

## MICROCOMPUTADORA BASICA DE DESARROLLO

" M I C D Z "

Rodriguez Espinosa M. Rodriguez Rossini G. Rojas Villanueva M.  
Depto. Instrumentacion Electronica. Desarrollo Tecnológico.  
Instituto Nacional de Cardiología " Ignacio Chavez ".

## R E S U M E N

La MICDZ es una microcomputadora enfocada al desarrollo de instrumentación electrónica en base al microprocesador Z80, así mismo esta dirigida a la enseñanza del lenguaje ensamblador y técnicas de control automatizado.

En el trabajo se explica la estructura de organización del programa monitor junto con sus rutinas, el mapa de memoria, el mapa de puertos y modos de operación de la máquina.

## INTRODUCCION.

El alcance y potencialidad de una microcomputadora cubre áreas tan diversas como pueden ser: la industrial, la médica, la administrativa e instrumentación en general. Dirigiendo nuestra atención al desarrollo de instrumentación computarizada dedicada tenemos las siguientes soluciones: el usar sistemas comerciales avanzados o tabletas de evaluación, en el primer caso resultan ser de muy alto costo, mientras que las tabletas resultan demasiado limitadas y a su vez implican el uso forzoso de componentes innecesarios para la aplicación una vez finalizado el diseño. La idea del presente trabajo es poner a juicio los conceptos y posibles alcances de una microcomputadora básica de desarrollo.

## ESTRUCTURA DE LA MICROCOMPUTADORA MICDZ.

Cuenta con un programa monitor configurado en base a estructuras jerárquicas que permiten organizar la programación según el nivel de complejidad.

Para explotar los recursos de la microcomputadora cuenta con comandos de control simples que son activados desde un teclado de membrana de retroalimentación auditiva, así mismo cuenta con un display pseudoalfanumérico de seis caracteres y dos arreglos de diodos como indicadores de registros internos de la máquina.

La microcomputadora MICDZ esencialmente está constituido por dos tabletas de circuito impreso, una contiene el procesador CPU, el cronometro/contador CTC, memoria ram, memoria eprom, decodificador de puertos y lógica de control, la otra constituye la interfase de interacción con el usuario, ésta maneja el teclado y despliegue pseudoalfanumérico, estas dos tabletas se comunican entre sí a través de un conector, marcado en el circuito impreso como J3.

Los conectores son para el tipo de expulsión rápida y las señales que se pueden tomar de ellos se organizan según la fig. 1

1	D0	P3	26
2	D1	P2	25
3	D2	P1	24
4	D3	P0	23
5	D4	A2	22
6	D5	A1	21
7	D6	A0	20
8	D7	RD	19
9	INT	WR	18
10	NMI	RES	17
11	CLK	-12	16
12	+5	+12	15
13	GND	GND	14

FIG. 1 Señales de los conectores de MICDZ.

El número máximo de periféricos que se pueden controlar son ocho agrupados en dos conectores J1 y J3 las señales de decodificación que se pueden acceder desde cada uno de ellos son: P0, P1, P2, P3 y P0, P1, P2, P3 respectivamente la señal P0 del

conector J1 esta inaccesible al usuario por ser de uso interno, controlandose con esta al CTC. Por cada uno de estos periféricos se controlan hasta ocho puertos de escritura y ocho de lectura, las direcciones que corresponden a cada periférico se desglozan en la tabla 1.

Periférico	Puert. Salida/Escritura	Puert. Entrada/Lectura
J1-P1	10, 11, ... 17	10, 11, ... 17
J1-P2	20, 21, ... 27	20, 21, ... 27
J1-P3	30, 31, ... 37	30, 31, ... 37
J3-P0	40, 41, ... 47	40, 41, ... 47
J3-P1	50, 51, ... 57	50, 51, ... 57
J3-P2	60, 61, ... 67	60, 61, ... 67
J3-P3	70, 71, ... 77	70, 71, ... 77

TABLA 1. Puertos de la microcomputadora MIC-DZ.

Observese que MICDZ decodifica 8 grupos de periféricos con capacidad de 8 puertos de entrada y 8 puertos de salida cada uno dando un total de 64 puertos de los 256 puertos que puede direccionar el microprocesador Z80.

