

89
2 ej.



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

DISEÑO Y CONSTRUCCION DE UN CONTROLADOR DE DISPOSITIVOS COMPATIBLES CON EL ESTANDAR IEEE - 488

TESIS CON
FALLA DE ORIGEN

TESIS PROFESIONAL
QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
P R E S E N T A :
ARMANDO LOZANO RAMIREZ

DIRECTOR DE TESIS: M.I. LAURO SANTIAGO CRUZ

MEXICO, D. F.

1991



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

I N D I C E

Pág

Introducción	1
1. Descripción general de la Norma IEEE-488	3
Antecedentes históricos de la Norma IEEE-488	
1.1 Objetivos de la Norma IEEE-488	
1.2 Limitaciones de la Norma IEEE-488	
1.3 Descripción general de la Norma	
1.3.1 Especificaciones mecánicas	
1.3.2 Especificaciones eléctricas	
1.3.3 Aspecto funcional	
1.4 Descripción de las señales eléctricas manejadas por la Norma	
1.4.1 Líneas de datos	
1.4.2 Líneas de control de transferencia de datos	
1.4.3 Líneas de control	
1.5 Comandos de la Norma IEEE-488	
1.6 Funciones de la interfaz	
2. La Computadora Personal XT	18
Descripción general de una PC XT	
2.1 El microprocesador 8088	
2.1.1 Arquitectura de la CPU	
2.1.2 Modos de operación	
2.1.3 Registros Internos	
2.2 Descripción de las señales eléctricas manejadas en la ranura de expansión de una PC-XT	
2.3 Clasificación de las líneas eléctricas del bus XT	

- 2.3.1 Líneas de datos
- 2.3.2 Líneas de direcciones
- 2.3.3 Líneas de control de memoria y de E/S
- 2.3.4 Líneas de interrupción
- 2.3.5 Líneas de arbitraje
- 2.3.6 Líneas de error y sincronía
- 2.3.7 Líneas de espera e inicialización
- 2.3.8 Voltajes de alimentación

3. Diseño del Controlador I8255A 27

3.1 Alternativas de diseño

- 3.1.1 El CI 8291 y el CI 8292
- 3.1.2 El CI MC68488
- 3.1.3 El μ P7210

3.2 Diseño de la tarjeta

- 3.2.1 El CI 8255A
 - a) Descripción de las señales de control del CI 8255A
 - b) Modos de operación
- 3.2.2 Decodificación de memoria
- 3.2.3 Manejo de las líneas de control del CI 8255A
- 3.2.4 Manejo de las líneas de datos del *bus* XT y del *bus* 488
- 3.2.5 Manejo de las líneas de protocolo del *bus* 488
- 3.2.6 Manejo de las líneas de control del *bus* 488
- 3.2.7 Asignación de los puertos del CI 8255A

4. Descripción de los programas desarrollados 42

Consideraciones de programación

4.1 Descripción de los diagramas de flujo	
5. Aplicaciones y consideraciones sobre el uso de la tarjeta	46
Como usar los Comandos	
5.1 Resumen de Comandos	
5.2 Programa de aplicación	
6. Resultados y conclusiones	58
Bibliografía	
Apéndices	
Apéndice A	Diagramas de flujo
Apéndice B	Listado del programa fuente del controlador
Apéndice C	Diagrama eléctrico del controlador IEEE-488
Apéndice D	Hojas de especificaciones de los circuitos integrados

INTRODUCCION

En la última década los sistemas de adquisición de datos, que se basan en la facilidad de uso de las señales digitales, han alcanzado un gran desarrollo, por lo que la tendencia actual es el uso de sistemas de medición controlados por computadora.

En términos generales, un sistema de adquisición de datos se puede definir como una combinación de instrumentos de medición controlados por una unidad central. Esta unidad central debe ser capaz de controlar la operación individual o en conjunto de todos los instrumentos integrantes del sistema, para que en este haya un intercambio de datos e instrucciones adecuado.

Una de las respuestas a la necesidad de un intercambio de información adecuado, fué la creación de una norma conocida hoy en día como: "Norma de Interconexión Digital para Instrumentos Programables", conocida más comunmente como la Norma IEEE-488. Esta Norma permite que un dispositivo equipado con ella actúe como controlador, emisor o receptor.

Hoy en día existen numerosos dispositivos que vienen equipados para ser controlados a través del *bus* IEEE-488, y se espera que en un futuro próximo la gran mayoría de los dispositivos vengan provistos de las funciones necesarias para conectarse al *bus*. Obviamente, a los dispositivos se les implementa únicamente las funciones necesarias para su buen desempeño dentro del sistema. Existen pocos equipos, que además de las funciones propias para las que fue creado, puede tener un control del *bus* IEEE-488 y en general esta capacidad aumenta en forma considerable el costo del equipo.

El problema existente en un sistema de adquisición de datos, es que forzosamente uno de nuestros equipos tendría que venir equipado con las funciones de controlador. Una solución es la de diseñar un controlador que pueda ejercer un mando sobre todos los equipos y que no forme parte de ningún dispositivo, asegurándonos así que este controlador pueda ser instalado en cualquier sistema que sea compatible con la Norma.

Existen en el mercado tarjetas capaces de realizar esta labor, variando en costo de acuerdo a la funciones que sea capaz de efectuar. El objetivo de este trabajo se centra en el diseño y construcción de una tarjeta que pueda conectarse en la ranura de expansión de una computadora personal y que además pueda llevar a cabo la mayoría de las funciones definidas por la Norma IEEE-488. También, es necesario que los componentes con los que se integra

el controlador sean de fácil adquisición en el mercado nacional, para que el costo final sea reducido.

La presente tesis se ha dividido en seis capítulos principales:

En el capítulo 1, Descripción General de la Norma IEEE-488, se hace una descripción general de lo que es la Norma IEEE-488, con el fin de dar una idea de los alcances y limitaciones de la Norma, además de que es una guía forzosa en el diseño de la tarjeta.

En el capítulo 2, La computadora personal XT, se da un panorama general de como es una computadora personal, basándose en un modelo general de una XT. Además, se hace énfasis acerca del microprocesador 8088 de Intel, debido a que las computadoras personales que harán uso de esta tarjeta cuentan con este microprocesador como base de su arquitectura. Para finalizar este capítulo, se da una explicación de las señales que conforman la ranura de expansión de una computadora XT, ya que será necesario conocer la función de cada una de ellas para el diseño final del Controlador

En el capítulo 3, Diseño del controlador IEEE-488, se explican las diferentes alternativas de diseño que se tuvieron y la elección final que se escogió en base a los objetivos iniciales del presente trabajo. Se incluye en este capítulo la explicación del diseño final de la tarjeta y de los componentes más importantes que la forman.

El capítulo 4, Descripción de los programas desarrollados, es en donde se explica el diseño de las rutinas de programación necesarias para que la tarjeta pueda funcionar como controlador de la Norma IEEE-488 y llevar a cabo las funciones más importantes establecidas en la misma. Se incluyen diagramas de flujo y los programas fuente.

En el capítulo 5, Aplicaciones y consideraciones sobre el uso de la tarjeta, se incluye un manual de uso de la tarjeta, en el que se explica como usarla una vez que ha sido instalada. También se incluyen un ejemplo de aplicación en el que se ilustra el funcionamiento de la tarjeta y la manera en que debe programar el usuario sus programas de aplicación.

Finalmente en el capítulo 6, Resultados y conclusiones, se da una perspectiva de los problemas que se tuvieron en el desarrollo del presente trabajo y una breve explicación de las limitantes y funciones de la tarjeta.

Capítulo 1

DESCRIPCION GENERAL DE LA NORMA IEEE-488

ANTECEDENTES HISTORICOS DE LA NORMA IEEE-488

La Norma para la interconexión digital de instrumentos programables, conocida normalmente como la Norma IEEE-488, fue elaborada para facilitar el diseño, construcción y uso de sistemas de instrumentación.

Una discusión detallada de la viabilidad, los alcances y objetivos de una interconexión se inició en los EUA y Europa, casi al mismo tiempo, a principios de los años setenta. Tales discusiones tuvieron lugar en respuesta a la urgente necesidad de minimizar, si no eliminar, los recursos consumidos en soluciones únicas para interconectar las diversas componentes de sistemas de medición. Los diseñadores de equipos de medición tendían a seleccionar métodos de interconexión que satisficieran únicamente necesidades específicas. La selección de códigos, niveles lógicos, etc. eran obligados a satisfacer necesidades primarias. Los costos de desarrollo y fabricación eran relativamente altos, cada solución tenía una aplicación única. Cada vez que un instrumento ya tenía un medio para manejar datos de entrada/salida, se presentaba el problema de que los diseñadores de sistemas buscaban mejorar las características del sistema más allá de los objetivos iniciales. El resultado final eran recursos adicionales e ingeniería especial para hacer compatible, a un nivel rudimentario, la comunicación entre los instrumentos. Esta situación existía tanto para sistemas configurados con instrumentos de diferentes fabricantes, así como para aquellos que se desarrollaban con equipo de una sola empresa.

Esta situación exigía encontrar una solución común, una norma de interconexión aplicable a una amplia variedad de productos, una norma que beneficiara tanto a fabricantes como a los usuarios. Bajo la dirección de la Comisión Electrotécnica Internacional (IEC) e instigación del Comité Nacional Alemán fue autorizado un proyecto de desarrollo en 1972. Al mismo tiempo un Comité Nacional de Estados Unidos y un subcomité de la IEEE fueron formados para considerar las necesidades de los fabricantes y usuarios de los Estados Unidos. Los conceptos de la norma que se estaba desarrollando en la Compañía Hewlett-Packard fueron seleccionados como modelo para discusión y desarrollo. El primer encuentro internacional fue celebrado en Munich, Alemania, en octubre de 1972. Una vez puestos de acuerdo en los objetivos generales, la propuesta de Estados Unidos fue seleccionada como base para la elaboración de una norma internacional.

Durante los siguientes dos años, los conceptos básicos fueron elaborados tanto a nivel nacional como internacional y se incluyeron algunos refinamientos para la mejora de la norma. En diciembre de 1974, el Consejo de Normas de la IEEE aprobó lo que hoy es conocido como la Norma IEEE-488. Durante este proceso, se realizó un gran esfuerzo para clarificar las definiciones por medio del uso de diagramas de estado y notaciones.

La Norma IEEE 488-1975 fue publicada por primera vez en abril de 1975. Su contraparte internacional, la publicación 625-1 de la IEC, llamada "Un sistema de interconexión para instrumentos de medición programables, byte serie, bit paralelo", fue aprobado y publicado en 1979. Dos revisiones posteriores a las bases de la Norma 488 han tenido lugar desde su publicación inicial. Primero, la Norma fue revisada en 1978 para añadir algunas clarificaciones editoriales y para quitar oraciones potencialmente ambiguas. En 1980, el Suplemento A fue aprobado y ha sido añadido a todas las subsecuentes impresiones de la Norma.

1.1 OBJETIVOS DE LA NORMA IEEE 488

Con base en las reuniones de los Comités encargados del desarrollo de las especificaciones de la Norma IEEE 488 se plantearon los siguientes objetivos:

- 1) Definir un sistema de interconexión de propósito general para usarse en aplicaciones a distancias limitadas.
- 2) Especificar los requerimientos mecánico, eléctrico y funcional de la interfaz que el dispositivo deberá tener con el fin de que sea interconectado al sistema y pueda comunicarse de una manera no ambigua.
- 3) Definir la terminología y los definiciones relacionados al sistema de interconexión.
- 4) Permitir la interconexión de instrumentos fabricados por distintas empresas en un sistema funcional único.
- 5) Permitir la utilización simultánea de aparatos con un amplio rango de capacidades, desde la más simple a la más compleja, en el sistema.
- 6) Permitir la comunicación directa en el sistema de los aparatos sin que se requiera que todos los mensajes sean mandados a una unidad de control o unidad intermedia.
- 7) Definir un sistema de interconexión que imponga el mínimo de restricciones a las características operativas de cada elemento.

- 8) Definir un sistema que permita la comunicación en forma asincrónica dentro de un amplio margen de velocidades de transmisión.
- 9) Definir un sistema que, en si mismo, sea de un costo relativamente bajo y permita la interconexión de dispositivos de bajo costo.
- 10) Definir un sistema de interconexión que sea fácil de utilizar y de realizar.

1.2 LIMITACIONES DE LA NORMA IEEE 488

Las características mecánicas, eléctricas y funcionales definidas en la Norma, han impuesto una serie de limitaciones en el funcionamiento del sistema de interconexión. Entre las que podrían causar mayores problemas se encuentran:

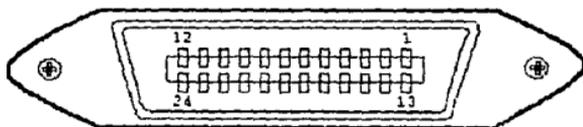
- 1) **Número de dispositivos:** Como máximo pueden estar conectados simultáneamente al *bus* 15 elementos, esto es debido a las características eléctricas del *bus*. Estos 15 elementos son elementos primarios que a su vez pueden tener elemento secundarios. Los elementos primarios tienen asignada una dirección principal, para enviar o recibir mensajes, de un total de 31 direcciones disponibles. Los elementos secundarios conectados a ellos, se asignan combinando la dirección del elemento principal al que pertenecen con otro grupo de 31 direcciones para formar un total de 961 direcciones secundarias.
- 2) **Velocidad de transmisión:** La velocidad máxima de transmisión es de 1 M byte/segundo y la típica de 250 K byte/segundo. Para tener una idea de lo que esto significa hay que tener en cuenta que la mayoría de los sistemas van a ser controlados por una computadora y que sus diferentes periféricos operan a diferentes velocidades tales como: 31 ó 62 K bytes/s para los discos flexibles (sencilla o doble densidad); 5 a 10 k bytes/s para las cintas de casete y un disco duro opera a velocidades de 781 K bytes/s. Hay que aclarar que la velocidad máxima de transmisión no se alcanza fácilmente ya que requiere un diseño muy cuidadoso, que tenga en cuenta los retardos digitales y tiempos con un cable de longitud mínima, las cargas adecuadas y transceptores especiales.
- 3) **Distancia entre elementos:** Esta limitada a 2 metros por el número de elementos, hasta un máximo total de 20 metros. Esta distancia máxima entre elementos obedece a la configuración de carga distribuida del *bus* y la necesidad de tener un elemento de carga por cada dos metros para mantener sus características eléctricas.

1.3 DESCRIPCION GENERAL DE LA NORMA

Las Normas IEEE 488 e IEC únicamente definen los aspectos mecánico, eléctrico y funcional del sistema de interconexión, dejando en plena libertad al diseñador en el aspecto operativo de cada elemento integrante del sistema.

1.3.1 Especificaciones mecánicas

El aspecto mecánico especifica un tipo de conector único, en configuración macho/hembra, para permitir el apilamiento de varios conectores en un solo punto y la formación de enlaces entre los elementos del sistema. El conector es trapezoidal de 24 contactos, en los que las señales se distribuyen de la forma indicada en la figura 1.1a. En la figura 1.1b. se muestra un conector con las medidas especificadas en la Norma.



PIN 1	DIO1	PIN 13	DIO5
PIN 2	DIO2	PIN 14	DIO6
PIN 3	DIO3	PIN 15	DIO7
PIN 4	DIO4	PIN 16	DIO8
PIN 5	E01	PIN 17	REN
PIN 6	DAV	PIN 18	DAV/GND
PIN 7	NRFD	PIN 19	NRFD/GND
PIN 8	NDAC	PIN 20	NDAC/GND
PIN 9	IFC	PIN 21	IFC/GND
PIN 10	Srq	PIN 22	Srq/GND
PIN 11	ATN	PIN 23	ATN/GND
PIN 12	SHIELD	PIN 24	SIGNAL GROUND

Fig. 1.1a. Distribución de las señales en el conector GPIB

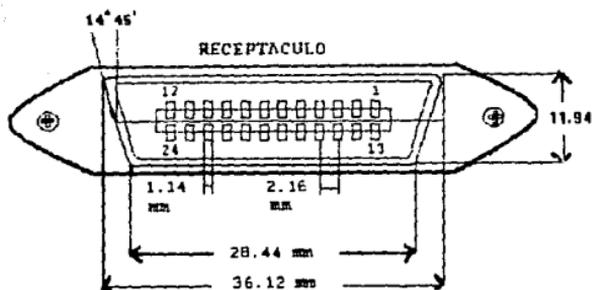


Fig. 1.1b. Conector GPIB con las medidas especificadas por la Norma

1.3.2 Especificaciones eléctricas

Las especificaciones eléctricas de la Norma están basadas para circuitos construidos con tecnología TTL (lógica de transistor-transistor), entre las más importantes cabe destacar:

- a) Niveles lógicos: Los estados alto y bajo están basados en los niveles normalizados de los circuitos TTL. La codificación de los mensajes es hecha utilizando lógica negativa, es decir, cumple con la siguiente convención:

Estado lógico	Nivel eléctrico de la señal
0	Mayor o igual a 2.0 volts Llamado Estado ALTO
1	Menor o igual a 0.8 volts Llamado Estado BAJO

- b) Tipo de manejador: Se deberán usar manejadores de tipo colector abierto para manejar las señales SRQ, NFRD y NDAC. Para manejar las señales DIO 1-8, DAV, IFC, ATN, REN y EOI se pueden usar manejadores de tipo colector abierto o tres estados, con la excepción de que en una encuesta en paralelo las líneas de datos deberán ser de tipo colector abierto.

c) Tipo de receptor: Se recomienda usar receptores de tipo Schmitt o equivalentes en todas las señales. Las especificaciones para los receptores deben cumplir con:

Estado bajo: Voltaje de entrada $\leq +0.8$ V

Estado alto: Voltaje de entrada $\geq +2.0$ V

1.3.3 Aspecto funcional

El aspecto funcional incluye el uso de líneas específicas para enviar y recibir mensajes, el protocolo para la emisión y recepción de éstos, la relación lógica y la temporización entre las líneas y la serie de 10 funciones de interconexión seleccionables.

El bus transfiere los datos y comandos entre los elementos del sistema a través de 16 líneas. Los elementos están conectados a las señales del bus tal como se muestra en la figura (1.2).

La transferencia de mensajes se hace por medio de las líneas DIO 1-8 de manera asincrónica y controlada por medio de las señales DAV (dato válido), NRFD (no listo para dato) y NDAC (dato no aceptado). Las otras cinco líneas: EOI, REN, SRQ, ATN e IFC, son para el control de la actividad en el bus.

Los elementos integrantes del sistema pueden ser receptores, emisores o controladores. El controlador asigna el papel de cada uno de los otros elementos a través de la línea ATN (atención).

1.4 DESCRIPCIÓN DE LAS SEÑALES ELÉCTRICAS MANEJADAS POR LA NORMA

Para la descripción de las 16 señales eléctricas del bus es conveniente dividir las en tres grandes grupos:

- a) Líneas de datos (8 líneas)
- b) Líneas de control de transferencia de datos (3 líneas)
- c) Líneas de control (5 líneas)

1.4.1 Líneas de datos

El bus de datos es un conjunto de 8 líneas de tipo bidireccional, por medio de las cuales se pueden transmitir o

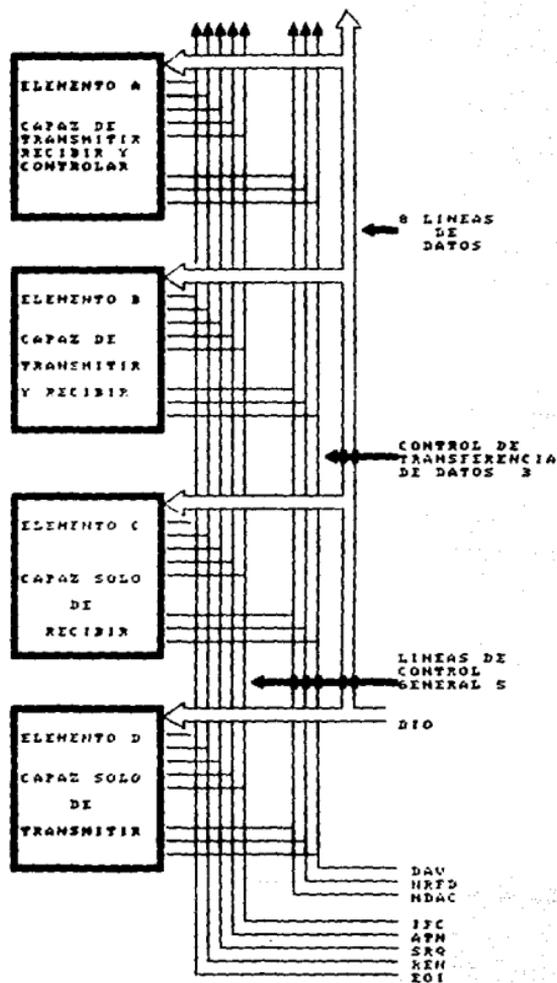


Fig. 1.2. Estructura funcional del Bus GPIB

recibir datos o mensajes. La denominación de las líneas es la siguiente:

DIO_n (Dato de entrada/salida n)

Donde n es el número de línea y puede ser del dato 1 al dato 8.

1.4.2 Líneas de control de transferencia de datos

Para la transferencia de datos es usado un conjunto de tres líneas: 1) NRFD (No listo para dato), 2) DAV (Dato valido) y 3) NDAC (Dato no aceptado). La secuencia de estas señales en el protocolo de comunicación es crítica para el correcto funcionamiento de este. La velocidad de transferencia de los datos está limitada por el dispositivo más lento que intervenga en la comunicación. Una descripción de los diagramas de tiempo se da en la figura 1.3.

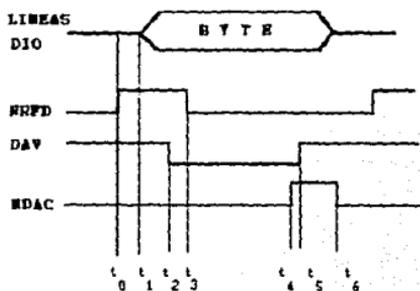


Fig. 1.3. Diagrama de tiempos del protocolo de comunicación

t_0 en este instante todos los receptores permiten a la señal NRFD tomar un estado alto, cuando están listos para más datos. La señal NDAC es mantenida baja por el receptor.

t_1 el controlador detecta que NRFD es alta y NDAC es baja y coloca el siguiente byte de datos en las líneas DIO.

- t2 el controlador lleva la señal DAV al estado bajo cuando sabe que el dato es válido.
- t3 el primer receptor en recibir el dato lleva a bajo la línea NRRFD, después todos los demás receptores lo siguen.
- t4 cuando todos los receptores han recibido el dato, NDAC es puesta en estado alto.
- t5 el controlador detecta la señal NDAC en alto y ajusta la señal DAV al estado alto. Hecho esto se ha terminado la transferencia de un byte y el dato en las líneas DIO puede ser cambiado.
- t6 cuando DAV es ajustada a alto, entonces NDAC es conducida a bajo por el receptor para estar listo para otro byte de datos.

1.4.3 Líneas de control

El bus de control está formado por cinco líneas que se ocupan para mantener un orden dentro del bus. La función de cada señal se da a continuación:

Línea ATN (Atención): Cuando ATN es verdadera, sólo se transmite por siete de las ocho líneas de datos, caracteres ASCII en forma de direcciones de órdenes universales; es decir, que todos los elementos deben atender a la información que aparece en las líneas de datos.

Cuando ATN es falsa, sólo los elementos que han sido previamente direccionados por el controlador envían o reciben datos, en los cuales se puede utilizar cualquier código de 8 bits o menos, mientras que el resto ignora las líneas de datos.

Línea IFC (Limpia interconexión): Únicamente un dispositivo con la función de controlador del sistema puede enviar un mensaje IFC, que consiste en una indicación para limpiar la interconexión; es decir, coloca a todos los dispositivos en un estado predefinido. De esta manera se puede obtener un punto uniforme de inicio en el sistema.

Línea SRQ (Requerimiento de servicio): Los dispositivos que no tengan la función de controlador pueden usar esta línea para indicar al controlador la necesidad de atención. Así, el controlador interrumpe la secuencia de eventos y atiende al dispositivo.

Línea REN (Habilita control remoto): Esta línea solamente puede ser activada por un dispositivo con la función de controlador del sistema y permite conmutar a los dispositivos de un control local a un control a través del bus.

