento seguro desde  $P(X) = 1$ . Entonces  $P(A) = P(A/X)$  o para el subset local  $P(A) = S(X,A)$ .

En general  $S(A,B) = M(A \cap B) / M(A)$ ..... ec.4.29a denota las cantidades difusas del vector que define a A.  $M(A) = a + \dots + a$  .... ec.4.30 esta ecuación tendrá valores entre "0" y "1". De aquí podemos definir la entropía difusa como la relación entre el traslape y no traslape, de la ley de la contradicción y exclusión media  $A \cap A^c \neq 0$  y  $A \cup A^c \neq X$  [16].

$$E(A) = M(A \cap A^c) / M(A \cup B^c)$$

ec.4.31

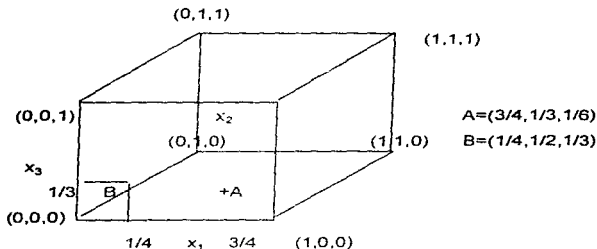
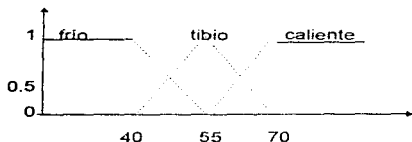


Figura 4.26 EL SUBCONJUNTO A EN UNA TERCERA DIMENSION

#### 4.3.4.-VARIABLES LINGÜÍSTICAS Y RAZONAMIENTO APROXIMADO

Una variable lingüística es el nombre que se le da a una característica que representa "algún" conocimiento sobre "algo". Esta variable es un subconjunto del conjunto  $X$  universal, en donde a la variable se le atribuyen conjuntos difusos, donde las funciones miembro indican  $m(X)$ , que es el grado de pertenencia a cada uno de los conjuntos difusos involucrados [14]. Una variable lingüística podrá ser caracterizada por una función con 5 elementos ( $X$ ,  $T(X)$ ,  $U, G, M$ ) en donde  $X$  es el nombre de la variable;  $T(X)$  es la característica o el

término set de  $X$ , que es el conjunto de nombres de valores lingüísticos o características, los cuales describirán algún conocimiento de "algo", siendo cada uno de ellos un conjunto difuso, definidos en el universo  $U$  y  $G$  es una regla sintáctica para generar nombres de valores de  $X$  y  $M$  en una regla semántica para asociar con cada valor el significado. De tal forma que  $M(X)$  denota un subconjunto de  $U$ . Por ejemplo la temperatura es interpretada como una variable lingüística donde estarán los conjuntos difusos involucrados con esta variable  $T(\text{temperatura}) = \{\text{helado, frío, tibio, templado, caliente, etc.}\}$  donde el universo estará definido por  $U=[0, 100]$  definiendo los límites para cada conjunto difuso como sigue [14]:



**Figura 4.27** REPRESENTACIÓN DE LOS CONJUNTOS DIFUSOS DE LA VARIABLE TEMPERATURA

Otros ejemplos pueden ser:

$T(\text{peso}) = \text{delgado} + \text{muy delgado} + \text{medio} + \text{robusta} + \text{muy robusta} + \text{obesa, etc.}$

ec.4.32



$T(\text{apariencia}) = \text{bonita} + \text{atractiva} + \text{fea} + \dots + \text{más o menos}$   
 $\text{bonita} + \text{horrible} + \dots + \text{etc, etc, etc.}$

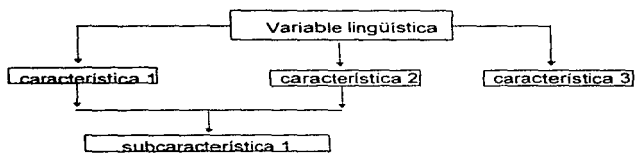
ec.4.32

Por otra parte es conveniente ejemplificar el concepto de característica mencionado anteriormente esto puede hacerse de la siguiente forma: supongamos que hablamos de las características mas comunes de un equipo de trabajo que quedarán conformadas de la siguiente forma:



**Figura 4.28** CONJUNTOS DIFUSOS DE UN EQUIPO

Dichas características serán los conjuntos difusos del conjunto equipo. Por otra parte la estructura jerárquica de una variable lingüística podrá ser visualizada a continuación donde se puede apreciar que una subcaracterística puede pertenecer a más de una característica : nivel 2 CARACTERÍSTICA 1  
 CARACTERÍSTICA 2p CARACTERÍSTICA 3p



**Figura 4.29** ESTRUCTURA JERÁRQUICA DE UNA VARIABLE LINGÜÍSTICA

## **5.- TRANSDUCTORES Y ACTUADORES**

### **5.1.- INTRODUCCIÓN.**

Los sistemas de control son un hecho fundamental en la industria moderna y en la sociedad. Aplicaciones tales como el control del aire acondicionado de un edificio, el control de un elevador, ó el control de aparatos domésticos, etc. es posible usando dispositivos de control, en donde el dispositivo de salida son los actuadores que accionan de acuerdo a un procesador y los dispositivos de entrada son los sensores que responden a estímulos de luz, presión, calor, movimiento, etc.

### **5.2.- CLASIFICACIÓN DE LOS TRANSDUCTORES.**

Un transductor es un dispositivo que recibe energía de un sistema y la transmite a menudo en forma diferente a otro sistema. El término transductor denota entonces, la magnitud de un estímulo aplicado y convertido a una señal eléctrica proporcional a la cantidad del estímulo. De otra manera, un sensor está definido como un dispositivo que es sensible a la luz, temperatura, impedancia eléctrica o nivel de

radiación, y transmite una señal a un dispositivo de control [24]. La cantidad de entrada en la mayoría de los sistemas de instrumentación no es eléctrica. Para utilizar métodos eléctricos para la medición, manipulación o control, la cantidad no eléctrica debe convertirse a una señal eléctrica por medio de un transductor de tal forma que el transductor se puede definir también como el dispositivo que actuado por energía en un sistema de transmisión, suministra energía en la misma forma o en otra a un segundo sistema de transmisión. La energía que se transmite puede ser eléctrica, mecánica, química, óptica, térmica, etc.

Un transductor es parte de un sistema de instrumentación electrónico, que consiste generalmente de tres elementos principales: un dispositivo de entrada, un procesador (acondicionador de la señal) y un dispositivo de salida. Los transductores se pueden clasificar de acuerdo a su aplicación, método de conversión de energía, y del tipo de señal de salida; a continuación se muestra una clasificación general de acuerdo a los principios eléctricos que involucran [26]:

TRANSDUCTORES PASIVOS

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
Galga Extensiométrica - Resistiva	La resistencia de un alambre o semiconductor se modifica por elongación debido a esfuerzos externos aplicados	Fuerza Torque(par)
- Termómetro de Resistencia	La resistencia de un alambre de metal duro cambia con la temperatura (el metal debe tener un coeficiente resistivo de temperatura positivo)	Temperatura Calor Radiante
- Higrómetro Resistivo	La resistencia de una cinta conductora cambia con la humedad	Humedad Relativa
- Celda Fotoconductora	La resistencia de la celda varía con la luz incidente. (la celda es parte de un circuito)	Relé fotosensible
- Termistor	La resistencia de un óxido de metal con un coeficiente resistivo de temperatura negativo, cambia con la temperatura	Temperatura
- Medidor de Alambre Caliente	La resistencia de un elemento caliente varía al enfriarse por medio de un chorro de gas	Flujo de Gas Presión de Gases
- Equipo Potenciométrico	El posicionamiento de un cursor por medio de una fuerza eléctrica varía la resistencia eléctrica de un circuito puente o un potenciometro	Presión Desplazamiento

Tabla 5.1 TRANSDUCTORES CON PRINCIPIO DE OPERACION: LA RESISTENCIA [26]

