

300 617
47
2ei



UNIVERSIDAD LA SALLE

ESCUELA DE INGENIERIA
INCORPORADA A LA U. N. A. M.

MICROTERMINAL INTELIGENTE CON EL MICROPROCESADOR 8085A

T E S I S
QUE PARA OBTENER EL TITULO DE
INGENIERO MECANICO ELECTRICO
P R E S E N T A :
EDUARDO ANTONIO PINO DOMINGUEZ

México, D. F.

1993

TESIS CON
FALLA DE ORIGEN



Universidad Nacional
Autónoma de México



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 está 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.

MICROTERMINAL INTELIGENTE CON EL
MICROPROCESADOR 8085A

T E M A R I O

I N T R O D U C C I O N

C A P I T U L O I

DESCRIPCION GENERAL

C A P I T U L O II

DISPOSITIVOS REQUERIDOS

- 2.1 CPU
- 2.2 MEMORIA
- 2.3 PUERTOS DE ENTRADA/SALIDA PARALELOS,
DISPLAY Y TECLADO
- 2.4 PUERTO DE ENTRADA/SALIDA SERIE
- 2.5 DISPOSITIVOS SELECCIONADOS

C A P I T U L O III

DESCRIPCION DE DISPOSITIVOS

- 3.1 MICROPROCESADOR INTEL 8085A
- 3.2 MEMORIA EPROM 8755A
- 3.3 MEMORIA RAM 8156
- 3.4 DISPLAYS MATRICIALES
- 3.5 TECLADO
- 3.6 CIRCUITOS DS14C88 Y DS14C89
- 3.7 ALIMENTACION DE VOLTAJE
- 3.8 INVERSOR 7406
- 3.9 MM74138 DECODIFICADOR DE LINEA DE 3 A 8

C A P I T U L O I V

DISEÑO DEL CIRCUITO ELECTRONICO

- 4.1 MICROCOMPUTADOR
- 4.2 DISPLAY MATRICIAL
- 4.3 TECLADO
- 4.4 FUENTE DE ALIMENTACION

C A P I T U L O V

INTERFACE DE COMUNICACION UTILIZADA

- 5.1 REGISTROS DEL NCS858
- 5.2 ACOPLAMIENTO AL MICROPROCESADOR
- 5.3 NIVELES RS232 Y TTL

C A P I T U L O V I

TEORIA DE FUNCIONAMIENTO

- 6.1 AUTODIAGNOSTICO
- 6.2 CONTROL DE LOS DISPLAYS
- 6.3 CONTROL DEL TECLADO
- 6.4 COMUNICACION AL COMPUTADOR

C O N C L U S I O N E S

A P E N D I C E S

- APENDICE A: CARACTERISTICAS DEL 8085A
- APENDICE B: CARACTERISTICAS DEL 8156 Y 8755A
- APENDICE C: NEMONICOS DEL 8085A
- APENDICE D: CARACTERISTICAS DEL DISPLAY MATRICIAL
- APENDICE E: CARACTERISTICAS DEL TECLADO
- APENDICE F: CARACTERISTICAS DEL DECODIFICADOR
MM74138
- APENDICE G: CARACTERISTICAS DEL LM123A Y 7406
- APENDICE H: CARACTERISTICAS DEL DS14C88 Y DS14C89
- APENDICE I: CODIGO ASCII
- APENDICE J: INTERFACE RS-422A

B I B L I O G R A F I A

I N T R O D U C C I O N

Es una realidad que la mayoría de las grandes empresas resuelven muchas de sus necesidades utilizando equipos de cómputo. Cada vez es más común encontrar terminales de video, computadores personales e impresoras en establecimientos y oficinas.

Sin embargo hay compañías o partes de ellas que necesitan consultar, procesar, etc., información en lugares remotos, esto es, fuera de las oficinas, como puede ser dentro de una planta petroquímica o eléctrica, la cual tenga todos sus procesos automatizados por computadora, y en la que los técnicos e ingenieros necesiten realizar pruebas y verificación del proceso en un punto determinado, lejos del centro de operaciones. En este ambiente las terminales de video, son la gran mayoría voluminosas y pesadas, para que el personal técnico las lleve consigo.

Es por lo anterior, que surgió la idea de diseñar el circuito para una terminal pequeña y portátil, que pueda enlazarse con la mayoría de los computadores, necesitando el tendido de un cable del puerto del computador central, a zonas claves, para que sólo se conecte la microterminal y se logre establecer la comunicación.

El usar los componentes estrictamente necesarios, no solo ayuda a ahorrar espacio dentro de la microterminal, sino también minimiza la posibilidad de un mal funcionamiento. El teclado es totalmente sellado, y como adentro de él, va implementado el circuito electrónico, éste queda totalmente protegido, no es grande ni estorboso; tomando en cuenta que es tan delgado como una mica, el orden en que se presentan las teclas, tienen la misma disposición de las de una máquina de escribir estándar, por lo que su manejo se facilita. La presentación de la información al usuario, se realiza por medio de displays matriciales, que tienen la suficiente capacidad para desplegar información, con una intensidad luminosa adecuada, y de esta forma poder leer los caracteres que en ellos aparezcan. En cuanto al circuito electrónico, se diseñó de una manera sencilla y dinámica, usando un microprocesador comercial, así como sus componentes periféricos.

Cabe mencionar que todos los elementos que en esta tesis se mencionan, se pueden encontrar en el mercado nacional.

CAPITULO I

DESCRIPCION GENERAL:

Entre los principales problemas, con los que se encuentra uno al estar diseñando alguna pieza, alguna idea, un sistema, etc., es el de poner en su debido orden todas y cada una de las piezas, que no falte nada, que la secuencia de funcionamiento sea la correcta. Pues bien, basándonos en este principio, nos apoyaremos durante toda esta tesis, en diagramas de bloque, pues es un método muy confiable. Se empezará con diagramas muy básicos, pero conforme se vaya avanzando, se irán detallando más cada uno de ellos, hasta llegar a un diagrama de bloque que nos simule cada uno de los pasos.

Para empezar a determinar los objetivos de la microterminal inteligente, empezaremos a describir la arquitectura básica, en torno a la cual gira su estructura.

Existen, en términos generales, dos tipos de terminales de computadoras, las denominadas "no inteligentes" y las "inteligentes". Las primeras son aquellas que el computador central las tiene que controlar totalmente, esto es, le manda toda la información lista, sólo para que la despliegue

y la estará monitoreando constantemente. Lo anterior con-
trae, que el computador tiene que emplear tiempo para estar al tanto
de ella, tiempo para procesar información que ésta tenga que
desplegar, decodificar la información proveniente del teclado
para su entendimiento y procesamiento, en fin, el computador
central tendrá que consumir recursos, que se reflejarán, en un
momento dado, en el tiempo de respuesta .

Las terminales inteligentes tienen la capacidad
de realizar ciertas funciones que le dan autonomía, esto es, el
computador central ya no le tiene que dedicar mucho tiempo,
pues en cierta forma la misma terminal se autocontrola. Por
ejemplo, el computador central puede enviar la información a
desplegar directamente, pues la microterminal podrá interpretar
esa información y convertirla a datos de despliegue, cada vez
que la terminal tenga que enviar información (si algo se ha
teclado), enviará la información de tal manera que el
computador central la interprete inmediatamente, esto es, en el
mismo formato que el computador trabaja.

De todo lo anterior podemos concluir que cada vez
que conectemos una terminal inteligente, el computador central
ahorra tiempo, mismo que tendría que invertir en una terminal
no-inteligente, pudiendo destinar ese tiempo a procesos que
realmente lo necesiten.

Es obvio que la estructura de una terminal inteli-

gente es más elaborada que una que no lo es. A continuación se describe de la arquitectura que logra que una terminal sea inteligente.

La figura 1 representa un microcomputador en su nivel más básico. El elemento principal sobre el cual se basa todo es la CPU, y es éste el que controla absolutamente todos los dispositivos acoplados en el sistema. La CPU (Unidad Central de Proceso) será el encargado de procesar toda la información.

Sin embargo la CPU necesita un lugar en donde pueda almacenar toda aquella información que necesite guardar, por lo que es necesario una memoria, la cual, ofrece al CPU la facilidad de grabar, leer y alterar la información que contenga, pues es imposible que la CPU procese datos y los almacene en sí mismo. Así pues utiliza la memoria como un almacén de datos, en los cuales alojará la información que va a ser procesada, la que ya fue, y temporalmente los resultados parciales. En resumen, toda información la CPU maneja, está almacenada en memoria o entra por el bloque de entrada/salida de datos, el orden de la información almacenada y todo es controlado por la CPU.

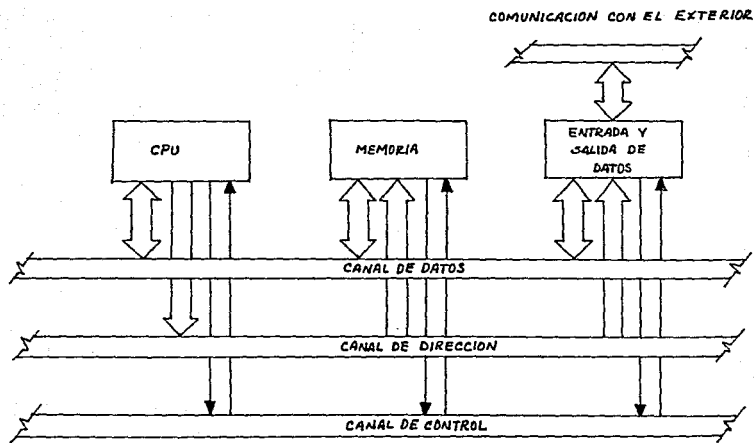


FIGURA 1

Toda la información que se ha procesado dentro del microcomputador tiene que entrar de alguna forma y es obvio que también tenga que salir, una vez procesada. El bloque de Entrada y Salida de datos, es por donde fluye la entrada y salida de información.

Todo microcomputador debe ser programado para cumplir la función específica para el cual está siendo empleado. Esta programación no forma parte en ningún momento del computador central con el que va a trabajar, por el contrario, dicha programación se almacenará en la memoria propia de la microterminal, que de ahora en adelante llamaremos sistema operativo. Esta terminal se puede acoplar a cualquier computador, siempre y cuando éste ofrezca una línea estándar de comunicación, lo cual se explica más adelante en esta tesis. El sistema operativo es el que le dirá al CPU, qué hacer con la información, en otras palabras es el "alma" del sistema; la CPU consultará incansablemente al sistema operativo cada vez que así se requiera.

Como podemos apreciar, la figura 1 se puede considerar como un modelo básico universal de un microcomputador, y este puede ser preprogramado para cumplir un fin específico mediante el sistema operativo. En esta tesis, el sistema operativo está diseñado para que el microcomputador se comporte como una interfaz inteligente, entre el usuario y el computador.

Hasta ahora se ha explicado de una manera muy general cuál es la relación entre la memoria y la CPU, cómo se debe el microcomputador comportar gracias al sistema operativo, y que gracias al mismo funcionará como una terminal, pero ¿cómo logrará el sistema la comunicación entre el usuario y la computadora?

Para poder lograr esa comunicación, se debe considerar un dispositivo que permita la entrada y salida de datos. En la figura 1 podemos observar un bloque, el cual ejemplifica la forma en que puede fluir la información hacia adentro y hacia afuera del sistema. Tal y como se aprecia, al igual que la memoria, está acoplado al CPU, esto asegura el absoluto control del ingreso y egreso de información, por lo que todo dato que se encuentre en el microcomputador estará plenamente identificado y registrado, así la probabilidad de pérdida de información es prácticamente nula.

La forma en que está comunicado cada bloque es muy importante, esta comunicación se lleva a cabo mediante 3 canales, mismos que se pueden apreciar en la figura 1, éstos son:

- a) canal de datos
- b) canal de dirección
- c) canal de control

Los tres canales actúan conjuntamente y por lo general uno es complemento del otro y todos están controlados por la CPU.

Para la CPU, la memoria y el control de entrada y salida de datos son dos elementos que están registrados bajo una cierta dirección, por lo que cada vez que necesita algo de ellos, los llama por medio del canal de dirección. Este canal cobra mayor importancia cuando, por ejemplo, la CPU necesita algún dato que se encuentra almacenado en la memoria o se quiera grabar en ella, ésta se divide internamente como un anaquel con varias divisiones y cada división tiene un número, pues bien, cada vez que la CPU necesite guardar o almacenar algún dato, habilitará la localidad que desea por medio del canal de dirección. Este canal es unidireccional para la CPU, pues es él, el que direcciona y nadie lo direcciona a él. También este canal es unidireccional para la memoria y para el bloque de entrada y salida de datos.

El canal de control permitirá que por él fluyan las señales que controlarán el proceso, mismas que darán sincronía al sistema; si la información que está en el bloque de entrada y salida de datos, es para enviarla al exterior del sistema o va a entrar al mismo; si algún dato va a ser grabado o leído en la memoria, etc; casi todas las señales de control son generadas por la CPU.

Esta canal es unidireccional en ambos sentidos, es decir, algunas líneas son salidas y otras entradas, sin embargo, ninguna de ellas, es bidireccional.

Como podemos apreciar, toda la figura 1 ha sido explicada y su estructura también. Se ha desglosado su principio de funcionamiento, para que de esta manera, se pueda comprender más ampliamente, cómo podemos adecuar el microcomputador para que funcione como terminal. Para lograr esto, es necesario, a partir de la figura 1 empezar a seleccionar los dispositivos requeridos, pues tal y como se ha venido mencionando, es la configuración básica y universal de un microcomputador.

CAPITULO II

DISPOSITIVOS REQUERIDOS:

Dentro de este capítulo se empezará a determinar que necesitamos para llevar a cabo el diseño de la microterminal.

2.1 CPU

Para la elección de un microprocesador, se deben considerar inicialmente el uso que tendrá y de ahí evaluar si es necesario que sea uno de alta o media velocidad, así como la cantidad de procesamiento y direccionamiento.

Actualmente los microprocesadores de 8 bits son muy confiables, pues tanto su fabricación, como su desarrollo se encuentra muy perfeccionado; además la velocidad a la que trabajan es aceptable para considerarlos en el control de terminales de cómputo. La gran mayoría de estos dispositivos ofrecen además de los canales de Dirección y de Control, otro de Dirección/Datos, con el cual, el poder de direccionamiento aumenta considerablemente.

2.2 MEMORIA

Se empezará por ampliar el concepto de las memorias, las cuales se dividen básicamente en dos tipos, las memorias RAM (Memorias de Acceso Aleatorio), y del tipo ROM (Memoria sólo para leer)

La diferencia entre estos dos tipos de memoria es la forma en que se usan, pues las dos pueden almacenar información.

MEMORIAS RAM:

Tal y como se mencionó, este tipo de memoria es de acceso aleatorio, es decir puede guardar y leer información en cualquier localidad, lo principal es que podemos grabar información cuando se desee, esto es, alterar los datos almacenados a nuestra voluntad, por lo que es común que se utilice para guardar información temporal.

MEMORIAS ROM:

Esta memoria es solo de lectura. Paradójicamente a lo que se supone debe ser una memoria, parecería que no

cumple con este requisito, sin embargo se ajusta perfectamente a la palabra "memoria" pues guarda información y el que una vez almacenada ésta no se pueda alterar, es una particularidad que aprovecharemos al máximo.

Ahora bien, en algunas ocasiones se necesita este tipo de memoria (ROM), pero a la cual, se le pueda grabar la información que se desee, pues la programación de los ROM's se hacen sobre pedido, y no se puede alterar el ya existente. A raíz de esta necesidad nacieron los PROM's y los EPROM's, ambos trabajando bajo el concepto ROM.

Una memoria PROM (Memoria Programable sólo para leer) es un ROM pero ofrece la particularidad que podemos grabar la información que se necesita; esto sólo se puede realizar una vez, ya que cuando se introduce la información no se puede borrar.

La moderna tecnología nos ofrece también, la memoria EPROM (Memoria Programable y Borrable sólo para leer). Este tipo de memoria tiene la misma propiedad que el PROM y además pueden borrarse y volverse a usar. Para borrar la información que contiene, se debe realizar un procedimiento especial, ya que si no se ejecuta, no se podrá limpiar la información que contiene.

Es común que en estos tipos de memoria se

almacenen los sistemas operativos.

El que el microprocesador use un mismo canal para dirección y para datos, obliga a usar circuitos demultiplexadores, incrementando la complejidad y el costo, entre otras cosas, sin embargo hay componentes que ya traen implementados los demultiplexadores dentro de sus circuitos.

2.3 PUERTOS DE ENTRADA/SALIDA PARELELO, DISPLAY Y TECLADO

Como se sabe, la función principal de una terminal es ser el medio de comunicación entre el usuario y el computador, y la microterminal debe cumplir con este principio. Hasta ahora, sólo se ha empezado a ver la estructura básica interna (CFU-MEMORIAS) y su conexión. A continuación se introducen los medios por los cuales la microterminal se comunicará al computador y con el usuario.

En la figura 2 se incluyen siete nuevos bloques, éstos son: los puertos de entrada y salida paralelos, el teclado, y el display, los decodificadores, los inversores, los puertos de entrada/salida seriales y los convertidores de amplitud.

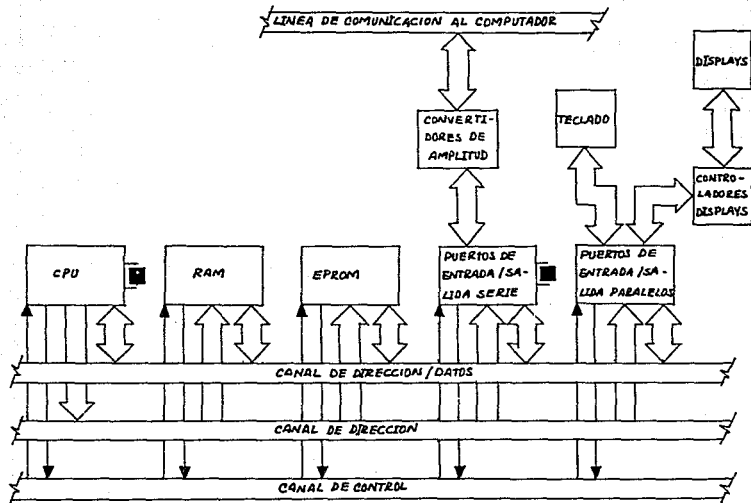


FIGURA 2

El teclado y el display estan conectados a los puertos de ENTRADA/SALIDA paralelos.

DISPLAY:

El display es el elemento que nos podrá presentar la información del computador en caracteres comprensibles para

nosotros. Como el display no es un elemento inteligente, deberá ser controlado por la CPU de la microterminal. Desgraciadamente la conexión entre el display y los puertos no se puede realizar directamente debido a la incompatibilidad que existe (el motivo se explica más adelante), por lo que se necesitan decodificadores e inversores para su control. Es importante recalcar que para cuando la CPU desee usar el display, los puertos serán usados como puertos de salida.

TECLADO

Lo que si se puede conectar directamente a los puertos de entrada/salida es el teclado, pues cada vez que se pulse una tecla, el teclado generará un código, que podrá ser leído e interpretado por la CPU gracias al sistema operativo. Para este caso los puertos estarán en modo de puertos de entrada.

2.4 PUERTO DE ENTRADA/SALIDA SERIE

Ahora se explicará cómo se debe comunicar la microterminal con el computador central. La forma más sencilla y común es por medio de una comunicación asíncrona, por lo que entonces se necesita un puerto de entrada/salida serial, que genere las señales de control correspondientes y de esta manera poder acoplar la microterminal a cualquier computador que ofrezca puertos seriales asíncronos y obtener una línea de transferencia de datos.

Para lograr lo anterior, dentro de la figura 2, se ha incluido el Puertos de Entrada/Salida Serial, al cual está conectado a los tres canales del sistema. El canal de control, le indica a este bloque, si la CPU desea sacar información; los puertos seriales, también usarán este canal para informar al CPU el estatus que guarda. Por el canal de dirección, los puertos seriales sabrán si la CPU desea trabajar con ellos, y por el canal de dirección/datos, fluirán los datos a que van ser transmitidos o que fueron recibidos. Los convertidores de amplitud, tienen como función el adecuar el nivel de las señales, pues dentro de la microterminal son de una amplitud y fuera de ella, son de otra. Junto a este bloque se encuentra un cristal de cuarzo, éste permite que se pueda programar la velocidad de transmisión/recepción.

2.5 SELECCION DE DISPOSITIVOS

Basándonos en la figura 2 ahora se procede a seleccionar los dispositivos que más se apequen a nuestro último diagrama de bloques. Este punto es muy importante pues es el paso que se da entre lo que se ha planeado en papel y la realidad.

Los dispositivos que a continuación se mencionan son de la familia INTEL, es una marca muy conocida y comercial, su uso es frecuente, además de ser los que utilizamos en el curso profesional.

El microprocesador 8085A se apega perfectamente al modelo de CPU que hemos venido explicando. Tiene una buena velocidad y es fácil encontrarlo en el mercado mexicano.

Para la memoria RAM el circuito integrado 8156 es el adecuado pues además de tener buena respuesta se puede acoplar directamente al microprocesador 8085A, por ser de la misma familia.

El 8755A será el que represente al bloque EPROM, que al igual que el 8156, es totalmente compatible con los pines del 8085A.

Otra de las razones por las que se eligieron las memorias 8156 y 8755A, es porque ya traen implementados puertos de entrada y salida paralelos, por lo que al desaparecer este bloque de la figura 2, simplifica el hardware y minimiza la probabilidad de falla.

Tomando en cuenta que es una microterminal, el teclado debe ser pequeño y dinámico, al investigar y analizar los diferentes tipos que hay en el mercado, se encontró que un

fabricante (Rapid Circuit) ofrece teclados ultraplano y pequeños, además de no tener un precio elevado.

Lo más difícil fue encontrar el display. En un principio se consideró uno alfanumérico (de 16 segmentos), ya que no se había encontrado uno matricial que es el ideal, pues permite desplegar cualquier tipo de caracter. Sin embargo, un distribuidor de Hewlett-Packard los ofrece, su número de modelo es el 5082-7101.

La arquitectura de un display matricial, como su nombre lo dice, se maneja por columnas e hileras (como una matriz matemática) Si además consideramos que cada matriz consta de un arreglo de LED's (Diodo Emisor de Luz -Light Emission Diode) de 7 x 5, y cada display contiene 4 matrices, se puede tener una idea muy clara que para controlar cada diodo se requiere de un código, esto puede hacer crecer el sistema

operativo a niveles alarmantes, por lo que el microprocesador tendría que dedicar más tiempo en controlar al display. El número de líneas que nos ofrecen los puertos de entrada y salida paralelos, no son suficientes para alojar todos los pines de los displays (son más los pines de los displays que el de los puertos paralelos), así que es necesario usar decodificadores. Estos permitirán transferir la información de los puertos a los pines de los displays.

Los decodificadores enviarán un código de salida (a los displays matriciales) dependiendo de la información que reciban de los puertos a los que estén conectados. Los decodificadores, en su etapa de salida, si pueden alojar todos los pines de los displays a partir de unos cuantos que reciba del puerto de entrada/salida paralelo.

