

Z80 DSK

Manual del usuario

Gabriel Rosso
Rafael Meza

Mendoza, Agosto de 1996

1. Índice

1.1. General

1. INDICE	2
1.1. General	2
1.2. Tablas	3
1.3. Gráficos	3
1.4. Diagramas	3
2. INTRODUCCIÓN	5
3. COMPOSICIÓN	5
4. REQUERIMIENTOS PARA SU OPERACIÓN	5
4.1. Hardware:	5
4.2. Software:	5
5. CONEXIÓN E INSTALACIÓN DEL SOFTWARE DE PC	5
5.1. Conexión:	5
5.2. Instalación:	5
6. DESCRIPCIÓN DEL SISTEMA	6
6.1. Parte compuesta por la placa de desarrollo:	6
6.1.1. Hardware	6
6.1.1.1. Introducción	6
6.1.1.2. Descripción	6
6.1.1.2.1. Componentes:.....	6
6.1.1.2.1.1. CPU:	6
6.1.1.2.1.2. Memoria:	6
6.1.1.2.1.3. Buffers:	7
6.1.1.2.1.4. Decodificadores:	7
6.1.1.2.1.5. Inversores:	7
6.1.1.2.1.6. Interface RS232C:.....	7
6.1.1.2.2. Capacitores:.....	7
6.1.1.2.3. Led:.....	7
6.1.1.2.4. Jumpers:	7
6.1.1.2.5. Osciladores:	8
6.1.1.2.6. Switch de reset:.....	8
6.1.1.2.7. Buses de Expansión:.....	8
6.1.1.2.8. Alimentación y comunicación serie:.....	8
6.1.1.3. Disponibilidades del usuario:.....	8
6.1.1.4. Mapa de memoria	9
6.1.1.5. Mapa de Entrada / Salida	9
6.1.1.6. Circuitos Esquemáticos.....	9
6.1.2. Software.....	9
6.1.2.1. Introducción	9
6.1.2.2. Descripción:	10
6.1.2.3. Disponibilidades del usuario:.....	10
6.2. Parte compuesta por la PC	10
6.2.1. Introducción	10
6.2.2. Descripción:	10
6.2.2.1. Composición	10
6.2.2.2. Ejecución:	10
7. OPERACIÓN DEL SISTEMA	11
7.1. Descripción de la ventana de administración del sistema:	12
7.2. Descripción de la inicialización:	12
7.3. Descripción de los menús:	13

7.3.1. Menú Archivos	13
7.3.1.1. Submenú Nuevo:	13
7.3.1.2. Submenú Abrir:	13
7.3.1.3. Submenú Guardar:.....	14
7.3.1.4. Submenú Guardar Como...:	14
7.3.1.5. Submenú Imprimir:.....	15
7.3.1.6. Submenú Salir:.....	16
7.3.2. Menú Editar:.....	16
7.3.3. Menú Memoria:	16
7.3.3.1. Submenú Definir...:	17
7.3.3.2. Submenú Mostrar:.....	17
7.3.3.2.1. Comando Mostrar Datos:	18
7.3.3.2.2. Comando Mostrar Programa del Usuario:	18
7.3.3.2.3. Comando Mostrar Programa del Sistema:	19
7.3.3.3. Submenú Llenar	20
7.3.3.4. Submenú Comparar	20
7.3.3.5. Submenú Trasladar:.....	22
7.3.4. Menú Sistema	22
7.3.4.1. Submenú Registros.....	23
7.3.4.1.1. Comando Registros Reset	23
7.3.4.1.2. Comando Registros Examinar.....	23
7.3.4.2. Submenú Definir	23
7.3.4.3. Submenú Ejecutar.....	24
7.3.4.3.1. Comando Ejecutar en Tiempo Real	24
7.3.4.3.2. Comando Ejecutar en Tiempo Virtual.....	24
7.3.4.3.3. Comando Ejecutar Paso A Paso	25
7.3.4.3.4. Comando Ejecutar con Break Point	25
7.3.4.4. Comando Detener.....	25
7.3.4.5. Submenú Ensamblar.....	25
7.3.5. Menú Puertos.....	26
7.3.5.1. Submenú Ver Contenido	26
7.3.5.2. Submenú Escribir	27
7.3.6. Menú Opciones	27
7.3.7. Menú Ventanas	27
7.3.8. Menú Ayuda	28
8. EJEMPLOS DE APLICACIONES	29
8.1. Aplicación con el PIO.....	29
8.2. Aplicación con el CTC	31
9. VENTAJAS DE TRABAJAR EN ENTORNO WINDOWS.....	33
10. BIBLIOGRAFÍA	34
11. APÉNDICE A: PROTOTIPO DEL Z80 DEVELOPMENT SYSTEM KIT	35
12. APÉNDICE B: ESQUEMÁTICOS	36

1.2. Tablas

TABLA 1: JUMPERS.....	8
TABLA 2: CONEXIÓN SERIE	8
TABLA 3: MAPA DE MEMORIA.....	9
TABLA 4: MAPA DE ENTRADA/SALIDA	9

1.3. Gráficos

FIGURA 1: ICONO INICIAL	11
FIGURA 2: DEFINIR PUERTO SERIE	11
FIGURA 3: MENSAJE DEL RESET	11

FIGURA 4: VENTANA PRINCIPAL	12
FIGURA 5: MENÚ ARCHIVOS.....	13
FIGURA 6: ABRIR ARCHIVOS	14
FIGURA 7: LÍMITES DEL BLOQUE A GUARDAR	14
FIGURA 8: GUARDAR ARCHIVO.....	15
FIGURA 9: OPCIONES DE IMPRESIÓN	15
FIGURA 10: MENÚ EDITAR	16
FIGURA 11: MENÚ MEMORIA	16
FIGURA 12: DEFINICIÓN DEL COMIENZO DE LOS BLOQUES DE MEMORIA	17
FIGURA 13: OPCIONES PARA VER LA MEMORIA.....	17
FIGURA 14: DATOS DE LA MEMORIA	18
FIGURA 15: OPCIONES PARA VER EL PROGRAMA DEL USUARIO	19
FIGURA 16: PROGRAMA DE MEMORIA EN MNEMÓNICO	19
FIGURA 17: PROGRAMA DEL SISTEMA	20
FIGURA 18: LLENAR MEMORIA.....	20
FIGURA 19: BLOQUES A COMPARAR.....	21
FIGURA 20: MENSAJE POR IMPOSIBILIDAD DE COMPARACIÓN	21
FIGURA 21: MENSAJE DE BLOQUES IGUALES	21
FIGURA 22: MENSAJE DE BLOQUES DESIGUALES	21
FIGURA 23: TRASLADAR BLOQUE DE MEMORIA.....	22
FIGURA 24: MENÚ SISTEMA.....	22
FIGURA 25: SUBMENÚ REGISTROS.....	23
FIGURA 26: CONTENIDO DE LOS REGISTROS	23
FIGURA 27: PARÁMETROS PARA EL BREAK POINT	24
FIGURA 28: SUBMENÚ EJECUTAR	24
FIGURA 29: INSTRUCCIÓN A ENSAMBLAR.....	25
FIGURA 30: MENÚ PUERTOS.....	26
FIGURA 31: CONTENIDO DE LOS PUERTOS	26
FIGURA 32: PROGRAMAR PUERTO	27
FIGURA 33: SUBMENÚ OPCIONES	27
FIGURA 34: MENÚ VENTANAS.....	28
FIGURA 35: SUBMENÚ AYUDA.....	28
FIGURA 36: INFORMACIÓN DEL SISTEMA	29

1.4. Diagramas

DIAGRAMA 1: APLICACIÓN PARA EL PIO	30
DIAGRAMA 2: APLICACIÓN PARA EL CTC	32

2. Introducción

El Z80 Development System Kit consiste en un sistema, el cual posibilita y facilita el análisis, desarrollo, implementación y verificación de programas y sistemas basados en el microprocesador Z80 y periféricos de la línea Zilog.

El Z80-DSK está conectado a una PC a través del canal de acceso serie, con lo cual se permite el acceso remoto del sistema. El sistema se encuentra administrado por un software operado a través de la PC actuando bajo entorno Windows.

3. Composición

El Z80 Development System Kit se compone de las siguientes partes constitutivas:

- Placa de desarrollo para microprocesador Z80 con la circuitería suficiente y estándar para lograr un estudio acabado del procesador.
- Conexión serie RS232C a PC.
- Software de PC (ejecutable bajo entorno Windows).
- Software de placa (dentro de EPROM).

4. Requerimientos para su operación

4.1. Hardware:

El Z80 Development System Kit necesita para su operación una PC AT 386 o superior con canal de acceso serie a través de un conector DB9. Se necesita también una disketera de 3 ½ para la ejecución del software de PC.

4.2. Software:

El Z80-DSK es administrado a través de un software dedicado que opera en entorno Windows 3.1 o superior.

5. Conexión e instalación del software de PC

5.1. Conexión:

Para operar el Z80 Development System Kit hay que alimentar la placa de desarrollo con 5 Voltios (con una tolerancia del 5%), interconectarla a la PC a través de un conector DB9 estándar y ejecutar el software de PC que corre bajo entorno Windows.

5.2. Instalación:

El software de PC se provee en un diskette de 3 ½. En el mismo se encuentran los programas necesarios para su correcta instalación. Este software se opera en entorno Windows, por lo que para su instalación se deberán seguir los siguientes pasos:

1. Ejecutar el sistema operativo Windows.
2. Insertar el diskette donde se encuentra el software de PC.
3. Del administrador de programas seleccionar el menú Archivos y elegir Ejecutar.
4. Escriba a:\setup y pulse ENTER.

Luego de unos instantes el programa de instalación le preguntará si desea cambiar el directorio de trabajo del sistema.

Al final de la instalación verifique que se haya creado un nuevo grupo de programas con un icono identificatorio en su interior referente al sistema.

6. Descripción del sistema

El Z80 Development System Kit consta de dos partes interdependientes, a saber: una parte la compone la PC y la otra parte la placa de desarrollo. A continuación se describirá a cada una de ellas en detalle:

6.1. Parte compuesta por la placa de desarrollo:

La placa de desarrollo es un diseño de hardware operado a través de un software dedicado a la administración de la misma.

6.1.1. Hardware

6.1.1.1.Introducción

La placa de desarrollo del Z80 Development System Kit consiste en un PCB de 24 cm de largo por 16,5 cm de ancho, en la cual se encuentra toda la circuitería necesaria para permitir el análisis, desarrollo, implementación y verificación de programas y sistemas basados en el microprocesador Z80 y periféricos de la línea Zilog. Para llevar a cabo dicha función, dentro de la misma se encuentran los siguientes componentes:

- Unidad Central de Proceso (CPU).
- Memoria Ram y Rom seleccionables.
- Interface paralelo PIO.
- Interface serie SIO.
- Temporizador CTC.
- Bus Buffers.
- Decodificadores.
- Osciladores para el reloj de la CPU y para el temporizador.
- Jumpers para la selección del modo de operación de la placa.
- Pulsador para la inicialización del sistema.
- Bus de expansión de periféricos de entrada-salida.
- Bus de expansión de memoria.
- Conexión serie RS232C a PC.
- Entrada de alimentación multipunto.

6.1.1.2.Descripción

Se hará una descripción detallada de la composición de la placa, función de los componentes y características de los mismos. A medida que se vaya nombrando cada componente se indicará el nombre que lo identifica dentro de la placa.

6.1.1.2.1.Componentes:

6.1.1.2.1.1.CPU:

La placa de desarrollo del Z80 Development System Kit tiene en su estructura principal un microprocesador Z80A de 4 (cuatro) Mhz. (U8) cuya función es la de organizar el sistema a través de un software de placa residente en EPROM (U3) y permitir al usuario la organización de sistemas propios actuando de acuerdo al software que se escriba para tal propósito.