#### ORGANIZACION DE LA MEMORIA.

La memoria direccionable de la MICDZ es de 8 Kbytes de EPROM y 2 Kbyte de RAM, la memoria EPROM de sólo lectura o permanente esta organizada en dos grupos de 4 Kbytes cada uno debido a que el circuito integrado que se usa es de ésta capacidad (2732), la memoria RAM es un solo grupo de 2 Kbyte del tipo estático, formado por un circuito integrado (6116).

Las direcciones iniciales de cada grupo de memoria de sólo lectura y del grupo de escritura/lectura se organiza según la tabla 2.

GRUPO	DIR. INICIAL	DIR. FINAL	USO
Eprom0	0000	0FFF	Rutinas del monitor de MICDZ.
Eprom1	2000	2FFF	Rutinas de Aplicación y tablas.
Ram4	4000	47FF	Variabes, stack, usuario.

TABLA 2. Organización de memoria.

El grupo Eprom0 contiene, el programa monitor, rutinas de monitor, rutinas generales y rutinas básicas.

El grupo Eprom2 se reserva para el desarrollo del programa que determinará finalmente la aplicación, también se puede utilizar para tablas o constantes.

El grupo Ram4 almacena las variables de la micro y es donde se encuentra la zona libre para el usuario que es aproximadamente el 75 % del total de las localidades de RAM, las variables de la micro son: del monitor, de aritmética, del depurador y de aplicación, también en esta zona Ram existe la zona reservada para el STACK que es aprox. el 25 % de la Ram.

#### DISTRIBUCION DE LA MEMORIA RAM.

La distribución real de la memoria ram se desgloza en la tabla de distribución de memoria ram (tabla 3), la cual no debe ser modificada ya que es manejada por el programa monitor que reside en el grupo Eprom1.

47FF -> 47B0	(80 localidades)	Variabes de Monitor.
47AF -> 4780	(48 localidades)	Variabes de Aritmética.
477F -> 4750	(48 localidades)	Variabes de Aplicación.
474F -> 4650	(256 localidades)	Area del depurador.
464F -> 45F0	(96 localidades)	Area del Stack.
45EF -> 4000	(1520 localidades)	Area libre para el usuario

TABLA 3. Distribución de Memoria Ram.

Se sugiere que tanto esta distribución como el programa monitor y las rutinas del grupo Eprom1 no deben ser modificados ya que manejan tanto la máquina como el instrumento en desarrollo.

Las variables del instrumento que se este desarrollando deben

ocupar de preferencia las localidades marcadas en la tabla 3 con el nombre de Variables de Aplicación y en el último de los casos en la zona libre de usuario distribución de la memoria ram.

#### COMANDOS DE OPERACION.

Para explotar los recursos de la microcomputadora cuenta con comandos de control simples organizados como sigue:

Comandos principales.

Comandos de función.

Comandos auxiliares.

Para activar estos comandos se presiona la tecla marcada con la letra inicial de la función que se quiere realizar.

#### Comandos Principales.

##### Comando A.

Manda el control al Archivador donde se pueden realizar funciones de manejo de programas como son: Cargar, Depositar y Ejecutar programas, estas funciones también son activadas presionando la tecla marcada con la letra inicial de la función a realizar, para desactivar el Comando A se logra presionando la letra F (Fin).

##### Comando C.

Manda el control a Calculadora donde se pueden realizar funciones aritméticas y lógicas, así como cálculo de desplazamientos relativos. Se puede trabajar en sistema decimal, hexadecimal y octal.

##### Comando D.

Manda el control al Depurador donde se pueden realizar funciones de manejo de registros del Z80, correr programas en diferentes zonas de memoria, ejecución de programas paso a paso escritura y lectura de puertos.

##### Comando E.

Manda el control al Editor donde se realizan funciones de manejo de memoria de la micro NICDZ como son: agregar, borrar y

cambiar códigos, desplazar bloques de memoria de una zona a otra, encontrar direcciones al azar y visualizar cualquier región de memoria incrementando y decrementando la dirección de la localidad desplegada por medio de las teclas de control de movimiento (flechas).

Comandos de función.

Comandos Fn.

Los comandos de función F1, F2, F3 y F4 mandan el control a diferentes zonas de memoria, donde se ejecutarán los programas que ahí se encuentren, específicamente es a Eprom2 en las direcciones 2000, 2400, 2800, 2C00 respectivamente.

Comandos auxiliares.