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
- Galga de Presión de Capacitancia Variable	La distancia entre dos placas paralelas se modifica al aplicar una fuerza externa.	Desplazamiento Presión
-Micrófono de Condensador	La presión del sonido modifica la distancia entre una placa fija y un diafragma móvil.	Voz Música Ruido
- Galga Dieléctrica	Cambio en la capacitancia por cambios dieléctricos.	Nivel de líquidos Espesor

Tabla 5.2 TRANSDUCTORES CON PRINCIPIO DE OPERACION: LA CAPACITANCIA [26]

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
- Transductor de Circuito Magnético	La inductancia mutua de una bobina excitada con AC se modifica cambiando su circuito magnético.	Presión Desplazamiento
- Detector de Reluctancia	La reluctancia de un circuito magnético se modifica variando la posición del núcleo de hierro de una bobina.	Presión Desplazamiento Vibración Posición

Tabla 5.3 TRANSDUCTOR CON PRINCIPIO DE OPERACION: LA INDUCTANCIA [26]

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
- Transformador Diferencial	El voltaje diferencial de dos devanados secundarios de un transformador se modifica variando la posición de un núcleo magnético por medio de una fuerza externa aplicada.	Presión Fuerza Desplazamiento Vibración
- Galga de Corrientes Inducidas	La inductancia de una bobina se modifica acercándole un plato con corrientes inducidas	Desplazamiento Espesor
- Galga Magnetométrica	Las propiedades magnéticas se modifican por presión y esfuerzos mecánicos	Fuerza Presión Sonido

Tabla 5.3 continuación TRANSDUCTOR CON PRINCIPIO DE OPERACION LA INDUCTANCIA [26]

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
- Detector por Efecto Hall	Una diferencia de potencial se genera a través de una placa de germanio cuando el flujo magnético interactúa con una corriente aplicada.	Flujo Magnético Corriente

Tabla 5.4 TRANSDUCTORES QUE EMITEN UN VOLTAJE Y CORRIENTE[26]

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
- Cámara de Ionización	Flujo electrónico inducido por ionización de un gas debido a radiación radiactiva.	Conteo de Partículas Radiación Luz y Radiación
- Celda Fotoemisiva	Emisión electrónica debido a radiación incidente sobre superficies fotoemisivas.	Luz y Radiación

**Tabla 5.4** continuación TRANSDUCTORES QUE EMITEN UN VOLTAJE Y CORRIENTE[26]

**TRANSDUCTORES DE AUTOGENERACION**

TIPO DE TRANSDUCTOR	PRINCIPIO DE OPERACIÓN	APLICACIÓN
- Termopar	Se genera una fem cuando la unión de dos metales diferentes se expone al calor.	Temperatura Flujo de Calor Radiación
- Generador de Bobina Móvil	El movimiento de una bobina dentro de un campo magnético genera un voltaje.	Velocidad Ubicación
- Detector Piezoeléctrico	Se genera una fem cuando se aplica una fuerza externa a ciertos materiales como el cuarzo.	Sonido Vibración Cambios de Presión Aceleración

**Tabla 5.5** TRANSDUCTORES QUE GENERAN VOLTAJE [26]



### **5.3.- TRANSDUCTORES DE PRESIÓN Y TEMPERATURA**

El transductor debe ser físicamente compatible con su función. En la operación de un transductor existen ocho importantes parámetros [24]:

1. El principio de operación usado.
2. Voltaje externo y/o corriente aplicada al transductor para hacer su trabajo.
3. La salida eléctrica del transductor.
4. Repetitividad del transductor para reproducir las lecturas de salida bajo toda las condiciones ambientales.
5. Estabilidad del transductor.
6. Confiabilidad.
7. El transductor sea firme y que no sea fácilmente de quitar.
8. El rango del transductor debe ser suficiente.

Los principios físicos incorporados en el diseño de un transductor de presión depende principalmente del rango de presión a medir. La mayoría de los transductores de presión consiste de un dispositivo que

convierte la presión en un desplazamiento o fuerza y un elemento de transducción eléctrica (resistivo, capacitivo, inductivo piezoelectrónico, etc.) que responde al desplazamiento o fuerza con una salida eléctrica.

A muy altas presiones, éste dispositivo puede ser eliminado usando directamente un elemento sensible eléctrico. También los materiales piezoelectrónicos son usados como dispositivos sensibles a altas presiones.

En cuanto a los transductores de temperatura algunos pueden convertir la expansión de un líquido ó sólido a señales eléctricas a través de medios mecánicos. Otro tipo son los termistores, termoresistencia de platino y el termopar, etc. que al sentir los cambios de temperatura, transmiten una señal eléctrica al procesador. Este tipo de transductores es más utilizado para sistemas de control con microprocesadores.

#### 5.4.- ACTUADORES

Como se mencionó anteriormente en un sistema de control, el dispositivo de entrada recibe la cantidad a medir y le entrega al procesador de la señal una forma de onda eléctrica proporcional a ésta [26]. En el procesador la señal es amplificada o modificada para ser aceptada en el dispositivo de salida. El dispositivo de salida son los actuadores (electroválvulas, relevadores, arrancadores de motor, etc.) que operan de acuerdo a la señal entregada por el procesador.

## **6.- CONSIDERACIONES BÁSICAS EN EL DISEÑO DEL CONTROLADOR DIFUSO.**

### **6.1.- INTRODUCCIÓN.**

Como humanos, pensamos, actuamos, creamos y realizamos diferentes actividades diariamente. Hablamos, leemos y formulamos opiniones basados en la palabra hablada y escrita. Tratamos con términos inexactos y aproximados tales como "poco", "mucho" y "algo". Mucho de nuestro diario quehacer involucra una gran cantidad de pensamiento abstracto vago.

Con la lógica difusa, podemos modelar tales transiciones vagas. Podemos comparar la seriedad de crímenes, la importancia de diferenciar leyes, la variedad de tonos del color verde. Podemos ver y utilizar el área gris que existe entre el negro y el blanco. Estas áreas grises pueden entonces ser comparadas, y una decisión humana puede ser simulada. Esta es la razón porqué, la lógica difusa es usada en control y/o en situaciones de toma de decisiones. Un controlador difuso no es representado por un modelo matemático convencional sino por reglas de inferencia, aunque hay que hacer notar que ya teniendo dichas reglas se pueden trasladar a un modelo discreto para efectos de análisis de comportamiento del sistema. Una desventaja del controlador difuso es que se necesita generar en ocasiones un número importante de reglas de inferencia que simulen el comportamiento del

proceso en una forma lo más precisa posible; esto ocasiona que entre más reglas existan se requiera más tiempo del cálculo y más localidades de memoria.

## **6.2.- CARACTERÍSTICAS DE LOS ALGORITMOS DE CONTROL.**

Los algoritmos de control pueden ser de lazo abierto ó de lazo cerrado, lo que depende de la estructura de control que se tenga, así como del proceso a controlar. Una segunda clasificación es de acuerdo al modelo matemático del controlador, ya sea si es un modelo continuo o un modelo discreto. Otra clasificación se basa en la estructura de control utilizada, como controladores proporcionales, integrales, derivativos, adaptables, robustos, autosintonizables, difusos, etc. Cada uno puede tener más de una característica y ser utilizado para resolver determinadas tareas, de acuerdo con su diseño. Los algoritmos de control que permiten el activado o desactivado de un determinado actuador mediante un rango de lectura de una variable sensada, representa una estructura de un algoritmo difuso en su forma más simple.