Por otro lado, el teclado puede funcionar compartiendo los puertos paralelos con los displays matriciales, pues gracias a los decodificadores, la CPU puede dar en un momento entrada a la información proveniente del teclado a los puertos, no permitiendo que ésta se vaya al display mostrando caracteres raros. Cuando la información vaya a ser desplegada, no importa que el código se vaya al teclado, pues no producirá ningún error en él.

El NCS858 es un UART programable y será el encargado de realizar las funciones de los puertos de entrada/salida seriales. Aun y cuando este circuito es

fabricado por National Semiconductor Corporation, es compatible 100% con el microprocesador 8085A. El implementar un circuito de este tipo, trae como beneficio, que el microprocesador puede dedicarle más tiempo a otras cosas, en lugar de consumir recursos para el control de las comunicaciones, pues una vez programado el NSC858, sólo necesita que le envíen la información a transmitir y extraer de él los datos recibidos.

A esta altura, la forma de conexión y el principio de funcionamiento de la microterminal esta planteada, por lo que en el próximo capítulo describiremos cada uno de los elementos anteriormente mencionados.

CAPITULO III

DESCRIPCION DE DISPOSITIVOS

Es de suma importancia conocer los dispositivos elegidos, saber cómo están constituidos, sus propiedades, sus alcances y sus restricciones, así como los beneficios que se pueden aprovechar y lo más importante, cómo funcionan. Así pues, dentro de este capítulo se explica cada dispositivo al nivel que esta tesis lo necesita.

3.1 MICROPROCESADOR INTEL 8085A

Producto de la compañía INTEL, es un microprocesador de 8 bits en paralelo que se fabrica en un circuito integrado DIP de 40 patas terminales, usando la Alta Escala de Integración, apéndice A

La 8085A utiliza al canal de datos/dirección (AD0-AD7) para la entrada/salida de datos y también para enviar la parte baja (8bits) de las direcciones. Realiza una función de "multiplexor". Esto obliga a la microcomputadora a tener

circuitos que realicen la función de multiplexor para los circuitos (memorias e interfaces) que se conectan al canal de dirección/datos o usar circuitos que tengan la función de multiplexor, tales como el 8156 y el 8755A.

El reloj con frecuencia de 3.03MHz (período= 330 nseg) permite un rango de tiempo de ejecución de las instrucciones de la 8085A de 1.3 a 5.75 microsegundos.

La 8085A contiene 6 registros de propósitos general (B, C, D, E, H y L) de 8 bits y el Acumulador de 8 bits, también de propósito general pero con características que lo hacen diferente (por ejemplo, el Acumulador siempre se utiliza en las instrucciones aritméticas y lógicas). Los 6 registros de propósito general se pueden direccionar en forma particular (para 8 bits) o en pares de registros (para 16 bits) proporcionando precisión sencilla y doble en sus operaciones al momento de programarlo. La serie de instrucciones de la 8085A "ponen" (envían a nivel 1) o "limpian" (envían a nivel cero) a cuatro banderas y a las que por medio de instrucciones se les puede investigar su estado, estas banderas son: Signo (S), Acarreo (CY), Paridad (P) y Cero (Z). Una quinta bandera se utiliza para operaciones de aritmética BCD y se le conoce como bandera de Acarreo Auxiliar (AC). Estas banderas se manejan como un registro conocido como registro de Banderas de Estado.

La 8085A tiene una característica de "pila de almacenar" (STACK) externo en una parte de la memoria RAM, la cual esta disponible para el microprocesador, se puede usar como pila de "último en llegar/primerero en salir" (Last In/First Out, LIFO) para almacenar/obtener el contenido de cualquiera de los siguientes registros: Acumulador, Banderas, Contador de Programa y los tres registros pares. Un registro de 16 bits, Apuntador del Stack (SP, Stack Pointer) controla la dirección de este Stack externo. La 8085A utiliza un canal de Dirección de 16 líneas y un canal de Datos de 8 bits para facilitar la interfaz directa de la 8085A con la memoria y con los dispositivos de Entrada/Salida.

El control final de los canales de Dirección y de Datos, reside en la señal de control HOLD, la cual proporciona la habilidad de suspender las operaciones del procesador y forzar a los canales de Dirección y de Datos a su estado de alta impedancia. Esto permite a que se realice la función OR de estos canales con otros dispositivos de control para acceso directo a memoria (DMA, Direct Memory Access) u operaciones de multiproceso. El microprocesador 8085A (fig. 3), se implementa en un circuito integrado de 40 patas con la siguiente distribución:

- Un Canal de Dirección de 8 líneas.
- Un Canal de Dirección/Datos de 8 bits
- 11 Salidas de control desde la 8085A.
- 9 entradas de control a la 8085A.

- 2 entradas para la fuente de alimentación.
- 2 entradas para el cristal de cuarzo.

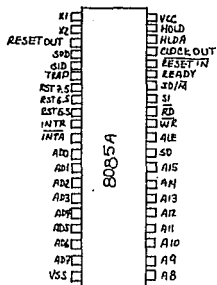


Figura 3

ARQUITECTURA DE LA 8085A:

A continuación se mencionan las siguientes unidades funcionales que nos ayudarán como guía para la explicación del CPU 8085A:

- a) Arreglo de registros y una lógica de dirección.
- b) Unidad Lógica Aritmética.
- c) Registro de instrucción.
- d) Decodificador de instrucción.
- e) Selección de control y tiempos.

- f) Buffer del canal de datos bidireccional de 3 Edos.
- g) Control de las interrupciones.
- h) Control de entrada/salida de datos en serie.

La figura 4 ilustra los bloques funcionales dentro de la 8085A

REGISTROS Y SECCION DE CONTROL DE LA 8085A:

El arreglo de registros consiste de los siguientes registros y circuitos auxiliares:

- Contador del Programa (Program Counter, PC), 16 bits.
- Apuntador del Stack (Stack Pointer, SP), 16 bits
- Seis registros de propósito general de 8 bits arreglados por pares y referidos como B, C, D, E, H y L
- Un circuito incrementador/decrementador, 16 bits
- Un Latch de dirección, 16 bits

Un programa es una serie de actividades que tiene que efectuar el microprocesador. Estos pasos se le ordenan por medio de las instrucciones, que tienen código único. El apéndice C muestra la tabla de instrucciones con sus códigos para el microprocesador 8085A.

Los códigos de instrucción que forman el programa se deben almacenar en memoria, en localidades contiguas.

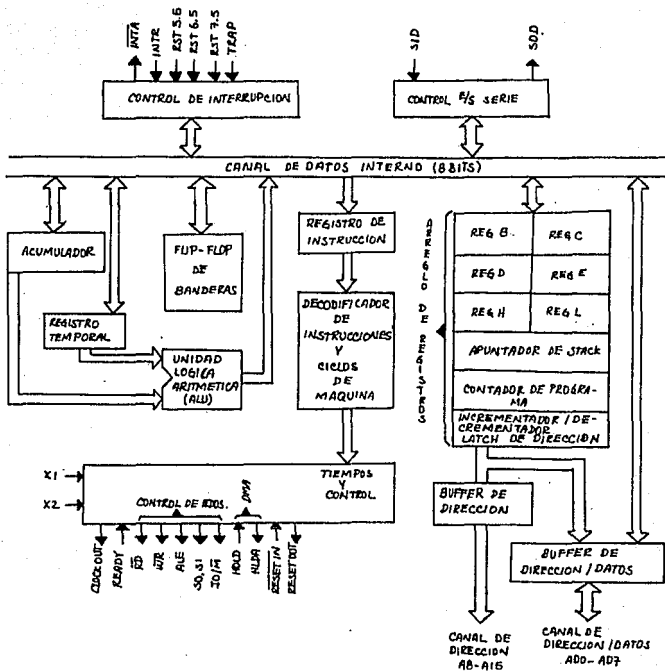


Figura 4

Una vez que se ha cargado la memoria con los códigos de instrucción y con los datos, debemos indicarle al microprocesador a partir de que localidad se encuentra el programa.

ma. Esto se realiza cargando un registro de 16 bits que se le da el nombre de contador de programa (PC) con el valor de la dirección de la localidad que contiene el primer código de instrucciones del programa.

CONTADOR DEL PROGRAMA (PC):

El PC 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 la 8085A envía el contenido de PC al canal de dirección, por medio del cual, direcciona a una localidad específica de memoria.

El PC se incrementa en uno cada vez que el microprocesador lee el código de la instrucción contenida en la localidad direccionada. De esta forma, el PC direcciona secuencialmente a las localidades de la memoria donde se encuentra almacenado el programa, Figura 5.

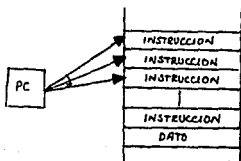


Figura 5

REGISTRO DE INSTRUCCION:

Una vez que se carga el PC con la dirección de la primera localidad del programa, el microprocesador inicia la ejecución del programa enviando al buffer de dirección el contenido del PC, el cual se transmite por el canal de dirección hacia la memoria. Además el microprocesador envía las señales de control, en este caso genera las señales de RD e IO/M = 0. La memoria responde enviando el contenido de la localidad direccionada al canal de datos. Posteriormente la 8085A lee este contenido en un registro conocido como "Registro de Instrucción" (RI).

Cada vez que un valor llega al Registro de Instrucción el microprocesador lo interpreta como un código de instrucción.

DECODIFICADORES DE INSTRUCCION Y CICLOS DE MAQUINA:

El contenido del RI a su vez es disponible por el Decodificador de Instrucciones. La salida del decodificador al combinarse con varias señales de tiempos, genera la Sección de Control, proporciona señales de control a todos los componentes del microprocesador (arreglo de registros, registro de instrucción, ALU, etc), figura 6

SECCION DE CONTROL Y DE TIEMPOS:

Las salidas del decodificador de instrucciones y las señales de control externas (INT, READY, HOLD, etc.), alimentan a la Sección de Control y Tiempos para que genere las señales de tiempos de los estados y ciclos de máquina de la instrucción en proceso.

6 REGISTROS, LATCH DE DIRECCION E INCREMENTADOR/DECREMENTADOR:

La memoria no únicamente se carga con los códigos de las instrucciones del programa sino que también almacena datos. Cuando la 8085A lee los datos no los debe almacenar en el RI, porque los interpretaría como códigos de instrucciones. Por lo tanto, la 8085A debe contar con algunos de los registros especiales para recibir este tipo de datos.

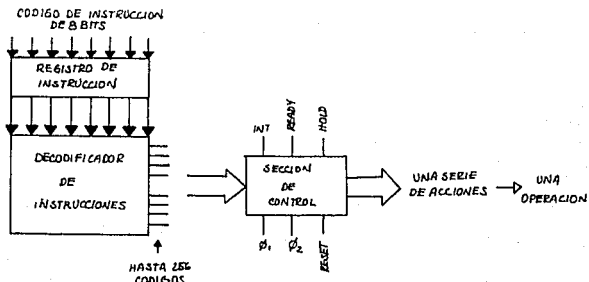


Figura 6

El microprocesador 8085A cuenta con 6 registros de 8 bits de propósito general en el Arreglo de Registros: B, C, D, E, F, H y L, figura 4. Estos registros se pueden utilizar en forma individual como registros de 8 bits o en registros pares para manejarse como registros de 16 bits en la forma siguiente: B-C, D-E, y H-L. En cada par, los registros B, D y H contienen los bits de más orden, y los registros C, E, y L los bits de más bajo orden. Estos registros en forma de parejas toman los nombres de B (para B-C), D (para D-E) y H (para H-L). A la dirección de memoria que se forman con los contenidos de H y L se conoce como dirección M.

Se pueden transferir datos de 8 bits entre el canal interno de la 8085A y cualquiera de los registros del Arreglo de Registros, incluyendo el PC y el SP. Estos dos últimos registros requieren de dos datos de 8 bits. La selección del registro se realiza por medio de un registro multiplexor selector, que recibe comandos de la sección de control. El PC y el SP alimentan al Latch de Dirección.

El Latch de Dirección también recibe datos de 16 bits de cualquiera de los tres registros pares y los conduce al Buffer de Dirección de 16 bits (A15-A0), o al circuito Incrementador / Decrementador. El circuito Incrementador / Decrementador recibe datos del Latch de Dirección y el resultado de la función ordenada se envía a un registro par del Arreglo de Registros. El dato de 16 bits se puede incrementar,

decrementar o simplemente efectuar con él una transferencia entre registros.

Con los 6 registros de propósito general se permiten efectuar, por medio de las instrucciones, las siguientes funciones:

- Recibir datos desde la memoria
- Enviar datos a la memoria
- Incrementar o decrementar en uno su contenido
- Formar una dirección con sus contenidos (registro par) o datos de 16 bits.
- Transferir datos entre ellos
- Tener un operando durante las funciones de la ALU

APUNTAADOR DEL STACK (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 no tener la necesidad de buscar la dirección de esa área cada vez que se desee almacenar un dato o rescatar el último dato almacenado. Esta área de memoria se llama "Stack".

El microprocesador 8085A cuenta con un registro

"Apuntador del Stack" SP (Stack Pointer) que contiene una dirección de memoria RAM a partir de la cual, en forma descendente, se pueden salvar los contenidos de un registro par o a partir de la cual, en forma ascendente, se obtienen los últimos dos datos almacenados en esa área para salvar a un registro par. El SP es un registro de 16 bits. En esta tesis se indicarán los términos Contador del Programa y Apuntador del Stack por sus siglas en inglés PC y SP.

Para salvar los contenidos de un registro par se utiliza la instrucción PUSH, y para cargar a un registro par se utiliza la instrucción POP. La figura 7 ilustra al SP.

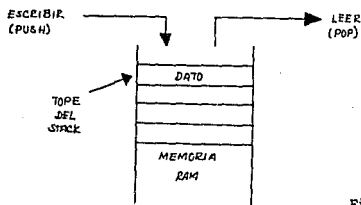


Figura 7

UNIDAD LÓGICA-ARITMÉTICA:

Las operaciones de los datos dentro de la 8085A se realizan por un grupo de componentes lógicos conocidos como Unidad Lógica-Aritmética, que cada vez que se haga referencia a ella se llamará por sus siglas en Inglés ALU (Arithmetic-Logic

Unit). La ALU tiene la lógica necesaria para llevar a cabo al menos las siguientes operaciones:

- 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 esta información se conoce como Registro de Banderas.

Si se necesita cualquier otra manipulación más completa de una palabra de datos, se debe recurrir a la combinación de estas funciones de la ALU. La ALU de la 8085A contiene los siguientes registros:

- Un Acumulador de 8 bits
- Un registro de banderas de ocho bits: cero, acarreo, signo, paridad y acarreo auxiliar; tres no se usan
- Un registro Temporal, TMP de 8 bits

La ALU se alimenta del Registro Temporal TMP, del Acumulador y del flip-flop de acarreo del Registro de Banderas. El resultado de la operación usualmente se deposita

en el Acumulador A o también se puede depositar en el canal de datos interno para enviarlo a los registros. La ALU también alimenta al Registro de Banderas.

ACUMULADOR Y REGISTROS TEMPORALES:

Uno de los registros más importantes de la 8085A es el Acumulador, A. El Acumulador es un registro de 8 bits, usualmente contiene uno de los dos operandos en las operaciones aritméticas y lógicas de la 8085A, y en donde normalmente se deposita el resultado de las operaciones.

Con el Acumulador se pueden realizar las mismas funciones que en los 6 registros individuales del Arreglo de Registros. Además, es el único que puede enviar o recibir datos a y de los puertos de salida y entrada respectivamente.

El Acumulador se puede cargar desde la ALU (después de una operación aritmética o lógica), desde el canal de datos internos (desde un registro, desde la memoria o desde un dispositivo de entrada), y puede transferir datos a la ALU y al canal de datos internos (dirigiéndose a un registro, a la memoria o a un dispositivo de salida).

El Registro Temporal TMP recibe información desde el canal de datos interno (desde la memoria o un registro), para

enviarla a la ALU. El registro Temporal se carga temporalmente con uno de los dos operandos de la función aritmética o lógica (el otro se carga en el Acumulador) que proviene de la memoria o de un registro.

BANDERAS DE ESTADO:

El microprocesador 8085A tiene un registro de 8 flip-flops para monitorear ciertos resultados de las operaciones de la ALU, figura 8. A la información que almacenan estos flip-flops se conoce como banderas de estado. Las banderas se actualizan después de cada operación con alguno de los registros y no todas las operaciones modifican a todas las banderas.

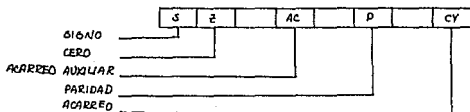


FIGURA 8

BUFFER DEL CANAL DE DATOS:

Este buffer bidireccional de 8 bits, de 3 estados, se usa para aislar el canal de datos interno de la 8085A del canal de datos externo. En el modo de salida, el microprocesador carga el contenido del canal interno en el latch

de ocho bits y maneja el buffer para salida del dato al canal de datos externo. Durante el modo de entrada los datos del canal de datos externo se transfieren al canal de datos interno, para que posteriormente la 8085A la cargue en uno de sus registros.

CONTROL DE LAS INTERRUPCIONES Y DATOS EN SERIE:

La Unidad de Control de las Interrupciones contiene la lógica que define cuándo se reciben solicitudes de interrupción, las prioridades de las mismas y la secuencia de eventos que se tienen que llevar a cabo para darle servicio. La Unidad de Control de Datos en Serie contiene la lógica para la recepción y transmisión de datos en serie.

CIRCUITOS COMPATIBLES CON LA 8085A:

La compañía INTEL fabrica circuitos integrados que fueron diseñados especialmente para soportar a la 8085A, estos circuitos son: 8156 y 8755A. Con estos dos integrados y la 8085A se pueden diseñar microcomputadoras con memorias RAM, y EPROM, puertos de entrada/salida y timers. Los dos circuitos realizan internamente la función de multiplexar el canal de dirección/datos (AD0-AD7) de la 8085A.

una memoria desde que recibe una dirección estable hasta que presenta en un canal de datos el dato de la localidad direccionada o hasta que esté listo para recibir un dato. Una característica de este circuito es de que los dos puertos de entrada/salida pueden ser programados para que sus 8 líneas se definan en forma particular (línea por línea) como entradas o como salidas. Esto significa que las líneas, por ejemplo, 1,4 y 7 de un puerto se pueden programar como entradas, las otras 5 líneas como salidas. Esta característica da una gran flexibilidad al diseñar.

DESCRIPCION FUNCIONAL DEL 8755A:

Este circuito cuenta con latches internos para multiplexar el canal ADO-AD7 utilizando la señal ALE para la sincronización.

La memoria EPROM (figura 10) está direccionada por 11 bits (ADO-AD7 y A8-A10) y los puertos por 2 bits (ADO-AD1). El valor de las líneas ADO-AD7, A8-A10, CE, \overline{CE} , e IO/M se cargarán en los circuitos latch internos del 8755A con la transición alto-bajo de la señal ALE. Si las líneas CE y \overline{CE} están activas, e $IO/\overline{M}=0$ cuando la línea RD toma nivel 0, el contenido de la localidad de memoria direccionada por los latches de dirección se envían a través de los buffers de salida ADO-AD7.

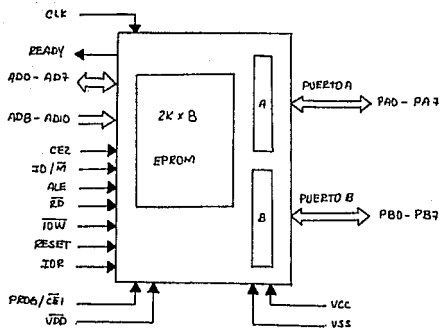


Figura 10

DIRECCIONAMIENTO DEL 8755A:

Las líneas ADO-AD7 y AB-A10 se deben conectar, directamente, a las líneas del mismo nombre de la 8085A. Las líneas A11-A15 se pueden decodificar para seleccionar de 1 a 32 integrados.

La lógica del decodificador debe seleccionar al integrado (activa CE o \overline{CE}) cuando la 8085A genera una dirección de memoria y además debe poder seleccionar al integrado cuando la 8085A genera un código de selección de puerto de entrada/salida (con las instrucciones IN y OUT). Durante una instrucción de entrada/salida, la 8085A envía los 8 bits de número

ro de puerto o código de selección del puerto involucrado, tanto por las líneas AD0-AD7 como por las líneas A8-A15.

El 8755A cuenta con dos puertos y dos registros DDR, por lo que requiere de 4 direcciones entrada/salida o códigos de selección, las cuales se pueden generar con las líneas AD0-AD1 ó A8-A9. Esto podría indicar que las líneas AD2-AD7 ó A10-A15, se pueden utilizar para la selección del integrado. Sin embargo, la línea A10 ya está conectada al integrado. 11 líneas de dirección AD0-A10 permiten direccionar los 2K Bytes de memoria EPROM, por lo tanto, sólo las líneas A11-A15 se pueden usar para la selección del integrado. Esto significa que las líneas A2 y A10 no influyen en la selección del puerto ni del integrado en operaciones de entrada/salida, es decir, un bit de los 8 que utiliza la 8085A para direccionar los puertos no es utilizado para la 8755A.

SECCION DE ENTRADA/SALIDA DE LA 8755A:

La sección de entrada/salida está direccionada por los valores cargados en los latches internos AD0-AD1. El 8755A tiene dos registros de dirección de datos (DDR, Data Direction Register) para especificar el listado, entrada o salida de cada pata en el puerto correspondiente.

La figura 11 ilustra las funciones de los puertos de entrada/salida y los registros DDR para el caso del bit cero del puerto A y del registro DDR A. La misma lógica se aplica para los otros 7 bits, y para los 8 del puerto B y registro DDR B. Se puede observar que un bit con valor 0 en el registro DDR indica que la línea correspondiente (PA0 en la figura 11) en el puerto está programada en el "modo de entrada", el buffer Habilitar Salida está inactivo.

Un bit con valor 1 en el registro DDR indica que la línea correspondiente (PA0 en la Fig. 11) en el puerto está programada en el "modo de salida", el Buffer Habilitar Salida está activo. Los contenidos de los puertos y registros DDR en base al valor cargado en los latches internos ADO y AD1.

La configuración de la Fig. 11 ilustra en forma clara, porque, en base al valor cargado en el registro DDR de un puerto, se puede programar la dirección de las líneas en forma particular. La tabla A muestra la asignación de los puertos y registros DDR en base al valor cargado en los latches internos ADO y AD1.

Para acceder los puertos A y B o los registros DDR A y DDR B del 8755A, la línea IO/M del integrado debe tener nivel 1, el cual se genera cuando la 8085A está procesando una instrucción de entrada/salida. Los puertos A y B se pueden usar para operaciones de entrada y salida de datos, mientras que

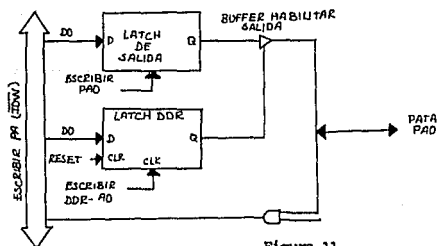


Figura 11

los registros DDR sólo se pueden usar en operaciones de salida (para programar la dirección de los puertos). La 8085A no puede leer (entrada) el contenido de los registros DDR. La dirección de los latches internos ADO y AD1. (cargados previamente con la transición alto-bajo de ALE) se utilizan en el 8755A para identificar los puertos (A o B) o los registros DDR (A o B), Tabla A.

