

300617

32
2ej



UNIVERSIDAD LA SALLE

**ESCUELA DE ING. MECANICA ELECTRICA
INCORPORADA A LA U.N.A.M.**

**EL MICROPROCESADOR 8085 APLICADO A JUEGOS DE VIDEO,
USANDO MANIPULADORES**

TESIS PROFESIONAL

**Que para obtener el Título de:
INGENIERO MECANICO ELECTRICO
CON AREA PRINCIPAL EN
ELECTRONICA Y COMUNICACIONES**

P r e s e n t a

HUMBERTO SANCHEZ SABBAGH

**TESIS CON
FALLA DE ORIGEN**

México, D. F.

1988



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

INDICE

		Pág.
CAPITULO 1	DESCRIPCION GENERAL DE UN SISTEMA	1
1.1	Descripción por bloques del "Hardware".	1
1.2	Necesidades de "Software".	8
CAPITULO 2	DESCRIPCION DEL SDK-85 Y DE CADA UNA DE SUS PARTES	10
2.1	BOBS y su Interface.	11
2.2	Direccionamiento de Memoria.	32
2.3	Direccionamiento de Puertos de Entrada-Salida.	33
CAPITULO 3	DESARROLLO DEL CIRCUITO DE VIDEO Y DE TECLADO	40
3.1	Descripción del Circuito de Video.	40
3.2	Conexión del Circuito de Video al Microcomputador.	53
3.3	Conexión del Teclado al Microcomputador.	54
CAPITULO 4	DESARROLLO DE LOS MANIPULADORES	57
4.1	Manipuladores y Arreglo de Circuitos Necesarios.	57
4.2	Conexión del Manipulador al Microcomputador.	66
CAPITULO 5	PROGRAMACION	68
5.1	Diseño de Caracteres Gráficos y Alfanuméricos.	68
5.2	Programación de los Puertos de Entrada-Salida.	69
5.3	Desarrollo del Sistema Operativo.	78
5.4	Desarrollo de un Programa Ejemplo.	87

CONCLUSIONES

95

APENDICE

1-a) LISTADO DEL PROGRAMA.	76
1-b) LISTADO DEL PROGRAMA EJEMPLO.	114
2-a) DIAGRAMA DEL SDK-05.	145
2-b) CIRCUITO PROPUESTO.	152

BIBLIOGRAFIA

158

CAPITULO 1.

DESCRIPCION GENERAL DE UN SISTEMA

Este primer capítulo tiene como objetivo dar una introducción a lo que se conoce como SISTEMA dentro de la computación y las partes que lo forman.

Definiendo, un Sistema es una combinación de componentes que reciben información que procesan siguiendo una secuencia organizada de pasos de programa, a los que se les conoce como instrucciones, con el fin de obtener determinados resultados.

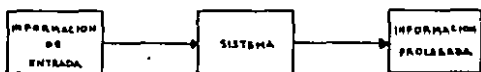


Fig. 1.1 Diagrama de bloques de un Sistema.

De la misma definición se puede determinar cuales son las partes que lo forman: a) al conjunto de componentes electrónicos (la máquina en sí), se conoce como "Hardware" y b) a la secuencia organizada de instrucciones se conoce como "Software".

1.1 DESCRIPCION POR BLOQUES DEL "HARDWARE".

Una computadora en forma general consta de tres partes: Unidad Central de Proceso (CPU), Memoria y Puertos de Entrada/Salida. Estos bloques se encuentran a su vez comunicados por medio de tres canales: Canal de Datos, Canal de Dirección y el Canal de Control, figura 1.2.

1.1.1 UNIDAD CENTRAL DE PROCESO (CPU).

La "CPU" es la parte principal de todo sistema ya que en ella se realizan las operaciones aritméticas y lógicas, y en donde se lleva el control y sincronización de las señales de control, figura 1.3.

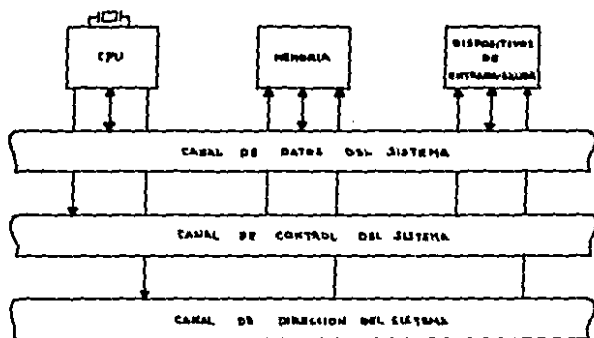


Fig. 1.2 Diagrama de bloques de una Computadora.

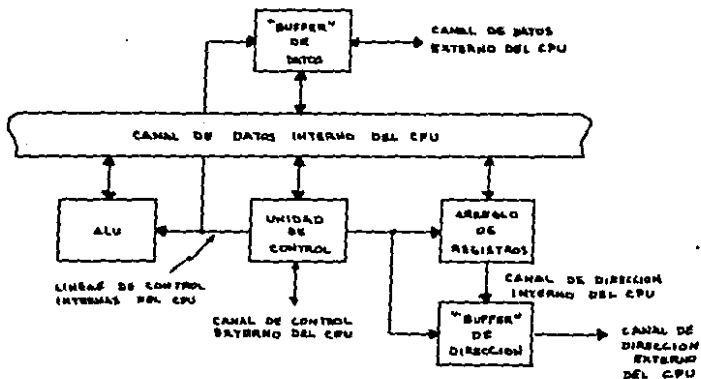


Fig. 1.3 Unidad Central de Proceso.

En las computadoras la "CPU" se encuentra generalmente implementada en un solo circuito integrado conocido como microprocesador. La Unidad Central de Proceso está formada por: a) la Unidad Aritmética-Lógica (ALU), b) el Arreglo de Registros y c) la Unidad de Control.

a) Unidad Aritmética-Lógica.

Esta unidad puede recibir datos y efectuar con ellos operaciones aritméticas, lógicas, de comparación y movimiento. La "ALU" tiene algunos registros que sirven para almacenar datos con los que va a realizar operaciones o el resultado de alguna operación, como es el caso del registro más importante de la "ALU", el Acumulador. Generalmente al comienzo de una operación, el Acumulador contiene uno de los operandos y al finalizar la operación contiene el resultado.

b) Arreglo de Registros.

La "CPU" cuenta con dos tipos de registros: los registros de propósito general y los registros de propósito específico. Los primeros se utilizan para el manejo de datos y los últimos tienen funciones ya establecidas como el Registro Contador del Programa (PC) y el Apuntador de Pila (SP).

c) Unidad de Control.

Esta unidad recibe las instrucciones desde la memoria y decide cuando, como y que operaciones se deben realizar para ejecutar cada instrucción. La Unidad de Control se considera como el cerebro de la computadora.

1.1.2 MEMORIA

La memoria es un dispositivo destinado para almacenar instrucciones y datos que son suministrados a la "CPU" para ser ejecutados y procesados cuando esta lo demande. Existen varias clasificaciones de las memorias, las más importantes son:

a) Por su forma de Acceso.

Acceso Directo o Aleatorio.- Memorias en las cuales se asocia una dirección de memoria y cuyo acceso es inmediato, ejemplo: memorias RAM, ROM, PROM, EPROM, EEPROM. Ver figura 1.4 a).

Acceso Secuencial.- En estas memorias para poder pasar de una dirección a otra se tiene que pasar por todas las direcciones intermedias, ejemplo: La unidad de cintas. Ver figura 1.4 b).

Asociativas.- Memorias en las que el acceso a las direcciones se realiza mediante la información contenida en un índice, ejemplo: La unidad de discos.



Fig. 1.4 Acceso de memoria a) Aleatorio y b) Secuencial.

b) Volátiles y no volátiles.

Volátiles.- Este tipo de memorias pierden toda la información en el momento en que se desconecta la fuente de poder, ejemplo: memorias RAM.

No volátiles.- Son aquellas memorias que conservan los datos aún después de desconectar la fuente de poder, ejemplo: unidad de cintas, de discos, memorias ROM, PROM, EPROM, etc.

c) Según la operación a efectuar.

Memorias Vivas.- En estas memorias se puede leer y escribir durante su operación normal, en otras palabras se puede tomar el contenido de las localidades de memoria y/o modificarlo, ejemplo: unidad de cintas, discos, memorias RAM, etc.

Memorias Muertas.- Se puede leer el contenido de cada dirección de memoria, pero no se puede modificar. En esta clasificación se encuentran las memorias ROM, PROM, EPROM, EEPROM.

El desarrollo de las memorias con semiconductores han modificado los conceptos de fabricación de las computadoras. De las características que proporcionan las memorias fabricadas con semiconductores se pueden mencionar las siguientes:

- a) Memorias más rápidas, compactas y de bajo consumo de potencia.
- b) Han causado un cambio total en la arquitectura de las computadoras con la introducción de la microcomputación.
- c) Su versatilidad y compatibilidad con otros circuitos integrados han sido útiles para nuevos propósitos.

A continuación se describen a grandes rasgos, las características principales de las memorias fabricadas con semiconductores:

Memorias RAM.-

Las memorias de acceso aleatorio o RAM (Random Access Memory), son memorias a las que se les puede cambiar el contenido de sus localidades (con la función de escribir), lo mismo que obtener los contenidos de sus localidades (con la función de leer), durante su proceso normal de operación. Esta clase de memorias se les conoce también como memorias de lectura-escritura.

Estas memorias son importantes porque con ellas se trabaja bajo condiciones normales de operación de la computadora, almacenando instrucciones y datos. En otras palabras los datos e instrucciones almacenados en discos, cintas y otros dispositivos, se pasan a memoria RAM antes de su ejecución.

Hay dos tipos principalmente de memorias RAM: estáticas y dinámicas. Las memorias estáticas están formadas por un "flip-flop" que al ser activado se carga con el dato de entrada 0 o 1, figura 1.5. El dato cargado se conserva hasta que se vuelve a activar el "flip-flop" o se quita la alimentación, de ahí el nombre de memoria estática.



Fig. 1. 5 RAM Estática.

Las memorias dinámicas, figura 1.6, utilizan una estructura capacitiva por lo que pueden conservar por muy pocos milisegundos la carga depositada en ellas, de ahí el término de dinámicas refiriéndose a que cambian de estado. Para no perder la información, las memorias dinámicas se deben de refrescar (recargar) el contenido de cada localidad.

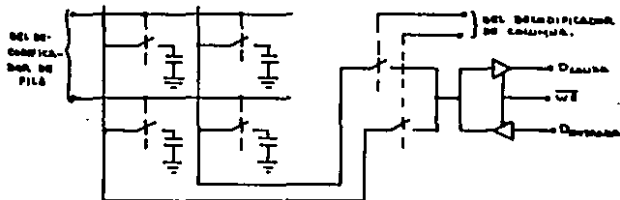


Fig. 1.6 RAM Dinámica.

Memorias ROM.-

Las memorias de solo lectura o ROM (Read Only Memory), son memorias a las que solo se puede leer el contenido de sus localidades, pero no se puede escribir en ellas. El contenido de la memoria de solo lectura es fijo e inalterable, establecido en el momento de la fabricacion. Al igual que las memorias de lectura-escritura (RAM), la memoria de solo lectura es de acceso aleatorio. Otra caracteristica importante es que las memorias ROM son memorias no volátiles.

Las memorias ROM se utilizan para almacenar en forma permanente datos e instrucciones. Estos en conjunto forman programas que son muy importantes, tales como el monitor que es el que permite que el usuario pueda llegar a tener el control de la "CPU".

Las memorias de solo lectura al igual que todos los componentes electrónicos han sido objeto de que se les mejore de acuerdo a las necesidades y los costos. Por tal motivo se han fabricado diferentes tipos de memorias que permiten una gran versatilidad, como son:

a) PROM.- Las memorias de solo lectura programables (Programmable Read Only Memory), constan de semiconductores que permiten ser programadas una sola vez posteriormente a su fabricacion.

b) EPROM.- Las memorias de solo lectura programables y borrables (Erasable Programmable Read Only Memory), son mas populares que las ROM y las PROM, porque tienen la ventaja de que pueden programarse y borrarse varias veces. Estas memorias almacenan cargas que permanecen atrapadas hasta que no se les aplique

energía externa, como es el caso de la luz ultravioleta.

c) EEPROM.- Las memorias de solo lectura programables y eléctricamente borrables (Electrically Erasable Programmable Read Only Memory), son memorias semejantes a las EPROM's, pero en lugar de requerir luz ultravioleta para borrar el contenido de sus localidades, requiere de un voltaje aplicado a una de sus patas terminales.

1.1.3. PUERTOS DE ENTRADA/SALIDA (E/S).

Cualquier tipo de comunicación que se quiere establecer con la "CPU" se realiza a través de ciertos arreglos de circuitos conocidos como interfaces, que proveen el medio de comunicación entre el sistema y los dispositivos periféricos. Las interfaces deben contar por lo menos de los siguientes dispositivos:

- a) Decodificador de códigos de selección del dispositivo, para seleccionar únicamente a un dispositivo.
- b) Registro de Control o de Comandos para recibir órdenes de la computadora.
- c) Registro de Datos de Entrada y Salida para efectuar la transferencia de datos entre computadora-interface e interface-dispositivo periférico.
- d) Registro de Bandejas que sirve para informar a la "CPU" que se encuentra listo para recibir o transmitir datos.

A los registros de la interface se les da el nombre de puertos. Los puertos son el conjunto de circuitos necesarios por donde se recibe y transmite la información desde y hacia un dispositivo periférico, y de acuerdo a la función que estén realizando se les conoce como Puerto de Entrada y Puerto de Salida respectivamente. Los puertos en fan o reciben información en paralelo (varios "bits" simultáneamente) y en serie (un "bit" a la vez), en una gran variedad de aplicaciones (puerto paralelo de 8 "bits", puerto paralelo de 4 "bits", puerto serial síncrono, puerto serial asíncrono, etc.).

1.1.4. DISPOSITIVOS DE ENTRADA/SALIDA.

Finalmente el usuario puede establecer comunicación con el sistema por los dispositivos periféricos o de Entrada/Salida.

Entre los dispositivos de entrada más comunes tenemos: el

teclado, unidad de cinta magnética, disco magnético, manipuladores, etc.

Y entre los dispositivos de salida tenemos a los siguientes: tubo de rayos catódicos, impresoras, unidad de cinta magnética, disco magnético, etc.

1.1.5. CANALES.

Este concepto consiste en un grupo de líneas por las cuales se transiere información de un dispositivo a otro, mientras que los dispositivos restantes que se encuentren conectados al mismo canal se deberán comportar como si no existieran. El propósito fundamental de la aplicación de los canales, es el de reducir el número de líneas de conexión requeridas para la transferencia de información. En las computadoras se utilizan tres canales para los diferentes tipos de información, y estos son:

- 1) Canal de Datos, (Bidireccional).
- 2) Canal de Dirección, (Unidireccional).
- 3) Canal de Control, (Unidireccional).

1.2 NECESIDADES DE SOFTWARE.

Una vez determinado el "Hardware" del sistema, es necesario contar con una serie de instrucciones básicas con el fin de proporcionar al usuario acceso a los recursos de la computadora y la forma de controlarlos durante el desarrollo y ejecución de un programa. A lo anterior se le conoce comúnmente como Sistema Operativo.

Los Sistemas Operativos constan de lo siguiente:

- 1.- Programa Monitor.
- 2.- Programa Soporte (Condiciones normales de operación).

1.2.1. PROGRAMA MONITOR.

En el momento de encender la fuente de alimentación, el microprocesador comienza a ejecutar las "instrucciones" que lo indica el registro "Contador del Programa". Estas "instrucciones" no son significativas por lo que se requiere empezar desde una localidad específica de memoria ROM (puede ser también de PROM, EPROM o EEPROM), en donde se inicie una secuencia de

pasos de programa, que le permita al usuario el control de las acciones y que pueda indicarle a la computadora la acción a seguir.

Los microprocesadores cuentan con una entrada RESET que cuando se activa, el Contador del Programa toma el valor 0000H por lo que el programa monitor debe de encontrarse cargado en esta localidad.

1.2.2 PROGRAMA SUFORTE.

Una vez que se ha establecido la comunicación entre usuario y computadora, esta debe de ser capaz de interpretar las instrucciones que le ordene el usuario, saber en donde almacenará los datos y en donde emitirá los resultados.

CAPITULO 2.
DESCRIPCION DEL SDK-85
Y DE CADA UNA DE SUS PARTES

En la actualidad existen microcomputadores educativos en el mercado que tienen como finalidad facilitar el entendimiento y proporcionar al usuario práctica en la elaboración de programas.

El diagrama de bloques de la figura 2.1, muestra al microcomputador educativo SDK 85, en donde se puede observar que cuenta con el microprocesador 8085 como "CPU", que su utilización es uno de los objetivos de este trabajo de tesis. Los componentes mostrados en los bloques representan lugares en el circuito impreso del SDK 85. Para simplificar el diagrama algunas líneas de control han sido omitidas.

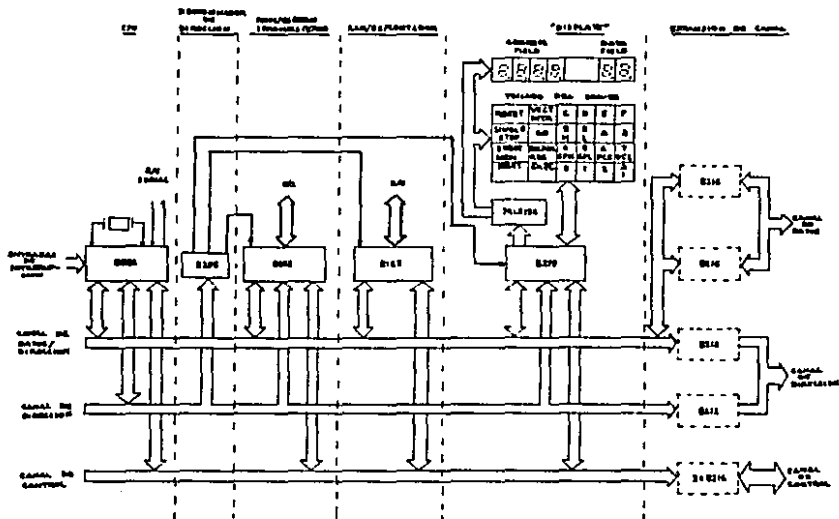


Fig. 2.1 Diagrama de Bloques del SDK 85.

El objetivo de este capítulo es proporcionar características funcionales del 8085 e interfaces así como describir como se realizará el direccionamiento de memoria y de los puertos de Entrada/Salida.

2.1 8085 Y SU INTERFACE.

El 8085 de la compañía Intel es un microprocesador de 8 "bits" en paralelo que se fabrica en un circuito integrado de 40 patas terminales. El 8085 (figura 2.2), tiene como características que: requiere de una alimentación de +5 volts de corriente directa; utiliza únicamente una señal de reloj; sus entradas de tiempo se conectan directamente al cristal; opera con un reloj con periodo de 370 nanosegundos y frecuencia de 3 MHz; cuenta con la capacidad de transmisión serie, de entrada/salida de datos; tiene 5 entradas para solicitudes de interrupción.

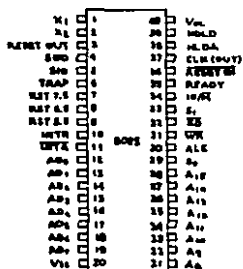


Fig. 2.2 Distribución del 8085.

En forma superficial a continuación se describirán las funciones de las patas terminales del 8085:

A15-A8 (Salidas de 3 estados). Canal de Dirección.- El "CPU" envía por este canal los 8 "bits" de mayor orden de una dirección de memoria o el código de selección de un puerto de

Entrada/Salida. Toman el tercer estado (estado de alta impedancia), durante los modos "Hold" y "Halt".

AD7-AD0 (Entrada/Salida de 8 estados). Canal de dirección/datos.- El "CPU" envía por este canal los 8 "bits" de menor orden de una dirección de memoria o el código de un dispositivo de E/S durante el estado T1 del ciclo de máquina. Un período de reloj es un estado, de 3 a 6 estados constituyen un ciclo de máquina y de 1 a 5 ciclos de máquina forman un ciclo de instrucción, figura 2.3. El "CPU" utiliza también este canal de datos durante los estados T2 y T3. Toman el tercer estado durante los modos "Hold" y "Halt".

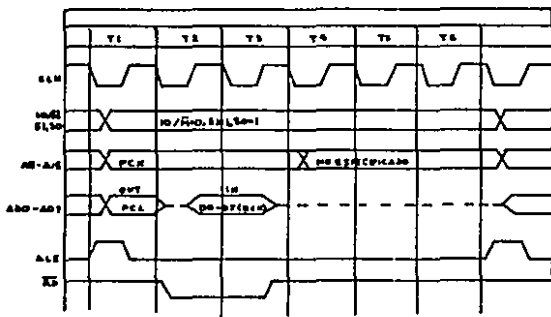


Fig. 2.3 Diagrama de tiempos de un ciclo de lectura de memoria.

ALE (Salida). Habilitar "latch" de dirección (Address Latch Enable).- Esta señal toma nivel alto durante el estado T1 del ciclo de máquina y la utilizan los dispositivos externos para cargar en un registro "latch" los 8 "bits" que el microprocesador está enviando por el canal AD7-AD0, figura 2.3.

READY (Entrada). Listo.- Esta señal indica que el dato desde memoria o desde un dispositivo de entrada se encuentra en el canal de datos del 8085. La señal "READY" se utiliza para sincronizar al microprocesador con memorias o puertos de E/S más

lentos que él. Si después de enviar la dirección por el canal de dirección, el 8085 no recibe un nivel alto en esta entrada, esto entra en estados de espera TW , tantos como la línea "READY" tenga nivel bajo, figura 2.4.

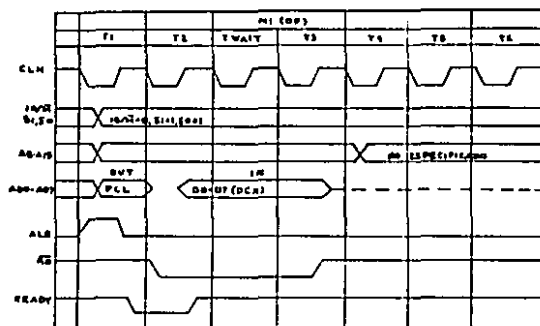


Fig. 2.4 Diagrama de tiempos con estados de espera (TW).

\overline{WR} (Salida de tres estados). Escribir.- La señal \overline{WR} la utiliza el microprocesador para indicar que el dato presente en el canal de datos se escribirá o cargará en una localidad de memoria o en un puerto de E/S. Toma el tercer estado durante los modos "Hold" y "Halt".

\overline{RD} (Salida de tres estados). Leer.- La señal \overline{RD} la utiliza el 8085 para indicar que va a leer una localidad de memoria o un puerto de E/S y que el canal de datos está disponible para la transferencia. Toma el tercer estado durante los modos "Hold" y "Halt".

IO/\overline{M} (Salida).- El microprocesador utiliza esta salida para indicar que la función de "leer" o "escribir" indicada por las salidas \overline{RD} y \overline{WR} respectivamente es con la memoria o con un puerto de E/S, como se muestra a continuación:

$$IO/\overline{M} = 1 \quad \text{y} \quad \overline{RD} = 0 ; \text{ leer puerto de E/S.}$$

$\overline{IO/\overline{M}} = 1$ y $\overline{WF} = 0$; escribir en puerto de E/S.
 $\overline{IO/\overline{M}} = 0$ y $\overline{RD} = 0$; leer memoria.
 $\overline{IO/\overline{M}} = 0$ y $\overline{WR} = 0$; escribir en memoria.

*Toma el tercer estado durante los modos "Hold" y "Halt".

HOLD (Entrada). Flotar. - Una señal con nivel alto en esta entrada le indica al microprocesador que un dispositivo está solicitando los canales de dirección y datos. El 8085 entra en el modo "HOLD", liberando (poniendo a flotar en el tercer estado) los canales mencionados y las líneas \overline{RD} , \overline{WR} , $\overline{IO/\overline{M}}$ y ALE tan pronto como termina la ejecución del ciclo de máquina en proceso. Esto permite que se realice la operación "OR" de estos canales con otros dispositivos para poder efectuar el acceso directo a memoria (DMA Direct Memory Access). Este tipo de operaciones la utilizan los dispositivos como el disco duro y los discos ligeros, quienes toman el control de los canales para la transferencia directa de datos memoria-dispositivo. El 8085 toma el control de los canales de nuevo hasta que la línea "HOLD" toma nivel bajo.

HOLD (Salida). Reconocimiento de "HOLD" (Hold Acknowledge). - Cuando esta línea toma nivel alto indica que el 8085 ha recibido una solicitud de estados "HOLD" y que liberará los canales y otras líneas en el próximo ciclo de máquina. Esta línea regresa a nivel bajo hasta que se retira la solicitud de estados "HOLD".

S0 y S1 (Salidas). Estado del canal de datos. - Define el estado del canal de datos durante los procesamientos del 8085. Los estados posibles que pueden tomar las salidas S0 y S1 indican:

S1	S0	=	Alto (Halt).
0	0	=	Escribir.
0	1	=	Leer.
1	0	=	"Fetch".
1	1	=	

INTR (Entrada). Solicitud de interrupción. - Se utiliza para solicitudes de interrupción al 8085. Esta línea es muestreada en el último estado de cada instrucción. Si la línea toma nivel bajo el contador del programa (PC), se inhibe para que no se incremente; se activa la señal \overline{INTA} y el 8085 entra en un "ciclo de interrupción" después de terminar la instrucción en proceso. La línea "INTR" se deshabilita al activar la línea "RESET" del microprocesador.

INTÁ (Salida). Reconocimiento de interrupción (Interrupt Acknowledge).- El 8085 lo utiliza para indicar que acepta la interrupción.

RST 5.5, RST 6.5, RST 7.5 (Entradas). Iniciar interrupciones (Restart).- Estas tres entradas tienen el mismo diagrama de tiempos que la línea INTA con la excepción de que generan internamente en forma automática una instrucción de reinicio RST. La prioridad de interrupción de estas líneas es la siguiente:

RST 7.5 = Mayor
RST 6.5 = Media
RST 5.5 = Menor

A su vez las tres líneas tienen mayor prioridad que la línea INTR.

TRAP (Entrada).- La línea "TRAP" sirve para hacer una interrupción no mascarable y es la que tiene mas alta prioridad de todas las líneas de interrupción del 8085.

RESET IN (Entrada).- Cuando toma el nivel bajo, el contador del programa se carga con ceros y limpia (toma nivel bajo), los contenidos internos de habilitar interrupción y "HLDA". Ninguno de los otros registros o banderas (excepto el registro de instrucción), es afectado. El 8085 permanece en la condición de "RESET" (Reiniciar), tanto como la línea "RESET IN" este activa.

RESET OUT (Salida).- Indica que la línea "RESET IN" está activa. Se puede utilizar como una línea de reinicio del sistema. La señal está sincronizada con el reloj del 8085.

X1 y X2 (Entradas).- Estas líneas se conectan a un cristal, a un reloj externo o a un circuito RC (resistencia-capacitor), para activar el reloj interno del 8085. La frecuencia de entrada (para el caso es de 6 MHz) se divide entre dos para obtener la frecuencia interna de operación (3 MHz) del 8085.

CLK (Salida).- La frecuencia de la señal "CLK" es la frecuencia interna de operación del 8085, es decir la frecuencia de "CLK" es la mitad de la frecuencia de entrada de las líneas X1 y X2.

SID (Entrada). Entrada de Datos en Serie (Serial Input Data).- El dato presente en la línea se carga en el "bit" 7 del acumulador, cuando el 8085 ejecuta la instrucción RIM.

SOD (Salida). Salida de Datos en Serie (Serial Output Data).- La línea SOD toma nivel alto o bajo según se especifique en la instrucción SIM cuando esta se ejecuta por el 8085.

Vcc.- Alimentación de +5 Volts.

Vss.- Referencia a tierra.

2.1.1. ARQUITECTURA DEL 8085.

La arquitectura del 8085, figura 2.5, consta de las siguientes partes funcionales:

- 1) Arreglo de Registros.
- 2) Registro de Instrucción.
- 3) Decodificador de Instrucción.
- 4) Sección de Control y de Tiempos.
- 5) Unidad Aritmética y Lógica.
- 6) "Buffer" del Canal de Datos.
- 7) Control de Interrupciones y Datos en Serie.

1) Arreglo de Registros.- Consta de los siguientes registros y circuitos auxiliares:

- Contador del Programa (Program Counter, PC), de 16 "bits".
- Apuntador de Pila (Stack Pointer, SP), de 16 "bits".
- Seis Registros de Propósito General de 8 "bits" arreglados en pares y referidos como: B, C, D, E, H y L.
- Un circuito Incrementador-Decrementador de 16 "bits".
- Un "Latch" de dirección de 16 "bits".

-Contador del Programa (PC).-

El Contador del Programa es un registro que siempre tiene la dirección de la próxima localidad de memoria que se va a acceder para obtener el código de la próxima instrucción a ejecutarse por el microprocesador. Al comienzo del ciclo de la instrucción, el 8085 envía el contenido del "PC" al canal de dirección, por medio del cual direcciona a una localidad de memoria específica. El "PC" se incrementa en uno cada vez que el microprocesador lee el

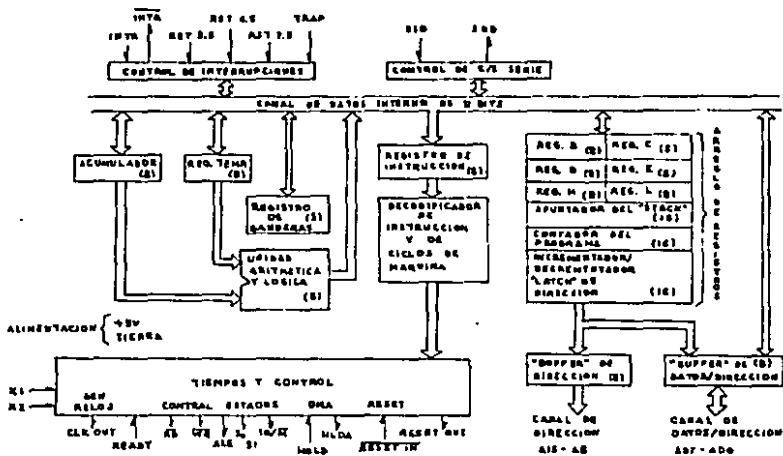


Fig. 2.5 Arquitectura del 8085.

código de la instrucción contenida en la localidad direccionada. De esta forma el "PC" direcciona secuencialmente a las localidades de memoria donde se encuentra almacenado el programa. Figura 2.6.

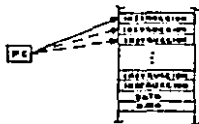


Fig. 2.6 El "PC" apunta a localidades que contienen instrucciones.

-Apuntador de Pila (SP).-

Durante el procesamiento de la información es muy útil contar con área en la memoria RAM, donde se puedan almacenar temporalmente datos. Para agilizar estas funciones es muy importante tener la necesidad de buscar la dirección de esa área cada vez que se desea almacenar un dato o rescatar el último dato almacenado. Este área se llama "Stack". Figura 2.7.

El apuntador de Pila es un registro de 16 "bits", que contiene una dirección de memoria RAM a partir de la cual en forma descendente, se pueden salvar los contenidos de un registro por o a partir del cual, en forma ascendente se obtienen los últimos dos datos almacenados en esa área.

-Seis Registros de Propósito General.-

La memoria no únicamente se carga con los códigos de las instrucciones del programa sino que también almacena datos. Cuando el "CPU" lee los datos los recibe en seis Registros de 8 "bits" de Propósito General. Estos seis Registros B, C, D, E, H y L se pueden utilizar en forma individual como registros de 8 "bits", o en registros pares para manejarse como registros de 16 "bits" como sigue: B-C, D-E y H-L. Los registros B, D y H contienen los "bits" de más alto orden, y los registros C, E y L los "bits" de más bajo orden. Como registros se designan como: B (para B-C), D (para D-E) y H (para H-L). A la dirección de memoria que se forma con los contenidos de H y L se conoce como dirección M.

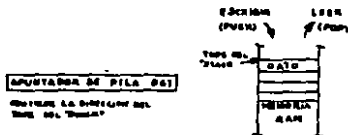


Fig. 2.7 Apuntador de Pila.

-Circuito Incrementador/Decrementador.-

El Circuito Incrementador/Decrementador recibe datos del "Latch" de Dirección y después de efectuar la operación deseada, el resultado se envía a un registro par del Arreglo de Registros.

-"Latch" de Dirección.-

El "Latch" de Dirección también recibe datos de la "bits" de cualquiera de los tres registros pares, del Contador del Programa o del Apuntador de Pila y los conduce al "Buffer" de Dirección (A15-A0), o al Circuito Incrementador/Decrementador.

2) Registro de Instrucción (RI).- Cuando el Contador del Programa se carga con la dirección de la primera instrucción del programa, el microprocesador envía el contenido del "PC" al "Buffer" de Dirección y lo transmite por el Canal de Dirección a la memoria indicada. Después de recibir la dirección y las señales de control correspondientes, la memoria responde enviando el contenido de la localidad direccionada al microprocesador por el Canal de Datos. Posteriormente el B035 lee este contenido en un registro conocido como "Registro de Instrucción" (RI). Este registro es exclusivo para las instrucciones, así que cualquier valor que se encuentre en este registro el B035 lo interpretará como una instrucción.

3) Decodificador de Instrucción.- El contenido del "RI" posteriormente pasa por el Decodificador de Instrucciones. La salida del Decodificador al combinarse con varias señales de tiempos, que genera la sección de Control, proporciona señales de control a todos los componentes del microprocesador (Arreglo de Registros, Registro de Instrucción, "ALU", etc.), figura 2.8.

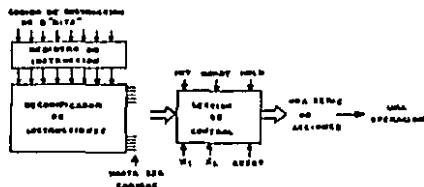


Fig. 2.8 Decodificador de Instrucciones.

4) Sección de Control y de Tiempos.- La salida del Decodificador de Instrucción y las señales de control externas como INTR, READY, HOLD, etc., alimentan a la Sección de Control y Tiempos para que genere las señales de tiempos y ciclos de máquina de la instrucción en proceso. Por ejemplo, si el "PC" contiene la dirección 1500H (INR E), al iniciarse el próximo ciclo de instrucción se efectúan los siguientes pasos:

- 1.- El 8085 envía el contenido del Contador del Programa por el Canal de Dirección, al igual que las señales de control correspondientes, figura 2.9 (a).
- 2.- La memoria responde a la dirección y a las señales de control, con el contenido de la localidad 1500H por el Canal de Datos. El microprocesador carga el contenido en el Canal de Datos al Registro de Instrucción e incrementa en uno el contenido en el Contador del Programa, figura 2.9 (b).
- 3.- El Decodificador de Instrucción efectúa la decodificación y la Sección de Control manda las señales adecuadas para incrementar en uno al registro E, figura 2.9 (c). Y es de esta forma como el 8085 queda listo para iniciar otro ciclo de instrucción.

5) Unidad Aritmética y Lógica.- La Unidad Aritmética y Lógica puede realizar las siguientes funciones:

- 1.- Suma binaria.
- 2.- Operaciones aritméticas y lógicas.
- 3.- Complementar una palabra de dato.
- 4.- Correr un bit a la derecha o a la izquierda una palabra de dato.
- 5.- Registrar información importante del resultado de las operaciones aritméticas y lógicas como acarreo, signo, acarreo auxiliar, paridad y si el resultado es cero. El registro que almacena todo esto se conoce como Registro de Banderas.

La "ALU" del 8085 utiliza a los siguientes registros para almacenar datos de la operación a efectuar o ya realizada:

- a) Acumulador de 8 "bits".
- b) Registro de Banderas de ocho "bits": cero, acarreo, signo, paridad y acarreo auxiliar; 3 "bits" no se usan.
- c) Registro Temporal, "TMP" de 8 "bits".

La "ALU" se alimenta del Registro Temporal "TMP", del Acumulador y del "bit" de acarreo del Registro de Banderas. El resultado de la operación se deposita usualmente en el Acumulador

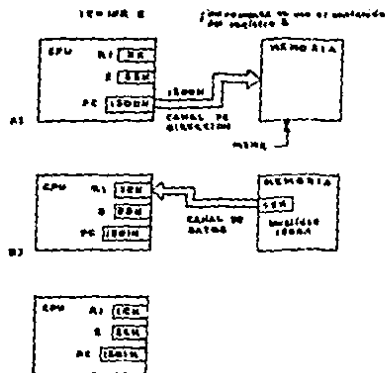


Fig. 2.9 Pasos en la ejecución de la instrucción INR E.

o también se puede depositar en el Canal de Datos interno para después pasarlo a los registros.

a) Acumulador.- El Acumulador (A), es uno de los registros más importantes del microprocesador. Este registro es de 8 "bits", usualmente contiene uno de los dos operandos de las operaciones aritméticas y lógicas y en donde normalmente se deposita el resultado de las operaciones.

