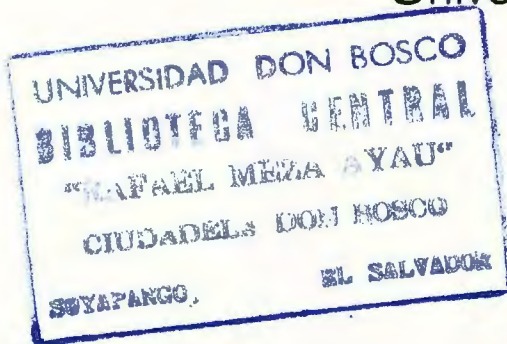


Universidad Don Bosco



**PROGRAMADOR DE MEMORIAS EPROM POR
COMPUTADORA**

**TRABAJO DE GRADUACION PREPARADO PARA LA
FACULTAD DE INGENIERIA**

**PARA OPTAR AL GRADO DE
TECNICO EN INGENIERIA ELECTRONICA**

POR:

**DAVID ESTUPINIAN CASTRO
ROBERTO EDMUNDO HERNANDEZ RODAS
RODRIGO ENRIQUE HERRERA NOSTHAS**

**Abril 1995
SOYAPANGO, EL SALVADOR. CENTRO AMERICA**

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA

PROGRAMADOR DE MEMORIAS EPROM
POR COMPUTADORA



ING. JORGE SALOMON HANDAL IBARRA.

ASESOR.

INDICE

OBJETIVOS	I
INTRODUCCION	II
 CAPITULO I. PROGRAMADORES DE MEMORIAS	 1
1.1 INTRODUCCION	1
1.2 DESCRIPCION GENERAL	1
1.3 ESTRUCTURA BASICA	3
 CAPITULO II. INTRODUCCION A LAS COMPUTADORAS	 7
2.1 CONTROL POR COMPUTADORA	7
2.1.1 INSTRUCCION OUT	9
2.1.2 FORMANDO LA DIRECCION DE SALIDA	10
2.1.3 INSTRUCCION INP	13
2.2 HARDWARE DE ENTRADA Y SALIDA PARA IBM PC	14
2.2.1 SALIDA ELECTRONICA POR IBM - PC	14
2.2.2 HARDWARE PARA INTRODUCIR DATOS A LA IBM PC	22
2.2.3 PASOS QUE OCURREN DURANTE UNA LECTURA O UNA OPERACION INP DESDE EL CKTO E/S	 25
 CAPITULO III. PROGRAMADOR DE EPROM POR PC	 27
3.1 INTRODUCCION	27
3.2 DESCRIPCION GENERAL DEL FUNCIONAMIENTO	28
3.3 HARDWARE DEL PROGRAMADOR	30
3.3.1 DECODIFICADOR DE TARJETA	31
3.3.2 DECODIFICADOR DE LATCH Y MONOESTABLE	33

3.3.3 CAPTURA DE DATOS, DIRECCIONES, PALABRA DE CONTROL	
Y ENTRADA / SALIDA DE DATOS	35
3.3.4 DECODIFICADOR DE PALABRA DE CONTROL	37
3.3.5 DECODIFICADORES DE PINES	41
3.4 SOFTWARE DEL PROGRAMADOR	43
3.4.1 FLUJOGRAMA	44
3.4.2 FUNCIONES GENERALES DEL PROGRAMADOR	57
3.4.2.1 MENUS	57
3.4.2.2 ARCHIVO	58
3.4.2.2.1 CREAR ARCHIVO	58
3.4.2.2.2 ABRIR ARCHIVO	59
3.4.2.2.3 IMPRIMIR ARCHIVO	61
3.4.2.3 SALIR	62
3.4.2.4 GRABAR MEMORIA EPROM	63
3.4.2.5 LEER MEMORIA EPROM	64
3.5 ALCANCES Y LIMITACIONES	65
CONCLUSIONES	66
APENDICE A PROGRAMA ETIQUETADO	67
APENDICE B DIAGRAMAS ELECTRONICOS	107
APENDICE C CALCULOS	111
APENDICE D HOJAS TECNICAS	115
APENDICE E MANUAL DEL USUARIO	155
BIBLIOGRAFIA	

OBJETIVOS

1. En la actualidad las memorias EPROM tienen una gran demanda en el campo de la publicidad, sistemas industriales de control, computadoras, aplicaciones domésticas, etc. Es nuestra intención al presentar este trabajo, dar una alternativa de solución a la necesidad de programadores de memorias EPROM que tienen estudiantes, ingenieros, personal de mantenimiento, etc.

2. Con la implementación de este programador de memorias EPROM se esperará poder programar y leer las memorias EPROM 2716, 2732, 2764, 27128, 27256 y 27512, através de una computadora personal compatible con IBM.

3. Al diseñar este programador de memorias EPROM por computadora, se pretende llegar a conocer en una forma práctica una de las muchas aplicaciones que tienen las computadoras personales. Para nuestro caso es experimentar el control por computadora de un dispositivo periférico de diseño propio.

4. La mayoría de los programadores de memorias EPROM permiten grabar varias EPROM de diferentes características, tienen la capacidad de leerlas, y algunos hasta pueden almacenar el contenido de las EPROM en otros medios de almacenamiento. Deseamos que este programador cumpla con las funciones básicas de los programadores de memorias que se pueden encontrar en nuestro medio.

INTRODUCCION

En este trabajo de graduación, titulado PROGRAMADOR DE MEMORIAS EPROM POR COMPUTADORA se presenta una aplicación de las computadoras como control de un dispositivo de hardware. También se hace mención sobre la importancia que tienen en la actualidad los programadores de memorias EPROM, y las diversas formas que hay para programar las EPROM. Como un programador de memorias por computadoras es un dispositivo de hardware controlado por software, es necesario hacer un estudio breve a lo que es el control por computadora, en el cual veremos como sacar información de la computadora hacia un dispositivo externo o por el contrario obtener información de un dispositivo externo hacia la computadora.

Finalmente se hablará específicamente sobre el hardware del programador de memorias EPROM por computadora, explicando cada una de sus partes y las formas en que se van generando las señales que controlan a estas secciones del programador. Como el programador es controlado por la computadora se presenta el software utilizado para realizar las diferentes funciones que el programador contempla.

CAPITULO I

PROGRAMADORES DE MEMORIAS

1.1 DESCRIPCION GENERAL

Los programadores de memorias (PM), son aparatos que permiten almacenar información en las memorias, con el objetivo de ser retomada cuantas veces se quiera la información para diversas aplicaciones, las cuales pueden ser sistemas de computadoras, controladores de luces, avisos luminosos, etc. Inicialmente se puede dividir el sistema de programación en dos partes básicas: el programador memoria propiamente dicho y la memoria.

Se puede mencionar que el programador de memorias debe ser capaz de indicar a la memoria en que parte de ella será almacenada la información que se desea, que información será almacenada, proporcionar las alimentaciones adecuadas para la operación de programación y la generación de la señal que efectúa la programación. Estas se puede decir, a rasgos generales, que son las funciones básicas de un programador de memoria. Dependiendo de la complejidad del programador, es decir de los alcances que este tenga, se pueden encontrar programadores en el mercado que presenten otras funciones entre las que se pueden mencionar el duplicado de memoria, la programación de diferentes clases de memorias, programación desde un teclado, programación por sistema microprocesado, etc. La figura 1.1 muestra un sistema de programación de memoria formado por el programador de memoria y la memoria.

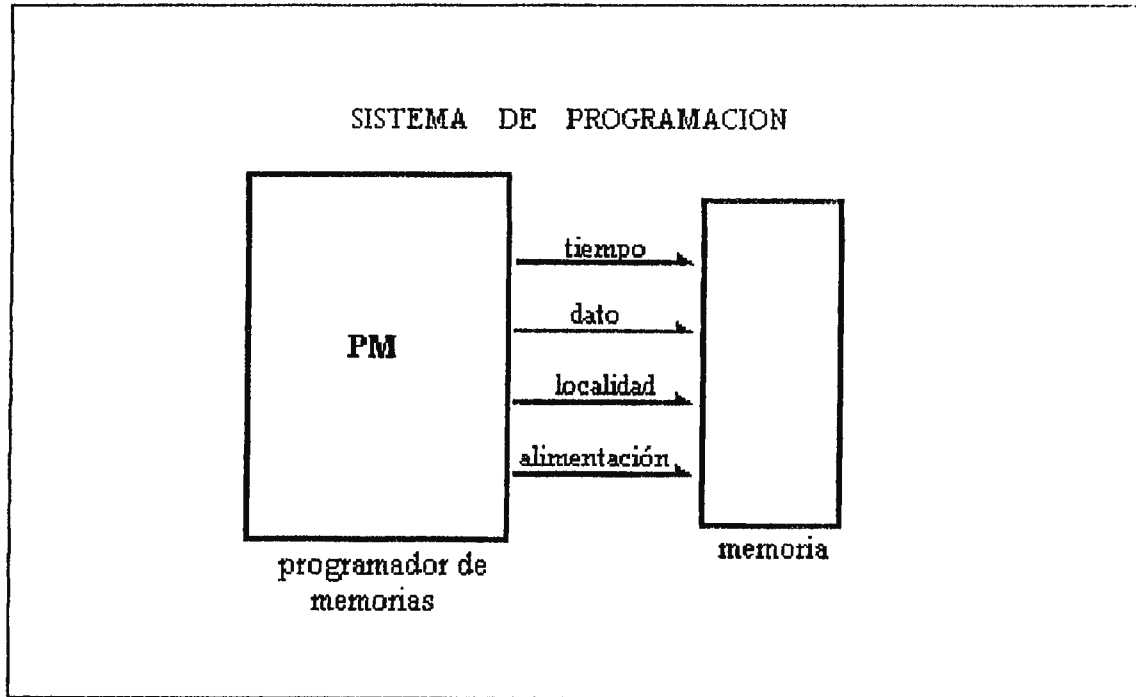


Figura 1.1 Sistema de Programación de Memoria.

1.2 ESTRUCTURA BASICA

Partiendo de las funciones básicas que se mencionaron en la sección anterior el programador de memorias se divide en 4 bloques, los cuales son: Generador de localidades de memoria, Generador de datos a programar, fuente de alimentación y generador de tiempos de programación. La figura 1.2 muestra los cuatro bloques básicos de un programador de memoria. Cada uno de estos bloques entra a funcionar en una forma secuencial de tal modo que cuando se de el último paso (señal de programación) la información , la localidad y las alimentaciones hayan sido seleccionadas.

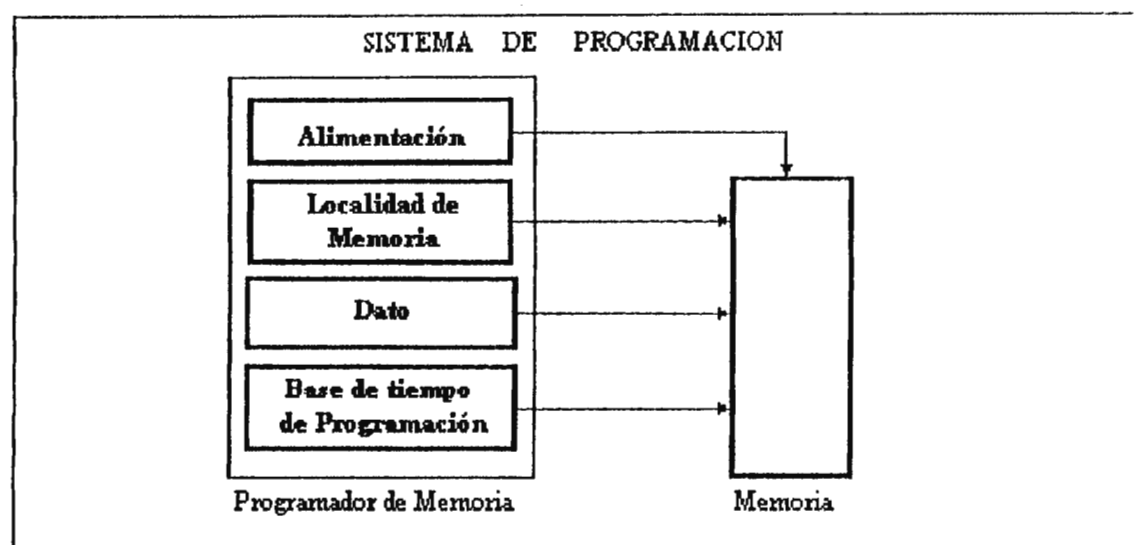


Figura 1.2 Diagrama a Bloques de un Programador de Memorias

La figura 1.3 muestra la secuencia para la programación de una memoria.

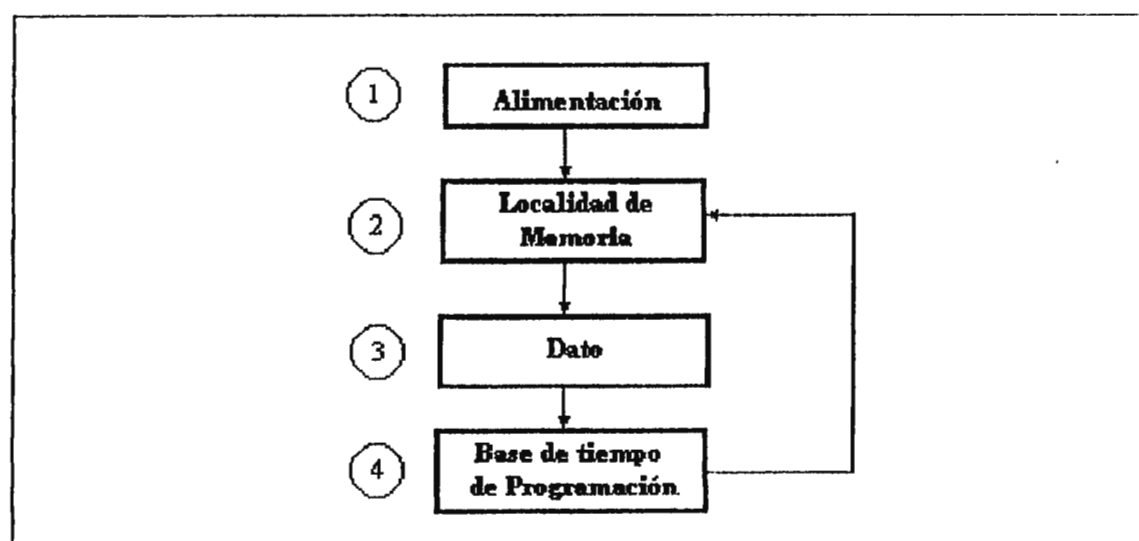


Figura 1.3 Secuencia de Programación de una Memoria

La sección de alimentación es la encargada de proporcionar la alimentación que requiere la memoria para ser programada, y está dispuesta de acuerdo al tipo de memoria; los valores de las alimentaciones son indicadas en las hojas técnicas del fabricante (apendice D).

La sección generadora de localidades de memoria es la que proporciona la información necesaria a la memoria con respecto a las direcciones de las celdas donde se programará una información cualquiera. Esta sección puede estar formada por contadores, por cerrojos que almacenan el número de la localidad, interruptores, etc.

La sección generadora de datos es la que genera la información que será almacenada en la memoria. Estos datos pueden ser generados apartir de un teclado, de otra memoria, de una etapa de cerrojos donde han sido almacenados previamente, contadores, etc. La figura 1.4 , muestra varias formas para generar datos que serán programados en una memoria y la información que indica a la memoria cual de sus celdas serán utilizadas para almacenar un dato.

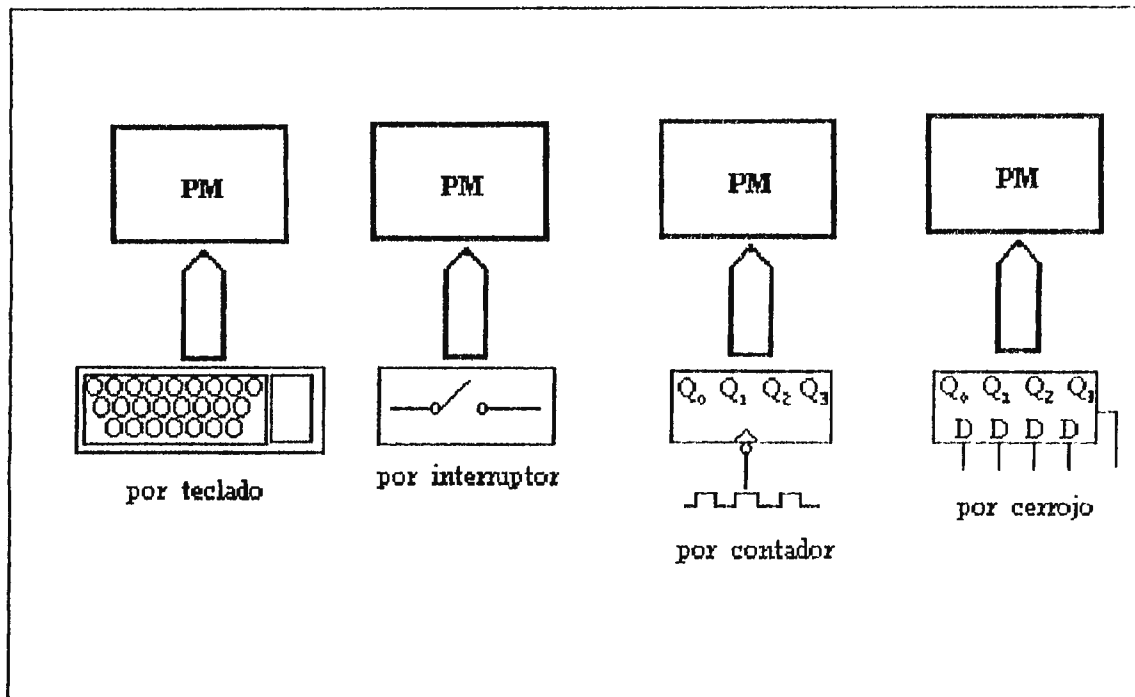


Figura 1.4 Generadores de Localidades de Memoria y Datos

La sección generadora de tiempo de programación puede ser un temporizador monoestable que es disparado manual o automáticamente, o ser generado apartir de otro circuito generador de base de tiempo, etc. La figura 1.5, muestra un oscilador monoestable como generador de tiempo de programación.

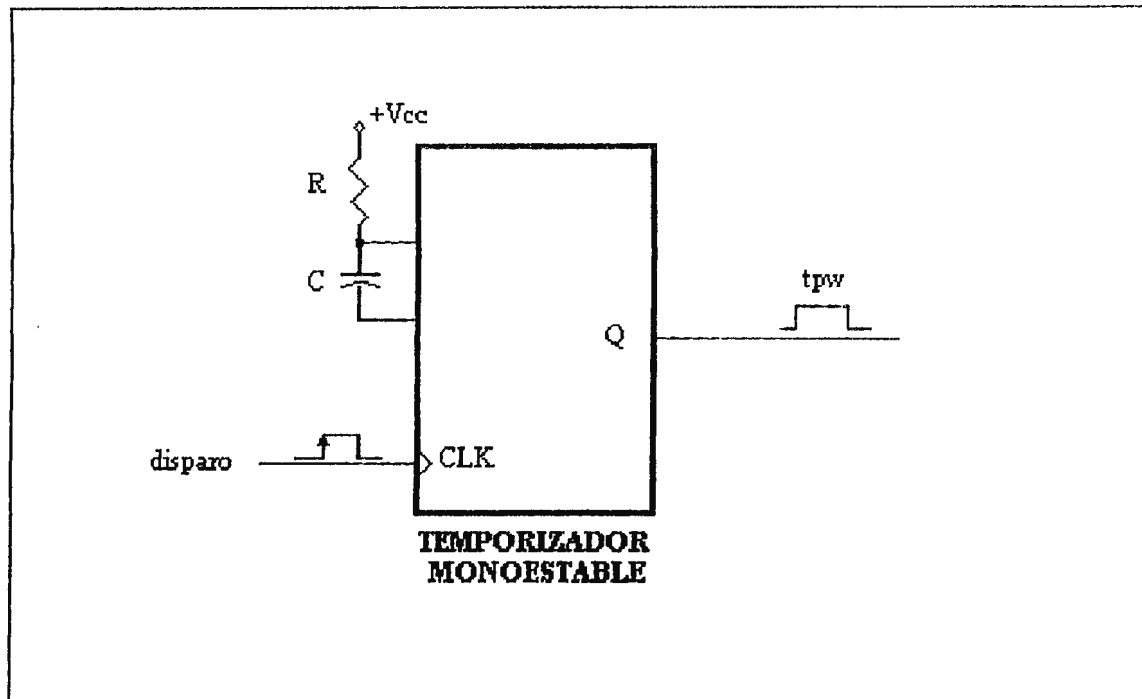


Figura 1.5 Generador de Tiempo de Programación a partir de un Multivibrador Monoestable.

CAPITULO II

CONTROL POR COMPUTADORA

2.1 CONTROL POR COMPUTADORA

Las computadoras personales IBM serán el marco de referencia como control para la aplicación de este proyecto, sin embargo podría utilizarse cualquier computadora no compatibles a IBM teniendo en cuenta los ajustes necesarios en lo que al software y hardware respecta. Se comenzará por definir lo que es un control por computadoras observando la figura 2.1, la cual muestra, como una computadora puede ser conectada para dirigir el funcionamiento de otro dispositivo periférico, esto en esencia, indica que el control por computadora es: hacer que la computadora dirija la

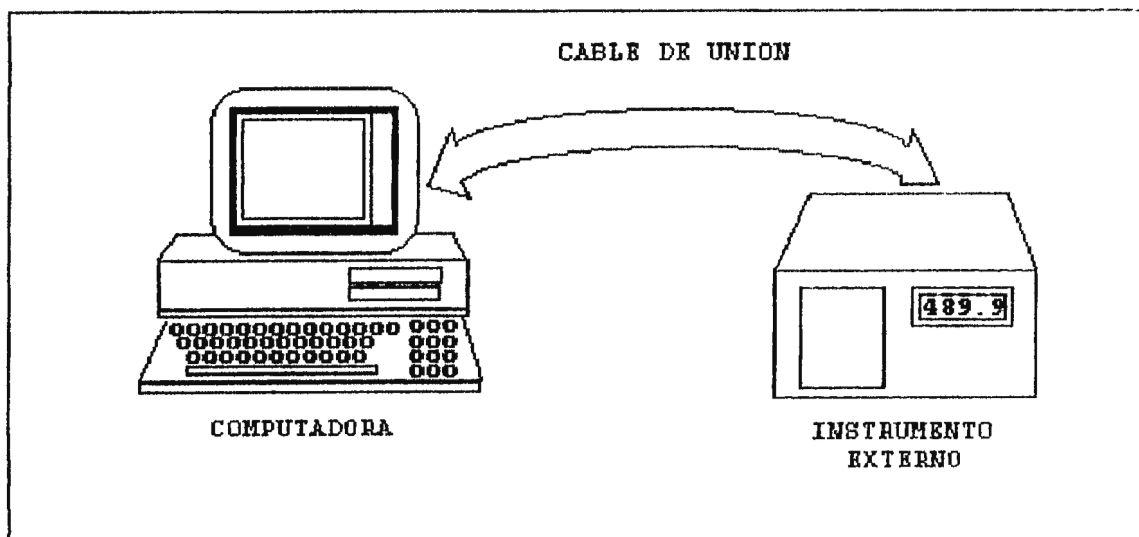


Figura 2.1 Una computadora conectada para dirigir un equipo externo. The IBM PC Connection, Coffron-James W.

sección física o eléctrica de un dispositivo de hardware externo. En la mayoría de aplicaciones de control por computadora debe haber un medio de entendimiento como el hardware externo esta respondiendo a su control. En otras palabras la computadora no sólo dirige la acción de un hardware externo , sino que también lo monitorea.

La figura 2.2 muestra la parte del proceso en la cual interviene la recepción de información proveniente del hardware externo o de la computadora basándose en la información que recibe, la computadora puede modificar las ordenes que le da al hardware externo. Este ejemplo ilustra los elementos básicos del control por computadora. Nótese que hay dos procesos involucrados.

- La computadora manda ordenes al hardware externo.
- La computadora recibe información del hardware externo.

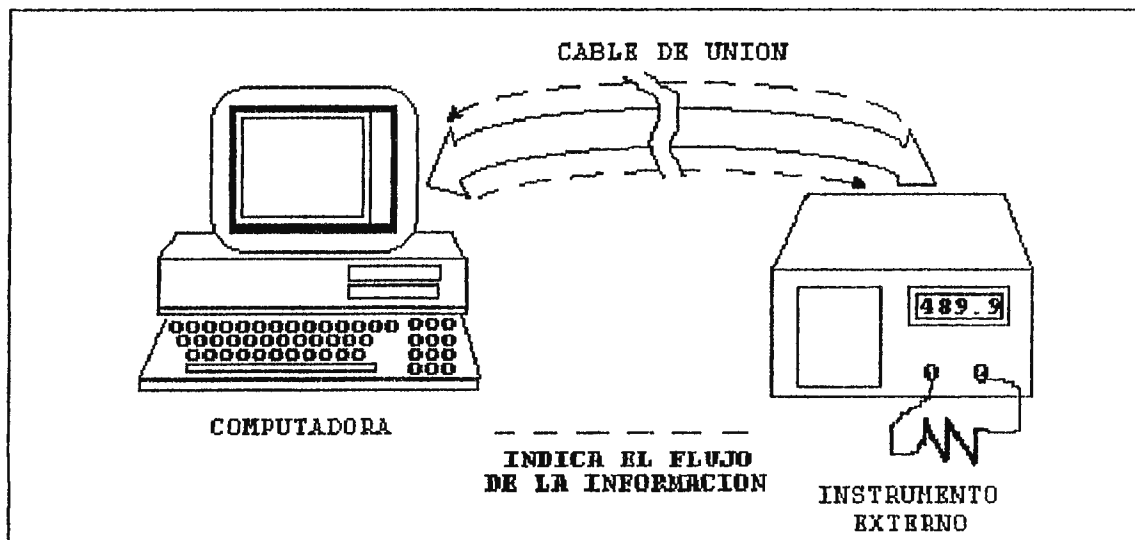


Figura 2.2 La computadora envia y recibe información del equipo externo. The IBM PC Connection, Coffron-James W.

2.1.1 INSTRUCCION OUT

Para llevar acabo la implementación de los dos procesos básicos mencionados anteriormente se escribirá el programa utilizando QBASIC, debido a que presenta una gran similitud con BASIC y fácil programación. Corriendo un programa desde QBASIC se puede direccionar información de cualquier tipo a la tarjeta E/S (entrada o salida), y también se puede tomar información en la tarjeta E/S para procesarla. Para sacar información digital desde un programa de QBASIC se usará la instrucción OUT la cual describe a continuación: La sintaxis de la instrucción es:

OUT (DIRECCIÓN,DATO)

La dirección usada por la instrucción OUT indica el espacio físico en la IBM-PC, donde la información va a ser depositada desde el programa. Algunas de las direcciones de salida son reservados para el uso exclusivo de la IBM-PC.

Para el programador solo se utilizarán 8 de los 65,535 direcciones de salida. La figura 2.3, muestra como se especifica la dirección donde se desea enviar la información. El segundo elemento de la instrucción OUT es el dato (la información digital actual que va a ser entrada a la dirección específica. La dirección específica a donde, de todo el sistema, va a ser enviada la información).

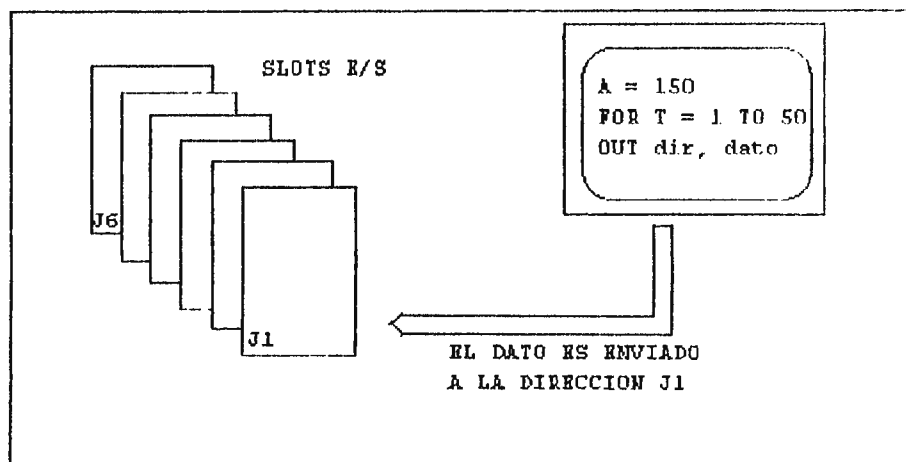


Figura 2.3 Como un programa en ejecución, selecciona uno de varios dispositivos. The IBM PC Connection, Coffron-James W.

2.1.2 FORMANDO LA DIRECCION DE SALIDA

A continuación se presentará como formar la dirección correcta, dependiendo de las direcciones E/S del dispositivo de salidas.

Cada SLOT de salida de la IBM-PC, tiene acceso a cualquiera de las direcciones de salidas que se puedan formar (cualquier dirección de 0 hasta 65,535) algunas de las direcciones de salidas son usadas solamente por la IBM-PC, la tabla 2.1, muestra las direcciones de salida que no pueden ser usadas para ningún otro propósito que no sea para el cual el fabricante lo diseño.

Al revisar la tabla 2.1, se pueden observar unos espacios libres, en los rangos de dirección, algunas direcciones que no son usadas por ninguno de los dispositivos periféricos de la IBM-PC.

Por ejemplo, el rango de 200_H - $20F_H$ son usados por el adaptador E/S de juegos, mientras el rango 210_H - $21F_H$ no son usados por ningún dispositivo periférico. y entonces no es mostrado en la tabla 2.1. Como resultado se puede diseñar un dispositivo periférico para que responda a una dirección entre 210_H y $21F_H$.

RANGO HEX	9	8	7	6	5	4	3	2	1	0	ELEMENTO
00-0F	0	0	0	0	0	Z	A3	A2	A1	A0	DMA 8237
20-21	0	0	0	0	1	Z	Z	Z	Z	A0	INT 8259
40-43	0	0	0	1	0	Z	Z	Z	A1	A0	TIM 8253
60-63	0	0	0	1	1	Z	Z	Z	A1	A0	PPI 8255
80-83	0	0	1	0	0	Z	Z	Z	A1	A0	DMA PG R
AX	0	0	1	0	1						NMI MASK
CX	0	0	1	1	0						RESERV
EX	0	0	1	1	1						RESERV
3F8-3FF	1	1	1	1	1	1	1	A2	A1	A0	RS-323-C
3F0-3F7	1	1	1	1	1	1	0	A2	A1	A0	5 $\frac{1}{4}$ DRV
2F8-2FF	1	0	1	1	1	1	1	A2	A1	A0	RESERV
378-37F	1	1	0	1	1	1	1	Z	A1	A0	PRTR PRT
3D0-3DF	1	1	1	1	0	1	A3	A2	A1	A0	CGA
278-27F	1	0	0	1	1	1	1	Z	A1	A0	RESERV
200-20F	1	0	0	0	0	0	A3	A2	A1	A0	GAME I/O
3B0-3BF	1	1	1	0	1	1	A3	A2	A1	A0	IBM MD

Tabla 2.1 Mapa de direcciones E/S para una computadora IBM.

* Z= no importa. The IBM PC Connection, Coffron-James W.

Para el caso se utilizará el rango de direcciones E/S entre 2B8_h y 2BF_h, estas direcciones no son usadas por ningún dispositivo periférico lo cual lo deja disponible para el uso del programador. Hay varios rangos que puedan ser utilizados, este rango puede ser seleccionado arbitrariamente.

Asumiendo que todas las direcciones de salida son mayores o iguales que 2B8_h y menores o iguales 2BF_h. Si se hace esto la operación de la IBM-PC no se verá afectada.

Usando esta información se puede seleccionar la dirección apropiada para la instrucción OUT, simplemente conociendo en que dirección se encuentra el circuito que se va a controlar. El hardware con que se quiere comunicar debe tener una dirección o direcciones de salida. Para mandar información a este hardware se debe conocer las direcciones exactas. Para mandar información al hardware se usará la instrucción de la siguiente manera:

OUT 2BF,DATO

Nótese que, en este ejemplo de la instrucción OUT, sólo se especifica la dirección, el contenido del dato no es relevante todavía. El dato va ha ser enviado a cualquier dispositivo de salida que resida en la dirección especificada. En este caso, va ha ser el Programador. Luego el dato va a ser enviado a él electricamente.

2.1.3 INSTRUCCIÓN INP

La instrucción INP es la instrucción complementaria de la instrucción OUT, tomando en cuenta que las direcciones de salidas consideradas anteriormente en la instrucción OUT. ($2B8_H - 2BF_H$), se puede tomar una como dirección de entrada y específicamente para el caso $2BB_H$.

VARIABLE = INP DIRECCIÓN

Al ejecutar esta instrucción con la dirección apropiada la IBM-PC toma el dato que se encuentra en el dispositivo periférico que reside en esta dirección.

Hay que tomar en cuenta que para la instrucción OUT se debe controlar el sentido de la información que es bilateral, de la IBM-PC hacia el dispositivo periférico y por el contrario con la instrucción INP el sentido deberá ser desde el dispositivo periférico hacia la IBM-PC.

Esto es de suma importancia cuando estos dos modos de dar y obtener información se combinan en un sólo programa y un dispositivo de programa como lo es el circuito del programador de memorias.

2.2 HARDWARE DE ENTRADA Y SALIDA PARA IBM-PC

En esta sección se verá el hardware interno y externo necesario para la entrada y salida de datos desde y para una IBM-PC.

2.2.1 SALIDA ELECTRONICA POR IBM-PC

Se puede dividir este proceso en cuatro principales secciones electrónicas digitales:

- 1.- El circuito habilitador para el dispositivo externo .
- 2.- La señal de escritura para salida desde la computadora.
- 3.- La señal de escritura para el circuito.
- 4.- El almacenamiento o captura de datos de salida

Estas cuatro secciones operan juntas para realizar la función de salida de datos. Si una de ellas falla al realizar la operación de salida física, no trabajará.

CIRCUITO HABILITADOR

Cuando un circuito habilitador esta activo significa que la computadora esta direccionando electricamente el circuito de salida. La IBM-PC puede comunicarse con más de 64000 circuitos diferentes. Es por eso que tiene que haber un medio de información eléctricamente a un circuito de salida que esta siendo seleccionado para comunicarse con él. Esta comunicación es realizada internamente en la computadora através de las líneas llamadas

líneas de dirección, estas líneas son salidas, en los conectores E/S de la IBM-PC. La figura 2.4 muestra el pin-out de un conector E/S de una IBM-PC 8088, el cual es la base para las siguientes generaciones de máquinas, la figura muestra el número de pin y el nombre de la señal de cada pin.

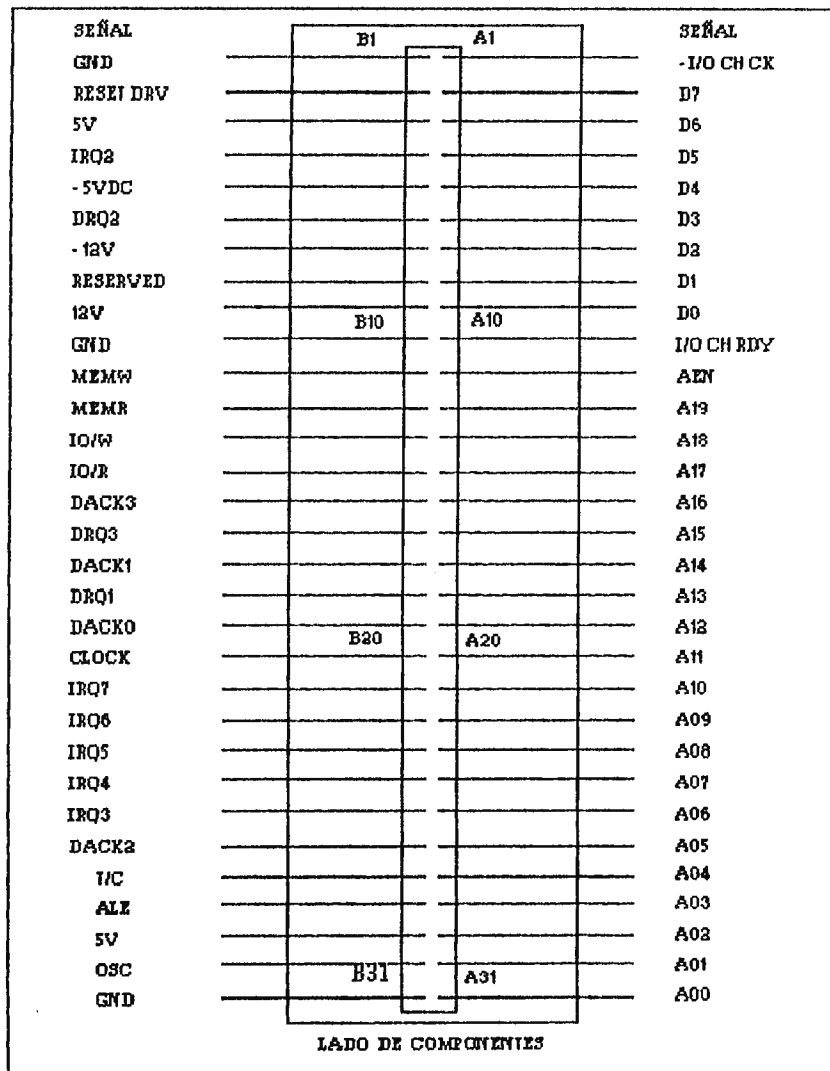


Figura 2.4 Diagrama de un Slot E/S de una computadora IBM. The IBM PC Connection, Coffron-James W.