TABLA A

AD1	ADO	SELECCION
0	0	Puerto A
0	1	Puerto B
1	0	DDR (*) del puerto A - DDR A
1	1	DDR (*) del puerto B - DDR B

(*)= Registros DDR (Data Direction Register)

El contenido de un puerto de entrada/salida se puede leer cuando se cumplen las condiciones siguientes: El integrado está habilitado ($CE=1$ y $\overline{CE}=0$), $IO/\overline{M}=1$ y $\overline{RD}=0$. 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 IO/\overline{M} . Al activarse la línea \overline{IOR} el integrado realiza la misma función que la combinación $IO/\overline{M}=1$ y $\overline{RD}=0$.

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 de entrada/salida o registro DDR seleccionado se afectan no importando el modo (entrada o salida) del puerto y del estado de la línea IO/\overline{M} . El dato de salida para un puerto o registro se carga hasta que \overline{IOW} regresa al nivel 1. Lo anterior indica que la entrada \overline{RD} del 8755A se usa para designar la operación de "leer" de memoria o de puertos de entrada/salida, en estos casos la línea IO/\overline{M} debe tener el valor de cero o uno respectivamente. Para cargar (escribir) en los puertos y en los registros DDR, el 8755A cuenta con la entrada \overline{IOW} no importando el valor de IO/\overline{M} . La línea \overline{IOW} se puede conectar directamente a la línea \overline{WR} de la 8085A. El 8755 también tiene la línea \overline{IOR} para la función de leer puertos, misma función que se realiza cuando $\overline{RD}=0$ e $IO/\overline{M}=1$. En todos estos casos el integrado debe estar habilitado ($CE=1$ y $\overline{CE}=0$).

En la figura 11 se puede observar que cuando se activa la línea RESET, las salidas Q de los latches DOR toman valor cero desactivando los Buffers de Habilitar Salidas. Esto es equivalente a poner el puerto en el modo de entrada.

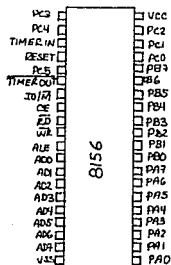


Figura 12.

3.3 MEMORIA RAM 8156

Este circuito integrado ofrece 256 bytes (8 bits) de memoria estática de leer/escribir (RAM), de dos puertos de entrada/salida de 8 bits, un puerto de entrada/salida de 6 bits y un "timer" de 14 bits.

Está implementado en un circuito integrado DIP de 40 patas terminales utilizando la tecnología MOS. Requiere de una alimentación de +5 Volts. Todas las salidas y entradas son compatibles TTL. La figura 12 ilustra la distribución de las patas terminales, y la figura 13 el diagrama fun-

cional de este circuito integrado.

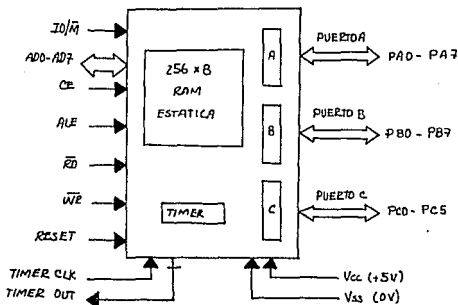


Figura 13

Tiene un tiempo de acceso máximo de 400 nseg por lo que no requiere estado de espera cuando se usa con la 8085A.

CARACTERÍSTICAS ELÉCTRICAS:

El 8156 tiene las mismas características eléctricas que el circuito explicado en la sección anterior en cuanto a la compatibilidad con la 8085A (ver apéndice B). El consumo de corriente para este integrado es de 180 mA.

DIRECCIONAMIENTO DEL 8156

El 8156 cuenta con tres puertos de entrada/salida, un t mer con dos registros, y los registros de comando/estados, por lo que requiere de seis direcciones de entrada/salida o c digos de selecci n. El n mero del puerto deseado se ordena en las entradas A0-A2, (tabla B). Los c digos de selecci n que se usan son del XXXX X000 al XXXX X101, no se usan el XXXX X110 y el XXXX X111. Para la selecci n del 8156 no se pueden usar las l neas A3-A7 porque est n conectadas al 8156 ni tampoco las l neas A8-A10 porque tienen un duplicado de las l neas A0-A2. Las l neas A11-A15 son las  nicas que se pueden usar en la decodificaci n de los c digos de selecci n.

Tabla B

AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	PUERTO O REGISTRO
X	X	X	X	X	0	0	0	Reg.de comando/Eidos
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 del Timer
X	X	X	X	X	1	0	1	6 bits altos del Timer

Direcciones de los Puertos y Registros del 8156

Con las líneas A11-A15 se pueden decodificar hasta 32 códigos de selección, con lo cual se pueden seleccionar 32 integrados 8156. Esto permite un máximo de 8K bytes. Ya que las líneas A8-A10 no se utilizan y no se puede generar direccionamiento absoluto. Al no usarse las líneas A8-A10 cada 8156 ocupará un área de direcciones de 2K (A0-A10), aunque realmente sólo utiliza 256 bytes.

SECCION DE ENTRADA SALIDA

La operación de los puertos y el timer se programa o configura por medio del registro de Comandos, mientras que el estado de estos puertos y el timer se pueda conocer leyendo el registro de estados, figura 14.

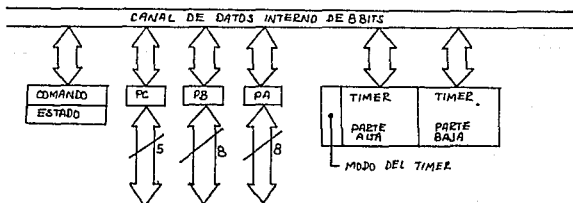


Figura 14

Durante una transferencia de entrada/salida (la 8085A está ejecutando una instrucción OUT o IN), el 8156 utiliza el valor de las líneas A0-A2 para seleccionar los puer

tos y los registros. La tabla B muestra la relación de los valores de las líneas AD0-AD2 con los puertos y registros. Las líneas A11-A15 directamente o por medio de decodificadores se utilizan para habilitar el integrado activando la entrada CE del 8156.

REGISTROS DE COMANDOS

El registro de Comandos consiste de 8 latches que controlan la operación de los tres puertos de entrada/salida y el timer, figura 15. El registro de comandos se puede cargar en cualquier momento con una instrucción OUT. El contenido de este registro no se puede leer. El código de selección o dirección de este registro es XXXXX000.

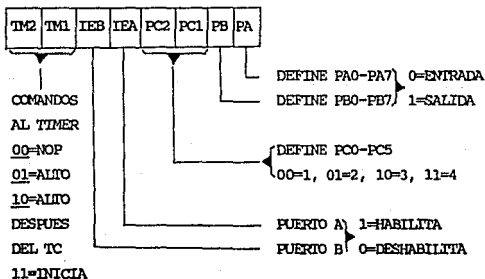


Figura 15

Los bits 0 y 1 del registro de Comandos 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 cuatro posibles configuraciones o alternativas de entrada/salida, tabla C. Los bits 4 y 5 se utilizan para habilitar (=1) o deshabilitar (=0). Las solicitudes de interrupción por el puerto A o por el puerto B respectivamente.

TABLA C
Posibles Configuraciones del Puerto C

PATA	ALT1	ALT2	ALT3	ALT4
PC0	*	**	A INTR (Int. del Pto A)	A INTR
PC1	*	**	A EF (Buffer Pto A lleno)	A EF
PC2	*	**	A STB (Strobe del Pto. A)	A STB
PC3	*	**	**	B INTR
PC4	*	**	**	B EF
PC5	*	**	**	B STB

* = Puerto de entrada ** = Puerto de salida

Los bits 7 y 6 se utilizan para darle comandos al

timer. El Valor 00 no afecta la operación del contador. El valor 01 detiene el conteo si el contador del timer está corriendo, no afecta la operación del contador si no está corriendo. El valor 10 ordena detener el conteo cuando el contador termina el conteo en proceso. Con el valor 11 existen dos posibilidades, (1) si el contador no está corriendo carga al timer y al contador y arranca inmediatamente, (2) si el contador está corriendo carga al timer y cuando se termina el conteo en proceso, inicia un nuevo conteo cargando una copia del contenido del timer en el contador.

REGISTRO DE ESTADOS:

El registro de estados consiste de 7 latches que registran información acerca del "estado" de los puertos A y B, y del timer. La 8085A puede leer el contenido de este registro pero no puede escribir en él. La Figura 16 muestra la asignación de los bits en el registro de estados.

El bit cero toma nivel uno cuando el puerto A tiene un dato para el exterior (puerto A está programado como salida) o recibió dato del exterior para la 8085A (el puerto A está programado como entrada). El bit cero toma nivel cero cuando el 8156 recibe un pulso negativo en la entrada A STB (pata PC2). Cuando el puerto C está programado en los modos ALT3 ó ALT4. El bit uno indica si el contenido del puerto A ya

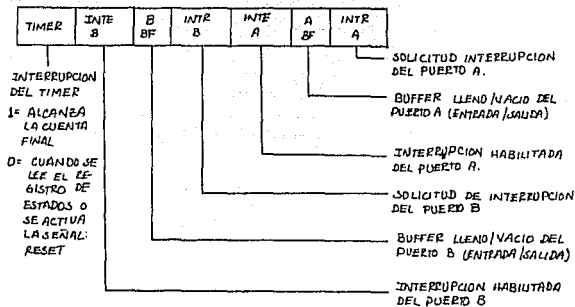


Figura 16

se leyó por la 8085A (puerto A como entrada) o por el exterior (puerto A como salida).

Si no se ha leído el puerto A, bit 1=1 (buffer lleno) y si ya se leyó, bit 1=0. Cuando el puerto A se carga, el bit 1 toma nivel 1 y nivel 0 cuando se lee.

El bit 2 indica si el puerto A está (=1) o no (=0) habilitado 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 (interrupción del timer) toma nivel 1 --

cuando el contador termina su cuenta y toma nivel 0 en el momento de leer el registro de estado o cuando se activa la entrada RESET del 8156.

FUERIOS A, B Y C:

Los puertos A y B se pueden programar como entradas o salidas, pero no se pueden programar individualmente los bits como entradas o salidas dependiendo cómo se programe el puerto C, los puertos A y B pueden usar las líneas del puerto C como líneas de protocolo e interrupción. El puerto C se puede programar en una de cuatro funciones alternativas. (Tabla C) El puerto C se puede usar como puerto de entrada o salida, ALT1 y ALT2 respectivamente en el registro de comando. En la configuración ALT3, 3 bits del puerto C quedan como salidas y 3 bits se usan como líneas de protocolo entre el puerto A y el exterior. En la configuración ALT4, 3 bits se usan como líneas de protocolo entre el puerto A y el exterior, y 3 bits se usan como líneas de protocolo, entre el puerto B y el exterior.

Al momento de programar el puerto C con las configuraciones ALT3 y ALT4, las señales de control para los puertos A y B se inician como se muestra en la Tabla D.

Tabla D

CONTROL	ENTRADA	SALIDA
BF	0	0
INIR	0	1
STB	Control de entrada	Control de entrada

La figura 17 muestra cómo están estructurados los bits de los puertos A y B en el 8156. Aún y cuando los puertos de entrada/salida están programados como puertos de salida, el contenido del puerto de salida se puede leer con una instrucción IN.

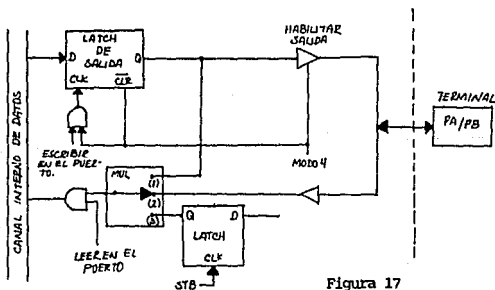


Figura 17

Los latches de salida se limpian automáticamente cuando el puerto se programa como salida. No se puede escribir o cargar un puerto cuando el puerto está programado como entrada. En la Figura 17 se puede observar que el punto (4) tie

ne nivel 0 cuando el puerto está en modo de entrada. El nivel 0 limpia el latch de salida. Esto da como resultado que cuando el puerto pasa del modo de entrada al de salida las líneas del puerto toman nivel 0. Cuando se activa la línea RESET del 8156 se limpian los latches de salida y los tres puertos quedan en modo de entrada.

La Figura 18 muestra una configuración de los puertos A, B y C, cuando los puertos A y B se programan como salida y entrada respectivamente usando el puerto C como línea de protocolo.

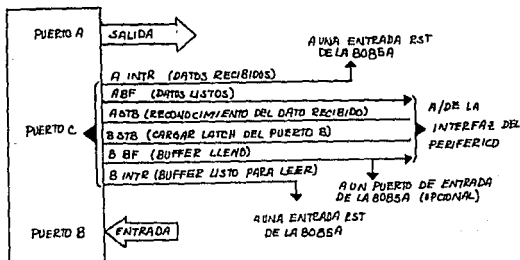


Figura 18

3.4 DISPLAYS MATRICIALES:

Son un producto de la familia Hewlett-Packard y están constituidos por pequeños diodos emisores de luz (LED's), arreglados en una forma matricial de 5x7. Se encuentran en el mercado nacional en encapsulados tipo circuito integrado (DIP-Dual In Package) de 28 patas terminales, y cada uno ofrece 4 matrices, Figura 19.

En la figura 20 podemos apreciar la distribución de las patas terminales, en las que podemos distinguir que están agrupadas por letras y numeradas.

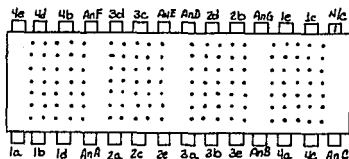


Figura 19

bajo la siguiente nomenclatura: "An" significa ánodo común, son en total 7, y van de la letra "A" a la "G" y cruzan las cuatro matrices que el encapsulado ofrece. Con respecto a los cátodos, éstos se manejan por el número de columnas que son 5 por matriz, por lo que si se consideran 4 matrices, nos dan un -

total de 20 hileras, y para las de la primera matriz serán del 1a al 1e, para la segunda matriz del 2a al 2e y así hasta la cuarta matriz.

Por ejemplo, si deseamos que funcione el primer led de la primera matriz, de izquierda a derecha, tendríamos que poner a funcionar el ánodo A y energizar la fila 1a. Si queremos que encienda el último led de la tercera matriz (el led inferior derecho) habilitaríamos el ánodo G y la hilera 3e, y así.

Por lo anterior se dice que el direccionamiento de este display matricial es tipo x-y (como los ejes cartesianos).

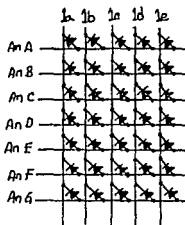


Figura 20

Su alimentación es tipo TTL (+5VDC), por lo que su implementación para ser controlada por microprocesador se puede llevar a cabo directamente, además que su presentación tipo DIP facilita mucho el poderlo incluir en los diseños lógi-

cos. (ver apéndice D).

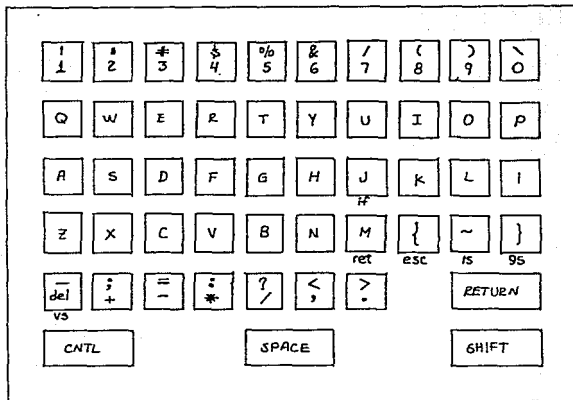
3.5 TECLADO:

Después de mucho analizar los diferentes tipos de teclados que se encuentran en el mercado, se eligió uno que ofrece "Rapid Circuit" pues además de ser pequeño, tenía otra facultad que los otros no reunían y es que es ultraplano (2 mm) esto es, se consideran de tacto fino, y es autoadherible. Su rango de operación se encuentra de -40 a 70°C.

La arquitectura interna es tipo matricial (figura 21), por lo que su control es de fácil manejo. Este teclado ofrece un conector como terminal de todas sus conexiones internas. Esta dividido en seis renglones (ejes "X") y 10 columnas (ejes "Y"). A su vez, el teclado se divide en dos partes, como si separáramos dos micas unidas, por una cara va la mitad del circuito, y por la otra el complemento del mismo. Cuando se unen las dos caras, los únicos puntos en donde se pueden tocar ambos circuitos, son en las teclas, esto es, cada vez que oprimimos una tecla (con 50g mínimo), hacemos que las pistas tengan contacto, generando que se cierre el circuito (que haya continuidad). Las teclas se comportan como switches normalmente abiertos.

Este teclado acepta alimentación tipo TTL, además

Figura 21



de tener un buen tiempo de respuesta. Su cubierta es de tipo acrílico plástico, por lo que su elección, es adecuada. Es de una sola pieza y no tiene huecos por donde pueda entrar polvo o partículas, los que son muy comunes encontrar en dispositivos de este tipo. Ver Apéndice E.

3.6 CIRCUITOS DS14C88 Y DS14C89:

Los circuitos integrados DS14C88 y DS14C89 (apéndice H), fueron diseñados para realizar las conversiones de niveles RS232 a TTL y viceversa. Estos circuitos monolíticos están fabricados con tecnología MOS, utilizando el proceso estándar CMOS. Se ofrecen en encapsulado tipo DIP de 14 patas terminales.

Su uso es muy frecuente dentro del campo de las comunicaciones, por lo que encontrar estos circuitos en terminales, modems, equipo criptográfico es común, pues cumple con los estándares de una interface en cuanto a características eléctricas se refiere, para manejar equipos en un ambiente de telecomunicaciones.

EL CONTROLADOR DE LINEA DS14C88

El DS14C88 es un controlador de líneas (driver li

ne) de 4 compuertas, y tiene la misma disposición de patas que el circuito bipolar MC1488.

La ventaja del DS14C88 sobre el MC1488, está principalmente en el consumo de energía, comparándolos en el peor de los casos, el primero consume 425 microamperes, mientras que el segundo necesita 25 miliamperes, por lo que usar el DS14C88, nos ahorra un 98.3% en el consumo de corriente.

La conversión de niveles TTL (0v -5V) a RS232 (+/-12V) lo realiza a través de una etapa de inversión, y minimiza la interferencia de radiofrecuencias. Otra característica, es que a la salida de este controlador de líneas, ya no es necesario agregar capacitores (mientras que en el MC1488, sí) ahorrándonos de esta manera 7 de estos elementos. La alimentación de voltaje debe ser de +9V y de -9V, para que este circuito trabaje, el uso de pilas es aceptado por el fabricante.

RECEPTOR DE LINEAS DS14C89

El DS14C89 es un receptor de líneas (line receiver) de cuatro compuertas, construidos con sistemas CMOS, siendo éste, el equivalente del MC1489 de tecnología bipolar.

La principal característica de este circuito, al -

igual que el anterior, es que la demanda de corriente con respecto al MC1489 se reduce en un 97.7% (de 25mA a 600 microamperes). Tiene la capacidad de ser un convertidor de voltaje en un rango de +/-30V a nivel lógico TTL, a través de una etapa de inversión.

El DS14C89 elimina la necesidad de conectar capacitores de filtro externos y tiene la capacidad de eliminar ruido que se induce durante el viaje de la señal, debido a que implementa un circuito interno de filtrado.

Con suministrarle +5VDC a este circuito, es suficiente para que trabaje.

3.7 ALIMENTACION DE VOLTAJE:

Todos los circuitos que en este capítulo hemos mencionado, necesitan ser energizados de una forma continua y estable; esto es, sin variaciones tanto en amplitud como en frecuencia. Para lograrlo, se debe agregar una fuente de alimentación, el cual debe de estar constituida por elementos capaces de manejar voltajes y corrientes necesarios para el correcto funcionamiento de los elementos de la micro terminal.

El circuito LM123A, es un regulador de voltaje, ca

paz de manejar hasta 3 amperes y entregar +5V. Se ofrece en encapsulado de metal (Metal Can Package); su alimentación debe ser de 9V. Sus características se pueden apreciar en el apéndice G.

Para lograr los +9V y los -9V, se necesita una batería y un 7660. La batería nos suministra los +9V en una forma estable, necesarios para el DS14C88 y para el 7660, que es un convertidor de voltaje, el cual se alimenta con +9V y nos entrega -9V, también necesarios para el DS14C88. El 7660 viene implementado en un circuito integrado tipo DIP de 8 patas terminales, sus características están disponibles en el apéndice G.

3.8 INVERSOR 7406:

Para las compuertas inversoras, se eligió al modelo 7406, el cual viene implementado en un circuito integrado, con encapsulado tipo DIP, de 14 patas terminales. Su alimentación es de +5VDC y sus salidas son TTL.

El 7406, trae dentro de su estructura, 6 compuertas inversoras y tiene entradas/salidas TTL (ver apéndice G).

3.9 DECODIFICADOR DE LINEAS 74HCT138

Es un decodificador de 3 a 8 líneas, utiliza la tecnología CMOS. Dentro de sus principales aplicaciones esta la decodificación de direcciones de memoria o encaminar datos. Se ofrece en encapsulado tipo DIP (Dual In Package) de 16 patas terminales. Necesita +5VDC para alimentarse y sus salidas son TTL.

Su consumo de energía es muy bajo, como sucede con los circuitos CMOS (80 microamperes). Este circuito es el reemplazo del 74LS138, con quien es compatible pata a pata

El 74HCT138, tiene tres selectores binarios como entradas (A, B y C); si el dispositivo es habilitado, esas entradas determinarán cual de las 8 salidas (que generalmente tienen nivel alto), tomará estado lógico cero. Tres habilitadores, dos que se activan con nivel bajo ($\overline{G2A}$ y $\overline{G2B}$) y uno con nivel alto (G1); se ofrecen para el manejo de decodificadores en cascada.

Sus características se pueden observar en el apéndice F.

CAPITULO IV

DISEÑO DEL CIRCUITO ELECTRONICO:

En este capítulo se verá cómo se debe conectar cada dispositivo. En los diagramas se puede notar la ventaja de haber elegido componentes compatibles. El acoplamiento se va a dividir en 5 partes: el microcomputador, alimentación de voltaje, teclado y display matricial. La interface RS232-C se explica en el capítulo V de esta tesis.

4.1 MICROCOMPUTADOR:

Este se divide en microprocesador, memorias RAM, EPROM y puertos de E/S; para una mejor explicación, nos basaremos en la figura 22.