Línea EOI (Fin o Identificación): Esta línea tiene dos funciones en combinación con ATN. Con ATN falsa un emisor puede usar EOI para indicar el final de una transferencia de bytes múltiple. Cuando ATN es verdadera el controlador usa la línea EOI para indicar la ejecución de una secuencia de encuesta (*polling*).

1.5 COMANDOS DE LA NORMA IEEE-488

Las líneas de datos llevan una variedad de mensajes a través del *bus*, siendo éstos: direcciones, datos para configurar un instrumento, resultados de mediciones, comandos universales y datos de estado. La identificación de qué es lo que se está transmitiendo es lograda por el mensaje ATN (Atención). Con la línea en su estado válido se transmiten direcciones o comandos universales cuya codificación está basada en el código ASCII.

La totalidad de los mensajes definidos por la Norma se encuentran resumidos en la tabla 1, cuya equivalencia de tipo y clase se da en la siguiente tabla:

TIPO	U: Mensaje que utiliza una línea del <i>bus</i>
	M: Mensaje que ocupa varias líneas del <i>bus</i>
CLASE	UC: Comandos Universales
	AC: Comandos Direccionados
	AD: Direcciones
	SE: Secundarios
	ST: De Estado
	SH: Protocolo de Manejo de datos
	DD: Dependiente del dispositivo

Comandos Universales: Son comandos que deben ser ejecutados por todos los instrumentos que tengan los circuitos necesarios para hacerlo.

Comandos Direccionados: Son comandos que sólo pueden ser interpretados por él o los dispositivos que fueron direccionados para hacerlo.

Direcciones: Se ocupan estos mensajes para indicar que dispositivo se está seleccionando en el *bus*. Los dispositivos que tienen la función de Emisor y Receptor poseen la misma dirección.

Mensajes Secundarios: Son mensajes que complementan a otro que ha sido transmitido previamente.

Mensajes de Estado: Estos mensajes son los que mandan los dispositivos para indicar su estado interno.

Mensajes de Protocolo: Son los mensajes ocupados por las

TABLA 1 (Continuación)

Abaco- técnico	Nombre del mensaje	LINEAS DEL BUS Y CODIGOS QUE CORRESPONDEN AL MEN- SAJE.			
		C	D	A	M
		T L O	D	A	M
		J A I	I	O R D A E S I R	
		P S O	O	A F A I D R F E	
		D E B 7 6 4 3 2 1		V D C N I D C N	
SDC	Línea dispositivos seleccionados	M AC	Y 2 2 0 0 1 0 0	X X X X X X X X	
SPD	Deshabilita encuesta en serie	M UC	Y 2 2 1 1 0 2 1	X X X X X X X X	
SPE	Habilita encuesta en serie	M UC	Y 2 2 1 1 0 0 0	X X X X X X X X	
SAR	Petición de servicio	D SI	X X X X X X X X	X X X X X X X X	
STB	Byte de estado (Notas 9,7)	M ET	S I S S S S S S	S S S S S S S S	
		B	6 5 4 3 2 1		
TCT	Toca el control	M AC	Y 2 2 0 1 0 0 1	X X X X X X X X	
TAS	Grupo de direcciones de locutor	M AD	Y 1 2 1 1 1 1 1	X X X X X X X X	
UCG	Grupos comandos universales	M UC	Y 0 2 1 1 1 1 1	X X X X X X X X	
UNL	No oír	M AD	Y 2 1 1 1 1 1 1	X X X X X X X X	
UNT	No hablar (Nota 11)	M AD	Y 1 0 1 1 1 1 1	X X X X X X X X	

NOTAS:

- (1) DI-DB Especifican bits de datos dependientes de los dispositivos.
- (2) EI-EB Especifican el código usado por el dispositivo para indicar el mensaje EOS.
- (3) LI-LS Especifican la dirección de receptor del dispositivo.
- (4) TI-TS Especifican la dirección de locutor del dispositivo.
- (5) SI-SS Especifican la dirección secundaria del dispositivo.
- (6) Especifica el sentido del PFR.

S	PPR
0	0
1	1

P1-P3 especifican el mensaje PFR que se envía cuando es llevada a cabo una encuesta en paralelo.

P1	P2	P3	MENSAJE PFR
0	0	0	PFR1
.	.	.	.
.	.	.	.
.	.	.	.
1	1	1	PFR3

- (7) DI-D4 especifican bits que no son decodificados por el dispositivo receptor. Se recomienda que todos los bits en cero.
- (8) SI-S6, SB especifican el estado del dispositivo. D107 es usado para el mensaje EOS.
- (9) La fuente de mensaje en la línea ATN es siempre la función C, mientras que los mensajes en las líneas D10 y EDI son habilitados por la función T.
- (10) La fuente de los mensajes en la línea ATN y EQ1 es siempre la función C, mientras que la fuente de los mensajes en las líneas D10 es siempre la función PP.
- (11) Este código es proporcionado para uso del sistema.

funciones de Protocolo Emisor y Protocolo Receptor para la transmisión de datos.

Mensajes Dependientes del Dispositivo: Son los mensajes de datos enviados por el Emisor al Receptor, después de que el Controlador les ha cedido el control del *bus*.

1.6 FUNCIONES DE LA INTERFAZ

La Norma IEEE 488 define 10 funciones del sistema de interconexión. Un dispositivo, para ser compatible con la Norma, debe ser capaz de interpretar las señales y de generar la secuencia de acciones necesarias. Normalmente, los dispositivos no responden a todas las funciones y solamente realizan las específicas de su función. La descripción de cada una de estas funciones se da a continuación:

CONTROLADOR : La función del controlador permite a un elemento actuar como controlador y enviar al *bus* las direcciones primarias y secundarias y las órdenes universales con dirección. También le permite responder a la línea SRQ y dirigir una llamada en serie o en paralelo para determinar el estado de cada elemento.

El controlador puede en cualquier momento validar la línea ATM e interrumpir la transferencia de datos que se estuviera realizando en ese instante.

En un momento determinado, sólo puede haber un dispositivo ejerciendo las funciones de controlador. El resto de los dispositivos capaces de ejercer esta función deben mantenerla en estado inactivo

EMISOR : Esta función proporciona al dispositivo la capacidad de transmitir datos (incluyendo los datos de estado durante una secuencia de *polling*) a través de la interconexión a otros dispositivos. Para ejercer esta función, el dispositivo debe ser direccionado previamente como emisor por el controlador. Todo dispositivo emisor tiene asignada una dirección que le diferencia de los otros dispositivos emisores del *bus*. Esta dirección es llamada dirección de locutor y puede estar formada de uno o dos bytes. En el caso de que se forme de dos bytes la función es llamada de Emisor Extendida y el proceso de direccionado se realiza en dos fases.

RECEPTOR: La función de receptor proporciona a un dispositivo la capacidad de recibir datos provenientes de un dispositivo (incluyendo datos de estado) a través de la interconexión. La

función de receptor solamente puede ser ejercida si el controlador previamente seleccionó al dispositivo como receptor. Esta dirección es conocida como dirección de receptor y también puede ser transmitida en uno o dos bytes. En este último caso la función es conocida como Receptor Extendido y el proceso de direccionado se realiza en dos fases.

REQUERIMIENTO DE SERVICIO: Esta función permite al dispositivo solicitar asincrónicamente al controlador el uso del *bus*. Para esto activa la señal SRQ del *bus*, y la mantiene en este estado hasta que el controlador notifique al dispositivo que ha recibido la solicitud de servicio.

CONTROL REMOTO O LOCAL: Esta función permite seleccionar al dispositivo. La función indica al dispositivo si debe usar la información que viene del panel frontal, control local, ó utilizar la recibida a través de la interconexión, control remoto.

LIMPIADO DE DISPOSITIVO: Esta función permite limpiar (inicializar) a un dispositivo individualmente o como parte de un grupo de ellos. El grupo puede ser un subconjunto o la totalidad de los dispositivos direccionables conectados en el sistema.

DISPARO DEL DISPOSITIVO : Esta función permite poner en marcha las funciones básicas del dispositivo, individualmente o como parte de un grupo de ellos. El grupo puede ser un subconjunto o la totalidad de los dispositivos direccionables conectados en el sistema.

ENCUESTA EN PARALELO (PPE): Esta función proporciona al dispositivo la capacidad de presentar un bit de estado al controlador sin haber sido direccionado como emisor. Las líneas DIO son usadas para llevar los bits de estado de los dispositivos durante una encuesta en paralelo. Esto permite al controlador discernir entre ocho dispositivos preseleccionados, uno por cada línea de datos, con una sola pregunta. Los bits 1 al 3 del mensaje PPE (encuesta en paralelo) enviado por el controlador designan la línea de datos que el instrumento debe hacer válida si la situación de su estado interno (1 ó 0) coincide con el bit 4 del mensaje.

PROTOCOLO EMISOR: Esta función permite al dispositivo enviar un dato cualquiera por el *bus*, ya sea parte de un mensaje o una respuesta a una orden del controlador. Cabe aclarar que la función de emisor capacita al dispositivo para adquirir el control del *bus* de datos, pero se requiere de esta función para controlar el protocolo de las señales, que asegura la llegada del dato transmitido a todos los dispositivos que fueron direccionados para aceptarlo.

PROTOCOLO RECEPTOR: Esta función permite al dispositivo recibir los datos transmitidos por el *bus*. Esta función está relacionada íntimamente con la función de protocolo emisor. Dado que la

transferencia de datos es asincrónica y que puede haber más de un dispositivo recibiendo el dato. el protocolo receptor permite al elemento más lento del sistema obligar a todos los demás integrantes del sistema a esperar a que esté listo para recibir más datos.

Capítulo 2

LA COMPUTADORA PERSONAL XT

DESCRIPCION GENERAL DE UNA PC XT

En la fig. 2.1 se muestra la tarjeta principal de una computadora (PC) XT. En ella podemos distinguir la tarjeta principal y las ranuras de expansión, en estas últimas se pueden conectar tarjetas adicionales.

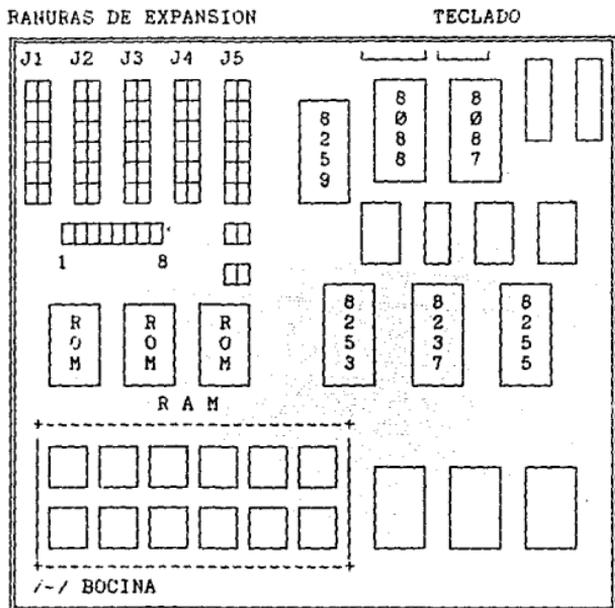


Fig. 2.1. Visión general de una tarjeta para computadora PC XT

Como se puede apreciar, contiene varios circuitos de memoria RAM cuya capacidad total puede variar, siendo el más común un total de 640 Kb ó 640 Kb. De memoria ROM contiene, por lo regular, 40.960 bytes, en los que se incluye un intérprete de BASIC y rutinas especiales para hacer la interfaz con dispositivos de E/S tales como: teclado, pantalla, disco, comunicación serial, impresoras en paralelo, etc..

A un lado de los circuitos de memoria se encuentra el circuito integrado (CI) 8253, que es un circuito contador/temporizador que permite a la PC mantener la hora y la fecha aunque sea apagada. Este circuito también produce señales que permiten manejar a la bocina y producir diferentes tonos.

Junto al CI 8253 se encuentra el controlador de Acceso Directo a Memoria, CI 8237, que permite a los dispositivos tales como las unidades controladoras de disco la transferencia de datos desde y hacia la memoria de la computadora, sin hacer uso del microprocesador. De esta manera, el microprocesador puede realizar otra tarea mientras la transferencia se lleva a cabo. Además uno de los cuatro canales del DMA se conecta con uno de los canales del CI 8253 para refrescar la memoria dinámica RAM.

Al lado del circuito DMA encontramos el CI 8255, que es un circuito que contiene varios puertos paralelos, y es ocupado para leer los interruptores de la tarjeta que determinan que tanta memoria se ha instalado en la PC, así como el número de manejadores de disco y el tipo de pantalla que se está usando. Otro de los puertos es usado para la lectura del teclado.

En la parte superior del diagrama encontramos al CI 8259A, el cual es un circuito controlador de interrupciones de 8 canales. A este circuito se conectan varios dispositivos que pueden acceder al microprocesador mediante solicitudes de interrupción.

En la parte superior derecha encontramos al microprocesador 8088 ó 8086, dependiendo el modelo, y ambos son totalmente compatibles en *software* variando en que el *bus* externo de datos del 8086 es de 16 bits y el del 8088 es de 8 bits.

Junto al CI 8088 se encuentra una zona vacía, en donde se puede colocar el coprocesador de punto flotante CI 8087, el cual está diseñado para realizar operaciones matemáticas a gran velocidad.

Junto a estos procesadores se encuentra una fuente de poder conmutada, la cual proporciona la alimentación necesaria a todos los circuitos y es capaz de alimentar hasta con 1 ampere de corriente cada una de las ranuras de expansión.

Por último, encontramos las ranuras de expansión, que pueden variar en número de un modelo a otro, en donde se conectan por lo regular una tarjeta controladora de disco, una tarjeta para manejar el video y quizás una tarjeta de memoria RAM con un

puerto serial RS232, un puerto paralelo y reloj con respaldo de batería. Las demás ranuras quedan para tarjetas de aplicación que el usuario diseñe o necesite comprar.

2.1 EL MICROPROCESADOR 8088

El μ P 8088 es un circuito integrado que se fabrica utilizando tecnología HMOS con un total de 40 terminales de conexión. Dos de estas terminales corresponden a la tierra del circuito y una más corresponde a la polarización de 5 volts, requerida por el circuito para trabajar

Este microprocesador posee un *bus* externo de 8 bits, pero internamente es idéntico al microprocesador de 16 bits 8086. Su capacidad de direccionamiento es de un Megabyte de memoria ya que posee un *bus* de direcciones de 20 líneas. Para evitar un número excesivo de líneas se multiplexan las líneas de datos con las de direcciones, de tal manera que los 8 bits de datos comparten terminales con los 8 bits menos significativos de la dirección.

Además, posee características especiales que le permiten trabajar en ambientes de multiproceso y procesamiento en paralelo. Una explicación más detallada de sus propiedades y funciones se da a continuación.

2.1.1 Arquitectura de la CPU

Este microprocesador internamente esta dividido en dos sub-procesadores: la unidad de ejecución (EU) y la unidad interfaz del *bus* (BIU). La unidad de ejecución es la encargada de codificar y ejecutar las instrucciones, mientras que la unidad de Interfaz del *bus* es la que realiza las funciones necesarias para localizar las instrucciones y transferir los datos entre los registros y el *bus*.

El CI 8088 utiliza el concepto de "Cola de instrucciones" con el fin de aumentar la velocidad de proceso. Existe en el circuito un área especial que recibe el nombre de cola de instrucciones, que es la que guarda los bytes de la instrucción. Cuando la BIU localiza en memoria un byte de código de máquina, lo coloca en la cola que tiene 4 bytes de longitud en el CI 8088 y 6 bytes en el CI 8086.

Este sistema posee la ventaja de que cada instrucción puede

ser obtenida de la memoria mientras otras se están ejecutando, reduciéndose en consecuencia el tiempo de proceso. Por ejemplo, las instrucciones que incluyen directamente al dato, se ejecutan casi a la misma velocidad que aquellos que necesitan datos de los registros de la CPU. Este sistema también es útil cuando se trabaja con memorias lentas. Esta repartición de trabajo entre la unidad de ejecución y la unidad interfaz del bus es lo que hace que el rendimiento del CI 8088 pueda ser comparable a un microprocesador de 16 bits.

2.1.2 MODOS DE OPERACION

El CI 8088 puede operar en dos modos : el modo máximo y el modo mínimo. El modo mínimo se ocupa en sistemas en donde el CI 8088 genera directamente todas las señales necesarias para el control del bus . El modo máximo es ocupado en sistemas que regularmente incluyen dos o más procesadores.

El modo de operación se define en función del valor lógico de la terminal MN/MX del circuito. Cuando esta terminal se conecta a tierra el CI 8088 es usado en configuración de modo máximo y cuando la terminal se conecta a 5 volts se selecciona el modo mínimo.

Cabe recalcar que en el modo máximo, el 8088 necesita de circuitos adicionales para generar todas las señales de control necesarias en el bus.

2.1.3 Registros internos

El CI 8088 tiene un total de 14 registros de 16 bits, los cuales pueden ser divididos en cuatro grupos funcionales diferentes, como se muestra en la figura 2.2. Varios de los registros tienen funciones especiales cuando se usan con ciertas instrucciones, aunque también pueden ser usados como registros de propósito general.

Se tienen 4 registros de datos de 16 bits (AX, BX, CX, DX), donde la letra A representa Acumulador, la B representa Base, la C representa Contador, la D representa Datos, y la letra X indica que se trata de un registro extendido. Estos registros a su vez pueden ser divididos en 8 registros de 8 bits (AH, AL, BH, BL, CH, CL, DH, DL), donde la H representa que es la parte alta del registro de 16 bits, y la L indica que es la parte baja, es decir, la menos significativa.

Los registros de datos se comportan idénticamente para la mayoría de las instrucciones, pero hay algunas diferencias. La primera de ellas es que algunas instrucciones se ejecutan más rápido y/o tienen una instrucción de lenguaje de máquina más corto cuando el registro AX o AL es usado. También, el registro AX ó AL es usado para todas las instrucciones de entrada/salida. El registro BX se utiliza especialmente como registro base para los direccionamientos.

AX	AH	AL	ACUMULADOR
BX	BH	BL	BASE
CX	CH	CL	CONTADOR
DX	DH	DL	DATOS
	SP		PUNTERO DE PILA
	BP		PUNTERO DE BASE
	SI		INDICE FUENTE
	DI		INDICE DESTINO
	IP		PUNTERO DE INSTRUCCION
	FLAGS	IFLAGS I	INDICADORES DE ESTADO
	CS		SECTORIO DE CODIGO
	DS		SECTORIO DE DATOS
	SS		SECTORIO DE PILA
	ES		SECTORIO EXTRA

Fig. 2.2. Conjunto de registros del CI 8088/8086

El registro BX se utiliza especialmente como registro base para los direccionamientos.

El registro CX se usa a menudo para almacenar datos, para contar iteraciones y para desplazamientos y rotaciones.

El registro DX tiene un uso especial, y es el de contener las direcciones de los puertos de entrada/salida para las instrucciones de entrada/salida.

Otros 4 registros sirven como apuntadores y de índice (SP, BP, SI y DI), de los cuales ninguno puede dividirse como en el caso anterior, pero pueden usarse como registros de propósito general. El registro SP es el apuntador de pilas y el registro BP es el apuntador base. Ambos son usados típicamente para lectura y escritura del *STACK*. SI es el registro índice fuente ;

DI es el registro indice destino, estos registros son de propósito específico y son utilizados como partes de los modos de direccionamiento.

Se cuenta con un registro de indicadores (banderas) de 16 bits, los cuales son bits que indican el estado interno de las banderas. Estas banderas son: bandera de cero (ZF), bandera de signo (SF), bandera de paridad (PF), bandera de acarreo (CR), bandera auxiliar (AF), bandera de direccion (DF), bandera de interrupción (IF), bandera de overflow (OF) y bandera de desvío (TF).

Un registro apuntador de instrucciones (IP). Este registro es usado para controlar el flujo propio del programa y contiene la dirección de la siguiente instrucción a ser ejecutada en memoria.

Se cuenta con cuatro registros segmento : CS, DS, SS y ES. Estos son los registros segmento de código, segmento de datos, segmento de pila y segmento extra respectivamente. Estos registros son usados en el concepto de segmentación de memoria.

2.2 DESCRIPCION DE LAS SEÑALES ELECTRICAS MANEJADAS EN LA RANURA DE EXPANSION DE UNA PC-XT

La descripción del bus XT y la de sus señales es necesaria debido a que la tarjeta va a quedar instalada en una de las ranuras de expansión de la PC.

El bus XT fue diseñado en un principio para trabajar con el microprocesador 8088 de Intel y fue dispuesto con los 8 bits de datos y las 20 líneas de direcciones del microprocesador. Su uso es básicamente para aplicaciones monoprocesador, pero también tiene las facilidades para efectuar las operaciones de controladores de DMA. Consta de cuatro canales de DMA, tres de los cuales están disponibles en el conector del bus y uno es usado para el refresco de la memoria dinámica RAM. Incluye 8 señales de interrupción, de las cuales 6 son accesadas en el conector y 2 son utilizadas por el sistema.

Los conectores son del tipo hembra para fin de tarjeta, de 62 contactos, numerados del A1 al A31 y del B1 al B31. La distribución de señales se muestra en la figura 2.3.

GND	B1	A1	-I/O CH CK
RESET DRV	B2	A2	D7
+5 V	B3	A3	D6
IRQ2	B4	A4	D5
Y	B5	A5	D4
DRQ2	B6	A6	D3
-12V	B7	A7	D2
RESERVED	B8	A8	D1
+12 V	B9	A9	D0
GND	B10	A10	I/O CH RDY
-MEMW	B11	A11	AEN
-MEMR	B12	A12	A19
-IOW	B13	A13	A18
-IOR	B14	A14	A17
-DACK3	B15	A15	A16
DRQ3	B16	A16	A15
-DACK1	B17	A17	A14
DRQ1	B18	A18	A13
-DACK0	B19	A19	A12
CLOCK	B20	A20	A11
IRQ7	B21	A21	A10
IRQ6	B22	A22	A9
IRQ5	B23	A23	A8
IRQ4	B24	A24	A7
IRQ3	B25	A25	A6
-DACK2	B26	A26	A5
T/C	B27	A27	A4
ALE	B28	A28	A3
+5 V	B29	A29	A2
OSC	B30	A30	A1
GND	B31	A31	A0

Fig. 2.3. Distribución de señales en el Bus PC XT

2.3 CLASIFICACION DE LAS LINEAS ELECTRICAS DEL BUS XT

Para la descripción de las líneas del bus es conveniente agruparlas en la forma siguiente:

- a) Líneas de datos
- b) Líneas de control de memoria y E/S
- c) Líneas de interrupción
- d) Líneas de arbitraje
- e) Líneas de error y sincronía
- f) Líneas de espera e inicialización
- g) Líneas de alimentación

2.3.1 Líneas de datos

Las líneas D0-D7 (data bits) son 8 líneas bidireccionales utilizadas para la transferencia de datos en el bus.

2.3.2 Líneas de direcciones

Las líneas A0-A19 (Address bits) son 20 líneas de direcciones del microprocesador y se ocupan para direccionar la memoria o los dispositivos de E/S.

2.3.3 Líneas de control de memoria y puertos de E/S

ALE (Address Latch Enable). Señal de salida que sincroniza la dirección existente en las líneas de direcciones. La sincronía se realiza en el flanco de bajada de la señal.

MEMR (Memory Read Command). Es una señal que puede generar la CPU ó un controlador de DMA, indicando que se está realizando una operación de lectura en memoria y que por lo tanto el dispositivo debe colocar el dato correspondiente a la dirección que se desea leer en las líneas del bus. Su estado activo corresponde a un cero lógico.

MEMW (Memory Write Command). Al igual que la anterior puede ser generada por la CPU o por un controlador DMA y es usada para indicar una operación de escritura en memoria.

IOR (I/O Read Command). Cuando se quiere llevar a cabo una función de lectura de un puerto de E/S, esta señal es llevada a un "0" lógico por el microprocesador o por un controlador de DMA.

IOW (I/O Write Command). Cuando esta señal es un "0" lógico, indica que la PC está enviando datos a un circuito de E/S. Puede ser generada por la CPU o por un controlador de DMA.

2.3.4 Líneas de interrupción

IRQ2-IRQ7 (Interrupt Request). Son 6 líneas de entrada que indican al microprocesador que un dispositivo requiere de su atención. Estas señales permanecen activas hasta que el microprocesador ejecute la correspondiente subrutina de atención a la interrupción. La señal IRQ7 corresponde a la interrupción de más baja prioridad.

