

T-UEB  
1504  
S 961d  
1992  
Ej. 2

# UNIVERSIDAD DE EL SALVADOR

FACULTAD DE INGENIERIA Y ARQUITECTURA  
ESCUELA DE INGENIERIA ELECTRICA



## “DISEÑO Y CONSTRUCCION DE INTERFASE Y MODULO DE PRUEBA PARA COMPONENTES INTEGRADOS DIGITALES EN LA COMPUTADORA PC-501-AT LEMMON”

TRABAJO DE GRADUACION PRESENTADO POR:

**JOSE HERIBERTO SURIA MONTES  
WERNER DAVID MELENDEZ VALLE**

PARA OPTAR AL TITULO DE  
**INGENIERO ELECTRICISTA**

ABRIL 1992



SAN SALVADOR, EL SALVADOR, CENTRO AMERICA

*Rec. abril 6/92*

*15101407*

*15101407*

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

SECRET

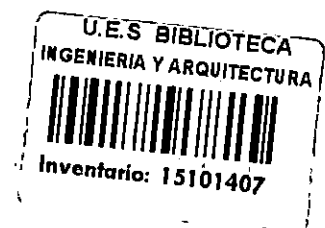
SECRET

SECRET

SECRET

SECRET

UNIVERSIDAD DE EL SALVADOR



RECTOR:

DR. FABIO CASTILLO FIGUEROA

SECRETARIO GENERAL:

LIC. MIGUEL ANGEL AZUCENA

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO:

ING. JUAN JESUS SANCHEZ SALAZAR

SECRETARIO:

ING. JOSE RIGOBERTO MURILLO CAMPOS

ESCUELA DE INGENIERIA ELECTRICA

DIRECTOR:

ING. RICARDO CORTEZ

*Jose Heriberto S.M.*

UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERIA Y ARQUITECTURA  
ESCUELA DE INGENIERIA ELECTRICA

Trabajo de Graduación previo a la opción al grado de:

INGENIERO ELECTRICISTA

Título: "DISEÑO Y CONTRUCCION DE INTERFASE Y MODULO  
DE PRUEBA PARA COMPONENTES INTEGRADOS  
DIGITALES EN LA COMPUTADORA PC-501-AT LEMMON"

Presentado por:

JOSE HERIBERTO SURIA MONTES Y  
WERNER DAVID MELENDEZ VALLE

Trabajo de Graduación aprobado por:

Coordinador y Asesor:

ING. RICARDO ERNESTO CORTEZ.

Asesor:

ING. SALVADOR DE JESUS GERMAN.

Asesor:

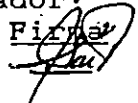
ING. JOSE MAURICIO QUIÑONEZ.

San salvador, Abril de 1992.

ACTA DE CONSTANCIA DE NOTA Y DEFENSA FINAL

En esta fecha, 23 de Marzo de 1992<sup>2</sup>  
en el local de Sala de Lectura de la Escuela de Ingeniería Eléctrica  
a las 10:05 horas, con la presencia de las siguientes autoridades de la  
Escuela de Ingeniería Eléctrica de la Universidad de El Salvador.

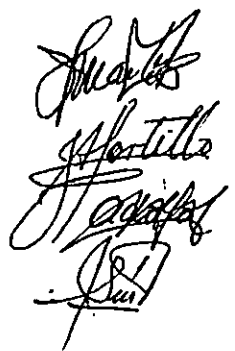
- 1- Ing. Salvador de Jesús German  
Director E.I.E.
- 2- Ing. Osbaldo Adolfo Campos  
Secretario E.I.E.
- 3- Ing. Ricardo Ernesto Cortez  
Director de Investigaciones E.I.E.

*Firma*  




Y con el Honorable Jurado de evaluación integrado por las personas  
siguientes:

- 1- Ing. Jesús Félix Martínez
- 2- Ing. José Aquiles Rodríguez
- 3- Ing. Osbaldo Adolfo Campos
- 4- Ing. Salvador de Jesús German
- 5- \_\_\_\_\_
- 6- \_\_\_\_\_



Se efectuó la defensa final reglamentaria del Trabajo de  
Graduación: " DISEÑO Y CONSTRUCCION DE INTERFASE Y MODULO DE PRUEBA PARA COMPONENTES  
INTEGRADOS DIGITALES EN LA COMPUTADORA PC-501-AT LEMMON"

a cargo del (los) Br(es): José Heriberto Suria Montes y  
Werner David Meléndez Valle

Habiendo obtenido el presente trabajo una nota final, global de 8.4  
( Ocho punto Cuatro - \_\_\_\_\_ )

Trabajo Dedicado a:

Al que me lavo con su sangre de mis pecados, nuestro Señor Jesucristo.

A mis padres con especial agradecimiento, ya que me apoyaron tanto material, como moralmente y merecen todo mi cariño y respeto.

A mi hermana Janeth con especial cariño.

Y a todos mis compañeros y amigos; con los cuales compartimos aflicciones y triunfos, con mucho aprecio.

JOSE HERIBERTO SURIA MONTES

## DEDICATORIA

A DIOS TODOPODEROSO, Por acompañarme en todo momento y permitirme llegar hasta aquí.

A MIS PADRES, Por su cariño, comprensión, apoyo y sacrificio.

A MIS HERMANOS, Que con paciencia y afecto, me alentaron a seguir adelante.

A MI SOBRINOS, Que en la distancia recuerdo y que con su inocencia infantil llenaron de alegría tantos momentos tristes.

A MIS AMIGOS, Compañeros de estudio y trabajo, por ser precisamente eso : Mis Amigos.

Y A TODOS aquellos que sin interés personal, me ayudaron a alcanzar esta meta y muy especialmente a BRISEIDA EURIDICE, por llegar justo en el momento en que más la necesitaba.

A TODOS ELLOS VA DEDICADO ESTE TRIUNFO

WERNER DAVID

## P R E F A C I O

En los últimos años nuestro país, ha tomado un gran auge por el "mundo de las computadoras", lo cual nos lleva a tomar un contexto mas activo en la investigación acerca de los usos que podría tener en el campo práctico de Ingeniería Eléctrica. Con esta clase de trabajo se promueve la utilización de las computadoras AT LEMON compatible con la IBM en aplicaciones de control en tiempo real, esto conlleva poder establecer las bases para aplicaciones de la mismas máquinas en el control de procesos industriales complejos en trabajos futuros, y además contribuye con el desarrollo del área de instrumentación y control mediante la aplicación natural de la potencia de la AT LEMON en los trabajos del área.

Los objetivos que se persiguen con este trabajo, es construir una interfase para el usuario de la computadora AT LEMON que permita el control de Hardware, en una aplicación dada, también persigue proveer al área de control e instrumentación un medio de comprobación del estado de operación de ciertos componentes integrados. Al mismo tiempo de impulsar el desarrollo de programas de control de dispositivos utilizando al AT LEMON.

Los alcances que pretende este trabajo incluyen construir y diseñar una tarjeta de interfase para la comunicación entre la computadora y el mundo exterior, también diseñar y construir un módulo de prueba de la componentes integradas digitales a comprobar y finalmente diseñar el programa de control del módulo de prueba. Esto incluye obligatoriamente instrucciones para el usuario, un menú de componentes que puede comprobarse, rutinas de control para cada componente escogida y la presentación de resultados de la comprobación en forma comprensible y clara.



## RESUMEN DEL TRABAJO

En el presente trabajo, se explica en el Capítulo I el medio por el cual se logra la comunicación de la computadora con el medio exterior, esto es posible por medio de una interfase basada en el circuito integrado 8255 PPI (Interfase Programable de Periféricos), se explican sus modos de operación y las bondades que ofrece.

El Capítulo II trata del módulo de prueba con todas las especificaciones para el diseño e implementación del mismo. Tanto la interfase como el módulo de prueba están hechos en tarjeta impresa con la tecnología de la Escuela de Ingeniería Eléctrica esto da un aporte muy significativo en futuros trabajos en los cuales es necesario usar estos medios. La metodología usada en esta implementación y prueba es eminentemente experimental ya que no se cuenta con bibliografía suficiente ni experiencia real, para solventar los múltiples problemas que surgieron en la fabricación de estas tarjetas.

Finalmente el capítulo 3 trata acerca de la comunicación por medio de un programa de control entre la computadora y el módulo de prueba. Para este caso en particular se ha trabajado en un lenguaje de alto nivel específicamente en Turbo Pascal versión 4.0 utilizando su potencial de instrucciones para presentar en un programa los resultados de una manera clara y vistosa a la altura de un Trabajo de Graduación, esto involucra una investigación ya que no se tiene conocimiento de toda su potencialidad.

## TABLA DE CONTENIDOS

	PAG.
CAPITULO I .....	1
1. Diseño del Circuito de Interfase .....	1
1.1 Interconexión de la Interfase con la Computadora .....	1
1.2 Descripción de los canales I/O de Expansión de las computadoras IBM PC o Compatibles .....	1
1.3 Descripción del Circuito de Interfase .....	4
1.4 El PPI 8255A (Interfase Programable de Periféricos) ....	5
1.4.1 Modos de Operación del 8255A .....	7
1.4.2 Modo 0: Entrada/Salida Elemental.....	7
1.4.3 Modo 1: Entrada/Salida con Protocolo .....	10
1.4.4 Modo 2: Puerto Bidireccional con Protocolo .....	11
1.5 Descripción de Pines del 8255A .....	13
Conclusiones .....	15
Referencias Bibliográficas .....	16
CAPITULO II .....	17
2. Diseño del Módulo de Prueba .....	17
2.1 Base 1. Flujo de Señales. Análisis de Funcionamiento ...	17
2.1.1 Multiplexador de Pines .....	22
2.1.2 Señales de Prueba .....	24
2.1.3 Lectura de Señales .....	26
2.1.4 Señales Especiales .....	26
2.2 Base 2. Análisis y Funcionamiento .....	28
2.3 Base 3. Análisis y Funcionamiento .....	29
Conclusiones .....	31
Referencias Bibliográficas .....	32

CAPITULO III .....	33
3. Diseño de Programa para el Probador de Circuitos	
Integrados .....	33
3.1 Descripción del Problema .....	33
3.2 Análisis del Problema .....	34
3.3 Estructura del Programa .....	35
3.3.1 Variables .....	35
3.3.2 Inicialización .....	35
3.3.3 Portada e Indicaciones .....	36
3.3.4 Lista .....	36
3.3.5 Impresión .....	36
3.3.6 Validación .....	37
3.3.7 Pruebas .....	37
3.4 Guía del Usuario .....	37
3.5 Mantenimiento del Programa .....	41
Conclusiones .....	43
Referencias Bibliográficas .....	44
Conclusiones y Recomendaciones Generales .....	45
Anexo 1. Recomendaciones para la Implementación de circuitos Impresos .....	47
Anexo 2. Listado del Programa Principal .....	50
Anexo 3. Ejemplo del Listado de la Unidad Pruebas y Dibujos .....	61
Anexo 4. Especificaciones de los componentes utilizados .....	69
Anexo 5. Diagrama del Módulo de Prueba .....	87

## LISTADO DE FIGURAS

	PAG.
1.1 Diagramas de Bloques de las partes mas importantes ..... circuito de Interfase	3
1.2 Pin Out de las ranuras de expansión .....	4
1.3 Diagramas del Circuito de Interfase .....	5
1.4 Diagramas de Bloques del 8255A .....	6
1.5 Byte de Control .....	8
1.6 Modo 1: Entrada con Protocolo para el grupo A .....	11
1.7 Modo 1: Salida con Protocolo para el grupo B .....	11
1.8 Pin Out del 8255A .....	13
2.1 Modulo 1 mostrado en diagramas de bloques .....	19
2.2 Diagrama del Circuito decodificador .....	20
2.3 Circuito Multiplexador de Canales .....	23
2.4 Divisores de voltaje para proporcionar 2 y 3.5 volt. ...	25
2.5 Circuito para proporcionar diferentes voltajes .....	25
2.6 Circuito para proporcionar niveles de referencia a diferentes pines de la base .....	27
2.7 Circuito de la base 2 .....	28
3.1 Diagramas de bloques del programa .....	35
3.2 Presentación del Programa .....	38
3.3 Librería de Integrados disponibles .....	38
3.4 Confirmación del Integrado .....	39
3.5 Impresión de Integrado y resultados .....	39
3.6 Elección de Probar otro integrado .....	40

## LISTADO DE TABLAS

	PAG.
1.1 Configuración de Puertos en el Modo 0 .....	9
1.2 Resumen de los modos de operación del 8255A.....	13
2.1 Rangos de valores del puerto B, a los cuales responden las diversas salidas del decodificador 1 .....	21
2.2 Rangos de valores de las cinco líneas menos signifi- cativas del puerto B, a los cuales responden las di- versas salidas del decodificador 2. ....	22

# CAPITULO I

## DISEÑO DEL CIRCUITO DE INTERFACE

### Introducción

La importancia que tiene el circuito de una interfase, es la comunicación que puede existir entre la computadora y otro circuito fuera de esta, ya que es trascendental para el manejo y control de señales en una gran variedad de propósitos. La elección de una interface se hace dependiendo de la comunicación de datos que se va a tener, ya sea serie o paralelo y de las características que tiene el circuito externo a controlar. Para los propósitos de este trabajo se diseñó una interfase de comunicación paralelo basado en el chip 8255 PPI (Interfase Programable de Periféricos) que permitirá el control del modulo de prueba. El diseño de la interfase se realizó tomando como base circuitos sugeridos para ser usados con microcomputadoras IBM o compatibles referencia 1.

En este capítulo se describen las características y el funcionamiento del IC 8255 así como también los canales I/O de expansión de la computadora que sirven para la conexión entre la interfase y la computadora.

#### 1.1 Interconexión de la Interfase con la Computadora.

La conexión de la interfase con la computadora se hace por medio de los canales I/O de expansión (slots), que poseen las computadoras PC o compatibles. La figura 1 muestra las partes mas importantes que forman parte del circuito de interfase y la relación que tiene con los conectores de la computadora.

La descripción de los canales I/O de expansión se muestran en la figura 2. En los parafos que siguen se trata la función de cada uno de ellos.

#### 1.2 Descripción de los canales I/O de Expansión de las Computadoras IBM PC o Compatibles.

##### SA0-SA19 (E/S):

Representa el bus de direcciones del computador y son usados para direccionar memoria y dispositivos de Entrada/Salida dentro del sistema. Son 20 líneas de direccionamiento que permiten acceder hasta un megabyte de memoria.

##### CLK (S):

Es la señal de reloj de 8 ó 10 Mhz. del sistema. Debe usarse solo para efectos de sincronización.

RESET DRV (S):

Inicializa la lógica del sistema al encender la máquina o durante un bajo voltaje.

SD0-SD7 (E/S):

Son los 8 bits menos significativos del bus de datos del sistema. En transferencias de 8 bits son los únicos utilizados.

BALE (S):

Abreviatura de "Buffered Address Latch Enable ". Esta señal proviene del controlador del bus (82288) y es utilizada para enclavar direcciones válidas del microprocesador. Se utiliza como indicador de direcciones válidas del DMA o del CPU.

I/O CH CK (E):

Esta señal proporciona al sistema información de error de paridad en memoria o dispositivos en los canales de entrada/salida. Es activa en bajo.

I/O CH RDY (E):

Cuando esta señal está en bajo, indica que el dispositivo periférico no está listo, permitiendo prolongar los ciclos de entrada/salida. Es utilizada para manejar dispositivos periféricos lentos.

IRQ3-IRQ7 (E):

Son señales de solicitud de interrupción, IRQ7 es la de mayor prioridad y IRQ3 la de menor.

IOR (E/S):

Esta señal le indica a un periférico que maneje sus datos a través del bus de datos. La señal es controlada por el microprocesador o el controlador del DMA o similares, presentes en el canal entrada/salida. Es activa en bajo.

IOW (E/S):

Esta señal le indica a un periférico que lea los datos presentes en el bus de datos. Es controlada en igual forma que IOR. Es activa en bajo.

SMEMR (S):

Esta señal indica a dispositivos de memoria manejar datos a través del bus de datos. Es controlada por el CPU o el controlador del DMA. Es activa solo para rangos menores de 1 megabyte de memoria.

SMEMW (S):

Esta señal indica a los dispositivos de memoria almacenar los datos presentes en el bus de datos. Controlada por el CPU o por el controlador del DMA. Es activa solo para rangos menores de un megabyte.

DRQ1, DRQ2, DRQ3 (E):

Son señales de solicitud de transferencia por canales DMA. DRQ1 es la de mayor prioridad y DRQ3 la de menor prioridad. Se mantiene en alto hasta que el correspondiente DACK se activa.

DACK1,-DACK2,-DACK3 (S):

Se utiliza para acceder solicitudes de DMA. Son activas en bajo.

AEN (S):

Cuando esta señal se activa (alto), el CPU cede el control sobre el bus de direcciones y bus de datos al controlador del DMA, permitiendo la transferencia de DMA.

REFRESH (E/S):

Esta señal indica un ciclo de refrescamiento. Es activa en bajo y puede controlarla el microprocesador en el canal de entrada/salida.

T/C (S):

Proporciona un pulso alto cuando es alcanzada la cuenta final por cualquier canal DMA.

OSC (S):

Señal de reloj de 14.31818 Mhz, no sincronizada.

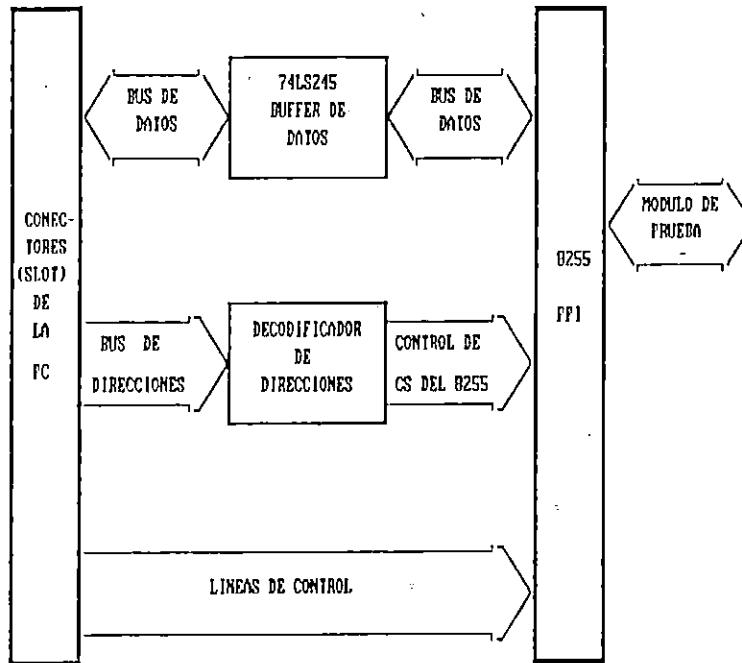


Figura 1.1 Diagrama de bloques de las partes más importantes del circuito de interfase.



### 1.3 Descripción del Circuito de Interfase.

Los canales que se utilizan en el presente diseño son: (ver figura 1.2).

De la A2-A9, conforman el bus de datos entrada/salida de 8 bits, con el bit menos significativo (LSB) en el A2 (SD7) y el más significativo (MSB) en el A9 (SD0) estas líneas son manejadas por el 74HC245, un transmisor/receptor de datos.

La dirección del flujo de datos, es controlada por la línea de IOR (B14), la cual se conecta al pin 1 del 74HC245.

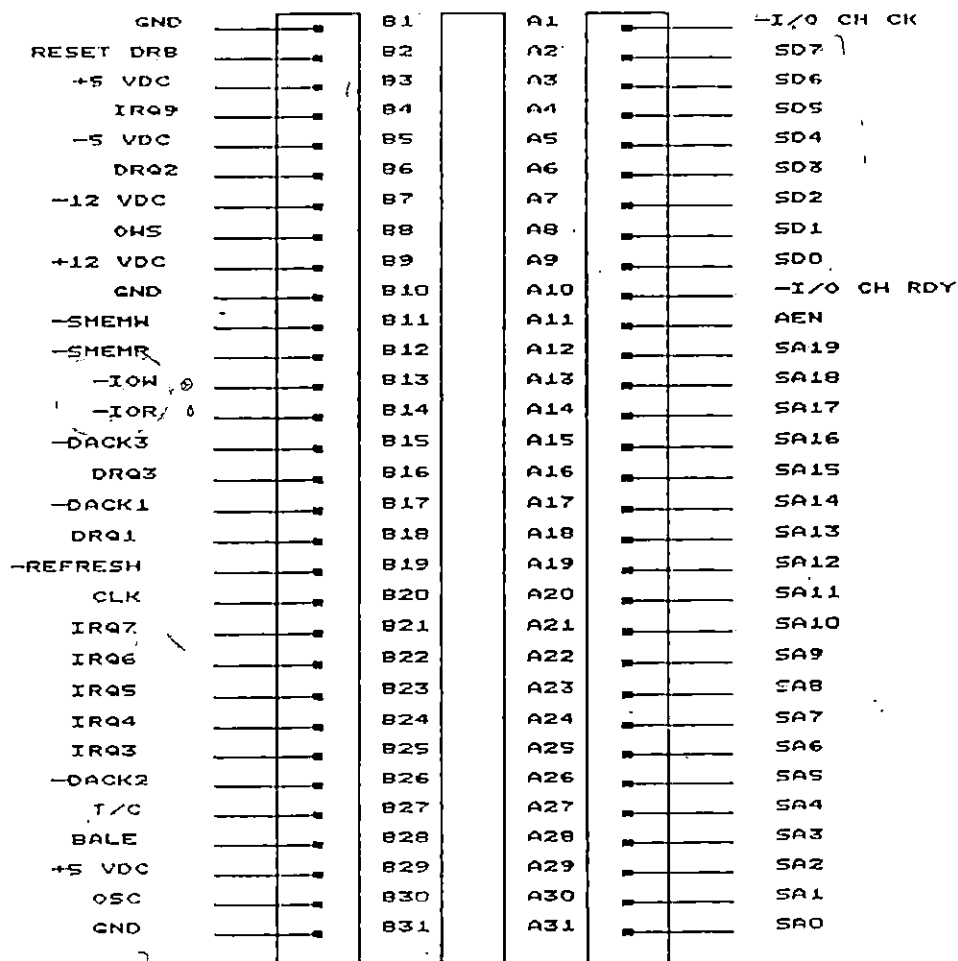


Figura 1.2 Pin Out de las ranuras de expansión.

La habilitación del 74HC245 (pin 19) es manejada por la salida del decodificador de direcciones; esta debe de ser baja para que el 74HC245 permita el paso de datos.

El decodificador de direcciones no es mas que una compuerta NAND de ocho entradas (74C30), a cuyas entradas se conectan los canales A22-A29 que corresponden a las líneas de direcciones SA9-SA2, además se conecta la línea de AEN (A11) a la entrada del 74C30 para evitar que la interfase sea habilitada accidentalmente durante la transferencia del DMA. La salida del 74C30 controla la habilitación del 74HC245 y además, junto con la línea SA4 (A27) controlan a través de una compuerta OR el "chip selec" del 8255.

Las líneas de dirección SA1 Y SA0, direccionan los puertos del 8255; de la manera siguiente :

- SA1,SA0 = 00, es el puerto A
- SA1,SA0 = 01, es el puerto B
- SA1,SA0 = 10, es el puerto C
- SA1,SA0 = 11, corresponde a la dirección del Byte de control

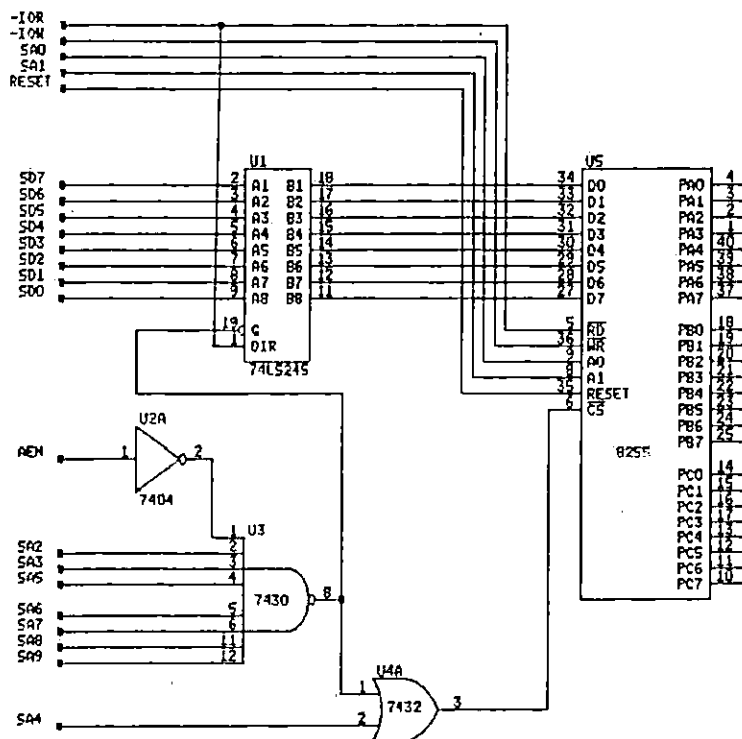


Figura 1.3 Diagrama del Circuito de Interfase.

Así la Interfase responde a 4 direcciones (que son direcciones disponibles en el rango I/O del mapa de memoria de la AT), tales direcciones son :

- 3EC Hex: puerto A
- 3ED Hex: puerto B
- 3EE Hex: puerto C
- 3EF Hex: dirección del byte de control

El diagrama del circuito de la Interfase puede verse en la figura 1.3.