La comunicación entre estos elementos es en forma directa, esto es, no hay ningún circuito adicional entre el microprocesador y sus memorias, sólo los canales y esto se debe a que se seleccionaron dispositivos de la misma familia y totalmente compatibles, por lo que los tres elementos principales usan un canal multiplexado de dirección/datos y

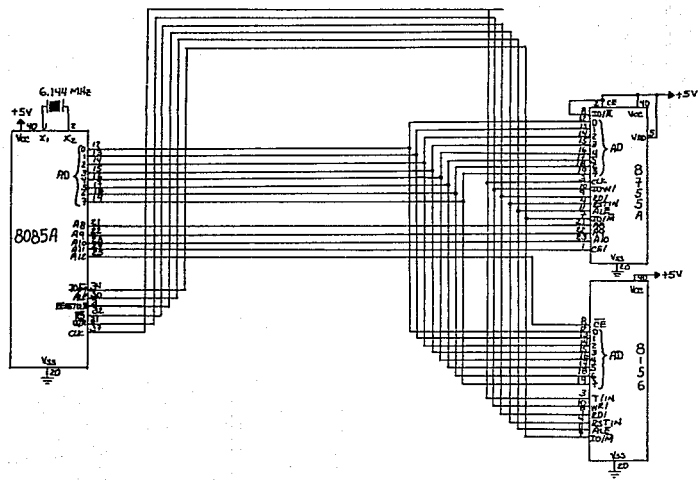


Figura 22

necesitan las mismas líneas de control y de dirección.

Por lo anterior, las líneas ADO a la AD7 del microprocesador tienen que ir conectadas a las mismas líneas del 8156 y del 8755, aprovechando que realizan internamente la demultiplexación.

De las líneas del canal de dirección (A8-A15), A11 al A15 se usan para direccionar dispositivos. A8, A9 y A10, van conectadas a sus similares del EPROM. Las líneas A11-A12 servirán para conectarse a las líneas \overline{CE} (Chip Enable Negado) y CE (Chip Enable) de las memorias EPROM-RAM y de esta manera poder elegir con qué memoria se desea trabajar. También por este canal se deberá enviar la parte alta de la dirección.

No todas las señales que componen el canal de control, tienen una línea correspondiente en los circuitos RAM, EPROM y en el UART, Fig 22. El canal de control de la 8085A se compone, por ejemplo, de las siguientes líneas entre otras:

8085A	NOMBRE	EPROM	RAM	UART
TRAP	INTERRUPCION NO MASCARABLE	---	---	---
RIS 6.5	INTERRUPCION	---	---	RTI
\overline{INTR}	SOLICITUD DE INTERRUPCION	---	---	---
\overline{INTA}	RECONOCIMIENTO DE INTERRUPCION	---	---	---
SO	ESTADO DE CANAL DE DATOS	---	---	---
ALE	HABILITAR LATCH DE DIRECCION	ALE	ALE	ALE
CLK OUT	RELOJ (SALIDA)	CLK	TIMER IN	---

8085A	DESCRIPCION	EPROM	RAM	UART
WR	ESCRIBIR	WR	WR	WR
RD	LEER	IOR	RD	RD
SI	ESTADOS DEL CANAL DE DATOS	---	---	---
IO/M	PTOS. ENTRADA-SALIDA MEMORIA	IO/M	IO/M	IO/M
READY	INFORMACION SOLICITADA ESTA EN EL CANAL DE DATOS.	---	---	---
RESET IN	LIMPIAR	---	---	---
HOLD	RECONOCIMIENTO DE HOLD	---	---	---
HOLD	MANTENER ESTADO	---	---	---

Por lo general, la mayoría de estas líneas interactúan unas con otras y son de mucha utilidad pues son las que dan cauce a la información que el microprocesador va manejando. Es bueno enfatizar que este canal es controlado totalmente por el microprocesador junto con el sistema operativo. La comunicación que este canal mantiene entre las memorias, el microprocesador y demás elementos acoplados a él, es de vital importancia.

El microprocesador 8085A necesita de un reloj externo con una frecuencia de 6.144MHz, éste le permite un rango de tiempo de ejecución de las instrucciones de 1.3 a 5.75 microsegundos. Actualmente se cuenta con cristales de cuarzo, los cuales son muy confiables tanto en la frecuencia que generan, como en la duración del elemento. Lo único que hay que cuidar es que el voltaje de alimentación sea constante, pues de este factor depende que este elemento piezoeléctrico sea estable. La línea de RESET IN va conectada a un boton (push button), para que el usuario pueda accionarlo cuando se

necesita.

4.2 DISPLAY MATRICIAL:

El display matricial será controlado por el microprocesador, a través de los puertos de entrada y salida de las memorias RAM y EPROM con apoyo de los decodificadores.

La conexión se puede ver claramente en el diagrama de la figura 23, la cual nos da una idea bastante clara de cómo se debe hacer el acoplamiento de los dispositivos involucrados en el control del display matricial.

Como se puede observar, son demasiados pines por parte de los displays matriciales, que demandan control para el encendido/apagado de cada led (81 pines en total). Es obvio que sumando todos los puertos disponibles de entrada/salida del sistema, no se acerca a la cifra anteriormente mencionada.

La única forma de controlar tantos pines con tan pocas líneas de entrada/salida relativamente, es manteniendo un solo led prendido en un solo instante de tiempo, barriendo todos los displays a una frecuencia suficientemente alta para que el ojo humano no perciba el apagado/encendido de cada led.

Siguiendo el principio anterior, se pueden agrupar

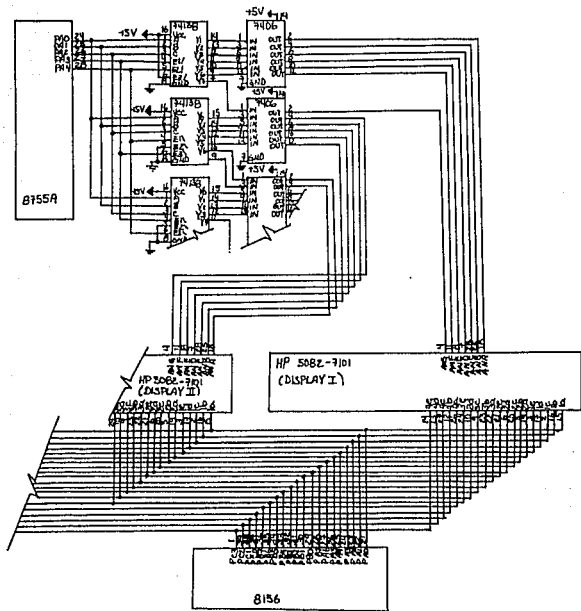


Figura 23

en un solo canal todos los cátodos de las primeras matrices de cada encapsulado, las cuales nombraremos 1a, 1b, 1c, 1d, 1e. Lo mismo para las segundas, terceras y cuartas matrices, obteniendo así cuatro canales, cada uno albergando cinco

líneas, dándonos un total de veinte líneas. Esto es:

8156	NOMBRE	DISPLAY I	DISPLAY II	DISPAY III
PA0	FUERTO A-0	1a	1a	1a
PA1	FUERTO A-1	1b	1b	1b
PA2	FUERTO A-2	1c	1c	1c
PA3	FUERTO A-3	1d	1d	1d
PA4	FUERTO A-4	1e	1e	1e
PA5	FUERTO A-5	2a	2a	2a
PA6	FUERTO A-6	2b	2b	2b
PA7	FUERTO A-7	2c	2c	2c
PB0	FUERTO B-0	2d	2d	2d
PB1	FUERTO B-1	2e	2e	2e
PB2	FUERTO B-2	3a	3a	3a
PB3	FUERTO B-3	3b	3b	3b
PB4	FUERTO B-4	3c	3c	3c
PB5	FUERTO B-5	3d	3d	3d
PB6	FUERTO B-6	3e	3e	3e
PB7	FUERTO B-7	4a	4a	4a
PC0	FUERTO C-0	4b	4b	4b
PC1	FUERTO C-1	4c	4c	4c
PC2	FUERTO C-2	4d	4d	4d
PC3	FUERTO C-3	4e	4e	4e

Tenemos los puertos A y B del 8755 y cada uno de 8 bits en paralelo nos da 16 líneas, más tres del puerto C del

8156, nos da un total de 19 líneas, por lo que son insuficientes. Para solucionar este problema, se pueden usar decodificadores de 3 a 8 líneas, con tres habilitadores. Si se utiliza un decodificador por display, se puede conseguir un buen control, consumiendo sólo 5 pines del sistema, dejando 14 para el teclado. La conexión se puede efectuar de la siguiente manera:

8755A	NOMBRE	DECODIF I	DECODIF II	DECODIF III
PA0	FUERTE A-0	A	A	A
PA1	FUERTE A-1	B	B	B
PA2	FUERTE A-2	C	C	C
PA3	FUERTE A-3	E1	E3	E1
PA4	FUERTE A-4	E2	E1	E3
Vcc	ALIMENTACION	E3	--	--
GND	TIERRA	--	E2	E2

Para ver cómo habilitar cada 74138, ver el apéndice F.

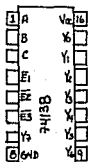


Figura 24

Por medio de los pines selectores (G1, $\overline{G2A}$ y $\overline{G2B}$) de cada decodificador (fig.24), el microcomputador seleccionará con cual 74138 desea trabajar y por consiguiente con que display matircial.

Por la forma en que se encuentran conectados en la figura 23, y siguiendo la tabla de verdad de este circuito (apéndice F), para seleccionar un ánodo de un display en especial, el 8085A deberá enviar la siguiente información por el puerto A del 8755 al los 74138:

ANODO	DISPLAY I	DISPLAY III	DISPLAY III
"A"	(00)H	(09)H	(11)H
"B"	(01)H	(0A)H	(12)H
"C"	(02)H	(0B)H	(13)H
"D"	(03)H	(0C)H	(14)H
"E"	(04)H	(0D)H	(15)H
"F"	(05)H	(0E)H	(16)H
"G"	(06)H	(0F)H	(17)H

En caso que el microcomputador no desee trabajar con los displays matriciales, con enviar un (1F)H se desactivan todos los decodificadores.

Una característica del 74138 es que cada vez que se le habilite correctamente, dependientes del estado lógico que reciban los selectores, el decodificador enviará por sus 8 patas de salida (Y0-Y7) siete estados lógicos altos y uno bajo, esto significaría la habilitación de 6 ánodos y la desactivación de uno; sin embargo, gracias a que hemos colocado

inversores a las salidas de los mismos decodificadores, estamos habilitando sólo un ánodo y los demás no reciben voltaje, facilitando así la multiplexación de los displays.

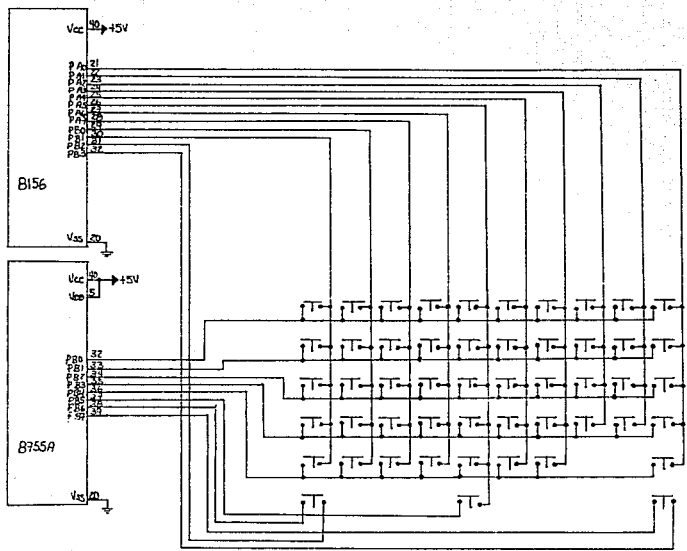
Para activar cualquier diodo del display no basta suministrarle un nivel alto a su ánodo, se le debe hacer llegar un cero lógico a sí cátodo, y de esta forma crear el ambiente de polarización para que funcione. Todos los cátodos de los displays están controlados por los puertos A, B y C del 8156 (fig. 23). A continuación se mencionan los hexadecimales que debe generar el 8085A, para habilitar un cátodo determinado:

CATODO SELEC.	PTO A	CATODO SELEC.	PTO A	PTO B	CATODO SELEC.	PTO B	CATODO SELEC.	PTO B	PTO C
1a	---->FEH	2a	---->DFH->xxx		3a	---->FBH	4a	---->7FH->xx	
1b	---->FDH	2b	---->BFH->xxx		3b	---->F7H	4b	---->xxx->EH	
1c	---->FEH	2c	---->7FH->xxx		3c	---->EFH	4c	---->xxx->DH	
1d	---->F7H	2d	---->xxx->FEH		3d	---->DFH	4d	---->xxx->EH	
1e	---->EFH	2e	---->xxx->FDH		3e	---->BFH	4e	---->xxx->7H	

4.3 TECLADO

La conexión del teclado al microcomputador, se puede apreciar en la figura 25. El puerto B del EPROM seleccionará el renglón a monitorear, y por los puertos A y B del RAM se recolectará la información del teclado por medio de barridos que el 8085A realizará. La secuencia que lleva el barrido y que encuentra la tecla de la letra "g" accionada, es el siguiente:

Figura 25



EVENIO	REGLON EPROM (PTO B)	COLUMNA RAM (PTO B/PTO A)	SIGNIFICADO
1	01H	00H/00H	No hubo pulsación 1er. renglón
2	02H	00H/00H	No hubo pulsación 2do. renglón
3	04H	00H/20H	Se pulsó la 5ta tecla, letra G
4	08H	00H/00H	No hubo pulsación 4to. renglón
5	10H	00H/00H	No hubo pulsación 5to. renglón
6	20H	00H/00H	No hubo pulsación 6to. renglón
7	40H	00H/00H	No se pulso la telca de "CNTRL"
8	80H	00H/00H	No se pulsó la tecla de "SHIFT"

En el apéndice E se puede ver los códigos hexadecimales que se generan en el teclado para la identificación de cada tecla

Por otra parte, no importa que los cátodos de los displays matriciales compartan los puertos del RAM con las columnas del teclado, pues son los puertos del EPROM los que seleccionarán si el microcomputador desea trabajar con el teclado (puerto "B") o con los displays (puerto "A").

4.4 FUENTE DE ALIMENTACION

A continuación se expone cómo energizaremos todo el sistema. Una alimentación eléctrica estable, sin variaciones, eleva considerablemente la confiabilidad del buen funcionamiento de cualquier dispositivo electrónico.

Como se sabe, el voltaje que nos entrega la Cía. de Luz, es de tipo alterno (60Hz), y su nivel de regulación no es del todo muy bueno. Toda la microterminal se debe alimentar

con voltaje continuo y estable, y es por esto que se debe agregar un convertidor de voltaje alterno a voltaje directo, así como un sistema de regulación.

Los niveles de tensión que se necesitan son tres: de +9V, otro de -9V y uno más de +5V. Los +/-9V se requieren para alimentar el DS14C88 y el 7660, y todos los demás circuitos necesitan +5V.

En la figura 26, el voltaje de fuente no regulada excita un regulador de voltaje que produce la salida final. Este voltaje de salida es idealmente constante, no varía ni con los cambios de voltaje de línea, ni con la corriente de carga. En los reguladores prácticos, el voltaje de final es casi constante y el rizado ha sido reducido notablemente.

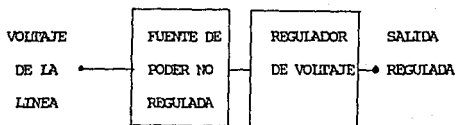


Figura 26

El circuito que se propone para la alimentación de la microterminal, se ve claramente en la figura 27; en él se está usando un transformador (127/10.5 VAC), el voltaje de salida de éste, entra a un puente rectificador de diodos, el cual como su nombre lo indica, rectificará la señal alterna de

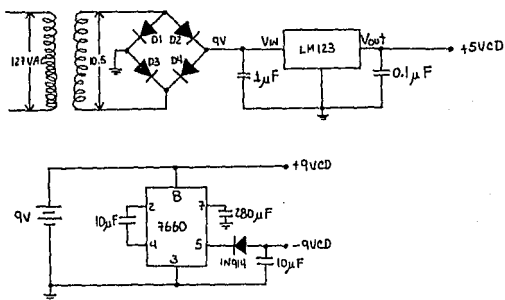


Figura 27

10.5V. Dicho puente esta constituido por cuatro diodos, y si consideramos la señal alterna como una senoide (figura 28) y observando la figura 27, podemos observar que durante un semiciclo, la onda periódica polariza directamente los diodos D2 y D3. Durante el otro semiciclo polariza en directo los diodos D1 y D4, obteniéndose la señal rectificada, figura 29. Cabe mencionar que cada diodo consume 0.7V durante su polarización, mismos que toma de la misma señal que lo alimenta. Lo anterior representa una caída de tension de 1.4V por semiciclo, si tomamos en cuenta que son dos diodos los que se activan por cada medio periodo, la señal que entrega dicho puente es de 9V.

Esta señal rectificada, pasa por capacitores de filtrado para darle una semejanza a un voltaje continuo, sin

embargo todavía lleva pequeñas variaciones. En un inicio, la señal (figura 29), carga al capacitor durante el primer semiciclo, inmediatamente éste tiende a descargarse, pero como el siguiente semiciclo lo vuelve a cargar antes de que se descargue por completo, se obtiene una señal semicontinua, figura 30.

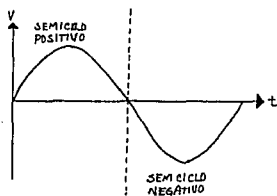


Figura 28

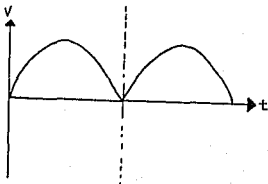


Figura 29

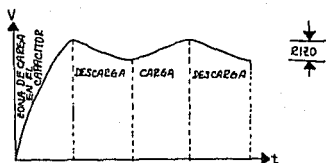


Figura 30

Como esta señal no es del todo continua, se debe pasar por un regulador de voltaje, para lograr que no tenga variaciones. El LM123A es un regulador, con el cual, podemos obtener un nivel de salida estable. Los +9V representados en la figura 30 excita al LM123A, obteniéndose un voltaje de salida de +5VDC, (figura 31) suficiente para alimentar los circuitos: 8085A, 8156, 8755A, DS14C89, 7406, 74HCT138, NCS858 y los displays matriciales. Por otra parte, una batería de +9V tiene la capacidad suficiente para energizar al DS14C88 y al 7660. El 7660 nos entregará los -9V necesarios para energizar también al DS14C88. El uso de una batería para los alimentar al controlador de línea y al convertidor de voltaje, es aceptado por el fabricante de estos circuitos, National Semiconductor Corp. Las señales continuas obtenidas de la batería y del 7660 se pueden apreciar en la figura 32.

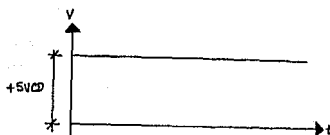


Figura 31

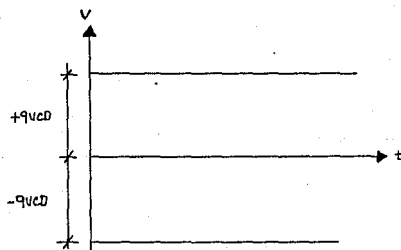


Figura 32

CAPITULO V

INTERFACE DE COMUNICACION UTILIZADA:

Es de vital importancia determinar la forma en como se comunicará la microterminal con el exterior. Se ha elegido el enlace serial RS232 en su modalidad asíncrona. El motivo se debe a que la gran mayoría de los fabricantes de equipo de cómputo ofrecen puertos asíncronos RS232 para poder acoplar periféricos de diversas marcas. Últimamente se pueden encontrar puertos seriales con interface RS422A, sin embargo, aún no es muy común su implementación en los computadores comerciales (apéndice J).

En el capítulo II, se menciona que para el control de los puertos seriales de la microterminal con un UART NSC858 es más que suficiente.

Para indicarle al NSC858 la forma en como debe funcionar, se le dictan diferentes parámetros, mismos que almacena en 11 registros, los cuales se mencionan a continuación:

Temporal de Recepción

Temporal de Transmisión

No Existe

PAGINA

De Modo de Recepción
De Modo de Transmisión
De Modo Global
De Comandos
De Latches Divisores del Generador de Bauds (parte baja)
De Latches Divisores del Generador de Bauds (parte alta)
Del Estatus de las Máscaras de Recepción-Transmisión
Del Estatus de las Máscaras del Modem
Del Modem

5.1 REGISTROS DEL NSC858

REGISTROS TEMPORALES DE RECEPCION Y TRANSMISION

Durante el proceso de recepción, esta localidad guarda la información que se ha recibido del computador en un formato paralelo (8 bits).

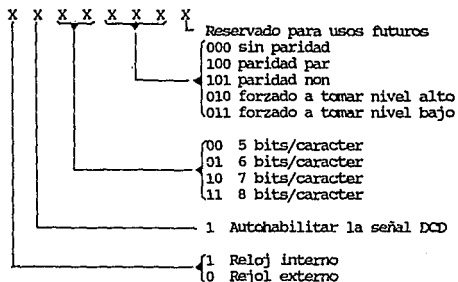
Para la etapa transmisora, en este registro se debe de depositar el byte que se desea transmitir.

REGISTRO DE MODO DE RECEPCION

Aquí se debe indicarle al UART, cómo deberá esperar la información que llegue del computador. Los parámetros que en este registro se manejan son los siguientes: el tipo de paridad, la cantidad de bits/caracter, si se desea

la autohabilitación de la señal DCD para que el NCS858 reciba información; y si se desea que la fuente de reloj de la etapa receptiva sea el generador de bauds interno. El formato del byte se puede apreciar en la tabla E.

TABLA E

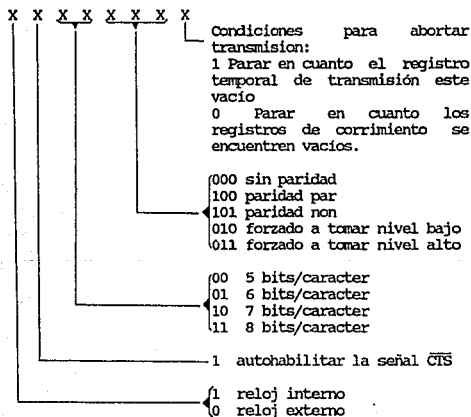


REGISTRO DE MODO DE TRANSMISION

En este registro, se declara la forma en como se deben transmitir los datos al computador. Se programa de forma similar al registro anteriormente explicado en cuanto a la paridad y bits/caracter, así como si se desea que se autohabilite la señal CIS, para que el UART transmita en cuanto se den las condiciones necesarias. Se puede seleccionar que el generador de velocidad para los bauds sea el reloj interno para la etapa transmisora, o se elija un reloj externo. Lo único en que difiere del registro de modo de recepción, es en el bit cero

de este byte, el cual se indica si cada vez que deseemos interrumpir la transmisión, ésta se suspenderá en cuanto el registro temporal de transmisión se encuentre vacío, o en cuanto sus registros de corrimiento (shifts registers), hayan enviado el último bit al exterior. El formato del byte, se puede ver en la tabla F.