2.3.5 Líneas de arbitraje

DRQ1-DRQ3 (DMA Request). Son tres canales asincronos utilizados por los dispositivos para hacer uso del DMA. Estas señales se deben mantener activas hasta que se les conceda el bus.

DACK0-DACK3 (DMA Acknowledge). Son cuatro líneas, de las cuales 3 se ocupan para responder a las correspondientes peticiones de servicio de DMA anteriores. La línea restante, DACK0, se utiliza para refrescar la memoria RAM dinámica.

AEN (Address Enable). Esta línea es usada para informar al sistema que la dirección, los datos y las líneas de control están siendo controlados por el microprocesador. Cuando esta señal es un "0" lógico la dirección de salida en las líneas A0-A19 es válida y puede ser usada en la decodificación de direcciones.

T/C (Terminal Count). Esta línea de salida emite un pulso cada vez que el valor preseleccionado del contador de cualquier DMA es alcanzado.

2.3.6 Líneas de error y sincronía

I/O CHECK (I/O Channel Check). Es usada para indicar al microprocesador que ocurrió un error de paridad en una operación en la memoria o un dispositivo de E/S.

OSC (Oscillator). Es una señal de reloj con frecuencia de 14.31818 Mhz.

CLK (Clock). Es una señal de salida que constituye el reloj del sistema y representa 1/3 de la frecuencia del oscilador.

2.3.7 Líneas de espera e inicialización

I/O RDY (I/O Channel Ready). Esta señal es activada por dispositivos lentos cuando son accedidos a través del bus. La señal I/O RDY no debe mantenerse a nivel bajo por un periodo superior a 10 ciclos de reloj.

RESET (Reset). Es utilizada para inicializar la lógica del sistema. Se sincroniza con el flanco de bajada del reloj del sistema y se considera activa cuando esta en estado lógico "1".

2.3.8 Voltajes de alimentación.

Se cuenta con cuatro voltajes de alimentación distintos: +5 V, -5 V, +12 V y -12 V.

Capítulo 3

DISEÑO DEL CONTROLADOR IEEE-488

Actualmente existen en el mercado circuitos diseñados especialmente para la construcción de un controlador para dispositivos compatibles con la Norma IEEE-488, cada uno de ellos ofrece el cumplimiento de los comandos establecidos en la Norma, por lo que la utilización de alguno de ellos permitirá un ahorro sustancial en la programación ya que los circuitos generan la totalidad o gran parte de las señales necesarias en el *bus*.

A continuación se da un panorama general de los circuitos investigados a lo largo del presente trabajo y por último se llega a la elección de los componentes que intervendrán en el diseño final de la tarjeta.

3.1 ALTERNATIVAS DEL DISEÑO

3.1.1 Los circuitos integrados 8291A y el 8292

EL CI 8291A es un circuito EMISOR/RECEPTOR GPIB, diseñado para comunicarse con microprocesadores tales como el 8051, el 8080/85 y el 8086/88. Implementa todas las funciones de interfaz definidas en la Norma, excepto aquellas inherentes a las del controlador.

El CI 8291A maneja la comunicación entre el microprocesador y el GPIB. Entre sus capacidades más importantes se incluyen:

- El índice de transferencia de datos programables.
- El protocolo completo de Emisor y Receptor.
- Las funciones de Emisor y Receptor con direccionamiento extendido.
- Las funciones de requerimiento de servicio, limpiado de dispositivo, disparo de dispositivo, encuesta en paralelo y Remoto/Local.
- El manejo de interrupciones.
- La capacidad de manejo de DMA.
- La facilidad en el manejo de transferencias Multi-Byte.

Para implementar las funciones de Controlador es necesario conectar el CI 8291A con el CI 8292, el cual es un Controlador del GPIB. Algunas de sus funciones importantes son:

- Implementar todas las funciones del Controlador.

- Enviar el comando de limpiado de la interfaz (IFC).
- Responder a los requerimiento de servicio.
- Enviar el comando de operación Remoto/Local a los dispositivos.
- Implementar el protocolo completo para la transferencia de Control.

Con estos dos circuitos conectados se implementa una Interfaz IEEE-488 completa con las funciones EMISOR/RECEPTOR/CONTROLADOR, además Intel pone a la disposición el CI 8293 que contiene transceptores especialmente diseñados para conectarse al GPIB.

Si no se desea utilizar el CI 8293 existen algunos transceptores que cumplen con los requisitos establecidos para conectarse al bus, éstos son fabricados por National Semiconductor:

- Los transceptores IEEE-488, DS75160A/DS75161A/DS75162A. Las líneas de datos siempre se manejan con el CI DS75160A y las otras señales se manejan con el CI DS75161A en un sistema con un solo controlador, y con el CI DS75162A en un sistema que contenga varios dispositivos capaces de tomar el control del bus.
- EL Transceptor IEEE-488, DS3666. Para implementar la interfaz con estos circuitos se requieren dos de ellos conectados a través de sus entradas de expansión. A cada dispositivo se le asignan 4 líneas de datos y 4 señales para manejar la totalidad de las 16 líneas.

3.1.2 EL Circuito integrado MC68488

El CI MC68488 es un Adaptador para Interfaz de Propósito General GPIA, distribuido por Motorola, y está diseñado para conectarse directamente al microprocesador MC68000; sin embargo, con hardware adicional, puede adaptarse para ser conectado a otros microprocesadores. Entre las funciones más importantes que puede realizar se encuentran:

- El manejo de los protocolos de Emisor y Receptor.
- Las interrupciones programables.
- La capacidad de hacer encuestas en serie y paralelo (*polling*).
- Se puede operar con un controlador de DMA.
- La selección automática de funciones para minimizar la programación

Para realizar la conexión con el bus, Motorola también fabrica transceptores especialmente diseñados para cumplir con los requerimientos de la Norma. Estos son:

- El CI MC3446A que es un transceptor cuádruple GPIB, del cual se necesitarían 4 unidades para llevar a cabo la interfaz completa.
- El CI MC3448A, el cual integra 4 transceptores para el GPIB y puede manejar la interfaz entre el bus y lógica TTL o CMOS. Cada par manejador/receptor tiene su propia terminal de habilitación independiente a las demás.

3.1.3 EL μ PD 7210

EL μ PD 7210 es un circuito controlador del GPIB fabricado por NEC, con el cual se puede construir un controlador que cumpla los requerimientos de la Norma.

Algunas de sus características más importantes son:

- Su capacidad de manejar las funciones de Controlador.
- Por lo regular requiere de ser conectado junto con FLA's.
- Su capacidad de manejo de DMA.

3.2 Diseño de la Tarjeta

En los puntos anteriores se ha dado una breve explicación de los circuitos que podrían ser la parte central del diseño. Analizando cada una de ellas encontramos que cada uno presenta el inconveniente de que son componentes que no pueden ser conseguidos en el mercado nacional por lo que es necesaria su importación. Este inconveniente aumenta el costo de los componentes y por tratarse de circuitos para un uso específico los proveedores aumentan en gran porcentaje el costo del circuito que en sí ya es elevado. Esto se contrapone con los objetivos iniciales de este trabajo en el que especificamos que el Controlador que se desea construir sea de muy bajo costo.

Por otra parte, podemos darnos cuenta que el alto costo de los componentes se debe en gran parte a la gran cantidad de funciones de la Norma que pueden llevar a cabo; sin embargo, estudiando detenidamente todas las funciones llegamos a la conclusión de que difícilmente en el trabajo se requiere hacer uso de todas ellas y que son sólo algunas las que son ocupadas continuamente.

Por lo que respecta al CI MC68480, que es el más barato de todos tiene la desventaja de requerir de *hardware* adicional para

que pueda ser adaptado al bus de la PC. Esto es por que originalmente esta pensado para trabajar con otro microprocesador.

Otro problema adicional lo constituyen los transeptores, ya que tampoco estos pueden ser conseguidos en el mercado y también es necesaria su importación. Este tipo de transeptores, por estar específicamente diseñados para cumplir con la Norma, también son de un costo alto y se requiere que sean usados si se quiere hacer un uso completo de todas las funciones de los circuitos anteriores.

Con base en esto, decidimos deshechar la idea de utilizar algunos de estos circuitos y hacer un diseño sencillo, pero que a la vez fuera capaz de llevar a cabo las funciones de EMISOR/RECEPTOR/CONTROLADOR, y que al mismo tiempo todos sus componentes se pudieran comprar fácilmente en el mercado nacional.

La manera más fácil de hacer esto es que nosotros mismos generemos todos las señales y protocolos necesarios a través de la programación. En este capítulo se discute todo el diseño referente a la parte de *hardware* y en el siguiente se explicarán todas las rutinas de programación que se necesitan.

3.2.1 El circuito integrado 8255A

Para el manejo de todas las señales eléctricas del bus XT y del bus 488 haremos uso del CI 8255A, diseñado para tener funciones de una interfaz de periféricos programables. La configuración funcional del circuito es hecha por el *software* del sistema. Y debido a que a través de los puertos vamos a tener acceso a las señales, para que puedan ser procesadas por los programas de control de la tarjeta, es necesario dar una explicación del funcionamiento de este circuito.

El diagrama de bloques del CI 8255A se muestra en la figura 3.1 en donde se observa claramente la conexión interna de sus tres puertos, así como las líneas de control.

Grupo de control A y B

La configuración del 8255A se hace a través de la palabra de control, la cual va a contener información sobre el comportamiento del CI 8255A.

Cada uno de los bloques de control (grupo A y grupo B) acepta comandos de la lógica de control y recibe palabras de control del bus interno de datos y manda los comandos apropiados a sus puertos. Cada bloque de control manda los siguientes puertos:

Grupo de control A : Puerto A y Puerto C superior (C7-C4)
 Grupo de Control B : Puerto B y Puerto C inferior (C3-C0)

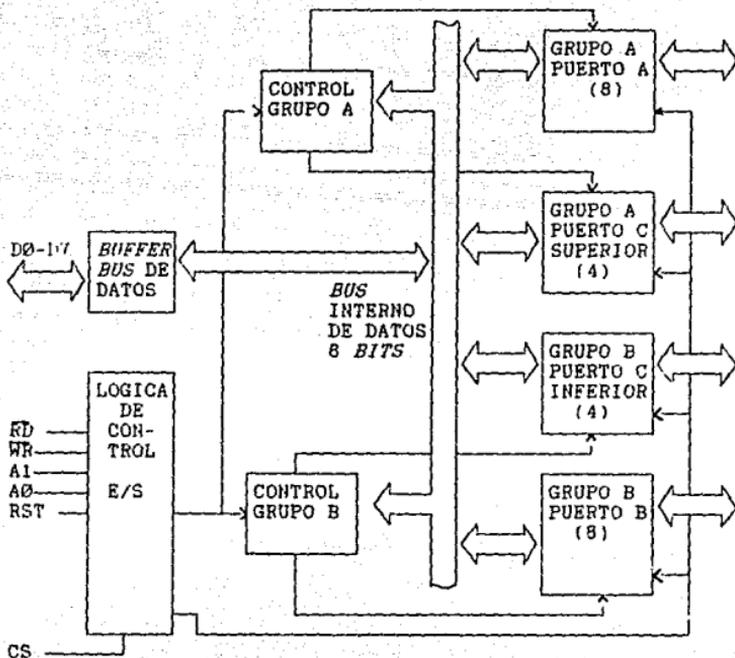


Fig. 3.1. Diagrama de bloques del CI 8255A

Puertos A, B y C

El CI 8255A contiene tres puertos de 8 bits que pueden ser configurados en una amplia variedad de características funcionales.

Puerto A : 8 bits con salida *latch*/buffer y entrada tipo *latch*.
Puerto B : 8 bits con salida/entrada *latch*/buffer
Puerto C : Este puerto puede ser dividido en dos puertos de 4 bits. Cada puerto de cuatro bits contiene un *latch* y puede ser usado para salidas de señales de control y entradas de información de estado en conjunción con los puertos A y B.

a) Descripción de las señales de control del CI 8255A

- Líneas de datos

El bus de datos es un bus de 8 bits de tres estados que se ocupa para hacer la interfaz con el bus del sistema. El dato es transmitido o recibido por el *buffer* después de la ejecución de una instrucción de entrada o salida efectuada por la CPU. Las palabras de control y la de información de estado también son transmitidas por este conducto.

- Líneas de control y de lectura/escritura

La función de este bloque es manejar todas las transferencias internas o externas de datos o de palabras de control y estado.

CS (Chip Select). Un cero lógico en esta línea habilita la comunicación entre el CI 8255A y la CPU.

RD (Read). Un cero lógico en esta línea le indica al CI 8255A que puede enviar los datos o la información de estado a la CPU.

WR (Write). Un cero lógico en esta línea habilita al CI 8255A para que la CPU escriba datos o una palabra de control en sus puertos o en su registro interno.

A0 y A1 (Selección del puerto 0 y selección del puerto 1). Estas señales junto con las señales RD y WR controlan la selección de uno de los tres puertos o el registro de la palabra de control. Se conectan normalmente a los bits menos significativos del bus de direcciones.

Reset. Un uno lógico en esta línea limpia el registro de control y los puertos se configuran en modo de entrada.

b) Modos de Operación

Hay tres modos de operación que pueden ser seleccionados a través de la programación. Las características de cada modo se ilustran en la figura 3.2

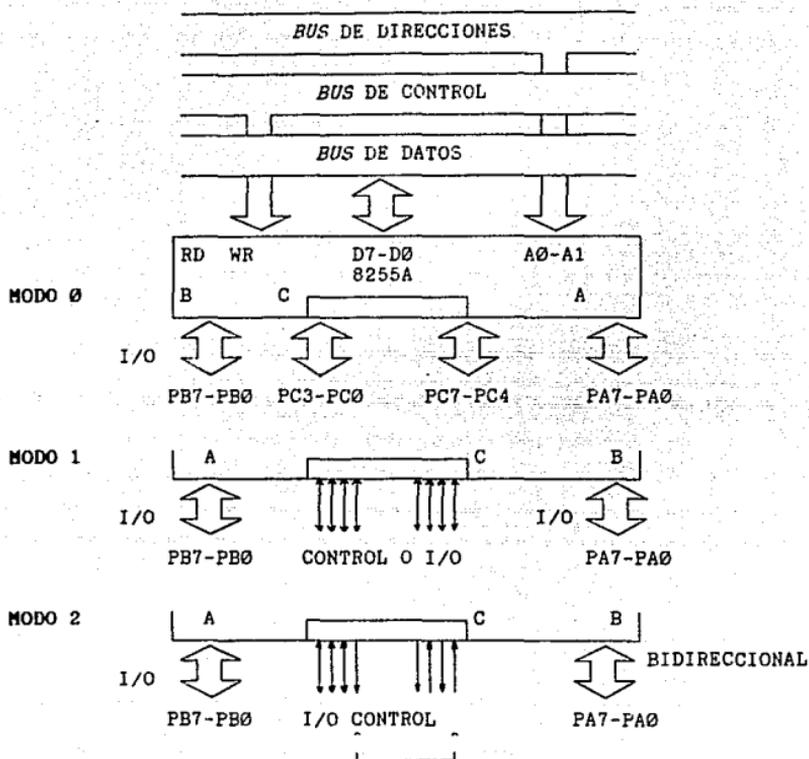


Fig. 3.2. Modos de configuración del CI 8255A

Modo 0

Esta configuración proporciona operaciones de entrada/salida simples para los tres puertos. Ningún protocolo es requerido, el dato es simplemente leído o escrito desde un puerto especificado. Sus características principales son:

- Dos puertos de 8 bits y dos puertos de 4 bits.
- Cualquier puerto puede ser entrada o salida.
- Las salidas son del tipo latch.
- Las entradas no son del tipo latch.
- Se tienen 16 configuraciones de entrada/salida diferentes.

Modo 1

Esta configuración proporciona un medio de transferir datos de entrada/salida hacia o desde un puerto especificado junto con señales de habilitación o de protocolo. El puerto A y el puerto B usan las líneas del puerto C para generar o aceptar estas señales. Sus características básicas son:

- Grupo A y Grupo B: cada uno controla un puerto de 8 bits de datos y un puerto de 4 bits de control/datos
- Los puertos de 8 bits pueden ser de entrada o salida, las cuales son de tipo *latch*
- El puerto de 4 bits es usado para control y estado del puerto de datos.

Modo 2

Con esta configuración se tiene un medio para comunicarse con un dispositivo periférico o un *bus* de 8 bits tanto para transmitir como recibir datos. Las señales de protocolo son provistas para mantener un flujo apropiado en el *bus* de manera similar al modo 1. También se cuenta con generación de interrupción y funciones de habilitado/deshabilitado. Sus características son:

- Usado solamente con el Grupo A.
- El puerto de *bus* bidireccional (puerto A) de 8 bits y un puerto de control de 5 bits (puerto C).
- El puerto de 5 bits es usado como de control y de estado del puerto A.

3.2.2 Decodificación de memoria

El CI 8086/8088 tiene 65,535 direcciones de salidas disponibles para puertos de entrada/salida, las cuales se generan con la combinación de los valores de las líneas A0 a la A15. De todas estas direcciones la computadora tiene varias reservadas para uso exclusivo de funciones propias, por lo que es necesario consultar el mapa de entrada/salida del equipo que vayamos a utilizar.

La tarjeta ocupa 4 direcciones consecutivas, debido a que utiliza dos líneas de direcciones para acceder sus registros internos, y debe cuidarse que éstas no interfieran con el mapa de entrada/salida de la PC. En base a esto se decidió diseñar un circuito que fuera capaz de decodificar el número total de direcciones y que el usuario definiera la dirección que activa la tarjeta. La manera de lograr esto es haciendo uso de un comparador, cuyas entradas sean la dirección del usuario y la

dirección mandada por el microprocesador, de esta manera cuando sean iguales se manda la señal de que la tarjeta necesita ser accesada.

Para hacer esto se ocuparon los CI's 74LS85, que son comparadores de magnitud de cuatro bits, conectados en cascada para hacer la comparación de una palabra de 14 bits. Estos circuitos dan una señal de mayor, menor e igual al comparar dos palabras. Una de estas va a ser la definida por el usuario y la otra la generada por el sistema. La palabra definida por el usuario se va a alimentar a través de interruptores de la siguiente manera:

Se cuenta con 2 interruptores de 8 posiciones, cada uno se identifica en el diagrama como DP-H y DP-L. Cada interruptor corresponde a un bit de la palabra de 14 bits. Cuando se ponen en la posición ON se pone el bit en "0", cuando se ponen en la posición OFF se coloca el bit correspondiente en "1". El interruptor DP-L maneja los bits correspondientes a las líneas A2 a la A9 y el interruptor de DP-H maneja los que corresponden a las líneas de A10 hasta A15. La manera de formar la palabra consiste únicamente en ajustar los interruptores en la posición adecuada para formar el equivalente binario de la dirección en la cual queremos que quede alojada la tarjeta.

La palabra generada por el microprocesador se toma directamente de las líneas del bus XT y es alimentada a nuestro circuito comparador junto con la palabra generada por el usuario a través de los interruptores. Se manejan las líneas A2 hasta A15, ignorándose las líneas A16 hasta la A19. Por lo que respecta a las líneas A0 y A1 son las usadas para acceder los puertos del CI 8255A.

En el momento que el microprocesador coloca una dirección igual a la formada con los interruptores, el arreglo de comparadores genera un "1" lógico (5 volts) indicando que la tarjeta necesita ser accesada. Esta señal es válida durante cuatro direcciones consecutivas, a partir del valor del equivalente binario formado por los interruptores.

3.2.3 Manejo de las líneas de control del CI 8255A

Necesitamos activar básicamente 4 señales del CI 8255A, con las cuales vamos a leer, escribir, activar la tarjeta y reinicializar el circuito. La forma de manejar estas señales se explica a continuación.

Para activar la señal CS vamos a ocupar la señal que indica que la dirección de la tarjeta está siendo enviada por el microprocesador, tomada del circuito decodificador de memoria, y la señal del bus XT, AEN. Esta señal es "0" cuando la dirección que está en el bus es válida. Conociendo que el circuito decodificador de memoria entrega una salida alta cuando reconoce que se está enviando la dirección de la tarjeta, podemos construir la tabla siguiente:

AEN	Decodificador	Salida
0	0	1
0	1	0
1	0	1
1	1	1

De la tabla se observa que la señal de salida es verdadera únicamente cuando se tiene una dirección válida y el circuito decodificador indica que es la dirección esperada. Del análisis de la tabla vemos que tenemos que invertir la salida del circuito decodificador y después hacer una OR con la señal AEN. La salida de la compuerta es la señal que habilita al CI 8255A y al transceptor de datos CI 74LS245.

La señal de escritura se obtiene fácilmente con la señal IOW del bus XT, la cual por seguridad únicamente es válida con la señal que habilita la tarjeta a través de una OR, antes de ser conectada a la señal WR del CI 8255A.

La señal de lectura se obtiene de la misma forma que la anterior pero ocupando la señal IOR del bus XT. Esta señal es la que controla la señal de lectura RD del CI 8255A y la señal de control de dirección de datos DIR, del CI 74LS245.

Por último la señal de *Reset* se conecta a la señal de *Reset* del sistema que se encuentra en el bus XT. De esta manera cualquier reinicialización del sistema afecta también a la tarjeta.

3.2.4 Manejo de las líneas de datos del bus XT y del bus 488

Se cuenta con 8 líneas de datos bidireccionales, de salida cuando actúa como Controlador o Emisor y de entrada cuando se funciona como Receptor. En este punto hay que tomar en cuenta que tenemos las líneas de datos que se comunican con el bus XT y las líneas que se comunican con el bus 488. Para el primer caso usamos el CI 74LS245, el cual es un transceptor octal con salida de 3 estados. Este circuito está conectado por medio de su bus A a los 8 bits de datos del bus XT y al puerto A del CI 8255A a

través del bus B. De esta manera tenemos la función de flujo bidireccional necesaria y sólo tenemos que preocuparnos por manejar adecuadamente el habilitador del circuito y la terminal de control del flujo de dirección con el fin de escribir o leer datos del bus.

El problema de habilitar el circuito se resuelve fácilmente si se ocupa la misma señal que habilita al CI 8255A, como se había mencionado anteriormente.

La dirección del flujo de datos en el CI 74LS245 se controla por medio de la señal DIR. Si DIR toma el valor de cero lógico, el flujo de datos va del bus B (Puerto A del CI 8255A) hacia el bus A (datos del bus XT). En el caso contrario los datos fluyen del bus A hacia el bus B. Utilizando la misma señal que habilita la señal de lectura del CI 8255A, se logra que el CI 74LS245 esté normalmente leyendo los datos del bus 488 hacia la tarjeta, evitando así activar algún dispositivo por equivocación.

Para el caso del manejo de las líneas de datos del bus 488 la situación se complica porque el manejo debe cumplir con los requisitos establecidos en el capítulo 1. Además, tenemos que hacer uso de circuitos de fácil adquisición que emulen el comportamiento de aquellos especialmente diseñados para este fin.

Con base en lo establecido en la Norma se plantea la utilización del arreglo de la figura 3.3 para cada línea de datos:

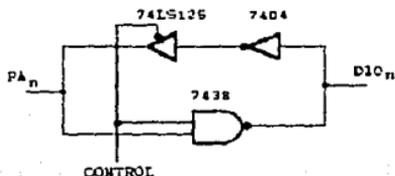


Fig. 3.3. Transceptor utilizado para los datos del bus-488

Su funcionamiento es sencillo y se explica a continuación.

Todas las líneas de datos las vamos a obtener del puerto A del CI 8255A. Para el caso de salida hacia el *bus* 488 utilizamos una AND con salida de tipo Colector Abierto (*Open Collector*), tal como es recomendado en la Norma, y en el caso de recepción de datos del *bus* 488 un inversor CI SN7404 y el *buffer* CI SN74LS125 con salida de tres estados. Es decir, todo lo que necesitemos mandar hacia el *bus* 488 tiene que pasar por el CI 74LS245, el Puerto A y el arreglo de la fig. 3.3. En el caso de recepción de información del *bus* 488, ésta pasa por el arreglo de adaptación, el Puerto A y el transceptor 74LS245, hasta llegar al *bus* XT, de donde se toma para ser procesada.

De lo anterior establecemos que el Puerto A tiene que ser programado para funcionar en forma bidireccional; además debemos tener líneas para controlar el flujo de información del circuito de adaptación por medio de la señal de control indicado en la fig. 3.3. Aprovechando que el Puerto C puede ser dividido en dos, de 4 líneas cada uno, vamos a tomar la parte menos significativa para generar esta señal.