#### 1.4 El PPI 8255 (Interfase Programable de Periféricos).

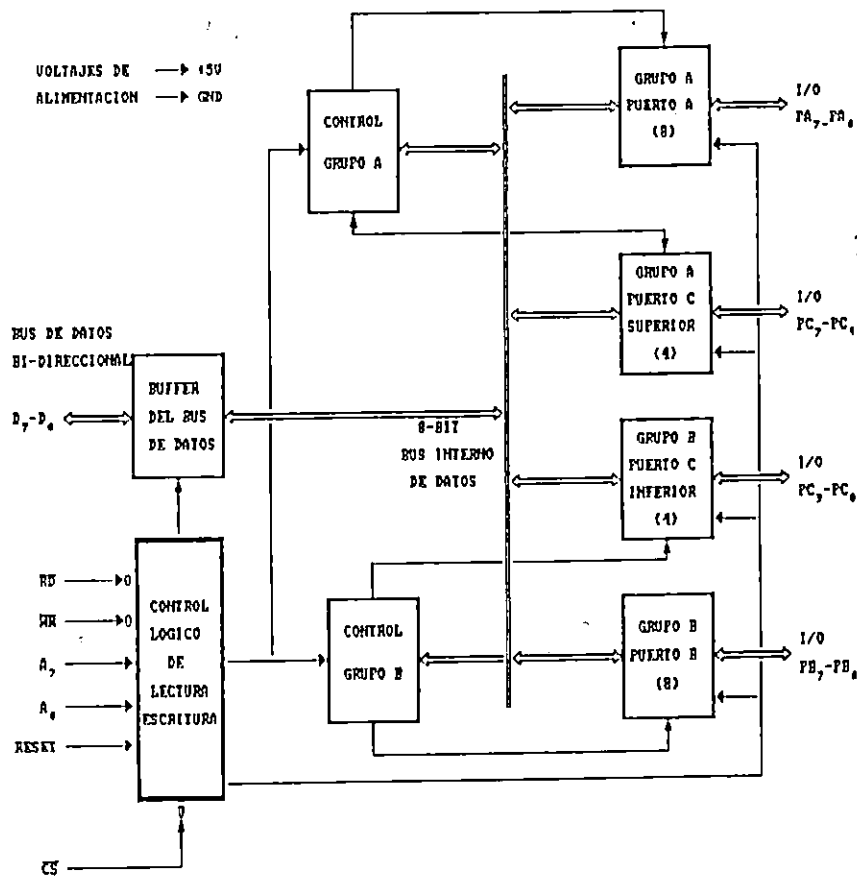


Figura 1.4 Diagrama de Bloques del 8255A.

Para comprender más claramente el funcionamiento de la Interfase es necesario describir las operaciones que puede realizar el 8255, el cual es un periférico programable de entrada/salida paralelo para propósitos generales y es la parte central del circuito de la Interfase, el diagrama del bloques del 8255A es mostrado en la figura 1.4. Es de la familia INTEL de circuitos de soporte para microprocesadores, capaz de proporcionar 24 pines programables de entrada/salida, divididos en tres puertos paralelos: puerto A, puerto B, y puerto C; los cuales pueden ser configurados de manera especial como se describe mas adelante. La configuración se hace por medio de software y no requiere circuitos lógicos externos para ello.

Antes de seguir describiendo las características de operación del 8255, cabe mencionar que todo lo que el hace, podría lograrse con simples compuertas y dispositivos similares, sin embargo esto involucra mayor costo por bit de entrada/salida y mucho mas espacio físico, complicando el diseño.

#### 1.4.1 Modos de Operación del 8255

El 8255 posee tres modos de operación conocidos como modo 0, modo 1 y modo 2. En el modo 0 los 8 pines de cada uno de los tres puertos son utilizados como entrada/salida. Los modos 1 y 2 utilizan las líneas del puerto C como entradas de solicitud de interrupción al CPU, estas interrupciones pueden ser enmascaradas o no mediante el formato bit set/reset.

El 8255 presenta tres modos básicos de operación :

- MODO 0 : Entrada/Salida elemental
- MODO 1 : Entrada/Salida con protocolo (Handshaking)
- MODO 2 : Bus bidireccional con protocolo (Handshaking)

El 8255 esta conformado por dos grupos de buses. El grupo A que es el puerto A más los 4 bits más significativos del puerto C y el grupo B formado por el puerto B más los cuatro bits menos significativos del puerto C.

Cuando se inicializa el 8255 (al encender la máquina), sus tres puertos quedan configurados como puertos de entrada. Los puertos A, B y C pueden ser configurados en modos de operación diferentes según sea necesario, escribiendo la palabra de control en el registro de control.

La palabra de control consta de ocho bits y es la que determina en que modos pueden trabajar los tres puertos; el registro de control es la dirección a la que responde la palabra de control.

Las direcciones a que responde la Interfase son las siguientes: (tres de estas direcciones corresponden a los puertos A, B y C y el cuarto es un registro de control de "solo escritura").

SA9	SA8	SA7	SA6	SA5	SA4	SA3	SA2	SA1	SA0		
1	1	1	1	1	0	1	1	0	0	3ECH	PUERTO A
1	1	1	1	1	0	1	1	0	1	3EDH	PUERTO B
1	1	1	1	1	0	1	1	1	0	3EEH	PUERTO C
1	1	1	1	1	0	1	1	1	1	3EFH	Palabra de control

El formato del byte de control se muestra a continuación:

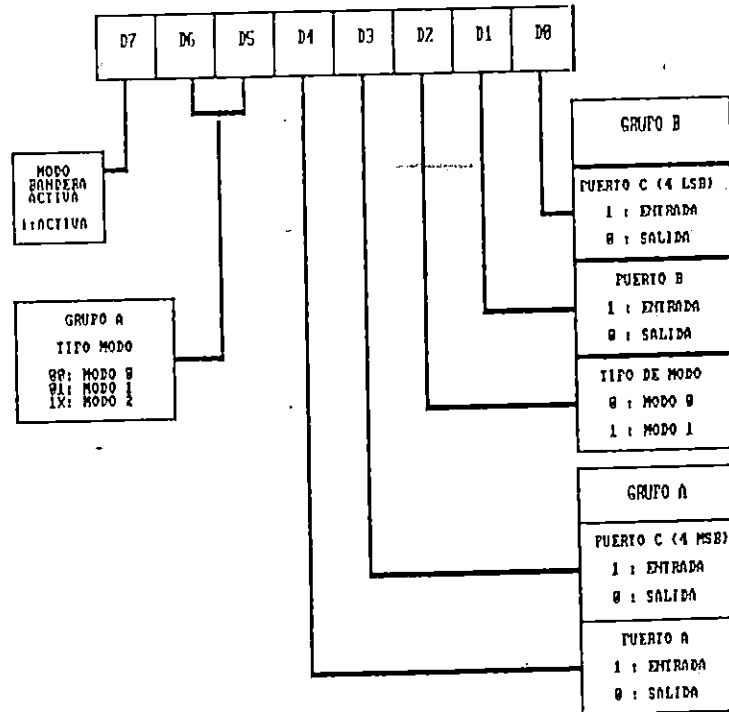


Figura 1.5 Byte de Control.

#### 1.4.2 Modo 0: Entrada/Salida Elemental.

Este modo configura a los tres puertos solamente como entrada/salida. Este modo de operación presenta las siguientes características:

- Los puertos A y B de 8 bits cada uno y el puerto C dividido en dos grupos de 4 bits cada uno los cuales pueden ser entrada o salida.
- Cualquier puerto puede ser entrada o salida

- Las salidas son enclavadas
  - Las entradas no son enclavadas
  - 16 formas de configurar en este modo
- A continuación se muestran las 16 posibilidades de configuración en el modo 0:

Tabla 1.1 Cofiguración de Puertos en el Modo 0.

BYTE DE CONTROL	PUERTO A	4MSB PUERTO C	PUERTO B	4LSB PUERTO C
01 Hex	salida	salida	salida	salida
81 "	salida	salida	salida	entrada
41 "	salida	salida	entrada	salida
C1 "	salida	salida	entrada	entrada
11 "	salida	entrada	salida	salida
91 "	salida	entrada	salida	entrada
51 "	salida	entrada	entrada	salida
D1 "	salida	entrada	entrada	entrada
09 "	entrada	salida	salida	salida
89 "	entrada	salida	salida	entrada
49 "	entrada	salida	entrada	salida
C9 "	entrada	salida	entrada	entrada
19 "	entrada	entrada	salida	salida
99 "	entrada	entrada	salida	entrada
59 "	entrada	entrada	entrada	salida
D9 "	entrada	entrada	entrada	entrada

Por razones de facilitar las conexiones en el diseño de la Interfase, ya que esta debe implementarse en circuito impreso, el orden del bus de datos esta invertido; teniendo D7 como el LSB y el D0 como el MSB, así tambien las salidas de los puertos estan de igual manera. Unicamente es trasposición de líneas por lo tanto cambio de palabra de control, el cual es mostrado en la tabla 1.1.

Para una mejor comprensión la selección de la palabra de control se hace según la necesidad que se va a tener por ejemplo: si se quiere tener como puertos de salida el A y el B y entrada el puerto C con la instrucción en Pascal que permite tener acceso directo a los puertos, se escribe de la siguiente manera `Port[$3EF]:=91`, y el 8255 queda así configurado. Para los propósitos de control en el Módulo de prueba la elección del modo 0 para la interfase es el mas idóneo y por esta razón se ha desarrollado con mayor profundidad, dejando los otros dos modos como información general.

#### 1.4.3 Modo 1 : Entrada/Salida con protocolo (Handshaking)

Este modo permite comunicarse con el computador con el uso de señales de protocolo o handshaking. Las características

principales de este modo son :

- Presenta dos grupos (A y B) de líneas de comunicación.
- Cada grupo esta formado por un puerto de 8 bit para datos y un puerto de 4 bit para control (Handshaking).
- El puerto de 8 bit puede ser de entrada o de salida. Las entradas y las salidas son enclavadas.
- El puerto de 4 bit es usado para controlar el estado del puerto de 8 bits.

En este modo de operación hay dos posibilidades para cada bloque; entrada con protocolo o salida con protocolo. En la configuración como puerto de entrada con protocolo, las señales de Handshaking son las siguientes:

1. -STB : Un cero en esta señal de entrada, carga el dato en el latch de entrada.
2. IBF : Un uno en esta señal de salida, indica que el dato ha sido cargado en el latch de entrada.
3. INTR : Un uno en esta señal de salida puede ser utilizada para interrumpir el CPU cuando un dispositivo de entrada solicita servicio. Se activa cuando -STB y IBF son unos y hay habilitación de interrupciones. Se activa y desactiva cuando ocurre una operación de lectura.

La interrupción puede ser "enmascarada" o no mediante el bit set/reset del PC4 para el grupo A y el set/reset del PC2 para el grupo B.

GRUPO A. BYTE DE CONTROL

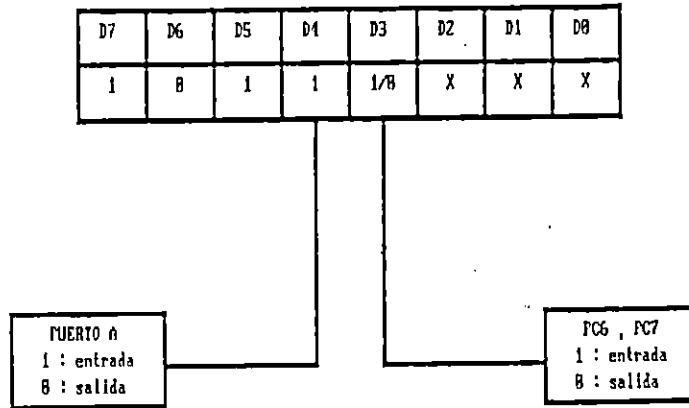


Figura 1.6 Modo 1: Entrada con protocolo (Handshaking) para el grupo A.

En la configuración de puerto de salida con protocolo, las señales de Handshaking son las siguientes:

1. -OBF : Un cero en esta señal de salida, indica que el CPU ha escrito un dato al puerto de salida.
2. -ACK : Un cero en esta señal de entrada, le indica al 8255A que el dato que se envió ha sido aceptado.
3. INTR : Un uno en esta señal de salida, puede ser utilizada para interrumpir el CPU cuando un dispositivo de salida ha aceptado datos enviados por el CPU. Se activa cuando -OBF y -ACK son unos y hay habilitación de interrupción. Se desactiva cuando se hace una operación de escritura.

La interrupción puede "enmascarse" o no mediante el bit set/reset del PC6 para el grupo A o el bit set/reset del PC2 para el grupo B.

#### GRUPO B. BYTE DE CONTROL

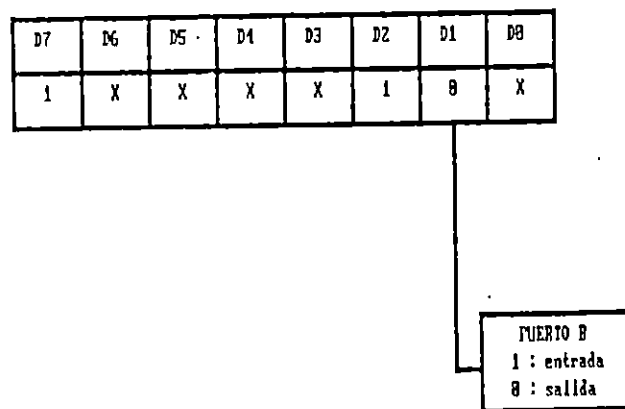


Figura 1.7 Modo 1: Salida con Protocolo (Handshaking) para el grupo B.

#### 1.4.4 Modo 2: Puerto Bidireccional con Protocolo.

Este modo opera con un solo puerto que sirve para enviar y recibir datos utilizando señales de protocolo. Sus características principales son:

- Usado en el grupo A únicamente.
- Un puerto de 8 bits bidireccional y un puerto de control de 5 bit.
- Entradas y salidas enclavadas.



Las señales INTR, -OBF, IBF, -ACK y -STB, tienen el mismo significado en este modo de operación que en el modo 1. El modo 2 es exclusivo del puerto A y lo convierte en un puerto bidireccional con señales de protocolo. El puerto B no puede operar en esta configuración.

Existen muchas maneras de combinar los modos de operación en el 8255A. El puerto A puede operar de 3 modos diferentes, el puerto B de dos modos diferentes. La forma de configurar los puertos depende del tipo de aplicación que se desee.

Para mayor detalle sobre los modos de operación consultar la referencia 4 al final del capítulo.

Los modos de operación del 8255 se resumen de la siguiente manera:

Tabla 1.2 Resumen de los Modos de Operación del 8255A.

	MODO 0		MODO 1		MODO 2
	ENT	SAL	ENT	SAL	SOLO GRUPO A
PA0	E	S	E	S	<----->
PA1	E	S	E	S	<----->
PA2	E	S	E	S	<----->
PA3	E	S	E	S	<----->
PA4	E	S	E	S	<----->
PA5	E	S	E	S	<----->
PA6	E	S	E	S	<----->
PA7	E	S	E	S	<----->
PB0	E	S	E	S	----
PB1	E	S	E	S	----
PB2	E	S	E	S	----
PB3	E	S	E	S	----
PB4	E	S	E	S	----
PB5	E	S	E	S	----
PB6	E	S	E	S	----
PB7	E	S	E	S	----
PC0	E	S	INTRB	INTRB	E/S
PC1	E	S	IBFB	-OBFB	E/S
PC2	E	S	-STBB	-ACKB	E/S
PC3	E	S	INTRA	INTRA	INTRA
PC4	E	S	-STBA	E/S	-STBA
PC5	E	S	IBFA	E/S	IBFA
PC6	E	S	E/S	-ACKA	-ACKA
PC7	E	S	E/S	-OBFA	-OBFA

### 1.5 Descripción de Pines del 8255A.

Los pines del 1 al 4 representa PA3, PA2, PA1, PA0 respectivamente, es decir los 4 bits menos significativos del puerto A.

El pin 5 es el -RD, señal que es bajo cuando se realiza una operación de lectura en algún puerto.

El pin 6 es el -CS, señal que habilita el funcionamiento del 8255A cuando esta en bajo.

El pin 7 es GND, la tierra del sistema.

Los pines 8 y 9 son A1 y A0. Estas líneas de direccionamiento accesan los puertos de siguiente manera:

A1	A0	PUERTO SELECCIONADO
0	0	A
0	1	B
1	0	C
1	1	Accesa el byte de control.

Los pines 10 al 13 son PC7, PC6, PC5, PC4, son los 4 bits menos significativos del puerto C (esto es por el cambio de líneas en el bus de datos para el presente diseño).

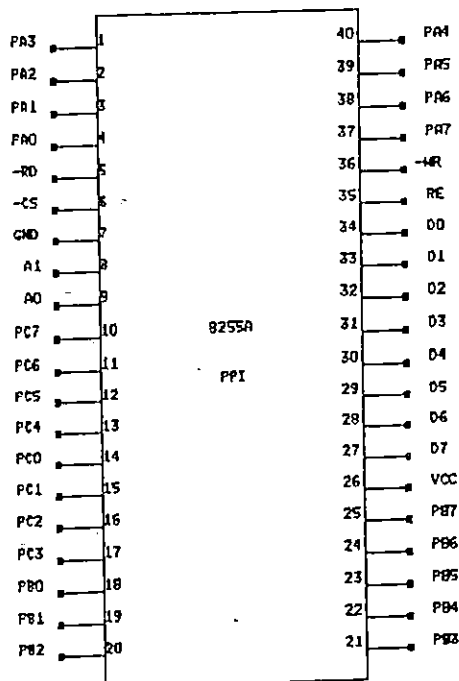


Figura 1.8 Pin Out del 8255A.

Los pines 14 al 17 son PC3, PC2, PC1, PC0, son los bits más significativos del puerto C.

Los pines 18 al 25 son PB0 a PB7, el puerto B.

El pin 26 es VCC, +5 voltios para el 8255A.

Los pines 27 al 34 son D0 a D7, el bus de datos que se conecta al computador. El pin 35 es el RESET, un alto en esta señal inicializa al 8255A colocando todos sus puertos como puertos de entrada.

El pin 36 es -WR, señal que es baja cuando se realiza una operación de escritura en algún puerto.

Los pines 37 al 40 PA7, PA6, PA5, PA4, respectivamente, es decir, los 4 bits más significativos del puerto A.

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

- El trabajo que realiza la interfase discutida en este capítulo puede ser hecho utilizando solamente dispositivos combinacionales y secuenciales, pero esto aumentaría el costo del espacio físico necesario. Este espacio esta limitado debido a que la Interfase debe de colocarse dentro de la computadora utilizando uno de sus puertos de expansión.
- Con la Interfase tratada alcanzamos el objetivo económico de mayor eficiencia al menor costo que en nuestro caso sería el menor costo por bit de entrada/salida.
- El diseño y construcción de una interfase da como resultado la base para la comunicación de la computadora con el mundo exterior, razón importante en los futuros trabajos de investigación en el control por medio de sistemas basados en computadoras IBM PC o Compatibles.
- La versatilidad de la interfase así como su facilidad de manejo y construcción la hace propicia para casi cualquier tipo de aplicación y por tanto susceptible a comercialización.

REFERENCIAS BIBLIOGRAFICAS

- 1- SARGENT, MURRAY : THE IBM PC FROM THE INSIDE OUT.  
ADDISON WESLEY, USA, 1986.
- 2- BRENNER, ROBERT C. : THE IBM PC TROUBLESHOOTING AND REPAIR  
GUIDE. MICROTRAND INC, USA, 1984.
- 3- IBM AT REFERENCE MANUAL, INTERNATIONAL BUSSINESS  
MACHINES CORP., FLORIDA, USA 1983.
- 4- MICROPROCESSOR AND PERIPHERAL HANDBOOK, INTEL, 1987.

## CAPITULO II DISEÑO DEL MÓDULO DE PRUEBA

### Introducción.

El objetivo del presente capítulo pretende establecer las bases del funcionamiento del módulo de prueba y la interconexión con el circuito de interfase. Una vez encontrada la forma de establecer el flujo de señales digitales entre la computadora y el "mundo exterior", se pasa a describir la aplicación que se da a la interfase estudiada en el capítulo anterior; es decir, se discutirá en el presente capítulo el diseño del módulo que permitirá realizar las pruebas a los diferentes circuitos integrados que forman parte de la lista o librería presentada.

Es claro que un circuito integrado puede ser sometido a una variedad de pruebas que determinarán su estado actual, pruebas que van desde la simple medición óhmica, hasta el trazado de curvas de respuesta de sus componentes a diferentes condiciones de voltaje, potencia, etc. En un sistema completo, todas esas pruebas deberían hacerse, para dar el mejor diagnóstico posible de sus parámetros dados por el fabricante; en este caso en especial únicamente se hará una prueba funcional en los dispositivos bajo prueba. Uno de los objetivos del presente trabajo es dar una aplicación un tanto variada y compleja de la interfase, aplicación que pone de manifiesto algunas bondades y formas de operación de la interfase, siendo el presente trabajo uno de los primeros en utilizar una interfase con la computadora IBM PC o compatible AT LEMON sistema 80286, como elementos controladores de un sistema periférico externo.

La importancia de este trabajo de graduación es considerarlo como punto de partida para el desarrollo de futuros trabajos en el control de periféricos más complejos basados en la interfase antes mencionada y también proporcionar un medio de comprobación del estado de operación de ciertos componentes integrados.

### Generalidades.

La prueba a que se someten los dispositivos es de tipo funcional, ya que se aplica señales de entrada (1 ó 0 lógicos), esto se hace dando las combinaciones respectivas de las tablas lógicas de cada componente; para verificar el funcionamiento del dispositivo; si existe una diferencia entre las tablas lógicas y las distintas combinaciones, se establece que el dispositivo se encuentra en mal estado.

Los dispositivos que pueden probarse con el módulo aquí presentado, puede ser tanto de la familia TTL o CMOS, con la característica que el VCC y GND deben estar ubicados en los extremos, diagonalmente. Además se presenta una librería dentro

del programa que contiene los dispositivos que pueden probarse.

Los niveles lógicos de prueba que se utilizarán, serán aquellos que establezcan las peores condiciones de trabajo del dispositivo, es decir, los niveles de voltaje que se usan serán:

Para familia TTL:

Voltaje de entrada alto mínimo:  $V_{IH(\min)} = 2$  voltios  
Voltaje de entrada bajo máximo:  $V_{IL(\max)} = 0.8$  voltios

Para familia CMOS:

Voltaje de entrada alto mínimo:  $V_{IH(\min)} = 0.7V_{DD} = 3.5$  volt.  
Voltaje de entrada bajo máximo:  $V_{IL(\max)} = 0.3V_{DD} = 1.5$  volt.

Las secciones que siguen contienen en forma más detallada lo que se ha dicho anteriormente, reforzando con los diagramas circuitales correspondientes.

Cabe mencionar que el módulo en sí, nada más es una parte del sistema de prueba, que dependerá de las señales generadas por la Computadora, esto implica necesariamente el desarrollo del programa o software respectivo; que será tema de discusión en el tercer capítulo.

## 2.1 Base 1. Flujo de Señales. Análisis de Funcionamiento.

Esta base (y la circuitería asociada con ella) servirá para probar la mayoría de dispositivos considerados en la librería propuesta, es decir aquí pueden probarse entre otros: compuertas, flip-flops, monoestables, contadores, buffers, decodificadores, desplazadores de registro, etc., (estos son llamados circuitos lógicos MSI en el transcurso del documento); además en esta base también podrán probarse memorias RAM estáticas.

La base usada para los circuitos MSI es apropiada para colocar dispositivos que posean de 14 hasta 20 pines; debido a la circuitería asociada, únicamente pueden probarse circuitos integrados que tengan la característica siguiente: poseer el pin de VCC y GND en sus pines extremos, por ejemplo: Los de catorce pines, VCC en el pin 14 y GND el pin 7, y así sucesivamente.

La prueba para los dispositivos MSI, comienza estableciendo los valores de los niveles lógicos, dependiendo si el dispositivo a probar es de la familia TTL, los niveles lógicos alto y bajo tienen los valores de 2 volt. y 0.8 volt. respectivamente, con el fin de que el dispositivo trabaje en las peores condiciones.

Como se mencionó anteriormente, la prueba consiste en la aplicación de los valores de la tabla de verdad correspondiente

al dispositivo bajo prueba y la lectura de la respuesta del dispositivo para cada combinación de valores presentes en sus entradas, luego estas respuestas son comparadas con las que debería dar el dispositivo, esta comparación determina si el elemento se encuentra en buen o mal estado.

Si en caso el integrado bajo prueba tenga varios dispositivos del mismo tipo en su interior, se probará cada dispositivo, por ejemplo, el chip 7400 contiene cuatro compuertas NAND; la prueba se hace a cada compuerta sucesivamente, es decir primero se prueba la compuerta NAND1, luego la NAND2, etc., estableciendo el estado de cada compuerta, esto se hace porque podría ser que la 1, 2 y 3 estarían en buen estado pero la 4 no.

En el caso de las memorias RAM estáticas, la prueba consiste en escribir una determinada palabra a cada una de las localizaciones de memoria y luego "leer" esas localizaciones, a fin de establecer si lo que se "escribió" está realmente almacenado en ella. De encontrarse alguna diferencia se establece el mal estado del dispositivo, indicando además la localización de la memoria en la cuál se encuentra la(s) diferencia(s).

El módulo asociado con la base 1 es mostrado en forma de diagramas de bloques en la figura 2.1.

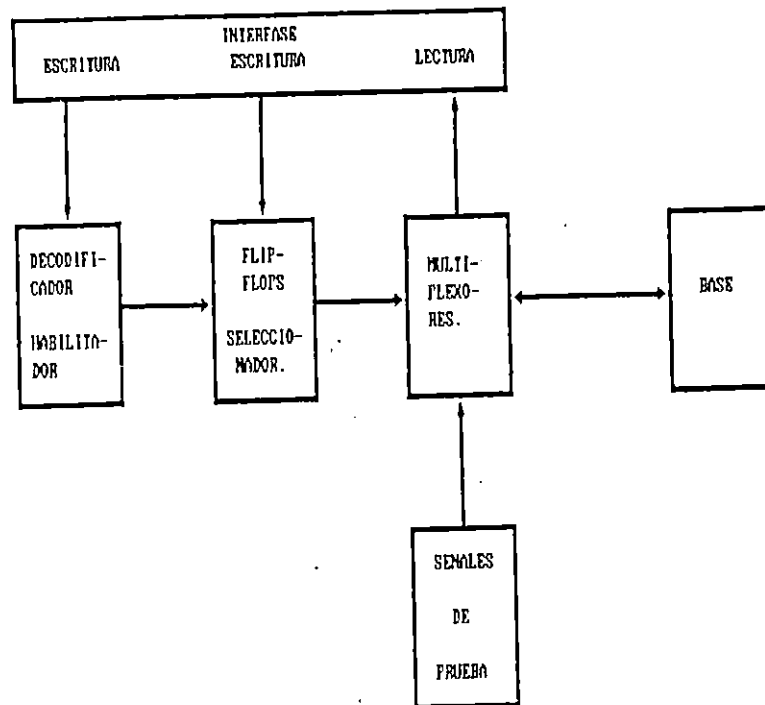


Figura 2.1 Modulo 1 mostrado en diagramas de bloques.