La placa se comunica con el exterior de manera serie y de manera paralela. Esto lo hace a través de dos integrados: el Z80A SIO (Serial Input Output) de 4 Mhz (U12) y el Z80A PIO (Parallel Input Output) de 4 Mhz (U11).

Para lograr la temporización de la comunicación serie se utiliza un Z80A CTC (Counter Timer Controller) de 4 Mhz. (U13) que permite la programación de la velocidad requerida para la comunicación.

6.1.1.2.1.2.Memoria:

La memoria del sistema está compuesta por cuatro bancos de 8 KBytes cada uno, formando así un banco total de memoria de 32 Kbytes. Cada banco de memoria de 8 Kbytes esta formado por un integrado (U3 - U6), el banco n°1 (U3) está constituido por el integrado 2764A (EPROM) cuya función es la de albergar el software de placa. El banco n°2 (U4) puede estar compuesto por un integrado 2764A (ampliar la cantidad de memoria no volátil) o dar comienzo a la memoria RAM (integrado 6264), la selección se consigue a través del jumper W5. Los bancos n°3 (U5) y n°4 (U6) están compuestos por integrados 6264 (RAM).

6.1.1.2.1.3. Buffers:

La CPU se encuentra aislada del resto de la placa a través de buffers, esto se hace para asegurar la correcta operación de la misma bajo condiciones de carga máxima. Los integrados que cumplen esta función son los 74LS245 (U2, U9, U10, U17). El U10 es el encargado del bus de datos, el U2 y el U9 son los encargados del bus de dirección y el U17 es el encargado de las señales de control.

6.1.1.2.1.4. Decodificadores:

Los decodificadores 74LS138 (U7 y U14) se utilizan para poder conformar el mapa de memoria (U7) y el mapa de entrada-salida (U14).

6.1.1.2.1.5. Inversores:

El inversor lógico 74LS14 (U1) se utiliza para lograr una mejoría en la forma de onda del reloj para la CPU proveniente del oscilador a cristal de 4 Mhz. XTAL1.

6.1.1.2.1.6. Interface RS232C:

Para la comunicación serie y para atender a lo establecido por la norma RS232C se utiliza un MAX232 (U15) para lograr los niveles de tensión adecuados para la comunicación.

6.1.1.2.2. Capacitores:

Cada integrado tiene un capacitor de 0,1 microfaradio con el objeto de eliminar cualquier ruido de alta frecuencia que venga por la línea de alimentación del mismo.

6.1.1.2.3. Led:

La placa posee un led identificador de cuando la misma se encuentra con alimentación.

6.1.1.2.4. Jumpers:

Los jumpers son los encargados de permitir la elección de la forma de operación que se desea de la placa. Hay jumpers para la selección del modo de operación de la CPU (W1 - W4), para la conformación del banco de memoria (W5), para permitir que ciertas señales de los integrados se queden dentro de la placa o se permita su envío hacia el exterior (W6 - W12).

La utilización y el significado de cada conexión se puede ver en la siguiente tabla:

Jumper	Utilización	Conexión	
		1-2	2-3
W1	BUSRQ	No	Sí
W2	NMI	No	Sí
W3	INT	No	Sí
W4	WAIT	No	Sí
W5	EPROM 2	Sí	No
W6	ZC/T00	INT	EXT
W7	CLK/T0	INT	EXT
W8	CTSB	INT	EXT

W9	RTSB	INT	EXT
W10	TXDB	INT	EXT
W11	RXTXCB	INT	EXT
W12	RXDB	INT	EXT

Tabla 1: Jumpers

Nota: INT significa que se usa dentro de la placa y EXT que se utiliza fuera de la placa.

6.1.1.2.5.Osciladores:

La placa posee dos osciladores a cristal (XTAL1 y XTAL2). El XTAL1 se utiliza como reloj para la CPU y tiene una frecuencia de operación de 4 Mhz, mientras que el XTAL2 tiene una frecuencia de 1,8432 Mhz., ideal para atender las distintas velocidades estandarizadas para la comunicación serie.

6.1.1.2.6.Switch de reset:

La placa posee un switch que permite la inicialización de la CPU y del CTC en forma simultánea como lo permite el manual técnico de Zilog.

6.1.1.2.7.Buses de Expansión:

La placa posee dos buses de expansión del sistema (J1 y J2) los cuales permiten que la placa no sea cerrada en si misma, sino permitir la expansión del sistema hacia cualquier circuitería que sea considerada necesaria para llevar a cabo un diseño en particular. El bus J1 da la posibilidad de usar el bus de datos, el bus de direcciones y las patas de control de la CPU como también la posibilidad de ampliar el mapa de memoria (hasta 64 Kbytes) y el de entrada-salida (hasta 7FH), mientras que el bus J2 permite el uso de las señales provenientes de los integrados periféricos (PIO, SIO, CTC).

6.1.1.2.8.Alimentación y comunicación serie:

La placa se puede alimentar desde tres posiciones diferentes de acuerdo a las necesidades del usuario. Estas tres posiciones corresponden a tres de los cuatro lados que posee la placa, dejando el cuarto lado para permitir la conexión serie, cuya interconexión física es la siguiente:

PC	Placa
2	3
3	2
5	5
6	7
7	8
8	7

Tabla 2: Conexión serie

6.1.1.3.Disponibilidades del usuario:

A través del hardware el usuario puede utilizar toda la potencia de un sistema basado en el microprocesador Z80 trabajando a una velocidad de 4 Mhz. para desarrollar sistemas de acuerdo a su necesidad. Puede expandir el mapa de memoria, modificar la cantidad de memoria no volátil de acuerdo a su necesidad (W5), seleccionar modos de operación de la CPU (W1 - W4), exteriorizar o interiorizar señales provenientes de integrados dedicados a una función especial como la temporización y la comunicación serie (W6 - W12). También puede elegir la extensión del mapa de entrada/salida de acuerdo a los requisitos de su aplicación.

6.1.1.4. Mapa de memoria

0000 - 1FFF	ROM 0
2000 - 3FFF	ROM 1 - RAM 0
4000 - 5FFF	RAM 1
6000 - 7FFF	RAM 2
8000 - 9FFF	Externo
A000 - BFFF	Externo
C000 - DFFF	Externo
E000 - FFFF	Externo

Tabla 3: Mapa de Memoria

6.1.1.5. Mapa de Entrada / Salida

00 - 0F	SIO
10 - 1F	CTC
20 - 2F	PIO
30 - 3F	Externo
40 - 4F	Externo
50 - 5F	Externo
60 - 6F	Externo
70 - 7F	Externo
80 - FF	No implementado

Tabla 4: Mapa de Entrada/Salida

6.1.1.6. Circuitos Esquemáticos

En el Apéndice B se puede ver los circuitos esquemáticos y PCB overlay de la placa de desarrollo.

6.1.2. Software

6.1.2.1. Introducción

El software de placa del Z80-DSK, que se encuentra en la EPROM de la placa (y en el archivo Z80-DSK.HEX), atiende las operaciones básicas de comunicación entre el software de PC y la placa en sí misma. Se encuentra a partir de la posición 0 (cero) de memoria y se extiende hasta la posición de memoria 04B3H, por lo que se utiliza un solo banco de 8 Kbytes para la memoria no volátil (EPROM).

Este software se compone de rutinas que atienden la comunicación de la placa con la PC, como asimismo de rutinas que llevan a cabo funciones básicas dentro de la placa, a saber:

Llenar un bloque de memoria con datos determinados.

- Enviar el contenido de un bloque de memoria a la PC.
- Llenar los puertos del sistema completo (usados y disponibles) con determinados valores.
- Enviar el contenido de los puertos del sistema completo (usados y disponibles) a la PC.
- Llenar los registros de la CPU con determinados valores.
- Enviar a la PC el contenido de los registros de la CPU.
- Ejecutar un programa a partir de una posición de memoria determinada.
- Ensambalar la dirección de bifurcación hacia una rutina que le devuelve el control de la placa a la PC después de la ejecución de un programa.

6.1.2.2.Descripción:

Para lograr un mejor entendimiento del software de placa del Z80 Development System Kit le sugerimos que se remita al Manual de Diseño del sistema.

6.1.2.3.Disponibilidades del usuario:

El usuario puede disponer de las rutinas que componen el software de placa del Z80 Development System Kit para atender a alguna necesidad en particular al desarrollar programas para sistemas propios. Para un mejor entendimiento y alcance de las mismas le sugerimos que se remita al Manual de Diseño del sistema.

6.2. Parte compuesta por la PC

6.2.1. Introducción

El Z80 Development System Kit se opera desde un software de PC que permite la admistración del sistema con la consiguiente explotación de su potencial.

6.2.2. Descripción:

6.2.2.1.Composición

El software de PC esta provisto en un diskette de 3 ½ pulgadas y posee los siguientes archivos:

- CMDIALOG.VB_
- COMMDLG.VB_
- MSCOMM.VB_
- SETUP.EXE
- SETUP.LST
- SETUP1.EX_
- SETUPKIT.DL_
- VBRUN300.DL_
- VER.DL_
- Z80-DSK.EX_

Todos estos archivos permiten la correcta instalación del software de PC del Z80 Development System Kit. El archivo SETUP.EXE se utiliza, como se vio al principio de este manual, para la instalación del software, mientras que los otros archivos que comienzan con SETUP hacen de soporte a la instalación. Los archivos con extensión VB_ y DL_ sirven para que el software se comunique con otras aplicaciones. El archivo Z80-DSK.EX_ es el que, una vez descomprimido, permitirá la ejecución del software de PC del sistema.

NOTA: la ausencia de alguno de los archivos arriba especificados puede traer como consecuencia el mal funcionamiento del sistema.

6.2.2.2.Ejecución:

Para ejecutar el software de PC del Z80 Development System Kit deberá hacer doble click en el icono identificatorio del sistema.



Figura 1: Icono inicial

7. Operación del sistema

Para comenzar a operar el Z80 Development System Kit deberá cerciorarse que la placa de desarrollo se encuentre alimentada con 5 Voltios (5% de tolerancia) y que se encuentre conectada a la PC a través del RS232C. Una vez chequeados estos requerimientos podrá ejecutar el software de PC desde el sistema operativo Windows haciendo doble click en el icono del sistema.

Lo primero que le aparecerá en pantalla será una ventana en la cual se le requiere que informe al sistema cual es el puerto serie que tiene disponible (Figura 2). Este puerto debe ser el puerto al que se encuentra conectada la placa de desarrollo. Una vez que haya ingresado el numero del puerto que utilizara el sistema, haga click en OK y resetee la placa de desarrollo a través del switch de inicialización. Si todo esta en condiciones de operar aparecerá un mensaje indicándole que se recibió el reset desde la placa de desarrollo (Figura 3) y posteriormente se visualizará la ventana de administración del sistema (Figura 4).