Siguiendo la tabla de verdad de una compuerta AND, establecemos que cuando haya un "1" en la señal de control activamos la compuerta, para que el dato presente en la otra entrada aparezca en el *bus* 488, y al mismo tiempo, como esta señal la conectamos al control del CI 74LS125, lo ponemos en alta impedancia en el mismo momento y así evitamos cualquier falla por realimentación. En este momento el circuito se encuentra funcionando como salida.

Cuando mandamos un "0" en la señal de control activamos al CI 74LS125 y ponemos en corte al transistor de salida de la compuerta, para que no interfiera en el flujo de datos. Así el circuito esta funcionando como entrada.

Se debe notar que existe un inversor que al parecer no interviene en la lógica explicada anteriormente; sin embargo, se debe recordar que el *bus* 488 trabaja con lógica negada, por lo que es muy comodo hacer este arreglo para evitarse el estar trabajando con instrucciones de complemento en la programación.

Como se explicó anteriormente, el flujo de datos lo establecen la señal de control que se manda a través del Puerto C, pero hay que notar que se tienen que manejar las 8 señales de datos, que requieren de un arreglo similar cada uno, por lo que es conveniente generar la señal e invertirla para alimentar a 2 inversores y obtener 2 señales de control iguales, llamadas I/O1

e I/O2 en el diagrama del controlador, que manejen la mitad de las líneas cada una y evitan el exceso de carga que pueden generar fallas difíciles de detectar.

3.2.5 Manejo de las líneas de protocolo del bus 488

Para el manejo de las líneas de protocolo la situación es un poco diferente, en el sentido de que no todas las señales llevan el mismo sentido al mismo tiempo, como en el caso de las líneas de datos. En este caso el sentido de las líneas depende de como este funcionando la tarjeta. Una revisión de las funciones nos conduce a la siguiente conclusión, respecto a la dirección de las señales de protocolo:

Quando funciona como Controlador y Emisor

DAV y EOI Salidas
NRFD y NDAC Entradas

Quando funciona como Controlador y Receptor

DAV y EOI Entradas
NRFD y NDAC Salidas

El principal problema que se presenta es que el CI 8255A no permite la programación individual de cada una de las líneas, por lo que se decidió manejar estas líneas por los Puertos B y C. Por el Puerto B se van a manejar cuando estén funcionando como salidas y por la parte más significativa del Puerto C se van a leer cuando estén funcionando como entradas. Como el Puerto C está dividido en dos, la parte más significativa va a estar programada como entrada para leer las cuatro señales de protocolo.

El circuito para acoplar estas señales al bus es básicamente el mismo que en el caso del manejo de las líneas de datos. Este circuito se muestra en la figura 3.4.

El funcionamiento es el mismo excepto en que este se está manejando a través de dos puertos. La línea de control la generamos también por la terminal 1 del puerto C, y al igual que en el caso anterior la invertimos para evitar el exceso de carga. La señal así generada es la señal I/O3 del diagrama, la cual vuelve a invertir para obtener la señal negada necesaria, a la cual llamamos I/O4. Lo que tenemos que cuidar es conectar a una misma línea de control las señales DAV y EOI y su negada a las señales NRFD y NDAC para cumplir con el flujo adecuado de las

señales. La parte menos significativa del Puerto C queda programada como salida para generar las dos líneas de control utilizadas.

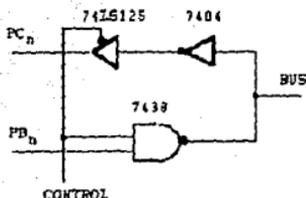


Fig. 3.4. Transceptor empleado en las líneas de protocolo de comunicación del bus 488

3.2.6 Manejo de las líneas de control del bus 488

El caso de las líneas de Control del bus 488 es mucho más sencillo, ya que consideramos que nuestra tarjeta va a ser la única con funciones de Controlador dentro del sistema.

Estas líneas son las necesarias para que la tarjeta tenga las funciones de Controlador y siempre van a funcionar como salidas, por lo que pueden ser generadas a través del Puerto B, que necesita ser programado como salida y del cual tenemos varias líneas no utilizadas.

La adaptación al bus de estas señales la vamos a realizar mediante el circuito inversor SN7406, con salida de colector abierto. Las salidas que se van a generar a través de este puerto son: ATN, IFC y REN, cada una de las cuales son de uso exclusivo del Controlador del Bus.

3.2.7 Asignación de los puertos del CI 8255A

Como resumen se presenta la asignación final de las líneas del bus 488 a los puertos del CI 8255A:

Puerto A	línea del CI 8255A	línea del bus 488	sentido
	PA0	DIO1	bidireccional
	PA1	DIO2	bidireccional
	PA2	DIO3	bidireccional
	PA3	DIO4	bidireccional
	PA4	DIO5	bidireccional
	PA5	DIO6	bidireccional
	PA6	DIO7	bidireccional
	PA7	DIO7	bidireccional
Puerto B	PB0	DAV	salida
	PB1	NRFD	salida
	PB2	NDAC	salida
	PB3	Sin conexión	
	PB4	EOI	salida
	PB5	REN	salida
	PB6	IFC	salida
	PB7	ATN	salida
Puerto C	PC0	Señal de control	salida
	PC1	Señal de control	salida
	PC2	Sin conexión	
	PC3	Sin conexión	
	PC4	DAV	entrada
	PC5	NRFD	entrada
	PC6	NDAC	entrada
	PC7	EOI	entrada

Capítulo 4

DESCRIPCION DE LOS PROGRAMAS DESARROLLADOS

CONSIDERACIONES DE PROGRAMACION

Debido a que todos los programas que se desarrollarán pueden ser llamados como subrutinas a los programas de aplicación del usuario, se pensó en programarlas en un lenguaje que fuera sencillo de programar y que fuera conocido por cualquier persona no experta en programación. Uno de los lenguajes que cumplen con los requisitos anotados líneas arriba es el Basic.

Se escogió el Turbo Basic porque aporta ventajas con respecto al intérprete GW BASIC, sus programas son compilables y se pueden generar archivos ejecutables que pueden llegar a operar hasta 100 veces más rápido que sus equivalentes en intérprete. Además, se puede programar en forma estructurada, lo que hace la programación y las correcciones más fáciles. Para personas expertas, o aplicaciones en las que se requiera velocidad en el proceso, se tiene la facilidad de llamar a rutinas escritas en lenguaje Ensamblador de manera mucho más fácil y versátil, logrando programas de alta eficiencia.

Por todo lo anterior el Turbo Basic aporta todas las ventajas a un programador experimentado y conserva la facilidad de uso, inclusive a personas no expertas y que requieran de aplicaciones demasiado sencillas.

A continuación se presenta la explicación de los diagramas de flujo de las secciones más importantes del programa fuente.

4.1 DESCRIPCION DE LOS DIAGRAMAS DE FLUJO

Por facilidad de entendimiento de los diagramas de flujo, incluidos en el apéndice A, se emplean los valores lógicos de las señales de la Norma. Hay que recordar que el bus 488 trabaja con lógica negada.

El primer diagrama, figura 4.1, describe los pasos que se llevan a cabo con el comando INICIA. El primer paso es obtener la dirección en la cual se encuentra instalada la tarjeta, ya que las operaciones de escritura y lectura a los puertos del CI 8255A

se hacen tomando esta dirección como base. A continuación se obtiene la dirección de la tarjeta dentro del sistema, tomada en cuenta cuando actúe como emisor o receptor dentro del bus.

Tomando como dirección inicial la obtenida, se programan los puertos del CI 8255A para iniciar su operación como controlador del bus:

Puerto A	salida
Puerto B	salida
Puerto C (PC7-PC4)	entrada
Puerto C (PC3-PC0)	salida interna

Las terminales PC0 y PC1 se colocan de tal manera que las señales EO1, DAV y DIO se configuren como salida y las líneas NRFD y NDAC como entradas. A continuación se colocan todas las señales, a través del puerto A y B, en cero lógico, a excepción de la señal REN, con el fin de indicar a todos los dispositivos que estarán en control remoto cuando detecten su dirección en el bus.

El diagrama de flujo de la figura 4.2 es una subrutina utilizada para manejar el protocolo de comunicación del controlador con algún dispositivo que se quiera direccionar como emisor, o bien mandar cualquier comando de la Norma.

Cuando el dispositivo coloca la señal NRFD en "0" es una indicación de que se puede validar el dato que se encuentre en ese momento, por lo que el emisor coloca la línea DAV en "1". El dispositivo responde colocando la señal NDAC en "0". En este momento el emisor hace la línea DAV igual a "0", dándose por terminado, en este momento, el protocolo de comunicación.

En la figura 4.3 se muestra el diagrama de flujo empleado para colocar una dirección de receptor en el bus 488.

El proceso se inicia colocando la línea ATN en "1" para indicar a todos los dispositivos que estén atentos al mensaje de las líneas de datos. Si todos los dispositivos están listos colocan la señal NDAC en "0" y se coloca la dirección en el bus, en caso contrario, existe un error en el bus. La comunicación con los dispositivos se lleva a cabo con la subrutina de protocolo, explicada anteriormente. En caso de existir más de una dirección de receptor se repite el protocolo cuantas veces sea necesario para el número de direcciones existentes. El proceso termina colocando la línea ATN en su estado de reposo, "0", sin olvidar mantener la señal REN en su estado activo.

El proceso para colocar un comando en el *bus* es similar al descrito anteriormente y está ilustrado en el diagrama de flujo de la figura 4.4. La variación consiste en que se coloca el código del comando que se quiera ejecutar y no la dirección en las líneas de datos. Además, sólo es posible ejecutar un comando cada vez que esta subrutina es llamada.

El proceso de colocar una dirección de locutor en el *bus* está ilustrado en la figura 4.5. El valor final de la dirección de locutor del dispositivo colocada en el *bus* se obtiene sumándole un 64 decimal (40 hexadecimal), lo cual equivale a poner en 1 el séptimo bit de la dirección, evitando así que el usuario se preocupe por esto. A continuación se valida la línea ATN y se verifica que la señal NDAC sea igual a "0". Si no es así, existe un error en el *bus*. En caso afirmativo se coloca la dirección en las líneas de datos y se inicia el protocolo de comunicación. En este momento se libera la línea ATN y el dispositivo queda designado como locutor hasta que se envíe otra dirección o el mensaje UNTALK.

El proceso de transmitir datos varía un poco con respecto al hecho de colocar una dirección o comando en el *bus*. La diferencia reside en el protocolo de comunicación, el cual tiene que saber si el dato que se transmite es o no el último, con el fin de manejar adecuadamente las señales DAV y EOI.

El protocolo, ilustrado en el diagrama de flujo de la figura 4.6, se inicia leyendo el estado de la línea NREFD hasta que el dispositivo la ponga en "0". Si no es el último dato se hace válida la señal DAV, mientras que la línea EOI se mantiene en su estado de no válido, "0". En caso de que sea el último carácter a transmitirse se validan simultáneamente las señales DAV y EOI, esta última le indica al receptor que se está transmitiendo el dato final. Cuando el receptor haga la señal NDAC igual a "0", el controlador maneja la línea DAV a "0", finalizando así el protocolo de transmisión de un dato.

El proceso completo de transmisión de una cadena de caracteres, la cual puede ser por ejemplo datos de programación para un dispositivo, se ve ilustrado en el diagrama de flujo de la figura 4.7.

El primer paso es programar al controlador como locutor del *bus*, para lo cual empleamos la subrutina de locutor, colocando la dirección de la tarjeta indicada en el comando INICIA. A continuación, por seguridad, se envía el mensaje UNLISTEN para evitar que algún dispositivo no deseado detecte los datos transmitidos. El siguiente paso es programar a todos los dispositivos indicados para actuar como receptores en el *bus*.

Cuando los dispositivos estén listos para recibir los datos colocan la señal NRFD igual a "0", indicando así que se puede iniciar la transmisión. El controlador coloca el dato en la línea DIO y se emplea el protocolo emisor. Este proceso es ejecutado hasta que se vaya a transmitir el último dato. En este momento el controlador coloca la señal EOI igual a "1" para indicar que el dato que va a colocar en el bus es el último dato a transmitirse. Después de llevar a cabo el protocolo de comunicación, la transmisión se termina colocando las líneas de control en su estado inactivo.

El diagrama de flujo de la figura 4.8. ilustra el proceso de leer datos provenientes de algún dispositivo. Lo primero es configurar al dispositivo como emisor y la tarjeta como receptor del bus. El siguiente paso es reprogramar los puertos del CI 8255A de la siguiente manera:

Puerto A	Entrada
Puerto B	Salida
Puerto C (PC7-PC4)	Entrada
Puerto C (PC3-PC0)	Salida

Nuestra señal de control interna, escrita en los líneas PC0 y PC1, programa las líneas DAV y DOI como entradas y las señales de control NRFD y NDAC como salidas. En este momento, se procede al protocolo de lectura.

Si la línea EOI es igual a "0", significa que no es el último dato, y se espera a que la señal DAV sea igual a "0", dato válido, se lee el dato y se manejan las señales NRFD igual a "1" y NDAC igual a "0" hasta que el locutor coloque la línea DAV igual a "0", con lo cual el receptor, nuestra tarjeta, coloca la señal NDAC igual a "1". Cuando se detecte que la línea EOI sea igual a "1" es que se leyó el último dato, por lo que los puertos del CI 8255A se programan a su estado inicial.

En el apéndice B se proporciona un listado del programa fuente desarrollado. El programa está completamente desarrollado en Turbo Basic versión 1.0 de Borland International, Inc.

Capítulo 5

APLICACIONES Y CONSIDERACIONES DE USO DE LA TARJETA

COMO USAR LOS COMANDOS

A continuación se da una breve explicación de los formatos requeridos y de las convenciones tomadas para el correcto funcionamiento de los Comandos.

En el formato requerido para cada comando se siguen las siguientes convenciones:

1. Todo debe ser escrito en letra mayúscula.
2. No se puede insertar espacios en blanco.
3. Todos los parámetros se separan por una diagonal (/).
4. Los parámetros por ser de tipo carácter se encierran entre comillas.
5. El último carácter siempre debe ser la diagonal.
6. Las direcciones de Receptor se anotan en base decimal, en caso de que exista más de una se separan por comas. No se permiten direcciones secundarias.
7. La dirección de Locutor se anota en base decimal, solamente se permite una dirección.
8. Un asterisco indica que se puede repetir el mismo parámetro varias veces.
9. Cada vez que se requiera usar un comando se debe igualar la variable PARAMETROS\$ con el formato requerido y mandar llamar a la subrutina COMANDOS. Cuando regrese de la subrutina el comando indicado habrá sido ejecutado.
10. El comando inicia se debe ejecutar 1 SOLA VEZ y antes de hacer uso de cualquier comando. Recuerde que este comando inicializa todos los valores de las líneas del Bus.

5.1 RESUMEN DE COMANDOS

A continuación se da un resumen de los Comandos disponibles, al mismo tiempo que se anota su formato correcto y se da una breve explicación del manejo que hace sobre las líneas del bus.

COMANDOS

Nombre : Inicia
Formato: PARAMETROS\$="INICIA/"
Función: Debe ser usado sólo una vez para iniciar la tarjeta. Configura los puertos para funcionar como un Locutor e inicializa todas las líneas del bus a sus valores de reposo.

Nombre : Limpia Interfaz
Formato: PARAMETROS\$="IFC/"
Función: Valida la línea de Limpia Interfaz (IFC) , provocando que todas las líneas del bus sean inicializadas.

Nombre : Todos los dispositivos a Local
Formato: PARAMETROS\$="TLDL/"
Función: Invalida la línea REN provocando que todos los dispositivos regresen a la disposición de ser controlados desde el panel de control local.

Nombre : Todos los dispositivos a Remoto
Formato: PARAMETROS\$="TLDR/"
Función: Valida la línea REN provocando que los dispositivos sean colocados en control remoto cuando detecten su dirección en el bus

Nombre : Limpia a todos los dispositivos
Formato: PARAMETROS\$="ADC/"
Función: Coloca en el bus el mensaje "ALL DEVICES CLEAR" el cual es procesado por todos los dispositivos capaces de hacerlo.

Nombre : Seguro contra control local
Formato: PARAMETROS\$="SCCL/"
Función: Coloca en el bus el mensaje "Local Lock Out" provocando que todos los dispositivos queden deshabilitados de ser controlados por medio de control local.

Nombre : Dispositivos Seleccionados a Local
Formato: PARAMETROS\$="DSL/dirección receptor*/"
Función: Configura a los dispositivos listados para ser operados desde el panel de control frontal, liberándolos del control remoto.

Nombre : Dispositivos Seleccionados a Remoto
Formato: PARAMETROS\$="DSR/dirección receptor*/"
Función: Configura a los dispositivos listados para ser operados desde el panel de control frontal.

Nombre : Limpia dispositivos seleccionados
Formato: PARAMETROS\$="SDC/dirección receptor*/"
Función: Manda el mensaje "Selected Device Clear" a todos los dispositivos listados provocando que inicien su estado de operación.

Nombre : Habilitador de disparo por grupo
Formato: PARAMETROS="GOT/dirección receptor/"
Función: Transmitir el mensaje "GROUP ENABLE TRIGGER" a los dispositivos listados provocando que inicien una operación simultáneamente.

Nombre : Escribe carácter
Formato: PARAMETROS="ESCCAR/dirección receptor/"
Función: Configura a la lista de dispositivos para actuar como receptores y al controlador como emisor del bus. Se transmite el código ASCII de los caracteres contenidos por la variable DATOS.
- Solamente se tiene implementado el final de transmisión mediante la línea EOI, la cual será válida en el momento de transmitir el último carácter.

Nombre : Lee dato
Formato: PARAMETROS="LEE.DATO/dirección locutor/"
Función: Configura a la tarjeta para funcionar como receptor y al dispositivo listado para trabajar como locutor del bus. Los datos que son leídos se interpretan como caracteres ASCII y añadidos a la variable DATO.STRING\$. La cantidad de caracteres está limitada por el Turbo Basic y corresponde a 32767.

En caso de que se necesite hacer una lectura de datos cuyo formato no este en ASCII, el valor hexadecimal de cada byte puede ser obtenido con las siguientes instrucciones:

```
FOR INICIO%=1 TO LEN(DATO.STRING$)
  PRINT HEX$(ASC(MID$(DATO.STRING$,INICIO%,1)))
NEXT INICIO%
```

El último dato leído será cuando se detecte que el locutor haga válida la línea EOI.

Nombre : Lee dato hacia archivo
Formato: PARAMETROS="LEE.A.ARCHIVO/dirección locutor/nombre de archivo/"

Función: Lee datos del bus GPIB, interpretándolos como caracteres ASCII y los escribe en el archivo indicado en el comando. La rutina de operación es idéntica al comando anterior.

Nombre : Poleo en serie
Formato: PARAMETROS="POLEO.SERIE/dirección receptor/"
Función: Configura a el dispositivo listado para ser poleado. La respuesta del dispositivo es regresada en la variable STATUS% y es necesario hacer la operación STATUS% AND &H40 para saber si el dispositivo validó la línea SRQ. Para conocer el significado de los otros 7 bits hay que referirse al manual del usuario del dispositivo.

Nombre : Locutor
Formato: PARAMETROS\$="LOCUTOR/dirección emisor/"
Función: Coloca en el bus la dirección de locutor indicada, habilitando al dispositivo para operar como locutor del bus

Nombre : Receptor
Formato: PARAMETROS\$="RECEPTOR/dirección receptor/"
Función: Este comando es usado para configurar una lista de dispositivos como receptores dentro del bus.

Nombre : Nolocutor
Formato: PARAMETROS\$="NOLOCUTOR/"
Función: Manda el mensaje universal "UNTALK" provocando que todos los dispositivos en el bus queden inhabilitados como emisores.

Nombre : Noreceptor
Formato: PARAMETROS\$="NORECEPTOR/"
Función: Este comando emite el mensaje universal "UNLISTEN" dando como resultado que ningún dispositivo quede habilitado como receptor.

5.2 PROGRAMA DE APLICACION

El siguiente programa es un ejemplo de la forma en que se puede programar la tarjeta. La demostración se hace con un procesador de señales marca SANEI modelo 7T26S de un solo canal y todos los comandos de operación del mismo fueron obtenidos del manual del usuario del equipo. Básicamente se demuestra como programar un equipo, y la forma de leer datos, con lo cual se involucran las funciones de controlador, emisor y receptor. Este programa debe ser compilado junto con el programa fuente del controlador para formar el programa de aplicación completo. El programa debe ser ejecutado en una computadora con tarjeta de video CGA y monitor a color.

PROGRAMA DE APLICACION DEL USO DE LA TARJETA IEEE-488
ESCRITO EN TURBO BASIC VERSION 1.0
AUTOR: ARMANDO LOZANO RAMIREZ.

```
DIBUJA LA PANTALLA DE PRESENTACION
SCREEN 1
COLOR 1
FOR RADIO%= 1 TO 100
  CIRCLE(100,100),RADIO%.1
NEXT RADIO%
```

```

FOR RADIO%=60 TO 1 STEP -1
  CIRCLE(160,100),RADIO%.2
NEXT RADIO%
FOR I=0 TO 3
  VIEW (10*(I+1),10*(I+1))-(25*(I+1),25*(I+1)),I,3-I
  CIRCLE (10,10),I+1,3-I
  VIEW (320-25*(I+1),10*(I+1))-(320-10*(I+1),25*(I+1)),I,3-I
  CIRCLE (10,10),I+1,3-I
NEXT I
VIEW (120,150)-(200,198),6,5
PI=ATN(1)*4
VAL1=5:VAL2=6:INCREM=400
WINDOW(-1,1)-(1,-1)
FOR ANGULO=0 TO 2*PI STEP 2*PI/INCREM
  RADIOS=COS(2*THETA)
  X=RADIOS*COS(VAL1*THETA)
  Y=RADIOS*SIN(VAL2*THETA)
  PSET(X,Y)
NEXT ANGULO
LOCATE 13,17:PRINT "IEEE-488"
LOCATE 1,9:PRINT "INSTITUTO DE INGENIERIA"
WHILE NOT INSTAT : WEND
SCREEN 0:WIDTH "SCRN:".80

*INICIA LAS OPERACIONES Y CONFIGURA AL SANEI COMO RECEPTOR DEL
BUS
PARAMETROS$="INICIA/2/&H310/":GOSUB COMANDOS
PARAMETROS$="IFC/":GOSUB COMANDOS
PARAMETROS$="DSR/1/":GOSUB COMANDOS
GOSUB PRINCIPAL

*CREA LA PANTALLA DEL MENU PRINCIPAL
PRINCIPAL:
FIN%=0
WHILE FIN%=0
  COLOR 0,1
  CLS
  COLOR 4
  CALL CUADRO(1,3,13,65,219)
  COLOR 15
  LOCATE 2,17:PRINT "MENU PRINCIPAL DEL PROCESADOR DE SEÑALES
SANEI"
  CALL CUADRO(8,18,13,65,177)
  COLOR 15,4
  LOCATE 10,29:PRINT " 1.- PROGRAMAR SANEI "
  LOCATE 12,29:PRINT " 2.- LEER DATOS "
  LOCATE 14,29:PRINT " 3.- IMPRIMIR "
  LOCATE 16,29:PRINT " 4.- SALIR "
  OPCION%=0
  WHILE OPCION% < 1 OR OPCION% > 4
    LOCATE 22,13:INPUT: "SELECCIONE OPCION :".OPCION%:BEEP
  WEND
  SELECT CASE OPCION%
    CASE 1:GOSUB PROGRAMA.SANEI