El bloque marcado como INTERFASE, ya se discutió ampliamente en el capítulo anterior, se limitará a decir que se utilizan los tres puertos que posee (A, B, y C), para establecer la comunicación entre la Computadora y el dispositivo bajo prueba, lo cual implica señales de escritura para efectuar la simulación y la habilitación a cada pin, y de lectura para establecer como responde el dispositivo y determinar si está en buen o mal estado.

Las operaciones de escritura en la mayoría de casos, se realizan con los puertos A y B de la Interfase, mientras que la lectura de señales se hará con el puerto C.

El circuito decodificador-habilitador que se muestra en la figura 2.2, está constituido por dos circuitos integrados 74LS138 (decodificadores), los cuales tienen como función habilitar aquellos dispositivos en los cuales se desee recibir o enviar una determinada información. Presenta a la salida 16 líneas habilitadoras, las cuales se activan dependiendo del código binario presente en sus entradas, cada salida está controlando a un solo dispositivo.

Cada uno de los dos decodificadores-multiplexores 74LS138 posee ocho líneas de salida ( $Y_0$  a  $Y_7$ ), haciendo un total de 16 líneas decodificadas, las cuales se mantienen en un estado lógico uno cuando los dispositivos se encuentren desactivados. Cuando se habilita uno de los dispositivos se activa una de las salidas, la cual depende de la combinación binaria presente en las entradas de selección A, B y C.

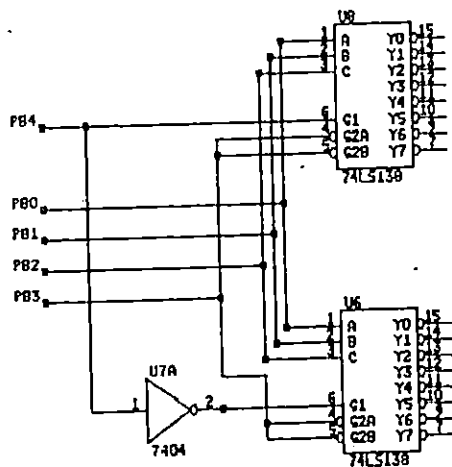


Figura 2.2 Diagrama del circuito decodificador.

La salida activada toma el estado lógico CERO mientras el dispositivo esté habilitado. A las entradas de selección A, B y C se conectan las líneas B0, B1 y B2 respectivamente del puerto B, previamente configurado como puerto de salida, el valor de estas líneas hará que se habilite una de las ocho líneas de salida.

Las entradas G1, G2A, y G2B son de habilitación al dispositivo. G2A y G2B, se accesan simultáneamente, las cuales habilitan la decodificación cuando su estado es un cero lógico, y lo desactivan al estar en un lógico uno; a estas entradas se conectan las línea B3 del puerto B.

G1 habilita al dispositivo al encontrarse en un estado lógico uno y lo desactiva en el estado contrario, a la entradas G1 del decodificador 1 se conectará la línea B4 y a la entrada G1 del decodificador 2 se conectará la línea B4 negada, es decir, pasará antes por un inversor. Con esto se asegura que en un momento determinado solo un decodificador este habilitado y por consiguiente solo una línea de salida este activada, evitando así una habilitación simultánea que podría generar un funcionamiento no coordinado del sistema.

Las tablas 1 y 2, presentan los rangos de valores del puerto B a los cuales responden los decodificadores, así también muestran los dispositivos que controlan cada una de las 16 líneas de salida que se disponen.

Tabla 2.1 Rangos de valores del puerto B, a los cuales responden las diversas salidas del decodificador 1.

Valores en las líneas del puerto B					Salida activada	Dispositivo Controlado
B4	B3	B2	B1	B0	Decodificador 1	Flip-Flop
1	0	0	0	0	Y0	FF1
1	0	0	0	1	Y1	FF2
1	0	0	1	0	Y2	FF3
1	0	0	1	1	Y3	FF4
1	0	1	0	0	Y4	FF5
1	0	1	0	1	Y5	FF6
1	0	1	1	0	Y6	FF7
1	0	1	1	1	Y7	FF8

Tabla 2.2 Rangos de valores de las cinco líneas menos significativas del puerto B, a los cuales responden las diversas salidas del decodificador 2.

Valores en las líneas del puerto B					Salida Activada	Dispositivo Controlado
B4	B3	B2	B1	B0	Decodificador 2	Flip-Flop
0	0	0	0	0	Y0	FF9
0	0	0	0	1	Y1	FF10
0	0	0	1	0	Y2	FF11
0	0	0	1	1	Y3	FF12
0	0	1	0	0	Y4	
0	0	1	0	1	Y5	
0	0	1	1	0	Y6	
0	0	1	1	1	Y7	

Por ejemplo, si se desea habilitar al FF1, se debe enviar por el puerto B la combinación  $XXX10000_2$ , mientras que si se quiere habilitar al FF9, se debe enviar la combinación  $XXX00000_2$ . Note que el estado de los tres bits más significativos del puerto B, no tienen nada que ver con el funcionamiento de los decodificadores, por lo cual el estado se toma como una condición de "no importa".

### 2.1.1 Multiplexador de Pines.

Ya que entre los dispositivos que se prueban se encuentran: compuertas, buffers, inversores, contadores, memorias, flip-flop, etc., y estos no presentan compatibilidad con el número de pines, ya que en su mayoría la distribución de pines no es la misma, es decir, que para unos, un pin determinado corresponde a una de las entradas del dispositivos y para otros es una salida o la señal del reloj, como es el caso de los flip-flops y contadores, necesitando en este caso una señal diferente para cada dispositivo de prueba. Es por ello que se necesita tener para cada pin, diferentes señales, para seleccionar la que deberá ser proporcionada al pin, dependiendo del tipo de dispositivo a probar.

La función del multiplexador mostrando en la figura 2.3, es proporcionar una de las señales a cada uno de los pines en el momento que se realiza la prueba, esto se logra con la ayuda de multiplexores analógicos (4051B), los cuales contienen 8 canales de entrada analógica y una salida, además su entrada habilitadora y tres entradas de selección son controladas digitalmente.

Como la base que se utilizará es de 20 pines, con VCC conectado siempre al pin 20 y GND conectado al pin 10, la razón para esto es que la mayoría de circuitos integrados comerciales poseen esta configuración de alimentación en sus pines extremos; para aplicar

las señales adecuadas a los pines de la base se utiliza 18 multiplexores analógicos (4051B) y 9 74LS377 (8 flip-flop D en cada uno).

Los multiplexores analógicos proporcionan las señales a cada uno de los pines que contiene la base donde se coloca el dispositivo que será probado. Ellos envían la señal o la reciben, dependiendo si el pin del dispositivo es entrada o salida respectivamente. De los 8 flip-flops tipo D que contiene cada integrado 6 son utilizados para controlar a los diferentes 4051B; tres sirven para controlar un 4051B y los otros tres a otro, por lo tanto cada 74LS377 puede manejar dos multiplexores cada uno.

Es de recordar que un flip-flop tipo D, coloca en su salida Q el valor lógico presente en su entrada D, cuando en su entrada de reloj se da la transición necesaria; en el caso del IC 74LS377, la entrada de reloj es común para los 6 flip-flop y necesitan una transición de cero a uno (transición positiva) en su entrada de reloj (pin 11) para colocar el valor presente en sus entradas D, en sus correspondientes salidas Q. La forma como se realiza la selección de uno de los ocho canales diferentes de cada multiplexor se muestra en la figura 2.3. Como puede observarse las entradas D de los flip-flops se conectan a las seis líneas menos significativas del puerto A, previamente configurado como puerto de salida, una a cada flip-flop; ahora bien los nueve 74LS377, así conectados se encuentran en paralelo, pero solo uno a la vez dejará pasar a sus salidas las señales presentes en sus entradas, esto se logra por medio de la entrada de reloj de los 74LS377 y de los decodificadores los cuales solo podrán hacerlo uno por uno para cada 74LS377.

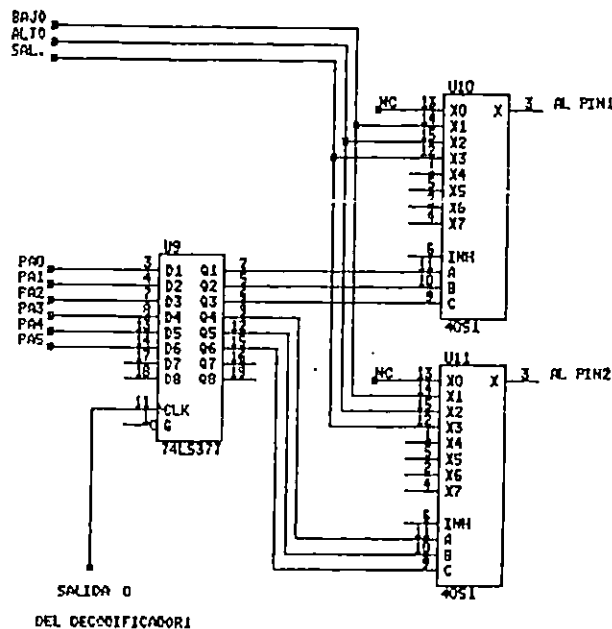


Figura 2.3. Circuito Multiplexador de Canales.

Luego las salidas de los flip-flop estan conectadas a las entradas de selección ABC de los multiplexores, las cuales dependiendo del valor lógico que se tenga, así dejarán pasar a la salida, la entrada correspondiente.

La salidas Q1, Q2 y Q3 controlan un multiplexor y las salidas Q4, Q5 y Q6 controlan al otro. Por ejemplo si se quiere conectar al pin 1 la señal presente en el canal 1 y al pin 2 en el canal 4, se debe habilitar al FF1 y luego enviar por el puerto A, la combinación XX100001<sub>2</sub>.

### 2.1.2 Señales de Prueba.

Como se puede observar, debido al tipo de multiplexores analógicos utilizados, se tiene la disponibilidad de proporcionar a cada pin ocho tipo de señales distintas, las cuales estan dadas de acuerdo al elemento a probar. Debido al análisis realizado, las ocho señales no serán las mismas para cada pin, inclusive para algunos no será necesario utilizar los ocho canales mientras que para otros sí. Sin embargo existen cinco señales básicas que son comunes a todos los pines y por tanto deberán estar disponibles para cada uno.

Esto significa que los ocho canales con que se cuentan, cinco ya están reservados para estas señales, quedando los otros tres canales disponibles para conectar señales "especiales" que se describirán más adelante; por ahora solo se describen las cinco señales básicas que son:

- Canal 0, se deja libre de señales.
- Canal 1, para conectar el nivel lógico 0.
- Canal 2, para conectar el nivel lógico 1.
- Canal 3, se conecta una línea del puerto C, la cual servirá para leer la respuesta que da el dispositivo bajo prueba, a las distintas combinaciones de entrada, esto se puede apreciar mejor observando la figura 2.3.

El objeto de dejar el canal 0, sin conexión es para que al inicializarse la prueba, el dispositivo bajo prueba se encuentre libre de señales. Esto significa que al inicializar la simulación los multiplexores se ponen con la combinación binaria 000<sub>2</sub> en las entradas CBA de selección, por ejemplo antes de conectar cualquier integrado a probar se debe poner todos los multiplexores en el canal 0, con esto se asegura que el integrado puesto en la base no tenga ninguna señal indeseada que pueda dañarlo es decir, si un determinado pin fuese salida y el multiplexor en ese pin estuviese en un nivel alto, podría haber daño en el circuito bajo prueba si en ese momento su combinación de entrada origina un nivel bajo en su salida. Esta condición de "no conexión" sirve como prevención de posibles riesgos.

El nivel lógico 1 que estará presente en el canal 2 de los

multiplexores tendrá un valor tal, que dependerá de la familia lógica a la cual pertenezca el dispositivo bajo prueba. Como se mencionó antes, los valores serán: 2 V. para TTL y 3.5 V. para CMOS. Estos valores se dan con el fin de que el dispositivo trabaje en las peores condiciones. Estos voltajes se obtienen por medio de divisores de tensión tal como se muestra en la figura 2.4. Estos divisores se implementan con amplificadores operacionales LM741 en modo seguidor, estos proporcionan un voltajes "aislado" con el fin de evitar el efecto de carga que podría ocasionar la conexión con los dispositivos bajo prueba. El circuito divisor en conjunto establece un consumo de potencia de 3.8 mW.

Para el nivel uno lógico, los dos niveles de voltaje son multiplexados, dependiendo del dispositivo bajo prueba; el nivel elegido se conecta al canal 2 de todos los multiplexores de pines de la base 1, tal como se muestra esquemáticamente en figura 2.5.

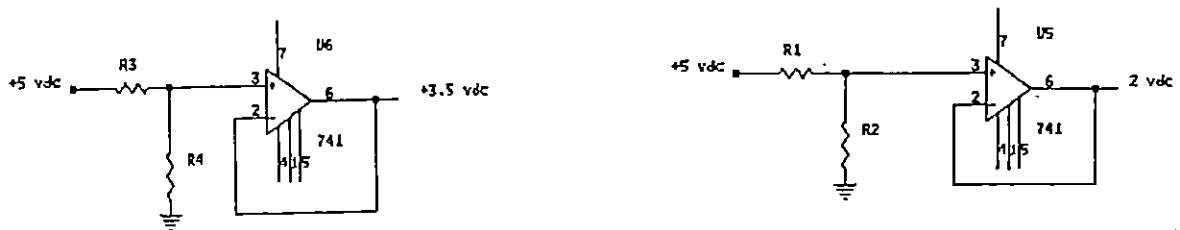


Figura 2.4 Divisores de voltaje para proporcionar 2 y 3.5 Volt.

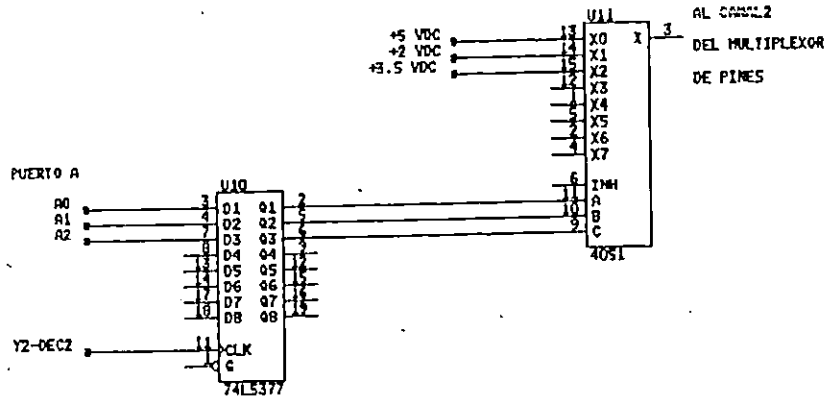


Figura 2.5 Circuito para proporcionar diferentes voltajes.

Además de los niveles de 2 y 3.5 Volt., se tiene disponibilidad del nivel de 5 Volt., nuevamente utilizamos flip-flop tipo D y

multiplexores 4051B para seleccionar el nivel de voltaje deseado. Por ejemplo, si el dispositivo bajo prueba es una compuerta TTL, se utiliza el nivel de 2 Volt. para el uno lógico, para proporcionarlo al canal 2 de los multiplexores de pines de la base, la secuencia a seguir es la siguiente:

- Habilitar al flip-flop tipo D, nombrado como FF10.
- Enviar por las líneas del puerto A, la combinación en sus tres líneas menos significativas, 010. Este nivel se mantiene hasta que sea sustituido por otro valor o por ausencia de energía.

### 2.1.3 Lectura de señales.

Hasta ahora se ha descrito algunas señales que pueden proporcionarse al dispositivo bajo prueba con la ayuda de los flip-flops y los multiplexores; considerando operaciones de escritura de señales. Sin embargo la interacción entre todos los elementos del sistema se completa cuando se realizan operaciones de lectura. La versatilidad de la Interfase permite realizar dichas operaciones de lectura de forma un tanto simple es decir definiendo nada más un puerto determinado como puerto de entrada con una simple instrucción de software que permite a la computadora leer la señal presente en las líneas del puerto considerado.

En el presente diseño se utiliza uno de los dos puertos de 4 bits del puerto C para lectura de señales, es decir, se define al puerto C como puerto de entrada y sus líneas se conectan al canal 5 de los multiplexores de pines, así cuando un determinado pin del dispositivo sea salida, éste se conecta con una de las líneas del puerto C y así se puede leer los resultados que presenta el dispositivo.

### 2.1.4 Señales Especiales.

Las señales especiales que se consideran aquí, son aquellas que no son comunes o que no están disponibles para todos los pines de la base, en este caso las señales son: las salidas de un contador, para cuando se prueben memorias y una constante de tiempo (esto es para los monoestables).

La memoria RAM estática con mayor capacidad que se puede probar con este circuito es de 4K, lo que implica utilizar un contador de 12 bits 4040B para poder acceder todas las localizaciones de memoria, las memorias de menor capacidad podrán ser utilizadas con este contador, controlando su límite de cuenta por medio de software.

Asimismo a los IC que tienen la característica de ser de colector abierto es necesario conectarles a la salida una resistencia de carga, para limitar la magnitud de la corriente que pueden generar. Es de hacer notar que solo se pueden probar compuertas

de colector abierto. El valor de la resistencia que se elija debe ser lo más bajo posible a fin de no afectar mucho la rapidez de respuesta del elemento y además debe asegurar que la magnitud de corriente no sea grande. En este caso se utilizan las siguientes entradas del multiplexor:

Los pines del 1 al 6, son controlados por un multiplexor, y los pines del 14 al 19, son controlados por el otro. El resto de los pines no es necesario conectarles resistencia de carga, ya que todos los integrados que contienen compuertas solamente poseen 14 pines.

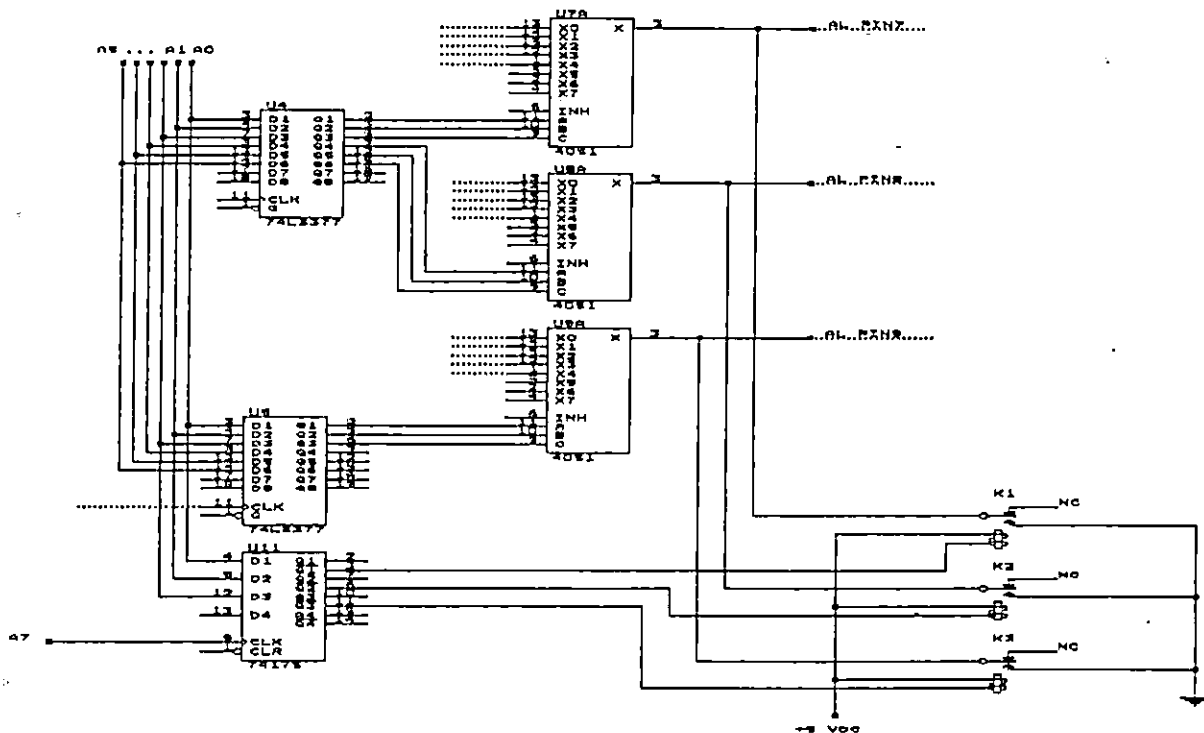


Figura 2.6 Circuito para proporcionar niveles de referencia a diferentes pines de la base.

Los pines 7, 8 y 9, poseen la característica de estar conectados a un relé cada uno, que al ser activados se conectan directamente a tierra y son utilizados cuando se someten a prueba dispositivos que contengan 14 ó 16 pines o cuando la base se utiliza para probar memorias RAM estáticas. La razón por la cual no se utiliza un canal del multiplexor conectado directamente a tierra, es por el manejo de corriente que poseen, la cual no es suficiente para drenar el flujo de todo el dispositivo bajo prueba; otra razón es por que cada canal presenta una resistencia interna



(aproximadamente 300 ohmios), la cual influye en dicha conexión, obteniéndose de esta forma una diferencia de tensión, la cual no asegura el funcionamiento del dispositivo, puesto que no se obtiene la referencia de tierra necesaria.

En la figura 2.6 se muestra la forma en que se hace la conexión a tierra para los pines 7, 8 y 9. Cada relé es controlado por las salidas -Q de un flip-flop los cuales son activados por la línea PA7 del puerto A. A las entradas D se conectan las líneas A0, A1, A2 y A3 del puerto A. En la figura 2.6 A0, A1 y A2 controlan los pines 7, 8 y 9 respectivamente.

La forma de funcionamiento se entiende con un ejemplo: para el caso que el IC a probar sea de 14 pines, se cierra el relé 1 (k1), conectado al pin 7 y se selecciona el canal 0 del multiplexor conectado a este pin para evitar cualquier tipo de problema. Además los restantes relés deben estar abiertos. Si el dispositivo es de 20 pines y del tipo MSI, todos los relés deben estar abiertos.

## 2.2. Base 2. Análisis y Funcionamiento.

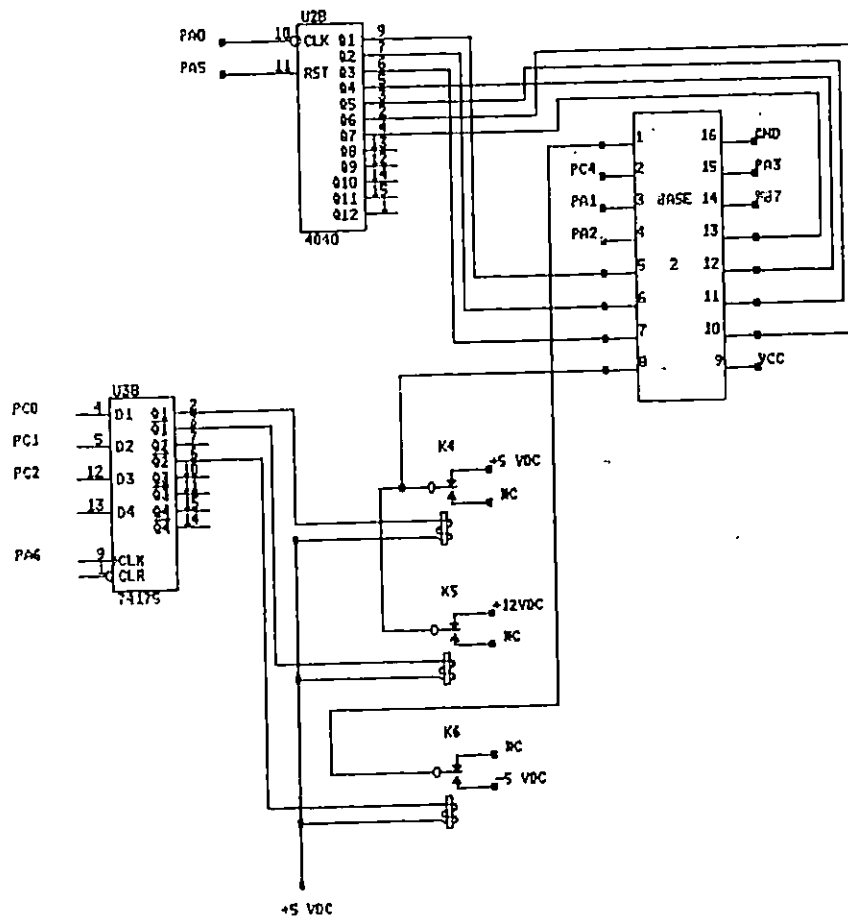


Figura 2.7 Circuito de la base 2.

Esta base esta reservada para probar memorias del tipo RAM dinámicas, de las más comúnmente utilizadas. A diferencia de las estáticas, las dinámicas requieren que la localización de memoria requerida ya sea para lectura o escritura se de en dos pasos: primero dando la posición de fila y luego la de columna, para lo cual se usan señales identificadas como RAS y CAS.

Además algunas necesitan de cuatro niveles de voltaje distintos para trabajar: +5, -5, +12 y 0 voltios. Esta son algunas razones por las cuales es necesario utilizar una base exclusiva para RAM dinámicas. Otra característica importante de las RAM dinámicas es su necesidad de refrescamiento de la localizaciones de memoria para evitar perder información. Comunmente el refrescamiento deberá hacerse cada 2 ms. Debido a que el software de manejo se hará con programación PASCAL y la velocidad de la computadora (10 MHZ) es suficiente rápida no será necesario el refrescamiento.

La base será de 16 pines, y la circuitería asociada constará de un flip-flop 74LS175, un contador 4040B y tres relés SPDT de 1 amp. como se muestra en la figura 2.7.