Con el Acumulador se pueden hacer las mismas funciones que con los Registros de Propósito General, además de que es el único que puede enviar o recibir datos de los puertos de salida y entrada respectivamente.

Resumiendo, el Acumulador se puede cargar desde el "ALI" después de una operación aritmética o lógica y desde el Canal de Datos interno (desde un registro, desde la memoria o desde un puerto de entrada). Además puede transferir datos a la "ALU" y al Canal de Datos interno (dirigiéndose a un registro, a la memoria o a un puerto de salida).

b) Registro de Banderas.- El 8085 tiene un registro de 5 "bits" que sirve para monitorear ciertos resultados de las operaciones

de la "ALU", figura 2.10. A la información que almacena este registro se conoce como banderas de estado y esta se actualiza después de cada operación con alguno de los registros. Es importante mencionar que no todas las operaciones modifican a todas las banderas (5 banderas, 3 bits no se usan).

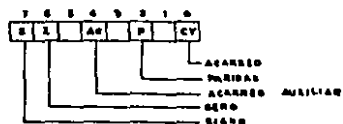


Fig. 2.10 Registro de Banderas.

c) Registro Temporal.- El Registro Temporal "TMP", recibe información desde el Canal de Datos interno (desde la memoria o de algún registro), para enviarla a la "ALU". El Registro Temporal se carga temporalmente con uno de los dos operandos de la función aritmética o lógica y el otro se carga en el Acumulador.

6) "Buffer" del Canal de Datos.- El "buffer" interno bidireccional de 8 "bits", de tres estados, se usa para aislar el Canal de Datos interno de la "CPU" del Canal de Datos externo (D0-D7).

7) Control de Interrupciones y Datos en Serie.- La Unidad de Control de las interrupciones determina cuando se reciben las interrupciones, las prioridades de las mismas y la secuencia de eventos que se tienen que llevar a cabo para poder responder a la solicitud.

La Unidad de Control de Datos en Serie contiene la lógica para la recepción y transmisión de datos en serie.

2.1.2. INTERFACE.

Los circuitos conocidos como Interfaces, permiten el control de la comunicación entre la "CPU" y los dispositivos periféricos

de Entrada-Salida y memorias. Anteriormente se mencionó las partes en que se divide la interface, así que en este inciso se describirán los circuitos que forman la interface del SDK-85.

-Decodificador 8203.-

El SDK 85 utiliza al Decodificador 8203 (figura 2.11), para la selección tanto de la memoria como del 8279 que es el controlador del teclado y de los "displays".

Cuando el 8203 está habilitado una de sus ocho salidas toma nivel 0 en función del valor binario de las entradas A0, A1 y A2. Cuando no está habilitado, todas sus salidas toman nivel 1.

La figura 2.12 muestra como se encuentra conectado el 8203 en el SDK 85, y como se podrá observar, el Decodificador se activa cuando las líneas A14 y A15 toman nivel 0. Las líneas A11 - A13 del 8085 se conectan directamente a las entradas A0 - A2 del 8203, de tal forma que cuando éste se encuentre habilitado, y dependiendo del valor que se encuentre en estas líneas, se seleccionará una memoria o un dispositivo de Entrada-Salida.

-Memorias 8355 y 8755.-

La memoria 8355 es una ROM de 2048 palabras de 8 "bits" y tiene además dos puertos de Entrada-Salida de 8 "bits". El 8755 es una memoria EPROM de 2048 palabras de 8 "bits" y cuenta también con dos puertos de Entrada-Salida de 8 "bits". La única diferencia que existe entre las dos memorias es que una es ROM y la otra es EPROM, siendo inclusive compatibles para a para terminal, figura 2.13.

Una característica del 8355 y del 8755 es que sus dos puertos de Entrada-Salida pueden ser programados para que sus ocho líneas se definan en forma particular como entradas o como salidas. Esto significa que por ejemplo las líneas 1, 4 y 7 de un puerto se pueden programar como entradas y las otras cinco como salidas.

El SDK-85 viene originalmente con un 8355 o con un 8755 y tiene un espacio adicional para alojar a otro más. A continuación se describirán el funcionamiento de sus patas terminales, pero, para no mencionar continuamente a los dos integrados haremos referencia sólo al 8355, a menos que se indique lo contrario.

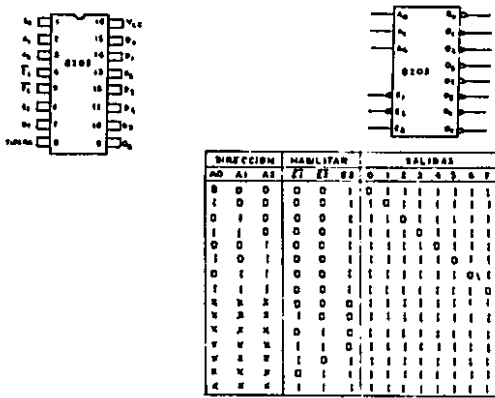


Fig. 2.11 Decodificador 8205.

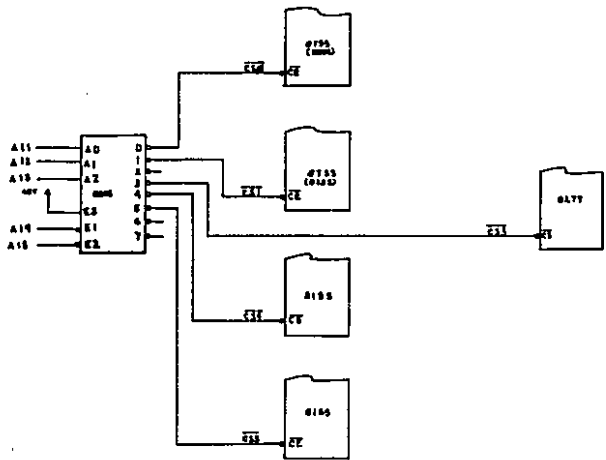


Fig. 2.12 Decodificador 8205 en el SDK 85.

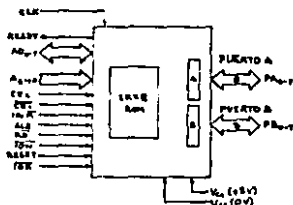
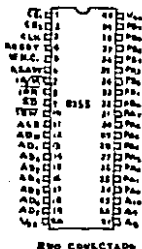
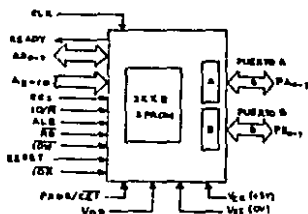
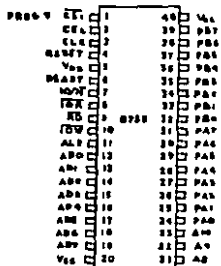


Fig. 2.13 Memorias 8355 y 8755.

ALE (Entrada). Habilitar "latch" de dirección. Las líneas AD0-AD7, AB-A10, IO/ \bar{R} , CE1 y CE2 se cargan en "latches" internos con la transición alto-bajo de la línea ALE.

AD0-AD7 (Entrada-Salida). Canal Bidireccional de Dirección/Datos. Los ocho "bits" de menor orden de una dirección de memoria o de un puerto de Entrada-Salida se encuentran presentes, cuando la línea ALE tiene nivel 1. Durante un ciclo de Entrada/Salida se selecciona el puerto A o B en base al valor de la línea A0 (Tabla 2.4). Si las líneas RD = 0 o WR = 0, cuando se encuentra seleccionado el circuito integrado, los "buffers" de salida presentarán un dato en el Canal AD0-AD7 del sistema. Si la línea IO = 0, los "buffers"

de entrada estarán listos para tomar el dato presente en el Canal ADO-AD7.

AB-A10 (Entradas). Estas líneas tienen los "bits" de mayor orden de una dirección de memoria exclusivamente.

CE2 y CE1 (Entradas). Habilitar integrado. Las dos entradas deben de estar activas (CE2 = 1 y CE1 = 0), para que el 8085 pueda ser habilitado. Si alguna de las dos líneas no está activa, las líneas ADO-AD7 y READY se encuentran en alta impedancia.

IO/M (Entrada). Si IO/M = 1 se selecciona un puerto de Entrada-Salida y si IO/M = 0 se selecciona una localidad de memoria.

RD (Entrada). Si el integrado se encuentra habilitado y RD pasa a nivel 0, se habilitan los "buffers" de salida ADO-AD7, de tal forma que el dato que se encuentre en un puerto de E/S o en una localidad de memoria pasa al canal de datos del sistema. Si RD = 1, los "buffers" de salida toman el tercer estado.

OW (Entrada). Si el integrado se encuentra habilitado y la entrada OW = 0, se habilitan los "buffers" de entrada, de tal manera que un puerto de salida o un registro DDR (Data Direction Register), seleccionado por el valor de los "latches" internos ADO y AD1 se cargue con el dato presente en el canal ADO-AD7, sin influir el estado de IO/M.

CLK (Entrada). Se utiliza para poner a la salida READY en tercer estado, cuando CE1 = 0, CE2 = 1, ALE = 1 después de la transición alto-bajo en esta entrada.

READY (Salida de tres estados). Controlada por CE1, CE2, ALE y CLK. Cuando el integrado está habilitado READY toma nivel 0 cuando ALE tiene nivel 1, permanece en nivel 0 hasta la próxima transición bajo-alto de CLK, después que ALE regresa a nivel 0.

PA0-PA7 (Entrada-Salida). Líneas de Entrada-Salida de propósito general del puerto A. Las direcciones (Entrada-Salida) de las líneas se determinan en el registro DDR A (Data Direction Register A). El puerto A se carga cuando el integrado está

de entrada estarán listos para tomar el dato presente en el Canal AD0-AD7.

AS-A10 (Entradas). Estas líneas tienen los "bits" de mayor orden de una dirección de memoria exclusivamente.

CE2 y CE1 (Entradas). Habilitar Integrado. Las dos entradas deben de estar activas (CE2 = 1 y CE1 = 0), para que el 8355 pueda ser habilitado. Si alguna de las dos líneas no está activa, las líneas AD0-AD7 y READY se encuentran en alta impedancia.

IO/M (Entrada). Si IO/M = 1 se selecciona un puerto de Entrada-Salida y si IO/M = 0 se selecciona una localidad de memoria.

RD (Entrada). Si el integrado se encuentra habilitado y RD pasa a nivel 0, se habilitan los "buffers" de salida AD0-AD7, de tal forma que el dato que se encuentre en un puerto de E/S o en una localidad de memoria pasa al canal de datos del sistema. Si RD = 1, los "buffers" de salida toman el tercer estado.

RDW (Entrada). Si el integrado se encuentra habilitado y la entrada RDW = 0, se habilitan los "buffers" de entrada, de tal manera que un puerto de salida o un registro DER (Data Direction Register), seleccionado por el valor de los "latches" internos AD0 y AD1 se cargue con el dato presente en el canal AD0-AD7, sin influir el estado de IO/M.

CLK (Entrada). Se utiliza para poner a la salida READY en tercer estado, cuando CE1 = 0, CE2 = 1, ALE = 1 después de la transición alto-bajo en esta entrada.

READY (Salida de tres estados). Controlada por CE1, CE2, ALE y CLK. Cuando el integrado está habilitado READY toma nivel 0 cuando ALE tiene nivel 1, permanece en nivel 0 hasta la próxima transición bajo-alto de CLK, después que ALE regresa a nivel 0.

PA0-PA7 (Entrada-Salida). Líneas de Entrada-Salida de propósito general del puerto A. Las direcciones (Entrada-Salida) de las líneas se determinan en el registro DDP A (Data Direction Register A). El puerto A se carga cuando el integrado está

habilitado, $\overline{IOW} = 0$ y el "latch" $\overline{AD0}$ de dirección tiene nivel 0. El contenido del puerto A se lee cuando el integrado está habilitado, $\overline{IOR} = 0$ y el "latch" $\overline{AD0} = 0$, o cuando el integrado está habilitado, $\overline{FE} = 0$, $\overline{IO/\overline{M}} = 1$ y el "latch" $\overline{AD0} = 0$.

PB0-PB7 (Entrada-Salida). Líneas del puerto B, con las mismas características del puerto A, pero las direcciones (Entrada-Salida), las proporciona el registro DDR B, y el "latch" $\overline{AD0}$ debe tener nivel 1.

RESET (Entrada). El nivel 1 en esta entrada causa que todas las líneas de los puertos A y B tomen el modo de entrada.

\overline{IOR} (Entrada). Si el integrado se encuentra habilitado o $\overline{IOR} = 0$, causa que el contenido del puerto o registro DDR seleccionado por el valor de los "latches" internos de $\overline{AD0}$ y $\overline{AD1}$ se envíen a $\overline{AD0-AD7}$, el estado de $\overline{IO/\overline{M}}$ no influye en esta operación. El nivel 0 en \overline{IOR} realiza la misma función que $\overline{IO/\overline{M}} = 1$ y $\overline{RD} = 0$. Cuando \overline{IOR} no se usa en el sistema se debe conectar a Vcc.

Vcc y Vss. Alimentación de + 5 Volts y referencia a tierra respectivamente.

Patas terminales exclusivas del 8755:

PROG/ $\overline{CE1}$ (Entrada). Cuando el 8755 está programado, esta entrada se utiliza para habilitar al integrado (igual que el $\overline{CE1}$ del 8155), en combinación con $\overline{CE2}$. Cuando se está programando el 8755, esta entrada debe tener nivel 0 y recibir un pulso positivo cuando el dato a cargarse está listo en el Canal de Datos.

Vdd (Entrada). Debe conectarse a + 5 Volts cuando el 8755 está programado. Cuando el 8755 se está programando, esta entrada debe recibir un pulso de + 25 Volts como máximo, cuando el dato a cargarse está listo en el Canal de Datos.

-Memoria 8155.-

El circuito integrado 8155 consiste en una memoria RAM de 256 "bytes", dos puertos de Entrada-Salida de 8 "bits", un puerto de Entrada-Salida de 6 "bits" y un "timer" de 14 "bits", figura 2.14.

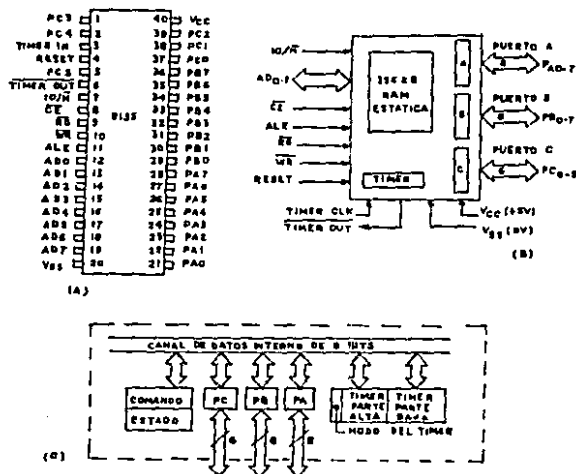


Figura 2.14 a) Configuración de las patas terminales, b) Diagrama de bloques y c) Registros internos del 8155.

A continuación se hará una descripción del funcionamiento de sus patas terminales:

AD0-AD7 (Entrada-Salida de tres estados). Estas ocho líneas se conectan directamente al Canal de Dirección-Datos del 8085.

ALE (Entrada). Control para habilitar los "latches" internos de Dirección. Esta línea se conecta a la salida del 8085 y le

permite a la RAM reconocer y cargar en sus "latches" internos de dirección el "byte" de menor orden de la dirección que se presenta en el Canal de Dirección-Datos del 8085 durante el estado 11.

CE (Entrada). Habilitar integrado. El 8155 se habilita con un 0 lógico en esta línea.

RD (Entrada). Leer. El nivel 0 en esta entrada, y con el integrado habilitado ($\overline{CE} = 0$), causa que los "buffers" internos AD0-AD7 del 8155 se comporten como salidas. Si la línea $IO/\overline{M} = 1$, el contenido de la localidad de memoria o el puerto de Entrada-Salida seleccionado, respectivamente, por los "latches" internos de dirección se presenta en el Canal de Dirección-Datos del sistema.

WR (Entrada). Escribir. El nivel 0 en esta entrada, y con el integrado habilitado ($\overline{CE} = 0$), causa que los "buffers" internos AD0-AD7 del 8155 se comporten como entradas. Si la línea $IO/\overline{M} = 0$ o $IO/\overline{M} = 1$, el dato presente en el Canal de Datos se carga en la memoria o en el puerto de Entrada-Salida respectivamente.

IO/\overline{M} (Entrada). Si $IO/\overline{M} = 0$ se selecciona a la memoria RAM y si $IO/\overline{M} = 1$ se selecciona a un puerto de Entrada-Salida.

PA0-PA7 (Entrada-Salida). Ocho líneas del puerto A de Entrada-Salida de propósito general. La dirección de las líneas se selecciona programando el Registro de Comando.

PB0-PB7 (Entrada-Salida). Ocho líneas del puerto B tienen las mismas características del puerto A.

PC0-PC5 (Entrada-Salida). Seis líneas del puerto C pueden funcionar como puerto de entrada, como puerto de salida, o como líneas de protocolo o de control de los puertos A y B. La función se selecciona programando el Registro de Comando. La tabla 2.1 muestra las funciones de las líneas del puerto C cuando se programa como señales de control.

TIMER IN (Entrada). Entrada al "Timer".

PATA	ALT 1	ALT 2	ALT 3	ALT 4
PC0	A	**	A INTR (Int. del Puerto A)	A INTR (Int. del Puerto A)
PC1	A	**	A BF (Buffer del Pto. A lleno)	A BF (Buffer del Pto. A lleno)
PC2	A	**	A STB (Strobe del Pto. A)	A STB (Strobe del Pto. A)
PC3	B	**	**	B INTR (Int. del Puerto B)
PC4	B	**	**	B BF (Buffer del Pto. B lleno)
PC5	B	**	**	B STB (Strobe del Pto. B)

* = Puerto de Entrada ** = Puerto de Salida

Tabla 2.1 Posibles configuraciones del Puerto C.

TIMER OUT (Salida). Es una salida del "timer" que puede ser una onda cuadrada o un pulso dependiendo de el modo del "timer".

RESET (Entrada). Se conecta a la salida RESET OUT del 8085. Un nivel lógico en esta línea "limpia" al integrado y programa a los tres puertos de Entrada-Salida en el modo de entrada. El ancho del pulso en RESET debe ser de dos ciclos de reloj del 8085.

Vcc y Vss (Entrada). Entrada de alimentación de + 5 Volts y referencia a tierra respectivamente.

-Controlador Teclado-Display 8279.-

El Circuito Integrado 8279 sirve como interfaz entre el 8085 y el teclado y los "displays" del SDR-85. El 8279 actualiza a los "displays" cuando detecta que se está introduciendo instrucciones o datos por el teclado. Este integrado se describirá en el siguiente capítulo.

-Circuitería Adicional para Ampliación del SDR-85.-

El SDR-85 cuenta con un grupo de circuitos que sirven para conectar otros dispositivos, adicionales al "kit" original. Este grupo, figura 2.1, cuenta con los siguientes circuitos

integrados que demultiplexan el canal de datos/dirección para
baja y además amplifican la corriente de las señales de los
canales de control, datos y dirección :

-Integrado 8212.-

El 8212 consiste en ocho circuitos "latch" con salidas de
tres estados y a los cuales se conectan las líneas del Canal de
Dirección, figura 2.15.

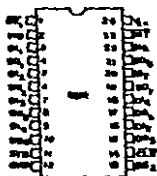


Fig. 2.15 Configuración del 8212.

-Integrado 8216.-

El 8216 tiene cuatro "buffers" bidireccionales con salidas
de tres estados, a los cuales se les conectan las líneas del
Canal de Datos y las del Canal de Control, figura 2.16.

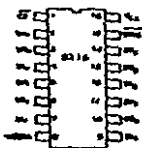


Fig. 2.16 Configuración del 8216.

2.2 DIRECCIONAMIENTO DE MEMORIA.

Cada memoria en el sistema básico del SDK-85, figura 2.1, es habilitada por una señal proveniente del decodificador 8205 o del 74LS138. La tabla 2.2 muestra las señales de habilitación de cada memoria, así como el espacio de dirección que abarca cada una.

SALIDA	RANGO DE DIRECCION ACTIVA	DISPOSITIVO SELECCIONADO
$\overline{CS0}$	0000-07FF	8755/8355 ROM MONITOR (A14)
$\overline{CS1}$	0800-0FFF	8755/8355 ROM DE EXPANSION (A15)
$\overline{CS2}$	1000-17FF	N/C
$\overline{CS3}$	1800-1FFF	8278 CONTROLADOR TECLADO/DISPLAY (A12)
$\overline{CS4}$	2000-27FF	8155 RAM (A16)
$\overline{CS5}$	2800-2FFF	8155 RAM DE EXPANSION (A17)
$\overline{CS6}$	3000-37FF	N/C
$\overline{CS7}$	3800-3FFF	N/C

AXX = No. de Circuito Integrado en el Diagrama de la Fig. 3.11 a)
 N/C = no conectado

Tabla 2.2 Señales de Habilitación de Memoria.

Como se puede observar en la figura 2.12, el decodificador selecciona a los circuitos integrados con las líneas A11-A15, lo que implica que cada espacio de dirección es de 2,048 "bytes" A0-A10, tabla 2.2.

La figura 2.17 muestra como están aprovechados estas 2,048 direcciones en el SDK-85. Como se puede observar las memorias 8355 y 8755 utilizan las 2,048 localidades de memoria porque son memorias de 2,048 "bytes". Para las memorias RAM 8155 solo se utilizan 256 localidades de memoria quedando desaprovechadas las localidades restantes. Las áreas de la figura 2.17 marcadas como "ABIERTO" están libres para expansión.

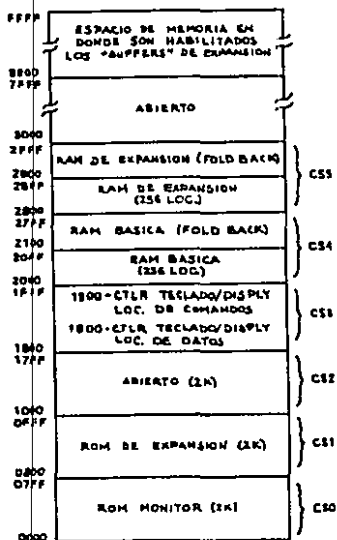


Fig. 2.17 Mapeo de la memoria del SDK-85.

2.3 DIRECCIONAMIENTO DE PUERTOS DE ENTRADA-SALIDA.

Como se mencionó anteriormente, el SDK-85 cuenta con memorias 8155 y 8355/8755 que tienen puertos de Entrada-Salida en paralelo.

PUERTO	FUNCION
00	PUERTO A del ROM Monitor
01	PUERTO B del ROM Monitor
02	Registro de Dirección de Datos (DDR) del PUERTO A del ROM Monitor
03	Registro de Dirección de Datos (DDR) del PUERTO B del ROM Monitor
08	PUERTO A del ROM de Expansión
09	PUERTO B del ROM de Expansión
0A	Registro de Dirección de Datos (DDR) del PUERTO A del ROM de Expansión
0B	Registro de Dirección de Datos (DDR) del PUERTO B del ROM de Expansión
20	Registro COMANDO/ESTADOS de la RAM BASICA
21	PUERTO A de la RAM BASICA
22	PUERTO B de la RAM BASICA
23	PUERTO C de la RAM BASICA
24	Byte de Menor Orden del Conteo del Timer de la RAM BASICA
25	Byte de Mayor Orden del Conteo del Timer de la RAM BASICA
28	Registro COMANDO/ESTADOS de la RAM de Expansión
29	PUERTO A de la RAM de Expansión
2A	PUERTO B de la RAM de Expansión
2B	PUERTO C de la RAM de Expansión
2C	Byte de Menor Orden del Conteo del Timer de la RAM de Expansión
2D	Byte de Mayor Orden del Conteo del Timer de la RAM de Expansión

Tabla 2.3 Direccionamiento de los Puertos de Entrada-Salida.

El decodificador B205 además de seleccionar a un integrado cuando el B085 genera una dirección de memoria, debe seleccionar al

integrado cuando el 8085 genere un código de selección de un puerto de Entrada-Salida (con las instrucciones "IN" y "OUT"). La tabla 2.3 contiene todas las direcciones de los puertos de Entrada-Salida del SCI-85. Durante una instrucción de Entrada-Salida, el 8085 envía los ocho "bits" del código de selección por las líneas AD0-AD7 y las repite por las líneas AB-A15. Por ejemplo, si se ejecuta una instrucción OUT 03, el 8085 envía por el Canal de Dirección lo siguiente:

A15	AD	A7	A0
0 0 0 0	0 0 1 1	0 0 0 0	0 0 1 1
03H		03H	

-Sección de Entrada-Salida del 8055/8755.-

Como ya se menciona el 8055 tiene dos Registros de Dirección de Datos (DDR), que sirven para especificar la función (Entrada o Salida), de cada pata terminal de los dos puertos.

La figura 2.18 muestra como se encuentran interconectados el "bit" 0 del puerto A y el "bit" 0 del registro DDR A. Esta interconexión la tienen los "bits" restantes del puerto A con los "bits" del registro DDR A y por otro lado los del puerto B con los del registro DDR B. Aquí se puede observar que un nivel 0 en este "bit" del registro indica que la línea correspondiente, para el caso PA0, del puerto está programada en el "modo de entrada". Un "bit" con valor 1 en el registro DDR indica que la línea correspondiente en el puerto está programada en el "modo de salida".

La tabla 2.4 muestra la asignación de los puertos y registros DDR en base al valor cargado en los "latches" internos AD0 y AD1.

Es importante tener en cuenta que el 8055 no puede leer el contenido de los registros DDR, sólo escribir en ellos.

El contenido de un puerto de Entrada-Salida se puede leer cuando se cumplen las condiciones siguientes: el integrado está habilitado ($\overline{CE} = 1$ y $\overline{CE} = 0$), $\overline{IO/\overline{M}} = 1$ y $\overline{OE} = 0$. También se puede leer cuando el integrado está habilitado y la línea \overline{IOR} se activa (nivel 0), sin importar el valor de la línea $\overline{IO/\overline{M}}$.

Cuando el integrado está habilitado y la línea \overline{IOW} se activa, el dato presente en el Canal AD0-AD7 se carga en el puerto de Entrada-Salida o el registro DDR seleccionado. Durante esta operación todos los "bits", del puerto o registro seleccionado, se alteran no importando el modo (Entrada-Salida) del puerto y del estado de la línea $\overline{IO/\overline{M}}$.

Cuando se activa la línea RESET, las salidas Q de los "latches" del registro DDR, toman valor 0, desactivando a los "buffers" de salida. Esto equivale a poner al puerto en el "modo de entrada".

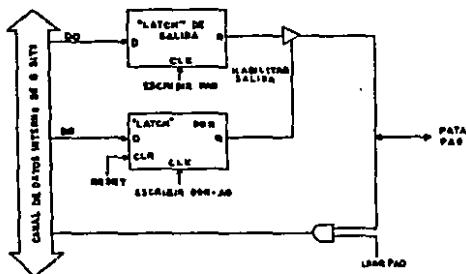


Figura 2.18 Configuración de un "bit" del Puerto A y DDRA.

AD1	AD0	Selección
0	0	Puerto A
0	1	Puerto B
1	0	DDR del puerto A - DDRA
1	1	DDR del puerto B - DDRB

Tabla 2.4 Asignación de los puertos y registros DDR.

-Sección de Entrada-Salida del BISS,-

La sección de Entrada-Salida del BISS cuenta con tres puertos de Entrada-Salida (dos de ocho "bits" y uno de seis

"bits"), un registro de Comandos, un registro de Estados y un "Timer" de 14 "bits". La operación de los puertos y el "Timer" se programa o configura por medio del registro de Comandos, mientras que el estado de los puertos y el "Timer" se pueden conocer leyendo el registro de Estados.

Durante una transferencia de Entrada-Salida, el 8155 utiliza el valor de las líneas AD0-AD2 para seleccionar a los puertos y a los registros. La tabla 2.5 muestra la relación de los valores de las líneas AD0-AD2 con los puertos y registros.

AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	Puerto o Registro
X	X	X	X	X	0	0	0	Registros de Comandos/Estados
X	X	X	X	X	0	0	1	Puerto A
X	X	X	X	X	0	1	0	Puerto B
X	X	X	X	X	0	1	1	Puerto C o de Control
X	X	X	X	X	1	0	0	8 "bits" bajos (LSB) del Timer
X	X	X	X	X	1	0	1	6 "bits" altos (MSB) del Timer

Tabla 2.5 Direcciones de los puertos y registros del 8155.

a) Registro de Comandos.-

El registro de Comandos consiste en ocho "latches" que controlan la operación de los tres puertos de Entrada-Salida y el "Timer", figura 2.19. El registro de Comandos se puede cargar en cualquier momento con una instrucción OUT, pero no se puede leer el contenido del mismo.

Los "bits" 0 y 1 de este registro definen el modo de los puertos A y B respectivamente, si el "bit" tiene valor 0 se define como puerto de entrada y si tiene valor 1 se define como puerto de salida.

Los "bits" 2 y 3 seleccionan una de las cuatro posibles alternativas de Entrada-Salida del puerto C, tabla 2.1.

Los "bits" 4 y 5 se utilizan para habilitar (1) y deshabilitar (0) las solicitudes de interrupción por el puerto A o el puerto B respectivamente.

Los "bits" 6 y 7 se utilizan para darle comandos al "Timer".

El valor 00 no afecta la operación del "Timer". El valor 01 detiene el conteo si el "Timer" está corriendo, no afectando la operación si no lo está. El valor 10 ordena detener el conteo cuando el "Timer" termine el conteo en proceso. Con el valor 11 existen dos posibilidades:

- 1.- Si el "Timer" no está corriendo, carga al "Timer" el valor y el modo, y arranca inmediatamente,
- 2.- Si el "Timer" está corriendo, termina el conteo en proceso, e inicia un nuevo conteo y el nuevo modo.

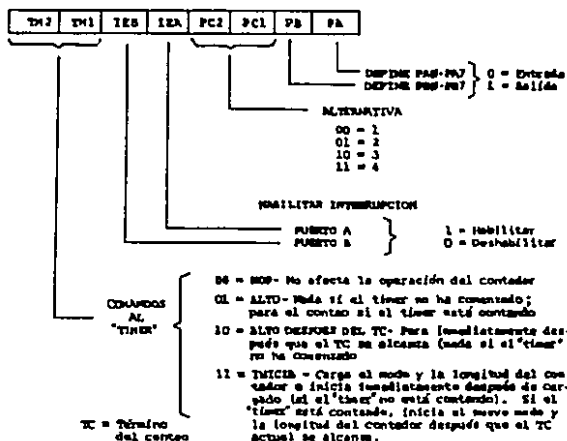


Fig. 2.19 Asignación de los "bits" del Registro de Comandos.

b) Registro de Estados.-

Este registro consiste en siete "latches" que almacenan información acerca del estado de los puertos A, B y del "Timer". El 8085 puede leer el contenido de este registro pero no puede escribir en él. La figura 2.20 muestra la asignación de los "bits" en el Registro de Estados.

El "bit" 0 toma nivel 1 cuando el puerto A tiene un dato para el exterior (el puerto A está programado como salida) o recibió un dato del exterior para el CPU (el puerto A está programado como entrada). El "bit" 0 toma nivel 0 cuando el 8155 recibe un pulso negativo en PC2 del puerto C cuando éste está programado en los modos ALT3 o ALT4.

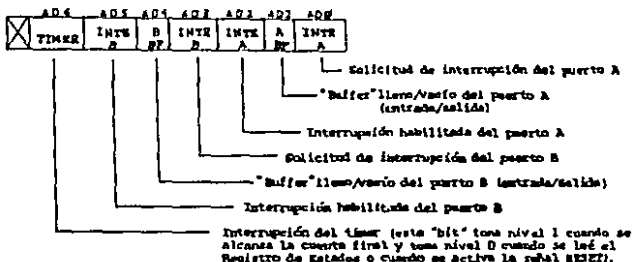


Fig. 2.20 Asignación de los "bits" del Registro de Estados.

El "bit" 1 indica si el contenido del puerto A ya se leyó por el microprocesador (puerto A como entrada) o por el exterior (puerto A como salida). Cuando el puerto A se carga, el "bit" 1 toma el nivel 1 y cuando se lee toma el nivel 0.

El "bit" 2 indica si el puerto A está habilitado (nivel 1) o no (nivel 0) para efectuar una solicitud de interrupción en el momento en que se carga con un dato.

Los "bits" 3, 4 y 5 indican las mismas funciones que los "bits" 0, 1 y 2 respectivamente, pero para el puerto B.

El "bit" 6 toma nivel 1 cuando el Contador termina su cuenta y toma nivel 0 en el momento de leer el Registro de Estados o cuando se activa la entrada RESET del 8155.

CAPITULO 3.

DESARROLLO DEL CIRCUITO DE VIDEO Y DE TECLADO.

El propósito de este capítulo es describir el Circuito de Video, su conexión con el SDK-85 y la conexión entre el teclado y el SDK-85.

El circuito de video que se propone está basado en el circuito integrado 8275 que es un Controlador Programable para el Tubo de Rayos Catódicos. Por otro lado, la conexión del teclado al SDK-85 se realizará con un circuito basado en el 8279 que es una Interface Programable Teclado/"Displays".

3.1 DESCRIPCION DEL CIRCUITO DE VIDEO.

Con el objeto de entender mejor el diseño del circuito de video, es necesario comentar los principios de operación de un monitor de Tubo de Rayos Catódicos. El Tubo de Rayos Catódicos figura 3.1, es de una estructura de vidrio al vacío que tiene un recubrimiento de fósforo sobre la superficie interna de la región frontal rectangular (pantalla). Un filamento contenido en la parte conocida como cuello del Tubo de Rayos Catódicos que excita, por medio de calor al cátodo para que ceda electrones. Una fuente de alto voltaje aplicado entre el cátodo y el electrodo del cinescopio (ánodo), acelera a los electrones que van hacia la superficie interna de la pantalla. Los electrones en el momento en que golpean a la superficie fluorescente producen luz. Para controlar la posición del rayo de electrones que golpea a la pantalla son utilizadas dos técnicas, la deflexión electromagnética y la deflexión electrostática.

Las imágenes desplegadas en el monitor son realizadas por el movimiento del rayo de electrones de tal forma, que se genere una serie de líneas a lo largo de la pantalla. Este movimiento se inicia en la esquina superior izquierda y simultáneamente se mueve de izquierda a derecha y de arriba a bajo formando líneas en zig-zag, figura 3.2.

Dos circuitos independientes de control operan simultáneamente para generar el movimiento horizontal y vertical del rayo y un tercer circuito controla el flujo de corriente en el rayo, que al variar provoca que la imagen pueda hacerse brillante u opaca según se desea.

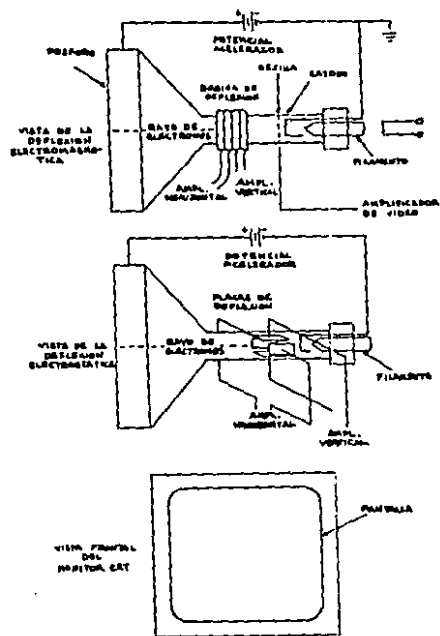


Fig. 3.1 Tubo de Rayos Catódicos.

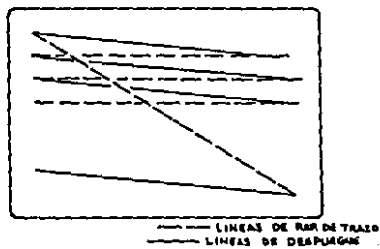


Fig. 3.2 Barrido en el Tubo de Rayos Catódicos.

Cuando el rayo llega hasta el fin de una línea, es regresado al principio de la próxima línea a una velocidad mayor que a la usada para generar la línea. Esta acción es conocida como "repetición de trazo" y durante este periodo no se disparan electrones.

Cuando el rayo llega finalmente hasta la esquina inferior derecha de la pantalla se "repite el trazo" verticalmente, regresando el rayo a la esquina superior izquierda. El tiempo que le toma al rayo moverse desde la parte superior de la pantalla a la parte inferior y regresar a la parte superior es conocido como un "cuadro". En los Estados Unidos, la televisión comercial ha difundido el uso de 15,750 Hz como frecuencia de barrido horizontal (63.5 microsegundos por cada línea horizontal) y 60 Hz como frecuencia de barrido vertical o de "cuadro" (16.67 milisegundos por cuadro vertical). La frecuencia horizontal dentro del estándar mencionado tiene una tolerancia de +/- 500 Hz, en otras palabras, pueden ser desplegadas de 256 a 270 líneas por cuadro.

Como ya se mencionó, los caracteres e imágenes que son vistos en la pantalla son formados por series de puntos que son enviados por el controlador de video al mismo tiempo de que el rayo de electrones se mueve a lo largo de la pantalla. Hay dos circuitos de tiempo que forman los pulsos de tiempo necesarios y son el "reloj de punto" y el "reloj de caracter".