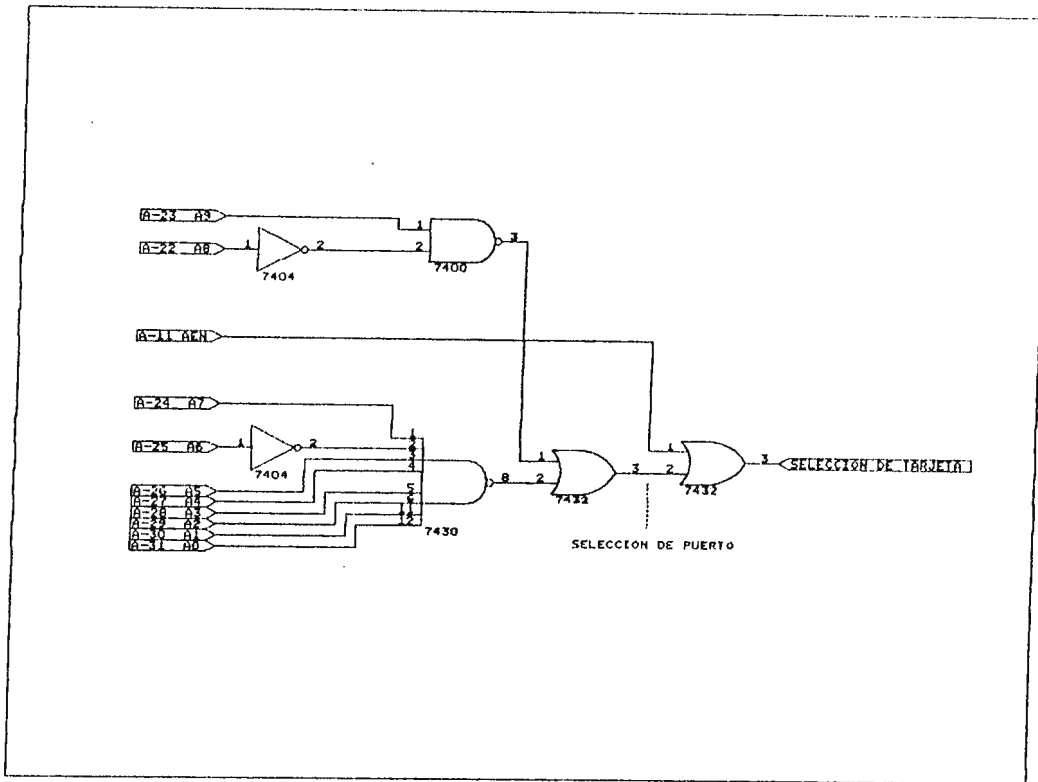


Figura 2.5 Circuito decodificador. The IBM PC Connection, Coffron-James W.

La señal habilitadora que interesa será generada por la combinación lógica que existe en las líneas de dirección $A_0 - A_7$. Retomando la instrucción OUT en QBASIC es necesario especificar una dirección. Esta dirección tendrá que ser puesta en las líneas de dirección de la computadora. Teniendo que reconocer electricamente cuando esta dirección está en las líneas de dirección de la computadora, la figura 2.5 muestra un circuito simple que reconocerá la dirección $2BF_{16}$. La figura 2.5 nos muestra que cuando las líneas de dirección $A_0 - A_7$ son iguales a : 1010111111 ($2BF_{16}$), la señal denominada SELECCION DE PUERTO es un cero lógico. Para todas las demás veces la señal selección de puerto es un 1 lógico. La idea principal con

este ejemplo es que se tiene un significado lógico de la dirección cuando el número 2BF es puesto en el bus de direcciones.

Además la figura 2.5 muestra la señal llamada AEN, esta línea es usada para indicar eléctricamente al sistema que la línea de dirección, dato y bus de control están siendo controlados por el microprocesador.

Cuando esta señal AEN es un 0 lógico la dirección de salida en la línea de dirección $A_0 - A_9$ es válida y puede ser usada para la decodificación de una dirección de salida seleccionada. Siempre que AEN sea un 1 lógico, se indicará que la dirección deseada no está siendo generada por el microprocesador, y no puede ser usada para decodificar una salida de dirección.

Al diseñar un dispositivo hardware de E/S, el AEN, debe ser usado y detectado para un cero lógico, como podemos ver en la figura 2.5. La línea de selección de tarjeta no será un cero lógico a menos que la línea selección de puerto y AEN sean 0 lógicos al mismo tiempo.

SEÑAL DE ESCRITURA DE SALIDA.

La siguiente señal es llamada -IOW, cuando esta señal es un 0 lógico la IBM-PC está sacando datos a un circuito E/S. Durante la ejecución de una instrucción OUT esta señal será un 0 lógico porque la computadora estará escribiendo datos en una dirección de

salida especificada en el comando OUT. Esta señal esta cambiando constantemente porque la IBM-PC está comunicándose con los circuitos E/S aún cuando no los estemos dirigiendo. La computadora realiza esta acción cuando se esta comunicando con el teclado, un impresor o una pantalla de video.

SEÑAL DE ESCRITURA PARA EL CIRCUITO

Esta señal es generada por la combinación de la señal -IOW y la señal selección de tarjeta del circuito de la figura 2.5, cuando ambas señales son 0 lógicos se genera una nueva señal, llamada señal de escritura de tarjeta. La figura 2.6 ,muestra un diagrama a bloque de este concepto.

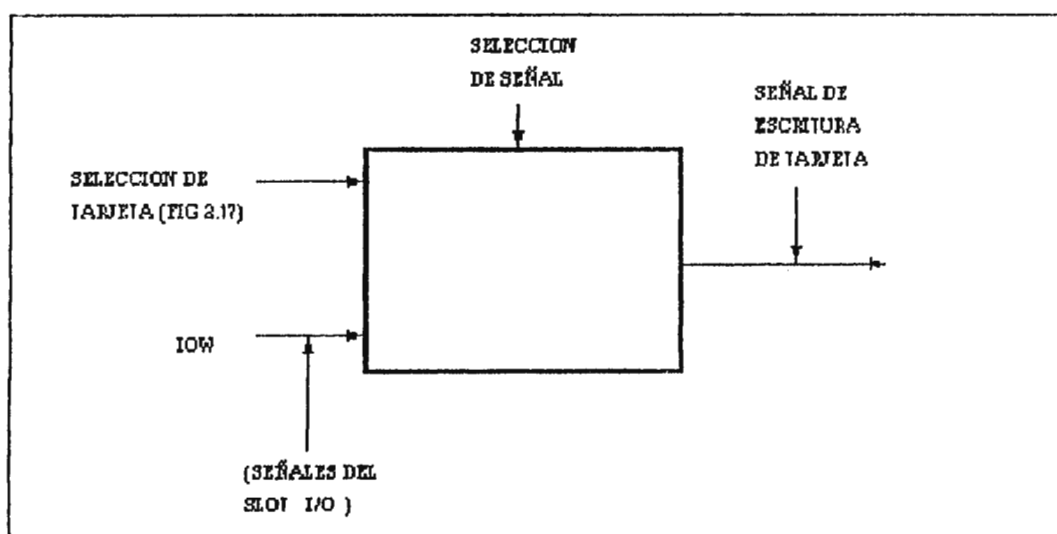


Figura 2.6 Diagrama a bloques que muestra como la señal Selección de Tarjeta y la IOW generan la señal de escritura de tarjeta. The IBM PC Connection, Coffron-James W.

La figura 2.7 muestra un esquema completo para generar la señal de escritura de tarjeta, esta señal se vuelve activa cuando la IBM-PC ejecuta la instrucción OUT.

CAPTURA DE DATOS DE SALIDA

Esta captura se realiza con la utilización de Flip-flop tipo "D" los cuales capturan el dato presente en las líneas de datos $D_0 - D_7$. Estos cerrojos tienen una función importante, ya que almacenan temporalmente el dato enviado por la IBM-PC al circuito externo. Recordemos que la instrucción OUT incluye una dirección y un dato, la dirección selecciona el circuito E/S y el dato especificado es enviado al circuito externo, los cerrojos de salida almacenan el dato.

La computadora presenta el dato en los pines de entrada de los cerrojos de salida (figura 2.8). En este momento la señal de escritura de tarjeta da un pulso de escritura a los cerrojos, y se mantienen almacenados ahí hasta que la computadora sea instruida nuevamente para escribir otro dato en la misma dirección.

El dato en la salida de los cerrojos puede ser usado para controlar cualquier hardware que deseemos. La figura 2.9 muestra un dispositivo de cerrojo (74LS374), que podría utilizarse.

La IBM-PC transferirá un dato de 8 bits cada vez que se ejecuta una operación OUT. Los datos desde la IBM-PC serán aplicados a los pines de entrada de datos del cerrojo 74LS374.

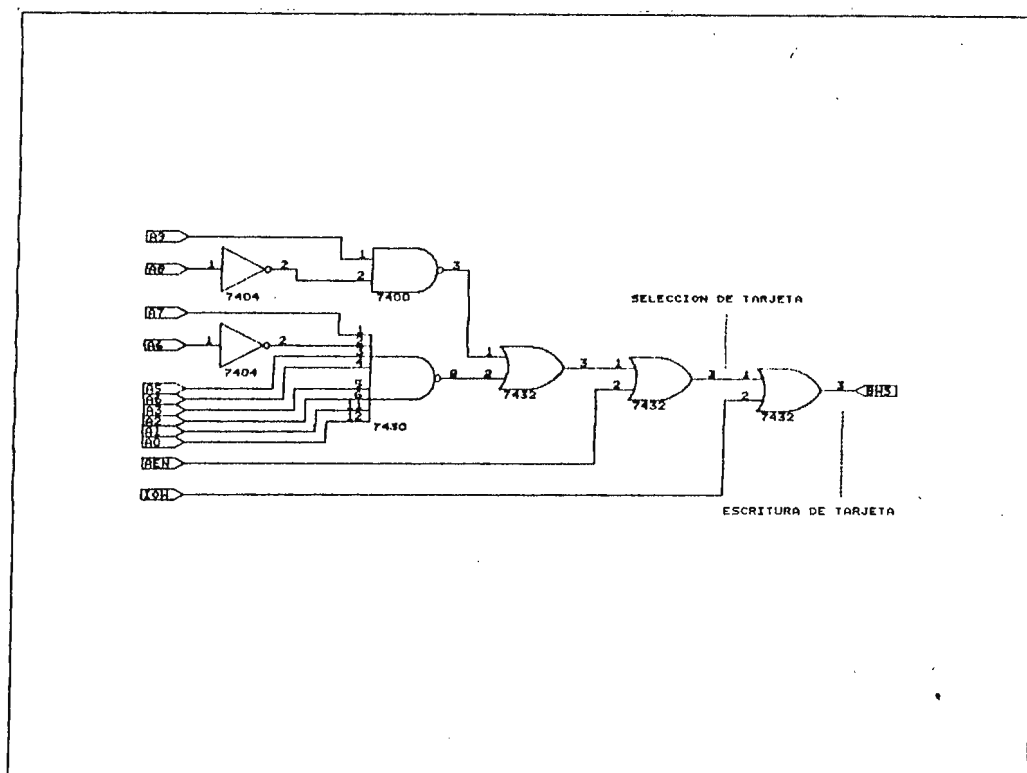


Figura 2.7 Generación de señales. The IBM PC Connection, Coffron-James W.

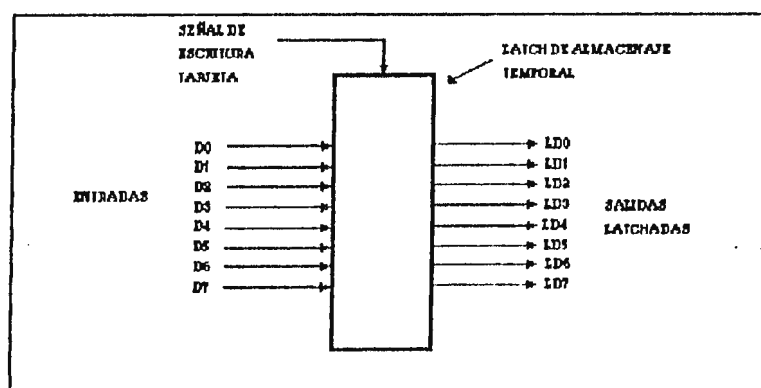


Figura 2.8 Diagrama a bloques mostrando como la salida de datos de una IBM PC será almacenada temporalmente. The IBM PC Connection, Coffron-James W.

El pin 11 del cerrojo es la entrada de habilitación, cuando esta línea es activa , cambia de 0 a 1 lógico, el dato presente a la entrada del cerrojo es capturado y almacenado internamente en el cerrojo.

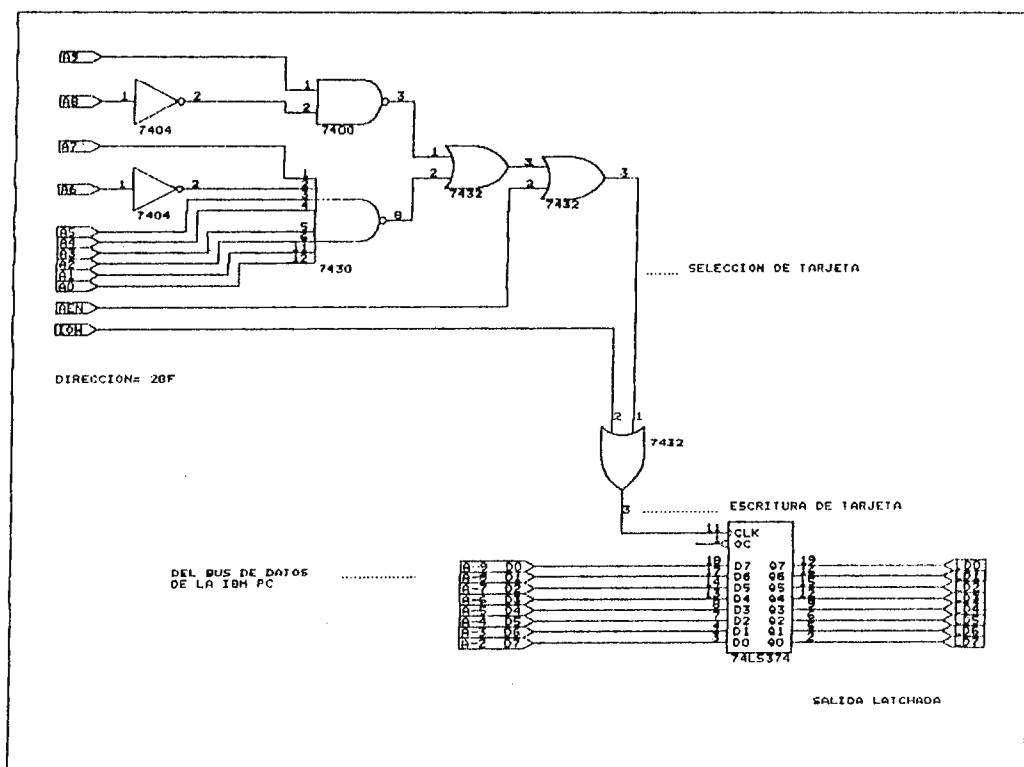


Figura 2.9 Latchador de datos. The IBM PC Connection, Coffron-James W.

2.2.2 HARWARE PARA INTRODUCIR DATOS A LA IBM PC

La idea fundamental es que un dispositivo externo sacará datos y meter estos datos a la IBM-PC. La computadora hará decisiones basándose en el dato de entrada, almacenará el dato o lo alterará de alguna manera.

El hardware de entrada de datos se puede dividir en tres secciones principales.

- 1.- Circuito habilitador para el dispositivo externo
- 2.- Señal de lectura de entrada (-IOR).
- 3.- Señal de lectura para circuito.

Para meter datos a un programa QBASIC, se usará la instrucción INP descrita anteriormente. La instrucción INP utilizará una dirección de donde tomará el dato, esta dirección será decodificada lógicamente para habilitar la línea de selección de tarjeta igual a la del hardware de salida del dato. Esto siempre que el software este realizando una instrucción INP en la dirección correcta para el slot E/S, la línea de selección de tableta será activa con un cero lógico. La instrucción INP estará realizando una función de lectura y la línea *IOR del conector de salida E/S será también un 0 lógico durante una instrucción OUT, las líneas -IOR y las líneas de selección de tarjetas son combinadas lógicamente para generar la línea de habilitación de tarjeta. Esto se muestra en la figura 2.10.

Durante el tiempo que la IBM-PC esta leyendo datos del slot E/S seleccionado, el dato del circuito es eléctricamente habilitado en las líneas de dato $D_0 - D_7$, como lo muestra la figura 2.11. La figura 2.12, muestra un diagrama esquemático completo de un circuito que introducirá datos a la IBM - PC desde una fuente externa. En el centro de la figura 2.12 está el buffer de 3 estados 74LS244, Las entradas son llamadas $CD_0 - CD_7$. Asumiendo que en estas líneas se encuentra presente alguna información digital para ser enviada a la IBM-PC.

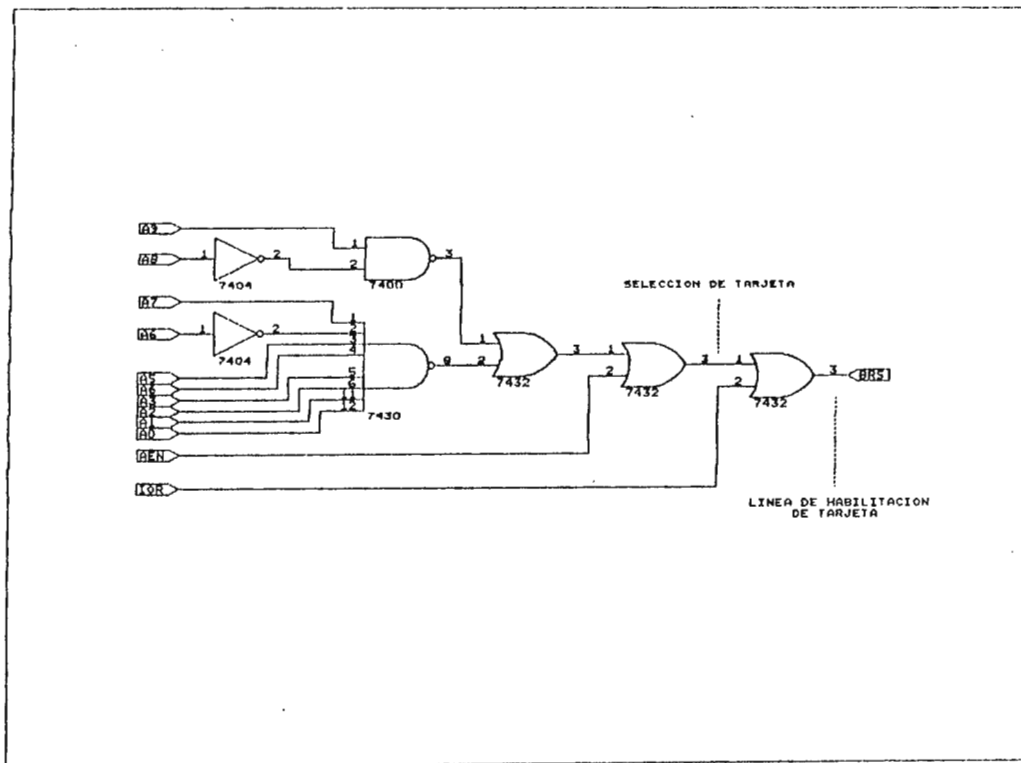


Figura 2.10 Hardware para lectura. The IBM PC Connection, Coffron-James W.

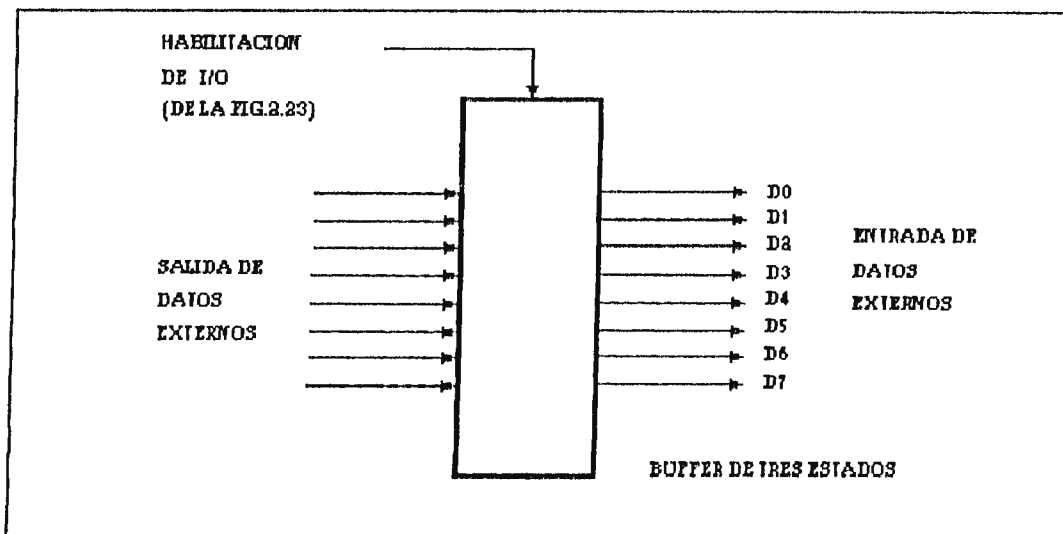


Figura 2.11 Diagrama a bloques indicando que los datos de un equipo externo están entrando a la IBM PC a través de un buffer de tres estados. The IBM PC Connection, Coffron-James W.

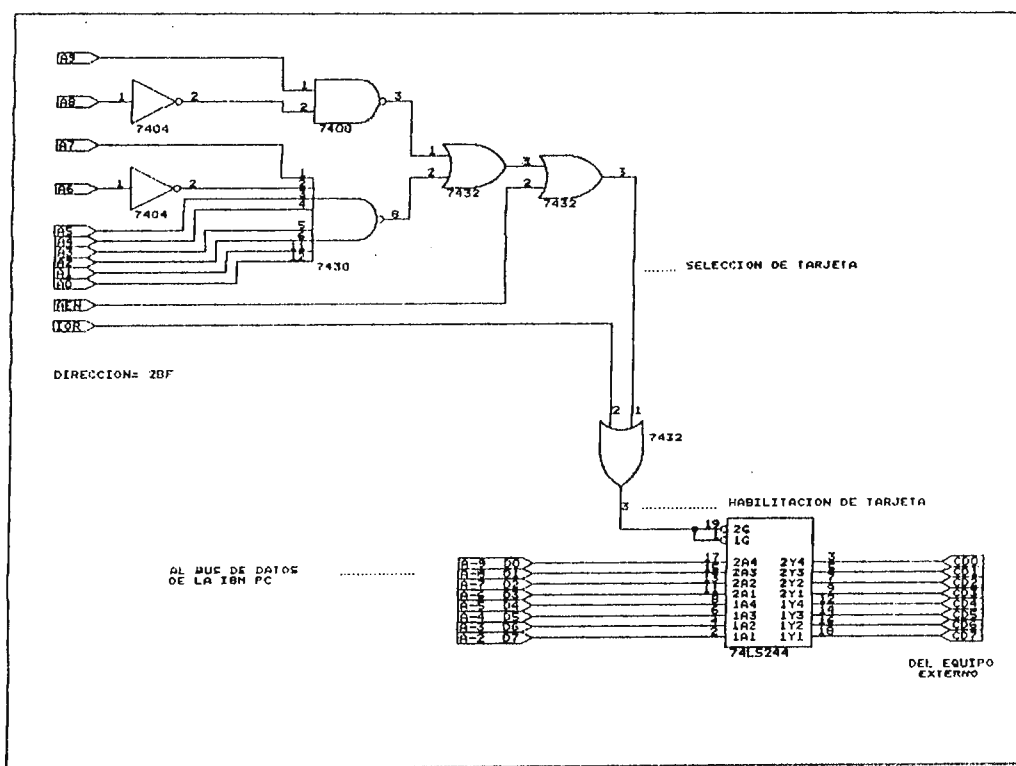


Figura 2.12 Diagrama completo de lectura. The IBM PC Connection, Coffron-James W.

Las salidas del 74LS244 de la figura 2.12, son conectadas al bus de datos $D_7 - D_0$ de la IBM-PC. Estas salidas no deberán estar habilitadas a menos que la IBM-PC este requiriendo eléctricamente un dato del circuito de entrada. Para todos los demás tiempos la salida del dato del buffer deben estar en alta impedancia (3 estados). Este modo remueve eléctricamente las salidas del 74LS244 de las líneas del bus de datos solo cuando el pin de habilitación será activado por las líneas - IOR y selección de tarjeta através de una compuerta OR.

2.2.3 PASOS QUE OCURREN DURANTE UNA LECTURA O UNA OPERACIÓN INP DESDE EL CKTO E/S

1.- La IBM-PC sacará la dirección correcta que habilitará la línea de selección de tarjeta, en este momento la línea AEN será puesta en un 0 lógico para la IBM-PC.

2.- La línea -IOR se pondrá a 0 lógico. Esto indica eléctricamente al sistema que la IBM-PC está en la espera de un dispositivo externo para poner el dato en el bus de datos del sistema. Cuando la línea -IOR va a 0 lógico habilitará el 74LS244 y el dato a la entrada del buffer es puesto en el bus de datos de la IBM-PC.

3.- Durante el tiempo que el dato externo queda habilitado en el bus de datos del sistema, la IBM-PC leerá automáticamente el dato.

4.- Después de un período de tiempo (menor que la millonésima parte de un segundo) la línea -IOR irá a 1 lógico. esto ocurre automáticamente através de los circuitos de la IBM-PC. Cuando esto ocurre el 74LS244 es deshabilitado poniéndolo en alta impedancia removiendo eléctricamente la salida del buffer de las líneas del bus de datos.

CAPITULO III

PROGRAMADOR DE EPROM POR PC

3.1 INTRODUCCION

En este capítulo se describe el funcionamiento del programador de memoria por computadora, visualizando los alcances de este así como sus limitaciones y las posibles alternativas de solución. También se tratará la parte de hardware y software del programador de memorias independientemente, para detallar en cada una de estas secciones las partes básicas de las mismas y la interrelación que van teniendo unas con otras.

En lo que al hardware respecta, se dividirá en cinco secciones principales las cuales realizan una función específica, siendo estas DECODIFICADOR DE TARJETA, DECODIFICADOR DE LATCH Y MONOESTABLE, CAPTURA DE INFORMACION Y ENTRADA/SALIDA DE DATOS, DECODIFICADOR DE PALABRA DE CONTROL Y LOS DECODIFICADORES DE PINES.

Para cada una de estas funciones se presentan tablas de verdad, calculos, señales de entrada/salida y control. Para el software se presenta el flujograma desglosado en las secciones principales, también se presentan las funciones del programador explicando lo que se hace al accesarlas.

3.2 DESCRIPCION GENERAL DEL FUNCIONAMIENTO

Como mencionamos anteriormente que un programador de memorias es un aparato que puede programar un dispositivo de memoria. Se hará referencia a un programador de memorias EPROM.

Las memorias EPROM poseen la ventaja de que pueden ser borradas y ser programadas cuantas veces quiera, por lo que se hará alusión a la programación de este tipo de memorias, a través de un dispositivo, el cual tendrá como función general ser una interface de la IBM-PC a la memoria EPROM y de la memoria EPROM a la IBM-PC.

Podemos definir los pasos básicos de un programador de memorias como sigue:

- 1.- Seleccionar la dirección de la localidad de memoria.
- 2.- Mantener el dato que deseamos almacenar en la dirección de memoria deseada.
- 3.- Dar una señal de grabación que almacene el dato en la dirección seleccionada.

Ahora bien, si se quiere un programador que permita programar algunos números de memorias EPROM, duplicar la información de una memoria a otra, almacenar el contenido de una memoria o simplemente

ver el contenido de una memoria, se puede diseñar un aparato que reúna todas estas funciones en un solo HARDWARE, el cual puede ser dirigido por un programa de computadora que realiza cada una de las funciones antes mencionadas. Planteando las funciones generales que realizará el programador de memorias tenemos:

- 1.- Guardar el contenido de una EPROM en un disco flexible o duro, en forma de archivo.
- 2.- Guardar un archivo en un disco flexible o duro en una memoria EPROM.

La función de guardar el contenido de una memoria EPROM a un disco flexible o duro constará de los siguientes pasos:

- 1.- Seleccionar el número de EPROM.
- 2.- Seleccionar el rango de localidades de memoria a leer.
- 3.- Dar un nombre al archivo que se creará con su respectiva ruta de acceso.

Después de todos estos pasos se puede visualizar el contenido del archivo el cual presentará los datos con su respectiva localidad de memoria.

Para guardar un archivo de disco flexible o duro en una EPROM . De antemano debemos de grabar un archivo con los datos y las localidades de memoria que deseamos en la memoria EPROM.

1.- Seleccionar el número de EPROM donde se grabará el contenido del archivo.

2.- Se indica al programa el nombre del archivo de donde tomará la información.

3.- El programa presentará en el programador de memoria las localidades seleccionadas.

4.- El programador sacará el dato que corresponde a esa localidad

5.- El programador activará la señal de grabación.

3.3 HARDWARE DEL PROGRAMADOR

Dividiremos el programador de memorias EPROM por PC en cuatro bloques principales, los cuales serán:

1.- Decodificador de tarjetas.

2.- Decodificador de latch y monoestable.

3.- Captura de datos, direcciones , palabra control y entrada/salida de datos.

4.- Decodificadores de pines.

3.3.1 DECODIFICADOR DE TARJETA

Esta etapa es la encargada de seleccionar ocho localidades de puerto E/S, las cuales serán utilizadas para habilitar los dispositivos que permiten almacenar temporalmente las direcciones de la memoria que se grabará o se leerá, para almacenar temporalmente el dato que se grabará en la memoria; leer el dato de una memoria hacia la máquina y generar el pulso de grabación para la EPROM; almacenar temporalmente la palabra binaria que determina el número de la memoria EPROM y el modo de operación. La figura 3.1 muestra el diagrama del Decodificador de Tarjeta.

El decodificador de tarjeta se puede sub-dividir en cuatro bloques:

1.- DECODIFICADOR DE DIRECCIÓN. Este circuito se encarga de responder con un cero lógico a una cierta dirección obtenida del slot de la IBM-PC. Debido a que se necesitan ocho localidades de puerto, el decodificador debe ser capaz de responder a estas ocho direcciones, la figura 3.1, muestra la etapa del decodificador de tarjeta en la cual se puede observar que el bus de direcciones de la IBM-PC se ha dividido en dos partes, la primera esta formada por las dirección A0 - A2 las cuales se utilizarán para habilitar cada

uno de los dispositivos de captura de datos, monoestable, etc. a través de un decodificador. La segunda parte del bus de direcciones está formada por A3 - A9, el circuito decodificador de dirección está conectado a esta parte del bus de

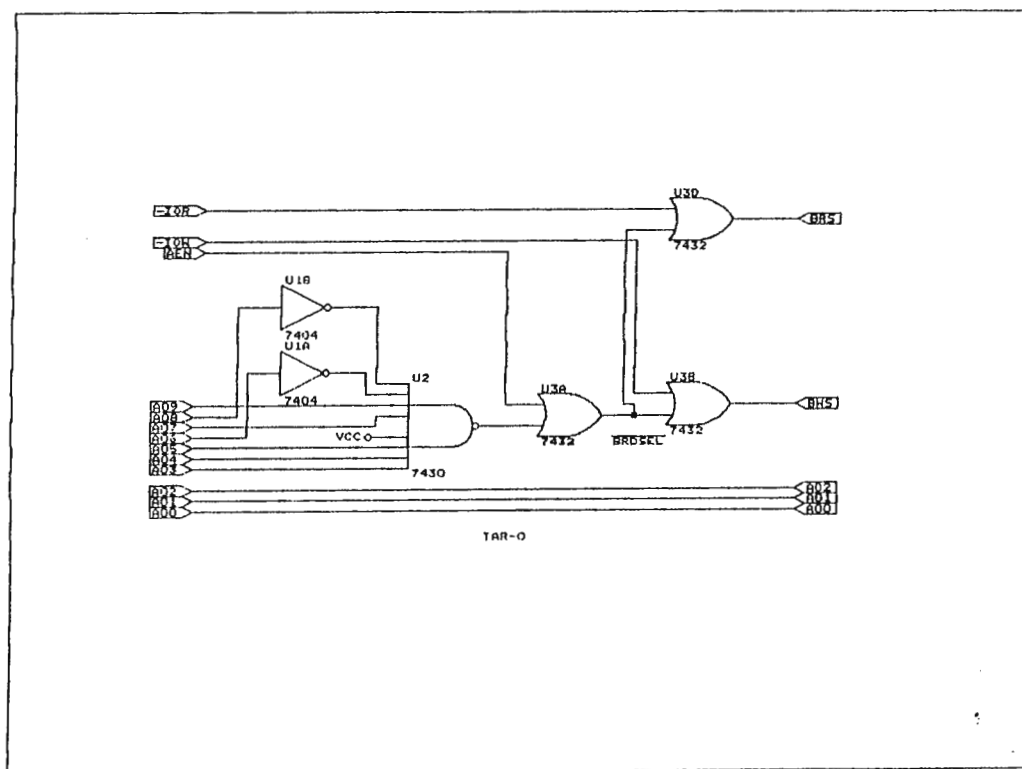


Figura 3.1 Diagrama del Decodificador de Tarjeta.

direcciones y generará un CERO lógico cuando en A3 - A9 tengamos 1010111.

Esta dirección y la línea de señal AEN generará la señal BRDSEL la cual indicará que la dirección ha sido efectiva, a través de una compuerta OR (U2C). La señal BRDSEL indica que la dirección decodificada es válida en ese momento y junto con la línea -IOW del slot generará la señal BWS, la cual será utilizada para habilitar la escritura de los demás dispositivos del programador que tienen la función de capturar información del bus de datos de la IBM-PC ó el disparo de los pulsos de grabación. La misma señal BRDSEL pero con -IOR del slot generarán la señal BRS, la cual será utilizada para habilitar el buffer de entrada de datos.

El bus de datos del slot D0 - D7 solo pasa a través de la tarjeta TAR-0, la cual será instalada dentro de uno de los slot.

3.3.2 DECODIFICADOR DE LATCH Y MONOESTABLE

Esta etapa es la destinada a habilitar los flip-flop que capturan los datos que de la IBM-PC se grabarán en la EPROM, la captura de las 16 líneas que direccionan las localidades de la EPROM que se grabarán, la captura de la palabra de control, y el disparo de los monoestable que generan los pulsos de grabación de la EPROM. La figura 3.2 muestra el circuito del decodificador de Latch y Monoestable.