```

```

CASE 2:GOSUB LEER.DATOS
CASE 3:GOSUB IMPRIMIR
CASE 4:CLS:END
END SELECT
WEND

'SOMBREA UNA REGION DE LA PANTALLA
SUB CUADRO(REN.INF%,REN.SUP%.COL.INF%.COL.SUP%,FONDO%)
FOR J%=REN.INF% TO REN.SUP%
  FOR I%=COL.INF% TO COL.SUP%
    LOCATE J%,I%:PRINT CHR$(FONDO%)
  NEXT I%
NEXT J%
END SUB

'BORRA UNA REGION DE LA PANTALLA
SUB BORRA(REN.INF%,REN.SUP%.COL.INF%.COL.SUP%)
COLOR 1
FOR J%=REN.INF% TO REN.SUP%
  FOR I%=COL.INF% TO COL.SUP%
    LOCATE J%,I%:PRINT CHR$(219)
  NEXT I%
NEXT J%
COLOR 14.1
END SUB

'ROUTINA DE LECTURA DEL DATO INDICADO POR EL CURSOR
LEER.DATOS:
COLOR 0.9
CLS
COLOR 4
CALL CUADRO(1.3,13,65,219)
COLOR 15.4
LOCATE 2.19:PRINT "MENU DE LECTURA DEL PROCESADOR DE SENALES"
COLOR 15.6
CALL CUADRO(5.10,1.16,177)
LOCATE 6.8:print "MENU 1":LOCATE 8.4:PRINT " PANTALLA "
CALL CUADRO(5.10,21.36,177)
LOCATE 6.28:print "MENU 2":LOCATE 8.26:PRINT "CURSOR"
CALL CUADRO(5.10,41.56,177)
LOCATE 6.48:print "MENU 3":LOCATE 8.46:PRINT "LECTURA"
CALL CUADRO(5.10,61.76,177)
LOCATE 6.68:print "MENU 4":LOCATE 9.66:PRINT "SALIR "
TECLA%=0
WHILE TECLA% = 0
  OPCION%=0
  WHILE OPCION% < 1 OR OPCION% > 4
    LOCATE 15.1:INPUT: "SELECCIONE OPCION : ".OPCION%
  WEND
  SELECT CASE OPCION%
    CASE 1:
      CALL BORRA(15,15,1,30)
      LOCATE 12.15:PRINT "1.- LIBERAR PANTALLA"
      LOCATE 12.40:PRINT "2.- FIJAR PANTALLA"

```

```

MENU%=0
WHILE MENU% < 1 OR MENU% > 2
  LOCATE 15.5:INPUT: "SELECCIONE OPCION :".MENU%:BEEP
WEND
SELECT CASE MENU%
  CASE 1:DATO$="TH1":CASE 2:DATO$="TH2":END SELECT
  PARAMETROS$="ESCCAR/1/":GOSUB COMANDOS
  CALL BORRA(15.15,1.35)
  CALL BORRA(12.12,13.60)
CASE 2:
  CALL BORRA(15.15,1.30)
  LOCATE 12.5:INPUT: "INTRODUZCA DESPLAZAMIENTO (0-1023)
: ".DESP$:BEEP
  DATO$="CM"+DESP$
  PARAMETROS$="ESCCAR/1/":GOSUB COMANDOS
  CALL BORRA(12.12,5.50)
CASE 3:
  CALL BORRA(15.15,1.30)
  LOCATE 12.15:PRINT "1.- CURSOR SUPERIOR"
  LOCATE 12.40:PRINT "2.- CURSOR INFERIOR"
  MENU%=0
  WHILE MENU% < 1 OR MENU% > 2
    LOCATE 15.5:INPUT: "SELECCIONE OPCION :".MENU%:BEEP
  WEND
  SELECT CASE MENU%
    CASE 1:DATO$="CR1":CASE 2:DATO$="CR2":END SELECT
    PARAMETROS$="ESCCAR/1/":GOSUB COMANDOS
    DELAY 1
    PARAMETROS$="POLEO.SERIE/1/":GOSUB COMANDOS
    PARAMETROS$="LEE.DATO/1/":GOSUB COMANDOS
    CALL BORRA(12.12,13.60)
    CALL BORRA(15.15,5.45)
    LOCATE 15.5:PRINT "D A T O   L E I D O
: ".DATO.STRING$
    WHILE NOT INSTAT:WEND
    CALL BORRA(15.15,5.70)
  CASE 4:GOSUB PRINCIPAL
END SELECT
WEND:RETURN

'RUTINA PARA IMPRIMIR CONTENIDO DE LA PANTALLA DEL SANEI
IMPRIMIR:
CLS
CALL CUADRO(9.15,15.65,177)
LOCATE 11.19:PRINT "ESTE COMANDO IMPRIME LO QUE SE TENGA EN LA"
LOCATE 12.19:PRINT "PANTALLA DEL PROCESADOR DE SENALES EN EL "
LOCATE 13.19:PRINT "MOMENTO DE EJECUTARSE..."
RESPUESTA$=""
WHILE RESPUESTA$<>"N" AND RESPUESTA$<>"n" AND RESPUESTA$
<>"S" AND RESPUESTA$<>"s"
  LOCATE 18.15:INPUT "EJECUTAR COMANDO S/N :".RESPUESTAS
WEND
SELECT CASE RESPUESTA$
  CASE "N":RETURN:CASE "n":RETURN

```

```
CASE "S":DATOS="PS":CASE "B":DATOS="PS"
END SELECT
PARAMETROS3= ESCCAR/1/:GOSUB COMANDOS
GOSUB PRINCIPAL
```

```
*DIBUJA MENU DE PROGRAMACION DEL PROCESADOR DE SEÑALES
PROGRAMA.SANEI:
COLOR 0,1:CLS
COLOR 4
CALL CUADRO(1,3,13,65,219)
COLOR 15,4
LOCATE 2,17:PRINT "MENU DE PROGRAMACION DEL PROCESADOR DE
SEÑALES"
COLOR 15,6
CALL CUADRO(5,10,1,16,177)
LOCATE 6,6:print "MENU 1":LOCATE 8,4:PRINT "DESPLIEGUE"
CALL CUADRO(5,10,21,36,177)
LOCATE 6,26:print "MENU 2":LOCATE 8,23:PRINT "FRECUENCIA "
CALL CUADRO(5,10,41,56,177)
LOCATE 6,46:print "MENU 3":LOCATE 8,43:PRINT "SENSIBILIDAD"
CALL CUADRO(5,10,61,76,177)
LOCATE 6,66:print "MENU 4":LOCATE 8,65:PRINT "DOMINIO "
CALL CUADRO(12,17,1,16,177)
LOCATE 13,6:print "MENU 5":LOCATE 15,5:PRINT "ETIQUETA"
CALL CUADRO(12,17,21,36,177)
LOCATE 13,26:print "MENU 6":LOCATE 15,25:PRINT "LIMPIAR "
CALL CUADRO(12,17,41,56,177)
LOCATE 13,46:print "MENU 7":LOCATE 15,46:PRINT "AC/DC "
CALL CUADRO(12,17,61,76,177)
LOCATE 13,66:print "MENU 8":LOCATE 15,66:PRINT "SALIR "
TECLA%=0
WHILE TECLA% = 0
  OPCION%=0
  WHILE OPCION% < 1 OR OPCION%>8
    LOCATE 22,1:INPUT: "SELECCIONE OPCION : ".OPCION%
  WEND
  SELECT CASE OPCION%
    CASE 1:GOSUB MENU1
    CASE 2:GOSUB MENU2
    CASE 3:GOSUB MENU3
    CASE 4:GOSUB MENU4
    CASE 5:GOSUB MENU5
    CASE 6:GOSUB MENU6
    CASE 7:GOSUB MENU7
    CASE 8:GOSUB PRINCIPAL
  END SELECT
WEND
```

```
*MENU DE CARACTERISTICAS DE DESPLIEGUE EN PANATALLA
MENU1:
CALL BORRA(22,22,1,35)
LOCATE 19,1:PRINT "1.- REJILLA":LOCATE 19,15:PRINT "2.- CURSOR"
LOCATE 19,28:PRINT "3.- EJE V.":LOCATE 19,41:PRINT "4.- EJE H."
LOCATE 19,55:PRINT "5.- SALIR"
```

```

DESPLIEGUE%=0
WHILE DESPLIEGUE%=0
MENU%=0
WHILE MENU% < 1 OR MENU% > 5
LOCATE 23,1:INPUT: "SELECCIONE TOPICO DE DESPLIEGUE
: ",MENU%:BEEP
WEND
SELECT CASE MENU%
CASE 1:
CALL BORRA(22,23,1,50)
LOCATE 21,10:PRINT "1.-SIN REJILLA"
LOCATE 21,30:PRINT "2.-CON REJILLA"
MENU%=0
WHILE MENU% < 1 OR MENU% > 2
LOCATE 23,1:INPUT: "SELECCIONE OPCION : ",MENU%:BEEP
WEND
SELECT CASE MENU%
CASE 1:DATOS="GR1":CASE 2:DATOS="GR2":END SELECT
PARAMETROS$="ESCCAR/1/":GOSUB COMANDOS
CALL BORRA(21,23,1,60)
CASE 2:
CALL BORRA(22,23,1,50)
LOCATE 21,1:PRINT "1.- CURSOR APAGADO"
LOCATE 21,21:PRINT "2.- CURSOR DE LINEA"
LOCATE 21,42:PRINT "3.- CURSOR DE MARCA"
MENU%=0
WHILE MENU% < 1 OR MENU% > 3
LOCATE 23,1:INPUT: "SELECCIONE OPCION : ",MENU%:BEEP
WEND
SELECT CASE MENU%
CASE 1:DATOS="CO1":CASE 2:DATOS="CO2":CASE 3:DATOS="CO3"
END SELECT
PARAMETROS$="ESCCAR/1/":GOSUB COMANDOS
CALL BORRA(21,23,1,60)
CASE 3:
CALL BORRA(22,23,1,50)
LOCATE 21,1:PRINT "1.- EJE VERTICAL LINEAL"
LOCATE 21,30:PRINT "2.- EJE VERTICAL LOGARITMICO"
MENU%=0
WHILE MENU% < 1 OR MENU% > 2
LOCATE 23,1:INPUT: "SELECCIONE OPCION : ",MENU%:BEEP
WEND
SELECT CASE MENU%
CASE 1:DATOS="VL1":CASE 2:DATOS="VL2":END SELECT
PARAMETROS$="ESCCAR/1/":GOSUB COMANDOS
CALL BORRA(21,23,1,60)
CASE 4:
CALL BORRA(22,23,1,50)
LOCATE 21,1:PRINT "1.- EJE HORIZONTAL LINEAL"
LOCATE 21,30:PRINT "2.- EJE HORIZONTAL LOGARITMICO"
MENU%=0
WHILE MENU% < 1 OR MENU% > 2
LOCATE 23,1:INPUT: "SELECCIONE OPCION : ",MENU%:BEEP
WEND

```

```

SELECT CASE MENU%
  CASE 1:DATOS="HL1":CASE 2:DATOS="HL2":END SELECT
  PARAMETROS$="ESCCAR/1":GOSUB COMANDOS
  CALL BORRA(21.23,1.60)
  CASE 5:DESPLIEGUE%=1
END SELECT
WEND
CALL BORRA(19.23,1.65)
RETURN

```

MENU DE SELECCION DE ESCALA EN FRECUENCIA

```

MENU2:
CALL BORRA(22.22,1.35)
LOCATE 19.1:PRINT "1.- 40 KHZ":LOCATE 19.21:PRINT "5.- 2 KHZ"
LOCATE 19.42:PRINT "9.- 100 HZ":LOCATE 20.1:PRINT "2.- 20 KHZ"
LOCATE 20.21:PRINT "6.- 1 KHZ":LOCATE 20.41:PRINT "10.- 40 HZ"
LOCATE 21.1:PRINT "3.- 10 KHZ":LOCATE 21.21:PRINT "7.- 400 HZ"
LOCATE 21.41:PRINT "11.- 20 HZ":LOCATE 22.1:PRINT "4.- 4 KHZ"
LOCATE 22.21:PRINT "8.- 200 HZ":LOCATE 22.41:PRINT "12.- 10 HZ"
MENU%=0
WHILE MENU% < 1 OR MENU% > 12
  LOCATE 23.1:INPUT: "SELECCIONE ESCALA DE FRECUENCIA
  ":"MENU%:BEEP
WEND
SELECT CASE MENU%
  CASE 1:DATOS="FR1":CASE 2:DATOS="FR2":CASE 3:DATOS="FR3"
  CASE 4:DATOS="FR4":CASE 5:DATOS="FR5":CASE 6:DATOS="FR6"
  CASE 7:DATOS="FR7":CASE 8:DATOS="FR8":CASE 9:DATOS="FR9"
  CASE 10:DATOS="FR10":CASE 11:DATOS="FR11":CASE 12:DATOS="FR12"
END SELECT
PARAMETROS$="ESCCAR/1":GOSUB COMANDOS
CALL BORRA(19.23,1.52)
RETURN

```

MENU DE SELECCION DE ESCALA VERTICAL

```

MENU3:
CALL BORRA(22.22,1.35)
LOCATE 19.22:PRINT "1. +30 dB":LOCATE 19.41:PRINT " 5. -10 dB"
LOCATE 20.22:PRINT "2. +20 dB":LOCATE 20.41:PRINT " 6. -20 dB"
LOCATE 21.22:PRINT "3. +10 dB":LOCATE 21.41:PRINT " 7. -30 dB"
LOCATE 22.22:PRINT "4.  0 dB"
MENU%=0
WHILE MENU% < 1 OR MENU% > 7
  LOCATE 23.1:INPUT: "SELECCIONE ESCALA DE SENSIBILIDAD
  ":"MENU%:BEEP
WEND
SELECT CASE MENU%
  CASE 1:DATOS="SE1":CASE 2:DATOS="SE2":CASE 3:DATOS="SE3"
  CASE 4:DATOS="SE4":CASE 5:DATOS="SE5":CASE 6:DATOS="SE6"
  CASE 7:DATOS="SE7"
END SELECT
PARAMETROS$="ESCCAR/1":GOSUB COMANDOS
CALL BORRA(19.23,1.52)
RETURN

```

*MENU DE DOMINIO DE DESPLIEGUE

MENU4:

CALL BORRA(22.22,1,35)

LOCATE 19.25:PRINT "1.- FRECUENCIA"

LOCATE 20.25:PRINT "2.- TIEMPO"

LOCATE 21.25:PRINT "3.- FRECUENCIA Y TIEMPO"

LOCATE 22.25:PRINT "4.- TIEMPO Y FRECUENCIA"

MENU%=0

WHILE MENU% < 1 OR MENU% > 4

LOCATE 23.1:INPUT: "SELECCIONE DOMINIO DE TRABAJO

:" ,MENU%:BEEP

WEND

SELECT CASE MENU%

CASE 1:DATOS="BH1.VS2":CASE 2:DATOS="BH1.VS1"

CASE 3:DATOS="BH1.VS2.BH2.VS1":CASE 4:DATOS="BH1.VS1.BH2.VS2"

END SELECT

PARAMETROS\$="ESCCAR/1/":GOSUB COMANDOS

CALL BORRA(19,23,1,52):RETURN

*MENU PARA ESCRIBIR O BORRAR ETIQUETA EN LA PANTALLA DEL SANEI

MENU5:

CALL BORRA(22.22,1,35)

LOCATE 19.25:PRINT "1.- ESCRIBIR ETIQUETA"

LOCATE 20.25:PRINT "2.- BORRAR ETIQUETA"

MENU%=0

WHILE MENU% < 1 OR MENU% > 2

LOCATE 23.1:INPUT: "SELECCIONE OPCION : " ,MENU%:BEEP

WEND

SELECT CASE MENU%

CASE 1:

CALL BORRA(21,21,1,79)

LOCATE 21,1:INPUT "ETIQUETA (MAX. 52): " ,ETIQUETAS

DATOS="LA"+ETIQUETAS+" ,"

CASE 2:DATOS="LE"

END SELECT

PARAMETROS\$="ESCCAR/1/":GOSUB COMANDOS

CALL BORRA(19,23,1,60)

RETURN

*EJECUTA EL COMANDO ALL DEVICE CLEAR DE LA NORMA

MENU6:

PARAMETROS\$="ADC/":GOSUB COMANDOS

CALL BORRA(22.22,1,35)

RETURN

*MENU DE TIPO DE MEDICION AC/DC

MENU7:

CALL BORRA(22.22,1,35)

LOCATE 20.30:PRINT "1.- MEDIR AC"

LOCATE 21.30:PRINT "2.- MEDIR DC"

MENU%=0

WHILE MENU% < 1 OR MENU% > 2

LOCATE 23.1:INPUT: "SELECCIONE TIPO DE MEDICION : " ,MENU%:BEEP

WEND

```
SELECT CASE MENUX  
  CASE 1:DATOS="AC1":CASE 2:DATOS="AC2"  
END SELECT  
PARAMETROS="ESCCAR/1/":GOSUB COMANDOS  
CALL BORRA(19.23.1.52)  
RETURN
```

Capítulo 6

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

Las conclusiones aquí presentadas están basadas en las pruebas que se hicieron con un Procesador de Señales de un solo canal marca Sanei modelo 7T26S y la tarjeta instalada en una computadora XT marca Printaform modelo 5710.

La tarjeta está instalada en la dirección hexadecimal 0310 (784 decimal) y todas las instrucciones de programación al procesador de señales fueron obtenidos del manual del usuario. La conexión entre la PC y el procesador de señales es hecha a través del cable recomendado en la Norma.

Las limitantes de la tarjeta son:

1. La tarjeta es considerada a ser la única Controladora del bus, por lo que no reconocerá ningún comando para transferir el control del bus.
2. No soporta el direccionamiento extendido.
3. No es capaz de hacer encuestas (*polling*) en Paralelo.
4. No reconoce la línea de Interrupción SRQ.
5. La velocidad de transferencia de datos no es muy alta debido a que se está trabajando con los "seudo-transceptores". Recuerde que las velocidades altas de transmisión requieren de transceptores especialmente diseñados para conectarse al bus.
6. No permite el manejo de acceso directo a memoria (DMA).

Entre sus funciones más importantes se encuentran:

1. Implementa las Funciones de Controlador, Emisor y Receptor
2. Realiza los protocolos de Emisor y Receptor
3. Tiene la capacidad de enviar el limpiado de interfaz (IFC).
4. Puede enviar la señal REN para conmutar entre control Remoto/Local.
5. Tiene la capacidad de encuestar en serie (*polling*)
6. Contiene funciones de Limpia Dispositivo, Disparo de Dispositivo y Funciones de Remoto/Local.
7. Es un medio sencillo de enlace con el *software* a través de Turbo Basic
8. Efectúa una selección completa en el rango posible de direccionamiento
9. Su costo es mucho menor comparado con el de las tarjetas comerciales

Una discusión favorable sobre las desventajas es que no es

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

común que haya más de un Controlador dentro de un sistema. Por lo regular siempre que haya más de un dispositivo con esta capacidad sóloamente uno puede controlar todo el sistema.

En cuanto al direccionamiento extendido será un problema en caso de que el equipo tenga funciones que únicamente se puedan acceder a través de un direccionamiento de este tipo. En caso de que se requiera habilitar esta función se tiene que modificar el *software* en la subrutina de Control, para las direcciones de Receptor, y la de Locutor para las direcciones de Emisor. Las modificaciones tienen que contemplar que después de haber colocado la dirección primaria en el *bus* se tiene que colocar la dirección secundaria en caso de que exista.

Por lo que se refiere a la incapacidad de encuesta en paralelo, se puede decir que se tiene la alternativa, aunque más lenta, de encuesta en serie. Sin embargo, si es indispensable agregar estas funciones se tienen que programar todas las rutinas necesarias.

En relación a la línea SRQ, se tiene la posibilidad de conectarse a través del CI SN7404. La línea se conectaría directamente a una línea de interrupción del microprocesador y se tendría que prever las rutinas de acción necesarias para atender esta señal. Es común que después de detectar la línea SRQ se haga cualquiera de los dos tipos de encuesta, que en nuestro caso solamente es uno. La línea SRQ es activada cuando alguno de los dispositivos detecta un problema en su funcionamiento, como en el caso de un pedirle que ejecute un comando no existente.

Por último, la velocidad de transmisión, no es posible que sea mejorada a menos que se sustituyan los transceptores por los adecuados, con las consecuentes mejoras al *hardware* y *software* de la tarjeta. Por esto último, tampoco es lógico hablar de una transferencia DMA cuando de antemano sabemos que nuestra velocidad está limitada por nuestros propios dispositivos.

En cuanto a la operación de nuestra tarjeta se comprobó que puede funcionar como Controlador, Emisor y Receptor del *bus*, lo cual satisface nuestros objetivos iniciales. Además el costo final de la tarjeta es aproximadamente un 20% del total de una tarjeta comercial y todos los componentes son de fácil adquisición.

En el programa de demostración, en donde se utilizan una gran cantidad de los comandos disponibles, se muestra como la tarjeta funciona correctamente.

Todo esto nos lleva a la conclusión de que la tarjeta funcionará correctamente para una aplicación específica. Recuerde que el programa desarrollado solamente es demostrativo y que el usuario deberá desarrollar su programa de aplicación con base en sus necesidades y apoyándose en los manuales de usuario del equipo que se quiera emplear, y haciendo uso de los comandos disponibles con base en lo indicado en los capítulos previos.

BIBLIOGRAFIA

- 1.- A. J. Bouwens. Digital Instrumentation. Mc Graw-Hill USA 1984
- 2.- Morgan Christopher, Waite Mitchell. Introducción al microprocesador 8086/8088 (16 bit). McGraw-Hill. México 1984
- 3.- Sargent Murray, Shoemaker Richard. The IBM Personal Computer from the inside out. Addison-Wesley. USA 1986
- 4.- Tocci Ronald. Sistemas Digitales. Prentice Hall Hispanoamericana. México 1987
- 5.- Bernardini Miguel. Interconexión de instrumentos de medida. EL GPIB: una norma internacional efectiva. Mundo electrónico de Boixareu editores número 103 enero 1981 p 49-54
- 6.- López F.J., Gutierrez P.M. y Castells F.J. Interface microprocesador bus IEEE488. Interconexión de periféricos a microprocesadores. Serie Mundo Electrónico. Barcelona 1980 p 146-152
- 7.- Young Richard. Implementing an IEEE-488 bus controller with microprocesor software. IEEE transactions on industrial electronics and control instrumentation volumen IECI-27 febrero 1980 p 10-15
- 8.- IEEE Standard Digital Interface for Programmable Programmable Instrumentation. The Institute of Electrical and Electronics Engineers, Inc.
- 9.- Manual Turbo Basic version 1.0 Borland International, Inc. 1987
- 10.- Microprocessor and Peripheral Handbook. Vol. 1 Intel. Santa Clara 1988
- 11.- Manual Signal Processor 7T26S. NEC SAN-EI Instruments. LTD
- 12.- SSM AH88 IEEE-488 controller instruction manual. SSM Microcomputer Products. Inc. Junio 1981
- 13.- LS/S/TTL Logic Databook. National Semiconductor Corporation.
- 14.- IEEE488 Interface Installation manual User's Guide Scientific Solutions, Inc.

APENDICE A

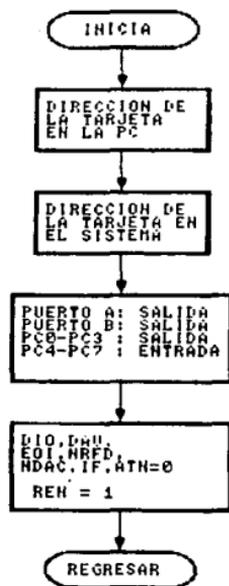


Fig. 4.1. Diagrama de flujo para iniciar el CI 8255A y las líneas del bus 488.

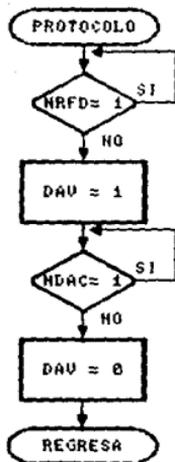


Fig. 4.2. Diagrama de flujo para manejar el protocolo GPIB para transmitir un comando o dirección de receptor.

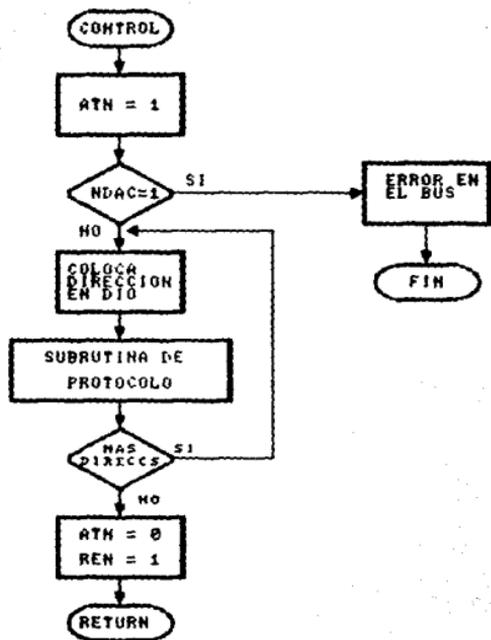


Fig. 4.3 Diagrama de flujo para colocar las direcciones de receptor en el bus.