El número de líneas por cuadro puede ser determinado con la aplicación de la siguiente ecuación:

$$L = (H * Z) + V$$

Donde: H es el número de líneas horizontales por caracter,
Z es el número de líneas de caracter por cuadro y
V es el número de líneas horizontales durante la "repetición de trazo" vertical.

En este diseño, una matriz de punto de 7 x 10 es empleada para formar a los caracteres (fig. 3.3), así que H = 10. Veinticinco renglones de caracter serán desplegados por cuadro, así que Z = 25. Y por último dejaremos un margen superior e inferior de 10 líneas cada uno para que se realice la "repetición de trazo" vertical, en otras palabras V = 20. Haciendo operaciones obtenemos que L = 270 líneas por cuadro, lo que implica una frecuencia horizontal de 16.20 kHz en un cuadro de 60 Hz.

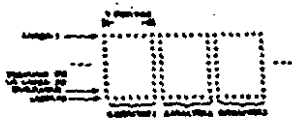


Fig. 3.3 Formato del Renglón de Carácter.

El reloj de carácter es igual al reloj de punto dividido entre el número de puntos usados para formar un carácter a lo largo del eje horizontal y el reloj de punto es calculado de acuerdo a la siguiente ecuación:

$$\text{Reloj de Punto (Hz)} = (N + R) * D * L * F$$

Donde: N es el número de caracteres desplegados por renglón,
 R es el número de caracteres aumentados para el tiempo de la "repetición de trazo" horizontal,
 D es el número de puntos por carácter,
 L es el número de líneas horizontales por cuadro y
 F es la velocidad por cuadro en Hz.

Para este diseño N = 80, R = 20, D = 7, L = 270 y F = 60. Sustituyendo los valores en la fórmula nos queda que se requiere un reloj de punto de 11.74 MHz.

El valor de la "repetición de trazo" horizontal N, puede cambiar de sistema a sistema porque además de su función principal es usado para establecer márgenes izquierdo y derecho en la pantalla.

-Controlador 8275.-

La figura 3.4 muestra el diagrama de bloques y la configuración de las patas terminales del Controlador Programable para el Tubo de Rayos Catódicos 8275. Su función principal es refrescar lo desplegado en la pantalla, almacena información internamente y además mantiene la pista de la posición de la información desplegada.

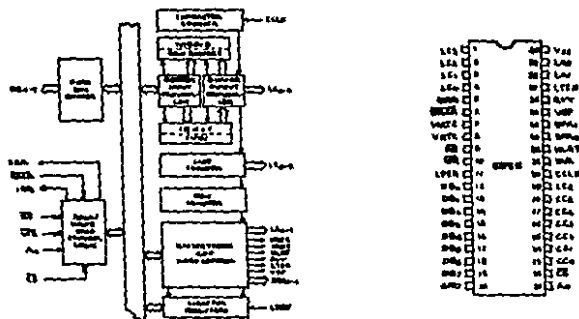


Fig. 3.4 Configuración de las patas terminales y diagrama de bloques del 8275.

A continuación se dará una descripción funcional de las patas terminales del 8275:

LCO-LC3 (Salidas). Conteo de Línea. Salidas del contador de línea que son usadas para direccionar al generador de carácter externo que indican la posición de la línea que está en la pantalla.

DRQ (Salida). Requerimiento de Acceso Directo a Memoria. Señal de salida que va (en nuestro caso), al Controlador de Acceso Directo a Memoria 8257 solicitándole un ciclo de acceso directo a la memoria.

DACT (Entrada). Reconocimiento de una Solicitud de Acceso Directo a Memoria. Esta línea toma nivel cero cuando el Controlador 8257 ha otorgado un ciclo de acceso directo a memoria.

HRTC (Salida). Repetición del Trazo Horizontal. Esta salida toma nivel uno, durante el intervalo programado de la Repetición de Trazo Horizontal. Dentro de este periodo la salida VSF está en uno lógico y la salida L7EN en cero.

VRTC (Salida). Repetición de Trazo Vertical. Esta línea toma nivel uno, durante el intervalo programado de la Repetición de Trazo Vertical. Dentro de este período la salida VSP está en un lógico y la salida LTEN en cero.

RD (Entrada). Leer. Señal de control para leer registros.

WR (Entrada). Escribir. Señal de control que se utiliza para escribir comandos en el registro de control o para escribir datos en los "buffers" de renglón durante un ciclo de acceso directo a memoria.

LPEN (Entrada). Pluma de Luz. Esta línea se activa cuando la señal de la pluma de luz ha sido detectada.

DBO-DD7 (Entradas-Salidas). Líneas del Canal de Datos Bidireccionales de Tres Estados. Estas líneas son habilitadas durante una operación de escritura o lectura.

LA0-LA1 (Salidas). Códigos de Atributo de Línea. Estos códigos de atributo tienen que ser decodificados externamente por la lógica de tiempo de punto para generar las combinaciones de línea horizontal y vertical para los despliegues gráficos especificados por los códigos de atributo de carácter.

LTEN (Salida). Habilitación de Luz. Esta línea de salida es usada para habilitar la señal de video para el Tubo de Rayos Catódicos. Esta salida está activa por la posición programada del cursor y por la posición especificada por los códigos de atributo.

RVV (Salida). Video Invertido. Esta línea sirve para indicarle al circuito de Tubo de Rayos Catódicos que invierta la señal de video. Esta línea está activa por la posición del cursor, si es programado un cursor de bloque de video invertido o por la posición especificada por los Códigos de Atributo de Campo.

VSP (Salida). Supresión de Video. Esta línea es usada para interrumpir la señal de video. Esta señal está activa:

- Durante los intervalos de Repetición de Trazo Horizontal y Vertical.
- En las líneas superior e inferior del renglón si el bajo-línea es programada para que sea la línea número ocho o mayor.
- Cuando es detectado un código de fin de renglón o fin de pantalla.
- Cuando se está efectuando un Acceso Directo a Memoria.
- En intervalos regulares para crear centelleo de acuerdo a lo especificado por la programación del cursor, atributo de carácter o atributo de campo.

GPA0-GPA1 (Salidas). Códigos de atributo de Propósito General. Salidas que son habilitadas por los códigos de atributo de campo de propósito general.

HLGT (Salida). Aumento de Luz. Señal usada para intensificar lo desplegado en una posición en particular en la pantalla, según lo especificado por los códigos de atributo de carácter o códigos de atributo de campo.

IRQ (Salida). Requerimiento de Interrupción. Con esta línea se hace un requerimiento para entrar en un ciclo de interrupción.

CCLK (Entrada). Reloj de Carácter. Línea que requiere de una señal de reloj de la sección de tiempo de punto.

CC0-CC6 (Salidas). Códigos de Carácter. Salidas de los "buffers" de renglón usadas para seleccionar al carácter en el generador de caracteres externo.

CS (Entrada). Selección del Integrado. Un cero lógico en esta entrada habilita al 8275, para poder escribir un comando o leer un estado.

AO (Entrada). Direccionamiento de Puerto. Un uno lógico en esta línea se selecciona el puerto "C" o registros de comando y un cero lógico se selecciona al puerto "F" o registros de parámetro.

Vcc. + 5 Volts.

GND. Referencia a tierra.

El 8275 proporciona las siguientes características generales:

1.- Refrescamiento de lo desplegado en el Tubo de Rayos Catódicos.- El 8275 ha sido programado por el diseñador para un formato específico de pantalla, generando una serie de señales de requerimiento de Acceso Directo a Memoria, dando como resultado la transferencia de un renglón de caracteres desde la memoria de despliegue, por medio del Controlador de Acceso Directo a Memoria 8257, a los "buffers" de renglón del 8275. El 8275 presenta los códigos de carácter a un generador de carácter externo ROM o EPROM usando las salidas CC0-CC6. La lógica externa del tiempo de punto es utilizada posteriormente para transferir la salida de datos en paralelo del generador de carácter externo, en forma serial a la entrada de video del Tubo de Rayos Catódicos. Los renglones de carácter son desplegados en el Tubo de Rayos Catódicos una línea a la vez. Las salidas del conteo de línea

LC0-LC3 son utilizadas por el generador de caracteres externo para realizar la función de selección de la línea del carácter a desplegar. El proceso de despliegue en la pantalla se ilustra en la figura 3.5. El proceso completo es repetido para cada renglón desplegado. A el principio del último renglón desplegado, el 8275 emite una solicitud de interrupción por la activación de la línea de salida IRQ. Y la salida de interrupción del 8275 es conectada a la entrada de interrupción del procesador central del sistema.

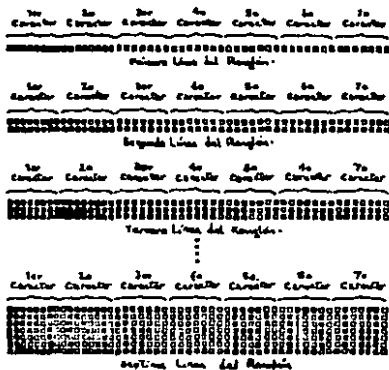


Fig. 3.5 Despliegue en la pantalla de un Renglón.

La interrupción da lugar a que el microprocesador ejecute una subrutina de servicio de interrupción. La subrutina de servicio reinicializa los parámetros del Controlador de Acceso Directo a Memoria para el próximo ciclo de refrescamiento de lo desplegado, encuesta al sistema controlador del teclado, y/o ejecuta otras funciones apropiadas. Un refrescamiento adecuado del Tubo de Rayos Catódicos requiere que los parámetros del 8275 sean programados antes de iniciar la operación de despliegue. El 8275 tiene dos tipos de registros programables, el registro de comandos (CREG) y el registro de parámetros (PREG). Así como también del registro de estados (SREG). Los registros de comando solo pueden ser escritos y los registros de estados solo

pueden ser leídos. El 8275 espera recibir un comando seguido por una secuencia de 0 a 4 parámetros, dependiendo del comando. El juego de instrucciones del 8275 consiste en 8 comandos que se muestran en la tabla 3.1.

COMANDO	NÚM. DE PARÁMETROS (BYTES)	NOTAS
RESET	0	Parámetros del formato de datos
START DISPLAY	0	--
STOP DISPLAY	0	--
READ LIGHT PEN	2	--
LOAD CURSOR	2	Parámetros X,Y de la posición del cursor
ENABLE INTERRUPT	0	--
DISABLE INTERRUPT	0	--
PRESET COUNTERS	0	Setea todos los contadores inter- nos

Tabla 3.1 Juego de Instrucciones del 8275.

Para poder establecer el formato a desplegar, el 8275 proporciona al usuario un número de parámetros programables del formato a desplegar. Los formatos a desplegar tienen de 1 a 80 caracteres por renglón, de 1 a 64 renglones por cuadro y de 1 a 16 líneas horizontales por renglón.

Además de transferir caracteres desde la memoria a la pantalla del Tubo de Rayos Catódicos, el 8275 le da importancia al control de la posición del cursor. La posición del cursor pueda ser programada, por los registros de posición del cursor X y Y, para cualquier posición de carácter en la pantalla. Se puede seleccionar una de las cuatro formas diferentes de desplegar al cursor y estas son: bajo-línea parpadeante o no parpadeante y ce bloque de video invertido parpadeante o no parpadeante.

2.- Tiempos para el Tubo de Rayos Catódicos.- El 8275 tiene dos salidas de tiempo, HRTC Y VRTC, las cuales son

utilizadas en sincronización con los osciladores horizontal y vertical del Tubo de Rayos Catódicos para el ciclo de refrescamiento del 8275. Siempre que son activados HRTC o VRTC, una tercera salida de tiempo, VSP (Supresión de Video) es activada, proporcionando una señal "en blanco" a la lógica de tiempo de punto. La lógica de tiempo de punto normalmente inhibirá la salida de video al Tubo de Rayos Catódicos, durante el tiempo en que permanece activada la señal de supresión de video. Una salida adicional de tiempo, LTEN (Habilitación de Luz), es usada para proporcionar a la fuerza, la habilitación de la salida de video sin hacer caso de el estado de VSP. Esta característica es usada por el 8275 para poner el cursor en la pantalla y para funciones de control de atributo.

La salida HGLT (Aumento de Luz) permite a una función de atributo incrementar la intensidad del rayo en el Tubo de Rayos Catódicos a un nivel mayor al normal. La quinta señal de tiempo RVV (Video Invertido) provocará, cuando este activa, que la salida de video del sistema se invierta.

3.- Funciones Especiales.-

a) Atributos Visuales.- Los atributos visuales son códigos especiales que cuando son recuperados desde la memoria por el 8275, afectan las características visuales de un carácter o campo de caracteres. Existen dos tipos de atributos visuales, atributos de carácter y atributos de campo.

Códigos de Atributo de Carácter: Los códigos de atributo de carácter pueden ser usados para generar símbolos gráficos sin utilizar un generador de caracteres. Esto se logra mediante la activación selectiva de las salidas de Atributo de Línea (LAO-LAL), la salida de Supresión de Video (VSP) y la salida de Habilitación de Luz (LTEN). La lógica de tiempo de punto usa estas señales para generar los símbolos adecuados. Los atributos de carácter pueden ser programados individualmente para que parpaden o para que sean aumentados en intensidad de luz. El parpadeo es logrado con la activación de la salida de Supresión de Video (VSP). La frecuencia de parpadeo es igual a la frecuencia de refrescamiento de la pantalla dividida entre 32. El aumento en la intensidad de luz es realizado por la activación de la salida Aumento de Luz (HGLT). Los atributos de carácter fueron diseñados para producir los símbolos gráficos mostrados en la tabla 3.2.

Códigos de Atributo de Campo: Los atributos de campo son códigos de control los cuales afectan las características visuales de un campo de caracteres, empezando en el carácter siguiente al código de atributo de campo hasta el carácter que precede al próximo código de atributo de campo.

Existen 6 Atributos de Campo y son:

-Parpadeo.- Los caracteres siguientes de este código parpadearán por la activación de la salida de Supresión de Video (VSP). La frecuencia de parpadeo es igual a la frecuencia de refrescamiento de la pantalla dividida entre 32.

-Aumento de Luz.- Los caracteres siguientes de este código son aumentados en su intensidad de luz, por la activación de la salida de Aumento de Luz (HGLT).

-Video Invertido.- Los caracteres siguientes de este código son desplegados invertidos (blanco por negro y negro por blanco), por la activación de la salida Video Invertido (RVV).

-Bajo Línea.- Los caracteres que siguen de este código son subrayados por la activación de la salida de Habilitación de Luz (LTEN).

-Propósito General.- Hay dos salidas adicionales del 8275 que actúan como de propósito general, independientemente de los atributos de campo programados. Estos atributos pueden ser usados para seleccionar colores o ejecutar otras funciones de control deseadas.

El 8275 puede ser programado para proporcionar caracteres de atributo de campo visibles o invisibles como se muestra en la figura 3.6. Si el 8275 es programado en el modo de atributo de campo visible, todos los atributos de campo ocuparán una posición en la pantalla en forma de espacios en blanco, por la activación de la salida de Supresión de Video (VSP). Esto quiere decir que los atributos visuales escogidos son activados después de un carácter en blanco. Si el 8275 es programado en el modo de atributo de campo invisible, son activados los FIFO's del "buffer" de renglón del 8275. Estos dos registros Primero en Entrar-Primero en Salir (FIFO's) para 16 caracteres, son usados para proporcionar un espacio adicional a los "buffers" de renglón. Cuando es proporcionado un atributo de campo en el "buffer" de renglón durante un Acceso Directo a Memoria, el "buffer" controlador de entrada lo reconoce y pone el siguiente carácter en el FIFO adecuado. Durante el despliegue, cuando el "buffer" controlador de salida detecta al atributo de campo, pone inmediatamente un carácter desde el FIFO en las salidas de código de carácter (CCO-CC6) al mismo tiempo que son activados los atributos escogidos.

b) Detección de la Pluma de Luz.- Una pluma de luz consiste fundamentalmente de un interruptor y un sensor de luz. Cuando la pluma de luz es presionada contra la pantalla del Tubo de

Character attributes were designed to produce the following graphics:

CHARACTER ATTRIBUTE CODE "CCCC"	OUTPUTS				SYMBOL	DESCRIPTION
	LA	LAG	VSP	LTEN		
0000	Above Underline	0	0	1	0	Top Left Corner
	Below Underline	1	0	0	0	
0001	Above Underline	0	1	0	0	Top Right Corner
	Below Underline	1	1	0	0	
0012	Above Underline	0	0	0	1	Bottom Left Corner
	Below Underline	1	0	0	0	
0014	Above Underline	0	0	1	0	Bottom Right Corner
	Below Underline	1	1	0	0	
0109	Above Underline	0	0	0	1	Top Intersect
	Below Underline	0	1	0	0	
0101	Above Underline	0	1	0	0	Right Intersect
	Below Underline	1	1	0	0	
0110	Above Underline	0	1	0	0	Left Intersect
	Below Underline	1	0	0	0	
0111	Above Underline	1	1	0	0	Bottom Intersect
	Below Underline	0	0	1	0	
1003	Above Underline	0	0	0	1	Horizontal Line
	Below Underline	0	0	1	0	
1001	Above Underline	0	1	0	0	Vertical Line
	Below Underline	0	1	0	0	
1013	Above Underline	0	1	0	1	Crossed Lines
	Below Underline	0	0	0	1	
1011	Above Underline	0	0	0	0	Not Recommended *
	Below Underline	0	0	0	0	
1100	Above Underline	0	0	1	0	Special Codes
	Below Underline	0	0	1	0	
1101	Underline		Undefined			Illegal
	Below Underline		Undefined			
1110	Underline		Undefined			Illegal
	Below Underline		Undefined			
1111	Underline		Undefined			Illegal
	Below Underline		Undefined			

*Character Attribute Code 1011 is not recommended for normal operation. Since none of the attribute outputs are active, the character generator will not be disabled and an indeterminate character will be generated.

Character Attribute Codes 1101, 1110, and 1111 are illegal. Blinking is active when B = 1. Highlight is active when H = 1.

Tabla 3.2 Atributos de Carácter.

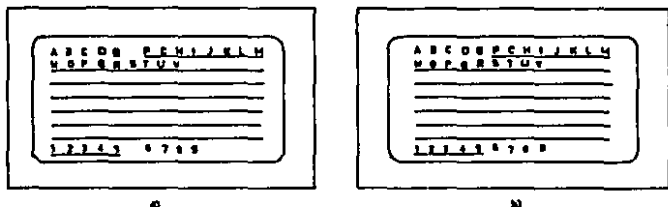


Fig. 3.6 Ejemplos de los Atributos de Campo: a) Visible; b) Invisible.

Rayos Catódicos, el interruptor habilita al sensor de luz. Cuando la línea de barrido coincide con la posición del sensor de luz en la pantalla, la salida de la pluma de luz es recibida y la coordenada de la posición de carácter y renglón es almacenada en dos registros internos del 8275. Estos registros pueden ser leídos por el microprocesador.

c) Códigos Especiales.- Cuatro códigos especiales pueden ser usados para ayudar a reducir el exceso de memoria, "software", o Acceso Directo a Memoria. Estos códigos son puestos en la posición de carácter en la memoria.

1.- Código de Fin de Renglón.- Activa a VSP que permanece activa hasta que es concluido el fin de renglón. El tiempo que VSP esté activa, la pantalla está en blanco.

2.- Código de Fin de Renglón-Parar el Acceso Directo a Memoria.- Este código provoca que la lógica de Control de Acceso Directo a Memoria detenga el acceso para la parte de renglón restante.

3.- Código de Fin de Pantalla.- Activa a VSP que permanece activa hasta que es realizado el fin de pantalla.

4.- Código de Fin de Pantalla-Parar el Acceso Directo a Memoria.- Hace que la lógica de Control de Acceso Directo a Memoria pare el acceso para la parte del cuadro restante, cuando este código es escrito en el "buffer" de renglón.

d) Control Programable de Acceso Directo a Memoria por Bloque.- El 8275 puede ser programado para solicitar una transferencia de un simple "byte" en el Acceso Directo a Memoria o realizar una transferencia de 2, 4 u 8 caracteres por bloque. Cabe aquí señalar que el intervalo entre bloques es también programable.

3.2 CONEXION DEL CIRCUITO DE VIDEO AL MICROCOMPUTADOR.

Una vez descrito las características principales del circuito de video, es necesario establecer el circuito y su conexión con el SDK-85. El diagrama de bloques de la figura 3.7 muestra a los circuitos básicos del circuito de video conectados al microcomputador.

Como se puede observar, en el diagrama de bloques de la figura 3.7 no se están considerando todas las líneas de control para facilitar su entendimiento.

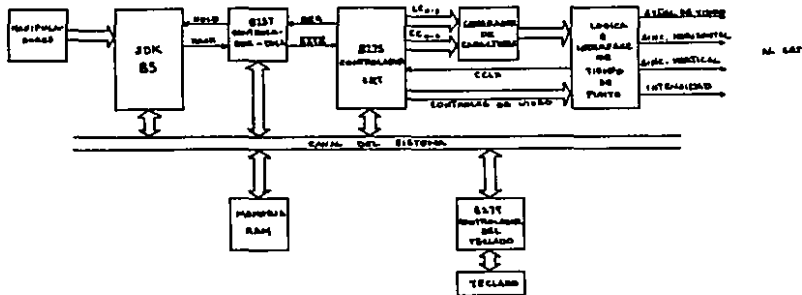


Fig. 3.7 Diagrama de Bloques del Sistema.

El SDK-85 programa en primera instancia al Controlador 8275, de acuerdo a la forma en como se va a desplegar. Posteriormente y de acuerdo a los datos de entrada por medio del manipulador o del teclado, estos se almacenan en el bloque de memoria RAM externo, es entonces que el 8275 genera una señal de requerimiento de Acceso Directo a Memoria obligando al 8257 para que envíe la señal de requerimiento de los Canales del Sistema, una vez liberados, son llenados los "buffers" de renglón del 8275 con los datos (en código ASCII), a desplegar. La dirección de una línea de cada carácter de cada renglón es mandada al generador de caracteres ROM, que a su vez responde con un código en paralelo, que pasa por la lógica e interfaz de tiempo de punto de alta velocidad y junto con los controles de video se genera la señal de video serial, sincronía Horizontal y sincronía Vertical, finalmente estas señales se envían al monitor.

Partiendo de este diagrama y tomando en cuenta que se puede disponer de los Canales de Dirección, Datos y de Control del SDK-85, el circuito total queda como se muestra en el apéndice 2.

3.3 CONEXION DEL TECLADO AL MICROCOMPUTADOR.

Como ya se menciona anteriormente la conexión del teclado al microcomputador se realiza por medio del circuito integrado 8279 que es una interfaz programable de teclado/"display". El microcomputador SDK-85 cuenta con un 8279 un teclado y una sección de "displays", pero para nuestros propósitos haremos unas pequeñas modificaciones ya que no requerimos la sección de "displays" y necesitamos un teclado más completo. Es por esta razón que adicionaremos otro 8279 y un teclado mayor.

El diagrama de bloques y la configuración de las patas terminales del 8279 se muestran en la figura 3.8. Como ya se mencionó que el 8279 será usado como interfaz entre el teclado y el microcomputador, así que sólo las características asociadas con esta función serán descritas en este inciso.

La función principal del 8279 en este sistema comprende el muestreo de 54 teclas del teclado, determinando si una ha sido oprimida y transmite la dirección de la misma al microprocesador. El controlador inicia el proceso de muestreo transmitiendo por las líneas de muestreo SLO-SL2 un dato que puede ser decodificado para seleccionar a un renglón del teclado. Las líneas de salida de las columnas del teclado son conectadas a las entradas RLO-RL7 del 8279. La información de estas líneas es almacenada por el

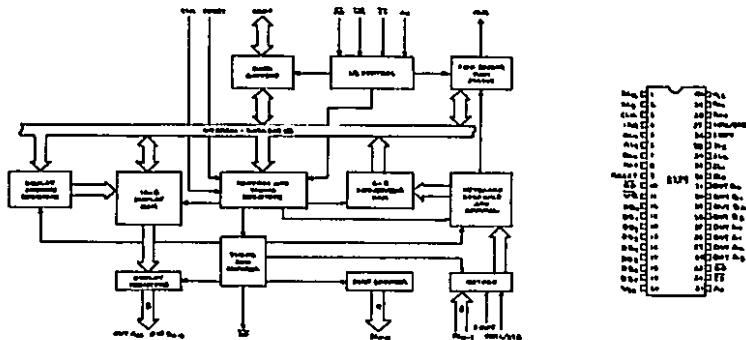


Fig. 3.8 Configuración de las patas terminales / diagrama de bloques del B279.

B279 internamente y son muestreadas con el objeto de encontrar si una tecla fue oprimida en el rengion seleccionado. Si el controlador detecta una digitación, espera aproximadamente 10 ms para determinar si la tecla permanece oprimida. Si esto es así, la dirección de la tecla más el estado de las líneas de SHIFT y CONTROL son transferidos al registro Primero en Entrar Primero en Salir (FIFO) del B279. El formato de datos de este registro se muestra en la figura 3.9. El registro puede almacenar el "byte"

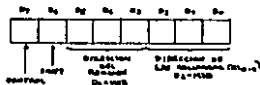


Fig. 3.9 Formato del "byte" de dato del Registro Primero en Entrar Primero en Salir.

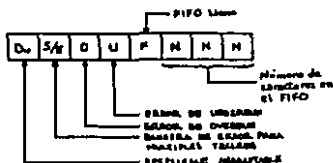


Fig. 3.10 Palabra de Estados del Registro Primero en Entrar Primero en Salir.

de información, esto quiere decir que pueden ocurrir 8 digitaciones antes de que el microprocesador inicie la operación de lectura. El número de caracteres dentro del registro es indicado por el contenido del conteo de carácter dentro de la palabra de estados del FIFO, figura 3.10.

El 8085 puede determinar que ha ocurrido una digitación de dos formas:

- 1) La línea de interrupción del 8279 puede ser conectada a la línea de entrada de interrupción del microprocesador, forzando al 8085 a llamar un servicio de subrutina de interrupción para poder leer el carácter del FIFO.
- 2) Un acceso alternado, en donde periódicamente el 8085 examina al 8279 leyendo la palabra de estados del registro Primero en Entrar Primero en Salir, de tal forma que si el conteo de carácter no es cero, indicando que al menos un carácter está presente en el registro, entonces el 8085 lee el contenido del FIFO.

Este acceso es el que será utilizado en este diseño. Una operación de lectura pondrá el contenido del registro en el Canal de Datos del Sistema y decrementará en uno el conteo de carácter del registro, contenido dentro de la palabra de estados del FIFO.

CAPITULO 4.

DESARROLLO DE LOS MANIPULADORES

A lo largo de la evolución de las computadoras los métodos para comunicarse con ellas han tenido una importante evolución. Una de estas formas que no ha sido la excepción es por medio del manipulador o "joystick" que es un dispositivo electromecánico que tiene como finalidad facilitar la comunicación hombre-maquina.

Unos de los sistemas computarizados de mayor comercialización son los juegos de video, en donde más se ha aprovechado el uso de los manipuladores.

En este capítulo se mencionará en forma general algunos manipuladores y los circuitos necesarios para conectarlos a algún sistema; el manipulador que usaremos en este trabajo y su conexión con el SDK-85.

4.1 MANIPULADORES Y ARREGLO DE CIRCUITOS NECESARIOS.

El uso de los manipuladores en los juegos de video, se inició para darle a este un ambiente adecuado, dependiendo claro está del tipo de juego, del objeto y sus movimientos.

Se dice para darle una ambientación, porque como por ejemplo si se tiene un juego en donde el objeto a mover es un coche, para darle un toque de realismo se podría pensar en un volante, un pedal para acelerar y una palanca de velocidades como manipuladores.

La mayoría de los manipuladores que existen para juegos de video, son de tres tipos:

- 1) Interruptor.
- 2) Resistencia Variable.
- 3) Generador de pulsos.

Cualquier tipo de manipulador tiene como función principal convertir una actividad física en una señal eléctrica. Hasta ahora hemos estado tratando con señales digitales, o sea que solo tienen dos niveles de voltaje o "uno" (+ 5V) o cero (0 V). Pero existen manipuladores que generan señales analógicas, esto quiere decir que la señal varía desde un mínimo hasta un máximo de voltaje o corriente.

Para que el sistema pueda entender estas señales es

necesario contar con un circuito que convierta de señal analógica a digital.

Un voltaje analógico desconocido alimentado en un Convertidor Analógico-Digital, tendrá una señal de salida digital proporcional a la entrada.

Cualquier tipo de convertidor ya sea analógico-digital o digital-analógico, tiene las siguientes características:

a) Precisión.- La precisión es el grado de error, arriba o abajo del valor de salida actual con respecto al valor de salida esperado.

b) Resolución.- Es la fidelidad de detalle en la reproducción de una señal de entrada a una señal de salida, y está determinada por el número de "bits". Esto es semejante a los dibujos de "conexión de punto" que dibujan los niños, entre más puntos haya es mejor la resolución del dibujo.

c) Tiempo de arreglo.- Es el tiempo que necesita el convertidor para fijar el valor correcto antes de empezar con el próximo.

A continuación mencionaremos las cinco técnicas más usadas de conversión analógico-digital:

1) Convertidor A/D Comparador en Paralelo, Simultáneo o "Flash".- El convertidor A/D más simple en concepto y el más rápido es el comparador en paralelo, mostrado en la figura 4.1.

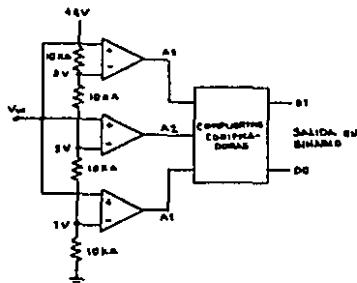


Fig. 4.1 Convertidor A/D Comparador en Paralelo.

La salida de cada comparador irá al valor alto, si la entrada de voltaje en la entrada positiva es mayor que el voltaje de referencia de su entrada negativa. Aquí la señal analógica se aplica a todos los comparadores en paralelo. La tabla 4.1 muestra las diferentes salidas para el circuito de la figura 4.1 para varios voltajes de entrada.

V_m (Volts)	Salidas del Comparador			Salida Binaria	
	A1	A2	A3	D1	D0
0 a 1	0	0	0	0	0
1 a 2	1	0	0	0	1
2 a 3	1	1	0	1	0
3 a 4	1	1	1	1	1

Tabla 4.1 Salidas para varios voltajes de entrada.

Si el voltaje de entrada es menor a 1 volt, los tres comparadores tendrán a la salida un valor bajo. Una entrada entre 1 y 2 volts, provocará que el comparador más bajo tenga un valor de salida alto. Una entrada entre 2 y 3 volts, se dará un valor de salida alto en A1 y A2; y un voltaje superior a 3 volts producirá una salida alta en todos los comparadores.

Los convertidores sólo analizarán un voltaje de entrada con respecto a 4 niveles, por lo que esto es equivalente a dos "bits" binarios de resolución. Para aumentar la resolución del convertidor se pueden usar más comparadores, de tal forma que siete comparadores son necesarios para una resolución de 3 "bits" y 15 comparadores son necesarios para una resolución de 4 "bits". De lo anterior tenemos que para tener N -"bits" de resolución se requieren:

$2^N - 1$ comparadores en un convertidor de este tipo.

El excesivo número de comparadores al intentar de aumentar la resolución es la mayor desventaja de este convertidor. Otra desventaja es que se requiere de una serie de circuitos para convertir la salida de los comparadores a binario.

Pero la mayor ventaja del convertidor A/D comparador en paralelo es su velocidad, porque la salida digital completa será presentada a la salida con un retardo de acuerdo al tiempo de propagación de los comparadores y de las compuertas de codificación. Es por esto que el nombre de "flash" es usado para este tipo de convertidor.

2) Convertidor A/D de Rampa Simple.- La figura 4.2 muestra un circuito de un convertidor A/D que tiene un generador de rampa, un comparador, unos contadores binarios, unos circuitos "latches" y un decodificador.

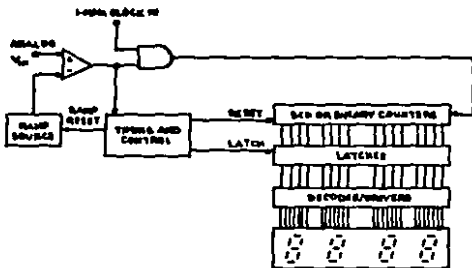


Fig. 4.2 Convertidor A/D de Rampa Simple.

Al principio de un ciclo de conversión, el generador de rampa y los contadores son puestos en cero y se aplica un voltaje a la entrada (+) del comparador. Como esta entrada es más positiva que la entrada invertida del comparador, su salida va a valor alto habilitando a la compuerta "AND" y activando para que empiece la generación de la rampa. Al habilitarse la compuerta "AND", la señal de reloj llega a los contadores iniciando así su función. Y por otro lado el voltaje de rampa va siendo más positivo hasta que excede el voltaje de entrada. Esto provoca que la salida del comparador tome el valor bajo, deshabilitando a la compuerta "AND", interrumpiendo la señal de reloj que va a los contadores. Es entonces que el circuito de control carga el valor del conteo en los "latches" y carga a los contadores y al generador de carga en cero.

Para entender mejor como trabaja este circuito, supongamos una señal de reloj de 1 MHz, cuatro contadores y un voltaje de entrada de 2 volts. También asumiremos que la rampa tiene una pendiente de 1 volt por milisegundo como se muestra en la figura 4.3. Esto implica que la rampa tardará 2 ms para que alcance 2 volts e interrumpir el pulso de reloj, en estos 2 ms, 2,000 pulsos de reloj serán recibidos por los contadores. El circuito de control almacenará el valor final del conteo en los "latches" e iniciará otro ciclo de conversión.

Este tipo de convertidor puede llegar a tener problemas de estabilidad, porque los generadores de rampa pueden ser sensibles a la temperatura, o que presenten variaciones en el tiempo.

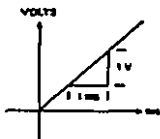


Fig. 4.3 Pendiente de la Rampa.

3) Convertidor A/D de Declive-Doble.- El diagrama de bloques de un circuito convertidor A/D de Declive-Doble se muestra en la figura 4.4. El circuito es muy similar al circuito de Rampa-Simple con la excepción de la utilización de un interruptor a la entrada, para seleccionar un voltaje de entrada o el voltaje de referencia y el circuito adicional a la entrada del comparador.

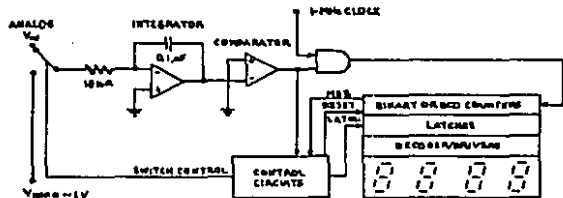


Fig. 4.4 Convertidor A/D de Declive Doble.

La primera etapa de este circuito es un generador de rampa 0, como es algunas veces llamado, integrador. Debido a que la impedancia de entrada del integrador es muy baja, se dice que la entrada inversora es retenida a tierra virtual. Esto quiere decir que la entrada inversora se comporta como un punto de tierra en el sentido de que el voltaje de error es aproximadamente cero. Es importante recordar estas dos ideas de

la tierra virtual:

- 1.- La entrada inversora del amplificador operacional se encuentra (idealmente) a un potencial de tierra.
- 2.- Toda la corriente de entrada (idealmente) circula por la realimentación.

Supongamos, por ejemplo, un voltaje de 2 volts aplicado a la resistencia de 10^7 causando una corriente de 0.2 mA, esta corriente llega hasta uno de los platos del capacitor. Como el capacitor se carga, el voltaje de salida del amp op debe ir más y más negativo para mantener el flujo de corriente constante. Esta salida del integrador se muestra en el de la figura 4.5.

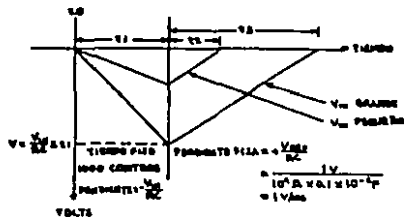


Fig. 4.5 Salida del Integrador Contra Tiempo.

La pendiente de la rampa puede ser fácilmente determinada teniendo que $q = CV$ y $q = It$. Combinando estas dos ecuaciones tenemos que $\Delta V/\Delta t = I/C$. Como la corriente es igual a V_{en}/R , la ecuación anterior nos queda:

$$\Delta V/\Delta t = V_{en}/RC$$

Con esto último podemos observar que para un V_{en} dado la pendiente de salida del integrador será una constante. Tomando los valores del ejemplo y con una entrada de + 2 Volts, la pendiente de salida será - 2 V/ms.

Ahora bien un ciclo de conversión comienza con la rampa en cero, los contadores en cero también y el interruptor conectado al voltaje de entrada. Un voltaje positivo a la entrada provoca a la salida del integrador una rampa negativa. Cuando el integrador alimenta la entrada invertida del comparador, la salida de éste va a positivo, dejando pasar la señal de reloj a través de la compuerta "AND". La salida del integrador está

disponible en rampa negativa para un número fijo de conteos. Esto sucede para dos entradas de voltaje diferentes en t_1 de la figura 4.5.