### 6.3 ESTRUCTURA DE UN CONTROLADOR DIFUSO.

Un controlador difuso es un sistema el cual usa una máquina de razonamiento y el conocimiento específico de un especialista, para simular el conocimiento experto y para proveer decisiones inteligentes realizando un control óptimo. El diagrama a bloques de un típico sistema de control difuso se muestra en la figura 6.1

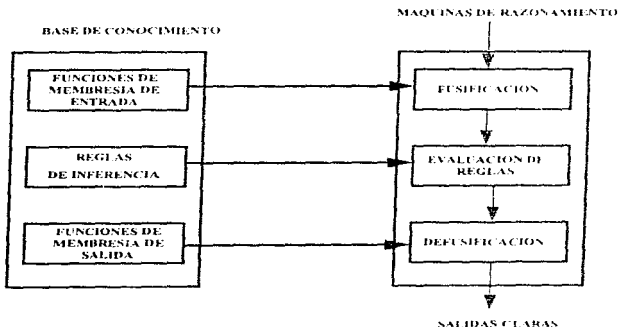


Figura 6.1 DIAGRAMA A BLOQUES DE UN CONTROLADOR DE LÓGICA DIFUSA

**La Base del Conocimiento.-** Está formada por funciones de membresía y reglas de inferencia. La base del conocimiento es usualmente generada y sintonizada manualmente por expertos o se

pueden emplear sistemas que incorporan generación automática de la base de conocimiento y sintonía del sistema.

**La Máquina de Razonamiento.-** Realiza tres funciones principales: Fusificación, Evaluación de Reglas y Defusificación:

- **Fusificación.-** Involucra una transformación de dominio, esto es, entradas claramente definidas son transformadas a entradas difusas, además, se deberán determinar funciones de membresía para cada entrada clara.

- **Evaluación de Reglas.-** Las reglas son combinadas para formar una tabla de decisiones. Esta tabla consiste de valores que muestran las diferentes situaciones experimentadas por el sistema y la correspondiente entrada de control.

- **Defusificación** - Es similar a la fusificación; a partir de las entradas difusas y de acuerdo a la tabla de decisiones se obtiene las salidas difusas. Estos valores toman inmediatamente después un valor de salida en tiempo real.

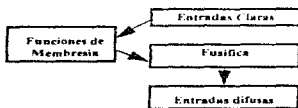


Figura 6.2 PASO DE FUSIFICACION

#### 6.4.- DESCRIPCIÓN GENERAL DEL PROCESO.

En un control de nivel, el nivel del líquido actual, se compara con el nivel deseado, dependiendo de esto se abre o cierra la (s) válvula (s) para mantener el nivel deseado. Nuestro proceso esta diseñado para el control y monitoreo del nivel del líquido de dos tanques cerrados interconectados entre sí, como se muestra en la fig. 6.3 Los datos de entrada son la presión y la temperatura que crea la columna de líquido en cada tanque, por lo que se usaran cuatro sensores: dos de temperatura y dos de presión.

Los sensores de presión y temperatura del primer tanque ( $P_1, T_1$ ), activan dos actuadores que corresponden a una válvula ( $V_1$ ) y a un interruptor ( $Y_1$ ) que energiza el arrancador eléctrico del motor que acciona la bomba; por consiguiente los sensores de presión y temperatura del segundo tanque ( $P_2, T_2$ ), activan dos actuadores que corresponden a dos válvulas ( $V_2, V_3$ ).



Figura 6.3 ESQUEMA DEL CONTROL DEL NIVEL

## 6.5.- ALGORITMO DIFUSO.

El algoritmo difuso utilizado en nuestro proceso tiene cuatro entradas: Presión 1(P1), Temperatura 1(T1), Presión 2(P2) y Temperatura 2 (T<sub>2</sub>) y dos salidas que corresponden al nivel de líquido del primer tanque (H<sub>1</sub>) y al nivel de líquido del segundo tanque (H<sub>2</sub>), de acuerdo al proceso. Las entradas están agrupadas en pares(presión y temperatura) y de acuerdo a los valores que toman, se tiene un valor de salida con la cual se evalúan los actuadores.

### 6.5.1.- FUSIFICACIÓN.

Después de definir las entradas y salidas de nuestro proceso, representaremos éstas en conjuntos difusos (fuzzy sets). A estos elementos les hemos asignado una forma triangular en la parte del centro y una forma de trapecio en los extremos. Esta organización va de acuerdo con nuestro planteamiento . Ver figura 6.4.



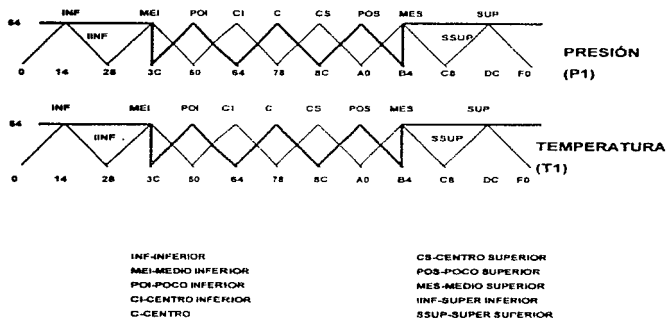


Figura 6.4 CONJUNTOS DIFUSOS DE LAS ENTRADAS P1,T1  
(DATOS NUMERICOS EN CÓDIGO HEXADECIMAL)

Cada elemento tiene una magnitud que está entre cero y uno. Esto es el grado de pertenencia en el cual el elemento existe. También podemos apreciar algunos elementos que se traslapan de tal forma que habrá valores que corresponderán tanto a un elemento como a otro, esto desde el punto de vista geométrico, pero que tendrán como consecuencia un incremento en exactitud, ya que en el momento de defusificar, tendremos un número mayor de datos a procesar, y

logrando una mayor precisión en el valor numérico de la salida. Para las entradas ( $P_2, T_2$ ) sucede exactamente lo mismo (figura 6.5) Para la salida ( $H_2$ ) también hemos asignado elementos triangulares con elementos de traslape en el centro (figura 6.6).

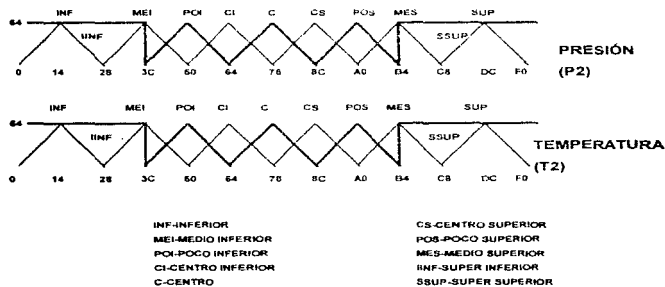


Figura 6.5 CONJUNTOS DIFUSOS DE LAS ENTRADAS  $P_2, T_2$ .  
(DATOS NUMERICOS EN CODIGO HEXADECIMAL)

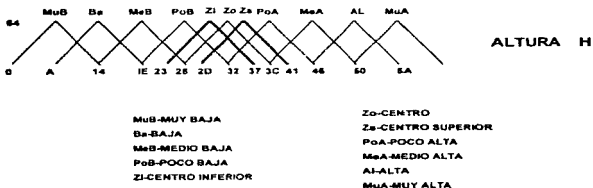


Figura 6.6 CONJUNTOS DIFUSOS DE SALIDA (H)  
(DATOS NUMERICOS EN CÓDIGO HEXADECIMAL)

### 6.5.2.- GENERACIÓN DE LAS REGLAS DE INFERENCIA

Estas reglas son la base del sistema, se generan a partir de valores que muestran las diferentes situaciones experimentadas por el sistema.

Estas reglas son implementadas como una serie de expresiones si.....entonces. Las diferentes combinaciones nos dan las diferentes posibles salidas difusas. Las reglas que estamos utilizando tienen como condiciones de entrada una presión y una temperatura y como consecuencia a la salida el nivel de líquido de cualquier tanque.

Enseguida presentamos un ejemplo de algunas reglas generadas:

Si la presión es SUP y la temperatura es SUP, MES, POS, CS, CO, IIT entonces la salida es 10.

Si la presión es MES y la temperatura es SUP, MES, POS, CS, CO entonces la salida es 9.

Si la presión es MES y la temperatura es CI, POI, MEI, INF entonces la salida es 8.

Si la presión es POS y la temperatura es SUP, MES, POS, CS, CU entonces la salida es 8.

Si la presión es POS y la temperatura es CI, POI, MEI, INF entonces la salida es 7.

Si la presión es CS y la temperatura es SUP, MES, POS, CS, IIT entonces la salida es 7.

