



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Control de temperatura con lógica difusa

TESIS

Que para obtener el título de

Ingeniero Eléctrico Electrónico

P R E S E N T A N

Juan Luis González Mancilla

Carlos Maya León

DIRECTOR DE TESIS

Ing. Roberto Macías Pérez



Ciudad Universitaria, Cd. Mx., 2003

CAPITULO 1

INTRODUCCION

En la época prehistórica, el fuego fue el elemento primordial que utilizó el ser humano para calentar su morada. La sombra y el agua fueron los elementos que utilizó para poder contrarrestar el exceso de calor. A lo largo de la historia se ha podido observar que el ser humano ha tratado de manipular las condiciones térmicas que prevalecen en un cierto medio, disipando calor o realizando el enfriamiento del medio que lo rodea. En la época medieval prevalecieron los grandes castillos con sus enormes chimeneas que sólo calentaban el área que las rodeaba. Los romanos eran propietarios de construcciones que contaban con una calefacción notablemente buena, calentando el aire y haciéndolo circular por paredes y pisos cóncavos. En los pueblos del Medio Oriente, en donde prevalece el clima seco, colgaban mantas húmedas frente a las puertas y, de esta manera, poder obtener una forma primitiva de enfriamiento por evaporación. Más tarde, con un poco más de desarrollo tecnológico y evolución de la ciencia, Leonardo da Vinci diseñó un gran enfriador por evaporación.

Sin embargo, el verdadero desarrollo de la calefacción y del sistema de aire acondicionado comenzó aproximadamente a fines del siglo pasado, aunque los sistemas de aire acondicionado, por medio de refrigeración mecánica, han tenido su desarrollo en los últimos cincuenta años.

En las últimas décadas se ha percibido un gran desarrollo en el ámbito eléctrico y electrónico, las aplicaciones se han incrementado y se ha llegado a campos que no se habían considerado, el campo de acondicionamiento de aire y calefacción no es la excepción.

Así, se han desarrollado sistemas de calefacción y aire acondicionado, en los cuales la mayoría del control es electrónico, se ha podido automatizar el sistema para acrecentar el confort del ser humano y, gracias a la nueva tecnología, se ha logrado reducir el tamaño de dichos sistemas en una proporción considerable.

Actualmente es muy común encontrar sistemas de calefacción y acondicionamiento de aire en grandes industrias, oficinas, casas habitación, centros de comercio, instalaciones deportivas y en la mayor parte de los transportes como son los automóviles, camiones, aviones, barcos, trenes, etc.

El proyecto que se presenta es un controlador de temperatura y su diseño adopta los principios del acondicionamiento de aire y calefacción. El sistema basa su estructura y control en la lógica difusa, herramienta que permite sustituir a sistemas clásicos “on/off” cuyo funcionamiento era controlado por la lógica binaria.

A lo largo de la historia, los principios de probabilidad han sido la principal herramienta para representar la incertidumbre en modelos matemáticos y por ello, toda la incertidumbre era asumida como una caracterización de incertidumbre aleatoria.

Sin embargo, la teoría de los conjuntos difusos permiten adoptar una forma de representar incertidumbre asociada con vaguedad, con imprecisión y con la carencia de información con respecto a un elemento particular. De igual forma, la teoría difusa da la posibilidad de utilizar variables lingüísticas en sustitución de variables cuantitativas que permitan representar conceptos imprecisos.

Obviamente, los principios de calefacción y acondicionamiento de aire no varían mucho con respecto a lo que aquí se menciona, pero si cabe mencionar que el control para acondicionar la temperatura a un estado confortable para el ser humano, si difiere un poco de los que generalmente se encuentran en diversos libros.

Las condiciones de temperatura, humedad, limpieza y movimiento del aire, para acondicionar algún sistema cerrado, son de enorme importancia, ya que su interacción con el comportamiento físico del cuerpo humano permite establecer si una persona se siente confortable o molesta con el ambiente que prevalece en un cierto medio.

La mayor parte de los sistemas de control ambiental son usados para dar mayor confort a las personas, pero también son usados para obtener las condiciones que se requieren en determinados procesos.

Se presenta un proyecto en el cual se tienen que considerar las condiciones ambientales a controlar para poder proporcionar un confort adecuado para el ser humano, con esto, se presentará un diseño apto para la aplicación deseada.

En el capítulo dos se analizará el sistema como una aplicación general, considerando al sistema una caja negra, esto ayudará a analizar los diversos principios del sistema de control ambiental: calefacción, ventilación y acondicionamiento de aire. Posteriormente se analizará las partes que constituyen a un sistema de control ambiental. Con esto se podrá tener una idea de las partes que serán de utilidad para conformar el sistema que tendrá la aplicación deseada. Este mismo capítulo permitirá analizar sistemas muy sencillos de control ambiental, así como diseños muy completos cuyo control electrónico es la base de su éxito. De la misma manera, se justificará el porqué utilizar lógica difusa para el control y no otro principio.

Al final del capítulo dos, se sugerirá un diseño con el cual se controlará la temperatura en el interior de un recinto utilizando los principios de lógica difusa y las partes necesarias para el desarrollo del sistema, realizando los pasos adecuados para cubrir las necesidades planteadas.

En el capítulo tres se realiza una descripción de lo que es la lógica difusa. La herramienta difusa es una alternativa para describir con precisión lo vago o no plenamente definido, permite adoptar un concepto ambiguo y establecer el grado de pertenencia de un elemento a un conjunto; en cambio lo aleatorio describe la incertidumbre en la ocurrencia de un evento.

Se exponen sus orígenes y principios, su teoría, la comparación con la lógica común o que todos conocen (lógica Aristotélica, binaria o “crisp”), las aplicaciones en la que es

partícipe, las herramientas que hasta la fecha se tienen para poder trabajar con ella y las ventajas y desventajas que se presentan al desarrollar un sistema difuso.

El capítulo cuatro hace referencia a toda la parte física del sistema (Hardware). La parte física esta dividida en partes: la parte de entrada o de sensado, la parte de control, en donde se aplica la lógica difusa y la parte de salida o de potencia. Toda la electrónica del sistema es expuesta en este capítulo, se hace una descripción de los componentes que conforman cada etapa, de la comunicación entre ellos mismos y de la comunicación entre una etapa y otra.

Todo lo referente a la programación del sistema (Software) es expuesto en el capítulo cinco. Se hace mención a la parte microcontroladora, a las herramientas ocupadas para la realización de programas, diagramas de flujo y listados de cada operación realizada por el sistema. Se mostrará la capacidad del programa principal y se hará mención a las diversas contrariedades que se puedan presentar debido a posibles modificaciones en el programa principal, y de esta manera poder recomendar soluciones para un mejor funcionamiento en el control.

Finalmente se tendrán las conclusiones en las cuales se expondrán los puntos de vista finales y definitivos acerca del comportamiento del sistema. Dentro de este capítulo se hará mención a las diversas impresiones que se experimentaron en el momento en que se realizaron pruebas al sistema, si se detectaron errores se mencionarán y se expondrán las soluciones que se tomaron.

CAPITULO 2

PLANTEAMIENTO DEL PROYECTO

2.1 INTRODUCCION.

En muchas ocasiones el ser humano ha tenido que ingeniárselas para poder luchar contra las situaciones adversas que presenta el clima en un cierto ambiente o en cierta región, por ejemplo, el exceso de calor, el frío o las heladas, la lluvia o la humedad, etc.. Así, existe una tendencia a crear espacios ambientales con características favorables que puedan cumplir con las exigencias del cuerpo humano.

Sin embargo, para poder considerar los diversos parámetros que intervienen en el control y acondicionamiento de un ambiente es conveniente abundar en ciertos conceptos y comportamientos, tanto del medio ambiente como del cuerpo humano y de esta manera, concluir en adecuadas herramientas que permitan plantear diversas soluciones para la obtención de condiciones ambientales interiores que proporcionen un confort adecuado al ser humano.

Las soluciones para poder adaptar el clima de un medio a las necesidades del ser humano pueden ser diversas, muy rudimentarias o sistemas altamente sofisticados; en realidad, los sistemas pueden ser pequeños o enormes, aplicaciones domésticas o en vehículos, así como aplicaciones industriales, pero lo importante es que el ser humano utiliza lo que tiene (ciencia y tecnología) para obtener bienestar propio, adaptando el medio que lo rodea a sus necesidades.

2.2 MOTIVACIONES DEL PROYECTO

Se sabe que la temperatura de confort para el ser humano está entre los 21° y 27° centígrados, pero esto depende de la humedad que prevalezca en el sitio donde se desea mantener este rango.

A lo largo de la historia, uno de los principales objetivos del ser humano ha sido mantener, dentro de un grado aceptable de perfección y estabilidad, los principales estándares que permiten la comodidad de los compartimientos que el ser humano utiliza para su estancia.

Ciertamente, para que un medio sea confortable, es necesario que mantenga un cierto grado de temperatura compatible con el nivel de temperatura al cual el ser humano se siente a gusto. En este punto es donde surge la dificultad: ¿Cómo saber a que nivel de temperatura se siente a gusto una persona?, ¿todas las personas se sienten a gusto a un mismo nivel de temperatura?. Es ilógico pensar que todas las personas sienten comodidad en un mismo grado de temperatura, sin embargo, la tecnología tiene que permitir desarrollar técnicas para dicho fin y que se adapten a las exigencias de cada persona.

De esta manera, dentro de las necesidades que se podrían establecer para el diseño de un compartimiento óptimo, la parte correspondiente al control de temperatura ha originado una severa importancia, no solamente por el confort, sabemos que el funcionamiento de diversos equipos eléctricos y electrónicos exigen, para su buen funcionamiento, un sistema el cual mantenga rangos de temperatura establecidos por los propios fabricantes.

Para poder entender la necesidad de desarrollar un control de temperatura, sería conveniente y necesario mencionar el comportamiento térmico del cuerpo humano. Se sabe que el cuerpo del ser humano es una máquina de calor. Al cuerpo se le introduce combustible llamado alimento y éste lo transforma en energía. Dicha energía es la utilizada en cada actividad de la vida cotidiana.

El alimento que se ingiere diariamente sirve para que el cuerpo conserve una temperatura interna, así como en la profundidad de los tejidos de 37° C. Se sabe que la temperatura a la que permanece la piel es cercana a los 32° C. La temperatura ambiente promedio a la que el ser humano se siente cómodo es de 24° C. Se sabe también que el cuerpo irradia calor en forma continua. Como el calor se desplaza siempre de los sitios calientes a los fríos, existe un desplazamiento de calor debido a esa diferencia que hay entre la temperatura interior del cuerpo (37° C) y la temperatura ambiente promedio (24° C).

El cuerpo tiene una forma de asegurar que se irradiará hacia el aire la cantidad apropiada de calor. Si por alguna razón el cuerpo no puede liberar suficiente calor, se percibe calor excesivo o se sufre de postración a causa del calor. En caso contrario, si se pierde demasiado calor se presenta la hipotermia. De esta manera, para poder controlar esta disipación de calor, el cuerpo perspira o suda.

El cuerpo suda en todo momento. Lo normal es que a medida que aparece humedad sobre la piel, de inmediato se evapora hacia el aire. Este mecanismo funciona bien en tanto el aire que rodea al cuerpo humano está seco. Si el aire ambiente contiene demasiada humedad no absorberá la perspiración del cuerpo humano y de inmediato surgirá la incomodidad. Se percibirá calor y sudor porque el cuerpo pierde su mecanismo normal de disipar calor.

Por otro lado, resulta de la misma importancia bajar la temperatura ambiente como disminuir la humedad del aire. En una habitación caliente surgirá la comodidad siempre y cuando, el aire se mantenga seco.

De esta manera, se considera de vital importancia, para obtener la comodidad en el interior de un compartimiento, el control de la temperatura, considerando los parámetros del aire que se relacionan con ella: humedad y movimiento.

2.3 PROBLEMA

El calor puede describirse como una forma de energía en transferencia. El calor es la forma de energía que se transmite de un cuerpo a otro debido a una diferencia de temperatura. Hay que hacer notar que el calor solamente puede pasar, en forma

natural, desde una temperatura mayor a una menor. Desde luego, si no existe una diferencia de temperaturas, no presentará flujo de calor. Recordemos que la temperatura es una medida de la actividad térmica de un cuerpo, la cual depende de la velocidad de las moléculas y demás partículas, de las cuales se compone cualquier cuerpo.

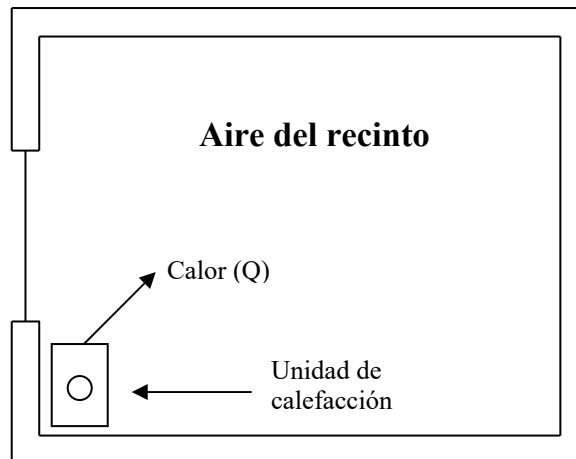


Fig. 2.1 Flujo del calor dentro del compartimiento

Se sabe que el calor es una manifestación de energía y por lo tanto cumple con la primera ley de la termodinámica o la ecuación de la energía: “La energía no se crea ni se destruye”. De esta forma, se dice que en un sistema, el cambio de energía total es igual a la energía agregada al sistema menos la energía eliminada del mismo. Esto puede expresarse de la siguiente forma:

$$E_{\Delta} = E_i - E_o \dots (2.a)$$

donde:

E_{Δ} = Variación de energía almacenada en el sistema.

E_i = Energía de entrada al sistema.

E_o = Energía de salida del sistema.

De esta forma, se concluye que el calor no puede destruirse, ya que cumple con la ley de la energía, pero sí se puede controlar y desplazar. Un acondicionador de aire, por ejemplo, ayuda a retirar exceso de calor en un determinado compartimiento y puede disiparlo al exterior.

Para poder adoptar el planteamiento de este problema, donde se desea controlar el calor de un compartimiento, buscando el óptimo confort para el ser humano, es necesario abundar en ciertos temas que serán base primordial en la solución que se tome para la realización del proyecto que se expone.

A lo largo de los años, se ha mencionado que la energía puede clasificarse en energía en transferencia y energía almacenada. Dos formas de energía almacenada son la energía cinética y la energía potencial. La primera es la energía que almacena un cuerpo debido a su movimiento o velocidad, mientras que la segunda es la energía que

almacena un cuerpo debido a su posición o elevación. Pero se sabe que los cuerpos también almacenan energía debido a su temperatura y presión. Por ejemplo, un gas a alta presión tiene energía almacenada, el agua a alta temperatura mantiene almacenada energía calorífica.

De tal manera que, se le llama entalpía a la energía almacenada en forma de temperatura y presión. Teniendo conocimiento de lo que es entalpía, podemos hacer mención del calor sensible y calor latente.

Cuando se agrega o elimina calor a una sustancia y esto provoca un cambio de temperatura, se conoce a el cambio de entalpía en la sustancia como cambio de calor sensible. Sin embargo, cuando este calor agregado o eliminado a la sustancia provoca un cambio de estado, entonces se conoce a el cambio de entalpía en la sustancia como cambio de calor latente.

Auxiliándose con la ecuación 2.a y aplicándose a una sustancia cualquiera, se puede determinar la relación entre la cantidad de calor que se agrega o que se elimina y la variación de calor sensible. De esta manera tenemos: variación de calor sensible o entalpía igual al calor neto agregado o eliminado, es decir:

$$Q = mC(T_1 - T_2) \dots (2.b)$$

y si se involucra al tiempo:

$$Q' = mCT' \dots (2.c)$$

donde:

Q' = Variación de calor con respecto al tiempo.

m = Masa de la sustancia.

C = Calor específico de la sustancia.

T' = Variación de la temperatura con respecto al tiempo.

El principal objetivo que plantea el proyecto es poder controlar la temperatura interna de un compartimiento, controlando el calor absorbido y transferido, dependiendo del comportamiento de la temperatura ambiente.

La absorción de calor de un compartimiento se puede presentar de diferentes formas. La energía calorífica que emiten los rayos solares puede ser absorbida por medio de la superficie del mismo recinto, así como los accesorios que puedan formar parte de su estructura.

Otra fuente de energía calorífica que se presenta es la irradiada por el mismo cuerpo humano.

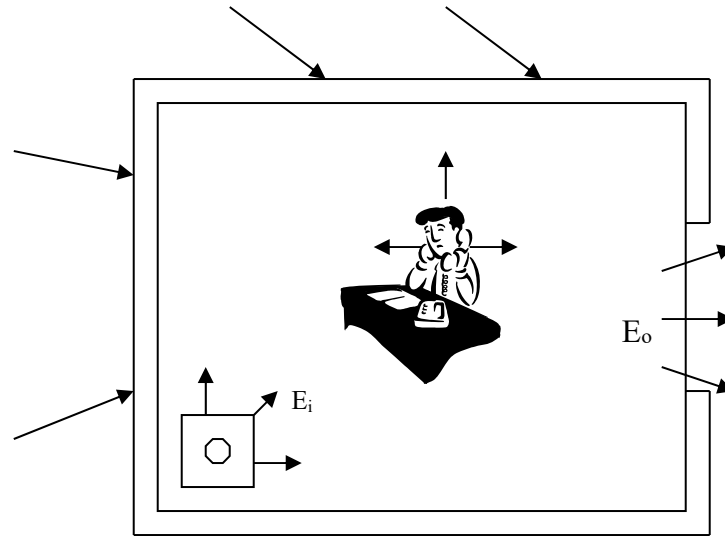


Fig. 2.2 Diversas formas de transferencia de calor.

Entre las alternativas para reducir la absorción de calor, podemos mencionar, en primer término, el tipo de color de la estructura del compartimiento. Los colores claros reflejan más y absorben en menor grado la energía solar, de esta manera, se obtendrá un grado menor de temperatura en el interior. Los colores oscuros absorben más y reflejan menos, por lo tanto, se mantendrá un mayor grado de temperatura en el interior.

El calor del sol irradiado al interior del compartimiento tiende a aumentar debido a que accesorios, como los vidrios, lo pueden reflejar hacia el interior, evitando la disipación al exterior, a lo que se le llama efecto de invernadero. Pero si se tiene un vidrio de color (polarizado), éste refleja cantidades considerables de energía solar.

Así, podemos encontrar diversas formas para poder disminuir la absorción de energía calorífica al interior del compartimiento, pero la realidad es que no siempre tendrán éxito, ya que tiene que considerarse un parámetro de gran importancia que envuelve al problema que se está exponiendo: la temperatura ambiente. Sin embargo, en este proyecto se planea controlar la temperatura de un sistema, controlando el flujo de calor absorbido, obviamente considerando la temperatura exterior al compartimiento y considerando la transferencia de calor que se pueda presentar.

Por otro lado, la disminución de temperatura, en un compartimiento, se debe, principalmente a la transferencia de calor que existe desde el aire caliente que pueda existir en el interior del compartimiento, hasta el aire frío del exterior; esta transferencia de calor se presenta a través de las paredes, ventanas y demás partes del compartimiento. La infiltración es otra forma de perder calor debido a las fugas de aire frío que se presentan.

Para poder contrarrestar las pérdidas de calor que se presentan, lo que produce una disminución de temperatura, es necesario agregar continuamente energía al interior del compartimiento, y de esta manera mantener la temperatura deseada.

La cantidad de energía calorífica que se debe suministrar, con el objetivo de mantener el aire del compartimento a una temperatura deseada, se llama carga de calefacción, la cuál es consecuencia de pérdida de calor por transferencia de calor y pérdidas por infiltración o ventilación.

La transferencia de calor puede efectuarse de tres modos distintos: radiación, conducción y convección.

La transferencia de calor por radiación se presenta entre dos cuerpos separados como resultado de la llamada radiación electromagnética, conocida también como movimiento ondulatorio.

La transferencia de calor por conducción es una forma de transferencia que se presenta a través de un cuerpo que no manifiesta movimiento alguno, puede decirse que es el resultado de acciones moleculares o electrónicas. Esta forma de transferencia es muy común a través de los sólidos y de cuerpos en contacto.

En cambio, existe otra forma de transferencia más común en líquidos y gases, es la convección. Esta forma de transferencia de calor es aquella que resulta del movimiento global de líquidos o gases.

Si el movimiento de un fluido se origina por una diferencia de densidad que acompaña a un cambio en la temperatura, a la transferencia de calor producida de esta manera se le llama convección natural. Cuando un fluido es obligado a moverse por la acción de una bomba o ventilador, la transferencia de calor producida de esta manera se llama convección forzada.

Teniendo la mayoría de las herramientas que se ocuparán en el desarrollo del proyecto, se procederá plantear el diseño que se propone para la solución del problema que se expone. Si en la evolución del presente proyecto, se llegara a solicitar la explicación de alguna herramienta, base o antecedente que no se haya mencionado con anterioridad, se proporcionarán las correspondientes fichas bibliográficas, o si fuese indispensable mencionar dicha parte, se abundará en ella sobre la marcha.

2.4 SISTEMA DE CONTROL AMBIENTAL

Un Sistema de Control Ambiental (SCA) se puede definir como aquel sistema que está conformado por los siguientes subsistemas: calefacción, ventilación y acondicionamiento de aire. Obviamente, al considerarlos como tal, cada uno realiza una función diferente, pero siempre con un fin común.

El acondicionamiento de aire es el proceso en el cual se realiza un tratamiento de aire en un ambiente interior con el objetivo de establecer y mantener los estándares requeridos de temperatura, humedad, limpieza y movimiento, para una comodidad exigida por el ser humano.

Es decir, los principales parámetros del aire pueden ser manipulados o controlados para determinar un cierto comportamiento del mismo. Si se desea controlar la temperatura del aire, esto se logra calentando o enfriando al mismo. Con lo que respecta a la humedad (contenido de vapor de agua en el aire), se controla aumentando o eliminando vapor de agua en el aire. La limpieza puede lograrse mediante la filtración eliminando contaminantes indeseables con filtros u otros dispositivos. Otra medida para la limpieza es la ventilación, haciendo introducir aire del exterior al interior, con lo cual se diluye la concentración de contaminantes. Cuando se habla de movimiento de aire, se refiere a la velocidad y a los lugares donde se puede desplazar y distribuir.

La calefacción es aquel sistema que permite suministrar calor dentro de un compartimento, mientras que al sistema que elimina este calor se le llama sistema de enfriamiento.

De esta manera, podemos resumir que para la realización del proyecto se tienen que tomar en consideración tres importantes partes: una fuente de calefacción, una fuente de enfriamiento y un sistema de distribución.

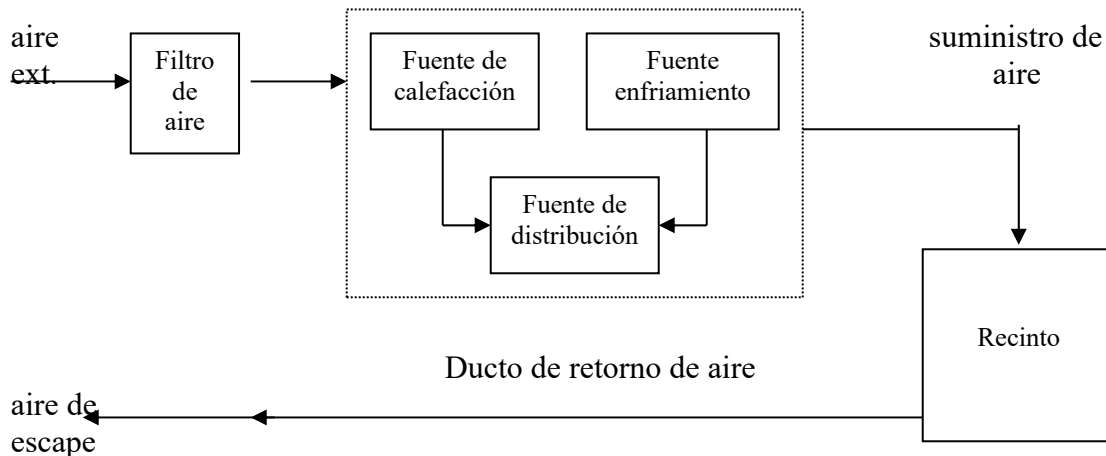


Fig. 2.3 Componentes de un sistema de calefacción y enfriamiento

2.5 PLANTEAMIENTO

Asimismo, la solución que se presenta en este proyecto es una de una diversidad de soluciones que se pueden encontrar en el campo técnico y comercial, ya que muchos compartimentos tienen instalado algún sistema que tenga como objetivo nivelar la temperatura interior a un cierto rango, teniendo como objetivos: confort y condiciones de operación.

La particularidad que presenta este proyecto es el uso de la Lógica Difusa para el sistema automatizado, el cual controlará el nivel de temperatura en el interior del compartimento.

El sistema constará de dos modos de selección para su funcionamiento: manual y automático, pero ambos tomarán como herramienta, para su funcionamiento, la Lógica Difusa.

¿Por qué utilizar un sistema de control automático?. Un sistema de control automático permite seleccionar la temperatura deseada. Primeramente se fija la temperatura deseada, posteriormente, el sistema ajustará de manera automática la variable temperatura, velocidad del ventilador y la potencia de la resistencia, la cual será la fuente de calefacción.

Los controles automáticos para un sistema de calefacción y acondicionamiento de aire, se puede comparar, en funcionamiento e importancia, con el sistema nervioso y el cerebro de un ser humano.

El diseño e instalación de los controles automáticos para un sistema de acondicionamiento de aire deben ser compatibles con el mismo sistema; asimismo deben trabajar en forma correcta. De no ser así, el sistema no producirá condiciones satisfactorias.

Los controles automáticos deben mantener las condiciones de diseño en el compartimento. Dependiendo de dichas condiciones, se seleccionará la capacidad de calefacción y enfriamiento del sistema. Además, gracias a los controles automáticos la intervención humana se reduce considerablemente, dentro del funcionamiento y servicio del sistema, esto reduce costos y posibilidades de errores. Una de las consideraciones más importantes que surgen al hacer uso de un sistema de control automático es que éste tiene la capacidad para reducir el uso de energía.

El sistema que se expone está integrado de la siguiente forma:

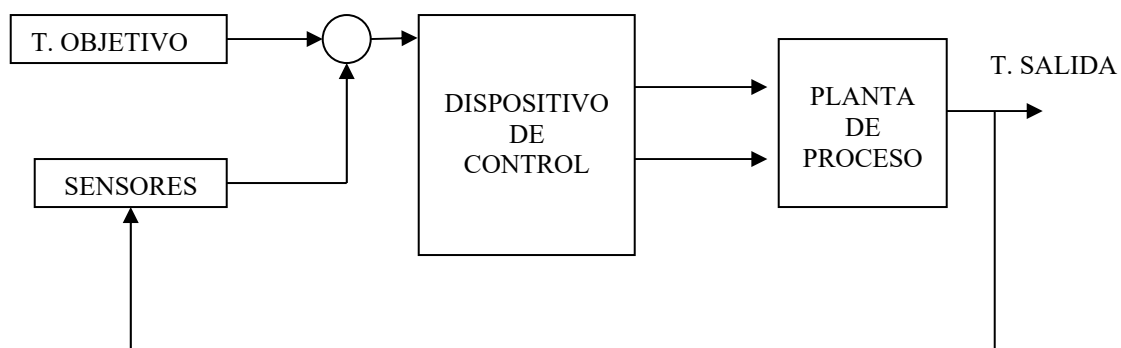


Fig. 2.4 Diagrama de bloques del sistema de control propuesto

De acuerdo a la figura 2.4 se tiene, en la entrada del sistema, la variable a controlar que es la temperatura. Dicha variable será sensada en el interior y exterior del compartimento.

Se tiene el dispositivo de control, el cual percibe los diversos cambios en la variable controlada y transmite una acción a un dispositivo que se encargará de corregir la condición que cambió.

El otro dispositivo mencionado es la planta de proceso, llamada también etapa de potencia. Este dispositivo se encargará de percibir la regulación del sistema de control, con la cual se varía la capacidad u operación de la misma planta.

Al final del proceso, se tendrá como salida el cambio deseado en la variable, pero sólo en un instante. Esta salida se retroalimenta al sistema para compararla con otra muestra de la variable controlada en otro instante de tiempo. Por lo tanto, la característica del control propuesto es proporcional derivativa, la variable a controlar tiende a un determinado valor en un cierto rango de tiempo.

Por otro lado, para establecer las condiciones necesarias y específicas que requiere el sistema de control para su funcionalidad, se hace uso de las herramientas matemáticas que hasta el momento se exponen.

Para realizar un intercambio de temperatura con el cual se pueda adquirir o disipar calor, el proceso realizará una convección forzada con el aire que entra al sistema.

Ahora, para poder transmitir calor hacia el interior del compartimiento, es necesario hacer uso de un actuador que realice dicho trabajo en un intervalo de tiempo considerable.

Para poder considerar las características de dicho actuador, se tomará como base la ecuación 2.b :

$$Q = mC(T_1 - T_2)$$

Inicialmente, considérese una variación de temperatura ΔT en un intervalo de tiempo, es decir el objetivo del control es llegar a obtener una temperatura deseada a partir de un valor de referencia, por ejemplo, podemos tener una temperatura ambiente de 14°C y se desea tener una temperatura de 21°C después de un tiempo determinado. De esta manera , tenemos un ΔT definido como $\Delta T = T_2 - T_1$, donde $T_2 = 21^\circ \text{C}$ y $T_1 = 14^\circ \text{C}$.

La constante C corresponde al calor específico del aire, ya que lo que estará entrando o saliendo al sistema será el mismo aire a diferentes grados de temperatura.

En lo que se refiere a la variable m , corresponderá al gasto másico que pueda mantener el sistema, ya que es necesario tener conocimiento de que cantidad de aire entra al sistema en cierto intervalo de tiempo.

Para determinar esta constante fue necesario auxiliarse de un medidor de flujo, de esta manera y considerando la lectura obtenida en el mismo, se determinó el gasto másico. En la figura 2.5 se ilustra el tipo de medidor de flujo que auxilió a esta parte del proceso experimental.

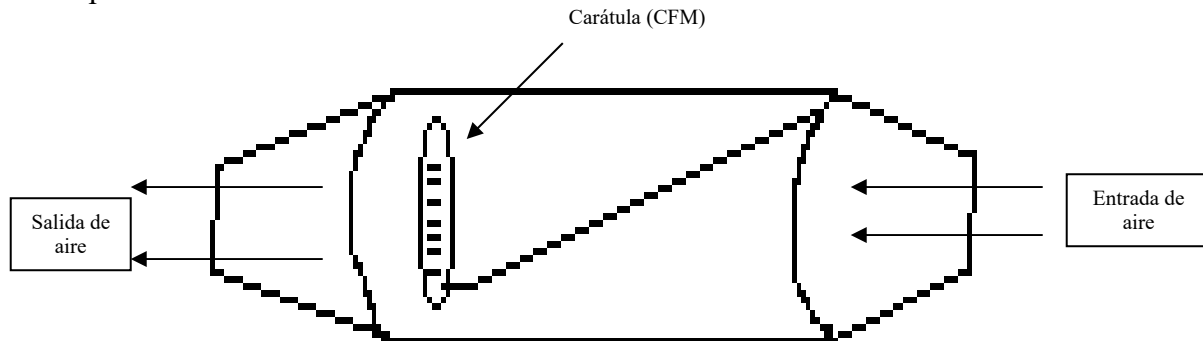


Fig. 2.5 Medidor de flujo de aire

Haciendo uso de un ventilador, se genera el flujo de aire, el cual se hace pasar por el medidor, éste marcará un valor en pies cúbicos por minuto (CFM).

El valor leído, una vez realizada la prueba, fue de $15 \text{ ft}^3 / \text{min}$.

Tenemos la siguiente equivalencia: $1 \text{ CFM} = 1.7 \text{ m}^3 / \text{h}$

De esta manera, realizando la conversión: $15 \text{ CFM} = 25.5 \text{ m}^3 / \text{h}$, este valor equivale al gasto volumétrico que tiene el sistema.

Pero, para poder obtener el gasto másico ($g_m = m$), el cual es necesario para poder obtener los datos completos de la ecuación 2.b, se multiplica el valor obtenido anteriormente por la densidad del aire $\rho = 1.29 \text{ kg} / \text{m}^3$:

$$g_m = m = (1.29 \text{ kg} / \text{m}^3) (25.5 \text{ m}^3 / \text{h}) = 32.86 \text{ Kg} / \text{h}$$

De esta manera, tenemos que pasan 32.86 Kg de aire por cada hora.

Realizando la conversión:

$$(32.86 \text{ Kg} / \text{h}) (2.2 \text{ lb} / 1 \text{ kg}) = 72.292 \text{ lb} / \text{h}$$

Por otro lado, el calor específico correspondiente al aire tiene un valor de $C_{\text{esp a}} = 0.24 \text{ BTU} / \text{lb} \text{ } ^\circ\text{F}$.

Suponga que se tiene 0° C y se desea tener 20° C , para lograr lo anterior es necesario considerar cierta potencia del actuador encargado de proporcionar energía calorífica, pero si esto se requiere en un tiempo menor a una hora, dicha potencia deberá variar considerablemente.

Consideremos una variación de temperatura de $\Delta T = 30^\circ \text{ C}$ en una hora. Aún sabiendo que el comportamiento no es lineal, se puede aceptar el grado de respuesta que proporcione el actuador una vez teniendo la máxima potencia, ya que la variación de 0° a 20° C podrá obtenerse en un intervalo de tiempo menor a una hora.

Ya establecidos los parámetros que requiere la ecuación 2.b, se procede a realizar el proceso de cálculo para obtener la disipación de energía por unidad de tiempo que permita determinar las características del actuador a utilizar.

Tenemos:

$$Q = mC\Delta T$$

donde: $m = 72.292 \text{ lb / h}$
 $C_{\text{esp a}} = 0.24 \text{ BTU / lb }^\circ\text{F}$.
 $\Delta T = 30^\circ \text{ C}$, $\Delta T = 111.6^\circ \text{ F}$

así:

$$Q = (72.292 \text{ lb / h}) (0.24 \text{ BTU / lb }^\circ\text{F}) (111.6^\circ\text{F})$$
$$Q = 1936.48 \text{ BTU / h}$$

pero, $1 \text{ BTU} = 1055 \text{ J}$, entonces:

$$Q = (1936.48 \text{ BTU / h}) (1055 \text{ J / 1BTU}) (1\text{h} / 3600 \text{ s})$$
$$Q = 567.5 \text{ J/ s} = 567.5 \text{ W}$$

De esta manera, el actuador que proporcionará energía calorífica será una resistencia que tenga una potencia de aproximadamente 500 Watts.

En el capítulo cuatro y cinco se mencionará todo acerca de cómo está constituida físicamente cada parte del sistema. En el capítulo cuatro corresponderá mencionar todo lo correspondiente al hardware del sistema, se hace mención de toda la estructura física, tanto de la parte de sensado, como de la parte de control y la etapa de potencia, se abunda en los principios que se adoptaron para la realización de cada etapa y los pasos que se siguieron para su respectivo desarrollo.

En el capítulo cinco corresponderá mencionar todo acerca del software del sistema, su estructura y desarrollo, los principios que se siguieron y el resultado que se obtuvo una vez finalizado.

CAPITULO 3 CONTROL DIFUSO

3.1 PRINCIPIOS DE CONTROL

El diseño del control de temperatura se desarrolla con un arreglo de componentes físicos, organizados de tal forma que el funcionamiento del mismo puede alterar, regular o mandar, a través de una acción de control, a otro sistema físico para que éste exhiba ciertas características o comportamientos deseados.

Por otro lado, de acuerdo a la teoría de control, un control de lazo abierto es aquel en el cual la acción de control es independiente de la salida del sistema físico. Sin embargo, en un sistema de control de lazo cerrado (también conocido como un sistema de control retroalimentado), la acción de control depende de la salida del sistema físico.

Un tostador donde la cantidad de calor es determinada manualmente o una máquina de lavado en la cual la temperatura y el tiempo de ciclo de centrifugado son determinados de igual forma, son ejemplos de sistema de control de lazo abierto, donde la acción de control no es una función de la salida.

El termostato de un cuarto, el cual sensa la temperatura de un compartimiento y activa una unidad de enfriamiento o calentamiento cuando un cierto umbral de temperatura es alcanzado o, un mecanismo de pilotaje automático, el cual realiza correcciones del curso de un aeroplano cuando son sensados valores de algunos parámetros por instrumentos del mismo, son ejemplos de sistemas de control retroalimentado.

De esta manera, las ventajas que proporciona un control de lazo cerrado permite desarrollar un control más versátil que se adapte a las condiciones deseadas por el diseñador.

Si el objetivo de un sistema de control es mantener una variable física en un valor constante ante la presencia de disturbios, el sistema es de tipo regulatorio o llamado regulador. El control de temperatura de un cuarto y el pilotaje automático son ejemplos de controladores regulados.

Sin embargo, existen controladores en los cuales se requiere que la variable física siga alguna función de tiempo deseada. Por ejemplo, un sistema automático de aterrizaje, en el cual se requiere seguir una rampa para llegar al punto deseado.

La salida o respuesta del sistema de control es ajustada por una señal de error. Esta señal de error es la diferencia que existe entre la respuesta actual del sistema y la señal de entrada que proporciona el sensor, pero cabe mencionar que la respuesta deseada está especificada por una entrada de referencia.

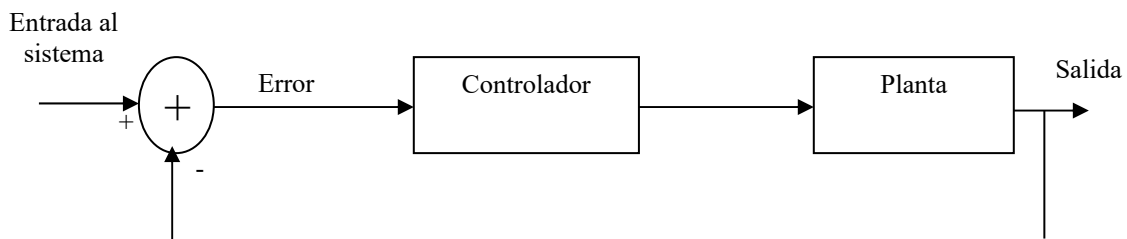


Fig. 3.1 Sistema de control de lazo cerrado

Para controlar una variable física, primeramente se debe medir. De esta manera, inicialmente se tiene un dispositivo que mida dicha variable llamado sensor; una parte posterior al sensor es la planta. En un sistema de control de lazo cerrado, ciertas señales del sistema, específicamente las señales de entrada, son determinadas por la respuesta del sistema o señales de salida. Al conectar un compensador o controlador dentro del lazo se pueden obtener resultados y características satisfactorias durante el control.

Es indiscutible que la teoría de control es extensa y, conforme aumenta la estructura y funcionalidad del sistema, la complejidad del control y todo lo que él conlleva, aumenta su desarrollo hasta llegar a un máximo de complejidad que origine un sistema de control poco práctico y costoso.

Sin embargo, el objetivo es realizar un sistema de control, donde se mantenga la temperatura dentro de los parámetros deseados en el interior de un recinto y que base su desarrollo en los principios expuestos por la teoría difusa y sus aplicaciones al control de procesos, la cual puede brindar, en diversas ocasiones, una menor complejidad a la que pueda existir al considerar un control clásico.

3.2 LOGICA DIFUSA

En el año de 1965 el Dr. Lotfi A. Zadeh dio a conocer el primer documento sobre la teoría de Lógica Difusa (Fuzzy Logic) siendo el pionero de esta teoría. Puede decirse que la lógica difusa es otra forma de pensar, ya que a diferencia de la lógica clásica o aristotélica donde el todo se puede clasificar como falso ó verdadero, en Lógica Difusa se puede tener diferentes grados de pertenencia, es decir, ya no se tiene la restricción de los valores falso o verdadero, sino que se pueden tener varios grados como muy falso, poco falso, poco cierto, muy cierto, etc. Numéricamente, la representación de grados de pertenencia sería como: 0.8falso, 0.2falso, 0.2cierto, 0.8 cierto, etc.