Las tres líneas de dirección del slot A0 - A2 seleccionan una de las ocho salidas negadas del 74138 (U4), correspondiendo cada una de ellas a una localidad de puerto. Las señales generadas en TAR-0

(BWS) y (BRS), a través de una compuerta AND (U3A), generarán la señal de habilitación para cada uno de los dispositivos (flip-flop, E/S y monoestable). El pin G2B de U4 permanecerá en BAJO y G1 permanecerá en ALTO, quedando G2A como la entrada de la señal BWS/BWR, con la cual se habilitará la escritura o lectura en cualquiera de los dispositivos seleccionados. Así para la ejecución de una instrucción OUT se escribirá en uno de los Flip-Flop o se disparará uno de los tres monoestables, ya que BRS se encuentra en alto y BWS pasaría a bajo, generando un cero a la salida de la AND

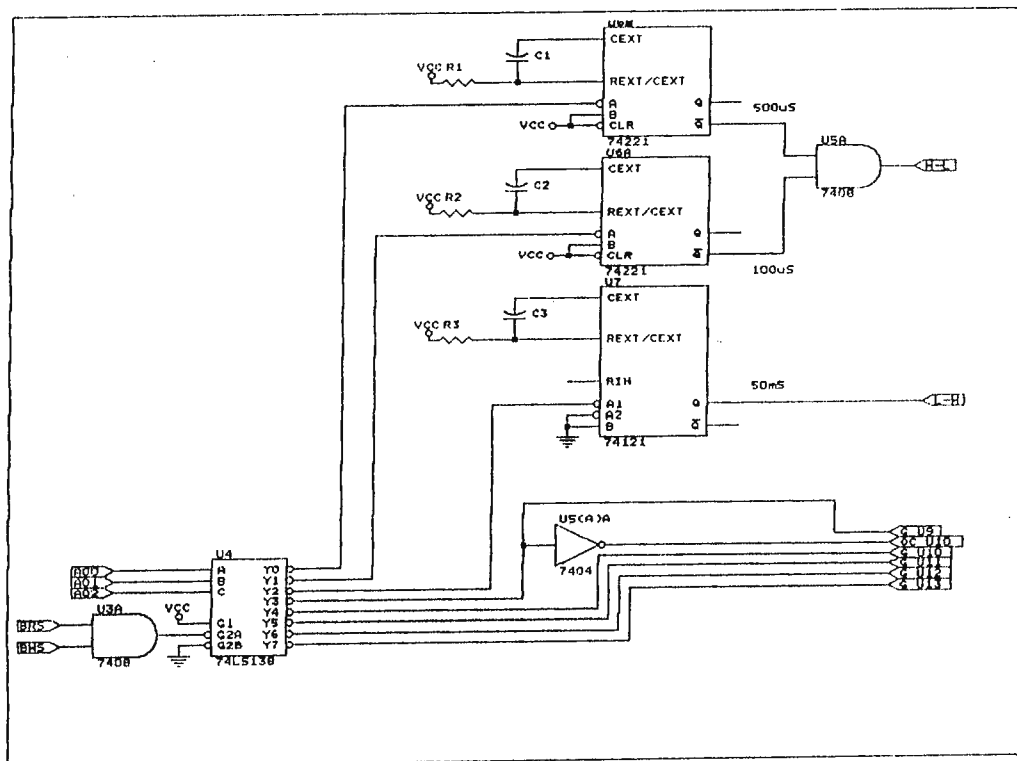


Figura 3.2 Diagrama del decodificador de latch y monoestable.

(U3A). Este cero generado habilitará las salidas de U4, siendo una de éstas la que corresponda a la dirección (A0-A2 de U4) del dispositivo seleccionado. Las direcciones de cada dispositivo de Escritura y Lectura aparecen en la tabla 3.1.

DISPOSITIVO SELECCIONADO	DIR HEX
CAPTURA PALABRA DE CONTROL F/F U13	2BF
CAPTURA DE DIRECCION M00-M07 F/F U12	2BE
CAPTURA DE DIRRECCION M08-M15 F/F U11	2BD
CAPTURA DE DATO D0-D7 PC-EPROM F/F U10	2BC
ENTRADA DE DATO EPROM-PC U9	2BB
DISPARO DE MONOESTABLE 50mS	2BA
DISPARO DE MONOESTABLE 100µS	2B9
DISPARO DE MONOESTABLE 500µS	2B8

Tabla 3.1 Direcciones de los dispositivos del programador.

3.3.3 CAPTURA DE DATOS, DIRECCIONES , PALABRA DE CONTROL Y ENTRADA / SALIDA DE DATOS

Esta etapa es la encargada de permitir la salida de datos que se grabarán en la EPROM, la entrada de datos de una EPROM grabada hacia el programa, la captura de la direcciones de las localidades de la EPROM que se grabarán y la captura de la palabra de control. La figura 4.3 muestra el circuito de los Flip Flops que captura el dato, la direccion de la EPROM y la Palabra de Control, así como el buffer de tres estados que permite el ingreso de datos de la EPROM a la computadora.

El bus de datos D0-D7, que se obtiene desde el slot es conectado a esta etapa, para ser direccionado en los diversos flip-flops. Esto indica que para grabar una memoria cualquiera será necesario almacenar temporalmente :

1-palabra de control.

2-dirección baja de la localidad que se grabará (m0-m7).

3-dirección alta de la localidad que se grabará (m8-m15).

4-dato que se grabará en la localidad capturada.

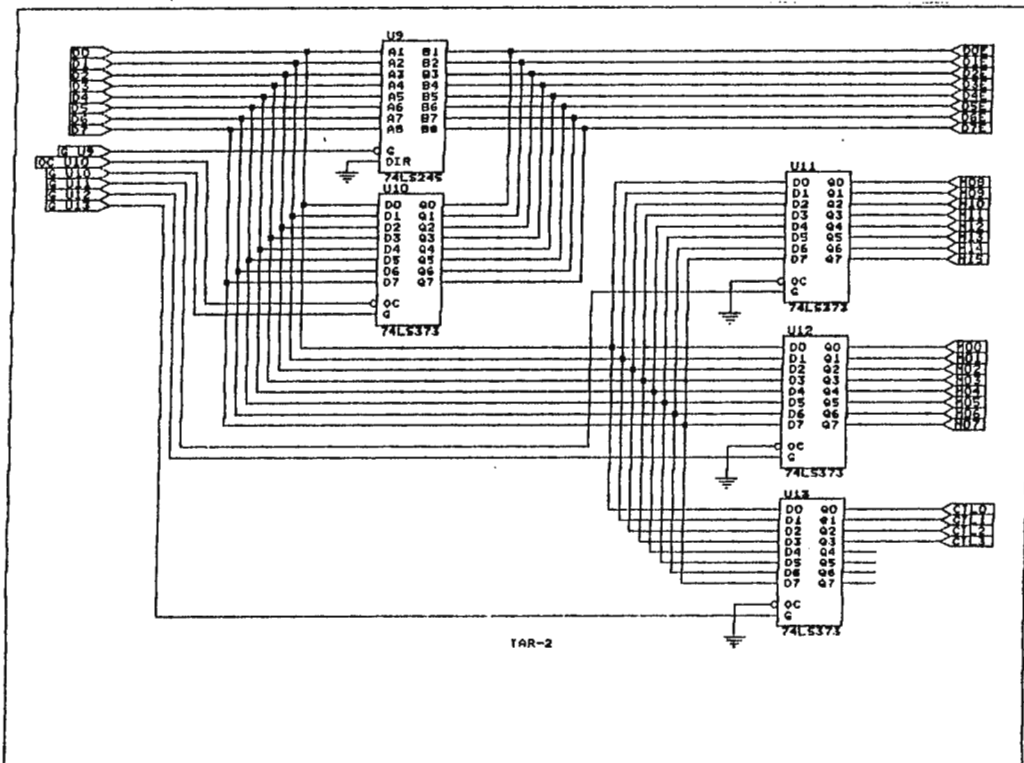


Figura 4.3 Diagrama de E/S, direcciones y palabra de control.

Estos datos capturados serán de 8 bits, cada uno de los cuales quedará almacenado hasta que la localidad haya sido grabada, y el proceso se repetirá hasta grabar todos los datos necesarios. La habilitación de cada uno de estos dispositivos se obtiene de las líneas de salida del demultiplexor U4 de la etapa TAR-1, que se explico anteriormente. Esto ocurre para el f/f U11, U12 y U13, ya que estos solo capturan el dato del bus de datos de la IBM-PC y lo presentan en sus salidas, no así para el f/f U10, debido a que este esta interrelacionado con el buffer bilateral U9. El funcionamiento conjunto de estos dos dispositivos permite hacer una transferencia bilateral en el bus de datos, es decir que permite la captura del dato a grabar en U10, habilitando las salidas 3-estados de éste y al mismo tiempo poniendo en alta impedancia las salidas de U9, logrando así que el dato quede capturado en U10. De la misma forma si se quiere introducir un dato al programa desde la EPROM, Se habilitan las salidas de U9 y se deshabilitan las de U10, dejando pasar el dato desde el puerto B al puerto A de U9. Las habilitaciones para estos dos integrados son generadas en la etapa el circuito de la figura 3.2.

3.3.4 DECODIFICADOR DE PALABRA DE CONTROL

La etapa de decodificación de palabra de control es la que determina el número de la memoria EPROM que se grabará o leerá, pre-estableciendo el modo de operación de la EPROM, esto se logra habilitando ciertas señales que intervienen para el grabado o lectura de un memoria específica, lo que implica un cambio en

algunas líneas ya sean de control, de dirección y voltajes de alimentación o de programación.

Del flip - flop U13 obtenemos una palabra de ocho bits, de la cual sólo utilizaremos 4 bits, dicha palabra deberá permanecer constante durante todo un proceso de lectura o escritura de una EPROM.

Esta palabra es un valor fijo asignado para cada número de memoria EPROM y su modo de operación, es decir que cada memoria EPROM tendrá dos valores para la palabra de control, por ejemplo: para la memoria 2716 habrán dos palabras de control, una que indicará que la memoria a grabar es una 2716 poniendo las líneas de control correspondiente a esa memoria así como los voltajes de alimentación y de programación.

La otra palabra de control para la 2716 será la que indicará que dicha memoria entrará en el modo de lectura poniendo así las líneas de control en los valores adecuados, habilitando el bus de datos de la EPROM e igualando el voltaje de programación con el de alimentación. Las señales que corresponden a cada pin de cada una de las memorias son presentadas en la tabla 3.2.

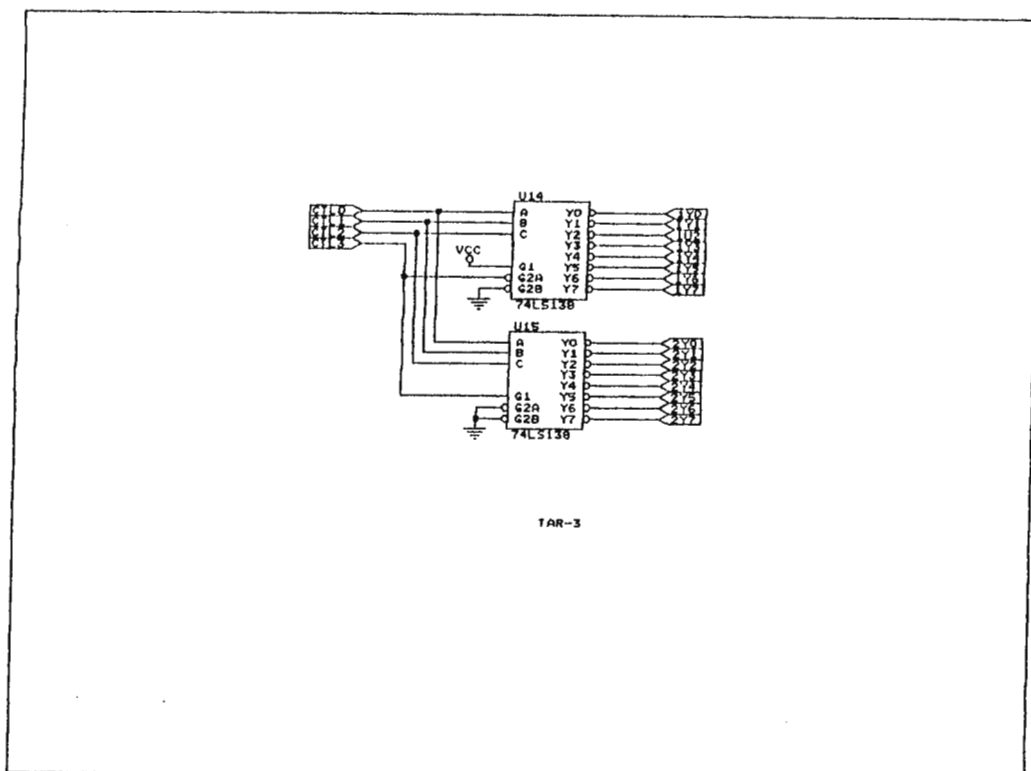


Figura 3.4 Diagrama del decodificador de palabra de control.

El decodificador de la palabra de control consta de dos demultiplexores (U14 y U15), cuyas líneas de dirección (A,B y C), están conectadas paralelamente teniendo U14 la línea de habilitación G2A, como la cuarta línea de dirección habilitada con cero, por el contrario U15 utilizará la línea G1 , la cual requiere de un 1 para habilitar el integrado.

P	G	L	G	L	G	L	G	L	G	L	G	L
I	1Y0	1Y1	1Y2	1Y3	1Y4	1Y5	1Y6	1Y7	2Y0	2Y1	2Y2	2Y3
N	'16	'16	'32	'32	'64	'64	128	128	256	256	512	512
28					6	5	6	5	6	5	6	5
27					HL	5	HL	5	A14	A14	A14	A14
26	5	5	6	5			A13	A13	A13	A13	A13	A13
23	21	5	A11	A11	A11	A11	A11	A11	A11	A11	A11	A11
22	5	L	13	L	5	L	5	L	5	L	13	L
20	LH	L	LH	L	L	L	L	L	HL	L	HL	L
1					13	5	13	5	13	5	A15	A15
CT	00	01	02	03	04	05	06	07	08	09	0A	0B
L												

Tabla 3.2 Señales en los pines conmutados para cada memoria y modo

Uniendo la líneas como se muestra en la figura 3.4, se obtendrá un demultiplexor de 1 a 16. Como el número de memorias con las que trabajaremos es de 6, el número de palabras de control será de 12, es decir que por cada modo de operación de una EPROM, tendremos un ordenamiento de pines diferente, ejemplo: si se desea leer la 2732 tendríamos que poner la señales indicadas en las tabla 3.2 que corresponden a cada pin lo que nos llevaría a utilizar una palabra de control en las líneas de dirección de U14 y U15 que nos genere un cero lógico en la línea 1Y4 a la cual los circuitos decodificadores de pines responderán habilitando las señales de control y aplicando los voltajes necesarios para efectuar la lectura de la 2732.

3.3.5 DECODIFICADORES DE PINES

Debido a que son 6 memorias las que se programarán, se debe disponer de un circuito que disponga las líneas de dirección, control y voltajes de alimentación de acuerdo a cada memoria.

De los 28 pines que tiene la base donde se colocará la EPROM a programar, 7 deben cambiar la señal que se aplica a la memoria en cada uno de estos 7 pines, los demás permanecen constantes para todas las memorias.

En la tabla 3.2 se presentan los pines en los que debe cambiarse la señal aplicada a la memoria, así como la señal que debe ser aplicada de acuerdo a la memoria seleccionada y al modo de operación de ésta.

La figura 3.5 y 3.6 muestran los siete circuitos decodificadores de pines, y la figura 3.7 muestra las conecciones de los pines de la base donde se colocará la EPROM que será programada.

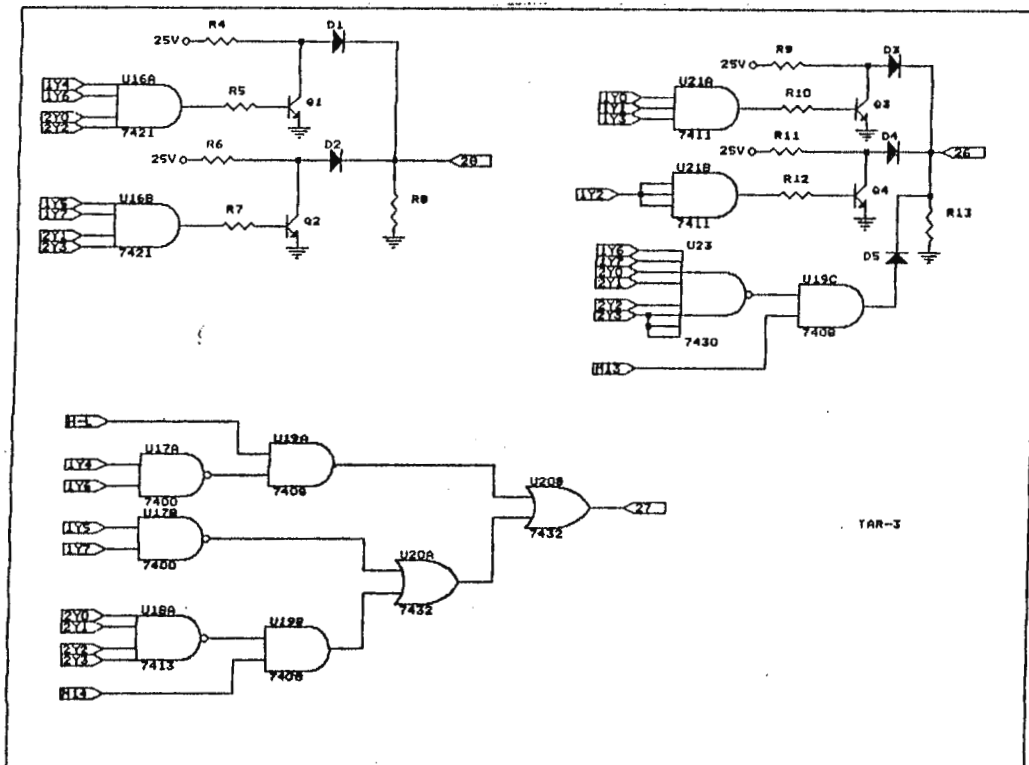


Figura 3.5 Decodificadores de pines 28, 27 y 26.

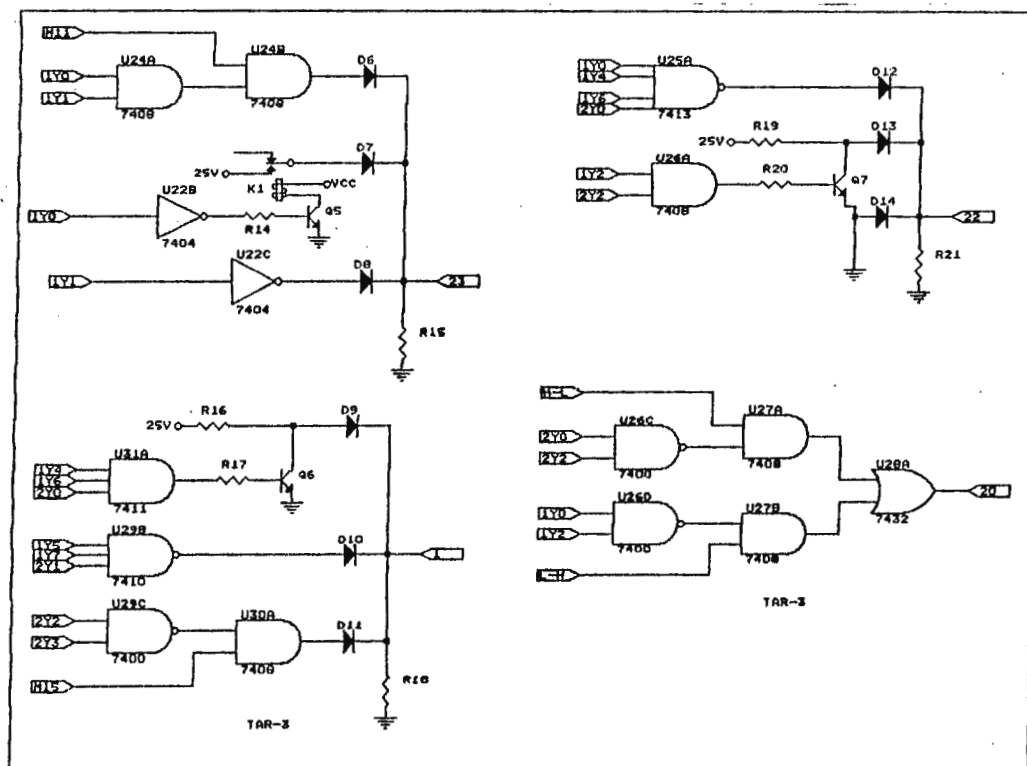


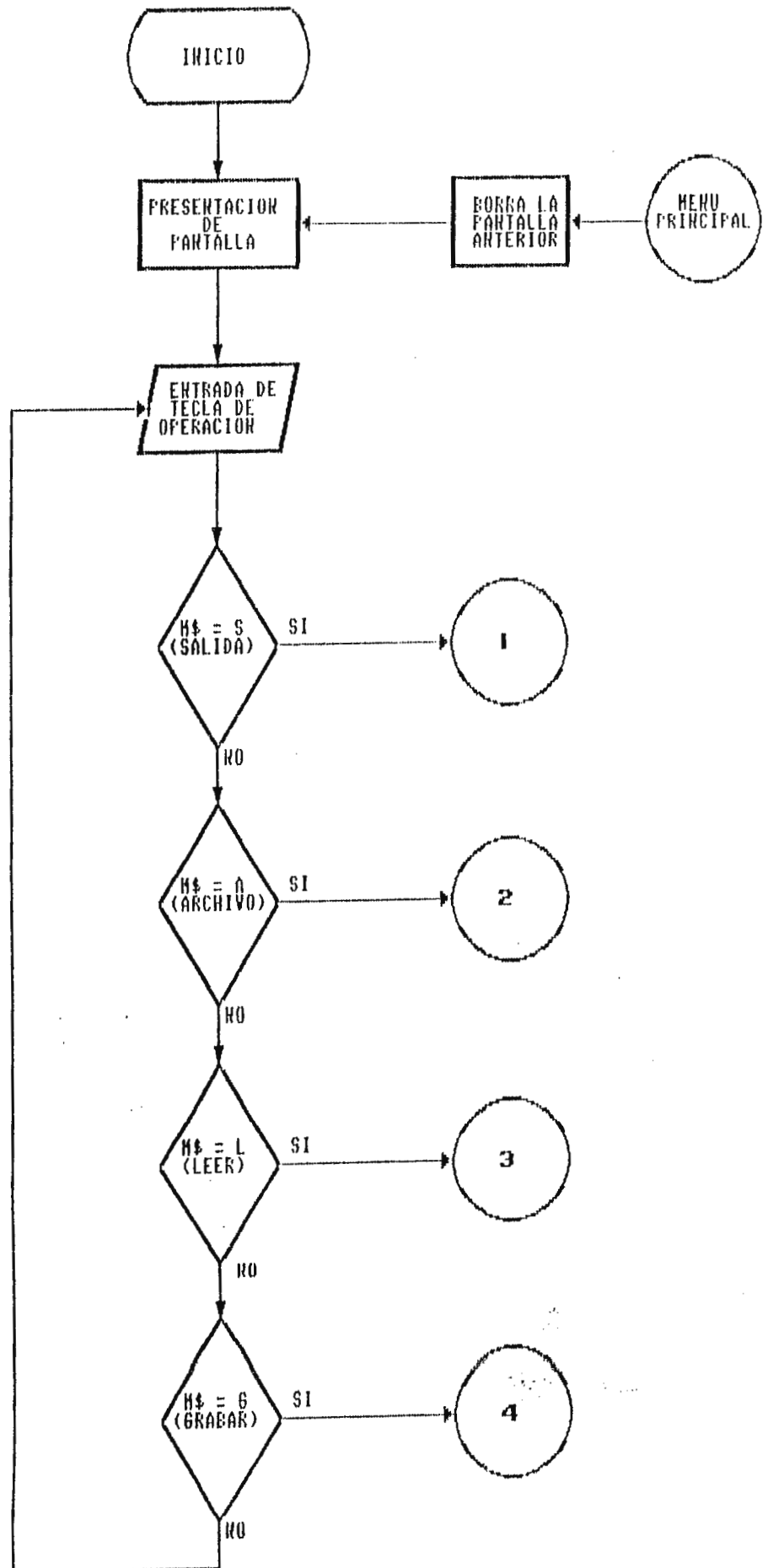
Figura 3.6 Decodificadores de pines 23, 22, 20 y 1.