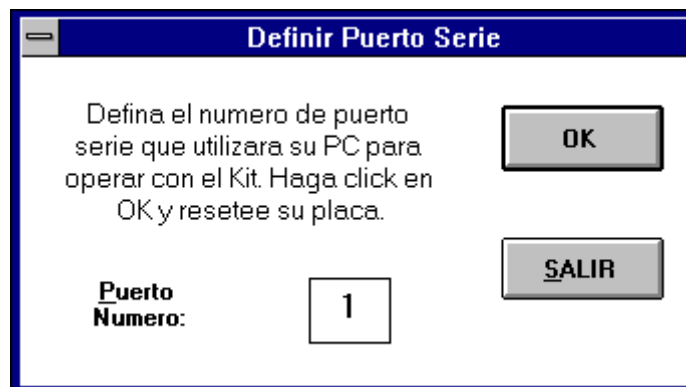


Figura 2: Definir Puerto Serie

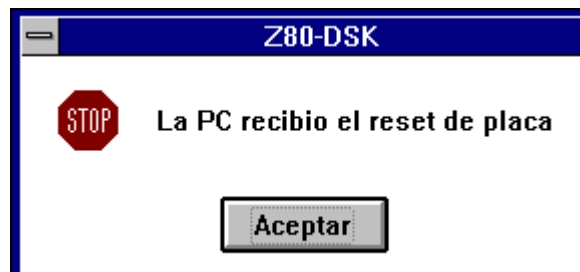


Figura 3: Mensaje del reset

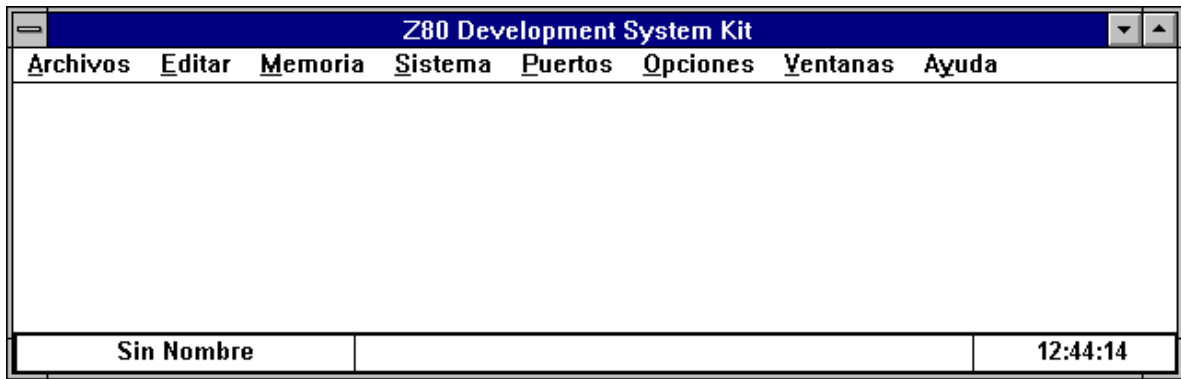


Figura 4: Ventana principal

NOTA: si define como puerto disponible para uso del sistema el mismo puerto que está utilizando el mouse, este dejará de funcionar. Si ello ocurriese y desea utilizarlo, deberá instalarlo nuevamente.

7.1. Descripción de la ventana de administración del sistema:

La ventana desde donde se administra el sistema tiene en su parte superior una barra de títulos en donde se puede leer el nombre del sistema, justo debajo de la barra de títulos se encuentra la barra de menús, la cual tiene los 8 menús que posee el sistema, a saber:

1. Archivos
2. Edición
3. Memoria
4. Sistema
5. Puertos
6. Opciones
7. Ventanas
8. Ayuda

Desde los 8 menús se accede a toda la potencialidad del sistema.

En la parte inferior a la izquierda se ubica un pequeño rectángulo en el cual se puede ver el nombre del archivo que se encuentra actualmente cargado. Si no hubiese ningún archivo cargado se visualizará la leyenda SIN NOMBRE. A la derecha del rectángulo antes mencionado, y al centro de la ventana, hay un rectángulo de tamaño superior en donde se puede ver una pequeña ayuda respecto de la función de cada objeto que se ve en pantalla y que se encuentre focalizado (focalizar es el procedimiento por el cual un objeto dentro de la pantalla se encuentra activo y susceptible para recibir eventos. La focalización se consigue pulsando reiteradamente la tecla TAB). En la parte inferior derecha de la ventana principal se muestra la hora que tenga el reloj de la PC y su valor se actualiza cada un segundo.

7.2. Descripción de la inicialización:

Cuando se ejecuta el software de PC del Z80 DSK, el sistema de administración se inicializa. La inicialización consiste en asignar valores por defecto a las distintas partes del sistema que pida valores al usuario, anular del nombre del archivo que se esté utilizando y resetear el contenido de los registros.

NOTA: la memoria RAM solamente es inicializada (se llena con 0) cuando se resetea la placa de desarrollo, como así también los integrados periféricos (SIO, PIO, CTC).

7.3. Descripción de los menús:

7.3.1. Menú Archivos

El menú Archivos es el que se encuentra en el extremo izquierdo de la barra de menús. Con este menú se despliega una lista de comandos para el trabajo con programas que se encuentren en archivos.

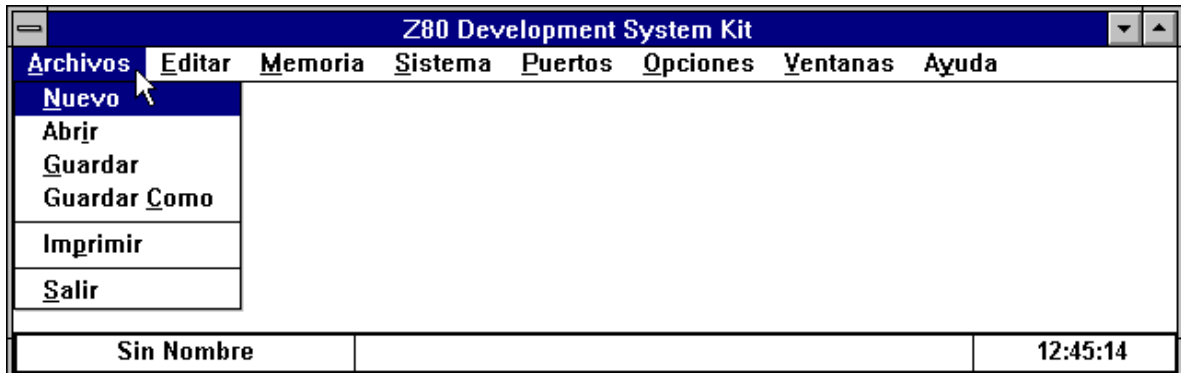


Figura 5: Menú Archivos

7.3.1.1. Submenú Nuevo:

Este comando permite la inicialización del software de PC como si se hubiera vuelto a ejecutar el mismo.

NOTA: la pantalla no vuelve al estado inicial como lo muestra la figura anterior, sino que conserva la disposición que tenga en ese momento.

7.3.1.2. Submenú Abrir:

Este comando permite al usuario el ingreso de archivos con formato hexadecimal (extensión .HEX) a la memoria del sistema.

Al hacer click en este submenú se abrirá una ventana de gestión de archivos (Figura 6), con la cual se seleccionará el archivo .HEX a ingresar. Si el archivo es compatible con el sistema, a continuación se mostrará la porción de memoria donde se ha colocado el programa recientemente ingresado.

Problemas con el archivo a abrir: pueden darse varias causas por las que el sistema muestre un mensaje indicando un problema con el archivo que se desea ingresar, ellas pueden ser:

- El archivo no tiene la extensión deseada.
- El archivo no corresponde a un archivo en formato Hexadecimal.
- El archivo que se desea ingresar es demasiado largo como para caber en la memoria física del sistema.

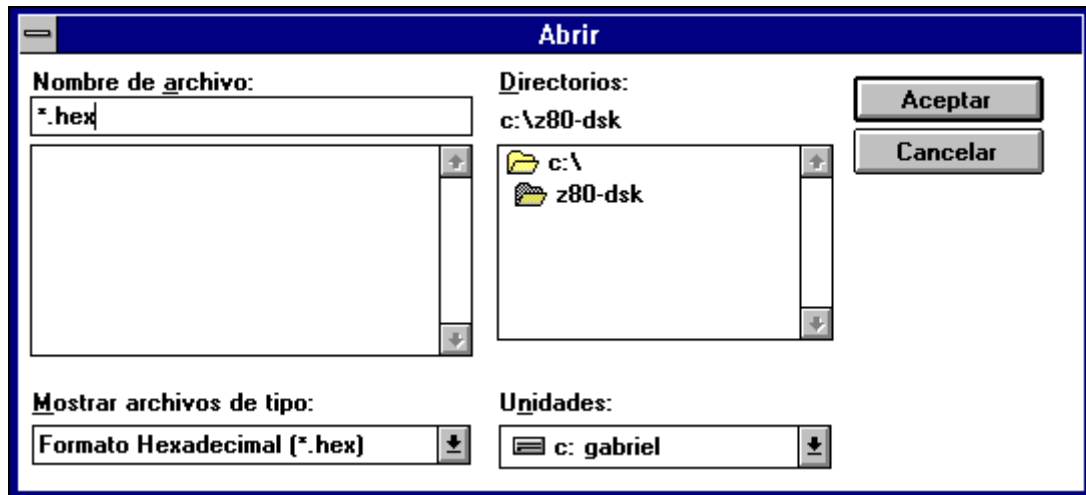


Figura 6: Abrir archivos

NOTA: debido a las restricciones físicas propias del sistema, los archivos que se deseen ingresar deberán estar organizados a partir de la posición de memoria 3000H.

7.3.1.3.Submenú Guardar:

Este comando permite al usuario guardar una porción de la memoria en un archivo dándole automáticamente el formato hexadecimal (extensión .HEX). La forma de trabajar de este comando depende del archivo que se haya abierto previamente. Si ningún archivo se encuentra abierto al momento de la ejecución de este comando, el sistema automáticamente invocará el comando Guardar Como... (ver punto siguiente); si por el contrario se encuentra un archivo en uso, el bloque de memoria a guardarse será igual al bloque de memoria del archivo que se ingresó previamente y el nombre del archivo no cambiará. El objetivo de este comando es dar una actualización rápida al último archivo ingresado sin alterar su longitud.

7.3.1.4.Submenú Guardar Como...:

Este comando permite al usuario guardar una porción de la memoria en un archivo dándole automáticamente el formato hexadecimal (extensión .HEX) al igual que el comando Guardar con la excepción de que le solicita al usuario que defina los límites de la porción de memoria que desea guardar y el nombre con el que desea archivar esa porción de la memoria.

Al hacer click sobre el comando aparecerá una ventana como la siguiente:

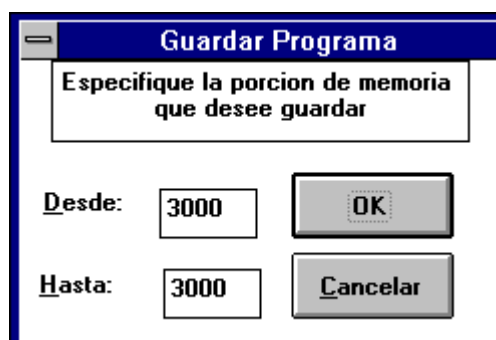


Figura 7: Límites del bloque a guardar

esta ventana le permite al usuario que defina los límites de la memoria que desea que se guarden en un archivo. Si se elige el botón de OK se pasara a una ventana de gestión de

archivos con la cual se puede dar el nombre del archivo con el que se guardara el contenido del bloque de memoria deseado.



Figura 8: Guardar archivo

Si se elige el botón CANCELAR de la ventana de gestión de archivos se anula por completo la operación de Guardar Como...

7.3.1.5. Submenú Imprimir:

Este comando permite al usuario la salida por impresora de:

- Contenido de la memoria
- Contenido de los puertos
- Valores de los Registros
- Contenido del portapapeles

Al hacer click en este comando se abrirá una ventana como la siguiente:

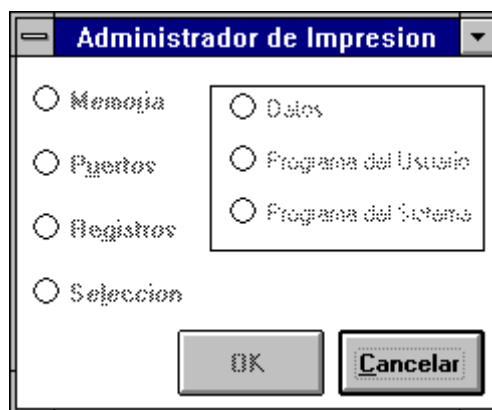


Figura 9: Opciones de impresión

Como se puede observar, las distintas opciones de impresión se encuentran desactivadas, por lo que el usuario no puede acceder a ninguna de ellas. Para poder hacerlo se debe tener en pantalla el contenido de alguna de las opciones a imprimir, o sea, se haya abierto previamente alguna ventana con la información deseada. Por ejemplo: si se quiere imprimir el contenido de los registros se deberá abrir previamente la ventana que muestra el contenido de los registros para luego acceder a la impresión.