Si la presión es CS y la temperatura es CO, CI, POI, MEI, INF entonces la salida es 6.

Si la presión es CO y la temperatura es SUP, MES, POS, CS, CO, CI POI, MEI, INF, IIT entonces la salida es 5.

Si la presión es CI y la temperatura es SUP, MES, POS, CS, CO, entonces la salida es 4.

Si la presión es CI y la temperatura es CI, POI, MEI, INF entonces la salida es 3.

Si la presión es POI y la temperatura es SUP, MES, POS, CS, CO, entonces la salida es 3.

Si la presión es POI, CI, POI, MOI, INF y la temperatura es POS entonces la salida es 2.

Si la presión es MEI y la temperatura es SUP, MES, POS, CS, CO, entonces la salida es 2.

Si la presión es MEI y la temperatura es CI, POI, MEI, INF, entonces la salida es 1.

Si la presión es INF y la temperatura es SUP, MES, POS, CS, CO entonces la salida es 1.

Si la presión es INF y la temperatura es CI, POI, MEI, INF entonces la salida es 0.

En la figura 6.7 se muestra la matriz generada de acuerdo a las reglas citadas anteriormente, en donde los renglones representan los conjuntos difusos relacionados con la temperatura y las columnas a la presión. En las intersecciones se localizan los conjuntos difusos de salida.

Como en el caso de las variables de entrada, en los conjuntos difusos de salida existen traslapes, teniendo mayor cobertura en la variable de salida. Se debe hacer notar que entre más traslapes existan se tendrá mayor exactitud del valor de salida, aunque el cálculo se complica, debido a que aumentan los elementos para obtener el valor numérico en la etapa de defusificación. Además se genera una mayor experiencia incrementando el número de reglas de inferencia ya que se aumenta el número de posibilidades. Para el caso de segundo tanque sucede exactamente lo mismo.

CONSIDERACIONES BASICAS EN EL DISEÑO DEL CONTROLADOR DIFUSO

PRESIÓN

IP	INF	MEI	POI	CI	CO	CS	POS	MES	SUP	SSP	SST	T
1	1	2	3	4	5	7	8	9	10	10	SUP	T
2	1	2	3	4	5	7	8	9	10	10	SUP	E
2	1	2	3	4	5	7	8	9	10	10	MES	M
2	1	2	3	4	5	7	8	9	10	10	POS	P
1	1	2	3	4	5	7	8	9	10	9	CS	E
1	1	2	3	4	5	6	8	9	10	9	CO	R
1	0	1	2	3	5	6	7	8	9	9	CI	A
0	0	1	2	3	5	6	7	8	9	8	POI	T
0	0	1	2	3	5	6	7	8	9	8	MEI	U
0	0	1	2	3	5	6	7	8	9	8	INF	R
0	1	2	3	4	5	7	8	9	10	8	IP	A

- 0-Muy abajo
- 1-Abajo
- 2-Medio Abajo
- 3-Poco Abajo
- 4-Centro Inferior
- 5-Centro
- 6-Centro Superior
- 7-Poco Arriba
- 8-Media Arriba
- 9-Arriba
- 10-Muy Arriba

Figura 6.7 MATRIZ DE INFERENCIA

### 6.5.3.- DEFUSIFICACIÓN.

#### 6.5.3.1.- DETERMINACIÓN DE LAS ALTURAS

El método utilizado es el criterio de mínimos; en el que se escoge el mínimo valor de altura que corresponde al valor numérico de la variable de entrada. Esto es, la altura mínima que se obtiene de la intersección de las regiones al trazar una línea vertical al partir de la variable de entrada por ejemplo localizamos el punto X ( $P=7E, T=70$ ). Vease la siguiente figura.

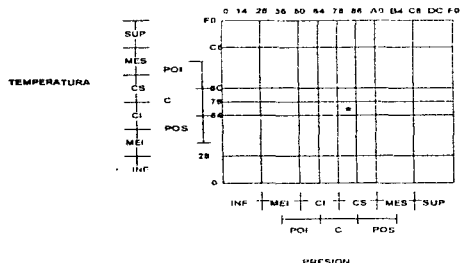


Figura 6.8 UBICACION DEL PUNTO X (7E 70)

Se traslada las variables de entrada a los respectivos conjuntos difusos a partir de estos valores se traza una línea vertical para interceptar la o las regiones a las que pertenecen obteniendo la o las alturas correspondientes a dichos valores numéricos, por último las

alturas se comparan de una variable con las alturas de la otra variable para obtener la mínima altura. Ver Fig. 6.9.

### 6.5.3.2.- DETERMINACIÓN DEL VALOR NUMÉRICO DE LA SALIDA

Una vez obtenidas las alturas mínimas se procede a determinar de acuerdo a las reglas de inferencia a que región de salida pertenece el dato. Con los datos de las alturas mínimas y el centroide de cada región, podemos obtener el centroide de las regiones determinadas que corresponde a el valor numérico de la salida. Del ejemplo:

$$I = 28(32) + 19(37) / (28 + 19) = 33$$

### 6.6.- ESTABILIDAD DEL CONTROLADOR DIFUSO

Condiciones para que un controlador difuso sea estable [18],[19]:

- 1.- Si al aplicar una regla o conjunto de ellas la salida no diverge, y tiende a un estado de equilibrio, mientras la entrada no cambie.
- 2.- Que al aplicar una regla ó conjunto de ellas no genere diferentes salidas ó alternancia de valores para una misma entrada.
- 3.- Entre las reglas no existan conflictos ó dualidades siendo la relación siempre única.
- 4.- Para todas y cada una de las reglas el punto uno y el punto dos.





## **7.- IMPLEMENTACIÓN DEL SISTEMA PROPUESTO.**

### **7.1.- INTRODUCCIÓN.**

Nuestro controlador esta formado por tres microcontroladores trabajando en paralelo, incrementándose así, la rapidez y el algoritmo. El arreglo está formado por dos PIC16C74 de Microchip y un MC68HC11A8 de Motorola. Los primeros por sus características de velocidad e integración de unidades funcionales y el segundo por la simplicidad e integración(ver apéndice), permiten tener desde el punto de vista hardware arquitecturas pequeñas.

El algoritmo de control consta de dos etapas principales: Fusificación y Defusificación, las cuales fueron asignadas en las dos etapas de la arquitectura, de tal manera de lograr el mínimo tiempo de procesamiento y además optimizar el uso de espacio de memoria.

### **7.2.- DESCRIPCIÓN GENERAL DE LA ARQUITECTURA DEL CONTROLADOR.**

La arquitectura que conforma el controlador es una arquitectura en paralelo de bus compartido, formada por dos microcontroladores PIC16C74 y un MC68HC11A8. Los PIC16C74 toman los valores de las variables de entrada y realizan el proceso de fusificación para

posteriormente entregar los valores difusos al MC68HC11A8. Éste realiza la defusificación, obteniendo las indicaciones necesarias para activar los actuadores, así como mantener la comunicación con el monitor central como lo muestra la figura 7.1. Nuestro algoritmo de control fue repartido para cubrir de manera más óptima las necesidades del sistema.