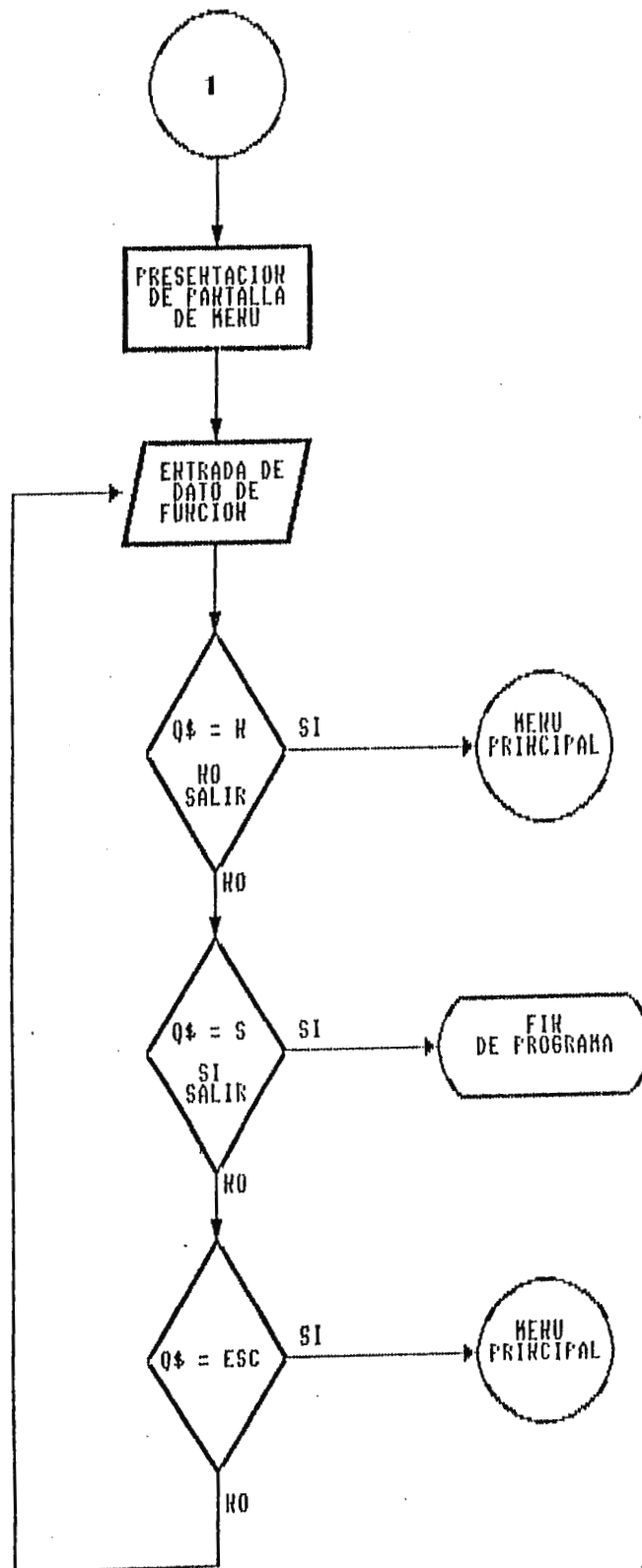
MENU PRINCIPAL

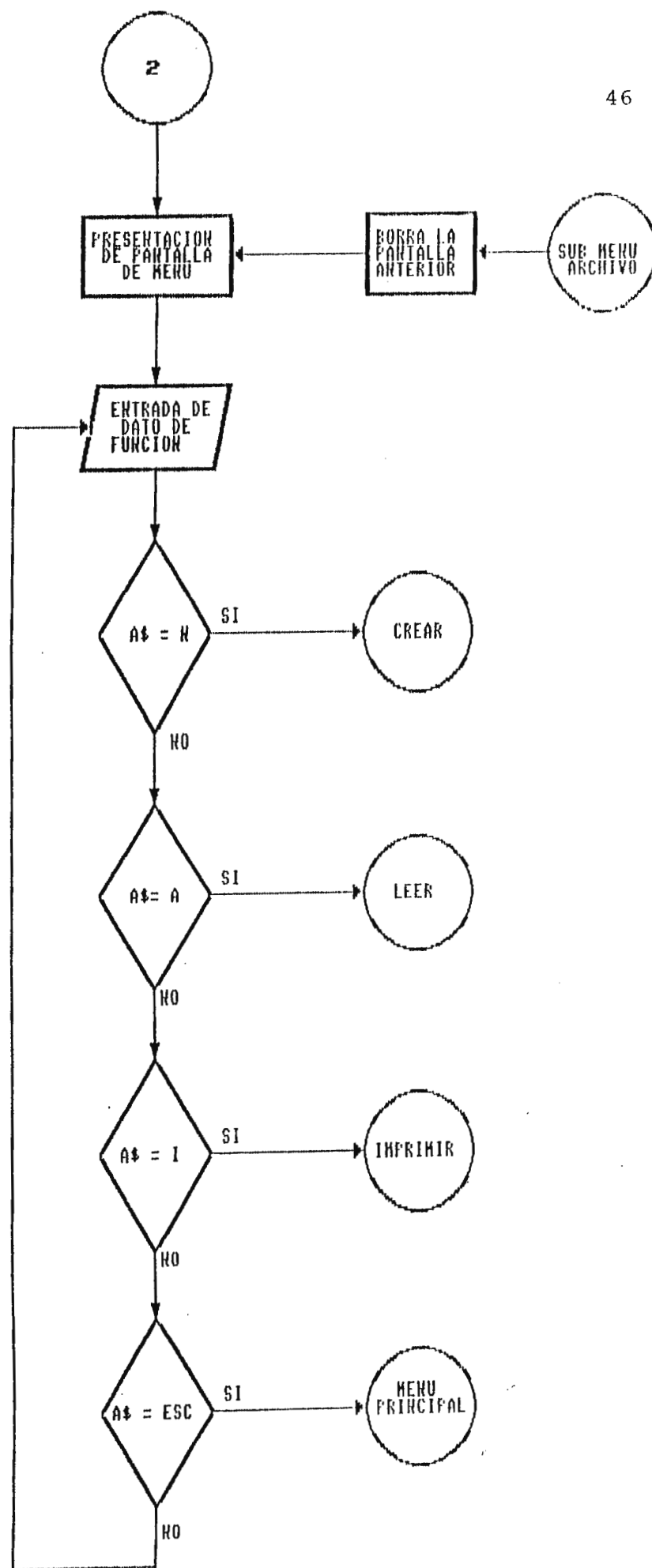
44

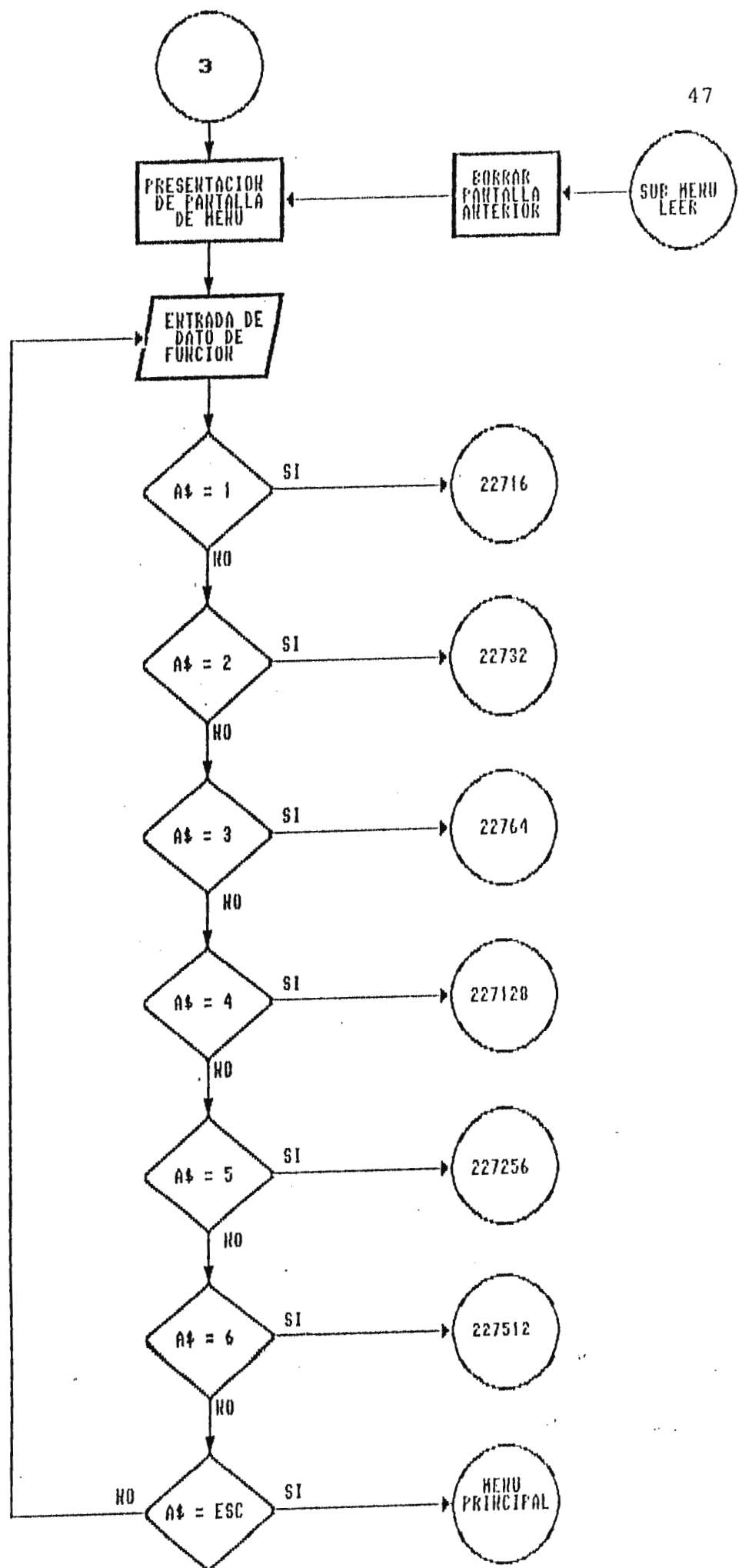
4.4.1 FLUJOGRAMA

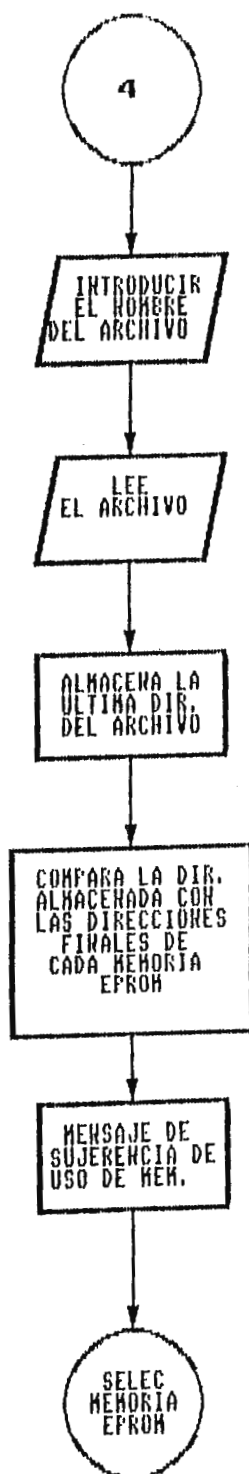
3.4.1 FLUJOGRAMA

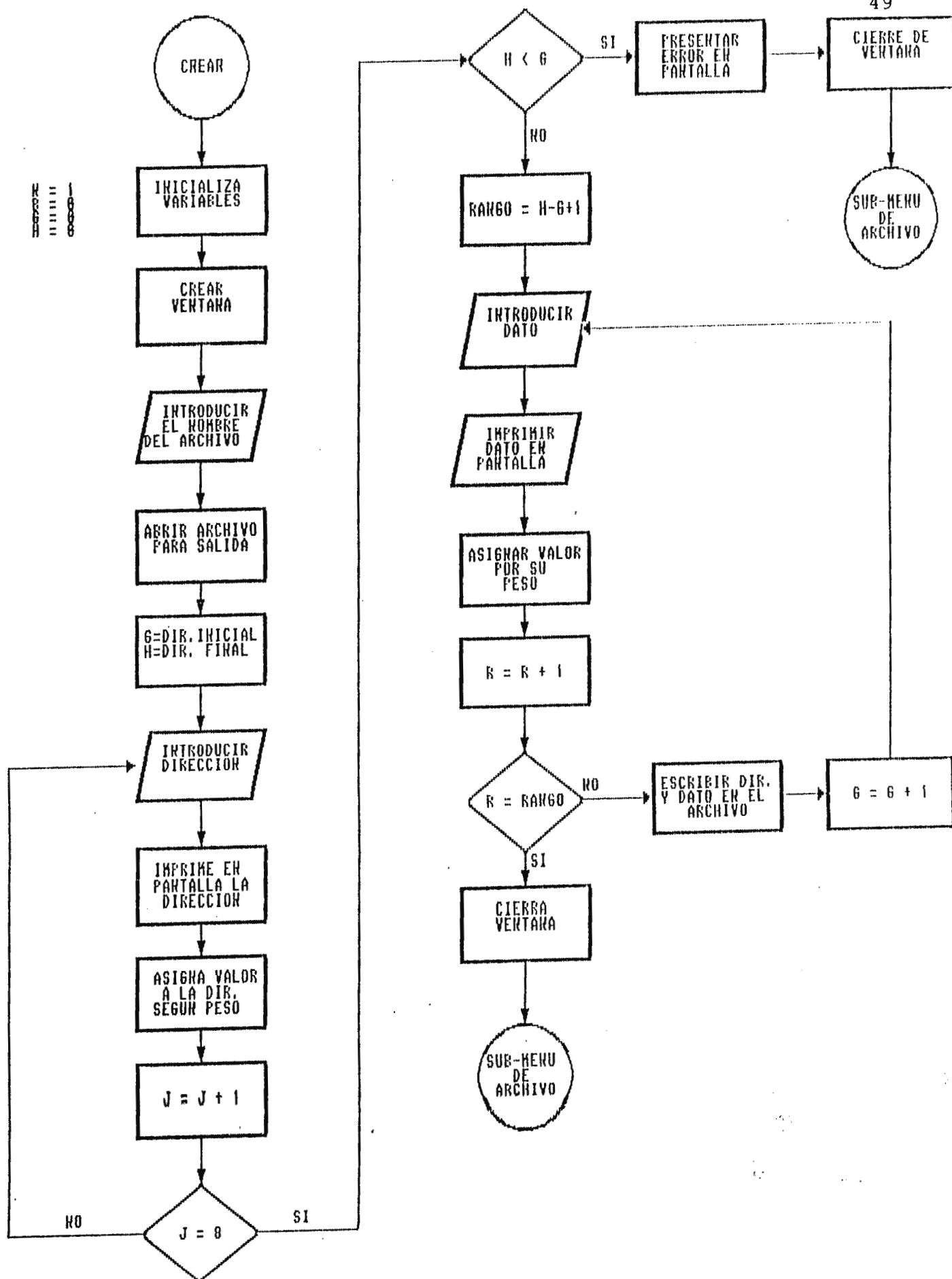


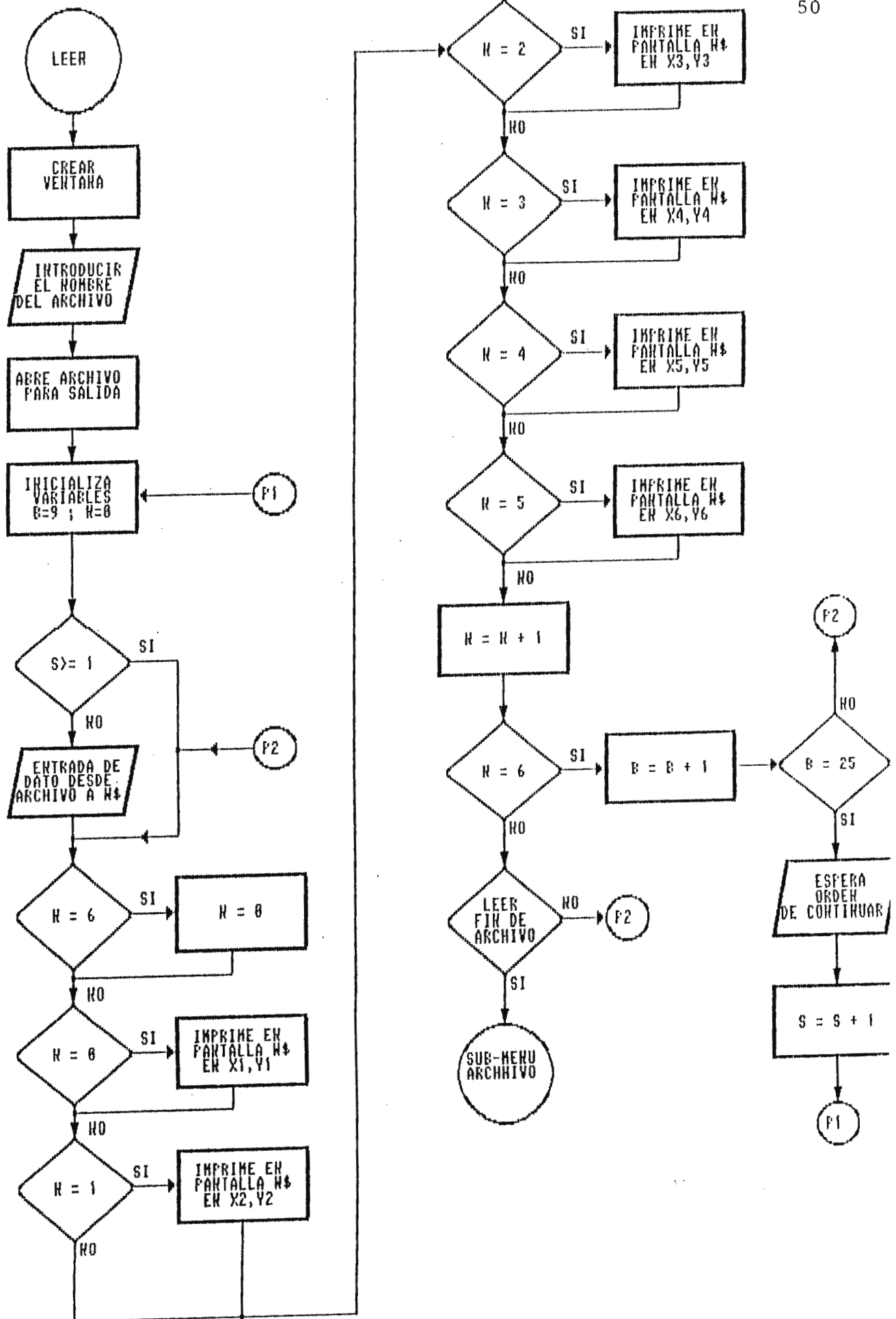


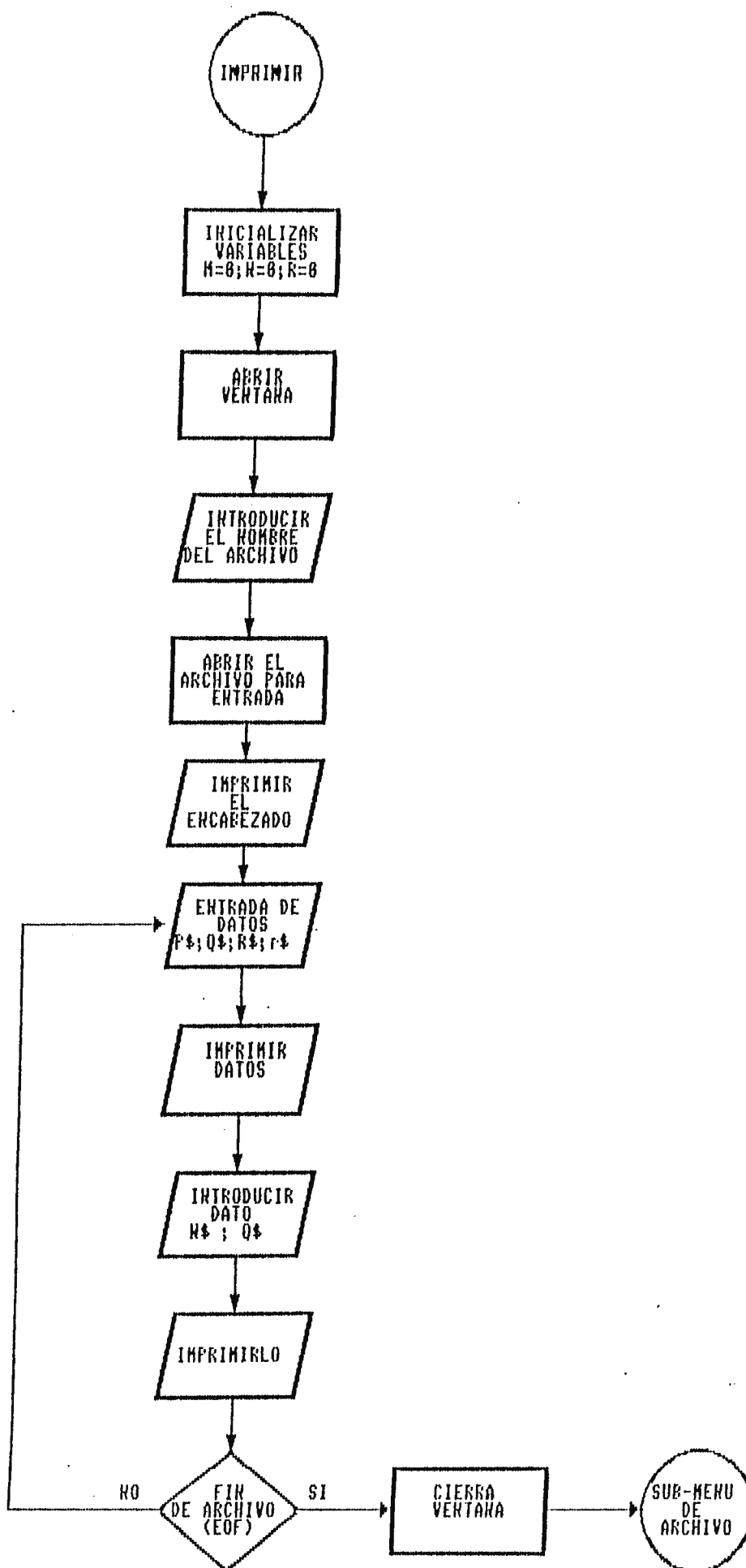


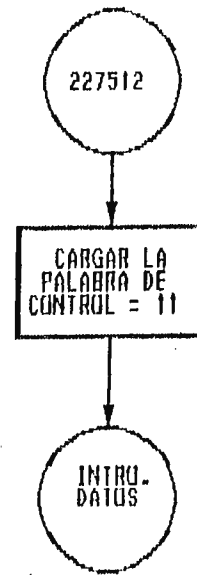
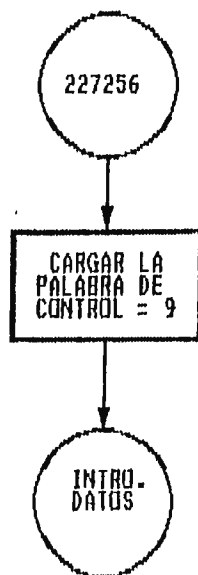
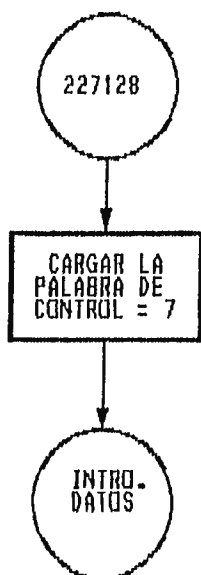
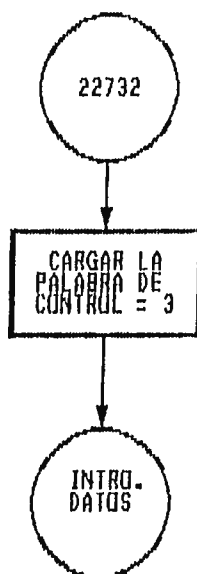
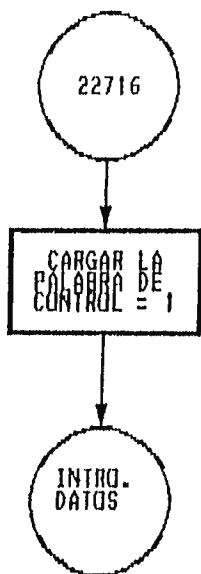


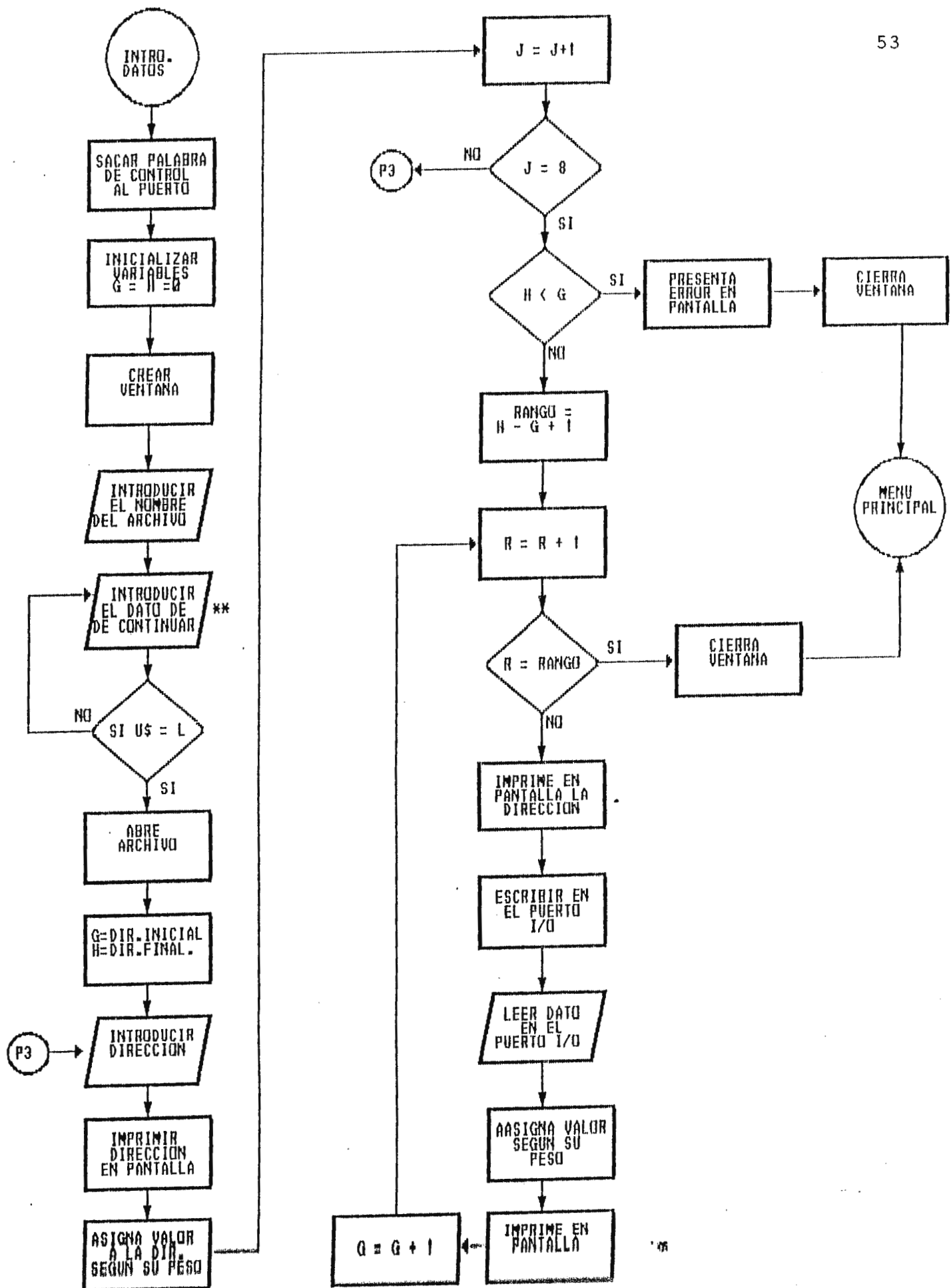


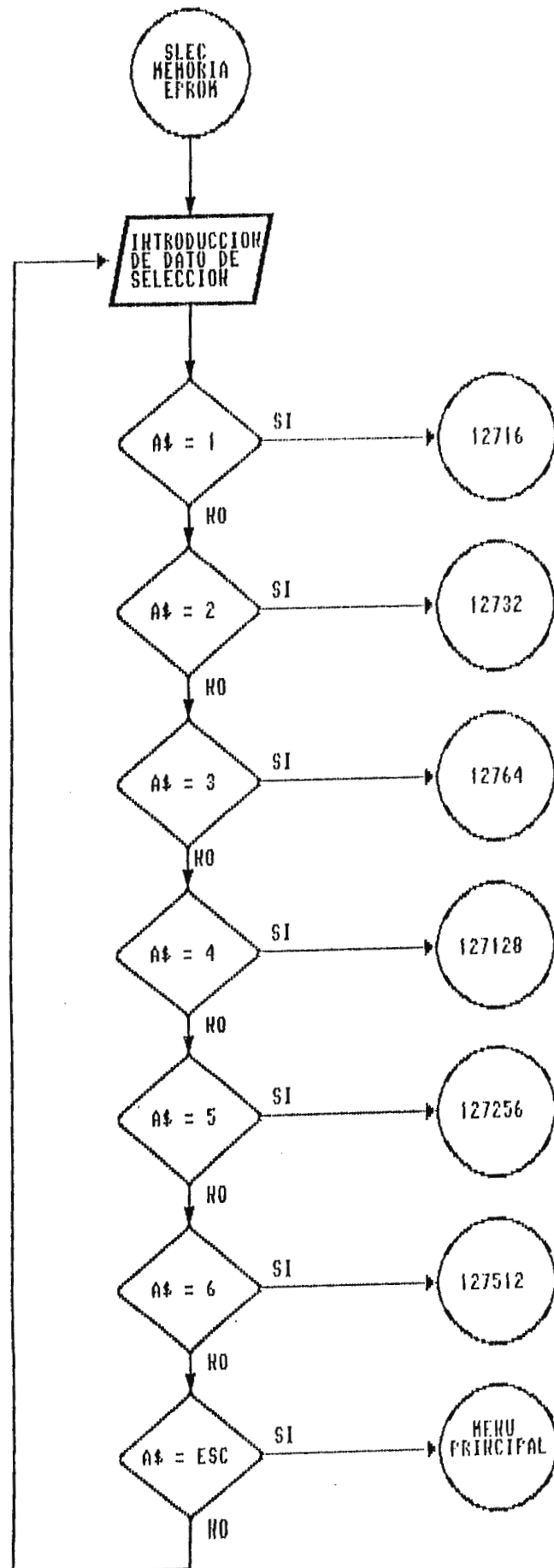


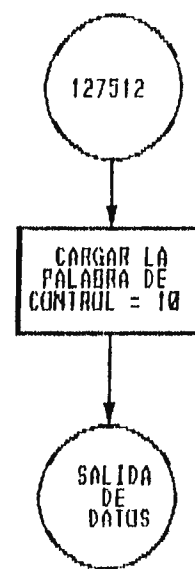
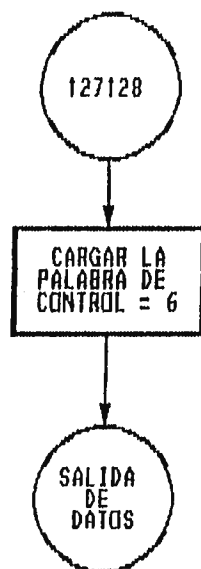
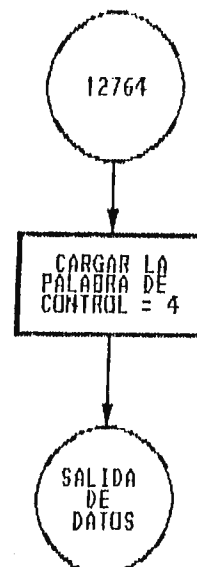
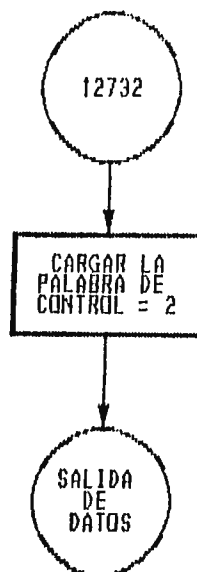
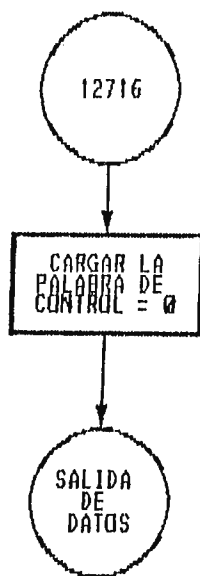


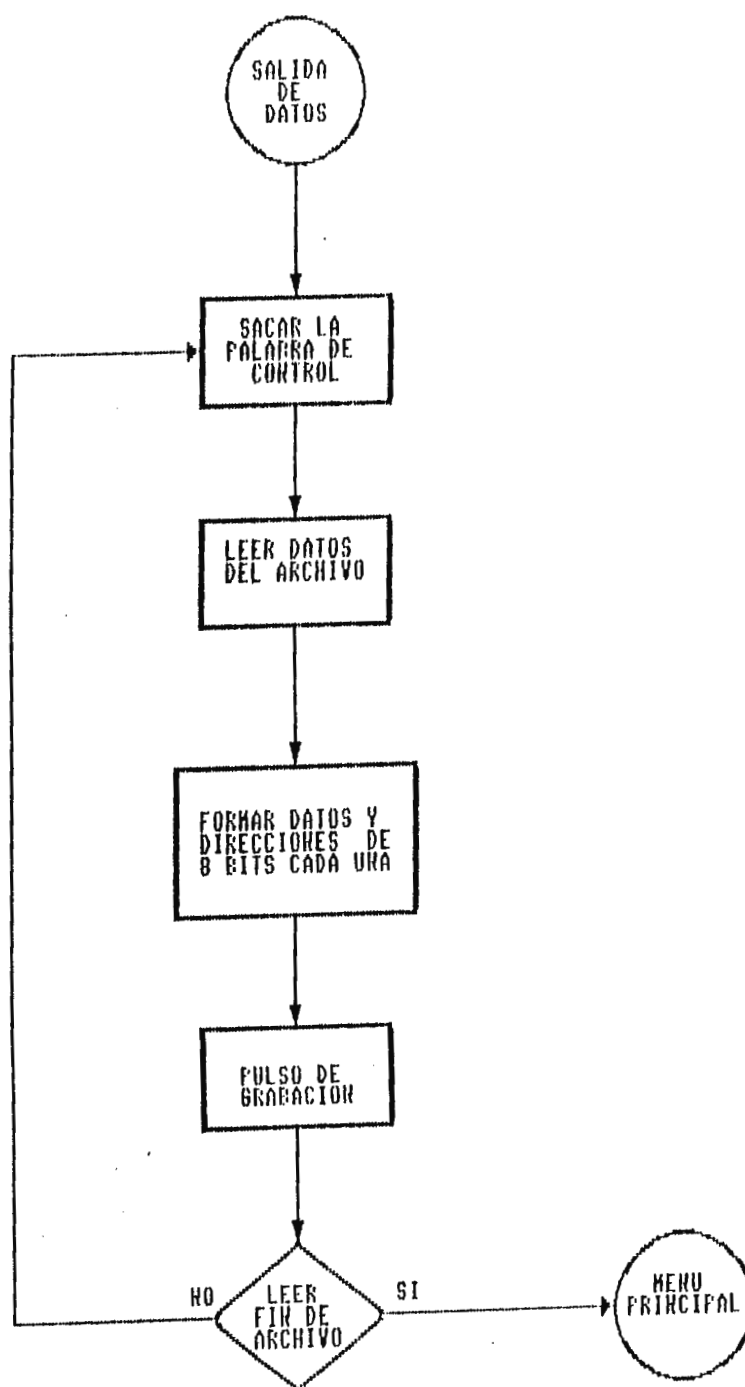












3.4.2 FUNCIONES GENERALES DEL PROGRAMADOR

Las funciones principales con las que se diseñó el Programador de Memorias EPROM por PC son:

1. Grabar memorias EPROM de la serie 27XXX, menos la 2710, 2708 y 271024, transfiriendo el contenido de un archivo a la EPROM.
2. Leer memorias EPROM de la serie 27XXX, menos la 2710, 2708 y 271024 y almacenar el contenido de ésta en un archivo.
3. Imprimir el contenido de un archivo que tiene almacenada la información que tenía la EPROM leída.

3.4.2.1 MENUS

Los menus permiten al usuario trasladarse de una función a otra, es decir realizar una función y al final de ésta ir a realizar otra o la misma. El acceso através de los menus se logra presionando la letra indicada de color diferente al color de el nombre de la función. Al finalizar la función seleccionada previamente se puede retornar al menu anterior presionando la tecla de escape [ESC]. Al seleccionar una opción del Menú Principal, se crean ventanas que contienen sub-menus que corresponden a cada una de las funciones principales del programador. Al finalizar la función y presionar [ESC], se abandonará el menú seleccionado y se regresará al menu anterior hasta volver al Menú Principal.

3.4.2.2 ARCHIVO.

La función de Archivo es la que se encarga de realizar la creación de archivos, la lectura de estos y la impresión de los mismos. Dentro de este sub-menú se puede determinar en que rango de EPROM se almacenará la información, la información que le programaremos, el contenido de una EPROM o la impresión de la información de la EPROM o del archivo.

3.4.2.2.1 CREAR ARCHIVO.

Esta opción permite abrir un archivo en una unidad de disco y almacenar en éste las direcciones y los datos que se programarán en la EPROM. La forma en que se almacena la información en el archivo es la siguiente:

-Se da nombre y ruta al archivo.

-Se indica la dirección inicial de la EPROM donde queremos que el programador empiece a grabar.

-Se indica la dirección final de la EPROM, es decir hasta donde queremos que sea grabada.

-Se inicia el ingreso de los datos que se quieren que sean grabados, y se finaliza cuando se llega hasta la dirección final.

ARCHIVO:C:\UDB\CREAR	
INICIO	FINAL
20AA	20AF
RANGO=6	
20AA	01
20AB	23
20AC	DE
20AD	A9
20AE	FF
20AF	0__

Figura 3.8 Ventana de creación de Archivo.

Cabe recalcar que una vez iniciado el proceso de creación de archivo no se puede salir de este hasta alcanzar la dirección final. Al finalizar de introducir todos los datos, se presiona [ESC] y se vuelve a la ventana del sub-menú de Archivo. La figura 3.8 muestra la ventana de creación de un archivo.

3.4.2.2.2 ABRIR ARCHIVO.

En este caso la función tiene como objeto ver el contenido de un archivo creado, ya sea por el usuario o por el programa al leer una memoria EPROM. Si el archivo fue creado por el usuario, esta

función permite verificar visualmente si las localidades que se utilizarán de la EPROM y los datos que estas contendrán son las correctas. Si el archivo fue creado por el programador de memorias al leer una EPROM, podemos ver las direcciones que están libres y que pueden ser utilizadas, así como también ver si la información grabada en la memoria es la correcta. En caso contrario se podrá hacer uso de cualquier procesador de texto, como EDIT de Microsoft o Word Perfect, para corregir la información del archivo. La disposición de la información del archivo aparece de la siguiente forma:

Linea 1	"1", "3", "9", "F", "E", "8"
Linea 2	"1", "3", "A", "0", "A", "F"
	LSB lsb
	MSB msb
	DIRECCION DATO

Los primero cuatro caracteres entre comillas representan la dirección (139F) de la EPROM, leida de izquierda a derecha y los ultimos dos caracteres representan el dato (E8) que se almacenará en esa dirección de memoria.

Al vizualizar un archivo, se presentará en la pantalla en páginas de 16 líneas, teniendose que presionar la tecla [C] para continuar viendo las demás líneas hasta ver todo el contenido del archivo.

Cuando se alcanza el final del archivo podemos optar por ver otro archivo presionando la tecla [O] , y repetir el mismo proceso.

La figura 3.9 muestra la ventana de lectura de un archivo.

ARCHIVO: C:\UDB\CREAR	
C para continuar	
DIRECCION	DATO
20AA	01
20AB	23
20AC	DE
20AD	A9
20AE	FF
20AF	00

Figura 3.9 Ventana de lectura de Archivo.

3.4.2.2.3. IMPRIMIR ARCHIVO

Esta opción nos permite enviar el contenido de un archivo a la impresora para tener en forma escrita el contenido de éste.

En el reporte impreso aparece el nombre del archivo, las direcciones de memoria y los datos, así como el total de

direcciones que tiene el archivo seleccionado. Al igual que en las otras funciones de Archivo, al querer imprimir un archivo, se indica la ruta del archivo y su nombre. La figura 3.10 muestra la ventana de impresión de archivo. La figura 3.11 muestra una impresión de archivo.

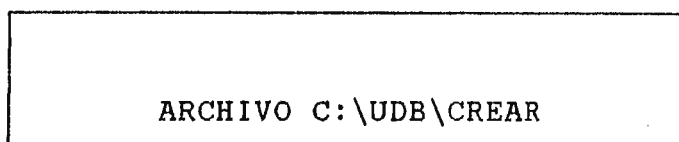


Figura 3.10 Ventana de impresión de archivo.

UNIVERSIDAD DON BOSCO	
NIVEL TECNOLÓGICO	
PROGRAMADOR DE MEMORIAS POR PC	

=====	
Archivo: CREAR	
=====	
DIRECCION	DATO
=====	
20AA	01
20AB	23
20AC	DE
20AD	A9
20AE	FF
20AF	00
+++++	
NUMERO DE LINEAS	6 /{RDR}

Figura 3.11 Impresión de un archivo

3.4.2.3 SALIR

Esta opción nos permite salir del programa y volver al sistema operativo, previa confirmación, ya que si la respuesta es negativa

se volverá al Menú Principal. La figura 3.12 muestra la ventana de confirmación de salida del programa.

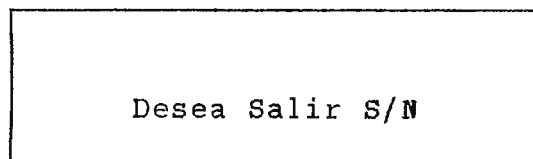


Figura 3.12 Ventana de confirmación de Salida del Programa

3.4.2.4 GRABAR MEMORIA EPROM.

Esta función del programador de memorias es la que permite seleccionar una de las 6 memorias de la serie 27XXX, para programarle el contenido de un archivo que nosotros indicamos.

Al elegir este sub-menú aparece una ventana en la que indicamos la ruta del archivo que se quiere transferir a la EPROM. Después de seleccionado el archivo, se almacena en el programa la última dirección del programa, y se compara con las direcciones últimas de cada una de las memorias EPROM, inhabilitando las memorias que no son aptas para ser grabadas debido a que el rango de dirección de la EPROM es inferior al indicado en el archivo que grabaremos. Después de este proceso aparece la ventana que contiene los números de las seis memorias EPROM (2716, 2732, 2764, 27128, 27256 y 27512), de las cuales se selecciona una al presionar un número correlativo a cada una. Sólo se podrán seleccionar las que indique el mensaje de sugerencia de EPROM.

Después de seleccionada la memoria, aparece una ventana con el mensaje de "COLOCAR EPROM EN EL PROGRAMADOR, PRESIONE C PARA CONTINUAR", al presionar [C] el programa comienza a leer el archivo y transferir la información a los flip flops del programador y a generar el pulso de grabación correspondiente, y repite el proceso hasta que alcanza la dirección final contenida en el archivo.

3.4.2.5. LEER MEMORIA EPROM

Esta función nos permite visualizar en la pantalla en una forma muy acelerada el contenido de una EPROM, seleccionada previamente, que está siendo transferida a un archivo. El archivo puede ser visto en una forma más detenida en la opción de leer archivo de la sección 3.4.2.2.2. Al entrar a esta función aparece la ventana en la cual se selecciona el número de la EPROM que queremos leer, presionando el número correlativo a la EPROM, después se indica la ruta y el nombre del archivo a crear en el cual se almacenará el contenido de la EPROM, después debe indicarse la dirección inicial y final del rango de direcciones que se leerán, y el programa verifica que ese rango no sea negativo y que la dirección final del mismo esté dentro del rango de direcciones de la EPROM seleccionada, caso contrario presentará los respectivos mensajes de error y deberá volver a iniciarse el proceso. Si los errores no se cometen las direcciones serán almacenadas en el archivo con los respectivos datos obtenidos de la EPROM.

3.5 ALCANCES Y LIMITACIONES

El programador de memorias EPROM por computadora ha sido diseñado considerando las memorias EPROM de la National Semiconductor, por lo que el programador puede trabajar con las memorias detalladas en las hojas técnicas del Apéndice D. Pero esta misma razón es la que no garantiza el hecho de que se logre programar otra memoria EPROM de otro fabricante con características eléctricas diferentes.

CONCLUSIONES

1. Con el programador de memorias EPROM por computadora se puede grabar información generada por el usuario u obtenida de otras memorias ya programadas y reproducirla en nuevas EPROM, las cuales pueden ser utilizadas en rótulos luminosos, juegos, aplicaciones industriales, etc.

2. Con la ayuda del programador de memorias EPROM por computadora se puede programar y leer las memorias EPROM 2716, 2732, 2764, 27128, 27256 y 27512. Con esto se puede reproducir una EPROM cuantas veces se desee, leyendo una EPROM, almacenar su contenido en un archivo y reproducir éste en otras EPROM.

3. Como se comprueba en este trabajo, las computadoras personales compatibles con IBM pueden ser de mucha utilidad para comandar por medio de un software un hardware periférico, reduciendo así la circuitería de éste último.

4. Comparando el programador de memorias presentado en este trabajo con el programador de EPROM de Lab-Volt, el cual puede programar y leer una EPROM 2716 o 2732, ingresando los datos desde un teclado o obteniendolos de otro dispositivo de memoria. También se puede comparar con el programador A.R.T., el cual opera através de un software y puede programar y verificar lo programado en las memorias 2716, 2732, 2764, 27128, 27256 y 27512. Al igual que este programador se puede almacenar en un archivo el contenido de cualquier EPROM.

APENDICE A.

PROGRAMA ETIQUETADO.

En esta sección se presenta el programa etiquetado, dividido en cuatro partes, SALIR, ARCHIVO, LEER y GRABAR.