Cuando los contadores alcanzan el conteo fijo, los circuitos de control ponen en cero a los contadores y selecciona al interruptor de entrada del integrador hacia el voltaje negativo de referencia. Un voltaje de entrada negativo causará que la salida del integrador se tenga una rampa positiva como se muestra en t_2 de la figura 4.5. Cuando la salida del integrador llega justamente arriba de cero, otra vez la salida del comparador va a bajo, deteniendo el paso de la señal de reloj a los contadores. El circuito de control por su parte detecta esta transición y almacena el contenido de los contadores en los "latches". Entonces el circuito de control pone en ceros a los contadores y selecciona al interruptor de entrada del integrador al voltaje de entrada para empezar otro ciclo de conversión. El número de conteos almacenado en los "latches" será proporcional al voltaje de entrada. La salida del integrador en un tiempo fijo t_1 , produce una rampa de voltaje V igual a $(V_{in}/RC)t_1$. Para regresar a cero el integrador deberá producir una rampa hacia arriba la misma cantidad de voltaje durante t_2 , quedando que V es igual a $(V_{ref}/RC)t_2$. Igualando las dos ecuaciones nos queda:

$$\frac{V_{in}}{RC} \cdot t_1 = \frac{V_{ref}}{RC} \cdot t_2 \qquad t_2 = V_{in} \cdot \frac{t_1}{V_{ref}}$$

Las ventajas del convertidor A/D de Declive-Doble son su exactitud, su bajo costo y su inmunidad a la temperatura, y su desventaja es su baja velocidad.

4) Convertidor A/D usando Contadores y Convertidores D/A.- En esta parte vamos a mencionar dos tipos de convertidores:

-Tipo Contador Simple.- La figura 4.6 muestra como un convertidor D/A y unos contadores pueden ser usados en lugar de un generador de rampa para formar un convertidor A/D de Rampa Simple. Al principio del ciclo de conversión, los contadores son puestos en cero por lo que la salida del convertidor D/A es Cero. Cuando un voltaje es aplicado a la entrada del comparador, su salida toma el valor alto y la señal de reloj va hacia los contadores. Cada pulso de reloj avanza a los contadores un conteo e incrementa el valor del voltaje de salida en forma proporcional. Cuando la salida del convertidor D/A empieza a ser mayor que el voltaje de entrada, la salida del comparador toma el valor bajo, deteniéndose el paso de los pulsos de reloj a los contadores. Entonces los circuitos de control almacenan el contenido de los contadores y posteriormente los ponen en cero para iniciar otro ciclo de conversión.

Este método es más rápido que el de Declive Doble pero tiene

la desventaja que requiere de un convertidor D/A muy preciso. Otra desventaja es de que cada ciclo de conversión, el conteo tiene que empezar en cero y contar ascendentemente hasta que la salida del convertidor D/A iguale al V_{in} .

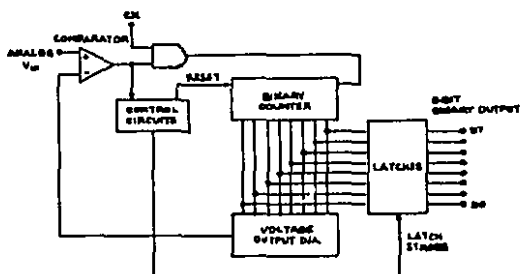


Fig. 4.6 Convertidor A/D de Rampa Simple usando un Convertidor D/A y Contadores Ascendentes.

-Convertidor A/D de Seguimiento.- El convertidor A/D de Seguimiento es un método un poco más rápido para digitalizar señales. Como se muestra en la figura 4.7, un convertidor de este tipo usa un contador ascendente-descendente en lugar del contador solo ascendente. Al inicio de una conversión, teniendo un voltaje de entrada, la salida del convertidor D/A es cero, así que la salida del comparador toma el nivel alto. Esto dirige la señal de reloj a la entrada del conteo ascendente. Los contadores realizan la cuenta ascendente hasta que la salida del convertidor D/A sea mayor que V_{in} .

La salida del comparador cambia tomando un valor bajo y dirige la señal de reloj a la entrada del conteo descendente. No obstante que V_{in} no tenga cambio, un conteo descendente hará que la salida del convertidor D/A esté por debajo del valor del V_{in} . La salida del comparador cambiará al valor alto y otra vez dirigirá la señal de reloj a la entrada del conteo ascendente. Un conteo ascendente cambiará la salida del comparador que cambiará al valor bajo quedándose en un ciclo. El "bit" de salida del convertidor A/D oscila para un V_{in} constante, siendo esto una desventaja. Pero es un método más rápido porque así como la señal de entrada cambia, el contador cuenta en forma

ascendente o descendente para "seguirlos" de aquí el nombre de este convertidor.

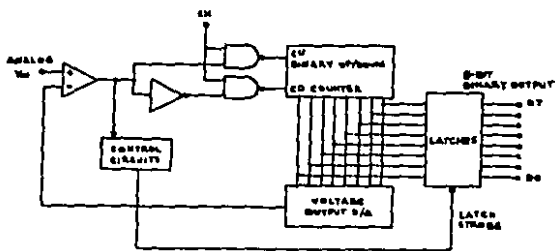


Fig. 4.7 Convertidor A/D usando un Convertidor D/A y un Contador Ascendente-Descendente.

5) Convertidor A/D de Aproximación Sucesiva.- Esta técnica de conversión es la más usada principalmente por sistemas digitales de alta velocidad, porque el tiempo de conversión es fijo e independiente de la magnitud del voltaje de entrada, además de contar con una alta resolución. Su alta velocidad se debe a que N "bits" de resolución pueden ser producidos con solo N pulsos de reloj. El diagrama de bloques de un convertidor A/D de Aproximación Sucesiva se muestra en la figura 4.8. La parte principal de este circuito es el registro de aproximación sucesiva (RAS) que funciona como sigue. Al principio de un ciclo de conversión el RAS, en el primer pulso de reloj, activa su "bit" más significativo, que va hasta el convertidor D/A. Entonces el RAS espera por una señal del comparador indicando si la salida del convertidor D/A es mayor o menor que el voltaje de entrada. Si la salida del comparador tiene el valor alto implica que la salida del convertidor D/A es menor que V_{en} , y el RAS mantendrá al "bit" más significativo activado. Pero si la salida del comparador tiene nivel bajo implica que la salida del convertidor D/A es mayor que V_{en} y el RAS pondrá en 0 lógico el "bit" más significativo. El RAS entonces, continúa con el siguiente "bit" más significativo y así sucesivamente manteniendo el valor del "bit" mayor anterior, requiriendo solo un pulso de reloj para cada "bit". Cuando todos los "bits" han sido obtenidos, el RAS envía una señal de fin de conversión a los "latches", indicándoles que las líneas de salida en paralelo

contienen una palabra válida en binario proporcional al voltaje de entrada.

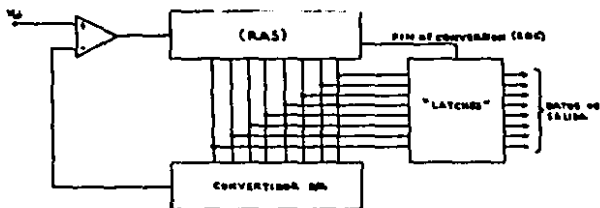


Fig. 4.9 Convertidor A/D de Aproximación Sucesiva.

4.2 CONEXION DEL MANIPULADOR AL MICROCOMPUTADOR.

Como se puede observar es necesario una interfaz para poder establecer una comunicación adecuada entre el manipulador y el microcomputador. Con el fin de aprovechar los circuitos con que cuenta el SDK-85, utilizaremos los puertos de entrada/salida de la primera memoria 8755 (que se describió en el capítulo 2).

El manipulador que usaremos es el modelo 2,600 de Atari, que es un "joystick" del tipo interruptor de "burbuja".

Este manipulador cuenta con cinco interruptores que manejan + 5 Volts y tierra, por tal motivo y por las características del 8755, no es necesario el uso de un Convertidor A/D.

En la figura 4.9 se muestra el diagrama de la conexión del manipulador al 8755. Como se puede observar los interruptores son del tipo "normalmente abiertos". Conectando la línea común de los interruptores a + 5 Volts, implicará que cuando se cierre el interruptor habrá un uno lógico y cuando esté abierto exista un 0 lógico. Para que esto último se cumpla es necesario conectar a las entradas del puerto una resistencia "pull down".

La distribución de la información de entrada del puerto A se muestra en la figura 4.10.

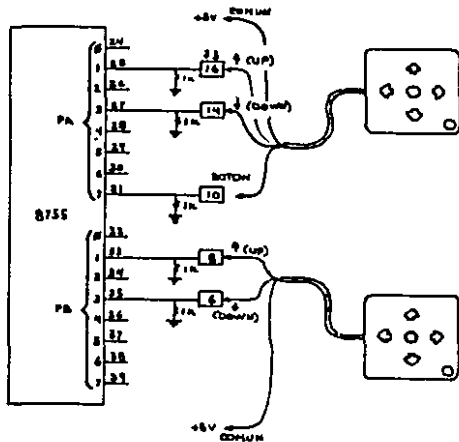


Fig. 4.9 Conexión de los manipuladores al 8755 del SDK-85.

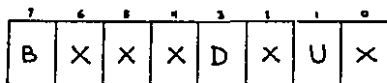


Fig. 4.10 Distribución de los datos de entrada.

Una vez que se ha determinado los circuitos que forman el "Hardware" del sistema, se procederá a su programación. En este capítulo se describirán las secuencias de programa necesarias para diseñar los caracteres gráficos y alfanuméricos, habilitar los puertos de Entrada-Salida y los dispositivos especiales. También se desarrollará el sistema operativo y se verá un programa ejemplo.

Para no extenderse en la explicación de las instrucciones del PDS, se recomienda consultar el libro "Programación del 8080 y 8085" de Octavio F. García García, o directamente en los manuales de INTEL.

También es necesario contar con una lista en código ASCII (American Standard Code for Information Interchange), de los caracteres y que se puede encontrar en la mayoría de los libros que tratan de programación en lenguaje máquina.

3.1 DISEÑO DE CARACTERES GRÁFICOS Y ALFANUMÉRICOS PARA EL CIRCUITO DE VIDEO.

Como se puede observar en la figura 3-11, el generador de caracteres usado en este diseño es un EPROM 2716. Los tres "bits" menos significativos del contador de líneas (CC0-CC2) del 8275 están conectados a las tres líneas de dirección de menor orden del generador de caracteres y los siete líneas del código de carácter (CC0-CC6) están conectadas a A0-A6 del generador de carácter.

Teniendo en cuenta lo anterior, cada carácter en realidad tendrá ocho líneas desplegadas en la matriz de 7 x 16 y usando 1,024 "Bytes" de memoria, implica un espacio suficiente para almacenar 128 caracteres.

Ahora asumamos que la letra "E" va a ser desplegada. El código ASCII para "E" es 45H, así que 45H estará presente en las líneas de dirección, CC0-CC6 del generador de caracteres. A partir de este momento las líneas de la pantalla serán contadas de cero a siete para formar el carácter como se muestra en la figura 3-1.

45H = 1 0 0 0 1 0 1 1

La dirección al PROM = 1 0 0 0 1 0 1 LC2 LC1 LC0
 = (228H-22FH).

Por lo que la salida del generador de caracteres será:

DIRECCION DE MEMORIA	SALIDA HEX	"BITS" DE SALIDA							
		0	1	2	3	4	5	6	7
228H	3E	X	X	X	X	X			
229H	02	X							
22AH	02	X							
22BH	1E	X	X	X	X				
22CH	02	X							
22DH	02	X							
22EH	3E	X	X	X	X	X			
22FH	00								

FIG. 5.1 Generación de un Caracter.

Este mismo procedimiento es usado para formar los demás caracteres tanto gráficos como alfanuméricos.

5.2 PROGRAMACION DE LOS PUERTOS DE ENTRADA/SALIDA Y LOS DISPOSITIVOS ESPECIALES.

Por cuestiones de orden y para ayudar al entendimiento de las partes del programa se propone la siguiente secuencia:

- 1) Programación del 8275.
- 2) Programación del 8237.
- 3) Programación del 8279.
- 4) Programación de los Puertos 0 y 1 del SDK-83.

1) Programación del 8275.-

El 8275 tiene dos registros de programación, el de Comando (CREG) y el de Parámetro (PREG). También el 8275 cuenta con el Registro de Estados (SREG) que tiene como característica que solo puede ser leído. Ellos son direccionados como sigue:

AO	OPERACION	REGISTRO
0	Leer	FREG
0	Escribir	FREG
1	Leer	SREG
1	Escribir	CREG

Para poder iniciar operaciones con el 8275 es importante empezar con la programación de los siguientes comandos y parámetros:

Comando "Reset".....Con sus cuatro parámetros.
 Comando "Cargar Cursor".....Con sus dos parámetros.
 Comando "Preset Counters"
 Comando "Empezar Despliegue"

A continuación se enlista los pasos de programa necesarios para la inicialización del 8275:

```

MVI A,00H           ; Borrar y parar despliegue.
OUT AFH
MVI A,4FH           ; Parámetro 1.
OUT AEH
MVI A,5BH           ; Parámetro 2.
OUT AEH
MVI A,B9H           ; Parámetro 3.
OUT AEH
MVI A,D9H           ; Parámetro 4.
OUT AEH
MVI A,80H           ; Carga la posición del
OUT AFH             cursor.
MVI A,00H           ; Posición X.
OUT AEH
MVI A,00H           ; Posición Y.
OUT AEH
MVI A,E0H           ; Poner a los contadores de
OUT AFH             tiempo internos.
MVI A,23H           ; Iniciar despliegue.
OUT AFH
EI                  ; Habilitación de
                   interrupciones.

LOOP:  NOP
      JMP LOOP

```

Las dos primeras instrucciones permiten que el 8085 cargue al 8275 un comando "Reset", posteriormente el 8275 espera 4

"Bytes" correspondientes a los cuatro parámetros de dicho comando. En estos parámetros se define la composición de la pantalla como se puede observar a continuación:

Comando "Reset".-

				Canal de Datos									
		A0	DESCRIPCION	BMS				bms					
Comando	Escribir	1	Comando "Reset"	0	0	0	0	0	0	0	0	0	0
	Escribir	0	"Byte" 1	S	H	H	H	H	H	H	H	H	H
Parámetro	Escribir	0	"Byte" 2	V	V	R	R	R	R	R	R	R	R
	Escribir	0	"Byte" 3	U	U	U	U	L	L	L	L	L	L
	Escribir	0	"Byte" 4	M	F	C	C	Z	Z	Z	Z	Z	Z

En donde:

"Byte" 1 .-

G.- Renglones Espaciados.

- S FUNCION
- 0 Renglones Normales.
- 1 Renglones Espaciados.

H.- Caracteres Horizontales por Renglon.

H H H H H H H	CARACTERES POR RENGLON
0 0 0 0 0 0 0	1
0 0 0 0 0 0 1	2
0 0 0 0 0 1 0	3
.	.
.	.
.	.
1 0 0 1 1 1 1	80
1 0 1 0 0 0 0	Indefinido
.	.
.	.
1 1 1 1 1 1 1	Indefinido

"Byte"2.-

V.- Contador de Repetición de Trazo Vertical.

V V	No. DE CONTEOS DE RENGLON POR VRTC
0 0	1
0 1	2
1 0	3
1 1	4

R.- Renglones Verticales por Cuadro.

R R R R R R	No. DE RENGLONES POR CUADRO
0 0 0 0 0 0	1
0 0 0 0 0 1	2
0 0 0 0 1 0	3
.	.
.	.
.	.
1 1 1 1 1 1	64

"Byte" 3.-

U.- Posicionamiento de la Línea de Subrayar.

U U U U	No. DE LINEA DE LA LINEA DE SUBRAYAR
0 0 0 0	1
0 0 0 1	2
0 0 1 0	3
.	.
.	.
.	.
1 1 1 1	16

L.- No. de Líneas por Renglon de Caracter.

L L L L	No. DE LINEAS POR RENGLON
0 0 0 0	1
0 0 0 1	2
0 0 1 0	3
.	.
.	.
.	.
1 1 1 1	16

"Byte" 4.-

M.- Modo del Contador de Línea.

M	MODOS DEL CONTADOR DE LINEA
0	Modo 0
1	Modo 1

F.- Modo de Atributo de Campo.

F	MODOS DE ATRIBUTO DE CAMPO
0	Transparente
1	No-Transparente

C.- Formato del Cursor.

C C	FORMATO DEL CURSOR
0 0	Bloque de video invertido parpadeante.
0 1	Linea de subrayar parpadeante.
1 0	Bloque de video invertido no parpadeante.
1 1	Linea de subrayar no parpadeante.

Z.- Conteo de Repetición de Trazo Horizontal.

Z Z Z Z	No. DE CONTEOS DE CARACTER POR HRTIC
0 0 0 0	2
0 0 0 1	4
0 0 1 0	4
.	.
.	.
1 1 1 1	32

Por lo tanto con el "Byte" 1 = 4FH, se seleccionan los renglones normales y 80 caracteres por renglón. Con el "Byte" 2 igual a 5BH tenemos dos conteos de renglón para la repetición de trazo vertical que corresponden a 20 líneas, así como 25 renglones por cuadro. Con el siguiente parámetro igual a 89H tenemos en la línea nueve la línea de subrayar, y diez líneas por renglón de carácter. Por último con el valor D9H seleccionamos el Modo 1 del contador de línea, Modo de atributo de campo no transparente, una línea de subrayar parpadeante como formato del cursor y 20 conteos de carácter para la repetición de trazo horizontal.

Con las siguientes instrucciones: MVI A, 80H; OUT AFH; MVI A, 00H; OUT AEH; MVI A, 00H; OUT AEH; se carga la posición del cursor para el caso corresponde a la parte superior izquierda de la pantalla.

Comando "Cargar posición del Cursor".-

	OPERACION	A0	DESCRIPCION	Canal de Datos
				BMS bms
Comando	Escribir	1	Cargar Cursor	1 0 0 0 0 0 0 0
Parámetro	Escribir	0	No. de Caracter	X X X X X X X X
	Escribir	0	No. de Renglón	X X X X X X X X

Para finalizar se ponen los contadores de tiempo interno con el valor correspondiente a la esquina superior izquierda; se inicia el despliegue, seleccionando 8 ciclos de acceso directo a memoria por bloque y se habilitan las interrupciones.

Comando "Preset Counters".-

Comando	OPERACION	A0	DESCRIPCION	Canal de Datos BMS	bms
Comando	Escribir	1	Poner Contadores	1 1 1 0 0 0 0 0	0

Comando "Iniciar Despliegue".-

Comando	OPERACION	A0	DESCRIPCION	Canal de Datos BMS	bms
Comando	Escribir	1	Empezar Despliegue	0 0 1 5 5 5 5 5	B B

S.- Código de Espacio por Bloque.-

S S S	No. DE CICLOS DE RELOJ DE CARACTER ENTRE REQUERIMIENTOS DE DMA'S
0 0 0	0
0 0 1	7
0 1 0	15
0 1 1	23
1 0 0	31
1 0 1	39
1 1 0	47
1 1 1	55

D.- Código de Conteo de Bloque.-

B'B	No. DE CICLOS DE DMA POR BLOQUE
0 0	1
0 1	2
1 0	4
1 1	8

Las interrupciones serán aceptadas por el 8085 de acuerdo a la instrucción EI. El ciclo final es solamente de espera.

2) Programación del 8257.-

El 8257 cuenta con cuatro pares de "Registros de Canal" (cada par consiste de un registro de dirección de Acceso Directo a Memoria y un registro de Conteo Terminal de los "Bits", cada uno), y dos "Registros Generales" (un registro de Selección de Modo y un registro de Estados, cada uno de 8 "Bits").

El "bit" 3 de dirección selecciona a un registro de Canal, si A3 = 0; o a el registro de Selección de Modo (solo para programar) o al registro de Estados (solo para leer), si A3 = 1.

Los tres "bits" menos significativos de dirección A0-A2, especifican el registro de Canal a ser accedido. Cuando se

accesa el registro de Selección de Modo o el de Estados, A0-A2 deben ser cero lógico.

Cuando se accesa un registro de Canal, si A0 = 0 se selecciona al registro de Dirección de DMA y si es igual a 1, se tiene acceso al registro de Conteo Terminal. Mientras que los "bits" A1 y A2 especifican uno de los cuatro canales.

El registro de Dirección de DMA se cargará con la primera localidad de memoria a ser accedida. El valor cargado en los 14 "bits" menos significativos del registro de Conteo Terminal debe ser el número de ciclos de DMA menos uno. Y los dos "bits" más significativos del registro de Conteo Terminal especifican el tipo de operación de Acceso Directo a Memoria para el canal, esto es:

"BIT" 15	"BIT" 14	TIPO DE OPERACION DE DMA
0	0	Verificación de DMA.
0	1	Escritura de DMA.
1	0	Lectura de DMA.
1	1	(Illegal).

Estos modos de operación son:

- 1.- Lectura de DMA.- Operación en donde el dato se transfiere de memoria al periférico.
- 2.- Escritura de DMA.- El dato se transfiere del periférico a la memoria.
- 3.- Verificación de DMA.- No se realizan transferencias de datos, su uso es exclusivo para verificar información.

Por último, se debe tomar en cuenta que existe un orden para programar los registros del 8257, en primer lugar se debe programar al registro (s) de Dirección de DMA, después el registro (s) de Conteo Terminal y por último al registro de Selección de Modo.

Teniendo en cuenta todo lo anterior tenemos:

```

MVI A,00H           ; Borra registro Selección
OUT 78H            ; de Modo.
LHLD TOPAD         ; Carga en H-L la dirección
MOV A,L           ; de TOPAD.
OUT 94H           ; Carga el registro de Di-
MOV A,H           ; rección del Canal 2 con
OUT 94H           ; la direc. de TOPAD.
MOV A,L           ; Secuencia para restarle
CMA               ; a la dirección límite
MOV L,A           ; superior de memoria la
MOV A,H           ; dirección límite inte-
CMA               ; rior (TOPAD), para deter-
```


MOV H,A	; minar el valor del regis-
INX H	; tro de Conteo Terminal.
LXI D,87CFH	
DAD D	
LXI D,8000H	; Secuencia Para seleccionar
DAD D	; el modo de operacion de
MOV A,L	; lectura de DMA, así como
OUT 95H	; cargar al registro de Con-
MOV A,H	; teo terminal del Canal 2.
OUT 95H	
LXI H,8000H	; Carga el registro de Di-
MOV A,L	; rección del Canal 3 con
OUT 96H	; 8000H.
MOV A,H	
OUT 96H	
LXI H,87CFH	; Carga el registro de Con-
MOV A,L	; teo Terminal con 87CFH.
OUT 97H	
MOV A,H	
OUT 97H	
MVI A,84H	; Carga el registro de Se-
OUT 98H	; lección de Modo con 84H.

Esta secuencia tiene ya parte de explicacion aunque falta por explicar el uso de la direccion TCPAD que se hara durante el desarrollo del sistema operativo. Aquí solo añadiremos que con el valor de 84H en el registro de Selección de Modo se esta habilitando el "bit" de Autocarga y el "bit" de habilitar Canal 2 para Accesos Directos a Memoria. La razon de haber escogido el Canal 2 del 8257 es que es el unico que puede aprovechar el servicio de Autocarga. Este servicio consiste en que los parámetros almacenados en los registros del Canal 3 son transferidos al Canal 2 después de haber sido realizado el primer bloque de ciclos de DMA por el Canal 2.

3) Programación del 8279.-

Todos los comandos de programación del 8279 como interface de teclado y "displays", son enviados por el canal de Datos del sistema, siempre y cuando CS = 0 y A0 = 1 durante el borde ascendente de la señal WR del 8279. Los siguientes comandos programan el modo de operación del 8279, exclusivamente como interface de teclado.

-Activación del Modo de teclado.-

	BMS	bms
Código	0 0 0 D D I K K	

En donde D⁰ es el modo de "display" y KKK es el modo de teclado.

K K K	
0 0 0	Muestreo de Teclado Codificado - "2 Key Lockout" *
0 0 1	Muestreo de Teclado Decodificado - "2 Key Lockout"
0 1 0	Muestreo de Teclado Codificado - "N-Key Rollover"
0 1 1	Muestreo de Teclado Decodificado - "N-Key Rollover"
1 0 0	Muestreo de Matriz Sensora Codificada
1 0 1	Muestreo de Matriz Sensora Decodificada
1 1 0	(Pertenece a la Programación del "Display")
1 1 1	(Pertenece a la Programación del "Display")

* Se selecciona siempre después de la activación de la línea RESET.

El Modo de Teclado seleccionado para este trabajo es el de Muestreo Codificado - "2 Key Lockout", por lo que no requiere ser programado de acuerdo a lo anterior.

-Programación del Reloj.-

	BMS	bms
Código	0 0 1 F P P P F	

Todas las señales de tiempo y multiplexión para el 0279 son generadas por una pre-escala interna. Esta pre-escala divide el reloj externo entre un entero programable. Los "bits" PFFFF determinan el valor de este entero que va desde 2 a 31. Escogiendo un divisor que permita 100 kHz, se darán los tiempos de muestreo necesarios. Para nuestro caso la pata terminal 3 del B279 recibe una señal de reloj de 3.072 MHz, por lo tanto para poder tener una frecuencia de operación de 100 kHz es necesario un divisor igual a 31 (1 1111H).

Por esta razón los pasos de programa de inicialización del B279 quedan resumidos así:

MVA A, JFH	; Programación del Reloj
OUT BFH	; interno de operación.

4) Programación de los Puertos 0 y 1 del SDI-05.-

La sección de Entrada/Salida del B355 u B755 está direccionada por los valores cargados en los "latches" internos ADO-ADI. Cada memoria tiene dos puertos de entrada (A y B) y para cada puerto corresponde un Registro de Dirección de Datos (DDR, Data Direction Register), que sirve para especificar el

estado, entrada o salida, de cada pata terminal de determinado puerto. Esto es:

ADI	ADO	SELECCION
0	0	Puerto A
0	1	Puerto B
1	0	DDR A
1	1	DDR B

Un "bit" con valor igual a 0 en un registro DDR indica que la línea correspondiente a dicho "bit" de ese puerto, está programada en el Modo de Entrada. Y un "bit" con valor 1 indicará que está en el Modo de Salida.

Esta memoria tiene la característica de que cuando su entrada de "RESET" toma nivel 1, todas las líneas de los puertos A y B toman el Modo de Entrada, siendo este el modo en que los ocuparemos. Así que no se requiere de una programación adicional.

5.3 DESARROLLO DEL SISTEMA OPERATIVO.

Las operaciones fundamentales realizadas por el Sistema Operativo son presentadas en la figura 5.2. En los pasos de programación que corresponden al "Reset" del sistema, se deshabilitan las interrupciones, se carga el Apuntador de Pila (SP), se llena el área de memoria de despliegue con ceros de espacio, se inicializan a todos los periféricos y se habilitan las interrupciones. En este momento el microprocesador se queda en un ciclo hasta que recibe una interrupción del 0275, es entonces que el 0085 realiza un servicio de subrutina de interrupción. Aquí el 0257 es reinicializado y se muestra al 8279 para chequear si se ha oprimido un tecla del teclado. Si es así se procede a desplegar en la pantalla, si no, al igual que lo anterior se procede a habilitar las interrupciones, por último se ejecuta una instrucción de regreso al ciclo de espera hasta que ocurra otra interrupción.

Para apreciar la operación del "Software" del sistema en detalle, es necesario considerar los siguientes puntos:

- 1.- La relación entre la posición del carácter en la pantalla, los apuntadores Conteo de Rowspan, Conteo de Columna y el apuntador de los límites superior e inferior de la memoria.

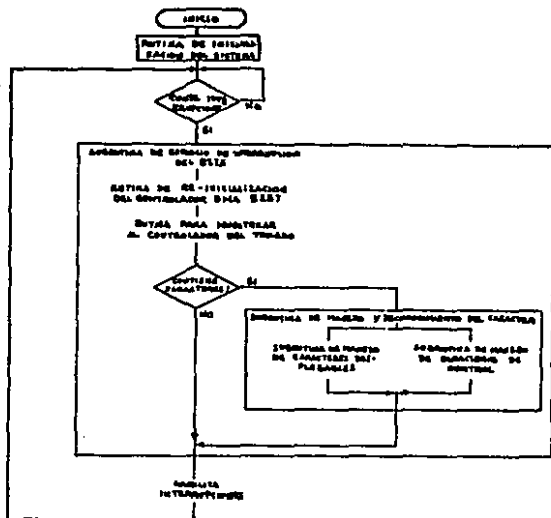


FIG. 5.2 Diagrama de Bloques de la Operación del "Software".

- 2.- La relación entre los apuntadores Conteo de Región, Conteo de Columna y los registros de la posición del cursor X y Y.
- 3.- El concepto de "Scrolling".

- Relación Posición del Caracter y el Apuntador de la Pantalla.-

Para determinar la ubicación de un caracter en la pantalla, son almacenados en memoria dos apuntadores, uno el Conteo de Región y el otro el Conteo de Columna, figura 5.3. El Conteo

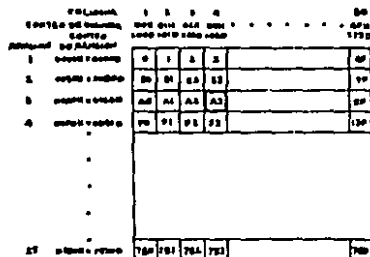


FIG. 5.3 Relación Ubicación del Carácter/Apuntador.

de Renglón y de Columna son almacenados en las localidades de memoria RCTAD y CCTAD, respectivamente. El Conteo de Renglón representa la posición del primer carácter en un renglón dado. Para el primer renglón, el apuntador de renglón será igual a 0000H, para el segundo renglón será igual a 0050H. El Conteo de Columna representa a la columna en donde está posicionado el carácter. Por tal motivo la localización del carácter en la pantalla, puede ser calculada por la suma del Contador de Renglón con el Contador de Columna; por ejemplo un carácter que se encuentra en el renglón A0H y en la columna 03H su posición será:

$$A0H + 03H = A3H$$

- Relación del Apuntador de Memoria y el Registro de la Posición del Cursor del 8275.-

Es necesario establecer una relación entre los apuntadores del Conteo de Renglón, Conteo de Columna y los registros X y Y de la posición del cursor en el 8275, para generar el cursor en la posición adecuada en la pantalla. Esta relación se observa en la tabla 5.1.

El valor transferido al 8275 para la posición X del cursor es idéntico al del Conteo de Columna, cosa que no sucede con la posición Y del cursor y el Conteo de Renglón. Por tal motivo la posición Y será almacenada en la localidad de memoria CURGY.

El parámetro "IDP" será usado en conjunto con el Conteo de Renglón y Columna para determinar la localidad de memoria en donde se almacenará un carácter desplegable. La localidad de

CONTADOR	CONTENIDO DE REGISTRO	VALOR DEL REGISTRO DE LA POSICIÓN Y DEL APUNTADE	COLUMNA	CONTENIDO DE LA COLUMNA	VALOR DEL REGISTRO DE LA POSICIÓN Y DEL APUNTADE
1	8000H	80H	1	00H	80H
2	8050H	01H	2	01H	01H
3	80A0H	02H	3	02H	02H
4	80F0H	03H	4	03H	03H
25	0780H = 1820D	18H = 24D	8D	4FH = 79D	4FH = 79D

TABLA 5.1. Relación entre la Posición del Cursor y los Apuntadores.

memoria en donde será almacenado un carácter se calcula mediante la suma de TOP + Conteo de renglón + Conteo de columna, en donde TOP en un principio es igual a 8000H. Teniendo en cuenta que el número máximo de caracteres desplegados en la pantalla es de 2,000, el último carácter será guardado en la localidad 87CFH. Si se introdujera un carácter más existiría una condición de "scrolling" y TOP será modificado para que apunte a la dirección de memoria 8050H.

- "Scrolling".-

En términos generales el "scrolling" o caracolero implica un movimiento continuo de información, ya sea en forma horizontal o vertical, en la pantalla, como si la información estuviera impresa sobre un papel que se está enrollando por detrás de ésta. Para nuestro caso el movimiento que se implementará aquí es únicamente vertical. La condición de caracolero ocurrirá cuando sean ejecutadas ciertas funciones de manipulación del cursor o cuando se introduce un carácter en la última posición desplegable en la pantalla, indicando que existe una condición de cuadro completo en memoria.

Los caracteres son normalmente introducidos secuencialmente en la memoria de despliegue. Cuando es almacenado el carácter número 2,000, se obtiene la capacidad máxima de memoria de despliegue, en este punto se dará lugar para que se efectúe el caracolero. Para que esto sea posible el Canal 2 del Controlador de DMA que es usado para extraer los caracteres de la memoria, debe ser re-inicializado con valores apropiados de dirección y de conteo terminal. Por lo tanto el apuntador TOP de memoria será usado para establecer la dirección de comienzo para el Canal 2.

Antes del caracolero, TOP = 8000H y cada operación de

caracoleo se sumará 80D (50H) a TOP, de tal forma que se mueve el apuntador como se muestra en la figura 5.4. El efecto de caracoleo hace que se recorra los datos en la pantalla, un renglón hacia arriba. Antes de suceder esto el valor del conteo terminal para el Canal 2 es igual en magnitud al tamaño de la memoria de despliegue - 1; esto es 87CFH - 8000H. Para poder seleccionar la operación de lectura al 8257, es necesario activar el "bit" 15 del registro de conteo terminal. Es por esto que el valor del conteo terminal es igual a $87CFH - 8000H + 8000H$. Como el apuntador de la dirección de inicio TOP estará cambiando, el conteo terminal será igual a $87CFH - TOP + 8000H$.

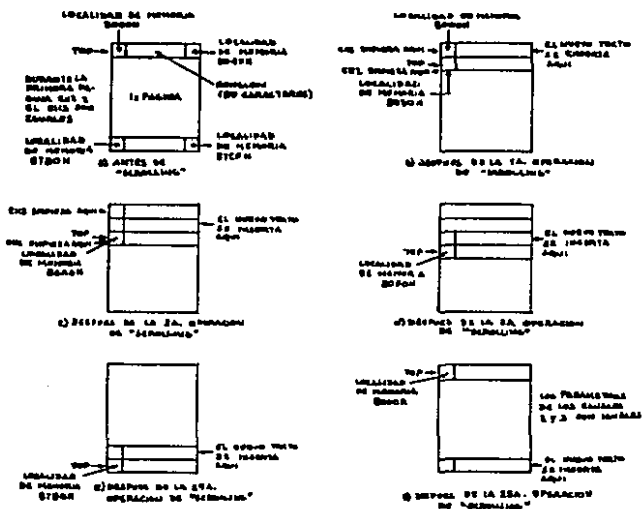


FIG. 5.4 Manipulación del Apuntador durante el Caracoleo.

En otras palabras, la transferencia de caracteres por el Canal 2 va desde la dirección especificada por la dirección TOP hasta el límite superior de la memoria de despliegue. Para transferir los caracteres que se encuentran antes de la localidad TOP se utiliza la característica de auto carga del 8257.

Cuando el Canal 2 termina el conteo terminal, siguiendo la transferencia de caracteres desde TOP hasta el final de la memoria de despliegue, los parámetros de los registros del Canal 1 son cargados dentro del Canal 2. Para que se pueda realizar un caracoleo completo, el registro de dirección del Canal 3 debe ser igual a 8000H y el conteo terminal debe tener el valor de B7CFH. Si se encuentra completa la pantalla, se generará una señal de interrupción deteniendo la transferencia de información de memoria al 8275, por lo que posteriormente se reinicializará al Controlador de DMA para iniciar otro cuadro.

A continuación se hará mención de las subrutinas del Sistema:

1) Rutina de Inicialización del Sistema.-

La Rutina de Inicialización establece un punto de partida para la operación del Sistema. Durante esta rutina, el Controlador de Teclado 8279 solo requiere que se le active la frecuencia de operación interna de 100 KHz. Se proporcionan los parámetros de inicialización para el 8275, como la Repetición de Trazo Horizontal, Repetición de Trazo Vertical, posición de la línea de subrayar, etc. Los registros de la posición del cursor son inicializados en cero, obligando que el cursor se despliegue en la esquina superior izquierda de la pantalla.

Posteriormente, se activa el comando de comienzo de despliegue y se activa la aceptación de las interrupciones, que anteriormente fueron desactivadas.

2) Subrutina de Servicio de Interrupción.-

Esta subrutina del 8275 reinicializa al Controlador de DMA 8257 y chequea los estados del FIFO del 8279. Si un carácter ha sido transmitido desde el teclado, se realiza una operación de revisión de tabla, para obtener el correcto código ASCII o de control de dicho carácter.

3) Subrutina de Manejo de Caracteres Desplegables (DISPL).-

La Subrutina de Manejo de Caracteres Desplegables determina si el cursor es localizado en la última columna del renglón, la última posición desplegable y llama a las subrutinas apropiadas.

4) Subrutina Uno de Despliegue (DIS1).-

La Subrutina Uno de Despliegue calcula la dirección de memoria en la cual el carácter desplegable está siendo insertado. Si el cálculo nos da una dirección fuera de los límites de la memoria de despliegue, se toma la acción de compensación correspondiente.