Los comandos auxiliares son para ejecutar programas de autodiagnóstico, pruebas de la máquina y ejecución de programas desarrollados por el usuario en la memoria de trabajo de la máquina.

#### OPERACION Y USO DE LA MICROCOMPUTADORA MICDZ.

Se puede usar de tres modos:

- a). Generando los programas directamente en la zona Ram4, ejecutándolos y depurándolos por medio del programa y rutinas del monitor de la MICDZ.
- b). Transfiriendo programas de zona Eprom2 a la zona Ram4 para depurarlos y corregirlos o en caso contrario ejecutándolos directamente en esa zona por lo que se debe tener un programa control generado por el usuario que permita esta función.
- c). Considerando la Eprom2 sólo como almacén de rutinas depuradas de diferentes aplicaciones para ser llamadas y ejecutadas desde la memoria de trabajo de la máquina (Ram4).

Los pasos a seguir en el modo "a" son:

1. Encienda la máquina. Después de encendida se producirá una tonada de bienvenida y aparecerá el mensaje A.C.D.E.
2. Presione un comando principal o de función (es decir una de

las letras que aparecen en el display: A, C, D, E, o una Fn ).

3. Otras teclas que se pueden presionar son: 1, 2, 3, 4, 5, ENTRADA y C/P, si se presiona cualquier otra será ignorada.

4. Si se presiona una tecla del paso 2 tenemos:

[A]. Manda el control al Archivador donde se podrán realizar las funciones de manejo de programas como son: Cargar, Depositar, Ejecutar, para lograr una de estas funciones se debe presionar la letra inicial de ellas, para salir de este modo presione la tecla F de fin.

[C]. Manda el control a Calculadora en este modo se pueden realizar operaciones aritméticas en sistema decimal o hexadecimal; Al entrar al modo Calculadora estaremos en sistema decimal y sólo se podrán ejecutar operaciones enteras, si se presiona C/P el número que este en el display cambia a hexadecimal, quedando la máquina en este sistema, se vuelve a presionar C/P el número que está en el display pasa a decimal, para salir de este modo estando en sistema decimal presione la tecla F de fin.

[D]. Manda el control a Depurador donde se pueden realizar funciones de manejo de registros del Z80, ejecución de programas en pasos, Escritura y Lectura de puertos; Para realizar estas funciones se debe presionar una tecla que se relacione con la función deseada.

Al iniciar el comando D o Depurador aparecerá el mensaje "DEPU." un instante e inmediatamente después un número hexadecimal de cuatro caracteres y las letras AF, esto significa que los dos primeros caracteres corresponden al contenido del registro A ó Acumulador y los otros dos al contenido de las banderas (Flags); En este estado si presiona las teclas ENTRADA o ↵ se desplegará el siguiente par de registros BC y así sucesivamente hasta desplegar todos los pares de registros pasando por los primos y repitiéndose la secuencia si se continua presionando cualquiera de esas dos teclas, si se quiere el despliegue en el sentido contrario debe presionarse la tecla !; En cualquier posición de despliegue de registros si se presiona

la tecla A se actualizará al par AF, sucede lo mismo si se presiona la tecla C/P con la diferencia de que se indica que se esta en el comando Depurador por medio del mensaje "dEPU."; Para salir del comando D debe presionarse la tecla F de Fin regresando a los comandos principales o menú base.

Si se presiona la tecla C ó Cambia en cualquier posición de despliegue de registros se pasa al comando de Cambio de contenido de registros, para salir de éste comando presione C/P.

La tecla D ó Despliegue de puertos pasa el control a Puertos apareciendo el mensaje Pu00 y contenido del puerto es decir queda en el modo de entrada o lectura; Para cambiar a modo de salida o escritura presione la tecla O donde permanecerá el mensaje Pu00 pidiendo el dato a sacar por el puerto. Con las teclas ENTRADA, ↑, ↓ se incrementa o decrementa el número del puerto desplegado, con la tecla C/P se sale del comando D ó Despliegue de puertos.

[E]. Manda el control a Editor donde se realizan funciones de manejo de memoria de la MICDZ como son: agregar, borrar y cambiar códigos, desplazar bloques de memoria de una zona a otra, encontrar direcciones al azar y visualizar cualquier región de memoria incrementando y decrementando la dirección de la localidad desplegada por medio de comandos simples.