-SECCION DE INICIO

```
CLS                                'INICIO
SCREEN 12
COLOR 15
LOCATE 8, 28: COLOR 14: PRINT " PROGRAMADOR DE MEMORIAS "
LOCATE 9, 18: PRINT " EPROM POR COMPUTADORA/ IBMPC Y COMPATIBLES"
LOCATE 10, 22: PRINT "RETIRE TODA MEMORIA DEL PROGRAMADOR"
LOCATE 11, 22: PRINT " PRESIONE C CUANDO ESTE LISTO"
SEGUIR1:
K$ = INKEY$
IF K$ = "C" OR K$ = "c" THEN GOTO SEGUIR
GOTO SEGUIR1
SEGUIR:
a$ = "MBT180o2P2P8L8GGGL2E-P24P8L8FFFL2D"
PLAY a$
WHILE PLAY(0) > 5: WEND
    COLOR 1
    FOR a = 0 TO 19
        PRINT CHR$(219);
    NEXT a
COLOR 15                                'PANTALLA PRINCIPAL
PRINT " PROGRAMADOR DE MEMORIAS EPROM POR PC'S "; 'ENCABEZADO
COLOR 1
FOR a = 0 TO 19
    PRINT CHR$(219);
NEXT a
FOR a = 0 TO 79
    PRINT CHR$(178);
NEXT a
COLOR 7                                'CREACION DEL FONDO
FOR a = 0 TO 2159
    PRINT CHR$(178);
NEXT a
```

-SECCION DE MENU PRINCIPAL

'MENU PRINCIPAL

'CREACION DE BARRA DE MENU

```

LOCATE 2, 13: COLOR 12: PRINT "S"; : COLOR 14: PRINT "alir";
LOCATE 2, 25: COLOR 12: PRINT "A"; : COLOR 14: PRINT "rchivo";
LOCATE 2, 45: COLOR 12: PRINT "L"; : COLOR 14: PRINT "eer";
LOCATE 2, 60: COLOR 12: PRINT "G"; : COLOR 14: PRINT "rabar";

```

'ROUTINA DE SELECCION DE MENU

```

menuppal:

```

```

m$ = INKEY$

```

```

    IF m$ = "S" OR m$ = "s" THEN GOTO SALIR
    IF m$ = "A" OR m$ = "a" THEN GOTO ARCHIVO
    IF m$ = "L" OR m$ = "l" THEN GOTO LEER
    IF m$ = "G" OR m$ = "g" THEN GOTO saldat

```

```

GOTO menuppal

```

-SUB-MENU DE SALIDA

'SUB-MENU DE SALIDA

'CREACION DE VENTANA

```

SALIR:

```

```

LOCATE 14, 30

```

```

FOR i = 0 TO 16

```

```

    PRINT CHR$(0);

```

```

NEXT i

```

```

LOCATE 15, 30

```

```

COLOR 14

```

'MENSAJE DE SALIDA

```

PRINT " Desea Salir S/N "; : COLOR 8: PRINT CHR$(178);

```

```

LOCATE 16, 30

```

```

FOR i = 0 TO 16

```

```

    PRINT CHR$(0);

```

```

NEXT i

```

```

COLOR 8: PRINT CHR$(178); 'ELIMINACION DE VENTANA

```

```

LOCATE 17, 31

```

```

COLOR 8

```

```

FOR i = 0 TO 16

```

'SOMBRA INFERIOR DE LA VENTANA

```

    PRINT CHR$(178);

```

```

NEXT i

```

'CONFIRMACION DE SALIDA

SAL:

q\$ = INKEY\$

IF q\$ = "S" OR q\$ = "s" THEN

CLS

SYSTEM

END IF

IF q\$ = "N" OR q\$ = "n" THEN

COLOR 7: K = 0

FOR i = 0 TO 3

LOCATE K + 14, 30

FOR J = 0 TO 21

PRINT CHR\$(178);

NEXT J

K = K + 1

NEXT i

GOTO menuppall

'RETORNO A MENU PRINCIPAL

END IF

GOTO SAL

-SUB-MENU DE ARCHIVO

'SUB-MENU DE FUNCIONES DE ARCHIVO

'CREACION DE VENTANA

ARCHIVO:

LOCATE 4, 25: COLOR 12: PRINT " N";:COLOR 15: PRINT "uevo ";

LOCATE 5, 25: COLOR 12: PRINT " A";:COLOR 15: PRINT "brir ";

COLOR 8: PRINT CHR\$(178);

LOCATE 6, 25: COLOR 12: PRINT " I";:COLOR 15: PRINT "mprimir ";

COLOR 8: PRINT CHR\$(178);

LOCATE 7, 26: COLOR 8

FOR i = 0 TO 12

PRINT CHR\$(178);

NEXT i

'SUB-MENU DE ARCHIVO

ARCHIVO1:

a\$ = INKEY\$

IF a\$ = "n" OR a\$ = "N" THEN GOTO CREAR 'SALTO A CREAR ARCHIVO

IF a\$ = "a" OR a\$ = "A" THEN GOTO LECTURA 'SALTO A LEER ARCHIVO

IF a\$ = "I" OR a\$ = "i" THEN GOTO IMPRIMIR 'SALTO A IMPRIMIR ARCH

IF a\$ = CHR\$(27) THEN 'SALIR

K = 0

FOR i = 0 TO 4 'ELIMINACION DE VENTANA

LOCATE K + 4, 25

FOR J = 0 TO 13

COLOR 7: PRINT CHR\$(178);

NEXT J

K = K + 1

NEXT i

GOTO menuppal 'RETORNO A MENU PRINCIPAL

END IF

GOTO ARCHIVO1 'RETORNO A SUB-MENU DE ARCHIVO

-SECCION PARA CREAR ARCHIVO

'CREAR ARCHIVO

CREAR:

LOCATE 6, 23 'CREACION DE LA VENTANA DE ESCRITURA

FOR i = 0 TO 35

PRINT CHR\$(0);

NEXT i

K = 0

FOR i = 0 TO 19

LOCATE K + 7, 23

FOR J = 0 TO 35

PRINT CHR\$(0);

NEXT J

COLOR 8: PRINT CHR\$(178);

K = K + 1

NEXT i

LOCATE 27, 24

```

FOR i = 0 TO 35
    COLOR 8: PRINT CHR$(178);
NEXT i
J = 0
COLOR 14
LOCATE 6, 25
INPUT "ARCHIVO"; ARCH$ 'ENTRADA DEL NOMBRE DEL ARCHIVO A CREAR
OPEN ARCH$ FOR OUTPUT AS #1 'APERTURA DE ARCHIVO COMO SALIDA
COLOR 15
LOCATE 7, 25: PRINT "INICIO      FINAL";
ini:
G = I0 + I1 + I2 + I3          'DIRECCION INICIAL
H = F0 + F1 + F2 + F3          'DIRECCION FINAL
LOCATE 9, 27: PRINT HEX$(G)
LOCATE 9, 37: PRINT HEX$(H)
IF J = 8 THEN
    IF H < G THEN
LOCATE 7, 25:PRINT "!!!ERROR:RANGO NEGATIVO!!!" 'MENSAJE DE ERROR
QUIT:
        K$ = INKEY$
        IF K$ = CHR$(27) THEN
            CLOSE #1
            I0 = 0
            I1 = 0
            I2 = 0
            I3 = 0
            F0 = 0
            F1 = 0
            F2 = 0
            F3 = 0 'HACER CERO LAS DIRECCIONES ERRONEAS
            KILL ARCH$ 'ELIMINACION DE ARCHIVO CON ERROR
            GOTO SAL1
        END IF

```

```

        GOTO QUIT
    END IF
    GOTO FINAL
END IF

        'BUCLE PARA ENTRADA DE DIRECCIONES

INI1:
K$ = INKEY$
    IF K$ = "0" THEN GOTO 0
    IF K$ = "1" THEN GOTO 1
    IF K$ = "2" THEN GOTO 2
    IF K$ = "3" THEN GOTO 3
    IF K$ = "4" THEN GOTO 4
    IF K$ = "5" THEN GOTO 5
    IF K$ = "6" THEN GOTO 6
    IF K$ = "7" THEN GOTO 7
    IF K$ = "8" THEN GOTO 8
    IF K$ = "9" THEN GOTO 9
    IF K$ = "A" OR K$ = "a" THEN GOTO a
    IF K$ = "B" OR K$ = "b" THEN GOTO b
    IF K$ = "C" OR K$ = "c" THEN GOTO c
    IF K$ = "D" OR K$ = "d" THEN GOTO d
    IF K$ = "E" OR K$ = "e" THEN GOTO e
    IF K$ = "F" OR K$ = "f" THEN GOTO F
GOTO INI1

        'RETORNO DEL BUCLE
        'ASIGNACION DE VALORES NUMERICOS

0 :
x = 0
GOTO BACK
1 :
x = 1
GOTO BACK
2 :
x = 2
GOTO BACK

```

```
3 :  
x = 3  
GOTO BACK  
4 :  
x = 4  
GOTO BACK  
5 :  
x = 5  
GOTO BACK  
6 :  
x = 6  
GOTO BACK  
7 :  
x = 7  
GOTO BACK  
8 :  
x = 8  
GOTO BACK  
9 :  
x = 9  
GOTO BACK  
a:  
x = 10  
GOTO BACK  
b:  
x = 11  
GOTO BACK  
c:  
x = 12  
GOTO BACK  
d:  
x = 13  
GOTO BACK
```

e:

x = 14

GOTO BACK

F:

x = 15

GOTO BACK

'ROUTINA PARA FORMAR LAS DIRECCIONES

BACK:

'FORMACION DE DIRECCION INICIAL

IF J = 0 THEN

J = J + 1: I3 = x * 4096: GOTO ini 'SALTO A INI PARA FORMAR

END IF

'DIRECCIONES

IF J = 1 THEN

J = J + 1: I2 = x * 256: GOTO ini

END IF

IF J = 2 THEN

J = J + 1: I1 = x * 16: GOTO ini

END IF

IF J = 3 THEN

J = J + 1: I0 = x: GOTO ini

END IF

'FORMACION DE DIRECCION FINAL

IF J = 4 THEN

J = J + 1: F3 = x * 4096: GOTO ini 'SALTO A INI PARA

END IF

'FORMAR DIRECCIONES

IF J = 5 THEN

J = J + 1: F2 = x * 256: GOTO ini

END IF

IF J = 6 THEN

J = J + 1: F1 = x * 16: GOTO ini

END IF

IF J = 7 THEN

J = J + 1: F0 = x: GOTO ini

END IF

'FORMACION DEL RANGO

```

FINAL:
RANGO = H - G + 1
LOCATE 8, 29: PRINT "RANGO="; RANGO
o = 1
FOR r = 1 TO RANGO
  IF o = 17 THEN
    o = 1
  FOR S = 0 TO 20
    LOCATE (S + 6), 23
    FOR i = 0 TO 35
      PRINT CHR$(0);
    NEXT i
  NEXT S
END IF
o = o + 1
LOCATE (o + 9), 35: PRINT ;
DIR$ = HEX$(G)
PRINT DIR$; : LOCATE o + 9, 30
U = 0

```

```

ini3:                                'BUCLE PARA FORMAR LOS DATOS

```

```

q$ = INKEY$
  IF q$ = "0" THEN GOTO B0
  IF q$ = "1" THEN GOTO B1
  IF q$ = "2" THEN GOTO B2
  IF q$ = "3" THEN GOTO B3
  IF q$ = "4" THEN GOTO B4
  IF q$ = "5" THEN GOTO B5
  IF q$ = "6" THEN GOTO B6
  IF q$ = "7" THEN GOTO B7
  IF q$ = "8" THEN GOTO B8
  IF q$ = "9" THEN GOTO B9
  IF q$ = "A" OR q$ = "a" THEN GOTO BA
  IF q$ = "B" OR q$ = "b" THEN GOTO BB
  IF q$ = "C" OR q$ = "c" THEN GOTO BC

```

```
        IF q$ = "D" OR q$ = "d" THEN GOTO BD
        IF q$ = "E" OR q$ = "e" THEN GOTO BE
        IF q$ = "F" OR q$ = "f" THEN GOTO Bf
GOTO ini3          'RETORNO DEL BUCLE
                   'ASIGNACION DE VALORES

B0:
    x = 0
    GOTO BACK1
B1:
    x = 1
    GOTO BACK1
B2:
    x = 2
    GOTO BACK1
B3:
    x = 3
    GOTO BACK1
B4:
    x = 4
    GOTO BACK1
B5:
    x = 5
    GOTO BACK1
B6:
    x = 6
    GOTO BACK1
B7:
    x = 7
    GOTO BACK1
B8:
    x = 8
    GOTO BACK1
B9:
    x = 9
    GOTO BACK1
```

```

BA:
    x = 10
    GOTO BACK1
BB:
    x = 11
    GOTO BACK1
BC:
    x = 12
    GOTO BACK1
BD:
    x = 13
    GOTO BACK1
BE:
    x = 14
    GOTO BACK1
Bf:
    x = 15
    GOTO BACK1
BACK1:
    IF U = 0 THEN
        U = U + 1: D1 = x * 16: LOCATE o + 9, 45: PRINT HEX$(D1);
        GOTO ini3
    END IF
    IF U = 1 THEN
        U = U + 1: D0 = x: LOCATE o + 9, 46: PRINT HEX$(D0);
    END IF

        ' ESCRITURA DE DIRECCIONES Y DATOS EN ARCHIVO
    DAT = D1 + D0
    DATA$ = HEX$(DAT)
    IF G = 0 THEN
        M0$ = "0"
        M1$ = "0"

```

```
M2$ = "0"
M3$ = "0"
GOTO PARP
END IF
IF G <= 15 THEN
    M1$ = "0"
    M2$ = "0"
    M3$ = "0"
    M0$ = MID$(DIR$, 1, 1)
    GOTO PARP
END IF
IF G <= 255 THEN
    M2$ = "0"
    M3$ = "0"
    M1$ = MID$(DIR$, 1, 1)
    M0$ = MID$(DIR$, 2, 1)
    GOTO PARP
END IF
IF G <= 4095 THEN
    M3$ = "0"
    M2$ = MID$(DIR$, 1, 1)
    M1$ = MID$(DIR$, 2, 1)
    M0$ = MID$(DIR$, 3, 1)
    GOTO PARP
END IF
IF G <= 65535 THEN
    M3$ = MID$(DIR$, 1, 1)
    M2$ = MID$(DIR$, 2, 1)
    M1$ = MID$(DIR$, 3, 1)
    M0$ = MID$(DIR$, 4, 1)
END IF
PARP:
IF DAT = 0 THEN
    D0$ = "0"
    D1$ = "0"
```

```
GOTO PARP1
END IF
IF DAT <= 15 THEN
    D1$ = "0"
    D0$ = MID$(DATA$, 1, 1)
    GOTO PARP1
END IF
IF DAT <= 255 THEN
    D1$ = MID$(DATA$, 1, 1)
    D0$ = MID$(DATA$, 2, 1)
END IF
PARP1:
    WRITE #1, M3$, M2$, M1$, M0$, D1$, D0$
    G = G + 1
NEXT r
SAL1:
CLOSE #1
T$ = INKEY$
IF T$ = CHR$(27) THEN
    LOCATE 6, 20: COLOR 7
    FOR Y = 0 TO 40
        FOR x = 0 TO 21
            LOCATE (x + 6), (Y + 20): PRINT CHR$(178);
        NEXT x
    NEXT Y
    G = H = 0
    CLOSE #1: GOTO ARCHIVO
END IF
GOTO SAL1
```

-LECTURA DE ARCHIVO CREADO

'RUTINA DE LECTURA DE ARCHIVO CREADO

LECTURA:

S = 0

LECTURAL:

LOCATE 6, 20 'CREACION DE VENTANA

FOR i = 0 TO 35

PRINT CHR\$(0);

NEXT i

FOR p = 0 TO 19

LOCATE (p + 7), 20

FOR i = 0 TO 35

PRINT CHR\$(0);

NEXT i

COLOR 8: PRINT CHR\$(178);

IF p = 19 THEN

LOCATE 27, 21: COLOR 8

FOR i = 0 TO 35

PRINT CHR\$(178);

NEXT i

END IF

NEXT p

IF S >= 1 THEN GOTO PASO

LOCATE 6, 22: COLOR 14

INPUT "ARCHIVO"; F\$ 'NOMBRE Y RUTA DEL ARCHIVO A LEER

OPEN F\$ FOR INPUT AS #2

PASO:

COLOR 14

LOCATE 8, 25: PRINT "C para continuar" 'SI EL ARCHIVO CONTIENE
'MAS DE 16 LINEAS

LOCATE 7, 25: PRINT " DIRECCION DATO";

b = 9

n = 0

```

DO WHILE NOT EOF(2)
    INPUT #2, W$                                'ENTRADA DE PALABRAS
    IF n = 6 THEN n = 0
    IF n = 0 THEN LOCATE b, 30
    IF n = 1 THEN LOCATE b, 31
    IF n = 2 THEN LOCATE b, 32
    IF n = 3 THEN LOCATE b, 33
    IF n = 4 THEN LOCATE b, 40
    IF n = 5 THEN LOCATE b, 41
    COLOR 15
    PRINT W$;
L:
    n = n + 1
    IF n = 6 THEN
        b = b + 1
        IF b = 25 THEN GOTO CONT
    END IF
LAZO:
LOOP
IF EOF(2) THEN
CONT:                                'ROUTINA PARA CONTINUAR VISUALIZANDO
    a$ = INKEY$
    IF a$ = "C" OR a$ = "c" THEN
        S = S + 1
        GOTO LECTURAL1
        GOTO LAZO
    END IF
    IF a$ = CHR$(27) THEN GOTO SALIDA
END IF
IF EOF(2) THEN
SALIDA:                                'ROUTINA PARA LEER OTRO DATO
    b$ = INKEY$
    IF b$ = CHR$(27) THEN
        CLOSE #2: S = 0: GOTO SAL2
    END IF

```

```

IF b$ = "o" OR b$ = "O" THEN
    S = 0
    CLOSE #2: GOTO LECTURA 'REGRESO A RUTINA DE LECTURA
END IF
GOTO SALIDA
END IF
GOTO CONT

SAL2:                                'ELIMINACION DE VENTANA
LOCATE 6, 20: COLOR 7:
FOR Y = 0 TO 40
    FOR x = 0 TO 21
        LOCATE (x + 6), (Y + 20): PRINT CHR$(178);
    NEXT x
NEXT Y
GOTO ARCHIVO                        'RETORNO A SUB-MENU DE ARCHIVO

-SECCION PARA IMPRIMIR UN ARCHIVO CREADO
                                'RUTINA PARA IMPRIMIR
IMPRIMIR:
LOCATE 6, 20                        'CREACION DE VENTANA
FOR i = 0 TO 31
    PRINT CHR$(0);
NEXT i
K = 0
FOR i = 0 TO 1
    LOCATE K + 7, 20
    FOR J = 0 TO 31
        PRINT CHR$(0);
    NEXT J
    COLOR 8: PRINT CHR$(178);
    K = K + 1
NEXT i
LOCATE 8, 20

```



```

FOR i = 0 TO 31
    PRINT CHR$(0);
NEXT i
COLOR 8: PRINT CHR$(178);
LOCATE 9, 21: COLOR 8
FOR i = 0 TO 31
    PRINT CHR$(178);
NEXT i
LOCATE 7, 22: COLOR 14
INPUT "ARCHIVO"; b$ 'ENTRAR NOMBRE DEL ARCHIVO A IMPRIMIR
OPEN b$ FOR INPUT AS #1
m = 0
n = 0
r = 0

                                'PRESENTACION DE LA IMPRESION
LPRINT "      UNIVERSIDAD DON BOSCO      "
LPRINT "      NIVEL TECNOLOGICO          "
LPRINT "  PROGRAMADOR DE MEMORIAS POR PC  "
LPRINT "*****"
LPRINT "===== "
LPRINT "Archivo: "; b$
LPRINT "===== "
LPRINT "      DIRECCION      DATO"
LPRINT "===== "
DO WHILE NOT EOF(1) 'ENTRADA DE DIRECCION Y DATOS A IMPRIMIR
    INPUT #1, M3I$, M2I$, M1I$, MOI$, D1I$, DOI$
    m = m + 1
    LPRINT "      "; M3I$; M2I$; M1I$; MOI$; "      "; D1I$; DOI$
LOOP
LPRINT "+++++"
LPRINT "NUMERO DE LINEAS "; m; "{RDR}"
CLOSE #1

```

```

FOR Y = 0 TO 3                                'ELIMINACION DE VENTANA
  FOR x = 0 TO 32
    LOCATE (Y + 6), (x + 20)
    COLOR 7
    PRINT CHR$(178);
  NEXT x
NEXT Y
U = 0
GOTO ARCHIVO                                'RETORNO A SUB-MENU DE ARCHIVO

```

-SECCION DE SUB-MENU DE LECTURA DE EPROM

```

'SUB-MENU DE LECTURA DE EPROM
'CREACION DE VENTANA DE SUB-MENU

```

LEER:

```

LOCATE 4, 45: COLOR 15: PRINT "  LEER  EPROM # "
LOCATE 5, 45: COLOR 15: PRINT "      2716  "; : COLOR 12: PRINT "►
1 ";
COLOR 8: PRINT CHR$(178);
LOCATE 6, 45: COLOR 15: PRINT "2732"; : COLOR 12: PRINT "►  2 ";
COLOR 8: PRINT CHR$(178);
LOCATE 7, 45: COLOR 15: PRINT "2764"; : COLOR 12: PRINT "►  3 ";
COLOR 8: PRINT CHR$(178);
LOCATE 8, 45: COLOR 15: PRINT "27128"; :COLOR 12: PRINT "►  4 ";
COLOR 8: PRINT CHR$(178);
LOCATE 9, 45: COLOR 15: PRINT "27256"; :COLOR 12: PRINT "►  5 ";
COLOR 8: PRINT CHR$(178);
LOCATE 10, 45: COLOR 15: PRINT "27512";:COLOR 12: PRINT "►  6 ";
COLOR 8: PRINT CHR$(178);
LOCATE 11, 46: COLOR 8
FOR i = 0 TO 16                                'SOMBRA INFERIOR DE LA VENTANA
  PRINT CHR$(178);
NEXT i

```

'ROUTINA DE SELECCION DE EPROM A LEER

FUNCION2:

a\$ = INKEY\$

IF a\$ = "1" THEN GOTO 22716 'SALTOS A RUTINAS DE LECTURA

IF a\$ = "2" THEN GOTO 22732 'PARA CADA EPROM

IF a\$ = "3" THEN GOTO 22764

IF a\$ = "4" THEN GOTO 227128

IF a\$ = "5" THEN GOTO 227256

IF a\$ = "6" THEN GOTO 227512

IF a\$ = CHR\$(27) THEN 'SALIDA DE SUB-MENU DE LECTURA

K = 0

FOR i = 0 TO 15 'ELIMINACION DE VENTANA DE SUB-MENU

LOCATE K + 4, 45

FOR J = 0 TO 17

COLOR 7: PRINT CHR\$(178);

NEXT J

K = K + 1

NEXT i

GOTO menuppall 'RETORNO A MENU PRINCIPAL

END IF

GOTO FUNCION2 'RETORNO A RUTINA DE SUB-MENU DE SELECCION

'ROUTINA DE CREACION DE ARCHIVO

'ROUTINA PARA SELECCIONAR PALABRA DE CONTROL DE EPROM A LEER

22716 :

CTL = 1 'PALABRA DE CONTROL PARA ESTA MEMORIA A LEER

GOTO INTRODAT

22732 :

CTL = 3

GOTO INTRODAT

22764 :

CTL = 5

GOTO INTRODAT

```
227128 :  
CTL = 7  
GOTO INTRODAT  
227256 :  
CTL = 9  
GOTO INTRODAT  
227512 :  
CTL = 11  
GOTO INTRODAT
```

```
INTRODAT:                'LECTURA DE UNA EPROM  
IO = 0: I1 = 0: I2 = 0: I3 = 0: F0 = 0: F1 = 0: F2 = 0: F3 = 0  
OUT &H2BF, CTL  
'PRINT CTL  
IF CTL = 1 THEN  
    DF = 2047                'DIRECCION FINAL DE ESTA EPROM  
    DG$ = "[07FF]"  
END IF  
IF CTL = 3 THEN  
    DF = 4095  
    DG$ = "[0FFF]"  
END IF  
IF CTL = 5 THEN  
    DF = 8191  
    DG$ = "[1FFF]"  
END IF  
IF CTL = 7 THEN  
    DF = 16383  
    DG$ = "[3FFF]"  
END IF  
IF CTL = 9 THEN  
    DF = 32767  
    DG$ = "[7FFF]"
```

```

END IF
IF CTL = 11 THEN
    DF = 65535
    DG$ = "[FFFF]"
END IF
K = 9
G = H = 0
LOCATE 6, 23          'CREACION DE LA VENTANA
FOR i = 0 TO 35
    PRINT CHR$(0);
NEXT i
K = 0
FOR i = 0 TO 19
    LOCATE K + 7, 23
    FOR J = 0 TO 35
        PRINT CHR$(0);
    NEXT J
    COLOR 8: PRINT CHR$(178);
    K = K + 1
NEXT i
LOCATE 27, 24
FOR i = 0 TO 35
    COLOR 8: PRINT CHR$(178);
NEXT i
J = 0
COLOR 15
COLOR 14
LOCATE 6, 25
INPUT "ARCHIVO"; ARCH$    'ENTRADA DEL NOMBRE DEL ARCHIVO
OPEN ARCH$ FOR OUTPUT AS #1    'APERTURA DE ARCHIVO PARA SALIDA
colocar2:
    LOCATE 8, 25
    COLOR 15          'MENSAJE DE COLOCAR EPROM Y CONTINUAR
    PRINT "COLOQUE EPROM EN PROGRAMADOR"
    LOCATE 9, 25

```

```

    PRINT "  PRESIONE C PARA CONTINUAR "
COLOCAR12:
    K$ = INKEY$
    IF K$ = "C" OR K$ = "c" THEN GOTO ABRIR1
    GOTO COLOCAR12
ABRIR1:
    LOCATE 8, 25
    FOR p = 0 TO 30
        PRINT CHR$(0);
    NEXT p
    LOCATE 9, 25
    FOR p = 0 TO 30
        PRINT CHR$(0);
    NEXT p
    LOCATE 7, 25: PRINT "INICIO      FINAL "; DG$
INI5:
    G = I0 + I1 + I2 + I3          'DIRECCION INICIAL
    H = F0 + F1 + F2 + F3          'DIRECCION FINAL
    LOCATE 9, 25: PRINT HEX$(G)
    LOCATE 9, 35: PRINT HEX$(H)
    IF J = 8 THEN
        IF H < G THEN
            LOCATE 7, 25: PRINT "!!!ERROR:  RANGO NEGATIVO!!!"
            GOTO quit5
        END IF
        IF H > DF THEN
            LOCATE 7, 25: PRINT "!!!ERROR:  RANGO SOBREPASADO!!!"
            GOTO quit5
        END IF
        GOTO final5
quit5:
    K$ = INKEY$
    IF K$ = CHR$(27) THEN
        CLOSE #1

```

```

        I0 = 0
        I1 = 0
        I2 = 0
        I3 = 0
        F0 = 0
        F1 = 0
        F2 = 0
        F3 = 0      'HACER CERO LAS DIRECCIONES ERRONEAS
        KILL ARCH$   'ELIMINACION DE ARCHIVO CON ERROR
        GOTO SAL5

    END IF
    GOTO quit5

END IF

                                'BUCLE PARA ENTRADA DE DIRECCIONES

INI6:
K$ = INKEY$
IF K$ = "0" THEN GOTO C0
IF K$ = "1" THEN GOTO C1
IF K$ = "2" THEN GOTO C2
IF K$ = "3" THEN GOTO C3
IF K$ = "4" THEN GOTO C4
IF K$ = "5" THEN GOTO C5
IF K$ = "6" THEN GOTO C6
IF K$ = "7" THEN GOTO C7
IF K$ = "8" THEN GOTO C8
IF K$ = "9" THEN GOTO C9
IF K$ = "A" OR K$ = "a" THEN GOTO CA
IF K$ = "B" OR K$ = "b" THEN GOTO CB
IF K$ = "C" OR K$ = "c" THEN GOTO CC
IF K$ = "D" OR K$ = "d" THEN GOTO CD
IF K$ = "E" OR K$ = "e" THEN GOTO CE
IF K$ = "F" OR K$ = "f" THEN GOTO CF

```

GOTO INI6

'RETORNO DEL BUCLE
'ASIGNACION DE VALORES NUMERICOS

C0:

x = 0

GOTO BACK4

C1:

x = 1

GOTO BACK4

C2:

x = 2

GOTO BACK4

C3:

x = 3

GOTO BACK4

C4:

x = 4

GOTO BACK4

C5:

x = 5

GOTO BACK4

C6:

x = 6

GOTO BACK4

C7:

x = 7

GOTO BACK4

C8:

x = 8

GOTO BACK4

C9:

x = 9

GOTO BACK4

CA:

x = 10

GOTO BACK4

CB:

x = 11

GOTO BACK4

CC:

x = 12

GOTO BACK4

CD:

x = 13

GOTO BACK4

CE:

x = 14

GOTO BACK4

CF:

x = 15

GOTO BACK4

'ROUTINA PARA FORMAR LAS DIRECCIONES

BACK4:

'FORMACION DE DIRECCION INICIAL

IF J = 0 THEN

J = J + 1: I3 = x * 4096: GOTO INI5

END IF

IF J = 1 THEN

J = J + 1: I2 = x * 256: GOTO INI5

END IF

IF J = 2 THEN

J = J + 1: I1 = x * 16: GOTO INI5

END IF

IF J = 3 THEN

J = J + 1: IO = x: GOTO INI5

END IF

'FORMACION DE DIRECCION FINAL

IF J = 4 THEN

J = J + 1: F3 = x * 4096: GOTO INI5

```

END IF
IF J = 5 THEN
    J = J + 1: F2 = x * 256: GOTO INI5
END IF
IF J = 6 THEN
    J = J + 1: F1 = x * 16: GOTO INI5
END IF
IF J = 7 THEN
    J = J + 1: F0 = x: GOTO INI5
END IF

                                'FORMACION DEL RANGO

final5:
RANGO = H - G + 1
LOCATE 8, 29: PRINT "RANGO="; RANGO
o = 1
FOR r = 1 TO RANGO
    IF o = 17 THEN
        o = 1
        FOR S = 0 TO 20
            LOCATE (S + 6), 23
            FOR i = 0 TO 35
                PRINT CHR$(0);
            NEXT i
        NEXT S
    END IF
    o = o + 1
    LOCATE (o + 9), 35: PRINT ;
    DIR$ = HEX$(G)
    PRINT DIR$;
    U = 0
    IF G = 0 THEN
        M3$ = "0"
        M2$ = "0"

```

```

        M1$ = "0"
        M0$ = "0"
        GOTO PARP8
    END IF
    IF G <= 15 THEN
        M3$ = "0"
        M2$ = "0"
        M1$ = "0"
        M0$ = MID$(DIR$, 1, 1)
        GOTO PARP8
    END IF
    IF G <= 255 THEN
        M3$ = "0"
        M2$ = "0"
        M1$ = MID$(DIR$, 1, 1)
        M0$ = MID$(DIR$, 2, 1)
        GOTO PARP8
    END IF
    IF G <= 4095 THEN
        M3$ = "0"
        M2$ = MID$(DIR$, 1, 1)
        M1$ = MID$(DIR$, 2, 1)
        M0$ = MID$(DIR$, 3, 1)
        GOTO PARP8
    END IF
    IF G <= 65535 THEN
        M3$ = MID$(DIR$, 1, 1)
        M2$ = MID$(DIR$, 2, 1)
        M1$ = MID$(DIR$, 3, 1)
        M0$ = MID$(DIR$, 4, 1)
    END IF
    FOR H = 0 TO 1          'BUCLE PARA FORMAR 2 PALABRAS DE 8 BITS
        FOR d = 0 TO 3      'BUCLE PARA CONVERTIR 4 EN 2 PALABRAS
            IF d = 0 THEN b$ = M0$
            IF d = 1 THEN b$ = M1$

```

```

IF d = 2 THEN b$ = M2$
IF d = 3 THEN b$ = M3$
IF b$ = "0" THEN b = 0      'ASIGNACION DE VALORES
IF b$ = "1" THEN b = 1
IF b$ = "2" THEN b = 2
IF b$ = "3" THEN b = 3
IF b$ = "4" THEN b = 4
IF b$ = "5" THEN b = 5
IF b$ = "6" THEN b = 6
IF b$ = "7" THEN b = 7
IF b$ = "8" THEN b = 8
IF b$ = "9" THEN b = 9
IF b$ = "A" THEN b = 10
IF b$ = "B" THEN b = 11
IF b$ = "C" THEN b = 12
IF b$ = "D" THEN b = 13
IF b$ = "E" THEN b = 14
IF b$ = "F" THEN b = 15
IF d = 0 THEN e = b
IF d = 1 THEN F = b * 16
IF d = 2 THEN m = b
IF d = 3 THEN n = b * 16
NEXT d
DIR0 = e + F      'PALABRA FORMADA
DIR1 = m + n
      'SALIDA DE PALABRAS A DIRECCIONES DE E/S
IF H = 1 THEN      'SALIDA DE PALABRA A DIR ALTA
    LOCATE 20, 15
    OUT &H2BD, DIR1      'SALIDA DE DIR ALTA A DIR 2BD
END IF
IF H = 0 THEN      'SALIDA DE PALABRA A DIR BAJA
    LOCATE 20, 20
    OUT &H2BE, DIR0      'SALIDA DE DIR BAJA A DIR 2BE
END IF
NEXT H

```

PARP8:

 DAT = INP(&H2BB)

 IF DAT = 0 THEN

 D1\$ = "0"

 D0\$ = "0"

 GOTO PARP18

 END IF

 IF DAT <= 15 THEN

 D1\$ = "0"

 D0\$ = MID\$(DATA\$, 1, 1)

 GOTO PARP18

 END IF

 IF DAT <= 255 THEN

 DATA\$ = HEX\$(DAT)

 D1\$ = MID\$(DATA\$, 1, 1)

 D0\$ = MID\$(DATA\$, 2, 1)

 END IF

PARP18:

 WRITE #1, M3\$, M2\$, M1\$, M0\$, D1\$, D1\$

 LOCATE (o + 9), 45

 PRINT D1\$; D0\$

 G = G + 1

NEXT r

SAL5:

CLOSE #1

T\$ = INKEY\$

IF T\$ = CHR\$(27) THEN

 LOCATE 6, 20: COLOR 7

 FOR Y = 0 TO 40

 FOR x = 0 TO 21

 LOCATE (x + 6), (Y + 20): PRINT CHR\$(178);

 NEXT x

```

NEXT Y
G = H = 0
CLOSE #1: GOTO LEER
END IF
GOTO SAL5

```

-SECCION PARA TRANSFERIR UN ARCHIVO A UNA EPROM

```

                                'ROUTINA PARA SACAR DATOS DE ARCHIVO Y
                                'TRANSFERIRLOS A LA EPROM SELECCIONADA
saldat: 'SALIDA DE DATOS AL PROGRAMADOR
LOCATE 6, 20                    'CREACION DE VENTANA
FOR i = 0 TO 38
    PRINT CHR$(0);
NEXT i
K = 0
FOR i = 0 TO 1
    LOCATE K + 7, 20
    FOR J = 0 TO 38
        PRINT CHR$(0);
    NEXT J
    COLOR 8: PRINT CHR$(178);
    K = K + 1
NEXT i
LOCATE 8, 20
FOR i = 0 TO 38
    PRINT CHR$(0);
NEXT i
COLOR 8: PRINT CHR$(178);
LOCATE 9, 21: COLOR 8
FOR i = 0 TO 38
    PRINT CHR$(178);
NEXT i
LOCATE 7, 22: COLOR 14
INPUT "ARCHIVO"; a$ 'ENTRADA DEL ARCHIVO A TRANSFERIR
OPEN a$ FOR INPUT AS #1

```

```

DO WHILE NOT EOF(1)
    INPUT #1, m$, n$, o$, p$, q$, r$
LOOP
CLOSE #1
FOR a = 0 TO 3
    IF a = 0 THEN b$ = m$
    IF a = 1 THEN b$ = n$
    IF a = 2 THEN b$ = o$
    IF a = 3 THEN b$ = p$
    IF b$ = "0" THEN d = 0
    IF b$ = "1" THEN d = 1
    IF b$ = "2" THEN d = 2
    IF b$ = "3" THEN d = 3
    IF b$ = "4" THEN d = 4
    IF b$ = "5" THEN d = 5
    IF b$ = "6" THEN d = 6
    IF b$ = "7" THEN d = 7
    IF b$ = "8" THEN d = 8
    IF b$ = "9" THEN d = 9
    IF b$ = "A" THEN d = 10
    IF b$ = "B" THEN d = 11
    IF b$ = "C" THEN d = 12
    IF b$ = "D" THEN d = 13
    IF b$ = "E" THEN d = 14
    IF b$ = "F" THEN d = 15
    IF a = 0 THEN e = d * 4096
    IF a = 1 THEN F = d * 256
    IF a = 2 THEN G = d * 16
    IF a = 3 THEN H = d
NEXT a
dirfin = e + F + G + H 'DIRECCION FINAL DEL ARCHIVO
                        'VENTANA DE SELECCION DE EPROM A GRABAR

GRABAR:
LOCATE 4, 60: COLOR 15: PRINT "  GRABAR EPROM # "
LOCATE 5, 60: COLOR 15: PRINT "2716"; : COLOR 12: PRINT "►  1 ";