La impresión se encuentra administrada por el sistema operativo Windows. El sistema solamente le indica al sistema operativo Windows lo que quiere imprimir para que luego se encargue el de realizar la impresión, por lo que si se quiere cambiar alguno de los parámetros de impresión (tamaño de papel, orientación, resolución, etc.) se deberá hacer desde la parte de impresión del Panel de Control del sistema operativo.

7.3.1.6.Submenú Salir:

Este comando permite al usuario finalizar la ejecución del software de PC del Z80 Development System Kit.

NOTA: este comando no brinda ninguna opción de salida, por lo que si se ha modificado un archivo y se desea guardar la modificación hecha se deberá ejecutar el comando Guardar o Guardar Como... previamente a la ejecución de este comando.

7.3.2. Menú Editar:

Dentro de este menú se encuentran los comandos más comunes para la edición como se puede ver en la figura.

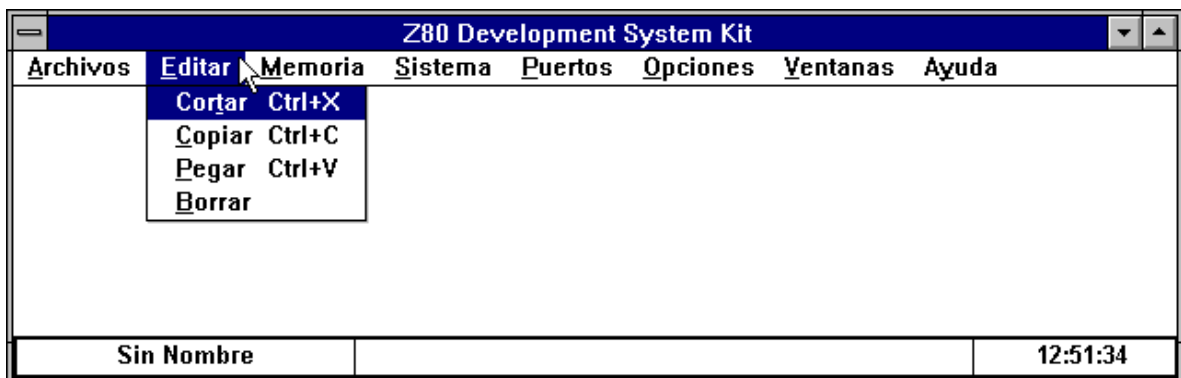


Figura 10: Menú Editar

Estos comandos cumplen la misma función que sus pares en las distintas aplicaciones para Windows. Se ha puesto énfasis en conservar las combinaciones de acceso rápido a los comandos para permitir una mejor adaptación al sistema por parte del usuario.

7.3.3. Menú Memoria:

Este menú contiene todos los comandos relacionados con el manejo de la memoria del sistema. Las operaciones relacionadas con la misma son las que muestra la figura:

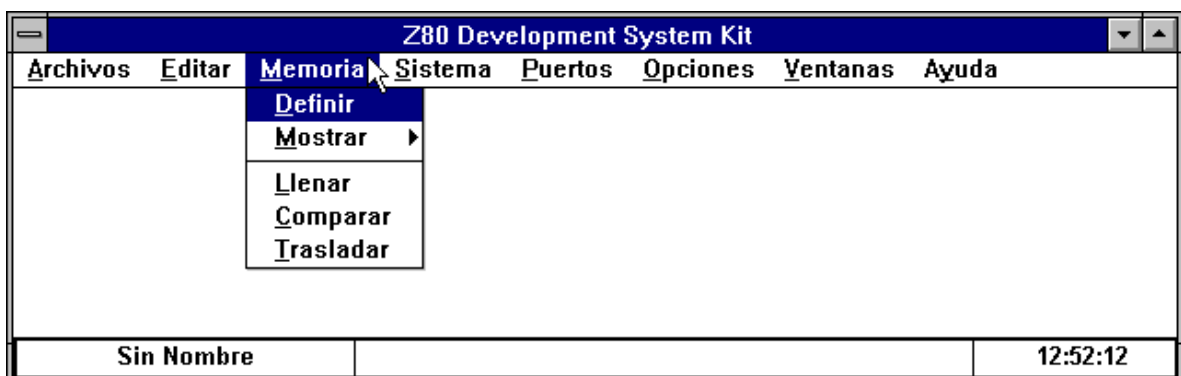


Figura 11: Menú Memoria

7.3.3.1.Submenú Definir...:

Este comando interactúa con el comando Mostrar y como su nombre lo indica permite que el usuario defina el comienzo de los distintos bloques de memoria que desea visualizar. Al hacer click sobre este comando se abre una ventana como la siguiente:

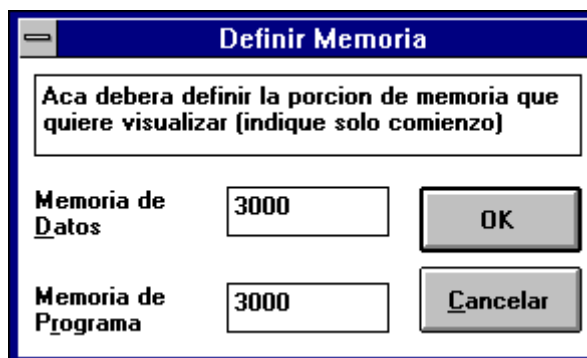


Figura 12: Definición del comienzo de los bloques de memoria

En esta ventana se puede definir los comienzos de los bloques de memoria de datos y de programa que el usuario desea visualizar a través del comando Mostrar. Si se elige la opción CANCELAR esta ventana se cerrará al igual que si se elige OK con la salvedad de que las modificaciones efectuadas no tendrán efecto.

7.3.3.2.Submenú Mostrar:

A través de este comando se puede visualizar el contenido de la memoria del sistema. El sistema da la opción de tener tres bloques de memoria simultáneamente activados (en pantalla).

Estos bloques se los diferencia de acuerdo a la función lógica que cumplen; así, se puede visualizar el contenido del software de placa del sistema (memoria del sistema), el contenido del software escrito por el usuario (memoria del usuario) y la porción de la memoria que alberga los datos que utiliza el usuario en sus programas como lo muestra la siguiente figura:

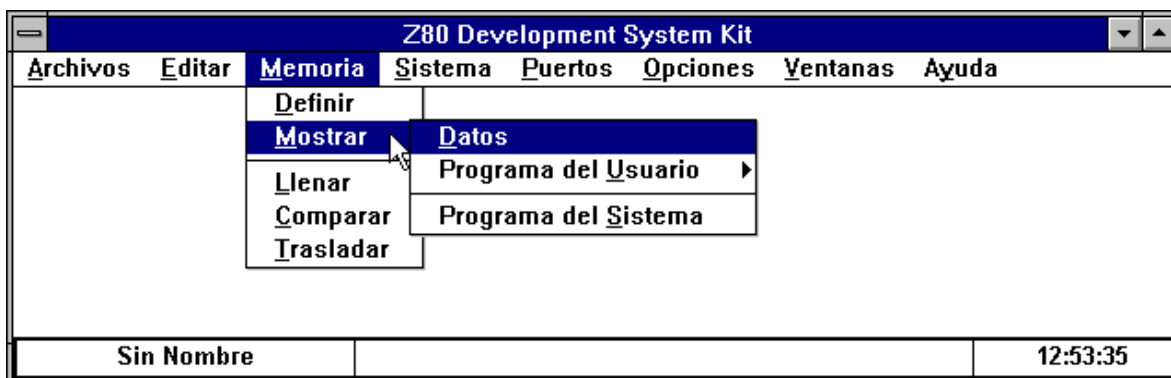


Figura 13: Opciones para ver la memoria

Cualquiera sea la opción que se elija, se encontrará en la ventana que se abre una barra de botones en la parte superior la cual tiene en su interior cuatro botones; estos botones se utilizan para INCREMENTAR la porción de memoria mostrada, DECREMENTAR la porción de memoria mostrada, ACTUALIZAR el contenido de la porción de memoria mostrada o cerrar la ventana (CANCELAR), y en la parte inferior el detalle del contenido de la porción de memoria que se muestra.

7.3.3.2.1. Comando Mostrar Datos:

A este comando se accede a través del menú Memoria y del submenú Mostrar. Este comando permite que el usuario visualice el contenido de una porción de la memoria RAM en forma hexadecimal con el correspondiente ASCII a su derecha si el código ASCII es representable, si no lo fuera se encontrara un punto en su lugar. Al hacer click sobre este comando aparecerá una ventana como la siguiente:

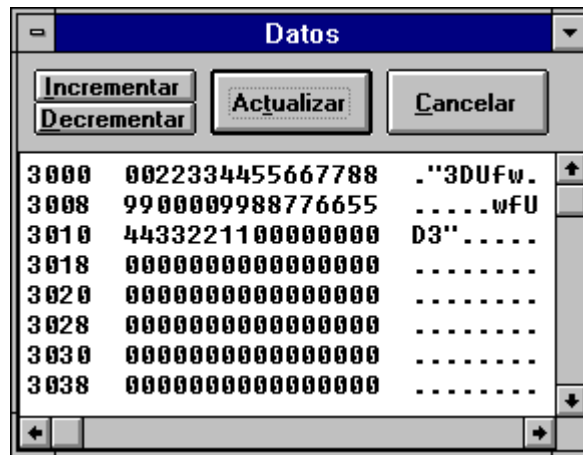


Figura 14: Datos de la memoria

En ella se puede ver que en la columna situada en la izquierda se encuentran representadas las posiciones de memoria, en la columna del centro se muestra el contenido de las posiciones de memoria en forma hexadecimal y en la columna de la derecha se da el código ASCII del contenido de la posición de memoria.

El rango de operación de este comando es: desde la posición de memoria 3000H hasta la posición de memoria FFFFH dependiendo de si se elige trabajar con mapa de memoria expandido. Si no se elige trabajar con mapa de memoria expandido el límite es la posición de memoria 7FFFH.

NOTA: al INCREMENTAR la porción de memoria mostrada se puede llegar a la máxima posición de memoria que el software de PC es capaz de mostrar (7FFFH); si ello ocurriese, al INCREMENTAR nuevamente la porción de memoria a mostrar, el software de PC mostrará a partir del menor valor de memoria que puede mostrar (3000H). Lo mismo es aplicable al comando DECREMENTAR con la salvedad de que operará en forma invertida respecto de INCREMENTAR.

7.3.3.2.2. Comando Mostrar Programa del Usuario:

Este comando se utiliza para visualizar el contenido de la memoria en donde el usuario aloja sus programas. A través de este comando el usuario puede ver el contenido de la memoria RAM que tiene disponible para su uso (3000H - FFFFH) de dos formas diferentes según sea su necesidad:

- Con el contenido de la memoria representado en forma hexadecimal.
- Con el contenido de la memoria representado a través de mnemónicos.

Al hacer click en el comando Programa del Usuario se abre un nuevo submenú con las opciones antes descritas.