Si comparamos la representación de los conjuntos en un diagrama de Venn-Euler sería como se muestra en la fig. 3.2:

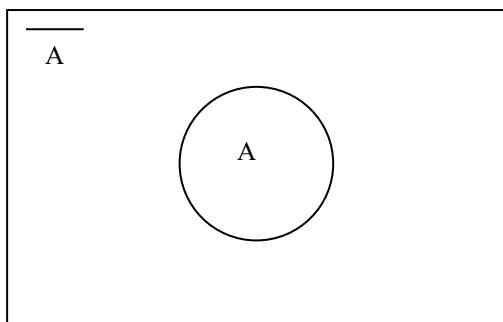


FIG. 3.2(a)

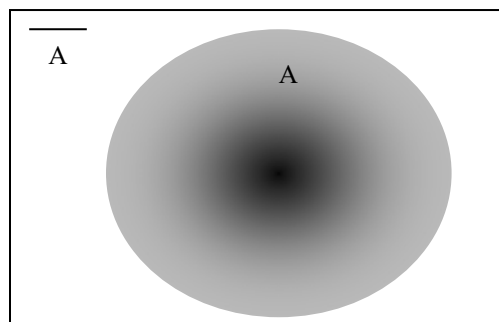


FIG. 3.2 (b)

Fig. 3.2 Representación Euler de conjunto crisp y difuso.

En la figura 3.2(a) se muestra la representación de los conjuntos “Crisp” o aristotélicos, donde se observa que el contorno de los conjuntos están definidos y se nota con claridad la división entre conjuntos, mientras que en la fig 3.2(b) se esquematiza a los conjuntos difusos, donde se observa que la división de estos no están bien definida.

Se observa que el centro de los conjuntos es más obscuro esto es al grado de pertenencia de los elementos, cabe mencionar que el grado de pertenencia es complementario, es decir, mientras un elemento pertenezca más a un conjunto pertenece menos al otro. Esto último se puede representar numéricamente, por ejemplo, si tiene un grado de pertenencia de 0.75 falso tendrá el 0.25 de verdadero.

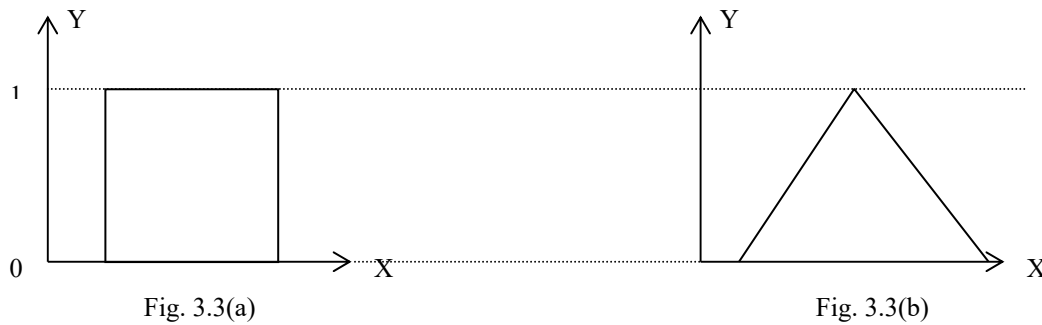


Fig. 3.3 Representación cartesiana de conjuntos crisp y difusos.

Otra forma de representar a los conjuntos es por medio de planos cartesianos. En la figura 3.3(a) se observa la representación de un conjunto “Crisp” en donde el falso (cuya representación numérica es de un “0”) pasa al verdadero (que se puede representar por un “1”) de manera abrupta, mientras que en la figura 3.3 (b) se puede ver a la representación de un conjunto difuso, donde la transición de falso a verdadero y viceversa se desarrolla por medio de una curva, la cual es una línea donde el cambio es paulatino. Cabe mencionar que los conjuntos “Crisp” se pueden considerar como una situación especial de un conjunto difuso, es decir, la lógica difusa abarca a la lógica aristotélica. Una de las características de los conjuntos difusos es que no cumplen con las propiedades de la Ley del Medio Excluido y Ley de la Contradicción.

La gráfica que se muestra en la figura 3.3 (b) es llamada función de membresía y representa el valor correspondiente de una variable X a un valor difuso Y. A Y también se le llama $\mu(x)$. La forma del conjunto difuso representada en la figura 3.3(b) no solamente es triangular (tipo λ) sino que puede obtener diferentes formas tales como Gaussiana, Singleton, tipo Z, tipo S, tipo ∇ .

Un conjunto difuso tiene partes que se identifican como lo muestra la figura 3.4.

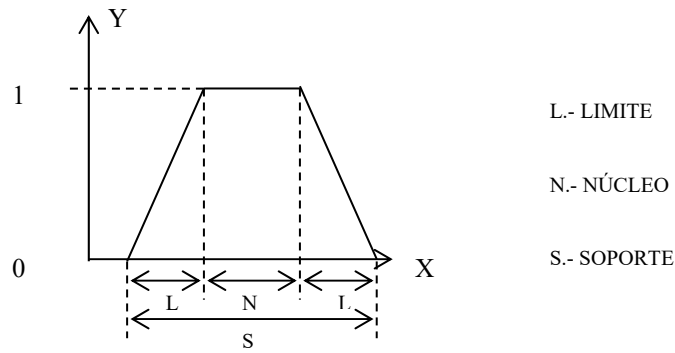


Fig. 3.4 Partes que identifican a un conjunto difuso.

La álgebra de los conjuntos difusos es similar a la de los conjuntos Crisp, ya que comparten muchas de sus respectivas propiedades diferenciándose por las dos leyes antes mencionadas, pero cabe mencionar que sus representaciones pueden variar, por ejemplo un conjunto difuso se representa por letras mayúsculas, al igual que un conjunto Crisp, pero sobre una línea curvada y su definición estaría representada como sigue:

$$\underset{\sim}{A}(x) = \{x_i / 0 \leq \mu(x_i) \leq 1\} \text{-----}(3.1)$$

La representación del complemento, sería de la siguiente forma:

$$\overline{\underset{\sim}{A}} = \underset{\sim}{\mu_A}(x) = 1 - \underset{\sim}{\mu_A}(x) \text{-----}(3.2)$$

Para poder trabajar con estos conjuntos debe ser considerada la forma en que se aplican los operadores de unión, intersección y complemento. Este último se representó en la ecuación (3.2) cuya interpretación sería: si se tiene un determinado valor difuso, por ejemplo, 0.35 el complemento será lo que falte para un valor de uno, (0.65).

El operador unión puede visualizarse si consideramos una tabla binaria, ya que la unión es semejante al operador "OR" como se muestra en la tabla 3.1. Si se observa con detalle el resultado es el mayor (1) de los valores, de esta manera, podemos decir que el resultado de la unión de dos conjuntos serán los elementos con mayor valor de pertenencia, y su representación es la siguiente:

$$\underset{\sim}{\mu_{A \cup B}}(x) = \max(\underset{\sim}{\mu_A}(x), \underset{\sim}{\mu_B}(x)) \text{-----}(3.3)$$

A	B	C = (AUB)
0	0	0
0	1	1
1	0	1
1	1	1

Tabla 3.1

Para la intersección al igual que en la unión puede interpretarse a partir de una tabla binaria, donde la operación de intersección es semejante a la “AND”. A partir de la tabla 3.2 se deduce que el resultado sigue al valor menor de la tabla, de esta manera, la solución de la intersección son los elementos con menor grado de pertenencia y su representación es la siguiente:

$$\mu_{A \wedge B}(x) = \min(\mu_A(x), \mu_B(x)) \text{-----(3.4)}$$

A	B	C = (A ∧ B)
0	0	0
0	1	0
1	0	0
1	1	1

Tabla 3.2

A partir de estos operadores y de sus propiedades pueden realizarse prácticamente todas las operaciones entre conjuntos difusos como son las relaciones, composiciones y otras.

De esta manera, se visualiza la facilidad de la lógica difusa en diferentes áreas y con ello tener la posibilidad de desarrollar sistemas más rápidos, más robustos y de menor costo, tanto para los requerimientos de cálculo, de software y en algunos casos de hardware.

Con la lógica difusa es posible manejar un lenguaje más sencillo, es decir, menos rígido. Por ejemplo: al referirse al parámetro temperatura, ya no es necesario que tome un valor rígido de 5 grados centígrados, sino que puede ubicarse en una etiqueta lingüística llamada temperatura fría con un cierto grado de pertenencia, de igual forma 40 grados centígrados se podría ubicar en otra etiqueta llamada temperatura caliente, también con un cierto grado de pertenencia.

Para pasar una variable a un valor difuso (fuzzificación) se tienen diferentes métodos, estos métodos dan el valor de pertenencia de la variable a los diferentes conjuntos difusos. Estos métodos pueden ser: Intuición, Inferencia, Ordenación por Rango, Conjuntos Difusos, Redes Neuronales, Algoritmos Genéticos, Razonamiento Inductivo.

Para la aplicación expuesta se utilizan conjuntos difusos cuyos puntos son determinados. Con los valores difusos se evalúan las reglas que son las que determinan el comportamiento del sistema, al evaluar realizan una modulación de los conjuntos de salida y a partir de ésta modulación se origina la salida difusa para la cual se tiene diferentes métodos de desfusión como son: Principio del Máximo, Centroide, Promedio Pesado, Promedio de Máximos, Centro de Sumas, Centro del Área Mayor, Primero o Último de Máximos, Singletons.

Para conocer la estructura de un control difuso se tiene el siguiente diagrama en el cual se muestran todos los bloques que intervienen en el proceso.

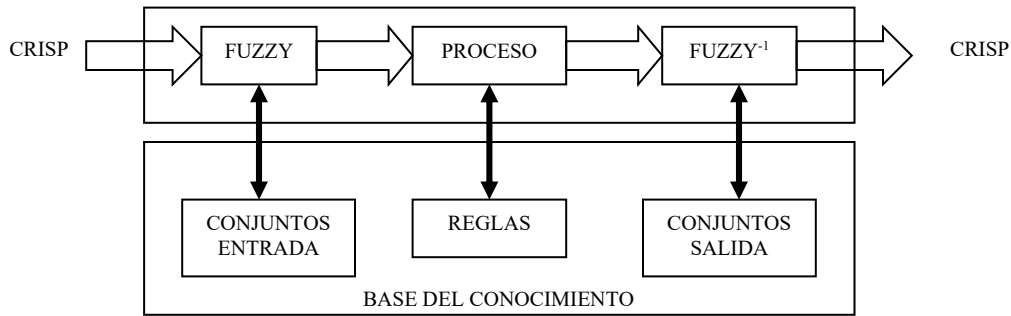


Fig. 3.5 Diagrama s bloques de un control difuso.

El bloque FUZZY junto con el bloque CONJUNTOS DE ENTRADA se encarga de pasar un valor físico a un valor difuso, este valor depende básicamente de cómo estén conformados los conjuntos de entrada (tipo, soporte, límites, núcleo). El siguiente bloque participante es el de PROCESO que junto con el bloque de REGLAS evalúan las condiciones establecidas en el diseño de un sistema y de esta forma se obtienen los primeros resultados, los cuales pasan a los últimos bloques que son los de FUZZY⁻¹ y al de CONJUNTOS DE SALIDA, los cuales se encargan de evaluar los resultados preliminares junto con los conjuntos de salida para, finalmente, dar un el valor nítido de salida, pero este valor deberá ser congruente con los actuadores de control. Esta es la forma simplificada de como trabaja un sistema de control difuso.

A continuación se expone un ejemplo que permita ilustrar, con más claridad, el proceso de un control difuso. La forma de proceder es la siguiente: se definen los conjuntos difusos de tal manera que sean congruentes con las variables a utilizar, por ejemplo: Poco Caliente (0 °C a 20 °C), Caliente (10 °C a 30 °C) y Muy Caliente (20 °C a 40 °C). La representación de los conjuntos se ilustra en la figura 3.6.

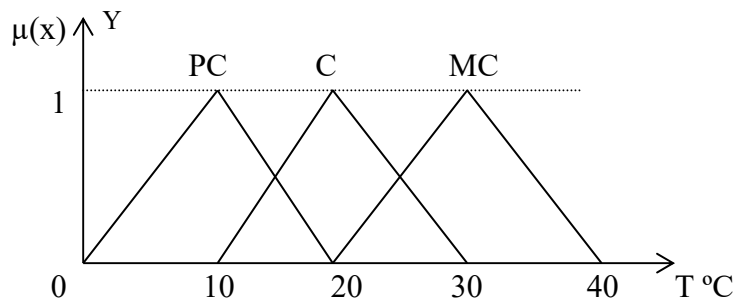


Fig. 3.6 Representación de conjuntos difusos de entrada

donde

PC = Poco Caliente

C = Caliente
MC = Muy Caliente

Ahora definimos de forma similar nuestros conjuntos de salida, donde la variable será el ciclo de trabajo de la resistencia. Los conjuntos definidos son Poco Intensa, Intensa, Muy Intensa. Ver fig. 3.7.

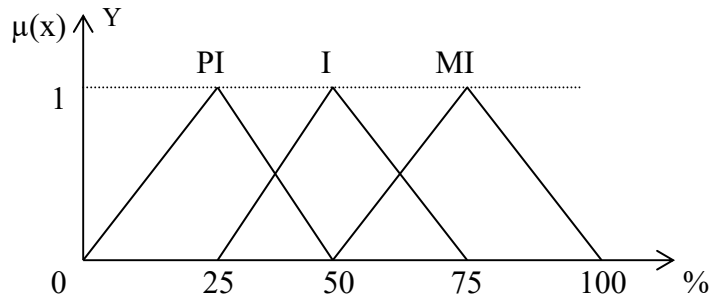


Fig. 3.7 Representación de conjuntos difusos de salida

donde

PI = Poco Intenso
I = Intenso
MI = Muy Intenso

Teniendo ya definidos los conjuntos de entrada y de salida se procede a realizar las reglas. Estas reglas se basan en los Sistemas Basados en Reglas y una regla lingüística esta formada por juicios de asignación, juicios de condición y juicios incondicionales, es decir, la estructura de la regla es de la siguiente forma: IF A^S THEN B^S , donde A^S representa a los conjuntos de entrada y B^S los conjuntos de salida.

Como se menciono con anterioridad, la relación que se da entre la entrada y la salida es por medio de reglas, por ejemplo: si es poco caliente entonces la potencia es muy intensa, si es muy caliente entonces la potencia es poco intensa, etc. El sistema se encarga, por medio de algún método de fuzzificación, de pasar la variable física a un valor difuso, esto se realiza por medio de los conjuntos de entrada, posteriormente se evalúan las reglas que proporcionan valores difusos de salida, los cuales son evaluados a través de los conjuntos de salida y por último mediante un método de desfuzzificación se obtiene la salida nítida correspondiente, para esto el sistema cuenta con un Kernel, quien se encarga de decodificar y manejar la información entre las variables difusas, los conjuntos de entrada, la evaluación de reglas y el correspondiente conjunto de salida.

El Kernel es una parte medular del sistema, ya que es el encargado de llevar la lógica difusa a la plataforma a utilizar. El ejemplo anterior tiene una entrada con su conjunto difuso definido y una salida también con su conjunto difuso definido por lo que las reglas son relativamente pocas, pero conforme el número de entradas aumenten las

reglas aumentan de tal forma que el Kernel deberá ser lo suficientemente robusto para manejar el sistema.

El capítulo 5 se encargará precisamente de la constitución del Kernel y del software en general.

Continuando con la solución del ejercicio se debe tener las reglas definidas para poder llevar a cabo el control, para esto se tienen las siguientes reglas:

Si x es PC entonces MI

Si x es C entonces I

Si x es MC entonces PI

Puesto que es un ejemplo con una sola entrada y una sola salida las reglas resultan muy sencillas y fáciles de definir. Teniendo ya las reglas definidas se tiene completo el sistema de control, ahora sólo se tiene que evaluar las diferentes entradas al sistema, para ello se considera que en un instante de tiempo la entrada x es de 13°C . Se debe recordar que es un control continuo, por lo que se tienen que evaluar muchos valores en un periodo corto de tiempo, pero solo se considera un valor para simplificar el ejercicio. Primero se transfiere el valor “CRISP” a un valor “Difuso” como se muestra en la figura 3.8.

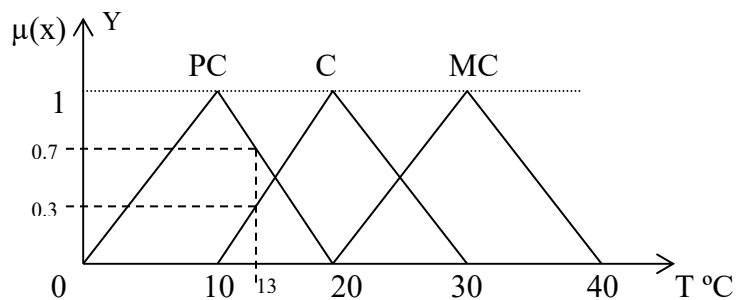


Fig. 3.8

Como se puede observar en la figura 3.9, los conjuntos habilitados son PC y C por lo que tenemos dos valores difusos, uno para cada conjunto. Ahora, evaluando las reglas se tiene que para la primera corresponde al conjunto de salida MI, mientras que para la segunda corresponde I, el tercer conjunto no se habilita, es decir tendrá un valor cero, esto se puede ver más claramente en la figura 3.9.

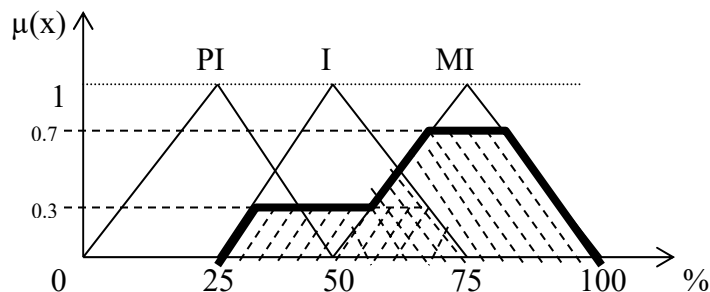


Fig. 3.9

Si aislamos el resultado quedará como se indica en la figura 3.10.

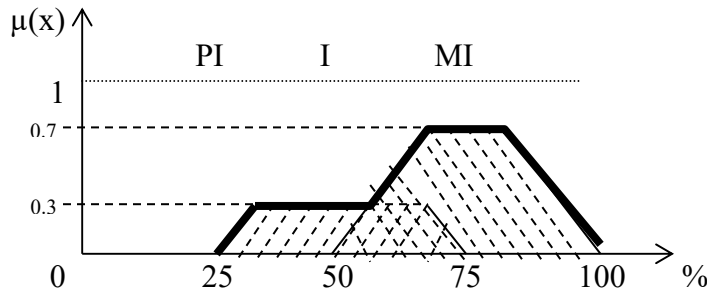


Fig. 3.10

La solución final de este problema se obtiene al aplicar un método de desfusión y de este modo se obtiene el ciclo de trabajo a la cual deberá operar el actuador. Cabe mencionar que el método de desfusión se selecciona a partir de la respuesta requerida al sistema, por ejemplo, que tan exacto debe ser, que tan rápido debe responder, etc. Para muchos sistemas el método de SINGLETONS es suficiente para obtener resultados bastante aceptables, además es el método que requiere, hasta cierto punto, menor requerimiento de software y calculo, lo que lo hace más rápido y más sencillo de manejar. Retomando el ejemplo, primeramente debe definirse los Singletons de la salida tal como muestra la figura 3.11. Como se puede observar, la representación consisten en unas flechas que apunta al valor asignado y dependiendo de este valor es como responderá la salida, es decir, al hacer cualquier cambio en los Singletons existirá un cambio en la salida. Se colocaron también los conjuntos anteriores con una línea punteada solo para mostrar la diferencia que existe.

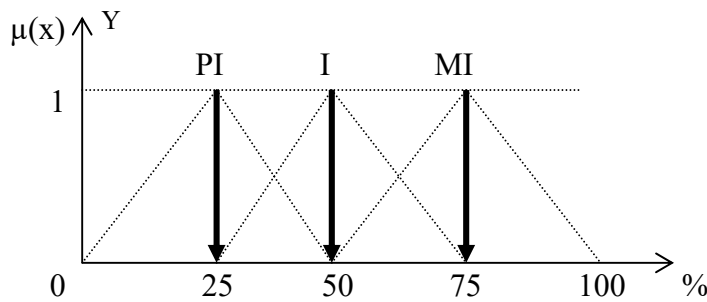


Fig. 3.11 Singletons

Procediendo a evaluar reglas, se tiene respuesta mostrada en la figura 3.12

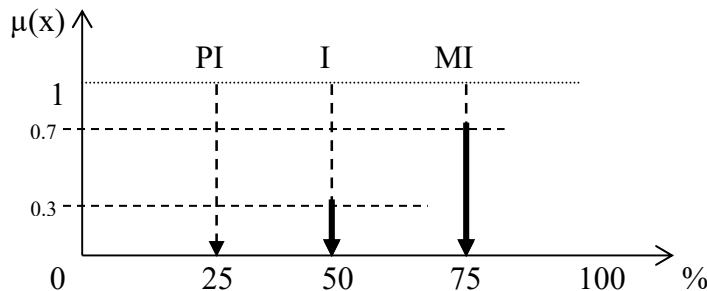


Fig. 3.12

Al aplicar la ecuación 3.5 se obtiene el resultado buscado, el cual puede ser ilustrado en la figura 3.13.

$$Z^* = \frac{\sum \mu(z)_i \cdot z_i}{\sum \mu(z)_i} \text{-----(3.5)}$$

$$Z^* = ((0.3 \cdot 50) + (0.7 \cdot 75)) / (0.3 + 0.7)$$

$$Z^* = 67.5 \%$$

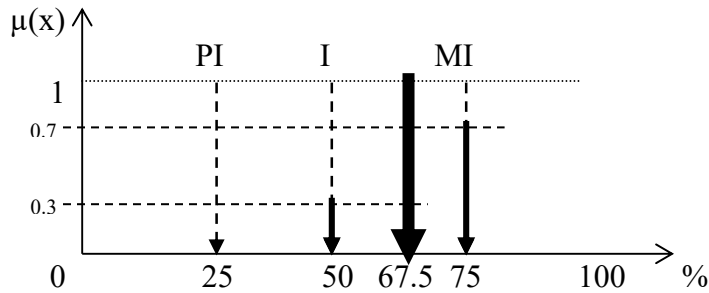
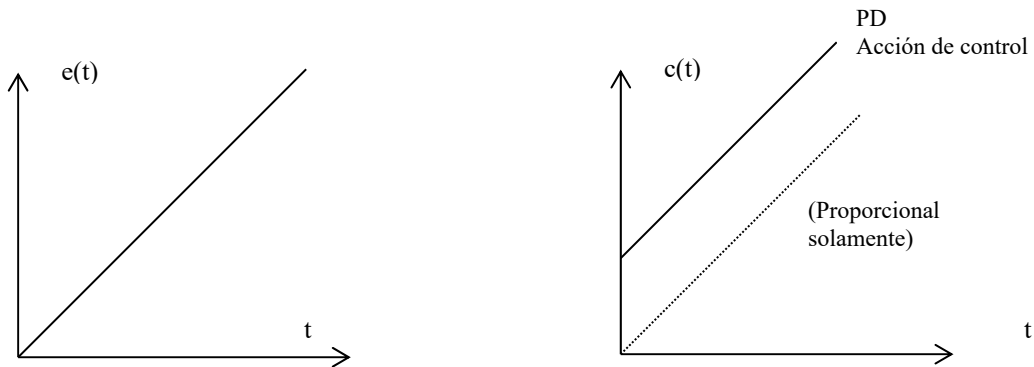


Fig. 3.13

3.3 PLANTEAMIENTO DEL CONTROL DIFUSO

Es conveniente establecer un orden para el planteamiento del diseño del sistema de control difuso que permita abordar, de una forma coherente, la solución del problema.

Primeramente, cabe mencionar que la acción de control a desarrollar tendrá característica derivativa, es decir, considérese el valor de salida del control proporcional a la velocidad de variación de la señal de error actuante (sistema retroalimentado). Así, la acción del control derivativa presenta características de anticipación. (Ver figura 3.14)



3.14 (a) Entrada: rampa unitaria

3.14 (b) Salida de control: proporcional derivativa

Fig. 3.14 Acción de Control Derivativa

Sin embargo, la acción derivativa nunca puede derivarse a una acción que aún no ha tenido lugar.

En la figura 3.15 se muestra la Máquina de Inferencia Difusa, la cual esquematiza la estructura del control a realizar:

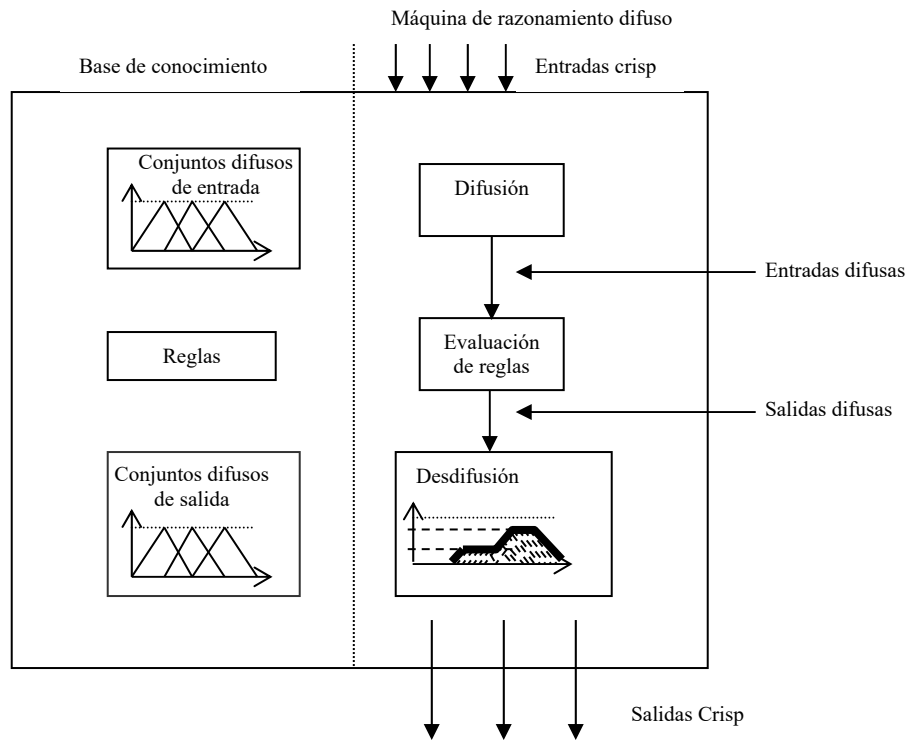


Fig. 3.15 Máquina de Inferencia Difusa

Lo anterior permite identificar las partes difusas del sistema. Hay que observar que partes presentan mayor ambigüedad para el uso de un control difuso y, en caso contrario, apegarse a la metodología de un control clásico.

Una vez identificadas las partes difusas del sistema deben ser señaladas las entradas y salidas del mismo, con el fin de obtener variables de control.

Cabe mencionar que hay que analizar y simplificar el problema planteado, lo que permitirá particionar y resolver de una forma modular, simplificando el trabajo.

De esta manera se definen funciones de membresía de entrada y salida, para ello es necesario especificar el universo para definir cual es el rango de variación de las mismas, así como su comportamiento para establecer de forma correcta los conjuntos difusos.

Una vez establecido el universo se prosigue a particionarlo, dependiendo de las entradas y salidas se ajunta su escala lineal, logarítmica o cualquier otra.

De acuerdo a la figura 3.1 la respuesta del sistema de control es ajustada por una señal de error, así primeramente se considera al error como una variable de entrada y estará definido como $E=T_0-T_f$, donde:

E = Error

T_0 = Temperatura inicial

T_f = Temperatura final

La temperatura inicial es aquella que se sensa una vez que se ha iniciado el control.

La temperatura final es aquella que se presenta después de que el proceso de control ha realizado sus funciones en un intervalo de tiempo dado.

Por otro lado, el sistema basará su funcionalidad en los principios del control proporcional derivativo, por lo cual se define como segunda entrada a la variación con respecto al tiempo de la señal de error actuante que presenta el sistema de control. Es decir:

E' = Velocidad de variación del error => Control proporcional derivativo.

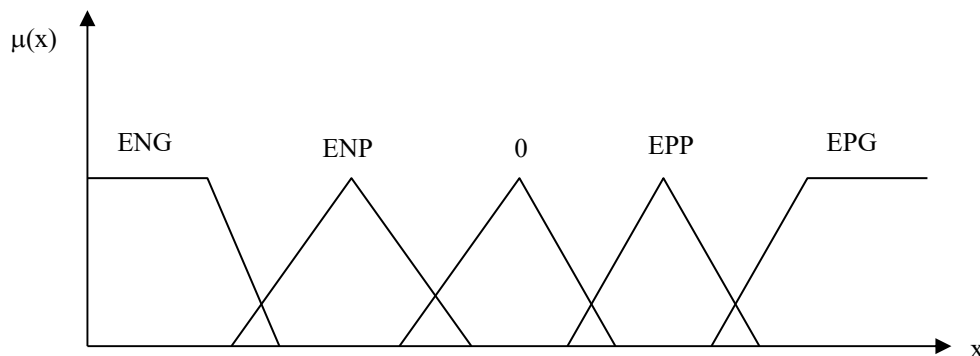
De esta manera, se tiene:

$$E' = \Delta E / \Delta t = \Delta(T_0 - T_f) / \Delta t$$

$$\text{Así, } E' = \Delta E / \Delta t = (E_2 - E_1) / (t_2 - t_1)$$

Se define el universo del discurso y se particiona para establecer los conjuntos correspondientes a las dos entradas.

Se establecen los conjuntos correspondientes a la primera entrada, cuya variable es el error, de la siguiente manera:

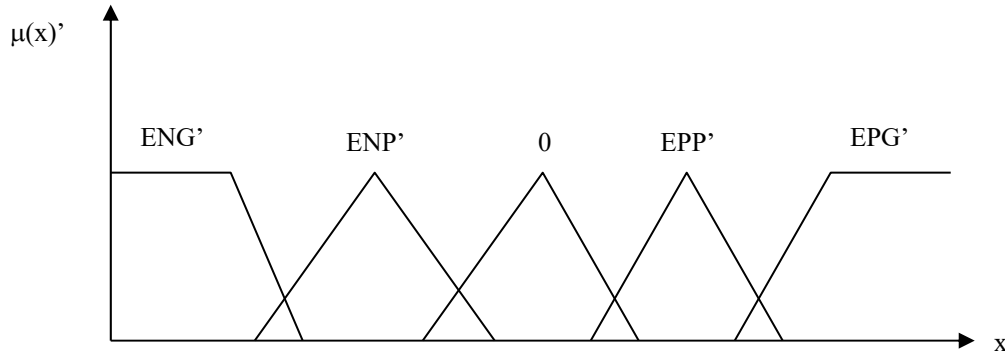


Donde:

$\mu(x)$ corresponde a la función de membresía que representa al error

ENG: Error Negativo Grande
 ENP: Error Negativo Pequeño
 0 : Cero
 EPP: Error Positivo Pequeño
 EPG: Error Positivo Grande

Por otro lado, se definen los conjuntos que corresponden a la segunda entrada, cuya variable representa la variación con respecto al tiempo del error.



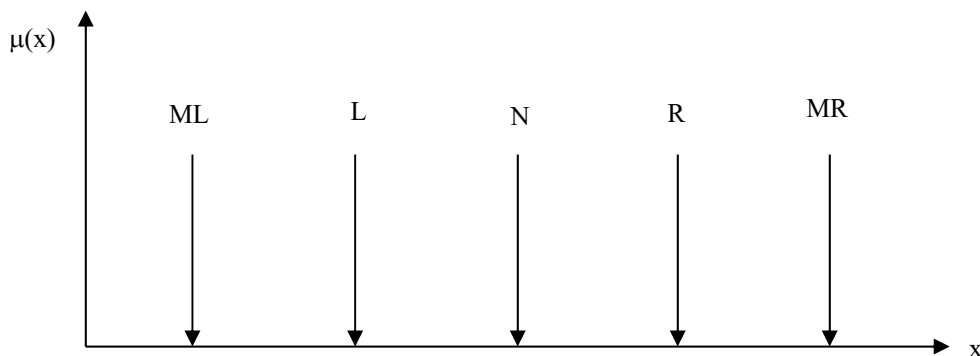
Donde:

$\mu(x)'$ corresponde a la derivada del error
 ENG': Error Negativo Grande
 ENP': Error Negativo Pequeño
 0 : Cero
 EPP': Error Positivo Pequeño
 EPG': Error Positivo Grande

Las salidas determinadas en el proceso serán aplicadas directamente a los actuadores del sistema y se verán reflejadas en el funcionamiento de un motor y/o una resistencia.

De esta manera, se definen los siguientes conjuntos difusos de salida:

Para el motor, tenemos:

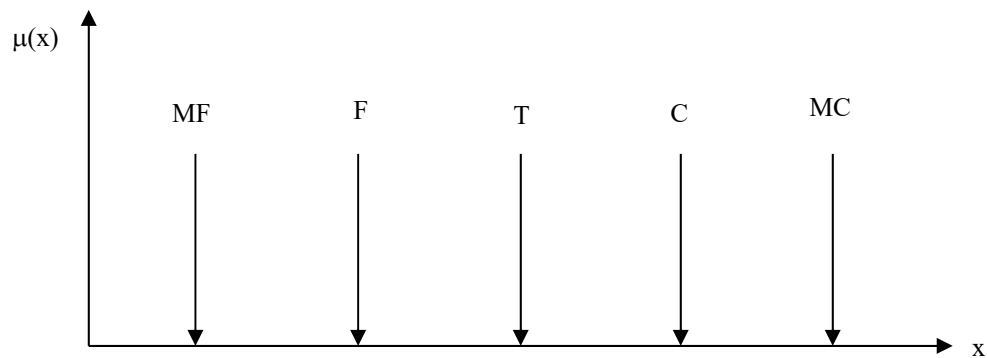


Donde:

ML: Muy Lento
L : Lento
N : Nulo
R : Rápido
MR: Muy Rápido

Las definiciones lingüísticas corresponden a la velocidad del motor.

Para la resistencia se considera lo siguiente:



Donde:

MF: Muy Fría
F : Fría
T : Templada
C : Caliente
MC: Muy Caliente

Estas definiciones lingüísticas corresponden a la disipación de energía por unidad de tiempo que presente la resistencia.

Una vez, habiendo establecido los conjuntos difusos de entrada y salida, así como su respectivo universo escalado, se procede a determinar las acciones de control de manera lingüística, es decir, la elaboración de las reglas de control.

En una primera instancia, puede realizarse la escritura de reglas obvias que implican el determinar, a grosso modo, el proceso de control.

Una vez que se determinan todas las reglas puede ser exentas algunas de ellas, o bien determinar casos especiales que no resulten tan obvios al implicar criterios contradictorios. Cabe señalar que en diversas ocasiones, la exención de reglas puede realizarse con la ayuda de un experto en el campo donde se desea realizar el control.

De acuerdo a los conjuntos definidos, las reglas propuestas son las siguientes:

Reglas correspondientes a la salida del primer actuador (resistencia):

IF ENG AND ENG' THEN F
 IF ENG AND ENP' THEN MF
 IF ENG AND Z' THEN MF
 IF ENG AND EPP' THEN MF
 IF ENG AND EPG' THEN MF

IF ENP AND ENG' THEN F
 IF ENP AND ENP' THEN F
 IF ENP AND Z' THEN MF
 IF ENP AND EPP' THEN MF
 IF ENP AND EPG' THEN MF

IF Z AND ENG' THEN T
 IF Z AND ENP' THEN F
 IF Z AND Z' THEN F
 IF Z AND EPP' THEN MF
 IF Z AND EPG' THEN MF

IF EPP AND ENG' THEN C
 IF EPP AND ENP' THEN T
 IF EPP AND Z' THEN T
 IF EPP AND EPP' THEN F
 IF EPP AND EPG' THEN MF

IF EPG AND ENG' THEN MC
 IF EPG AND ENP' THEN C
 IF EPG AND Z' THEN C
 IF EPG AND EPP' THEN T
 IF EPG AND EPG' THEN F

La FAM correspondiente a estas reglas es la siguiente:

	ENG	ENP	Z	EPP	EPG
EPG'	MF	MF	MF	MF	F
EPP'	MF	MF	MF	F	T
Z	MF	MF	F	T	C
ENP'	MF	F	F	T	C
ENG'	F	F	T	C	MC

Reglas correspondientes a la salida del segundo actuador (motor):

IF ENG AND ENG' THEN L
 IF ENG AND ENP' THEN ML
 IF ENG AND Z' THEN ML
 IF ENG AND EPP' THEN ML
 IF ENG AND EPG' THEN ML

IF ENP AND ENG' THEN L
 IF ENP AND ENP' THEN L
 IF ENP AND Z' THEN L
 IF ENP AND EPP' THEN ML
 IF ENP AND EPG' THEN ML

IF Z AND ENG' THEN N
 IF Z AND ENP' THEN L
 IF Z AND Z' THEN L
 IF Z AND EPP' THEN ML
 IF Z AND EPG' THEN ML

IF EPP AND ENG' THEN R
 IF EPP AND ENP' THEN N
 IF EPP AND Z' THEN N
 IF EPP AND EPP' THEN L
 IF EPP AND EPG' THEN ML

IF EPG AND ENG' THEN MR
 IF EPG AND ENP' THEN R
 IF EPG AND Z' THEN R
 IF EPG AND EPP' THEN N
 IF EPG AND EPG' THEN L

La FAM correspondiente a estas reglas es la siguiente:

	ENG	ENP	Z	EPP	EPG
EPG'	ML	ML	ML	ML	L
EPP'	ML	ML	ML	L	N
Z	ML	ML	L	N	R
ENP'	ML	L	L	N	R
ENG'	L	L	N	R	MR

Una vez que el sistema realiza la evaluación de reglas, se entregarán salidas difusas, las cuales deberán manipularse de acuerdo al proceso de desdifusión para obtener salidas

nítidas que puedan ser captadas por los actuadores. Dicho proceso se realizará basándose en el método de “singletons”.

El kernel del sistema fue desarrollado dentro del rubro correspondiente al software. De esta manera, en el capítulo 5 se expondrá todo lo relacionado al desarrollo de dicho kernel, indicando el método utilizado para su realización.

CAPITULO 4 HARDWARE

4.1 INTRODUCCIÓN

Actualmente se cuenta con una gran variedad de dispositivos electrónicos con los cuales se pueden desarrollar una gran variedad de sistemas, pero para poder seleccionarlos se necesita, por un lado, conocer las necesidades y alcance del sistema a desarrollar y por otro conocer los diferentes dispositivos electrónicos disponibles. Otro punto importante es el de saber las aplicaciones recomendables de dichos componentes ya que algunos se especializan para aplicaciones destinadas para la investigación y para aplicaciones militares mientras otros son diseñados para trabajar en condiciones que no requieren características tan especiales.

Un aspecto que no se debe perder de vista es el costo beneficio que puede dar un diseño determinado, pues por un lado esta el tamaño físico del sistema contra el precio de los componentes además también se debe tomar en cuenta el número de dispositivos a utilizar y la complejidad del software para llegar al objetivo del sistema.

En la figura 4.1 se muestra un diagrama a bloques del sistema a desarrollar tomando en cuenta el costo beneficio que nos proporciona. En el transcurso del presente capítulo se dará a detalle cada uno de los componentes utilizados.