```

```

COLOR 8: PRINT CHR$(178);
LOCATE 6, 60: COLOR 15: PRINT "2732"; : COLOR 12: PRINT "► 2 ";
COLOR 8: PRINT CHR$(178);
LOCATE 7, 60: COLOR 15: PRINT "2764"; : COLOR 12: PRINT "► 3 ";
COLOR 8: PRINT CHR$(178);
LOCATE 8, 60: COLOR 15: PRINT "27128"; :COLOR 12: PRINT "► 4 ";
COLOR 8: PRINT CHR$(178);
LOCATE 9, 60: COLOR 15: PRINT "27256"; :COLOR 12: PRINT "► 5 ";
COLOR 8: PRINT CHR$(178);
LOCATE 10, 60: COLOR 15: PRINT "27512";:COLOR 12: PRINT "► 6 ";
COLOR 8: PRINT CHR$(178);
LOCATE 11, 61: COLOR 8
FOR i = 0 TO 16          'SOMBRA INFERIOR DE LA VENTANA
    PRINT CHR$(178);
NEXT i
IF dirfin <= 2047 THEN
    LOCATE 10, 10
    COLOR 15
    PRINT "Use la 2716 [1] o Superior"  'MENSAJE DE MEMORIA
    GOTO seleccion1          'SUGERIDA
END IF
IF dirfin <= 4095 THEN
    LOCATE 10, 10
    COLOR 15
    PRINT "Use la 2732 [2] o Superior"
    GOTO seleccion2
END IF
IF dirfin <= 8191 THEN
    LOCATE 10, 10
    COLOR 15
    PRINT "Use la 2764 [3] o Superior"
    GOTO seleccion3
END IF
IF dirfin <= 16383 THEN
    LOCATE 10, 10

```



```

    COLOR 15
    PRINT "Use la 27128 [4] o Superior"
    GOTO seleccion4
END IF
IF dirfin <= 32767 THEN
    LOCATE 10, 10
    COLOR 15
    PRINT "Use la 27256 [5] o Superior"
    GOTO seleccion5
END IF
IF dirfin >= 65535 THEN
    LOCATE 10, 10
    COLOR 15
    PRINT "Use la 27512 [6]"
    GOTO seleccion6
END IF

```

'ROUTINA DE SELECCION DE EPROM

'INHABILITACION DE OPCIONES DE MENU LE

EPROM

seleccion1:

ab\$ = INKEY\$

```

    IF ab$ = "1" THEN GOTO 12716      'SALTOS A RUTINA DE GRABACION
    IF ab$ = "2" THEN GOTO 12732      'DE CADA EPROM
    IF ab$ = "3" THEN GOTO 12764
    IF ab$ = "4" THEN GOTO 127128
    IF ab$ = "5" THEN GOTO 127256
    IF ab$ = "6" THEN GOTO 127512
    IF ab$ = CHR$(27) THEN GOTO ventanaoff 'SALIDA DE SUB-MENU
                                          'DE SELECCION DE EPROM

```

```

GOTO seleccion1      'RETORNO A LA RUTINA DE SUB-MENU DE
SELECCION

```

seleccion2:

ab\$ = INKEY\$

```

    IF ab$ = "2" THEN GOTO 12732
    IF ab$ = "3" THEN GOTO 12764

```

```

    IF ab$ = "4" THEN GOTO 127128
    IF ab$ = "5" THEN GOTO 127256
    IF ab$ = "6" THEN GOTO 127512
    IF ab$ = CHR$(27) THEN GOTO ventanaoff
GOTO seleccion2      'RETORNO A LA RUTINA DE SUB-MENU DE SELECCION
seleccion3:
ab$ = INKEY$
    IF ab$ = "3" THEN GOTO 12764
    IF ab$ = "4" THEN GOTO 127128
    IF ab$ = "5" THEN GOTO 127
    IF ab$ = "6" THEN GOTO 127512
    IF ab$ = CHR$(27) THEN GOTO ventanaoff
GOTO seleccion3      'RETORNO A LA RUTINA DE SUB-MENU DE SELECCION
seleccion4:
ab$ = INKEY$
    IF ab$ = "4" THEN GOTO 127128
    IF ab$ = "5" THEN GOTO 127256
    IF ab$ = "6" THEN GOTO 127512
    IF ab$ = CHR$(27) THEN GOTO ventanaoff
GOTO seleccion4      'RETORNO A LA RUTINA DE SUB-MENU DE SELECCION
seleccion5:
ab$ = INKEY$
    IF ab$ = "5" THEN GOTO 127256
    IF ab$ = "6" THEN GOTO 127512
    IF ab$ = CHR$(27) THEN GOTO ventanaoff
GOTO seleccion5      'RETORNO A LA RUTINA DE SUB-MENU DE SELECCION
seleccion6:
ab$ = INKEY$
    IF ab$ = "6" THEN GOTO 127512
    IF ab$ = CHR$(27) THEN GOTO ventanaoff
GOTO seleccion6      'RETORNO A LA RUTINA DE SUB-MENU DE SELECCION
ventanaoff:
K = 0
FOR i = 0 TO 15      'ELIMINACION DE LA VENTANA
    LOCATE K + 4, 60

```

```

    FOR J = 0 TO 17
        COLOR 7: PRINT CHR$(178);
    NEXT J
    K = K + 1    'ELIMINACION DE VENTANA Y BARRA DE PORCENTAJE
NEXT i
LOCATE 10, 10
COLOR 7
FOR a = 0 TO 30
    PRINT "█";
NEXT a
FOR a = 0 TO 3
    LOCATE 6 + a, 20
    FOR b = 0 TO 39
        COLOR 7
        PRINT "█";
    NEXT b
NEXT a
FOR a = 0 TO 2
    LOCATE 23 + a, 19
    FOR b = 0 TO 36
        COLOR 7
        PRINT "█";
    NEXT b
NEXT a
GOTO menuppal
    'ROUTINA PARA CARGAR PALABRA DE CONTROL DE EPROM A GRABAR

12716 :
CTL = 0          'PALABRA DE CONTROL PARA ESTA MEMORIA A GRABAR
GOTO saldat1     'SALTO A ROUTINA DE GRABACION DE EPROM
12732 :
CTL = 2
GOTO saldat1
12764 :
CTL = 4

```

```

GOTO saldat1
127128 :
CTL = 6
GOTO saldat1
127256 :
CTL = 8
GOTO saldat1
127512 :
CTL = 10
GOTO saldat1
saldatl:
COLOR 15          'CREACION DE BARRA DE PORCENTAJE DE GRABACION
LOCATE 23, 19
FOR a = 0 TO 36
    PRINT " ";
NEXT a
LOCATE 24, 20
FOR a = 0 TO 29
    PRINT CHR$(178);
NEXT a
PRINT "      ";
LOCATE 25, 19
FOR a = 0 TO 36
    PRINT CHR$(0);
NEXT a
LOCATE 24, 19: PRINT " "; : LOCATE 24, 55: PRINT " ";

          'CARGA DIRECCION DE MONOESTABLE A USAR
          'DE ACUERDO A LA EPROM ELEGIDA

IF CTL = 0 THEN
    puerto = &H2BA
END IF
IF CTL = 2 THEN
    puerto = &H2B9
END IF

```

```

IF CTL = 4 THEN
    puerto = &H2B8
END IF
IF CTL = 6 THEN
    puerto = &H2B9
END IF
IF CTL = 8 THEN
    puerto = &H2B8
END IF
IF CTL = 10 THEN
    puerto = &H2B9
END IF
i = 0                                'INICIALIZACION DE VARIABLES
J = 0
x = 0
c = 0
K = 100 / 6
m = K
LOCATE 22, 20                        'SALIDA DE PALABRA DE CONTROL
OPEN a$ FOR INPUT AS #1
DO WHILE NOT EOF(1)
    INPUT #1, p$
    c = c + 1                        'NUMERO DE PALABRAS DEL ARCHIVO
LOOP
x = c / 6                            'NUMERO DE LINEAS DEL ARCHIVO
CLOSE #1
OUT &H2BF, CTL                      'SALIDA DE PALABRA DE CONTROL A PUERTO 2B8
LOCATE 10, 10
COLOR 7
FOR v = 0 TO 30
PRINT " ";
NEXT v
colocar:
    COLOR 15
    LOCATE 8, 22

```

```

PRINT "COLOQUE EPROM EN PROGRAMADOR"      'MENSAJE
LOCATE 9, 22
PRINT "  PRESIONE C PARA CONTINUAR "      'MENSAJE DE CONTINUAR
colocar1:
  K$ = INKEY$
  IF K$ = "C" OR K$ = "c" THEN GOTO ABRIR
  GOTO colocar1
ABRIR:
LOCATE 8, 22: FOR p = 0 TO 30
  PRINT CHR$(0);
NEXT p
LOCATE 9, 22: COLOR 8
  FOR p = 0 TO 30
  PRINT CHR$(178);
NEXT p
OPEN a$ FOR INPUT AS #1      'ABRE ARCHIVO SELECCINADO
DO WHILE NOT EOF(1)
  FOR H = 0 TO 2      'BUCLE PARA FORMAR 3 PALABRAS DE 8 BITS
  FOR d = 0 TO 1 'BUCLE PARA CONVERTIR 2 PALABRAS EN 1 PALABRA
  i = i + 1
  LOCATE 25, 21: COLOR 11: PRINT SIG$
  INPUT #1, b$      'ENTRADA DE PALABRA
  IF b$ = "0" THEN b = 0      'ASIGNACION DE VALORES
  IF b$ = "1" THEN b = 1
  IF b$ = "2" THEN b = 2
  IF b$ = "3" THEN b = 3
  IF b$ = "4" THEN b = 4
  IF b$ = "5" THEN b = 5
  IF b$ = "6" THEN b = 6
  IF b$ = "7" THEN b = 7
  IF b$ = "8" THEN b = 8
  IF b$ = "9" THEN b = 9
  IF b$ = "A" THEN b = 10
  IF b$ = "B" THEN b = 11
  IF b$ = "C" THEN b = 12

```

```

IF b$ = "D" THEN b = 13
IF b$ = "E" THEN b = 14
IF b$ = "F" THEN b = 15
IF d = 0 THEN e = b * 16      'LSB
IF d = 1 THEN F = b          'MSB
IF i = x THEN
    LOCATE 24, 50      'PORCENTAJE DE GRABACION
    COLOR 4
    PRINT CINT(m); "%"
    LOCATE 24, (20 + J)
    COLOR 9
    PRINT "██████";
    J = J + 5
    m = m + K
    i = 0
END IF
NEXT d
G = e + F      'PALABRA FORMADA
COLOR 7

                'SALIDA DE PALABRAS A DIRECCIONES DE E/S
IF H = 2 THEN      'SALIDA DE PALABRA AL DATO
    LOCATE 20, 30
    OUT &H2BC, G      'SALIDA DE DATO A DIR 2BB
    SIG$ = "\"
END IF
IF H = 1 THEN      'SALIDA DE PALABRA A DIR ALTA
    LOCATE 20, 25
    OUT &H2BE, G      'SALIDA DE DIR BAJA A DIR 2B9
    SIG$ = "|"
END IF
IF H = 0 THEN      'SALIDA DE PALABRA A DIR BAJA
    LOCATE 20, 20
    OUT &H2BD, G      'SALIDA DE DIR ALTA A DIR 2B8
    SIG$ = "/"
END IF

```

```
OUT puerto, 0          'DISPARO DE MONOESTABLE
LOCATE 20, 35
FOR Y = 0 TO 500
NEXT Y
NEXT H
LOOP
tor:                   'ELIMINACION DE VENTANA Y BARRA DE PORCENTAJE
FOR a = 0 TO 3
  LOCATE 6 + a, 20
  FOR b = 0 TO 39
    COLOR 7
    PRINT "█";
  NEXT b
NEXT a
FOR a = 0 TO 2
  LOCATE 23 + a, 19
  FOR b = 0 TO 36
    COLOR 7
    PRINT "█";
  NEXT b
NEXT a
CLOSE #1
GOTO ventanaoff
```


APENDICE B. DIAGRAMAS ELECTRONICOS.

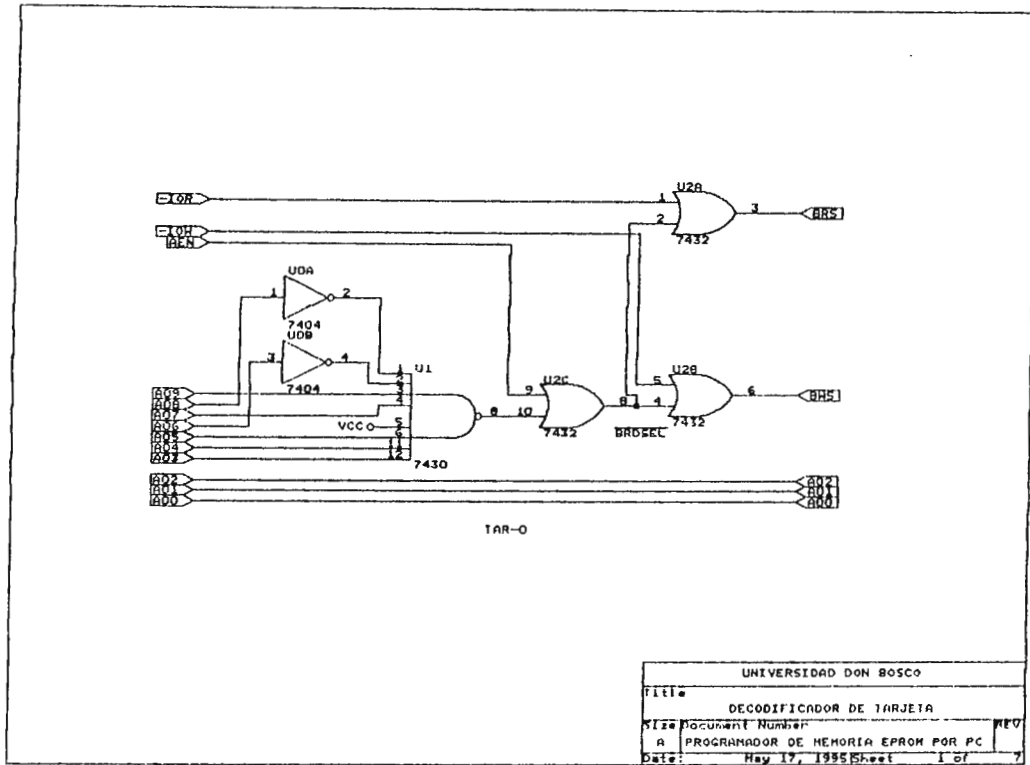


Diagrama 1/7. Decodificador de Tarjeta

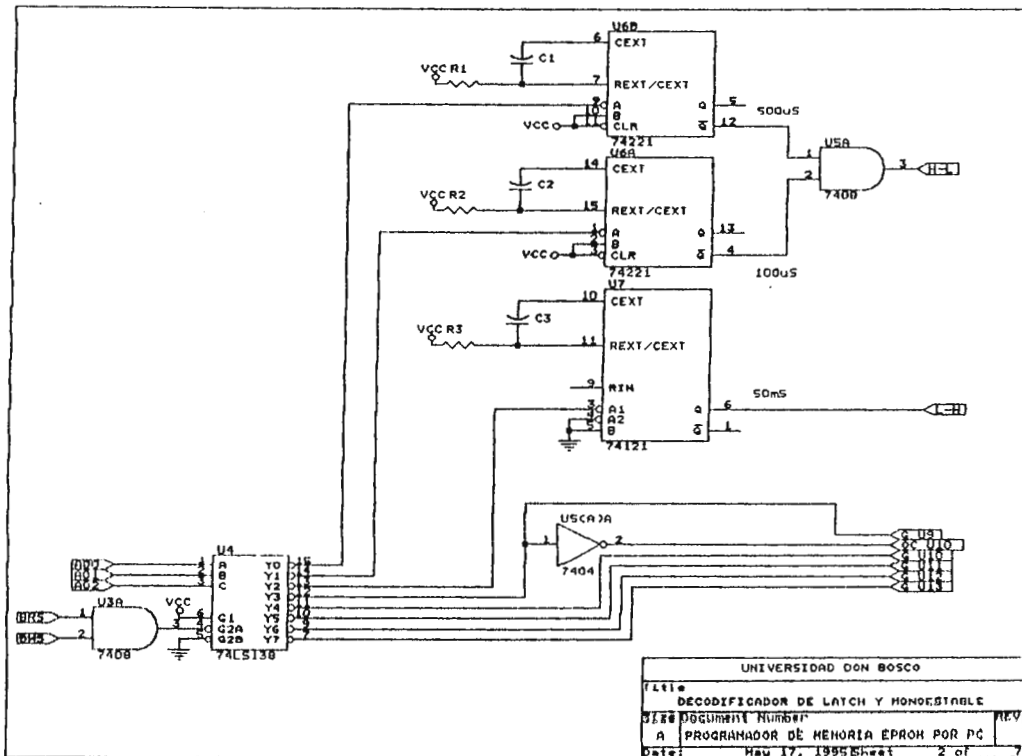


Diagrama 2/7. Decodificador de Latch y Monoestable.

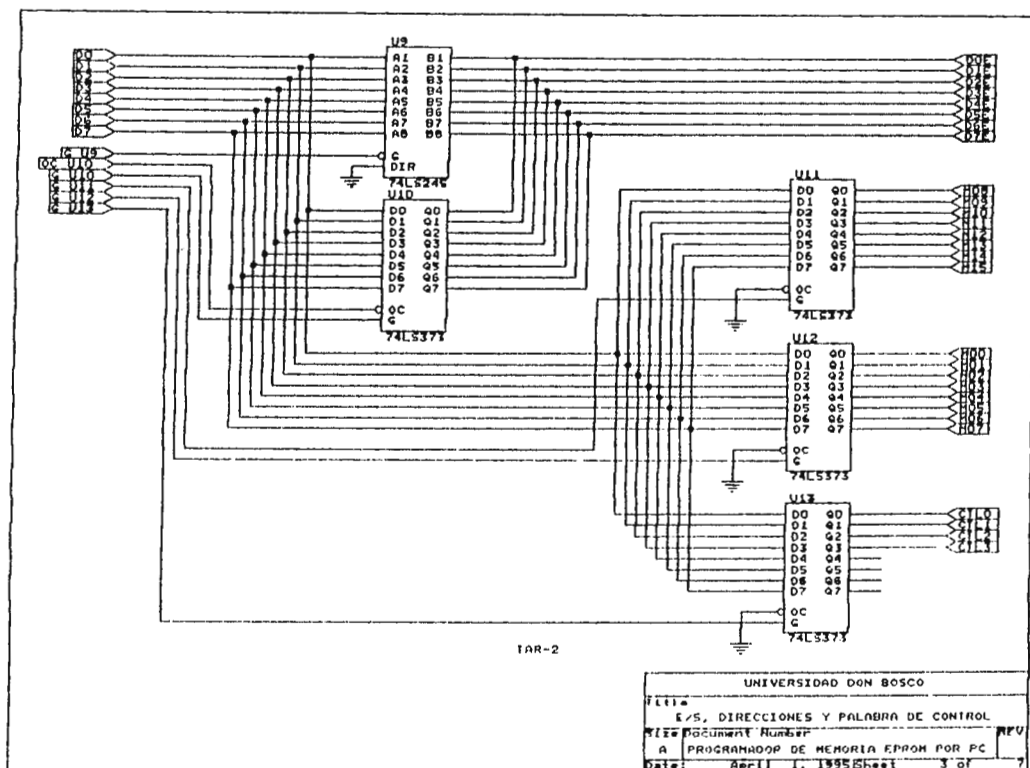


Diagrama 3/7. E/S, captura de Direcciones y Palabra de Control.

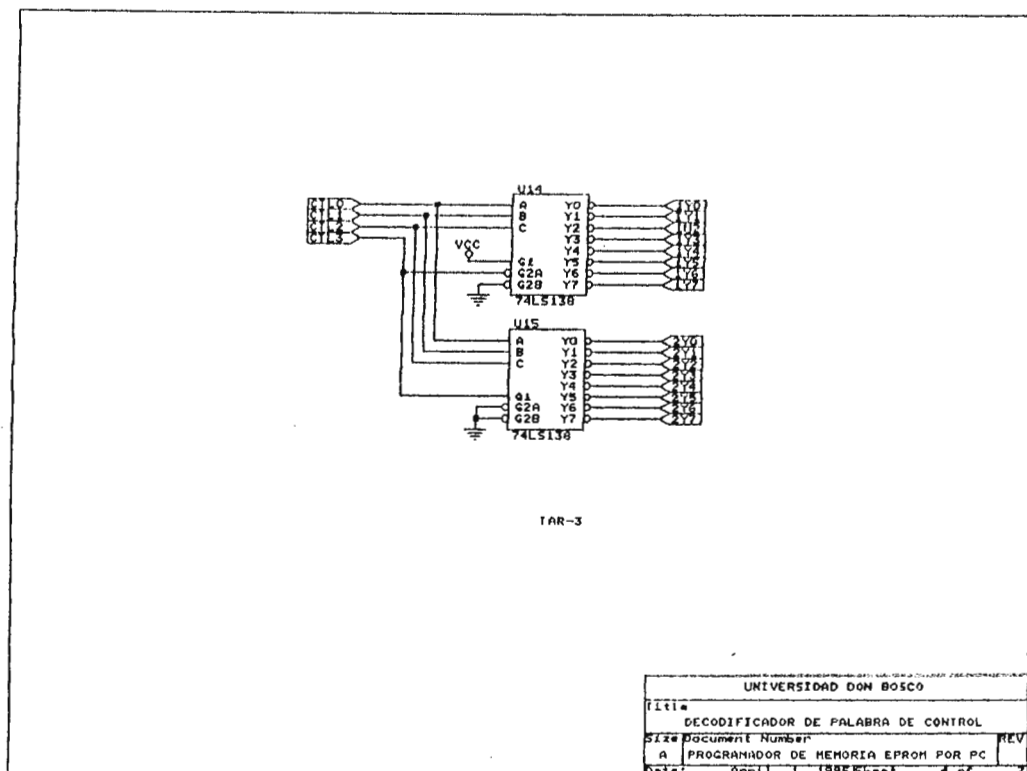


Diagrama 4/7. Decodificador de Palabra de Control.

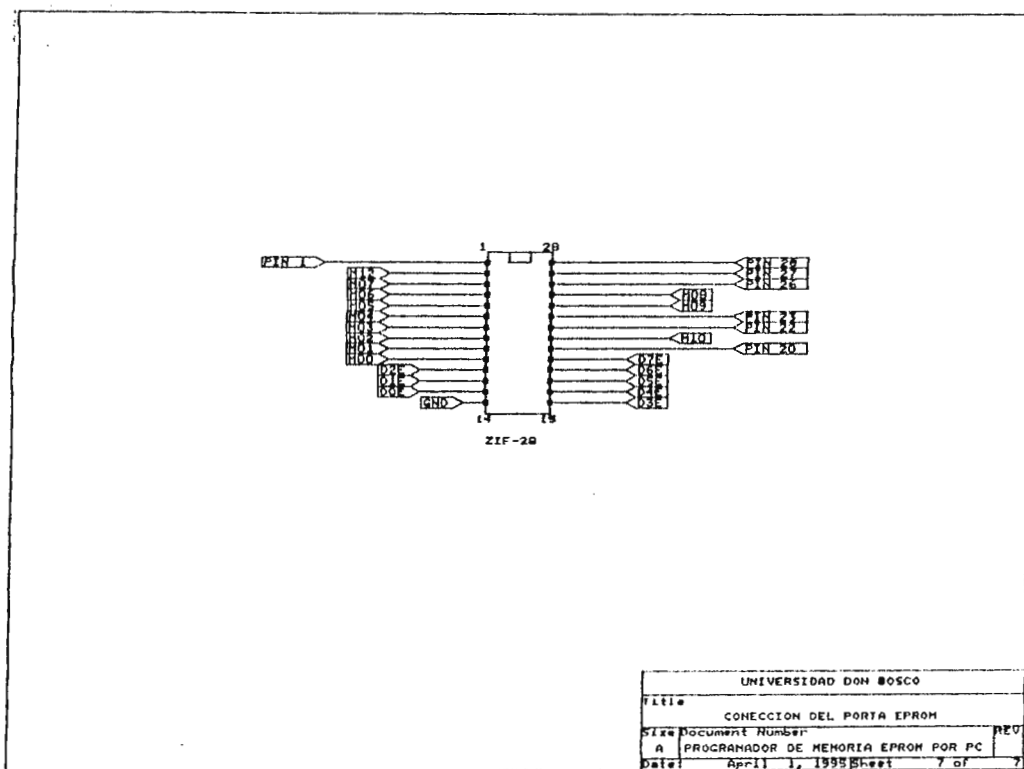


Diagrama 7/7. Conecciones del Porta EPROM.

APENDICE C

CALCULOS

- Calculo para resistencias R4 y R11

$$V_{cc} = 25 \text{ V}$$

$$I_c = 10\text{mA}$$

$$25 - (10\text{mA})(R) - 0.7 - 6 = 0$$

$$R = 1.83 \text{ k}\Omega \approx 1.8 \text{ K}\Omega$$

$$PR = (25)^2 \div 1.8 \text{ K}\Omega$$

$$PR = 0.347 \text{ Watt.} \approx 1/2 \text{ Watt.}$$

- Calculo para resistencias R5 y R12

$$V_{cc} = 25 \text{ V}$$

$$I_c = 10\text{mA}$$

$$I_b = 1/10 (10\text{mA}) \text{ ó } V_{cc} \div (10)(R4)$$

$$4.5 - [1/10 (10\text{mA})] R - 0.7 = 0$$

$$R = 2.7 \text{ K}\Omega$$

$$PR = 1/4 \text{ Watt.}$$

- Calculo para resistencia R6 y R9

$$I_c = 10\text{mA}$$

$$V_{cc} = 25 \text{ V}$$

$$25 - (10\text{mA})(R) - 0.7 - 5 = 0$$

$$R = 1.93 \text{ K}\Omega \approx 2.0 \text{ K}\Omega$$

$$PR = (25)^2 \div 2\text{K}\Omega$$

$$PR = 0.3125 \text{ WATT} \approx 1/2 \text{ WATT.}$$

- Calculo para resistencias R7 y R10

$$V_{cc} = 25$$

$$I_c = 10\text{mA}$$

$$I_b = 1/10 (10\text{mA})$$

$$4.5 - [1/10(10\text{mA})] R - 0.7 = 0$$

$$R = 2.93 \text{ K}\Omega \approx 3.0 \text{ K}\Omega$$

$$PR = 1/4 \text{ de Watt.}$$

- Calculo para resistencias R8,R18,R13 Y R21.

NOTA: Sirven para asegurar estado (Tierra), teniendo que ser de un valor elevado para no demandar mucha corriente al circuito.

El valor sugerido es de $1\text{M}\Omega$. (1/4 de Watt).

- Calculo para R16

$$I_c = 30\text{mA}$$

$$V_{cc} = 25 \text{ V}$$

$$25 - 30\text{mA}(R) - 0.7 - 12.8 = 0$$

$$R = 383.3 \text{ }\Omega \approx 390\Omega$$

$$PR = (25)^2 \div 390 \text{ }\Omega = 1.602 \text{ Watt} \approx 2 \text{ Watt.}$$

- Calculo para R17

$$V_{cc} = 25 \text{ V}$$

$$I_c = 30 \text{ mA}$$

$$I_b = 1/10 (30\text{mA})$$

$$4.5 - [1/10(30\text{mA})]R - 0.7 = 0$$

$$R = 592.8 \, \Omega \approx 560 \, \Omega$$

$$PR = 1/4 \text{ de Watt.}$$

- Calculo para R1 (MONOESTABLE)

$$C1 = 4.7 \, \mu\text{F}$$

$$0.7 (R)(4.7\mu\text{F}) = 500 \, \mu\text{S}$$

$$R = 150 \, \Omega$$

- Calculo para R2 (MONOESTABLE)

$$C2 = 0.1 \, \mu\text{F}$$

$$0.7 (R)(0.1\mu\text{F}) = 100 \, \mu\text{S}$$

$$R = 1.5 \, \text{K}\Omega$$

- Calculo para R3 (MONOESTABLE)

$$C3 = 4.7 \, \mu\text{F}$$

$$0.7 (R)(4.7\mu\text{F}) = 50 \, \text{mS}$$

$$R = 15 \, \text{K}\Omega$$

LISTA DE COMPONENTES CALCULADOS

C1, C3 = 4.7 μ F

C2 = 0.1 μ F

R1 = 150 Ω

R2 = 1.5K Ω

R3 = 15K Ω

R4, R11 = 1.8K Ω

R5, R12 = 2.7K Ω

R6, R9 = 2.0K Ω

R7, R10 = 3.0K Ω

R8, R18, R13, R21 = 1M Ω

R16, R19 = 390 Ω

R17, R20 = 560 Ω

Q1, Q2, Q3, Q4, Q5, Q6, Q7 = 2N2222 (0.8 Amp. Max)

Diodos = 1N914 (0.2 Amp. Max)

NOTA: Se seleccionaron los Diodos 1N914 tomando en cuenta que en los circuitos las corrientes que pasan por ellos son iguales o menores a 30mA.



NMC27C16 16,384-Bit (2048 x 8) UV Erasable CMOS EPROM

General Description

The NMC27C16 is a high speed 16k UV erasable and electrically reprogrammable CMOS EPROM, ideally suited for applications where fast turnaround, pattern experimentation and low power consumption are important requirements.

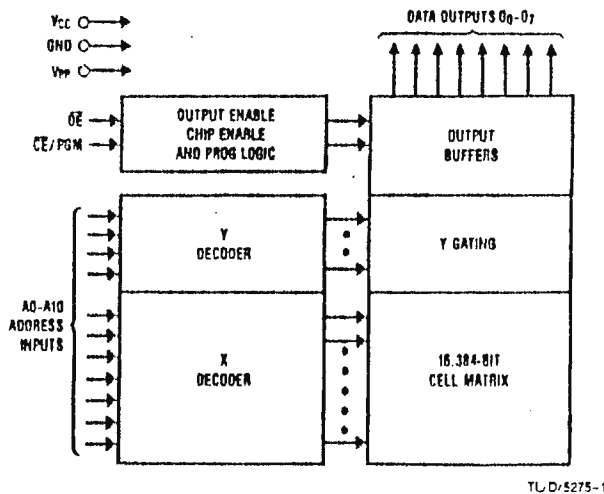
The NMC27C16 is packaged in a 24-pin dual-in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written into the device by following the programming procedure.

This EPROM is fabricated with the reliable, high volume, time proven, P²C²MOS[™] silicon gate technology.

Features

- Access time down to 170 ns
- Low CMOS power consumption
 - Active Power: 26.25 mW max
 - Standby Power: 0.53 mW max (98% savings)
- Performance compatible to iNS300[™] CMOS microprocessor
- Single 5V power supply
- Extended temperature range available (NMC27C16E-45), -40°C to +85°C, 450 ns ±5% power supply
- Pin compatible to MM2716 and higher density EPROMs
- Static—no clocks required
- TTL compatible inputs/outputs
- TRI-STATE[®] output

Block Diagram



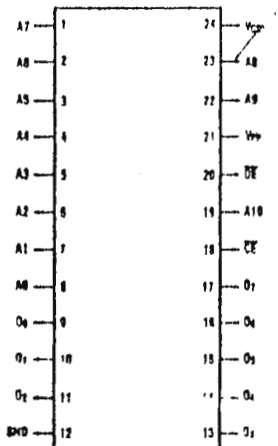
Pin Names

A0-A10	Addresses
CE	Chip Enable
OE	Output Enable
D0-D7	Outputs
PGM	Program
NC	No Connect

Connection Diagram

27C256	27C128	27C64	27C32
27256	27128	2764	2732
V _{PP}	V _{PP}	V _{PP}	
A12	A12	A12	
A7	A7	A7	A7
A6	A6	A6	A6
A5	A5	A5	A5
A4	A4	A4	A4
A3	A3	A3	A3
A2	A2	A2	A2
A1	A1	A1	A1
A0	A0	A0	A0
O ₀	O ₀	O ₀	O ₀
O ₁	O ₁	O ₁	O ₁
O ₂	O ₂	O ₂	O ₂
GND	GND	GND	GND

Dual-In-Line Package
NMC27C16



27C32	27C16	27C8	27C256
2732	2716	2708	27256
V _{PP}	V _{PP}	V _{PP}	V _{PP}
A8	A8	A8	A13
A9	A9	A9	A8
A11	A11	A11	A9
CE/V _{PP}	CE	CE	A11
A10	A10	A10	A10
CE	CE	CE	CE
O ₇	O ₇	O ₇	O ₇
O ₆	O ₆	O ₆	O ₆
O ₅	O ₅	O ₅	O ₅
O ₄	O ₄	O ₄	O ₄
O ₃	O ₃	O ₃	O ₃

TL D/5275-2

Top View

Note: Socket compatible EPROM pin configurations are shown in the blocks adjacent to the NMC27C16 pins.

Order Number NMC27C16

See NS Package Number J24AQ

Commercial Temp Range (0°C to +70°C) V_{CC} = 5V ± 5%

Parameter/Order Number	Access Time (ns)
NMC27C16-30	300
NMC27C16-35	350
NMC27C16-45	450
NMC27C16-55	550

DC CHARACTERISTICS

DC Programming Characteristics (Notes 2 & 3)

($T_A = +25^\circ\text{C} \pm 5^\circ\text{C}$, $V_{CC} = 5\text{V} \pm 5\%$, $V_{PP} = 25\text{V} \pm 1\text{V}$)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Current (for Any Input)	$V_{IH} = V_{CC}$ or GND			10	μA
I_{PO}	V_{CC} Supply Current During Programming Pulse	$\overline{CE}/PGM = V_{IH}$			30	mA
I_{CC}	V_{CC} Supply Current				10	mA
V_{IL}	Input Low Level		-0.1		0.0	V
V_{IH}	Input High Level		2.0		$V_{CC} + 1$	V

AC Programming Characteristics (Notes 2 & 3)

($T_A = +25^\circ\text{C} \pm 5^\circ\text{C}$, $V_{CC} = 5\text{V} \pm 5\%$, $V_{PP} = 25\text{V} \pm 1\text{V}$)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
t_{AS}	Address Setup Time		2			μs
t_{CES}	\overline{OE} Setup Time		2			μs
t_{DS}	Data Setup Time		2			μs
t_{AH}	Address Hold Time		2			μs
t_{CEH}	\overline{OE} Hold Time		2			μs
t_{DH}	Data Hold Time		2			μs
t_{CF}	Output Enable to Output Float Delay	$\overline{OE}/PGM = V_{IL}$	0		160	ns
t_{OE}	Output Enable to Output Delay	$\overline{OE}/PGM = V_{IH}$			160	ns
t_{PW}	Program Pulse Width		45	50	55	ms
t_{PRT}	Program Pulse Rise Time		5			ns
t_{PFT}	Program Pulse Fall Time		5			ns

AC Test Conditions

V_{CC}	$5\text{V} \pm 5\%$	Timing Measurement Reference Level	
V_{PP}	$25\text{V} \pm 1\text{V}$	Inputs	1V and 2V
Input Rise and Fall Times	$\leq 20\text{ ns}$	Outputs	0.8V and 2V
Input Pulse Levels	0.8V to 2.2V		

Absolute Maximum Ratings (Note 1)

If Military Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Temperature Under Bias -10°C to $+80^{\circ}\text{C}$

Storage Temperature -65°C to $+125^{\circ}\text{C}$

All Input Voltages with Respect to Ground -0.5V to $+0.3\text{V}$

All Output Voltages with Respect to Ground (Note 11) $V_{\text{CC}} + 0.3\text{V}$ to $\text{GND} - 0.3\text{V}$

V_{FD} Supply Voltage with Respect to Ground During Programming $+25.5\text{V}$ to -0.3V

Power Dissipation 1.0W

Lead Temperature (Soldering, 10 seconds) 260°C

Operating Conditions (Note 9)

Temperature Range
NMC27C16-30, -35, -45, -55 0°C to $+70^{\circ}\text{C}$
NMC27C16E-45 -40°C to $+85^{\circ}\text{C}$

V_{CC} Power Supply (Notes 2 and 3) $5\text{V} \pm 5\%$

V_{FD} Power Supply (Note 3) V_{CC}

READ OPERATION**DC Electrical Characteristics**

Symbol	Parameter	Conditions	Min	Typ (Note 4)	Max	Units
I_{LI}	Input Load Current	$V_{\text{IH}} = V_{\text{CC}}$ or GND			10	μA
I_{LO}	Output Leakage Current	$V_{\text{OUT}} = V_{\text{CC}}$ or GND, $\overline{\text{CE}} = V_{\text{IH}}$			10	μA
I_{CC1} (Note 3)	V_{CC} Current (Active) TTL Inputs	$\overline{\text{OE}} = \overline{\text{CE}} = V_{\text{IL}}$, $f = 1\text{ MHz}$ Inputs = V_{IH} or V_{IL} , $I/O = 0\text{ mA}$		2	10	mA
I_{CC2} (Note 3)	V_{CC} Current (Active) CMOS Inputs	$\overline{\text{OE}} = \overline{\text{CE}} = V_{\text{IL}}$, $f = 1\text{ MHz}$ Inputs = V_{CC} or GND, $I/O = 0\text{ mA}$		1	5	mA
I_{CCSB1}	V_{CC} Current (Standby) TTL Inputs	$\overline{\text{CE}} = V_{\text{IH}}$		0.1	1	mA
I_{CCSB2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{\text{CE}} = V_{\text{CC}}$		0.01	0.1	mA
V_{IL}	Input Low Voltage		-0.1		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{\text{CC}} + 1$	V
V_{OL1}	Output Low Voltage	$I_{\text{OL}} = 2.1\text{ mA}$			0.45	V
V_{OH1}	Output High Voltage	$I_{\text{OH}} = -400\text{ }\mu\text{A}$	2.4			V
V_{OL2}	Output Low Voltage	$I_{\text{OL}} = 0\text{ }\mu\text{A}$			0.1	V
V_{OH2}	Output High Voltage	$I_{\text{OH}} = 0\text{ }\mu\text{A}$	$V_{\text{CC}} - 0.1$			V

AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C16								Unit
			-30		-35		E-45, -45		-55		
			Min	Max	Min	Max	Min	Max	Min	Max	
t_{ACC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$		300		350		450		550	ns
t_{CE}	\overline{CE} to Output Delay	$\overline{OE} = V_{IL}$		300		350		450		550	ns
t_{OE}	\overline{OE} to Output Delay	$\overline{CE} = V_{IL}$		120		120		120		160	ns
t_{DF}	\overline{OE} High to Output Float	$\overline{CE} = V_{IL}$	0	100	0	100	0	100	0	100	ns
t_{OH} (Note 5)	Output Hold from Addresses, \overline{CE} or \overline{OE} , Whichever Occurred First	$\overline{CE} = \overline{OE} = V_{IL}$	0		0		0		0		ns

Functional Description

Programming multiple NMC27C16s in parallel with different data can be easily accomplished due to the simple programming requirements. Like inputs of the parallel NMC27C16s may be connected together when they are programmed with the same data. A high level TTL pulse applied to the \overline{CE}/PGM input programs the parallel NMC27C16s.