5) Subrutinas de Despliegue A, B, C (LISA, DISA, DISC).-

Las Subrutinas A, B y C modifican a los apuntadores de memoria de despliegue apropiados. Las modificaciones se basan en la posición presente del cursor. El resultado de la posición del cursor es transferido a los registros de posición del cursor X y Y del 8275. Si se llama a la subrutina DISC ocurre una operación de caracoleo.

6) Subrutina de Manejo de Caracteres de Control (ESCON).-

Esta subrutina toma el caracter de control teclado y enmascara sus 4 "bits" más significativos. Posteriormente se recorre un "bit" a la izquierda y se suma este resultado con la dirección de inicio de la tabla de control con el fin de determinar la secuencia de control a seguir.

7) Rutina Arriba Cursor (CTRLA).-

La Rutina Arriba Cursor determina si el cursor se encuentra en el primer renglón de despliegue. Si es así, los valores de Conteo de Renglón y Conteo de Columna son modificados y se mueve el cursor al último renglón desplegado sin cambiar la posición X. Si el cursor no se encuentra en el renglón superior de la pantalla, se llama a la subrutina de Arriba Renglón.

8) Rutina Abajo Cursor (CTRLD).-

Esta rutina determina si el cursor se encuentra en el último renglón desplegable. Si es así, se llama a la subrutina de caracoleo. Si el cursor no se localiza en el último renglón desplegable, se llama a la subrutina Abajo Renglón.

9) Rutina Derecha Cursor (CRTLC).-

Esta rutina chequea la posición del cursor, si éste se encuentra en la última columna de un renglón superior al último renglón, se realiza un salto de renglón hacia abajo. Si no es así se llama a la subrutina Derecha Columna. En caso en que se encuentre el cursor en la última posición desplegable se procederá a una operación de caracoleo. Los registros de la posición del cursor X y Y son actualizados.

10) Rutina Izquierda Cursor (CTRLD).-

Esta rutina también chequea la posición del cursor, si éste no se encuentra en la primera columna de un renglón inferior al primer renglón, se llama a la subrutina Izquierda Columna. Si no es así, se carga al conteo de columna con la última columna en un renglón y se llama a la subrutina Arriba Renglón. Pero si se encuentra en la primera posición desplegable, se carga el conteo de columna con la última columna, al conteo de renglón con el último renglón y a la posición / del cursor con el último renglón.

11) Rutina "Home" (CTRLC).-

La rutina "Home" carga con ceros al Conteo de renglón, Conteo de Columna y a la posición Y del cursor, sin afectar el valor de TOP.

12) Rutina para Borrar un Caracter (CTRLF).-

Con esta rutina se borra el caracter inmediato a la izquierda del cursor, recorriendo al mismo tiempo la posición del cursor un lugar a la derecha. En el caso en que se encuentre el cursor en cualquier renglón, con excepción del 1o., y en la primera columna, se sube un renglón el cursor y se posiciona en la última columna de dicho renglón.

Por último si se encuentra el cursor en la primera posición desplegable (renglón 0; columna 0), se ignora la petición para realizar esta rutina.

13) Rutina del Servicio de "Return" (CTRLG).-

Esta secuencia llena de espacios en blanco el renglón actual y posiciona el cursor en la primera columna un renglón abajo. En el caso que el cursor se encuentre en el último renglón, ocurre una operación de Caracoleo.

14) Subrutinas Renglón Arriba, Renglón Abajo (ROW UP), (ROW DOWN).-

La subrutina Renglón Arriba le resta 800 al Conteo de Renglón, decremента el apuntador de la posición Y del cursor y actualiza los registros de la posición del cursor del 8275. La subrutina Renglón Abajo suma 800 al Conteo de Renglón, incrementa la posición Y del cursor, y actualiza los registros de la posición del cursor del 8275.

15) Subrutinas Columna Derecha, Columna Izquierda (COLRT), (COLLT).-

La subrutina Columna Derecha incrementa el apuntador del Conteo de Columna y actualiza los registros de la posición del cursor del 8275. La subrutina Columna Izquierda difiere en que el Conteo de Columna se decremента.

16) Subrutina "Scroll" (SCROL).-

La subrutina de Caracoleo, llena el renglón que está apuntando TOP con caracteres de espacio por la subrutina de Llenado, y modifica el valor que contiene TOP para que apunte un renglón abajo del renglón apuntado anteriormente.

17) Subrutina de Llenado (FILL).-

La subrutina de Llenado toma la dirección apuntada por TOP y la llena con caracteres de espacio usando los registros B-C y la instrucción PUSH B.

18) Subrutina para Cargar la Posición del Cursor (WF 75).-

La subrutina para Cargar la Posición del Cursor transfiere los contenidos del Contador de Columna y del apuntador de la posición Y del cursor a los registros X y Y de la posición del cursor del 8275 respectivamente.

En el Apendice 1-a), se enlistan estas secuencias en hexadecimal, los mnemónicos de las instrucciones del microprocesador 8085, así como los comentarios necesarios de estas secuencias.

5.4 DESARROLLO DE UN PROGRAMA EJEMPLO.

Como se pudo ver en el capítulo anterior, se elaboraron las rutinas necesarias para programar a los componentes, bajo la consideración normal de operación de un monitor y su teclado. En otras palabras, de acuerdo a las características de los componentes (principalmente por las del 8275), este circuito que se propone ha sido diseñado para desplegar caracteres alfanuméricos y símbolos. Sin embargo como veremos a continuación se enlista el programa ejemplo con un juego de video.

El juego que se propone es uno de los primeros juegos que salieron al mercado el "Ping-Pong", pero es uno de los pocos que no sobrepasa el límite de caracteres que puede desplegar el 8275 que es de 128 en total.

Las operaciones fundamentales realizadas por el programa se muestran en la figura 5.5. Desde el momento en que se borra la memoria de despliegue se carga el primer cuadro del patron del juego.

A diferencia de la distribución del cuadro del programa anterior y con el fin de disminuir el número de caracteres desplegables, cada caracter se formará con una matriz de punto de 5 líneas horizontales por 7 verticales (en lugar de 10 por 7), por esta razón se aumentará la memoria de despliegue al doble. La distribución del número de renglones y columnas en un cuadro queda como se muestra en la figura 5.6.

Una vez programado el 8275 se habilitan las interrupciones y se queda en un ciclo, esperando una interrupción del controlador del CRT para realizar un servicio de interrupción, en donde se reinicializa al 8257, se revisa el estado de la bandera de juego empezado, si esta activada se procede a continuar el juego, si no, se muestra al manipulador izquierdo para verificar si se ha solicitado un inicio de juego para proseguir con los pasos de inicio de juego y si no es así, se pasa a la secuencia de demostración. Cualquiera de las tres opciones finalmente regresa al ciclo de espera hasta que se realice otra interrupción.

Es importante mencionar que se tendrán tres caracteres que estarán continuamente cambiando: en posición (la pelota y las paletas) y en forma (los marcadores), así que a continuación se explicarán a grandes rasgos estos cambios.

1.- Movimiento de la Pelota.-

La pelota tendrá cuatro movimientos ascendente a la izquierda, descendente a la izquierda, ascendente a la derecha y descendente a la derecha. La idea principal de su movimiento se basa en que antes de poner la pelota en el área de memoria de despliegue calculada por la secuencia de movimiento

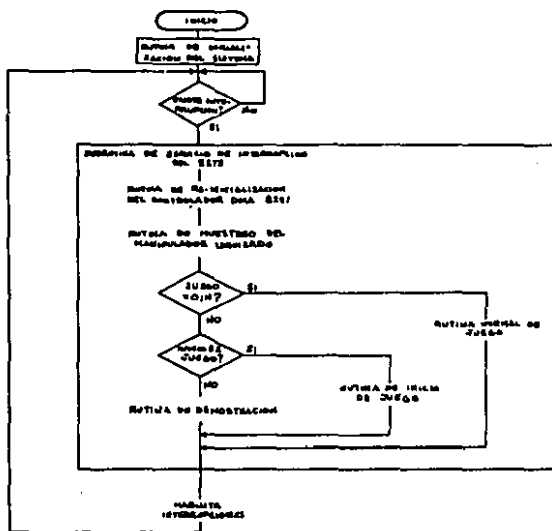


FIG. 5.5 Diagrama de Flujo del Juego.

correspondiente, se salva el caracter que se encuentre en esa misma área en CARAN, y posteriormente antes de realizar otro movimiento se regrese el caracter que se encuentre en CARAN en la posición de memoria desplegable a la que corresponde originalmente. De esta forma se evita perder parte de los marcadores así como la línea intermedia de la "cancha". Durante la parte de demostración las paletas no aparecen en la pantalla. En el caso que la pelota se encuentre en los renglones 0000H y 0F50H, y en las columnas 00H y 4FH se procede a cambiar su movimiento de acuerdo a las consideraciones que tomen las subrutinas de rebote.

En el juego normal se considerará la posición de la paleta correspondiente para que continúe su movimiento la pelota, rebote, o se salte a una subrutina de anotación.

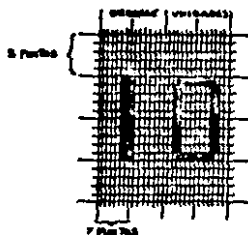


Fig. 5.7 Los Dígitos de los Marcadores.

códigos necesarios para formar los dígitos 10 en la pantalla.

Después de desplegar el número 10 en la pantalla la secuencia de los marcadores continúa con el primer procedimiento, hasta que llega a 15. Entonces deshabilita la bandera de juego empezado, borra las palotas de la pantalla, pone la pelota en la localidad de memoria desplegable 8026H y se carga en DATDE el movimiento de la pelota descendente a la izquierda.

Las rutinas y subrutinas del Sistema son:

1) Rutina de Inicialización del Sistema.-

Durante esta rutina se proporcionan los parámetros de inicialización para el 8275, como la Repetición de Trazo Horizontal, etc. Se activa el comando de comienzo de despliegue y se activa la aceptación de las interrupciones anteriormente desactivadas.

2) Subrutina de Servicio de Interrupción.-

En esta subrutina se reinicializa el 8257 y se verifica en que estado se encuentra el juego (demostración, inicio de juego o juego empezado), y se llaman a las subrutinas apropiadas.

3) Rutina de Demostración (DEMO).-

Esta rutina toma el valor que contiene DATDE y lo suma al valor de inicio de las tabla 1, para poder determinar el movimiento de la pelota durante el ciclo de demostración.

4) Rutinas que Controlan el Movimiento de la Pelota Durante la Demostración (CNT01, CNT02, CNT03, CNT04).-

Estas rutinas con ayuda de la subrutina auxiliar 1, regresan el caracter que se encontraba originalmente en la dirección de memoria desplegable antes de que, en esa misma dirección, se almacenara el código de la pelota.

Posteriormente se realizan cambios en los conteos de renglón y columna de la pelota, según la rutina de movimiento que se está efectuando, al mismo tiempo que se chequean los valores de estos contadores para saltar o no a las secuencias de rebote.

Una vez realizados los cambios con ayuda de la subrutina auxiliar 2, se procede a calcular la nueva dirección que ocupará la pelota, se guarda el caracter que se encuentre ahí en CARAN y se envía el código de la pelota a dicha dirección.

Finalmente se activa el código del movimiento que se acaba de efectuar en DNTDE, con el fin de que cuando ocurra otra interrupción se continúe con el último movimiento y se regresa al ciclo de espera de interrupción.

5) Subrutinas de Rebote Durante la Demostración (REB01, REB02, REB03, REB04).-

Estas subrutinas verifican otros puntos importantes de la posición de la pelota y de acuerdo a los resultados que se obtengan se procede a cambiar el movimiento de la pelota.

6) Rutina de Inicio de Juego (INJUE).-

En esta rutina se activa en primer lugar la bandera de juego empezado, se borran los contadores de los marcadores así como los dígitos en la pantalla, se posicionan las paletas en sus columnas respectivas y a la mitad de la pantalla y se carga el movimiento de la pelota durante el juego en MOVPE, de tal forma que el inicio del juego sea alternado, es decir una vez a la izquierda y otra a la derecha. Por último se regresa al ciclo de espera de interrupción.

7) Rutina Normal de Juego (JUEMP).-

Primeramente revisa si existe una solicitud de movimiento de las paletas, si es así toma determinadas consideraciones y realiza o no el movimiento de estas. Luego la secuencia continúa con el movimiento de la pelota que está determinado por el contenido de MOVPE y el valor del inicio de la tabla 2.

8) Rutinas de Movimiento de la Pelota Durante el Juego (CNT05, CNT06, CNT07, CNT08).-

Estas rutinas son iguales a las que se realizan durante el ciclo de demostración, con la diferencia que primero se realizan unos pasos para verificar las posiciones de las paletas y la pelota para poder realizar una secuencia de rebote de pelota o no.

Se continúa igual que las rutinas de movimiento anteriores

pero antes de hacer los cambios en los contadores de renglón y columna de la pelota se realizan ciertas consideraciones para continuar con el movimiento, pasar a una secuencia de rebote de pelota para el juego normal o hacer una secuencia de anotación.

Es igual que en las secuencias anteriores se activa el código del movimiento que se acaba de efectuar pero en MOVPE, para que cuando se regrese a la rutina de juego empezado se continúe con el último movimiento realizado por la pelota.

9) Subrutinas de Rebote Durante el Juego Normal (REB05, REB06, REB07, REB08).-

Estas subrutinas son iguales a las anteriores con la excepción de que pueden saltar a las subrutinas de anotación.

10) Subrutinas de Anotación (ANOT1, ANOT2).-

Estas subrutinas incrementan el contenido del conteo de los marcadores (izquierdo en ANOT1 y derecho en ANOT2), y lo comparan con el valor OAH, si es igual convierten este resultado de código hexadecimal al código decimal codificado en binario (BCD), y se despliega inmediatamente en la pantalla el número 10. Si después de incrementar el contador no es igual a OAH, se procede a incrementar el dígito de las unidades en el marcador desplegado.

Por último si el contenido del contador no es igual a 15H se continúa el juego cargando la pelota en la mitad de la pantalla correspondiente al lado en donde se efectuó la anotación, pero si llega el marcador a 15, se desactiva la bandera de juego empezado, se borran las paletas de la pantalla y se destina una nueva posición de la pelota para que a partir de ella se inicie el movimiento de la pelota en la rutina de demostración.

El listado del programa, que tiene partes comentadas que facilitan su seguimiento, se encuentra en el Apéndice 1-b).

La tabla con el código hexadecimal de los caracteres Gráficos del juego está a continuación.

CODIGO	CARACTER GRAFICO
10H	Pelota.
12H	Paleta izquierda.
14H	Paleta derecha.
20H	Espacio en blanco.
21H	Parte superior izquierda del dígito 0.
22H	Parte superior derecha del dígito 0.
23H	Parte inferior izquierda del dígito 0.
24H	Parte inferior derecha del dígito 0.

25H	Parte superior izquierda del dígito 1.
26H	Parte superior derecha del dígito 1.
27H	Parte inferior izquierda del dígito 1.
28H	Parte inferior derecha del dígito 1.
29H	Parte superior izquierda del dígito 2.
2AH	Parte superior derecha del dígito 2.
2BH	Parte inferior izquierda del dígito 2.
2CH	Parte inferior derecha del dígito 2.
2DH	Parte superior izquierda del dígito 3.
2EH	Parte superior derecha del dígito 3.
2FH	Parte inferior izquierda del dígito 3.
30H	Parte inferior derecha del dígito 3.
31H	Parte superior izquierda del dígito 4.
32H	Parte superior derecha del dígito 4.
33H	Parte inferior izquierda del dígito 4.
34H	Parte inferior derecha del dígito 4.
35H	Parte superior izquierda del dígito 5.
36H	Parte superior derecha del dígito 5.
37H	Parte inferior izquierda del dígito 5.
38H	Parte inferior derecha del dígito 5.
39H	Parte superior izquierda del dígito 6.
3AH	Parte superior derecha del dígito 6.

3BH	Parte inferior izquierda del dígito 6.
3CH	Parte inferior derecha del dígito 6.
3DH	Parte superior izquierda del dígito 7.
3EH	Parte superior derecha del dígito 7.
3FH	Parte inferior izquierda del dígito 7.
40H	Parte inferior derecha del dígito 7.
41H	Parte superior izquierda del dígito 8.
42H	Parte superior derecha del dígito 8.
43H	Parte inferior izquierda del dígito 8.
44H	Parte inferior derecha del dígito 8.
45H	Parte superior izquierda del dígito 9.
46H	Parte superior derecha del dígito 9.
47H	Parte inferior izquierda del dígito 9.
48H	Parte inferior derecha del dígito 9.
4CH	Línea intermedia de la "cancha".

CONCLUSIONES

- 1) El diseño partiendo de circuitos ya armados como el SDK-65, implican que se tiene que tomar muy en cuenta los direccionamientos tanto de memoria como de los puentes de entrada-salida, para evitar posibles conflictos durante el funcionamiento normal.
- 2) Existen diferentes formas de crear las señales de video, pero una de las más sencillas y fáciles de entender, aunque tiene limitantes, es usando el circuito integrado 0275.
- 3) Los convertidores Analógico-Digital y Digital-Analógico se aplican en infinidad de circuitos electrónicos, es por esto que no importando que los convertidores Analógico-Digital no se utilizaran directamente en el circuito propuesto, se tomaron en cuenta para ser mencionados.
- 4) Los efectos de simulación de movimiento de los juegos de video son variados y tan complicados como el juego lo requiera, porque la mayoría de los juegos simulan movimientos relacionados con fuerzas físicas.
- 5) La velocidad de la pelota y de las paletas en el juego será directamente proporcional a la frecuencia de la repetición de trazo vertical por posición en la memoria desplegable.

APENDICE

1-a) LISTADO DEL PROGRAMA.

LOCA	CON. HEX.	SIMBOLOGIA	COMENTARIOS
		§ RUTINAS DE INICIALIZACION DEL SISTEMA §	
0000	C34000	JMP ARCRT	§Salta a las rutinas de §inicialización.
0034	CCB200	JMP SERIN	§Salta a la rutina de §servicio de interrup- §ción.
003C	CC4000	JMP ARCRT	§Salta a las rutinas de §inicialización.
0040	F3	ARCRT: DI	§Deshabilita interrup- §ciones.
0041	31FF28	LXI SP,28FFH	§Carga el SP
		§ RUTINA PARA LIMPIAR MEMORIA DE DESPLIEGUE §	
0044	2100B0	LXI H,0000H	§Carga en H-L la ia. di- §rección de la memoria §desplegable.
0047	3E20	BLAN: MVI A,20H	§Mueve el código del ca- §racter de espacio al §registro A.
0049	77	MOV M,A	§Muevelo a memoria.
004A	7D	MOV A,L	§Mueve la parte baja de §la dirección a A.
004B	FECF	CPI CFH	§Última columna?
004D	CAS400	JZ PRUE1	§Si es así salta a §PRUE1.
0050	23	INX H	§Si no incrementa H-L.
0051	C34700	JMP BLAN	§Salta a BLAN.
0054	7C	PRUE1: MOV A,H	§Mueve la parte alta de §la dirección a A.
0055	FEB7	CPI 07H	§Último renglon?
0057	CASE00	JZ CONUA	§Si es así salta a §CONUA.
005A	23	INX H	§Si no, incrementa H-L.
005B	C34700	JMP BLAN	§Salta a BLAN.
		§ RUTINA PARA Borrar LOS APUNTAORES §	
005E	210000	CONUA: LXI H,0000H	§Carga en H-L, 0000H.

0061	220020	SHLD RCTAD	;Conteo de renglón en ;0000H.
0064	220220	SHLD LOCBUF	;Borra LOCBUF.
0067	220420	SHLD LQCAD	;Posición del carácter ;en 0000H.
006A	220620	SHLD LQCO1	;Posición del 1er. ca- ;racter del renglón en ;0000H.
006D	2100B0	LXI H,8000H	;Carga en H-L, 8000H.
0070	220820	SHLD TOPAD	;Límite inferior igual ;8000H.
0073	3E00	MVI A,00H	;Mueve 00H a A.
0075	320A20	STA COTAD	;Conteo de columna en ;00H.
0078	320B20	STA CURSY	;Apuntador Y del cursor ;en 00H.
007B	320C20	STA KYCHA	;Carácter del teclado.
007E	320D20	STA INDAT	;Carácter desplegable o ;de control = 00H.

;

RUTINA DE INICIALIZACION DEL B279

0081	3E3F	MVI A,3FH	;Programación del reloj.
0082	D3BF	OUT BFH	

;

RUTINA DE INICIALIZACION DEL B275

0085	3E00	MVI A,00H	;Borra y para el des- ;pliegue.
0087	D3AF	OUT AFH	
0089	3E4F	MVI A,4FH	;Parámetro 1.
008B	D3AE	OUT AEH	
008D	3E5B	MVI A,5BH	;Parámetro 2.
008F	D3AE	OUT AEH	
0091	3E09	MVI A,99H	;Parámetro 3.
0093	D3AE	OUT AEH	
0095	3E09	MVI A,99H	;Parámetro 4.
0097	D3AE	OUT AEH	
0099	3E80	MVI A,80H	;Carga la posición del ;cursor.
009B	D3AF	OUT AFH	
009D	3E00	MVI A,00H	;Posición X.
009F	D3AE	OUT AEH	
00A1	3E00	MVI A,00H	;Posición Y.
00A3	D3AE	OUT AEH	
00A5	3EE0	MVI A,E0H	;Poner a los contadores ;internos de tiempo.
00A7	D3AF	OUT AFH	
00A9	3E23	MVI A,23H	;Iniciar despliegue.
00AB	D3AF	OUT AFH	
00AD	FB	EI	;Habilita interrupcio- ;nes.
00AE	00	LOOP: NOP	
00AF	C3AE00	JMP LOOP	

;

SUBROUTINA DE SERVICIO DE INTERRUPCION

```

;
00B2 CDB700 SERIN: CALL RT75 ;Llama a RT75.
00B5 FD EI ;Habilita interrupciones.
00B6 C9 RET ;Regresa.
;
RUTINA DE REINICIALIZACION DEL B257
;
00D7 3E00 RT75: MVI A,00H ;Borra el registro de
;seleccion de modo.
00D9 D398 OUT 9BH
00DB 2A0B20 LMLD TDFAD ;Carga en H-L el contenido de TDFAD.
00DE 7D MOV A,L ;Carga el registro de
00DF D394 OUT 94H ;direccion del Canal 2
00E1 7C MOV A,H ;con el contenido de
00E2 D394 OUT 94H ;TDFAD.
;
00E4 7D MOV A,L ;Secuencia para calcular
00E5 2F CMA ;el tamaño de la memoria desplegable.
00E6 6F MOV L,A
00E7 7C MOV A,H
00EB 2F CMA
00E9 67 MOV H,A
00EA 23 INX H
00EB 11CFB7 LXI D,87CFH
00EC 19 DAD D
;
00EF 1100B0 LYI D,8000H ;Selección del modo de
00D2 19 DAD D ;operación de lectura de
00D3 7D MOV A,L ;DMA en el registro de
00D4 D395 OUT 95H ;conteo terminal del Canal 2.
00D6 7C MOV A,H
00D7 D395 OUT 95H
;
00D9 2100B0 LXI H,8000H ;Carga el registro de
00DC 7D MOV A,L ;direccion del Canal 3.
00DD D396 OUT 96H
00DF 7C MOV A,H
00E0 D396 OUT 96H
;
00E2 21CFB7 LXI H,87CFH ;Carga el registro de
00E5 7D MOV A,L ;conteo terminal del Canal 3.
00E6 D397 OUT 97H
00E8 7C MOV A,H
00E9 D397 OUT 97H
;
00EB 3EB4 MVI A,84H ;Carga el registro de
00ED D398 OUT 9BH ;selección de modo con
;(Auto Carga y Canal 2
;habilitado).
;

```

RUTINA DE MUESTREO DEL TECLADO

00CF	DBBF	IN BFH	¡Entrada del FIFO de es-
00CF1	E607	ANI 07H	tados.
00CF3	CAF900	JZ SAL	
00CF6	CDFA00	CALL ENCAR	
00CF9	C9	CALL RET	¡Regresa.
			¡
			RUTINA DE MANEJO/RECONOCIMIENTO DE LD TECLADO
			¡
00FA	DBDE	ENCAR: IN DEH	¡Caracter del teclado.
00FC	EEDV	XRI COH	¡Invierte los 2 "bits"
			¡más significativos.
00FE	320C20	STA FYCHR	
			¡
			RUTINA DE MANEJO/RECONOCIMIENTO DEL CARACTER
			¡
0101	3A0C20	RECAR: LDA KYCHR	¡Carga en A. el contenido en FYCHR.
0104	21B203	LXI H,INTA1	¡Carga en H-L la dirección de inicio de
			¡Tabla.
0107	110000	LXI D,0000H	
010A	5F	MOV E,A	
010E	17	DAD D	¡Obten el caracter de la
010C	7E	MOV A,M	¡tabla.
010D	320C20	STA INDAT	¡Almacena el código
			¡ASCII o de control de
			¡teclado en INDAT.
0110	FEB0	CFI 50H	¡Es un código de
			¡control?
0112	DA1901	JC RETAR	¡No, salta a RETAR.
0115	ED1D01	CALL ESCON	¡Si A es mayor o igual a
			¡0001, llama a la subrutina
			¡de caracteres de control.
0118	C9	RET	¡Regresa.
0119	CD9B02	RETAR: CALL DESFL	¡Llama a la subrutina de
			¡manejo de caracteres
			¡desplegables.
011C	C9	RET	¡Regresa.
			¡
			SUBROUTINA DE RECONOCIMIENTO/MANEJO DE
			CARACTERES DE CONTROL
			¡
011D	3A0C20	ESCON: LDA INDAT	¡Carga el código de control
			¡en A.
0120	E60F	ANI 0FH	¡Enmascara "bits" del 4
			¡al 7.
0122	07	RLC	¡Recorre a la izquierda
			¡un lugar sin que el
			¡carreos afecte.
0123	21A403	LXI H,BSET1	¡Carga en H-L la dirección
			¡del inicio de la
			¡tabla BSET1.
0126	110000	LXI D,0000H	¡Carga en D-E con 0000H.
0129	5F	MOV E,A	¡Mueve el contenido de A

012A	19	DAD D	tal registro E. ;Suma H-L con D-E para ;obtener la dirección de ;control.
012B	5E	MOV E,M	;Mueve el contenido de ;la tabla al registro E.
012C	23	INX H	;Incrementa la dirección ;de la tabla.
012D	56	MOV D,M	;Mueve el contenido de ;la tabla al registro D.
012E	EB	XCHG	;Intercambia contenidos ;entre registros D-E y ;H-L.
012F	E9	PCHL	;Carga al Contador de ;Programa con la direc- ;ción de control a se- ;guir.
; RUTINA PARA SUBIR CURSOR ;			
0130	2A0020	CTRLA: LHLD RCTAD	;Carga el contenido del ;conteo de renglón en ;H-L.
0133	7D	MOV A,L	;Mueve el "byte" menor ;del conteo a A.
0134	FE00	CPI 00H	;Comparalo con 00H.
0136	CA3D01	JZ ALPHA	;Si es igual a 00H con- ;tinúa comparación.
0139	CD5B02	CALL ROWUP	;No, llama a la subrutina ;ROWUP.
013C	C9	RET	;Regresa.
013D	7C	ALPHA: MOV A,H	;Mueve el "byte" alto ;del conteo de renglón ;a A.
013E	FE00	CPI 00H	;Comparalo con 00H.
0140	CA4701	JZ BETA	;Si es igual a 00H, sal- ;ta a BETA.
0143	CD5B02	CALL ROWUP	;No, llama a la subrutina ;ROWUP.
0146	C9	RET	;Regresa.
0147	21B007	BETA: LXI H,0790	;Carga el valor del úl- ;timo renglón en H-L.
014A	220020	SHLD RCTAD	;Almacénalo en la direc- ;ción RCTAD.
014D	3E18	MVI A,18H	;Mueve a A el último ;valor para CURSY.
014F	320B20	STA CURSY	;Almacénalo en la direc- ;ción CURSY.
0152	CD8C02	CALL WF75	;Llama a la subrutina ;para cargar la posición ;del cursor.
0155	C9	RET	;Regresa.

RUTINA PARA BAJAR CURSOR

;

0156	2A0020	CTRLB:	LHLD RCTAD	:Carga el contenido del :conteo de renglón en :H-L.
0159	7D		MOV A,L	:Mueve el "byte" menor :del conteo a A.
015A	FE80		CPI 80H	:Compara lo con 80H.
015C	C6301		JZ GAMMA	:Si es igual a 80H con- :tinda la comparacion.
015F	CD6A02		CALL ROWDN	:No, llama a la subrutina :ROWDN.
0162	C9		RET	:Regresa.
0163	7C	GAMMA:	MOV A,H	:Mueve el "byte" alto :del conteo de renglón a :A.
0164	FE07		CPI 07H	:Compara lo con 07H.
0166	CA6901		JZ DELTA	:Si es igual, salta a la :direccion DELTA.
0169	CD6A02		CALL ROWDN	:No, llama a la subrutina :ROWDN.
016C	C9		RET	:Regresa.
016D	CD6C02	DELTA:	CALL WP75	:Llama a la subrutina :para cargar la posición :del cursor.
0170	CD4602		CALL SCROL	:Llama a la subrutina :SCROL.
0173	C9		RET	:Regresa.
			:	
			RUTINA PARA MOVER A LA DERECHA EL CURSOR	
			:	
0174	3A6A20	CTRLC:	LDA CCTAD	:Carga el conteo de co- :lumna en A.
0177	FE4F		CPI 4FH	:Compara lo con 4FH.
0179	CAB001		JZ ZETA	:Si el cursor está en la :última columna salta a :la dirección ZETA.
017C	CD8A02		CALL COLRA	:No, llama a la subrutina :columna a la dere- :cha.
017F	C9		RET	:Regresa.
0180	2A0020	ZETA:	LHLD RCTAD	:Carga en H-L el conteo :del conteo de renglón en :H-L.
0183	7D		MOV A,L	:Mueve el "byte" menor :del conteo a A.
0184	FE80		CPI 80H	:Compara lo con 80H.
0186	C29201		JNZ CCTOA	:Si no es igual, salta a :la dirección CCTOA.
0189	7C		MOV A,H	:Mueve el "byte" alto :del conteo a A.
018A	FE07		CPI 07H	:Compara lo con 07H
018C	C29201		JNZ CCTOA	:Si no es el último ren- :glón, salta a CCTOA.
018F	C39B01		JMP CCTOB	:Si es el último renglón :salta a CCTOB.
0192	3E00	CCTOA:	MVI A,00H	:Mueve 00H a A.

0194	320A20		STA CCTAD	:Almacena en el conteo de columna.
0197	CD6A02		CALL ROWDN	:Llama a la subrutina :ROWDN.
019A	C9		RET	:Regresa.
019B	3E00	CCTDB:	MVI A,00H	:Mueve 00H en A.
019D	320A20		STA CCTAD	:Almacena en el conteo de columna.
01A0	CD8C02		CALL WR75	:Llama a la subrutina :para cargar la posicion del cursor.
01A3	CD4603		CALL SCROL	:Llama a la subrutina :SCROL.
01A6	C9		RET	:Regresa.
			:	
			RUTINA PARA MOVER A LA IZQUIERDA EL CURSOR	
			:	
01A7	3A0A20	CTRLD:	LDA CCTAD	:Carga el conteo de columna en A.
01AA	FE00		CP1 00H	:Compara lo con 00H.
01AC	CAB301		JZ NXTA	:Si el cursor esta en la fila, columna de un rengion salta a NXTA.
01AF	CD7C02		CALL COLLT	:No, llama a la subrutina :para mover a la izquierda.
01B2	C9		RET	:Regresa.
01B3	2A0020	NXTA:	LHLD RCTAD	:Carga en H-L el contenido de rengion.
01B6	7D		MOV A,L	:Mueve el "byte" menor del conteo a A.
01B7	FE00		CP1 00H	:Compara lo con 00H.
01B9	C2C501		JNZ CCTMA	:Si no es igual a 00H, salta a CCTMA.
01BC	7C		MOV A,H	:Mueve la parte alta del conteo a A.
01BD	FE00		CP1 00H	:Compara lo con 00H.
01BF	C2C501		JNZ CCTMA	:Si no es igual a 00H salta a CCTMA.
01C2	C3CE01		JMP CCTMB	:Si es igual a 00H implica que existe una condicón de Home.
01C5	3E4F	CCTMA:	MVI A,4FH	:Mueve 4FH a A.
01C7	320A20		STA CCTAD	:Almacena 4FH en el conteo de columna.
01CA	CD5802		CALL ROWUP	:Llama a la subrutina :ROWUP.
01CD	C9		RET	:Regresa.
01CE	218007	CCTMB:	LXI H,07B0H	:Carga en H-L, 07B0H
01D1	220020		SHLB RCTAD	:Almacena en el conteo de rengion con el valor del ultimo rengion.
01E4	3E4F		MVI A,4FH	:Mueve 4FH a A.
01E6	320A20		STA CCTAD	:Almacena en el conteo de columna, el valor de

01D7	3E18	MVI A,18H	!la última columna.
01DB	320B20	STA CURSY	!Mueve 18H a A.
01DE	CB8C02	CALL WP75	!Almacena en CURSY el va- !lor del último renglón.
01E1	C9	RET	!Llama a la subrutina pa- !ra cargar la posición !del cursor.
		↑	!Regresa.
		RUTINA DE LA POSICION DE "HOME"	
		↑	
01E2	210000	CTRL: LXI H,0000H	!Carga en H-L, 0000H.
01E5	320020	SHLD RCTAD	!Almacena 0000H en el !conteo de renglón.
01EB	3E00	MVI A,00H	!Mueve 00H a A.
01E6	320A20	STA CCTAD	!Almacena 00H en el con- !teo de columna.
01ED	320B20	STA CURSY	!Almacena 00H en la di- !rección CURSY.
01F0	CB8C02	CALL WP75	!Llama a la subrutina pa- !ra cargar la posición !del cursor.
01F3	C9	RET	!Regresa.
		↑	
		RUTINA PARA Borrar CARACTER	
		↑	
01F4	3A0A20	CTRLF: LDA CCTAD	!Carga en A, el conten- !do del conteo de colum- !na.
01F7	FE00	CPI 00H	!Comparalo con 00H.
01F9	C22D02	JNZ SEGUN	!Si no es cero, salta a !SEGUN.
01FC	3A0020	LHLD RCTAD	!Carga en H-L, el conte- !nido del conteo de ren- !glón.
01FF	7D	MOV A,L	!Mueve a A, el contenido !de L.
0200	FE00	CPI 00H	!Comparalo con 00H.
0202	C20E02	JNZ PRIM	!No es cero, salta a !PRIM.
0205	7C	MOV A,H	!Mueve a A, el contenido !de H.
0206	FE00	CPI 00H	!Comparalo con 00H.
0208	C20E02	JNZ PRIM	!No es el 1er. renglón?, !salta a PRIM.
020B	C33C02	JMP FIN	!Es el 1er. renglón, sal- !ta a FIN.
020E	3E4F	PRIM: MVI A,4FH	!Mueve a A el valor 4FH.
0210	320A20	STA CCTAD	!Almacena en el conteo !de columna.
0213	3A0020	LHLD RCTAD	!Carga en H-L, el conte- !nido en el conteo de !renglón.
0216	11B0FF	LXI D,FFB0H	!Carga en D-E, el valor !150H.