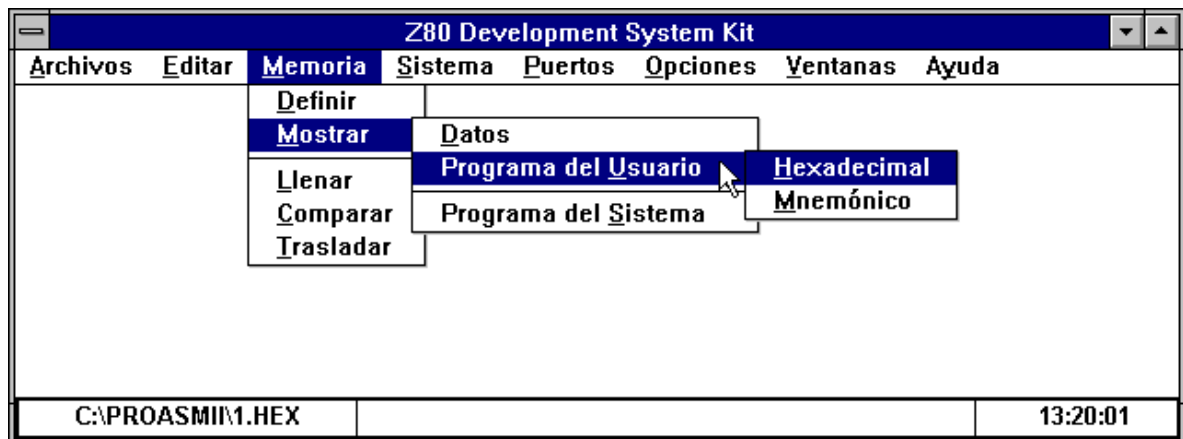


Figura 15: Opciones para ver el programa del usuario

Si se elige Mostrar en forma Hexadecimal, la ventana que se abre tiene características similares a la descrita por el comando Mostrar Datos. Si en cambio se elige Mostrar a través de los Mnemónicos, se encontrará que en la columna de la izquierda están representadas las posiciones de memoria, en la columna central el código de la instrucción y en la columna de la derecha el mnemónico de la instrucción tal como se puede observar en la siguiente figura:

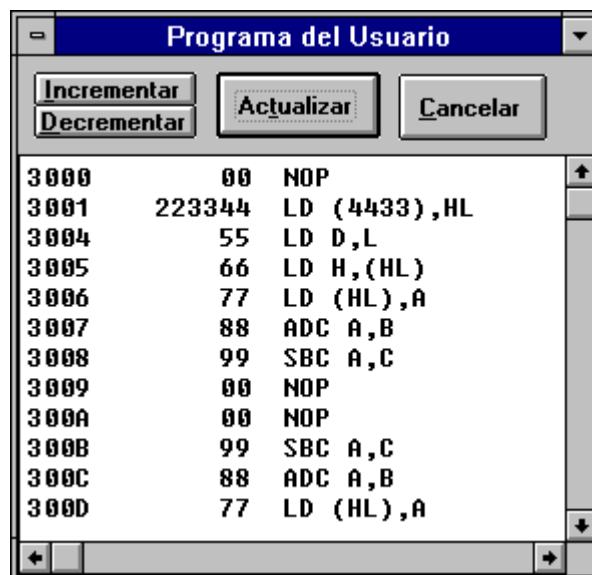


Figura 16: Programa de memoria en mnemónico

El rango de operación de este comando es: desde la posición de memoria 3000H hasta la posición de memoria FFFFH dependiendo de si se elige trabajar con mapa de memoria expandido. Si no se elige trabajar con mapa de memoria expandido el límite es la posición de memoria 7FFFH.

Las características de los botones INCREMENTAR y DECREMENTAR son iguales a las del comando Mostrar Datos.

7.3.3.2.3. Comando Mostrar Programa del Sistema:

Este comando se utiliza para visualizar el software de placa del sistema. El rango de operación de este comando va desde la posición de memoria 0H hasta la posición de memoria 2FFFH y la representación que se hace de la memoria del sistema es a través de los mnemónicos de las instrucciones.



Figura 17: Programa del sistema

7.3.3.3.Submenú Llenar

Este comando se utiliza para llenar un bloque de memoria con un contenido determinado. Al hacer click en este submenú se accede a una ventana como la siguiente:



Figura 18: Llenar memoria

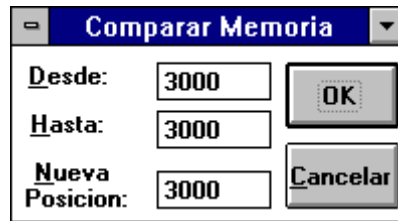
En esta ventana se puede observar que se puede definir los límites inferior y superior del bloque de memoria que se quiere llenar, y que se puede establecer el contenido que se desea que tenga dicho bloque de memoria. El valor que se observe como futuro contenido del bloque de memoria a llenar será el que tenga el portapapeles en ese momento (cuatro primeros caracteres) si dentro del portapapeles se encuentra algo, y si dentro del portapapeles no hubiera nada, se visualizaría el ultimo valor asignado a este parámetro.

El rango de operación de este comando es: desde la posición de memoria 3000H hasta la posición de memoria FFFFH dependiendo de si se elige trabajar con mapa de memoria expandido. Si no se elige trabajar con mapa de memoria expandido el límite es la posición de memoria 7FFFH.

NOTA: como contenido de un bloque de memoria se puede asignar desde un nibble hasta dos bytes.

7.3.3.4.Submenú Comparar

Este comando se utiliza para comparar los contenidos de dos bloques de memoria. Para que dos bloques sean comparables, es necesario que sean de igual tamaño. Al hacer click en este comando se accede a una ventana como la siguiente:



Comparar Memoria

Desde: 3000 OK

Hasta: 3000

Nueva Posicion: 3000 Cancelar

Figura 19: Bloques a comparar

En esta ventana se puede observar que se puede definir los límites inferior y superior del bloque n°1, y el límite inferior del bloque n°2. El sistema se encargará de determinar si los bloques tienen igual longitud para llevar a cabo la comparación, y si no lo fueran se podrá observar un mensaje como el siguiente:



Figura 20: Mensaje por imposibilidad de comparación

para llevar a cabo la comparación se utiliza como límite máximo superior para el bloque n°2 la posición de memoria final de la memoria RAM, cuyo valor no deber ser superado por la extensión del bloque n°2, la cual depende de la extensión del bloque n°1.

Si la comparación es posible, se observará un mensaje de:



Figura 21: Mensaje de bloques iguales

o de



Figura 22: Mensaje de bloques desiguales

de acuerdo al resultado de la comparación.

El rango de operación de este comando es: desde la posición de memoria 3000H hasta la posición de memoria FFFFH dependiendo de si se elige trabajar con mapa de memoria expandido. Si no se elige trabajar con mapa de memoria expandido el límite es la posición de memoria 7FFFH.

7.3.3.5.Submenú Trasladar:

Este comando se utiliza para trasladar el contenido de un bloque de memoria hacia otra porción del mapa de memoria. Al hacer click en este comando se accede a una ventana como la siguiente:

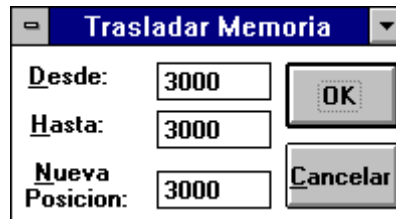


Figura 23: Trasladar bloque de memoria

En esta ventana se puede observar que se puede definir los límites inferior y superior del bloque n°1, y el límite inferior del bloque n°2. En este caso no es necesario que los dos bloques sean iguales, ya que si el límite superior del bloque n°2 (depende del tamaño del bloque n°1) es superior a la posición de memoria máxima de la memoria RAM (FFFFH), se trasladaran los datos del bloque n°1 hasta que se alcance la ultima posición de la memoria RAM.

El rango de operación de este comando es: desde la posición de memoria 3000H hasta la posición de memoria FFFFH dependiendo de si se elige trabajar con mapa de memoria expandido. Si no se elige trabajar con mapa de memoria expandido el límite es la posición de memoria 7FFFH.

7.3.4. Menú Sistema

Dentro de este menú se encuentran todos los comandos relacionados con el estado del microprocesador. La lista de comandos se puede ver en la siguiente figura:

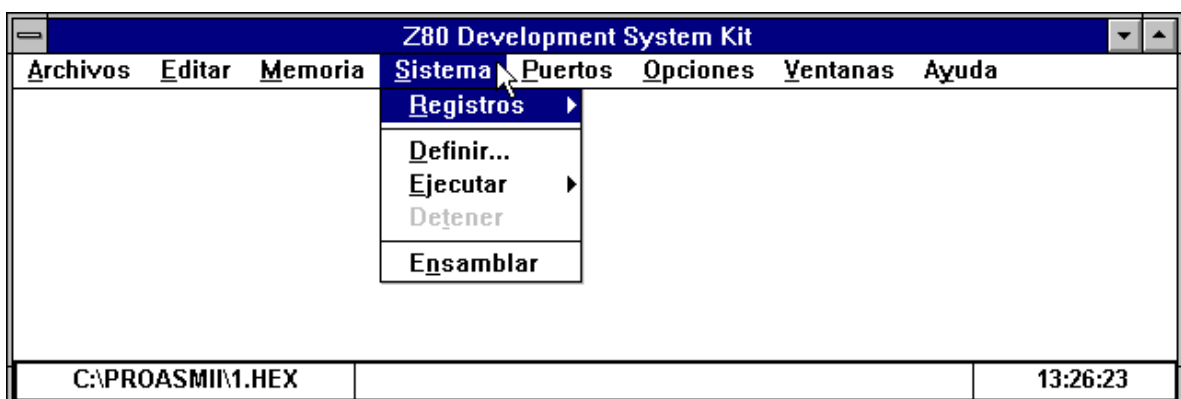


Figura 24: Menú Sistema

7.3.4.1.Submenú Registros

A través de este comando se puede acceder a los registros del microprocesador Z80 que se encuentra en la placa de desarrollo. Si se hace click se abrirá un nuevo submenú como lo muestra la figura:

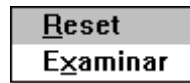


Figura 25: Submenú Registros

7.3.4.1.1.Comando Registros Reset

A través de este comando se puede inicializar el contenido de los registros. Los valores que tendrán los registros luego de la inicialización es el siguiente:

 A screenshot of a software window titled "Registros". At the top, there are three buttons: "OK", "Act.", and "Cancelar". Below the buttons, the registers are displayed in a grid:

A		I		R
00		00	00	
B	C	IX	0000	
00	00	IY	0000	
D	E	SP	3500	
00	00	PC	3000	
H	L	Flags		
00	00	s	z	x
		h	x	p/v
		n	c	
0	0	0	0	0
0	0	0	0	0

Figura 26: Contenido de los registros

NOTA: No es necesario que la ventana donde figuran los registros se encuentre abierta para que tenga efecto este comando.

7.3.4.1.2.Comando Registros Examinar

A través de este comando se puede visualizar y programar el contenido que se desea que tengan los registros del microprocesador Z80. Al hacer click sobre este comando se abrirá una ventana como la anterior.

En esta ventana se puede observar que aparecen los nombres y los contenidos de los registros. En la parte superior se encuentra un caja de botones con tres botones en su interior. El botón de OK se utiliza para programar el contenido de los registros del microprocesador con los valores que se encuentren en la ventana. El botón de ACTUALIZAR se utiliza para actualizar el contenido de los registros de la ventana con el contenido de los registros del microprocesador y el botón CANCELAR se utiliza para cerrar la ventana.

7.3.4.2.Submenú Definir

Este comando se utiliza juntamente con los comandos de ejecución. A través de este comando se accede a una ventana como la siguiente:

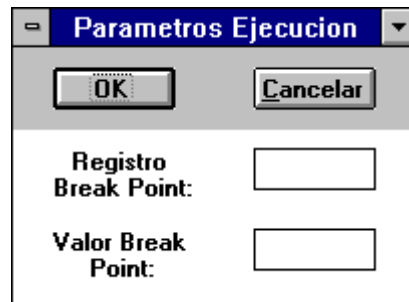


Figura 27: Parámetros para el Break Point

con la cual se puede definir los parámetros en el caso de que se desee ejecutar el software del usuario a través del uso de Break Points (ver Comando Ejecutar con Break Point). La ventana permite definir el registro que oficiará de Break Point y el valor que se debe alcanzar para detener el flujo del programa.

7.3.4.3.Submenú Ejecutar

Al hacer click en este comando se abre un nuevo submenú como lo muestra la figura:



Figura 28: Submenú Ejecutar

Este comando permite elegir la forma en la que el usuario desea que se ejecute su software dentro de la placa de desarrollo, con control del software de PC o sin su control.