Program Inhibit

Programming multiple NMC27C16s in parallel with different data is also easily accomplished. Except for \overline{CE}/PGM , all like inputs (including \overline{OE}) of the parallel NMC27C16s may be common. A TTL level program pulse applied to an NMC27C16's \overline{CE}/PGM input with V_{pp} at 25V will program that NMC27C16. A low level \overline{CE}/PGM input inhibits the other NMC27C16 from being programmed.

Program Verify

A verify should be performed on the programmed bits to determine whether they were correctly programmed. The verify may be performed with V_{pp} at 25V. V_{pp} must be at V_{CC} , except during programming and program verify.

ERASURE CHARACTERISTICS

The erasure characteristics of the NMC27C16 are such that erasure begins to occur when exposed to light with wavelengths shorter than approximately 4000 Angstroms (\AA). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000 \AA –4000 \AA range. Opaque labels should be placed over the NMC27C16 window to prevent unintentional erasure. Covering the window will also prevent temporary functional failure due to the generation of photo currents.

The recommended erasure procedure for the NMC27C16 is exposure to short wave ultraviolet light which has a wavelength of 2537 Angstroms (\AA). The integrated dose (i.e., UV intensity \times exposure time) for erasure should be a minimum of 15W-sec/cm². The erasure time with this dosage is approximately 21 minutes using an ultraviolet lamp with a

12,000 mW/cm² power. The NMC27C16 should be placed within 1 inch of the lamp during erasure. Some lamps have a filter on their face which should be removed before erasure.

Note: The NMC27C16-55 may take up to 10 minutes for complete erasure to occur.

An erasure system should be calibrated periodically. The distance from lamp to unit should be maintained at one inch. The erasure time increases as the square of the distance. (If distance is doubled the erasure time increases by a factor of 4.) Lamps lose intensity as they age. When a lamp is changed, the distance has changed, or the lamp has aged, the system should be checked to make certain full erasure is occurring. Incomplete erasure will cause symptoms that can be misleading. Programmers, components, and even system designs have been erroneously suspected when incomplete erasure was the problem.

SYSTEM CONSIDERATION

The power switching characteristics of EPROMs require careful decoupling of the devices. The supply current, I_{CC} , has three segments that are of interest to the system designer—the standby current level, the active current level, and the transient current peaks that are produced on the falling and rising edges of chip enable. The magnitude of these transient current peaks is dependent on the output capacitance loading of the device. The associated transient voltage peaks can be suppressed by properly selected decoupling capacitors. It is recommended that a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND. This should be a high frequency capacitor of low inherent inductance. In addition, a 4.7 μF bulk electrolytic capacitor should be used between V_{CC} and GND for each eight devices. The bulk capacitor should be located near where the power supply is connected to the array. The purpose of the bulk capacitor is to overcome the voltage drop caused by the inductive effects of the PC board traces.

TABLE I. Mode Selection

Mode	Pins \overline{CE}/PGM (16)	\overline{OE} (20)	V_p (21)	V_{CC} (24)	Outputs (9–11, 13–17)
Read	V_{IL}	V_{IL}	V_{CC}	5	D_{OUT}
Standby	V_{IH}	Don't Care	V_{CC}	5	Hi-Z
Program	Pulsed V_{IL} to V_{IH}	V_{IH}	25	5	D_{IH}
Program Verify	V_{IL}	V_{IL}	25	5	D_{OUT}
Program Inhibit	V_{IL}	V_{IH}	25	5	Hi-Z
Output Disable	X	V_{IH}	V_{CC}	5	Hi-Z

2. Input Rise/Fall Times $t_{in} = 1 \text{ ns}$ (Note 5)

Symbol	Parameter	Conditions	Typ	Max	Units
C_{in}	Input Capacitance	$V_{in} = 0\text{V}$	4	5	pF
C_{out}	Output Capacitance	$V_{out} = 0\text{V}$	8	12	pF

AC Test Conditions

Output Load

1 TTL Gate and
 $C_L = 100 \text{ pF}$

Timing Measurement Reference Level

Inputs
Outputs

Input Rise and Fall Times

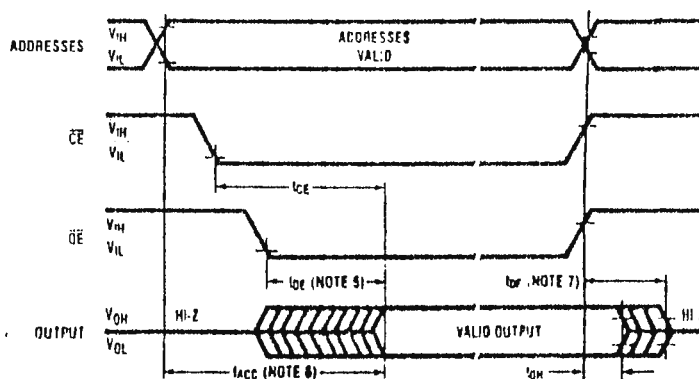
$\leq 20 \text{ ns}$

Input Pulse Levels

0.8V to 2.2V

1V and 2V
0.8V and 2V

AC Waveforms (Notes 2, 8, 9, 10)



TL/D/5275-3

Note 1: 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.

Note 2: V_{CC} must be applied simultaneously or before V_{pp} and removed simultaneously or after V_{pp} .

Note 3: V_{pp} may be connected to V_{CC} except during programming. $I_{CC1} \leq$ the sum of the I_{CC} active and I_{pp} read currents.

Note 4: Typical values are for $T_A = +25^\circ\text{C}$ and nominal supply voltages.

Note 5: This parameter is only sampled and is not 100% tested.

Note 6: \overline{OE} may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of \overline{CE} without impact on t_{ACC} .

Note 7: The t_{OE} compare level is determined as follows:

High to TRI-STATE, the measured V_{OH1} (DC) - 0.10V

Low to TRI-STATE, the measured V_{OL1} (DC) + 0.10V

Note 8: TRI-STATE may be attained using \overline{OE} or \overline{CE} .

Note 9: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND.

Note 10: The NMC27C16 requires one address transition after initial power-up to reset the outputs.

Note 11: The outputs must be restricted to $V_{CC} + 0.3\text{V}$ to avoid latch-up and device damage.



NMC27C32

32,768-Bit (4096 x 8) UV Erasable CMOS PROM

General Description

The NMC27C32 is a high speed 32k UV erasable and electrically reprogrammable CMOS EPROM. Ideally suited for applications where fast turnaround, pattern experimentation and low power consumption are important requirements.

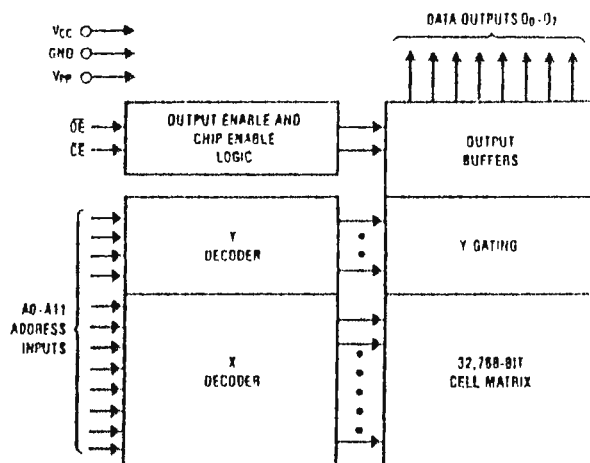
The NMC27C32 is packaged in a 24-pin dual-in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written into the device by following the programming procedure.

This EPROM is fabricated with the reliable, high volume, time proven, p²CMOSTM silicon gate technology.

Features

- Access time down to 160 ns
- Low CMOS power consumption
 - Active power: 26.25 mW max
 - Standby power: 0.53 mW max (98% savings)
- Extended temperature range available (NMC27C32E-45 and NMC27C32HE-45), -40°C to +85°C, 45°C max ±5% power supply
- 10 ms programming available (NMC27C32H), an 80% time savings
- Pin compatible to NMC2732 and higher density EPROMs
- Static-no clocks required
- TTL compatible inputs/ outputs
- Two-line control
- TRI-STATE[®] output

Block Diagram



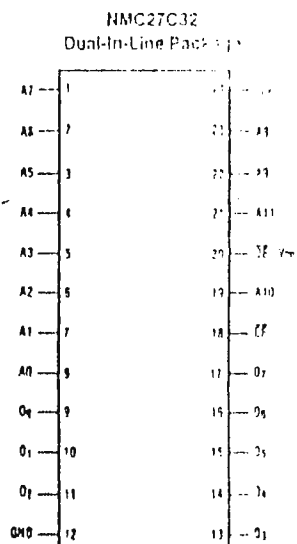
Pin Names

A ₀ -A ₁₁	Addresses
\overline{CE}	Chip Enable
\overline{OE}	Output Enable
O ₀ -O ₇	Outputs

TL/D/5274-1

Connection Diagram

27C256	27C128	27C64	27C16
27256	27128	2764	2716
V _{PP}	V _{PP}	V _{PP}	
A12	A12	A12	
A7	A7	A7	A7
A6	A6	A6	A6
A5	A5	A5	A5
A4	A4	A4	A4
A3	A3	A3	A3
A2	A2	A2	A2
A1	A1	A1	A1
A0	A0	A0	A0
O ₀	O ₀	O ₀	O ₀
O ₁	O ₁	O ₁	O ₁
O ₂	O ₂	O ₂	O ₂
GND	GND	GND	GND



27C216	27C01	27128	2716
27216	2764	27128	2716
	V _{PP}		
	PGM		
V _{PP}	PGM	A11	A11
A8	A8	A8	A8
A9	A9	A9	A9
V _{PP}	A11	A11	A11
CE	CE	CE	CE
A10	A10	A10	A10
CE	CE	CE	CE
O ₇	O ₇	O ₇	O ₇
O ₆	O ₆	O ₆	O ₆
O ₅	O ₅	O ₅	O ₅
O ₄	O ₄	O ₄	O ₄
O ₃	O ₃	O ₃	O ₃

Top View

Order Number NMC27C32
See NS Package Number J24AQ

Note: Socket compatible EPROM pin configurations are shown in the blocks adjacent to the NMC27C32 pins

Commercial Temp Range (0°C to +70°C) V_{CC} = 5V ± 5%

Parameter/Order Number	Access Time (ns)
NMC27C32-30, NMC27C32H-30	300
NMC27C32-35, NMC27C32H-35	350
NMC27C32-45, NMC27C32H-45	450
NMC27C32-55, NMC27C32H-55	550

Extended Temp Range (-40°C to +85°C) V_{CC} = 5V ± 5%

Parameter/Order Number	Access Time (ns)
NMC27C32E-45, NMC27C32EH-45	450

Capacitance (Note 5)

Symbol	Parameter	Conditions	Typ	Max	Units
C_{IN1}	Input Capacitance Except \overline{OE}/V_{PP}	$V_{IN} = 0V$	4	6	pF
C_{IN2}	\overline{OE}/V_{PP} Input Capacitance	$V_{IN} = 0V$		20	pF
C_{OUT}	Output Capacitance	$V_{OUT} = 0V$	8	12	pF

AC Test Conditions

Output Load

1 TTL Gate and
 $C_L \approx 100$ pF

Timing Measurement Reference Level

Input Rise and Fall Times

 ≤ 20 ns

Inputs

1V and 2V

Input Pulse Levels

0.45V to 2.4V

Outputs

0.8V and 2V

AC Waveforms (Notes 6 & 8)

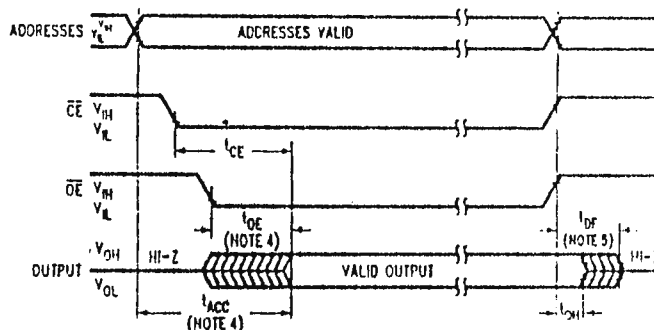


FIGURE 1

Note 1: 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.

Note 2: Typical values are for $T_A = +25^\circ\text{C}$ and nominal supply voltages.

Note 3: This parameter is only sampled and is not 100% tested.

Note 4: \overline{CE} may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of \overline{CE} without impacting t_{ACC} .

Note 5: The t_{DF} compare level is determined as follows:

High to TRI-STATE, the measured V_{OH1} (DC) $- 0.10V$;

Low to TRI-STATE, the measured V_{OL1} (DC) $+ 0.10V$.

Note 6: TRI-STATE may be attained using \overline{OE} or \overline{CE} .

Note 7: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that a $0.1 \mu\text{F}$ ceramic capacitor be used on every device between V_{CC} and GND.

Note 8: The outputs must be restricted to $V_{CC} + 0.3V$ to avoid latch-up and device damage.

Absolute Maximum Ratings

If military/airspace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Temperature under Bias	-10°C to +80°C
Storage Temperature	-65°C to +125°C
All Input Voltages with Respect to Ground	+6.5V to -0.3V
All Output Voltages with Respect to Ground	$V_{CC} + 0.3V$ to GND -0.3V
V_{PP} Supply Voltage with Respect to Ground during Programming	+26.5V to -0.3V

Operating Conditions

Temperature Range	
Commercial	NMC27C32-35, NMC27C32-55, NMC27C32H-35, NMC27C32HE-55
Industrial	NMC27C32H-55, NMC27C32HE-55
Automotive	NMC27C32E-45, NMC27C32E-55
V_{CC} Power Supply	5V ± 5%

READ OPERATION

DC Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ (Note 2)	Max	Units
I_{LI}	Input Load Current	$V_{IH} = V_{CC}$ or GND			10	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND, $\overline{CE} = V_{IH}$			10	μA
I_{CC1}	V_{CC} Current (Active) TTL Inputs	$\overline{OE} = \overline{CE} = V_{IL}$ Inputs = V_{IH} or V_{IL} , $f = 1$ MHz I/O = 0 mA		2	10	mA
I_{CC2}	V_{CC} Current (Active) CMOS Inputs	$\overline{OE} = \overline{CE} = V_{IL}$ Inputs = V_{CC} or GND, $f = 1$ MHz I/O = 0 mA		1	5	mA
I_{CCS1}	V_{CC} Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCS2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.01	0.1	mA
V_{IL}	Input Low Voltage		-0.1		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{CC} + 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.45	V
V_{OH1}	Output High Voltage	$I_{OH} = -400$ μA	2.4			V
V_{OL2}	Output Low Voltage	$I_{OL} = 0$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = 0$ μA	$V_{CC} - 0.1$			V

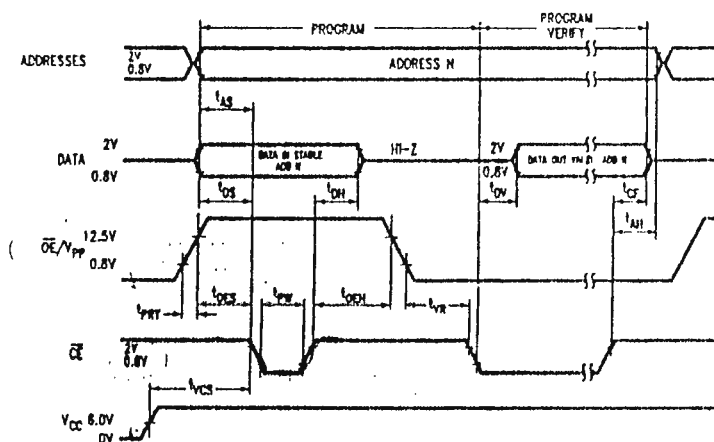
AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C32								Units
			-30, H-30		-35, H-35		-45, H-45 E-45; HE-45		-55, H-55		
			Min	Max	Min	Max	Min	Max	Min	Max	
t _{ACC}	Address to Output Delay	CE = OE = V _{IL}		300		350		450		550	ns
t _{CE}	CE to Output Delay	OE = V _{IL}		300		350		450		550	ns
t _{OE}	OE to Output Delay	CE = V _{IL}		150		150		150		150	ns
t _{PF}	OE High to Output Float	CE = V _{IL}	0	130	0	130	0	130	0	130	ns
t _{OH} (Note 3)	Output Hold from Address, CE or OE, Whichever Occurred First	OE = CE = V _{IL}	0		0		0		0		ns

Programming Characteristics (Notes 1, 2, 3 & 4)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
t_{AS}	Address Setup Time		1			μs
t_{AES}	\overline{OE} Setup Time		1			μs
t_{DS}	Data Setup Time		1			μs
t_{DSS}	V_{CC} Setup Time		1			μs
t_{AH}	Address Hold Time		0			μs
t_{DH}	Data Hold Time		1			μs
t_{CF}	Chip Enable to Output Float Delay	$\overline{OE} = V_{IL}$	0		60	ns
t_{PW}	Program Pulse Width		35	100	105	μs
t_{CEH}	\overline{OE} Hold Time		1			ns
t_{CV}	Data Valid from \overline{OE}	$\overline{OE} = V_{IL}$			250	ns
t_{PRT}	\overline{OE} Pulse Rise Time During Programming		50			ns
t_{VR}	V_{PP} Recovery Time		1			μs
I_{PP}	V_{PP} Supply Current During Programming Pulse	$\overline{OE} = V_{IL}$ $\overline{OE} = V_{CC}$			30	mA
I_{CC}	V_{CC} Supply Current				10	mA
T_A	Temperature Ambient		20	25	30	$^{\circ}C$
V_{CC}	Power Supply Voltage		6.0	6.25	6.5	V
V_{PP}	Programming Supply Voltage		12.5	12.75	13.0	V
t_{FR}	Input Rise, Fall Time		5			ns
V_{IL}	Input Low Voltage			0.0	0.45	V
V_{IH}	Input High Voltage		2.4	4.0		V
V_{IN}	Input Timing Reference Voltage		0.8	1.5	2.0	V
V_{OUT}	Output Timing Reference Voltage		0.8	1.5	2.0	V

Programming Waveforms



1L/D/8927-4

Note 1: National's standard product warranty applies only to devices programmed to specifications described herein.

Note 2: V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP} . The EPROM must not be inserted into or removed from a board with voltage applied to V_{PP} or V_{CC} .

Note 3: The maximum absolute allowable voltage which may be applied to the V_{PP} pin during programming is 14V. Care must be taken when switching the V_{PP} supply to prevent any overshoot from exceeding this 14V maximum specification. At least a 0.1 μF capacitor is required across V_{CC} to GND to suppress spurious voltage transients which may damage the device.

Note 4: Programming and program verify are tested with the fast Program Algorithm, at typical power supply voltages and timings.

FIGURE 3. AC Test Conditions (Note 2)

Symbol	Parameter	Conditions	Typ	Max	Units
C_{IN}	Input Capacitance except \overline{OE}/V_{pp}	$V_{IN} = 0V$	6	8	pF
C_{IN}	\overline{OE}/V_{pp} Input Capacitance	$V_{IN} = 0V$	25	29	pF
C_{OUT}	Output Capacitance	$V_{OUT} = 0V$	9	12	pF

AC Test Conditions

Output Load

1 TTL Gate and
 $C_L = 100$ pF (Note 8)

Timing Measurement Reference Level
Inputs

0.8V and 2.0V

Input Rise and Fall Times

≤ 5 ns

Outputs

0.8V and 2.0V

Input Pulse Levels

0.45V to 2.4V

AC Waveforms (Note 7)

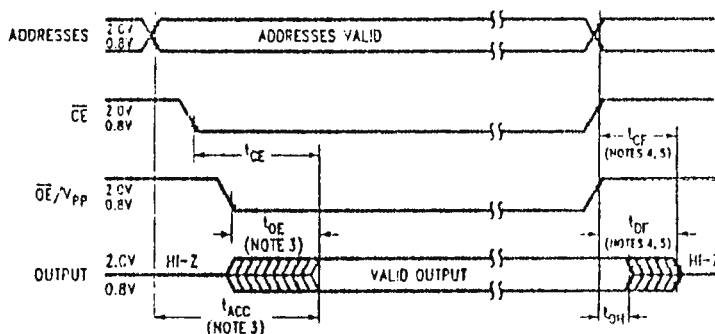


FIGURE 3

Note 1: 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.

Note 2: This parameter is only sampled and is not 100% tested.

Note 3: \overline{OE} may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of \overline{CE} without impacting t_{ACC} .

Note 4: The t_{OH} and t_{DR} compare level is determined as follows:

High to TRI-STATE, the measured V_{OH1} (DC) - 0.10V;
Low to TRI-STATE, the measured V_{OL1} (DC) + 0.10V.

Note 5: TRI-STATE may be attained using \overline{CE} or \overline{OE} .

Note 6: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that at least a 0.1 μ F ceramic capacitor be used on every device between V_{CC} and GND.

Note 7: The outputs must be restricted to $V_{CC} + 1.0V$ to avoid latch-up and device damage.

Note 8: 1 TTL Gate: $I_{OL} = 1.6$ mA, $I_{OH} = -400$ μ A.

C_L : 100 pF includes fixture capacitance.

Note 9: Inputs and outputs can undershoot to -2.0V for 20 ns Max, except for \overline{OE}/V_{pp} which cannot exceed -0.2V.

Program Verify

Program verify should be performed on the NMC27C32B in parallel with different data as it is programmed. Except for \overline{CE} , all like inputs and outputs of the parallel NMC27C32B may be common. A TTL low level program pulse applied to an NMC27C32B output with \overline{OE}/V_{pp} at 12.75V will program that NMC27C32B. A TTL high level \overline{CE} inhibits the other NMC27C32Bs from being programmed.

Program Verify

A verify should be performed on the programmed bit to determine whether they were correctly programmed. The verify is accomplished with \overline{CE}/V_{pp} and \overline{CE} at V_{IL} . Data should be verified t_{CV} after the falling edge of \overline{CE} .

MANUFACTURER'S IDENTIFICATION CODE

The NMC27C32B has a manufacturer's identification code to aid in programming. The code, shown in Table II, is two bytes wide and is stored in a ROM configuration on the chip. It identifies the manufacturer and the device type. The code for the NMC27C32B is, "8F01", where "8F" designates that it is made by National Semiconductor, and "01" designates a 32k part.

The code is accessed by applying $12.0V \pm 0.5V$ to address pin A9. Addresses A1-A8, A10-A11, \overline{CE} , and \overline{OE} are held at V_{IL} . Address A0 is held at V_{IL} for the manufacturer's code, and at V_{IH} for the device code. The code is read out on the 8 data pins. Proper code access is only guaranteed at $25^\circ C \pm 5^\circ C$.

The primary purpose of the manufacturer's identification code is automatic programming control. When the device is inserted in an EPROM programmer socket, the programmer reads the code and then automatically calls up the specific programming algorithm for the part. This automatic programming control is only possible with programmers which have the capability of reading the code.

ERASURE CHARACTERISTICS

The erasure characteristics of the NMC27C32B are such that erasure begins to occur when exposed to light with wavelengths shorter than approximately 4000 Angstroms (\AA). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000 \AA -4000 \AA range. After programming, opaque labels should be placed over the NMC27C32B's window to prevent unintentional

erasure. Erasures are due to the generation of photo currents.

The recommended erasure procedure for the NMC27C32B is exposure to short wave length light which has a wavelength of 2537 \AA . The integrated dose (i.e., UV intensity \times exposure time) for erasure should be a minimum of $15 W \text{ sec/cm}^2$.

The NMC27C32B should be stored within 1 inch of the lamp tubes during erasure. Some lamps have a filter on their tubes which should be removed before erasure. Table III shows the minimum NMC27C32B erasure time for various light intensities.

An erasure system should be calibrated periodically. The distance from lamp to unit should be maintained at one inch. The erasure time increases as the square of the distance. (If distance is doubled the erasure time increases by a factor of 4.) Lamps lose intensity as they age. When a lamp is changed, the distance has changed or the lamp has aged, the system should be checked to make certain full erasure is occurring. Incomplete erasure will cause symptoms that can be misleading. Programmers, components, and even system designs have been erroneously suspected when incomplete erasure was the problem.

SYSTEM CONSIDERATION

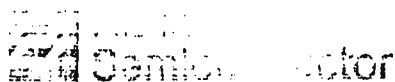
The power switching characteristics of EPROMs require careful decoupling of the devices. The supply current, I_{CC} , has three segments that are of interest to the system designer—the standby current level, the active current level, and the transient current peaks that are produced by voltage transitions on input pins. The magnitude of these transient current peaks is dependent on the output capacitance loading of the device. The associated V_{CC} transient voltage peaks can be suppressed by properly selected decoupling capacitors. It is recommended that at least a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND. This should be a high frequency capacitor of low inherent inductance. In addition, at least a 4.7 μF bulk electrolytic capacitor should be used between V_{CC} and GND for each eight devices. The bulk capacitor should be located near where the power supply is connected to the array. The purpose of the bulk capacitor is to overcome the voltage drop caused by the inductive effects of the PC board traces.

TABLE II. Manufacturer's Identification Code

Pins	A0 (8)	O7 (17)	O6 (16)	O5 (15)	O4 (14)	O3 (13)	O2 (11)	O1 (10)	O0 (9)	Hex Data
Manufacturer Code	V_{IL}	1	0	0	0	1	1	1	1	8F
Device Code	V_{IH}	0	0	0	0	0	0	0	1	01

TABLE III. Minimum NMC27C32B Erasure Time

Light Intensity ($\mu W/\text{cm}^2$)	Erasure Time (Minutes)
15,000	20
10,000	25
5,000	50



NMC27C64 65,536-Bit (8k x 8) UV Erasable CMOS PROM

General Description

The NMC27C64 is a high-performance UV erasable and electrically reprogrammable CMOS EPROM, ideally suited for applications where fast turn-around, pattern experimentation and low power consumption are important requirements.

The NMC27C64 is designed to operate with a single +5V power supply with $\pm 5\%$ or $\pm 10\%$ tolerance. The CMOS design allows the part to operate over extended and military temperature ranges.

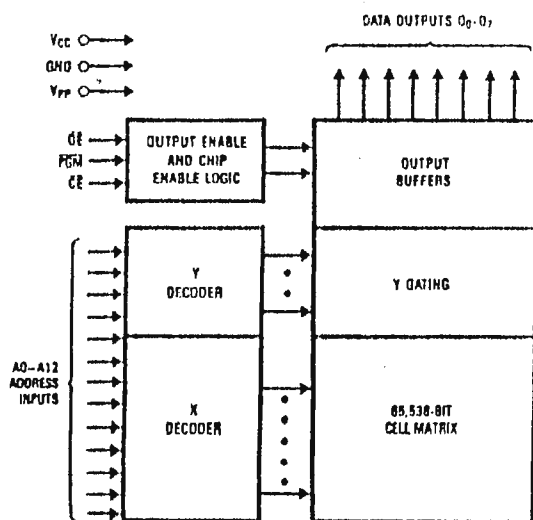
The NMC27C64 is packaged in a 28-pin dual-in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written electrically into the device by following the programming procedure.

This EPROM is fabricated with National's proprietary, time proven CMOS double-poly silicon gate technology which combines high performance and high density with low power consumption and excellent reliability.

Features

- Clock-free device trips for fast access times down to 150 ns
- Low CMOS power consumption
 - Active Power: 55 mW max
 - Standby Power: 0.55 mW max
- Performance compatible to NSC2001™ CMOS micro-processor
- Single 5V power supply
- Extended temperature range (NMC27C64CE), -10°C to $+85^{\circ}\text{C}$, and military temperature range (NMC27C64QM), -55°C to $+125^{\circ}\text{C}$, available
- Pin compatible with NMOS 64k EPROMs
- Fast and reliable programming
- Static operation—no clocks required
- TTL/CMOS compatible inputs/outputs
- TRI-STATE* output
- Optimum EPROM for total CMOS systems
- Manufacturer's identification code for automatic programming control

Block Diagram

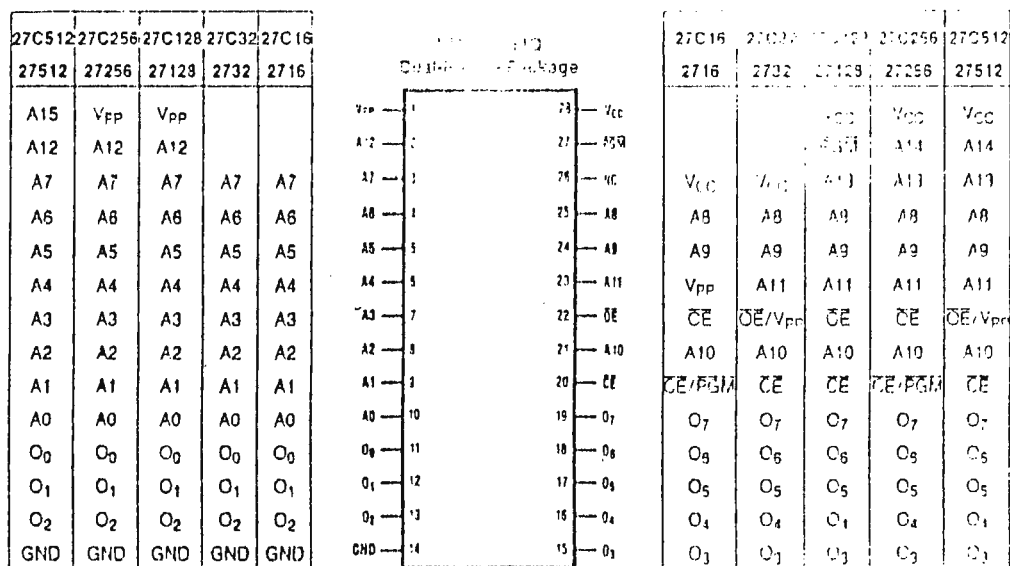


TL/D/0834-1

Pin Names

A0-A12	Addresses
CE	Chip Enable
OE	Output Enable
O ₀ -O ₇	Outputs
PGM	Program
NC	No Connect

Connection Diagram



TL/D/4514-2

Note: Socket compatible EPRCM pin configurations are shown in the blocks adjacent to the NMC27C64 pins.

Order Number NMC27C64Q
See NS Package Number J28AQ

Commercial Temp Range (0°C to +70°C)
V_{CC} = 5V ± 5%

Parameter/Order Number	Access Time (ns)
NMC27C64Q15	150

V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time (ns)
NMC27C64Q150	150
NMC27C64Q200	200
NMC27C64Q250	250
NMC27C64Q300	300

Extended Temp Range (-40°C to +85°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time (ns)
NMC27C64QE150	150
NMC27C64QE200	200

Military Temp Range (-55°C to +125°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time (ns)
NMC27C64QM200	200
NMC27C64QM250	250

NOTE: For plastic DIP requirements please refer to NMC27C64N data sheet.

Capacitance $C_A = 100 \text{ pF}$

Symbole	Parameter	Conditions	Typ	Max
C_{IN}	Input Capacitance	$V_{IN} = 0V$	5	10
C_{OUT}	Output Capacitance	$V_{OUT} = 0V$	3	10

AC Test Conditions

Output Load

1 TTL Gate and
 $C_L = 100 \text{ pF}$ (Note 8)

Measurement Reference Level

0.8V and 2V

Input Rise and Fall Times

 $\leq 5 \text{ ns}$

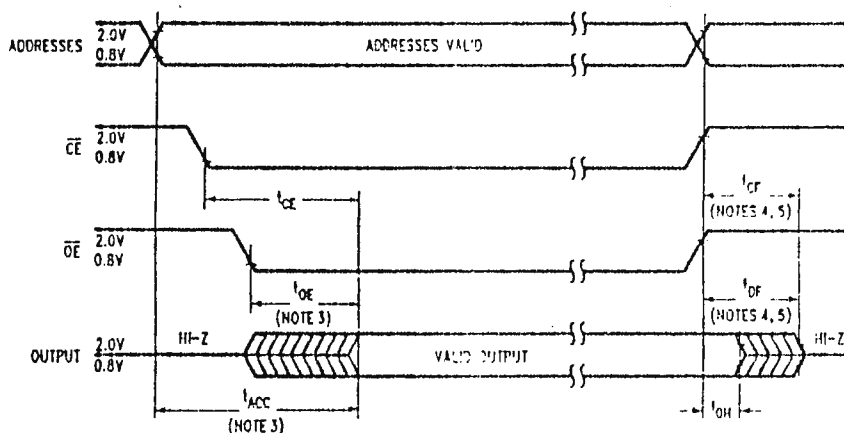
Inputs

0.8V and 2V

Input Pulse Levels

0.45V to 2.4V

AC Waveforms (Notes 6 & 9)



TLC6434-1

Note 1: 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 those 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.

Note 2: This parameter is only sampled and is not 100% tested.

Note 3: CE may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of CE without impacting t_{ACC} .

Note 4: The t_{OE} and t_{CE} compare level is determined as follows:

High to TRI-STATE, the measured V_{OH1} (DC) - 0.10V;

Low to TRI-STATE, the measured V_{OL1} (DC) + 0.10V.

Note 5: TRI-STATE may be attained using OE or CE.

Note 6: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that at least a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND.

Note 7: The outputs must be restricted to $V_{CC} + 1.0V$ to avoid latch-up and device damage.

Note 8: 1 TTL Gate: $I_{OL} = 1.6 \text{ mA}$, $I_{OH} = -400 \mu\text{A}$.

C_L : 100 pF includes fixture capacitance.

Note 9: V_{PP} may be connected to V_{CC} except during programming.

Note 10: Inputs and outputs can undershoot to -2.0V for 20 ns Max.

5 2.0
3 1.0

Absolute Maximums

If Military/Aerospace applications are required, please contact the nearest Analog Products Sales Office/Distributors for additional specifications.