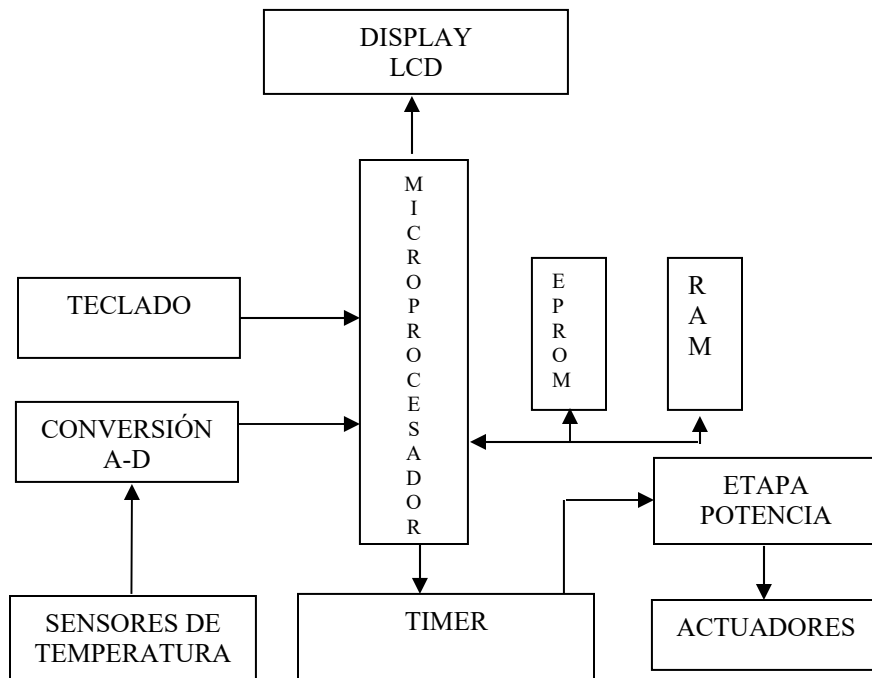
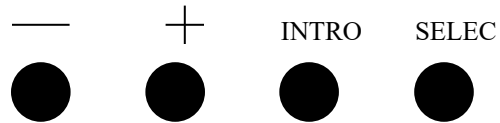


Fig. 4.1 Diagrama a bloques del sistema electrónico.

4.2 TECLADO y LCD

Para poder tener algunas opciones de funcionalidad del sistema de control, se cuenta con una interfaz que permita la comunicación del sistema con el usuario y viceversa, para ello se hará uso de un teclado y una pantalla de cristal liquido (LCD).

El teclado cuenta con 4 teclas las cuales tendrán un funcionamiento particular y se encontrarán dispuestos de la siguiente forma:



La tecla SELEC se encarga de desplegar, en el LCD, las diferentes opciones de funcionalidad que tiene el sistema de control por el LCD. La tecla INTRO habilita alguna de la opción elegida en despliegue. Las otras dos teclas restantes se utilizan al seleccionar el modo programa ya que en este modo se tiene la posibilidad de cambiar variables que intervienen en el proceso de control, la tecla “+” incrementará el valor indicado mientras que la tecla “ - “ lo disminuirá esto se explicará con más detalle en el capítulo siguiente.

Para establecer la comunicación con el teclado es necesario contar con una interfaz de comunicación, esto se logra con un circuito 74HC368, el cual es un buffer inversor con selección. Tal como indica la figura 4.2, tenemos el pin de selección, el cual llamaremos PT y le corresponde el numero de puerto 00H.

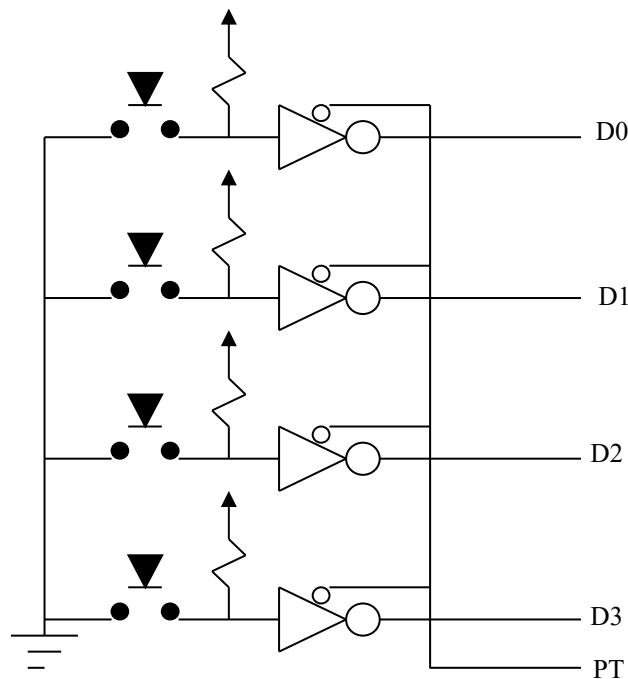


Fig. 4.2 Buffers inversores con selección

un control optimo, es decir, que la frecuencia de corte del filtro paso bajas sea adecuado al sistema y como en este caso las variaciones se dan de manera lenta, se considera un filtro con un ancho de banda $f_c = 10\text{Hz}$.

Así tenemos:

Respuesta del sensor de temperatura (LM35): 10mV por 1°C

Tenemos para 90°C un valor de $990\text{mV} = 0.99\text{V}$

La representación matemática de un amplificador no inversor es:

$$V_o = (1 + R_f / R_i) V_i$$

$$V_o / V_i = 1 + R_f / R_i$$

Sustituyendo los valores deseados:

$$R_f / R_i = 5.05 - 1$$

$$R_f / R_i = 4.05$$

Si $R_f = 33\text{K}\Omega$, entonces $R_i = 8.147\text{K}\Omega$

4.4 ADC

Los convertidores analógicos digital se conectan al microprocesador como se indica en la figura 4.5. en donde su conexión es direccionamiento de memoria, es decir, ya no se ocupan como puertos, ya que para ahorrar circuitería los convertidores analógicos digital se conectan en habilitadores de memoria que corresponden a una dirección dada. Para el convertidor 1 la habilitación será llamada ADC1 y para el convertidor 2 será ADC2. En la figura 4.5 se muestra la conexión de un solo ADC ya que son idénticos en la forma de conectarlos variando solamente en la habilitación.

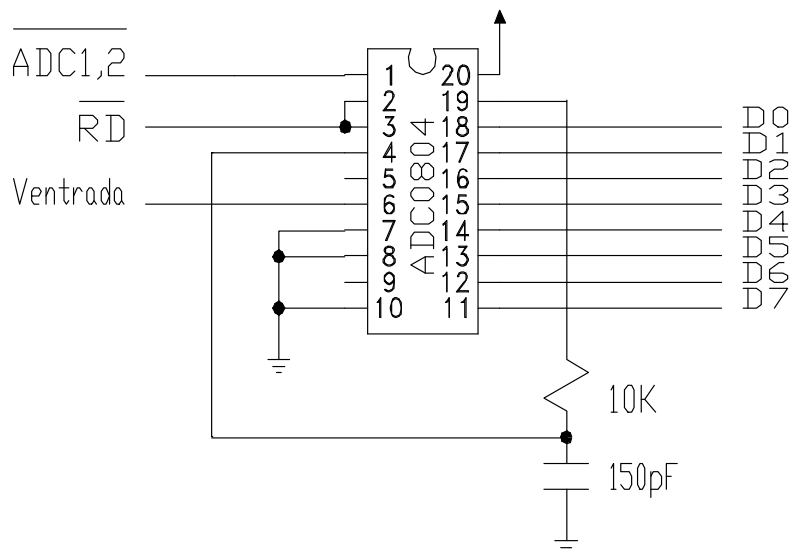


Fig. 4.5 Convertidor analógico Digital

4.5 MICROPROCESADOR

En lo que respecta al microprocesador, como ya se indico anteriormente, se ocupará el Z-80 en un sistema básico, es decir, tendrá una memoria EPROM, una RAM, su generador de reloj, su reset, y decodificadores para la selección de memorias y puertos. En la figura 4.6 se muestra el diagrama del circuito de reloj y de reset. Estos circuitos son de uso típico por lo que su funcionamiento es conocido.

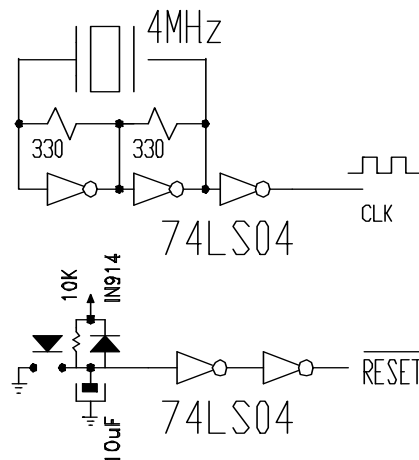


Fig. 4.6 Circuitos de reloj y reset utilizados en el sistema básico

Ahora en la figura 4.7 se ilustra, a nivel de bloques, el sistema básico completo del Z-80.

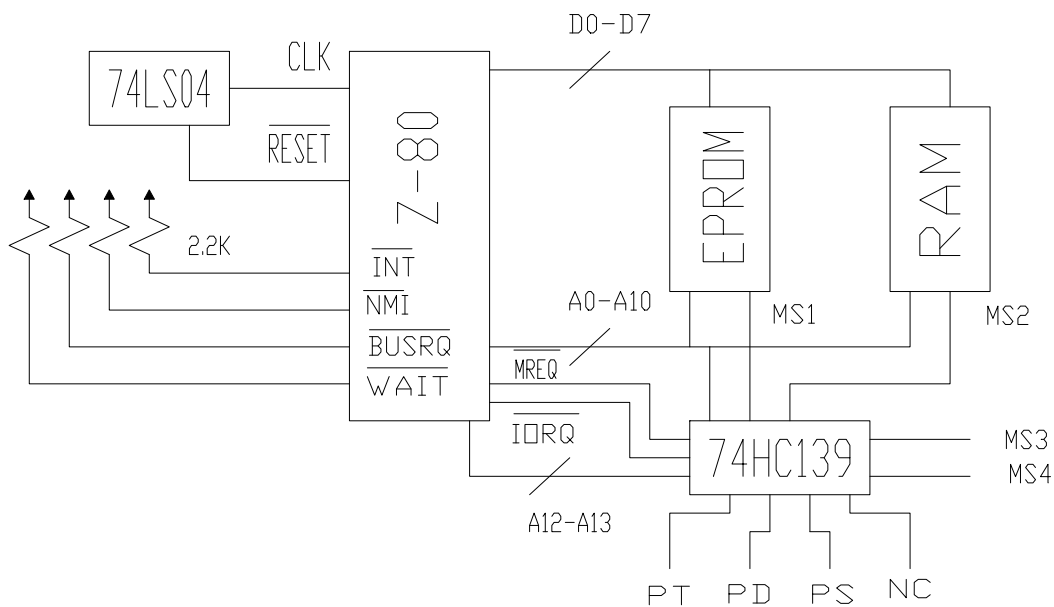


Fig. 4.7 Sistema básico del Z-80

Para la selección de puertos y memoria se requiere de un decodificador, el cual se encarga de discriminar la información hacia un lado u otro.

La forma de conectar el decodificador es de gran importancia ya que desde ese instante se determinan los bloques de memoria y números de puertos. Para esto se determina una memoria de programa de 4Kbytes y una memoria de datos de 2Kbytes, así como los puertos con el número que resulta al manejar A6 y A7. Los bloques de memoria que de manejan son, para EPROM de la \$0000 hasta \$0FFF y para RAM desde la \$1000

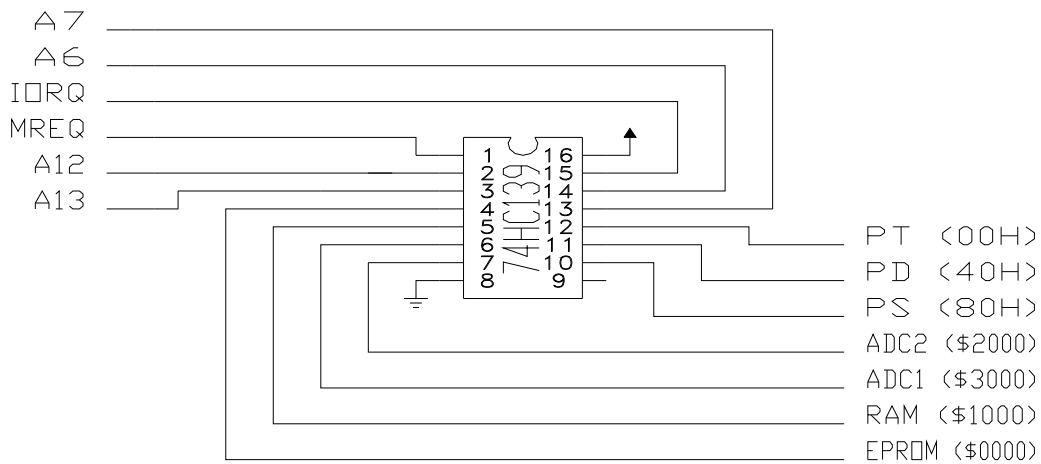


Fig. 4.8 Configuración del decodificador

hasta \$17FF. Para ello se utilizan A12 y A13. La conexión se muestra en la figura 4.8.

4.6 ETAPA DE SALIDA

Por último, se implementa la interfaz de potencia, la cual será la encargada de manejar los actuadores de salida. Los dispositivos que fungirán como tales serán un generador de calor y de un ventilador. Una resistencia será utilizada como generadora de calor, mientras que el ventilador ayudará el proceso de convección forzada y enfriamiento, ambos serán de corriente alterna. Cabe mencionar que el control de salida permitirá manipular la corriente alterna, es decir, su funcionamiento estará basado en la técnica de ángulo de disparo.

La razón por la cual es usada la técnica de ángulo de disparo es por la flexibilidad que presentan las fuentes de corriente alterna en, prácticamente, cualquier lugar, mientras que una corriente continua puede requerir de una mayor estructura que se ve reflejada directamente en los costos del proyecto, además el sentido del mismo retoma un valor más práctico.

Ahora, se considera un TIMER que permitirá manejar los tiempos de operación, es decir, llevará el cálculo de en que momento se deben activar las salidas, ya que se controlan

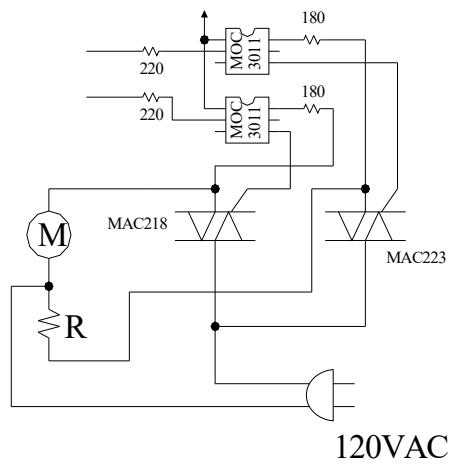


Fig. 4.11 Etapa de alta potencia (salida)

CAPITULO 5 SOFTWARE

5.1 INTRODUCCIÓN

Este capítulo hace referencia al desarrollo del software del sistema. En esta parte se desarrolla la programación del sistema con lógica difusa. La lógica difusa permite utilizar variables lingüísticas para el desarrollo de un sistema de control, del cual se debe tener previo dominio y conocimiento.

Para iniciar con la programación del sistema se consideran dos grandes bloques independientes: el primero será el Kernel de la lógica difusa y el segundo la interfaz con el usuario. Cabe mencionar que estos dos grandes bloques se unirán de tal forma que exista dependencia, pudiendo afectar a las condiciones de trabajo de la lógica difusa a través de la interfaz del usuario.

El diseño del sistema permite exhibir la mayor sencillez, flexibilidad y facilidad de uso, con el fin de tener acceso a varias opciones y parámetros de control.

5.2 EL KERNEL DE LA LÓGICA DIFUSA

Una parte importante del control es el Kernel, el cual es desarrollado bajo los principios de la lógica difusa y consta de las siguientes partes:

- a) La difusión
- b) Evaluación de reglas
- c) La desdifusión

Retomando la figura 3.5 del capítulo 3, podemos recordar que la *base del conocimiento* es la parte donde se definen los conjuntos de entrada, las reglas y los conjuntos de salida. Tanto las reglas como los conjuntos de entrada y de salida son definidas a partir del conocimiento que se tenga del proceso a controlar. Cabe mencionar que el Kernel sólo evalúa a partir de la base del conocimiento, por lo que una vez realizado el mismo y se desee una condición de respuesta diferente del sistema, bastará con cambiar las reglas, los conjuntos de entrada, los conjuntos de salida o bien todos.

El Kernel es el núcleo de la Lógica Difusa, por lo que esta formado por un conjunto de programas, los cuales se encargan de desarrollar la difusión de la entrada nítida (creación de valor difuso), la evaluación de las reglas y la desdifusión (proceso de transformación del resultado difuso a un valor reconocido por el actuador de salida). En el diagrama 5.1 se ilustra la estructura del Kernel.

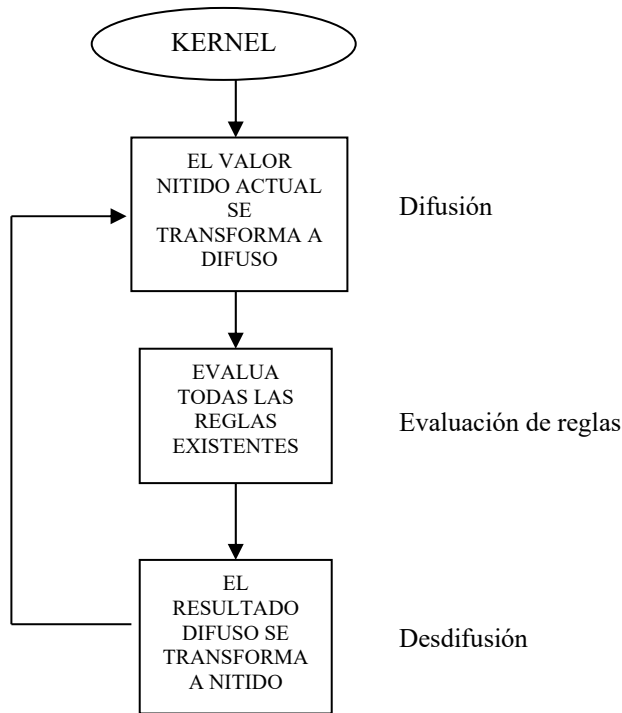


Diagrama 5.1 Descripción General del kernel

5.3 DESARROLLO DEL KERNEL

Como parte de la estructura del sistema, se define una palabra de control que establece 4 aspectos como se muestra en la figura 5.1

FIN	0	S	X	X	Δ	Δ	Δ
------------	----------	----------	----------	----------	----------	----------	----------

Donde

FIN. Si es “1” indica el final de las reglas.

S. Si es “1” indica que es salida, si es “0” es entrada.

XX. Indica la entrada o salida correspondiente (va desde 1 a 4).

ΔΔΔ. Indica la función de membresía en uso (va desde 1 a 8).

Fig. 5.1 Estructura de la palabra de control

La programación de esta parte es descrita posteriormente a detalle, pero cabe mencionar que las reglas se colocan en forma binaria en la memoria, es decir, con base en la palabra de control, se puede establecer a qué variable lingüística corresponde y a través de una operación AND (definida en la teoría de la Lógica Difusa) que se realiza entre ellas, se define el valor de la variable lingüística de salida.

5.3.1 MÉTODO DE DIFUSIÓN

El método utilizado para el proceso de difusión es el *punto pendiente*. Para ello es necesario contar con las pendiente ya calculadas de los conjuntos; para facilitar este proceso el sistema calcula las pendientes a partir de cuatro puntos por lo que se pueden manejar conjuntos triangulares y trapezoidales (ver figura 5.2). El cálculo de las pendientes se realiza al iniciar el sistema o bien al cambiar algún parámetro de los conjuntos.

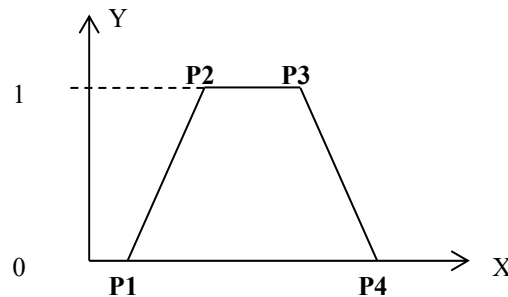


Fig. 5.2 Definición de Puntos

5.3.1.1 CALCULO DE LAS PENDIENTES DE LOS CONJUNTOS

Una vez definidos los cuatro puntos de los conjuntos difusos, se desarrolla el programa que permite realizar el cálculo de las pendientes en forma automática. Inicialmente, se deberán desarrollar algunas operaciones aritméticas importantes, como son la multiplicación y la división, ya que son herramientas que no están implementadas en el set de instrucciones del microprocesador Z – 80.

Multiplicación

El programa de multiplicación se desarrolla de la siguiente manera: Primeramente se verifica el multiplicador bit a bit de derecha a izquierda, si el valor de dicho bit es “1”, se suma el multiplicando al resultado acumulado (estado inicial = cero), se rota el resultado una posición a la derecha; en caso contrario, si el valor del bit en curso es “0”, solamente se rotará el resultado una posición a la derecha procediendo a verificar el siguiente bit del multiplicador. El diagrama de flujo de este proceso se ilustra en el diagrama 5.2

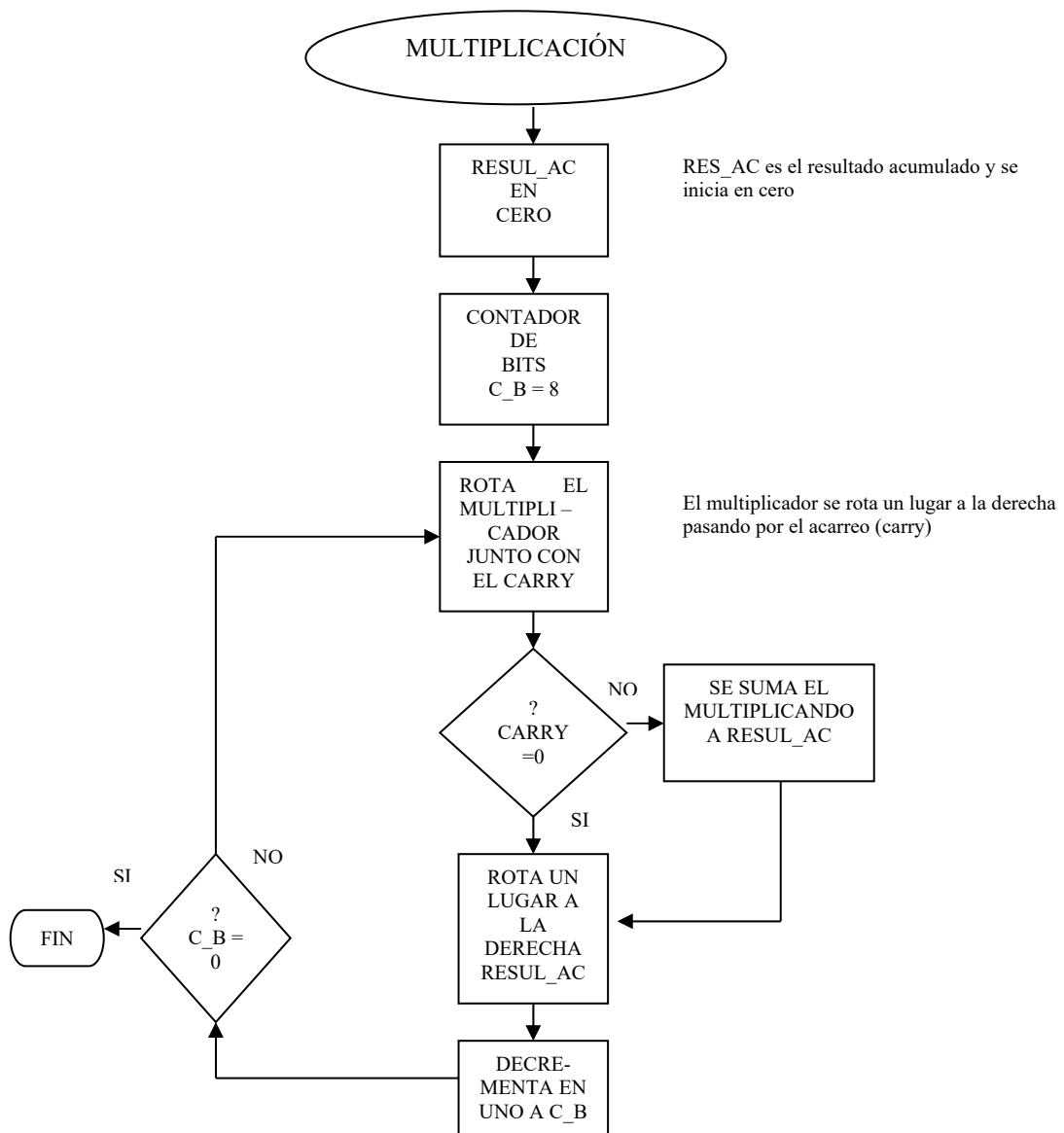


Diagrama 5.2 Descripción de la multiplicación

Con base en el diagrama de flujo 5.2 se desarrolla la decodificación de la multiplicación; el listado 5.1 expone el ensamblador para esta operación con sus respectivos comentarios.

```

;*****
;PROGRAMA QUE EFECTUA UNA MULTIPLICACION DE 8BITS X 8BITS Y UN RESULTADO
;*****DE 16BITS *****
MULTI:   PUSH AF           ;Se guarda el contenido de AF
         PUSH BC
         PUSH DE
         PUSH HL
         LD DE,00H        ;El resultado acumulado es igual a cero
         LD A,(IX+VAR1L)  ;Lugar del multiplicando
         CP 0H            ;Compara si no es cero, si lo es
         JR Z,MULRESUL    ;el resultado es cero
         LD C,A           ;Multiplicando -> C
         LD A,(IX+VAR2L)  ;Lugar del multiplicador
         CP 0H            ;Compara si no es cero, si lo es
         JR Z,MULRESUL    ;el resultado es cero
         LD H,A           ;Multiplicador -> H
         LD B,8H          ;C_B = 8 (contador de bit)
MULT1:   RR H              ;Rota multiplicador a la derecha con carry
         JR NC,MULNOSUM   ;Checa el bit en curso
         LD A,D           ;El bit en curso es "1" y se sumar
         ADD A,C           ;al resultado acumulado
         LD D,A
MULNOSUM: RR D             ;Rota el resultado acumulado
         RR E
         DJNZ MUL1        ;Se realizó para los ocho bits?
MULRESUL: LD (IX+VAR3L),E ;EL RESULTADO ACUMULADO ESTA EN DE
         LD (IX+VAR3H),D ;Y SE PASA A UN LUGAR DE MEMORIA
         POP HL           ;Regresa los valores de los registros
         POP DE
         POP BC
         POP AF           ;B,C,D,E,H,L,A,F
         RET              ;Regresa de la multiplicación

```

Listado 5.1 Código de la Multiplicación

División

La división se desarrolla de la siguiente manera: Consiste en introducir el divisor, iniciando por el bit menos significativo, bit a bit, por la izquierda a un registro temporal. Dicho registro es restado al dividendo; si se genera un acarreo, el registro temporal es sumado al resultado de la resta anterior, obteniendo así el minuendo original.

El complemento del acarreo será el primer bit de la estructura del cociente, el cual es guardado en un registro, bit a bit con corrimiento a la izquierda. De esta manera, el primer bit que se origina de la primera substracción será el menos significativo del cociente.

Si no existe acarreo (bit de acarreo = 0), de igual forma se complementa y se introduce ese bit en el registro correspondiente al cociente.

Se continúa con la siguiente sustracción. El número de iteraciones a realizar depende directamente del número de corrimientos del divisor en el temporal. Dichas iteraciones continuarán hasta finalizar los corrimientos del divisor en el registro temporal.

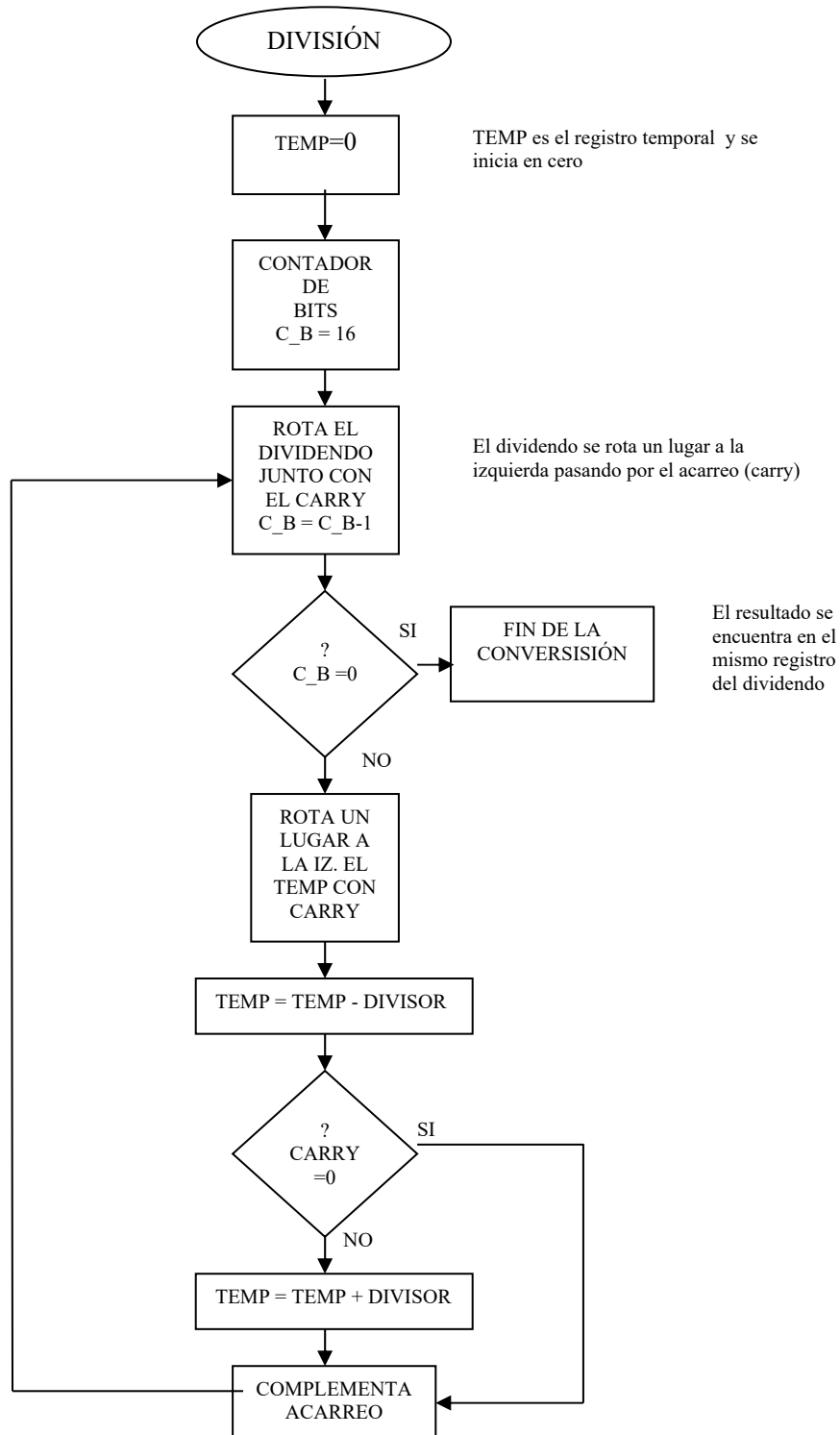


Diagrama 5.3 Descripción de la división

La estructura de la división se ilustra en el diagrama de flujo 5.3 y con base en él, se desarrolla el lenguaje ensamblador para la ejecución de la misma. Véase el listado 5.2.

```

;*****
;***PROGRAMA QUE EFECTUA UNA DIVISION ENTERA DE 16BITS X 16BITS Y UN ****
;*****RESULTADO DE 16 BITS *****
DIV:      PUSH AF
          PUSH BC
          PUSH DE
          PUSH HL
          LD HL,00H           ;Usados para guardar el resultado parcial
          LD B,11H           ;Numero de bits m s uno (16 bits + 1)
          LD E,(IX+VAR2L)    ;Obtiene el valor del dividendo
          LD D,(IX+VAR2H)
DIV1:     RL E                ;Rota el valor bajo del dividendo con carry
          RL D                ;Rota el valor alto del dividendo con carry
          DJNZ DIV2          ;? fin de bits
          LD (IX+VAR3L),E    ;Guarda el valor final a
          LD (IX+VAR3H),D    ;la variable VAR3
          POP HL             ;Regresa valores principales
          POP DE
          POP BC
          POP AF
          RET                ;Retorna de la subrutina
DIV2:     RL L                ;Rota el valor del dividendo parcial
          RL H
          LD A,L             ;Recupera el valor del dividendo parcial
          SUB (IX+VAR1L)     ;y réstalo al divisor
          LD L,A
          LD A,H
          SBC A,(IX+VAR1H)
          LD H,A
          JR NC,DIVNOSUM    ;? existe acarreo
          LD A,L             ;Si hay acarreo por lo que
          ADD A,(IX+VAR1L)  ;se regresa al valor inicial
          LD L,A
          LD A,H
          ADC A,(IX+VAR1H)
          LD H,A
DIVNOSUM: CCF                ;Complementa el acarreo
          JR DIV1           ;y repite para el siguiente bit

```

Listado 5.2 Código de la división

Una vez que las operaciones básicas han sido programadas, se procede a desarrollar el cálculo de las pendientes. Para ello, se manejan bloques, es decir, considérese el uso de direccionamiento indexado, así como el uso de pares de registros del Z-80.

Inicialmente se definen las localidades donde serán almacenadas las pendientes calculadas y los puntos de los conjuntos; de esta manera, se establece la **C1M1E1**, la cual queda definida como: Conjunto uno, pendiente uno y entrada uno. De igual forma, para los

puntos se tiene la **C1P1E1** definida como: Conjunto uno, punto uno y entrada uno. Representando estas localidades en un mapa de memoria quedaría como se ilustra en la figura 5.3.

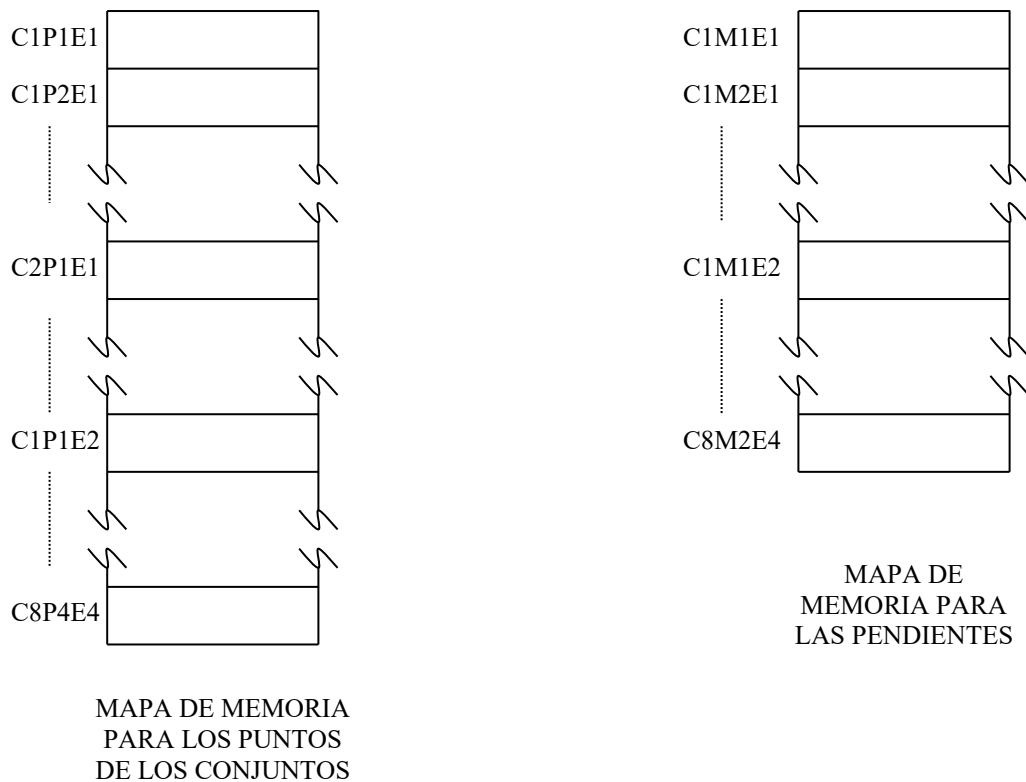


Fig. 5.3 Mapas de memoria de pendientes y puntos de los conjuntos difusos

Se desarrolla el algoritmo que consiste en un ciclo que repetirá el cálculo de las pendientes de ambos lados de un conjunto, es decir, $M_1 = (P_2 - P_1)/FFH$ y $M_2 = (P_4 - P_3)/FFH$. Considerando dos entradas, con una representación de cinco conjuntos por entrada y dos pendientes por conjunto, el ciclo se repetirá 20 veces en decimal o bien 14H en hexadecimal. El diagrama de flujo 5.4 ilustra el desarrollo del algoritmo correspondiente. Cabe mencionar que cuando $P_1 = P_2$ o $P_3 = P_4$, donde el valor es FFH en $\mu(x)$, el valor de la pendiente será también FFH, esto para facilitar el cálculo al evaluar $\mu(x)$.

Control de temperatura con lógica difusa

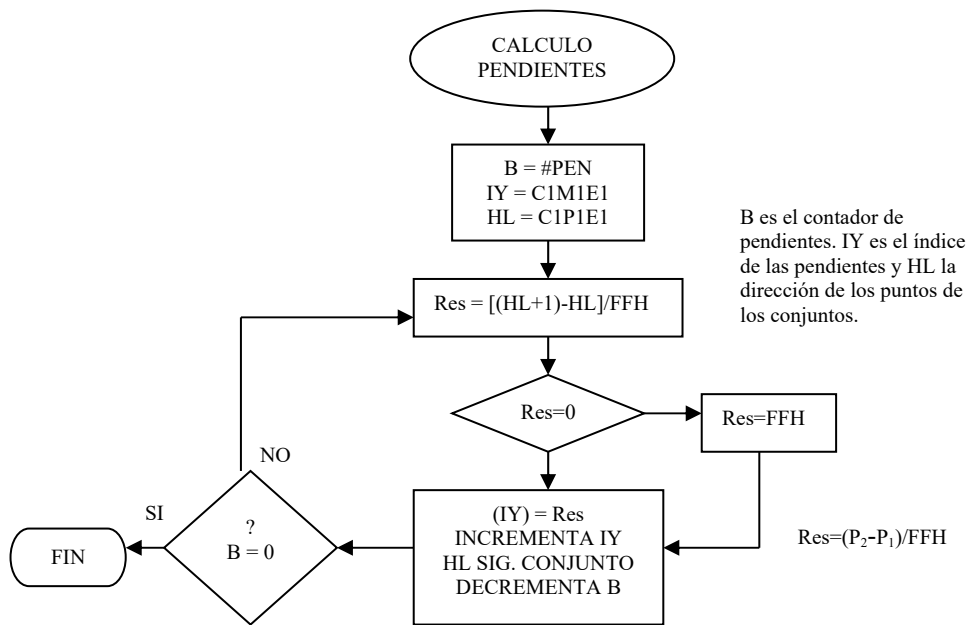


Diagrama 5.4 Diagrama de flujo para el cálculo de las pendientes

Con base en el diagrama anterior, se desarrolla el ensamblador que se ilustra en el listado 5.3.

```

;***** PROGRAMA QUE CALCULA LAS PENDIENTES DE LOS CONJUNTOS *****

PENDIEN: LD A,(NUMENT)      ;Calcula el número de pendientes que
RLCA      ;esta dado por el número de entradas
LD (IX+VAR1L),A      ;y número de conjuntos.
LD A,(NUMCON)      ;pendientes = conjuntos*2*entradas
LD (IX+VAR2L),A
CALL MULTI
LD B,(IX+VAR3L)      ;# de pendientes -> B
LD IY,C1M1E1      ;Lugar donde estarán las pendientes
LD HL,C1P1E1+1      ;Lugar de los puntos de los conjuntos
LD (IX+VAR2L),0FFH ;Inicia los valores de la división
LD (IX+VAR2H),00H
LD (IX+VAR1H),00H

LOOP1: LD A,(HL)      ; A = P2 & P4
DEC HL      ; HL se coloca en el punto anterior
SUB (HL)      ;VALOR DEL DIVIDENDO EN A
LD (IX+VAR1L),A ;Pasa el valor a la división
CALL DIV
LD A,(IX+VAR3L) ; A = (P2 - P1)/FF & A = (P4 - P3)/FF
CP 0H      ;Compara si los puntos son iguales el
JR NZ,PENOCERO ;resultado será FFH
LD A,0FFH

PENOCERO: LD (IY+0),A ;Se coloca el resultado donde corresponde
INC HL      ;-----
INC HL      ;Se coloca en el sig. conjunto
INC HL      ;-----
INC IY      ;Siguiente lugar de pendiente
DJNZ LOOP1 ;? están todas las pendientes
  
```

```
RET ; Si -> retorna
```

Listado 5.3 Código para cálculo de pendientes

El método *punto pendiente* que obtiene el valor difuso correspondiente al valor nítido de entrada se describe a continuación:

- a) se calculan dos incrementos :

$$\Delta_1 = \text{valor nítido} - P_1 \quad \text{y} \quad \Delta_2 = P_2 - \text{valor nítido}$$

Si cualquiera de los dos incrementos es negativo, entonces el valor nítido estará fuera del conjunto, por lo tanto el valor difuso (μ) será igual a cero.