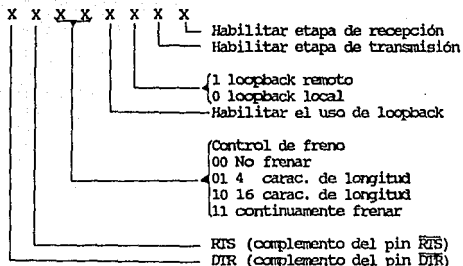
TABLA F



REGISTRO DE MODO GLOBAL

El UART consultará este registro interno de cuatro bits, para saber el factor de reloj que usará para el generador de velocidad para el control de bauds; y el número

TABLA H



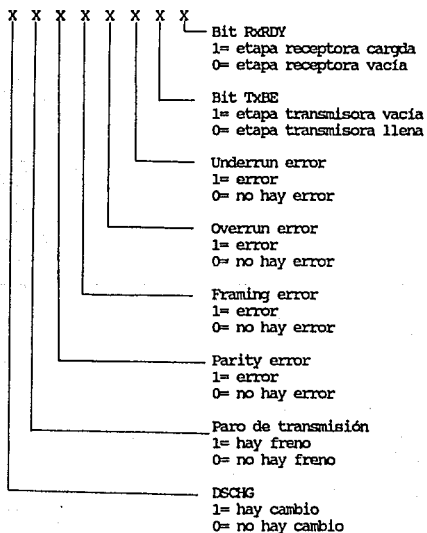
REGISTRO DEL ESTATUS DE TRANSMISION-RECEPCION

Este registro contiene la información del estatus que guarda el NSC858. Cada bit de este byte (tabla I), puede generar una interrupción de transmisión o de recepción ($\overline{\text{RTI}}$), según sea el caso, si cualquier bit se activa, junto con su bit asociado en el registro máscara de transmisión-recepción.

Las opciones que se manejan en este registro son las siguientes: sus etapas de recepción o transmisión estan llenas/vacías de información. Si intenta transmitir y el registro temporal de transmisión y los registros de corrimiento de transmisión estan vacíos (underrun error); puede detectar un error al tratar de leer un caracter disponible y no lo encuentra (overrun error); cuando un bit de parada válido no es detectado (framing error); si hay error de paridad (parity

error); si se detecta un paro de transmisión (transmit break) y el último bit (DSCHG) se activa si hay cambio en los tres primeros bits el registro del estatus de modem y sus bits máscaras asociados. Este último bit se limpia leyendo o escribiendo en los bits asociados del registro de estatus del modem. El formato del byte es el siguiente

TABLA I



REGISTRO MASCARA DE TRANSMISION-RECEPCION

Este registro se usa junto con el registro del estatus de transmisión-recepción, para habilitar o deshabilitar interrupciones. Estos bits máscaras afectarán sólo la interrupción, mas no al registro de estatus de transmisión - recepción.

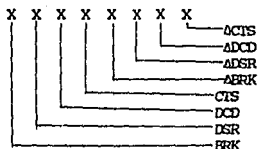
REGISTRO DE ESTATUS DEL MODEM

Es un registro de lectura y contiene bits de información del modem, como son: CTS, DCD, DSR y BRK, y sus bits asociados CTS, DCD, DSR, BRK. Los cuatro delta bits se limpian (toman nivel bajo), cada vez que este registro se lee. Si cualquier bit y su correspondiente delta bit se activa, forzará que el bit DSCG del registro del estatus de transmisión-recepción, tome nivel uno. Los bits CTS, DCD, DSR, representan el valor complementario que tengan las patas \overline{CTS} (26), \overline{DCD} (23) y \overline{DSR} (27). El bit 7 se activa indicando que la etapa receptora ha detectado una condición de freno y se desactiva cuando dicha condición desaparece. El formato del byte se puede observar en la tabla J.

REGISTRO DE LOS BIT MASCARA DE MODEM

Este registro de 4 bits, puede ser accedido para ser consultado o escribir en él. Contiene los bits máscara para los cuatro delta bits del registro de estatus de modem.

TABLA J



Un uno lógico en cualquiera de estos bits y otro en su delta bit asociado dentro del registro de estatus del modem, causará que se active el bit DSCHG en el registro de estatus de transmisión-recepción.

LATCHES DIVISORES PARA EL CONTROL DEL GENERADOR DE VELOCIDAD (BAUDS) Se componen de dos registros de 8 bits, uno almacena la parte baja y otro la parte alta del factor que servirá como divisor de la frecuencia del reloj. Estos registros no se alteran cuando se ejecuta la función de RESET y pueden ser consultados o escribir en ellos.

GENERADOR DE BAUDS PROGRAMABLE

El NCS858 ofrece un generador de bauds programable, y tiene la capacidad de tomar cualquier entrada de reloj (DC hasta 4.1 Mhz) y dividirlo por cualquier divisor desde 1 hasta $(2^{16} - 1)$. La salida de la frecuencia de este generador (esta disponible en las terminales TxC/ERGOUT y

RxC/BR300T, si se seleccionó el uso del reloj interno en los parámetros TxC y RxC) es igual al factor de reloj (1X, 16X, 32X, 64X) veces el rango de bauds.

Los latches de ocho bits almacenan el divisor en un formato de 16 bits. Estos latches deben ser cargados durante la inicialización, para garantizar que el generador de bauds opere como se desea.

La tabla K nos muestra los diferentes divisores de reloj (a 16X) que se deben usar para lograr el rango de bauds deseado, considerando el uso de un cristal de cuarzo de 1.8432MHz.

TABLA K

BAUDS	DIVISOR	PORCENTAJE DE ERROR
1200	96	-----
1800	64	-----
2000	58	0.69
2400	48	-----
3600	32	-----
4800	24	-----
7200	16	-----
9600	12	-----
19200	6	-----
38400	3	-----

5.2 ACOPLAMIENTO DEL NCS858 AL MICROCOMPUTADOR

El NCS858 se puede conectar al sistema por medio de dos de los tres canales básicos: Dirección/Datos y de

Control, figura 33.

El canal de dirección/datos de la 8085A, se conecta directamente al canal del mismo nombre del NSC858, pues este circuito ofrece la demultiplexación de este canal internamente.

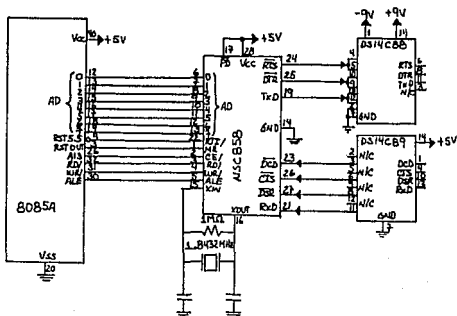


Figura 33

Del canal de control de la 8085A, solo son necesarias algunas de sus líneas, mismas que demandada el UART. Las conexiones son las siguientes: ALE-ALE, \overline{WR} - \overline{WR} , \overline{RD} - \overline{RD} , A13- \overline{CE} , RESETOUT-MR e RST16.5- \overline{RTI} (vía un inversor, pues por ser línea negada, envía un uno lógico cuando no hay interrupción).

Para lograr la comunicación entre el NCS858 y el computador central, se agregan los circuitos DS14C88 y DS14C89, los cuales, convertirán los niveles de voltaje, tal y como se explica mas adelante, figura 33.

5.3 Niveles RS232 y TTL

Los niveles de voltaje que el UART entrega en cada una de sus patas, son TTL (+5V y 0V), figura 34a; y la información que se maneja en el exterior de la microterminal, es de +/-12V (RS232), figura 34b; esto nos obliga a convertir la señal TTL a RS232 en transmisión (DS14C88) y de RS232 a TTL en recepción (DS14C89).

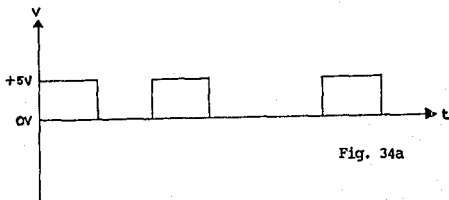
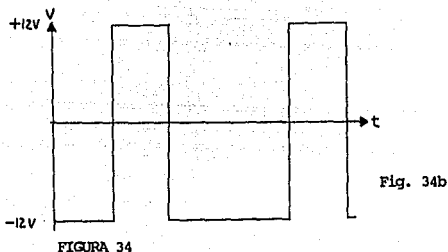
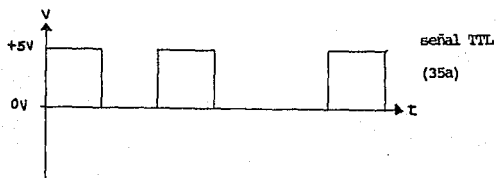


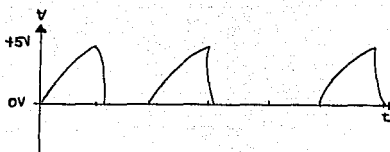
Fig. 34a



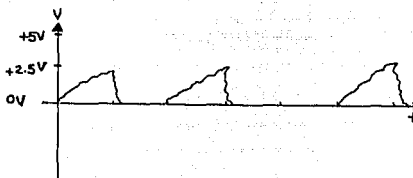
Tal y como se aprecia, la señal TTL sufre un aumento en amplitud al pasar por el convertidor en la etapa de transmisión, o una disminución en la etapa de recepción. La ventaja es la siguiente: cuando se transmiten pulsos TTL (fig. 35a) por alambre telefónico, estos pasan por un efecto capacitivo que se produce en el cable, por lo que la señal se atenúa y se distorsiona, figura 35b.

Figura 35





la señal sufre alteraciones por el efecto capacitivo en la línea (35b)

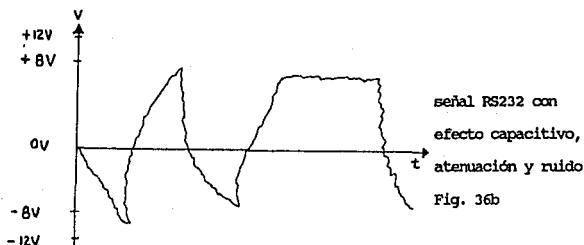
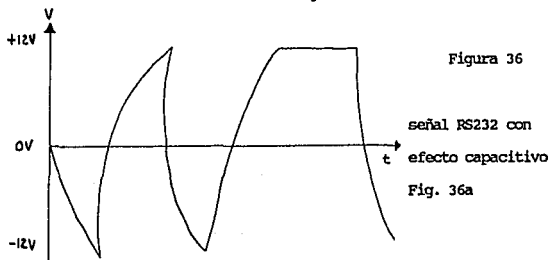


la señal aparte del efecto capacitivo, se encuentra atenuada y con ruido, también por la línea telefónica (35c)

A lo anterior hay que agregarle que si la señal sufre mucha atenuación (fig. 35c) y llega a niveles de los +2.2 VDC, el dispositivo discreto que trabaje a niveles TTL, no podrá distinguir si el bit recibido tiene un nivel lógico de 1 ó 0.

Ahora bien, si convertimos la señal de nivel TTL a RS232, suprimiremos los inconvenientes anteriores, pues al convertirse a señal bipolar +/- (más/menos), la atenuación pasa a ser un factor secundario al igual que la distorsión. En la figura 34 se aprecia fácilmente que para un + 5VDC TTL corresponde un -12 VDC y cero volts TTL un +12 VDC. Si sometemos una señal de +/-12 VDC como la que aparece en la

figura 34b, a un efecto capacitivo y a un efecto de atenuación, obtendremos las señales de la figura 36.



Obviamente, si al DS14C89 le llega una señal como la de la figura 36b, si puede distinguir si hay amplitud positiva o amplitud negativa, por lo que la interpretación para unos y ceros al convertir la señal RS232 a TTL se lleva a cabo, aún y cuando este atenuada, no pasando lo mismo con la señal de la figura 35c.

Puede darse el caso que la atenuación y efecto capacitivo para distancias cortas no sea demasiada y se pueda transmitir señal TTL por el puerto serial, vía un multiplicador de modem (null modem), pero esto nos recortaría la eficiencia de la microterminal, en cambio, si transmitimos a niveles RS232 obtendremos mas eficacia. Además, la mayoría de los fabricantes de computación se han estandarizado para que a sus equipos se les puedan conectar periféricos con interfaca RS232.

La figura 37 nos muestra la disposición de pines en un conector de 25 pines RS232.

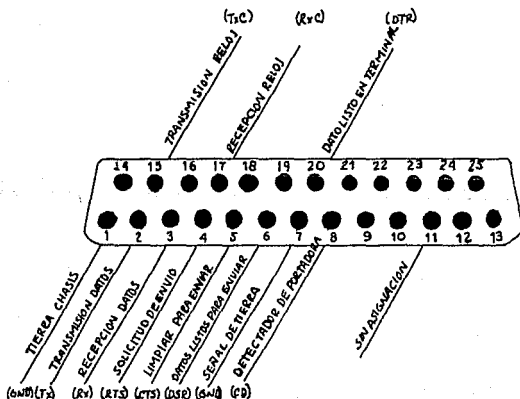


Figura 37

CAPITULO VI

TEORIA DE FUNCIONAMIENTO:

Tal y como está sucediendo últimamente en todo equipo de cómputo, el hardware tiende a ser miniaturizado y junto con el software, nos brinda dispositivos más poderosos. El que actualmente contemos con memorias en las que puedan guardar grandes cantidades de información, permite que podamos explotar el hardware por medio del software de una manera más eficiente.

La microterminal no es ajena a lo comentado en el párrafo anterior, por lo que se aprovechará al máximo las bondades que el sistema del microcomputador ofrece.

A continuación explicaremos por partes cómo debe funcionar la microterminal, a partir de la programación que grabemos en el EPROM del sistema. Empezaremos con lo que se haría como primer paso: encenderla, esto conlleva a una serie de autodiagnósticos, como son, en un primer término, probar su memoria RAM, luego (como opción) probar el teclado, y por

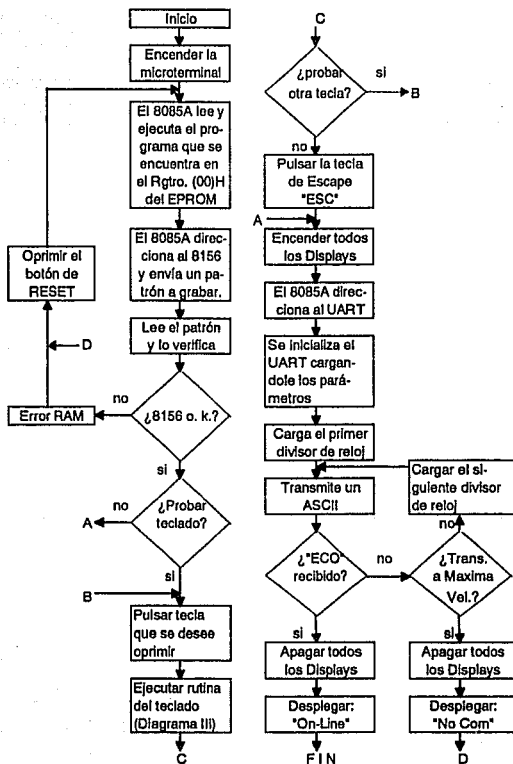
último, tratar de establecer comunicación con el computador central.

6.1 AUTODIAGNOSTICO:

Para que la microterminal funcione, hay que energizar todo el sistema. Al encenderla, provoca que se origine un impulso que limpia al sistema (reset), esto fuerza que las primeras direcciones que lea el microprocesador sea el registro cero del EPROM, luego el uno y así. Las primeras direcciones le dirán al microprocesador que se autodiagnostique, esto es, que las memorias, puertos de E/S y el microprocesador, estén bien y respondan a los comandos de prueba del programa.

Siguiendo el diagrama I, se pueden apreciar claramente las diferentes etapas por las que debe pasar el autodiagnóstico de la microterminal. Al inicio, al ser limpiado (encendido) el microprocesador, manda leer automáticamente el registro (00H) del EPROM, es por esto, que en sus primeras localidades se deben cargar las rutinas de autopruueba. Si por alguna causa (el impulso de reset no fue suficiente) no empezará el microprocesador a leer el registro cero, se origina un error de direccionamiento, por lo que se tendrá que oprimir el botón de reset, y así limpiar todos los

Diagrama I



registros.

El siguiente paso, es averiguar si la memoria RAM está funcionando bien, para eso, el microprocesador direcciona al 8156 y le envía un patrón predefinido a grabar, acto seguido, verifica que dicho patrón se encuentre almacenado, leyendo todas las localidades de memoria. Si se encuentran discrepancias durante la verificación, enviará un mensaje a los displays diciendo "ERR RAM". Ahora bien, si no encontró errores en escritura ni en lectura, el microprocesador seguirá ejecutando el programa de autoprueba.

La prueba del teclado es opcional, por lo que una vez que la microcomputadora se autodiagnosticó, ofrece la posibilidad de revisarlo. De ser así, la rutina exige que el usuario pulse las teclas que desee probar (no es necesario oprimir todas, ni llevar una secuencia). Una vez que se haya concluido la prueba, se deberá pulsar "ESC" (escape) para que la microterminal siga ejecutando su rutina de autodiagnostico.

Ahora bien, si no se desea probar el teclado, con solo pulsar la tecla de "RETURN" (retorno), la microterminal seguirá con la rutina de revision a la siguiente etapa.

Para la comunicación al computador, la

microterminal enviará caracteres para poder fijar la velocidad a la que habra de transmitir y recibir información.

Para realizar estas pruebas, el microprocesador programará al UART, enviándole información predefinida dentro del sistema operativo. Para inicializar el NSC858 hay que direccionarlo.

Los siguientes registros del NSC858 deberán recibir los siguientes hexadecimales, para habilitar las diferentes opciones que se explican en el capítulo V de esta tesis.

REGISTRO DE MODO DE RECEPCION: Recibirá un (E0)H, para declarar paridad non, 7bits/caracter, autohabilitación DCD y RxC interno. Se eligió sin paridad, son 7 bits/caracter los que se declararán, pues dentro del formato ASCII (American Standard Code for Interchange Communication) son el número que se maneja (apéndice I). Como activamos la opción DCD, el UART se autohabilitará tan pronto y como reciba una solicitud de recepción de datos. El reloj, se eligió interno, pues es la forma en la cual indicamos a que velocidad se espera recibir los datos.

REGISTRO DE MODO DE TRANSMISION: El hexadecimal

correspondiente es un (E0)H, para que las siguientes opciones se habiliten: que se pare el proceso de transmisión si los registros de corrimiento (que serializan la información a transmitir), están vacíos; que no tenga paridad; y que el UART sepa que el carácter a transmitir es de 7 bits; que la etapa transmisora se autohabilite en cuanto la línea CTS se active y se eligió el reloj interno para que el NCS858 sepa a que velocidad tiene que transmitir.

REGISTRO DE MODO GLOBAL: El microprocesador hará llegar un (01)H y de esta forma indicarle al UART que el factor de reloj con el que trabajará es de 16X y que son dos los bits de parada que debe enviar cada vez que transmita un ASCII. Se eligió el factor de 16X debido a que es el factor del que menos desviación obtuvimos en control el bauds, (tabla K).

REGISTRO DE COMANDOS: Se activan los dos primeros bits, que sirven para habilitar la recepción y la transmisión, además de ser un requisito si las opciones de autohabilitar DCD y CTS en los registros de modo de recepción y transmisión fueron elegidas. Los bits en los cuales se declaran si se desea trabajar con retornos (loopback) locales o remotos, no importa el valor que tomen, pues la opción de habilitar el uso de retornos esta desactivada. Para el control de freno, lo mas común dentro de las terminales de cómputo es usar NO BREAK (no

frenar). Los bits que controlan RTS y DTR no importa el valor de un inicio (y los declararemos como ceros), pues éstos cambian dependiendo si se esta transmitiendo o recibiendo información; para lograr lo anterior, a este registro se le debe hacer llegar un (03)H.

LATCH DIVISOR PARA EL CONTROL DEL GENERADOR DE VELOCIDAD (bauds): Una vez que se ha programado el NSC858, se procede a la última fase de su inicialización, esto es, indicarle a que velocidad debe de realizar la prueba de transmisión. Para esto se carga un divisor en dos registros, en uno guarda la parte alta y en otro la parte baja. Este divisor lo usa el UART junto con el factor de reloj (16X) para controlar la velocidad de transmisión (dentro del generador de velocidad para alcanzar el nivel de Bauds deseado), y a que velocidad debe esperar el la información a recibir. A continuación se mencionan los divisores que se usarán para las pruebas de velocidad:

HEX	DIVISOR	BAUDS
0060	96	1200
0040	64	1800
003A	58	2000
0030	48	2400
0020	32	3600
0018	24	4800
0010	16	7200
000C	12	9600
0006	06	19200

Una vez cargada la información en los registros del UART, el microprocesador intenta transmitir un ASCII, en caso que la línea de comunicación no este trabajando bien y la microterminal no pueda enviar la información al exterior, desplegará un mensaje "ERR COM"; si la línea esta bien, transmite la información a 1200bauds, que es el primer divisor que se le cargo al UART. La microterminal queda en espera del ECO del computador, esto es, la confirmación; en cuanto el UART le avise al 8085A que ha recibido el ECO, se procederá a comparar la información transmitida, con la que el computador envió (ECO), si no hay diferencias, manda a desplegar el mensaje "ON LINE" en los displays.

En caso que el ECO no llegue, el microprocesador direccionará los latches divisores para el control del generador de velocidad, y cargará el siguiente divisor, que será el (0040)H, para lograr una velocidad de 1800bauds, y se vuelve a repetir el proceso explicado en el párrafo anterior.

Si después de haber probado con todos los divisores, la microterminal no logró establecer comunicación con el computador central, desplegará un mensaje "NO COM".

Mientras la microterminal trata de establecer comunicación con el computador central, el microprocesador

encenderá todos los leds de los displays. Durante esta lapso de tiempo el usuario podrá realizar una inspección visual del estado que guardan éstos, los cuales se apagaran para mostrar el resultado final de la rutina de comunicación ("ON LINE" o "NO COM"). Cabe recordar que mientras los displays se encuentren encendidos, siempre estarán multiplexados.

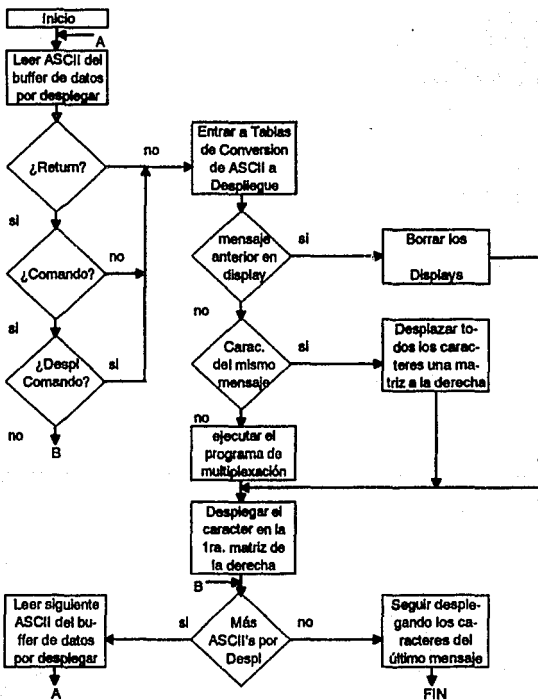
Es recomendable, que cada vez que se detecta un error en las rutinas de autodiagnóstico, se oprima el boton deRESET, y de esta manera provocar que la microterminal se reinicialize.

6.2 CONTROL DE LOS DISPLAYS:

El control de encendido/apagado de cada led del display matricial, tiene que ver más con el sistema operativo que con el hardware.

Para hacer que uno de los leds del display encienda, se necesita polarizarlo, esto es, crear un ambiente eléctrico, que ayude a que el flujo de electrones se realice, sin que se encuentre ningún obstáculo. El microprocesador le dará dicho ambiente a cada led para que encienda cada vez que así se requiera, con la ayuda de inversores y decodificadores (figura 23) Basándonos en la figura 19, para encender un led

DIAGRAMA II



con coordenadas 1b-And, se tiene que poner un 0 lógico en 1b y un 1 lógico en And. En el Diagrama II, se puede ver más detalladamente qué pasa cuando el microprocesador desea desplegar algún caracter.