Para el manejo de la base 2 los puertos A y C se configuran como puertos de salida y el puerto B como puerto de entrada. El puerto A dará las señales de control necesarias como por ejemplo RAS, CAS, WE, etc. y el puerto C se encargará de manejar la conexión de los voltajes adecuados dependiendo del tipo de memoria.

#### 2.4 Base 3. Análisis y Funcionamiento.

Como se menciona al principio esta base esta destinada a probar memorias de tipo EPROM, entendiéndose como prueba, a la lectura de todas sus localizaciones de memoria. La circuitería necesaria para ello no es tan complicada como la descrita para las bases 1 y 2, ya que simplemente se utiliza un contador 4040B (contador de 12 bits CMOS). Las ocho líneas del puerto C y la línea A7 del puerto A sirve para generar la frecuencia de reloj respectiva del contador. El funcionamiento de los otros dispositivos es de igual forma al descrito para las bases anteriores.

Con esta base se podrá probar EPROMS del tipo siguiente: NTE 2532, NTE 2708, NTE 2716 y NTE 2732, todas ellas de 24 pines y que en la mayoría de casos presentan gran semejanza en la distribución de pines, lo que facilita el diseño del circuito necesario para la prueba.

El diagrama final del módulo de prueba se presenta en el anexo 5, donde se reunen los circuitos para la base 1, 2 y 3 así como su interconexión con la interfase discutida en el capítulo 1.

Uno de los objetivos del presente trabajo es implementar el circuito total en tarjeta impresa. Dicha implementación se llevó

a cabo con los recursos que dispone la Escuela de Ingeniería Eléctrica, esto dió lugar a situaciones que no habian sido consideradas en dicha implementación, tales problemas así como la soluciones dadas ellos son discutidos en el anexo 1.

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

- La facilidad de manejo de la interfase así como la versatilidad de sus tres puertos facilitan la generación de las señales necesarias para el manejo del módulo de prueba.
- La multiplexación de señales para los distintos pines es adecuada para el tipo de prueba funcional que se esta haciendo sin embargo deberá modificarse para efectos de someter a pruebas paramétricas a los diferentes circuitos integrados.
- Debido al tipo de componentes utilizados en el diseño del módulo vuelve sencillo su comprensión y susceptible a modificaciones posteriores.
- El consumo de potencia total del circuito hace posible utilizar la fuente de alimentación que proporciona la computadora utilizada ya que en el peor de los casos (con todos los dispositivos funcionando), el consumo de energía no excede a 1 Watt.
- Los problemas presentados en la implementación del circuito en tarjeta impresa (dado sus dimensiones físicas) dió origen a situaciones no previstas en el diseño teórico debido a que estas son producto de la experiencia de la cual no se tiene mayor información escrita.
- La experiencia adquirida através del presente trabajo proporciona recomendaciones reales (dadas en el anexo 1) en los futuros trabajos de graduación en los cuales este involucrado la fabricación de circuitos impresos.

## REFERENCIAS BIBLIOGRAFICAS

- 1- SARGENT, MURRAY: THE IBM PC FROM THE INSIDE OUT;  
ADDISON WESLEY, USA, 1986.
- 2- BRENNER, ROBERT C. : THE IBM-PC TROUBLESHOOTING AND REPAIR  
GUIDE.  
MICROTRAND INC, USA 1984.
- 3- IBM - AT - REFERENCE MANUAL  
INTERNATIONAL BUSINESS  
MACHINES CORP., FLORIDA, USA 1983.
- 4- RONALD J. TOCCI: SISTEMAS DIGITALES PRINCIPIOS Y APLICACIONES  
PRENTICE HALL, USA, 1987.
- 5- THE TTL DATA BOOK, TEXAS INSTRUMENTS, USA 1982.
- 6- RCA COS/MOS INTEGRATED CIRCUITS  
USA 1980.

# CAPITULO III

## DISEÑO DE PROGRAMA PARA EL PROBADOR DE CIRCUITOS INTEGRADOS DIGITALES.

### Introducción.

Este programa cuyas siglas identificadoras son PCID (Probador de Circuitos Integrados Digitales), ha sido desarrollado en lenguaje de alto nivel, estructurado (Turbo Pascal version 4.0). El programa es sumamente sencillo en su estructura y en la forma de usarlo, asimismo su presentación de resultados es de forma gráfica para la mejor interpretación del usuario.

El programa presenta 2 características importantes en su desarrollo, en la primera se ha combinado los dos modos gráficos, el modo gráfico que se encuentra en el archivo TURBO.TPL y el modo gráfico del GRAPHIX TOOLBOX, esto es, para utilizar herramientas gráficas que no se encuentran en un solo modo; la segunda es que debido a que el programa es bastante grande se ha utilizado una herramienta que provee el Turbo Pascal llamada "Unidad" la cual es una colección de constantes, tipos de datos, variables, procedimientos y funciones los cuales pueden ser compiladas separadamente y ser invocadas en cualquier momento por el programa principal.

Además se incluye una guía del usuario que contiene aspectos sobre el diseño del programa, especialmente aquellos sobre el uso de las unidades gráficas, ya que la combinación de estas no son explicadas a fondo por las referencias bibliográficas consultadas, sin embargo estos tópicos son abarcados para los interesados en el campo de la programación gráfica; el manejo del programa es bastante sencillo y puede aprenderse en el primer contacto que se tenga con este programa ya que su estructura es pequeña en cuanto a manejo pero bastante grande en cuanto a espacio de memoria que ocupa. El propósito de este programa es de proveer un medio de medición rápida y confiable a los encargados de laboratorio o de personas encargadas de mantenimiento que están relacionados con la prueba de circuitos integrados.

### 3.1 Descripción del Problema.

El problema, como se ha descrito en los capítulos anteriores, consiste básicamente en enviar señales fuera del computador para direccionar a una circuitería la forma de proveer y recibir señales a los pines de los integrados que se deseen probar. Por supuesto el diseño de la circuitería tiene sus limitantes ya que "no" puede probarse cualquier integrado sin embargo se ha tratado de abarcar un considerable número de integrados que pueden probarse y es representado en el programa por medio de una librería.

### 3.2 Análisis del Problema.

En este capítulo se trata el problema desde el punto de vista de lenguaje de programación. El primer problema que surge es, ¿cómo poder comunicarse con el puerto del usuario?, en cuánto a lenguaje de programación existen diferentes instrucciones para hacerlo, para esta caso en particular, Turbo Pascal provee una instrucción llamada Port el cuál es un arreglo unidimensional y cada elemento representa un dato en el puerto y el índice corresponde a la dirección del puerto; cuando un elemento es asignado al componente de Port el valor es sacado por el puerto seleccionado, y cuando un componente de Port es refenciado en una expresión, su valor es introducido por el puerto seleccionado. Ejemplo de ellos tenemos:

```
Port[$3ec]:=10 (el dato dato es sacado por el puerto A).
```

```
S2:=Port[$3ee] (el valor del puerto C es introducido a S2).
```

Con estas instrucciones se pueden enviar señales que direccionan a la circuitería implementada para proveer a los integrados bajo prueba las diferentes combinaciones y así también poder recibir la respuesta a las distintas combinaciones.

El segundo problema es ¿cómo se pueden presentar los resultados obtenidos?, la mejor manera de representar cualquier tipo de resultados, es por medio de gráficos. El Turbo Pascal provee dos unidades gráficas; una es la unidad gráfica, llamada Graph que se encuentra en el archivo TURBO.TPL cuyas características principales son las de proveer de tipos de letras, elipses, círculos, cuadros, etc. La segunda unidad gráfica es llamada GRAPHIX TOOLBOX el cual provee de ventanas, y una instrucción de impresión muy importante que no posee la primera unidad. La combinación de estos dos modos gráficos provee una valiosa herramienta gráfica no sólo para este propósito, sino para cualquier representación de datos.

Debido a que la prueba de cada integrado es diferente, se simulan todas las combinaciones posibles según la aplicación que tenga el integrado (ejemplo compuertas, contadores, flip-flop, etc.) y además es representado gráficamente, esto conlleva a que el costo de memoria sea bastante elevado por cada integrado y también se pretende que el programa sea ejecutable, partiendo de estas premisas surge la siguiente pregunta, ¿cómo poder hacer para editar un programa tan grande?. La limitante que posee el Turbo Pascal es que únicamente soporta programas de 256Kbytes, esto puede depender del procesador utilizado y de la memoria instalada. Para resolver este problema Turbo Pascal provee una herramienta llamada "unidad", la cuál permite tener una colección de constantes, variables, tipos de datos, procedimientos y funciones las cuales pueden ser compiladas separadamente del programa principal, dando lugar a editar pequeños programas por

módulos e invocarlos como si ellos hubieren sido definidos en el programa principal.

### 3.3 Estructura del Programa

El programa es corto en su estructura pero grande en la cantidad de procedimientos o módulos que puede utilizar. Para una mejor comprensión del programa se presenta en la figura 3.1 un Flujograma que resume su contenido y las partes que lo componen. En el anexo se presenta el listado del programa principal.

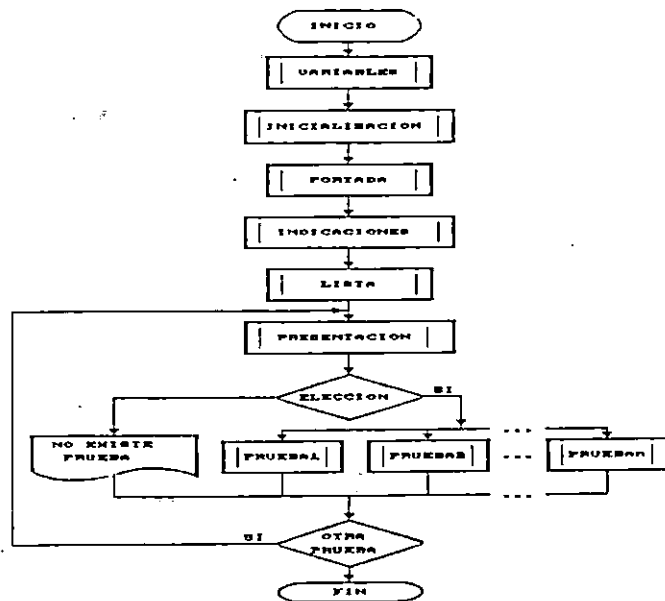


Figura 3.1 Diagrama de bloques del Programa

#### 3.3.1 Variables.

Este pequeño procedimiento comprende todas las variables, constantes y arreglos que utiliza el programa principal.

#### 3.3.2 Inicialización

Esta es una parte importante del programa ya que en este procedimiento se combinan las dos formas de inicializar los modos gráficos.

Esta inicialización comprende la selección de la tarjeta gráfica y el modo gráfico, para este caso en particular se solicita una autodetección del dispositivo por medio de la instrucción "graphdriver:=detect" ; el modo gráfico es hercomonohi el cual consiste de 720x348 pixeles (un pixel equivale a un punto en la matriz de la pantalla). Para que la detección sea satisfactoria



el código de error deberá ser igual a 0 de lo contrario presentará un mensaje de error y dará paso a una salida hacia el sistema operativo. La forma de inicializar este modo gráfico es:

```
graphdriver:=detect;
initgrapher(graphdriver,graphmode, ' ');
errorcode <> grok then
begin
  writeln('error gráfico:',grapherrormsg(errorcode));
  writeln('Programa abortado..');
  halt(1);
end.
```

La segunda inicialización es la del modo gráfico GRAPHIX TOOLBOX el cual simplemente inicia con la instrucción "Initgraphic", teniendo cuidado de usar las unidades que corresponden a éste modo gráfico (ejemplo grdriver, gkerner, etc.). Es de hacer notar que cuando se trabaja en modo gráfico y en modo texto se debe de respetar las salidas y entradas que corresponden a cada modo. Por ejemplo, cuando se trabaja en modo gráfico las instrucciones de rigor son:

"Entergraphic" (modo gráfico GRAPHIX TOOLBOX) y setgraphmode (hercmonohi) que corresponde a la unidad gráfica; de igual manera cuando se trabaja en modo texto las instrucciones que se usan son: leavegraphic y restorecrtmode respectivamente. También cuando se desea limpiar la pantalla si se está trabajando en modo texto la instrucción designada para esto es CLRSCR, en el modo gráfico CLEARDEVICE y en el GRAPHIX TOOLBOX CLEARSCREEN, sino se observan estas reglas se generan manchas en la pantalla o caracteres aleatorios dispersos. Estos ejemplos son uno de los muchos criterios que hay que considerar cuando se combinan los dos modos gráficos.

### 3.3.3 Portada e Indicaciones.

En estos procedimientos respectivamente, se encuentran el nombre del programa y las indicaciones para usarlo.

### 3.3.4 Lista.

Este procedimiento se encarga de almacenar en un vector los integrados que el programa tiene disponibles para probar, es decir la librería disponible de integrados.

### 3.3.5 Impresión.

Este módulo se encarga de presentar en la pantalla la librería de integrados que pueden probarse en una forma ordenada, para que el

usuario haga la elección del número del integrado de su interés. Esta constituido por dos procedimientos, uno es llamado "cuenta", el cual es el encargado de presentar en forma impresa en la pantalla la librería de integrados; y otra es "innumero" la cual se encarga de introducir el número que el usuario elige para probar.

### 3.3.6 Validación.

Este módulo presenta en la pantalla el número que el usuario eligió para asegurarse que es el integrado de su elección, posee dos opciones: la primera es que si no fué el integrado que eligió repita la elección del integrado, y la segunda es continuar con la prueba del integrado.

### 3.3.7 Pruebas.

Este bloque es una estructura conocida en Turbo Pascal como "CASE" la cual consiste de una expresión (selector) y una lista de sentencias, la cual es precedida por una etiqueta "CASE" del mismo tipo que el selector, cuando el valor del selector es igual a la etiqueta una sentencia será ejecutada, si ninguna de las etiquetas contine el valor de el selector, ninguna de las sentencias es ejecutada. Esta estructura es valiosa para la aplicación que se pretende en este trabajo, ya que cuando se elige un número a probar (número entero), el valor puede estar dentro del rango de las etiquetas, cuando se encuentra en este rango se ejecutarán sus respectivos procedimientos que se encuentran en las unidades correspondientes. Es decir si se elige el número 7400 debe de presionarse el número correlativo que le corresponde, este número se encuentra en un rango en el cual cualquier número contenido en este rango ejecutará las mismas sentencias o procedimientos. Para efectos de orden los procedimientos de dibujos se encuentran en una unidad llamada "dibujos", y la prueba de los integrados en una unidad llamada "pruebas". En el anexo xx aparece como está constituida una unidad y el listado para la prueba de un integrado y el programa para dibujar la configuración de este.

### 3.4 Guía del Usuario.

Las indicaciones que se dan en este apartado deberán seguirse con mucho cuidado, ya que una mala elección en la prueba de algún integrado podría causar daños es este.

El programa puede ser almacenado en disco flexible o duro según la conveniencia, el archivo ejecutable se llama PCID, el cual al correrse presentará una pantalla con el nombre del programa así:

Tal como lo dice esta pantalla el programa seguirá si se presiona la tecla <Enter>, a continuación se presenta otra pantalla con las respectivas indicaciones de su uso.

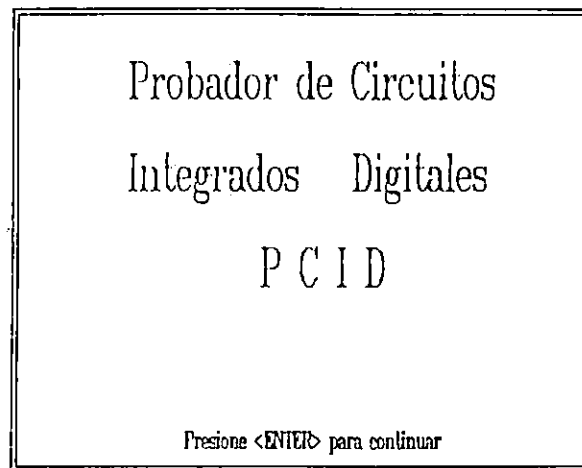


Figura 3.2 Presentación del programa.

A continuación se presenta una pantalla (ver figura 3.3) con la respectiva librería de los integrado que pueden ser probados, para seleccionar el número deseado basta con presionar las teclas en formas de flechas para desplazarse ya sea hacia arriba, abajo, izquierda o derecha. Una vez seleccionado el número se presiona <ENTER> para que comience la prueba.

1-7400	19-CD4011	37-7420	55-74F10	73-74AHC02	91-74C20
2-74F00	20-CD4093	30-74LS20	56-74H10	74-74AHC02	92-74H20
3-74H00	21-7403	39-CD4001	57-74ALS10	75-7411	93-74LS20
4-74L00	22-74H01	40-7433	58-74L10	76-74C14	94-74S20
5-74S00	23-7439	41-74LS33	59-74C10	77-74LS14	95-74F20
6-74ALS00	24-7430	42-7432	60-74HC10	78-74F14	96-74ALS20
7-74LS00	25-74LS30	43-74F32	61-7412	79-74HC14	97-74HC20
8-74C00	26-7401	44-74S32	62-74LS12	80-74AHC14	98-74L20
9-74HC00	27-74LS01	45-74LS32	63-CD4023	81-74HC14	99-7413
10-74AHC00	28-7402	46-74ALS32	64-7404	82-CD4069	100-74LS13
11-7437	29-74L02	47-74C32	65-74H04	83-CD4504	101-7440
12-74S37	30-74LS02	48-74HC32	66-74L04	84-7405	102-74H40
13-74LS37	31-74S02	49-74AHC32	67-74LS04	85-74S05	103-74LS40
14-74132	32-74F02	50-74AHC32	68-74S02	86-74LS35	104-74S40
15-74S132	33-74ALS02	51-CD4071	69-74F02	87-74H05	105-7422
16-74LS132	34-74C02	52-7410	70-74ALS02	88-74ALS05	106-74H22
17-74HC132	35-74HC02	53-74S10	71-74C02	89-74AHC05	107-74S22
18-74AHC132	36-74AHC02	54-74LS10	72-74HC02	90-7420	108-74LS22

Figura 3.3 Librería de Integrados disponibles.

Una vez introducido el número de elección aparecerá una pantalla como la figura 3.4:

En la cuál se pregunta si es el integrado que se eligió, esto es muy importante, ya que si no es el integrado que se eligió, y en la base se pone otro chip diferente podría sufrir daños. Cuando no se este seguro del integrado deberá presionarse una tecla la

cual regresará a la librería principal para hacer la elección correcta.

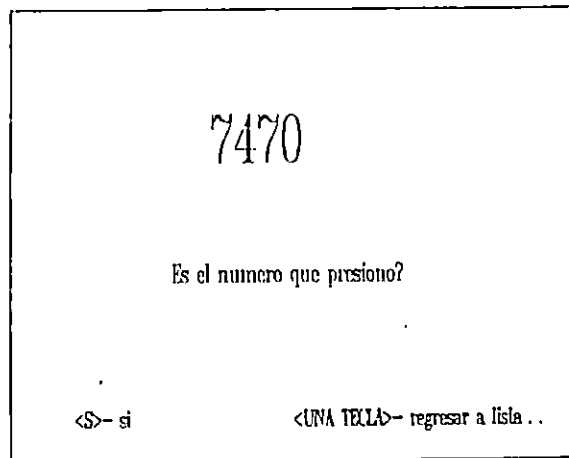


Figura 3.4 Confirmación del Integrado.

Una vez hecha la elección correcta aparecerá en la figura 3.5 la configuración del integrado y un mensaje "Introduzca el integrado a probar, presione <Enter> para comenzar", esto da tiempo para que el usuario ponga el integrado en la base respectiva para empezar la prueba. Esto se hace ya que si el integrado se coloca desde el principio podrían haber señales aleatorias en la base que podrían dañar al integrado.

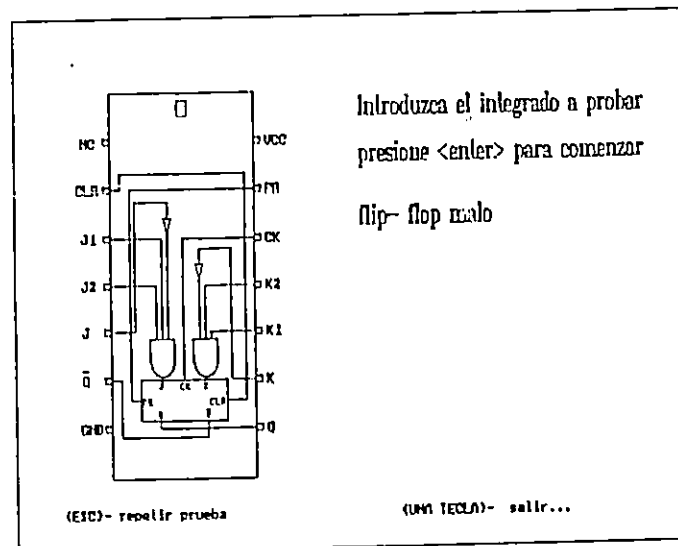


Figura 3.5 Impresión de Integrado y Resultados.

La última pantalla que aparece es la figura 3.6 en la cual indica si desea continuar probando otro integrado presionando la tecla <S> de lo contrario se presiona cualquier tecla para terminar con el programa.

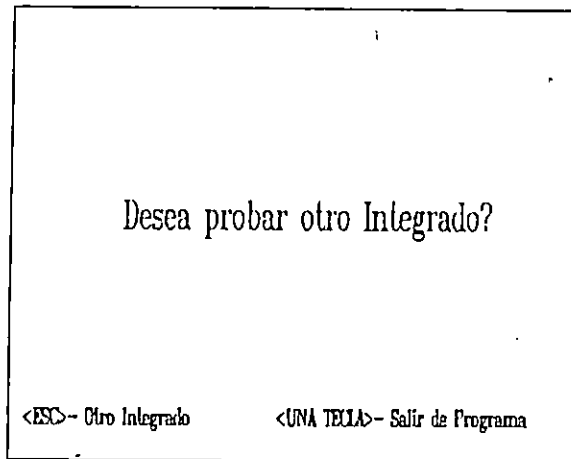


Figura 3.6 Elección de probar otro integrado.

### 3.5 Mantenimiento del Programa.

Debido a la configuración que presenta el módulo de prueba, es posible agregar nuevos dispositivos a probar. Los pasos a seguir para incluir un dispositivos específico son:

1- Hacer el programa individual del dispositivo, deseado, por ejemplo: tomando como base las palabras de control, tanto del puerto A, B y C para seleccionar el pin deseado a la señal de control elegida.

Para seleccionar el pin deseado las palabras de control son: (es de recordar que cada flip-flop controla a dos pines adyacentes).

Flip- Flop	Número en Decimal	Pines
FF1	16	1 Y 2
FF2	17	3 Y 4
FF3	18	5 Y 6
FF4	19	7 Y 8
FF5	20	9 Y 19
FF6	21	18 Y 17
FF7	22	16 Y 15
FF8	23	14 Y 13
FF9	0	12 Y 11

Una vez elegido el pin deseado, se elige la señal que se desea en ese pin; para elegir la señal deseada se escribe los números respectivos tomando en cuenta que cada flip-flop controla a dos pines adyacentes.

Pin(LSB)	Pin(MSB)	Número
0	-	1
1	-	2
lect.	-	3
-	0	8
-	1	10H
-	lect.	18H
0	0	9
0	1	11H

Pin(LSB)	Pin(MSB)	Número
0	lect.	19H
1	0	AH
1	1	12H
1	lect.	1AH
lect.	0	BH
lect.	1	13H
lect.	lect.	11H

Tómese como base el ejemplo para probar la 7400 mostrada el anexo 3. Si se desea hacer el dibujo del integrado, también se encuentra en el anexo 3 el ejemplo de la 7400.

2.- Una vez realizada la prueba para el dispositivo deseado se incluye en el programa principal. De asegurarse de que la prueba del dispositivo haya cumplido con los requerimientos deseados de la prueba, se procederá incluirlo en el programa principal mostrado en el anexo 2..

3.- Cuando sea cumplido el anterior requisito se procederá hacer los arreglos necesarios en el programa principal:

a- Redefinir el arreglo llamado "numero2" para agregar el integrado deseado.

b- Agregar en el PROCEDURE LISTA el número del integrado, en el programa actual el índice del vector "numeros" esta en 164.

c- Agregar en el PROCEDURE CUENTAL la instrucción necesaria para presentar el integrado en la pantalla .  
El PROCEDURE CUENTA, sirve para presentar en pantalla 16 números en una columna.

El PROCEDURE CUENTAL, sirve para presentar un número menor o igual a 16 números en una columna. Esto se hace porque si no se completa los 16 números se tiene la posibilidad de presentar en pantalla los que se encuentran disponibles en la última columna .

d- Ver el "Procedure Presentación" para entender la función que tienen los "Procedure Cuenta y Ventana". Ya que si se incluyen muchos nuevos dispositivos tendría que usarse una tercera pantalla.

e- Incluir en el programa principal la prueba o dibujo mostrado en la estructura CASE en una forma correlativa.

f- Compilar de nuevo el programa para renovar el programa fuente.

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

- El programa esta diseñado lo mas sencillo posible para que el usuario puede interactuar facilmente con él. Además puede ser usado por alguien que no tenga grandes conocimientos en las pruebas que se le hacen a los diferentes integrados.

- La combinación de las 2 unidades gráficas que proporciona Turbo Pascal se convierte en una poderosa herramienta gráfica para la presentación de resultados, no solo para los propósitos del presente trabajo, sino que servirá para una mejor utilización de los modos graáficos que podrán ser utilizados en futuros Trabajos de Graduación que incluyan presentación de resultados o incluso para personas interezadas en el campo de la programación.

- La impresión de resultados fué posible gracias a la combinación de los 2 modos gráficos ya que la unidad gráfica no posee esta herramienta, pero la unidad de GRAPHIX TOOLBOX sí, de lo contrario los gráficos presentados o cualquier gráfico hecho en la unidad gráfica no pudiera ser imprimido a menos que se hiziera en Lenguaje de Máquina.