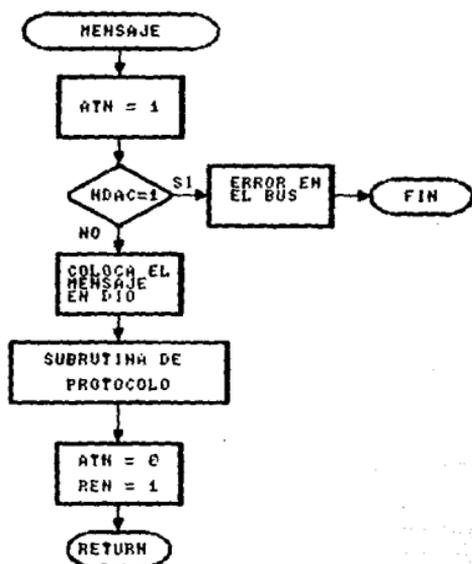


Fig. 4.4. Diagrama de flujo para colocar un comando sobre el bus 488.

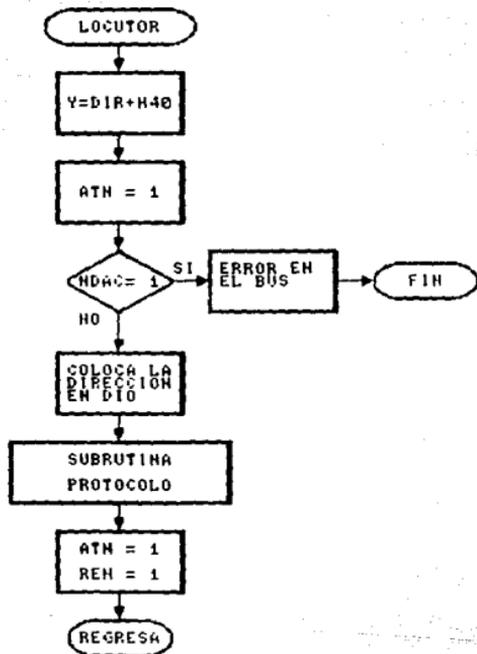


Fig. 4.5. Diagrama de flujo para configurar un elemento como locutor.

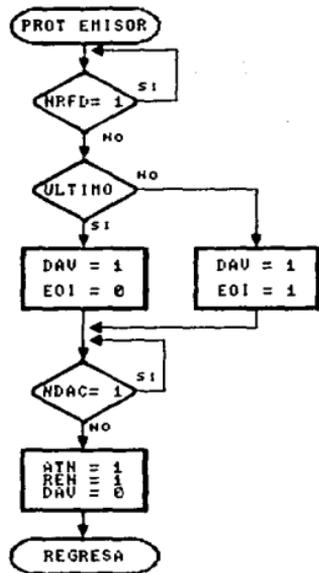


Fig. 4.3. Diagrama de flujo para manejar el protocolo GPIB en la transmisión de datos.

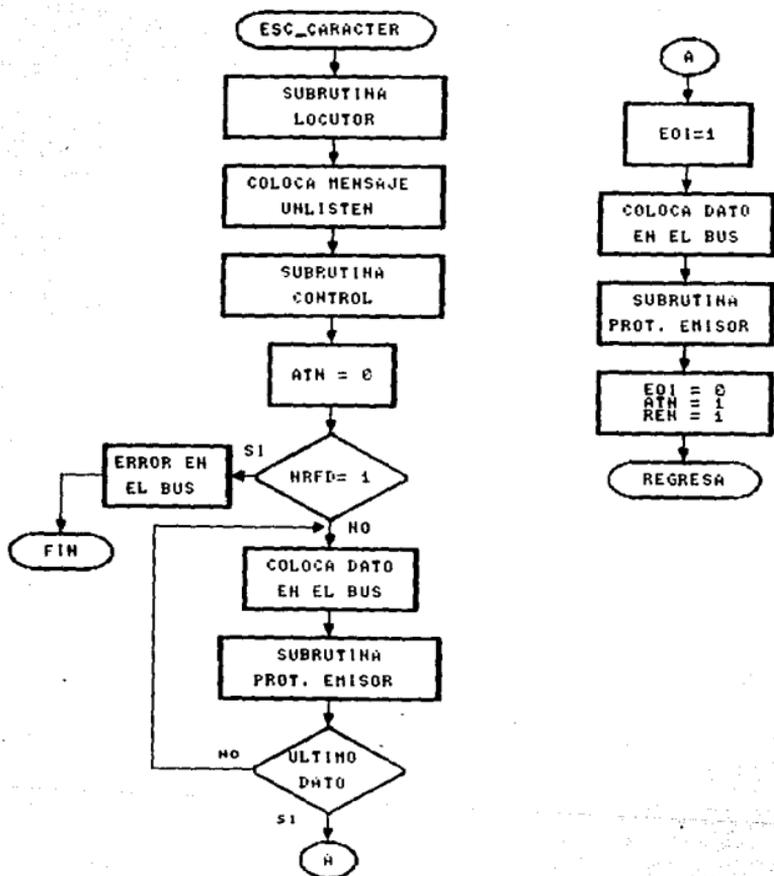


Fig. 4.7 Diagrama de flujo para controlar la escritura de datos en el bus.

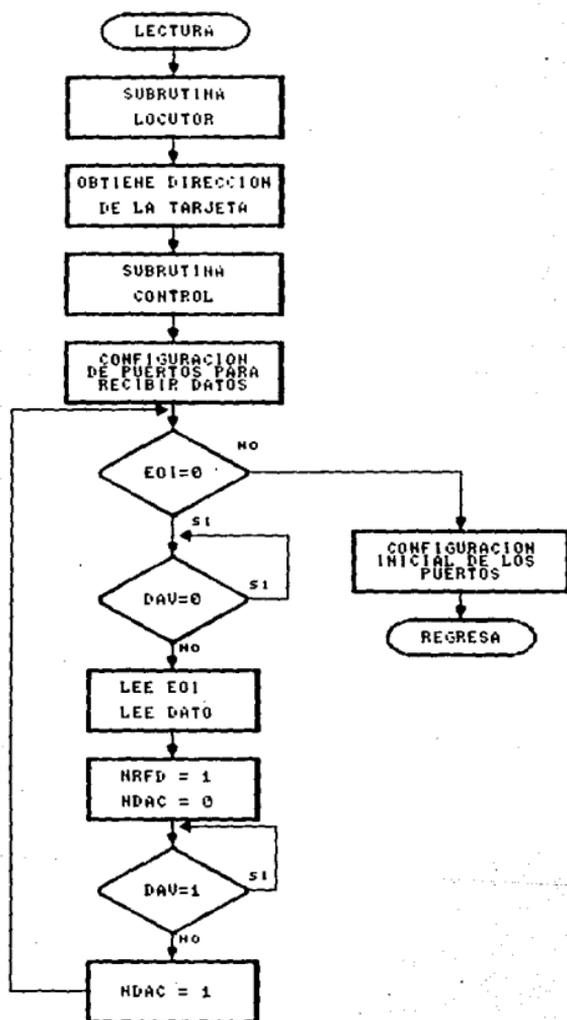


Fig. 4.8. Diagrama de flujo para controlar el proceso de lectura de datos.

APENDICE B

APENDICE B
PROGRAMA FUENTE DEL CONTROLADOR IEEE-488

```
REM          INTERFAZ IEEE488 PARA COMPUTADORA XT
REM          ESCRITO EN TURBO BASIC
REM          AUTOR: ARMANDO LOZANO RAMIREZ
REM
REM ***** COMIENZO DE SUBROUTINAS DE APLICACION *****
REM
COMANDOS:
GOSUB ANALIZA          'SUBROUTINA DE OBTENCION DE PARAMETROS
SELECT CASE CAMPOS(0) 'TOMADO DE PARAMETROSS
  CASE "INICIA"
    GOSUB INICIA
  CASE "IFC"
    GOSUB IFC
  CASE "LOCUTOR"
    GOSUB LOCUTOR
  CASE "RECEPTOR"
    GOSUB RECEPTOR
  CASE "NOLOCUTOR"
    GOSUB NOLOCUTOR
  CASE "NORECEPTOR"
    GOSUB NORECEPTOR
  CASE "DSL"
    GOSUB DSL
  CASE "TLDR"
    GOSUB TLDR
  CASE "TLDL"
    GOSUB TLDL
  CASE "SCCL"
    GOSUB SCCL
  CASE "GOT"
    GOSUB GOT
  CASE "SDC"
    GOSUB SDC
  CASE "ADC"
    GOSUB ADC
  CASE "DSR"
    GOSUB DSR
  CASE "ESCCAR"
    GOSUB ESCCAR
  CASE "POLEO.SERIE"
    GOSUB POLEO.SERIE
  CASE "LEE.DATO"
    GOSUB LEE.DATO
  CASE "LEE.A.ARCHIVO"
    GOSUB LEE.A.ARCHIVO
  CASE ELSE
    PRINT "COMANDO NO IDENTIFICADO... FIN DE EJECUCION"
END
END SELECT
RETURN
```

```

REM ***** SUBROUTINA ANALIZA *****
ANALIZA:
REM ANALIZA LA VARIABLE PARAMETROS$
ENTRADA: PARAMETROS$
SALIDA : ARREGLO CAMPOS(X%) CON LOS PARAMETROS DEL COMANDO
X%=0
FOR N1%=1 TO LEN(PARAMETROS$)
  N2%=INSTR(N1%,PARAMETROS$,"/") 'BUSCA LA POSICION DEL /
  CAMPOS(X%)=MID$(PARAMETROS$,N1%,N2%-N1%) 'CARGA CAMPOS(X%)
  X%=X%+1
  N1%=N2%
NEXT N1%
RETURN
REM ***** DIRECCION *****
DIRECCION:
REM OBTIENE LAS DIRECCIONES DE RECEPTOR
ENTRADA:CAMPOS(1)
SALIDA :DIR%(X%) CON LAS DIRECCIONES DE RECEPTOR
X%=0
N2%=INSTR(1,CAMPOS(1),".") 'BUSCA LA POSICION DE LA "."
IF N2% <> 0 THEN 'HAS DE UN RECEPTOR
  CAMPOS(1)=CAMPOS(1)+","
  FOR N1%=1 TO LEN(CAMPOS(1))
    N2%=INSTR(N1%,CAMPOS(1),".") 'POSICION DE LA "."
    DIR%(X%)=VAL(MID$(CAMPOS(1),N1%,N2%-N1%))+&H20
    X%=X%+1
    N1%=N2%
  NEXT N1%
ELSE 'UN SOLO RECEPTOR
  DIR%(0)=VAL(CAMPOS(1))+&H20 'DIRECCION DE RECEPTOR
  X%=1
END IF
RETURN
REM ***** SUBROUTINA DE PROTOCOLO *****
PROTOCOLO:
REM MANEJA LAS SEÑALES DE PROTOCOLO PARA COLOCAR UNA
DIRECCION O COMANDO EN EL BUS
ESTADO1%=INP(PTO2%) 'LEE NRFD
WHILE (ESTADO1% AND &000100010) = 32 'MIENTRAS NRFD=1
  ESTADO1%=INP(PTO2%)
WEND
OUT PTO1%,&HA1 'DAV=1
ESTADO2%=INP(PTO2%) 'LEE NDAC
WHILE (ESTADO2% AND &01000000) = 64 'MIENTRAS NDAC=1
  ESTADO2%=INP(PTO2%)
WEND
OUT PTO1%,&H20 'ATN=0,REN=1,DAV=0
RETURN
REM ***** SUBROUTINA DE CONTROL *****
CONTROL:
REM COLOCA UNA DIRECCION DE RECEPTOR EN EL BUS
GOSUB DIRECCION 'DETECCION DE DIRECCIONES
OUT PTO1%,&HAC 'REN=1,ATN=1,DAV=0 (LOGICOS)
ESTADO3%=INP(PTO2%) 'LEE EL ESTADO DE NDAU

```

```

IF (ESTADO% AND &B01000000) = 64 THEN 'DETECTA SI NDAC=1
FOR LX= 0 TO X%-1
  IF DIR%(LX) = &H20 OR DIR%(LX) >= &H3E THEN
    PRINT "DIRECCION DE RECEPTOR INVALIDA: "DIR%(LX)
    PRINT "EJECUCION TERMINADA..."
    END
  ELSE
    'DIRECCIONES VALIDAS
    OUT PTO3%,DIR%(LX) 'PONE LA DIRECCION EN EL BUS
    GOSUB PROTOCOLO 'MANEJO DE PROTOCOLO
  END IF
NEXT LX
ELSE
  PRINT "ERROR EN EL BUS SUBROUTINA CONTROL
  PRINT "EJECUCION TERMINADA..."
  END
END IF
OUT PT01%,&H22 'ATN=0,REN=1
RETURN
REM ***** MENSAJE *****
MENSAJE:
REM COLOCA EN EL BUS UN COMANDO
ENTRADA:COMANDO%
SALIDA:ACCION INDICADA POR COMANDO%
OUT PT01%,&H40 'REN=1,ATN=1,DAV=0
ESTADO%=INP(PTO2%) 'LEE NDAC
IF (ESTADO% AND &B01000000) = 64 THEN 'DETECTA SI NDAC=1
  OUT PTO0%,COMANDO% 'COLOCA MENSAJE EN EL BUS
  GOSUB PROTOCOLO 'MANEJO DE PROTOCOLO
ELSE
  PRINT "ERROR EN EL BUS. EJECUCION TERMINADA ..."
  END
END IF
OUT PT01%,&H22 'ATN=0,REN=1,DAV=0
RETURN
REM ***** SUBROUTINA DE INICIALIZACION *****
INICIA:
REM CONFIGURACION INICIAL DE LA TARJETA
MIDIR%=VAL(CAMPOS(1)) 'DIRECCION DE LA TARJETA
DIRBASE%=VAL(CAMPOS(2)) 'DIRECCION BASE
PTO0%=DIRBASE% 'PUERTO A DEL 8255A
PTO1%=DIRBASE%+1 'PUERTO B DEL 8255A
PTO2%=DIRBASE%+2 'PUERTO C DEL 8255A
PTO3%=DIRBASE%+3 'REGISTRO DE CONTROL
OUT PTO3%,&H88 'PTO A SALIDA;B SALIDA;CA ENTRADA;CB SALIDA
OUT PTO2%,&H02 'DAV/E01, DATOS EN SALIDA NDAC/NRFD ENTRADA
OUT PT01%,&H20 'REN=1 LAS RESTANTES=0
RETURN
REM ***** SUBROUTINA DE LOCUTOR *****
LOCUTOR:
REM CONFIGURA A DISPOSITIVO COMO LOCUTOR DEL BUS
ENTRADA:CAMPOS(1) O MIDIR%
SALIDA:VALIDA A UN DISPOSITIVO COMO LOCUTOR DEL BUS
IF CONTROLADOR%<1 THEN
  Y%=MIDIR%+&H40 'DIRECCION DE LOCUTOR DE LA TARJETA

```

```

ELSE
  Y%=VAL(CAMPOS(1))+&H40      DIRECCION DE LOCUTOR
END IF
OUT PTO1%,&H40                'REN=1.ATN=1.DAV=0
ESTADO%=INP(PTO2%)            'LEE NDAC
IF (ESTADO% AND %B00000100) <> 4 THEN 'DETECTA SI NDAC=1
  IF Y% <= &H40 OR Y% >= &H5E THEN
    PRINT "DIRECCION DE LOCUTOR INVALIDA: ";Y%
    PRINT "EJECUCION TERMINADA..."
  END
ELSE
  'DIRECCIONES VALIDAS DE LOCUTOR
  OUT PTO3%,Y%                'COLOCA LA DIRECCION EN EL BUS
  GOSUB PROTOCOLO            'MANEJO DE PROTOCOLO
END IF
ELSE
  PRINT "ERROR EN EL BUS (LOCUTOR). EJECUCION TERMINADA..."
  END
END IF
OUT PTO1%,&H40                'ATN=1.REN=1
RETURN
REM ***** SUBROUTINA DE RECEPTOR *****
RECEPTOR:
REM CONFIGURA UNA LISTA DE DISPOSITIVOS COMO RECEPTORES
GOSUB CONTROL                'COLOCA EN EL BUS LAS DIRECCIONES
OUT PTO1%,&H20                'LIBERA EL BUS
RETURN
REM ***** SUBROUTINA DE LIMPIA INTERFAZ *****
IFC:
REM INICIALIZA TODAS LAS TAREAS QUE SE ESTEN LLEVANDO A CABO
  EN EL BUS
OUT PTO1%,&H40                'IFC=1
DELAY .054
OUT PTO1%,&H20                'LIBERA EL BUS
RETURN
REM *** SUBROUTINA DE TODOS LOS DISPOSITIVOS A LOCAL ***
TLDL:
REM COLOCA A TODOS LOS DISPOSITIVOS EN CONTROL LOCAL
OUT PTO1%,&H00                'ATN=0.REN=0
RETURN
REM *** SUBROUTINA DE TODOS LOS DISPOSITIVOS A REMOTO ***
TLDR:
REM PONE A TODOS LOS DISPOSITIVOS EN CONDICION DE SER
  CONTROLADOS POR EL BUS
OUT PTO1%,&H40                'REN=1
RETURN
REM ***** SUBROUTINA DE SEGURO CONTRA CONTROL LOCAL *****
SCCL:
REM ENVIA EL COMANDO UNIVERSAL LOCAL LOCK OUT INHABILITANDO
  EL PANEL DE CONTROL
OUT PTO1%,&H40                'ATN=1
COMANDO%=&H11                'LOCAL LOCK OUT
GOSUB MENSAJE                'COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&H40                'ATN=1
RETURN

```

REM **** SUBROUTINA DE HABILITADOR DE DISPARO POR GRUPO ****
GOT:
REM TRANSMITE EL MENSAJE GROUP ENABLE TRIGGER A TODOS LOS
DISPOSITIVOS LISTADOS, PROVOCANDO QUE EMPIEZEN UNA
OPERACION SIMULTANEAMENTE

COMANDO%=&H3F UNLISTEN
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
GOSUB CONTROL PONE EN EL BUS LAS DIRECCIONES DE
RECEPTOR
COMANDO%=&H08 GROUP ENABLE TRIGGER
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&HA0 ATN=1
RETURN

REM ***** SUBROUTINA DE LIMPIA A TODOS LOS DISPOSITIVOS *****
ADC:

REM TRANSMITE EL MENSAJE UNIVERSAL "ALL DEVICES CLEAR"
OUT PTO1%,&HA0 ATN=1
COMANDO%=&H14 ALL DEVICES CLEAR
GOSUB MENSAJE COLOCA EL MENSAJE EN EL BUS
OUT PTO1%,&HA0 ATN=1
RETURN

REM *** SUBROUTINA DE LIMPIA DISPOSITIVOS SELECCIONADOS ***
SDC:

REM TRANSMITE EL MENSAJE "SELECTED DEVICE CLEAR" A TODOS LOS
DISPOSITIVOS LISTADOS
COMANDO%=&H3F UNLISTEN
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
GOSUB CONTROL PONE EN EL BUS LAS DIRECCS. DE
RECEPTOR
COMANDO%=&H04 SELECTED DEVICE CLEAR
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&HA0 ATN=1
RETURN

REM *** SUBROUTINA DE DISPOSITIVOS SELECCIONADOS A LOCAL ***
DSL:

REM MANDA EL MENSAJE GO TO LOCAL A LOS RECEPTORES INDICADOS
COMANDO%=&H3F UNLISTEN
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&HA0 ATN=1 (LOGICO)
GOSUB CONTROL COLOCA EN EL BUS LAS DIRECCS. DE
RECEPTOR
COMANDO%=&H01 GO TO LOCAL
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&HA0 REN=1, ATN=1
RETURN

REM ***** SUBROUTINA DE NO LOCUTOR *****

NOLOCUT:
REM MANDA EL MENSAJE UNIVERSAL UNTALK
COMANDO%=&H5F UNTALK
GOSUB MENSAJE COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&HA0 ATN=1
RETURN

```

REM ***** SUBROTINA DE NO RECEPTOR *****
NORECEPTOR:
REM MANDA EL MENSAJE UNIVERSAL UNLISTEN
COMANDO%=&H3F          UNLISTEN
GOSUB MENSAJE          COLOCA EN EL BUS EL MENSAJE
OUT PTO1%,&HA0        ATN=1
RETURN
REM *** SUBROTINA DE DISPOSITIVOS SELECCIONADOS A REMOTO ***
DER:
REM CONFIGURA A LOS DISPOSITIVOS PARA OPERAR BAJO EN BUS
COMANDO% = &H3F          UNLISTEN
GOSUB MENSAJE          COLOCA EN EL BUS EL MENSAJE
GOSUB CONTROL          COLOCA EN EL BUS DIRECCS. DE RECEPTOR
OUT PTO1%,&HA0        ATN=1
RETURN
REM ***** SUBROTINA DE PROT.EMISOR *****
PROT.EMISOR:
REM MANEJA LAS SENALES DE PROTOCOLO DEL BUS EN LA
TRANSMISION DE DATOS
ESTADO1%=INP(PTO2%)      LEE NRFD
WHILE (ESTADO1% AND &B00100010) = 32  MIENTRAS NRFD=1
    ESTADO1%=INP(PTO2%)
WEND
IF FIN%=0 THEN          HAY MAS CARACTERES POR TRANSMITIR
    OUT PTO1%,&H31      DAV=1
ELSE                    ULTIMO CHARACTER
    OUT PTO1%,&H31      DAV=1,EOI=1
END IF
ESTADO%=INP(PTO2%)      LEE NDAC
WHILE (ESTADO% AND &B01000000) = 64  MIENTRAS NDAC=1
    ESTADO%=INP(PTO2%)
WEND
OUT PTO1%,&HA2          ATN=1,REN=1,DAV=0
RETURN
REM ***** SUBROTINA ESCRIBE CARACTERES *****
ESCCAR:
REM COLOCA EN EL BUS UNA CADENA DE CARACTERES CONTENIDA EN
LA VARIABLE DATOS
CONTROLADOR%=1
GOSUB LOCUTOR          SE AUTODIRECCIONA COMO LOCUTOR
COMANDO% = &H3F          UNLISTEN
GOSUB MENSAJE          COLOCA MENSAJE EN EL BUS
GOSUB CONTROL          COLOCA EN EL BUS LA DIRECCION DE
RECEPTOR
OUT PTO1%,&H20        ATN=0
FIN%=0
ESTADO%=INP(PTO2%)
IF (ESTADO% AND &B01000000) = 64 THEN  NRFD=0, NDAC=0
    FOR X%=1 TO LEN(DATOS)-1
        CARACTER%=MID$(DATOS,X%,1)
        OUT PTO2% ASC(CARACTER%)      COLOCA CHARACTER
        GOSUB PROT.EMISOR            MANEJO DE PROTOCOLO
        DELAY 0.000065
    NEXT X%

```

```

FIN%=1                                'ULTIMO DATO
CARACTER%=ASC(MID$(DATOS,LEN(DATOS),1))
OUT PTO1%,&H30                          'EOI=1
OUT PTO3%,CARACTER%                     'COLOCA ULTIMO CARACTER
GOSUB PROT.EMISOR                        'MANEJO DE PROTOCOLO
DELAY 0.000065

ELSE
PRINT "ERROR EN EL BUS (ESC_CAR)... EJECUCION TERMINADA"
END
END IF
OUT PTO1%,&HA2                            'ATN=1.REN=1
RETURN
REM ***** LEE.DATO *****
LEE.DATO:
REM LEE EL DATO ENVIADO POR EL DISPOSITIVO
ARCHIVO%=0
GOSUB LECTURA 'MANEJO DE PROTOCOLO RECEPTOR
RETURN
REM ***** LEE.A.ARCHIVO *****
LEE.A.ARCHIVO:
REM ESCRIBE LOS DATOS LEIDOS EN UN ARCHIVO
OPEN CAMPO$(2) FOR OUTPUT AS R 1
ARCHIVO%=1
GOSUB LECTURA 'MANEJO DE PROTOCOLO RECEPTOR
CLOSE R 1 'CIERRA ARCHIVO
RETURN
REM ***** LECTURA *****
LECTURA:
REM MANEJO DE PROCESO DE LECTURA DE DATOS
CONTROLADOR%=0
GOSUB LOCUTOR 'DIRECCIONA DISPOSITIVO COMO LOCUTOR
CAMPOS(1)=STR$(MIDIR%) 'SE AUTOPROGRAMA COMO RECEPTOR
GOSUB CONTROL
OUT PTO3%,&H98 'PTO A ENTRADA;B SALIDA;CA ENTRADA;CB SALIDA
OUT PTO2%,&H01 'DAV/DATOS ENTRADA; NRFD/NDAC SALIDA
OUT PTO1%,&H24 'ATN=0;REN=1;NDAC=1;NRFD=0
DATO.STRING$=""
ULTIMO%=0
WHILE (ULTIMO% AND &B10000000) <> 128 'LEE EOI
ESTADO%=INP(PTO2%) 'MIENTRAS EOI=0
WHILE (ESTADO% AND &B00010000) <> 16 'LEE DAV
ESTADO%=INP(PTO2%) 'MIENTRAS DAV=0
WEND
ULTIMO%=INP(PTO2%) 'LEE EOI
IF ARCHIVO%=0 THEN 'ESCRIBE EN LA VARIABLE
DATO.STRING$=DATO.STRING$+CHR$(INP(PTO2%))
ELSE
PRINT R 1,CHR$(INP(PTO2%)); 'ESCRIBE EN ARCHIVO
END IF
OUT PTO1%,&H26 'NRFD=1
OUT PTO1%,&H22 'NDAC=0 (LOGICO)
ESTADO%=INP(PTO2%) 'LEE DAV
WHILE (ESTADO% AND &B00010000) = 16 'MIENTRAS DAV=1
ESTADO%=INP(PTO2%)

```


APENDICE C

APPENDICE D

8255A/8255A-5 PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-45™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel® Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- 40-Pin Dual In-Line Package
- Reduces System Package Count*
- Improved DC Driving Capability

The Intel® 8255A is a general purpose programmable I/O device designed for use with Intel® microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 3 lines, borrowing one from the other group, for handshaking.