- b) Si los incrementos no son negativos se procede a calcular el producto con las pendientes:

$$\text{Producto 1} = \Delta_1 \cdot S_1$$

$$\text{Producto 2} = \Delta_2 \cdot S_2$$

Donde, S_1 es la pendiente 1 y S_2 es la pendiente 2

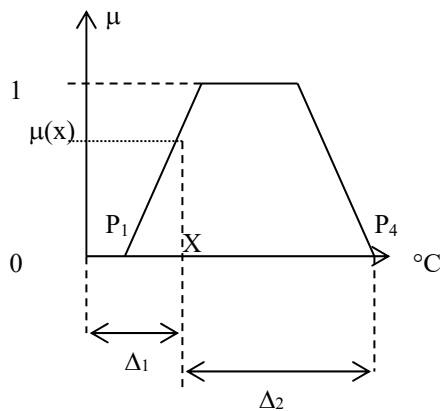
- c) Una vez teniendo los productos se obtiene el mínimo, el cual será el resultado difuso:

$$\mu = \min(\text{Producto 1}, \text{Producto 2})$$

Este proceso se repite para cada uno de los conjuntos de entrada. Al terminar, se tendrán los valores difusos para la entrada considerada. Dado que el sistema es de dos entradas, lo anterior se realiza para ambas. Al terminar con este bloque se desarrolla la evaluación de reglas utilizando la palabra de control descrita en la figura 5.1

Conversión del valor nítido a difuso

Para obtener un valor difuso a partir del valor nítido se consideran los puntos de los conjuntos y las pendientes. En la figura 5.4 se presenta la gráfica de un conjunto difuso con sus correspondientes puntos y pendientes, dicha gráfica ayuda a representar el proceso de conversión.



Donde:

$$\Delta_1 = (x - P_1)$$

$$\Delta_2 = (P_4 - x)$$

$$\mu(x) = \min(\Delta_1 \bullet M_1, \Delta_2 \bullet M_2, FF)$$

Fig. 5.4 Representación de puntos y pendientes de un conjunto difuso

El proceso de conversión es llamado punto pendiente y consiste en calcular los valores de los incrementos Δ_1 y Δ_2 . Si algún valor de incremento resultara negativo esto implica que el valor difuso será nulo ($\mu = 0$), ya que dicho valor de x no corresponde al conjunto en cuestión. Caso contrario, si los incrementos son positivos se obtienen los productos que resultan de multiplicar a los mismos con sus respectivas pendientes, es decir, al Δ_1 obtenido a partir del P_1 (punto uno), le corresponde la M_1 y al Δ_2 la M_2 (ver figura 5.4). El valor difuso $\mu(x)$ será el correspondiente al mínimo de los productos obtenidos.

$$PROD_1 = \Delta_1 M_1$$

$$PROD_2 = \Delta_2 M_2$$

$$\mu(x) = \min [PROD_1, PROD_2, FF]$$

Por otro lado, esta conversión se realiza en un ciclo, ya que es desarrollada para todas las pendientes de todos los conjuntos. El programa se divide en dos partes: La primera describe un ciclo para todas las pendientes de todos los conjuntos y ambas entradas, la segunda sólo describe cómo hacer el cálculo del valor difuso.

Este proceso se ilustra en los diagramas 5.5 a) y b).

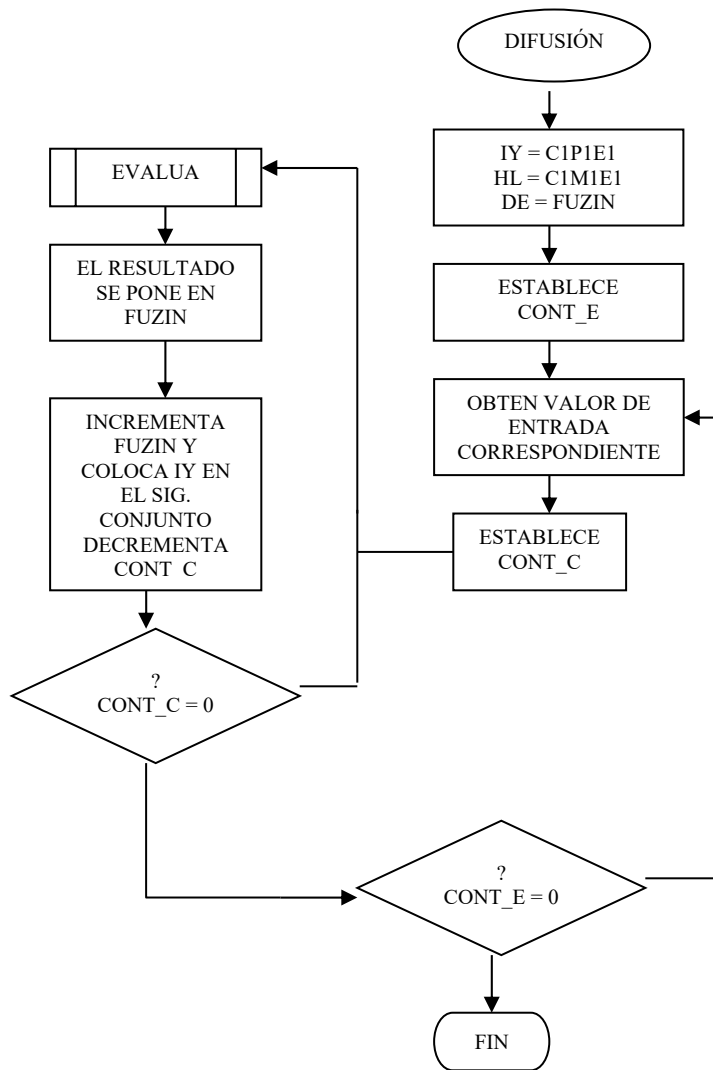
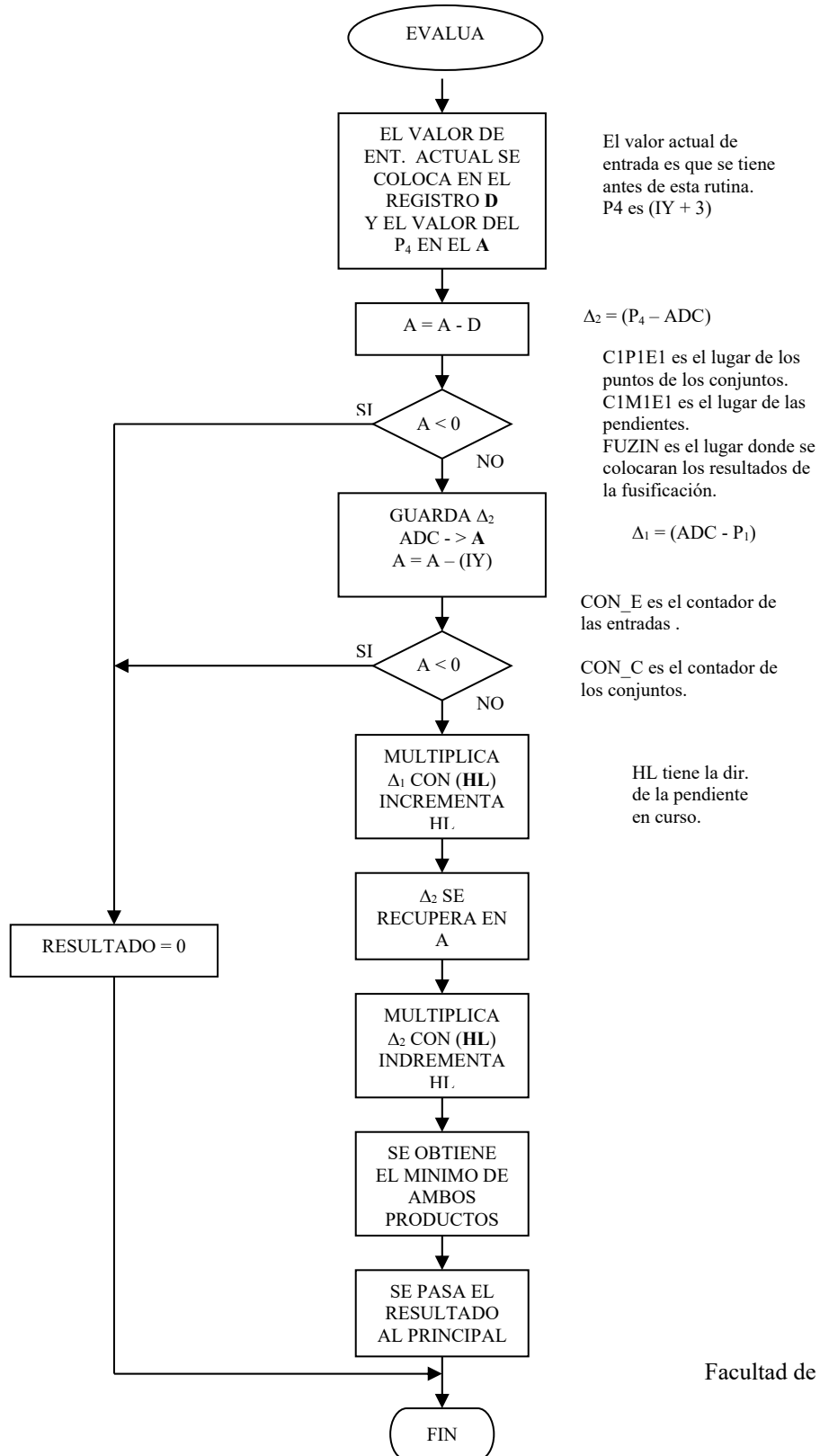


Diagrama 5.5 a) Descripción de la primera parte de conversión (ciclo)



Una vez descrito el método de conversión con sus respectivos diagramas de flujo, se exponen los listados de ensamblador de ambos algoritmos con sus respectivos comentarios.

;***** PROGRAMA QUE EFECTUA LA FUSIFICACIÓN *****

```

FUZY:      PUSH AF          ;-----
           PUSH BC        ; Retiene valores
           PUSH DE        ; anteriores
           PUSH HL        ;-----
           LD IY,C1P1E1   ;PUNTO 1 DEL CONJUNTO 1 Y ENTRADA 1
           LD HL,C1M1E1   ;Lugar de las pendientes calculadas
           LD DE,FUZIN    ;Lugar en donde se pone el valor difuso
           EXX
           LD HL,BLKENT   ;HL' Apuntará hacia el BLOQUE DE ENTRADAS
           EXX
           LD A,(NUMENT)
           LD C,A         ; # de entradas
FUZYB:     EXX
           LD A,(HL)
           INC HL
           EXX
           LD (IX+GEN1),A ;Variable que utiliza valúa para calcular
           LD A,(NUMCON)
           LD B,A         ;# de conjuntos
FUZYVAL:   CALL EVALUA    ;Función que realiza el calculo
           LD (DE),A      ;En A se encuentra el Min(P1,P2,FF)
           INC DE         ;lo pone en FUZIN e incrementa el lugar
           INC IY        ;-----
           INC IY        ;IY se ubica en el conjunto
           INC IY        ;siguiente
           INC IY        ;-----
           DJNZ FUZYVAL  ;Checa si ya se realizo para los conjuntos
           LD A,(NUMCON)
FUZYCPA:   CP 8          ;Se posiciona en el lugar correcto de
           JR Z,FUZYPOS  ;los resultados en FUZIN
           INC A
           INC DE
           JR FUZYCPA
FUZYPOS:   DEC C         ;Checa si ya se realizo para las entradas
           JR NZ,FUZYB
           POP HL        ;-----
           POP DE        ;Recupera los valores anteriores
           POP BC        ;de los registros

```

Control de temperatura con lógica difusa

```

POP AF          ;-----
RET             ;Regresa

;***** FUNCION EVALUA *****

EVALUA:        PUSH BC
                PUSH DE
                LD D,(IX+GEN1)      ;Recupera valor de la entrada
                LD A,(IY+P4)        ;Se coloca en el punto 4
                SUB D                 ; A = P4 - ADC
                JR C,VALNEG          ; Si es negativo Termina
                LD E,A               ; A -> E
                LD A,D
                SUB (IY+0)           ; A = ADC - P1
                JR C,VALNEG          ; Si es negativo Termina
                LD B,02H             ;Por ser dos pendientes por conjunto
                LD (IX+VAR1H),00H    ;Variables de la mult. sin utilizar
                LD (IX+VAR2H),00H    ;Variables de la mult. sin utilizar
VAL2:          LD (IX+VAR1L),A        ;Valor del incremento
                LD A,(HL)            ;Valor de la pendiente
                LD (IX+VAR2L),A
                CALL MULTI            ;Realiza la multiplicación
                LD A,(IX+VAR3H)      ;-----
                CP 0                  ;Verifica si el resultado es > FFH
                JR Z, VAL0            ;-----
                LD (IX+VAR3L),0FFH   ;Si es >FFH coloca FFH
VAL0:          LD C,(IX+VAR3L)        ;Resultado -> C
                INC HL                ;Sig. pendiente
                LD A,E                ;E -> A
                LD E,C                ;C -> E
                DJNZ VAL2             ;? para las dos pendientes
                LD B,A                ;*****
                SUB E                  ;           Obtiene el valor
                JR NC,MENE            ;           mínimo de los
                LD A,B                ;           productos de los
                JR VALFIN              ;           incrementos y
MENE:          LD A,E                  ;           pendientes
                JR VALFIN              ;*****
VALNEG:        LD A,00H                ; Valor cero si in incremento es (-)
                INC HL
                INC HL
VALFIN:        POP DE
                POP BC
                RET

```

Listado 5.4 Difusión

Al tener los valores difusos establecidos y calculados, se procede a evaluar las reglas. De esta manera, se tendrán los valores correspondientes de salida. Cabe mencionar que el valor de las salidas puede variar respecto a la forma de desdifusión, para este sistema se considera el método de *Centro de Máximos*.

5.3.2 EVALUACIÓN DE REGLAS

Una vez que se tienen los valores difusos en un bloque de memoria establecido, denominado **FUZIN**, se procede a operar y evaluar a las correspondientes reglas, respetando el formato indicado en la figura 5.1.

Retomando la figura 5.1, se considera que el bit 7 de la palabra de control corresponde a la etiqueta **FIN**, el cual es el bit que indica la continuación de la evaluación de las reglas o el final de ellas. Para ello, dicha etiqueta obtiene un valor lógico “1” cuando se presente el final de la evaluación y tendrá el valor “0” lógico para reglas válidas, es decir, continuará evaluando indefinidamente si el valor del bit 7 es “0”.

Con lo que respecta a **S** (bit 5), indica si es el antecedente (valor lógico “0”) o consecuente (valor lógico “1”), según el valor que tome corresponderá la función a realizar.

El valor de **X X** (bit 4 y 3) indica el número de antecedente o consecuente tal como se muestra en la tabla 5.1. Para establecer si es consecuente o antecedente se tiene que verificar el valor de **S**.

Bit 4 X	Bit 3 X	Consecuente o antecedente
0	0	1
0	1	2
1	0	3
1	1	4

Tabla 5.1 Número de antecedentes o consecuentes

Por último, los tres primeros bits (bit 0, bit 1 y bit 2) indican la función de membresía correspondiente, Esta palabra de control es capaz de contener máximo ocho funciones de membresía, para el sistema que se desarrolla se trataran solo cinco. En la tabla 5.2 se muestran los números de las membresías.

Bit 2 Δ	Bit 1 Δ	Bit 0 Δ	Número de membresía
0	0	0	1
0	0	1	2
0	1	0	3
0	1	1	4
1	0	0	5
1	0	1	6
1	1	0	7
1	1	1	8

Tabla 5.2 Número de funciones de membresía

Como ejemplo, se desarrollan tres reglas, para ello considérese que el sistema está compuesto por dos entradas y dos salidas. La primera entrada corresponde a la temperatura en el interior del sistema, la segunda entrada proporciona la lectura de la temperatura ambiente en el exterior del sistema. Las salidas corresponden al ciclo de trabajo de la resistencia de calentamiento y del ventilador. Cabe mencionar que al término del procesamiento de una salida, el sistema comienza con una nueva regla.

Para ejemplificar, considérese las siguientes tres reglas:

- Si la temperatura externa esta fría y la temperatura interna esta caliente entonces la resistencia fría y ventilador rápido.
- Si la temperatura externa esta caliente y la temperatura interna esta fría entonces la resistencia caliente y ventilador rápido.
- Si la temperatura externa esta muy fría y la temperatura interna esta muy fría entonces la resistencia muy caliente y ventilador muy lento.

Estas variables lingüísticas pueden ser reducidas a símbolos más sencillos, es decir, sintetizar las variables como:

MC = (muy caliente)
C = (caliente)
N = (normal)
F = (frío)
MF = (muy frío)

Para una mejor representación, la variable **E** representará una entrada, mientras que la variable **S** una salida; cada variable con su número de identificación correspondiente, por ejemplo:

E1 => Entrada 1
S2 => Salida 2

Para representar la salida del ventilador, se tiene:

ML = (muy lento)
L = (lento)
N = (normal)
R = (rápido)
MR = (muy rápido)

Asimismo, la conjunción “y” se puede reemplazar por **AND** y el “entonces” por **THEN**.

Cabe mencionar que la evaluación de reglas consiste, precisamente, de una operación **AND** entre ellas. Al presentarse un consecuente o respuesta a una salida, se inicia una nueva regla.

Representando las reglas anteriores con la simbolización definida, se tiene lo siguiente:

Si E2F AND E1C THEN S2F AND S1R

Si E2C AND E1F THEN S2C AND S1R

Si E2MF AND E1MF THEN S2MC AND S1ML

Así, se inicia con el análisis de la primera regla. Esta indica que se trata de la entrada o el antecedente 2, considerando la función de membresía 4 y no es final de reglas. De esta manera, se define el valor en hexadecimal a partir de la palabra de control.

FIN = 0 S = 0 X X= 0 1

Con una representación binaria, se tiene:

0	0	0	0	1	0	1	1
---	---	---	---	---	---	---	---

y su representación en hexadecimal:

0BH

Al continuar con la evaluación de reglas, se obtiene una tabla conformada por valores hexadecimales, que representan a cada una de ellas.

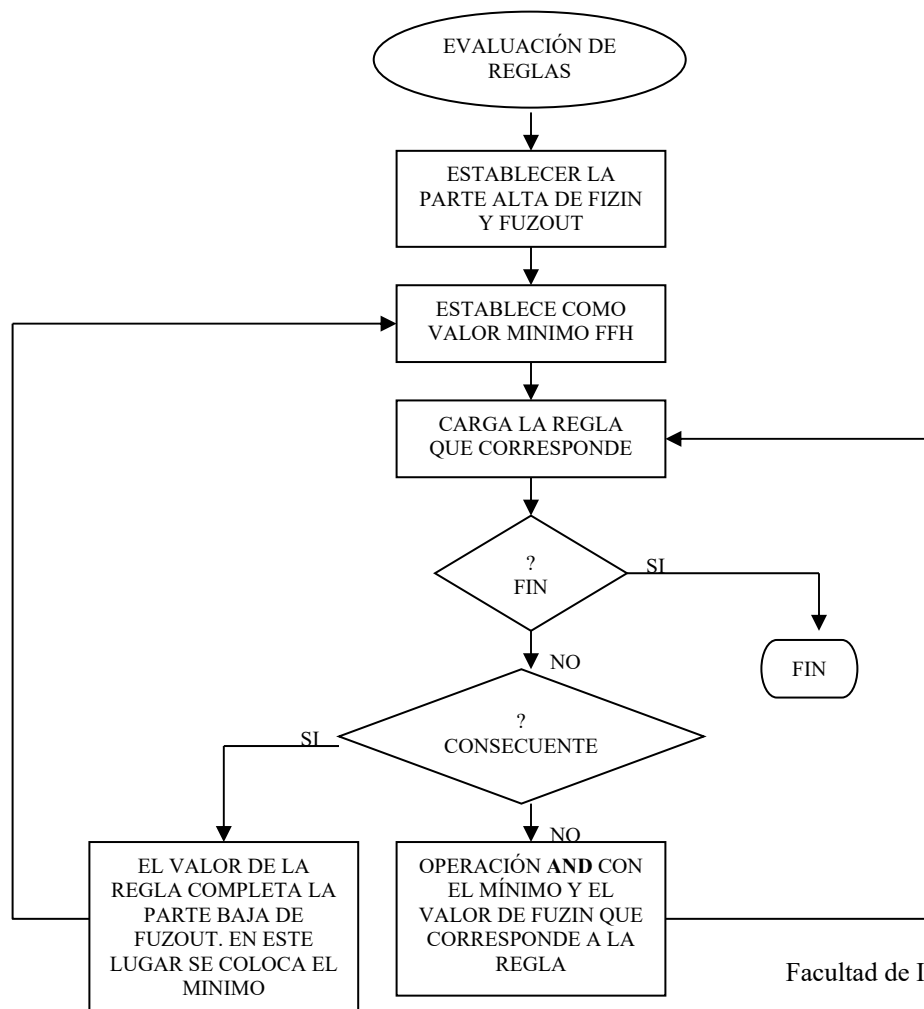
0BH
01H Si E2F AND E1C THEN S2F AND S1R
2BH
23H

09H
03H Si E2C AND E1F THEN S2C AND S1R
29H
23H

0CH

04H Si E2MF AND E1MF THEN S2MC AND S1ML
 28H
 20H

De esta manera, la evaluación de reglas consiste en un ciclo cuyo fin será cuando encuentre el código FFH (basta con que el bit 7 sea "1") en la tabla de las reglas, ya que para fines del sistema representará el FIN. En el diagrama de flujo presentado el diagrama 5.6 se ilustra la lógica que desarrolla el programa de evaluación de reglas.



Con base en el diagrama 5.6, se desarrolla el ensamblador para la evaluación de las reglas, con sus respectivos comentarios. Ver listado 5.5.

```

;***** FUNCION QUE EVALUA REGLAS *****
EREGLAS:  PUSH AF
          PUSH BC
          PUSH DE
          PUSH HL
          LD H,FUZH           ;Lugar de FUZIN y FUZOUT (Parte alta)
          LD DE,REGLAS       ;Dirección donde est n las reglas
RSAL:     LD B,0FFH         ;Establece como mínimo el valor FFH
RSIG2:    LD A,(DE)         ;Carga la regla en curso
          BIT 7,A           ;? fin de las reglas
          JR Z,RSIG         ;NO continua con la evaluación
          POP HL            ;SI termina
          POP DE
          POP BC
          POP AF
          RET
RSIG:     INC DE           ;Se posiciona a la siguiente regla
          LD L,A           ;Se completa la dirección (HL)
          BIT 5,A          ;? antecedente o consecuente
          JR Z,RSAL        ;Salta si es antecedente
          LD (HL),C        ;Consecuente -> se pone el resultado
          JR RSAL          ;Siguiente regla
RB50:    LD A,(HL)         ;Se realiza operación AND
          SUB B            ;La operación AND en la lógica difusa es el
          JR NC,RNG        ;obtener el mínimo y esto lo realiza
          LD B,(HL)        ;este bloque
RNG:     LD C,B
          JR RSIG2
    
```

Listado 5.5 Ensamblador para la evaluación de reglas

Hasta este momento se tienen los resultados a partir de la evaluación de reglas, los cuales se encuentran localizados en **FUZOUT** y son cinco valores por salida en este caso.

5.3.3 MÉTODO DE DESDIFUSIÓN

El método de desdifusión en este caso es el de Centro de Máximos que está dado por la ecuación 5.1.

$$Z^* = \frac{\sum \mu(z)_i \cdot z_i}{\sum \mu(z)_i} \text{-----(5.1)}$$

Con base en la ecuación 5.1 puede definirse el algoritmo correspondiente y se construirá a partir de un ciclo que realice las multiplicaciones y las divisiones, así como la

acumulación de los resultados, recordemos que los conjuntos de salida son del tipo SINGLETONS. En el diagrama de flujo 5.7 se observa la secuencia de dicho programa.

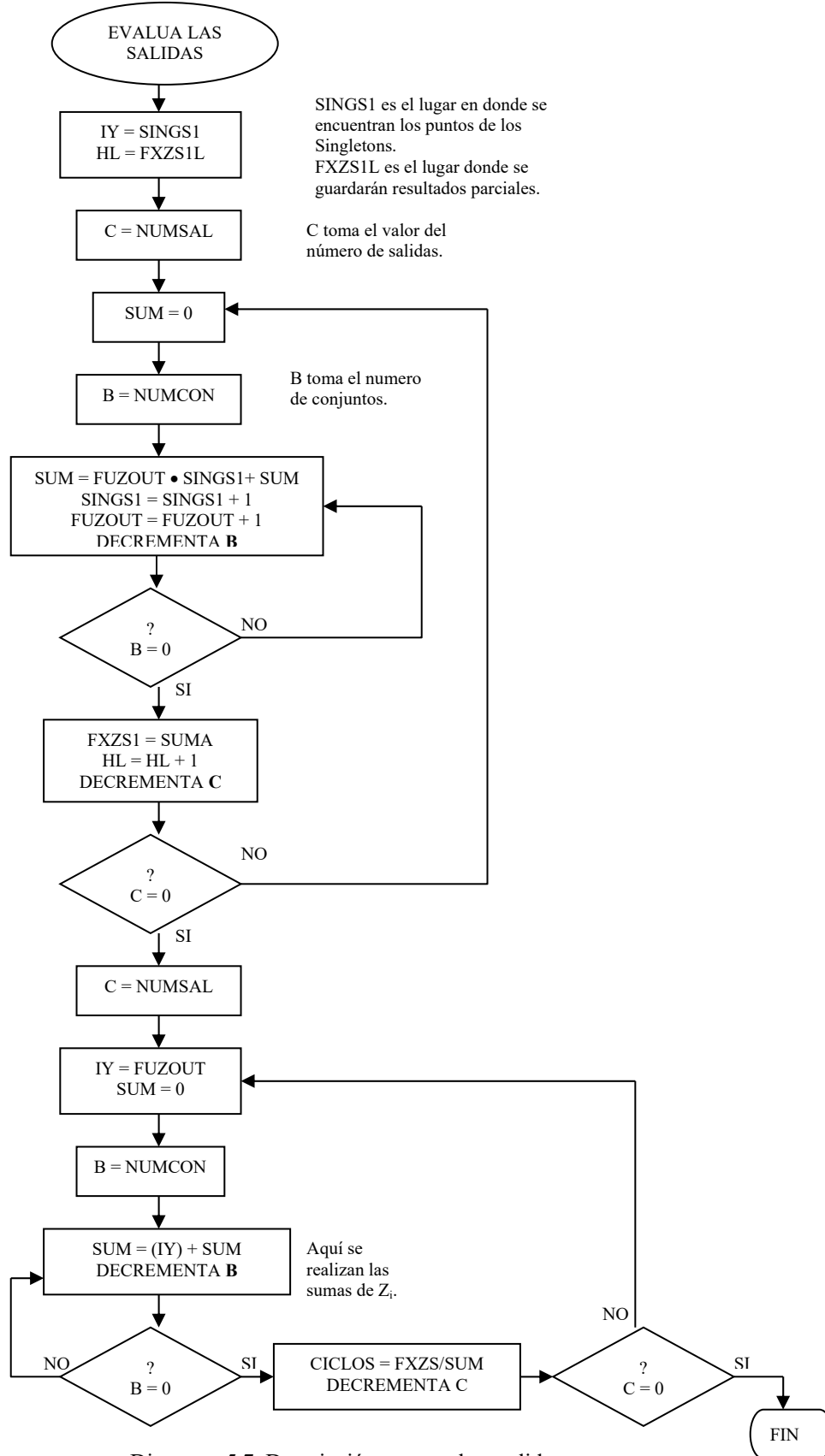


Diagrama 5.7 Descripción para evaluar salidas

De esta manera, los resultados finales quedan en un bloque de 4 registros, uno para cada salida. Estos registros se utilizarán en otra rutina, la cual se encargara de generar el ángulo (α) de disparo para controlar la potencia de salida. En el listado 5.6 se muestra el código para el cálculo de la salida de los dispositivos de AC: resistencia y motor.

;***** FUNCION QUE EVALUA LA SALIDA *****

```

SALIDA:    PUSH AF
           PUSH BC
           PUSH DE
           PUSH HL
           LD IY,SINGS1      ;Lugar de los SINGLETON de salida
           LD HL,FUZOUT      ;Lugar donde se colocara FUZOUT
           EXX               ;En HL' se tendrá F(X)Z
           LD HL,FXZS1L
           EXX
           LD A,(NUMSAL)
           LD C,A            ;# de salidas
SALC0:    LD DE,00H         ;Inicia los registros para la suma
           LD A,(NUMCON)
           LD B,A            ;# de conjuntos
SALBN0:   LD A,(IY+0)       ;Apunta al lugar de Singletons
           LD (IX+VAR1L),A
           LD A,(HL)        ;Apunta al lugar de FUZOUT
           LD (IX+VAR2L),A
           CALL MULTI       ;SINGSij*FUZOUTij
           LD A,(IX+VAR3L)
           ADD A,E
           LD E,A
           LD A,(IX+VAR3H)
           ADC A,D
           LD D,A           ;DE=SINGSij*FUZOUTij + DE
           INC IY
           INC HL
           DJNZ SALBN0
           LD A,(NUMCON)
SALCPA:   CP 8              ;Se posiciona en el lugar correcto de
           JR Z,SALPOS      ;FUZOUT
           INC A
           INC HL
           JR SALCPA
SALPOS:   LD A,E            ;Pone los resultados parciales a
           EXX              ;partir de FXZS1L
           LD (HL),A
           INC HL
           EXX
           LD A,D
           EXX
           LD (HL),A
           INC HL
           EXX

```

Control de temperatura con lógica difusa

```
DEC C ;Checa si se realizo para todas las salidas
JR NZ,SALCO

LD IY,FUZOUT ;Hasta aquí se tiene el numerador y a
LD HL,FXZS1L ;continuación se realiza la suma
EXX ;del bloque para el denominador
LD HL,CICLOS1 ;Aquí comienza SUMATORIA DE Zi
EXX
LD A,(NUMSAL)
LD C,A ;Se coloca el número de salida
SALB: LD A,(NUMCON) ;# de conjuntos
LD B,A
LD DE,00H
SALCAR: LD A,(IY+0)
ADD A,E
LD E,A
LD A,D
ADC A,00H
LD D,A
INC IY
DJNZ SALCAR
LD A,(NUMCON)
SALCPA1: CP 8 ;Se coloca en la correcta posición
JR Z,SALPOS1 ;de FUZOUT
INC A
INC IY
JR SALCPA1
SALPOS1: LD (IX+VAR1L),E ;Se pone el resultado en un registro
LD (IX+VAR1H),D ;temporal para realizar las operaciones
LD A,(HL)
LD (IX+VAR2L),A
INC HL
LD A,(HL)
LD (IX+VAR2H),A
INC HL
CALL DIV
LD A,(IX+VAR3L)
EXX
LD (HL),A ;CICLOS1=FXZS1/DE
INC HL
EXX
DEC C ;Lo realizo para todas las salidas
JR NZ,SALB
POP HL
POP DE
POP BC
POP AF
RET
```

Listado 5.6 Ensamblador para evaluar la salida

Hasta este momento se tienen los resultados generados por la lógica difusa, a continuación se procede a acondicionar los valores para los actuadores de salida. Cabe

mencionar que el acondicionamiento depende de la estructura del sistema, por lo que su desarrollo no es general.

Por último se tiene la función que controlará el ángulo de disparo, la cual comprende un rango establecido de FFH a 00H, con un equivalente aproximado de 0.7° por unidad hexadecimal. El diagrama de flujo se ilustra en la figura 5.8.

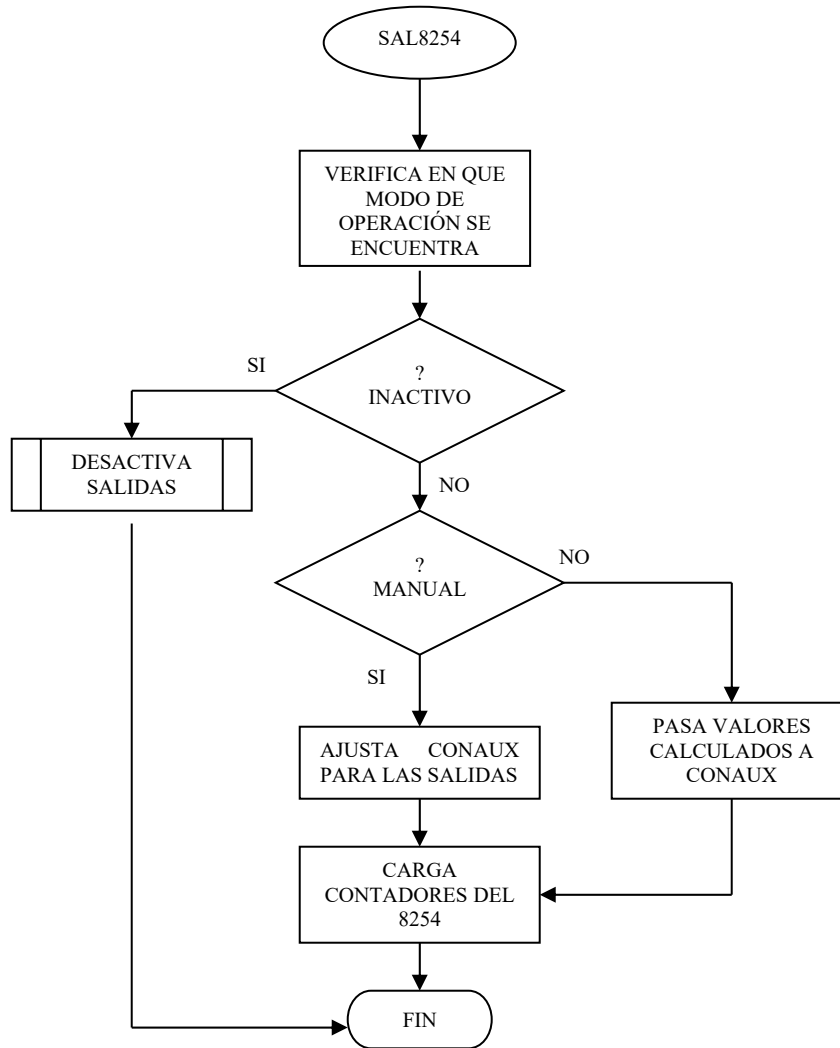


Diagrama 5.8 Descripción para controlar el ángulo de disparo

A continuación en el listado 5.7 se tiene el programa SAL8254 ensamblado. Cabe mencionar que el puerto de salida se denomina 8254. Como se menciona en el capítulo cuatro, el 8254 es un timer que nos da la funcionalidad para poder hacer el disparo a un

ángulo determinado por lo que de esa manera se libera carga de trabajo para el microprocesador y así también la complejidad de programación.

```
;***** FUNCION SAL8254 *****
SAL8254:  PUSH AF
          PUSH BC
          PUSH DE
          PUSH IY
          LD A,(IX+B_M)
          CP 04H                ;Verifica si esta en inactivo
          JR NZ,CONAC
          CALL DES8254          ;Si est en inactivo anula las salidas
          JR CONREG            ;el sistema no se altere
CONAC:   LD A,(IX+B_M)        ;Verifica si esta manual
          CP 02H
          JR Z,CONSIG
          LD B,04H
          LD IY,CICLOS1
CONREC:  LD A,(IY+0)
          LD (IY+CONAUX),A
          INC IY
          DJNZ CONREC
CONSIG:  LD IY,CICLOS1+CONAUX
          LD A,0
          LD B,(IY+0)         ;Valor inicial para el cont3
          SUB B
          CP 0H
          JR NZ,SALUNO
          LD A,0FFH
SALUNO:  OUT (P8254+2),A
          LD A,0
          OUT (P8254+2),A
          LD B,(IY+1)         ;Valor inicial para el cont2
          SUB B
          CP 0H
          JR NZ,SALDOS
          LD A,0FFH
SALDOS:  OUT (P8254+1),A
          LD A,0
          OUT (P8254+1),A
CONREG:  POP IY
          POP DE
          POP BC
          POP AF
          RET
;***** DES8254 *****
DES8254:  PUSH AF
          LD A,WORDCON1      ;Programa el Contador1 como generador de onda
          OUT (P8254+3),A    ;cuadrada a partir del CLK del sistema
          LD A,DIV255        ;para que entren 255 divisiones en el periodo
          OUT (P8254),A      ;de media onda senosoidal de la fuente
          LD A,0              ;eléctrica
          OUT (P8254),A
          LD A,WORDCON2      ;Programa el Contador2 como diparado por señal
```

```
OUT (P8254+3),A      ;de hardware
LD A,DIVINI          ;Valor inicial para el cont2****!!!!!!!
OUT (P8254+2),A
LD A,0
OUT (P8254+2),A
LD A,WORDCON3       ;Programa el Contador3 como diparado por señal
OUT (P8254+3),A     ;de hardware
LD A,DIVINI          ;Valor inicial para el cont3****!!!!!!!
OUT (P8254+1),A
LD A,0
OUT (P8254+1),A
POP AF
RET
```

Listado 5.7 Ensamblador para periferico de salida

5.2 COMUNICACIÓN CON EL USUARIO

La comunicación con el usuario es el medio por el cual se verifica que el sistema esté funcionando correctamente o bien para determinar parámetros de funcionamiento. Por ejemplo, poder determinar la temperatura a la que se desea controlar el sistema, actividad o inactividad del mismo, modo de uso: manual o automático, etc.

Para determinar esta parte del programa considérese algunos objetivos como son:

- Que la manipulación del sistema sea lo más sencillo posible
- El manejo de botones sea intuitivo
- Los despliegues sean cortos pero claros

Los objetivos anteriores son enfocados a usuarios que no necesariamente conozcan de Lógica Difusa, aquellos que sólo se limitan a controlar la operación del sistema.

Asimismo, es necesario una comunicación para modificar los parámetros de control. Esto, por la necesidad de examinar el sistema, así como para desarrollar la base de conocimiento en tiempo real sin la necesidad de estar grabando en memoria EPROM para cada prueba.