- Gracias a la herramienta que proporciona Turbo Pascal llamada "unidad" es posible la implementación de este programa ya que de lo contrario fuera muy difícil la realización de este, debido a la gran cantidad de memoria utilizada. Esto proporciona bases para la implementación de programas bastante grandes.



## REFERENCIAS BIBLIOGRAFICAS

- TURBO PASCAL GRAPHIX TOOLBOX VERSION 4.0  
BORLAND, INTERNATIONAL VERSION 1987
- TURBO PASCAL OWNER'S HANDBOOK VERSION 4.0  
BORLAND, INTERNATIONAL VERSION 1987

## CONCLUSIONES GENERALES Y RECOMENDACIONES

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

- La construcción de la Interfase tratada en este documento tiene su relevancia importante en la comunicación de la computadora con el "mundo exterior". Esta puede ser usada en futuros trabajos en los cuales involucre control por medio de la Computadora. Debido a su sencillez tanto en diseño como en la forma de usarla la convierte en un dispositivo de múltiples aplicaciones. Esto proporciona un salto significativo en la comunicación entre computadoras de la nueva generación comparada con los sistemas de los procesadores del 6502 usada en años anteriores.
- En el diseño del módulo de prueba puede notarse un gran número de dispositivos esto es debido a los requerimientos establecidos al inicio de este trabajo, que por la naturaleza y variedad de las señales de prueba y de control determinan que el multiplexado de señales para cada pin es la solución mas adecuada.
- El uso de un lenguaje de alto nivel para el diseño del programa de control permite presentar los resultados en forma bastante comprensible y por ser un programa bastante extenso este lenguaje proporciona las unidades para resolver este problema. Esto ayuda a futuros trabajos de graduación o diseñadores de software en los cuales involucre utilizar todos estos beneficios del Turbo Pascal versión 4.0 para presentar los resultados de una manera más clara y eficiente.
- Es conveniente recalcar las tres grandes áreas que involucran el presente trabajo a saber: el diseño y construcción de una interfase para comunicar la computadora con el módulo de prueba, el diseño y construcción del módulo de prueba y finalmente el desarrollo del software respectivo en lenguaje de alto nivel.

## R E C O M E N D A C I O N E S

- La fabricación de circuitos impresos requiere el empleo de químicos y materiales de buena calidad, una instalación física adecuada para su construcción y herramientas idóneas. En nuestro país no se dispone de lo anterior por lo que sería recomendable que la Escuela de Ingeniería Eléctrica hiziera los contactos pertinentes a fin de obtener dichos materiales de proveedores externos.
- Es conveniente también una actualización de la tecnología usada en la fabricación de impresos usada en la Escuela de Ingeniería Eléctrica a fin de hacerla competitiva en futuro cercano.
- Impulsar la investigación en circuitos de interfases para una mejor versatilidad en la comunación por medio de la computadora y nuevas aplicaciones en el área de control en la Industria Salvadoreña.

**A N E X O 1**

**RECOMENDACIONES PARA LA IMPLEMENTACION  
DE CIRCUITOS IMPRESOS**

La implementación de un diseño electrónico en circuito impreso plantea una serie de dificultades que raras veces son consideradas en textos o trabajos de investigación. Dichas dificultades son más evidentes si el diseño en cuestión, es de dimensiones relativamente considerables o si se tiene una densidad alta de pistas por pulgada cuadrada en el impreso, si a lo anterior se agrega el problema de no contar con la calidad mínima de materiales a usar (como por ejemplo: la tableta de cobre, químicos adecuados, lugar de trabajo y condiciones ambientales no óptimas), entonces estaremos en el centro de una fuente inagotable de problemas y de interminables dolores de cabeza, antes de lograr el funcionamiento "ideal" del sistema diseñado teóricamente.

El presente trabajo y en particular su elaboración en tableta impresa no escapó a dichos problemas porque tenía las características "óptimas" para que tales contratiempos se dieran a saber:

- Dimensiones grandes (relativamente hablando 13X13 pulgadas).
- Alta densidad de pistas por pulgada cuadrada.
- Falta de cobre de buena calidad.
- Químicos en dudoso estado y no preservados adecuadamente.
- Lugar de trabajo en condiciones inapropiadas para la fabricación de circuitos impresos.
- Herramientas no adecuadas.
- etc.

Con el propósito de ayudar a que otros trabajos de graduación o proyectos superen muchos de los problemas mencionados, se dan a continuación una serie de sugerencias y recomendaciones para solventar algunos de los problemas que se encuentran a la hora de implementar a un circuito electrónico en tableta impresa.

#### Recomendaciones para la fabricación de circuitos impresos.

Las recomendaciones dadas aquí, suponen el empleo de la técnica usada en la Escuela de Ingeniería Eléctrica que involucra el diseño de las pistas en el paquete de computación Smartword, fabricación del master por proceso fotomecánico y finalmente el uso de los químicos y equipo del laboratorio de circuitos impresos de la Escuela de Ingeniería Eléctrica.

- El diseño deberá hacerse, de tal forma que los circuitos integrados a usar queden orientados en la misma dirección, esto facilita las modificaciones o reparaciones futuras.
- Es recomendable que las líneas que proveen la alimentación de energía, VCC y GND, recorran la menor distancia posible para reducir la posible inducción de señales que pueda darse. Esto implica que si el circuito diseñado posee un conector donde se alimenta externamente, que las líneas VCC y GND lleguen a los

IC de la forma más directa posible.

- Las líneas de alimentación (VCC y GND) deberán hacerse más gruesas que las demás líneas a fin de aumentar el área de conductividad que ellas tengan.
- El paquete Smartword posee dos tipos de líneas una gruesa y una delgada, sin embargo comercialmente en nuestro medio es difícil encontrar cobre de buena calidad, por lo que las líneas delgadas son difíciles de reproducir en el cobre, por lo que se recomienda que cuando se obtenga la impresión del smartword llamada "2X ARTWORK" las líneas sean engrosadas lo más posible para asegurar su reproducción en el cobre, Esto puede hacerse con tinta china.
- Los espacios que no se utilicen en la tarjeta pueden "llenarse" con cobre, para formar un plano de tierra amplio, a fin de poder realizar un blindaje del circuito si fuera necesario.
- Es recomendable conectar capacitores de cerámica entre 0.01 $\mu$ F y 0.1 $\mu$ F entre VCC y GND, a fin de contrarrestar inductancias parásitas en el circuito.
- Asimismo es buena idea colocar un capacitor de tantalio entre 10 y 30  $\mu$ F entre VCC y GND en la entrada del conector de alimentación.
- Si se utiliza alambres en el circuito, asegúrese de que tengan la menor longitud posible, es decir que viajen lo más cerca de la tarjeta y si se utiliza varios alambres, entonces lo recomendable es que viajen trenzados y de ser posible blindados y conectar el blindaje a la tierra del sistema.
- Se recomienda que al la hora de perforar los agujeros en el impreso, se utilice un taladro de alta revoluciones (25000 rpm) para evitar arrancar el cobre de las pistas.

A N E X O 2

LISTADO DEL PROGRAMA PRINCIPAL

```

Program PCID;
uses
dos,crt,printer,graph,gdriver,gkernel,gwindow,dibujos,
pruebas;
const
    esc = #27;
type
    numero2 = array[1..164] of string[10];
var b,x,y,z,i,h,salida,iocode:byte;
eleccion,a,gdriver,graphmode,errorcode,result:integer;
    si,ch:char;
    temp:wrkstring;
    numero:numero2;
{*****}
Procedure inicio;
begin
    initgraphic;
    gdriver:=detect;
    initgraph(gdriver,graphmode,'');
    errorcode:=graphresult;
    if errorcode <> gok then
        begin
            writeln('graphics
error:',grapherrormsg(errorcode));
            writeln('Programa abortado..');
            halt(1);
        end;
    end;
{*****}
Procedure portada;
begin
    rectangle(0,0,700,343);
    rectangle(6,4,694,339);
    settextstyle(1,0,5);
    outtextxy(140,30,'Probador de Circuitos');
    outtextxy(140,100,'Integrados Digitales');
    outtextxy(300,170,'P C I D');
    settextstyle(1,0,1);
    outtextxy(210,310,'Presione <ENTER> para continuar');
    readln;
end;
{*****}
Procedure lista;
begin
cleardevice;
clearscreen;
    for i:=1 to 153 do
        numero[i]:='';
end;

```



```

(inicio de nand de 2) numero[11]:='7400'; numero[12]:='74F00';
numero[13]:='74H00'; numero[14]:='74L00'; numero[15]:='74S00';
numero[16]:='74ALS00'; numero[17]:='74LS00'; numero[18]:='74C00';
numero[19]:='74HC00'; numero[10]:='74AHC00'; numero[11]:='7437';
numero[12]:='74S37'; numero[13]:='74LS37';
numero[14]:='74132';
numero[15]:='74S132'; numero[16]:='74LS132';
numero[17]:='74HC132';
numero[18]:='74HCT132';
(inicio de nand cmos 2) numero[19]:='CD4011'; numero[20]:=
'CD4093';
(i. nand 2 o.c.) numero[21]:='7403'; numero[22]:='74H01';
numero[23]:='7439';
(i. nand 2 o.c.) numero[24]:='7438'; numero[25]:='74LS38';
(i. nand 2 o.c.) numero[26]:='7401'; numero[27]:='74LS01';
{nor 2 normal} numero[28]:='7402'; numero[29]:='74L02';
numero[30]:='74LS02';
numero[31]:='74S02'; numero[32]:='74F02';
numero[33]:='74ALS02';
numero[34]:='74C02'; numero[35]:='74HC02';
numero[36]:='74AHCT02';
numero[37]:='7428'; numero[38]:='74LS28';
{nor 2 cmos} numero[39]:='CD4001';
{nor 2 o.c.} numero[40]:='7433'; numero[41]:='74LS33';
{or de 2} numero[42]:='7432'; numero[43]:='74F32';
numero[44]:='74S32';
numero[45]:='74LS32'; numero[46]:='74ALS32';
numero[47]:='74C32';
numero[48]:='74HC32'; numero[49]:='74AHCT32';
numero[50]:='74HCT32';
{or de 2 cmos} numero[51]:='CD4071';
{nand 3 normal} numero[52]:='7410'; numero[53]:='74S10';
numero[54]:='74LS10';
numero[55]:='74F10'; numero[56]:='74H10';
numero[57]:='74ALS10';
numero[58]:='74L10'; numero[59]:='74C10';
numero[60]:='74HC10';
{nand 3 o.c.} numero[61]:='7412'; numero[62]:='74LS12';
{nand 3 cmos} numero[63]:='CD4023';
{inversores normal} numero[64]:='7404'; numero[65]:='74H04';
numero[66]:='74L04'; numero[67]:='74LS04';
numero[68]:='74S02';
numero[69]:='74F02'; numero[70]:='74ALS02';
numero[71]:='74C02';
numero[72]:='74HC02'; numero[73]:='74AHC02';
numero[74]:='74HCT02';
numero[75]:='7414'; numero[76]:='74C14';
numero[77]:='74LS14';
numero[78]:='74F14'; numero[79]:='74HC14';
numero[80]:='74AHCT14';
numero[81]:='74HC14'; numero[82]:='CD4069';

```

```

numero[83]:='CD4584';
{inversores o.c.} numero[84]:='7405'; numero[85]:='74805';
numero[86]:='74LS05'; numero[87]:='74H05';
numero[88]:='74ALS05';
numero[89]:='74HCT05';
{nand 4 normal} numero[90]:='7420'; numero[91]:='74C20';
numero[92]:='74H20';
numero[93]:='74LS20'; numero[94]:='74820';
numero[95]:='74F20';
numero[96]:='PROX. PAG.'; numero[97]:='PAG. ANTER.';
numero[98]:='74ALS20'; numero[99]:='74HC20';
numero[100]:='74L20';
numero[101]:='7413'; numero[102]:='74LS13';
numero[103]:='7440';
numero[104]:='74H40'; numero[105]:='74LS40';
numero[106]:='74S40';
{nand 4 o.c.} numero[107]:='7422'; numero[108]:='74H22';
numero[109]:='74S22'; numero[110]:='74LS22';
{nand 4 cmos} numero[111]:='CD4012';
{and 2 normal} numero[112]:='7408'; numero[113]:='74LS08';
numero[114]:='74F08'; numero[115]:='74H08';
numero[116]:='74L08';
numero[117]:='74S08'; numero[118]:='74ALS08';
numero[119]:='74C08';
numero[120]:='74HC08'; numero[121]:='74AHCT08';
numero[122]:='74HCT08';
{and 2 o.c.} numero[123]:='7409'; numero[124]:='74S09';
numero[125]:='74LS09'; numero[126]:='74ALS09';
{and 2 cmos} numero[127]:='CD4081';
{and 3 normal} numero[128]:='7411'; numero[129]:='74H11';
numero[130]:='74S11'; numero[131]:='74ALS11';
numero[132]:='74LS11';
numero[133]:='74HC11';
{and 3 o.c.} numero[134]:='74H15'; numero[135]:='74S15';
numero[136]:='74LS15';
{and 3 cmos} numero[137]:='CD4073';
{and 4 normal} numero[138]:='7421'; numero[139]:='74H21';
numero[140]:='74L21'; numero[141]:='74HC21';
numero[142]:='74LS21';
numero[143]:='74ALS21';
{and 4 cmos} numero[144]:='CD4082';
{nor 3 normal} numero[145]:='7427'; numero[146]:='74ALS27';
numero[147]:='74LS27'; numero[148]:='74HC21';
{nor 3 cmos} numero[149]:='CD4025';
{nand 8 normal} numero[150]:='7430'; numero[151]:='74L30';
numero[152]:='74LS30'; numero[153]:='74ALS30';
numero[154]:='74HC30';
{nand 8 cmos} numero[155]:='CD4068';
numero[156]:='7442'; numero[157]:='7443';
numero[158]:='7444';
numero[159]:='7478'; numero[160]:='7470';

```

```

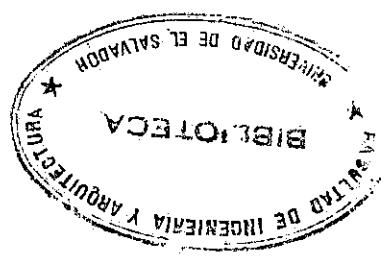
numero[161]:= '7472';
numero[162]:= '7474'; numero[163]:= '74107';
numero[164]:= '74110'end;
{*****}

```

```

procedure cuenta(z:integer);
begin
  i:=i+1; outtextxy(z,1,+numero[i]);
  i:=i+1; outtextxy(z,20,+numero[i]);
  i:=i+1; outtextxy(z,39,+numero[i]);
  i:=i+1; outtextxy(z,58,+numero[i]);
  i:=i+1; outtextxy(z,77,+numero[i]);
  i:=i+1; outtextxy(z,96,+numero[i]);
  i:=i+1; outtextxy(z,115,+numero[i]);
  i:=i+1; outtextxy(z,134,+numero[i]);
  i:=i+1; outtextxy(z,153,+numero[i]);
  i:=i+1; outtextxy(z,172,+numero[i]);
  i:=i+1; outtextxy(z,191,+numero[i]);
  i:=i+1; outtextxy(z,210,+numero[i]);
  i:=i+1; outtextxy(z,229,+numero[i]);
  i:=i+1; outtextxy(z,248,+numero[i]);
  i:=i+1; outtextxy(z,267,+numero[i]);
  i:=i+1; outtextxy(z,286,+numero[i]);

```



```

end;
{*****}

```

```

procedure cuenta1(z:integer);
begin
  i:=i+1; outtextxy(z,1,+numero[i]);
  i:=i+1; outtextxy(z,20,+numero[i]);
  i:=i+1; outtextxy(z,39,+numero[i]);
  i:=i+1; outtextxy(z,58,+numero[i]);

```

```

end;
{*****}

```

```

procedure ventana(var opcion:integer;k:integer);
const null=#0;enter=#13;
var i,j,fil,col,hor,ver,x,y:integer;
    ch:char;
    vector:array[1..6,1..16] of integer;
begin
  rectangle(0,0,680,304);
  gotoxy(1,24); write(char(24),char(25),'- Mov. Vert. ');
  settextstyle(0,0,1);
  outtextxy(160,322,+char(26));
  outtextxy(160,329,+char(27));
  gotoxy(20,24); write('- Mov. Horz. ');
  gotoxy(38,24); write('Enter- Eleccion');
  copyscreen;
  hor:=trunc(XMaxGlb/6);ver:=trunc(YMaxGlb/18);
  definewindow(1,0,0,hor,ver);selectwindow(1);

```

```

defineworld(1,0,0,10,10);selectworld(1);
invertwindow;
fil:=1;col:=1;x:=0;y:=0;
repeat
  ch:=readkey;
  if (ch=null) and keypressed then ch:=readkey;
  case ch of
    'A','H': begin
      if (fil=1) and (col=1) then movehor(0,true) else
      if (fil=1) and (col<>1) then begin invertwindow;
      col:=col-1;fil:=16;
      redefinewindow(1,x-hor,y+15*ver,x,y+16*ver);
      x:=x-hor;y:=y+15*ver;
      selectwindow(1);invertwindow end
      else begin
      invertwindow;fil:=fil-1;
      redefinewindow(1,x,y-ver,x+hor,y);
      y:=y-ver;selectwindow(1);invertwindow;end;
      end;
    'D','K': begin
      if col=1 then movehor(0,true)
      else begin col:=col-1;invertwindow;
      redefinewindow(1,x-hor,y,x,y+ver);
      x:=x-hor;selectwindow(1);invertwindow; end;
      end;
    'C','M': begin
      if col=6 then movehor(0,true)
      else begin col:=col+1;invertwindow;
      redefinewindow(1,x+hor,y,x+2*hor,y+ver);
      selectwindow(1);
      x:=x+hor;invertwindow; end;
      end;
    'B','P': begin
      if (fil=16) and (col=6) then movehor(0,true)
      else
      if (fil=16) and (col<>6) then begin invertwindow;
      col:=col+1;fil:=1;
      redefinewindow(1,x+hor,0,x+2*hor,ver);
      selectwindow(1);invertwindow;x:=x+hor;y:=0 end
      else begin
      fil:=fil+1;invertwindow;
      redefinewindow(1,x,y+ver,x+hor,y+2*ver);
      selectwindow(1);invertwindow;y:=y+ver end;
      end;
      end;
  until ch=enter;
  for i:=1 to 6 do
  for j:=1 to 16 do begin vector[i,j]:=k;k:=k+1;end;
  eleccion:=vect [col,fil];
  clearscreen;
  cleardevice; end;

```

```

(*****)
Procedure presentacion;
begin
repeat
cleardevice;
clearscreen;
i:=0;
setttextstyle(2,0,6);
cuanta(20);
cuanta(135);
cuanta(244);
cuanta(359);
cuanta(474);
cuanta(584);
ventana(eleccion,1);
if eleccion = 96 then
begin
cleardevice;
clearscreen;
setttextstyle(2,0,6);
cuanta(15);
cuanta(135);
cuanta(244);
cuanta(359);
cuanta(474);
ventana(eleccion,97);
end
else salida:=1;
if eleccion = 97 then salida:=0
else salida:=1;
until salida = 1;
end;
(*****inicio*****)
begin{inicio de programa principal}
inicio;
portada;
lista;
repeat
repeat
presentacion;
cleardevice;
clearscreen;
setttextstyle(1,0,6);
definewindow(1,1,1,XmaxGlb-2,YmaxGlb-5);
defineworld(1,0,0,1000,1000);
selectwindow(1);
outtextxy(250,70,+numero[eleccion]);
setttextstyle(1,0,2);
outtextxy(200,190,'Es el numero que presiono?');
setttextstyle(1,0,1);
outtextxy(80,300,'<S>- si');

```

```

outtextxy(350,300,<UNA TECLA>-ragresar a lista . . .);
invertwindow;
si:=readkey;
until ('a' = si) or ('S' = si);
case selection of
1..18: begin
repeat
clearscreen;
cleardevice;
delay(100);
dibujoi;
prueba1;
ch:=readkey;
until ch <> esc;
end;
28..38:begin
repeat
clearscreen;
cleardevice;
dibujoi;
prueba2;
ch:=readkey;
until ch <> esc;
end;
40..41:begin
repeat
clearscreen;
cleardevice;
dibujoi;
prueba19;
ch:=readkey;
until ch <> esc;
end;
52..60:begin
repeat
clearscreen;
cleardevice;
dibujoi;
prueba4;
ch:=readkey;
until ch <> esc;
end;
64..83:begin
repeat
clearscreen;
cleardevice;
dibujoi;
prueba3;
ch:=readkey;
end;
end;

```

```

until ch <> esc;
end;
84..89:begin
repeat
clearscreen;
cleardevice;
dibuf03;
prueba20;
ch:=readkey;
until ch <> esc;
end;
90..104:begin
repeat
clearscreen;
cleardevice;
dibuf05;
prueba25;
ch:=readkey;
until ch <> esc;
end;
112..122:begin
repeat
clearscreen;
cleardevice;
dibuf07;
prueba17;
ch:=readkey;
until ch <> esc;
end;
123..126:begin
repeat
clearscreen;
cleardevice;
dibuf01;
prueba11;
ch:=readkey;
until ch <> esc;
end;
148..152:begin
repeat
clearscreen;
cleardevice;
dibuf06;
prueba6;
ch:=readkey;
until ch <> esc;
end;
156
:begin
repeat
clearscreen;
cleardevice;

```

```

162      :begin
      repeat
      clearscreen;
      cleardevice;
      dibujoj2;
      pruebaj3;
      ch:=readkey;
      until ch <> esc;
      end;

```

```

161      :begin
      repeat
      clearscreen;
      cleardevice;
      dibujoj;
      pruebaj2;
      ch:=readkey;
      until ch <> esc;
      end;

```

```

160      :begin
      repeat
      clearscreen;
      cleardevice;
      dibujoj;
      pruebaj1;
      ch:=readkey;
      until ch <> esc;
      end;

```

```

158      :begin
      repeat
      clearscreen;
      cleardevice;
      dibujoj7;
      pruebaj;
      ch:=readkey;
      until ch <> esc;
      end;

```

```

157      :begin
      repeat
      clearscreen;
      cleardevice;
      dibujoj7;
      pruebaj;
      ch:=readkey;
      until ch <> esc;
      end;

```



```

:begin 163
repeat
clearscreen
display
prueba14
ch:=readkey
until ch <> esc
end
:begin 164
repeat
clearscreen
display
prueba14
ch:=readkey
until ch <> esc
end
else
:begin
inicio
clearscreen
cleardevice
rectangle(1,1,700,343);
settextstyle(1,0,2);
outtextxy(130,130,'El numero que eligio no se
encuentra');
outtextxy(170,190,'Corra de nuevo el programa');
repeat until keypressed;
leavegraphic;
halt(1);
end;
end;
clearscreen;
cleardevice;
rectangle(1,1,700,343);
settextstyle(1,0,4);
outtextxy(1,1,700,343);
outtextxy(140,140,'Desea probar otro Integrador?');
settextstyle(1,0,1);
outtextxy(20,300,'<ESC>- Otro Integrador');
outtextxy(330,300,'<UNA TECLA>- Salir de Programa');
repeat until keypressed;
set:=readkey;
until si <> esc;
closegraph;
leavegraphic;
end"

```

## A N E X O 3

### EJEMPLO DEL LISTADO DE LA UNIDAD

#### PRUEBAS Y DIBUJOS

Este es un ejemplo del contenido de las unidades en las cuales estan contenidos las pruebas de los integrados y los dibujos de cada uno de ellos.

Debido a lo extenso del programa no se presenta en su totalidad, pero el programa fuente con todo su listado se encuentra en Disco Flexible en la Escuela de Ingeniería Eléctrica.