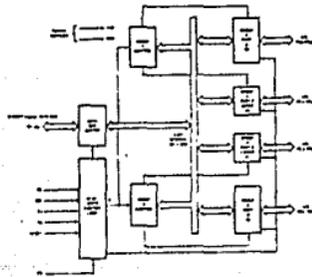


Figure 1. 8255A Block Diagram



Figure 2. Pin Configuration

8255A/8255A-5

8255A FUNCTIONAL DESCRIPTION

General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel® microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 8-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control buses and in turn, issues commands to both of the Control Groups.

(CS)

Chip Select, A "low" on this input pin enables the communication between the 8255A and the CPU.

(RD)

Read, A "low" on this input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

(WR)

Write, A "low" on this input pin enables the CPU to write data or control words into the 8255A.

(A₀ and A₁)

Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A₀ and A₁).

8255A BASIC OPERATION

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	INPUT OPERATION (READ)
0	0	0	1	0	0	0	0	PORT A = DATA BUS
0	1	0	1	0	0	0	0	PORT B = DATA BUS
1	0	0	1	0	0	0	0	PORT C = DATA BUS
0	0	0	0	1	0	0	0	OUTPUT OPERATOR (WRITE)
0	0	1	0	0	0	0	0	DATA BUS = PORT A
0	1	0	0	0	0	0	0	DATA BUS = PORT B
1	0	1	0	0	0	0	0	DATA BUS = PORT C
1	1	1	0	0	0	0	0	DATA BUS = CONTROL
0	0	0	0	0	0	0	0	DISABLE FUNCTION
0	0	0	0	1	0	0	0	DATA BUS = 3-STATE
1	1	0	1	0	0	0	0	ILLEGAL CONDITION
0	0	1	1	0	0	0	0	DATA BUS = 3-STATE

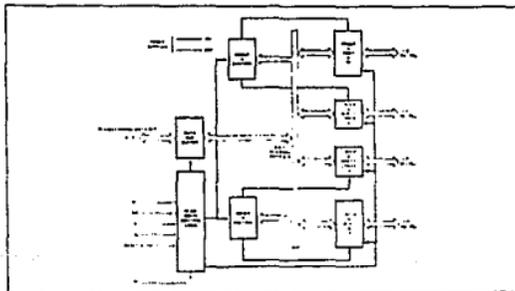


Figure 3. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

intel

8255A/8255A-5

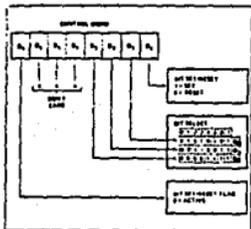
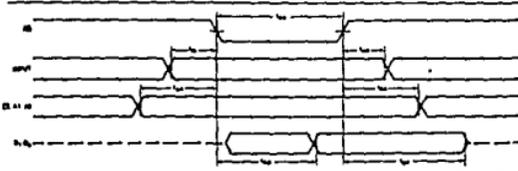


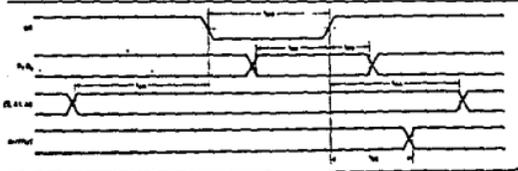
Figure 7. 8-bit Set/Reset Format

Operating Modes

MODE 0 (Strobe Input/Output) This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required; data is simply written to or read from a specified port.



MODE 0 (Strobe Output)



MODE 0 (Strobe Output)

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output bits.

Interrupt Control Functions

When the 8255A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip flop, using the bit set/reset function of port C.

This function allows the Programmer to disable or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip flop definition:

(BIT SET) = INTE IS SET = Interrupt enable
(BIT RESET) = INTE IS RESET = Interrupt disable

Note: All Mask flip flops are automatically reset during mode selection and device Reset.

Mode 0 Bank Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in 8-bit Mode.

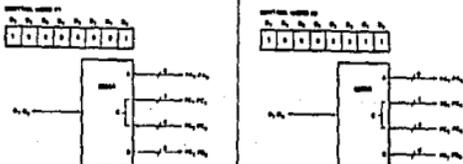
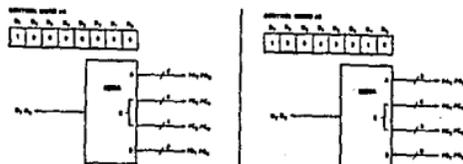
intel

8255A/8255A-5

MODE 0 Port Definition

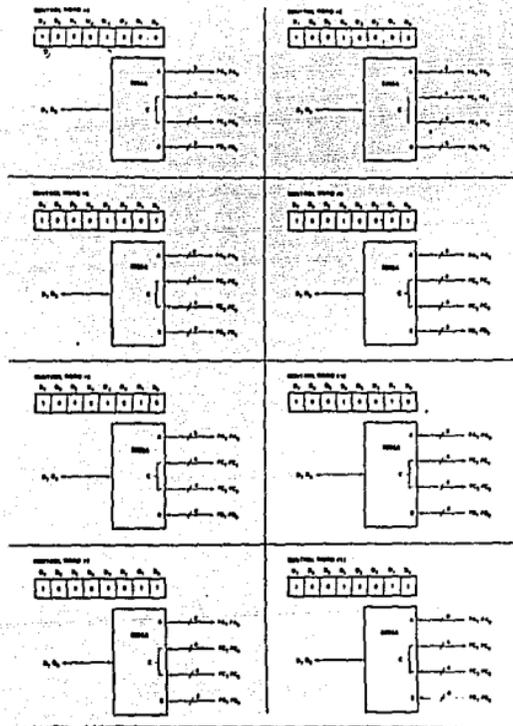
		GROUP A				GROUP B		GROUP C	
D ₇	D ₆	D ₅	D ₄	D ₃	PORT A (Latched)	PORT B (Latched)	PORT C (Latched)	PORT C (Latched)	PORT C (Latched)
0	0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	0	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	0	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	0	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	0	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	0	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	0	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	0	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	0	INPUT	INPUT	15	INPUT	INPUT

MODE 0 Configurations



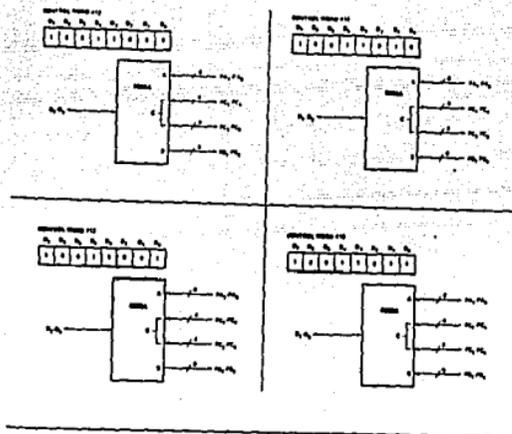
intel

8256A/8256A-5



intel

8255A/8255A-5



Operating Modes

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring IO data to or from a specified port in conjunction with strobes or "handshaking" signals. In mode 1, port A and Port B use the lines on port C to generate or accept these "handshaking" signals.

Mode 1 8-bit Functional Definitions:

- Two Groups (Group A and Group B)
- Each group contains one 8-bit data port and one 4-bit control/status port.
- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit data port.

Combinations of MODE 1

Part A and Part B can be individually defined as input or output as Mode 1 to support a wide variety of strobed I/O architectures.

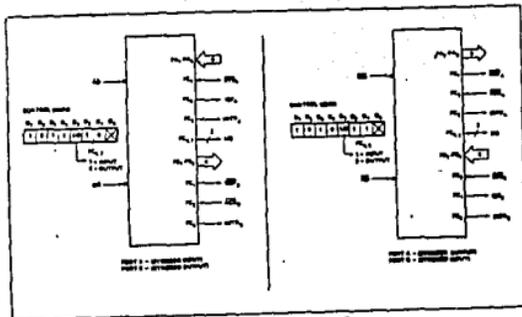


Figure 13. Combinations of MODE 1

Operating Modes

MODE 0 (Simple Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE 1. Interrupt generation and enable/disable functions are also available.

MODE 0 Basic Functional Definitions:

- Used in Group A only.
- One 8-bit, bidirectional bus port (Part A) and a 5-bit control port (Part C).
- Both inputs and outputs are latched.
- The 5-bit control port (Part C) is used for control and status for the 8-bit, bidirectional bus port (Part A).

Bidirectional Bus I/O Control Signal Definition

INTR (Interrupt Request). A high on this output can be used to interrupt the CPU for both input or output operations.

Output Operations

OBF (Output Buffer Full). The OBF output will go "low" to indicate that the CPU has written data out to part A.

ACK (Acknowledge). A "low" on this input enables the in-state output buffer of part A to send out the data. Otherwise, the output buffer will be in the high impedance state.

ITE 1 (The ITE Flip-Flop Associated with OBF). Controlled by bit 1 of control of PC₁.

Input Operations

STB (Strobe Input).

STB (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full FIF). A "high" on this output indicates that data has been loaded into the input latch.

ITE 2 (The ITE Flip-Flop Associated with IBF). Controlled by bit 1 of control of PC₂.

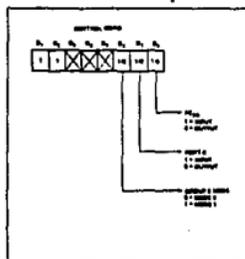


Figure 13. MODE Control Word

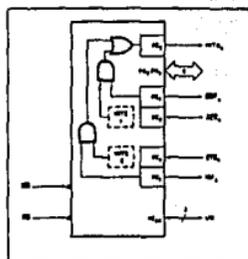


Figure 14. MODE 2

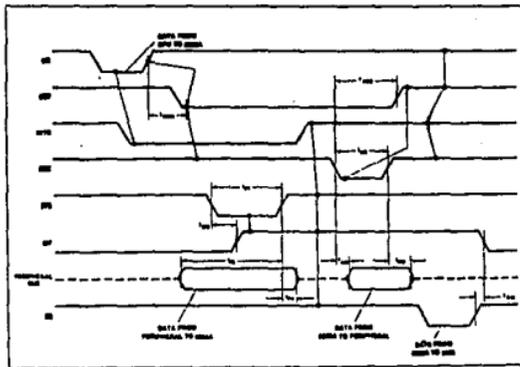


Figure 15. MODE 2 (Bidirectional)

NOTE: Any associated with STB occurs before ACK and STB occurs before ACK. RD is non-latching.
 (INTR = IBF + OBF + STB + ACK + INTR)

Input Control Signal Definition

STB (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch, in essence, an acknowledgment. IBF is set by STB input being low and is reset by the rising edge of the RD input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the STB is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of RD. This processor filters an input device to request service from the CPU by simply writing its data into the port.

INTE A

Controlled by bit 0/least of PC₂

INTE B

Controlled by bit 1/most of PC₂

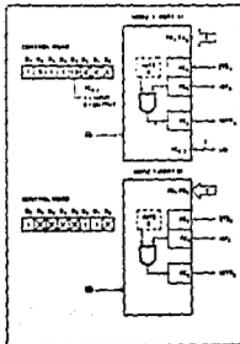


Figure 9. MODE 1 Input

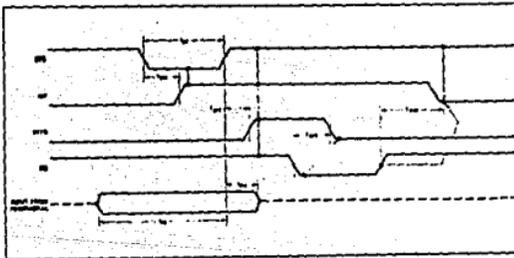


Figure 10. MODE 1 (Strobed Input)

Output Control Signal Definition

OCF (Output Buffer Full F/F). The OCF output will go "low" to indicate that the CPU has written data out to the specified port. The OCF F/F will be set by the rising edge of the WR input and reset by ACK input being low.

ACK (Acknowledge Input). A "low" on this input informs the 8255A that the data from port A or port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

INTR (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when ACK is a "one", OCF is a "one" and INTE is a "one". It is reset by the falling edge of WR.

INTE A

Controlled by bit 0/least of PC₂

INTE B

Controlled by bit 1/most of PC₂

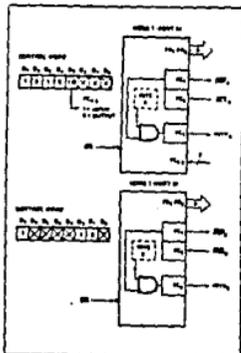


Figure 11. MODE 1 Output

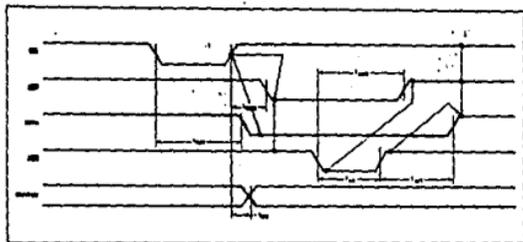


Figure 12. Mode 1 (Strobed Output)

APPLICATIONS OF THE 8255A

The 8255A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and is flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional definition of the 8255A is programmed by the I/O service routine and becomes an extension of the system software. By examining the I/O device interface characteristics for both data transfer and timing, and matching this information to the examples and tables in the detailed operational description, a control word can readily be developed to initialize the 8255A to exactly "fit" the application. Figures 19 through 25 present a few examples of typical applications of the 8255A.

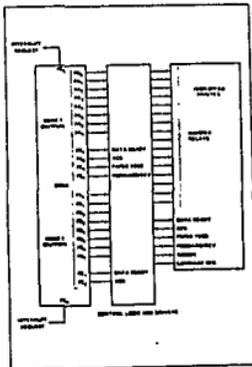


Figure 18. Printer Interface

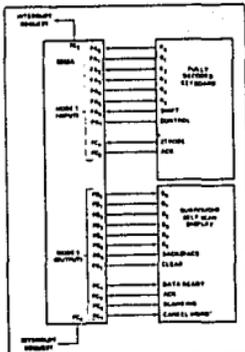


Figure 20. Keyboard and Display Interface

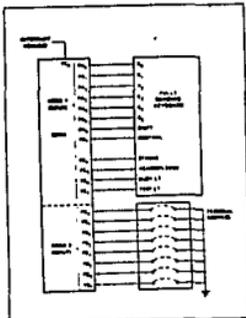


Figure 21. Keyboard and Terminal Address Interface

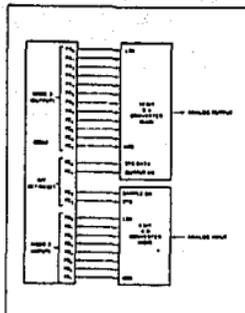


Figure 22. Digital to Analog, Analog to Digital

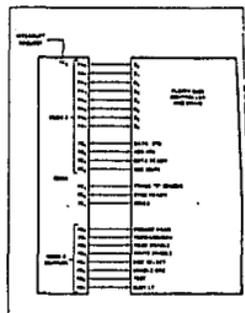


Figure 23. Basic CRT Controller Interface

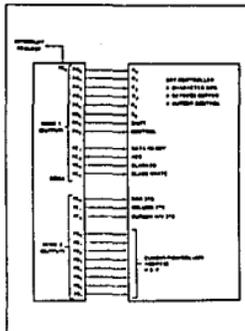


Figure 24. Basic Floppy Disc Interface

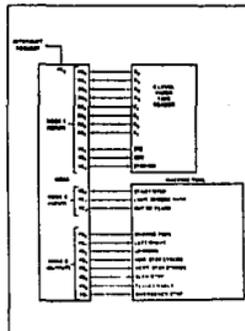


Figure 25. Machine Tool Controller Interface

intel

8255A/8255A-5

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias 0°C to 70°C
 Storage Temperature Without Bias -55°C to +150°C
 Voltage on Any Pin
 With Respect to Ground -0.5V to +17V
 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°C to 70°C, V_{CC} = +5V ± 5%, GND = 0V)

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
V _{OL}	Input Low Voltage	-0.5	0.8	V	
V _{OH}	Input High Voltage	2.0	V _{CC}	V	
V _{OL} (DB)	Output Low Voltage (Data Bus)	0.45	V		I _{OL} = 2.5mA
V _{OL} (PER)	Output Low Voltage (Peripheral Part)	0.45	V		I _{OL} = 1.7mA
V _{OH} (DB)	Output High Voltage (Data Bus)	2.4	V		I _{OH} = -400µA
V _{OH} (PER)	Output High Voltage (Peripheral Part)	2.4	V		I _{OH} = -200µA
I _{OL} (DB)	Output Drive Current	-1.2	-4.0	mA	R _{EXT} = 280Ω; V _{EXT} = 1.5V
I _{CC}	Power Supply Current	130	mA		
I _I	Input Load Current	110	µA		V _{IN} = V _{CC} to 0V
I _{OL}	Output Float Leakage	110	µA		V _{OUT} = V _{CC} to 0V

NOTES:

1. Available on any 8 pins from Part B and C.

CAPACITANCE (T_a = 30°C, V_{CC} = GND = 0V)

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Conditions
C _{IN}	Input Capacitance	10		10	pF	f _c = 1MHz
C _{IO}	I/O Capacitance	20		20	pF	Increasing and returned to GND

A.C. CHARACTERISTICS (T_a = 0°C to 70°C, V_{CC} = +5V ± 5%, GND = 0V)

See Parameters READ

Symbol	Parameter	8255A		8255A-5		Unit
		Min.	Max.	Min.	Max.	
t _{AS}	Address Stable Before READ	0		0		ns
t _{SA}	Address Stable After READ	0		0		ns
t _{WR}	READ Pulse Width	300		300		ns
t _{WD}	Data Valid From READ ¹⁾		250		300	ns
t _{OR}	Data Valid After READ	10	150	10	100	ns
t _W	Time Between READs and/or WRITEs	850		850		ns

intel

8255A/8255A-5

A.C. CHARACTERISTICS (Continued)

WRITE

Symbol	Parameter	8255A		8255A-5		Unit
		Min.	Max.	Min.	Max.	
t _{AS}	Address Stable Before WRITE	0		0		ns
t _{SA}	Address Stable After WRITE	20		20		ns
t _{WR}	WRITE Pulse Width	400		300		ns
t _{WD}	Data Valid to WRITE (T.E.)	100		100		ns
t _{OR}	Data Valid After WRITE	30		30		ns

OTHER TIMINGS

Symbol	Parameter	8255A		8255A-5		Unit
		Min.	Max.	Min.	Max.	
t _{DS}	W _{RT} = 1 to Output ¹⁾		310		330	ns
t _{DR}	Peripheral Data Before RD	0		0		ns
t _{DR}	Peripheral Data After RD	0		0		ns
t _{AS}	ACK Pulse Width	300		300		ns
t _{ST}	STB Pulse Width	500		500		ns
t _{SP}	Per. Data Before T.E. of STB	0		0		ns
t _{SD}	Per. Data After T.E. of STB	180		180		ns
t _{AD}	ACK = 0 to Output ¹⁾		300		300	ns
t _{OP}	ACK = 1 to Output ¹⁾ Hold	20	250	20	250	ns
t _{OPB}	W _{RT} = 1 to OBF = 0 ¹⁾		630		630	ns
t _{OPB}	ACK = 0 to OBF = 1 ¹⁾		350		350	ns
t _{OP}	ACK = 0 to IBF = 1 ¹⁾		300		300	ns
t _{OP}	RD = 1 to IBF = 0 ¹⁾		300		300	ns
t _{OP}	RD = 0 to INTR = 0 ¹⁾		400		400	ns
t _{OP}	STB = 1 to INTR = 1 ¹⁾		300		300	ns
t _{OP}	ACK = 1 to INTR = 1 ¹⁾		350		350	ns
t _{OP}	W _{RT} = 0 to INTR = 0 ¹⁾		450		450	ns

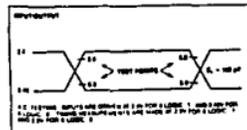
NOTES:

1. Test Conditions: 8255A C_L = 100pF, 8255A-5 C_L = 150pF

2. Period of Read pulse must be at least 50ns during or after power on. Subsequent Read pulse can be 500 ns min.

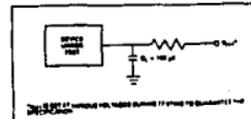
3. BITE¹⁾ may occur as early as WRT.

A.C. TESTING INPUT OUTPUT WAVEFORM



1. t_{AS} and t_{SA} apply only when t_{AS} is 10 ns for 8255A-5, and 20 ns for 8255A.
 2. t_{WD} and t_{OR} apply only when t_{WD} is 10 ns for 8255A-5, and 20 ns for 8255A.
 3. t_{OP} and t_{OPB} apply only when t_{OP} is 10 ns for 8255A-5, and 20 ns for 8255A.