Ahora bien, como sólo son 12 las matrices disponibles (una por cada caracter), el despliegado de la información se presentará de tal forma que el primer caracter de la primera palabra aparecerá en la primera matriz de la derecha, para presentar el segundo, el microprocesador hará que el primer caracter desaparezca de la primera matriz y aparezca en la segunda matriz y de esta forma poder desplegar el siguiente caracter de la palabra. En conclusión, los caracteres se van a ir corriendo de derecha a izquierda. En aquellos casos en los que en un instante se llegaran a utilizar las 12 matrices y haya más caracteres por desplegar, seguirán el recorrido normal, obligando a que el caracter que ocupe la última matriz de la izquierda desaparezca y así aparezca el decimotercer caracter en la primera matriz de la derecha.

Una vez que la microterminal ha recibido información del computador central, empieza a ejecutar la rutina de despliegue de caracteres, en la cual convierte el código ASCII en códigos para desplegar información en los displays matriciales.

La rutina de despliegue empieza cuando el microprocesador lee el primer caracter ASCII del buffer de datos que recibió por el UART, y decide si corresponde a un EOC o es un mensaje del computador; para poder llevar a cabo la decisión, comprobará si la tecla de "RETURN" fué pulsada, pues es ésta, la que separa los mensajes que el usuario envía de los que genera el computador.

En caso que la información a desplegar sea un mensaje del computador, entra a tablas de conversión de ASCII a código de despliegue. Una vez que se tiene lista la información para enviarla a los displays matriciales, entra a dos etapas de evaluación, en las cuales determina primero, si hay algunos caracteres que estén siendo presentados en los displays que correspondan a un mensaje anterior, de ser así, deshabilita los displays y acto seguido envía el caracter a desplegar en la primera matriz de la derecha. Cabe mencionar que como había un mensaje desplegado anteriormente, la subrutina de multiplexación de los displays matriciales está funcionando.

Si no había mensajes anteriores en los displays, quiere decir que el caracter a desplegar corresponde a un nuevo mensaje y averigua si hay caracteres de este mensaje desplegados. En caso afirmativo, desplaza todos los caracteres

una matriz a la izquierda y presenta el nuevo caracter en la primera matriz de la derecha. En caso que el ASCII a desplegar sea el primer caracter de un mensaje nuevo, se activa la subrutina de multiplexación y se presenta el caracter en la primera matriz de la derecha.

Inmediatamente averigua si hay mas caracteres por desplegar, consultando el contador de ASCII's del buffer de datos. En caso que no haya ninguno, se seguirán presentando los caracteres del mensaje que se este manejando en esos momentos en los displays matriciales. Si hay mas caracteres por desplegar, lee el siguiente ASCII, y vuelve averiguar si se trata de un EOO, o de un mensaje del computador, ejecutándose otra vez la rutina que se ha venido explicado.

Se puede dar el caso que al momento de estar evaluando un ASCII, éste represente un caracter de control, por lo que comprobará si ese caracter de control se debe desplegar o no, en caso que se pueda desplegar, le dará un trato normal, esto es lo enviará a las tablas de conversión de despliegue. En caso de que sea un comando no desplegable, lo desechará e irá a verificar si hay mas ASCII's por presentar en los displays.

Se ha venido manejando la idea de multiplexación en los displays matriciales. Es un programa de barrido sobre

12	(06)H	(FD)H	S
13	(06)H	(FE)H	S
14	(00)H	(7F)H	i
15	(02)H	(7F)H	i
16	(03)H	(7F)H	i
17	(04)H	(7F)H	i
18	(05)H	(7F)H	i
19	(06)H	(7F)H	i

y para lograrlo, el microprocesador ejecutará el programa de despliegue, enviando en un primer estado de tiempo, la información necesaria para habilitar el renglón de los displays. Esto se logra energizando con un 1 lógico el ánodo C (AnC), y luego que el cátodo 1e tenga un estado lógico 0; todos los demás ánodos deberán recibir nivel 0. En un segundo estado de tiempo el microprocesador volverá a suministrar un nivel alto exclusivamente al ánodo C (AnC), pero en esta ocasión hará que el cátodo 1d reciba un nivel bajo. Y así sigue hasta llegar al estado de tiempo 19, en donde el ánodo G (AnG) será el único que reciba un 1 y el cátodo 2c un 0.

Si unimos uno a uno todos los estados de tiempo, con una duración entre uno y otro de medio segundo, podremos apreciar el barrido que el microprocesador realiza sobre los displays. Ahora bien, si acortamos el tiempo entre un estado y otro, de tal manera que se vayan ejecutando a una frecuencia de unos 100 Hz mínimo, (desplegar la palabra "si" 100 veces en un segundo), el ojo humano no notará el barrido en los displays y parecerá que están siempre encendidos todos los leds.

6.3 CONTROL DEL TECLADO:

A continuación se explica, cómo la microcomputadora (8085A-8156-8755A) reconoce al teclado y todo lo que se requiere realizar para que se reconozca un dato.

El programa que controla el teclado, sigue la siguiente secuencia de barrido: Primero investiga si el Control esta siendo oprimido, luego revisa si la tecla de cambio "SHIFT" esta accionada y por último rastrea si alguna otra tecla es pulsada.

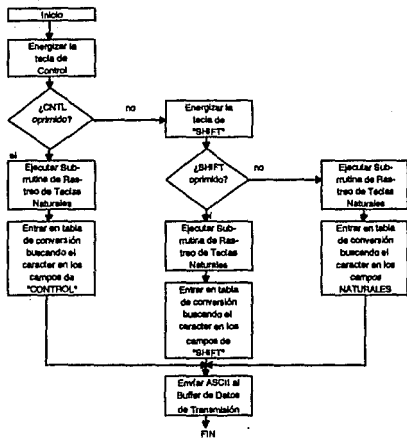
Es importante para la microterminal saber si las teclas de "CONTROL" o "SHIFT" estan siendo o no oprimidas, pues de esto depende en cual campo de las tablas de conversión va a realizar la búsqueda.

La forma en cómo lo realiza se explica a continuación: el 8085A envía un 1 lógico al renglón de la tecla de "CONTROL" (el puerto b del EPROM toma el valor (40)H) y verifica el estado que guarda la pata 31 (PB2) del RAM; si éste es alto, la tecla en cuestión está siendo accionada, el microcomputador la registra y el programa se va a la subrutina rastreo de teclas naturales. En caso contrario, el 8085A hace que el puerto B del EPROM deje salir un (80)H (energiza el

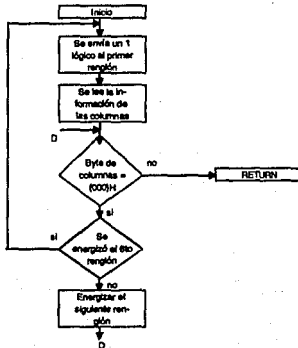
renglón de la tecla de CAMBIO) y evalúa el estado de esta tecla, leyendo la información de la pata 32 del 8156 (1= oprimida, 0= no oprimida), si esta tecla ha sido pulsada la registra en su memoria, en caso contrario, significa que la teclas de "CONTROL" y "CAMBIO" no fueron oprimidas y también se va a buscar si alguna otra tecla natural está siendo accionada.

Una vez que la microterminal sabe en que campos va a efectuar la búsqueda, necesita saber que tecla natural (dentro de los 6 primeros renglones) está pulsando el usuario. Para realizar lo anterior, se ejecuta una rutina de rastreo, el cual consiste en enviar un uno lógico al primer renglón, del teclado, esto es, envía un hexadecimal (01)H por el puerto B del EPRCM. Inmediatamente, el microprocesador manda a leer la información que se encuentra en los puertos A y B del RAM (columnas) y compara si el byte recibido por dichos puertos, es diferente a (00)H. En caso que sean iguales, se pregunta si el renglón energizado anteriormente fué el último del teclado, como no lo es, envía ahora al puerto B del 8755A, un (02)H y vuelve a repetir el proceso de lectura en los puertos A y B del 8156. Cuando ha energizado el último renglón, esto es, ha enviado un (200)H, por el puerto B del EPRCM, se ha completado un barrido de renglones, por lo que el siguiente renglón a energizar es el primero, y de esta forma se empieza un nuevo barrido.

DIAGRAMA III



(Subrutina de Rastreo de Teclas Naturales)



En caso que encuentre un byte diferente a (00)H en el momento de analizar la información proveniente de las columnas, guarda en memoria el último byte que envió por el puerto B del EPROM (el renglón energizado), y los hexadecimales que recibió por los puertos A y B del RAM (la columna).

En la tabla que se encuentra en el apéndice E, encontraremos las hexadecimales y los estados lógicos que el microprocesador debe enviar y esperar para la decodificación de las teclas según su ubicación.

Ahora se procede a ubicarse dentro de las tablas de conversión de código de teclado a código ASCII, éstas están estructuradas de la siguiente manera: la llave primaria es el renglón del teclado; la llave secundaria es la columna del teclado y la tercera llave sirve para establecer si se toma el carácter de cambio, de control o las naturales. Una vez averiguado lo anterior, se obtiene el código ASCII correspondiente, mismo que se deposita en el buffer de transmisión en RAM, se envía al computador central vía el UART, y el programa se vuelve a reiniciar.

Si consideramos que el barrido que el microprocesador ejerce sobre el teclado se ejecuta por lo menos 20 veces cada segundo, se puede decir que el teclado está

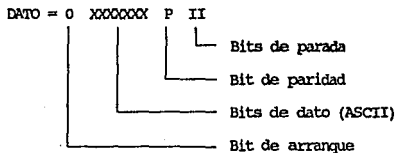
siempre atendido.

Es importante mencionar que no importa que los puertos A y B del 8156 estén compartidos con los cátodos de los displays matriciales, pues la lectura del teclado lo realiza en un instante de tiempo en los cuales los decodificadores, están anulificando los ánodos de todos los displays; y en otro instante de tiempo, el despliegue de información.

6.4 COMUNICACION CON EL COMPUTADOR:

La comunicación con el computador central, se logra por medio de una interface RS232, dentro de un formato asíncrono. Para el manejo de este protocolo se cuenta con un UART.

El formato característico de una comunicación asíncrona es el siguiente:



Nótese que el dato que se transmite es de 11 bits ya que el ASCII esta constituido por 7 bits, y se debe a que tal y como podemos ver en el apéndice I, que el bit cero es siempre cero, por lo que cuando se trabaja con este código, se entienda que el primer bit tiene estado lógico cero, procediéndose a buscar los 7 bits restantes. Lo anterior es un estándar dentro de las comunicaciones entre los fabricantes.

TRANSMISION

Cada vez que el microprocesador necesita enviar un dato al computador central, seguira la secuencia descrita en el Diagrama IV, mismo que a continuación se explica.

En un inicio, el 8085A verifica si en el buffer de transmisión de datos (que se encuentra en memoria RAM), existe información a transmitir, si no hay nada, vuelve a verificarlo cierto tiempo después. En caso que sí haya un ASCII por enviar, habilita al UART (le hace llegar un cero lógico a su pata \overline{CE}), luego envía a los latches de dirección del NCS858 (previamente activados por la señal ALE), la dirección (00)H para indicar que se accesará el registro temporal de transmisión; hace la la línea \overline{RD} tome nivel bajo, y envia por el canal de dirección/datos el ASCII a transmitir.

El bit TxBE toma nivel cero en el registro de

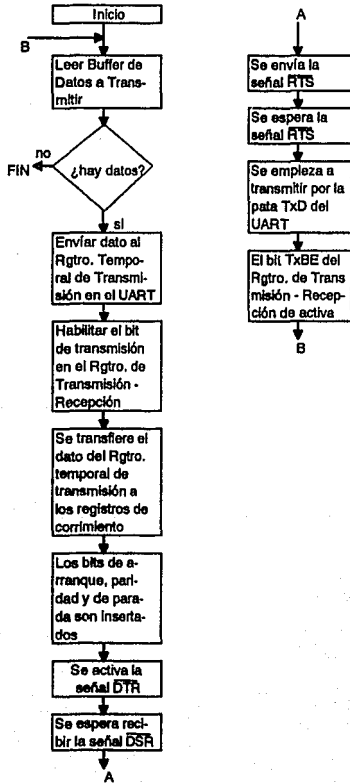
estatus de transmisión, indicando que el buffer de transmisión esta lleno. El UART procede a transferir el ASCII del registro temporal de transmisión (que contiene la información en paralelo) a los registros de corrimiento de transmisión y de esta forma serializar el dato. Se insertan los bits de arranque, de paridad y los de parada, tal y como se declararon durante la etapa de inicialización del NSC858.

Se habilita la pata \overline{DTR} (Data Terminal Ready) para indicar que hay un dato por transmitir. El UART queda en espera de recibir la confirmación por la pata \overline{DSR} (Data Set Ready), el cual significa que hay una línea de comunicación disponible. Se activa la pata \overline{RTS} (Request to Send), para solicitar la transmisión del ASCII, y queda en espera de la confirmación, la cual la detecta por la pata \overline{CTS} (Clear to Send).

Una vez que el protocolo anterior se ha llevado a cabo, se empieza a transmitir el ASCII por la pata TxD, en cada transición alto-bajo de la señal TxC. Una vez que se acabó de enviar la información al computador central, el bit TxBE toma nivel alto para indicar que el buffer de transmisión esta vacío y disponible si se desea transmitir más información.

Si hay más ASCII's por transmitir, el

DIAGRAMA IV



microprocesador, volverá a cargarlos en el UART siguiendo el procedimiento explicado en párrafos anteriores. Para separar una ASCII de otro, la línea TxD mantendrá una condición de "marca" (uno lógico) entre la transmisión de uno y de otro.

Si no hay mas ASCII's por enviar al computador central, la 8085A deshabilitará al UART, provocando con esto, que la señal RTS se desactive indicando, el fin de la sesión.

Cabe mencionar que puede darse el caso que al estar serializando el ASCII (del registro temporal de transmisión a los registros de corrimiento) el UART no le da tiempo de leer un bit y el siguiente se haga presente, provocando un error (overrun), en este caso, el byte es regresado al registro temporal de transmisión, y el UART lo volverá a tratar de serializar. El bit tres del registro del estatus de transmision-recepcion se activa.

RECEPCION

Para que el NCS858 sepa que va a recibir información (diagrama V), necesita que su pata \overline{DCD} se active, inmediatamente checa si se eligió la autohabilitación del bit DCD en el registro de modo de recepción y que $RxE=1$ en el registro de comandos (mismo que se declaró durante la

inicialización del NCS858), por lo que la etapa de recepción se activa automáticamente.

La línea RxD está normalmente en un estado alto. Una transición de alto a bajo sobre esta línea, significa el comienzo de un bit de arranque. La validez de este bit de arranque se chequea nuevamente muestreándolo en su centro nominal. Si RxD aún tiene nivel bajo, es un bit de arranque válido y el proceso de recepción del carácter se efectúa en los registros de corrimiento de recepción, mismo que es inmediatamente transferido al registro temporal de recepción, en donde el ASCII recibido es convertido de un formato serial a uno paralelo (sin bits de arranque, paridad y de parada), y de esta manera está disponible al microprocesador.

El bit RxDY se activa en el registro de estatus de transmisión-recepción, indicando que el registro temporal de recepción contiene información, y la pata \overline{RTI} (Receive-Transmit Interrupt) se activa.

En cuanto la 8085A reciba una interrupción (del \overline{RTI}), acabará de ejecutar la instrucción en proceso, direccionará al UART, envía la dirección para leer el registro de estatus de transmisión-recepción y chequeará si el bit RxDY=0 (la lectura de este registro, fuerza que RxDY=1 y \overline{RTI} =1), de

DIAGRAMA V

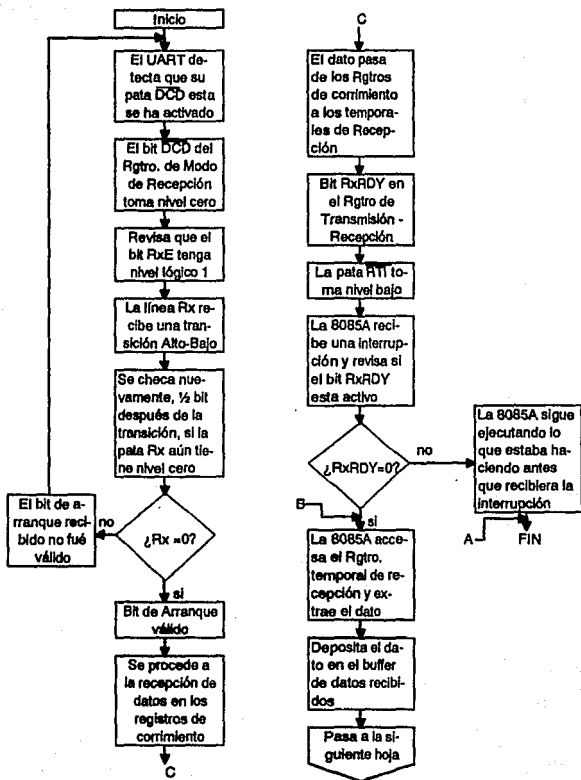
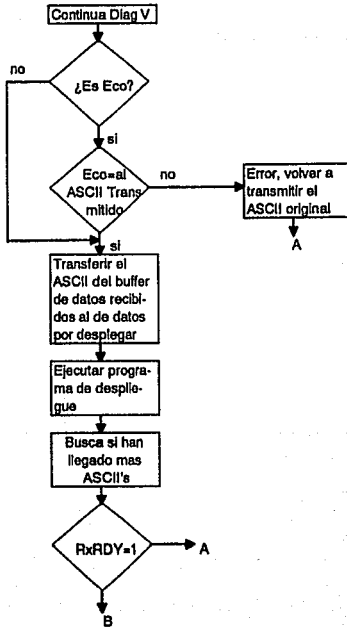


DIAGRAMA V

(Cont.)



ser así, accesa el registro temporal de recepción para sacar el ASCII y guardarlo en el buffer de datos recibidos, continuando el microprocesador con la tarea que estaba ejecutando cuando se le interrumpió.

En caso que la señal RRDY tenga nivel 1 cuando el microprocesador reciba una interrupción por parte del UART, éste ignorará dicha interrupción y seguirá realizando lo que estaba ejecutando

En caso que sea un EOO, lo comparará con el ASCII original, y si son iguales, transferirá el ASCII al buffer de datos a desplegar, y empezará el programa de despliegue de caracteres. Otra posibilidad es que el ASCII que recibió la microterminal, sea una respuesta del computador a una petición solicitada por el usuario, de la microterminal, para este caso, la transferencia del buffer de datos recibidos al de datos a desplegar se hace automáticamente, sin checar EOOs. La tecla de RETURN separa los mensajes de la microterminal (que sí necesitan EOO), de los que envía el computador (no necesitan EOO).

En caso que esten llegando más ASCII's, en una forma continua, esto es, en una sesión, el 8085A, los irá sacando del UART y los guardará en el buffer de datos recibidos

y de esta manera no provocar el bloqueo del NCS858.

Cabe mencionar que durante la transferencia del ASCII de los registros de corrimiento al registro temporal de recepción, el UART revisa la paridad del dato recibido. En caso que detecte algún error, la bandera de paridad se activará en el registro de estatus de transmisión-recepción. Si no recibe los bits de parada, también se genera un error (framing error), ó si al estar leyendo los caracteres, no puede leer uno y el siguiente se hace presnete, entonces se dice que hubo un error de overrun. Todos estos errores, generan que la señal RTI se active, informando al microprocesador que hubo un error, el cual estara a su disposición en el registro de estatus de transmisión-recepción.

C O N C L U S I O N E S

A continuación se detalla el punto a que se llegó en cuanto a la concepción inicial de la Microterminal Inteligente.

1. El concepto básico de la Microterminal Inteligente, se basa simplemente en la Portabilidad que ésta debe ofrecer al usuario.

Es por lo mismo que para cumplir con esta característica, se estudiaron que los elementos que constituyen la Microterminal fueran lo más pequeños posibles, apoyándonos plenamente en la tecnología electrónica ya existente.

Analizando los elementos que más representan voluminosidad en una terminal de video normal; resaltan dos elementos básicamente, estos son el Teclado y el Monitor. El monitor, por estar constituido por un Tubo de Rayos Catódicos, es necesario alto voltaje, lo que trae como consecuencia, circuitos de control y alimentación especiales, con el consecuente aumento en el tamaño de la fuente de alimentación en capacidad, peso y dimension. El teclado por el espacio que se guarda entre tecla y tecla y el tamaño de las mismas, lo vuelve también de dimensiones grandes para entrar en el concepto de portabilidad.

Para lograr el que sea maniobrable y ligero a la vez, se han tenido que sacrificar el Tubo de Rayos Catódicos y el espacio entre tecla y tecla, cuidando la distribución estandar. El primero lo sustituimos por tres circuitos constituidos por arreglos de leds en forma matricial; y el segundo lo

reducimos a una tercera parte del tamaño de un teclado de una computadora personal y a un grosor de escasos 3 milímetros.

2. Uno de los objetivos que no pudo ser cubierto al 100%, fué el de la independencia en la comunicación; esto es, en donde se desee trabajar con la microterminal, deberá haber un cable de comunicación conectado desde el computador central por uno de sus lados y rematado en un conector DB25 en su otro extremo.

Lo que en un inicio se pensó, fue el de implementar un circuito de radiofrecuencia, y que transmitiera y recibiera información desde otro transmisor-receptor, mismo que se acoplaría a un puerto del computador central. Lo anterior nos llevaba, a desarrollar los dos circuitos de comunicación (el que llevaría la microterminal, y el del computador central), lo que desviaría en un gran porcentaje el objetivo principal de esta tesis que es la Microterminal Inteligente con el Microprocesador INTEL 8085A.

3. No se pudo alimentar totalmente mediante baterías toda la Microterminal, ya que la demanda de corriente en los 5VDC es alta, por lo que energizarla por este medio, provocaría que éstas se descarguen rápidamente, provocando desconfianza en su uso y en una etapa comercial, sería un punto negativo. Es por lo anterior que se desarrollo una fuente de poder regulada, por lo que debe haber un contacto eléctrico para su alimentación.

4. Evaluando finalmente la Portabilidad de la microterminal, podemos decir que ésta se cumplio en cuanto a tamaño y

dimensiones, pues todos los componentes seleccionados, contribuyen a su miniaturización y el usuario puede transportarla consigo, sin embargo para su funcionamiento depende de una instalación de comunicación y de ambiente de fuerza.

Las conclusiones a que se llegaron dentro del desarrollo de la Microterminal Inteligente se mencionan a continuación.

5. No se pudo usar el puerto serial que el 8085A ofrece, para la comunicación de la Microterminal con el exterior, esto es, usando las patas: SID (entrada serial de datos/serial input data) y SOD (salida serial de datos/serial output data), como señales TXD y RXD respectivamente; ya que el manejo de las señales de control: RTS, CTS, DSR, etc, representaron un problema, pues el sistema operativo elevaba su grado de complejidad, además de consumir recursos del propio microcomputador. Es por lo mismo, que se utilizó un UART (NSC858), el cual, para ser acoplado al 8085A, sólo necesita de algunas líneas de los canales de control y de dirección/datos. El control de este dispositivo por parte del microcomputador mediante el sistema operativo, es fácil y sencillo, además de no consumir demasiada corriente por estar contruido con tecnología CMOS.