EJEMPLO DE LA UNIDAD QUE CONTIENE LA PRUEBA DE CADA UNO DE  
 LOS  
 CIRCUITOS INTEGRADOS CONTENIDOS EN LA UNIDAD PRUEBAS.  
 ESTE PROGRAMA PRUEBA EL IC 7400.

```

Unit pruebas;
Interface
  uses crt,graph,gkernel,gdriver;
  Procedure 7400;
Implementation;
Procedure 7400;
var a,c,b,s1,s2,s3,s4,i:byte;
Procedure libre;
begin
Port[#3ec]:=0;{inicialización de los multiplexer al canal 0}
for i:=16 to 23 do
Port[#3ed]:=i;
Port[#3ed]:=7;
for i:=0 to 3 do
Port[#3ed]:=i;
Port[#3ed]:=7;{fin de iniciliazación}
Port[#3ec]:=0;
Port[#3ec]:=128;
Port[#3ec]:=0;
end;
Procedure inicial;{inicialiaza los integrados de 14 pines}
begin
Port[#3ec]:=0;
for i:=16 to 18 do
port[#3ed]:=i;
for i:=20 to 23 do
port[#3ed]:=i;
Port[#3ed]:=7;
end;
Procedure tierra;{pone tierra para los integrados de 14
pines}
begin
Port[#3ec]:=1;
Port[#3ec]:=129;
port[#3ec]:=0;
end;
Procedure poner;
begin
setttextstyle(1,0,1);
outtextxy(360,50,'Introduzca el integrado a probar');
outtextxy(360,80,'presione <enter> para c6menzar');
readln;
end;
Procedure salida;
begin
setttextstyle(0,0,1);

```

```

outtextxy(50,320,'<ESC>- repetir prueba');
outtextxy(400,320,'<UNA TECLA>- salir...');
end;
Procedure pini23;
begin
Port[#3ed]:=16; {habilita el FF1}
Port[#3ed]:=7; {pivote}
Port[#3ec]:=3; {habilita el canal 4 para leer}
Port[#3ed]:=17; {habilita el FF2}
Port[#3ed]:=7; {pivote}
end;
Procedure pin987;
begin
Port[#3ed]:=22; {habilita el FF7}
Port[#3ed]:=7; {pivote}
Port[#3ec]:=3; {habilita el canal 4 para leer}
Port[#3ed]:=23; {habilita el FF8}
Port[#3ed]:=7; {pivote}
end;
Procedure compuerta;
begin
Port[#3ec]:=8; {pone 0 en el pin 4 y nc en pin3}
Port[#3ed]:=a; {habilita el FF2}
Port[#3ed]:=7; {pivote}
Port[#3ec]:=#19; {pone 0 en el pin 5 y lee en pin 6}
Port[#3ed]:=b; {habilita el FF3}
Port[#3ed]:=7; {pivote}
delay(10);
s1:=Port[#3ed]; {lee por el puerto C la salida}
Port[#3ec]:=#10; {pone 1 en el pin 4}
Port[#3ed]:=a; {habilita el FF2}
Port[#3ed]:=7; {pivote}
Port[#3ec]:=#19; {pone 0 en el pin 5 y lee en pin 6}
Port[#3ed]:=b; {habilita el FF3}
Port[#3ed]:=7; {pivote}
delay(10);
s2:=Port[#3ed]; {lee por el puerto C la salida}
Port[#3ec]:=8; {pone 0 en el pin 4}
Port[#3ed]:=a; {habilita el FF2}
Port[#3ed]:=7; {pivote}
Port[#3ec]:=#1a; {pone 1 en el pin 5 y lee en pin 6}
Port[#3ed]:=b; {habilita el FF3}
Port[#3ed]:=7; {pivote}
delay(10);
s3:=Port[#3ed]; {lee por el puerto C la salida}
Port[#3ec]:=#10; {pone 1 en el pin 4}
Port[#3ed]:=a; {habilita el FF2}
Port[#3ed]:=7; {pivote}
Port[#3ec]:=#1a; {pone 1 en el pin 5 y lee en pin 6}
Port[#3ed]:=b; {habilita el FF3}
Port[#3ed]:=7; {pivote}

```

```

delay(10);
s4:=PortI[#3eeI];(lee por el puerto C la salida)
end;
Procedure decision;
begin
if (s1=1) and (s2=1) and (s3=1) and (s4=0)
then begin
settextstyle(1,0,1); outtextxy(360,z,'compuerta '+char(c)+'
buena');
end
else
begin
settextstyle(1,0,1); outtextxy(360,z,'compuerta '+char(c)+'
mala');
end;
end;
(*****)
begin(inicio de 7400)
PortI[#3efI]:=#9I;(configuración de la interfase)
libre;
delay(1);
tierraI;
delay(2);
poner;
(inicio de compuerta 1)
PortI[#3ecI]:=#9;(pone 0,0 en el pin 1 y 2)
pin123;
delay(10);
s1:=PortI[#3eeI];(lee por el puerto C la salida)
PortI[#3ecI]:=#a;(pone 0,1 en el pin 1 y 2)
pin123;
delay(10);
s2:=PortI[#3eeI];(lee por el puerto C la salida)
PortI[#3ecI]:=#11;(pone 1,0 en el pin 1 y 2)
pin123;
delay(10);
s3:=PortI[#3eeI];(lee por el puerto C la salida)
PortI[#3ecI]:=#12;(pone 1,1 en el pin 1 y 2)
pin123;
delay(10);
s4:=PortI[#3eeI];(lee por el puerto C la salida)
c:=49;
z:=110;
decision;
(fin de compuerta 1)
iniciol;
a:=17;(inicio de compuerta 2)
b:=18;
compuerta;
c:=50;
z:=130;

```

```

decision;
{fin de compuerta 2}
iniciol;
{inicio de compuerta 3}
Port[#3ec]:=#9;{pone 0,0 en los pines 16,15}
pin987;
delay(10);
s1:=Port[#3ee];{lee por el puerto C la salida}
Port[#3ec]:=#a;{pone 1,0 en los pines 16,15}
pin987;
delay(10);
s2:=Port[#3ee];{lee por el puerto C la salida}
Port[#3ec]:=#11;{pone 0,1 en los pines 16,15}
pin987;
delay(10);
s3:=Port[#3ee];{lee por el puerto C la salida}
Port[#3ec]:=#12;{pone 1,1 en los pines 16,15}
pin987;
delay(10);
s4:=Port[#3ee];{lee por el puerto C la salida}
c:=51;
z:=150;
decision;
{fin de compuerta 3}
iniciol;
{inicio de compuerta 4}
a:=20;
b:=21;
compuerta;
c:=52;
z:=170;
decision;
salida;
end;

```

EJEMPLO DE LA UNIDAD DIBUJOS QUE CONTIENE LOS DIBUJOS DE LOS  
 DIFERENTES INTEGRADOS QUE SON PROBADOS.  
 ESTE PROGRAMA HACE EL DIBUJO DE LA 7400.

```

Unit dibujos;
Interfacs;
uses crt,graph,kernel,gdriver;
Procedure dibujos;
Implementation
Procedure dibujos;
(este dibujo es para una 7400 o compatibles)
var a,i,x,y:integer;
Procedure cuadro;
var i,x,a:byte;
begin
rectangle(1,700,347); (dibuja el emarcado)
rectangle(100,50,250,300);(dibuja el cuadro del integrado)
i:=48; (pivote Ymin)
x:=52; (pivote Ymax)
for a:=1 to 6 do
begin
i:=i+31;
x:=x+31;
delay(80);
rectangle(93,i,100,x); (hace los cuadros de los pines
izquierdos)
end;
rectangle(93,265,100,269);
end;
i:=48;
x:=52;
for a:=1 to 6 do
begin
i:=i+31;
x:=x+31;
delay(80);
rectangle(250,i,257,x); (hace los cuadros de los pines
derechos)
end;
rectangle(250,265,257,269);
end;
rectangle(170,55,180,65); (hace el cuadro de identificacion)
end;
begin
cuadro;
delay(80);
settextstyle(0,0,1);
rectangle(110,74,150,119); (hace el primer cuadro izquierdo)
outtextxy(130,98,1); (pone uno el primer cuadro izquierdo)
delay(80);

```

```

rectangle(110,167,150,212); {segundo cuadro izquierdo}
outtextxy(130,186,'2');
delay(80);
rectangle(200,198,240,243); {tercer cuadro derecho}
outtextxy(220,221,'3');
delay(80);
rectangle(200,105,240,150); {cuarto cuadro derecho}
outtextxy(220,126,'4');
delay(80);
line(100,81,110,81); {pone raya de arriba de primer cuadro}
delay(50);
line(100,112,110,112); {pone raya de abajo de primer cuadro}
delay(50);
line(100,143,130,143); {pone raya de respuesta de primer
cuadro}
delay(80);
line(100,174,110,174); {pone raya de arriba de segundo
cuadro}
delay(50);
line(100,205,110,205); {pone raya de abajo de segundo cuadro}
delay(50);
line(100,236,130,236); {pone raya de respuesta de segundo
cuadro}
delay(80);
line(220,267,250,267); {pone raya de respuesta de tercer
cuadro}
delay(50);
line(240,236,250,236); {pone raya de abajo de tercer cuadro}
delay(50);
line(240,205,250,205); {pone raya de arriba de tercer cuadro}
delay(80);
line(220,174,250,174); {pone raya de respuesta de cuarto
cuadro}
delay(50);
line(240,143,250,143); {pone raya de abajo de cuarto cuadro}
delay(50);
line(240,112,250,112); {pone raya de arriba de cuarto cuadro}
delay(80);
line(130,119,130,143); {raya vertical de respuesta de primer
cuadro}
delay(50);
line(130,212,130,236); {raya vertical de respuesta de segundo
cuadro}
delay(50);
line(220,243,220,267); {raya vertical de respuesta de tercer
cuadro}
delay(50);
line(220,150,220,174); {raya vertical de respuesta de cuarto
cuadro}
delay(80);
outtextxy(70,79,'1A'); {pone numeracion a pines del

```



```
integrado}
delay(80);
outtextxy(70,109,'1B');
delay(80);
outtextxy(70,140,'1Y');
delay(80);
outtextxy(70,171,'2A');
delay(80);
outtextxy(70,202,'2B');
delay(80);
outtextxy(70,233,'2Y');
delay(80);
outtextxy(68,264,'GND');
delay(80);
outtextxy(262,264,'3Y');
delay(80);
outtextxy(262,233,'3A');
delay(80);
outtextxy(262,202,'3B');
delay(80);
outtextxy(262,171,'4Y');
delay(80);
outtextxy(262,140,'4A');
delay(80);
outtextxy(262,109,'4B');
delay(80);
outtextxy(262,78,'VCC');
end;
```

A N E X O 4

ESPECIFICACIONES DE LOS COMPONENTES UTILIZADOS



8255A/8255A-5

**CAPACITANCE**  $T_A = 25^\circ\text{C}$ ,  $V_{CC} = \text{GND} = 0\text{V}$

Symbol	Parameter	Min	Typ	Max	Unit	Test Conditions
$C_{IN}$	Input Capacitance			10	pF	$f_c = 1 \text{ MHz}$
$C_{I/O}$	I/O Capacitance			20	pF	Unmeasured pins returned to GND

**A.C. CHARACTERISTICS**  $T_A = 0^\circ\text{C}$  to  $70^\circ\text{C}$ ,  $V_{CC} = +5\text{V} \pm 10\%$ ,  $\text{GND} = 0\text{V}$

**Bus Parameters**

**READ**

Symbol	Parameter	8255A		8255A-5		Unit
		Min	Max	Min	Max	
$t_{AR}$	Address Stable before READ	0		0		ns
$t_{RA}$	Address Stable after READ	0		0		ns
$t_{RR}$	READ Pulse Width	300		300		ns
$t_{RD}$	Data Valid from READ(1)		250		200	ns
$t_{DF}$	Data Float after READ	10	150	10	100	ns
$t_{RV}$	Time between READs and/or WRITEs	850		850		ns

**WRITE**

Symbol	Parameter	8255A		8255A-5		Unit
		Min	Max	Min	Max	
$t_{AW}$	Address Stable before WRITE	0		0		ns
$t_{WA}$	Address Stable after WRITE	20		20		ns
$t_{WW}$	WRITE Pulse Width	400		300		ns
$t_{DW}$	Data Valid to WRITE (T.E.)	100		100		ns
$t_{WD}$	Data Valid after WRITE	30		30		ns

**OTHER TIMINGS**

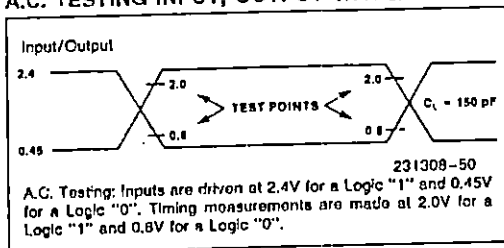
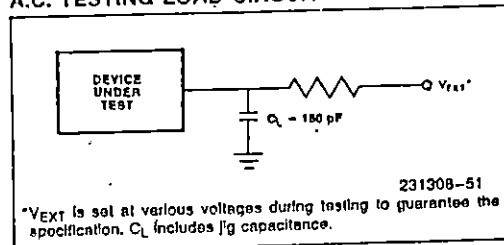
Symbol	Parameter	8255A		8255A-5		Unit
		Min	Max	Min	Max	
$t_{WB}$	WR = 1 to Output(1)		350		350	ns
$t_{PR}$	Peripheral Data before RD	0		0		ns
$t_{PR}$	Peripheral Data after RD	0		0		ns
$t_{AK}$	ACK Pulse Width	300		300		ns
$t_{ST}$	STB Pulse Width	500		500		ns
$t_{PS}$	Per. Data before T.E. of STB	0		0		ns
$t_{PH}$	Per. Data after T.E. of STB	180		180		ns
$t_{AD}$	ACK = 0 to Output(1)		300		300	ns
$t_{KD}$	ACK = 1 to Output Float	20	250	20	250	ns

**A.C. CHARACTERISTICS (Continued)**
**OTHER TIMINGS (Continued)**

Symbol	Parameter	8255A		8255A-5		Unit
		Min	Max	Min	Max	
$t_{WOB}$	WR = 1 to OBF = 0 <sup>(1)</sup>		650		650	ns
$t_{AOB}$	ACK = 0 to OBF = 1 <sup>(1)</sup>		350		350	ns
$t_{SIB}$	STB = 0 to IBF = 1 <sup>(1)</sup>		300		300	ns
$t_{RIB}$	RD = 1 to IBF = 0 <sup>(1)</sup>		300		300	ns
$t_{RIT}$	RD = 0 to INTR = 0 <sup>(1)</sup>		400		400	ns
$t_{SIT}$	STB = 1 to INTR = 1 <sup>(1)</sup>		300		300	ns
$t_{AIT}$	ACK = 1 to INTR = 1 <sup>(1)</sup>		350		350	ns
$t_{WIT}$	WR = 0 to INTR = 0 <sup>(1,3)</sup>		450		450	ns

**NOTES:**

- Test Conditions:  $C_L = 150$  pF.
  - Period of Reset pulse must be at least 50  $\mu$ s during or after power on. Subsequent Reset pulse can be 500 ns min.
  - INTR  $\uparrow$  may occur as early as WR  $\downarrow$ .
- \*For Extended Temperature EXPRESS, use M8255A electrical parameters.

**A.C. TESTING INPUT, OUTPUT WAVEFORM**

**A.C. TESTING LOAD CIRCUIT**

**ABSOLUTE MAXIMUM RATINGS\***

Ambient Temperature Under Bias ..... 0°C to 70°C  
 Storage Temperature ..... -65°C to +150°C  
 Voltage on Any Pin  
 with Respect to Ground ..... -0.5V to +7V  
 Power Dissipation ..... 1 Watt

\*Notice: 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. CHARACTERISTICS**  $T_A = 0^\circ\text{C to } 70^\circ\text{C}$ ,  $V_{CC} = +5V \pm 10\%$ ,  $GND = 0V$ 

Symbol	Parameter	Min	Max	Unit	Test Conditions
$V_{IL}$	Input Low Voltage	-0.5	0.8	V	
$V_{IH}$	Input High Voltage	2.0	$V_{CC}$	V	
$V_{OL (DB)}$	Output Low Voltage (Data Bus)		0.45*	V	$I_{OL} = 2.5$ mA
$V_{OL (PER)}$	Output Low Voltage (Peripheral Port)		0.45*	V	$I_{OL} = 1.7$ mA
$V_{OH (DB)}$	Output High Voltage (Data Bus)	2.4		V	$I_{OH} = -400$ $\mu$ A
$V_{OH (PER)}$	Output High Voltage (Peripheral Port)	2.4		V	$I_{OH} = -200$ $\mu$ A
$I_{DAR}^{(1)}$	Darlington Drive Current	-1.0	-4.0	mA	$R_{EXT} = 750\Omega$ ; $V_{EXT} = 1.5V$
$I_{CC}$	Power Supply Current		120	mA	
$I_{IL}$	Input Load Current		$\pm 10$	$\mu$ A	$V_{IN} = V_{CC}$ to 0V
$I_{OFL}$	Output Float Leakage		$\pm 10$	$\mu$ A	$V_{OUT} = V_{CC}$ to 0.45V

**NOTE:**

- Available on any 8 pins from Port B and C.

# CD4020B, CD4024B, CD4040B Types

## COS/MOS Ripple-Carry Binary Counter/Dividers

High-Voltage Types (20-Volt Rating)

CD4020B — 14 Stage

CD4024B — 7 Stage

CD4040B — 12 Stage

RCA-CD4020B, CD4024B, and CD4040B are ripple-carry binary counters. All counter stages are master-slave flip-flops. The state of a counter advances one count on the negative transition of each input pulse; a high level on the RESET line resets the counter to its all zeros state. Schmitt trigger action on the input-pulse line permits unlimited rise and fall times. All inputs and outputs are buffered.

The CD4020B and CD4040B types are supplied in 16-lead hermetic dual-in-line ceramic packages (D and F suffixes), 16-lead dual-in-line plastic packages (E suffix), and in chip form (H suffix).

The CD4024B types are supplied in 14-lead hermetic dual-in-line ceramic packages (D and F suffixes), 14-lead dual-in-line plastic packages (E suffix), and in chip form (H suffix).

### Features:

- Medium-speed operation
- Fully static operation
- Buffered inputs and outputs
- 100% tested for quiescent current at 20 V
- Standardized, symmetrical output characteristics
- Fully static operation
- Common reset
- 5-V, 10-V, and 15-V parametric ratings
- Maximum input current of 1  $\mu$ A at 18 V over full package-temperature range; 100 nA at 18 V and 25°C
- Noise margin (over full package-temperature range):
  - 1 V at  $V_{DD} = 5$  V
  - 2 V at  $V_{DD} = 10$  V
  - 2.5 V at  $V_{DD} = 15$  V

Meets all requirements of JEDEC Tentative Standard No. 13A, "Standard Specifications for Description of 'B' Series CMOS Devices"

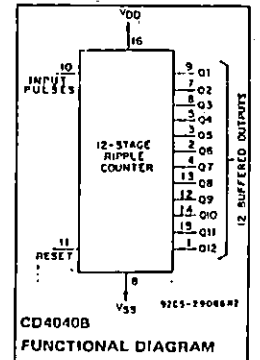
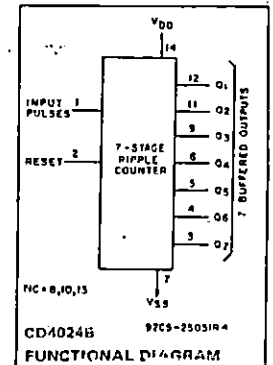
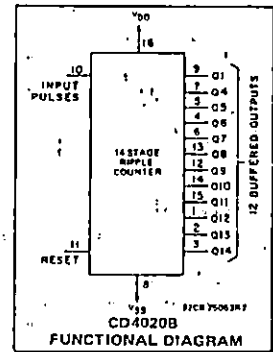
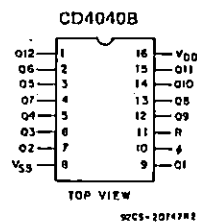
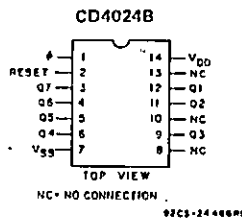
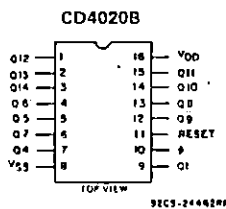
### Applications:

- Control counters
- Timers
- Frequency dividers
- Time-delay circuits

### MAXIMUM RATINGS, Absolute-Maximum Values:

DC SUPPLY-VOLTAGE RANGE, $V_{DD}$ (Voltages referenced to $V_{SS}$ Terminal)	-0.5 to +20 V
INPUT VOLTAGE RANGE, ALL INPUTS	-0.5 to $V_{DD} + 0.5$ V
DC INPUT CURRENT, ANY ONE INPUT	$\pm 10$ mA
POWER DISSIPATION PER PACKAGE ( $P_D$ ):	
For $T_A = -40$ to $+60^\circ\text{C}$ (PACKAGE TYPE E)	500 mW
For $T_A = +60$ to $+85^\circ\text{C}$ (PACKAGE TYPE E)	Derate Linearly at 12 mW/ $^\circ\text{C}$ to 200 mW
For $T_A = -55$ to $+100^\circ\text{C}$ (PACKAGE TYPES D, F)	500 mW
For $T_A = +100$ to $+125^\circ\text{C}$ (PACKAGE TYPES D, F)	Derate Linearly at 12 mW/ $^\circ\text{C}$ to 200 mW
DEVICE DISSIPATION PER OUTPUT TRANSISTOR FOR $T_A =$ FULL PACKAGE-TEMPERATURE RANGE (All Package Types)	100 mW
OPERATING-TEMPERATURE RANGE ( $T_A$ ):	
PACKAGE TYPES D, F, H	-55 to $+125^\circ\text{C}$
PACKAGE TYPE E	-40 to $+85^\circ\text{C}$
STORAGE TEMPERATURE RANGE ( $T_{stg}$ )	-65 to $+150^\circ\text{C}$
LEAD TEMPERATURE (DURING SOLDERING):	
At distance 1/16 $\pm$ 1/32 inch (1.59 $\pm$ 0.79 mm) from case for 10 s max.	+265 $^\circ\text{C}$

### TERMINAL ASSIGNMENTS



# CD4020B, CD4024B, CD4040B Types

RECOMMENDED OPERATING CONDITIONS at  $T_A = 25^\circ\text{C}$ . Unless Otherwise Specified  
 For maximum reliability, nominal operating conditions should be selected so that operation is always within the following ranges:

CHARACTERISTIC	$V_{DD}$	Min.	Max.	UNITS
Supply Voltage Range (at $T_A = \text{Full Package Temperature Range}$ )		3	18	V
Input-Pulse Frequency, $f_{\phi}$	5 10 15	—	3.5 8 12	MHz
Input-Pulse Width, $t_W$	5 10 15	140 60 40	—	ns
Input-Pulse Rise or Fall Time, $t_{r\phi}, t_{f\phi}$	5 10 15	Unlimited		$\mu\text{s}$
Reset Pulse Width, $t_W$	5 10 15	200 80 60	—	ns
Reset Removal Time, $t_{REM}$	5 10 15	350 150 100	—	ns

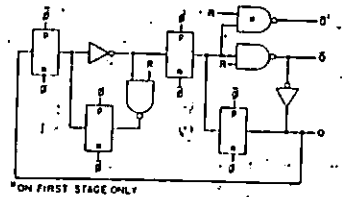


Fig. 4 — Detail of typical flip-flop stage.

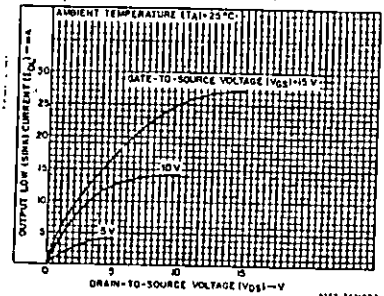


Fig. 5 — Typical output low (sink) current characteristics.

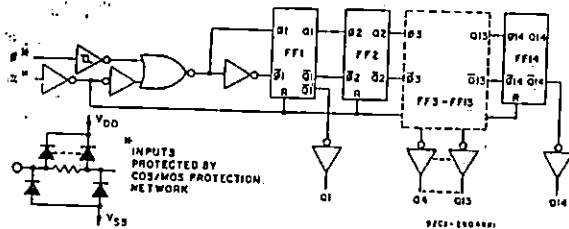


Fig. 1 — Logic diagram for CD4020B.

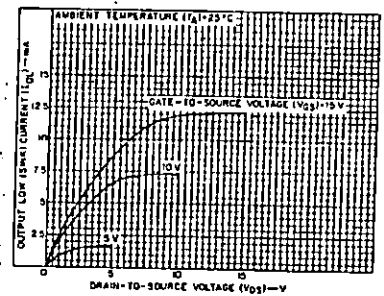


Fig. 6 — Minimum output low (sink) current characteristics.

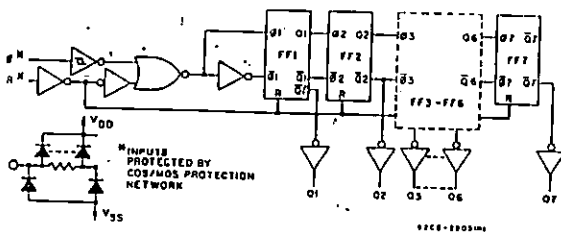


Fig. 2 — Logic diagram for CD4024B.

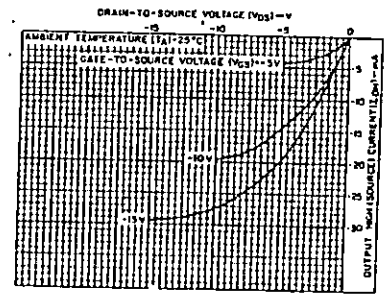


Fig. 7 — Typical output high (source) current characteristics.

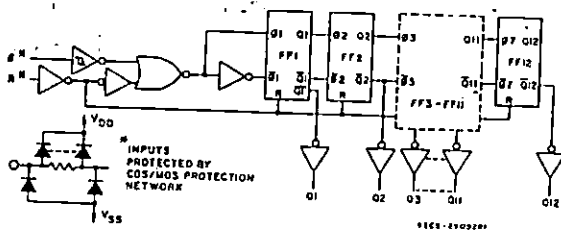


Fig. 3 — Logic diagram for CD4040B.

# CD4020B, CD4024B, CD4040B Types

## STATIC ELECTRICAL CHARACTERISTICS

CHARACTERISTIC	CONDITIONS			LIMITS AT INDICATED TEMPERATURES (°C)							UNITS
	V <sub>O</sub> (V)	V <sub>IN</sub> (V)	V <sub>DD</sub> (V)	Values at -55, +25, +125 Apply to D,F,H Packages			Values at -40, +25, +85 Apply to E Package				
				-55	-40	+85	+125	Min.	Typ.	Max.	
Quiescent Device Current, I <sub>DD</sub> Max.	-	0.5	5	5	5	150	150	-	0.04	5	μA
	-	0.10	10	10	10	300	300	-	0.04	10	
	-	0.15	15	20	20	600	600	-	0.04	20	
	-	0.20	20	100	100	3000	3000	-	0.08	100	
Output Low (Sink) Current I <sub>OL</sub> Min.	0.4	0.5	5	0.64	0.61	0.42	0.36	0.51	1	-	mA
	0.5	0.10	10	1.6	1.5	1.1	0.9	1.3	2.6	-	
	1.5	0.15	15	4.2	4	2.8	2.4	3.4	6.8	-	
Output High (Source) Current, I <sub>OH</sub> Min.	4.6	0.5	5	-0.64	-0.61	-0.42	-0.36	-0.51	-1	-	mA
	2.5	0.5	5	-2	-1.8	-1.3	-1.15	-1.6	-3.2	-	
	9.5	0.10	10	-1.6	-1.5	-1.1	-0.9	-1.3	-2.6	-	
	13.5	0.15	15	-4.2	-4	-2.8	-2.4	-3.4	-6.8	-	
Output Voltage: Low Level, V <sub>OL</sub> Max.	-	0.5	5	0.05			-	0	0.05	-	V
	-	0.10	10	0.05			-	0	0.05	-	
	-	0.15	15	0.05			-	0	0.05	-	
Output Voltage: High Level, V <sub>OH</sub> Min.	-	0.5	5	4.95			4.95	5	-	-	V
	-	0.10	10	9.95			9.95	10	-	-	
	-	0.15	15	14.95			14.95	15	-	-	
Input Low Voltage, V <sub>IL</sub> Max.	0.5, 4.5	-	5	1.5			-	-	1.5	-	V
	1.9	-	10	3			-	-	3	-	
	1.5, 13.5	-	15	4			-	-	4	-	
Input High Voltage, V <sub>IH</sub> Min.	0.5, 4.5	-	5	3.5			3.5	-	-	-	V
	1.9	-	10	7			7	-	-	-	
	1.5, 13.5	-	15	11			11	-	-	-	
Input Current I <sub>IN</sub> Max.		0.18	18	±0.1	±0.1	±1	±1	-	±10 <sup>-5</sup>	±0.1	μA

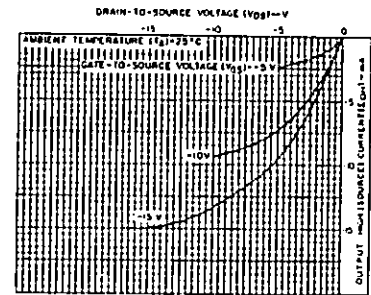


Fig. 8 - Minimum output high (source) current characteristics.