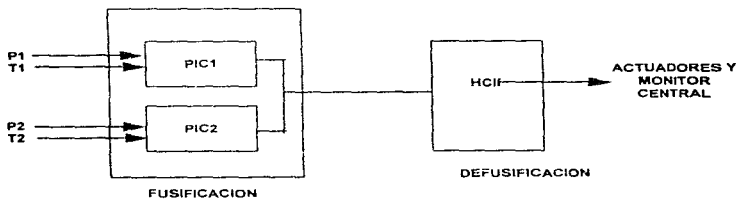


Fig. 7.1 ESQUEMA DE LA ARQUITECTURA DE CONTROL

### 7.3.- DIAGRAMA DE FLUJO.

En esta parte analizaremos por medio de un diagrama de flujo como esta formado nuestro algoritmo de control, explicando en forma general los pasos realizados. La primera etapa es la Fusificación que la realizan los PIC'S y el algoritmo sigue los siguientes pasos (ver figura 7.2):

- 1.- Toma los datos del convertidor analógico digital (PIC1:T<sub>1</sub>,P<sub>1</sub>; PIC2:T<sub>2</sub>,P<sub>2</sub>).
- 2.- Determina la o las regiones a las que pertenece el dato de presión.
- 3.- Inmediatamente proporciona el grado de pertenencia correspondiente al valor de presión.
- 4.- Transfiere el dato de presión, región (R), grado (G).
- 5.- Determina si existe región de traslape.
- 6.- Si existe esta región de traslape, inmediatamente proporciona el grado (GR).
- 7.- Transfiere al siguiente procesador la región de traslape (RT) y el grado (GR).
- 8.- En caso de que no exista región de traslape, entonces transfiere GR=64 y RT=00 ó 01.
- 9.- Completando estos cinco datos transferidos, realiza el mismo procedimiento para el dato de temperatura.
- 10.- Al finalizar los diez datos de la primera muestra vuelve a empezar con la segunda muestra.

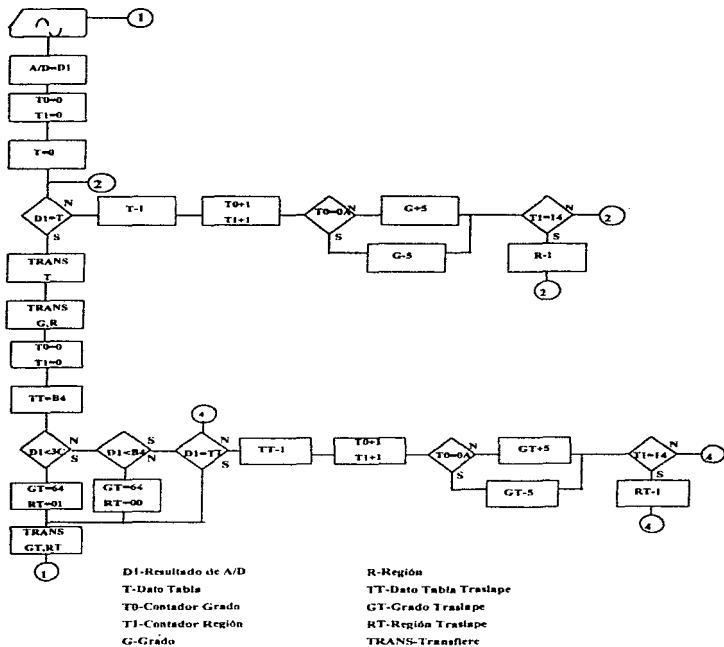
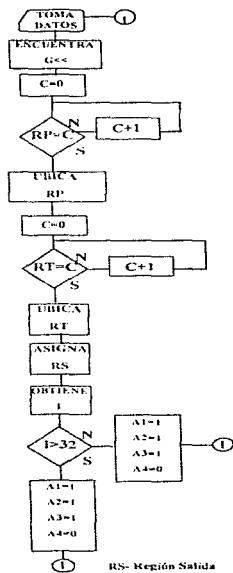


Figura 7.2 DIAGRAMA DE FLUJO PRIMERA PARTE (FUSIFICACION)

La segunda etapa es la Defusificación, que el 68HC11A8 en modo expandido realiza el siguiente algoritmo (ver figura 7.3):

- 1.- Recibe los diez datos que envió el PIC1, y los almacena.
- 2.- Determina por medio del método de los mínimos el menor de los grados de pertenencia para cada una de las regiones a las que pertenece el dato.
- 3.- Ubica regiones de presión y temperatura.
- 4.- Realiza la evaluación de reglas.
- 5.- Determina de acuerdo a la evaluación de reglas la ó las regiones de salida que corresponden a la entrada.
- 6.- Aplicando el método del centroide termina el proceso de Defusificación.
- 7.- Realiza la evaluación de los actuadores(analiza si están abiertos o cerrados).
- 8.- Una vez realizado, el proceso para los dos pares de datos ( $P_1, T_1$ ), inicia nuevamente el proceso para la siguiente muestra ( $P_2, T_2$ ).



C-Contador  
 RP-Región Presión  
 G-Grado  
 RT-Región Temperatura

RS- Región Salida  
 I- Centroide  
 AN- Actuadores

Figura 7.3 DIAGRAMA DE FLUJO SEGUNDA PARTE (DEFUSIFICACION)

Además de los pasos anteriores, que son parte del programa principal (algoritmo de control), ésta arquitectura mantiene comunicación con la PC (monitor central), para enviarle los datos que le solicite (valor de variables, estado de los actuadores), además recibe los comandos que desactivan ó activan al controlador. Todo esto se encuentra dentro de una rutina de servicio de interrupción que es activada desde la PC. Esta parte se compone de los siguientes pasos [28]:

1.- Pregunta por el código que le envían para saber que operación realizar sea:

- a) Enviar datos ( $T_1, P_1, T_2$  y  $P_2$ ).
- b) Activar ó desactivar el controlador.
- c) Manipular actuadores.

2.-Una vez atendida la rutina de interrupción el procesador se saldrá al principio del programa para reiniciar el proceso.

#### **7.4.- MANEJO DEL PROCESO DESDE EL MONITOR CENTRAL.**

Nuestro sistema forma parte de un sistema de control distribuido [9], monitoreando desde un monitor central (PC) [8], que cuenta con un sistema operativo en tiempo real QNX que nos permite estar sensando las variables de entrada de los diferentes controladores que este monitoreando. El monitor central está interconectado con nuestro



controlador mediante el puerto serie, y es capaz de realizar las siguientes actividades:

**a) Monitorear los valores que tienen las variables de entrada:** Esto se realiza mediante el uso de códigos que el monitor central le envía al controlador cada vez que el primero desea actualizar la información que tiene. El controlador interrumpe su proceso para atender la petición del monitor central. Al identificar que comando es el que le fue enviado, este contesta con la información solicitada por la PC terminando así la comunicación con el monitor central (PC), por el momento.

**b) Control de actuadores desde el monitor central:** Esta acción consiste en enviar desde el monitor central (PC), el código que le permita tomar el control del sistema. En el instante en que el controlador identifica este código interrumpe su proceso, permitiendo que la (PC), tome el control, permitiendo la manipulación de los actuadores de salida del sistema, quedando en ese momento el controlador simplemente como una interface de recepción y envío de los valores que en ese instante tengan las variables de entrada al sistema.

La PC en este momento puede controlar por completo el sistema, ya que cuenta con comandos que le permiten activar o desactivar los actuadores de salida sin que el algoritmo de control tenga que ser activado, remplazando con esto al controlador e introduciendo la experiencia de un experto que es quien maneja el monitor central. Si

desea devolver el control al controlador, también existe un código con el que podemos lograr, activando el algoritmo de control para retomar el control del sistema, a partir de las condiciones que nos haya dejado el experto. Los códigos que manejamos para que el monitor central pueda pedir información y/o tomar el control del sistema son los siguientes:

CODIGO DEL MENSAJE	DESCRIPCIÓN DEL MENSAJE
05	Pide información
07	Desactiva algoritmo de control
11	Activa algoritmo de control
38	Activa actuador 1
39	Activa actuador 2
40	Activa actuador 3
41	Activa actuador 4
50	Desactiva actuador 1
51	Desactiva actuador 2
52	Desactiva actuador 3
53	Desactiva actuador 4

**TABLA 7.1 CÓDIGOS UTILIZADOS PARA EL CONTROL DEL SISTEMA [28]**

## 7.5.- DIAGRAMA DE BLOQUES

En esta parte mostramos a manera de bloques todo el hardware de nuestro sistema:

- 1.- El PIC1 y el PIC2 a partir de los datos de presión y temperatura realiza la Fusificación entregando los datos en su puerto "D".
- 2.-El PIC1 avisa cuando un dato está listo para transmitir por medio de su terminal PA4 al HCII en su terminal PA7.
- 3.- El HCII da prioridad al PIC1 y habilita el primer buffer para recibir datos del PIC1 y avisa cuando recibe y almacena por medio de su terminal PA6 y un arreglo de compuertas al PIC1.
- 4.- Los dos primeros pasos se repiten hasta que el HC11 recibe los diez datos proporcionados por el PIC1.
- 5.- El HC11 realiza la Defusificación, evalúa actuadores.
- 6.- Inmediatamente después habilita el segundo buffer para recibir datos del PIC2.
- 7.- El PIC2 avisa cuando un dato está listo par transmitir por medio de su terminal PA4 al HC11 en su terminal PA7.