6. Para el manejador de líneas (line driver) y para el receptor de líneas (line receiver), se habían considerado los circuitos MC1488 y MC1489, sin embargo, éstos elevaban considerablemente la demanda de energía, exigiendo que la

fuente de alimentación fuera más grande. Con los circuitos DS14C88 y DS14C89, se logró eliminar prácticamente de la fuente de alimentación la etapa que estaba destinada a entregar +9 y -9 Volts, pues estos circuitos por su baja demanda de energía, pueden ser alimentados mediante baterías.

7. No se pudo implementar un display de cristal líquido a la microterminal, el cual era uno de los objetivos principales por su bajo consumo de energía, ya que, su disponibilidad en el comercio de la electrónica nacional, es casi nula, y su adquisición se debe manejar sobre pedido a ciertos proveedores o casas electrónicas, con un alto costo. Por lo anterior, tuvimos que considerar el display de leds, mismo que se encontró fácilmente, siendo su existencia en el mercado mexicano abundante. Es cierto que con su uso, el consumo de energía se incrementa; sin embargo al multiplexar el encendido de los leds, la demanda de corriente baja considerablemente, por lo que la fuente de alimentación lo soporta fácilmente.

8. Se ofrece 2K x 8 de memoria EPROM, misma que es suficiente para alojar el sistema operativo, así como las rutinas de comunicación, del control del teclado, de los displays y de autodiagnóstico. Si tomamos en cuenta que el sistema operativo del kit SDK-85, consume 1K x 8 de memoria EPROM, y considerando que este es el programa más grande, los demás programas podrán ocupar el resto de la memoria para este efecto.

Para el almacenamiento temporal de información, se cuenta con 256Kb de memoria de acceso aleatorio, cantidad suficiente, pues la microterminal necesita 4 buffers básicamente (displays, teclado, comunicaciones y para operaciones y funciones del sistema operativo).

De estos buffers, los que más área de memoria necesitan, es el de displays, pues en él se guarda el mensaje a desplegar. El buffer de comunicación aloja los datos a transmitir y que se reciban del computador. Para la etapa transmisora, sólo se ocupan poco bytes, pues cada vez que se deseé transmitir algo, se hara de forma inmediata, y el dato residira en memoria hasta que lo compare con el "ECO". Para la etapa receptora, la necesidad de espacio en memoria es prácticamente nula, pues los mensajes que provengan del computador se guardan casi automáticamente en el buffer de los displays.

El buffer del teclado tampoco demanda mucha memoria, pues sólo guradrá la información que provenga del teclado. Ahora bien, si por cad tecla corresponden 3 bytes y si se ofrece un buffer para que acepte un máximo de 20 teclas para ser convertidas, significa que necesitamos reservar 60 bytes. Una vez que se ha llevado la conversion de código de teclado a código ASCII, el sistema guarda el resultado en el buffer de datos de transmisión, por lo que no se necesita más memoria.

El área de memoria restante, se le puede dedicar al sistema operativo, para que en él, guarde los resultados temporales de sus operaciones, com es entre otras: el área de Stack, etc.

APENDICE A

8085A CHARACTERISTICS

D.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$; $V_{SS} = 0\text{V}$; unless otherwise specified)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	+0.8	V	
V_{IH}	Input High Voltage	2.0	$V_{CC} + 0.5$	V	
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2\text{mA}$
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = 400\mu\text{A}$
I_{CC}	Power Supply Current		170	mA	
I_{IL}	Input Leakage		± 10	μA	$V_{in} = V_{CC}$
I_{LO}	Output Leakage		± 10	μA	$0.45\text{V} \leq V_{out} \leq V_{CC}$
V_{ILR}	Input Low Level RESET	-0.5	+0.8	V	
V_{IHR}	Input High Level RESET	2.4	$V_{CC} + 0.5$	V	
V_{HY}	Hysteresis RESET	0.25		V	

TIMING CHARACTERISTICS

Bus Timing Specification as a T_{CYC} Dependent

t_{HL}	$(1/2)T - 50$	MIN
t_{LA}	$(1/2)T - 60$	MIN
t_{LL}	$(1/2)T - 20$	MIN
t_{LCK}	$(1/2)T - 60$	MIN
t_{LC}	$(1/2)T - 30$	MIN
t_{AD}	$(5/2 + N)T - 225$	MAX
t_{RD}	$(3/2 + N)T - 180$	MAX
t_{RAE}	$(1/2)T - 10$	MIN
t_{CA}	$(1/2)T - 40$	MIN
t_{DW}	$(3/2 + N)T - 60$	MIN
t_{WD}	$(1/2)T - 60$	MIN
t_{CC}	$(3/2 + N)T - 80$	MIN
t_{CL}	$(1/2)T - 110$	MIN
t_{ARY}	$(3/2)T - 260$	MAX
t_{HACK}	$(1/2)T - 50$	MIN
t_{HABF}	$(1/2)T + 50$	MAX
t_{HABE}	$(1/2)T + 50$	MAX
t_{AC}	$(2/2)T - 50$	MIN
t_1	$(1/2)T - 80$	MIN
t_2	$(1/2)T - 40$	MIN
t_{RV}	$(3/2)T - 80$	MIN

NOTE: N is equal to the total WAIT states

$$T = t_{CYC}$$

ABSOLUTE MAXIMUM RATINGS*
8085A

Ambient Temperature Under Bias	0°C to 70°C
Storage Temperature	-65°C to 150°C
Voltage on Any Pin With Respect to Ground	-0.5 to +7 V
Power Dissipation	1.5 Watt

**COMMENT: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

A P E N D I C E B

8755A CHARACTERISTICS

D.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	+0.8	V	
V_{IH}	Input High Voltage	2.0	$V_{CC} + 0.5$	V	
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2\text{mA}$
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = 400\mu\text{A}$
I_{CC}	Power Supply Current		180	mA	
I_{IL}	Input Leakage		10	μA	$V_{in} = V_{CC}$ to 0 V
I_{LO}	Output Leakage		± 10	μA	$0.45\text{V} \leq V_{out} \leq V_{CC}$

A.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$)

SYMBOL	PARAMETER	MIN	MAX	UNITS	TEST CONDITIONS
t_{CYC}	Clock Cycle Time	320		ns	$C_{LOAD} = 150\text{pF}$
t_1	CLK Pulse Width	80		ns	
t_2	CLK Pulse Width	120		ns	
t_r, t_f	CLK Rise and Fall Time		30	ns	
t_{AL}	Address to Latch Set Up Time	50		ns	
t_{AH}	Address Hold Time after Latch	80		ns	150 μF Load
t_{LC}	Latch to READ/WRITE Control	100		ns	
t_{RD}	Valid Data Out Delay from READ Control		170	ns	
t_{AD}	Address Stable to Data Out Valid		450	ns	
t_{LE}	Latch Enable Width	100		ns	
t_{RDF}	Data Bus Float after READ	0	100	ns	
t_{CL}	READ/WRITE Control to Latch Enable	20		ns	
t_{CC}	READ/WRITE Control Width	250		ns	
t_{OW}	Data in to WRITE Set Up Time	150		ns	
t_{WD}	Data in Hold Time After WRITE	30		ns	
t_{WP}	WRITE to Port Output		400	ns	
t_{PI}	Port Input Set Up Time	50		ns	
t_{PH}	Port Input Hold Time	50		ns	
t_{RHM}	READY HOLD TIME	0	160	ns	
t_{ARV}	ADDRESS (CE) to READY		160	ns	
t_{RV}	Recovery Time between Controls	300		ns	
t_{RDE}	Data Out Delay from READ Control	10		ns	
t_{LD}	ALE to Data Out Valid		350	ns	Preliminary

ABSOLUTE MAXIMUM RATINGS*

8755A

Ambient Temperature Under Bias	-10°C to +70°C
Storage Temperature	-65°C to 150°C
Voltage on Any Pin With Respect to Ground	-0.5 to +7 V*
Power Dissipation	1.5 Watt

**COMMENT: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. SPECIFICATION PROGRAMMING

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$; $V_{CC} = 0\text{V}$)

Symbol	Parameter	Min	Typ	Max	Unit
V_{pp}	Programming Voltage (during Write to EPROM)	24	25	26	V
I_{pp}	Prog Supply Current		15	30	mA

A.C. SPECIFICATION FOR PROGRAMMING

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$; $V_{CC} = 0\text{V}$)

Symbol	Parameter	Min	Typ	Max	Unit
t_{ps}	Data Setup Time	10			ns
t_{pd}	Data Hold Time	0			ns
t_s	Prog. Pulse Setup Time	2			μs
t_h	Prog Pulse Hold Time	2			μs
t_{pr}	Prog Pulse Rise Time	0.01	2		μs
t_{pf}	Prog Pulse Fall Time	0.01	2		μs
t_{pw}	Prog Pulse Width	45	50		msec

8156 CHARACTERISTICS

D.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	$V_{CC} + 0.5$	V	
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2\text{mA}$
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -400\mu\text{A}$
I_{CC}	V_{CC} Supply Current		180	mA	
	Input Leakage		± 10	μA	$V_{IN} = V_{CC}$ to 0 V
I_{O}	Output Leakage Curr.		± 10	μA	$0.45\text{V} < V_{OUT} <= V_{CC}$
I_{LE} (CE)	Chip Enable Leakage		-100	μA	$V_{IN} = V_{CC}$ to 0 V

ABSOLUTE MAXIMUM RATINGS*

8156

Ambient Temperature Under Bias	0°C to $+70^\circ\text{C}$
Storage Temperature	-65°C to 150°C
Voltage on Any Pin With Respect to Ground	-0.5 to $+7\text{V}^*$
Power Dissipation	1.5 Watt

*COMMENT: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

8156 A.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$)

SYMBOL	PARAMETER	MIN	MAX	UNITS
t_{AL}	Address to Latch Set Up Time	50		ns
t_{LA}	Address Hold Time after Latch	80		ns
t_{LC}	Latch to READ/WRITE Control	100		ns
t_{RD}	Valid Data Out Delay from READ Control		170	ns
t_{LE}	Latch Enable Width	100		ns
t_{DF}	Data Bus Float After READ	0	100	ns
t_{CL}	READ/WRITE Control to Latch Enable	20		ns
t_{CC}	READ/WRITE Control Width	250		ns
t_{OW}	Data in to WRITE Set Up Time	150		ns
t_{WD}	Data to Hold Time After WRITE	0		ns
t_{RV}	Recovery Time Between Controls	300		ns
t_{WP}	Write to Port Output		400	ns
t_{PI}	Port Input Setup Time *	70		ns
t_{SBF}	Strobe to Buffer Full		400	ns
t_{SS}	Strobe Width	200		ns
t_{RBE}	Read to Buffer Empty		400	ns
t_{SI}	Strobe to INTR On		400	ns
t_{ROI}	READ to INTR Off		400	ns
t_{PS}	Port Setup Time to Strobe Strobe	50		ns
t_{PHS}	Port Hold Time After Strobe	120		ns
t_{RBE}	Strobe to Buffer Empty		400	ns
t_{WBF}	WRITE to Buffer Full		400	ns
t_{WI}	WRITE to INTR Off		400	ns
t_{TL}	TIMER-IN to TIMER-OUT Low		400	ns
t_{TH}	TIMER-IN to TIMER-OUT High		400	ns
t_{RDE}	Data Bus Enable from READ Control	10		ns
t_L	TIMER-IN Low Time	80		ns
t_H	TIMER-IN High Time	120		ns
t_{AD}	Address Stable to Data Out Valid		400	ns
t_{HP}	Port Input Hold Time	50		ns

A P E N D I C E C

8085A MNEMONICS

JUMP	CALL	RETURN	RESTART	ROTATE**	MOVE(conv)	ACCUMULATOR	
C3 JMP	C0 CALL	C4 RET	C7 RST 0	07 RLC	58 MOV EB	80 ADD B	A8 XRA B
C2 JNZ	D4 CNZ	C0 RNZ	CF RST 1	0F RRC	59 MOV EC	81 ADD C	A9 XRA C
CA JZ	CC CZ	C8 RZ	D7 RST 2	17 RAL	5A MOV ED	82 ADD D	AA XRA D
D2 JNC	D4 CNC	DD RNC	DF RST 3	1F RAR	5B MOV EE	83 ADD E	AB XRA E
DA JC	DC CC	D8 FC	EF RST 4		5C MOV EH	84 ADD H	AC XRA H
E2 JPO	E4 CPO	E0 RPO	EF RST 5		5D MOV EL	85 ADD L	AD XRA L
EA JPE	EC CPE	E8 RPE	F7 RST 6		5E MOV EH	86 ADD M	AE XRA M
F2 JP	F4 CP	F0 RP	FF RST 7	CONTROL	5F MOV EA	87 ADD A	AF XRA A
FA JM	FC CM	F8 RM					
E9 PCHL							
				00 NOP	80 MOV HB	88 ADC B	80 ORA B
				78 HLT	81 MOV HC	89 ADC C	81 ORA C
				73 DI	82 MOV HD	8A ADC D	82 ORA D
				FB EI	83 MOV HE	8B ADC E	83 ORA E
					84 MOV HH	8C ADC H	84 ORA H
					85 MOV HL	8D ADC L	85 ORA L
					86 MOV HM	8E ADC M	86 ORA M
					87 MOV HA	8F ADC A	87 ORA A
				40 MOV B,B	88 MOV LB	90 SUB B	88 CMP B
				41 MOV C,C	89 MOV LC	91 SUB C	89 CMP C
				42 MOV D,D	8A MOV LD	92 SUB D	8A CMP D
				43 MOV E,E	8B MOV LE	93 SUB E	8B CMP E
				44 MOV H,H	8C MOV LH	94 SUB H	8C CMP H
				45 MOV L,L	8D MOV LL	95 SUB L	8D CMP L
				46 MOV M,M	8E MOV LM	96 SUB M	8E CMP M
				47 MOV A,A	8F MOV LA	97 SUB A	8F CMP A
				48 MOV B,B	70 MOV MB	98 SUB B	PSEUDO INSTRUCTION
				49 MOV C,C	71 MOV MC	99 SUB C	STANDARD SETS
				4A MOV D,D	72 MOV MD	9A SUB D	
				4B MOV E,E	73 MOV ME	9B SUB E	
				4C MOV C,H	74 MOV MH	9C SUB H	OR0 Atr
				4D MOV C,L	75 MOV ML	9D SUB L	END
				4E MOV C,M	76 MOV AL	9E SUB M	END D16
				4F MOV C,A	77 MOV MA	9F SUB A	SET D16
							D5 D16
				50 MOV D,B	78 MOV AB	AG ANA B	DB DB ()
				51 MOV D,C	79 MOV AC	A1 ANA C	DF D16 ()
				52 MOV D,D	7A MOV AD	A2 ANA D	
				53 MOV D,E	7B MOV AE	A3 ANA E	
				54 MOV D,H	7C MOV AH	A4 ANA H	
				55 MOV D,L	7D MOV AL	A5 ANA L	
				56 MOV D,M	7E MOV AM	A6 ANA M	MACRO ()
				57 MOV D,A	7F MOV AA	A7 ANA A	ENOM
							FLAG BYTE
							STACK FORMAT
							7 6 5 4 3 2 1 0
							S Z 0 A 0 P I C

* = constant, or logical/mathematic expression that evaluates to an 8 bit data quantity

** = only CARRY affected

Ad = 16 bit address

A P E N D I C E D

**DISPLAY MATRICIAL 5082-7101 DE
HEWLETT PACKARD**

D.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C ; $V_{CC} = 5\text{V} \pm 5\%$)

Symbol	Parameter	Min.	Max.	Units	Test Conditions
I_L	Input Led Current		100	mA	
I_{AL}	Input Led Average Curr		10	mA	
P_{CO}	Power Character Disipation		700	mW	

**CODIGO PARA EL ENCENDIDO
DE LEDS EN LOS DISPLAY'S
MATRICIALES**

	Anodo	EPROM (HEX)	Cátodo	Puerto A	Puerto B	Puerto C
	A	(00)H	1a	(FE)H	(00)H	(00)H
	B	(01)H	1b	(FD)H	(00)H	(00)H
	C	(02)H	1c	(FB)H	(00)H	(00)H
DISPLAY I	D	(03)H	1d	(F7)H	(00)H	(00)H
	E	(04)H	1e	(EF)H	(00)H	(00)H
	F	(05)H				
	G	(06)H	2a	(DF)H	(00)H	(00)H
			2b	(BF)H	(00)H	(00)H
	A	(09)H	2c	(7F)H	(00)H	(00)H
	B	(0A)H	2d	(00)H	(FE)H	(00)H
	C	(0B)H	2e	(00)H	(FD)H	(00)H
DISPLAY II	D	(0C)H				
	E	(0D)H	3a	(00)H	(FB)H	(00)H
	F	(0E)H	3b	(00)H	(F7)H	(00)H
	G	(0F)H	3c	(00)H	(EF)H	(00)H
			3d	(00)H	(DF)H	(00)H
	A	(11)H	3e	(00)H	(BF)H	(00)H
	B	(12)H				
	C	(13)H	4a	(00)H	(7F)H	(00)H
DISPLAY III	D	(14)H	4b	(00)H	(00)H	(0E)H
	E	(15)H	4c	(00)H	(00)H	(0D)H
	F	(16)H	4d	(00)H	(00)H	(0B)H
	G	(17)H	4e	(00)H	(00)H	(07)H
		(07)H				
		(08)H				
PARA		(10)H				
DES-		(18)H				
HABILITAR		(19)H				
LOS		(1A)H				
DISPLAYS		.				
		.				
		.				
		(1F)H				

A P E N D I C E E

CARACTERISTICAS DEL TECLADO DE MEMBRANA DE RAPID CIRCUIT.

MODELO 1610. CONEXIONES EN MATRICIAL.

CARACTERISITICAS ELECTRICAS:

- | | |
|----------------------|-----------------|
| - Resistencia máxima | 50 ohms |
| - Voltaje máximo | 30 volts |
| - Corriente máxima | 100 miliamperes |

CARACTERISTICAS MECANICAS:

- | | |
|--|------|
| - Fuerza para operar una tecla | 50gr |
| - La parte trasera del teclado es aislada. | |

CARACTERISTICAS AMBIENTALES:

- | | |
|----------------------------|---------------|
| - Temperatura de operación | -40°C a +70°C |
|----------------------------|---------------|

CARACTERISTICAS QUIMICAS:

-) Se recomienda la utilización de pegamentos a base de SILICON
-) No se debe manejar pegamentos de contactos o similares (ej. Resistol).
-) No utilizar solventes como tricloretileno, thinner, Xileno.
-) Se puede usar agua o alcohol.

SOLDAR SOLAMENTE AL CONECTOR.
EL TECLADO ES AUTOADHERIBLE.

FABRICADO FOR RAPID CIRCUIT.

ANICETO ORTEGA No. 714, COL. DEL VALLE, MEXICO, D.F.

RENGLON NUM	EPCOM 8155A						10	9	8	7	6	5	4	3	2	1	----> # DE COLUMNA
	B5	B4	B3	B2	B1	B0											
							0	0	0	0	0	0	0	0	0	1	A0
							0	0	0	0	0	0	0	0	1	0	A1
							0	0	0	0	0	0	0	1	0	0	A2
							0	0	0	0	0	0	1	0	0	0	A3
							0	0	0	0	0	1	0	0	0	0	A4
							0	0	0	0	1	0	0	0	0	0	A5
							0	0	0	1	0	0	0	0	0	0	A6
							0	0	1	0	0	0	0	0	0	0	A7
							0	1	0	0	0	0	0	0	0	0	B0
							1	0	0	0	0	0	0	0	0	0	B1
							00H	00H	80H	40H	20H	10H	08H	04H	02H	01H	----> PUERTO A
							02H	01H	00H	00H	00H	00H	00H	00H	00H	00H	----> PUERTO B
1	0	0	0	0	0	1	01H	1\!	2*	3\#	4\%	5\&	6\&	7\/	8\{	9\	0\
2	0	0	0	0	1	0	02H	Q	W	R	E	T	Y	U	I	O	P
3	0	0	0	1	0	0	04H	A	S	D	F	G	H	J	K	L	;
														if			
4	0	0	1	0	0	0	08H	Z	X	C	V	B	N	M	{	~	}
														ret	enc	rs	gs
5	0	1	0	0	0	0	10H	DEL- vs	+!;	-!;	*!;	/!?	,!<	!>			RTM
6	0	1	0	0	0	0	20H	CNTL									BARRA ESPACIADORA
																	SHEPT

} 8155 RAM

A P E N D I C E F

MM74138

3-to-8 Line Decoder

Electrical Characteristics $T_A=25^\circ\text{C}$, $V_{CC}=5.0\text{V}$, $t_r=t_f=6\text{ns}$, $C_L=15\text{pF}$ (unless otherwise specified)

Parameter	Conditions	Typ	Guaranteed Limit	Units
Maximum Propagation Delay, A, B or C to Output		20	35	ns
Maximum Propagation Delay, #1 to Y Output		14	25	ns
Maximum Propagation Delay, E2 or E3 to Y Output		17	30	ns

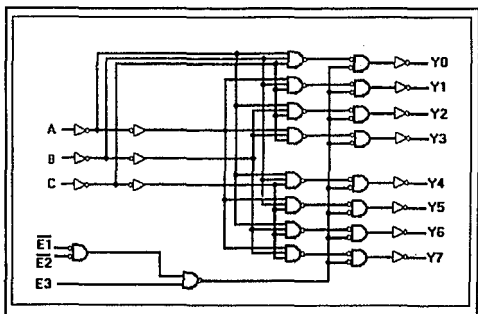
DC Electrical Characteristics $V_{CC}=5\text{V} \pm 10\%$ (unless otherwise specified)

Symbol	Parameter	Conditions	Typ	$T_A=25^\circ\text{C}$	Units
V_{IH}	Minimum High Level Input Voltage			2.0	V
V_{IL}	Maximum Low Level Input Voltage			0.8	V
V_{OH}	Minimum High Level Output Voltage	$V_{IH}=V_{IH}$ or V_{IL} $I_{OUT1}=20\mu\text{A}$	V_{CC}	$V_{CC}-0.1$	V
		$I_{OUT1}=4.0\text{mA}$, $V_{CC}=4.5\text{V}$	4.2	3.95	V
		$I_{OUT1}=4.0\text{mA}$, $V_{CC}=5.5\text{V}$	5.2	4.95	V
V_{OL}	Maximum Low Level Output Voltage	$V_{IH}=V_{IH}$ or V_{IL} $I_{OUT1}=20\mu\text{A}$	0	0.1	V
		$I_{OUT1}=4.0\text{mA}$, $V_{CC}=4.5\text{V}$	0.2	0.25	V
		$I_{OUT1}=4.0\text{mA}$, $V_{CC}=5.5\text{V}$	0.2	0.25	V
I_{IH}	Maximum Input Current	$V_{IH}=V_{CC}$ or GND V_{IH} or V_{IL}		± 0.1	μA
I_{CC}	Maximum Quiescent Supply Current	$V_{IH}=V_{CC}$ or GND $I_{OUT1}=0\mu\text{A}$		8.0	μA