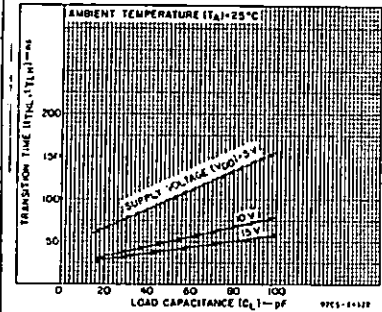
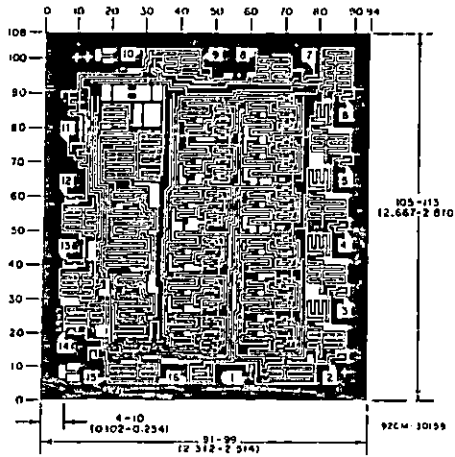
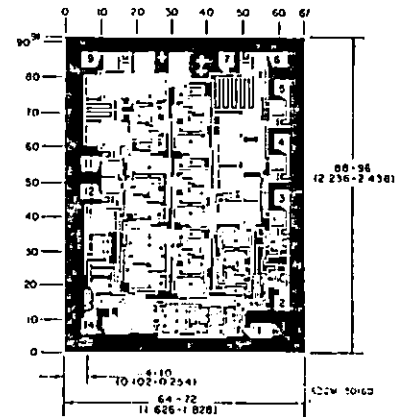


Fig. 9 - Typical transition time as a function of load capacitance.



Dimensions and Pad Layout for CD4020BH. Dimensions and pad layout for CD4040BH are identical.

Dimensions in parentheses are in millimeters and are derived from the basic inch dimensions as indicated. Grid graduations are in mils (10<sup>-3</sup> inch).



Dimensions and Pad Layout for CD4024BH.

The photographs and dimensions of each COS/MOS chip represent a chip when it is part of the wafer. When the wafer is cut into chips, the cleavage angles are 52° instead of 90° with respect to the face of the chip. Therefore, the isolated chip is actually 7 mils (0.17 mm) larger in both dimensions.

# CD4020B, CD4024B, CD4040B Types

DYNAMIC ELECTRICAL CHARACTERISTICS at  $T_A = 25^\circ\text{C}$ , Input  $t_r, t_f = 20 \text{ ns}$ ,  
 $C_L = 50 \text{ pF}$ ,  $R_L = 200 \text{ k}\Omega$ .

CHARACTERISTIC	TEST CONDITIONS	$V_{DD}$ (V)	LIMITS			UNITS
			Min.	Typ.	Max.	
<b>Input-Pulse Operation</b>						
Propagation Delay Time, $\phi$ to $Q_1$ Out; $t_{PHL}, t_{PLH}$		5	—	180	360	ns
		10	—	80	160	
		15	—	65	130	
$Q_n$ to $Q_{n+1}$ ; $t_{PHL}, t_{PLH}$		5	—	100	200	ns
		10	—	40	80	
		15	—	30	60	
Transition Time, $t_{THL}, t_{TLH}$		5	—	100	200	ns
		10	—	50	100	
		15	—	40	80	
Minimum Input-Pulse Width, $t_{Wp}$		5	—	70	140	ns
		10	—	30	60	
		15	—	20	40	
Input-Pulse Rise or Fall Time, $t_{r\phi}, t_{f\phi}$		5	Unlimited			$\mu\text{s}$
		10				
		15				
Maximum Input-Pulse Frequency, $f_{\phi}$		5	3.5	7	—	MHz
		10	8	16	—	
		15	12	24	—	
Input Capacitance, $C_i$	Any Input		—	5	7.5	pF
<b>Reset Operation</b>						
Propagation Delay Time, $t_{PHL}$		5	—	140	280	ns
		10	—	60	120	
		15	—	50	100	
Minimum Reset Pulse Width, $t_{Wp}$		5	—	100	200	ns
		10	—	40	80	
		15	—	30	60	
Reset Removal Time, $t_{REM}$		5	—	175	350	ns
		10	—	75	150	
		15	—	50	100	

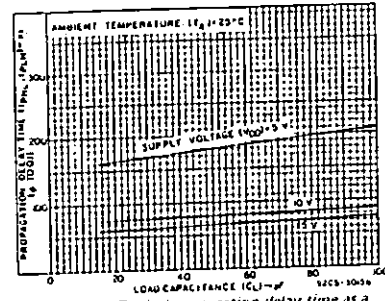


Fig. 10 - Typical propagation delay time as a function of load capacitance ( $\phi$  to  $Q_1$ ).

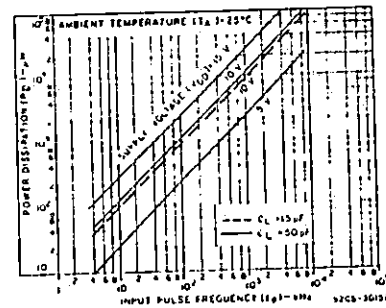


Fig. 11 - Typical dynamic power dissipation as a function of input pulse frequency for CD4020B.

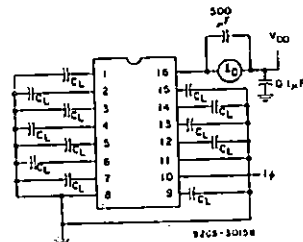


Fig. 12 - Dynamic power dissipation test circuit for CD4020B.

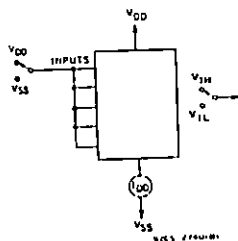


Fig. 13 - Quiescent device current test circuit.

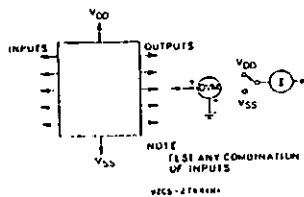


Fig. 14 - Input voltage test circuits.

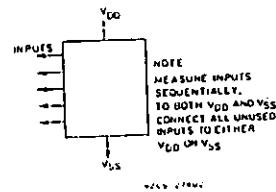


Fig. 15 - Input current test circuit.



# CD4051B, CD4052B, CD4053B Types

## COS/MOS Analog Multiplexers/Demultiplexers\*

With Logic-Level Conversion

High-Voltages Types (20-Volt Rating)

- CD4051B — Single 8-Channel
- CD4052B — Differential 4-Channel
- CD4053B — Triple 2-Channel

BCA-CD4051B, CD4052B, and CD4053B analog multiplexers/demultiplexers are digitally controlled analog switches having low ON impedance and very low OFF leakage current. Control of analog signals up to 20 V peak-to-peak can be achieved by digital signal amplitudes of 4.5 to 20 V (if  $V_{DD}-V_{SS} = 3$  V, a  $V_{DD}-V_{EE}$  of up to 13 V can be controlled; for  $V_{DD}-V_{EE}$  level differences above 13 V, a  $V_{DD}-V_{SS}$  of at least 4.5 V is required). For example, if  $V_{DD} = 14.5$  V,  $V_{SS} = 0$ , and  $V_{EE} = -13.5$  V, analog signals from  $-13.5$  V to  $14.5$  V can be controlled by digital inputs of 0 to 5 V. These multiplexer circuits dissipate extremely low quiescent power over the full  $V_{DD}-V_{SS}$  and  $V_{DD}-V_{EE}$  supply-voltage ranges, independent of the logic state of the control signals. When a logic "1" is present at the inhibit input terminal all channels are off.

The CD4051B is a single 8-channel multiplexer having three binary control inputs, A, B, and C, and an inhibit input. The three binary signals select 1 of 8 channels to be turned on, and connect one of the 8 inputs to the output.

The CD4052B is a differential 4-channel multiplexer having two binary control inputs, A and B, and an inhibit input. The two binary input signals select 1 of 4 pairs of channels to be turned on and connect the analog inputs to the outputs.

The CD4053B is a triple 2-channel multiplexer having three separate digital control inputs, A, B, and C, and an inhibit input. Each control input selects one of a pair of channels which are connected in a single-pole double-throw configuration.

The CD4051B, CD4052B, and CD4053B are supplied in 16-lead ceramic dual-in-line packages (D and F suffixes), 16-lead plastic dual-in-line packages (E suffix), and in chip form (H suffix).

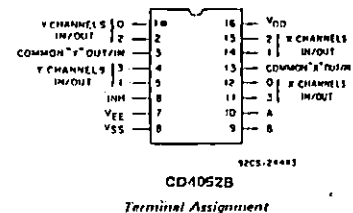
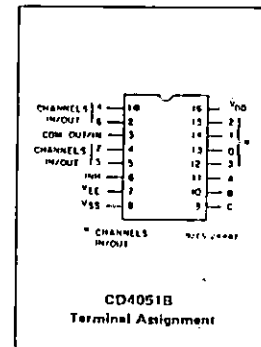
\* When these devices are used as demultiplexers, the "CHANNEL IN/OUT" terminals are the outputs and the "COMMON OUT/IN" terminals are the inputs.

### Applications:

- Analog and digital multiplexing and demultiplexing
- A/D and D/A conversion
- Signal gating

### Features:

- Wide range of digital and analog signal levels: digital 3 to 20 V, analog to 20 V<sub>p-p</sub>
- Low ON resistance: 125  $\Omega$  (typ.) over 15-V<sub>p-p</sub> signal-input range for  $V_{DD}-V_{EE} = 15$  V
- High OFF resistance: channel leakage of  $\pm 100$  pA (typ.) @  $V_{DD}-V_{EE} = 18$  V
- Logic-level conversion for digital addressing: signals of 3 to 20 V ( $V_{DD}-V_{SS} = 3$  to 20 V) to switch analog signals to 20 V<sub>p-p</sub> ( $V_{DD}-V_{EE} \approx 20$  V); see introductory text
- Matched switch characteristics:  $R_{ON} = 5 \Omega$  (typ.) for  $V_{DD}-V_{EE} = 15$  V
- Very low quiescent power dissipation under all digital-control input and supply conditions: 0.2  $\mu$ W (typ.) @  $V_{DD}-V_{SS} = V_{DD}-V_{EE} = 10$  V
- Binary address decoding on chip
- 5-, 10-, and 15-V parametric ratings
- 100% tested for quiescent current at 20 V
- Maximum input current of 1  $\mu$ A at 18 V over full package temperature range; 100 nA at 18 V and 25°C

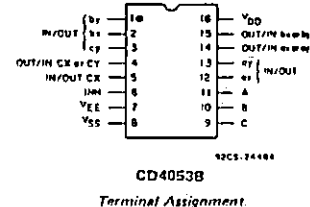


### RECOMMENDED OPERATING CONDITIONS AT $T_A = 25^\circ\text{C}$ (Unless Otherwise Specified)

For maximum reliability, nominal operating conditions should be selected so that operation is always within the following ranges. Values shown apply to all types except as noted.

CHARACTERISTIC	V <sub>DD</sub>	Min.	Max.	Units
Supply-Voltage Range ( $T_A =$ Full Package-Temp. Range)	—	3	18	V
Multiplexer Switch Input Current Capability*	—	—	25	mA
Output Load Resistance	—	100	—	$\Omega$

\* In certain applications, the external load resistive current may include both  $V_{DD}$  and signal-line components. To avoid drawing  $V_{DD}$  current when switch current flows into the transmission gate inputs, the voltage drop across the bidirectional switch must not exceed 0.8 volt (calculated from  $R_{ON}$  values shown in ELECTRICAL CHARACTERISTICS CHART). No  $V_{DD}$  current will flow through  $R_L$  if the switch current flows into terminal 3 on the CD4051; terminals 3 and 13 on the CD4052; terminals 4, 14, and 15 on the CD4053.



# CD4051B, CD4052B, CD4053B Types

## MAXIMUM RATINGS, Absolute-Maximum Values:

DC SUPPLY-VOLTAGE RANGE, (V <sub>DD</sub> )	-0.5 to +20 V
(Voltages referenced to V <sub>SS</sub> or V <sub>EE</sub> , whichever is more negative)	
INPUT VOLTAGE RANGE, ALL INPUTS	-0.5 to V <sub>DD</sub> +0.5 V
DC INPUT CURRENT, ANY ONE INPUT	±10 mA
POWER DISSIPATION PER PACKAGE (P <sub>D</sub> ):	
For T <sub>A</sub> = -40 to +60°C (PACKAGE TYPE E)	500 mW
For T <sub>A</sub> = +60 to +85°C (PACKAGE TYPE E)	Derate Linearly at 12 mW/°C to 200 mW
For T <sub>A</sub> = -55 to +100°C (PACKAGE TYPES D, F)	500 mW
For T <sub>A</sub> = +100 to +125°C (PACKAGE TYPES D, F)	Derate Linearly at 12 mW/°C to 200 mW
DEVICE DISSIPATION PER OUTPUT TRANSISTOR	
FOR T <sub>A</sub> = FULL PACKAGE-TEMPERATURE RANGE (All Package Types)	100 mW
OPERATING-TEMPERATURE RANGE (T <sub>A</sub> ):	
PACKAGE TYPES D, F, H	-55 to +125°C
PACKAGE TYPE E	-40 to +85°C
STORAGE TEMPERATURE RANGE (T <sub>stg</sub> )	-65 to +150°C
LEAD TEMPERATURE (DURING SOLDERING):	
At distance 1/16 ± 1/32 inch (1.59 ± 0.79 mm) from case for 10 s max.	+265°C

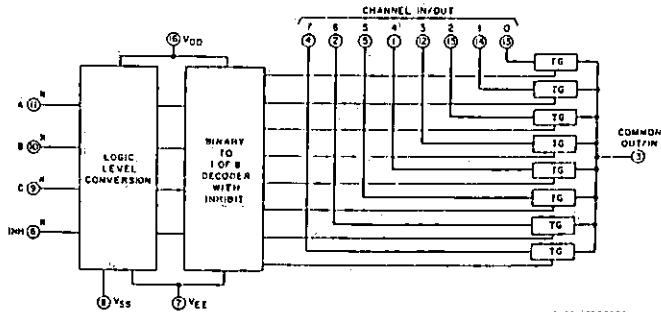


Fig. 1 - Functional diagram of CD4051B.

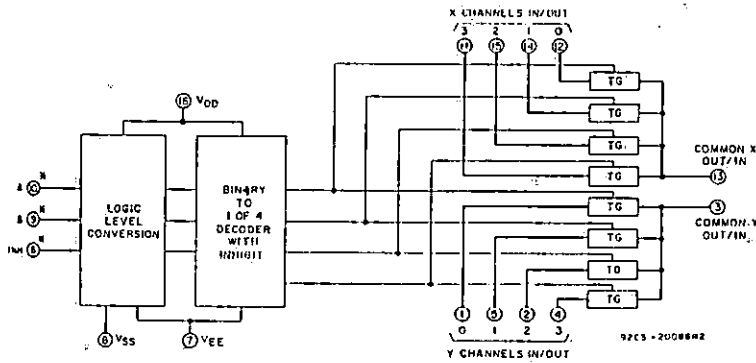


Fig. 2 - Functional diagram of CD4052B.

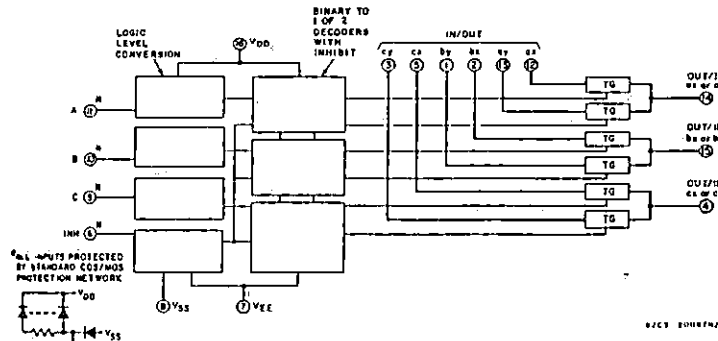


Fig. 3 - Functional diagram of CD4053B.

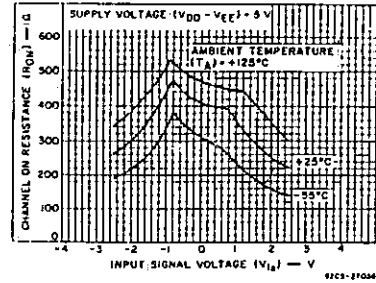


Fig. 4 - Typical channel ON resistance vs input signal voltage (all types).

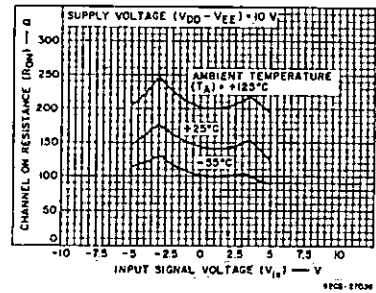


Fig. 5 - Typical channel ON resistance vs. input signal voltage (all types).

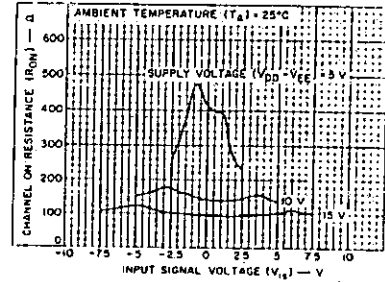


Fig. 6 - Typical channel ON resistance vs. input signal voltage (all types).

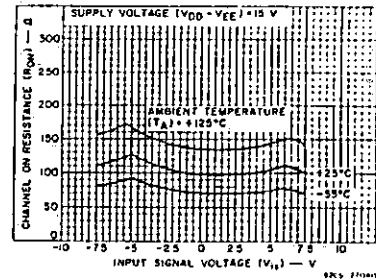


Fig. 7 - Typical channel ON resistance vs. input signal voltage (all types).

# CD4051B, CD4052B, CD4053B Types

## ELECTRICAL CHARACTERISTICS

CHARACTERISTIC	CONDITIONS				LIMITS at Indicated Temperature (°C)							Units		
	V <sub>IS</sub> (V)	V <sub>EE</sub> (V)	V <sub>SS</sub> (V)	V <sub>DD</sub> (V)	Values at -55, +25, +125, apply to D, F, H pkg				Values at -40, +25, +85, apply to E p kgs					
					-55	-40	+85	+125	Min.	Typ.	Max.			
<b>SIGNAL INPUTS (V<sub>IS</sub>) AND OUTPUTS (V<sub>OS</sub>)</b>														
Quiescent Device Current, I <sub>DD</sub> Max.				5	5	5	150	150	-	0.04	5	μA		
				10	10	10	300	300	-	0.04	10			
				15	20	20	600	600	-	0.04	20			
				20	100	100	3000	3000	-	0.08	100			
On-State Resistance 0 < V <sub>IS</sub> < V <sub>DD</sub> r <sub>on</sub> Max.				0	0	5	800	850	1200	1300	-	470	1050	Ω
				0	0	10	310	330	520	550	-	180	400	
				0	0	15	200	210	300	320	-	125	240	
Change in On-State Resistance (Between Any Two Channels) Δr <sub>on</sub>				0	0	5	-	-	-	-	-	15	-	Ω
				0	0	10	-	-	-	-	-	10	-	
				0	0	15	-	-	-	-	-	5	-	
OFF Channel Leakage Current: Any Channel OFF Max. or All Channels OFF (Common OUT/IN) Max.				0	0	18	±100*	±1000*	-	±0.01	±100*	nA		
Capacitance: Input, C <sub>IS</sub> Output, C <sub>OS</sub> CD4051 CD4052 CD4053				-	-	-	-	-	-	5	-	pF		
				-	-	-	-	-	-	30	-			
				-	-	-	-	-	-	18	-			
				-	-	-	-	-	-	9	-			
Feedthrough, C <sub>IOS</sub>				-	-	-	-	-	-	0.2	-			
Propagation Delay Time (Signal Input to Output)	V <sub>DD</sub>	R <sub>L</sub> = 200 kΩ	C <sub>L</sub> = 50 pF	t <sub>r</sub> , t <sub>f</sub> = 20 ns	5	-	-	-	-	30	60	ns		
					10	-	-	-	-	15	30			
					15	-	-	-	-	10	20			

\* Determined by minimum feasible leakage measurement for automatic testing.

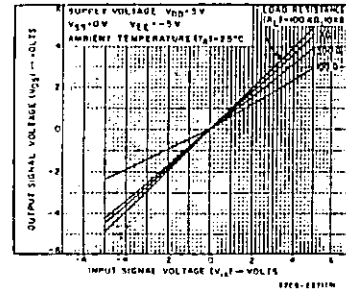


Fig. 8 - Typical ON characteristics for 1 of 8 channels (CD4051B).

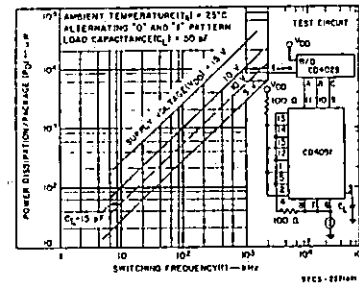


Fig. 9 - Typical dynamic power dissipation vs. switching frequency (CD4051B).

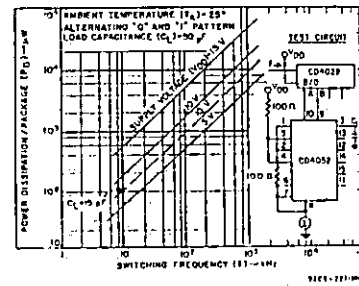


Fig. 10 - Typical dynamic power dissipation vs. switching frequency (CD4052B).

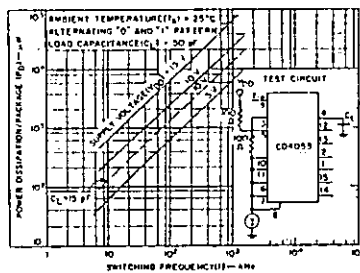
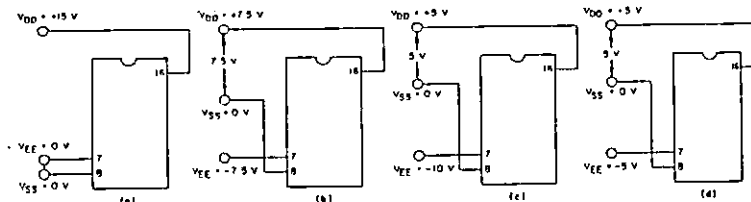


Fig. 11 - Typical dynamic power dissipation vs. switching frequency (CD4053B).



The ADDRESS (digital control inputs) and INHIBIT logic levels are: "0" = V<sub>SS</sub> and "1" = V<sub>DD</sub>. The analog signal (through the TG) may swing from V<sub>EE</sub> to V<sub>DD</sub>.

Fig. 12 - Typical bias voltages.



# CD4051B, CD4052B, CD4053B Types

## ELECTRICAL CHARACTERISTICS (Cont'd)

CHARACTERISTIC	CONDITIONS				LIMITS at Indicated Temperature (°C)							Units	
	$V_{IH}$	$V_{EE}$	$V_{SS}$	$V_{DD}$	Values at -66, +25, +125, apply to D, F, H pkg Values at -40, +25, +85, apply to E pkg								
	(V)	(V)	(V)	(V)	-65	-40	+85	+125	+25				
											Min.	Typ.	Max.
<b>CONTROL (ADDRESS or INHIBIT) <math>V_C</math></b>													
Input Low Voltage, $V_{IL}$ Max.	= $V_{DD}$ thru 1 k $\Omega$	$V_{EE}=V_{SS}$ $R_L=1\text{ k}\Omega$ to $V_{SS}$ $I_{IS}<2\text{ }\mu\text{A}$ on all OFF Channels	5	1.5				-		1.5	V		
			10	3				-		3			
			15	4				-		4			
Input High Voltage, $V_{IH}$ Min.			5	3.5				3.5					
			10	7				7					
			15	11				11					
Input Current, $I_{IN}$ Max.	$V_{IN} = 0, 1\text{B}$		1B	$\pm 0.1$	$\pm 0.1$	$\pm 1$	$\pm 1$	-	$\pm 10^{-5}$	$\pm 0.1$	$\mu\text{A}$		
Propagation Delay Time: Address-to-Signal OUT (Channels ON or OFF) See Figs. 14, 15, 18	$t_r, t_f = 20\text{ ns}, C_L = 50\text{ pF}$		0 0 5	-	-	-	-	-	360	720	ns		
			0 0 10	-	-	-	-	-	160	320			
			0 0 15	-	-	-	-	-	120	240			
			-5 0 5	-	-	-	-	-	225	450			
Inhibit-to-Signal OUT (Channel turning ON)	$R_L = 10\text{ k}\Omega, C_L = 50\text{ pF}$ $t_r, t_f = 20\text{ ns}$		0 0 5	-	-	-	-	-	360	720	ns		
			0 0 10	-	-	-	-	-	160	320			
			0 0 15	-	-	-	-	-	120	240			
			-10 0 5	-	-	-	-	-	200	400			
Inhibit-to-Signal OUT (Channel turning OFF)	$R_L = 300\Omega, C_L = 50\text{ pF}$ $t_r, t_f = 20\text{ ns}$		0 0 5	-	-	-	-	-	200	450	ns		
			0 0 10	-	-	-	-	-	90	210			
			0 0 15	-	-	-	-	-	70	160			
			-10 0 5	-	-	-	-	-	130	300			
Input Capacitance, $C_{IN}$ (Any Address or Inhibit Input)				-	-	-	-	-	5	7.5	$\text{pF}$		

INPUT STATES				"ON" CHANNEL(S)
INHIBIT	C	B	A	
<b>CD4051B</b>				
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	NONE
<b>CD4052B</b>				
INHIBIT	B	A		
0	0	0	0x, 0y	
0	0	1	1x, 1y	
0	1	0	2x, 2y	
0	1	1	3x, 3y	
1	X	X	NONE	
<b>CD4053B</b>				
INHIBIT	A or B or C			
0	0		ax or bx or cx	
0	1		ay or by or cy	
1	X		NONE	

X = Don't care

Fig. 13 - Truth tables.