8.- El HC11 avisa cuando recibe y almacena por medio de su terminal PA6 y un arreglo de compuertas al PIC2.

9.- Estos dos últimos pasos se repiten hasta que el HC11 recibe diez datos proporcionados por el PIC2.

10.- Inmediatamente el HC11 realiza la defusificación y evaluación de actuadores de acuerdo a los datos proporcionados por el PIC2.

11.- Una vez terminados estos pasos vuelve al inicio.

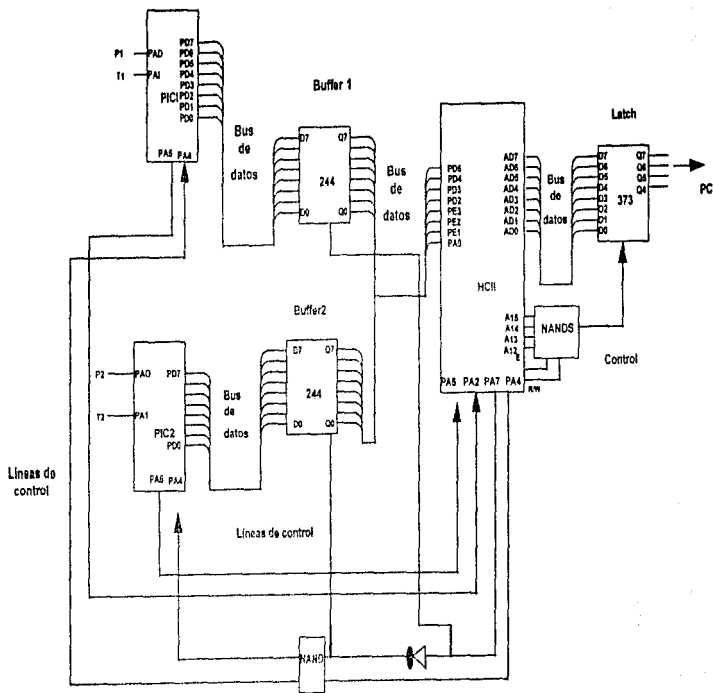


Figura 7.4 DIAGRAMA DE BLOQUES, MOSTRANDO LAS TERMINALES DE CONTROL

## 8.- PRUEBAS, RESULTADOS Y CONCLUSIONES.

### 8.1.- INTRODUCCIÓN

Una vez detallado el algoritmo del controlador difuso y descrito el hardware del sistema, se mencionará en que consistieron las pruebas realizadas; así como los resultados obtenidos en éstas.

### 8.2.- PRUEBAS

Para la fusificación, los valores de Presión y Temperatura (entradas numéricas) fueron convertidos a un código hexadecimal. De estos valores se formaron conjuntos difusos asignandoles su respectivo grado de pertenencia. Se formularon reglas de inferencia, de acuerdo a lo que nosotros estimamos será(n) la(s) posible(s) salida(s) difusa(s). Por lo que respecta a la defusificación se puede obtener por varios métodos, el método utilizado en éste sistema es el de "centroide" que está dado por la suma de los centros de gravedad de cada área; en términos generales:

$$I = \frac{\sum (U_n \times \mu_n)}{\mu_n} \quad \text{e.c. 8.1}$$

donde:

- I.- centroide
- $\mu$  .- grado de pertenencia
- I.- centroide
- U .- centro de gravedad de cada área.
- n .- número de las diferentes salidas difusas

Como mencionamos anteriormente nuestro sistema está formado por dos partes principales: fusificación y defusificación. La fusificación es realizada por los PIC'S, en donde el PIC1 atiende los datos de entrada del primer tanque y el PIC2 los del segundo tanque. La defusificación es realizada por el HC11, que recibe los datos fusificados de los dos PIC'S. La siguiente tabla muestra las operaciones que realizan los microcontroladores empleados.

PIC16C74 (1)	PIC16C74 (2)	68HC11A8
CONVERSIÓN A/D P1 y T1	CONVERSIÓN A/D P2 y T2	RECIBE DATOS DEL PIC1
ASIGNA SUBCONJUNTO DIFUSO	ASIGNA SUBCONJUNTO DIFUSO	DEFUSIFICA
ASIGNA GRADO DE PERTENENCIA	ASIGNA GRADO DE PERTENENCIA	EVALÚA ACTUADORES  RECIBE DATOS DEL PIC2
TRANSFIERE DATOS AL 68HC11A8	TRANSFIERE DATOS AL 68HC11A8	DEFUSIFICA EVALÚA ACTUADORES

**Tabla 8.1 FUNCIONES DE LOS MICROCONTROLADORES**

En la fusificación se tomaron en el puerto de entrada los valores de presión y temperatura en un código hexadecimal de un byte, en cada uno de los microcontroladores (PIC16C74) y a partir de ellos se obtuvieron los siguientes datos en el puerto de salida.

PRESIÓN O TEMPERATURA	GRADO	REGIÓN	GRADO TRASLAPE	REGIÓN TRASLAPE
7E	1E	03	46	12
70	28	04	36	12
B8	50	02	64	00
20	28	06	64	01
40	50	65	14	13
26	0A	06	64	01

**Tabla 8.2 DATOS FUSIFICADOS** (datos en código hexadecimal)

Estos valores fusificados son transferidos al 68HC11A8, por lo que este recibe el dato de presión con sus respectivos datos de grado y región, y el dato de temperatura con sus datos respectivos de grado y región, siendo diez datos transferidos por el PIC1. Realiza la defusificación y posteriormente continúa con el activado o desactivado de los actuadores, entregando una salida difusa y el resultado de la evaluación de los actuadores, para después recibir los siguientes diez datos del PIC2 y realizar la misma operación. Algunos datos obtenidos son los siguientes:



PRESIÓN O TEMPERATURA	SALIDA DIFUSA	ESTADO DE ACTUADORES	SALIDA DIFUSA (TEÓRICA)
70	2E	E0	2E
40	18	E0	1P
20	08	E0	09
C8	5A	10	52
BB	55	10	5A

Tabla 8.3 DATOS DEFUSICADOS (datos en código hexadecimal)

Comparando los datos de la salida difusa con los datos teóricos, de acuerdo a los datos de entrada, observamos que en cuanto nos acercamos a las regiones del centro, es decir las regiones donde existe los traslapes, hay mayor exactitud. Debemos aclarar que nuestro nivel de referencia se encuentra en subset "CENTRO" (32) de nuestros conjuntos difusos de salida (H).

### 8.3.- RESULTADOS.

Una vez presentada las pruebas realizadas, mencionaremos los tiempos de respuesta en cada una de las etapas.

PIC1		PIC2		68HC11A8	
RUTINA	TIEMPO (us)	RUTINA	TIEMPO (us)	RUTINA	TIEMPO (us)
CONFIGURACIÓN DE REGISTROS	12.5	CONFIGURACIÓN DE REGISTROS	12.5	CONFIGURACIÓN DE REGISTROS	36.5
CONVERSIÓN A/D P1 y T2	12.5	CONVERSIÓN A/D P2 y T2	12.5	ADQUISICIÓN DE DATOS PIC1	217
FUSIFICACIÓN	100	FUSIFICACIÓN	100	DEFUSIFICA	555
TRANSFIERE DATOS	82	ESPERA A SER ATENDIDO	963	EVALÚA ACTUADORES	191
ESPERA A SER ATENDIDO	963	TRANSFIERE DATOS	82	ADQUISICIÓN DE DATOS PIC2	217
<b>TOTAL</b>	<b>1170</b>		<b>1170</b>		<b>1216.5</b>
INTERRUPCIÓN EXTERNA	23	INTERRUPCIÓN EXTERNA	23	ACCIONES DE CONTROL	625

Tabla 8.4 TIEMPOS DE RESPUESTA

El 68HC11A8 recibe datos del PIC1, defusifica y evalúa actuadores; posteriormente toma datos del PIC2 defusifica, evalúa actuadores para volver a recibir datos del PIC1. Teniendo en cuenta los resultados anteriores, se obtiene una muestra en 2.3 ms.