A.C. TESTING LOAD CIRCUIT

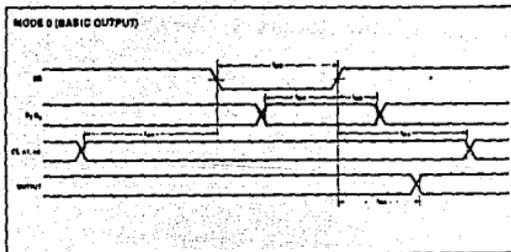
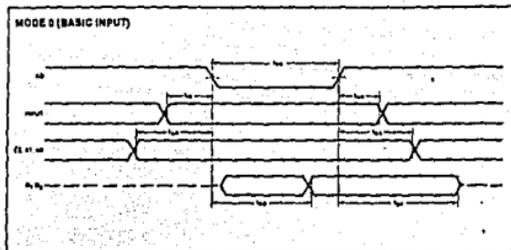


1. t_{AS} and t_{SA} apply only when t_{AS} is 10 ns for 8255A-5, and 20 ns for 8255A.
 2. t_{WD} and t_{OR} apply only when t_{WD} is 10 ns for 8255A-5, and 20 ns for 8255A.
 3. t_{OP} and t_{OPB} apply only when t_{OP} is 10 ns for 8255A-5, and 20 ns for 8255A.

intel

8255A/8255A-5

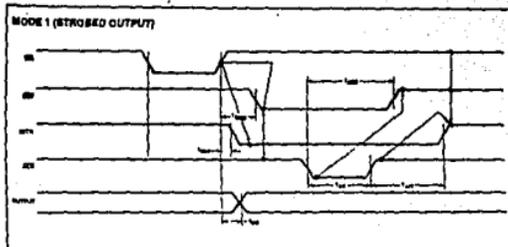
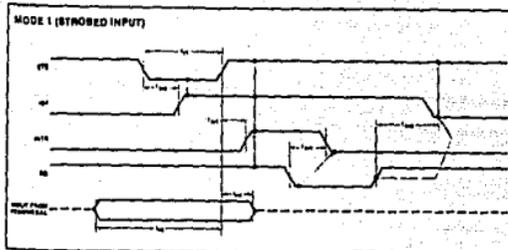
WAVEFORMS



intel

8255A/8255A-5

WAVEFORMS (Continued)



Absolute Maximum Ratings (Note)

Specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Supply Voltage	7V
Input Voltage	7V
Operating Free Air Temperature Range	
DM54LS	-55°C to +125°C
DM74LS	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	DM54LS85			DM74LS85			Units
		Min	Nom	Max	Min	Nom	Max	
V _{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V _{IH}	High Level Input Voltage	2			2			V
V _{IL}	Low Level Input Voltage			0.7			0.8	V
I _{OH}	High Level Output Current			-0.4			-0.4	mA
I _{OL}	Low Level Output Current			4			8	mA
T _A	Free Air Operating Temperature	-55		125	0		70	°C

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

Symbol	Parameter	Conditions	Min	Typ (Note 1)	Max	Units	
V _I	Input Clamp Voltage	V _{CC} = Min, I _I = -18 mA			-1.5	V	
V _{OH}	High Level Output Voltage	V _{CC} = Min, I _{OH} = Max V _{IL} = Max, V _{IH} = Min	DM54	2.5	3.4	V	
			DM74	2.7	3.4		
V _{OL}	Low Level Output Voltage	V _{CC} = Min, I _{OL} = Max V _{IL} = Max, V _{IH} = Min	DM54		0.25	0.4	V
			DM74		0.35	0.5	
			DM74		0.25	0.4	
I _I	Input Current @ Max Input Voltage	V _{CC} = Max V _I = 7V	A < B			0.1	mA
			A > B			0.1	
			Others			0.3	
I _{IH}	High Level Input Current	V _{CC} = Max V _I = 2.7V	A < B			20	μA
			A > B			20	
			Others			60	
I _{IL}	Low Level Input Current	V _{CC} = Max V _I = 0.4V	A < B			-0.4	mA
			A > B			-0.4	
			Others			-1.2	
I _{OS}	Short Circuit Output Current	V _{CC} = Max (Note 2)	DM54	-20		-100	mA
			DM74	-20		-100	
I _{CC}	Supply Current	V _{CC} = Max (Note 3)		10	20	mA	

Note 1: All typicals are at V_{CC} = 5V, T_A = 25°C

Note 2: Not more than one output should be shorted at a time, and the duration should not exceed one second

Note 3: I_{CC} is measured with all outputs open, A - B grounded and all other inputs at 4.5V.

Switching Characteristics at $V_{CC} = 5V$ and $T_A = 25^\circ C$ (See Section 1 for Test Waveforms and Output Load)

Symbol	Parameter	From Input	To Output	Number of Gate Levels	$R_L = 2\text{ k}\Omega$				Units
					$C_L = 15\text{ pF}$		$C_L = 50\text{ pF}$		
					Min	Max	Min	Max	
t _{PLH}	Propagation Delay Time Low-to-High Level Output	Any A or B Data Input	A < B, A > B	3		36		42	ns
			A = B	4		40		40	
			A < B, A > B	3		30		40	
t _{PHL}	Propagation Delay Time High-to-Low Level Output	Any A or B Data Input	A < B, A > B	3		30		40	ns
			A = B	4		30		40	
			A < B or A = B	A > B	1		22		
t _{PHL}	Propagation Delay Time High-to-Low Level Output	A < B or A = B	A > B	1		17		26	ns
t _{PLH}	Propagation Delay Time Low-to-High Level Output	A = B	A = B	2		20		25	ns
t _{PHL}	Propagation Delay Time High-to-Low Level Output	A = B	A = B	2		17		26	ns
t _{PLH}	Propagation Delay Time Low-to-High Level Output	A > B or A = B	A < B	1		22		26	ns
t _{PHL}	Propagation Delay Time High-to-Low Level Output	A > B or A = B	A < B	1		17		26	ns



National
Semiconductor
Corporation

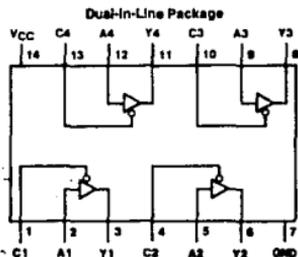
DM54LS125A/DM74LS125A Quad TRI-STATE® Buffers

General Description

This device contains four independent gates each of which performs a non-inverting buffer function. The outputs have the TRI-STATE feature. When enabled, the outputs exhibit the low impedance characteristics of a standard LS output with additional drive capability to permit the driving of bus lines without external resistors. When disabled, both the

output transistors are turned off presenting a high-impedance state to the bus line. Thus the output will act neither as a significant load nor as a driver. To minimize the possibility that two outputs will attempt to take a common bus to opposite logic levels, the disable time is shorter than the enable time of the outputs.

Connection Diagram



Order Number DM54LS125AJ, DM74LS125AM or DM74LS125AN
See NS Package Number J14A, M14A or N14A

TL/F/6387-1

Function Table

$$Y = A$$

Inputs		Output
A	C	Y
L	L	L
H	L	H
X	H	Hi-Z

H = High Logic Level

L = Low Logic Level

X = Either Low or High Logic Level

Hi-Z = TRI-STATE (Outputs are disabled)

Absolute Maximum Ratings (Note)

specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Supply Voltage	7V
Input Voltage	7V
Operating Free Air Temperature Range	
DM54LS	-55°C to +125°C
DM74LS	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at those limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	DM54LS125A			DM74LS125A			Units
		Min	Nom	Max	Min	Nom	Max	
V _{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V _{IH}	High Level Input Voltage	2			2			V
V _{IL}	Low Level Input Voltage			0.7			0.8	V
I _{OH}	High Level Output Current			-1			-2.8	mA
I _{OL}	Low Level Output Current			12			24	mA
T _A	Free Air Operating Temperature	-55		125	0		70	°C

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

Symbol	Parameter	Conditions	Min	Typ (Note 1)	Max	Units
V _I	Input Clamp Voltage	V _{CC} = Min, I _I = -18 mA			-1.5	V
V _{OH}	High Level Output Voltage	V _{CC} = Min, I _{OH} = Max V _{IL} = Max, V _{IH} = Min	2.4	3.4		V
V _{OL}	Low Level Output Voltage	V _{CC} = Min, I _{OL} = Max V _{IL} = Max	DM54	0.25	0.4	V
			DM74	0.35	0.5	
		I _{OL} = 12 mA, V _{CC} = Min	DM74	0.25	0.4	
I _I	Input Current @ Max Input Voltage	V _{CC} = Max, V _I = 7V			0.1	mA
I _{IH}	High Level Input Current	V _{CC} = Max, V _I = 2.7V			20	μA
I _{IL}	Low Level Input Current	V _{CC} = Max, V _I = 0.4V			-0.4	mA
I _{OZH}	Off-State Output Current with High Level Output Voltage Applied	V _{CC} = Max, V _O = 2.4V V _{IH} = Min, V _{IL} = Max			20	μA
I _{OZL}	Off-State Output Current with Low Level Output Voltage Applied	V _{CC} = Max, V _O = 0.4V V _{IH} = Min, V _{IL} = Max			-20	μA
I _{OS}	Short Circuit Output Current	V _{CC} = Max (Note 2)	DM54	-20	-100	mA
			DM74	-20	-100	
I _{CC}	Supply Current	V _{CC} = Max (Note 3)		11	20	mA

Note 1: All typicals are at V_{CC} = 5V, T_A = 25°C.

Note 2: Not more than one output should be shorted at a time, and the duration should not exceed one second.

Note 3: I_{CC} is measured with the data control (C) inputs at 4.5V and the data inputs grounded.

Switching Characteristics at $V_{CC} = 5V$ and $T_A = 25^\circ C$ (See Section 1 for Test Waveforms and Output Load)

Symbol	Parameter	$R_L = 607\Omega$				Units
		$C_L = 50\text{ pF}$		$C_L = 150\text{ pF}$		
		Min	Max	Min	Max	
t_{PLH}	Propagation Delay Time Low to High Level Output		15		21	ns
t_{PHL}	Propagation Delay Time High to Low Level Output		18		22	ns
t_{pZH}	Output Enable Time to High Level Output		25		36	ns
t_{pZL}	Output Enable Time to Low Level Output		25		40	ns
t_{PHZ}	Output Disable Time from High Level Output (Note 1)		20			ns
t_{PLZ}	Output Disable Time from Low Level Output (Note 1)		20			ns

 Note 1: $C_L = 50\text{ pF}$.



National
Semiconductor
Corporation

DM54LS245/DM74LS245 TRI-STATE® Octal Bus Transceiver

General Description

These octal bus transceivers are designed for asynchronous two-way communication between data buses. The control function implementation minimizes external timing requirements.

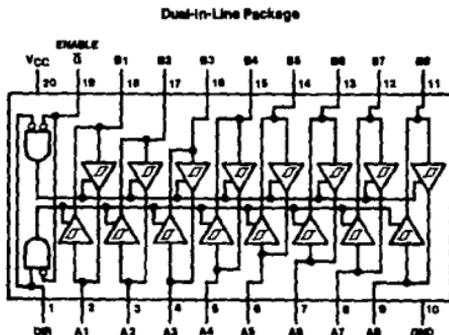
The device allows data transmission from the A bus to the B bus or from the B bus to the A bus depending upon the logic level at the direction control (DIR) input. The enable input (\bar{G}) can be used to disable the device so that the buses are effectively isolated.

- PNP inputs reduce DC loading on bus lines
- Hysteresis at bus inputs improve noise margins
- Typical propagation delay times, port-to-port 8 ns
- Typical enable/disable times 17 ns
- I_{OL} (sink current)
 - 54LS 12 mA
 - 74LS 24 mA
- I_{OH} (source current)
 - 54LS -12 mA
 - 74LS -15 mA

Features

- Bi-Directional bus transceiver in a high-density 20-pin package
- TRI-STATE outputs drive bus lines directly

Connection Diagram



Order Number DM54LS245J, DM74LS245WM or DM74LS245N
See NS Package Number J20A, M20B or N20A

TLF/M113-1

Function Table

Enable \bar{G}	Direction Control DIR	Operation
L	L	B data to A bus
L	H	A data to B bus
H	X	Isolation

H = High Level, L = Low Level, X = Irrelevant

Absolute Maximum Ratings (Note)

Specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Supply Voltage	7V
Input Voltage	
DIR or G	7V
A or B	5.5V
Operating Free Air Temperature Range	
DM54LS	-55°C to +125°C
DM74LS	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	DM54LS245			DM74LS245			Units
		Min	Nom	Max	Min	Nom	Max	
V _{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V _{HI}	High Level Input Voltage	2			2			V
V _{LI}	Low Level Input Voltage			0.7			0.8	V
I _{OH}	High Level Output Current			-12			-15	mA
I _{OL}	Low Level Output Current			12			24	mA
T _A	Free Air Operating Temperature	-55		125	0		70	°C

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

Symbol	Parameter	Conditions		Min	Typ (Note 1)	Max	Units
V _I	Input Clamp Voltage	V _{CC} = Min, I _I = -18 mA				-1.5	V
HYS	Hysteresis (V _{I+} - V _{I-})	V _{CC} = Min		0.2	0.4		V
V _{OH}	High Level Output Voltage	V _{CC} = Min, V _{HI} = Min V _{LI} = Max, I _{OH} = -1 mA	DM74	2.7			V
		V _{CC} = Min, V _{LI} = Min V _{LI} = Max, I _{OH} = -3 mA	DM54/DM74	2.4	3.4		
		V _{CC} = Min, V _{HI} = Min V _{LI} = 0.5V, I _{OH} = Max	DM54/DM74	2			
V _{OL}	Low Level Output Voltage	V _{CC} = Min V _{LI} = Max V _{HI} = Min	I _{OL} = 12 mA DM74			0.4	V
			I _{OL} = Max DM54			0.4	
			DM74			0.5	
I _{OZH}	Off-State Output Current, High Level Voltage Applied	V _{CC} = Max V _{LI} = Max V _{HI} = Min	V _O = 2.7V			20	μA
I _{OZL}	Off-State Output Current, Low Level Voltage Applied		V _O = 0.4V			-200	μA
I _I	Input Current at Maximum Input Voltage	V _{CC} = Max	A or B DIR or G	V _I = 5.5V V _I = 7V		0.1 0.1	mA
I _{HI}	High Level Input Current	V _{CC} = Max, V _I = 2.7V				20	μA
I _{LI}	Low Level Input Current	V _{CC} = Max, V _I = 0.4V				-0.2	mA
I _{OS}	Short Circuit Output Current	V _{CC} = Max (Note 2)		-40		-225	mA
I _{CC}	Supply Current	Outputs High	V _{CC} = Max		48	70	mA
		Outputs Low		62	90		
		Outputs at Hi-Z		64	95		

Note 1: All typicals are at V_{CC} = 5V, T_A = 25°C.

Note 2: Not more than one output should be shorted at a time, not to exceed one second duration.

Switching Characteristics $V_{CC} = 5V, T_A = 25^\circ C$ (See Section 1 for Test Waveforms and Output Load)

Symbol	Parameter	Conditions	DM54/74		Units
			LS245		
			Min	Max	
t_{PLH}	Propagation Delay Time, Low-to-High-Level Output	$C_L = 45 \text{ pF}$ $R_L = 687 \Omega$		12	ns
t_{PHL}	Propagation Delay Time, High-to-Low-Level Output			12	ns
t_{PZL}	Output Enable Time to Low Level			40	ns
t_{PZH}	Output Enable Time to High Level			40	ns
t_{PLZ}	Output Disable Time from Low Level	$C_L = 5 \text{ pF}$ $R_L = 667 \Omega$		25	ns
t_{PHZ}	Output Disable Time from High Level			25	ns
t_{PLH}	Propagation Delay Time, Low-to-High-Level Output	$C_L = 150 \text{ pF}$ $R_L = 687 \Omega$		16	ns
t_{PHL}	Propagation Delay Time, High-to-Low-Level Output			17	ns
t_{PZL}	Output Enable Time to Low Level			45	ns
t_{PZH}	Output Enable Time to High Level			45	ns



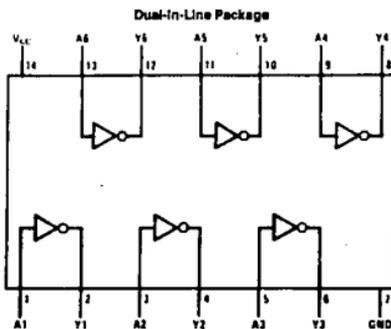
National
Semiconductor
Corporation

DM5404/DM7404 Hex Inverting Gates

General Description

This device contains six independent gates each of which performs the logic INVERT function.

Connection Diagram



TL9/0004-1

Function Table

$$Y = \bar{A}$$

Inputs	Output
A	Y
L	H
H	L

H = High Logic Level

L = Low Logic Level

Absolute Maximum Ratings (Note)

specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Supply Voltage	7V
Input Voltage	5.5V
Operating Free Air Temperature Range	
DM54	-55°C to +125°C
DM74	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	DM5404			DM7404			Units
		Min	Nom	Max	Min	Nom	Max	
V _{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V _{IH}	High Level Input Voltage	2			2			V
V _{IL}	Low Level Input Voltage			0.8			0.8	V
I _{OH}	High Level Output Current			-0.4			-0.4	mA
I _{OL}	Low Level Output Current			16			16	mA
T _A	Free Air Operating Temperature	-55		125	0		70	°C

Electrical Characteristics

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

Symbol	Parameter	Conditions	Min	Typ (Note 1)	Max	Units
V _I	Input Clamp Voltage	V _{CC} = Min, I _i = -12 mA			-1.5	V
V _{OH}	High Level Output Voltage	V _{CC} = Min, I _{OH} = Max V _{IL} = Max	2.4	3.4		V
V _{OL}	Low Level Output Voltage	V _{CC} = Min, I _{OL} = Max V _{IH} = Min		0.2	0.4	V
I _i	Input Current @ Max Input Voltage	V _{CC} = Max, V _i = 5.5V			1	mA
I _{IH}	High Level Input Current	V _{CC} = Max, V _i = 2.4V			40	μA
I _{IL}	Low Level Input Current	V _{CC} = Max, V _i = 0.4V			-1.6	mA
I _{OS}	Short Circuit Output Current	V _{CC} = Max (Note 2)	DM54	-20	-55	mA
			DM74	-18	-55	
I _{COH}	Supply Current with Outputs High	V _{CC} = Max		6	12	mA
I _{CCL}	Supply Current with Outputs Low	V _{CC} = Max		18	33	mA

Switching Characteristics at V_{CC} = 5V and T_A = 25°C (See Section 1 for Test Waveforms and Output Load)

Symbol	Parameter	Conditions	Min	Max	Units
t _{PLH}	Propagation Delay Time Low to High Level Output	C _L = 15 pF R _L = 400Ω		22	ns
t _{PHL}	Propagation Delay Time High to Low Level Output			15	ns

Note 1: All typicals are at V_{CC} = 5V, T_A = 25°C

Note 2: Not more than one output should be shorted at a time.



National
Semiconductor
Corporation

DM5438/DM7438

DM5438/DM7438 Quad 2-Input NAND Buffers with Open-Collector Outputs

General Description

This device contains four independent gates each of which performs the logic NAND function. The open-collector outputs require external pull-up resistors for proper logical operation.

Pull-Up Resistor Equations

$$R_{MAX} = \frac{V_{CC} (Min) - V_{OH}}{N_1 (I_{OH}) + N_2 (I_{IH})}$$

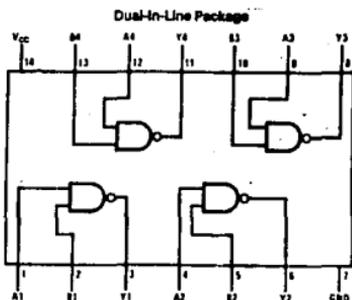
$$R_{MIN} = \frac{V_{CC} (Max) - V_{OL}}{I_{OL} - N_3 (I_{IL})}$$

Where: $N_1 (I_{OH})$ = total maximum output high current for all outputs tied to pull-up resistor

$N_2 (I_{IH})$ = total maximum input high current for all inputs tied to pull-up resistor

$N_3 (I_{IL})$ = total maximum input low current for all inputs tied to pull-up resistor

Connection Diagram



Order Number DM5438J, DM7438M or DM7438N
See NS Package Number J14A, M14A or N14A

TUF/6513-1

Function Table

$$Y = \bar{A}\bar{B}$$

Inputs		Output
A	B	Y
L	L	H
L	H	H
H	L	H
H	H	L

H = High Logic Level

L = Low Logic Level

Absolute Maximum Ratings (Note)

Specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Supply Voltage	7V
Input Voltage	5.5V
Output Voltage	7V
Operating Free Air Temperature Range	
DM54	-55°C to +125°C
DM74	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	DM5438			DM7438			Units
		Min	Nom	Max	Min	Nom	Max	
V _{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V _{IH}	High Level Input Voltage	2			2			V
V _{IL}	Low Level Input Voltage			0.8			0.8	V
V _{OH}	High Level Output Voltage			5.5			5.5	V
I _{OL}	Low Level Output Current			48			48	mA
T _A	Free Air Operating Temperature	-55		125	0		70	°C

Electrical Characteristics

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

Symbol	Parameter	Conditions	Min	Typ (Note 1)	Max	Units
V _I	Input Clamp Voltage	V _{CC} = Min, I _I = -12 mA			-1.5	V
I _{OEX}	High Level Output Current	V _{CC} = Min, V _O = 5.5V V _{IL} = Max			250	μA
V _{OL}	Low Level Output Voltage	V _{CC} = Min, I _{OL} = Max V _{IH} = Min			0.4	V
I _I	Input Current @ Max Input Voltage	V _{CC} = Max, V _I = 5.5V			1	mA
I _{IH}	High Level Input Current	V _{CC} = Max, V _I = 2.4V			40	μA
I _{IL}	Low Level Input Current	V _{CC} = Max, V _I = 0.4V			-1.6	mA
I _{COH}	Supply Current with Outputs High	V _{CC} = Max		5	8.5	mA
I _{COL}	Supply Current with Outputs Low	V _{CC} = Max		34	54	mA

Switching Characteristics at V_{CC} = 5V and T_A = 25°C (See Section 1 for Test Waveforms and Output Load)

Symbol	Parameter	Conditions	Min	Max	Units
t _{PLH}	Propagation Delay Time Low to High Level Output	C _L = 45 pF R _L = 133Ω		22	ns
t _{PHL}	Propagation Delay Time High to Low Level Output			18	ns

Note 1: All typicals are at V_{CC} = 5V, T_A = 25°C.



National
Semiconductor
Corporation

DM5406/DM7406 Hex Inverting Buffers with High Voltage Open-Collector Outputs

General Description

This device contains six independent buffers each of which performs the logic INVERT function. The open-collector outputs require external pull-up resistors for proper logical operation.

Pull-Up Resistor Equations

$$R_{MAX} = \frac{V_O(\text{Min}) - V_{OH}}{N_1(I_{OH}) + N_2(I_{IH})}$$

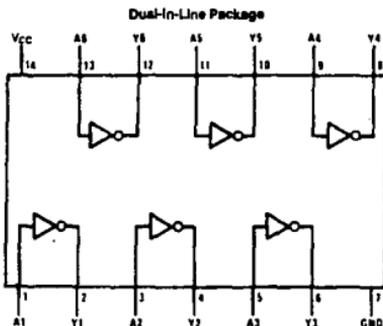
$$R_{MIN} = \frac{V_O(\text{Max}) - V_{OL}}{I_{OL} - N_3(I_{IL})}$$

Where: $N_1(I_{OH})$ = total maximum output high current for all outputs tied to pull-up resistor

$N_2(I_{IH})$ = total maximum input high current for all inputs tied to pull-up resistor

$N_3(I_{IL})$ = total maximum input low current for all inputs tied to pull-up resistor

Connection Diagram



Order Number DM5406J, DM7406M or DM7406N
See NS Package Number J14A, M14A or N14A

TL/F/8486-1

Function Table

$$Y = \bar{A}$$

Input	Output
A	Y
L	H
H	L

H = High Logic Level
L = Low Logic Level

Absolute Maximum Ratings (Note)

Specifications for Military/Aerospace products are not contained in this datasheet. Refer to the associated reliability electrical test specifications document.

Supply Voltage	7V
Input Voltage	5.5V
Output Voltage	30V
Operating Free Air Temperature Range	
DM54	-55°C to +125°C
DM74	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

Note: The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Recommended Operating Conditions

Symbol	Parameter	DM5406			DM7406			Units
		Min	Nom	Max	Min	Nom	Max	
V _{CC}	Supply Voltage	4.5	5	5.5	4.75	5	5.25	V
V _{IH}	High Level Input Voltage	2			2			V
V _{IL}	Low Level Input Voltage			0.8			0.8	V
V _{OH}	High Level Output Voltage			30			30	V
I _{OL}	Low Level Output Current			30			40	mA
T _A	Free Air Operating Temperature	-55		125	0		70	°C

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

Symbol	Parameter	Conditions	Min	Typ (Note 1)	Max	Units
V _I	Input Clamp Voltage	V _{CC} = Min, I _I = -12 mA			-1.5	V
I _{OEX}	High Level Output Current	V _{CC} = Min, V _O = 30V V _{IH} = Max			250	μA
V _{OL}	Low Level Output Voltage	V _{CC} = Min, I _{OL} = Max V _{IH} = Min			0.7	V
		I _{OL} = 16 mA, V _{CC} = Min			0.4	
I _I	Input Current @ Max Input Voltage	V _{CC} = Max, V _I = 5.5V			1	mA
I _{IH}	High Level Input Current	V _{CC} = Max, V _I = 2.4V			40	μA
I _{IL}	Low Level Input Current	V _{CC} = Max, V _I = 0.4V			-1.6	mA
I _{OCH}	Supply Current with Outputs High	V _{CC} = Max		30	42	mA
I _{OCL}	Supply Current with Outputs Low	V _{CC} = Max		27	38	mA

Switching Characteristics at V_{CC} = 5V and T_A = 25°C (See Section 1 for Test Waveforms and Output Load)

Symbol	Parameter	Conditions	Min	Max	Units
t _{PLH}	Propagation Delay Time Low to High Level Output	C _L = 15 pF R _L = 110Ω		15	ns
t _{PHL}	Propagation Delay Time High to Low Level Output			23	ns

Note 1: All typicals are at V_{CC} = 5V, T_A = 25°C