Al iniciar el comando E o Editor aparecerá el mensaje "Edit." e inmediatamente después el número hexadecimal 4000 y un par de caracteres, ésto es, una dirección de la zona de memoria Ram4 y su contenido, esta dirección es la primera localidad que se puede acceder en la zona RAM.

Por medio de las teclas ENTRADA, ↑ y ↓ se puede posicionar cualquier dirección en incrementos o decrementos de la dirección desplegada, si se quiere pasar a una dirección arbitraria presione la tecla E de Encontrar, la máquina preguntará la dirección por medio del mensaje "dir.", de la dirección y presione la tecla ENTRADA, inmediatamente después aparecerá el contenido de esa dirección. La zona de

Ram4 libre para el usuario o de trabajo esta limitada por las direcciones 4000 y 45EF por lo que se debe tener cuidado de no cambiar contenidos fuera de este rango pues serán ignorados por la máquina o se alterará seriamente la ejecución del programa. Estando en el rango de trabajo se puede agregar, borrar y cambiar códigos de memoria para lograr esto basta con presionar la letra inicial de cada función es decir A para agregar, B para borrar, C para cambiar, en esta zona y fuera de ella se puede desplazar bloques de memoria o encontrar direcciones presionando las teclas D para desplazar y E para encontrar.

[A] Agregar.

En este modo aparece la dirección y el contenido de esa localidad direccionada, el contenido se apaga después de un instante, prendiéndose el cursor pidiéndose de esa forma el código a ser agregado. Los códigos que se encuentren en las direcciones siguientes es decir las direcciones altas son recorridas para hacer espacio al nuevo código para salir de este comando presione la tecla C/P.

[B] Borrar.

Cualquier dirección que se este desplegando estará acompañada de su contenido, en este estado si se presiona la tecla B se borrará el contenido presente de esa localidad direccionada, reescribiéndose los códigos siguientes (direcciones altas) a partir de la dirección desplegada, se puede presionar cuantas veces sea necesario esta tecla borrándose los códigos presentes.

[C] Cambiar.

En este modo aparece la dirección y el contenido de esa localidad direccionada, el contenido se apaga después de un instante, prendiéndose el cursor pidiéndose de esa forma el código a ser cambiado, para lograr esto de el código y la tecla ENTRADA, inmediatamente después se despliega la siguiente dirección repitiéndose las operaciones anteriores, de esta forma se pueden cambiar los códigos que sean

necesarios, para salir de este modo presione la tecla C/P.

5. Si se presiona una de las teclas del paso 3 tenemos:

[1]. Se ejecuta un programa de prueba de teclado, desplegándose en el display el símbolo del carácter presionado y un símbolo raro que corresponde al valor posicional de bits prendidos o apagados según el número hexadecimal que corresponde a cada carácter. Los símbolos desplegados corresponden a una asignación de caracteres según la máscara del teclado usado, característica muy importante en el caso de usar el teclado de aplicación que generalmente es diferente al de la micro MICDZ.

[2]. Se ejecuta un programa de prueba de displays barriéndose segmento por segmento de cada carácter y de los arreglos de leds del acumulador y de banderas.

[3]. Se ejecuta un programa de despliegue de registro acumulador y estatus de la maquina en el arreglo de indicadores luminosos para este fin.

[4]. [5]. Para probar puertos de una manera inmediata de la micro a través de los conectores J1 y J3, 4 en escritura y 5 en lectura.

[ENTRADA]. Si se presiona esta tecla se ejecutan las instrucciones de máquina que se encuentren a partir de la dirección 4000 de la zona RAM. Para regresar al programa monitor después de ejecutar esas instrucciones insertese al final la instrucción RST 38 la cual obligará la ejecución de una rutina cuya función es salvar el estado final de los registros del microprocesador Z80 y mandar el control al programa monitor de la máquina.

[C/P]. Al presionar esta tecla se reinicia la máquina.

6. Los modos b y c de trabajo de la máquina son idénticos a los del modo a, con la diferencia de la habilidad del usuario y de la experiencia que se tenga con el uso de la misma.

## RUTINAS DE LA MICROCOMPUTADORA MUCDI.

Las rutinas están organizadas por complejidad de función y en forma modular. Las rutinas van desde funciones básicas como es el control de los circuitos electrónicos hasta la operación automatizada del instrumento en desarrollo. El flujo y organización de las rutinas se muestran en la fig. 2.