El sistema tiene cuatro modos de operación que son:

- 1.- AUTOMATICO
- 2.- MANUAL
- 3.- INACTIVO
- 4.- PROGRAMA

El modo AUTOMATICO es aquel que desarrolla el control con las herramientas de Lógica Difusa. En este modo se selecciona la temperatura deseada a través de los botones INTRO y SELEC, pero en el transcurso de la selección el control se desactiva.

El segundo modo es MANUAL, el cual se encarga de dar un ángulo de disparo determinado sin llevar a cabo cualquier tipo de control. Esto se implementa con el objetivo de realizar pruebas a los actuadores. Se puede variar los ángulos de disparo de las salidas en forma independiente.

El tercer modo permite poner INACTIVO el sistema, es decir, desactiva las salidas y se concreta a actualizar la temperatura de despliegue.

Por ultimo, se considera el modo PROGRAMA, en el cual se requiere tener conocimientos básicos de Lógica Difusa, ya que en esta parte puede modificarse la base del conocimiento.

En el modo PROGRAMA se tiene un submenú en donde se puede escoger diferentes características como *cambiar despliegue*, *cambiar de valores predeterminados a programados* y *cambiar rangos*.

Cambiar despliegue alterna la forma de salida de información del sistema, por un lado se tiene solo el despliegue de la temperatura interna actual y por el otro se despliega tanto la temperatura interna actual como la temperatura externa del sistema, ya que como se mencionó en un principio, cuenta con dos sensores de temperatura.

Cambiar valores predeterminados a programados y viceversa es una forma de modificar el sistema, es decir, al definir la base de conocimiento en forma diferente de cómo se tiene en memoria ROM se puede pasar de uno a otro sin problema.

El funcionamiento del sistema puede ser modificado al cambiar la base de conocimiento, esto puede realizarse al *cambiar rangos* de algunos parámetros como:

- Número de Entradas.
- Número de Salidas.
- Número de Conjuntos.
- Conjuntos de Entrada.
- Conjuntos de Salida.
- Reglas.

Tanto para el número de entradas como para el número de salidas se tiene un tope de 4, es decir, por efecto de la palabra de control sólo pueden manejarse en el sistema 4

entradas y 4 salidas. Cabe mencionar que estas entradas deben implementarse en el programa ya que pueden ser variables físicas o bien comportamiento de ellas como son la derivada o la integral permitiendo, de esta manera, realizar control del tipo PD, PI o PDI.

El número de Conjuntos está limitado a 8 por la misma razón que las entradas. Este número de conjuntos es considerado tanto para los conjuntos de entradas como para los conjuntos de salidas.

Los conjuntos de entrada estarán limitados por el número de conjuntos previamente seleccionado y se podrán dar acceso a 4 Puntos de un conjunto, esto para dar mas flexibilidad al sistema ya que los conjuntos podrán ser trapezoidales y triangulares.

Los conjuntos de salida están también limitados por el número de conjuntos anteriormente fijado y estos conjuntos de salida tienen la forma de SINGLETON, por lo que solo se admite un número que corresponde a la magnitud, en porcentaje, del ángulo de disparo, que va desde 00H, que es el 0%, hasta el FFH que implica un 100%.

Por ultimo se tienen las reglas que se refiere precisamente al valor correspondiente a una regla predeterminada. Se puede tener acceso a 255 reglas.

Para manejar los valores se tienen dos teclas que son las de “+” y “-“, ellas son las que permiten incrementar o decrementar un valor en particular. Estas teclas tienen la función de auto - incremento, es decir, si se mantiene presionada la tecla por un determinado tiempo, el incremento o decremento se realiza de manera acelerada.

En los modos de operación AUTOMATICO y MANUAL se tiene la opción de cambiar la temperatura deseada y el ángulo de disparo respectivamente.

En la presentación todo el listado del sistema operativo ya codificado se muestra en el apéndice C.

CAPITULO 6

RESULTADOS Y CONCLUSIONES

6.1 INTRODUCCION

Sin duda, el control juega una parte muy importante en muchos de los procesos de hoy en día y su alcance es simplemente ilimitado, mientras más complejo sea el proceso a controlar, el sistema de control tendrá que cumplir con las necesidades requeridas: velocidad, exactitud, ser óptimo, práctico y económico; de esta manera, deben considerarse nuevas y evolucionadas técnicas que permitan cumplir con todas las necesidades presentes.

Actualmente una de las técnicas que ha evolucionado, que se adecua, en buena medida, a las demandas actuales es el *Control con Lógica Difusa* (FLC), el cual se considera dentro de la clasificación de sistemas expertos, es decir, el desarrollo de ciertos aspectos de un sistema difuso depende, en gran parte, de la sensibilidad y experiencia humana, por ejemplo, el ajuste de un sistema de aire acondicionado de un recinto, en donde la persona que lo controla tiene la sensibilidad de controlar las condiciones de dicho sistema con base en la experiencia que se adquiere después de ajustarlo varias veces.

Durante el desarrollo del presente trabajo se percibió la importancia de tener un amplio conocimiento sobre el proceso a controlar, pues de ello depende el óptimo ajuste del sistema de control, no basta solamente con tener una buena arquitectura electrónica o una adecuada programación del kernel de la lógica difusa; en realidad lo más importante es tener una base de conocimiento sólida y eficaz para el proceso a controlar.

El planteamiento del presente proyecto fue pensado para que fuera un sistema que tuviera los recursos necesarios para poder ser una herramienta de diseño y ajuste de la base del conocimiento, esto por las necesidades mismas de ajustar y probar. Esto elevó en cierto grado la complejidad en el desarrollo del mismo, pero permitió, por otro lado, la realización de un sistema completo y didáctico, facilitando el ajuste de parámetros para su correcto funcionamiento.

La naturaleza de la estructura del sistema permitió el desarrollo de un sistema de control derivativo, con el firme propósito de seguir una entrada deseada hasta obtener el mínimo error posible.

6.2 RESULTADOS

Los resultados obtenidos, los cuales se muestran en la figura 6.1, proyectan el comportamiento del control, pero cabe mencionar que fue necesario modificar, en diversas ocasiones, conjuntos y reglas para poder establecer una sintonización del sistema óptima y que cumpla con las necesidades del usuario. Esa es la particularidad de la lógica difusa, ya que un control convencional binario no permitiría realizar esa sintonización tan accesible sin evitar cálculos matemáticos e incluso inserción de etapas en el hardware.

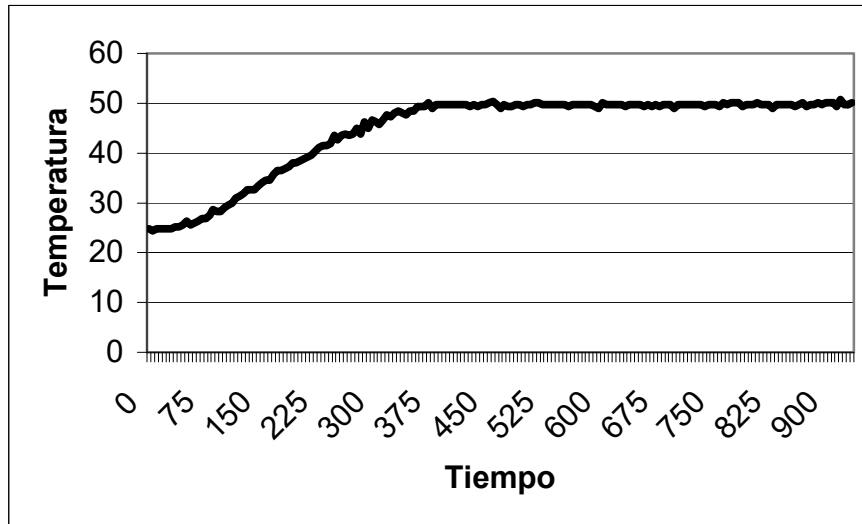


Figura 6.1.a) Respuesta de incremento de temperatura

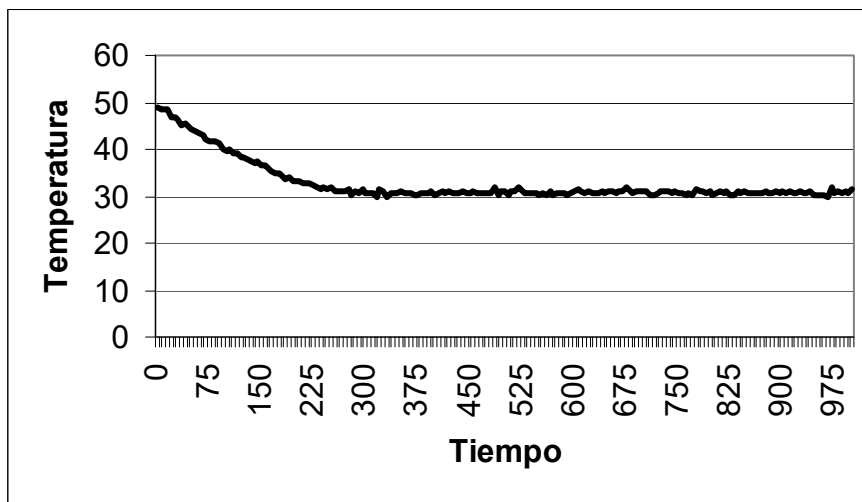


Figura 6.1.b) Respuesta del decremento de temperatura

Figura 6.1 Respuesta del control derivativo

En la figura 6.1 se muestra la respuesta del sistema de control derivativo donde se ilustra la parte ascendente y descendente de la misma.

El sistema resulto ser amigable y manejable gracias al diseño que se planteo desde un principio, controla la temperatura con una pequeña oscilación de $\pm 1^{\circ}\text{C}$, pero en aplicaciones como las que se proponen este error es tolerable.

6.3 CONCLUSIONES

Para desarrollar un sistema de control con lógica difusa fue necesario considerar la importancia del fenómeno a controlar determinando ciertas características y de esa forma, elegir los requerimientos necesarios y la electrónica asociada. Si se requiere controlar un determinado proceso donde los cambios de las variables son lentas (como la temperatura de una pecera) no es necesario tener grandes recursos en el sistema de control, bastaría con un microcontrolador de 8 bits con una velocidad modesta. Asimismo, si el sistema a controlar no requiere de mucha exactitud o precisión, es suficiente con determinar pocos conjuntos de entrada y salida con ciertas reglas.

Cuando se planteó la idea de desarrollar un control difuso se pensó en que tuviera la facilidad de modificar su funcionamiento con el objetivo de hacer pruebas sin tener que emplear otras herramientas, por lo que la arquitectura del software esta diseñada para soportar cambios en los parámetros, tanto en los conjuntos de entrada como de salida y también en las reglas, además de aceptar diferentes modos de trabajo. Todo esto facilito, en gran medida, el ajuste de la base del conocimiento sin la necesidad de borrar y grabar memorias.

La ventaja que presenta este sistema difuso es que se puede realizar y mantener una sintonización del control y un comportamiento físico y funcional del mismo, ya que haciendo uso del control convencional serían necesarias diversas herramientas físicas y matemáticas para realizar lo que con la lógica difusa se haría sólo cambiando condiciones lingüísticas en las reglas correspondientes.

Asimismo, presenta una buena respuesta al incrementar la temperatura, esto gracias a los cálculos que se realizaron para determinar las características de los actuadores que se utilizan, pero cuando se requiere bajar la temperatura, el sistema únicamente lo puede hacer hasta llegar a la referencia establecida por la temperatura ambiente, de exigirse menores temperaturas a esta, deben considerarse otras alternativas que se mencionan más adelante. Por lo tanto, considérese este sistema como un eficiente elevador de temperatura, el cual mantiene condiciones estables partiendo de la línea base que determina la temperatura ambiente.

El software desarrollado para el funcionamiento del sistema permite manipular la base del conocimiento prácticamente en tiempo real; con la visualización establecida en el LCD se pueden manejar otros parámetros del comportamiento del sistema. Pero cabe mencionar que las modificaciones de la base del conocimiento se llevan a cabo en lenguaje de maquina, lo ideal sería que fueran con variables lingüísticas, pero esto demandaba más tiempo de desarrollo y más recursos de memoria.

El sistema difuso permite sacar provecho de la vida útil de los actuadores, ya que basta con cambiar los SINGLETONS para poder seguir operando a los actuadotes que

posiblemente hayan bajado su potencia por cuestiones de uso y sea más caro el cambiarlo que el de modificar los conjuntos de salida. Sin embargo, el cambio de los actuadores implica un repaso del capítulo 2, ya que tomando algunas consideraciones se realizaron los cálculos pertinentes para establecer las condiciones de operación de los mismos.

Las dimensiones físicas de la arquitectura de desarrollo es bastante manejable, pero podría reducirse más realizando una nueva distribución en algún montaje de doble cara como el que presentan algunas compañías comerciales al vender sus arquitecturas microcontroladoras. Sin embargo, cabe aclarar que la arquitectura de este sistema está diseñada de acuerdo a las necesidades que se fueron presentando, difícilmente se hubiera encontrado una arquitectura comercial que cumpliera con las exigencias de esta aplicación. A pesar de que algunos componentes de la arquitectura de desarrollo han existido en el mercado por largo tiempo y pueden ser considerados “viejos”, el criterio empleado para su utilización tiene como base el desarrollar un sistema óptimo y práctico con los menores recursos posibles.

Mejoras en el proceso de control

Con el objetivo de agilizar el decremento en la temperatura, considérese un sistema “serpentin” donde se hace fluir refrigerante. Esto lleva a considerar, dentro de la etapa de actuadores, sistemas de inyección y flujo que pasen por el ventilador.

Podría considerarse una salida más donde se sense el flujo del refrigerante y con base en ese dato, determinar la cantidad de refrigerante que se inyecta. Aunque este rubro puede considerarse constante y dejar al ventilador como el actuador primario que controle el proceso de enfriado. Cabe mencionar que ambos actuadores pueden ser igualmente controlados.

De esta manera, podría agilizarse el decremento de la temperatura dentro de un recinto, el cual presentó un desarrollo lento en las pruebas realizadas.

Obviamente al sintonizar el sistema, variando condiciones de reglas y conjuntos, se obtiene una respuesta más estable y rápida del control, pero debido a que la única fuente externa que puede proporcionarme temperaturas bajas (tendencia negativa de calor) es la temperatura ambiente del propio aire, el sistema, en el proceso de bajar la temperatura, se vuelve un poco lento.

Todo lo contrario en el proceso de calentamiento, ya que para determinar las características de los actuadores participantes, se realizaron los cálculos pertinentes.

Así es como se sugiere utilizar un actuador para enfriamiento con el objetivo de agilizar el proceso de decremento de la temperatura. Esto implica modificar el sistema ya que se deberían tener 3 salidas y desde luego cambiar la base del conocimiento.

Otras aplicaciones del sistema

La aplicación de la lógica difusa en este trabajo permite sustituir la mayor parte de la matemática que se desarrolla al plantear un sistema de control convencional con herramientas vagas e intuitivas difundidas en la lógica difusa. Resumiendo cálculos algebraicos y obtención de ecuaciones de transferencia, lo difuso abunda en el manejo de reglas determinadas lingüísticamente y que describen, al igual que una ecuación de transferencia, el comportamiento físico de un sistema de control.

La importancia de la lógica difusa poco a poco se ha ido proyectando hacia un mejor desarrollo, según las aplicaciones estructuradas con la misma.

Se puede encontrar principios de lógica difusa en campos de investigación, medicina e industria con gran desarrollo creando tecnología vanguardista con diversas aplicaciones.

Dentro de los sistemas que regulan temperatura, se puede mencionar las incubadoras que se desarrollan para el tratamiento pediátrico de un recién nacido; el sólo hecho de establecer las condiciones deseadas en el interior de la misma, permiten al usuario una completa atención hacia otro tipo de factores, olvidándose de la temperatura.

Actualmente existen automóviles con climas inteligentes, los cuales, gracias a la lógica difusa, respetan un cierto rango de temperatura en su interior, manteniendo el confort del usuario, sin importar las condiciones externas de dicho vehículo. Siempre se desea una temperatura ideal y el control debe respetar dicho valor sin importar las perturbaciones externas.

Asimismo, edificios inteligentes se mantienen bajo un cierto rango de temperatura, con lo que se garantiza una buena estancia de las personas en el interior de los mismos.

No sólo es cuestión de confort, existen laboratorios de investigación, p.e. laboratorios de espectrometría gamma, física, química, donde se requiere mantener el sitio a ciertas condiciones de temperatura y humedad, equipos de instrumentación y elementos como el nitrógeno líquido requieren ciertas condiciones que necesitan ser controladas. La lógica difusa es una herramienta poderosa que puede permitir el control de dichos parámetros de estos sitios y más; con los debidos elementos es una forma muy poderosa de realizar control bajo cualquier condición y con características especiales.

APÉNDICE A OPERACIÓN DEL LCD

I. DESCRIPCIÓN

Este espacio presenta la operación del LCD en la aplicación expuesta, de esta manera sólo se presentan a detalle algunas características. Para complementar la información expuesta consultar el manual de operación del LCD HITACHI mod. LCD-II (HD44 780).

El LCD es un modulo controlador de matriz de puntos hechos con cristal liquido LSI. Puede manejar caracteres alfanuméricos y símbolos. Su control puede estar gobernado bajo microprocesadores o microcontroladores de 8-bits y 4-bits. Todas sus funciones requeridas por la matriz de puntos de cristal líquido están internamente provistas en sus controladores internos. El manejo del LCD puede realizarse con un mínimo de circuitos externos ya que cuenta con lo necesario para poder trabajar con él. Tiene además un consumo mínimo de potencia.

II. CARACTERISTICAS

- Capaz de ser manejado por 4-bit y 8-bits
- RAM de despliegue de 80 caracteres máxima.
- ROM generador de caracteres
 - Caracteres de una matriz de 5x7 160
 - Caracteres de una matriz de 5x10 32
- Acción de lectura de RAM y ROM por el CPU
- Funciones consideradas:
 - Limpia display, retorno de cursor, encender/apagar display,
 - encender/apagar cursor, blanquea caracteres del display,
 - corrimiento del cursor, corrimiento del display.
- Reset interno automático al recibir energía.

III. CONEXIÓN LCD-CPU

El LCD cuenta con 14 pines de conexión con lo que se tiene todo lo necesario para poder trabar con él. Dos pines son para la alimentación, uno para ajustar el contraste de despliegue, tres de control y por último cuenta con ocho pines de datos. Los pines se encuentran configurados como se muestra en la figura A.1.

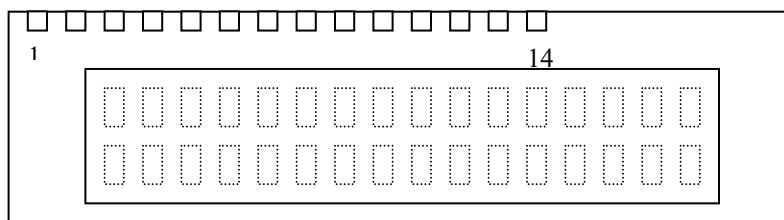


Fig. A.1 LCD

En la tabla A.1 se describen los pines.

PIN	NOMBRE	DESCRIPCIÓN
1	GND	TIERRA
2	V _{DD}	ALIMENTACIÓN 5V
3	V ₀	REGULACIÓN DE CONTRASTE
4	RS	SELECTOR DE REGISTRO. Esta señal se encarga de separa el registro de instrucción (IR) y el registro de datos(DR).
5	R/W	LECTURA/ESCRITURA
6	E	HABILITAR
7	DB ₀	DATO 0
8	DB ₁	DATO 1
9	DB ₂	DATO 2
10	DB ₃	DATO 3
11	DB ₄	DATO 4
12	DB ₅	DATO 5
13	DB ₆	DATO 6
14	DB ₇	DATO 7

Tabla A.1 Descripción de Pines

III.1 CONTRASTE

La conexión del contraste se puede realizar conectando una resistencia variable a entre V_{DD} y GND, donde el cursor estará conectado en V₀ tal como indica la figura A.2. Cabe mencionar que el contraste puede presentar variaciones ligeras en su funcionamiento debido a las variaciones de temperatura consideradas en el LCD, las cuales son originadas por la disipación de energía que presenta la estructura electrónica del mismo. De esta manera para eliminar dichas variaciones y obtener una invariabilidad en el funcionamiento es necesario hacer uso de componentes adicionales referidos en el manual de operación.

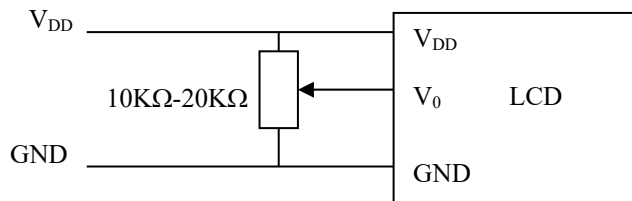


Fig. A.2 Regulación de contraste

Para la aplicación expuesta, la conexión del LCD se expone en la figura A.3. Cabe mencionar que la conexión esta ideada para ocupar el mínimo de hardware sin sacrificar el software. Las líneas GND y V_{DD} se conectar a la alimentación de 5V, V₀ se conecta al cursor de la resistencia variable, RS se encuentra en la línea A₀ del microprocesador y R/W

se conecta con A₁, E se conecta con el Chip Selec respectivo y el bus de datos del LCD se conecta con el bus de datos del microprocesador.

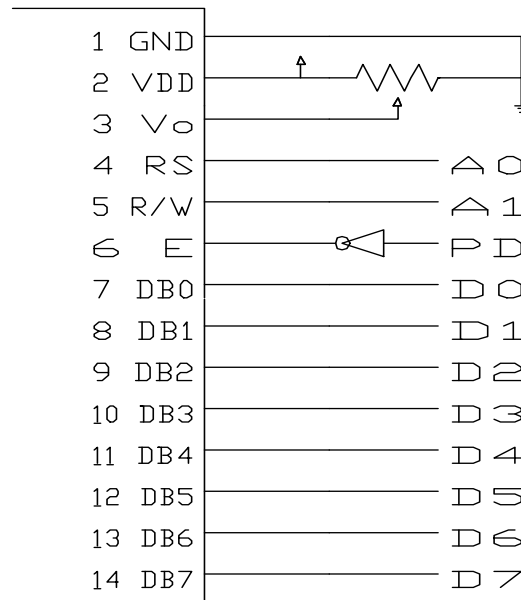


Fig. A.3 Conexión LCD-CPU

Como resultado de la configuración presentada en la figura A.3 se pueden considerar diversas operaciones las cuales son descritas en la tabla A.2.

PD	A ₀	A ₁	OPERACIÓN
1	X	X	NO SELECCIONADO
0	0	0	ESCRITURA DE INSTRUCCIÓN
0	1	0	LECTURA DE BANDERA Y DIRECCIONES
0	0	1	ESCRITURA DE DATOS
0	1	1	LECTURA DE DATOS

Tabla A.2 Operaciones

IV. SOFTWARE

Una vez discutidos los detalles de hardware se procede al análisis del software ya que es necesario conocer las funciones del LCD para poder tener un aprovechamiento máximo del mismo.

Primero lo que se tiene que hacer después del reset es establecer la longitud de datos, número de líneas en el display y fuente de caracteres, control ON/OFF del display, limpiar el display y por ultimo el modo de trabajo del display. En el diagrama A.1 se muestra el procedimiento.

En la tabla A.3 se muestran las funciones y códigos del display.

FUNCIÓN	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	DESCRIPCIÓN
Limpia display	0	0	0	0	0	0	0	0	0	1	Limpia todo el display y retorna el cursor al inicio (dirección 0)
Retorna	0	0	0	0	0	0	0	0	1	*	Retorna el cursor al inicio (dirección 0) sin alterar el contenido en la DD RAM
Modo	0	0	0	0	0	0	0	1	I/D	S	Determina el movimiento del cursor, determina el corrimiento o no del display
Control ON/OFF del display	0	0	0	0	0	0	1	D	C	B	Apaga o enciende a todo el display (D), enciende o apaga al cursor (C), y blanquea la posición del cursor.
Corrimiento del cursor y del display	0	0	0	0	0	1	S/C	R/L	*	*	Recorre el cursor o el display sin afectar el contenido de la DD RAM
Función del LCD	0	0	0	0	1	DL	N	F	*	*	Longitud de datos (DL), número de líneas en el display (L) y fuente de caracteres (F)
Pone dirección CG RAM	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Se establece la dirección de la CG RAM para poder escribir o leer de ella.
Pone dirección DD RAM	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Se establece la dirección de la DD RAM para poder escribir o leer de ella.
Lectura de la bandera BUSY y dirección	0	1	BF	AC	AC	AC	AC	AC	AC	AC	Lectura de la bandera de ocupado (BUSY) y también el valor del contador de dirección en curso
Escritura de datos al CG o DD RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Escritura de datos al CG o DD RAM
Lectura de datos al CG o DD RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Lectura de datos al CG o DD RAM

Tabla A.3 Tabla de funciones

Cabe mencionar que los caracteres del display corresponden al código ASCII a partir de la posición 20H hasta la 7DH.

Por otra parte, el LCD permite la definición de 8 posiciones cuya ubicación será de la localidad 00H hasta la 07H. Los caracteres de estas posiciones se definen al cargar los datos en las localidades apartadas para ellos en la CG RAM.

Ejemplo para desplegar “HOLA” a partir de la conexión propuesta y utilizando el microprocesador Z-80.

```
.ORG PROG           ;Inicio del programa

COMLCD EQU *
.DBYTE 380CH,0601H ;Comandos para programar el LCD

HOLA EQU *
.DBYTE 0FE4FH,5045H,5241H,52FEH,454EH,0FE4DH,4F44H,4FFE4H
           ;Letrero

LD BC,LCD+400H      ;Carga el periférico y #de datos
                   ;para el LCD
LD HL,COMLCD        ;Lugar de comandos para el LCD
SIG: CALL LET_LIS   ;LCD listo para recibir
      OUTI          ;Has una transferencia
      JR NZ,SIG
      LD BC,LCD+1001H ;Cargar el periférico y #de datos para el
                   ;LCD
      LD HL,HOLA     ;Lugar del letrero para el LCD
SIG: CALL LET_LIS   ;LCD listo para recibir
      OUTI          ;Has una transferencia
      JR NZ,SIG
FIN:  JR FIN        ;Fin del programa
LET_LIS: PUSH AF
ESP:  IN A,(LCD+2)  ;Lee la bandera
      AND 80H       ;Verifica si esta limpia
      JR NZ,ESP
      POP AF
      RET           ;retorna
```

APÉNDICE B

SITEMA DE DESARROLLO

En este apartado se expone la realización de un sistema de desarrollo cuyo funcionamiento facilite la manipulación y ejecución del proyecto. Cabe mencionar que este sistema se diseñó para tener el máximo aprovechamiento del microprocesador y poder visualizar su secuencia en el procesamiento.

Primeramente, para llevar a cabo esta herramienta se identificaron las principales características de funcionalidad con el objetivo de establecer las condiciones estructurales de la misma. A continuación se mencionan las características del sistema de desarrollo:

- Microprocesador Z80A
- Frecuencia de trabajo 4MHz
- Comunicación paralela con la PC
- Cuatro puertos I/O
- Cuatro selectores de memoria
- Acceso a todas las direcciones, datos y líneas de control del Z80
- 4Kbytes de EPROM
- 16Kbytes de RAM

Una vez identificadas las características se procede a la conexión del sistema para esto contamos con la siguiente lista de materiales:

- Microprocesador Z80
- 2716 (EPROM)
- 62256 (RAM)
- 8255 (PPI)
- 74LS139
- 74LS04
- 5 resistencias 4.7K Ω
- 2 resistencias 330 Ω
- 1 resistencia 10K Ω
- Diodo 1N914
- Capacitor de 10 μ F
- Cristal 4MHz
- Botón
- Conector DB-25
- Conector tipo Peine paso 100 de 50 pines
- Cabezales

El conector cuenta con 50 pines conectados a todas las líneas de control, direcciones y datos necesarios. En la figura B.1 se muestra la configuración del conector, como también se indicaron entre paréntesis el número de pin y de CI donde va conectado.

Control de temperatura con lógica difusa

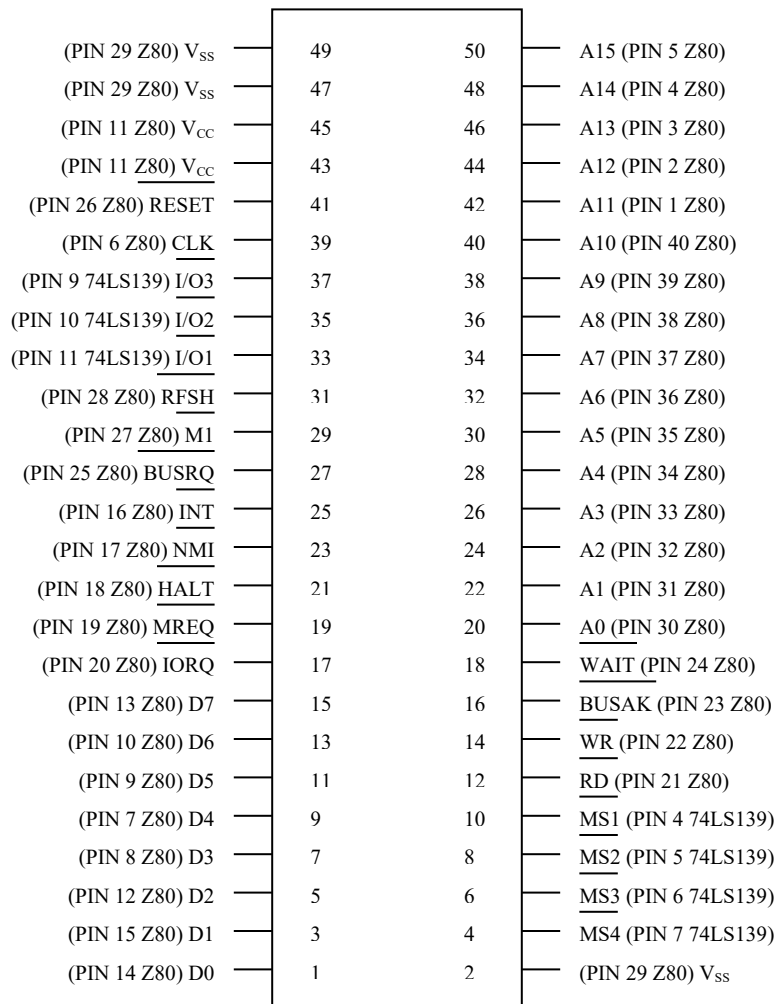


Fig. B.1 Configuración de pines

Los valores de los puertos de I/O y de los selectores de memoria son como siguen:

- I/O0 00H (Este puerto corresponde a la PPI 8255 por lo que no es externo)
- I/O1 40H Uso externo
- I/O2 80H Uso externo
- I/O3 C0H Uso externo
- MS1 0000H (Corresponde a la EPROM)
- MS2 4000H (Corresponde a la RAM)
- MS3 8000H Uso externo
- MS4 C000H Uso externo

Concluido lo que corresponde a la parte del hardware del sistema se procede ahora al software el cual tiene una gran importancia ya que ser en realidad quien determine el comportamiento de nuestro sistema. El software se compone de dos partes importantes: la primer parte es un programa escrito en ensamblador para el Z80 del sistema armado, grabado en la EPROM y la segunda es un programa escrito en lenguaje C para la PC. Al primer programa que esta en ensamblador lo llamaremos TDESZ80.ASM y para el segundo escrito en lenguaje C lo llamaremos TDESZ80.CPP. Para la realización de ambos programas se definen las funciones que se necesitan.

CC Y CM

Estas funciones se encargaran de cargar datos a la memoria RAM en una dirección específica. Cargar Código (**CC**) es específica para enviar un bloque de datos por el puerto paralelo de la PC donde esta última a su vez obtendrá estos datos de un archivo tipo INTEL-HEX. Cargar Memoria(**CM**) es la misma función que **CC** pero la diferencia radica en la PC en donde ahora se pone la dirección, el tamaño del bloque y el dato a cargar desde los comandos del programa TDESZ80.EXE.

CP

Cargar Periférico (**CP**) se encarga de enviar un dato al periférico (puerto I/O) indicado.

DR

Desplegar Registros (**DR**) se encarga de mandar los contenidos de los registros principales y auxiliares a la PC donde son desplegados en pantalla.

DM

Desplegar Memoria (**DM**) se encarga de mandar un bloque de ocho contenidos de memoria a partir de la dirección indicada y ser desplegados en pantalla.

DP

Desplegar Periférico (**DP**) se encarga de mandar el dato obtenido del periférico especificado y desplegarlo en pantalla.

P

Parar programa (**P**) detiene la ejecución de algún programa. Reset por comando.

E

Ejecutar programa (**E**) ocurre un salto hacia una dirección dada desde la PC y comienza o ejecutar las instrucciones a partir de ahí.

S

Este comando afecta solo a la ejecución de la PC e indica que salga del programa, para el sistema de desarrollo no tiene efecto.

L

Limpiar pantalla. Solo afecta a la pantalla de la PC.

Para el reconocimiento de estos comandos se definen sus claves como se indica a continuación:

- **CC Y CM** 01h
- **DM** 02h
- **DP** 04h
- **CP** 08h
- **DR** 10h
- **E** 20h
- **P** 40h

A continuación se presentan los programas tanto para el Z80 como para la PC.

Ensamblador para el Z80

```
;*****FUNCION INICIO*****  
.ORG 0000H  
P8255 EQU 00H  
SPINI EQU 4200H  
LUGARSP EQU 4000H  
LD SP,SPINI ;CARGA EL STAK POINTER  
LD A,0F8H ;PROGRAMA AL 8255  
OUT (P8255+3),A  
LD A,09H  
OUT (P8255+3),A  
IM 1  
ABINT: EI ;Lugar donde brinca de SI (ES A)  
HALT  
JR ABINT  
;*****FUNCION DE SERVICIO DE INTERRUPCION*****  
INT: DI  
PUSH IY  
PUSH IX  
PUSH HL  
PUSH DE  
PUSH BC  
PUSH AF  
LD (LUGARSP),SP  
CALL ENTRADA ;LLAMA A ENTRADA  
CP 01H  
JR NZ,NCC  
CALL CC ;LLAMA A LA FUNCION CC  
JR BRIN  
NCC: CP 02H ;COMPARA PARA LA FUNCION DM  
JR NZ,NDM  
CALL DM ; LLAMA A LA FUNCION DM  
JR BRIN  
NDM: CP 04H ;COMPARA PARA LA FUNCION DP  
JR *+5  
.ORG 0038H  
JR INT ;INTERUPCION  
JR NZ,NDP  
CALL DP ; LLAMA A LA FUNCION DP  
JR BRIN
```

Control de temperatura con lógica difusa

```

NDP:      CP 08H          ;COMPARA PARA LA FUNCION CP
          JR NZ,NCP
          CALL RCP        ; LLAMA A LA FUNCION CP
          JR BRIN
NCP:      CP 10H          ;COMPARA PARA LA FUNCION DR
          JR NZ,NDR
          CALL DR         ; LLAMA A LA FUNCION DR
          JR BRIN
NDR:      CP 20H          ;COMPARA PARA LA FUNCION EX
          JR NZ,NEX
          CALL EXE        ; LLAMA A LA FUNCION EX
          JR BRIN
NEX:      CP 40H          ;COMPARA PARA LA FUNCION ST
          JR Z,ST
BRIN:     POP AF
          POP BC
          POP DE
          POP HL
          JR  *+5
          .ORG 0066H
          JP 4400H        ;*****INTERRUPCION NO ENMASCARABLE*****
          POP IX
          POP IY
          EI
          RET
ST:       POP AF
          POP BC
          POP DE
          POP HL
          POP IX
          POP IY
          INC SP
          INC SP
          JP ABINT
;*****FUNCION CARGAR CODIGO (CC)*****
CC:       CALL ENTRADA    ;LLAMA A LA FUNCION ENTRADA
          LD B,A
          CALL ENTRADA    ;LLAMA A LA FUNCION ENTRADA
          LD C,A
BANPE:    CALL ENTRADA    ;LLAMA A LA FUNCION ENTRADA
          LD H,A
          CALL ENTRADA    ;LLAMA A LA FUNCION ENTRADA
          LD L,A
          LD D,H
          LD E,L
          CALL ENTRADA    ;LLAMA A LA FUNCION ENTRADA
          LD (HL),A
          LDI
          JP PE,BANPE
          RET
;*****FUNCION ENTRADA*****
ENTRADA:  IN A,(P8255+2)   ;SUBROUTINA ENTRADA
          BIT 3,A
          JR Z,ENTRADA
          IN A,(P8255)
          RET

```

Control de temperatura con lógica difusa

```
;*****FUNCION SALIDA*****
SALIDA:  PUSH AF          ;SUBROUTINA SALIDA
SALID:   IN A,(P8255+2)
         BIT 7,A
         JR Z,SALID
         POP AF
         OUT (P8255),A
         RET

;*****FUNCION DESPLEGAR MEMORIA (DM)*****
DM:      CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         LD H,A
         CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         LD L,A
         LD B,08H
SIGA:    LD A,(HL)
         INC HL
         CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
         DJNZ SIGA
         RET

;*****FUNCION CARGA PERIFERICO (CP)*****
RCP:     CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         LD C,A
         CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         OUT (C),A
         RET

;*****FUNCION DESPLIEGA PERIFERICO (DP)*****
DP:      CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         LD C,A
         IN A,(C)
         CALL SALIDA      ; LLAMA A LA FUNCION SALIDA
         RET

;*****FUNCION DE EXECICION (EX)*****
EXE:     LD SP,SPINI      ;CARGA EL STAK POINTER
         CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         LD H,A
         CALL ENTRADA     ;LLAMA A LA FUNCION ENTRADA
         LD L,A
         EI
         JP (HL)

;*****FUNCION DESPLEGAR REGISTROS (DR)*****
DR:      LD HL,(LUGARSP)
         LD B,0EH
AHL:     LD A,(HL)
         INC HL
         CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
         DJNZ AHL
         EX AF,AF'
         CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
         PUSH AF
         EX (SP),HL
         LD A,L
         CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
         EX (SP),HL
         POP AF
         EX AF,AF'
         EXX
         LD A,C
```

Control de temperatura con lógica difusa

```
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
LD A,B
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
LD A,E
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
LD A,D
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
LD A,L
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
LD A,H
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
LD A,I
CALL SALIDA      ;LLAMA A LA FUNCION SALIDA
EXX
RET
```

Programa en lenguaje C para la PC

```
/*ESTE PROGRAMA ES PARA LA TARGETA DE DESARROLLO TDES_Z80 QUE TIENE EL
SISTEMA
```

```
TDESZ80.ASM Y ES CAPAS DE CARGAR ARCHIVOS TIPO CML Y TIPO INTEL.HEX*/
```

```
#include<stdio.h>
#include<conio.h>
#include<dos.h>
#include<ctype.h>
#include<stdlib.h>
#include<process.h>
#include<bios.h>
#define CC 0x01 //Cargar codigo
#define DM 0x02 //Desplegar memoria
#define DP 0x04 //Desplegar periferico
#define CP 0x08 //Cargar periferico
#define DR 0x10 //Desplegar Registros
#define EX 0x20 //Ejecutar Programa
#define ST 0x40 //Parar programa

FILE *pf;
unsigned con=0,con2=0,exe=0x4200,ban=1,colum=1;
char nom[50]; //variables globales

void lissal(unsigned salida)
{
    unsigned est;long inter=0; //funcion de salida
    do{est=inp(0x379)&0x40;inter++;}while(est!=0&&inter<100000);
    if(inter==100000)
    {
        printf("\nNO RESPONDE\n");outp(0x378,0x00);sound(800);
        delay(500);nosound();delay(1000);exit(0);
    }
    else
    {
        outp(0x378,salida);outp(0x37a,inp(0x37a)|0x01);
        outp(0x37a,inp(0x37a)&0xfe);
    }
}

unsigned lisent(void)
```