Absolute Maximum Ratings

Parameter	Rating
Supply Voltage (V_{CC})	-0.5 to +7.0V
DC Input Voltage (V_{IH})	-1.5 to $V_{CC} + 1.5\text{V}$
DC Output Voltage (V_{OUT1})	-0.5 to $V_{CC} + 0.5\text{V}$
Clamp Diode Current (I_{IK} , I_{OK})	$\pm 20\text{mA}$
DC Output Current, per pin (I_{OUT1})	$\pm 25\text{mA}$
DC V_{CC} or GND Current per pin (I_{CC})	$\pm 50\text{mA}$
Storage Temperature Range (T_{STG})	-65°C to $+150^\circ\text{C}$
Power Dissipation (P_D)	500mW
Lead Temperature (T_L) (Soldering 10 seconds)	260°C

Logic Diagram



Truth Table

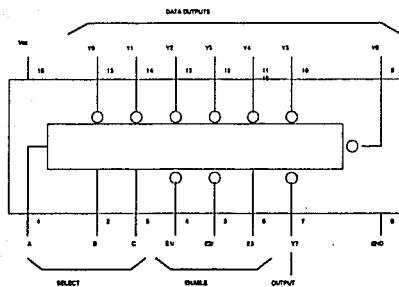
Inputs				Outputs							
Enable		Select		Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
E1	E2	C	B	A							
X	H	X	X	X	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H
H	L	H	L	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	L

*E3 = E1 • E2

H = High level L = low level X = don't care

Connection Diagram

Dual-In-Line Package



A P E N D I C E G

7406

Hex Inverter

Buffer Drivers with open-collector high voltage outputs

Supply Current

Parameter	Typ	Max
I_{CC} (mA) Total with outputs high	30	48
I_{CC} (mA) Total with outputs low	32	51
I_{CC} (mA) Average per gate	5.17	

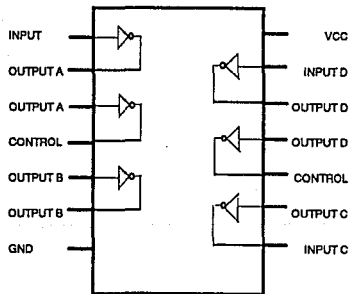
DC Electrical Characteristics

Symbol	Parameter	Min	Nom	Max	Units
V_{CC}	Voltage Supply	4.75	5	5.25	V
V_{OH}	High Level Output Voltage			30	V
I_{OH}	Low Level Output Current			40	mA
T_A	Operating Free air Temperature	0		70	°C

7406

Hex Inverter

DUAL-IN-LINE PACKAGE



LM123A CHARACTERISTICS

ELECTRICAL CHARACTERISTICS ⁽¹⁾

Parameter	Conditions	Min	Typ	Max	Units
Output Voltage	$T_J = 25^\circ\text{C}$ $V_{in} = 7.5\text{V}, I_{out} = 0$	4.7	5	5.3	V
Output Voltage	$7.5 < V_{in} < 15\text{V}$ $0 < I_{out} < 3\text{A}, P < 300\text{W}$	4.8		5.3	V
Line Regulation ⁽²⁾	$T_J = 25^\circ\text{C}$ $7.5 < V_{in} < 15\text{V}$		5	2.5	mV
Line Regulation ⁽²⁾	$T_J = 25^\circ\text{C}, V_{in} = 7.5\text{V}$ $0 < I_{out} < 3\text{A}$		25	100	mV
Quiescent Current	$7.5 < V_{in} < 15\text{V}$ $0 < I_{out} < 3\text{A}$		12	20	mA
Output Noise Voltage	$T_J = 25^\circ\text{C}$ $10\text{ Hz} < f < 100\text{ kHz}$		40		μVrms
Short Circuit Current Limit	$T_J = 25^\circ\text{C}$ $V_{in} = 15\text{V}$ $V_{in} = 7.5\text{V}$		3 4	45 5	A A
Long Term Stability				35	mV
Thermal Resistance Junction ⁽³⁾			2		$^\circ\text{C/W}$

NOTES:

(1) Unless otherwise noted, specification apply for $-55^\circ\text{C} < T_J < +150^\circ\text{C}$. Although power dissipation is internally limited, specifications apply only for $P < 30\text{W}$.

(2) Without a heatsink, the thermal resistance of the TO-3 package is about 35°C/W , with a heatsink, the effective thermal resistance can only approach the specified values of 2°C/W , depending on the efficiency of the heatsink

(3) Load and line regulation are specified at constant junction temperature. Pulse testing is required with a pulse width $< 1\text{ms}$ and a duty cycle $< 5\%$

ABSOLUTE MAXIMUM RATINGS**LM123A**

Input Voltage	20V
Power Dissipation Internally Limited	-
Operating Junction Temperature Range	-55 to +150 °C*
Storage Temperature Range	-65°C to +150°C
Lead Temperature (Soldering 10 sec.)	300°C

A P E N D I C E H

DS14C88

Quad CMOS Line Driver

DC Electrical Characteristics

$T_A = 0^\circ\text{C}$ to 70°C , $V^+ = +4.5\text{V}$ to 12V , $\text{GND} = 0\text{V}$, $V^- = -4.5\text{V}$ to 12V , unless otherwise specified

Symbol	Parameter	Conditions	Min	Max	Units
I_{IH} , I_{IH}	Maximum Input Current	$V_{IH} = \text{GND or } V^+$	-10	10	μA
V_{IH}	High Level Input Voltage		2.0	V_{DD}	V
V_{IL}	Low Level Input Voltage	$V^+ \Rightarrow 7\text{V}, V^- \leq -7\text{V}$	GND	0.8	V
		$V^+ \Rightarrow 7\text{V}, V^- \leq -7\text{V}$	GND	0.6	V
V_{OH}	High Level Output Voltage	$V_{IH} = V_{IL}$			
		$R_L = 3\text{ kohm or } 7\text{ kohm}$			
		$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$	3.0		V
		$V^+ = +9\text{V}, V^- = -9\text{V}$	6.5		V
		$V^+ = +12\text{V}, V^- = -12\text{V}$	9.0		V
V_{OL}	Low Level Output Voltage	$V_{IH} = V_{OH}$			
		$R_L = 3\text{ kohm or } 7\text{ kohm}$			
		$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$		-3.0	V
		$V^+ = +9\text{V}, V^- = -9\text{V}$		-6.5	V
		$V^+ = +12\text{V}, V^- = -12\text{V}$		-9.0	V
I_{OS}^+	High Level Output Short Circuit Current (Note 3)	$V_{OH} = V_{IL}$ $V_{OUT} = \text{GND}$ $V^+ = +12\text{V}, V^- = -12\text{V}$		+45	mA
I_{OS}^-	Low Level Output Short Circuit Current (Note 3)	$V_{OH} = V_{OH}$ $V_{OUT} = \text{GND}$ $V^+ = +12\text{V}, V^- = 12\text{V}$		-45	mA
R_{OUT}	Output Resistance	$V^+ = V^- = 0\text{V}$ $-2\text{V} \leq V_{OUT} \leq 2\text{V}$	300		ohm
I_{CC}^+	Positive Supply Current	$V_{OH} = V_{IL}, R_L = \text{open}$			
		$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$		10	μA
		$V^+ = +9\text{V}, V^- = -9\text{V}$		30	μA
		$V^+ = +12\text{V}, V^- = -12\text{V}$		60	μA
		$V_{OH} = V_{OH}, R_L = \text{open}$		30	μA
		$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$		190	μA
		$V^+ = +9\text{V}, V^- = -9\text{V}$		425	μA
		$V^+ = +12\text{V}, V^- = -12\text{V}$		425	μA
I_{CC}^-	Negative Supply Current	$V_{OH} = V_{IL}, R_L = \text{open}$			
		$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$		-10	μA
		$V^+ = +9\text{V}, V^- = -9\text{V}$		-10	μA
		$V^+ = +12\text{V}, V^- = -12\text{V}$		-10	μA
		$V_{OH} = V_{OH}, R_L = \text{open}$		-30	μA
		$V^+ = +4.5\text{V}, V^- = -4.5\text{V}$		-30	μA
		$V^+ = +9\text{V}, V^- = -9\text{V}$		-60	μA
		$V^+ = +12\text{V}, V^- = -12\text{V}$		-60	μA

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provide conditions for actual device operation.

Note 2: Power Dissipation N-Package: 1300 mW at 25°C , J-Package: 1000 mW at 25°C

Note 3: I_{OS}^+ and I_{OS}^- values are for one output at a time. If more than one output is shorted simultaneously, the device dissipation may be exceeded.

DS14C88 Absolute Maximum Ratings (Note 1)

Parameter	Rating
Voltage to Any Input Pin	(V _I) +0.3 to GND -0.3V
Voltage to Any Output Pin	-25V to +25V
Storage Temp.	-65°C to +150°C
Power Dissipation	(See Note 2)
Junction Temperature	+150°C
Lead Temp. (soldering 10 sec.)	+260°C

DS14C88 Operating conditions

Parameter	Min	Max
Supply Voltage V ₊ (GND = 0V)	+4.5V	+12.6V
Supply Voltage V ₋ (GND = 0V)	-4.5V	-12.6V
Temperature Range	0°C	+70°C

DS14C89A

Quad CMOS Line Receiver

DC Electrical Characteristics

$T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, $+4.5 < V_{CC} < 5.5\text{ V}$, $\text{GND} = 0\text{V}$, unless otherwise specified

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_{TH}	Input High Threshold Voltage		1.3		2.5	V
V_{TL}	Input Low Threshold Voltage		0.5		1.7	V
V_H	Typical Input Hysteresis			1.0		V
I_{IN}	Input Current	$V_{IN} = +25\text{V}$ $V_{IN} = -25\text{V}$ $V_{IN} = +3\text{V}$ $V_{IN} = -3\text{V}$	3.6 -3.6 +0.43 -0.43		8.3 -8.3 +1.0 -1.0	mA mA mA mA
V_{OH}	Output High Voltage	$V_{IN} = V_{TL}$ (min) $I_{OUT} = -3.2\text{ mA}$	2.8			V
V_{OL}	Output Low Voltage	$V_{IN} = V_{TH}$ (max) $V_{OUT} = +3.2\text{ mA}$			0.4	V
I_{CC}	Supply Current	$R_L = \text{open}$ $V_{IN} = V_{TH}$ (max) or V_{TL} (min)			+900	μA

AC Electrical Characteristics

$T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$, $+4.5 < V_{CC} < 5.5\text{ V}$, $\text{GND} = 0\text{V}$, $\text{CL} = 50\text{pF}$ unless otherwise specified (Note 2)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
t_{PLH}	Propagation Delay to a Logic 1	Input pulse width $\geq 10\mu\text{s}$			6.5	μs
t_{PLL}	Propagation Delay to a Logic 0	Input pulse width $\geq 10\mu\text{s}$			6.5	μs
t_{SK}	Typical Propagation Delay Skew			400		ns
t_r	Output Rise Time				300	ns
t_f	Output Fall Time				300	ns
t_{pw}	Pulse Width Assumed to be Noise				1.0	μs

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The table of "Electrical Characteristics" provide conditions for actual device operation.

Note 2: AC Input waveform for test purposes $t_r = t_f = 200\text{ns}$, $V_{IH} = +3\text{V}$, $V_{IL} = -3\text{V}$, $f = 20\text{ kHz}$.

DS14C89A Absolute Maximum Ratings (Note 1)

Parameter	Rating
Voltage to Any Input Pin	-30V to +30V
Voltage to Any Output Pin	(V_{CC}) +0.3V to GND -0.3V
Storage Temp.	-65°C to +150°C
Power Dissipation	500mW at +70°C
Junction Temperature	+150°C
Lead Temp. (soldering 10 sec.)	+260°C

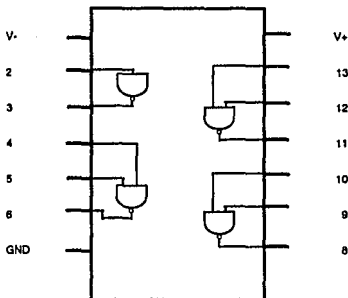
DS14C89A Operating conditions

Parameter	Min	Max
Supply Voltage V_{CC} (GND=0V)	+4.5V	+5.5V
Temperature Range	0°C	+70°C

DS14C88

Line Driver

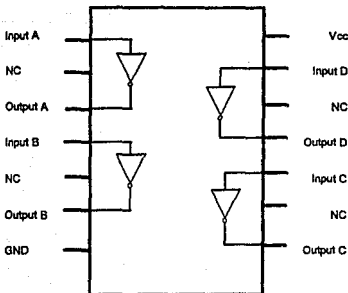
DUAL-IN-LINE PACKAGE



DS14C89A

Line Receiver

DUAL-IN-LINE PACKAGE



Features

Low power CMOS design

$\pm 0.2V$ sensitivity over the entire common mode range

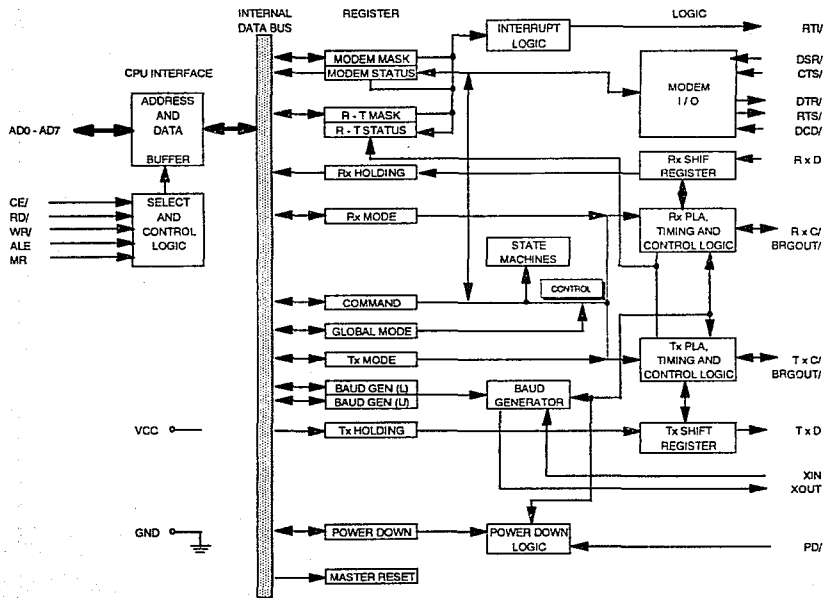
Typical propagation delays: 15ns

Typical input hysteresis: 50mV

Inputs won't load line when $V_{CC}=0V$

Meets the requirements of EIA standard RS-422

TRI-STATE outputs for connection to system buses



NSC858 Functional Block Diagram

NCS858 UART

DC Electrical Characteristics $V_{CC}=5V \pm 10\%$, $GND=0V$, unless otherwise specified

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_{IH}	Logical 1 Input Voltage		$0.8V_{CC}$		V_{CC}	V
V_{IL}	Logical 0 Input Voltage		0		$0.2V_{CC}$	V
V_{HY}	Hysteresis at RESET IN Input	$V_{CC} = 5V$	0.25	0.5		V
V_{OH1}	Logical 1 Output Voltage	$I_{OUT} = -1.0mA$	2.4			V
V_{OH2}	Logical 1 Output Voltage	$I_{OUT} = -10\mu A$	$V_{CC} - 0.5$			V
V_{OL1}	Logical 0 Output Voltage	$I_{OL} = 2mA$ except X_{OUT}	0		0.4	V
V_{OL2}	Logical 0 Output Voltage	$I_{OUT} = 10\mu A$	0		0.1	V
I_{IL}	Input Leakage Current	$0 < V_{IN} < V_{CC}$	-10.0		10.0	μA
I_{OL}	Output Leakage Current	$0 < V_{IN} < V_{CC}$	-10.0		10.0	μA
I_{CC}	Active Supply Current	$T_A = 25^\circ C$		2	10	μA
I_{HPD}	Current Hardware Power Down	Pin PD = 0, No resistive Output Loads. $T_A = 25^\circ C$ $V_{IN} = 0V$ or $V_{IN} = V_{CC}$		100		μA
I_{SPD}	Current Software Power Down	Power Down Reg Bit 0=1, No Resistive Outputs Load $V_{IN} = 0V$ or $V_{IN} = V_{CC}$ $T_A = 25^\circ C$		300		μA
C_{IN}	Input Capacitance			6	10	pF
C_{OUT}	Output Capacitance			8	12	pF
V_{CC}	Power Supply Voltage	(Note 1)	2.4	5	6	V

Note 1: Operation at lower power supply voltages will reduce the maximum operating speed. Operation at voltages other than $5V \pm 10\%$ is guaranteed by design, not tested.

Absolute Maximum Ratings (Note 2)

Parameter	Ratings
Storage Temperature	$-65^\circ C$ to $+150^\circ C$
Voltage on Any Pin with Respect to Ground	$-0.3V$ to $V_{CC} + 0.3V$
Maximum V_{CC}	7V
Power Dissipation	1W
Lead Temp. (soldering 10 sec.)	$300^\circ C$

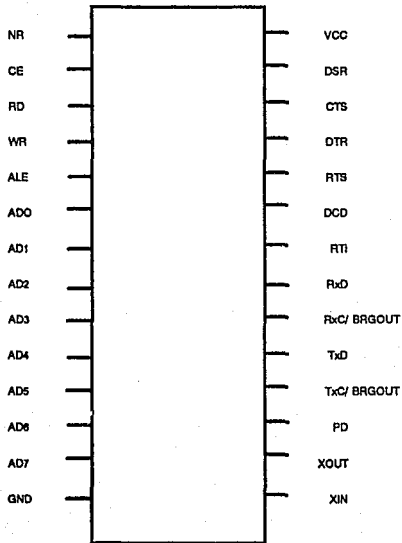
Note 2: Absolute Maximum Ratings indicate limits beyond which permanent damage may occur. Continuous operation at these limits is not intended and should be limited to those conditions specified under DC Electrical Characteristics.

Operating conditions ($V_{CC}=5V \pm 10\%$)

Ambient Temperature	Values
Industrial	$-40^\circ C$ to $+85^\circ C$
Commercial	$0^\circ C$ to $+70^\circ C$

NCS858

UART



A P E N D I C E I

USASCII CODE

								0	0	0	0	1	1	1	1	
								0	0	1	1	0	0	1	1	
								0	1	0	1	0	1	0	1	
b0	b1	b2	b3	b4	b5	b6	b7									
0				0	0	0	0	0	NUL	DLE	SP	@	P	\	~	
0				0	0	0	1	1	SOH	DC1	!	!	A	Q	a	q
0				0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0				0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0				0	1	0	0	4	EOT	DC4	\$	4	D	T	d	t
0				0	1	0	1	5	ENO	NAK	%	5	E	U	e	u
0				0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0				0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
0				1	0	0	0	8	BS	CAN	(8	H	X	h	x
0				1	0	0	1	9	HT	EM)	9	I	Y	i	y
0				1	0	1	0	A	LF	SUB	:	A	J	Z	j	z
0				1	0	1	1	B	VT	ESC	;	B	K	[k	[
0				1	1	0	0	C	FF	FS	<	C	L	\	l	l
0				1	1	0	1	D	CR	GS	=	D	M]	m]
0				1	1	1	0	E	SO	RS	>	E	N	^	n	^
0				1	1	1	1	F	SI	US	/	F	O	_	o	DEL

Main Storage Byte

A P E N D I C E J

INTERFACE RS422A

Al principio de los años setenta, la EIA introdujo el estándar RS422A, para mejorar los defectos de la interface RS232C. Lo que se hizo fue separar en dos partes las características del RS232C: Eléctricas-Mecánicas y las funcionales. El RS422A cubre y mejora las Eléctricas-Mecánicas otros circuitos como el RS449 las funcionales, es decir, las de control.

Por la interface RS422A se pueden transmitir datos a altas velocidades (hasta 10 M bound) (tabla J-A). Usa dos hilos por cada señal. A esta técnica se le llama "transmisión balanceada", duplica el número de alambres en el cable, pero permite lograr muy alta velocidad de datos y minimiza el problema de variaciones de potencial en la referencia. Como la tierra no es usada como voltaje de referencia, los requerimientos de tierra en la interface RS422A son menos críticos que los de la interface RS232C; y el uso de la señal de tierra en la RS422A es opcional. (figura J-1)

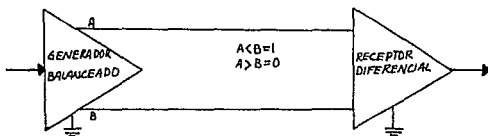


Figura J-1

Otra diferencia que hace que la interface RS422A sea mejor que RS232C, es la región de transición entre los estados de uno y cero lógicos. Con la eliminación de la señal de tierra, la región de transición puede ser más angosta. En la RS422A la diferencia de potencial entre los dos alambres determina si se mandó un uno o cero lógico. La diferencia es de sólo 0.4V, considerando los 24V (+12V y -12V) en la RS232C. Si la diferencia entre los alambres es positivo y mayor de 0.2V, el receptor lo interpreta como un uno lógico, ahora bien, si la diferencia es negativa o menor de -0.2V, la etapa receptora lo lee como cero lógico. Estos valores de voltaje permiten que los transmisores y receptores puedan ser alimentados con los +/- 5V, los cuales son muy comunes en los computadores. Se utiliza para enlaces directos, dejando la RS232C para enlaces vía modem. Como son más pequeños, se ahorra espacio en la circuitería de la microterminal, pues se elimina el UART.

TABLA J-1

<u>PARAMETRO</u>	<u>RS232C</u>	<u>RS422</u>
longitud máxima	50 ft	1200 m (4000 ft)
velocidad máxima	20 K baud	10 M baud
1 lógico	-3V	A < B
0 lógico	+3V	A > B
voltaje de salida para circuito abierto	3V-25V (5V-15V con 3-7 K ohm de carga)	<6V
nivel mínimo de entrada en el receptor	+/- 3V	200 mV diferencial

BIBLIOGRAFIA

8085A E INTERFACES

Octavio F. García Narcía.
Publicado por la ESIME
1ra. Edición

MCS-85 USER'S MANUAL

Publicado por INTEL CORPORATION
Copyright 1977

MICROCOMMUNICATIONS ELEMENTS

Publicado por NATIONAL SEMICONDUCTOR CORP.
Copyright 1987

NATIONAL SEMICONDUCTOR DATABOOK

Publicado por NATIONAL SEMICONDUCTOR CORP.
Copyright 1986

THE TTL DATA BOOK

Publicado por TEXAS INSTRUMENTS
Copyright 1974

INTERFACE DATA BOOK

Publicado por NATIONAL SEMICONDUCTOR CORP.
Copyright 1986

PRINCIPIOS DE ELECTRONICA

Mc Graw Hill
Albert Paul Malvino, Ph. D.
2da. Edición

ESTANDARES DE TELECOMUNICACIONES

Publicado por WANG LABORATORIES INC.
Copyright 1986

UNDERSTANDING DATA COMMUNICATIONS

Publicado por RADIO SHACK
Copyright 1986

SYSTEMS DESIGN WITH ADVANCED MICROPROCESSORS

Publicado por HOWARD W. SAMS & CO.
John Freer
3ra. Edición