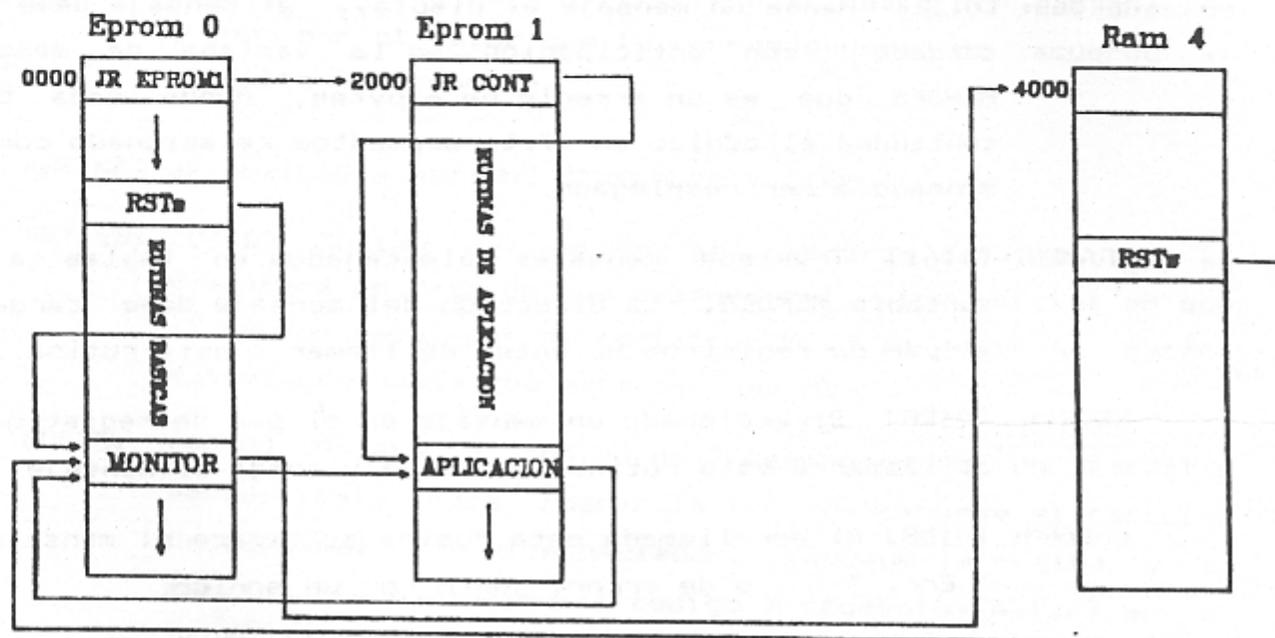


FIG. 2. Organización y flujo de las rutinas de la DZ.

Las rutinas cumplen varios objetivos como son: generación de sonidos, manejo de mensajes, manejo del teclado, de auxilio a los circuitos electrónicos, manejo de sistema decimal y hexadecimal, de aritmética entera y de apoyo al programa monitor, en seguida se desglosa cada una de ellas indicándose entre parentesis cuadrados la dirección física de su localización en la memoria permanente de la máquina (grupo Eprom1).

## RUTINAS DE GENERACION DE SONIDOS.

CAMP1: [0102] Genera un sonido agudo.

CAMP2: [010B] Genera un sonido grave.

TONO: [0114] Genera un sonido de cualquier tono.

Para usar esta rutina debe cargarse antes el par de registros DE con los parámetros de duración y tono donde

el registro D llevará la duración y E el Tono, los valores posibles para éstos parámetros van de 00 a FF.

TONENT: [0128] Genera varios tono generalmente se usa como tonada de entrada o de bienvenida.

#### RUTINAS DE MANEJO DE MENSAJES.

REFDES: [013E] Manda un mensaje al display, el mensaje debe ser cargado con anticipación en la variable de memoria MEMDES que es un arreglo de 6 bytes, donde cada byte contendrá el código en siete-segmentos relacionado con el mensaje a ser desplegado.

TRAMEN: [0161] Traslada mensajes almacenados en tablas a la variable MEMDES, la dirección del mensaje debe cargarse al par de registros HL antes de llamar a esta rutina.

ESCRI: [01BE] Direccionado un mensaje en el par de registros HL al llamar a esta rutina se prende y apaga el mensaje.