7.3.4.3.1.Comando Ejecutar en Tiempo Real

Este comando le permite al usuario ejecutar el software que haya escrito para el sistema sin la intervención del software de PC, por lo que es ideal para probar programas que se encuentren en un alto grado de desarrollo. Al ejecutar este comando el sistema permitirá la ejecución del software del usuario a partir de la posición de memoria que determine el registro PC, por lo que se deberá constatar su valor previamente a la ejecución para evitar efectos indeseados.

NOTA: si se desea que el software de PC vuelva a tener el control de la placa de desarrollo se deberá colocar al final del programa una instrucción de salto incondicional a la posición de memoria 03F6H.

7.3.4.3.2.Comando Ejecutar en Tiempo Virtual

Este comando le permite al usuario ejecutar el software que haya escrito para el sistema con la intervención del software de PC. Con el uso de este comando se puede ir siguiendo la ejecución del programa a través de los cambios que se producen en los contenidos de los registros y de los puertos. Al ejecutar este comando el sistema comenzara una ejecución Paso A Paso de las instrucciones, a partir de la posición de memoria indicada por el contenido del registro PC, en forma acelerada. Para detener el flujo de la ejecución se deberá hacer click en el comando DETENER que se habilita una vez que se comienza con este tipo de ejecución. Para una mejor comprensión de los alcances de este comando por favor lea el Comando Ejecutar Paso A Paso.

NOTA: este comando no abre las ventanas que contienen el contenido de los Registros y el de los Puertos.

7.3.4.3.3. Comando Ejecutar Paso A Paso

Este comando le permite al usuario ejecutar el software que haya escrito para el sistema a razón de una instrucción por vez. Cada vez que se haga click en este comando (o a través de su tecla de acceso rápido F5) se ejecutará la instrucción que se encuentre a partir de la posición de memoria que indique el contenido del registro PC.

IMPORTANTE: para más detalles sobre el funcionamiento y alcance de este comando le sugerimos que se remita al Manual de Diseño del sistema.

7.3.4.3.4. Comando Ejecutar con Break Point

Este comando le permite al usuario ejecutar el software que haya escrito para el sistema hasta que se cumpla la condición de Break Point (detención del flujo del programa). Al ejecutar este comando el sistema comenzará una ejecución Paso A Paso de las instrucciones a partir de la posición de memoria indicada por el contenido del registro PC hasta que se llegue a la condición de Break Point definida por el usuario o bien cuando se ejecute el comando DETENER.

NOTA: Este comando se encuentra normalmente deshabilitado mientras no se introduzcan los parámetros que oficiaran de Break Point para el flujo de la ejecución (ver Submenú Definir).

7.3.4.4. Comando Detener

Este comando permite al usuario detener el flujo del programa que se este ejecutando, ya sea en Tiempo Virtual como con el uso de Break Point. Normalmente este comando se encuentra deshabilitado hasta que se comienza con la ejecución del programa del usuario.

7.3.4.5. Submenú Ensamblar

A través de este comando se accede a una ventana como la siguiente:

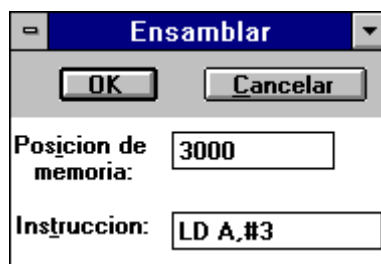


Figura 29: Instrucción a ensamblar

Este comando posibilita el ensamblado directo de instrucciones en la memoria de trabajo asignada para uso del usuario (3000H - FFFFH). Como se puede ver en la figura anterior se puede definir la posición de memoria a partir de la cual se quiere ensamblar la instrucción deseada, la cual se ajustará automáticamente, luego del ensamblado (usando el botón OK), a la posición de memoria siguiente a la última ocupada por la instrucción ensamblada.

El rango de operación de este comando es: desde la posición de memoria 3000H hasta la posición de memoria FFFFH dependiendo de si se elige trabajar con mapa de memoria expandido. Si no se elige trabajar con mapa de memoria expandido el límite es la posición de memoria 7FFFH.

7.3.5. Menú Puertos

Dentro de este menú se encuentran los comandos necesarios para la gestión de entrada y salida directa a los puertos del sistema.



Figura 30: Menú Puertos

7.3.5.1. Submenú Ver Contenido

Este comando permite la visualización del contenido de los puertos del sistema. Al hacer click en este comando se abre una ventana como la siguiente:

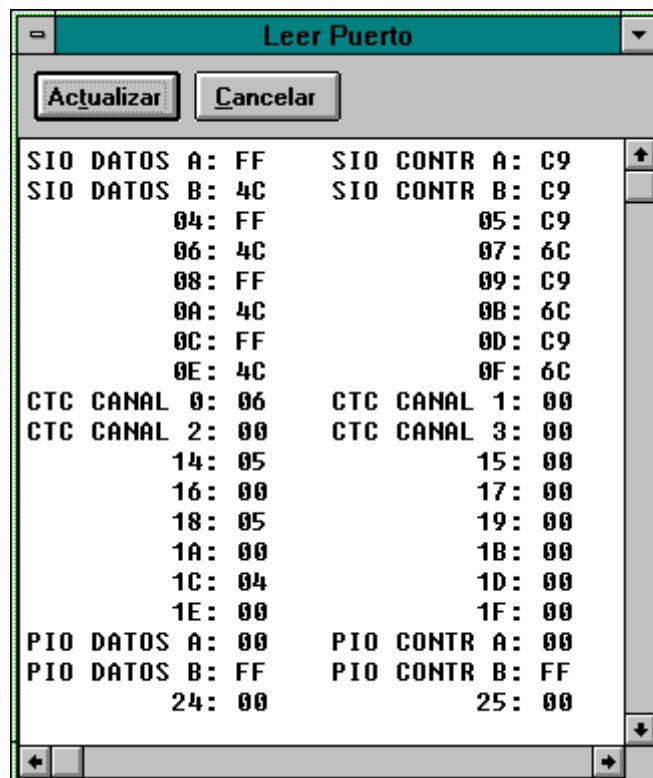


Figura 31: Contenido de los puertos

En ella se puede observar los contenidos de los puertos del sistema formando dos columnas en donde se indica el número del puerto con el contenido correspondiente a su derecha. Para actualizar los contenidos mostrados en pantalla basta hacer click en el botón ACTUALIZAR que se encuentra en la parte superior izquierda de la ventana.

El rango de operación de este comando depende de si se elige o no trabajar con mapa de puertos expandido. Si no se elige trabajar con mapa de puertos expandido el número de puerto máximo es 2FH, caso contrario el número de puerto máximo es FFH.

7.3.5.2. Submenú Escribir

Este comando permite la programación individual de los puertos. Al hacer click en este comando se abre una ventana como la siguiente:



Figura 32: Programar puerto

En ella se puede definir el número de puerto que se desee programar (escribir) y el contenido que se quiere que tenga.

El rango de operación de este comando depende de si se elige o no trabajar con mapa de puertos expandido. Si no se elige trabajar con mapa de puertos expandido el número de puerto máximo es 2FH, caso contrario el número de puerto máximo es FFH.

NOTA: el software de PC no permitirá la reprogramación de los puertos que utiliza el Z80 DSK para su funcionamiento.

7.3.6. Menú Opciones

Este Menú permite la selección de la extensión del mapa de memoria y de puertos con que se quiere trabajar como se puede ver en la siguiente figura.

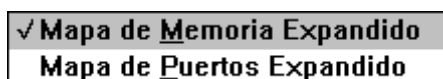


Figura 33: Submenú Opciones

En el caso de elegir Mapa de Memoria Expandido, el sistema operará con toda la capacidad de memoria que puede manejar (0 - FFFFH), caso contrario el sistema operará hasta la posición de memoria 7FFFH.

En el caso de elegir Mapa de Puertos Expandido, el sistema operará con toda la capacidad de puertos que puede manejar (0 - FFH), caso contrario el sistema operará hasta el puerto 2FH.

NOTA: recordar que el usuario tiene a su disposición el bloque de memoria que comienza a partir de la posición 3000H

7.3.7. Menú Ventanas

Dentro de este menú se encuentran todos los comandos relacionados con la organización de las ventanas que se encuentran abiertas (en pantalla).

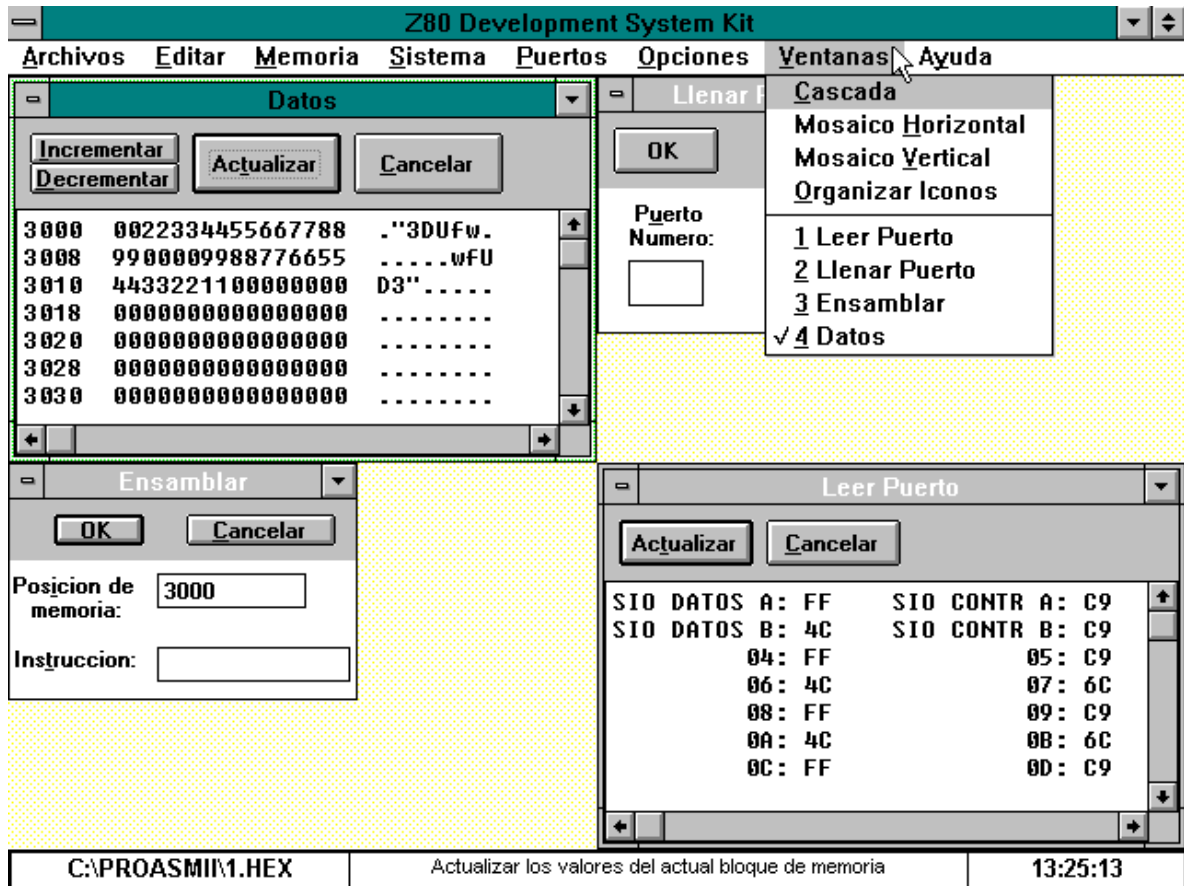


Figura 34: Menú Ventanas

Estos comandos funcionan de forma análoga a sus pares en otras aplicaciones Windows.

También se puede ver en este menú el listado de las ventanas que se encuentran abiertas, donde la que se encuentra marcada por un signo de verificación corresponde a la ventana activa.

7.3.8. Menú Ayuda

Este menú despliega el siguiente submenú:

Acerca del Z80-DSK...