Control de temperatura con lógica difusa

```
{
    unsigned est; long inter=0; //función de entrada
    do{est=inp(0x379)&0x10;inter++;}while(est!=0&&inter<100000);
    if(inter==100000)
    {
        printf("\nNO RESPONDE\n"); outp(0x378,0x00); sound(800);
        delay(500); nosound(); delay(1000); exit(0);
    }
    else
    {
        outp(0x37A,inp(0x37a)|0x08); est=inp(0x378);
        outp(0x37A,inp(0x37a)&0xf7);
    }
    return (est);
}

void cc(void)
{
    // función que carga desde un disco un
    unsigned dat, dat2, i, ac=1, ban=0, dir=0x4200, dirH, con_hex=0, b_c=0;
    // archivo que sea en ASCII escrito desde
    // algun editor.
    printf(" Nombre del archivo: ");
    gets(nom);
    if(nom[0]=='\x0') return;
    pf=fopen(nom, "r"); con=0;
    while(!ferror(pf)&&!feof(pf)&&pf!=NULL)
    {
        add: dat=fgetc(pf);
        if(con==0&&dat==58){b_c=1; goto hex1;} //indica que es un
        //archivo *.hex
        if(!isxdigit(dat))
        {
            if(dat==34)
            while(fgetc(pf)!=34);
            if(dat==36) ban=1;
        } //indica que es una dirección
        else
        {
            if(dat>96) dat=dat-32; //ASCII - > HEXADECIMAL
            if(dat>64) dat=dat-55;
            else dat=dat-48;
            if(ac!=1)
            {
                dat=dat2*16+dat; ac=1;
                if(ban==0)
                {
                    lissal(CC);
                    lissal(0); lissal(1);
                    lissal(dir/256); lissal(dir&0xff);
                    lissal(dat);
                    gotoxy(5,24); printf("Numero de byte
transmitido: %XH %i", con, con);
                    con++;
                    dir++;
                }
            }
            if(ban==1){ dirH=dat; ban=2; goto add;}
            if(ban==2){ exe=dir=dirH*256+dat; ban=0;}
        }
    }
}
```

Control de temperatura con lógica difusa

```

        do
        {
            dat=fgetc(pf);
        }
        while(dat!=10);

    }
    else
    {
        ac=2;dat2=dat;
    }
}
goto salta; hex1:
while(!ferror(pf)&&!feof(pf)&&pf!=NULL)
{
    hex:  dat=fgetc(pf);
    if(!isxdigit(dat))
    {
        if(dat==58){b_c=1;goto hex;}
        goto hex2;
    }
    if(dat>96)dat=dat-32;
    if(dat>64)dat=dat-55;
    else dat=dat-48;
    if(ac!=1)
    {
        dat=dat2*16+dat;ac=1;
        if(ban==0&&b_c==0&&con_hex!=0)
        {
            lissal(CC);
            lissal(0x00);lissal(0x01);
            lissal(dir/256);lissal(dir&0xff);
            lissal(dat);
            gotoxy(5,24);printf("Numero de byte transmitido:
%XH    %i    ",con,con);
            con++;
            dir++;
            con_hex--;
        }
        if(ban==1){ dirH=dat;ban=2;goto hex;}

    if(ban==2){exe=dir=dirH*256+dat;ban=0;dat=fgetc(pf);dat=fgetc(pf);}
    if(b_c==1){con_hex=dat;b_c=0;ban=1;if(dat==0)goto salta;}
        }
        else
        {
            ac=2;dat2=dat;
        }
        hex2:
    }
salta:
if(ferror(pf)||pf==NULL)
{
    error:  printf("\nERROR DE LECTURA");
    sound(800);delay(500);nosound();
    delay(1000);clearerr(pf);exit(0);
}

```

Control de temperatura con lógica difusa

```
    }
    delay(500);
}

void cm(void)
{ //función Cargar Memoria
  unsigned b,c,d;
  printf(" Dir ini: $");scanf("%x",&b);gets(nom);
  gotoxy(25,24);printf(" No de loc: $");scanf("%x",&c);gets(nom);
  gotoxy(45,24);printf(" Dato:
  $");scanf("%x",&d);gets(nom);lissal(CC);
  if(c>0){lissal((c/256));lissal((c&0xff));}else return;
  while(c>0)
  {
    lissal((b/256));lissal((b&0xff));
    lissal(d);b++;c--;
  }
}

void cp(void)
{ //función Cargar Perif,rico
  unsigned p,d;
  printf(" Perif,rico: $");scanf("%x",&p);gets(nom);
  gotoxy(30,24);printf(" Dato: $");scanf("%x",&d);gets(nom);
  lissal(CP);
  lissal(p);
  lissal(d);
}

void dm(void)
{
  unsigned i,p,fi,co;unsigned d; // función Desplegar Memoria
  printf(" Dirección: $");scanf("%x",&d);gets(nom);
  lissal(DM);
  lissal((d/256));
  lissal((d&0xff));outp(0x37a,inp(0x37a)|0x20);
  if(colum==1||colum==2)fi=2;else fi=12;
  if(colum==1||colum==3)co=2;else co=20;
  for(i=0;i<8;i++)
  {
    gotoxy(co,fi);p=d/256;if(p<16)printf("0%X",p);else
printf("X",p);
    p=d&0xff;if(p<16)printf("0%XH - ",p);else printf("%XH - ",p);
    p=lisent();fi++;d++;
    if(p<16)printf("$0%X ",p);else printf("$%X ",p);
  }
  if(colum==4)colum=1;else colum++;
  outp(0x37a,inp(0x37a)&0xde);outp(0x37a,inp(0x37a)&0xf7);
  outp(0x378,0x00);
}

void dp(void)
{
  unsigned p,d;//función Desplegar Perif,rico
  printf(" Perif,rico: $");scanf("%x",&p);gets(nom);
  lissal(DP);
  lissal(p);outp(0x37a,inp(0x37a)|0x20);
  d=lisent();
  gotoxy(76,20);if(p<16)printf("$0%X ",p);else printf("$%X ",p);
  gotoxy(67,21);
```


Control de temperatura con lógica difusa

```
    if(d<16)printf("Dato: $0%X  ",d);else printf("Dato: $%X  ",d);
    outp(0x37a,inp(0x37a)&0xde);outp(0x37a,inp(0x37a)&0xf7);
    outp(0x378,0x00);
}
void dr(void)
{
    unsigned rL,rH; //funci n Desplegar Registros
    lissal(DR);outp(0x37a,inp(0x37a)|0x20);
    rL=lisent();rH=lisent();gotoxy(40,3);
    if(rH<16)printf("A:0%X  ",rH);else printf("A:%X  ",rH);
    if(rL<16)printf("F:0%X  ",rL);else printf("F:%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(40,5);
    if(rH<16)printf("B:0%X  ",rH);else printf("B:%X  ",rH);
    if(rL<16)printf("C:0%X  ",rL);else printf("C:%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(40,7);
    if(rH<16)printf("D:0%X  ",rH);else printf("D:%X  ",rH);
    if(rL<16)printf("E:0%X  ",rL);else printf("E:%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(40,9);
    if(rH<16)printf("H:0%X  ",rH);else printf("H:%X  ",rH);
    if(rL<16)printf("L:0%X  ",rL);else printf("L:%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(42,11);
    if(rH<16)printf("IX:0%X",rH);else printf("IX:%X",rH);
    if(rL<16)printf("0%X  ",rL);else printf("%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(42,13);
    if(rH<16)printf("IY:0%X",rH);else printf("IY:%X",rH);
    if(rL<16)printf("0%X  ",rL);else printf("%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(42,16);
    if(rH<16)printf("PC:0%X",rH);else printf("PC:%X",rH);
    if(rL<16)printf("0%X\n",rL);else printf("%X\n",rL);
    rL=lisent();rH=lisent();gotoxy(60,3);
    if(rL<16)printf("A':0%X  ",rL);else printf("A':%X  ",rL);
    if(rH<16)printf("F':0%X  ",rH);else printf("F':%X  ",rH);
    rL=lisent();rH=lisent();gotoxy(60,5);
    if(rH<16)printf("B':0%X  ",rH);else printf("B':%X  ",rH);
    if(rL<16)printf("C':0%X  ",rL);else printf("C':%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(60,7);
    if(rH<16)printf("D':0%X  ",rH);else printf("D':%X  ",rH);
    if(rL<16)printf("E':0%X  ",rL);else printf("E':%X  ",rL);
    rL=lisent();rH=lisent();gotoxy(60,9);
    if(rH<16)printf("H':0%X  ",rH);else printf("H':%X  ",rH);
    if(rL<16)printf("L':0%X  ",rL);else printf("L':%X  ",rL);
    rL=lisent();gotoxy(62,11);
    if(rL<16)printf("I:0%X  ",rL);else printf("I:%X  ",rL);
    outp(0x37a,inp(0x37a)&0xde);outp(0x37a,inp(0x37a)&0xf7);
    outp(0x378,0x00);
}
void ex(void)
{ //funci n Executar programa
    printf("  Executar en $");scanf("%x",&exe);gets(nom);
    lissal(EX);
    lissal((exe/256));
    lissal((exe&0xff));ban=0;
}
void main(void)
{ // programa maestro
    char op='n';
    outp(0x37a,inp(0x37a)&0xde);
```

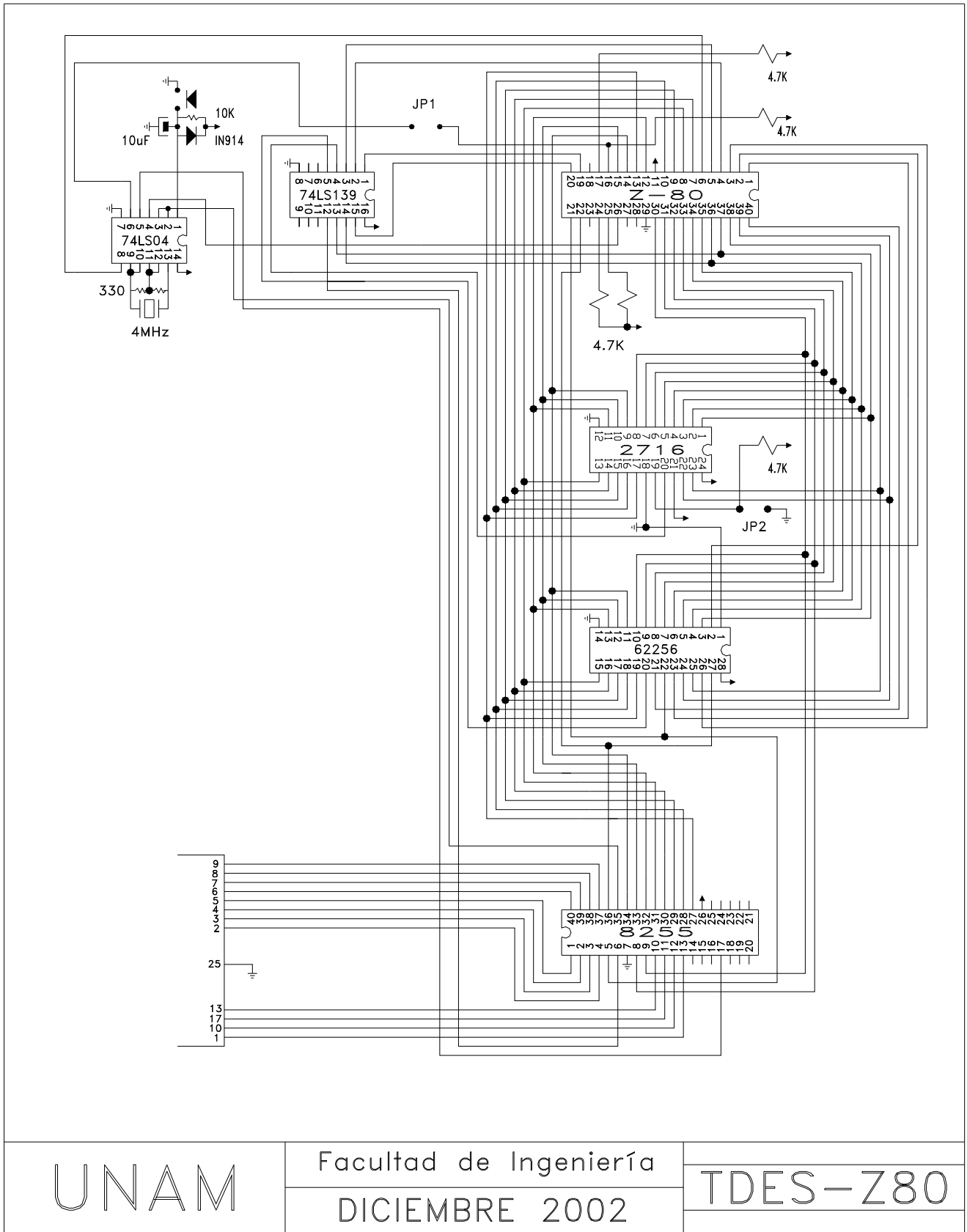
Control de temperatura con lógica difusa

```
    outp(0x37a,inp(0x37a)&0xf7);
    outp(0x378,0x00);
inicio:
    clrscr();
    gotoxy(36,1);printf("TDES-Z80");
    gotoxy(65,20);printf("Perif,rico");
    dr();
    do
    {
        gotoxy(65,23);
        if(ban==1)printf("SIN EJECUCION");
            else printf("EJECUTANDO          ");
        gotoxy(1,24);printf("Z80> ");
        op=getche();if(bioskey(1)!=0)getch();
        if(op=='c' || op=='C')
        {
            op=getche();if(bioskey(1)!=0)getch();
            if(op=='c' || op=='C')cc();
            if(op=='m' || op=='M')cm();
            if(op=='p' || op=='P')cp();
        }
        if(op=='d' || op=='D')
        {
            op=getche();
            if(op=='m' || op=='M')dm();
            if(op=='p' || op=='P')dp();
            if(op=='r' || op=='R')dr();
        }
        if(op=='e' || op=='E')ex();
        if(op=='p' || op=='P'){lissal(ST);ban=1;}
        if(op=='l' || op=='L')goto inicio;
        if(op=='h' || op=='H')
        {
            clrscr();
            printf("Cargar codigo          CC\n");
            printf("Cargar memoria          CM\n");
            printf("Cargar perif,rico       CP\n");
            printf("Desplegar memoria       DM\n");
            printf("Desplegar perif,rico    DP\n");
            printf("Desplegar registros     DR\n");
            printf("Ejecutar programa       E \n");
            printf("Parar programa          P \n");
            printf("Limpiar pantalla        L \n");
            printf("Salir                    S \n");
            getch();goto inicio;
        }
        if(op=='s' || op=='S'){printf("    Seguro? (s/n)");op=getch();}
        gotoxy(1,24);
        printf("
            ");
    }
    while(op!='s'&&op!='S');
    clrscr();
}
```

Con esto ésta tarjeta de desarrollo se obtiene un gran dominio para el desarrollo del sistema que se realizo y para cualquier otro que se base en Z-80 ya que se ideo como un sistema abierto y no dedicado a una sola aplicación.

Para poder realizar aplicaciones se contó con el ensamblador ASMZ80 que genera archivos con el formato INTEL-HEX.

El diagrama de TDESZ80 se muestra en la siguiente hoja.



UNAM

Facultad de Ingeniería
DICIEMBRE 2002

TDES-Z80

APENDICE C

En este apartado se enlista el programa desarrollado para el presente trabajo, el cual ya cuenta con las rutinas de la lógica difusa, el control del LCD, control de actuadores y de teclado.

Se compilo con el programa ASMZ80.COM el cual general el presente listado.

Al final del apéndice se encuentra el diagrama completo del sistema.

Control de temperatura con lógica difusa

```

0000: ;*****
0000: ;*Este programa se encarga de la interfaz con el usuario y el control del *
0000: ;*teclado, es el programa principal y de aqui parten los demas subprogramas *
0000: ;*****
0000: ;//////////////////// Lugar de las constantes del sistema //////////////////////
0000:
0B00: .ORG 0B00H
0B00: COMLCD EQU *
0B00: 38 0C 06 01 .DBYTE 380CH,0601H;Comandos para programar el LCD
0B04: CODENTER EQU * ;Datos para la generaci3n de caracteres especiales del LCD
0B04: 01 01 01 05 .DBYTE 0101H,0105H,091FH,0804H,0609H,0906H,0000H,0000H
0B08: 09 1F 08 04
0B0C: 06 09 09 06
0B10: 00 00 00 00
0B14:
0B14: ;Estos valores contienen el numero de salidas, entradas y conjuntos del
0B14: ;sistema
0B14: SAENCOP EQU *
0B14: 02 02 05 .BYTE 02H,02H,05H
0B17:
0B17: ;Estos son los puntos de los conjuntos de entrada. En este caso se cuentan
0B17: ;con 5 conjuntos por entrada y son dos entradas por que son 40 datos
0B17: C1PLEIP EQU *
0B17: 00 00 2A 54 .DBYTE 0000H,2A54H,2A54H,5480H,5480H,80AEH,80AEH,0AED6H,0AED6H,0FFFFH
0B1B: 2A 54 54 80
0B1F: 54 80 80 AB
0B23: 80 AB AB D6
0B27: AB D6 FF FF
0B2B: 00 00 2A 54 .DBYTE 0000H,2A54H,2A54H,5480H,5480H,80AEH,80AEH,0AED6H,0AED6H,0FFFFH
0B2F: 2A 54 54 80
0B33: 54 80 80 AB
0B37: 80 AB AB D6
0B3B: AB D6 FF FF
0B3F: ;Estos son los puntos de los singletons de salida, son dos salidas
0B3F: ;y 5 singletons por cada una. Da un total de 10 datos
0B3F: SINGSLP EQU *
0B3F: 00 3F 7F BE .BYTE 00H,3FH,7FH,0BEH,0FFH
0B43: FF
0B44: 4D 6A 87 A4 .BYTE 4DH,6AH,87H,0A4H,0C2H
0B48: C2
0B49:
0B49: ;En este lugar se colocan las reglas para evaluar
0B49: REGLASP EQU * ;Las reglas terminan cuando el bit7 = 1
0B49: 00 08 20 2C .DBYTE 0008H,202CH,0108H,202BH,0208H,212BH,0308H,222AH,0408H,2329H
0B4D: 01 08 20 2B
0B51: 02 08 21 2B
0B55: 03 08 22 2A
0B59: 04 08 23 29
0B5D: 00 09 20 2C .DBYTE 0009H,202CH,0109H,212BH,0209H,222AH,0309H,2329H,0409H,2429H
0B61: 01 09 21 2B
0B65: 02 09 22 2A
0B69: 03 09 23 29
0B6D: 04 09 24 29
0B71: 00 0A 20 2C .DBYTE 000AH,202CH,010AH,212BH,020AH,222AH,030AH,2329H,040AH,2429H
0B75: 01 0A 21 2B
0B79: 02 0A 22 2A
0B7D: 03 0A 23 29
0B81: 04 0A 24 29
0B85: 00 0B 21 2B .DBYTE 000BH,212BH,010BH,222AH,020BH,2329H,030BH,2429H,040BH,2428H
0B89: 01 0B 22 2A
0B8D: 02 0B 23 29
0B91: 03 0B 24 29
0B95: 04 0B 24 28
0B99: 00 0C 22 2A .DBYTE 000CH,222AH,010CH,232AH,020CH,2429H,030CH,2428H,040CH,2428H
0B9D: 01 0C 23 2A
0BA1: 02 0C 24 29
0BA5: 03 0C 24 28
0BA9: 04 0C 24 28
0BAD: FF .BYTE 0FFH ;Fin de las reglas
0BAE: ;Los siguientes datos son los letreros que se utilizan en el sistema
0BAE: MODO EQU *

```

Control de temperatura con lógica difusa

```

OBAA: 20 4F 50 45          .BYTE " OPERAR EN MODO "
OBB2: 52 41 52 20
OBB6: 45 4E 20 4D
OBBA: 4F 44 4F 20
OBEE:          AUTOMATI EQU *
OBEE: 20 20 20 41          .BYTE " AUTOMATICO "
OBC2: 55 54 4F 4D
OBC6: 41 54 49 43
OBCA: 4F 20 20 20
OBCE:          ; .DBYTE OFEFEH,0E545H,0FE53H,32FEH,0FEFEH,0FEFEH,0FEFEH,0FEFEH ;DEBUG
OBCE:          AUTOINCO EQU *
OBCE: 20 20 20 01          .BYTE " ",01H,"C ","AUTO " ,01H,"C "
OBD2: 43 20 41 55
OBD6: 54 4F 20 20
OBDA: 20 01 43 20
OBDE:          MANUAL EQU *
OBDE: 20 20 20 20          .BYTE " MANUAL "
OBE2: 20 4D 41 4E
OBE6: 55 41 4C 20
OBEA: 20 20 20 20
OBEE:          MANUINCO EQU *
OBEE: 20 20 20 01          .BYTE " ",01H,"C ","MANU " ,01H,"C "
OBF2: 43 20 4D 41
OBF6: 4E 55 20 20
OBFA: 20 01 43 20
OBFE:          INACTIVO EQU *
OBFE: 20 20 20 20          .BYTE " INACTIVO "
OC02: 49 4E 41 43
OC06: 54 49 56 4F
OC0A: 20 20 20 20
OC0E:          INACINCO EQU *
OC0E: 20 20 20 01          .BYTE " ",01H,"C ","INAC " ,01H,"C "
OC12: 43 20 49 4E
OC16: 41 43 20 20
OC1A: 20 01 43 20
OC1E:          PROGRAMA EQU *
OC1E: 20 20 20 20          .BYTE " PROGRAMA "
OC22: 50 52 4F 47
OC26: 52 41 4D 41
OC2A: 20 20 20 20
OC2E:          PRINCI1 EQU *
OC2E: 54 65 6D 70          .BYTE "Temp. Act. " ,01H,"C"
OC32: 2E 20 41 63
OC36: 74 2E 20 20
OC3A: 20 20 01 43
OC3E:          ; .DBYTE OFEFEH,45FEH,0FE53H,31FEH,0FEFEH,0FEFEH,0FEFEH,0143H ;DEBUG
OC3E:          PRINCI2 EQU *
OC3E: 54 20 69 6E          .BYTE "T int. T ext."
OC42: 74 2E 20 20
OC46: 20 20 54 20
OC4A: 65 78 74 2E
OC4E:          CAMBIAR EQU *
OC4E: 20 20 20 43          .BYTE " CAMBIAR DE "
OC52: 41 4D 42 49
OC56: 41 52 20 44
OC5A: 45 20 20 20
OC5E:          DESPLIE EQU *
OC5E: 20 20 20 44          .BYTE " DESPLIEGUE "
OC62: 45 53 50 4C
OC66: 49 45 47 55
OC6A: 45 20 20 20
OC6E:          RANGOS EQU *
OC6E: 20 20 20 20          .BYTE " RANGOS "
OC72: 20 52 41 4E
OC76: 47 4F 53 20
OC7A: 20 20 20 20
OC7E:          PREPRO EQU *
OC7E: 20 20 50 52          .BYTE " PRED.",7EH,"PROG. "
OC82: 45 44 2E 7E
OC86: 50 52 4F 47
OC8A: 2E 20 20 20

```

Control de temperatura con lógica difusa

```

0C8E:          PROPRE      EQU *
0C8E: 20 20 50 52          .BYTE " PROG.",7EH,"PRED.  "
0C92: 4F 47 2E 7E
0C96: 50 52 45 44
0C9A: 2E 20 20 20
0C9E:          CAMANU1    EQU *
0C9E: 53 61 6C 69          .BYTE "Salida 1:  % CT"
0CA2: 64 61 20 31
0CA6: 3A 20 20 20
0CAA: 25 20 43 54
0CAE:          CAMANU2    EQU *
0CAE: 28 2D 29 20          .BYTE "(-) (+)  (",7FH,") (",7EH,")"
0CB2: 28 2E 29 20
0CB6: 20 28 7F 29
0CBA: 20 28 7E 29
0CBE:          CAMANU3    EQU *
0CBE: 28 2D 29 20          .BYTE "(-) (+)  (",00H,") (",7EH,")"
0CC2: 28 2E 29 20
0CC6: 20 28 00 29
0CCA: 20 28 7E 29
0CCE:          LAUTO      EQU *
0CCE: 54 65 6D 70          .BYTE "Temperatura:  ",01H,"C"
0CD2: 65 72 61 74
0CD6: 75 72 61 3A
0CDA: 20 20 01 43
0CDE:          LENIRA     EQU *
0CDE: 23 20 45 4E          .BYTE "# ENTRADAS:  "
0CE2: 54 52 41 44
0CE6: 41 53 3A 20
0CEA: 20 20 20 20
0CEE:          LSALID     EQU *
0CEE: 23 20 53 41          .BYTE "# SALIDAS:  "
0CF2: 4C 49 44 41
0CF6: 53 3A 20 20
0CFA: 20 20 20 20
0CFE:          LCONJU     EQU *
0CFE: 23 20 43 4F          .BYTE "# CONJUNTOS:  "
0D02: 4E 4A 55 4E
0D06: 54 4F 53 3A
0DOA: 20 20 20 20
0DOE:          AJENIR     EQU *
0DOE: 41 6A 75 73          .BYTE "Ajusta Entradas "
0D12: 74 61 20 45
0D16: 6E 74 72 61
0D1A: 64 61 73 20
0D1E:          AJSALI     EQU *
0D1E: 41 6A 75 73          .BYTE "Ajusta Salidas "
0D22: 74 61 20 53
0D26: 61 6C 69 64
0D2A: 61 73 20 20
0D2E:          AJREGL     EQU *
0D2E: 41 6A 75 73          .BYTE "Ajusta Reglas  "
0D32: 74 61 20 20
0D36: 52 65 67 6C
0D3A: 61 73 20 20
0D3E:          AJCEN1     EQU *
0D3E: 45 20 43 20          .BYTE "E C P1:  P2:  "
0D42: 20 50 31 3A
0D46: 20 20 20 50
0D4A: 32 3A 20 20
0D4E:          AJCEN2     EQU *
0D4E: 50 33 3A 20          .BYTE "P3:  P4:  ",00H," ",7EH
0D52: 20 20 50 34
0D56: 3A 20 20 20
0D5A: 20 00 20 7E
0D5E:          AJSAL      EQU *
0D5E: 53 20 20 53          .BYTE "S SINGLETON :  "
0D62: 49 4E 47 4C
0D66: 45 54 4F 4E
0D6A: 20 3A 20 20
0D6E:          AJREG      EQU *

```


Control de temperatura con lógica difusa

```

OD6E: 52 45 47 4C      .BYTE "REGLA  VAL:  "
OD72: 41 20 20 20
OD76: 20 56 41 4C
OD7A: 3A 20 20 20
OD7E:      LCONREG  EQU *
OD7E: 28 2D 29 20      .BYTE "(-) (+)  (" ,00H,") SAL"
OD82: 28 2B 29 20
OD86: 20 28 00 29
OD8A: 20 53 41 4C
OD8E:
OD8E:      ;///////////////////////////////// Definición de variables ///////////////////////////////////
OD8E:
OD8E:      BASEFUZ  EQU 1200H      ;Lugar de memoria RAM del Kernel
OD8E:      BLKPROM  EQU 200H       ;Lugar de variables
OD8E:
OD8E:      FUZIN    EQU BASEFUZ    ;Las entradas van desde XX00H a XX1FH
OD8E:
OD8E:      FUZOUT   EQU BASEFUZ+20H;Las salidas van desde XX20H a XX3FH
OD8E:
OD8E:      FXZS1L   EQU BASEFUZ+40H;Suma de u(z)z de las salidas
OD8E:      ;Va desde XX40H a XX47H
OD8E:
OD8E:      CICLOS1  EQU BASEFUZ+48H;Lugar donde se ponen los resultado
OD8E:      ;Va desde XX48H a XX4BH
OD8E:      ;De XX4CH a XX4FH esta CONAUX
OD8E:
OD8E:      BLKENT   EQU BASEFUZ+50H;Bloque de entradas que va desde
OD8E:      ;Va desde XX50H hasta XX53H
OD8E:
OD8E:      CIMLE1   EQU BASEFUZ+54H;Lugar de las pendientes calculadas
OD8E:
OD8E:      TEMSEL   EQU BASEFUZ+BLKPROM;Temperatura seleccionada
OD8E:      NUMSAL   EQU BASEFUZ+BLKPROM+1;# de salidas
OD8E:      NUMENT   EQU BASEFUZ+BLKPROM+2;# de entradas
OD8E:      NUMCON   EQU BASEFUZ+BLKPROM+3;# de conjuntos (antecedentes como con
OD8E:      C1PLE1   EQU BASEFUZ+BLKPROM+4;Lugar de los puntos
OD8E:      SINGS1   EQU BASEFUZ+BLKPROM+85H;Singletons
OD8E:      REGLAS   EQU BASEFUZ+BLKPROM+0A6H;Reglas
OD8E:      RESPAL   EQU REGLAS+100H
OD8E:      ;-----
OD8E:      VAR1L    EQU 0           ;-----Estas variables se ocupan en la-----
OD8E:      VAR1H    EQU 1           ;----- multiplicación y en la -----
OD8E:      VAR2L    EQU 2           ;----- división. -----
OD8E:      VAR2H    EQU 3           ;-----
OD8E:      VAR3L    EQU 4           ;-----Resultado BAJO -----
OD8E:      VAR3H    EQU 5           ;-----Resultado ALTO -----
OD8E:      ;-----
OD8E:      GENL     EQU 6           ;Variable de uso general
OD8E:
OD8E:      VART1    EQU 7           ;-----
OD8E:      VART2    EQU 8           ;-----Variables de tiempo de espera-----
OD8E:
OD8E:      ;B_M=1H AUTOMATICO B_M=2H MANUAL B_M=4H INACTIVO B_M=8H PROGRAMA
OD8E:      B_T      EQU 9           ;Bandera de teclado
OD8E:      B_M      EQU 10          ;Bandera de modo
OD8E:
OD8E:      FCONT1   EQU 11          ;Bandera de control 1
OD8E:      FCONT2   EQU 12          ;Bandera de control 2
OD8E:
OD8E:      TECLAT   EQU 13          ;Variable tecla temporal
OD8E:
OD8E:      ;Contadores para la entrada de datos
OD8E:      CONENT   EQU 14          ;Contador de entrada
OD8E:      CONSAL   EQU 15          ;Contador de salida
OD8E:      CONCON   EQU 16          ;Contador de conjuntos
OD8E:      CONPUN   EQU 17          ;Contador de puntos
OD8E:
OD8E:      ARRANL   EQU 18          ;Variable para retardar incremento o decremento
OD8E:      DIR      EQU 19          ;Indica el lugar de despliegue del LCD
OD8E:      HEXA     EQU 20          ;Vandera que indica si la operación es en hexa o decim
OD8E:      TOPE     EQU 21          ;Indica hasta que valor llegara alguna variable

```

Control de temperatura con lógica difusa

```

OD8E:      CONFF2 EQU 22      ;Variable que retarda la lectura de entradas
OD8E:      ERROR EQU 23      ;Variable en la cual se guarda el error
OD8E:      ERRORS EQU 24     ;Variable de signo del error
OD8E:
OD8E:      ;//////////////////// Definición de constantes //////////////////////
OD8E:
OD8E:      P4 EQU 3           ;Punto cuatro
OD8E:      FUZH EQU 65H      ;12H;Parte alta de FUZIN y FUZOUT
OD8E:      FILTRAT EQU 0FH   ;Filtro de teclado
OD8E:      MENOS EQU 1       ;Valor correspondiente a la tecla menos
OD8E:      MAS EQU 2         ;Valor correspondiente a la tecla mas
OD8E:      ENTER EQU 4       ;Valor correspondiente a la tecla enter
OD8E:      SELEC EQU 8       ;Valor correspondiente a la tecla selección
OD8E:      LIMPIA EQU 1      ;Codigo para limpiar el LCD
OD8E:      LIN_DOS EQU 0COH  ;Codigo para pasar a la linea dos del LCD
OD8E:      CONAUX EQU 4      ;Desplazamiento para auxiliares
OD8E:      VALBP EQU MODO-SAENCOP;# de datos a pasar a la RAM
OD8E:      RETLL EQU 10H     ;Retorna un lugar el cursor a la izquierda
OD8E:      FACTOR EQU 64H    ;Factor para convertir de Hexa a Dec.
OD8E:      DIV255 EQU 74H    ;82H ;Divisor de frecuencia para el cont1 del 8254
OD8E:      DIVINI EQU 0FFH   ;El valor inicial para el cont2 y cont3 del 8254
OD8E:      WORDCON1 EQU 3EH  ;Palabra de control para el cont1 del 8254
OD8E:      WORDCON2 EQU 0BAH  ;Palabra de control para el cont2 del 8254
OD8E:      WORDCON3 EQU 7AH   ;Palabra de control para el cont3 del 8254
OD8E:      P8254 EQU 80H     ;Salida (Número de periférico para el 8254)
OD8E:      TECLA EQU 40H     ;Numero correspondiente al Teclado
OD8E:      LCD EQU 00H       ;Numero correspondiente al LCD
OD8E:      ADC1 EQU 2000H    ;Valor correspondiente al ADC1
OD8E:      ADC2 EQU 3000H    ;Valor correspondiente al ADC2
OD8E:
OD8E:      INDEXX EQU 1100H   ;Lugar de variables
OD8E:
OD8E:      ;***** INICIO *****
OD8E:
OD8E:      .ORG 0000H
0000:
0000: 31 FF 10 LD SP,10FFH ;Apuntador de pila
0003: DD 21 00 11 LD IX,INDEXX ;Lugar de variables y registros a utilizar
0007:
0007: DD 36 0B 00 LD (IX+FCONT1),00H;Inicia banderas de control
000B: DD 36 0C 00 LD (IX+FCONT2),00H
000F: DD 36 18 00 LD (IX+ERRORS),00H
0013: DD 36 17 00 LD (IX+ERROR),00H
0017: DD 36 0A 01 LD (IX+B_M),01H;Inicia registro B_M = 01H
001B: CD 99 08 CALL DES8254
001E:
001E: CD 9D 00 CALL BLKDATP ;Pasa el bloque de EPROM a la RAM
0021:
0021: 01 00 04 LD BC,LCD+400H ;Cargar el periférico y #de datos para el LCD
0024: 21 00 0B LD HL,COMLCD ;Lugar de comandos para el LCD
0027: CD 89 09 SIG: CALL LET_LIS ;LCD listo para recibir
002A: ED A3 OUTI ;Has una transferencia
002C: 20 F9 JR NZ,SIG
002E: CD 89 09 CALL LET_LIS ;Prepara la carga de nuevos caracteres
0031: 3E 40 LD A,40H ;al LCD correspondientes a los códigos
0033: D3 00 OUT (LCD),A ; 00H y 01H
0035:
0035: 01 01 10 LD BC,LCD+1001H;Carga los nuevos caracteres
0038: 21 04 0B LD HL,CODENTER
003B: CD 89 09 SIGDOS: CALL LET_LIS
003E: ED A3 OUTI
0040: 20 F9 JR NZ,SIGDOS
0042:
0042:
0042: DD 7E 0A LA: LD A,(IX+B_M) ;Carga al registro B_T con
0045: DD 77 09 LD (IX+B_T),A ;lo que tiene B_M
0048: DD 36 0C 00 LD (IX+FCONT2),00H; TEMP = 0
004C: CD 1A 05 CALL LET_PRIN ;Llama al letrero principal
004F: DB 40 INDATA: IN A,(TECLA) ;Obtiene el dato del teclado
0051: E6 0F AND FILTRAT ;quitando los cuatro últimos bits

```

Control de temperatura con lógica difusa

```

0053: 28 33          JR Z,CON          ;Si A=0 -> CON
0055: DD 77 0D          LD (IX+TECLAT),A;Si a!=0 guarda el valor de la tecla
0058: DD CB 0C 7E        BIT 7,(IX+FCOINT2);Esta en PROGRAMA?
005C: 28 05          JR Z,NPPROG      ;si es asi llama a la funcion programa
005E: CD 38 02          CALL PPROG       ;si no entonces compara con la siguiente
0061: 18 1C          JR TECO
0063: DD 7E 0A  NPPROG: LD A,(IX+B_M)    ;Compara si es AUTOMATICO
0066: FE 01          CP 01H
0068: 20 05          JR NZ,NPAUTO
006A: CD EF 00          CALL PAUTO
006D: 18 10          JR TECO
006F: FE 02  NPAUTO: CP 02H          ;Compara si es MANUAL
0071: 20 05          JR NZ,NPMANU
0073: CD 53 01          CALL PMANU
0076: 18 07          JR TECO
0078: FE 04  NPMANU: CP 04H          ;Esta comparacion corresponde a INACTIVO pero
007A: 20 03          JR NZ,TECO      ;como solo afecta a la vanderla se procedera
007C: CD EF 00          CALL PAUTO      ;con AUTOMATICO
007F:
007F: CD F2 05  TECO:   CALL CONTROL    ;Llama a control
0082: DB 40          IN A,(TECLA)    ;Espera a que se desactive la tecla
0084: E6 0F          AND FILTRAT
0086: 20 F7          JR NZ,TECO
0088:
0088: CD F2 05  CON:     CALL CONTROL
008B: DD CB 0C 46        BIT 0,(IX+FCOINT2)
008F: 28 BE          JR Z,INDATO
0091: DD 35 07          DEC (IX+VART1)  ;***** VARIABLES DE ESPERA *****
0094: 20 B9          JR NZ,INDATO    ;Sirven para dejar un determinado
0096: DD 35 08          DEC (IX+VART2) ;tiempo el letrero y esperando tambi,n
0099: 20 B4          JR NZ,INDATO    ;una respuesta de parte del usuario
009B: 18 A5          JR LA
009D:
009D: ;***** BLKDATP *****
009D:
009D: C5          BLKDATP:  PUSH BC
009E: D5          PUSH DE
009F: E5          PUSH HL          ;-----
00A0: 21 14 0B        LD HL,SAENCOOP ;-- Esta función se encarga de --
00A3: 11 01 14        LD DE,NUMSAL   ;-- pasar el bloque predetermi_--
00A6: 01 2B 00        LD BC,SINGS1P-SAENCOOP;-- nado de datos grabados en --
00A9: ED B0          LDIR           ;-- la EPROM a la RAM.          --
00AB: 21 3F 0B        LD HL,SINGS1P ;-----
00AE: 11 85 14        LD DE,SINGS1
00B1: 01 0A 00        LD BC,REGLASP-SINGS1P
00B4: ED B0          LDIR
00B6: 21 49 0B        LD HL,REGLASP
00B9: 11 A6 14        LD DE,REGLAS
00BC: 01 65 00        LD BC,MODO-REGLASP
00BF: ED B0          LDIR
00C1: E1          POP HL
00C2: D1          POP DE
00C3: C1          POP BC
00C4: CD 9D 06        CALL PENDIEN    ;Calcula las pendientes preterminadas
00C7: C9          RET
00C8:
00C8: ;***** BLKRAM *****
00C8:
00C8: C5          BLKRAM:  PUSH BC
00C9: D5          PUSH DE
00CA: E5          PUSH HL          ;-----
00CB: 21 01 14        LD HL,NUMSAL   ;-- Esta función se encarga de --
00CE: 11 A6 15        LD DE,RESPAL   ;-- pasar el bloque de desplie_--
00D1: 01 A5 01        LD BC,RESPAL-NUMSAL;-- gue de RAM a otra parte de --
00D4: ED B0          LDIR           ;-- la RAM.          --
00D6: E1          POP HL          ;-----
00D7: D1          POP DE
00D8: C1          POP BC
00D9: C9          RET
00DA:
00DA: ;***** RAMBLK *****