ERRORM: [01D8] Al ser llamada esta rutina se genera el mensaje " Err. " o de error, junto con un sonido.

#### RUTINAS DE ACTIVACION Y MANEJO DEL TECLADO.

PRELED: [01F2] Prende LEDs de tecla de función siempre que el valor de la tecla este entre 10H y 1FH.

LECAR: [0221] Lee el teclado, activa el display y permanece en este estado hasta que sea presionada una tecla, para entregar el valor de esta en el registro Acumulador.

LECAR2: [0261] Lee el teclado, activa el display, después de un tiempo aproximado de 6 seg. si no se presiona ninguna tecla la rutina es abortada entregando el carácter especial 83.

MUESTE: [02B1] Lee el teclado una sola vez, si durante la lectura es presionada una tecla se modifican dos variables de memoria que son: MEMTEC donde se actualiza el valor de la tecla presionada y en BANTEC se almacena un 1 si se presiono cualquier tecla y 0 si no es presionada ninguna.

TECDZ: [02FA] Rutina llamada por LECAR y MUESTE, su función es entregar un caracter equivalente a la distribución de teclas en la máscara del teclado es decir por asignación y no por el valor que le corresponderia de acuerdo al circuito electrónico.

#### RUTINAS DE AUXILIO A RUTINAS GENERALES Y CIRCUITOS.

RET16B: [034B] Rutina de retardo al invocarse esta rutina se ejecutará un retardo que depende del valor inicial en que se encuentre el par de registros HL, es decir se estará ejecutando hasta que este par sea 0.

COO7S: [0350] Genera el código de 7-segmentos de un carácter hexadecimal. Para lograr la función cargese el caracter hexadecimal en el Acumulador, invoque la rutina y al terminar la ejecución el código 7-segmentos estará en el Acumulador nuevamente.

LIMMEM: [0385] Rutina para limpiar n localidades de memoria contiguos a partir de la dirección inicial dada en el par de registros DE.

ACTMEM: [0392] Rutina para actualizar n localidades de memoria, transfiriendo la información de una localidad baja a la siguiente alta inmediata.

#### RUTINAS GENERALES SISTEMA HEXADECIMAL Y DECIMAL.

LEEHEX: [03A0] Lee un número hexadecimal en el rango de 0000 a FFFF a través del teclado diferenciando tres tipos de teclas: de caracteres hexadecimales, de entrada, de escape y especiales. El número a ser leído se teclea

carácter por carácter desplegándose al mismo tiempo en el display, solo es válido el número que se visualiza hasta antes de ser presionada la tecla [ENTRADA]. El número leído es almacenado en la variable MEMTEC.

DESHEX: [0417] Rutina para desplegar el contenido de una localidad de memoria (en hexadecimal), la dirección de la localidad debe ser cargada en el par de registros HL.

LEEDEC: [043D] Lee un número entero decimal entre 0000 y 9999 a través del teclado diferenciando tres tipos de teclas: de caracteres decimales, de entrada, de escape y especiales. El número a ser leído se teclea dígito por dígito desplegándose al mismo tiempo en el display, solo es válido el número que se visualiza hasta antes de ser presionada la tecla [ENTRADA]. El número leído es almacenado en la variable MEMTEC.

DESDEC: [04C3] Rutina para desplegar el contenido de una localidad de memoria en sistema decimal, la dirección de la localidad debe ser cargada en el par de registros HL antes de llamar a esta rutina.

DESDIR: [051F] Despliega el contenido del par de registros HL es decir puede ser un número hexadecimal entre 0000 y FFFF.

LEBYTE: [055C] Lee un byte, es decir un número entre 00 y FF, entregándolo en el Acumulador, la rutina discrimina entre tres tipos de teclas: caracteres hexadecimales, de entrada, de escape y especiales. El número a ser leído se teclea carácter por carácter desplegándose al mismo tiempo en el display siendo solo válido el byte que se visualice hasta antes de ser presionada la tecla [ENTRADA].

#### RUTINAS DE ARITMETICA.

SUMA: [05C1] Suma dos palabras de 32 bites a través de la memoria, es decir toma dos bloques de cuatro bytes que son las variables SUM1 y SUM2 suma byte a byte

considerando el carry y almacenando el resultado en la variable SUM1. La rutina puede ser expandida a sumas de palabras de n bits.