Figura 35: Submenú Ayuda

que al hacer click sobre el se puede ver una ventana donde se da una breve información del sistema como se puede observar en la figura siguiente:

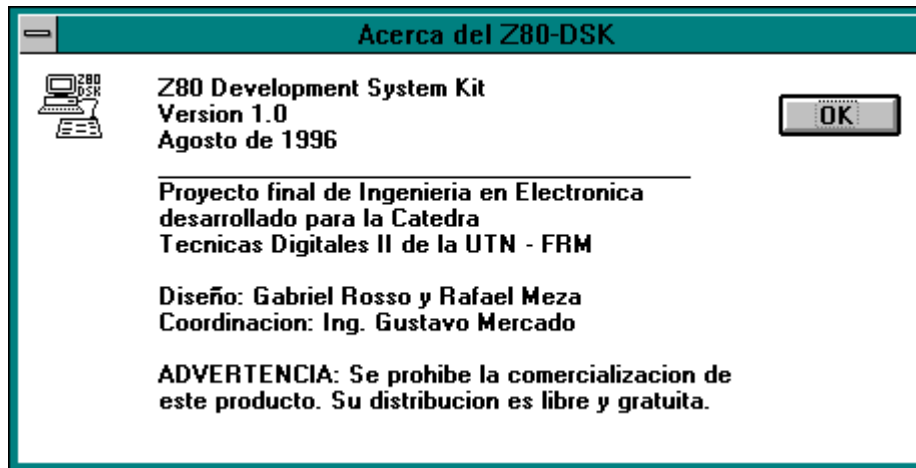


Figura 36: Información del sistema

8. Ejemplos de aplicaciones

Durante el ciclo lectivo 1995 se implementaron dos ejemplos de aplicaciones utilizando el Z80 Development System Kit. Dichos ejemplos se presentaron ante los alumnos de 3º año de la Cátedra Técnicas Digitales II de la carrera de Ingeniería en Electrónica. Estos ejemplos pueden servir de base para la comprensión de la forma de realizar futuras aplicaciones utilizando esta herramienta de trabajo.

NOTA: estos ejemplos fueron exhibidos utilizando la versión prototipo del Z80 DSK, cuya descripción funcional se hace en el Apéndice A; no obstante, son compatibles con la versión final del Z80 DSK bajo ciertas condiciones (para más detalles sobre este tema remítase al Apéndice A).

Las aplicaciones presentadas consistieron en la exposición de una aplicación para el PIO y otra para el CTC.

8.1. Aplicación con el PIO

La aplicación con el PIO consiste en la toma de datos a través de una llave BCD (implementada en el puerto A) y la excitación de displays de 7 segmentos (implementados en el puerto B) los cuales deben reflejar el dato obtenido de la llave BCD. El programa que maneja esta aplicación debe tomar los datos de la llave, convertirlos a 7 segmentos y excitar los displays 7 segmentos. Se utiliza visualización dinámica para la presentación en los displays, utilizando para ello el pin PB7 del PIO.

A continuación se muestran el diagrama de flujo y el programa que realiza esta aplicación.

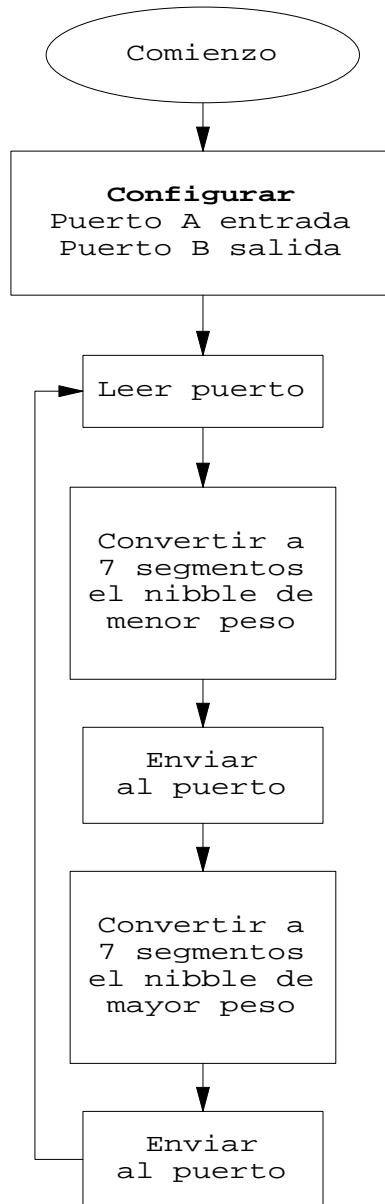


Diagrama 1: Aplicación para el PIO

```

TITLE PROGRAMA PARA LA PRACTICA DEL PIO
CSEG
RADIX 16
ENTRY COMIENZO
  
```

```

ORG 3000
  
```

;este programa ingresa los datos colocados en un par de llaves
;BCD (puerto A) y decodifica lo ingresado para excitar un par
;de displays de 7 segmentos (puerto B).

```

COMIENZO LD A, 4F      ;configura el puerto A como entrada
          OUT (22), A
          LD A, 0F     ;configura el puerto B como salida
          OUT (23), A
  
```

```

JUMP_5    IN A, (20)    ;lee el puerto A
           LD C, A      ;almacena temporalmente el dato ingresado
           LD D, 00     ;D=0 nibble 0, D=1 nibble 1
JUMP_4    LD B, 09     ;numero máximo de comparaciones
           LD HL, DIR   ;final de la tabla de conversión
           AND 0F       ;eliminar nibble alto
JUMP_2    CP B         ;compara el nibble elegido con los valores
           ;posibles
           JP Z, JUMP_1 ;si coinciden ir a JUMP_1
           DEC B        ;cambiar el valor con el que comparamos
           DEC HL       ;decrementar el puntero de la tabla de conversión
           JP JUMP_2    ;repetir hasta que coincida
JUMP_1    LD A, D       ;cargar el numero de nibble que se esta decodificando
           ADD A, 00
           JP NZ, JUMP_3 ;si es el nibble 0 continuar
           LD A, (HL)   ;cargar el valor decodificado
           ADD A, 80    ;setear el bit 7 para permitir una visualización dinámica
           ;(activar el display de menor peso)
           OUT (21), A  ;enviar al puerto B
           LD D, 01     ;cambio de nibble
           LD A, C       ;recuperar el dato ingresado por el puerto A
           RLC A        ;rotar a la izquierda cuatro veces
           RLC A
           RLC A
           RLC A
           JP JUMP_4    ;convertir el nibble 1
JUMP_3    LD A, (HL)   ;cargar el valor decodificado
           OUT (21), A  ;enviar por el puerto B
           JP JUMP_5    ;repetir todo el proceso indefinidamente

           DB 3F        ;0
           DB 06        ;1
           DB 5B        ;2
           DB 4F        ;3
           DB 66        ;4
           DB 6D        ;5
           DB 7D        ;6
           DB 07        ;7
           DB 7F        ;8
DIR        DB 67        ;9

```

END

El hardware necesario para esta aplicación se puede ver en el Apéndice B

8.2. Aplicación con el CTC

La aplicación con el CTC consiste en modificar el periodo de visualización de una cadena de leds conectados en el puerto B del PIO. El programa que maneja esta aplicación debe enviar en forma secuencial excitación a los distintos leds del puerto B, cuya velocidad depende de la frecuencia modificable de un astable.

A continuación se muestran el diagrama de flujo y el programa que realiza esta aplicación.

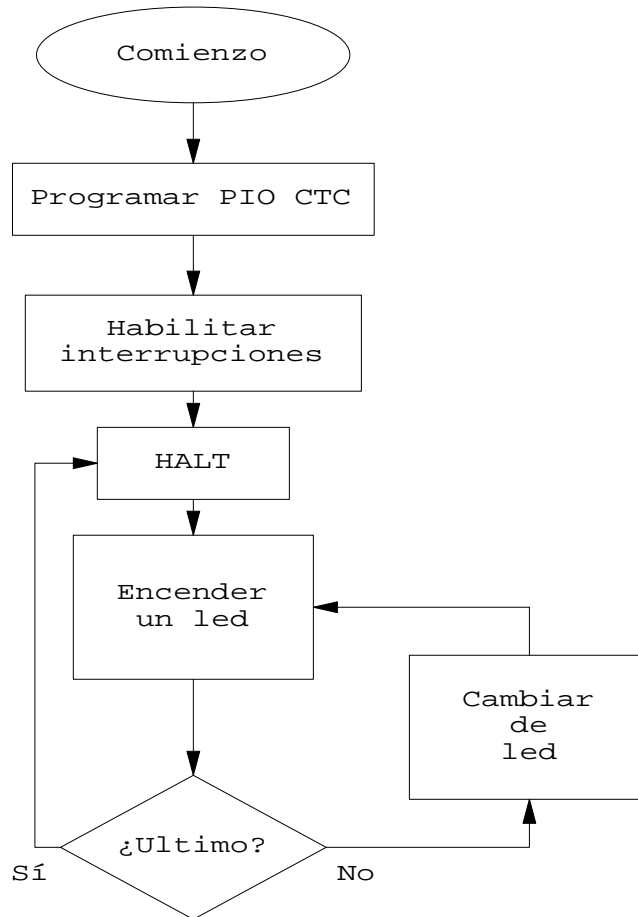


Diagrama 2: Aplicación para el CTC

TITLE SECUENCIADOR DE LEDS IMPLEMENTADO CON PIO Y CTC
 CSEG
 RADIX 16
 ENTRY COMIENZO

;este programa permite una visualización barrida de LEDs, cuya
 ;velocidad de barrido se puede controlar en forma externa a través
 ;de un potenciómetro conectado a un 555 funcionando como astable.
 ;cuando la constante de tiempo del CTC llega a cero se produce una
 ;interrupción, la cual permite que se vaya barriendo la cadena
 ;de LEDs

ORG 3000

```

COMIENZO  DI          ;deshabilitar las interrupciones
          LD A, 31    ;inicializar el registro índice
          LD I, A
          LD A, 0F
          OUT (23), A ;puerto B del PIO como salida
          LD A, 0F5
          OUT (11), A ;canal 1 del CTC
          ;habilitar las interrupciones
          ;sigue constante de tiempo
  
```



```

                                ;modo contador
                                LD A, 64
                                OUT (11), A ;constante de tiempo
                                LD A, 02
                                OUT (11), A ;valor del vector de interrupción
                                LD B, 08 ;cantidad de leds que se activan
                                LD HL, DIR ;principio de la tabla
ESPERA EI ;habilitar las interrupciones
        HALT ;esperar una interrupción
        JP ESPERA ;volver

RUTINA_PROGRAMA LD A, (HL) ;enviar el valor de la tabla
                OUT (21), A ;al puerto B del PIO para excitar los leds
                DJNZ JUMP_1 ;activar todos los leds
                LD B, 08 ;cantidad de leds que se activan
                LD HL, DIR ;principio de la tabla
                RETI ;retornar de la interrupción
JUMP_1 DEC HL ;cambiar de posición en la tabla
        RETI ;retornar de la interrupción

                                DB 01 ;bit 0
                                DB 02 ;bit 1
                                DB 04 ;bit 2
                                DB 08 ;bit 3
                                DB 10 ;bit 4
                                DB 20 ;bit 5
                                DB 40 ;bit 6
DIR      DB 80 ;bit 7

                                ORG 3102

DW      RUTINA_PROGRAMA ;etiqueta que tiene la dirección a la que irá el programa
                                ;cuando ocurra una interrupción

                                END