0219	29	DAD H	:Sumalo a H-L.
021A	220020	SHLD RCTAD	:Almacena el contenido de H-L en el conteo de trenllon.
021D	3E20	MVI A,20H	:Mueve el valor del código de espacio en A.
021F	320D20	STA INDAT	:Almacenaló en la dirección INDAT.
0222	CDCE02	CALL DISI	:Llama a la subrutina DISI.
0225	210B20	LXI H,CURSY	:Carga en H-L, la dirección de CURSY.
0228	35	DCR M	:Decrementa la posición Y del cursor.
0229	CDBC02	CALL WF75	:Llama a la subrutina para cargar la posición del cursor.
022C	C9	RET	:Regresa.
022D	3D	SEGUN: DCR A	:Decrementa el contenido del conteo de columna.
022E	320A20	STA CCTAD	:Almacenaló en CCTAD.
0231	3E20	MVI A,20H	:Mueve el código de espacio en A.
0233	320D20	STA INDAT	:Almacenaló en la dirección INDAT.
0236	CDCE02	CALL DISI	:Llama a la subrutina DISI.
0239	CDBC02	CALL WF75	:Llama a la subrutina para cargar la posición del cursor.
023C	C9	FIN: RET	:Regresa.
		+	
		RUTINA PARA EL SERVICIO DE "RETURN"	
		+	
023D	3E20	CTRLG: MVI A,20H	:Mueve el código de espacio a A.
023F	320D20	STA INDAT	:Almacenaló en la dirección INDAT.
0242	3A0A20	OTROS: LDA CCTAD	:Carga en A, el contenido del conteo de columna.
0245	FE4F	CPI 4FH	:Comparalo con el valor de la última columna.
0247	CAS402	JZ FORTA	:Última columna?, salta a FORTA.
024A	CDCE02	CALL DISI	:Llama a la subrutina DISI.
024D	210A20	LXI H,CCTAD	:Carga en H-L, la dirección del conteo de columna.
0250	34	INR M	:Incrementa el conteo de columna.
0251	C34202	JMF OTROS	:Salta a OTROS.
0254	CD9B02	FORTA: CALL DESPL	:Llama a la subrutina de manejo de caracteres

0257	C9	RET	};desplegables. };Regresa.
		↓	
		SUBROUTINA PARA SUBIR RENGLON	
		↓	
025B	2A0020	ROWUP: LHLD RCTAD	};Carga el contenido del };conteo de renglón en };H-L.
025D	11B0FF	LXI D,FPBOH	};Carga -50H en D-E.
025E	19	DAD D	};Sumalo con el contenido };del conteo de renglón.
025F	220020	SHLD RCTAD	};El resultado almacenalo };en el conteo de reng- };glón.
0262	210B20	LXI H,CURSY	};Carga la dirección del };apuntador Y del cursor };en H-L.
0265	35	DCR M	};Decrementa la posición };Y del cursor.
0266	CD8C02	CALL WP75	};Llama a la subrutina pa };ra cargar la posición };del cursor.
0269	C9	RET	};Regresa.
		↓	
		SUBROUTINA PARA BAJAR RENGLON	
		↓	
026A	2A0020	ROWDN: LHLD RCTAD	};Carga lo contenido en };el conteo de renglón en };H-L.
026D	115000	LXI D,0050H	};Carga 50H en D-E.
0270	19	DAD D	};Sumalo con el contenido };del conteo de renglón.
0271	220020	SHLD RCTAD	};El resultado almacenalo };en el conteo de ren- };glón.
0274	210B20	LXI H,CURSY	};Carga la dirección del };apuntador Y del cursor };en H-L.
0277	34	INR M	};Incrementa la posición };Y del cursor.
027B	CD8C02	CALL WP75	};Llama a la subrutina pa };ra cargar la posición };del cursor.
027B	C9	RET	};Regresa.
		↓	
		SUBROUTINA COLUMNA A LA IZQUIERDA	
		↓	
027C	210A20	COLLT: LXI H,CCTAD	};Carga en H-L la direc- };ción del conteo de co- };lumna.
027F	35	DCR M	};Decrementa el conteo de };columna.
0280	CD8C02	CALL WP75	};Llama a la subrutina pa };ra cargar la posición };del cursor.

0283	C9	RET	¡Regresa.
		¡	
		SUBROUTINA COLUMNA A LA DERECHA	
		¡	
0284	210A20	CDLRT: LXI H, CCTAD	¡Carga en H-L la dirección del conteo de columna.
0287	34	INR M	¡Incrementa el conteo de columna.
0288	CD8C02	CALL WP75	¡Llama a la subrutina para cargar la posición del cursor.
028B	C9	RET	¡Regresa.
		¡	
		SUBROUTINA PARA CARGAR LA POSICION DEL CURSOR	
		¡	
028C	3E60	WP75: MVI A, 80H	¡Carga A con el comando para cargar la posición del cursor.
028E	D3AF	OUT AFH	
0290	3A0A20	LDA CCTAD	¡Carga en A la posición X del cursor.
0293	D3AE	OUT AEH	
0295	3A0B20	LDA CURSY	¡Carga en A la posición Y del cursor.
0298	D3AE	OUT AEH	
029A	C9	RET	¡Regresa.
		¡	
		SUBROUTINA DE MANEJO DE CARACTERES DESPLEGABLES	
		¡	
029B	3A0A20	DESFL: LDA CCTAD	¡Carga el conteo de columna en A.
029E	FE4F	CPI AFH	¡Última columna?
02A0	CAAA02	JZ CTA,	¡Si, salta a CTA.
02A3	CDCE02	CALL DIS1	¡No, llama a la subrutina DIS1.
02A6	CD0603	CALL DISA	¡Llama a la subrutina DISA.
02A9	C9	RET	¡Regresa.
02AA	2A0020	CTA: LHLD RCTAD	¡Carga el conteo de renglón en H-L.
02AD	7D	MOV A, L	¡Mueve la parte baja a A.
02AE	FEB0	CPI 80H	¡Compara la con 80H.
02B0	CABA02	JZ CTB	¡Si es igual, salta a CTB.
02B3	CDCE02	CALL DIS1	¡No, llama a DIS1.
02B6	CD0E03	CALL DISB	¡Llama a DISB.
02B9	C9	RET	¡Regresa.
02BA	7C	CTB: MOV A, H	¡Mueve parte alta del conteo de renglón a A.
02BB	FE07	CPI 07H	¡Último renglón?
02BD	CAC702	JZ CTC	¡Si, salta a CTC.
02C0	CDCE02	CALL DIS1	¡No, llama a DIS1.
02C3	CD0E03	CALL DISB	¡Llama a DISB.

02C6	C9	RET	:Regresa.
02C7	CDCE02	CTC: CALL DIS1	:Llama a DIS1.
02CA	CD2503	CALL DISC	:Llama a DISC.
02CD	C9	RET	:Regresa.
		;	
		SUBROUTINA DIS1	
		;	
02CE	2A0820	DIS1: LHLD TOPAD	:Carga en H-L el contenido de TOPAD.
02D1	EB	XCHG	:Intercambia contenidos entre H-L y D-E.
02D2	2A0020	LHLD RCTAD	:Carga en H-L el contenido del conteo de ren-
			:gion.
02D5	19	DAD D	:Suma los contenidos de TOPAD y RCTAD.
02D6	220620	SHLD LOCO1	:El resultado almacenado en la localidad de memoria LOCO1.
02E9	EB	XCHG	:Intercambia contenidos entre H-L y D-E.
02EA	210000	LXI H,0000H	:Carga 0000H en H-L.
02ED	3A0A20	LDA COTAD	:Carga en A el contenido del conteo de columna.
02E0	6F	MOV L,A	:Mueve el conteo de columna a L.
02E1	19	DAD D	:Suma los contenidos de LOCO1 y COTAD.
02E2	220420	SHLD LOCAD	:Almacena el resultado en la direccion LOCAD.
02E5	3EB7	MVI A,B7H	:Mueve B7H a A.
02E7	BC	CMP H	:Compara con la parte alta del contenido de LOCAD.
02EB	D2F102	JNC NXTCM	:Si no hay acarreo, continúa la comparacion.
02ED	CD3103	CALL COMRT	:Si hay acarreo, llama a la rutina de compensacion.
02EE	C2FD02	JMP XSTAD	:Salta a XSTAD.
02F1	C2FD02	NXTCM: JNZ XSTAD	:Si no son iguales, salta a XSTAD.
02F4	3ECF	MVI A,CFH	:Mueve CFH a A.
02F6	BD	CMP L	:Compara L con CFH.
02F7	D2FD02	JNC XSTAD	:Si no hay acarreo, la localidad es menor o igual a B7CFH.
02FA	CD3103	CALL COMRT	:Si hay acarreo, llama a la rutina de compensacion.
02FD	210D20	XSTAD: LXI H,INDAT	:Carga la direccion INDAT en H-L.
0300	7E	MOV A,M	:Mueve el contenido de INDAT a A.
0301	2A0420	LHLD LOCAD	:Carga en H-L el contenido

0304	77	MOV M,A	ndo de LOCAD.
0305	C9	RET	;Mueve a memoria el ca- ;racter desplegable. ;Regresa.
		↓	
		SUBROUTINA DISA	
		↓	
0306	210A20	DISA: LXI H,CCTAD	;Carga la dirección del ;conteo de columna en ;H-L.
0309	34	INR M	;Incrementa el conteo de ;columna.
030A	CD8C02	CALL WP75	;Llama a la subrutina pa ;ra cargar la posición ;del cursor.
030D	C9	RET	;Regresa.
		↓	
		SUBROUTINA DISD	
		↓	
030E	3E00	DISD: MVI A,00H	;Mueve 00H a A.
0310	320A20	STA CCTAD	;Almacénalo en el conteo ;de columna.
0313	2A0020	LHLD RCTAD	;Carga en H-L el conten ;do del conteo de ren- ;glón.
0316	115000	LXI D,0050H	;Carga 50H en D-E.
0319	19	DAD D	;Sumale 80D al conteo de ; renglón.
031A	220020	SHLD RCTAD	;Almacena el resultado ;en el conteo de ren- ;glón.
031D	210B20	LXI H,CURSY	;Carga la dirección de ;CURSY en H-L.
0320	34	INR M	;Incrementa la posición ;Y del cursor.
0321	CD8C02	CALL WP75	;Llama a la subrutina pa ;ra cargar la posición ;del cursor.
0324	C9	RET	;Regresa.
		↓	
		SUBROUTINA DISC	
		↓	
0325	3E00	DISC: MVI A,00H	;Mueve 00H a A.
0327	320A20	STA CCTAD	;Almacénalo en el conteo ;de columna.
032A	CD8C02	CALL WP75	;Llama a la subrutina pa ;ra cargar la posición ;del cursor.
032D	CD4603	CALL SCROL	;Llama a la subrutina de ;SCROL.
0330	C9	RET	;Regresa.
		↓	
		SUBROUTINA DE COMPENSACION DE DIRECCION	
		↓	
0331	2A0420	COMRT: LHLD LOCAD	;Carga en H-L la direc-

0334	1130FB	LXI D,FB30H	ición contenida en ;LOCAD.
0337	19	DAD D	;Carga en D-E el valor de compensación.
0338	220420	SHLD LOCAD	;Sumalo a la dirección de memoria contenida en ;LOCAD.
033B	2A0620	SHLD LOCAD ; LHLD LOCO1	;Almacena el nuevo valor en LOCAD.
033E	1130FB	LXI D,FB30H	;Carga la dirección del ter. caracter del ren- gión en H-L.
0341	19	DAD D	;Carga en D-E el valor de compensación.
0342	220620	SHLD LOCO1	;Sumalo a la dirección contenida en LOCO1.
0345	C9	RET	;Almacena el nuevo valor en LOCO1.
		↑	;Regresa.
		SUBROUTINA DE "SCROL"	
		↑	
0346	2A0820	SCROL: LHLD TOPAD	;Carga en H-L el conteni- do de TOPAD.
0349	220220	SHLD LOCBUF	;Almacenaló en LOCBUF.
034C	CD6D03	CALL FILL	;Llama a la subrutina de llenado.
034F	2A0820	LHLD TOPAD	;Carga en H-L el conteni- do de TOPAD.
0352	7D	MOV A,L	;Mueve el contenido de L a A.
0353	FE90	CPI 80H	;Comparalo con 80H.
0355	C26503	JNZ DUCK	;Si no son iguales conti- núa el "Scroll".
0358	7C	MOV A,H	;Mueve el contenido de H a A.
0359	FE87	CPI 87H	;Comparalo con 87H.
035B	C26503	JNZ DUCK	;Si no son iguales, con- tínua el "Scroll".
035E	210080	LXI H,8000H	;Si son iguales, TOPAD ;llegó al límite por lo ;tanto carga en H-L ;8000H.
0361	220820	SHLD TOPAD	;Almacenaló en TOPAD.
0364	C9	RET	;Regresa.
0365	115000	DUCK: LXI D,0050H	;Carga en D-E, 50H.
036D	19	DAD D	;Sumalo al contenido de ;TOPAD.
0369	220820	SHLD TOPAD	;El resultado almacenalo en TOPAD.
036C	C9	RET	;Regreso.
		↑	
		SUBROUTINA DE LLENADO	
		↑	

0340	012020	FILL: LXI B,2020H	¡Carga el código de carácter de espacio en B-C.
0370	310000	LXI H,0000H	¡Carga 0000H en H-L.
0373	35	DAD SP	¡Suma H-L con el contenido del Apuntador del "Stack".
0374	EB	XCHG	¡Intercambia contenidos entre H-L y D-E.
0375	2A0220	LHLD LOCBUF	¡Carga en H-L el contenido de LOCBUF.
0378	F9	SPHL	¡Carga el SP con el contenido de H-L.
0379	05	PUSH B	¡Con la instrucción
037A	05	PUSH B	¡PUSH, llena el siguiente
037B	05	PUSH B	¡registro desplegable
037C	05	PUSH B	¡con caracteres en blanco.
037D	05	PUSH E	
037E	05	PUSH D	
037F	05	PUSH B	
0380	05	PUSH B	
0381	05	PUSH B	
0382	05	PUSH B	
0383	05	PUSH B	
0384	05	PUSH B	
0385	05	PUSH B	
0386	05	PUSH B	
0387	05	PUSH B	
0388	05	PUSH B	
0389	05	PUSH B	
038A	05	PUSH B	
038B	05	PUSH B	
038C	05	PUSH D	
		!	
038D	05	PUSH B	
038E	05	PUSH B	
038F	05	PUSH B	
0390	05	PUSH B	
0391	05	PUSH B	
0392	05	PUSH B	
0393	05	PUSH B	
0394	05	PUSH B	
0395	05	PUSH B	
0396	05	PUSH B	
0397	05	PUSH B	
0398	05	PUSH B	
0399	05	PUSH B	
039A	05	PUSH B	
039B	05	PUSH B	
039C	05	PUSH B	
039D	05	PUSH B	
039E	05	PUSH B	
039F	05	PUSH B	
03A0	05	PUSH B	
03A1	EB	XCHG	¡Carga en H-L el SP ori-

03A2	F9	SFHL	Signal.
			¡Carga el SF con el con-
03A3	C9	RET	tenido de M-L.
			¡Regresa.

TABLA DE LOS CARACTERES DE CONTROL

03A4	3001	BSET1:	DW CTRLA
03A6	5601		DW CTRLB
03A8	7401		DW CTRLC
03AA	A701		DW CTRLD
03AC	E201		DW CTRELE
03AE	F401		DW CTRLF
03B0	3D02		DW CTRLG

TABLA DE LOS CARACTERES TECLADOS

03B2	30	INTA1:	DB 30H	
03B3	83		DB 83H	¡ BS
03B4	30		DB 30H	
03B5	3D		DB 3DH	¡ =
03B6	2D		DB 2DH	¡ -
03B7	30		DB 30H	¡ 0
03B8	39		DB 39H	¡ 9
03B9	3B		DB 3BH	¡ B
03BA	85		DB 85H	¡ (DELETE)
03BE	86		DB 86H	¡ (RETURN)
03BC	30		DB 30H	
03BD	5B		DB 5BH	¡ [
03BE	50		DB 50H	¡ P
03BF	4F		DB 4FH	¡ 0
03C0	49		DB 49H	¡ I
03C1	55		DB 55H	¡ U
03C2	37		DB 37H	¡ 7
03C3	30		DB 30H	
03C4	84		DB 84H	¡ (HOME)
03C5	27		DB 27H	¡
03C6	3B		DB 3BH	¡
03C7	4C		DB 4CH	¡ L
03C8	4B		DB 4BH	¡ K
03C9	4A		DB 4AH	¡ J
03CA	82		DB 82H	¡ (FLECHA A LA DERECHA)
03CB	81		DB 81H	¡ (FLECHA A ABAJO)
03CC	80		DB 80H	¡ (FLECHA A ARRIBA)
03CD	83		DB 83H	¡ (FLECHA A LA IZQUIERDA)
03CE	2F		DB 2FH	¡ /
03CF	2E		DB 2EH	¡ .
03D0	2C		DB 2CH	¡ .
03D1	4D		DB 4DH	¡ M
03D2	4E		DB 4EH	¡ N
03D3	42		DB 42H	¡ B

03D4	56	DB	56H	:	V
03D5	47	DB	47H	:	C
03D6	58	DB	58H	:	X
03D7	5A	DB	5AH	:	Z
03D8	41	DB	41H	:	A
03D9	30	DB	30H	:	
03DA	48	DB	48H	:	H
03DB	47	DB	47H	:	G
03DC	46	DB	46H	:	F
03DD	44	DB	44H	:	D
03DE	20	DB	20H	:	SF
03DF	30	DB	30H	:	
03E0	30	DB	30H	:	
03E1	59	DB	59H	:	
03E2	30	DB	30H	:	
03E3	54	DB	54H	:	T
03E4	52	DB	52H	:	R
03E5	45	DB	45H	:	E
03E6	53	DB	53H	:	S
03E7	57	DB	57H	:	W
03E8	51	DB	51H	:	Q
03E9	30	DB	30H	:	
03EA	30	DB	30H	:	
03EB	36	DB	36H	:	L
03EC	35	DB	35H	:	S
03ED	34	DB	34H	:	A
03EE	33	DB	33H	:	C
03EF	32	DB	32H	:	2
03F0	31	DB	31H	:	1
03F1	30	DB	30H	:	

03F2	30	DB	30H	:	
03F3	83	DB	83H	:	BD
03F4	30	DB	30H	:	
03F5	2B	DB	2BH	:	+
03F6	5F	DB	5FH	:	-
03F7	29	DB	29H	:]
03F8	2B	DB	2BH	:	(
03F9	2A	DB	2AH	:	*
03FA	85	DB	85H	:	(DELETE)
03FB	86	DB	86H	:	(RETURN)
03FC	30	DB	30H	:	
03FD	5D	DB	5DH	:	J
03FE	50	DB	50H	:	P
03FF	4F	DB	4FH	:	0
0400	49	DB	49H	:	1
0401	55	DB	55H	:	U
0402	26	DB	26H	:	2
0403	30	DB	30H	:	
0404	84	DB	84H	:	(HOME)
0405	22	DB	22H	:	"

0406	3A	DB 3AH	!
0407	4C	DB 4CH	!
0408	4B	DB 4BH	!
0409	4A	DB 4AH	!
040A	02	DB 2TH	!
040B	81	DB 81H	!
040C	80	DB 80H	!
040D	83	DB 83H	!
040E	3F	DB 3FH	!
040F	3E	DB 3EH	!
0410	3C	DB 3CH	!
0411	4D	DB 4DH	!
0412	4E	DB 4EH	!
0413	42	DB 42H	!
0414	56	DB 56H	!
0415	43	DB 43H	!
0416	3B	DB 3BH	!
0417	5A	DB 5AH	!
0418	41	DB 41H	!
0419	30	DB 30H	!
041A	48	DB 48H	!
041B	47	DB 47H	!
041C	46	DB 46H	!
041D	44	DB 44H	!
041E	20	DB 20H	!
041F	30	DB 30H	!
0420	30	DB 30H	!
0421	59	DB 59H	!
0422	30	DB 30H	!
0423	54	DB 54H	!
0424	52	DB 52H	!
0425	45	DB 45H	!
0426	53	DB 53H	!
0427	57	DB 57H	!
0428	51	DB 51H	!
0429	30	DB 30H	!
042A	30	DB 30H	!
042B	5E	DB 5EH	!
042C	25	DB 25H	!
042D	24	DB 24H	!
042E	23	DB 23H	!
042F	5C	DB 5CH	!
0430	21	DB 21H	!
0431	20	DB 20H	!

1-b) LISTADO DEL PROGRAMA EJEMPLO.

LOCA	CON. HEX.	SIMBOLOGIA	COMENTARIOS
		↑ RUTINAS DE INICIALIZACION DEL SISTEMA	
0000	C34000	↑ JMP ARCRT	↑Salta a las rutinas de inicialización.
0024	C3A900	↑ JMP SERIN	↑Salta a la rutina de servicio de interrupción.
002C	C34000	↑ JMP ARCRT	↑Salta a las rutinas de inicialización.
0040	F3	ARCRT: DI	↑Deshabilita interrupciones.
0041	31FF29	LXI SP,28FFH	↑Carga el SP
		↑ RUTINA PARA LIMPIAR MEMORIA DE DESPLIEGUE	
0044	CD1D02	CALL PRIME	↑Llama a la subrutina del primer despliegue.
0047	212680	LXI H,80C6H	↑Carga la 1a. dirección de la pelota.
004A	3E10	MVI A,10H	↑Mueve al Acumulador el código de la pelota.
004C	77	MOV M,A	↑Muevelo a memoria.
		↑ RUTINA PARA BORRAR LOS AFUNTADORES	
004D	210000	↑ LXI H,0000H	↑Carga en H-L. 0000H.
0050	220020	SHLD CRFE	↑Almacénalo en el conteo de renglon de la pelota.
0053	220220	SHLD PEAN	↑Almacénalo en la dirección que mantendrá la dirección anterior.
0056	220420	SHLD CRJ1	↑Almacénalo en el conteo de renglon del manipuleador izquierdo.
0059	220620	SHLD CRJD	↑Almacénalo en el conteo de renglon del manipuleador derecho.
005C	220820	SHLD RMARC	↑Almacénalo en el conteo de renglon de los marcadores.
005F	3E00	MVI A,00H	↑Mueve 00H al Acumulador.

0061	320A20	STA JUEGO	¡Almacenalo en JUEGO.
0064	320B20	STA CON12	¡Almacenalo en el conteo del marcador izquierdo.
0067	320C20	STA CONDE	¡Almacenalo en el conteo del marcador derecho.
006A	320D20	STA MAIZ	¡Almacenalo en el estado del manipulador izquier- do.
006D	320E20	STA MADE	¡Almacenalo en el estado del manipulador dere- cho.
0070	320F20	STA CHARD	¡Almacenalo en el conteo de columna de los marca- dores.
0073	3E26	MVI A, 26H	¡Mueve 26H al Acumula- dor.
0075	321020	STA COFE	¡Almacenalo en el conteo de columna de la pelot- ta.
0078	321120	STA CPIN	¡Almacenalo en el conteo de columna inicial de la pelota.
007B	3E20	MVI A, 20H	¡Mueve a A, el código del espacio en blanco.
007D	321220	STA CARAN	¡Almacenalo en la direc- ción con el carácter an- terior.
0080	3E02	MVI A, 02H	¡Mueve el código del mo- vimiento de la pelota descendente-izquierda al Acumulador.
0082	321320	STA DATDE	¡Almacenalo en la direc- ción DATDE.
0085	321420	STA MOVPE	¡Almacenalo en la direc- ción MOVPE.

¡
RUTINA DE INICIALIZACION DEL 8275

008B	3E00	MVI A, 00H	¡Borra y para el des- pliegue.
008A	D3AF	OUT AFH	
008C	3E4F	MVI A, 4FH	¡Parámetro 1.
008E	D3AE	OUT AEH	
0090	3EF1	MVI A, F1H	¡Parámetro 2.
0092	D3AE	OUT AEH	
0094	3E61	MVI A, 64H	¡Parámetro 3.
0096	D3AE	OUT AEH	
0098	3ED9	MVI A, D9H	¡Parámetro 4.
009A	D3AE	OUT AEH	
009C	3EE0	MVI A, E0H	¡Poner a los contadores internos de tiempo.
009E	D3AF	OUT AFH	
00A0	3E25	MVI A, 25H	¡Iniciar despliegue.
00A2	D3AF	OUT AFH	
00A4	FB	EI	¡Habilita interrupcio-


```

                                nes.
00A5 00      LOOP: NOP
00A6 C3A500      JNF LOOP
                                ;
                                SUBROUTINA DE SERVICIO DE INTERRUPCION
                                ;
00A9 CDAE00      SERIN: CALL RT75      ;Llama a RT75.
00AC FB          EI                    ;Habilita interrupcio-
                                ;nez.
00AD C9          RET                    ;Regresa.
                                ;
                                RUTINA DE REINICIALIZACION DEL 8257
                                ;
00AE 3E00      RT75: MVI A,00H        ;Borra el registro de
                                ;seleccion de modo.
00B0 D39B          OUT 9BH
00B2 2100B0     LXI H,8000H          ;Carga en H-L la 1a. di-
                                ;reccion de memoria.
00B5 7D          MOV A,L            ;Carga el registro de
00B6 D394          OUT 94H          ;direccion del Canal 2
00B8 7C          MOV A,H            ;con 8000H.
00B9 D394          OUT 94H
                                ;
                                LXI H,BF30H
00BB 2150BF     LXI H,BF30H          ;Selecciona el modo de
00BE 7D          MOV A,L            ;operacion de lectura de
00BF D395          OUT 95H          ;DMA y carga el registro
00C1 7C          MOV A,H            ;de conteo terminal del
00C2 D395          OUT 95H          ;Canal 2.
                                ;
00C4 3E04      MVI A,04H            ;Carga el registro de
00C6 D398          OUT 98H          ;seleccion de modo con
                                ;el Canal 2 habilitado.
                                ;
                                RUTINA DE MUESTREO DEL MANIPULADOR IZQUIERDO
                                ;
00C8 3A0A20     MANIZ: LDA JUEGO      ;Verifica la bandera de
                                ;inicio de juego.
00CB FE01          CPI 01H            ;Esta activada la bande-
                                ;ra JUEGO?
00CD C2D600     JNZ VEJU            ;No, salta a VEJU.
00CE CDDC02     CALL JUEMP           ;Si, llama a la rutina
                                ;JUEMP.
00D3 C3E600     JMP FINMU           ;Salta a "fin de mues-
                                ;treo.
00D6 DB00      VEJU: IN 00H          ;Lee el puerto 0 del
                                ;SDK-85.
00DB E6B0      ANI 80H              ;Añade el valor del
                                ;"bit" más significati-
                                ;vo.
00DA C2E300     JNZ INICO           ;Se esta solicitando que
                                ;inicie juego, salta a
                                ;INICO.
00DD CDE700     CALL DEMO           ;Si no, llama a la rut-
                                ;ina DEMO.
00DE C3E600     JMP FINMU           ;Salta a "fin de mues-

```

00E3	CD7902	INICIO:	CALL INJUE	¡treeo. ¡Llama a la rutina de ¡"inicio de juego". ¡Regrese.
00E6	C9	FINMU:	RET	
↓ RUTINA DE DEMOSTRACION ↓				
00E7	CA1320	DEMO:	LDA DATDE	¡Carga el código de la ¡dirección del moviemi- ¡to de la pelota.
00EA	210407		LXI H, TAB01	¡Carga en H-L la direc- ¡ción del inicio de la ¡tabla TAB01.
00ED	116000		LXI D, 0000H	¡Carga 0000H en D-E.
00F0	5F		MOV E, A	¡Mueve el contenido de A ¡a E.
00F1	17		DAD D	¡Suma H-L con D-E para ¡obtener la dirección ¡del movimiento de la ¡pelota.
00F2	5E		MOV E, M	¡Mueve el contenido de ¡la tabla al registro E.
00F3	27		INX H	¡Incrementa la dirección ¡de la tabla.
00F4	56		MOV D, M	¡Mueve el contenido de ¡la tabla al registro D.
00F5	EB		XCHG	¡Intercambia contenidos ¡entre registros D-E y ¡H-L.
00F6	E9		PCHL	¡Carga al Contador de ¡Programa con la direc- ¡ción del movimiento a ¡seguir.
↓ RUTINA MOVIMIENTO DE DEMOSTRACION ASCENDENTE A LA IZQUIERDA ↓				
00F7	CD7301	CHTO1:	CALL AUXL1	¡Llama a la subrutina ¡auxiliar AUXL1.
00FA	2A0020	CAMB1:	LHLD CRPE	¡Carga en H-L, el conte- ¡tido de CRPE.
00FD	3E00		MVI A, 00H	¡Mueve 00H a A.
00FF	BD		CMP L	¡Compara lo con el conte- ¡tido de L.
0100	C20A01		JNZ CONCI	¡No son iguales, salta a ¡CONCI.
0103	BC		CMP H	¡Compara lo con el conte- ¡tido de H.
0104	C20A01		JNZ CONCI	¡No son iguales, salta a ¡CONCI.
0107	C3CF01		JMP REB01	¡Sí, salta a REB01.
010A	3A1020	CONCI:	LDA CCPE	¡Carga en A, el conten- ¡do del conteo de colum- ¡na de la pelota.
010D	FE00		CPI 00H	¡Compara lo con 00H.

010F	C21501	JNZ NOPA1	:No son iguales, salta a :NOPA1.
0112	C3EA01	JMP REB04	:Si, salta a REB04.
0115	2A0020	NOPA1: LHL D CRPE	:Carga en H-L el conte- :nido del conteo de ren- :gion de la pelota.
011B	11B0FF	LXI D,FFB0H	:Carga -50H en D-E.
011B	19	DAD D	:Sumalo con el conten- :ido de CRPE.
011C	220020	SHLD CRPE	:El resultado almacenado :en el conteo de ren- :gion de la pelota.
011F	211020	LXI H,CCPE	:Carga en H-L la direc- :cion del conteo de co- :lumna de la pelota.
0122	35	DCR H	:Decrementa el conteo de :columna de la pelota.
0123	CDFF01	CALL AUXL1	:Llama a la subrutina :auxiliar AUXL1.
0126	3E00	MVI A,00H	:Mueve el codigo del mo- :vimiento de demostra- :cion ascendente a la :izquierda.
012B	321320	STA DATDE	:Almacena en la direc- :cion DATDE.
012B	C9	RET	:Regresa.
: RUTINA MOVIMIENTO DE DEMOSTRACION DESCENDENTE A LA IZQUIERDA :			
012C	CDFF01	CINT02: CALL AUXL1	:Llama a la subrutina :auxiliar AUXL1.
012F	2A0020	CAMB2: LHL D CRPE	:Carga en H-L el conte- :nido de CRPE.
0132	3E50	MVI A,50H	:Mueve 50H a A.
0134	BD	CMF L	:Compara lo con el conte- :nido de L.
0135	C24101	JNZ CONC2	:No son iguales, salta a :CONC2.
013B	3E0F	MVI A,0FH	:Mueve 0FH a A.
013A	DC	CMF H	:Compara lo con el conte- :nido de H.
013B	C24101	JNZ CONC2	:No son iguales, salta a :CONC2.
013E	C3D001	JMP REB02	:Si, salta a REB02.
0141	3A1020	CONC2: LDA CRPE	:Carga en A, el conten- :ido del conteo de colum- :na de la pelota.
0144	FE00	CPI 00H	:Compara lo con 00H.
0146	C24C01	JNZ NOPA2	:No son iguales, salta a :NOPA2.
0149	C3E101	JMP REB03	:Si, salta a REB03.
014C	2A0020	NOPA2: LHL D CRPE	:Carga en H-L el conte- :nido del conteo de ren- :gion de la pelota.

014F	115000	LXI D,0050H	;Carga 50H en D-E.
0152	19	DAD D	;Sumalo con el contenido de CRPE.
0153	220020	SHLD CRPE	;El resultado almacénalo en el conteo de renglón de la pelota.
0156	211020	LXI H,CCPE	;Carga en H-L la dirección del conteo de columna de la pelota.
0159	35	DCR M	;Decrementa el conteo de columna de la pelota.
015A	CDFF01	CALL AUXLD	;Llama a la subrutina auxiliar AUXLD.
015D	3E02	MVI A,02H	;Mueve el código del movimiento de demostración descendente a la izquierda.
015F	321020	STA DATDE	;Almacénalo en la dirección DATDE.
0162	C9	RET	;Regresa.

↓
 Rutina movimiento de demostración
 ascendente a la derecha

0163	CDFF01	CNTO: CALL AUXL1	;Llama a la subrutina auxiliar AUXL1.
0166	2A0020	CAMB: LHLD CRPE	;Carga en H-L, el contenido de CRPE.
0169	3E00	MVI A,00H	;Mueve 00H a A.
016B	BD	CMP L	;Compáralo con el contenido de L.
016C	C27601	JNZ CONCC	;No son iguales, salta a CONCC.
016F	BC	CMP H	;Compáralo con el contenido de H.
0170	C27601	JNZ CONCC	;No son iguales, salta a CONCC.
0173	CCE101	JMP REB03	;Si, salta a REB03.
0176	3A1020	CONCC: LDA CCPE	;Carga en A, el contenido del conteo de columna de la pelota.
0179	FE4F	CPI 4FH	;Compáralo con 4FH.
017B	C28101	JNZ NOPA3	;No son iguales, salta a NOPA3.
017E	C3DB01	JMP REB02	;Si, salta a REB02.
0181	2A0020	NOPA3: LHLD CRPE	;Carga en H-L el contenido del conteo de renglón de la pelota.
0184	11B0FF	LXI D,FFB0H	;Carga -50H en D-E.
0187	19	DAD D	;Sumalo con el contenido de CRPE.
018B	220020	SHLD CRPE	;El resultado almacénalo en el conteo de renglón de la pelota.
018D	211020	LXI H,CCPE	;Carga en H-L la direc-

018E	34	INR M	Incrementa el conteo de columna de la pelota.
018F	CDFF01	CALL AUXL2	Incrementa el conteo de columna de la pelota.
0192	3E04	MVI A,04H	Llama a la subrutina auxiliar AUXL2.
0194	321320	STA DATDE	Mueve el código del movimiento de demostración ascendente a la derecha.
0197	C9	RET	Almacénalo en la dirección DATDE.
			Regresa.
			RUTINA MOVIMIENTO DE DEMOSTRACION
			DESCENDENTE A LA DERECHA
0198	CDFF01	CNF04: CALL AUXL1	Llama a la subrutina auxiliar AUXL1.
019B	2A0020	CAMB4: LHLD CRPE	Carga en H-L, el contenido de CRPE.
019E	3E50	MVI A,50H	Mueve 50H a A.
01A0	BD	CMP L	Compara lo con el contenido de L.
01A1	C2AD01	JNZ CONC4	No son iguales, salta a CONC4.
01A4	3E0F	MVI A,0FH	Mueve 0FH a A.
01A6	BC	CMP H	Compara lo con el contenido de H.
01A7	C2AD01	JNZ CONC4	No son iguales, salta a CONC4.
01AA	C3EA01	JMP REB04	Si, salta a REB04.
01AD	3A1020	CONC4: LDA CCPE	Carga en A, el contenido del conteo de columna de la pelota.
01B0	FE4F	CPI 4FH	Compara lo con 4FH.
01B2	C2BB01	JNZ NOFA4	No son iguales, salta a NOFA4.
01B5	C3CF01	JMP REB01	Si, salta a REB01.
01B8	2A0020	NOFA4: LHLD CRPE	Carga en H-L, el contenido del conteo de renglón de la pelota.
01BB	115000	LXI D,0050H	Carga 50H en B-E.
01BE	19	DAD D	Súmalo con el contenido de CRPE.
01BF	220020	SHLD CRPE	El resultado almacénalo en el conteo de renglón de la pelota.
01C2	211020	LXI H,CCPE	Carga en H-L la dirección del conteo de columna de la pelota.
01C5	34	INR M	Incrementa el conteo de columna de la pelota.
01C6	CDFF01	CALL AUXL2	Llama a la subrutina auxiliar AUXL2.

01C9	3E06	MVI A,06H	;Mueve el código del movimiento de demostración descendente a la derecha.
01CB	321320	STA DATDE	;Almacena en la dirección DATDE.
01CE	C9	RET	;Regresa.
		↓	
		SUBROUTINA DE REBOTE 1	
		↓	
01CF	CD1102	REBO1: CALL AUXL3	;Llama a la subrutina auxiliar AUXL3.
01D2	CA9B01	JZ CAMB4	;Son iguales, salta a CAMB4.
01D5	C32F01	JMP CAMB2	;No, salta a la CAMB2.
		↓	
		SUBROUTINA DE REBOTE 2	
		↓	
01D9	CD1102	REBO2: CALL AUXL3	;Llama a la subrutina auxiliar AUXL3.
01DB	CA6601	JZ CAMB3	;Son iguales, salta a CAMB3.
01DE	C3FA00	JMP CAMB1	;No, salta a CAMB1.
		↓	
		SUBROUTINA DE REBOTE 3	
		↓	
01E1	CD1702	REBO3: CALL AUXL4	;Llama a la subrutina auxiliar AUXL4.
01E4	CA2F01	JZ CAMB2	;Son iguales, salta a CAMB2.
01E7	C39B01	JMP CAMB4	;No, salta a CAMB4.
		↓	
		SUBROUTINA DE REBOTE 4	
		↓	
01EA	CD1702	REBO4: CALL AUXL4	;Llama a la subrutina auxiliar AUXL4.
01ED	CAFA00	JZ CAMD1	;Son iguales, salta a CAMD1.
01F0	C36601	JMP CAMB3	;No, salta a CAMB3.
		↓	
		SUBROUTINA AUXILIAR 1	
		↓	
01F3	CDC004	AUXL1: CALL DIPA	;Llama a la subrutina de direccionamiento de la posición de la pelota.
01F6	211220	LXI H,CARAN	;Carga en H-L, la dirección de CARAN.
01F9	7E	MOV A,M	;Mueve el carácter anterior a A.
01FA	2A0220	LHLD PEAN	;Carga en H-L, el contenido de PEAN.
01FD	77	MOV M,A	;Mueve a memoria el carácter anterior.
01FE	C9	RET	;Regresa.
		↓	

SUBROUTINA AUXILIAR 2

01FF	CDC004	AUXL2: CALL DIPA	¡Llama a la subrutina de direccionamiento de la pelota.
0202	2A0220	LHLD PEAN	¡Carga en H-L, el contenido de PEAN.
0205	7E	MOV A,M	¡Mueve el caracter desplegado a A, para salvarlo.
0206	211220	LXI H,CARAN	¡Carga en H-L, la dirección del caracter anterior.
0209	77	MOV M,A	¡Mueve el contenido de A a memoria.
020A	3E10	MVI A,10H	¡Mueve a A, el código de la pelota.
020C	2A0220	LHLD PEAN	¡Carga en H-L, el contenido de PEAN.
020F	77	MOV M,A	¡Mueve el código de la pelota a memoria.
0210	C9	RET	¡Regresa.