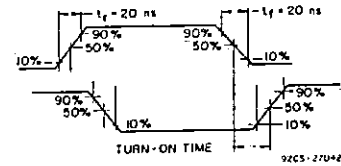


Fig. 14 - Waveforms, channel being turned ON ( $R_L = 10\text{ k}\Omega$ ).

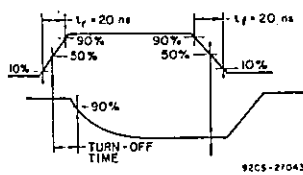


Fig. 15 - Waveforms, channel being turned OFF ( $R_L = 300\Omega$ ).

## TEST CIRCUITS

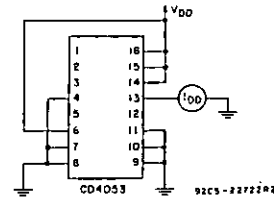
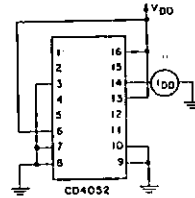
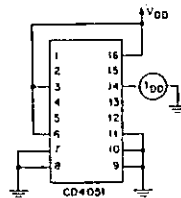


Fig. 16 - OFF channel leakage current - any channel OFF.

# TYPES SN54LS377, SN54LS378, SN54LS379, SN74LS377, SN74LS378, SN74LS379 OCTAL, HEX, AND QUAD D-TYPE FLIP-FLOPS WITH ENABLE

BULLETIN NO. DL-S 7612474, OCTOBER 1976

- 'LS377 and 'LS378 Contain Eight and Six Flip-Flops, Respectively, with Single-Rail Outputs
- 'LS379 Contains Four Flip-Flops with Double-Rail Outputs
- Individual Data Input to Each Flip-Flop
- Applications Include:  
Buffer/Storage Registers  
Shift Registers  
Pattern Generators

### description

These monolithic, positive-edge-triggered flip-flops utilize TTL circuitry to implement D-type flip-flop logic with an enable input. The 'LS377, 'LS378, and 'LS379 devices are similar to 'LS273, 'LS174, and 'LS175, respectively, but feature a common enable instead of a common clear.

Information at the D inputs meeting the setup time requirements is transferred to the Q outputs on the positive-going edge of the clock pulse if the enable input  $\bar{G}$  is low. Clock triggering occurs at a particular voltage level and is not directly related to the transition time of the positive-going pulse. When the clock input is at either the high or low level, the D input signal has no effect at the output. The circuits are designed to prevent false clocking by transitions at the  $\bar{G}$  input.

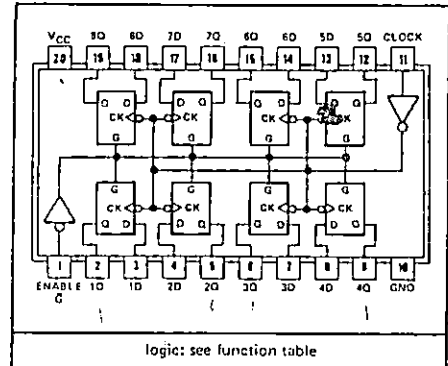
These flip-flops are guaranteed to respond to clock frequencies ranging from 0 to 30 MHz while maximum clock frequency is typically 40 megahertz. Typical power dissipation is 10 milliwatts per flip-flop.

FUNCTION TABLE  
(EACH FLIP-FLOP)

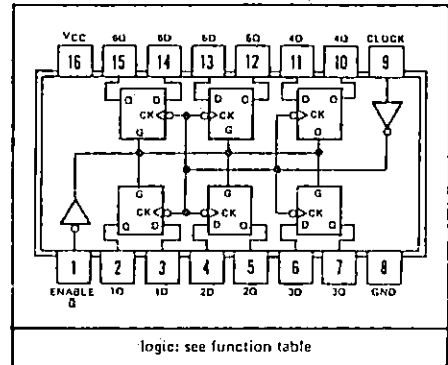
INPUTS			OUTPUTS	
$\bar{G}$	CLOCK	DATA	Q	$\bar{Q}$
H	X	X	$Q_0$	$\bar{Q}_0$
L	↑	H	H	L
L	↑	L	L	H
X	L	X	$Q_0$	$\bar{Q}_0$

See explanation of function tables on page 3-8.

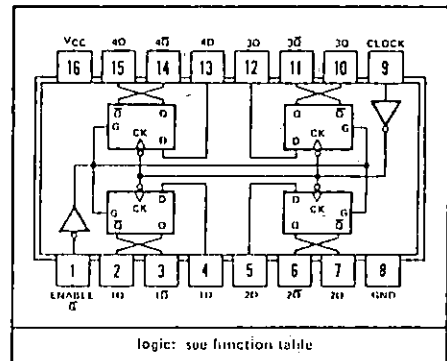
SN54LS377 ... J PACKAGE  
SN74LS377 ... J OR N PACKAGE  
(TOP VIEW)



SN54LS378 ... J OR W PACKAGE  
SN74LS378 ... J OR N PACKAGE  
(TOP VIEW)

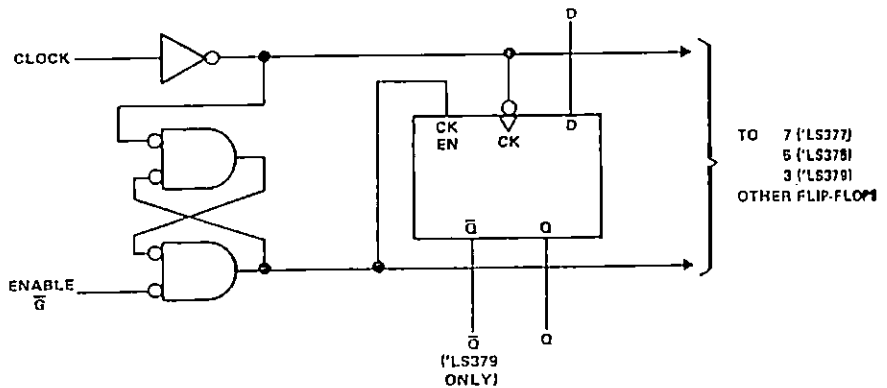


SN54LS379 ... J OR W PACKAGE  
SN74LS379 ... J OR N PACKAGE  
(TOP VIEW)

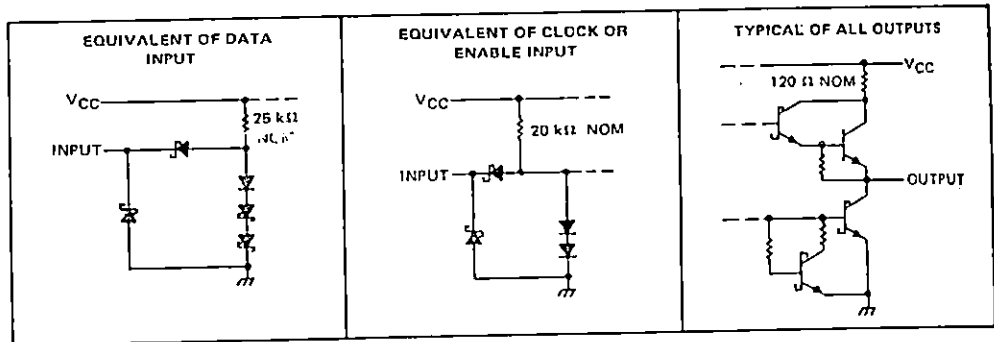


TYPES SN54LS377, SN54LS378, SN54LS379,  
 SN74LS377, SN74LS378, SN74LS379  
 OCTAL, HEX, AND QUAD D-TYPE FLIP-FLOPS WITH ENABLE

functional block diagram



schematics of inputs and outputs



absolute maximum rating over operating free-air temperature range (unless otherwise noted)

Supply voltage, VCC (see Note 1)	7 V
Input voltage	7 V
Operating free-air temperature range: SN54LS*	-55°C to 125°C
SN74LS*	0°C to 70°C
Storage temperature range	-65°C to 150°C

NOTE 1: Voltage values are with respect to network ground terminal.

10V

# TYPES SN54LS377, SN54LS378, SN54LS379, SN74LS377, SN74LS378, SN74LS379 OCTAL, HEX, AND QUAD D-TYPE FLIP-FLOPS WITH ENABLE

recommended operating conditions

	SN54LS'			SN74LS'			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, $V_{CC}$	4.5	5	5.5	4.75	5	5.25	V
High-level output current, $I_{OH}$	-400			-400			$\mu A$
Low-level output current, $I_{OL}$	4			8			mA
Clock frequency, $f_{clock}$	0	30		0	30		MHz
Width of clock or clear pulse, $t_w$	20			20			ns
Setup time, $t_{su}$	Data Input			20†			ns
	Enable active-state			25†			
	Enable inactive-state			10†			
Hold time, $t_h$	Data and enable			5†			ns
Operating free-air temperature, $T_A$	-55		125	0	70		$^{\circ}C$

† The arrow indicates that the rising edge of the clock pulse is used for reference.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†	SN54LS'			SN74LS'			UNIT
		MIN	TYP‡	MAX	MIN	TYP‡	MAX	
$V_{IH}$ High-level input voltage		2			2			V
$V_{IL}$ Low-level input voltage		0.7			0.8			V
$V_{IK}$ Input clamp voltage	$V_{CC} = \text{MIN}, I_I = -18 \text{ mA}$	-1.5			-1.5			V
$V_{OH}$ High-level output voltage	$V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V},$	2.5	3.5		2.7	3.5		V
	$V_{IL} = V_{IL \text{ max}}, I_{OH} = -400 \mu A$							
$V_{OL}$ Low-level output voltage	$V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V},$			0.25	0.4		V	
	$V_{IL} = V_{IL \text{ max}}$					0.35		0.5
$I_I$ Input current at maximum input voltage	$V_{CC} = \text{MAX}, V_I = 7 \text{ V}$	0.1			0.1			mA
$I_{IH}$ High-level input current	$V_{CC} = \text{MAX}, V_I = 2.7 \text{ V}$	20			20			$\mu A$
$I_{IL}$ Low-level input current	$V_{CC} = \text{MAX}, V_I = 0.4 \text{ V}$	-0.4			-0.4			mA
$I_{OS}$ Short-circuit output current§	$V_{CC} = \text{MAX}$	-20			-100			mA
$I_{CC}$ Supply current	$V_{CC} = \text{MAX},$ See Note 2	'LS377		17	28		mA	
		'LS378		13	22			
		'LS379		9	15			

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

‡ All typical values are at  $V_{CC} = 5 \text{ V}, T_A = 25^{\circ}C$ .

§ Note more than one input should be shorted at a time, and duration of the short-circuit should not exceed one second.

NOTE 2: With all outputs open and ground applied to all data and enable inputs,  $I_{CC}$  is measured after a momentary ground, then 4.5 V, is applied to clock.

switching characteristics,  $V_{CC} = 5 \text{ V}, T_A = 25^{\circ}C$

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
$f_{max}$ Maximum clock frequency	$C_L = 15 \text{ pF},$	30	40		MHz
$t_{PLH}$ Propagation delay time, low-to-high-level output from clock	$R_L = 2 \text{ k}\Omega$		17	27	ns
$t_{PHL}$ Propagation delay time, high-to-low-level output from clock	See Note 3		18	27	ns

NOTE 3: Load circuit and voltage waveforms are shown on page 3-11.



TTL  
MSI

**TYPES SN54LS138, SN54LS139, SN54S138, SN54S139,  
SN74LS138, SN74LS139, SN74S138, SN74S139  
DECODERS/DEMULTIPLEXERS**

ILLUSTRATION NO. DL-S 7611004, DECEMBER 1972—REVISED OCTOBER 1974

- Designed Specifically for High-Speed: Memory Decoders  
Data Transmission Systems
- 'S138 and 'LS138 3-to-8-Line Decoders Incorporate 3 Enable Inputs to Simplify Cascading and/or Data Reception
- 'S139 and 'LS139 Contain Two Fully Independent 2-to-4-Line Decoders/ Demultiplexers
- Schottky Clamped for High Performance

TYPE	TYPICAL PROPAGATION DELAY (3 LEVELS OF LOGIC)	TYPICAL POWER DISSIPATION
'LS138	22 ns	32 mW
'S138	8 ns	245 mW
'LS139	22 ns	34 mW
'S139	7.5 ns	300 mW

**description**

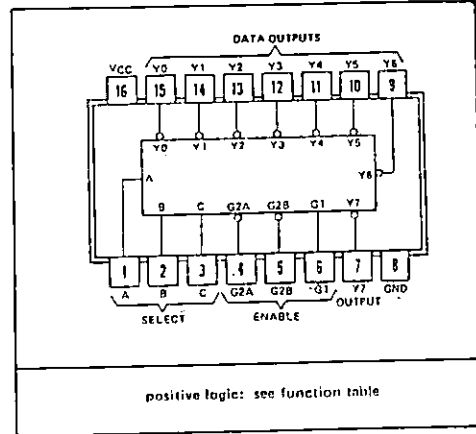
These Schottky-clamped TTL MSI circuits are designed to be used in high-performance memory-decoding or data-routing applications requiring very short propagation delay times. In high-performance memory systems these decoders can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast-enable circuit the delay times of these decoders and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the Schottky-clamped system decoder is negligible.

The 'LS138 and 'S138 decode one-of-eight lines dependent on the conditions at the three binary select inputs and the three enable inputs. Two active-low and one active-high enable inputs reduce the need for external gates or inverters when expanding. A 24-line decoder can be implemented without external inverters and a 32-line decoder requires only one inverter. An enable input can be used as a data input for demultiplexing applications.

The 'LS139 and 'S139 comprise two individual two-line-to-four-line decoders in a single package. The active-low enable input can be used as a data line in demultiplexing applications.

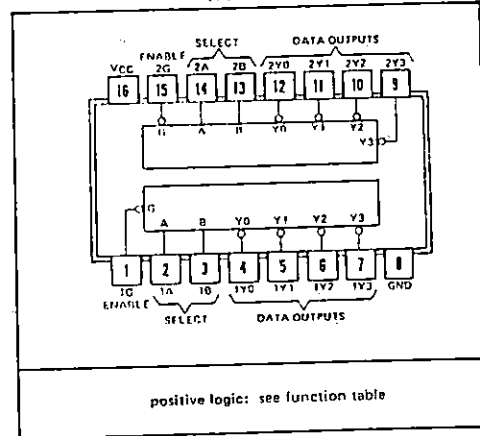
All of these decoders/demultiplexers feature fully buffered inputs each of which represents only one normalized Series 54LS/74LS load ('LS138, 'LS139) or one normalized Series 54S/74S load ('S138, 'S139) to its driving circuit. All inputs are clamped with high-performance Schottky diodes to suppress line-ringing and simplify system design. Series 54LS and 54S devices are characterized for operation over the full military temperature range of -55°C to 125°C. Series 74LS and 74S devices are characterized for 0°C to 70°C industrial systems.

SN54LS138, SN54S138 . . . J OR W PACKAGE  
SN74LS138, SN74S138 . . . J OR N PACKAGE  
(TOP VIEW)



positive logic: see function table

SN54LS139, SN54S139 . . . J OR W PACKAGE  
SN74LS139, SN74S139 . . . J OR N PACKAGE  
(TOP VIEW)

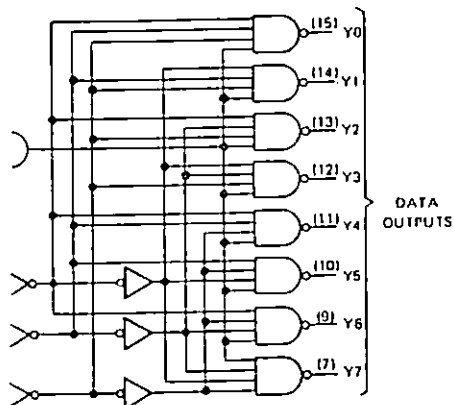


positive logic: see function table

# SN74LS138, SN74S138, SN74LS139, SN74S139 DECODERS/DEMULTIPLEXERS

grams and logic

'LS138, 'S138



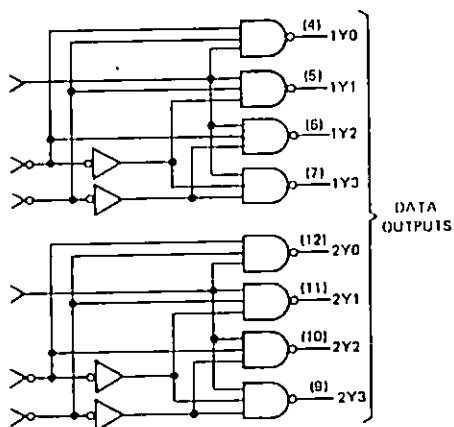
'LS138, 'S138  
FUNCTION TABLE

INPUTS					OUTPUTS							
ENABLE		SELECT										
G1	G2*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	L	H	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

\*G2 = G2A + G2B

H = high level, L = low level, X = irrelevant

'LS139, 'S139

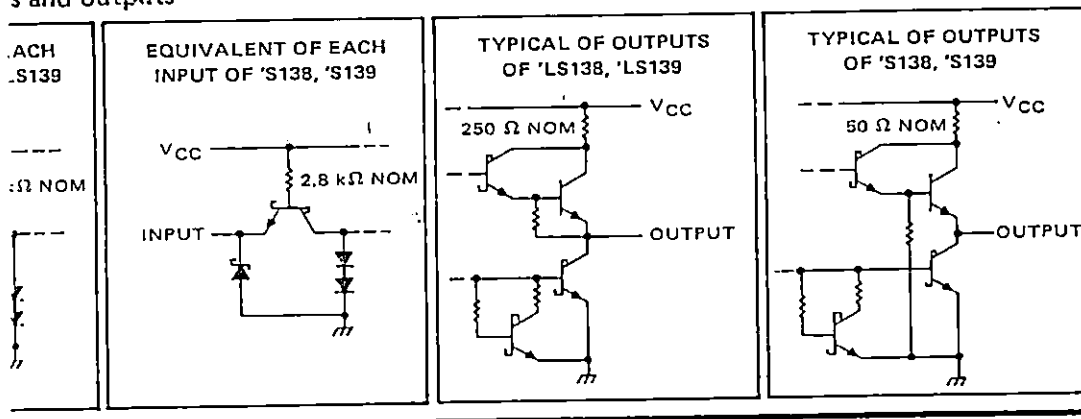


'LS139, 'S139  
(EACH DECODER/DEMULTIPLEXER)  
FUNCTION TABLE

INPUTS			OUTPUTS			
ENABLE	SELECT					
G	B	A	Y0	Y1	Y2	Y3
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	L	H	H	L	H	H
L	H	L	H	H	L	H
L	H	H	H	H	H	L

H = high level, L = low level, X = irrelevant

s and outputs



# TYPES SN54LS138, SN54LS139, SN74LS138, SN74LS139, DECODERS/DEMULTIPLEXERS

REVISED OCTOBER 1976

absolute maximum ratings over operating free-air temperature range (unless otherwise specified)

Supply voltage, $V_{CC}$ (see Note 1)	4.5 to 5.5
Input voltage	-0.5 to 5.5
Operating free-air temperature range: SN54LS138, SN54LS139 Circuits	0 to 70
SN74LS138, SN74LS139 Circuits	-55 to 125
Storage temperature range	-55 to 125

NOTE 1: Voltage values are with respect to network ground terminal.

## recommended operating conditions

	SN54LS138 SN54LS139		
	MIN	NOM	MAX
Supply voltage, $V_{CC}$	4.5	5	5.5
High-level output current, $I_{OH}$			-400
Low-level output current, $I_{OL}$			4
Operating free-air temperature, $T_A$	-55		125

## electrical characteristics over recommended operating free-air temperature range (unless otherwise specified)

PARAMETER	TEST CONDITIONS†	SN54LS138 SN54LS139	
		MIN	TYP‡ M/
$V_{IH}$ High-level input voltage		2	
$V_{IL}$ Low-level input voltage			
$V_{IK}$ Input clamp voltage			
$V_{OH}$ High-level output voltage	$V_{CC} = \text{MIN}, I_I = -18 \text{ mA}$ $V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V},$ $V_{IL} = V_{IL \text{ max}}, I_{OH} = -400 \mu\text{A}$	2.5	3.4
$V_{OL}$ Low-level output voltage	$V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V},$ $V_{IL} = V_{IL \text{ max}}$		0.25
$I_I$ Input current at maximum input voltage	$V_{CC} = \text{MAX}, V_I = 7 \text{ V}$		
$I_{IH}$ High-level input current	$V_{CC} = \text{MAX}, V_I = 2.7 \text{ V}$		
$I_{IL}$ Low-level input current	$V_{CC} = \text{MAX}, V_I = 0.4 \text{ V}$		
$I_{OS}$ Short-circuit output current§	$V_{CC} = \text{MAX}$		-6
$I_{CC}$ Supply current	$V_{CC} = \text{MAX},$ Outputs enabled and open	LS138	6.3
		LS139	6.8

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating type.

‡ All typical values are at  $V_{CC} = 5 \text{ V}, T_A = 25^\circ \text{C}$ .

§ Not more than one output should be shorted at a time.

## switching characteristics, $V_{CC} = 5 \text{ V}, T_A = 25^\circ \text{C}$

PARAMETER¶	FROM (INPUT)	TO (OUTPUT)	LEVELS OF DELAY	TEST CONDITIONS	SN54LS138 SN74LS139	
					MIN	TYP
$t_{PLH}$	Binary Select	Any	2	$C_L = 15 \text{ pF},$ $R_L = 2 \text{ k}\Omega,$ See Note 2		13
$t_{PHL}$						27
$t_{PLH}$			3			18
$t_{PHL}$						26
$t_{PLH}$	Enable	Any	2			12
$t_{PHL}$						21
$t_{PLH}$			3			17
$t_{PHL}$						25

¶  $t_{PLH}$  = propagation delay time, low to high level output;  $t_{PHL}$  = propagation delay time, high-to-low level output.

NOTE 2: Load circuits and waveforms are shown on page 3-11.

maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, $V_{CC}$ (see Note 1)	7 V
Input voltage	5.5 V
Operating free-air temperature range: SN54S138, SN54S139 Circuits	-55°C to 125°C
SN74S138, SN74S139 Circuits	0°C to 70°C
Storage temperature range	-65°C to 150°C

† All voltage values are with respect to network ground terminal.

### Recommended operating conditions

	SN54S138 SN74S139			SN74S138 SN74S139			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, $V_{CC}$	4.5	5	5.5	4.75	5	5.25	V
Input current, $I_{OH}$			-1			-1	mA
Output current, $I_{OL}$			20			20	mA
Free-air temperature, $T_A$	-55		125	0		70	°C

Characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†	SN54S138 SN74S139		SN54S139 SN74S139		UNIT
		MIN	TYP‡	MAX	MIN	
Level input voltage		2		2		V
Level input voltage		0.8		0.8		V
Clamp voltage	$V_{CC} = \text{MIN}, I_I = -18 \text{ mA}$	-1.2		-1.2		V
Level output voltage	$V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V}, V_{IL} = 0.8 \text{ V}, I_{OH} = -1 \text{ mA}$	SN54S* 2.5 3.4		SN54S* 2.5 3.4		V
Level output voltage	$V_{CC} = \text{MIN}, V_{IH} = 2 \text{ V}, V_{IL} = 0.8 \text{ V}, I_{OL} = 20 \text{ mA}$	0.5		0.5		V
Current at maximum input voltage	$V_{CC} = \text{MAX}, V_I = 5.5 \text{ V}$	1		1		mA
Level input current	$V_{CC} = \text{MAX}, V_I = 2.7 \text{ V}$	50		50		µA
Level input current	$V_{CC} = \text{MAX}, V_I = 0.5 \text{ V}$	-2		-2		mA
Circuit output current§	$V_{CC} = \text{MAX}$	-40	-100	-40	-100	mA
Supply current	$V_{CC} = \text{MAX}, \text{Outputs enabled and open}$	49	74	60	90	mA

Values shown as MIN or MAX, use the appropriate value specified under recommended operating conditions for the applicable device.

† All values are at  $V_{CC} = 5 \text{ V}, T_A = 25^\circ\text{C}$ .

‡ When one output should be shorted at a time, and duration of the short-circuit test should not exceed one second.

§ Characteristics,  $V_{CC} = 5 \text{ V}, T_A = 25^\circ\text{C}$

TER†	FROM (INPUT)	TO (OUTPUT)	LEVELS OF DELAY	TEST CONDITIONS	SN54S138, SN74S138			SN54S139, SN74S139			UNIT
					MIN	TYP	MAX	MIN	TYP	MAX	
H	Binary select	Any	2	$C_L = 15 \text{ pF}, R_L = 280 \Omega, \text{See Note 3}$	4.5	7		5	7.5		ns
L					7	10.5		6.5	10		
H			3		7.5	12		7	12		ns
L					8	12		8	12		
H	Enable	Any	2		5	8		5	8		ns
L					7	11		6.5	10		
H			3		7	11					ns
L					7	11					

† Propagation delay time, low-to-high-level output

‡ Propagation delay time, high-to-low-level output

§ Test circuits and waveforms are shown on page 3-10.

**A N E X O 5**

**DIAGRAMA DEL MODULO DE PRUEBA**