```

Control de temperatura con lógica difusa

```

00DA:
00DA: C5      RAMBLK:  PUSH BC
00DB: D5      PUSH DE
00DC: E5      PUSH HL      ;-----
00DD: 21 A6 15 LD HL,RESPAL ;-- Esta función se encarga de --
00E0: 11 01 14 LD DE,NUMSAL ;-- pasar el bloque de la RAM --
00E3: 01 A5 01 LD BC,RESPAL-NUMSAL;-- a la RAM de despligue --
00E6: ED B0      LDIR ;-----
00E8: E1      POP HL
00E9: D1      POP DE
00EA: C1      POP BC
00EB: CD 9D 06 CALL PENDIEN ;Calcula las pendientes programadas
00EE: C9      RET
00EF:
00EF:          ;***** PAUIO *****
00EF:
00EF: DD 7E 0D PAUIO:  LD A,(IX+TECLAT);Recupera la tecla y verifica
00F2: FE 04      CP ENTER      ;si se trata de la tecla ENTER
00F4: 28 14      JR Z,PAUTOCON ;entonces salta
00F6:
00F6: FE 08      PAUTOSEL: CP SELEC      ;si se trata de la tecla SELEC
00F8: C0      RET NZ      ;continua de lo contrario regresa.
00F9: DD CB 09 06 RLC (IX+B_T) ;Se pone en el siguiente modo pero
00FD: DD CB 0C C6 SET 0,(IX+FCONT2);sin activarlo. Activa el tiempo.
0101: CD 32 05 CALL LET_B_T ;despliega el modo a seleccionar
0104: 06 05      LD B,05H
0106: CD EA 05 CALL PON_TIE
0109: C9      RET
010A: DD CB 0C 46 PAUTOCON: BIT 0,(IX+FCONT2)
010E: 28 1F      JR Z,SIPAUTO
0110:
0110: DD 7E 09 PAUTOSIG: LD A,(IX+B_T) ;Coloca el nuevo modo activo
0113: FE 08      CP 08H      ;excepto si es PROGRAMA
0115: 28 0B      JR Z,VAPROG ;salta
0117: DD 77 0A      LD (IX+B_M),A
011A: DD CB 0C 86 PAUTOREG: RES 0,(IX+FCONT2);Limpia tiempo
011E: CD 1A 05 CALL LET_PRIN
0121: C9      RET
0122: 06 05      VAPROG:  LD B,05H      ;Aqui se pone activo el tempo de espera
0124: CD EA 05 CALL PON_TIE ;ademas de que se pone el letrero
0127: DD CB 0C FE SET 7,(IX+FCONT2);correspondiente a la seleccion
012B: CD A6 05 CALL LETTEC ;esta tambi,n en modo PROGRAMA
012E: C9      RET
012F: DD 36 14 00 SIPAUTO: LD (IX+HEXA),0 ;No se trata de un valor hexadecimal
0133: 3E 01      LD A,LIMPIA ;Limpia el LCD para desplegar
0135: D3 00      OUT (LCD),A ;los letreros correspondientes
0137: 21 CE 0C LD HL,LAUTO
013A: CD 96 05 CALL DESPLIEG
013D: 3E C0      LD A,LIN_DOS
013F: D3 00      OUT (LCD),A
0141:
0141: DD 36 13 8D LD (IX+DIR),8DH;DDRAM para los digitos
0145: 21 BE 0C LD HL,CAMANU3
0148: CD 96 05 CALL DESPLIEG
014B:
014B: 21 00 14 LD HL,TEMSEL ;Lugar donde se tiene la temperatura
014E: CD BF 01 CALL TECLAON ;seleccionada
0151:
0151: 18 C7      JR PAUTOREG
0153:
0153:          ;***** PMANU *****
0153:
0153: DD 7E 0D PMANU:  LD A,(IX+TECLAT);Recupera la tecla y verifica
0156: FE 04      CP ENTER      ;si se trata de la tecla ENTER
0158: 28 02      JR Z,PMANUSIG ;entonces salta
015A: 18 9A      JR PAUTOSEL
015C:
015C: DD CB 0C 46 PMANUSIG: BIT 0,(IX+FCONT2);Verifica si se encuentra en seleccción
0160: 28 02      JR Z,SIMANUAL
0162: 18 AC      JR PAUTOSIG
0164:

```

Control de temperatura con lógica difusa

```

0164: DD 36 0F 31 SIMANUAL: LD (IX+CONSAL),31H;Pone el valor de 1 al contador de salida
0168: DD 36 14 00 LD (IX+HEXA),0 ;No se trata de un valor hexadecimal
016C: 3E 01 LD A,LIMPIA ;Limpia el LCD para desplegar
016E: D3 00 OUT (LCD),A ;los letreros correspondientes
0170: 21 9E 0C LD HL,CAMANU1
0173: CD 96 05 CALL DESPLIEG
0176: 3E C0 LD A,LIN_DOS
0178: D3 00 OUT (LCD),A
017A:
017A: DD 36 13 8B LD (IX+DIR),8BH;DDRAM para los digitos
017E: 21 AE 0C LD HL,CAMANU2
0181: CD 96 05 CALL DESPLIEG
0184:
0184: 21 4C 12 LD HL,CICLOSI+CONAUX;Lugar donde se tiene los ciclos de las salid
0187: 3A 01 14 PMANU1: LD A,(NUMSAL)
018A: C6 31 ADD A,31H ;Este tipo de comparacion se realiza
018C: 47 LD B,A ;para que no exceda el limite de salidas
018D: DD 7E 0F LD A,(IX+CONSAL)
0190: B8 CP B
0191: 28 87 JR Z,PAUTOREG
0193:
0193: 3E 87 PMANU2: LD A,87H ;DDRAM para el numero
0195: D3 00 OUT (LCD),A ;Actualiza DDRAM el numero actual
0197: CD 89 09 CALL LET_LIS
019A: DD 7E 0F LD A,(IX+CONSAL)
019D: D3 01 OUT (LCD+1),A
019F: CD 89 09 CALL LET_LIS
01A2: CD BF 01 CALL TECLACON
01A5:
01A5: DD 7E 0D LD A,(IX+TECLAT);Al regresar de la funcion
01A8: FE 04 CP ENTER ;compara si se trata de regresar
01AA: 20 0D JR NZ,PMANU3 ;un lugar
01AC: DD 7E 0F LD A,(IX+CONSAL)
01AF: FE 31 CP 31H
01B1: 28 E0 JR Z,PMANU2
01B3: DD 35 0F DEC (IX+CONSAL);Regresa un lugar atras
01B6: 2B DEC HL
01B7: 18 DA JR PMANU2
01B9: DD 34 0F PMANU3: INC (IX+CONSAL);En este caso se adelanta un lugar
01BC: 23 INC HL
01BD: 18 C8 JR PMANU1
01BF:
01BF: ;***** TECLACON *****
01BF:
01BF: CD 99 08 TECLACON: CALL DES8254 ;Desactiva las salidas ya que se cambiar algfn
01C2: DD 46 0A LD B,(IX+B_M) ;parametro
01C5: DD 36 0A 04 LD (IX+B_M),04H
01C9: 3E 0E LD A,0EH ;Se acativa el cursor
01CB: D3 00 OUT (LCD),A ;para resaltar el número que
01CD: CD 89 09 CALL LET_LIS ;cambiar
01D0: DD 7E 14 LD A,(IX+HEXA) ;Verifica si se trata de un número
01D3: FE 00 CP 0 ;hexadecimal o decimal
01D5: CC EF 08 CALL Z,DBINBCD ;Se coloca el primer número de memoria
01D8: C4 3A 09 CALL NZ,DHEXA ;ya sea decimal o hexadecimal
01DB: DB 40 TECCERO: IN A,(TECLA) ;Espera para la desactivacion de cualquier
01DD: E6 0F AND FILTRAT ;tecla
01DF: 20 FA JR NZ,TECCERO
01E1:
01E1: DD 36 12 0A TECLACO: LD (IX+ARRAN1),0AH;Inicia el arranque
01E5:
01E5: CD 00 0A TECLAIN: CALL TIEMPO_1
01E8: DB 40 IN A,(TECLA) ;Obtiene el dato del teclado
01EA: E6 0F AND FILTRAT ;quitando los cuatro últimos bits
01EC:
01EC: FE 08 CP SELEC ;Checa si se trata de alguna tecla de control
01EE: 28 3A JR Z,TECREG ;si es así retorna al programa anterior
01F0: FE 04 CP ENTER
01F2: 28 36 JR Z,TECREG
01F4:
01F4: DD BE 0D CP (IX+TECLAT) ;Comprueba si la tecla continua activa
01F7: 20 C8 JR NZ,TECACIAA

```

Control de temperatura con lógica difusa

```

01F9:
01F9: DD 7E 12          LD A,(IX+ARRAN1);Compara si arranque esta en cero
01FC: FE 00            CP 0
01FE: 28 08            JR Z,TECDECIN
0200:
0200: DD 35 12            DEC (IX+ARRAN1)
0203: 18 E0            JR TECLAIN
0205:
0205: DD 77 0D            TECACIAA: LD (IX+TECLAT),A;Guarda la tecla actual
0208: DD 7E 0D            TECDECIN: LD A,(IX+TECLAT)
020B: FE 02            CP MAS
020D: 28 06            JR Z,TECMAS ;Realiza la operacion correspondiente
020F: FE 01            CP MENOS ;de incremento o decremento
0211: 20 CE            JR NZ,TECLACO
0213: 35                DEC (HL)
0214: 35                DEC (HL)
0215: 34                TECMAS: INC (HL)
0216: DD 7E 14            LD A,(IX+HEXA)
0219: FE 00            CP 0
021B: CC EF 08            CALL Z,DBINBCD
021E: C4 3A 09            CALL NZ,DHEXA
0221: DD 7E 06            LD A,(IX+GEN1) ;Esta variable indica si el ultimo digito
0224: FE 00            CP 0 ;cambia ya que si no es asi se debe repetir
0226: 20 E0            JR NZ,TECDECIN ;la operaci3n de forma aut3noma para que
0228: 18 BB            JR TECLAIN ;exista un cambio en el despliegue
022A: DD 77 0D            TECREG: LD (IX+TECLAT),A
022D: 3E 0C            LD A,0CH
022F: D3 00            OUT (LCD),A
0231: CD 89 09            CALL LET_LIS
0234: DD 70 0A            LD (IX+B_M),B
0237: C9                RET
0238:
0238: ;***** PPROG *****
0238:
0238: DD 7E 0D            PPROG: LD A,(IX+TECLAT);Recupera la tecla presionada
023B: FE 04            CP ENIER
023D: 28 0C            JR Z,PPROGSIG
023F: FE 08            CP SELEC ;Compara si se trata de SELEC
0241: C0                RET NZ
0242: CD A6 05            CALL LETTEC ;Despliega entonces las siguientes
0245: 06 05            LD B,05H ;siguientes opciones
0247: CD EA 05            CALL PON_TIE
024A: C9                RET
024B:
024B: ;En esta parte del programa se maneja un
024B: DD CB 0C 76          PPROGSIG: BIT 6,(IX+FCONT2);submenu derivado del modo programa.
024F: CA C8 04            JP Z,PPN6 ;Verifica que se trata del menu o submenu
0252: ;programa
0252:
0252: ;Entra en el modo de subprograma
0252: DD 36 0C 00            LD (IX+FCONT2),0;Quita vanderas
0256: DD 36 14 FF            LD (IX+HEXA),0FFH
025A: 3E 01            LD A,LIMPIA ;Limpia el LCD para desplegar
025C: D3 00            OUT (LCD),A ;los letreros correspondientes
025E: CD 89 09            CALL LET_LIS
0261: 3E C0            LD A,LIN_DOS
0263: D3 00            OUT (LCD),A
0265: 21 AE 0C            LD HL,CAMANU2
0268: CD 96 05            CALL DESPLIEG
026B:
026B: 3E 02            PPCHECA: LD A,02H ;Regresa al LCD al inicio del letreo
026D: D3 00            OUT (LCD),A ;sin cambiar la DDRAM
026F: CD 89 09            CALL LET_LIS
0272: DD 36 15 04            LD (IX+TOPE),4 ;Se tendra un tope de 4
0276: DD 36 13 8C            LD (IX+DIR),8CH
027A: DD 7E 0C            LD A,(IX+FCONT2);Verifica en que opci3n de cambio de
027D: FE 00            CP 0 ;encuentra
027F: 20 0B            JR NZ,PPNOCERO
0281: 21 DE 0C            LD HL,LENIRA ;Se encuentra en entradas
0284: CD 96 05            CALL DESPLIEG ;y prepara lo necesario
0287: 21 02 14            LD HL,NUMENT ;para poder hacer los

```

Control de temperatura con lógica difusa

```

028A: 18 20          JR PPTTECCON   ;cambios
028C: FE 01          PPNOCERO: CP 1
028E: 20 0B          JR NZ,PPNOUNO ;Se encuentra en salidas
0290: 21 EE 0C          LD HL,LSALID  ;y prepara lo necesario
0293: CD 96 05          CALL DESPLIEG ;para poder hacer los
0296: 21 01 14          LD HL,NUMSAL  ;cambios
0299: 18 11          JR PPTTECCON
029B: FE 02          PPNOUNO: CP 2
029D: 20 29          JR NZ,PPNODOS ;Se encuentra en conjuntos
029F: DD 36 15 08       LD (IX+TOPE),8 ;donde el tope es de 8 y
02A3: 21 FE 0C          LD HL,LCONJU  ;prepara lo necesario para
02A6: CD 96 05          CALL DESPLIEG ;poder hacer los cambios
02A9: 21 03 14          LD HL,NUMCON
02AC:
02AC: CD BF 01          PPTTECCON: CALL TECLACON ;En este apartado se lleva a cabo
02AF: DD 7E 0D          LD A,(IX+TECLAT);el proceso de cambio de las variables
02B2: FE 04          CP ENTER     ;o bien la navegaci3n entre ellas
02B4: 28 06          JR Z,PPMENOS
02B6: DD 34 0C          PPMAS: INC (IX+FCONT2)
02B9: DD 34 0C          INC (IX+FCONT2)
02BC: DD 7E 0C          PPMENOS: LD A,(IX+FCONT2)
02BF: FE 00          CP 0
02C1: 28 A8          JR Z,PPCHECA
02C3: DD 35 0C          DEC (IX+FCONT2)
02C6: 18 A3          JR PPCHECA
02C8:
02C8: FE 03          PPNODOS: CP 3          ;En este caso el proceso cambia ligeramente
02CA: 20 08          JR NZ,PPNOIRES ;ya que es el proceso de cambio de conjuntos
02CC: 21 0E 0D          LD HL,AJENIR  ;de entrada
02CF: CD 96 05          CALL DESPLIEG
02D2: 18 17          JR PPTTECDIF
02D4: FE 04          PPNOIRES: CP 4
02D6: 20 08          JR NZ,PPNO4
02D8: 21 1E 0D          LD HL,AJSALI  ;Cambio de conjuntos de salida
02DB: CD 96 05          CALL DESPLIEG
02DE: 18 0B          JR PPTTECDIF
02E0: FE 05          PPNO4: CP 5
02E2: C2 EF 04          JP NZ,PPSIG
02E5: 21 2E 0D          LD HL,AJREGL  ;Cambio de reglas
02E8: CD 96 05          CALL DESPLIEG
02EB:
02EB: CD 99 08          PPTTECDIF: CALL DES8254 ;Desactiva salida
02EE: DB 40          PPTTECOT: IN A,(TECLA) ;Espera para la desactivacion de cualquier tecla
02F0: E6 0F          AND FILIRAT
02F2: 20 FA          JR NZ,PPTECOT
02F4:
02F4: CD 00 0A          PPTTECIN: CALL TIEMPO_1
02F7: DB 40          IN A,(TECLA) ;Obtiene el dato del teclado
02F9: E6 0F          AND FILIRAT  ;quitando los cuatro ultimos bits
02FB:
02FB: FE 08          CP SELEC     ;Checa si se trata de alguna tecla de control
02FD: 28 B7          JR Z,PPMAS   ;si no es asi incrementa o decrementa
02FF: FE 04          CP ENTER
0301: 28 B9          JR Z,PPMENOS
0303: FE 02          CP MAS
0305: 28 06          JR Z,PPAJUS
0307: FE 01          CP MENOS
0309: 28 02          JR Z,PPAJUS
030B: 18 E7          JR PPTTECIN
030D:
030D: DD 7E 0C          PPAJUS: LD A,(IX+FCONT2)
0310: FE 05          CP 5
0312: 20 5D          JR NZ,PPCOM4 ;Verifica si se trata de las reglas
0314: DD 36 0F 00       LD (IX+CONSAL),00H;Pone el valor de 0 al contador de reglas
0318: DD 36 14 FF       LD (IX+HEXA),0FFH;Se trata de un valor hexadecimal
031C: DD 36 15 FF       LD (IX+TOPE),0FFH;No tiene un tope
0320: 3E 01          LD A,LIMPIA  ;Limpia el LCD para desplegar
0322: D3 00          OUT (LCD),A ;los letreros correspondientes
0324: 21 6E 0D          LD HL,AJREG
0327: CD 96 05          CALL DESPLIEG
032A: 3E C0          LD A,LIN_DOS

```

Control de temperatura con lógica difusa

```

032C: D3 00          OUT (LCD),A
032E:
032E: 21 7E 0D      LD HL,LCONREG
0331: CD 96 05      CALL DESPLIEG
0334:
0334: 21 A6 14      LD HL,REGLAS ;Lugar donde se tienen las reglas
0337:
0337: 3E 86          PPRREG2: LD A,86H ;DDRAM para los digitos
0339: D3 00          OUT (LCD),A
033B: CD 89 09      CALL LET_LIS
033E: DD 7E 0F      LD A,(IX+CONSAL)
0341: E6 F0          AND 0F0H ;Aqui se realiza el despligue del conteo
0343: 0F             RRCA ;que se lleva en hexadecimal
0344: 0F             RRCA
0345: 0F             RRCA
0346: 0F             RRCA
0347: CD 7B 09      CALL CONVHEX
034A: DD 7E 0F      LD A,(IX+CONSAL)
034D: E6 0F          AND 0FH
034F: CD 7B 09      CALL CONVHEX
0352:
0352: DD 36 13 8E     LD (IX+DIR),8EH;Se prepara para hacer el cambio de reglas
0356: CD BF 01      CALL TECLAON ;Cambia la regla actual
0359:
0359: DD 7E 0D          LD A,(IX+TECLAT);Al regresar de la funcion
035C: FE 08          CP SELEC ;compara si se trata de salir de raglas
035E: 20 03          JR NZ,PPREG3 ;Nota: se debe de terminar poniendo
0360: C3 1A 01          JP PAUTOREG ;como fin el código FFH
0363: DD 34 0F          PPRREG3: INC (IX+CONSAL);En este caso se adelanta un lugar
0366: 23             INC HL
0367: DD 7E 0F          LD A,(IX+CONSAL)
036A: FE 00          CP 0 ;Se pueden acceder hasta 256 reglas
036C: CA 1A 01          JP Z,PAUTOREG
036F: 18 C6          JR PPRREG2
0371:
0371: FE 04          PPCOM4: CP 4
0373: C2 11 04          JP NZ,PPCOM3 ;Se trata de cambiar los conjuntos de salida
0376: DD 36 0E 31      LD (IX+CONENT),31H;Pone el valor de 1 al contador de salida
037A: DD 36 0F 31      LD (IX+CONSAL),31H;Pone el valor de 1 al contador de Singleton
037E: DD 36 15 FF      LD (IX+TOPE),0FFH;No tiene tope
0382: DD 36 14 FF      LD (IX+HEXA),0FFH;Se trata de un valor hexadecimal
0386: 3E 01          LD A,LIMPIA ;Limpia el LCD para desplegar
0388: D3 00          OUT (LCD),A ;los letreros correspondientes
038A: 21 5E 0D          LD HL,AJSAL
038D: CD 96 05          CALL DESPLIEG
0390: 3E C0          LD A,LIN_DOS
0392: D3 00          OUT (LCD),A
0394:
0394: DD 36 13 8E     LD (IX+DIR),8EH;DDRAM para los digitos
0398: 21 AE 0C          LD HL,CAMANU2
039B: CD 96 05          CALL DESPLIEG
039E:
039E: 21 85 14          LD HL,SINGS1 ;Lugar donde se tiene los SINGLETON
03A1:
03A1: 3A 03 14          PPSAL1: LD A,(NUMCON) ;Se compara para no revasar el
03A4: C6 31          ADD A,31H ;el número de conjuntos
03A6: 47             LD B,A
03A7: DD 7E 0E          LD A,(IX+CONENT)
03AA: B8             CP B
03AB: 20 14          JR NZ,PPSAL2
03AD: DD 36 0E 31      LD (IX+CONENT),31H;Respectivo a contador de SINGLETON
03B1: DD 34 0F          INC (IX+CONSAL)
03B4: 3A 01 14          LD A,(NUMSAL)
03B7: C6 31          ADD A,31H ;Este tipo de comparacion se realiza
03B9: 47             LD B,A ;para que no exceda el limite de salidas
03BA: DD 7E 0F          LD A,(IX+CONSAL)
03BD: B8             CP B
03BE: CA 1A 01          JP Z,PAUTOREG
03C1:
03C1: 3E 81          PPSAL2: LD A,81H ;DDRAM para el numero
03C3: D3 00          OUT (LCD),A ;Actualiza DDRAM el numero actual

```


Control de temperatura con lógica difusa

```

03C5: CD 89 09          CALL LET_LIS ;de conjunto y de salida.
03C8: DD 7E 0F          LD A,(IX+CONSAL)
03CB: D3 01             OUT (LCD+1),A
03CD: CD 89 09          CALL LET_LIS
03D0: 3E 8C             LD A,8CH
03D2: D3 00             OUT (LCD),A
03D4: CD 89 09          CALL LET_LIS
03D7: DD 7E 0E          LD A,(IX+CONENT)
03DA: D3 01             OUT (LCD+1),A
03DC: CD 89 09          CALL LET_LIS
03DF:
03DF: CD BF 01          CALL TECLACON ;Para cambiar el valor del conjunto
03E2:
03E2: DD 7E 0D          LD A,(IX+TECLAT);Al regresar de la funcion
03E5: FE 04             CP ENIER ;compara si se trata de regresar
03E7: 20 22             JR NZ,PPSAL3 ;un lugar
03E9: DD 7E 0E          LD A,(IX+CONENT)
03EC: FE 31             CP 31H
03EE: 28 06             JR Z,PPSALA4
03F0: DD 35 0E          DEC (IX+CONENT);Regresa un lugar atras
03F3: 2B                DEC HL
03F4: 18 CB             JR PPSAL2
03F6: DD 7E 0F          PPSALA: LD A,(IX+CONSAL)
03F9: FE 31             CP 31H
03FB: 28 C4             JR Z,PPSAL2
03FD: DD 35 0F          DEC (IX+CONSAL)
0400: 3A 03 14          LD A,(NUMCON)
0403: C6 30             ADD A,30H
0405: DD 77 0E          LD (IX+CONENT),A
0408: 2B                DEC HL
0409: 18 B6             JR PPSAL2
040B:
040B: DD 34 0E          PPSAL3: INC (IX+CONENT);En este caso se adelanta un lugar
040E: 23                INC HL
040F: 18 90             JR PPSAL1
0411:
0411: DD 36 0E 31        PPCOMB: LD (IX+CONENT),31H;Pone el valor de 1 al contador de entradas
0415: DD 36 10 31        LD (IX+CONCON),31H;Pone el valor de 1 al contador de conjuntos
0419:
0419: DD 36 15 FF          LD (IX+TOPE),0FFH;No tiene tope
041D: DD 36 14 FF          LD (IX+HEXA),0FFH;Se trata de un valor hexadecimal
0421: 3E 01             LD A,LIMPIA ;Limpia el LCD para desplegar
0423: D3 00             OUT (LCD),A ;los letreros correspondientes
0425: 21 3E 0D          LD HL,AJCEN1
0428: CD 96 05          CALL DESPLIEG
042B: 3E C0             LD A,LIN_DOS
042D: D3 00             OUT (LCD),A
042F:
042F: 21 4E 0D          LD HL,AJCEN2
0432: CD 96 05          CALL DESPLIEG
0435: 21 04 14          LD HL,C1P1E1
0438:
0438: DD 36 11 31        PPEM1: LD (IX+CONPUN),31H;Inicia con 1 el contador de puntos
043C: 3A 03 14          LD A,(NUMCON)
043F: C6 31             ADD A,31H
0441: 47                LD B,A
0442: DD 7E 10          LD A,(IX+CONCON);Compara el el valor de conjuntos
0445: B8                CP B
0446: 20 14             JR NZ,PPEN2
0448: DD 36 10 31        LD (IX+CONCON),31H;Reestablece a 1 el contador de conjuntos
044C: DD 34 0E          INC (IX+CONENT);Incrementa contador de entradas
044F: 3A 02 14          LD A,(NUMENT)
0452: C6 31             ADD A,31H ;Este tipo de comparacion se realiza
0454: 47                LD B,A ;para que no exceda el limite de entradas
0455: DD 7E 0E          LD A,(IX+CONENT)
0458: B8                CP B
0459: CA 1A 01          JP Z,PAUTOREG
045C:
045C: 3E 81             PPEM2: LD A,81H ;DDRAM para el numero
045E: D3 00             OUT (LCD),A ;Actualiza DDRAM el numero actual
0460: CD 89 09          CALL LET_LIS

```

Control de temperatura con lógica difusa

```

0463: DD 7E 0E          LD A, (IX+CONENT);Despliega el número de entrada
0466: D3 01             OUT (LCD+1),A
0468: CD 89 09          CALL LET_LIS
046B: 3E 83             LD A,83H
046D: D3 00             OUT (LCD),A
046F: CD 89 09          CALL LET_LIS
0472: DD 7E 10          LD A, (IX+CONCON);Despliega el número de conjunto
0475: D3 01             OUT (LCD+1),A
0477: CD 89 09          CALL LET_LIS
047A: CD F7 04          CALL PONPUN ;Despliega los puntos correspondientes
047D: DD 36 13 88      LD (IX+DIR),88H;Prepara el lugar para el primer punto
0481: 18 26             JR PPEN5
0483: DD 7E 11   PPEN6: LD A, (IX+CONPUN)
0486: FE 32             CP 32H
0488: 20 06             JR NZ,*+8
048A: DD 36 13 8E      LD (IX+DIR),8EH;Prepara el lugar para el segundo punto
048E: 18 19             JR PPEN5
0490: FE 33             CP 33H
0492: 20 06             JR NZ,*+8
0494: DD 36 13 C3      LD (IX+DIR),0C3H;Prepara el lugar para el tercer punto
0498: 18 0F             JR PPEN5
049A: FE 34             CP 34H
049C: 20 06             JR NZ,*+8
049E: DD 36 13 C9      LD (IX+DIR),0C9H;Prepara el lugar para el cuarto punto
04A2: 18 05             JR PPEN5
04A4: DD 34 10          INC (IX+CONCON)
04A7: 18 8F             JR PPEN1
04A9:
04A9: CD BF 01   PPEN5: CALL TECLACON ;Cambia el punto correspondiente
04AC:
04AC: DD 7E 0D          LD A, (IX+TECLAT);Al regresar de la funcion
04AF: FE 04             CP ENTER ;compara si se trata de avanzar
04B1: 20 06             JR NZ,PPEN3 ;un conjunto
04B3: 23             INC HL ;Puesto que fue Enter se avanza un punto
04B4: DD 34 11          INC (IX+CONPUN)
04B7: 18 CA             JR PPEN6
04B9:
04B9: DD 34 10   PPEN3: INC (IX+CONCON);En este caso se adelanta un conjunto
04BC: DD 7E 11          LD A, (IX+CONPUN)
04BF: 23   PPENHL: INC HL ;Aqui se realiza el avance de puntos
04C0: 3C             INC A ;para pasar a otro punto
04C1: FE 35             CP 35H
04C3: 20 FA             JR NZ,PPENHL
04C5: C3 38 04          JP PPEN1
04C8:
04C8: DD CB 0C 6E PPN6: BIT 5, (IX+FCONT2);En este caso se procesa parte del menu programa
04CC: 28 18             JR Z,PPCAMD
04CE: DD CB 0B 1E      RR (IX+FCONT1)
04D2: 3F             CCF
04D3: DD CB 0B 16      RL (IX+FCONT1)
04D7: DD CB 0B 46      BIT 0, (IX+FCONT1)
04DB: C4 C8 00          CALL NZ,BLKRAM
04DE: C4 9D 00          CALL NZ,BLKDATP
04E1: CC DA 00          CALL Z,RAMELK
04E4: 18 09             JR PPSIG
04E6: DD CB 0B 16 PPCAMD: RL (IX+FCONT1)
04EA: 3F             CCF
04EB: DD CB 0B 1E      RR (IX+FCONT1)
04EF: DD 36 0C 00 PPSIG: LD (IX+FCONT2),00H;Quita vanderas
04F3: CD 1A 05          CALL LET_PRIN
04F6: C9             RET
04F7:
04F7: ;***** Función que despliega los puntos de un conjunto *****
04F7:
04F7: DD 36 13 88 PPNFUN: LD (IX+DIR),88H;En este caso se ponen los lugares
04FB: CD 3A 09          CALL DHEXA ;correspondientes de los puntos
04FE: 23             INC HL ;y se despliegan
04FF: DD 36 13 8E      LD (IX+DIR),8EH
0503: CD 3A 09          CALL DHEXA ;No hay cambio alguno en ellos
0506: 23             INC HL
0507: DD 36 13 C3      LD (IX+DIR),0C3H

```

Control de temperatura con lógica difusa

```

050B: CD 3A 09          CALL DHEXA
050E: 23                INC HL
050F: DD 36 13 C9       LD (IX+DIR),0C9H
0513: CD 3A 09          CALL DHEXA
0516: 2B                DEC HL
0517: 2B                DEC HL
0518: 2B                DEC HL
0519: C9                RET
051A:
051A: ;*****FUNCION LETRERO PRINCIPAL*****
051A:
051A: DD 36 16 01 LET_PRIN: LD (IX+CONFF2),1
051E: 21 3E 0C          LD HL,PRINCI2 ;Lugar del letrero principal 2
0521: DD CB 0B 7E       BIT 7,(IX+FCONT1);Verifica el despliegue actual
0525: 20 03              JR NZ,PRINCID2
0527: 21 2E 0C          LD HL,PRINCI1 ;Lugar del letrero principal 1
052A: DD 7E 0A PRINCID2: LD A,(IX+B_M)
052D: DD 77 09          LD (IX+B_T),A
0530: 18 03              JR SALTA
0532:
0532: ;*****FUNCION LETRERO B_T*****
0532:
0532: 21 AE 0B LET_B_T: LD HL,MOD0 ;Lugar del letrero de modo
0535: F5 SALTA: PUSH AF
0536: 3E 01 LD A,LIMPIA ;Limpia el LCD
0538: D3 00 OUT (LCD),A
053A: CD 96 05 CALL DESPLIEG
053D: 3E C0 LD A,LIN_DOS ;Pasa a la linea dos del LCD
053F: D3 00 OUT (LCD),A
0541: DD 7E 09 LD A,(IX+B_T) ;Carga en A el reg. B_T
0544: DD CB 0C 46 BIT 0,(IX+FCONT2)
0548: 20 06 JR NZ,SIESLETC
054A: DD CB 0B 7E BIT 7,(IX+FCONT1);En este lugar se verifica el tipo de
054E: 20 2D JR NZ,NOESLETC ;despliegue actual
0550: FE 01 SIESLETC: CP 01H ;En el siguiente proceso
0552: 20 05 JR NZ,SIG1 ;corresponde al primer tipo
0554: 21 BE 0B AUTO: LD HL,AUTOMATI ;de despliegue donde sólo se
0557: 18 1F JR DESP ;tiene la temperatura interna en el LCD
0559: FE 02 SIG1: CP 02H
055B: 20 05 JR NZ,SIG2
055D: 21 DE 0B LD HL,MANUAL
0560: 18 16 JR DESP
0562: FE 04 SIG2: CP 04H
0564: 20 05 JR NZ,SIG3
0566: 21 FE 0B LD HL,INACTIVO
0569: 18 0D JR DESP
056B: FE 08 SIG3: CP 08H
056D: 28 06 JR Z,SIG4
056F: DD 36 09 01 LD (IX+B_T),01H
0573: 18 DF JR AUTO
0575: 21 1E 0C SIG4: LD HL,PROGRAMA
0578: CD 96 05 DESP: CALL DESPLIEG
057B: F1 POP AF
057C: C9 RET
057D: FE 01 NOESLETC: CP 01H ;En este caso se procesan los letreros
057F: 20 05 JR NZ,SIGLINCO ;que corresponden al segundo tipo
0581: 21 CE 0B LD HL,AUTOINCO ;de despliegue donde se visualiza
0584: 18 F2 JR DESP ;tanto la temperatura interna como externa
0586: FE 02 SIGLINCO: CP 02H
0588: 20 05 JR NZ,SIG2INCO
058A: 21 EE 0B LD HL,MANUINCO
058D: 18 E9 JR DESP
058F: FE 04 SIG2INCO: CP 04H
0591: 21 0E 0C LD HL,INACINCO
0594: 18 E2 JR DESP
0596:
0596: ;*****FUNCIÓN DESPLIEGUE*****
0596:
0596: C5 DESPLIEG: PUSH BC ;-----
0597: 01 01 10 LD BC,LCD+1001H;-- Esta función se encarga --
059A: CD 89 09 OTRA: CALL LET_LIS ;-- de mandar los datos de --

```

Control de temperatura con lógica difusa

```

059D: ED A3          OUTI ;-- un letrero la LCD.  --
059F: 20 F9          JR NZ,OTRA ;-----
05A1: CD 89 09       CALL LET_LIS
05A4: C1             POP BC
05A5: C9            RET
05A6:
05A6:                ;***** FUNCION LETTEC *****
05A6:
05A6: 21 4E 0C   LETTEC: LD HL,CAMBIAR ;Lugar del letrero "CAMBIAR DE"
05A9: 3E 01     LD A,LIMPIA ;Limpia el LCD
05AB: D3 00     OUT (LCD),A
05AD: CD 96 05   CALL DESPLIEG
05B0: 3E C0     LD A,LIN_DOS ;Pasa a la linea dos del LCD
05B2: D3 00     OUT (LCD),A
05B4: DD CB 0C 76 BIT 6,(IX+FCONT2);-----
05B8: 20 0F     JR NZ,TN91 ;-- En las siguientes comparaciones --
05BA: DD CB 0C 6E BIT 5,(IX+FCONT2);-- se realiza la filtración de --
05BE: 20 1F     JR NZ,TN1 ;-- letreros para el menu PROGRAMA --
05C0: DD CB 0C F6 SET 6,(IX+FCONT2);-- en forma ciclica. --
05C4: 21 6E 0C   LD HL,RANGOS ;-----
05C7: 18 1D     JR TECLLET
05C9: DD CB 0C EE TN91: SET 5,(IX+FCONT2)
05CD: DD CB 0C B6 RES 6,(IX+FCONT2)
05D1: 21 8E 0C   LD HL,PROPRE
05D4: DD CB 0B 46 BIT 0,(IX+FCONT1)
05D8: 20 0C     JR NZ,TECLLET
05DA: 21 7E 0C   LD HL,PREPRO
05DD: 18 07     JR TECLLET
05DF: DD CB 0C AE TN1: RES 5,(IX+FCONT2)
05E3: 21 5E 0C   LD HL,DESPLIE
05E6: CD 96 05   TECLLET: CALL DESPLIEG
05E9: C9            RET
05EA:
05EA:                ;***** FUNCION PON TIEMPO *****
05EA:
05EA: DD 36 07 50 PON_TIE: LD (IX+VART1),50H;Esta función se encarga de poner el tiempo
05EE: DD 70 08     LD (IX+VART2),B;necesario para retardar un letrero desplegado,
05F1: C9            RET ;con el registro B se puede extender el tiempo.
05F2:
05F2:                ;***** FUNCION CONTROL *****
05F2:
05F2: CD FF 05   CONTROL: CALL ACTENT ;Llama a actualizar entradas
05F5: CD E0 06   CALL FUZY ;Llama a convetir valores reales a difusos
05F8: CD 6E 07   CALL EREGLAS ;Llama a evaluar reglas
05FB: CD A4 07   CALL SALIDA ;Llama a deteminar el estado de las salidas
05FE: C9            RET
05FF:
05FF:                ;***** ACTUALIZA EL VALOR DE LAS ENIRADAS *****
05FF:
05FF: DD 35 16   ACTENT: DEC (IX+CONFF2);Aqui se utiliza un retardo para definir
0602: C0            RET NZ ;una constante de tiempo
0603: CD 45 08   CALL SAL8254
0606: DD 36 16 40 LD (IX+CONFF2),40H;para la actualización de las entradas
060A: F5            PUSH AF
060B: C5            PUSH BC
060C: D5            PUSH DE
060D: E5            PUSH HL
060E: 21 52 12   LD HL,BLKENT+2
0611: 3A 00 20   LD A,(ADCL) ;Actualiza el valor de temperatura, el
0614: 77            LD (HL),A ;valor de temperatura lo pone BLKENT+2
0615: 47            LD B,A ;B=Tactual
0616: 3A 00 14   LD A,(TEMSEL) ;A=Tselec
0619: 90            SUB B ;A=Tselec-Tactual
061A: 5F            LD E,A ;E=Error2
061B: 16 00       LD D,0H ;D->Se encarga de definir que signo tiene Error2
061D: 30 02       JR NC,ACTUNO ;Se compara si hay acarreo
061F: 16 FF       LD D,0FFH ;Si D=FFH nos indica que es negativo E2 por lo
0621: CD 83 06   ACTUNO: CALL AJUSTIA ;tanto Tactual>Tsel.
0624: 32 50 12   LD (BLKENT),A ;A->BLKENT
0627: 7B            LD A,E ;A=Error2

```

Control de temperatura con lógica difusa

```

0628: DD 4E 17          LD C, (IX+ERROR);C=Error1
062B: 91                SUB C          ;A=Error2-Error1
062C: 07                RLCA
062D: 07                RLCA
062E: 07                RLCA
062F: DD 73 17          LD (IX+ERROR),E;Se pasa el Error2->Error1
0632: F5                PUSH AF      ;
0633: DD 7E 18          LD A, (IX+ERRORS);Retrae el signo del error anterior
0636: BA                CP D          ;compara si el signo es el mismo que el actual
0637: 28 14            JR Z,ACTIDOS ;Si es asi salta, si no entonces compara
0639: FE 00            CP 0          ;si el signo anterior es positivo
063B: 20 08            JR NZ,ACTCUA ;si no es asi entonces salta
063D: F1                POP AF      ;
063E: 37                SCF ;pone la bandera de acarreo en alto
063F: DD 36 18 FF        LD (IX+ERRORS),0FFH;Se pone la vandera de negativo
0643: 18 09            JR ACTRES
0645: F1                ACTCUA:     POP AF      ;
0646: B7                OR A          ;Con esto se logra limpiar la bandera de acarreo
0647: DD 36 18 00        LD (IX+ERRORS),0H;pone la el signo como positivo
064B: 18 01            JR ACTRES
064D: F1                ACTIDOS:   POP AF      ;
064E: CD 83 06          ACTIRES:   CALL AJUSTA ;Se pone la derivada del error
0651: 32 51 12          LD (BLKENT+1),A
0654:
0654: 3A 00 30          LD A,(ADC2) ;Actualiza el valor de ADC2
0657: 32 53 12          LD (BLKENT+3),A
065A:
065A: DD CB 0C 46          BIT 0, (IX+FCONT2)
065E: 20 1E            JR NZ,ACTIFIN
0660: DD CB 0B 7E          BIT 7, (IX+FCONT1)
0664: 20 09            JR NZ,ACTILET
0666: DD 36 13 8C          LD (IX+DIR),8CH;Actualiza despliegue de letrero 1
066A: CD EF 08          CALL DBINBCD
066D: 18 0F            JR ACTIFIN
066F:
066F: DD 36 13 C2 ACTILET: LD (IX+DIR),0C2H;Actualiza despliegue de letrero 2
0673: CD EF 08          CALL DBINBCD
0676: DD 36 13 CC          LD (IX+DIR),0CCH
067A: 23                INC HL
067B: CD EF 08          CALL DBINBCD
067E:
067E:                ACTIFIN:   ;LD (IX+TOPE),0FFH ;*****ES PARA DEBUG
067E:                ;LD HL,BLKENT ;*****ES PARA DEBUG
067E:                ;LD (IX+DIR),80H ;*****ES PARA DEBUG
067E:                ;CALL DHEXA ;*****ES PARA DEBUG
067E:                ;INC HL ;*****ES PARA DEBUG
067E:                ;LD (IX+DIR),0C0H ;*****ES PARA DEBUG
067E:                ;CALL DHEXA ;*****ES PARA DEBUG
067E:
067E: E1                POP HL
067F: D1                POP DE
0680: C1                POP BC
0681: F1                POP AF
0682: C9                RET
0683:                ;***** FUNCION QUE AJUSTA LOS VALORES DE LAS ENTRADAS*****
0683:                ;***** PASA COMO REFERENCIA "A" Y "B" RESULTADO EN "A"*****
0683: 4F                AJUSTA:     LD C,A
0684: 38 0B            JR C,AJUNEG
0686: D6 80            SUB 80H     ;Checa que el valor no exeda de 80H
0688: 30 04            JR NC,AJUNO
068A: 79                AJDOS:     LD A,C
068B: C6 80            ADD A,80H
068D: C9                RET
068E: 3E FF          AJUNO:     LD A,0FFH
0690: C9                RET
0691: C6 80          AJUNEG:   ADD A,80H
0693: 4F                LD C,A
0694: D6 80            SUB 80H
0696: 38 03            JR C,AJUIRES
0698: 3E 00            LD A,0H
069A: C9                RET