SUBROUTINA AUXILIAR 3

0211	3A1020	AUXL3: LDA CQPE	¡Carga en H, el contenido del conteo de columna de la pelota.
0214	FE00	CPI 00H	¡Compáralo con 00H.
0216	C7	RET	¡Regresa.

SUBROUTINA AUXILIAR 4

0217	3A1020	AUXL4: LDA CQPE	¡Carga en A, el contenido del conteo de columna de la pelota.
021A	FE4F	CPI 4FH	¡Compáralo con 4FH.
021C	C9	RET	¡Regresa.

SUBROUTINA DE PRIMER DESPLIEGUE

021D	2100B0	PRIME: LXI H,0000H	¡Carga en H-L, la dirección de la memoria desplegable.
0220	3E20	BLAN: MVI A,20H	¡Mueve el código del caracter de espacio al registro A.
0222	77	MOV M,A	¡Muévelo a memoria.
0223	7D	MOV A,L	¡Mueve la parte baja de la dirección a A.
0224	FE9F	CPI 9FH	¡Última columna?
0226	CA2D02	JZ PRUE1	¡Si, salta a PRUE1.
0229	23	INX H	¡No, incrementa H-L.
022A	C32002	JMP BLAN	¡Salta a BLAN.
022D	7C	PRUE1: MOV A,H	¡Mueve la parte alta de

022E	FEBF	CPI 8FH	!la dirección a A.
0230	CA3702	JZ PATRO	!ultimo renglón?
			!Si es así, salta a
			!PATRO.
0232	23	!H-X-H	!Si no, incrementa H-L.
0234	032002	JMP BLAN	!Salta a BLAN.
		;	
0237	21F80	PATRO: LXI H,80F8H	!Carga la 1a. dirección
			!del marcador izquierdo
			!en H-L.
023A	3E21	MVI A,21H	!Carga el 1er. caracter
			!del dígito 0 en A.
023C	77	MOV M,A	!Muévelo a memoria.
		;	
023D	215981	LXI H,8159H	!Carga la 1a. dirección
			!del marcador derecho
			!en H-L.
0240	77	MOV M,A	!Muévelo a memoria.
		;	
0241	21FC80	LXI H,80FC8H	!Carga la 2a. dirección
			!del marcador izquierdo
			!en H-L.
0244	3E22	MVI A,22H	!Carga el 2o. caracter
			!del dígito 0 en A.
0246	77	MOV M,A	!Muévelo a memoria.
		;	
0247	215A81	LXI H,815AH	!Carga la 2a. dirección
			!del marcador derecho
			!en H-L.
024A	77	MOV M,A	!Muévelo a memoria.
		;	
024B	214881	LXI H,8148H	!Carga la 3a. dirección
			!del marcador izquierdo
			!en H-L.
024E	3E23	MVI A,23H	!Carga el 3er. caracter
			!del dígito 0 en A.
0250	77	MOV M,A	!Muévelo a memoria.
		;	
0251	21A981	LXI H,81A9H	!Carga la 3a. dirección
			!del marcador derecho
			!en H-L.
0254	77	MOV M,A	!Muévelo a memoria.
		;	
0255	214C01	LXI H,814CH	!Carga la 4a. dirección
			!del marcador izquierdo
			!en H-L.
0258	3E24	MVI A,24H	!Carga el 4o. caracter
			!del dígito 0 en A.
025A	77	MOV M,A	!Muévelo a memoria.
		;	
025B	21AA81	LXI H,81AAH	!Carga la 4a. dirección
			!del marcador derecho
			!en H-L.
025E	77	MOV M,A	!Muévelo a memoria.
		;	

025F	21D77F		LXI H,7FD7H	¡Carga la 1a. dirección ¡para la línea interme- ¡dia -50H, en H-L.
0262	115000		LXI D,0050H	¡Carga 50H en D-E.
0265	3E4C	LINT:	MVI A,4CH	¡Mueve a A el caracter ¡de la línea intermedia.
0267	19		DAD D	¡Baja un renglón a la ¡mitad de la pantalla.
0268	77		MOV M,A	¡Mueve el contenido de A ¡a la localidad de memo- ¡ria direccionada por ¡H-L.
0269	7D		MOV A,L	¡Mueve el contenido de L ¡al Acumulador.
026A	FE77		CPI 77H	¡Última línea interme- ¡dia?
026C	C26502		JNZ LINT	¡No, baja otro renglón.
026F	7C		MOV A,H	¡Mueve el contenido de H ¡al Acumulador.
0270	FEBF		CPI 8FH	¡Última línea interme- ¡dia?
0272	CA7802		JZ CONUA	¡Si, sigue en CONUA.
0275	C36502		JMP LINT	¡No, baja otro renglón.
0278	C9	CONUA:	RET	¡Regresa.
			¡	
			RUTINA DE INICIO DE JUEGO	
			¡	
0279	210A20	INJUE:	LXI H,JUECO	¡Carga en H-L, la direc- ¡ción de la bandera
			¡	¡JUEGO.
027C	3601		MVI M,01H	¡Mueve 01H a memoria.
			¡	
027E	211220		LXI H,CARAN	¡Carga en H-L, la direc- ¡ción de memoria que con- ¡tiene el caracter ante- ¡rior.
0281	3620		MVI M,20H	¡Mueve el código de espa- ¡cio en blanco a CARAN.
			¡	
0283	210B20		LXI H,CONIZ	¡Carga en H-L, la direc- ¡ción del conteo del mag- ¡cador izquierdo.
0286	3600		MVI M,00H	¡Mueve 00H a memoria.
			¡	
0288	210C20		LXI H,CONDE	¡Carga en H-L, la direc- ¡ción del conteo del mag- ¡cador derecho.
028B	3600		MVI M,00H	¡Mueve 00H a memoria.
			¡	
028D	CD1D02		CALL PRIME	¡Llama a la subrutina ¡del primer despliegue.
			¡	
0290	21B007		LXI H,07B0H	¡Carga en H-L, el valor ¡del renglón intermedio ¡en la pantalla para la

0293	220420	SHLD CRJI	;paleta. ;Almacénalo en el con- ;teo de renglón del mani- ;pulador izquierdo.
0296	220620	SHLD CRJD	;Almacénalo en el con- ;teo de renglón del mani- ;pulador derecho.
0299	CDAE04	CALL JMOV1	;Llama a la subrutina de ;direccionamiento 1, pa- ;ra la paleta izquierda.
029C	3612	MVI M,12H	;Mueve el código de la ;paleta izquierda a me- ;moria.
029E	2A0620	LHLD CRJD	;Carga en H-L, el valor ;del conteo de renglón ;de la paleta derecha.
02A1	CDB704	CALL JMOV2	;Llama a la subrutina de ;direccionamiento 2, pa- ;ra la paleta derecha.
02A4	3614	MVI M,14H	;Mueve el código de la ;paleta derecha a memo- ;ria.
02A6	21D007	LXI H,07D0H	;Carga en H-L, el valor ;del siguiente renglón ;intermedio para la pa- ;leta.
02A9	CDAE04	CALL JMOV1	;Llama a la subrutina de ;direccionamiento 1, pa- ;ra la paleta izquierda.
02AC	3612	MVI M,12H	;Mueve el código de la ;paleta izquierda a mem- ;ria.
02AE	21D007	LXI H,07D0H	;Carga en H-L, el valor ;del siguiente renglón ;intermedio para la pa- ;leta.
02B1	CDB704	CALL JMOV2	;Llama a la subrutina de ;direccionamiento 2, pa- ;ra la paleta derecha.
02B4	3614	MVI M,14H	;Mueve el código de la ;paleta derecha a memo- ;ria.
02B6	3A1120	LDA CF1H	;Carga en A, el conteni- ;do de la columna ini- ;cial de la pelota.
02B9	FE26	CPI 26H	;Compara lo con la colum- ;na número 26D.
02BB	CACE02	JZ CARDE	;Son iguales, salta a ;CARDE.
02BE	3E26	MVI A,26H	;Mueve a A, el número de

02C0	321120	STA CPIN	columna 38D. :Almacénalo en la local- :lidad de la memoria en :la columna inicial de :la pelota.
02C3	321020	STA CCPE	:Almacénalo en el conteo :de columna de la pelot- :ta.
02C6	3E02	MVI A,02H	:Mueve el código del mo- :vimiento descendente a :la izquierda de la pel- :ta durante el juego.
02C8	321420	STA MOVPE	:Almacénalo en la direc- :ción MOVPE.
02CB	C3DB02	JMP PASA	:Salta a PASA.
02CE	3E2B	CARDE: MVI A,2BH	:Mueve a A, el número de :columna 40D.
02D0	321120	STA CPIN	:Almacénalo en la local- :lidad de la memoria en :la columna inicial de :la pelota.
02D3	321020	STA CCPE	:Almacénalo en el conteo :de columna de la pelot- :ta.
02D6	3E06	MVI A,06H	:Mueve el código del mo- :vimiento descendente a :la derecha de la pelota :durante el juego.
02D8	321420	STA MOVPE	:Almacénalo en la direc- :ción MOVPE.
02DB	C9	PASA: RET	:Regresa.
		↓	
		RUTINA NORMAL DE JUEGO	
		↓	
02DC	DB00	JUEMP: IN 00H	:Lee el contenido del :puerto 0 del SDF-85.
02DE	E60A	ANI 0AH	:Aísla los "bits" 1 y 3 :del Acumulador.
02E0	320D20	STA MAIZ	:Almacénalo en la direc- :ción con la información :del estado del manipu- :lador izquierdo.
02E3	FE02	CFI 02H	:Se está solicitando un :movimiento para arriba?
02E5	CA1B03	JZ PROC1	:Si, salta a PROC1.
		↓	
02E8	FE08	CFI 08H	:Se está solicitando un :movimiento para abajo?
02EA	C23903	JNZ JUMDE	:No, salta a JUMDE para :leer el contenido del :manipulador derecho.
		↓	
02ED	2A0420	LHLD CRJ1	:Carga en H-L, el conte- :nido del conteo de ren- :ción del manipulador

02F0	7D		MOV A,L	;Izquierdo.
02F1	FE00		CPI 00H	;Mueve el contenido de
02F3	C2FC02		JNZ MFDP1	;L, al Acumulador.
02F6	7C		MOV A,H	;Ultimo renglón para la
02F7	FE0F		CPI 0FH	;paleta izquierda?
02F9	CA3903		JZ JUMDE	;No, mueve para abajo la
				;palata izquierda.
02FC	11A000	MFDP1:	LXI D,00A0H	;Mueve el contenido de
				;H, al Acumulador.
02FF	19		DAD D	;Verifica si se trata
				;del último renglón.
0300	CDAE04		CALL JMOVI	;Si, salta a JUMDE para
				;leer el contenido del
0303	3612		MVI M,12H	;manipulador derecho.
0305	1160FF		LXI D,FF60H	;Carga en D-E, el valor
				;que corresponde a bajar
0308	19		DAD D	;dos renglones.
				;Suma el contenido de
0309	3620		MVI M,20H	;D-E al contenido de
				;H-L.
030B	2A0420		LHLD CRJI	;Mueve el código de espá-
				;cio en balanco a memo-
030E	115000		LXI D,0050H	;ria.
				;Carga en H-L, el conte-
0311	19		DAD D	;nido del conteo de ren-
				;glón del manipulador
0312	220420		SHLD CRJI	;izquierdo.
				;Carga en D-E, el valor
0315	C33903		JMP JUMDE	;que corresponde a bajar
				;un renglón.
0318	2A0420	PROCI:	LHLD CRJI	;Suma el contenido de
				;D-E al contenido de
				;H-L.
				;Almacénalo en el conteo
				;del renglón del manipu-
				;lador izquierdo.
				;Salta a JUMDE, para
				;leer el contenido del
				;manipulador derecho.

			¡Según del manipulador ¡izquierdo. ¡Mueve el contenido de ¡H. al Acumulador. ¡Primer renglón para la ¡paleta izquierda". ¡No, mueve para arriba ¡la paleta izquierda. ¡Mueve el contenido de ¡H. al Acumulador. ¡Verifica si se trata ¡del primer renglón. ¡Si, salta a JUMDE para ¡leer el contenido del ¡manipulador derecho.
031B	7D	MOV A,L	
031C	FE00	CPI 00H	
031E	C22703	JNZ JIZUF	
0321	7C	MOV A,H	
0322	FE00	CPI 00H	
0324	CA3903	JZ JUMDE	
		;	
0327	11B0FF	JIZUF: LXI D,FFB0H	¡Carga en D-E, el valor ¡correspondiente a -50H. ¡Suma los contenidos de ¡H-L y D-E. ¡Almacénalo en el conteo ¡de renglón del manipula ¡dor izquierdo. ¡Llene a la subrutina de ¡direccionamiento 1 de ¡los manipuladores. ¡Mueve a memoria el código ¡de la paleta izquier ¡da. ¡Carga en D-E, el valor ¡que corresponde a bajar ¡dos renglones. ¡Suma el contenido de ¡H-L y el de D-E. ¡Mueve a memoria el código ¡del espacio en blan ¡co.
032A	19	DAD D	
032B	220420	SHLD CRJI	
032E	CDAE04	CALL JMOV1	
0331	3a12	MVI M,12H	
0333	11A000	LXI D,00A0H	
0336	19	DAD D	
0337	3a20	MVI M,20H	
		;	
0339	DB01	JUMDE: IN 01H	¡Leo el contenido del ¡puerto 1 del SDF-85. ¡Aísla los "bits" 1 y 3 ¡del Acumulador. ¡Almacénalo en la direc ¡ción con la información ¡del estado del manipu ¡lador derecho. ¡Si está solicitando un ¡movimiento para arriba? ¡Si, salta a PROC2.
033B	E60A	ANI 0AH	
033D	320E20	STA MADE	
0340	FE02	CPI 02H	
0342	CA7503	JZ PROC2	
		;	
0345	FE0B	CPI 0BH	¡Se está solicitando un ¡movimiento para abajo? ¡No, salta a MUEVE para ¡mover la posición de la ¡pelota en la pantalla.
0347	C29603	JNZ MUEVE	

034A	2A0620	LHLD CRJD	¡Carga en H-L, el contenido del conteo de renglón del manipulador derecho.
034D	7D	MOV A,L	¡Mueve el contenido de L al Acumulador.
034E	FE00	CPI 00H	¡Último renglón para la pelota derecha?
0350	02590C	JNZ MPDFL	¡No, mueve para abajo la palota derecha.
0353	7C	MOV A,H	¡Mueve el contenido de H al Acumulador.
0354	FE0F	CPI 0FH	¡Verifica si se trata del último renglón.
0356	CA9&0C	JZ MUEVE	¡Si, salta a MUEVE para mover la posición de la pelota en la pantalla.
0359	11A000	MPDFD: LXI D,00A0H	¡Carga en D-E, el valor que corresponde a bajar dos renglones.
035C	19	DAD D	¡Suma el contenido de D-E al contenido de H-L.
035D	CDP704	CALL JMOV2	¡Llama a la subrutina de direccionamiento 2 de los manipuladores.
0360	3614	MVI M,14H	¡Mueve a memoria, el código de la palota derecha.
0362	11E0FF	LXI D,FF60H	¡Carga en D-E, el valor que corresponde a subir dos renglones.
0365	19	DAD D	¡Suma el contenido de D-E al contenido de H-L.
0366	3620	MVI M,20H	¡Mueve el código de espacio en blanco a memoria.
036B	2A0620	LHLD CRJD	¡Carga en H-L, el contenido del conteo de renglón del manipulador derecho.
036B	115000	LXI D,0050H	¡Carga en D-E, el valor que corresponde a bajar un renglón.
036E	19	DAD D	¡Suma el contenido de D-E al contenido de H-L.
036F	220620	SHLD CRJD	¡Almacénalo en el conteo de renglón del manipulador derecho.
0372	039603	JMP MUEVE	¡Salta a MUEVE, para mover la posición de la

				¡paleta en la pantalla.
0375	2A0620	PROC2:	LHLD CRJD	¡Carga en H-L el contenido del conteo de renglón del manipulador derecho.
0376	7D		MOV A,L	¡Mueve el contenido de L al Acumulador.
0379	FE00		CPI 00H	¡Primer renglón para la paleta derecha?
037B	C28403		JNZ JDEUF	¡No, mueve para arriba la paleta derecha.
037E	7C		MOV A,H	¡Mueve el contenido de H al Acumulador.
037F	FE00		CPI 00H	¡Verifica si se trata del primer renglón.
0381	CA9603		JZ MUEVE	¡Si, salta a MUEVE para mover la posición de la paleta en la pantalla.
0384	11B0FF	JDEUF:	LXI D,FFB0H	¡Carga en D-E, el valor correspondiente a -50H.
0387	19		DAD D	¡Suma los contenidos de H-L y D-E.
0388	220620		SHLD CRJD	¡Almacénalo en el conteo de renglón del manipulador derecho.
038B	CDB704		CALL JNOVD	¡Llama a la subrutina de direccionamiento E de los manipuladores.
038E	3614		MVI M,14H	¡Mueve a memoria el código de la paleta derecha.
0390	11A000		LXI D,00A0H	¡Carga en D-E, el valor que corresponde a bajar dos renglones.
0393	19		DAD D	¡Suma el contenido de H-L y el de D-E.
0394	3620		MVI M,20H	¡Mueve a memoria el código del espacio en blanco.
0396	3A1420	MUEVE:	LDA MOVPE	¡Carga el código de la dirección del movimiento de la pelota.
0399	210C07		LXI H,TABO2	¡Carga en H-L, la dirección del inicio de la tabla TABO2.
039C	110000		LXI D,0000H	¡Carga 0000H en D-E.
039F	5F		MOV E,A	¡Mueve el contenido de A a E.
03A0	19		DAD D	¡Suma H-L con D-E para obtener la dirección del movimiento de la pelota.

03A1	5E	MOV E,M	!Mueve el contenido de la tabla al registro E.
03A2	23	INX H	!Incrementa la dirección de la tabla.
03A3	56	MOV D,M	!Mueve el contenido de la tabla al registro D.
03A4	EB	XCHG	!Intercambia contenidos entre registros D-E y H-L.
03A5	ER	PCHL	!Carga al Contador de Programa con la direc- ción del movimiento a seguir.

↑
 Rutina durante el juego del movimiento
 ascendente a la izquierda
 ↓

03A6	C00304	CNT05: CALL VEIF1	!Llama a la subrutina de verificación 1 de la posición de la pelota.
03A9	CDF001	CALL AUX11	!Llama a la subrutina auxiliar AUX11.
03AC	2A0020	ASCIZ: LHLD CRPE	!Carga en H-L, el conte- nido de CRPE.
03AF	3E00	MVI A,00H	!Mueve 00H a A.
03B1	BD	CMF L	!Comparalo con el conte- nido de L.
03B2	C2BC03	JNZ COMD1	!No son iguales, salta a :COMD1.
03B5	EC	CMF H	!Comparalo con el conte- nido de H.
03B6	C2BC03	JNZ COMD1	!No son iguales, salta a :COMD1.
03B9	C3BA04	JMP REB05	!Si, salta a REB05.
03BC	3A1020	COMD1: LDA CCPE	!Carga en A, el conteni- do del conteo de colum- na de la pelota.
03BF	FE00	CFI 00H	!Comparalo con 00H.
03C1	C2C703	JNZ NOPAS	!No son iguales, salta a :NOPAS.
03C4	C3B805	JMP ANOT1	!Si, salta a :ANOT1.
03C7	2A0020	NOPAS: LHLD CRPE	!Carga en H-L el conte- nido del conteo de renglón de la pelota.
03CA	1100FF	LXI D,FFDAH	!Carga -50H en D-E.
03CD	19	DAD D	!Sumalo con el conteni- do de CRPE.
03CE	230020	SHLD CRPE	!El resultado almacenalo en el conteo de renglón de la pelota.
03D1	211020	LXI H,CCPE	!Carga en H-L, la direc- ción del conteo de colum- na de la pelota.
03D4	35	DCR H	!Decrementa el conteo de columna de la pelota.

03D5	CDF01	CALL AUXL2	;Llama a la subrutina ;auxiliar AUXL2.
03D8	3E00	MVI A,00H	;Mueve el código del mo- ;vimiento ascendente a ;la izquierda de la pelot ;ta durante el juego.
03DA	321420	STA MOVPE	;Almacena en la direc- ;cion MOVPE.
03DD	C9	RET	;Regresa.
;			
RUTINA DURANTE EL JUEGO DEL MOVIMIENTO			
DESCENDENTE A LA IZQUIERDA			
;			
03DE	C0D304	CNT06: CALL VERF1	;Llama a la subrutina de ;verificación 1 de la ;posición de la pelota.
03E1	C0F301	CALL AUXL1	;Llama a la subrutina ;auxiliar AUXL1.
03E4	2A0020	DESIZ: LHLD CRPE	;Carga en H-L, el conte- ;nido de CRPE.
03E7	3E50	MVI A,50H	;Mueve 50H a A.
03E9	BD	CMF L	;Compara lo con el conte- ;nido de L.
03EA	C2F603	JNZ COM02	;No son iguales, salta a ;COM02.
03ED	3E0F	MVI A,0FH	;Mueve 0FH a A.
03EF	BC	CMF H	;Compara lo con el conte- ;nido de H.
03F0	C2F603	JNZ COM02	;No son iguales, salta a ;COM02.
03F3	C39304	JMP RE006	;Si, salta a RE006.
03F6	3A1020	COM02: LDA CCPE	;Carga en A, el conteni- ;do del conteo de colum- ;na de la pelota.
03F9	FE00	CPI 00H	;Compara lo con 00H.
03FB	C20104	JNZ NOPA6	;No son iguales, salta a ;NOPA6.
03FE	C38B05	JMP ANDT1	;Si, salta a ANDT1.
0401	2A0020	NOPA6: LHLD CRPE	;Carga en H-L el conte- ;nido del conteo de rengl- ;ón de la pelota.
0404	115000	LXI D,0050H	;Carga 50H en D-E.
0407	19	DAD D	;Sumalo con el contenido ;de CRPE.
0408	220020	SHLD CRPE	;El resultado almacénalo ;en el conteo de renglón ;de la pelota.
040B	211020	LXI H,CCPE	;Carga en H-L la direc- ;cion del conteo de col- ;umna de la pelota.
040E	35	DCR M	;Decrementa el conteo de ;columna de la pelota.
040F	CDF01	CALL AUXL2	;Llama a la subrutina ;auxiliar AUXL2.
0412	3E02	MVI A,02H	;Mueve el código del mo-

			;	movimiento descendente a
			;	la izquierda de la pelg
			;	ta durante el juego.
0414	321420	STA MOVPE	;	Almacénalo en la direc-
0417	C9	RET	;	ción MOVPE.
			;	Regresa.
			;	
			;	RUTINA DURANTE EL JUEGO DEL MOVIMIENTO
			;	ASCENDENTE A LA DERECHA
			;	
0418	CD2F05	CNT07: CALL VERF2	;	Llama a la subrutina de
			;	verificación 2 de la
			;	posición de la pelota.
041B	CDFF01	CALL AUXL1	;	Llama a la subrutina
			;	auxiliar AUXL1.
041E	2A0020	ASCDE: LHLD CRPE	;	Carga en H-L el conte-
			;	nido de CRPE.
0421	3E00	MVI A,00H	;	Mueve 00H a A.
0423	0D	CMP L	;	Compara lo con el conte-
			;	nido de L.
0424	C22E04	JNZ COM03	;	No son iguales, salta a
			;	COM03.
0427	BC	CMP H	;	Compara lo con el conte-
			;	nido de H.
0428	C22E04	JNZ COM03	;	No son iguales, salta a
			;	COM03.
042B	C39C04	JMP REB07	;	Si, salta a REB07.
042E	3A1020	COM03: LDA CCPE	;	Carga en A, el conteni-
			;	do del conteo de colum-
			;	na de la pelota.
0431	FE4F	CPI 4FH	;	Compara lo con 4FH.
0433	C23904	JNZ NOPA7	;	No son iguales, salta a
			;	NOPA7.
0436	C30F06	JMP AN0T2	;	Si, salta a AN0T2.
0439	2A0020	NOPA7: LHLD CRPE	;	Carga en H-L el conte-
			;	nido del conteo de rengl-
			;	on de la pelota.
043C	1180FF	LXI D,FFB0H	;	Carga -50H en D-E.
043F	19	DAD D	;	Sumalo con el conteni-
			;	do de CRPE.
0440	220020	SHLD CRPE	;	El resultado almacénalo
			;	en el conteo de renglón
			;	de la pelota.
0443	211020	LXI H,CCPE	;	Carga en H-L la direc-
			;	ción del conteo de col-
			;	umna de la pelota.
0446	34	INR H	;	Incrementa el conteo de
			;	columna de la pelota.
0447	CDFF01	CALL AUXL2	;	Llama a la subrutina
			;	auxiliar AUXL2.
044A	3E04	MVI A,04H	;	Mueve el código del mo-
			;	vimiento ascendente a
			;	la derecha de la pelota
			;	durante el juego.
044C	321420	STA MOVPE	;	Almacénalo en la direc-

044F	C9	RET	ccion MOVPE. ;Regresa.
		;	
		RUTINA DURANTE EL JUEGO DEL MOVIMIENTO DESCENDENTE A LA DERECHA	
		;	
0450	CD2F05	CNT08: CALL VERF2	;Llama a la subrutina de ;verificación 2 de la ;posición de la pelota.
0453	CDF301	CALL AUXL1	;Llama a la subrutina ;auxiliar AUXL1.
0456	2A0020	DESDE: LHLD CRPE	;Carga en H-L, el conte- ;nido de CRPE.
0459	3E50	MVI A,50H	;Mueve 50H a A.
045B	BD	CMP L	;Compara lo con el conte- ;nido de L.
045C	C26004	JNZ COM04	;No son iguales, salta a ;COM04.
045F	3E0F	MVI A,0FH	;Mueve 0FH a A.
0461	BC	CMP H	;Compara lo con el conte- ;nido de H.
0462	C26004	JNZ COM04	;No son iguales, salta a ;COM04.
0465	C3A504	JMP REB08	;Si, salta a REB08.
0468	3A1020	COM04: LDA CCPE	;Carga en A, el conten- ;ido del conteo de colum- ;na de la pelota.
046B	FE4F	CPI 4FH	;Compara lo con 4FH.
046D	C27304	JNZ NOPAB	;No son iguales, salta a ;NOPAB.
0470	C30F06	JMP ANOT2	;Si, salta a ANOT2.
0473	2A0020	NOPAB: LHLD CRPE	;Carga en H-L, el conte- ;nido del conteo de rengl- ;on de la pelota.
0476	115000	LXI D,0050H	;Carga 50H en D-E.
0479	19	DAD D	;Sumalo con el conten- ;ido de CRPE.
047A	220020	SHLD CRPE	;El resultado almacenalo ;en el conteo de renglón ;de la pelota.
047D	211020	LXI H,CCPE	;Carga en H-L la direc- ;ción del conteo de co- ;lumna de la pelota.
0480	34	INR M	;Incrementa el conteo de ;columna de la pelota.
0481	CDF301	CALL AUXL2	;Llama a la subrutina ;auxiliar AUXL2.
0484	3E06	MVI A,06H	;Mueve el código del mo- ;vimiento descendente a ;la derecha de la pelota ;durante el juego.
0486	321420	STA MOVPE	;Almacenalo en la direc- ;ción MOVPE.
0489	C9	RET	;Regresa.
		;	

SUBROUTINA DE REBOTE 5

```

049A   CD1102   REB05: CALL AUXL3   ;Llama a la subrutina
                                ;auxiliar AUXL3.
049D   CA9B05   JZ ANOT1             ;Son iguales, salta a
                                ;ANOT1.
0490   C3E403   JMP DESI2           ;No, salta a DESI2.

```

SUBROUTINA DE REBOTE 6

```

0473   CD1102   REB06: CALL AUXL3   ;Llama a la subrutina
                                ;auxiliar AUXL3.
0496   CA9B05   JZ ANOT1             ;Son iguales, salta a
                                ;ANOT1.
0499   C5AC03   JMP ASCI2           ;No, salta a ASCI2.

```

SUBROUTINA DE REBOTE 7

```

049C   CD1702   REB07: CALL AUXL4   ;Llama a la subrutina
                                ;auxiliar AUXL4.
049F   CA9F06   JZ ANOT2             ;Son iguales, salta a
                                ;ANOT2.
04A2   C35604   JMP DESDE           ;No, salta a DESDE.

```

SUBROUTINA DE REBOTE 8

```

04A5   CD1702   REB08: CALL AUXL4   ;Llama a la subrutina
                                ;auxiliar AUXL4.
04A9   CA9F06   JZ ANOT2             ;Son iguales, salta a
                                ;ANOT2.
04AB   C31E04   JMP ASCDE           ;No, salta a ASCDE.

```

SUBROUTINA DE DIRECCIONAMIENTO 1
DE LOS MANIPULADORES

```

04AE   110080   JMOV1: LXI D,8000H   ;Carga en D-E, 8000H.
04B1   19       DAD D               ;Suma los contenidos de
                                ;H-L y D-E.
04B2   110200   LXI D,0002H        ;Carga en D-E, el número
                                ;de columna que usara la
                                ;paleta izquierda.
04B5   19       DAD D               ;Suma los contenidos de
                                ;H-L y D-E, guardando el
                                ;resultado en H-L.
04B6   C9       RET               ;Regresa.

```

SUBROUTINA DE DIRECCIONAMIENTO 2
DE LOS MANIPULADORES

```

04B7   110080   JMOV2: LXI D,8000H   ;Carga en D-E, 8000H.
04BA   19       DAD D               ;Suma los contenidos de
                                ;H-L y D-E.
04BB   114D00   LXI D,004DH        ;Carga en D-E, el número
                                ;de columna que usará la
                                ;paleta derecha.

```

04BE	19	DAD D	¡Suma los contenidos de H-L y D-E, aumentando el resultado en H-L.
04BF	C9	RET	¡Regresa.
		↓	
		SUBROUTINA DE DIRECCIONAMIENTO DE LA PELOTA	
		↓	
04C0	1100B0	DIPA: LXI D,8000H	¡Carga en D-E, el valor 8000H.
04C3	2A0020	LHLD CPFE	¡Carga el conteo de ren- gión de la pelota en H-L.
04C6	17	DAD D	¡Suma los contenidos de H-L y D-E.
04C7	110000	LXI D,0000H	¡Carga en D-E, 0000H.
04CA	3A1020	LDA CCPE	¡Carga en A, el contenido del conteo de columna de la pelota.
04CD	5F	MOV E,A	¡Mueve el contenido del acumulador a E.
04CE	19	DAD D	¡Suma los contenidos de H-L con los de D-E.
04CF	220220	SHLD FEAN	¡Invierte la dirección de la pelota anterior.
04D2	C9	RET	¡Regresa.
		↓	
		SUBROUTINA DE LA VERIFICACION 1 DE LA POSICION DE LA PELOTA	
		↓	
04D3	3A1020	VERF1: LDA CCPE	¡Carga en A, el contenido del conteo de columna de la pelota.
04E6	FE05	CPI 05H	¡La pelota está en la columna 05H?
04DB	C22E05	JNZ DIME2	¡No, salta a DIME2, para continuar con el movimiento de la pelota.
04DB	2A0420	LHLD CRJI	¡Carga en H-L, el contenido del conteo de ren- gión de la pelota iz- quierda.
04DE	EB	XCHG	¡Intercambia los contenidos de H-L y E-L.
04DF	2A0020	LHLD CRPE	¡Carga en H-L, el contenido del conteo de ren- gión de la pelota.
04E2	7D	MOV A,L	¡Mueve el contenido de L al Acumulador.
04E3	BB	CMPE	¡Compara con el contenido del registro E.
04E4	C2EC04	JNZ DIME1	¡No son iguales, salta a DIME1.
04E7	7C	MOV A,H	¡Si, mueve el contenido de H al acumulador.

04E3	BA		CMF D	:Compara lo con el contenido del registro D.
04E9	CAF004		JZ CHRE1	:Son iguales, salta a CHRE1.
04EC	210000	DIME1:	LAI H,0050H	:Carga en H-L, 0050H.
04EF	19		DAD D	:Suma los contenidos de H-L y L-R.
04F0	EB		XCHG	:Interchangea contenidos entre H-L y D-E.
04F1	2A0020		LHLD CAFE	:Carga en H-L, el contenido del conteo de region de la pelota.
04F4	7D		MOV A,L	:Mueve el contenido de L al acumulador.
04F5	BE		CMF A	:Compara lo con el contenido de L.
04F6	C22E05		JNZ DIME2	:No son iguales, salta a DIME2.
04F9	7C		MOV A,H	:Mueve el contenido de H al Acumulador.
04FA	BA		CMF D	:Compara lo con el contenido de D.
04FD	C22E05		JNZ DIME2	:No son iguales, salta a DIME2.
04FC	3E00		:MVI A,00H	:Mueve a A, 00H.
0500	BD	CHRE1:	CMF L	:Compara lo con el contenido de L.
0501	C2000B		JNZ CHUR1	:No, salta a CHUR1.
0504	BC		CMF H	:Compara lo con el contenido de H.
0505	CA5004		JZ CNT0B	:Son iguales, cambia el movimiento de la pelota al descendente-derecha.
0508	3E50		:MVI A,50H	:Mueve a A, 50H.
050A	BD	CHUR1:	CMF L	:Compara lo con el contenido de L.
050E	C21405		JNZ CHAD1	:No son iguales, salta a CHAD1.
050E	30CF		MVI A,0FH	:Mueve a A, 0FH.
0510	BC		CMF H	:Compara lo con el contenido de H.
0511	CA1604		JZ CNT07	:Son iguales, cambia el movimiento de la pelota al ascendente-derecha.
0514	3A0D20		:LDA MAIZ	:Carga en A, el contenido de MAIZ.
0517	FE02		CPI 02H	:Se movió para arriba?
0519	CA1604		JZ CNT07	:Sí, cambia el movimiento de la pelota al ascendente-derecha.
051C	FE08		:CPI 08H	:Se movió para abajo?

051E	CA5004	JZ CNT08	¿Si, cambia el movimiento de la pelota al descendente-derecha.
0521	3A1420	↓ LDA MOVPE	¿Carga en A, el contenido de MOVPE.
0524	FE00	CPI 00H	¿La pelota tiene un movimiento ascendente-izquierdo?
0526	CA5004	JZ CNT08	¿Si, cambia el movimiento de la pelota al descendente derecha.
0529	FE02	↓ CPI 02H	¿La pelota tiene un movimiento descendente-izquierda?
052B	CA1804	JZ CNT07	¿Si, cambialo por el movimiento ascendente-derecha.
053E	C9	DIME2: RET	¿Regresa.
		↓ SUBROUTINA DE LA VERIFICACION 2 DE LA POSICION DE LA PELOTA	
053F	3A1020	↓ VERF2: LDA CCPE	¿Carga en A, el contenido del conteo de columna de la pelota.
0532	FE4C	CPI 4CH	¿La pelota está en la columna 4CH?
0534	C28A05	JNC DIME4	¿No, salta a DIME4, para continuar con el movimiento de la pelota.
0537	3A0620	LHLD CRJD	¿Carga en H-L, el contenido del conteo de posición de la pelota derecha.
053A	ED	XCHG	¿Intercambia los contenidos de H-L y D-E.
053B	2A0020	LHLD CRTE	¿Carga en H-L, el contenido del conteo de posición de la pelota.
053E	7D	MOV A,L	¿Mueve el contenido de L al Acumulador.
053F	BB	CMP E	¿Compara lo con el contenido del registro E.
0540	C24005	JNZ DIME3	¿No son iguales, salta a DIME3.
0543	7C	MOV A,H	¿Si, mueve el contenido de H al Acumulador.
0544	BA	CMP D	¿Compara lo con el contenido del registro D.
0545	CA5A05	JZ CHRE2	¿Son iguales, salta a CHRE2.
0548	215000	DIME3: LXI H,0050H	¿Carga en H-L, 0050H.
054B	19	DAD D	¿Suma los contenidos de

0540	EB	XCHG	¡H-L y D-E. ¡Intercambia contenidos entre H-L y D-E.
0540	2A0020	LHLD CRPE	¡Carga en H-L, el conte- nido del conteo de ren- glón de la pelota.
0550	7D	MOV A,L	¡Mueve el contenido de L al Acumulador.
0551	85	CMF E	¡Compara lo con el conte- nido de E.
0552	C2BA05	JNZ DIME4	¡No son iguales, salta a ¡DIME4.
0555	7C	MOV A,H	¡Mueve el contenido de H al Acumulador.
0556	54	CMF D	¡Compara lo con el conte- nido de D.
0557	C2BA05	JNZ DIME4	¡No son iguales, salta a ¡DIME4.
055A	3E00	CHRE2: MVI A,00H	¡Mueve a A, 00H.
055C	BD	CMF L	¡Compara lo con el conte- nido de L.
055D	C15405	JNZ CHUR2	¡No, salta a CHUR2.
0560	BC	CMF H	¡Compara lo con el conte- nido de H.
0561	CADE05	JZ CNT06	¡Son iguales, cambia el movimiento de la pelota al descendente-izquier- da.
0564	3E50	CHUR2: MVI A,50H	¡Mueve a A, 50H.
0566	BD	CMF L	¡Compara lo con el conte- nido de L.
0567	C27005	JNZ CHAD2	¡No son iguales, salta a ¡CHAD2.
056A	3E0F	MVI A,0FH	¡Mueve a A, 0FH.
056C	BC	CMF H	¡Compara lo con el conte- nido de H.
056D	CAR605	JZ CNT05	¡Son iguales, cambia el movimiento de la pelota al ascendente-izquier- da.
0570	3A0E20	CHAD2: LDA MADE	¡Carga en A, el conten- ido de MADE.
0573	FE02	CFI 02H	¡Se movió para arriba?
0575	CAR605	JZ CNT05	¡Si, cambia el movimien- to de la pelota al as- cendente-izquierda.
0576	FE08	CFI 08H	¡Se movió para abajo?
057A	CADE05	JZ CNT06	¡Si, cambia el movimien- to de la pelota al des- cendente-izquierda.