```

El hardware necesario para esta aplicación se puede ver en el Apéndice B

9. Ventajas de trabajar en entorno Windows

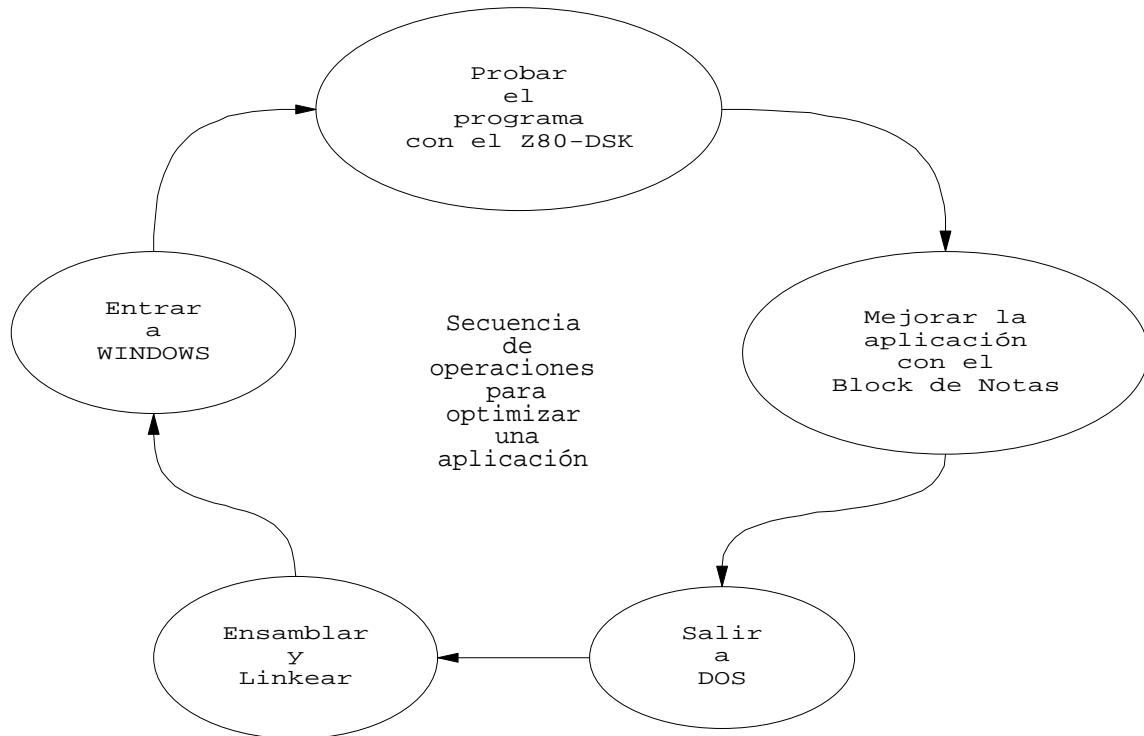
Cuando se desarrolla una aplicación particular muy rara vez “sale de una”, normalmente se pone a punto a través del método de prueba y error. Esto trae consigo que sea tedioso el tener que modificar un programa para que cumpla con las expectativas pretendidas cuando no se dispone de un entorno Windows, ya que se debe, la mayoría de las veces, salir de una aplicación, entrar en otra, hacer las modificaciones correspondientes, salir..., entrar... y así sucesivamente hasta que se consiga la modificación final del archivo que realmente interesa, es decir del archivo en formato hexadecimal. El Z80 DSK, al ser un sistema que se utiliza bajo entorno Windows, permite que la puesta a punto de un diseño sea más “agradable”, ya que los pasos que se siguen son los siguientes:

1. Ejecutar el utilitario Block de Notas, abrir el archivo de texto donde se encuentra el módulo del programa que se desea modificar, modificarlo y almacenarlo nuevamente.
2. Salir a DOS, ensamblar el módulo (se puede crear un archivo .BAT donde se consignen las órdenes para en ensamblado de los distintos módulos) y linkearlo.
3. Entrar en Windows nuevamente y probar el programa modificado a través del Z80 DSK.

4. Si el programa sigue teniendo problemas repetir los pasos anteriores hasta que se solucionen.

Nota: en Windows se puede tener varios programas “abiertos”, o sea ejecutándose, por lo que solamente se debe abrir una vez cada aplicación necesaria (excepto la salida a DOS) y se puede navegar entre ellas con el uso de la combinación ALT+TAB.

Lo antes expuesto se puede ver en el siguiente gráfico:



10. Bibliografía

- Z80 CPU Central Processing Unit Technical Manual de Zilog.
- Z80 PIO Technical Manual.
- Z80 SIO Technical Manual.
- Z80 CTC Counter/Timer Circuit Technical Manual.
- MOS Memory Data Book. Texas Instruments.
- MAXIM 1994 New Releases Data Book Volume III.
- ECG Semiconductors.
- ECG TTL Data Book.

11. Apéndice A: Prototipo del Z80 Development System Kit

Existe una versión DOS del Z80 DSK. Esta es una versión prototipo que se desarrolló antes de obtener la versión con la que se cuenta actualmente. Si se desea utilizar esta versión se debe cambiar únicamente el programa que se encuentra en la EPROM de la placa de desarrollo por el que se encuentra dentro del archivo Z80PROT.HEX. Esta versión utiliza la misma placa de desarrollo que la versión final y funciona con cualquier programa de PC que haga de la PC una terminal "boba". Obviamente esta versión tiene un set de instrucciones muy reducido comparado con la versión final, pero igualmente útil.

Conexión

La conexión es igual a la utilizada a la versión final.

Ejecución

A continuación se detallará la forma de proceder para comenzar con la ejecución de esta versión del Z80 DSK.

- Encienda la PC.
- Alimente la placa de desarrollo con la tensión recomendada.
- Resetee la placa de desarrollo.
- Ejecute el software que emule en la PC una terminal "boba".
- Pulse una tecla cualquiera.
- Verifique que aparezca un cartel indicándole que la placa se encuentra en funcionamiento.

Si se cumplen los pasos anteriores, Ud. se encontrará listo para operar la versión prototipo del Z80 DSK.

Comandos

El sistema cuenta con 8 comandos, cuyo detalle se da a continuación:

<u>Instrucción</u>	<u>Operación que realiza</u>	<u>Parámetros que utiliza</u>	<u>Sintaxis</u>
Ejecutar	Transfiere el valor del parámetro al contador de programa	Posición de memoria de comienzo de ejecución. (0-4FFF)	Exxxx
Mostrar	Muestra el contenido de 100 posiciones de memoria a partir del parámetro	Posición inicial de muestreo (0-4FFF)	Mxxxx
Llenar	Llena la memoria desde un valor inicial hasta un valor final con un valor deseado	Valor Inicial Valor Final Contenido (VI<VF) (0-4FFF)	Lxxxx,yyyy,zz donde xxxx=VI yyyy=VF zz=Contenido
Verificar	Comprueba si dos bloques de memoria son iguales	Valor Inicial Valor Final Nueva Posición (VI<VF) (0-4FFF)	Vxxxx,yyyy,zzzz donde xxxx=VI yyyy=VF zzzz=NP
Trasladar	Mueve un bloque de memoria a otra posición	Valor Inicial Valor Final Nueva Posición (VI<VF)	Txxxx,yyyy,zzzz donde xxxx=VI yyyy=VF zzzz=NP

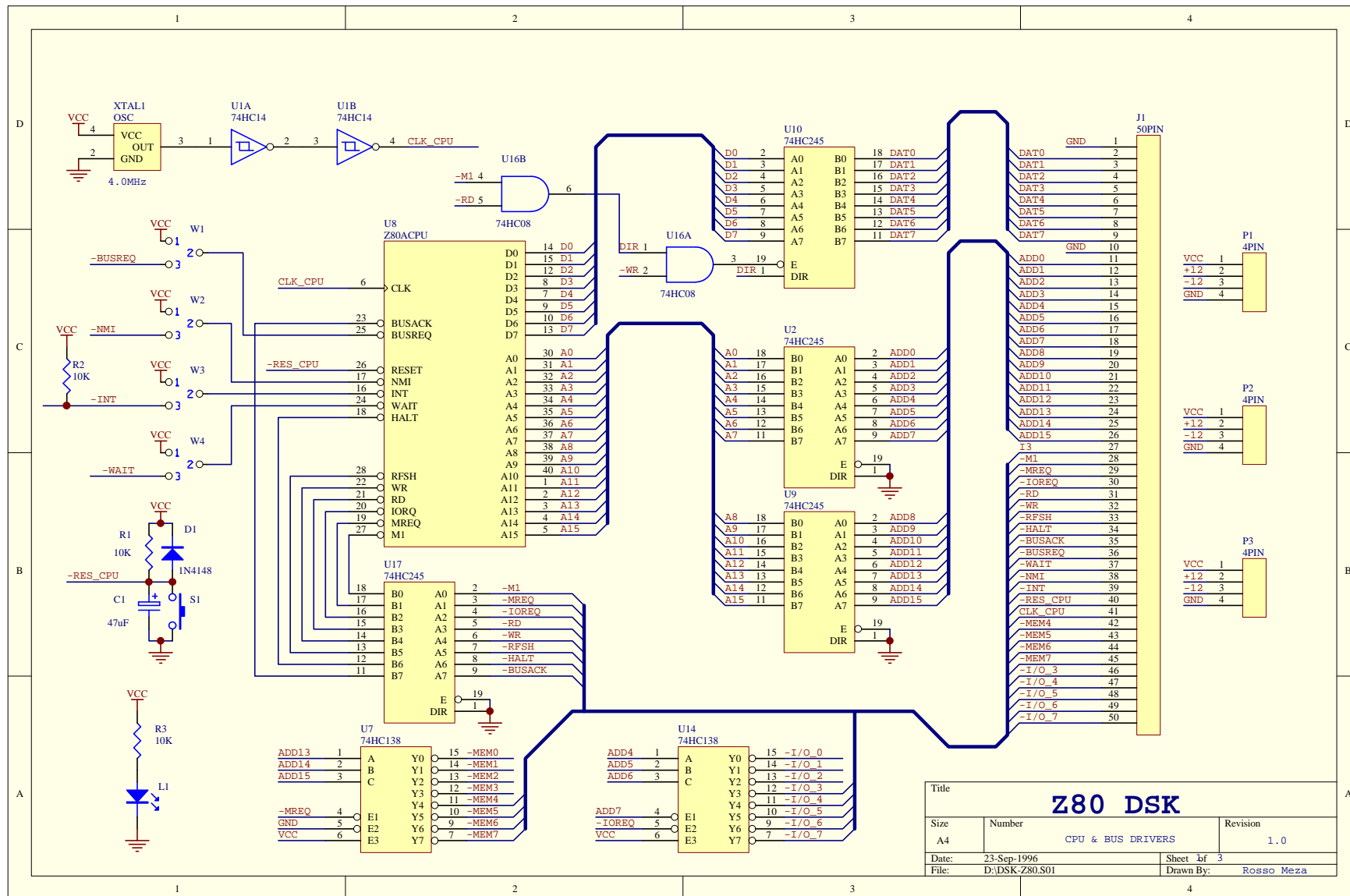
		(0-4FFF)	
Ingresar Archivo	Ingresar un archivo con formato .HEX		I
Puerto Salida	Envía al puerto seleccionado el valor deseado	Puerto deseado Valor deseado (0-FF)	PSxx,yy donde xx=PD yy=VD
Puerto Entrada	Devuelve el contenido del puerto seleccionado	Puerto deseado (0-FF)	PExx donde xx=Puerto deseado

Estas instrucciones se ingresan a través del teclado de la PC.

12. Apéndice B: Esquemáticos

Los esquemáticos que a continuación se detallan son los siguientes:

- Esquemáticos de la placa de desarrollo (pág. 38/40)
- Esquemático de la placa para la aplicación con el PIO (pág. 41)
- Esquemático de la placa para la aplicación con el CTC (pág. 42)
- Top Overlay de la placa de desarrollo (pág. 43)
- Top Overlay de la placa de extensión para el bus DATA/ADD J1 (pág. 44)
- Top Overlay de la placa de extensión para el bus PIO/SIO/CTC J2 (pág. 45)



Title			Z80 DSK		
Size	Number	Revision			
A4	CPU & BUS DRIVERS	1.0			
Date:	23-Sep-1996	Sheet	1	of	3
File:	D:\DSK-Z80.S01	Drawn By:	Rosso Meza		