#### **8.4.- CONCLUSIONES.**

El algoritmo difuso utilizado no se basa en un modelo matemático del proceso a controlar; de tal forma que se puede ahorrar tiempo de cálculo. Por lo que los algoritmos difusos son ideales para aplicaciones en tiempo real ó donde se necesita rapidez y predicibilidad. Aunque una desventaja sería tener que conocer el comportamiento del proceso y hacer pruebas, cambiando reglas de inferencia hasta que el funcionamiento de nuestro controlador sea adecuado al comportamiento del proceso. Esto lleva a tener un gran número de reglas y por ende puede llevar un tiempo de procesamiento grande. Por lo que al usar un algoritmo difuso tendrá que considerarse los siguientes puntos:

- 1.- Conocimiento del proceso a controlar.
- 2.- Evaluar el número de reglas de inferencia.
- 3.- Tiempo de ejecución.
- 4.- Precisión en la respuesta del controlador.
- 5.- Estabilidad dentro de los rangos determinados.
- 6.- Arquitecturas para este tipo de algoritmo, así como el software necesario para su aplicación.

Con respecto a este punto diremos que la arquitectura va de acuerdo a las características de dichos algoritmos difusos. En nuestro caso se utiliza una arquitectura en paralelo de bus compartido, es decir, mientras se identifican las variables de entrada en cierto instante, se están fusificando las variables de la muestra anterior, además comparten el mismo bus para transmitir. Ésta arquitectura utilizada, optimiza el desarrollo del sistema ya que ahorra tiempo de

cálculo incrementando la velocidad del proceso. Para cumplir con este tipo de arquitectura se dispuso de dos tipos de microcontroladores que utilizan diferente tecnología, esto es, el PIC16C74 usa un diseño R.I.S.C. y el 68HC11A8 un diseño C.I.S.C. haciendo un sistema versátil.

En nuestro sistema, la simulación de transductores fué hecha con potenciómetros, haciendo variar un nivel de voltaje simulaba una variación de presión o de temperatura; en cuanto a los actuadores se visualizó con leds el prendido y apagado. En la práctica se recomienda utilizar electroválvulas, que de acuerdo a una señal eléctrica que corresponde según el dato de salida, la electroválvula permite u obstruye el paso del líquido.

Hemos planteado un sistema que cumple con la arquitectura propuesta e implementado con microcontroladores de acuerdo a ella, obteniendo rapidez, optimización en el desarrollo del sistema, permitiendome tener una visión acerca del diseño e implemetación de cualquier sistema de control . Todos estos sistemas de control han dado lugar a sistemas de seguridad automáticos que se pueden monitorear mediante un comando central.

**9.- BIBLIOGRAFIA.**

- [1] APPLICATION OF FUZZY ALGORITHMS FOR CONTROL OF A SIMPLE DYNAMIC PLANT. E.H. Mamdami. Proc. IEEE, Vol 2, 1974.
- [2] A FUZZY LOGIC CONTROLLER FOR A DYNAMIC PLANT. E.H. Mamdami and S. Assilian. In the J. Man-Machine Stud. Vol. 7 1975.
- [3] THE APPLICATION OF FUZZY SET THEORY TO CONTROL A WARM WATER PROCESS. H.R. Van Nauta Lemke and W.J.M. Kickert. Journal A17, 1976.
- [4] APPLICATION OF A FUZZY CONTROLLER IN A WARM WATER PLANT W.J.M. Kickert and H.R. Van Nauta Lemke. Automática, Vol. 12, 1976.
- [5] FURTHER ANALYSIS AND APPLICATION OF FUZZY LOGIC CONTROL Internal Report F/WK/2/75, Queen Mary's College London, 1975.
- [6] THE APPLICATION OF FUZZY CONTROL SYSTEMS TO INDUSTRIAL PROCESS. P.J. King and E. H. Mamdami. Automática, Vol. 13, 1976.

- [7] A HEURISTIC ADAPTIVE CONTROLLER FOR A SINTER PLANT. D. Rutherford and G. Carter Proc. 2nd IFAC Symp. Johannesburg, 1976
- [8] RISC ARCHITECTURES, J.C. Heudin and C. Panetto. Chapman & Hall, 1a edición, 1992.
- [9] ARQUITECTURA DE COMPUTADORAS Y PROCESAMIENTO PARALELO, Kai Hwang/Faye A. Briggs, Mc Graw-Hill, 1a edición, 1988.
- [10] THE 8051 MICROCONTROLLER, I. Scott Mackenzie Merrill, New York Ny 1992.
- [11] MICROCONTROLLER TECHNOLOGY, Peter Spasov. Prentice Hall, Englewood Cliffs, Nueva Jersey 1993.
- [12] M68HC11 REFERENCE MANUAL, Motorola Inc. 1991.
- [13] FUZZY SYSTEMS THEORY AND ITS APPLICATIONS, Toshiro Terano, Kiyoji Asai, Michio Sugeno, Academic Press, Inc, San Diego California, 1987.
- [14] FUZZY SETS AND APPLICATIONS: Selected Papers by L.A. Zadeh, Edited by R.R. Yager, S. Ovchinnikov, R.M. Tong, H.T. Nguyen, John Wiley y Sons, U.S.A. 1987.

- [15] FUZZY LOGIC AND CONTROL, Mohammad Jamshidi, Nader Vadiee, Timothy, J. Ross, Prentice-Hall, Inc. Englewood Cliffs N.J. U.S.A.1993.
- [16] NEURAL NETWORKS AND FUZZY SYSTEMS, Bart Kosko, PrenticeHall, Englewood Cliffs, New Jersey. 1992.
- [17] FUZZY SYSTEMS AND OVERVIEW, Toshinori Munakata, Yashuant Jani, Communications of the ACM, 1994.
- [18] STABILITY ANALYSIS AND DESIGN OF FUZZY CONTROL SYSTEMS, Kazuo Tanaka and Michio Sugeno, Elsevier Science Publishers, 1992.
- [19] STABILITY A PPROACH TO FUZZY C ONTROL DESIGN FOR NONLINEAR SYSTEMS, Guang-Chyan Hwan and Shih-Chang Lin, Elseiver Science Publishers 1990.
- [20] MPSIM USER'S GUIDE, Microchip Technology Inc, 1993.
- [21] MPASM USER'S GUIDE, Microchip Technology Inc, 1993.
- [22] MICROCHIP DATA BOOK, Microchip Technology Inc. 1993.
- [23] THE TTL DATA BOOK FOR DESIGN ENGINERRS, Texas Instruments Incorporated, U.S.A. 1981.

[24] TRANSDUCERS THEORY & APPLICATIONS. Jhon A. Alloca, Allen Stuart. A Prentice-Hall Company. 1984

[25] LOGICA DIGITAL Y DISEÑO DE COMPUTADORES. M. Morris Mano. De. Prentice-Hall, 1a edición, 1979.

[26] INSTRUMENTACION ELECTRONICA Y MEDICIONES. William David Cooper. De. Prentice-Hall, 2a edición, 1988.

[27] COMPAQ TECHNICAL MANUAL, Compaq Inc. 1995

[28] DISEÑO Y DESARROLLO DE UN SISTEMA PARA EL CONTROL DISTRIBUIDO DE PROCESOS USANDO UN SISTEMA OPERATIVO EN TIEMPO REAL. Tesis de Licenciatura, Facultad de Ingeniería, UNAM, López Rodríguez Alejandra, México, D.F. 1995.