;

057D	3A1420	LDA MOVFE	;Carga en A, el contenido de MOVFE.
05E0	FE04	CFI 04H	;La pelota tiene un movimiento ascendente-derecho?
05E2	CADE03	JZ CNT05	;Si, cambia el movimiento de la pelota al descendente-izquierda.
05E5	FE06	CFI 06H	;La pelota tiene un movimiento descendente-derecho?
05E7	CA6603	JZ CNT05	;Si, cambialo por el movimiento ascendente-izquierda.
058A	C9	DIME4: RET	;Regresa.
		;	
		SUBROUTINA DE ANOTACION 1	
		;	
058B	3A0B20	ANOT1: LDA CONIZ	;Carga en A, el contenido del marcador izquierdo.
058E	3C	INR A	;Incrementa el contenido del marcador.
059F	FE04	CFI 04H	;Es igual a 04H?
0591	C2CC05	JNZ UNID1	;No, salta a UNID1.
0594	C606	ADI 06H	;Convierte del código hexadecimal al BCD.
0596	320B20	STA CONIZ	;Almacena en CONIZ.
		;	
0599	21F9B0	LXI H, 80F9H	;Carga la 1a. dirección del dígito de las decenas del marcador izquierdo en H-L.
059C	CD9306	CALL AMAR1	;Llama a la subrutina auxiliar 1 de los marcadores.
		;	
059F	21FA80	LXI H, 80FAH	;Carga la 2a. dirección del dígito de las decenas del marcador izquierdo en H-L.
05A2	CD9706	CALL AMAR2	;Llama a la subrutina auxiliar 2 de los marcadores.
		;	
05A5	21FB00	LXI H, 80FBH	;Carga la 1a. dirección del dígito de las unidades del marcador izquierdo en H-L.
05AB	CD9B06	CALL AMAR3	;Llama a la subrutina auxiliar 3 de los marcadores.
		;	
05AB	21FC80	LXI H, 80FCH	;Carga la 2a. dirección

05AE	CD9F06	CALL AMAR4		¡del dígito de las unidades del marcador izquierdo en H-L. ¡Llama a la subrutina auxiliar 4 de los marcadores.
05B1	2149B1	¡ LXI H, B147H		¡Carga la 3a. dirección del dígito de las decenas del marcador izquierdo en H-L.
05B4	CD1706	CALL AMAR5		¡Llama a la subrutina auxiliar 5 de los marcadores.
05B7	214AB1	¡ LXI H, B14AH		¡Carga la 4a. dirección del dígito de las decenas del marcador izquierdo en H-L.
05BA	CD1706	CALL AMAR6		¡Llama a la subrutina auxiliar 6 de los marcadores.
05BD	214BB1	¡ LXI H, B14BH		¡Carga la 5a. dirección del dígito de las unidades del marcador izquierdo en H-L.
05C0	CD1B06	CALL AMAR7		¡Llama a la subrutina auxiliar 7 de los marcadores.
05C3	214CB1	¡ LXI H, B14CH		¡Carga la 6a. dirección del dígito de las unidades del marcador izquierdo en H-L.
05C6	CD1F06	CALL AMAR8		¡Llama a la subrutina auxiliar 8 de los marcadores.
05C9	C30406	JMP CMIZQ		¡Salta a CMIZQ.
05CC	320B20	¡ UNID1: STA CONIZ		¡Almacénalo en CONIZ.
05CF	21F0B0	LXI H, B0F0H		¡Carga en H-L, el valor correspondiente al tercio de renglón de los marcadores.
05D2	320B20	SHLD RMARC		¡Almacénalo en RMARC.
05D5	3E0E	¡ UNID2: MVI A, 0BH		¡Carga la 1a. columna del marcador izquierdo.
05D7	320F20	STA CMARC		¡Almacénalo en CMARC.
05DA	CDB306	CALL ELQSI		¡Llama a la subrutina "el que sigue".
05DD	210F20	¡ LXI H, CMARC		¡Carga en H-L, la dirección del conteo de co-

05E0	34	INR M	;Incrementa el contenido del contador.
05E1	CDB306	CALL ELQSI	;Llama a la subrutina "el que sigue".
05E4	2A0820	LHLD RMARC	;Carga en H-L, el contenido del contador de renglón del marcador.
05E7	7D	MOV A,L	;Mueve el contenido de L al Acumulador.
05EB	FE40	CPI 40H	;Es el último renglón del marcador?
05EA	CAF605	JZ FUER1	;Si, salta a FUER1.
05ED	2140B1	LXI H,B140H	;Carga en H-L, el valor del 2o. renglón de los marcadores.
05F0	220820	SHLD RMARC	;Almacénalo en RMARC.
05F3	C3D505	JMP UNID2	;Salta a UNID2.
05F6	3A0B20	FUER1: LDA CONIZ	;Carga en A, el contenido del marcador izquierdo.
05F9	FE15	CPI 15H	;Compara lo con 15H.
05FB	C20406	JNZ CMIZQ	;No son iguales, salta a CMIZQ.
05FE	CDC306	CALL AULFN	;Llama a la subrutina auxiliar final.
0601	C30E06	JMP FINJ1	;Salta a FINJ1.
0604	3E20	CMIZQ: MVI A,20H	;Mueve el código del espacio en blanco al Acumulador.
0606	321220	STA CARAN	;Almacénalo en la dirección de memoria que contiene el carácter anterior.
0609	3E26	MVI A,26H	;Mueve el valor de la 1a. columna intermedia de inicio de movimiento a la izquierda.
060B	321020	STA CCPE	;Almacénalo en el conteo de columna de la posición.
060E	C9	FINJ1: RET	;Regresa.
		↓	
		SUBROUTINA DE ANOTACION 2	
		↓	
060F	3A0C20	ANOT2: LDA CONDE	;Carga en A, el contenido del marcador derecho.
0612	3C	INR A	;Incrementa el contenido del marcador.

0613	FE0A	CFI 0AH	¡Es igual a 0AH?
0615	C25006	JNZ UNIDS	¡No, salta a UNIDS.
0618	C606	ADI 06H	¡Convierte del código hexadecimal al BCD.
061A	320C20	STA CONDE	¡Almacénalo en CONDE.
061D	215791	↓ LXI H, 2157H	¡Carga la 1a. dirección del dígito de las decenas del marcador derecho en H-L.
0620	CD9306	CALL AMAR1	¡Llama a la subrutina auxiliar 1 de los marcadores.
0623	215981	↓ LXI H, 2159H	¡Carga la 2a. dirección del dígito de las decenas del marcador derecho en H-L.
0626	CD9706	CALL AMAR2	¡Llama a la subrutina auxiliar 2 de los marcadores.
0629	215981	↓ LXI H, 2159H	¡Carga la 1a. dirección del dígito de las unidades del marcador derecho en H-L.
062C	CD9B06	CALL AMAR3	¡Llama a la subrutina auxiliar 3 de los marcadores.
062F	215A81	↓ LXI H, 215AH	¡Carga la 2a. dirección del dígito de las unidades del marcador derecho en H-L.
0632	CD9F06	CALL AMAR4	¡Llama a la subrutina auxiliar 4 de los marcadores.
0635	21A781	↓ LXI H, 21A7H	¡Carga la 3a. dirección del dígito de las decenas del marcador derecho en H-L.
0638	CDA306	CALL AMAR5	¡Llama a la subrutina auxiliar 5 de los marcadores.
063B	21A881	↓ LXI H, 21A8H	¡Carga la 4a. dirección del dígito de las decenas del marcador derecho en H-L.
063E	CDA706	CALL AMAR6	¡Llama a la subrutina auxiliar 6 de los marcadores.
0641	21A981	↓ LXI H, 21A9H	¡Carga la 5a. dirección

0644	CDAB06	CALL AMAR7	;del dígito de las unidades del marcador derecho en H-L. ;Llama a la subrutina auxiliar 7 de los marcadores.
0647	21AAB1	; LXI H,812AH	;Carga la 4a. dirección del dígito de las unidades del marcador derecho en H-L.
0644	CDAB06	CALL AMAR6	;Llama a la subrutina auxiliar 6 de los marcadores.
064D	C36B06	JMP CMDEF.	;Salta a CMDEF.
0650	320C20	UNID3: STA CONDE	;Almacénalo en CONDE.
0653	21F0B0	LXI H,80F0H	;Carga en H-L, el valor correspondiente al primer renglón de los marcadores.
0656	220B20	SHLD RMARC	;Almacénalo en RMARC.
0659	3E69	UNID4: MVI A,6FH	;Carga la 1a. columna del marcador derecho.
065B	320F20	STA CMARC	;Almacénalo en CMARC.
065C	CDB306	CALL ELQSI	;Llama a la subrutina "el que sigue".
0661	210F20	; LXI H,CMARC	;Carga en H-L, la dirección del conteo de columna del marcador.
0664	34	INR M	;Incrementa en uno el contenido del conteo de columna del marcador.
0665	CDB306	CALL ELQSI	;Llama a la subrutina "el que sigue".
066B	2ACB20	; LHLD RMARC	;Carga en H-L, el contenido del conteo de renglón del marcador.
066B	7D	MOV A,L	;Mueve el contenido de L al Acumulador.
066C	FE40	CPI 40H	;Es el último renglón del marcador?
066E	CA7A06	JZ FUER2	;Si, salta a FUER2.
0671	2140B1	LXI H,8140H	;Carga en H-L, el valor del 2o. renglón de los marcadores.
0674	220B20	SHLD RMARC	;Almacénalo en RMARC.
0677	C35906	JMP UNID4	;Salta a UNID4.
067A	3A0C20	FUER2: LDA CONDE	;Carga en A, el contenido del marcador derecho.

067D	FE15	CPI 15H	;Compara lo con 15H.
067F	028806	JNZ CMDBR	;No son iguales, salta a ;CMDBR.
0682	000306	CALL AXLFN	;Llama a la subrutina ;auxiliar final.
0685	000206	JMP FINJ2	;Salta a FINJ2.
0688	3E20	CMDBR: MVI A,20H	;Mueve el código del es- ;pacio en blanco al Acu- ;mulador.
068A	321220	STA CAPAN	;Almacénalo en la direc- ;ción de memoria que con- ;tiene el carácter anter- ;rior.
068D	3E28	MVI A,28H	;Mueve el valor de la ;la. columna intermedia ;Je inicio de movimiento ;a la derecha.
068F	321020	STA CCPE	;Almacénalo en el conteo ;de columna de la pelot- ;ta.
0692	C9	FINJ2: RET	;Regresa.
		↓	
		SUBROUTINA AUXILIAR 1 DE LOS MARCADORES	
		↓	
0693	3E15	AMAR1: MVI A,25H	;Carga el 1er. carácter ;del dígito 1 en A.
0695	77	MOV M,A	;Muevelo a memoria.
0696	C9	RET	;Regresa.
		↓	
		SUBROUTINA AUXILIAR 2 DE LOS MARCADORES	
		↓	
0697	3E26	AMAR2: MVI A,26H	;Carga el 2o. carácter ;del dígito 1 en A.
0699	77	MOV M,A	;Muevelo a memoria.
069A	C9	RET	;Regresa.
		↓	
		SUBROUTINA AUXILIAR 3 DE LOS MARCADORES	
		↓	
069B	3E21	AMAR3: MVI A,21H	;Carga el 1er. carácter ;del dígito 0 en A.
069D	77	MOV M,A	;Muevelo a memoria.
069E	C9	RET	;Regresa.
		↓	
		SUBROUTINA AUXILIAR 4 DE LOS MARCADORES	
		↓	
069F	3E23	AMAR4: MVI A,23H	;Carga el 2o. carácter ;del dígito 0 en A.
06A1	77	MOV M,A	;Muevelo a memoria.
06A2	C9	RET	;Regresa.

			↓	
			SUBROUTINA AUXILIAR 2 DE LOS	
			MARCADORES	
			↓	
06A5	3E27	AMAR5:	MVI A, 27H	!Carga el 3er. caracter
				!del dígito 1 en A.
06A5	77		MOV M, A	!Muevelo a memoria.
06A5	C9		RET	!Regresa.
			↓	
			SUBROUTINA AUXILIAR 3 DE LOS	
			MARCADORES	
			↓	
06A7	3E28	AMAR6:	MVI A, 28H	!Carga el 4o. caracter
				!del dígito 1 en A.
06A7	77		MOV M, A	!Muevelo a memoria.
06A7	C9		RET	!Regresa.
			↓	
			SUBROUTINA AUXILIAR 4 DE LOS	
			MARCADORES	
			↓	
06AB	3E2D	AMAR7:	MVI A, 2DH	!Carga el 3er. caracter
				!del dígito 0 en A.
06AD	77		MOV M, A	!Muevelo a memoria.
06AE	C9		RET	!Regresa.
			↓	
			SUBROUTINA AUXILIAR 5 DE LOS	
			MARCADORES	
			↓	
06AF	3E24	AMAR8:	MVI A, 24H	!Carga el 4o. caracter
				!del dígito 0 en A.
06B1	77		MOV M, A	!Muevelo a memoria.
06B2	C9		RET	!Regresa.
			↓	
			SUBROUTINA EL QUE SIGUE	
			↓	
06B3	2A0B20	EL051:	LHLD BMARC	!Carga en H-L, el contenido
				!del conteo de memoria del
				!marcador.
06B6	110000		LXI D, 0000H	!Carga en D-E, el valor
				!0000H.
06B9	3A0F20		LDA CMARC	!Carga en A, el contenido
				!del conteo de columna del
				!marcador.
06BC	5F		MOV E, A	!Muevelo a E.
06BD	19		DAD D	!Suma los contenidos de
				!H-L con los de D-E.
06BE	7E		MOV A, M	!Mueve el contenido de
				!memoria direccionado
				!por H-L a A.
06BF	C604		ADI 04H	!Sumale 04H para incre-
				!mentar en uno al dígito
				!de las unidades del mar-
				!cador.
06C1	77		MOV M, A	!Regresalo a la locali-
				!dad de memoria ya direc-

06C2	C9	RET	ccionada. ;Regres.
		↓ SUBROUTINA AUXILIAR FINAL	
		↓	
06C3	3E00	AXLFI: MVI A,00H	;Mueve a A, 00H.
06C5	320A20	STA JUEGO	;Almacena en la bandera de juego.
		↓	
06C8	2A0420	LHLD CRJ1	;Carga en H-L, el contenido del conteo de rotación de la paleta izquierda.
06C5	CDAE04	CALL JMOV1	;Llama a la subrutina de direccionamiento 1 de los manipuladores.
06CE	3620	MVI M,20H	;Mueve el código de espacio a memoria.
06D0	2A0420	LHLD CRJ1	;Carga en H-L, el contenido del conteo de rotación del manipulador izquierdo.
06D3	115000	LXI D,0050H	;Carga en D-E, el número 0050H.
06D6	19	DAD D	;Suma los contenidos de H-L con los de D-E.
06D7	CDAE04	CALL JMOV1	;Llama a la subrutina de direccionamiento 1 de los manipuladores.
06DA	3620	MVI M,20H	;Mueve el código de espacio a memoria.
		↓	
06DC	2A0620	LHLD CRJD	;Carga en H-L, el contenido del conteo de rotación de la paleta derecha.
06DF	CDB704	CALL JMOV2	;Llama a la subrutina de direccionamiento 2 de los manipuladores.
06E2	3620	MVI M,20H	;Mueve a memoria el código de espacio.
06E1	2A0620	LHLD CRJD	;Carga en H-L, el contenido del conteo de rotación del manipulador derecho.
06E7	115000	LXI D,0050H	;Carga en D-E, el valor 0050H.
06EA	19	DAD D	;Suma los contenidos de H-L y D-E.
06EB	CDB704	CALL JMOV2	;Llama a la subrutina de direccionamiento 2 de los manipuladores.
06EE	3620	MVI M,20H	;Mueve a memoria el código de espacio.

;

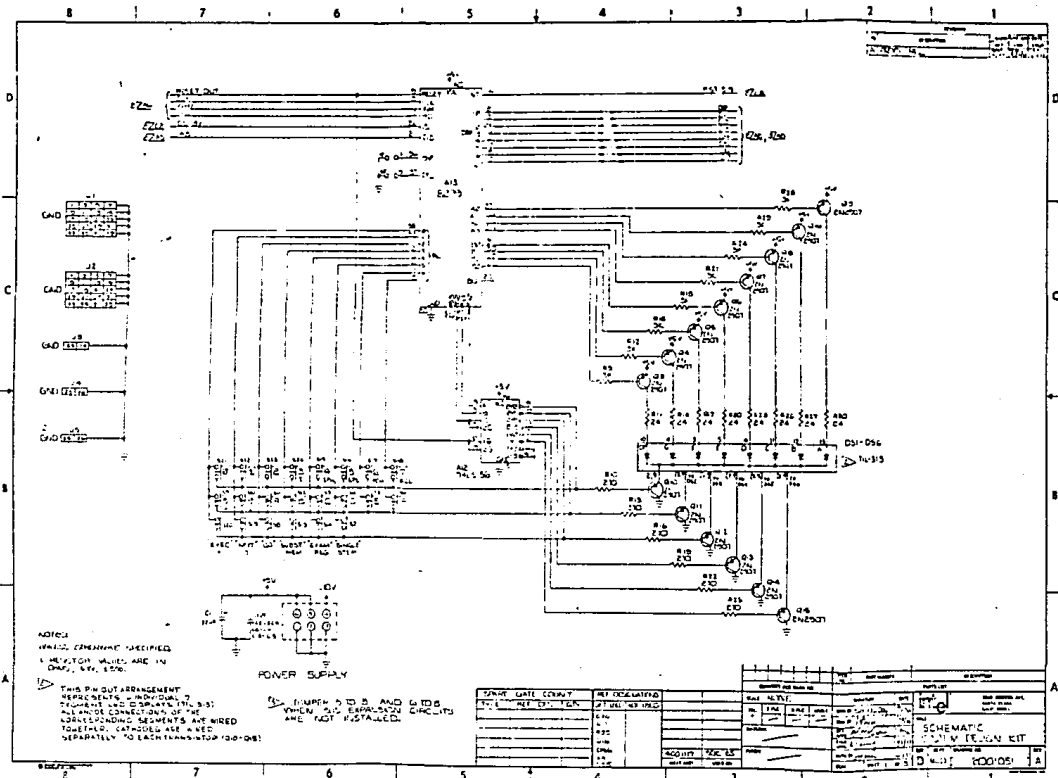
06F0	3E26	MVI A,26H	;Mueve a A, el valor ;26H.
06F2	321020	STA CCPE	;Almacénalo en el conteo ;de columna de la pelota.
06F5	CDC004	CALL DIFA	;Llama a la subrutina de ;direccionamiento de la ;posición de la pelota.
06F8	3E10	MVI A,10H	;Mueve a A, el código de ;la pelota.
06FA	2A0220	LHLD PEAN	;Carga en H-L, el conteo ;de PEAN.
06FD	77	MOV M,A	;Mueve a memoria el con- ;tenido del Acumulador.
06FE	3E20	MVI A,20H	;Mueve a A, el código de ;espacio.
0700	321220	STA CARAN	;Almacénalo en CARAN.
0703	C9	RET	;Regresa.

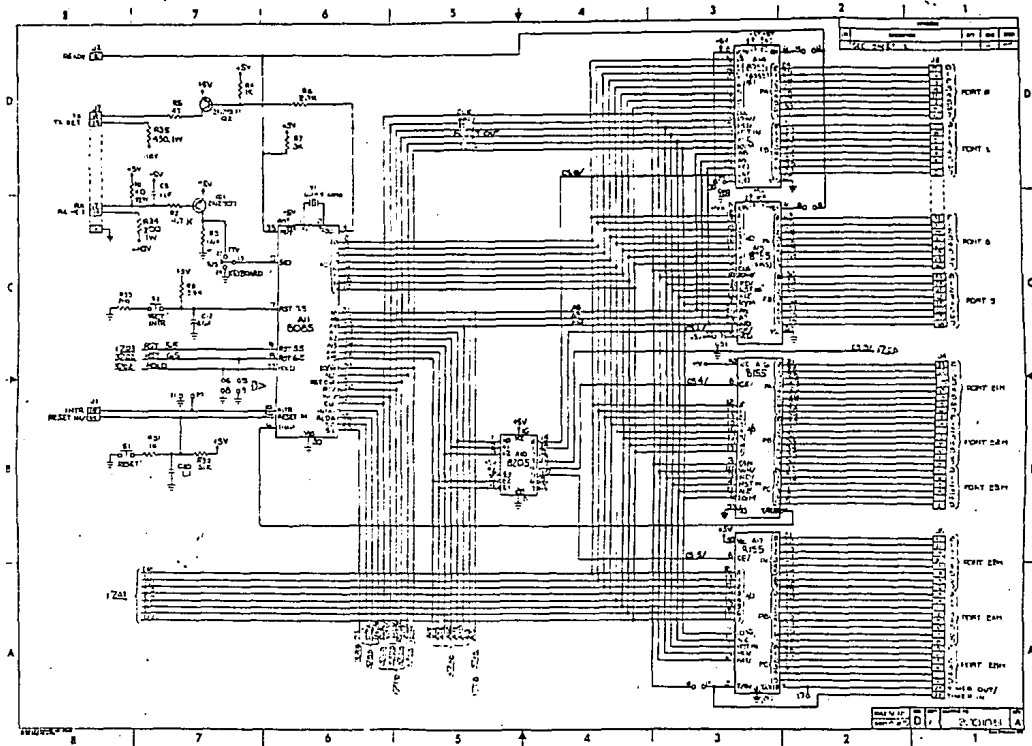
↓
↓
↓
TABLA DE LOS MOVIMIENTOS DURANTE
LA DEMOSTRACION

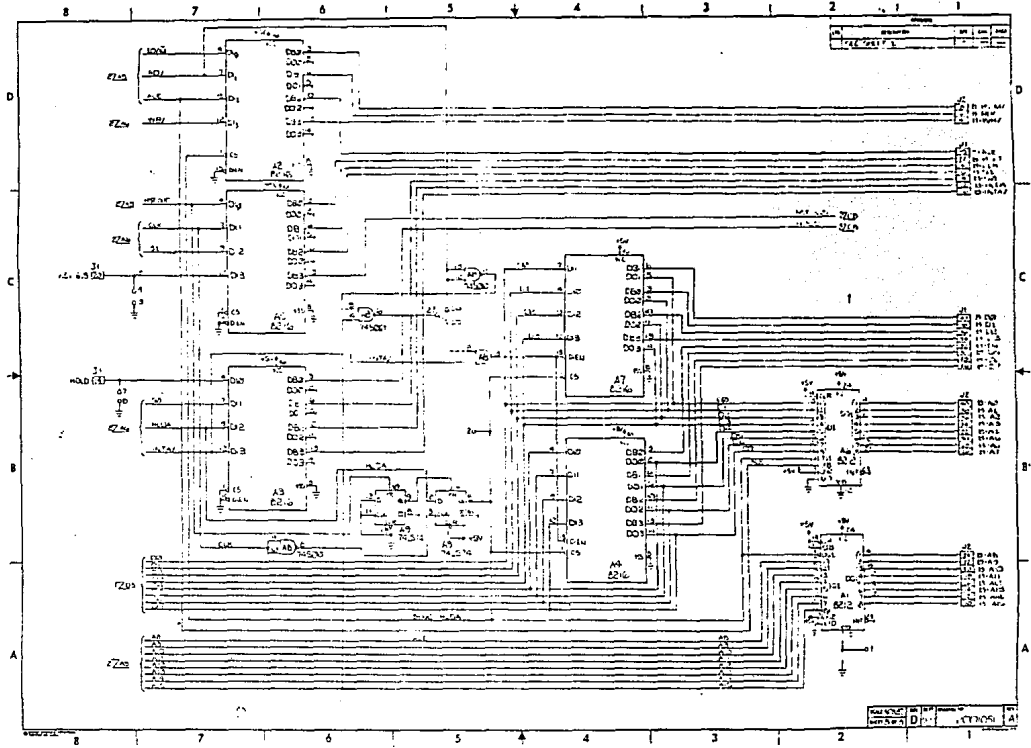
0704	F700	TAB01: DW CNT01
0706	2C01	DW CNT02
0708	6301	DW CNT03
070A	9B01	DW CNT04

↓
↓
↓
TABLA DE LOS MOVIMIENTOS DURANTE
EL JUEGO

070C	A603	TAB02: DW CNT05
070E	DE03	DW CNT06
0710	1804	DW CNT07
0712	5004	DW CNT08







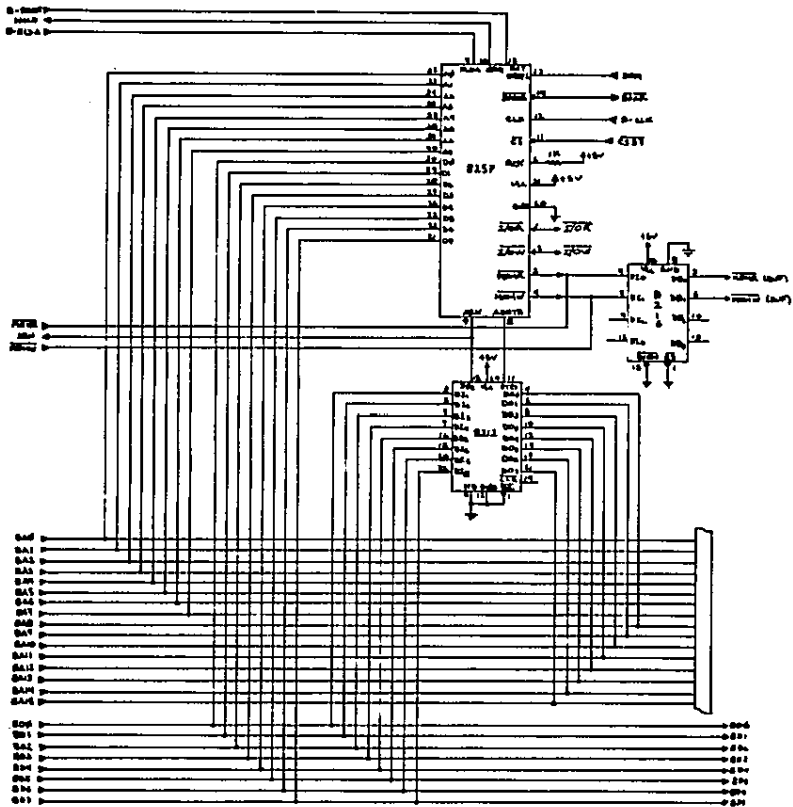
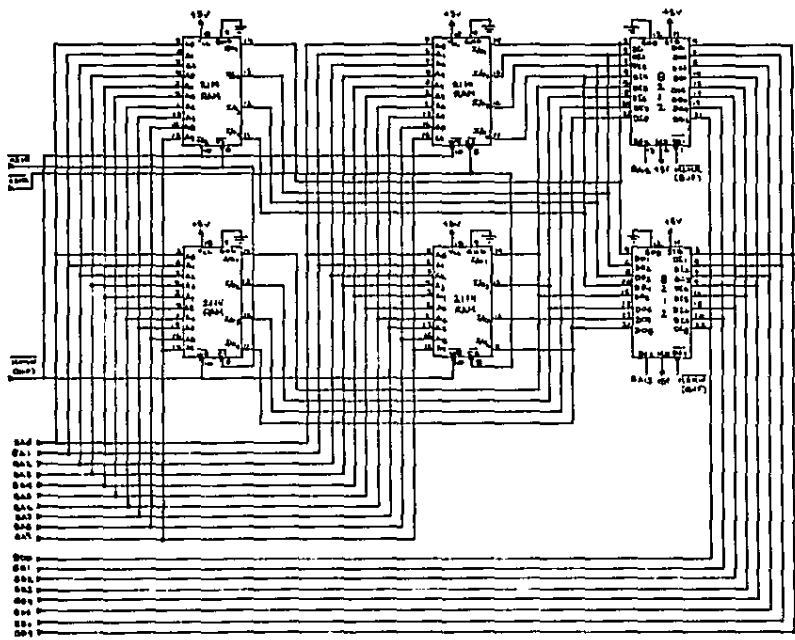
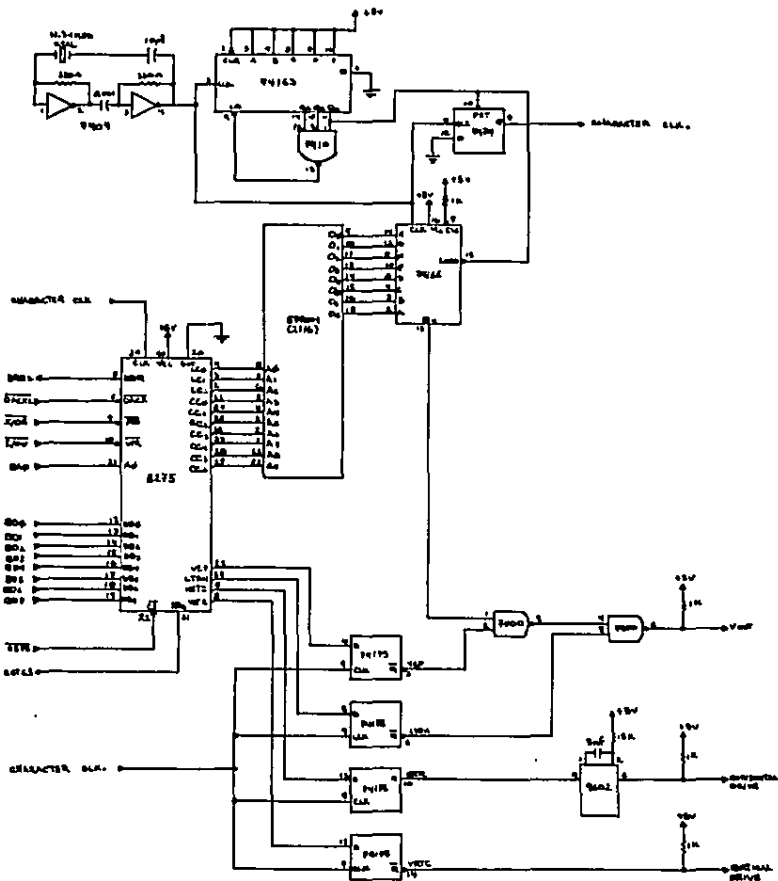
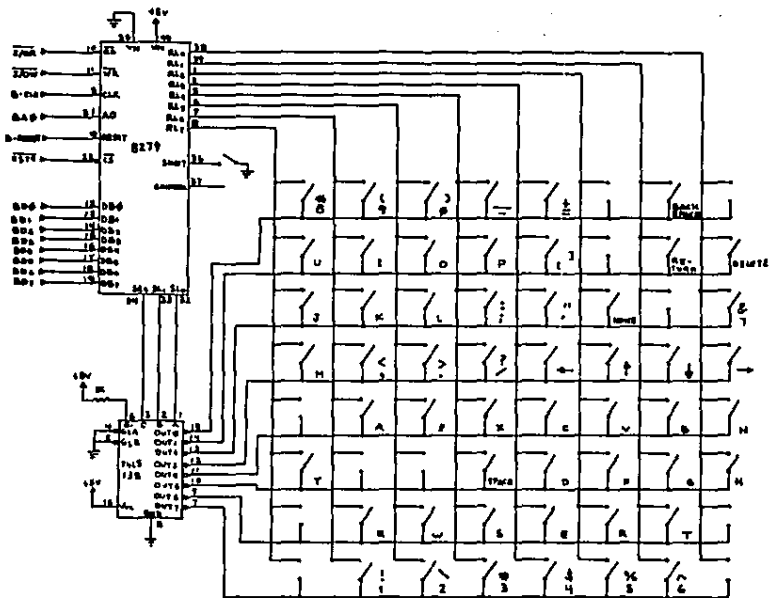
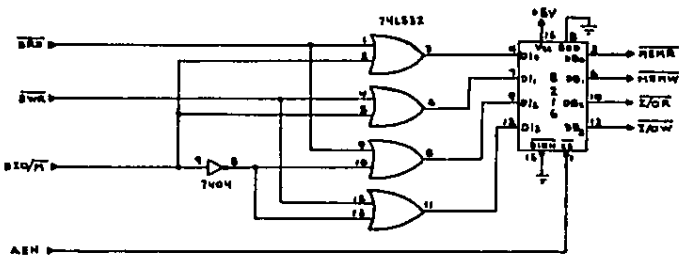
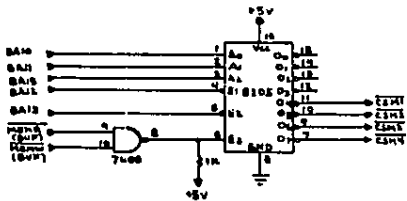
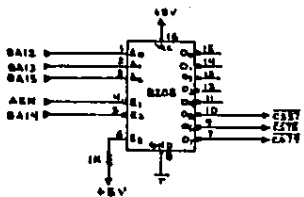


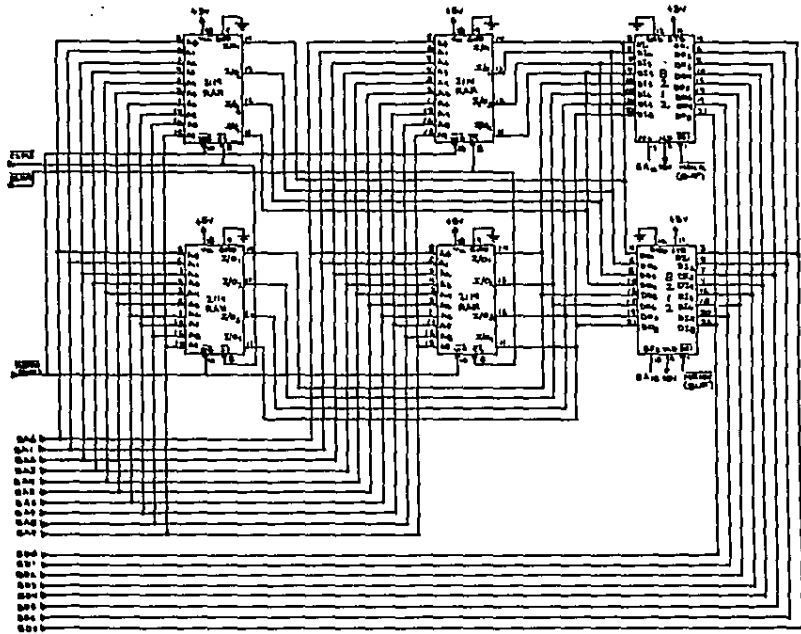
Diagram illustrating the wiring connections between various components, including inputs (B-2007, B-2008, B-2009, B201-B210) and outputs (B171-B180, B211-B220), and a transformer-like component.











BIBLIOGRAFIA

- BOBBA E INTERFACES
Aut. Octavio F. Garcia Nardcia.
- PROGRAMACION BOBO Y BOBS
Aut. Octavio F. Garcia Nardcia.
- DATA BASE DESIGN
Aut. Gjo Wiederhold
Computer Science Series
Mc. Graw Hill Book Corp.
- THE TTL DATA BOOK FOR DESIGN ENGINEERS
Texas Instruments
- SDK-85 USER'S MANUAL
Intel
- THEORY AND PROBLEMS OF DIGITAL PRINCIPLES
Aut. Roger L. Tokheim
Mc. Graw Hill
- PRINCIPIOS DE ELECTRONICA
Aut. Albert Paul Malvino
Mc. Graw Hill
- THE ART OF DIGITAL DESIGN
Aut.s David Winkel-Franklin Prosser
Frontice Hall Inc.
- MODULOS DIDACTICOS DE CONVERTIDORES
DIGITAL-ANALOGICO, ANALOGICO DIGITAL
Tesis Profesional de Guillermo Gerardo Ibarra Gonzalez
ULSA 1985.
- MICROPROCESSORS AND DIGITAL SYSTEMS
Aut. Douglas V. Hall
Mc. Graw Hill
- MICROSYSTEM COMPONENTS HANDBOOK
VOL 1, VOL2
Intel
- LOGIC DATA BOOK
National Semiconductors

-PROGRAMMING TECHNIQUES (SIMULATION)

VOL. 2

Byte Books

Blaise W. Liffick, Editor

-CIRCUITOS DIGITALES Y MICROPROCESADORES

Herbert Taub

Mc. Graw Hill