```

Control de temperatura con lógica difusa

```

069B: 79      AJUIRES:  LD A,C
069C: C9      RET
069D:
069D:        ;***** PROGRAMA QUE CALCULA LAS PENDIENTES DE LOS CONJUNTOS *****
069D:
069D: 3A 02 14   PENDIEN:  LD A,(NUMENT) ;Calcula el número de pendientes que
06A0: 07      RLC A ;esta dado por el número de entradas
06A1: DD 77 00      LD (IX+VAR1L),A;y número de conjuntos.
06A4: 3A 03 14      LD A,(NUMCON) ;pendientes = conjuntos*2*entradas
06A7: DD 77 02      LD (IX+VAR2L),A
06AA: CD CF 09      CALL MULTI
06AD: DD 46 04      LD B,(IX+VAR3L);# de pendientes -> B
06B0: FD 21 54 12   LD IY,C1P1E1 ;Lugar donde estaran las pendientes calculadas
06B4: 21 05 14      LD HL,C1P1E1+1 ;Lugar de lo puntos de los conjuntos
06B7: DD 36 02 FF   LD (IX+VAR2L),0FFH;Inicia los valores de la división
06BB: DD 36 03 00      LD (IX+VAR2H),00H
06BF: DD 36 01 00      LD (IX+VAR1H),00H
06C3: 7E      LOOP1:   LD A,(HL) ; A = P2 ÷ P4
06C4: 2B      DEC HL ; HL se coloca en el punto anterior
06C5: 96      SUB (HL) ;VALOR DEL DIVIDENDO EN A
06C6: DD 77 00      LD (IX+VAR1L),A;Pasa el valor a la división
06C9: CD 92 09      CALL DIV
06CC: DD 7E 04      LD A,(IX+VAR3L); A = (P2 - P1)/FF ÷ A = (P4 - P3)/FF
06CF: FE 00      CP 0H ;Compara si los puntos son iguales el resultado
06D1: 20 02      JR NZ,PENCERO ;ser FFH
06D3: 3E FF      LD A,0FFH
06D5: FD 77 00   PENCERO:  LD (IX+0),A ;Se coloca el resultado donde corresponde
06D8: 23      INC HL ;-----
06D9: 23      INC HL ;Se coloca en el sig. conjunto
06DA: 23      INC HL ;-----
06DB: FD 23      INC IY ;Siguiente lugar de pendiente
06DD: 10 E4      DJNZ LOOP1 ;? estan todas las pendientes
06DF: C9      RET ; Si -> retorna
06E0:
06E0:        ;***** PROGRAMA QUE EFECTUA LA FUSIFICACIÓN *****
06E0:
06E0: F5      FUZY:   PUSH AF ;-----
06E1: C5      PUSH BC ; Retiene valores
06E2: D5      PUSH DE ; anteriores
06E3: E5      PUSH HL ;-----
06E4: FD 21 04 14   LD IY,C1P1E1 ;PUNTO 1 DEL CONJUNTO 1 Y ENTRADA 1
06E8: 21 54 12      LD HL,C1M1E1 ;Lugar de las pendientes calculadas
06EB: 11 00 12      LD DE,FUZIN ;Lugar en donde se pone el valor difuso
06EE: D9      EXX
06EF: 21 50 12      LD HL,BLKENT ;HL' Tendra la dir para el BLOQUE DE ENTRADAS
06F2: D9      EXX
06F3: 3A 02 14      LD A,(NUMENT)
06F6: 4F      LD C,A ; # de entradas
06F7: D9      FUZYB:  EXX
06F8: 7E      LD A,(HL)
06F9: 23      INC HL
06FA: D9      EXX
06FB: DD 77 06      LD (IX+GEN1),A ;Variable que utiliza valua para calcular
06FE: 3A 03 14      LD A,(NUMCON)
0701: 47      LD B,A ;# de conjuntos
0702: CD 24 07   FUZYVAL:  CALL VALUA ;Función que realiza el calculo correspondiente
0705: 12      LD (DE),A ;En A se encuentra el Min(P1,P2,FF)
0706: 13      INC DE ;lo pone en FUZIN e incrementa el lugar
0707: FD 23      INC IY ;-----
0709: FD 23      INC IY ;IY se ubica en el conjunto
070B: FD 23      INC IY ;siguiente
070D: FD 23      INC IY ;-----
070F: 10 F1      DJNZ FUZYVAL ;Checa si ya se realizo para los conjuntos
0711: 3A 03 14      LD A,(NUMCON)
0714: FE 08      FUZYCPA:  CP 8 ;Se posiciona en el lugar correcto de
0716: 28 04      JR Z,FUZYPOS ;los resultados en FUZIN
0718: 3C      INC A
0719: 13      INC DE
071A: 18 F8      JR FUZYCPA
071C: 0D      FUZYPOS:  DEC C ;Checa si ya se realizo para las entradas
071D: 20 D8      JR NZ,FUZYB

```

Control de temperatura con lógica difusa

```

071F: E1          POP HL          ;-----
0720: D1          POP DE          ;Recupera los valores anteriores
0721: C1          POP BC          ;de los registros
0722: F1          POP AF          ;-----
0723: C9          RET ;Regresa
0724:
0724:              ;***** FUNCION VALUA *****
0724:
0724: C5          VALUA:   PUSH BC
0725: D5          PUSH DE
0726: DD 56 06     LD D,(IX+GEN1) ;Recupera valor de la entrada
0729: FD 7E 03     LD A,(IY+P4)  ;Se coloca en el punto 4
072C: 92          SUB D          ; A = P4 - ADC
072D: 38 38       JR C,VALNEG  ; Si es negativo Termina
072F: 5F          LD E,A        ; A -> E
0730: 7A          LD A,D
0731: FD 96 00     SUB (IY+0)    ; A = ADC - P1
0734: 38 31       JR C,VALNEG  ; Si es negativo Termina
0736: 06 02       LD B,02H     ;Por ser dos pendientes por conjunto
0738: DD 36 01 00 LD (IX+VAR1H),00H;Variables de la mult. sin utilizar
073C: DD 36 03 00 LD (IX+VAR2H),00H;Variables de la mult. sin utilizar
0740: DD 77 00     VAL2:   LD (IX+VAR1L),A;Valor del incremento
0743: 7E          LD A,(HL)     ;Valor de la pendiente
0744: DD 77 02     LD (IX+VAR2L),A
0747: CD CF 09     CALL MULTI   ;Realiza la multiplicacin
074A: DD 7E 05     LD A,(IX+VAR3H);-----
074D: FE 00       CP 0          ;Verifica si el resultado es > FFH
074F: 28 04       JR Z, VAL0   ;-----
0751: DD 36 04 FF     LD (IX+VAR3L),0FFH;Si es >FFH coloca FFH
0755: DD 4E 04     VAL0:   LD C,(IX+VAR3L);Resultado -> C
0758: 23          INC HL       ;Sig. pendiente
0759: 7B          LD A,E        ;E -> A
075A: 59          LD E,C        ;C -> E
075B: 10 E3       DJNZ VAL2   ;? para las dos pendientes
075D: 47          LD B,A        ;*****
075E: 93          SUB E        ; Obtiene el valor
075F: 30 03       JR NC,MENE  ; minimo de los
0761: 78          LD A,B        ; productos de los
0762: 18 07       JR VALFIN  ; incrementos y
0764: 7B          MENE:   LD A,E        ; pendientes
0765: 18 04       JR VALFIN  ;*****
0767: 3E 00     VALNEG:  LD A,00H    ; Valor cero si in incremento es (-)
0769: 23          INC HL
076A: 23          INC HL
076B: D1          VALFIN:  POP DE
076C: C1          POP BC
076D: C9          RET
076E:
076E:              ;***** FUNCION QUE EVALUA REGLAS *****
076E:
076E: F5          EREGLAS:  PUSH AF
076F: C5          PUSH BC
0770: D5          PUSH DE
0771: E5          PUSH HL
0772:
0772: 06 20       LD B,20H     ;Este bloque se encarga de borrar las
0774: 21 20 12     LD HL,FUZOUT ;reglas evaluadas en ocasiones anteriores
0777: 3E 00       LD A,0
0779: 77          BORRAREG: LD (HL),A
077A: 23          INC HL
077B: 10 FC       DJNZ BORRAREG
077D:
077D: 26 65       LD H,FUZH   ;Lugar de FUZIN y FUZOUT (Parte alta)
077F: 11 A6 14     LD DE,REGLAS ;Direccin donde est n las reglas
0782: 06 FF       RSAL:   LD B,0FFH ;Establece como mnimo el valor FFH
0784: 1A          RSIG2:  LD A,(DE)  ;Carga la regla en curso
0785: CB 7F       BIT 7,A    ;? fin de las reglas
0787: 28 05       JR Z,RSIG  ;NO continua con la evaluacin
0789: E1          POP HL     ;SI termina
078A: D1          POP DE
078B: C1          POP BC

```

Control de temperatura con lógica difusa

```

078C: F1          POP AF
078D: C9          RET
078E: 13          RSIG:   INC DE           ;Se posiciona a la siguiente regla
078F: 6F          LD L,A           ;Se completa la dirección (HL)
0790: CB 6F       BIT 5,A          ;? antecedente o consecuente
0792: 28 08       JR Z,RB50        ;Salta si es antecedente
0794: 79          LD A,C
0795: FE 00       CP 0
0797: 28 E9       JR Z,RSAL
0799: 77          LD (HL),A       ;Es consecuente por lo que se pone el resultado
079A: 18 E6       JR RSAL         ;Siguiente regla
079C: 7E          RB50:   LD A,(HL)       ;SE REALIZA LA OPERACION AND PARA LOGICA DIFUSA
079D: 90          SUB B           ;La operación AND en la lógica difusa es el
079E: 30 01       JR NC,RNG       ;obtener el mínimo y esto lo realiza
07A0: 46          LD B,(HL)       ;este bloque
07A1: 48          RNG:     LD C,B
07A2: 18 E0       JR RSIG2
07A4:
07A4:             ;***** FUNCION QUE EVALUA LA SALIDA *****
07A4:
07A4: F5          SALIDA:  PUSH AF
07A5: C5          PUSH BC
07A6: D5          PUSH DE
07A7: E5          PUSH HL
07A8: FD 21 85 14 LD IY,SINGS1   ;Lugar de los SINGLETON de salida
07AC: 21 20 12   LD HL,FUZOUT   ;Lugar donde se colocara FUZOUT
07AF: D9          EXX ;En HL' se tendra F(X)Z
07B0: 21 40 12   LD HL,FXZS1L
07B3: D9          EXX
07B4: 3A 01 14   LD A,(NUMSAL)
07B7: 4F          LD C,A         ;# de salidas
07B8: 11 00 00   SALC0:  LD DE,00H ;Inicia los registros para la suma
07BB: 3A 03 14   LD A,(NUMCON)
07BE: 47          LD B,A         ;# de conjuntos
07BF: FD 7E 00   SALENO: LD A,(IY+0) ;Apunta al lugar de Singletons
07C2: DD 77 00   LD (IX+VAR1L),A
07C5:
07C5: 7E          LD A,(HL)     ;Apunta al lugar de FUZOUT
07C6: DD 77 02   LD (IX+VAR2L),A
07C9: CD CF 09   CALL MULTI    ;SINGSij*FUZOUTij
07CC: DD 7E 04   LD A,(IX+VAR3L)
07CF: 83          ADD A,E
07D0: 5F          LD E,A
07D1: DD 7E 05   LD A,(IX+VAR3H)
07D4: 8A          ADC A,D
07D5: 57          LD D,A        ;DE=SINGSij*FUZOUTij + DE
07D6: FD 23       INC IY
07D8: 23          INC HL
07D9: 10 E4       DJNZ SALENO
07DB: 3A 03 14   LD A,(NUMCON)
07DE: FE 08       SALCPA:  CP 8         ;Se posiciona en el lugar correcto de
07E0: 28 04       JR Z,SALPOS   ;FUZOUT
07E2: 3C          INC A
07E3: 23          INC HL
07E4: 18 F8       JR SALCPA
07E6: 7B          SALPOS:  LD A,E         ;Pone los resultados parciales a
07E7: D9          EXX ;partir de FXZS1L
07E8: 77          LD (HL),A
07E9: 23          INC HL
07EA: D9          EXX
07EB: 7A          LD A,D
07EC: D9          EXX
07ED: 77          LD (HL),A
07EE: 23          INC HL
07EF: D9          EXX
07F0: 0D          DEC C         ;Checa si se realizo para todas las salidas
07F1: 20 C5       JR NZ,SALC0
07F3:
07F3: FD 21 20 12   LD IY,FUZOUT   ;Hasta aqui se tiene el numerador y a
07F7: 21 40 12   LD HL,FXZS1L   ;continuación se realiza la suma
07FA: D9          EXX ;del bloque para el denominador

```


Control de temperatura con lógica difusa

```

07FB: 21 48 12      LD HL,CICLOS1 ;Aqui comienza SUMATORIA DE Zi
07FE: D9            EXX
07FF: 3A 01 14      LD A,(NUMSAL)
0802: 4F            LD C,A ;Se coloca el numero de salida
0803: 3A 03 14      LD A,(NUMCON) ;# de conjuntos
0806: 47            LD B,A
0807: 11 00 00      LD DE,00H
080A: FD 7E 00      SALCAR: LD A,(IY+0)
080D: 83            ADD A,E
080E: 5F            LD E,A
080F: 7A            LD A,D
0810: CE 00         ADC A,00H
0812: 57            LD D,A
0813: FD 23         INC IY
0815: 10 F3         DJNZ SALCAR
0817: 3A 03 14      LD A,(NUMCON)
081A: FE 08         SALCPAL: CP 8 ;Se coloca en la correcta posición
081C: 28 05         JR Z,SALPOS1 ;de FUZOUT
081E: 3C            INC A
081F: FD 23         INC IY
0821: 18 F7         JR SALCPAL
0823: DD 73 00      SALPOS1: LD (IX+VAR1L),E;Se pone el resultado en un registro temporal
0826: DD 72 01      LD (IX+VAR1H),D;para realizar las operaciones correspondientes
0829: 7E            LD A,(HL)
082A: DD 77 02      LD (IX+VAR2L),A
082D: 23            INC HL
082E: 7E            LD A,(HL)
082F: DD 77 03      LD (IX+VAR2H),A
0832: 23            INC HL
0833: CD 92 09      CALL DIV
0836: DD 7E 04      LD A,(IX+VAR3L)
0839: D9            EXX
083A: 77            LD (HL),A ;CICLOS1=FXZS1/DE
083B: 23            INC HL
083C: D9            EXX
083D: 0D            DEC C ;Lo realizo para todas las salidas
083E: 20 C3         JR NZ,SALB
0840:
0840: ;LD (IX+TOPE),0FFH ;*****ES PARA DEBUG
0840: ;LD HL,CICLOS1 ;*****ES PARA DEBUG
0840: ;LD (IX+DIR),87H ;*****ES PARA DEBUG
0840: ;CALL DHEXA ;*****ES PARA DEBUG
0840: ;INC HL ;*****ES PARA DEBUG
0840: ;LD (IX+DIR),0C7H ;*****ES PARA DEBUG
0840: ;CALL DHEXA ;*****ES PARA DEBUG
0840:
0840: E1            POP HL
0841: D1            POP DE
0842: C1            POP BC
0843: F1            POP AF
0844: C9            RET
0845:
0845: ;***** FUNCION SAL8254 *****
0845:
0845: F5            SAL8254: PUSH AF
0846: C5            PUSH BC
0847: D5            PUSH DE
0848: FD E5         PUSH IY
084A: DD 7E 0A         LD A,(IX+B_M)
084D: FE 04         CP 04H ;Verifica si esta en inactivo
084F: 20 05         JR NZ,CONAC
0851: CD 99 08         CALL DES8254 ;Si est en inactivo anula las salidas
0854: 18 3D         JR CONREG ;el sistema no se altere
0856:
0856:
0856: DD 7E 0A         CONAC: LD A,(IX+B_M) ;Verifica si esta manual
0859: FE 02         CP 02H
085B: 28 10         JR Z,CONSIG
085D: 06 04         LD B,04H
085F: FD 21 48 12     LD IY,CICLOS1
0863: FD 7E 00         CONREC: LD A,(IY+0)

```

Control de temperatura con lógica difusa

```

0866: FD 77 04          LD (IX+CONAUX),A
0869: FD 23             INC IY
086B: 10 F6            DJNZ CONREC
086D:
086D:
086D: FD 21 4C 12 CONSIG: LD IY,CICLOS1+CONAUX
0871: 3E 00            LD A,0
0873: FD 46 00          LD B,(IY+0) ;Valor inicial para el cont2
0876: 90                SUB B
0877: FE 00            CP 0H
0879: 20 02            JR NZ,SALUNO
087B: 3E FF            LD A,0FFH
087D: D3 82            SALUNO: OUT (P8254+2),A
087F: 3E 00            LD A,0
0881: D3 82            OUT (P8254+2),A
0883:
0883: FD 46 01          LD B,(IY+1) ;Valor inicial para el cont1
0886: 90                SUB B
0887: FE 00            CP 0H
0889: 20 02            JR NZ,SALDOS
088B: 3E FF            LD A,0FFH
088D: D3 81            SALDOS: OUT (P8254+1),A
088F: 3E 00            LD A,0
0891: D3 81            OUT (P8254+1),A
0893:
0893: FD E1            CONREG: POP IY
0895: D1                POP DE
0896: C1                POP BC
0897: F1                POP AF
0898: C9                RET
0899:
0899: ;***** DES8254 *****
0899: F5            DES8254: PUSH AF
089A: 3E 3E          LD A,WORDCON1 ;Programa el Contador1 como generador de onda
089C: D3 83          OUT (P8254+3),A;cuadrada a partir de la frecuencia del sistema
089E: 3E 74          LD A,DIV255 ;para que entren 255 divisiones en el periodo
08A0: D3 80          OUT (P8254),A ;de media onda senosoidal de la fuente
08A2: 3E 00          LD A,0 ;el,ctrica
08A4: D3 80          OUT (P8254),A
08A6:
08A6: 3E BA          LD A,WORDCON2 ;Programa el Contador2 como diparado por secal
08A8: D3 83          OUT (P8254+3),A;de hardware
08AA: 3E FF          LD A,DIVINI ;Valor inicial para el cont2****!!!!!!
08AC: D3 82          OUT (P8254+2),A
08AE: 3E 00          LD A,0
08B0: D3 82          OUT (P8254+2),A
08B2:
08B2: 3E 7A          LD A,WORDCON3 ;Programa el Contador2 como diparado por secal
08B4: D3 83          OUT (P8254+3),A;de hardware
08B6: 3E FF          LD A,DIVINI ;Valor inicial para el cont2****!!!!!!
08B8: D3 81          OUT (P8254+1),A
08BA: 3E 00          LD A,0
08BC: D3 81          OUT (P8254+1),A
08BE: F1            POP AF
08BF: C9            RET
08C0:
08C0: ;*****FUNCIÒN DE TRANSFORMACIÒN DE BINARIO A DECIMAL*****
08C0: ;***** EN ASCII*****
08C0:
08C0: DD 77 00        BINBCD: LD (IX+VAR1L),A;Factor uno
08C3: DD 70 02        LD (IX+VAR2L),B;Factor dos
08C6: CD CF 09        CALL MULTI
08C9: DD 7E 05        LD A,(IX+VAR3H);A = A*B
08CC: DD 36 05 00    LD (IX+VAR3H),00H
08D0: D6 0A          BINSIG: SUB 10
08D2: 38 05          JR C,MEN10
08D4: DD 34 05        INC (IX+VAR3H)
08D7: 18 F7          JR BINSIG
08D9: C6 3A          MEN10: ADD A,30H+10 ;Valor en ASCII
08DB: DD 77 04        LD (IX+VAR3L),A
08DE: DD 7E 05        LD A,(IX+VAR3H)

```

Control de temperatura con lógica difusa

```

08E1: FE 00          CP 0
08E3: 20 04          JR NZ,BINNO
08E5: 3E FE          LD A,0FEH
08E7: 18 02          JR BINGUAR ;Las unidades en ASCII se encuentran en
08E9: C6 30          BINNO:  ADD A,30H ;(IX+VAR3L), las decenas en (IX+VAR3H)
08EB: DD 77 05       BINGUAR: LD (IX+VAR3H),A
08EE: C9            RET
08EF:
08EF: ;***** PROGRAMA QUE REALIZA EL DESPLIEGUE DE DATOS *****
08EF:
08EF: F5          DBINBCD:  PUSH AF ;La rutina de despliegue de datos requiere
08F0: C5          PUSH BC ;la direccion del dato en los registros HL.
08F1: 7E          LD A,(HL) ;Los datos son por dos digitos que se ponen
08F2: 06 64       LD B,FACTOR ;en el LCD
08F4: CD C0 08    CALL BINBCD ;El factor es por la cual hay que multiplicar
08F7: ;el número para usar toda la escala
08F7: DD 7E 13     LD A,(IX+DIR) ;Se accede a la dirección correspondiente
08FA: D3 00       OUT (LCD),A ;y se lee el dato actual, esto sirve
08FC: CD 89 09    CALL LET_LIS ;para posteriormente se comparen tanto
08FF: DB 03       IN A,(LCD+3) ;el número leído como el calculado
0901: 57          LD D,A ;y de esa manera verificar si hubo cambio
0902: CD 89 09    CALL LET_LIS ;en los digitos.
0905: DD 46 04     LD B,(IX+VAR3L)
0908: 58          LD E,B
0909: B8          CP B
090A:
090A: 3E 10       LD A,RETL
090C: D3 00       OUT (LCD),A
090E: CD 89 09    CALL LET_LIS
0911:
0911: 20 06       JR NZ,DBINSIG1
0913: DD 36 06 FF   LD (IX+GEN1),0FFH;Esto indica que no hubo cambio
0917: 18 04       JR DBINSIG2
0919:
0919: DD 36 06 00 DBINSIG1: LD (IX+GEN1),0H;Indica que si hubo cambio en el digito
091D:
091D: 78          DBINSIG2: LD A,B
091E: D3 01       OUT (LCD+1),A ;Realiza el despliegue de digitos
0920: CD 89 09    CALL LET_LIS
0923: 3E 10       LD A,RETL
0925: D3 00       OUT (LCD),A
0927: CD 89 09    CALL LET_LIS
092A: D3 00       OUT (LCD),A
092C: CD 89 09    CALL LET_LIS
092F: DD 7E 05     LD A,(IX+VAR3H)
0932: D3 01       OUT (LCD+1),A
0934: CD 89 09    CALL LET_LIS
0937:
0937: C1          POP BC
0938: F1          POP AF
0939: C9          RET
093A:
093A: ;***** DHEXA *****
093A:
093A: F5          DHEXA:  PUSH AF
093B: DD 7E 15     LD A,(IX+TOPE) ;Verifica si no tiene tope
093E: FE FF       CP 0FFH
0940: 28 14       JR Z,DLIBRE
0942: 7E          LD A,(HL)
0943: FE 00       CP 0
0945: 20 04       JR NZ,DMAY0 ;Si tiene tope compara para no
0947: 36 01       LD (HL),1 ;sobrepasarlo
0949: 18 0B       JR DLIBRE
094B: D6 01       DMAY0:  SUB 1
094D: DD 96 15     SUB (IX+TOPE)
0950: 38 04       JR C,DLIBRE
0952: DD 7E 15     LD A,(IX+TOPE)
0955: 77          LD (HL),A
0956:
0956: DD 7E 13     DLIBRE: LD A,(IX+DIR) ;Aqui separa la parte alta y baja para su
0959: D3 00       OUT (LCD),A ;transformación a ASCII

```

Control de temperatura con lógica difusa

```

095B: CD 89 09          CALL LET_LIS    ;con despligue en hexadecimal
095E: 7E                LD A,(HL)
095F: E6 F0             AND 0FOH
0961: 0F                RRCA
0962: 0F                RRCA
0963: 0F                RRCA
0964: 0F                RRCA
0965: CD 7B 09          CALL CONVHEX
0968: 7E                LD A,(HL)
0969: E6 0F             AND 0FH
096B: CD 7B 09          CALL CONVHEX
096E: 3E 10             LD A,RET1L
0970: D3 00             OUT (LCD),A
0972: CD 89 09          CALL LET_LIS
0975: DD 36 06 00      LD (IX+GEN1),0 ;Por default hay cambio de digito
0979: F1                POP AF
097A: C9                RET
097B:
097B: ;***** CONVHEX *****
097B: FE 0A          CONVHEX: CP 10          ;Se encarga de convertir el binario a
097D: 38 02          JR C,HEXMEN10 ;ASCII para despligue en hexadecimal
097F: C6 07          ADD A,7
0981: C6 30          HEXMEN10: ADD A,30H
0983: D3 01          OUT (LCD+1),A
0985: CD 89 09          CALL LET_LIS
0988: C9                RET
0989:
0989: ;*****FUNCIÓN DE LEIRERO LISTO*****
0989: F5                LET_LIS: PUSH AF      ;Esta función se dedica a esperar
098A: DB 02          ESP: IN A,(LCD+2) ;ejecutar algún comando o recibir
098C: E6 80          AND 80H ;datos verificando la bandera BUSY
098E: 20 FA          JR NZ,ESP ;del mismo.
0990: F1                POP AF
0991: C9                RET
0992:
0992: ;*****
0992: ;PROGRAMA QUE EFECTUA UNA DIVISION ENTERA DE 16BITS X 16BITS Y UN RESULTADO DE
0992: ;***** 16 BITS *****
0992: F5                DIV: PUSH AF
0993: C5                PUSH BC
0994: D5                PUSH DE
0995: E5                PUSH HL
0996: 21 00 00        LD HL,00H ;Seran usados para guardar el resultado parcial
0999: 06 11            LD B,11H ;Numero de bits más uno (16 bits + 1)
099B: DD 5E 02        LD E,(IX+VAR2L);Obtiene el valor del dividendo
099E: DD 56 03        LD D,(IX+VAR2H)
09A1: CB 13          DIV1: RL E ;Rota el valor bajo del dividendo con carry
09A3: CB 12          RL D ;Rota el valor alto del dividendo con carry
09A5: 10 0B          DJNZ DIV2 ;? fin de bits
09A7: DD 73 04        LD (IX+VAR3L),E;Guarda el valor final a
09AA: DD 72 05        LD (IX+VAR3H),D;la variable VAR3
09AD: E1                POP HL ;Regresa valores principales
09AE: D1                POP DE
09AF: C1                POP BC
09B0: F1                POP AF
09B1: C9                RET ;Retorna de la subrutina
09B2: CB 15          DIV2: RL L ;Rota el valor del dividendo parcial
09B4: CB 14          RL H
09B6: 7D                LD A,L ;Recupera el valor del dividendo parcial
09B7: DD 96 00        SUB (IX+VAR1L) ;y restalo al divisor
09BA: 6F                LD L,A
09BB: 7C                LD A,H
09BC: DD 9E 01        SEC A,(IX+VAR1H)
09BF: 67                LD H,A
09C0: 30 0A          JR NC,DIVNOSUM ;? existe acarreo
09C2: 7D                LD A,L ;Si hay acarreo por lo que

```

Control de temperatura con lógica difusa

```

09C3: DD 86 00          ADD A,(IX+VAR1L);se regresa al valor inicial
09C6: 6F              LD L,A
09C7: 7C              LD A,H
09C8: DD 8E 01          ADC A,(IX+VAR1H)
09CB: 67              LD H,A
09CC: 3F              DIVNOSUM: CCF ;Complementa el acarreo
09CD: 18 D2          JR DIV1          ;y repite para el siguiente bit
09CF:
09CF: ;*****
09CF: ;*PROGRAMA QUE EFECTUA UNA MULTIPLICACION DE 8BITS X 8BITS Y UN RESULTADO DE*
09CF: ;***** 16BITS *****
09CF:
09CF: F5              MULTI:   PUSH AF          ;Se guarda el contenido de AF
09D0: C5              PUSH BC
09D1: D5              PUSH DE
09D2: E5              PUSH HL
09D3: 11 00 00          LD DE,00H          ;El resultado acumulado es igual a cero
09D6: DD 7E 00          LD A,(IX+VAR1L);Lugar del multiplicando
09D9: FE 00          CP 0H              ;Compara si no es cero, si lo es
09DB: 28 18          JR Z,MULRESUL     ;el resultado es cero
09DD: 4F              LD C,A            ;Multiplicando -> C
09DE: DD 7E 02          LD A,(IX+VAR2L);Lugar del multiplicador
09E1: FE 00          CP 0H              ;Compara si no es cero, si lo es
09E3: 28 10          JR Z,MULRESUL     ;el resultado es cero
09E5: 67              LD H,A            ;Multiplicador -> H
09E6: 06 08          LD B,8H           ;C_B = 8 (contador de bit)
09E8: CB 1C          MUL1:   RR H          ;Rota al multiplicador a la derecha con carry
09EA: 30 03          JR NC,MULNOSUM    ;Checa el bit en curso
09EC: 7A              LD A,D            ;El bit en curso es "1" y se sumar
09ED: 81              ADD A,C           ;al resultado acumulado
09EE: 57              LD D,A
09EF: CB 1A          MULNOSUM: RR D          ;Rotar el resultado acumulado
09F1: CB 1B          RR E
09F3: 10 F3          DJNZ MUL1         ;"Se realiz  para los ocho bits?
09F5: DD 73 04          MULRESUL: LD (IX+VAR3L),E;EL RESULTADO ACUMULADO ESTA EN DE
09F8: DD 72 05          LD (IX+VAR3H),D;Y SE PASA A UN LUGAR DE MEMORIA
09FB: E1              POP HL            ;Regresa los valores de los registros
09FC: D1              POP DE
09FD: C1              POP BC
09FE: F1              POP AF           ;B,C,D,E,H,L,A,F
09FF: C9              RET ;Regresa de la multiplicaci n
0A00:
0A00: ;*****TIEMPO 1 DURACI N APROX. 50ms*****
0A00: C5              TIEMPO_1: PUSH BC
0A01: 0E 5A          LD C,5AH          ;Tiempo que se utiliza para eliminar
0A03: 06 00          SIG5:   LD B,00H     ;los rebotes de los botones.
0A05: 10 FE          BME:   DJNZ BME
0A07: 0D              DEC C
0A08: 20 F9          JR NZ,SIG5
0A0A: C1              POP BC
0A0B: C9              RET

```

Line #	Identifier Name	Value
959	ACTCUA	0645
963	ACTIDOS	064D
921	ACTIENT	05FF
984	ACTFIN	067E
978	ACTILET	066F
964	ACTIRES	064E
939	ACTUNO	0621
189	ADC1	2000
190	ADC2	3000
90	AJCEN1	0D3E
92	AJCEN2	0D4E
1003	AJDOS	068A
84	AJENIR	0D0E
96	AJREG	0D6E
88	AJREGL	0D2E
94	AJSAL	0D5E
86	AJSALI	0D1E

Control de temperatura con lógica difusa

1008	AJUNEG	0691
1006	AJUNO	068E
999	AJUSTA	0683
1014	AJUTRES	069B
158	ARRANL	0012
837	AUTO	0554
43	AUTOINCO	0BCE
40	AUTOMATI	0EBE
103	BASEFUZ	1200
1357	BINBCD	08C0
1374	BINGUAR	08EB
1373	BINNO	08E9
1362	BINSIG	08D0
268	BLKDATP	009D
117	BLKENT	1250
104	BLKFROM	0200
291	BLKRAM	00C8
145	BM	000A
1557	BME	0A05
1143	BORRAREG	0779
144	BT	0009
120	C1MLE1	1254
126	C1P1E1	1404
20	C1P1E1P	0B17
70	CAMANU1	0C9E
72	CAMANU2	0CAE
74	CAMANU3	0CBE
60	CAMBIAR	0C4E
113	CICLOS1	1248
10	CODENTER	0B04
8	COMLCD	0B00
257	CON	0088
1292	CONAC	0856
177	CONAUX	0004
155	CONCON	0010
153	CONENT	000E
162	CONF2	0016
156	CONPUN	0011
1297	CONREC	0863
1323	CONREG	0893
154	CONSAL	000F
1303	CONSIG	086D
913	CONTROL	05F2
1462	CONVHEX	097B
1379	DBINBCD	08EF
1403	DBINSIG1	0919
1405	DBINSIG2	091D
1330	DES8254	0899
852	DESP	0578
62	DESPLIE	0C5E
869	DESPLIEG	0596
1423	DHEXA	093A
159	DIR	0013
1483	DIV	0992
1491	DIV1	09A1
1501	DIV2	09B2
181	DIV255	0074
182	DIVINI	00FF
1516	DIVNOSUM	09CC
1438	DLIERE	0956
1432	DMAYO	094B
173	ENTER	0004
1135	EREGLAS	076E
163	ERROR	0017
164	ERRORS	0018
1473	ESP	098A
180	FACTOR	0064
147	FCONT1	000B
148	FCONT2	000C
170	FILTRAT	000F
169	FUZH	0065

Control de temperatura con lógica difusa

106	FUZIN	1200
108	FUZOUT	1220
1050	FUZY	06E0
1062	FUZYB	06F7
1078	FUZYCPA	0714
1083	FUZYPOS	071C
1069	FUZYVAL	0702
110	FXZSL	1240
138	GEN1	0006
160	HEXA	0014
1465	HEXMEN10	0981
51	INACINCO	0C0E
49	INACTIVO	0BFE
231	INDATO	004F
192	INDEXX	1100
227	LA	0042
76	LAUTO	0CCE
188	LCD	0000
82	LCONJU	0CFE
98	LCONREG	0D7E
78	LENIRA	0CDE
823	LETBT	0532
1472	LETLIS	0989
812	LETPRIN	051A
880	LETTEC	05A6
175	LIMPIA	0001
176	LINDOS	00C0
1031	LOOP1	06C3
80	LSALID	0CEE
45	MANUAL	0BDE
47	MANUINCO	0BEE
172	MAS	0002
1366	MEN10	08D9
1124	MENE	0764
171	MENOS	0001
38	MOD0	0BAE
1537	MUL1	09E8
1542	MULNOSUM	09EF
1545	MULRESUL	09F5
1523	MULTI	09CF
855	NOESLETC	057D
244	NPAUTO	006F
248	NPMANU	0078
239	NPPROG	0063
125	NUMCON	1403
124	NUMENT	1402
123	NUMSAL	1401
871	OTRA	059A
168	P4	0003
186	P8254	0080
320	PAUTO	00EF
332	PAUTOCON	010A
339	PAUTOREG	011A
324	PAUTOSEL	00F6
335	PAUTOSIG	0110
1019	PENDIEN	069D
1040	PENCERO	06D5
366	PMANU	0153
389	PMANU1	0187
396	PMANU2	0193
413	PMANU3	01B9
371	PMANUSIG	015C
794	PONPUN	04F7
907	PONTIE	05EA
577	PPAJUS	030D
785	PPCAMD	04E6
506	PPCHECA	026B
695	PPCOM3	0411
623	PPCOM4	0371
711	PPEN1	0438
727	PPEN2	045C

Control de temperatura con lógica difusa

767	PPEN3	04B9
758	PPEN5	04A9
742	PPEN6	0483
769	PPENHL	04BF
535	PPMAS	02B6
537	PPMENOS	02BC
775	PEN6	04C8
553	PPNO4	02E0
518	PPNOCERO	028C
543	PPNODOS	02C8
548	PPNOTRES	02D4
524	PPNOUNO	029B
595	PPREG2	0337
616	PPREG3	0363
480	PPROG	0238
491	PPROGSIG	024B
642	PPSAL1	03A1
657	PPSAL2	03C1
691	PPSAL3	040B
681	PPSAL4	03F6
788	PPSIG	04EF
531	PPTECCON	02AC
558	PPTECDIF	02EB
563	PPTECIN	02F4
559	PPTECOT	02EE
66	PREPRO	0C7E
55	PRINCI1	0C2E
58	PRINCI2	0C3E
817	PRINCID2	052A
53	PROGRAMA	0C1E
68	PROPRE	0C8E
305	RAMELK	00DA
64	RANGOS	0C6E
1167	RB50	079C
128	REGLAS	14A6
30	REGLASP	0B49
129	RESPAL	15A6
179	RETLI	0010
1171	RNG	07A1
1149	RSAL	0782
1158	RSIG	078E
1150	RSIG2	0784
15	SAENCOP	0B14
1281	SAL8254	0845
1231	SALB	0803
1190	SALENO	07BF
1187	SALCO	07B8
1234	SALCAR	080A
1206	SALCPA	07DE
1243	SALCPAL	081A
1319	SALDOS	088D
1176	SALIDA	07A4
1211	SALPOS	07E6
1248	SALPOS1	0823
824	SALTA	0535
1310	SALUNO	087D
174	SELEC	0008
835	SIESLETC	0550
213	SIG	0027
839	SIG1	0559
859	SIGLINCO	0586
843	SIG2	0562
863	SIG2INCO	058F
847	SIG3	056B
851	SIG4	0575
1556	SIG5	0A03
222	SIGDOS	003B
375	SIMANUAL	0164
127	SINGS1	1485
25	SINGS1P	0B3F
347	SIPAUTO	012F

Control de temperatura con lógica difusa

252	TECO	007F
454	TECACIAA	0205
429	TECCERO	01DB
455	TECDECIN	0208
187	TECLA	0040
433	TECLACO	01E1
419	TECLACON	01BF
435	TECLAIN	01E5
150	TECLAT	000D
902	TECLET	05E6
462	TECMAS	0215
471	TECREG	022A
122	TEMSEL	1400
1554	TIEMPO1	0A00
893	TIN91	05C9
900	TNA1	05DF
161	TOPE	0015
1114	VAL0	0755
1106	VAL2	0740
178	VALBP	009A
1129	VALFIN	076B
1126	VALNEG	0767
1093	VALJA	0724
342	VAPROG	0122
132	VAR1H	0001
131	VAR1L	0000
134	VAR2H	0003
133	VAR2L	0002
136	VAR3H	0005
135	VAR3L	0004
140	VART1	0007
141	VART2	0008
183	WORDCCN1	003E
184	WORDCCN2	00BA
185	WORDCCN3	007A

No. of lines : 1561
No. of errors : 0