[29] DISEÑO DE UN SISTEMA DE MONITOREO Y CONTROL DISTRIBUIDO EN TIEMPO REAL ,Tesis de maestría, Facultad de Ingeniería, UNAM, Avarez Castillo, Jesús México.D.F,1997



## ANEXO A

### INTRODUCCIÓN

En esta parte mostramos las características eléctricas de los microcontroladores utilizados.

( $V_{DD} = 5V_{DC} \pm 10\%$ ,  $V_{SS} = 0V_{DC}$ ,  $T_A = T_C$  a  $T_M$ )

Características		Símbolo	Mínimo	Máximo	Unidad
Voltaje de salida $I_{LOAD} = \pm 10 \mu A$ excepto RESET y MODA	todas las salidas todas las salidas	$V_{OH}$ $V_{OL}$	----- $V_{DD} - 0.1$	$0.1$ -----	volts volts
Alto voltaje de salida $I_{OEH} = -0.8 mA$ , $V_{DD} = 4.5$ XTAL Y MODA	Todas las salidas excepto RESET	$V_{OH}$	$V_{DD} - 0.8$	-----	Volts
Voltaje bajo de salida $I_{OEL} = 1.6 mA$	Todas las salidas excepto XTAL	$V_{OL}$	-----	0.4	Volts
Voltaje alto de entrada excepto RESET	Todas las entradas RESET	$V_{IH}$	$0.7 \times V_{DD}$ $0.8 \times V_{DD}$	$V_{DD}$ $V_{DD}$	Volts
Voltaje bajo de entrada	Todas las entradas	$V_{IL}$	$V_{OL}$	$0.2 \times V_{DD}$	Volts
Puertos E/S 3 estados	PA3, PA7, PC0-PC7, PD0, PD5, AS/STRA, MODA/LIR, RESET	$I_{OZ}$	-----	$\pm 10$	$\mu A$
Corriente de entrada $V_{in} = V_{OH}$ ó $V_{OL}$ $V_{in} = V_{DD}$ ó $V_{SS}$	PA0 - PA3, IRQ X IRQ MODB/STBY	$I_{in}$	-----	$\pm 1$ $\pm 10$	$\mu A$

Tabla A.1 CARACTERÍSTICAS ELÉCTRICAS DE D.C. DEL 68HC11AB

(V<sub>DD</sub> = 4.0V a 6.0V 0°C ≤ T<sub>A</sub> ≤ +70°C)

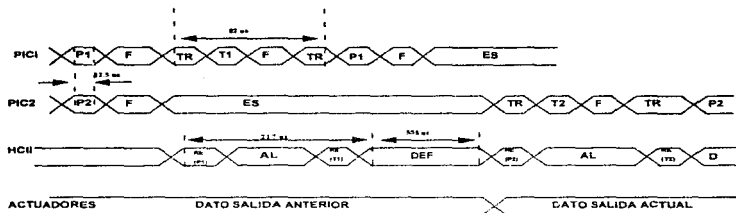
Características	Símbolo	Mínimo	Típico	Máximo	Unidad	Condiciones
Voltaje de entrada I/O puertos con TTL buffer con Schmitt Trigger buffer (MCLR, RA4/TOCKI, OSC1 (in RC modo) OSC1 (in XT, HS y LP))	V <sub>ih</sub>	V <sub>DD</sub> V <sub>DD</sub>	— —	0.8V 0.2 V <sub>DD</sub>	V V	
Voltaje de entrada Añta I/O puertos con TTL buffer con Schmitt Trigger buffer (MCLR, RA4/TOCKI, RC<7 4>, RD<7 4> INT RE<2:0>, OSC1 (XT, HS y LP))	V <sub>ih</sub>	2.0 0.85 V <sub>DD</sub> 0.85 V <sub>DD</sub> 0.7 V <sub>DD</sub>	— — — — — —	V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub> V <sub>DD</sub>	V V V V V V	4.5 ≤ V <sub>DD</sub> ≤ 5.5V para máximo V <sub>DD</sub>
Corriente weak pull-up Puerto B	I <sub>pullup</sub>	50	200	+400	µA	V <sub>DD</sub> = 5V, V <sub>DD</sub> = V <sub>DD</sub>
Corriente de entrada I/O puertos MCLR, RA4/TOCKI OSC1	I <sub>ih</sub>	— — —	— — —	±1 ±5 ±5	µA µA µA	V <sub>DD</sub> ≤ V <sub>DD</sub> ≤ V <sub>DD</sub> terminal a alta impedancia V <sub>DD</sub> ≤ V <sub>DD</sub> ≤ V <sub>DD</sub> V <sub>DD</sub> ≤ V <sub>DD</sub> ≤ V <sub>DD</sub> XT, HS y LP osc
Voltaje de salida bajo I/O puertos OSC2/CLKOUT (configuración RC osc)	V <sub>oh</sub>	— — — —	— — — —	0.6 0.5 0.6 0.6	V V V V	I <sub>OH</sub> = 8.5mA, V <sub>DD</sub> = 4.5V, -40°C a +85°C I <sub>OH</sub> = 7.0mA, V <sub>DD</sub> = 4.5V, -40°C a +125°C I <sub>OH</sub> = 1.6mA, V <sub>DD</sub> = 4.5V, -40°C a +85°C I <sub>OH</sub> = 8.5mA, V <sub>DD</sub> = 4.5V, 40°C a +125°C
Voltaje de salida aña I/O puertos OSC2/CLKOUT (configuración RC osc)	V <sub>oh</sub>	V <sub>DD</sub> -0.7 V <sub>DD</sub> -0.7 V <sub>DD</sub> -0.7 V <sub>DD</sub> -0.7	— — — —	— — — —	V V V V	I <sub>OH</sub> = 3.0mA, V <sub>DD</sub> = 4.5V, -40°C a +85°C I <sub>OH</sub> = 2.5mA, V <sub>DD</sub> = 4.5V, -40°C a +125°C I <sub>OH</sub> = 1.3mA, V <sub>DD</sub> = 4.5V, -40°C a +85°C I <sub>OH</sub> = 1.0mA, V <sub>DD</sub> = 4.5V, 40°C a +125°C
Carga capacitiva en las terminales de salida OSC2 terminal todas I/O terminal y OSC2 (en modo RG) SCL, SDA en modo I2C	C <sub>osc2</sub> C <sub>io</sub> C <sub>e</sub>			15 50 400	pF pF pF	En modo XT, HS y LP cuando el reloj externo es usado al manejar OSC1

Tabla A.2 CARACTERÍSTICAS ELÉCTRICAS DE D.C. DEL PIC16C74

## ANEXO B

### INTRODUCCIÓN

En esta parte mostramos los tiempos de operación del sistema.



P1 Conversión A/D de Presión 1

P2 Conversión A/D de Presión 2

T1 Conversión A/D de Temp. 1

T2 Conversión A/D de Temp. 2

F Fusificación

TR Transfere los primeros cinco datos

ES Espera ser atendido

RE Recibe datos

DEF Defusifica

AL Espera los cinco datos restantes

Figure B.1 SECUENCIA DE OPERACION DEL SISTEMA

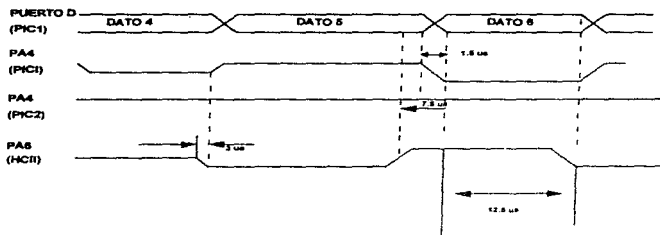


Figura 8.2 EL PIC1 TRANSMITE DATOS, MIENTRAS EL PIC2 ESPERA A SER ATENDIDO. AL ESTAR PRESENTE EL DATO EN EL PUERTO D DEL PIC1, AVISA (PA4) AL HC11. ESTE ALMACENARLO AVISA (PA5) AL PIC1