RESTA: [05D8] Resta dos palabras de 32 bits a través de la memoria, es decir toma dos bloques de cuatro bytes que son las variables SUM1 y SUM2 donde SUM1 es el minuendo y SUM2 el sustraendo resta byte a byte de SUM2 a SUM1 considerando el carry y almacenando el resultado en la variable SUM1. La rutina puede ser expandida a restas de palabras de n bits.

MULTIP: [05F7] Multiplica palabras de 16 bits, donde los factores se almacenan en las variables MDOR y MANDO, el resultado es almacenado en la variable SUM1. Esta operación se realiza por corrimientos o método de ponderación.

DIV: [0649] Divide palabras de 32 bits, donde el dividendo y divisor deben ser almacenados en las variables DIVNDO y DIVSOR, el resultado de la operación así como el residuo se entregan en las variables COCIEN y RESIDU. Esta operación se realiza por restas sucesivas.

LBLOCK: [0665] Esta rutina es una auxiliar a las rutinas de aritmética y se utiliza para limpiar o reiniciar variables aritméticas en cero.

#### RUTINAS DEL PROGRAMA MONITOR:

El programa monitor PROMON [0083] maneja y administra sus propias rutinas las cuales están distribuidas en cuatro grupos principales que son: ARCH, CALC, DEPU y EDIT. El programa monitor también puede ceder el control a un programa manejador de aplicación ó a un programa que se encuentre en la memoria ram.

DEPU: [ A)cum, C)ambReg, D)espPu, F)in ].

Rutina de uso interno de la micro DZ, su función es de depuración de programas, auxiliado por el manejo de puertos y registros.

EDIT: [ A)grega, B)orra, C)ambia, D)esplaza, E)ncontrar, F)in ]  
Rutina de uso interno de la micro MICDZ, su función es de manejo de información ó códigos de máquina. La rutina esta organizada en sub-grupos conocidos como: ANADE, BORRA, CAMBIA, DESBLO, ENCDIR además de esta subdivisión cuenta con las características de visualizar información de la memoria, de regresar al menú principal y de indicar el nivel donde se encuentra la máquina en caso de confusión de usuario.

TEXREG: Rutina para manejar textos de registros o mensajes de identificación de cada par de registros de procesador Z80

#### RUTINAS DE DIAGNOSTICO Y AUTOPRUEBA.

TECLAS: [0962] Rutina de prueba de teclado, despues de encendida la máquina presione la tecla [1] y entra esta rutina. Ver operación y uso (modo a inciso 3 y 5).

DESPLI: [0981] Rutina de prueba de displays, idem que rutina TECLAS presionando tecla [2].

DIAG3: [099D] Rutina de despliegue de Acum. y Banderas, idem que rutina TECLAS presionando tecla [3].

DIAG4: [09AB] Rutina de prueba inmediata de puertos en escritura, idem que rutina TECLAS presionando tecla [4].

DIAG5: [09B9] Rutina de prueba inmediata de puertos en lectura, idem que rutina TECLAS presionando tecla [5].

## ESPECIFICACIONES TECNICAS.

PROCESADOR	200 CPU de 4 MHz.
CRONOMETRO/CONTADORES	280 CTC.
MEMORIA RAM	2 KBytes CDM 6116.
MEMORIA EPROM (programa monitor)	4 KBytes 1MS2732.
MEMORIA EPROM socket (opcional)	4 KBytes
Decodificador	Para 8 periféricos en grupos de 8 puertos de entrada y 8 de salida.
Controlador de teclado	Con retroalimentación audible y capacidad de control para 64 teclas.
Membrana de teclado	16 teclas de caracteres hexadecimales. 2 de control y 5 de operaciones lógicas. 4 de funciones. 1 de Reset y 1 de Wait ocultas.
Teclado opcional	No. de teclas opcional máximo 64. 1 de Reset y 1 de Wait.
Despliegue	Seudo-alfanumérico 6 caracteres. Arreglo de leds indicador de Acumulador. Arreglo de leds indicadores de Banderas.
Alimentación de +5 volts 1 amp.	

Dimensiones: 2 tabletas de circuito impreso de 19.5 x 8.0 cm

La microcomputador puede ser expandida a 16 KBytes de RAM y 8 KBytes de EPROM ó 16 KBytes de EPROM y 8 KBytes de RAM.