Temperature Under Bias	
Commercial	-40°C to +80°C
Military and Extended	Operating Temp. Range
Storage Temperature	-65°C to +150°C

All Input Voltages except \overline{CE} and \overline{OE} with Respect to Ground (Note 10) $+5.5V$ to $-0.6V$

All Output Voltages with Respect to Ground (Note 10) $+1.0V$ to $GND - 0.6V$

V_{PP} Supply Voltage and A3 with Respect to Ground During Programming $+14.0V$ to $-0.6V$

Recommended Operating Conditions

VCC to GND	$+5.0V$ to $+0.6V$
Power Dissipation	1.0W
Lead Temperature (Soldering, 10 sec.)	300°C
ESD Rating (MIL-STD-883C Method 3015.2)	2000V

Operating Conditions (Note 1)

Temperature Range	
NMC27C64Q15, Q150, 200, 250	-40°C to +70°C
NMC27C64QE150, 200	-40°C to +85°C
NMC27C64QM200, M250	-40°C to +125°C
VCC Power Supply	$\pm 5V \pm 10\%$
except NMC27C64Q15	$\pm 5V \pm 5\%$

READ OPERATION**DC Electrical Characteristics**

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Load Current	$V_{IH} = V_{CC}$ or GND			1	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND, $\overline{CE} = V_{IH}$			1	μA
I_{CC1} (Note 9)	VCC Current (Active) TTL Inputs	$\overline{CE} = V_{IL}$, $f = 5$ MHz Inputs = V_{IH} or V_{IL} , I/O = 0 mA		5	20	mA
I_{CC2} (Note 9)	VCC Current (Active) CMOS Inputs	$\overline{CE} = GND$, $f = 5$ MHz Inputs = V_{CC} or GND, I/O = 0 mA		3	10	mA
I_{CCS1}	VCC Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCS2}	VCC Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.5	100	μA
I_{PP}	V_{PP} Load Current	$V_{PP} = V_{CC}$			10	μA
V_{IL}	Input Low Voltage		-0.1		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{CC} + 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.45	V
V_{OH1}	Output High Voltage	$I_{OH} = -400$ μA	2.4			V
V_{OL2}	Output Low Voltage	$I_{OL} = 0$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = 0$ μA	$V_{CC} - 0.1$			V

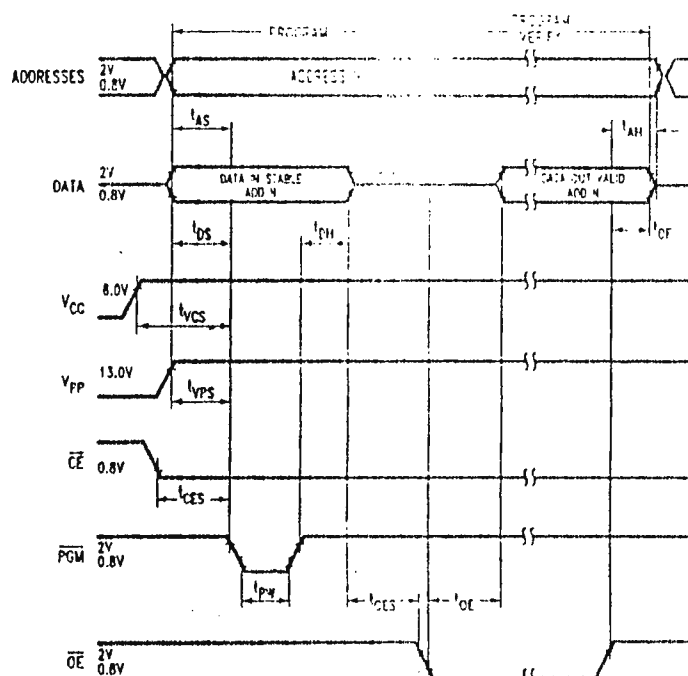
AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C64Q						Units
			15, 150, E150		200, E200, M200		250, M250		
			Min	Max	Min	Max	Min	Max	
t _{ACO}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$ $PGM = V_{IH}$		180		200		250	ns
t _{CE}	\overline{CE} to Output Delay	$\overline{OE} = V_{IL}, PGM = V_{IH}$		150		200		250	ns
t _{OE}	\overline{OE} to Output Delay	$\overline{CE} = V_{IL}, PGM = V_{IH}$		60		60		70	ns
t _{OF}	\overline{OE} High to Output Float	$\overline{CE} = V_{IL}, PGM = V_{IH}$	0	60	0	60	0	60	ns
t _{CF}	\overline{CE} High to Output Float	$\overline{OE} = V_{IL}, PGM = V_{IH}$	0	60	0	60	0	60	ns
t _{OH}	Output Hold from Addresses, \overline{CE} or \overline{OE} , Whichever Occurred First	$\overline{CE} = \overline{OE} = V_{IL}$ $PGM = V_{IH}$	0		0		0		ns

Programming Conditions

Symbol	Param	Conditions	Min	Typ	Max	Units
t_{AS}	Address Setup Time		2			μs
t_{OES}	OE Setup Time		2			μs
t_{CES}	CE Setup Time		2			μs
t_{DS}	Data Setup Time		2			μs
t_{VPS}	V_{PP} Setup Time		2			μs
t_{VCS}	V_{CC} Setup Time		2			μs
t_{AH}	Address Hold Time		0			μs
t_{DH}	Data Hold Time		2			μs
t_{DF}	Output Enable to Output Float Delay	$\overline{CE} = V_{IL}$	0		130	ns
t_{PW}	Program Pulse Width		0.45	0.5	0.55	ms
t_{CE}	Data Valid from \overline{CE}	$\overline{CE} = V_{IL}$			150	ns
I_{PP}	V_{PP} Supply Current During Programming Pulse	$\overline{CE} = V_{IL}$ $PGM = V_{IL}$			30	mA
I_{CC}	V_{CC} Supply Current				10	mA
T_A	Temperature Ambient		20	25	30	$^{\circ}C$
V_{CC}	Power Supply Voltage		5.75	6.0	6.25	V
V_{PP}	Programming Supply Voltage		12.2	13.0	13.3	V
t_{FR}	Input Rise, Fall Time		5			ns
V_{IL}	Input Low Voltage			0.0	0.45	V
V_{IH}	Input High Voltage		2.4	4.0		V
I_{IN}	Input Timing Reference Voltage		0.8	1.5	2.0	V
I_{OUT}	Output Timing Reference Voltage		0.8	1.5	2.0	V

Programming Waveforms



TL04614-5

Note 1: National's standard product warranty applies to devices programmed to specifications described herein.

Note 2: V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP}. The EPROM must not be inserted into or removed from a board with voltage applied to V_{PP} or V_{CC}.

Note 3: The maximum absolute allowable voltage which may be applied to the V_{PP} pin during programming is 14V. Care must be taken when switching the V_{PP} supply to prevent any overshoot from exceeding this 14V maximum specification. At least a 0.1 μ F capacitor is required across V_{PP}, V_{CC} to GND to suppress spurious voltage transients which may damage the device.

Note 4: Programming and program verify are tested with the Interactive Program Algorithm, at typical power supply voltages and timings.

Functional Description

CE OPERATION

The six modes of operation of the NMC27C64 are listed in Table I. It should be noted that all inputs for the six modes are at TTL levels. The power supplies required are V_{CC} and V_{PP} . The V_{PP} power supply must be at 13.0V during the three programming modes, and must be at 5V in the other three modes. The V_{CC} power supply must be at 6V during the three programming modes, and at 5V in the other three modes.

Read Mode

The NMC27C64 has two control functions, both of which must be logically active in order to obtain data at the outputs. Chip Enable (\overline{CE}) is the power control and should be used for device selection. Output Enable (\overline{OE}) is the output control and should be used to gate data to the output pins, independent of device selection. The programming pin (\overline{PGM}) should be at V_{IH} except during programming. Assuming that addresses are stable, address access time (t_{ACC}) is equal to the delay from \overline{CE} to output (t_{CE}). Data is available at the outputs t_{OE} after the falling edge of \overline{OE} , assuming that \overline{CE} has been low and addresses have been stable for at least $t_{ACC} - t_{OE}$.

The sense amps are clocked for fast access time. V_{CC} should therefore be maintained at operating voltage during read and verify. If V_{CC} temporarily drops below the spec. voltage (but not to ground) an address transition must be performed after the drop to insure proper output data.

Standby Mode

The NMC27C64 has a standby mode which reduces the active power dissipation by 99%, from 55 mW to 0.55 mW. The NMC27C64 is placed in the standby mode by applying a CMOS high signal to the \overline{CE} input. When in standby mode, the outputs are in a high impedance state, independent of the \overline{OE} input.

Output OR-Tying

Because NMC27C64s are usually used in larger memory arrays, National has provided a 2-line control function that accommodates this use of multiple memory connections. The 2-line control function allows for:

- the lowest possible memory power dissipation, and
- complete assurance that output bus contention will not occur.

To most efficiently use the 2-line control function, it is recommended that \overline{CE} (pin 20) be used as the primary device selecting function. Pin 20 should be made a common connection to all devices in the array and connected to the READ line from the memory controller. This assures that all deselected memory devices are in their low power standby modes and that the outputs are active only when data is desired from a selected memory device.

Programming

CAUTION: Exceeding 14V on pin 1 (V_{PP}) will damage the NMC27C64.

Initially, all bits of the NMC27C64 are in the "1" state. Data is introduced by selectively programming "0s" into the desired bit locations. Although only "0s" will be programmed, both "1s" and "0s" can be presented in the data word. A "0" cannot be changed to a "1" once the bit has been programmed.

The NMC27C64 is in the programming mode when the V_{PP} power supply is at 13.0V and \overline{OE} is at V_{IH} . It is required that at least a 0.1 μ F capacitor be placed across V_{PP} , V_{CC} to ground to suppress spurious voltage transients which may damage the device. The data to be programmed is applied 8 bits in parallel to the data output pins. The levels required for the address and data inputs are TTL.

For programming, \overline{CE} should be kept TTL low at all times while V_{PP} is kept at 13.0V.

When the address and data are stable, an active low, TTL program pulse is applied to the \overline{PGM} input. A program pulse must be applied at each address location to be programmed. The NMC27C64 is designed to be programmed with interactive programming, where each address is programmed with a series of 0.5 ms pulses until it verifies (up to a maximum of 20 pulses or 10 ms). The NMC27C64 must not be programmed with a DC signal applied to the \overline{PGM} input.

Programming multiple NMC27C64s in parallel with the same data can be easily accomplished due to the simplicity of the programming requirements. Like inputs of the paralleled NMC27C64s may be connected together when they are programmed with the same data. A low level TTL pulse applied to the \overline{PGM} input programs the paralleled NMC27C64s.

TABLE I. Mode Selection

Mode	Pins \overline{CE} (20)	\overline{OE} (22)	\overline{PGM} (27)	V_{PP} (1)	V_{CC} (28)	Outputs (11-13, 15-19)
Read	V_{IL}	V_{IL}	V_{IH}	5V	5V	D_{OUT}
Standby	V_{IH}	Don't Care	Don't Care	5V	5V	Hi-Z
Output Disable	Don't Care	V_{IH}	V_{IH}	5V	5V	Hi-Z
Program	V_{IL}	V_{IH}	L	13V	6V	D_{IN}
Program Verify	V_{IL}	V_{IL}	V_{IH}	13V	6V	D_{OUT}
Program Inhibit	V_{IH}	Don't Care	Don't Care	13V	6V	Hi-Z



PRELIMINARY

NMC27C128B High Speed Version 131,072-Bit (16k x 8) UV Erasable CMOS PROM

General Description

The NMC27C128B is a high-speed 128k UV erasable and electrically reprogrammable CMOS EPROM, ideally suited for applications where fast turnaround, pattern experimentation and low power consumption are important requirements.

The NMC27C128B is designed to operate with a single +5V power supply with $\pm 5\%$ or $\pm 10\%$ tolerance. The CMOS design allows the part to operate over extended and military temperature ranges.

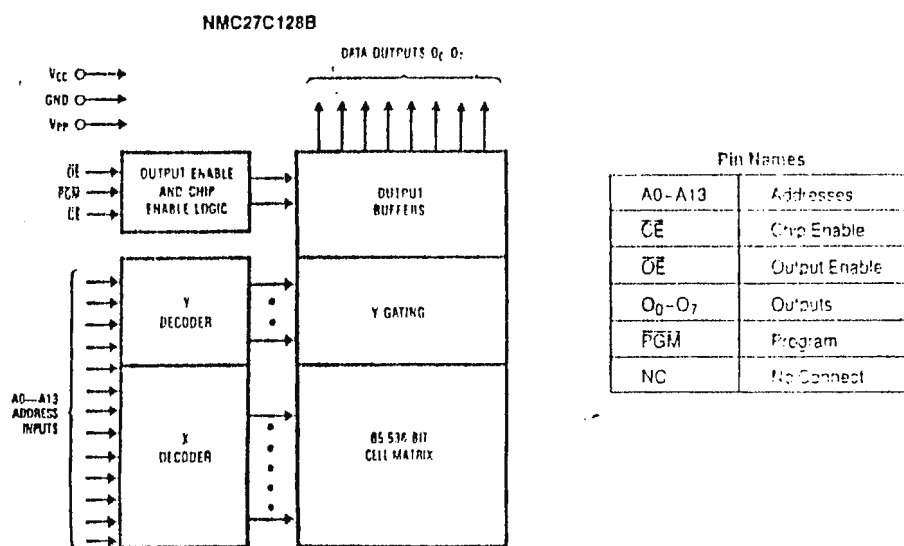
The NMC27C128B is packaged in a 28-pin dual-in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written electrically into the device by following the programming procedure.

This EPROM is fabricated with National's proprietary, time proven CMOS double-poly silicon gate technology which combines high performance and high density with low power consumption and excellent reliability.

Features

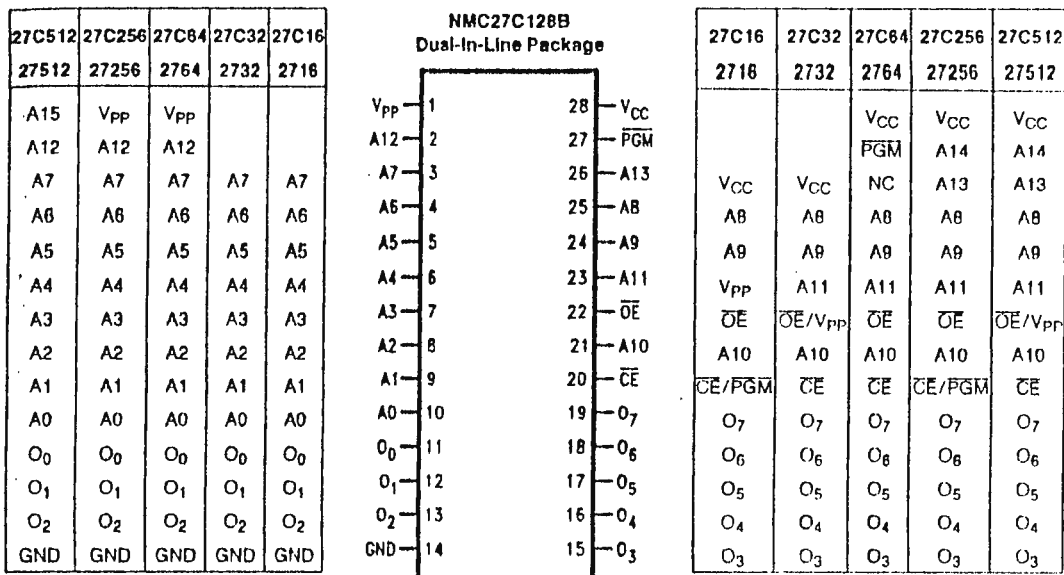
- Clock sense amps for fast access time down to 150 ns
- Low CMOS power consumption
 - Active Power: 110 mW max
 - Standby Power: 0.55 mW max
- Extended temperature range (NMC27C128BQE), -40°C to $+85^{\circ}\text{C}$, and military temperature range (NMC27C128BQM), -55°C to $+125^{\circ}\text{C}$ available
- Pin compatible with NMOS 128k EPROMs
- Fast and reliable programming—100 μs typical/byte
- Static operation—no clocks required
- TTL, CMOS compatible inputs/outputs
- TRI-STATE* output
- Optimum EPROM for total CMOS systems
- Manufacturer's identification code for automatic programming control
- High current CMOS level output drivers

Block Diagram



TEL D1669-1

Connection Diagrams



TL/D/P089-2

Note: Socket compatible EPROM pin configurations are shown in the blocks adjacent to the NMC27C128B plus.

Order Number NMC27C128BQ
See NS Package Number J28AQ

Commercial Temp Range (0°C to +70°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time (ns)
NMC27C128BQ150	150
NMC27C128BQ200	200
NMC27C128BQ250	250

Commercial Temp Range (0°C to +70°C)
V_{CC} = 5V ± 5%

Parameter/Order Number	Access Time (ns)
NMC27C128BQ15	150
NMC27C128BQ20	200
NMC27C128BQ25	250

Extended Temp Range (-40°C to +85°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time (ns)
NMC27C128BQE150	150
NMC27C128BQE200	200

Military Temp Range (-55°C to +125°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time (ns)
NMC27C128BQM150	150
NMC27C128BQM200	200

NOTE: For plastic DIP and surface mount PLCC package requirements please refer to NMC27C128BN datasheet.

MILITARY AND EXTENDED TEMPERATURE RANGE

Absolute Maximum Ratings (Note 1)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Temperature Under Bias Operating Temp. Range
Storage Temperature - 65°C to + 150°C

All Input Voltages except A9 with
Respect to Ground (Note 10) + 6.5V to - 0.6V

All Output Voltages with
Respect to Ground (Note 10) $V_{CC} + 1.0V$ to GND - 0.6V

V_{PP} Supply Voltage and A9
with Respect to Ground + 14.0V to - 0.6V
During Programming

V_{CC} Supply Voltage with
Respect to Ground + 7.0V to - 0.6V

Power Dissipation 1.0W

Lead Temperature (Soldering, 10 sec.) 300°C

ESD Rating
(Mil Spec 883C, Method 3015.2) 2000V

Operating Conditions (Note 7)

Temperature Range
NMC27C128BQE150, 200 - 40°C to + 85°C
NMC27C128BOM150, 200 - 55°C to + 125°C

V_{CC} Power Supply + 5V \pm 10%

READ OPERATION

DC Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Load Current	$V_{IH} = V_{CC}$ or GND			10	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND, $\overline{CE} = V_{IH}$			10	μA
I_{CC1} (Note 8)	V_{CC} Current (Active) TTL Inputs	$\overline{CE} = V_{IL}$, $f = 5$ MHz Inputs = V_{IH} or V_{IL} , $I/O = 0$ mA		10	30	mA
I_{CC2} (Note 9)	V_{CC} Current (Active) CMOS Inputs	$\overline{CE} =$ GND, $f = 5$ MHz Inputs = V_{CC} or GND, $I/O = 0$ mA		8	20	mA
I_{CCSB1}	V_{CC} Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCSB2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.5	100	μA
I_{PP}	V_{PP} Load Current	$V_{PP} = V_{CC}$			10	μA
V_{IL}	Input Low Voltage		0.2		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{CC} - 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.40	V
V_{OH1}	Output High Voltage	$I_{OH} = -1.6$ mA	3.5			V
V_{OL2}	Output Low Voltage	$I_{OL} = 10$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = -10$ μA	$V_{CC} - 0.1$			V

AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C128BQ				Units
			E150, M150		E200, M200		
			Min	Max	Min	Max	
t_{ACC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$ PGM = V_{IH}		150		200	ns
t_{CE}	\overline{CE} to Output Delay	$\overline{OE} = V_{IL}$, PGM = V_{IH}		150		200	ns
t_{OE}	\overline{OE} to Output Delay	$\overline{CE} = V_{IL}$, PGM = V_{IH}		60		75	ns
t_{CF}	\overline{CE} High to Output Float	$\overline{OE} = V_{IL}$, PGM = V_{IH}	0	50	0	55	ns
t_{OF}	\overline{OE} High to Output Float	$\overline{CE} = V_{IL}$, PGM = V_{IH}	0	50	0	55	ns
t_{OH}	Output Hold from Addresses, \overline{CE} or \overline{OE} , Whichever Occurred First	$\overline{CE} = \overline{OE} = V_{IL}$ PGM = V_{IH}			0		ns

COMMERCIAL TEMPERATURE RANGE

Absolute Maximum Ratings (Note 1)

Temperature Under Bias	-10°C to +80°C
Storage Temperature	-65°C to +150°C
All Input Voltages except A9 with Respect to Ground (Note 10)	+6.5V to -0.6V
All Output Voltages with Respect to Ground (Note 10)	$V_{CC} + 1.0V$ to $GND - 0.6V$
V_{PP} Supply Voltage and A9 with Respect to Ground During Programming	+14.0V to -0.6V
V_{CC} Supply Voltage with Respect to Ground	+7.0V to -0.6V

Power Dissipation	1.0W
Lead Temperature (Soldering, 10 sec.)	300°C
ESD Rating (Mil Spec 883C, Method 3015.2)	2000V

Operating Conditions (Note 7)

Temperature Range	0°C to +70°C
V_{CC} Power Supply	
NMC27C128BQ150, 200, 250	+5V ± 10%
NMC27C128BQ15, 20, 25	±5V ± 5%

READ OPERATION

DC Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Load Current	$V_{IH} = V_{CC}$ or GND		0.01	1	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND, $\overline{CE} = V_{IH}$		0.01	1	μA
I_{CC1} (Note 9)	V_{CC} Current (Active) TTL Inputs	$\overline{CE} = V_{IL}$, $f = 5$ MHz Inputs = V_{IH} or V_{IL} , I/O = 0 mA		10	30	mA
I_{CC2} (Note 9)	V_{CC} Current (Active) CMOS Inputs	$\overline{CE} = GND$, $f = 5$ MHz Inputs = V_{CC} or GND, I/O = 0 mA		8	20	mA
I_{CCSB1}	V_{CC} Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCSB2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.5	100	μA
I_{PP}	V_{PP} Load Current	$V_{PP} = V_{CC}$			10	μA
V_{IL}	Input Low Voltage		-0.2		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{CC} + 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.40	V
V_{OH1}	Output High Voltage	$I_{OH} = -2.5$ mA	3.5			V
V_{OL2}	Output Low Voltage	$I_{OL} = 10$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = -10$ μA	$V_{CC} - 0.1$			V

AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C128B						Units
			Q15, Q150		Q20, Q200		Q25, Q250		
			Min	Max	Min	Max	Min	Max	
t _{ACC}	Address to Output Delay	CE = OE = V _{IL} PGM = V _{IH}		150		200		250	ns
t _{CE}	CE to Output Delay	OE = V _{IL} , PGM = V _{IH}		150		200		250	ns
t _{OE}	OE to Output Delay	CE = V _{IL} , PGM = V _{IH}		60		75		100	ns
t _{DF}	OE High to Output Float	CE = V _{IL} , PGM = V _{IH}	0	50	0	55	0	60	ns
t _{CF}	CE High to Output Float	OE = V _{IL} , PGM = V _{IH}	0	50	0	55	0	60	ns
t _{OH}	Output Hold from Addresses, CE or OE whichever Generated First	CE = OE = V _{IL} PGM = V _{IL}	0		0		0		ns

Capacitance $T_A = +25^\circ\text{C}$, $f = 1\text{ MHz}$ (Note 2)

Symbol	Parameter	Conditions	Typ	Max	Units
C_{IN}	Input Capacitance	$V_{IN} = 0\text{V}$	6	12	pF
C_{OUT}	Output Capacitance	$V_{OUT} = 0\text{V}$	9	12	pF

AC Test Conditions

Output Load

1 TTL Gate and
 $C_L = 100\text{ pF}$ (Note 8)

Timing Measurement Reference Level

Input Rise and Fall Times

 $\leq 5\text{ ns}$

Inputs

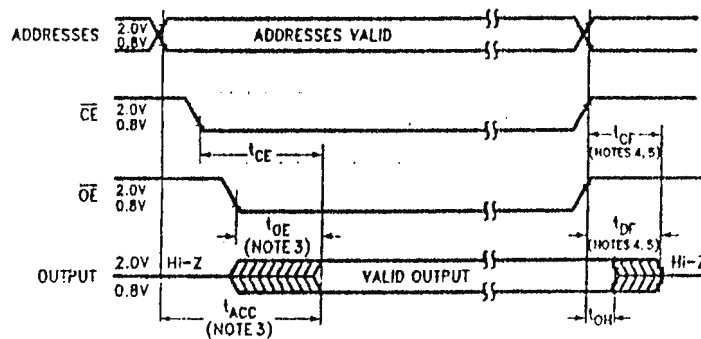
Input Pulse Levels

0.45V to 2.4V

Outputs

0.8V/1.6V

0.8V/1.6V

AC Waveforms (Notes 6, 7 & 9)

Note 1: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and 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 maximum rating conditions for extended periods may affect device reliability.

Note 2: This parameter is only sampled and is not 100% tested.

Note 3: OE may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of CE without impacting t_{ACC} .

Note 4: The t_{DF} and t_{CF} compare level is determined as follows:

High to TRI-STATE, the measured $V_{OH1}(\text{DC}) = 0.10\text{V}$

Low to TRI-STATE, the measured $V_{OL1}(\text{DC}) = 0.10\text{V}$

Note 5: TRI-STATE may be attained using OE or CE.

Note 6: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that at least a 0.1 μF ceramic capacitor be placed every device between V_{CC} and GND.

Note 7: The outputs must be restricted to $V_{CC} - 1.0\text{V}$ to avoid latch-up and device damage.

Note 8: 1 TTL Gate: $I_{OL} = 1.6\text{ mA}$, $I_{OH} = -400\text{ }\mu\text{A}$.

C_L : 100 pF includes fixture capacitance

Note 9: V_{PP} may be connected to V_{CC} except during programming.

Note 10: Inputs and outputs can undershoot to -2.0V for 20 ns Max.



NMC27C256

262,144-Bit (32k x 8) UV Erasable CMOS PROM

General Description

The NMC27C256 is a high-speed 256k UV erasable and electrically reprogrammable CMOS EPROM, ideally suited for applications where fast turnaround, pattern experimentation and low power consumption are important requirements.

The NMC27C256 is designed to operate with a single +5V power supply with $\pm 5\%$ or $\pm 10\%$ tolerance. The CMOS design allows the part to operate over extended and military temperature ranges.

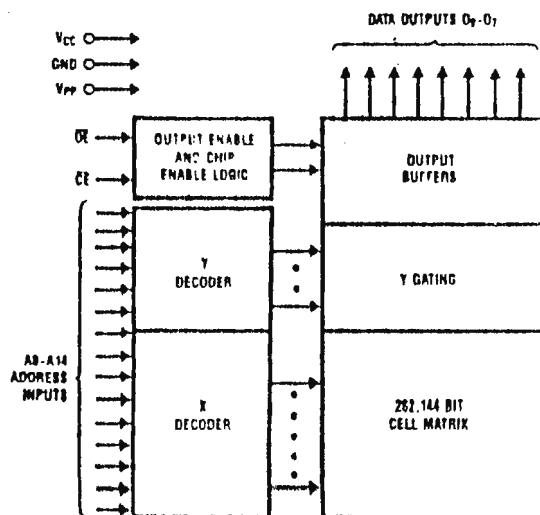
The NMC27C256 is packaged in a 28-pin dual in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written electrically into the device by following the programming procedure.

This EPROM is fabricated with National's proprietary, time proven CMOS double-poly silicon gate technology which combines high performance and high density with low power consumption and excellent reliability.

Features

- Clocked sense amps for fast access time down to 170 ns
- Low CMOS power consumption
 - Active power: 55 mW max
 - Standby power: 0.55 mW max
- Performance compatible to NSC800™ CMOS micro-processor
- Single 5V power supply
- Extended temperature range (NMC27C256OE), -40°C to $+85^{\circ}\text{C}$, and military temperature range (NMC27C256QM), -55°C to $+125^{\circ}\text{C}$, available
- Pin compatible with NMOS 256k EPROMs
- Fast and reliable programming (0.5 ms for most bytes)
- Static operation—no clocks required
- TTL, CMOS compatible inputs/outputs
- TRI-STATE® output
- Optimum EPROM for total CMOS systems

Block Diagram

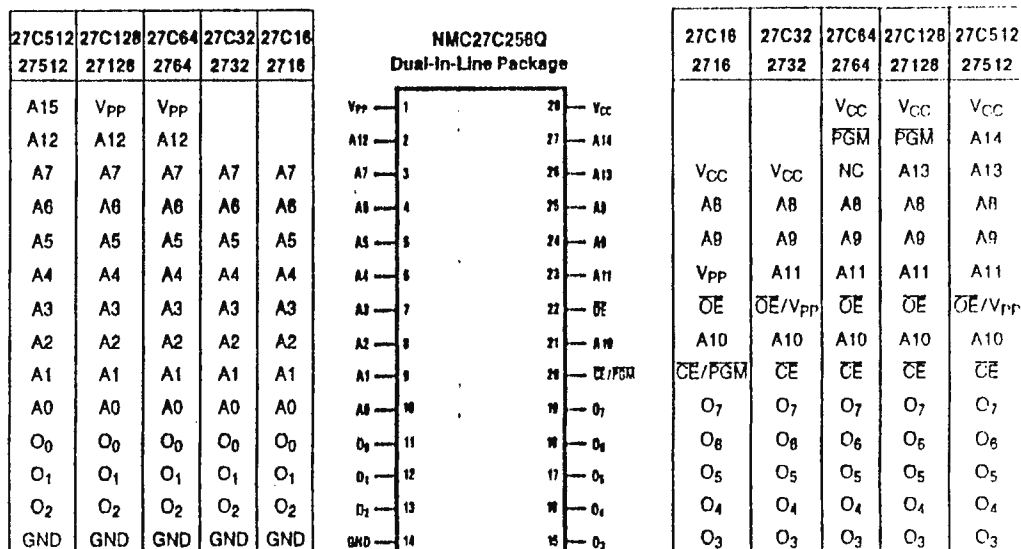


TL D7712-1

Pin Names

A0-A14	Addresses
CE	Chip Enable
OE	Output Enable
O ₀ -O ₇	Outputs
PGM	Program
NC	No Connect

Connection Diagram



TL/D/7512-2

Note: Socket compatible EPROM pin configurations are shown in the blocks adjacent to the NMC27C258Q part.

Order Number NMC27C258Q
See NS Package Number J28AQ

Commercial Temp Range (0°C to +70°C)
V_{CC} = 5V ± 5%

Parameter/Order Number	Access Time
NMC27C256Q17	170
NMC27C256Q20	200
NMC27C256Q25	250

Commercial Temp Range (0°C to +70°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time
NMC27C256Q200	200
NMC27C256Q250	250
NMC27C256Q300	300

Extended Temp Range (-40°C to +85°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time
NMC27C256QE200	200
NMC27C256QE250	250

Military Temp Range (-55°C to +125°C)
V_{CC} = 5V ± 10%

Parameter/Order Number	Access Time
NMC27C256QM250	250
NMC27C256QM350	350

NOTE: For plastic DIP and surface mount PLCC package requirements please refer to NMC27C256BH data sheet.

COMMERCIAL TEMPERATURE RANGE

Absolute Maximum Ratings (Note 1)

Temperature Under Bias	-10°C to +80°C
Storage Temperature	-65°C to +150°C
All Input Voltages with Respect to Ground (Note 10)	+6.5V to -0.6V
All Output Voltages with Respect to Ground (Note 10)	$V_{CC} + 1.0V$ to $GND - 0.6V$
V_{PP} Supply Voltage with Respect to Ground During Programming	+14.0V to -0.6V

Power Dissipation	1.0W
Lead Temperature (Soldering, 10 sec.)	300°C
V_{CC} Supply Voltage with Respect to Ground	+7.0V to -0.6V

Operating Conditions (Note 7)

Temperature Range	0°C to +70°C
V_{CC} Power Supply	
NMC27C256Q17, 20, 25	5V \pm 5%
NMC27C256Q200, 250, 300	5V \pm 10%

READ OPERATION

DC Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Load Current	$V_{IN} = V_{CC}$ or GND			10	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND , $\overline{CE} = V_{IH}$			10	μA
I_{CC1} (Note 9)	V_{CC} Current (Active) TTL Inputs	$\overline{CE} = V_{IL}$, $f = 5$ MHz Inputs = V_{IH} or V_{IL} , $I/O = 0$ mA		6	20	mA
I_{CC2} (Note 9)	V_{CC} Current (Active) CMOS Inputs	$\overline{CE} = GND$, $f = 5$ MHz Inputs = V_{CC} or GND , $I/O = 0$ mA		3	10	mA
I_{CCSB1}	V_{CC} Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCSB2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.5	100	μA
I_{PP}	V_{PP} Load Current	$V_{PP} = V_{CC}$			10	μA
V_{IL}	Input Low Voltage		-0.1		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{CC} + 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.45	V
V_{OH1}	Output High Voltage	$I_{OH} = -400$ μA	2.4			V
V_{OL2}	Output Low Voltage	$I_{OL} = 0$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = 0$ μA	$V_{CC} - 0.1$			V

AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C256								Units
			Q17		Q20, Q200		Q25, Q250		Q300		
			Min	Max	Min	Max	Min	Max	Min	Max	
t _{AOC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$		170		200		250		300	ns
t _{OE}	\overline{CE} to Output Delay	$\overline{OE} = V_{IL}$		170		200		250		300	ns
t _{CE}	\overline{CE} to Output Delay	$\overline{CE} = V_{IL}$		75		75		100		120	ns
t _{OE}	\overline{OE} High to Output Float	$\overline{CE} = V_{IL}$	0	50	0	50	0	60	0	105	ns
t _{OF}	\overline{CE} High to Output Float	$\overline{OE} = V_{IL}$	0	50	0	50	0	60	0	105	ns
t _{OH}	Output Hold from Addresses, \overline{CE} or \overline{OE} Whichever Occurred First	$\overline{CE} = \overline{OE} = V_{IL}$	0				0		0		ns

MILITARY AND EXTENDED TEMPERATURE RANGE

Absolute Maximum Ratings (Note 1)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Temperature Under Bias	Operating Temp Range
Storage Temperature	-65°C to +150°C
All Input Voltages with Respect to Ground (Note 10)	+6.5V to -0.6V
All Output Voltages with Respect to Ground (Note 10)	$V_{CC} + 1.0V$ to $GND - 0.6V$
V_{PP} Supply Voltage with Respect to Ground During Programming	+14.0V to -0.6V

Power Dissipation	1.0W
Lead Temperature (Soldering, 10 sec.)	300°C
V_{CC} Supply Voltage with Respect to Ground	+7.0V to -0.6V

Operating Conditions (Note 7)

Temperature Range	-40°C to +85°C
NMC27C256QE200, 250	-55°C to +125°C
NMC27C256QM250, M350	
V_{CC} Power Supply	5V ± 10%

READ OPERATION

DC Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Load Current	$V_{IN} = V_{CC}$ or GND			10	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND, $\overline{CE} = V_{IH}$			10	μA
I_{CC1} (Note 9)	V_{CC} Current (Active) TTL Inputs	$\overline{CE} = V_{IL}$, $f = 5$ MHz Inputs = V_{IH} or V_{IL} , I/O = 0 mA		6	20	mA
I_{CC2} (Note 9)	V_{CC} Current (Active) CMOS Inputs	$\overline{CE} = GND$, $f = 5$ MHz Inputs = V_{CC} or GND, I/O = 0 mA		3	10	mA
I_{CCSB1}	V_{CC} Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCSB2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.5	100	μA
I_{PP}	V_{PP} Load Current	$V_{PP} = V_{CC}$			12	μA
V_{IL}	Input Low Voltage		-0.1		0.9	V
V_{IH}	Input High Voltage		2.0		$V_{CC} - 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.45	V
V_{OH1}	Output High Voltage	$I_{OH} = -400$ μA	2.4			V
V_{OL2}	Output Low Voltage	$I_{OH} = 0$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = 0$ μA	$V_{CC} - 0.1$			V

AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C256Q						Units
			E200		E250 M250		M350		
			Min	Max	Min	Max	Min	Max	
t _{ACC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$		200		250		350	ns
t _{CE}	\overline{CE} to Output Delay	$\overline{OE} = V_{IL}$		200		250		350	ns
t _{OE}	\overline{OE} to Output Delay	$\overline{CE} = V_{IL}$		75		100		120	ns
t _{OF}	\overline{OE} High to Output Float	$\overline{CE} = V_{IL}$	0	60	0	60	0	105	ns
t _{OH}	Output Held from Addresses, $\overline{CE} = \overline{OE}$ Whichever Occurs First	$\overline{CE} = \overline{OE} = V_{IL}$	0		0		0		ns
t _{OF}	Output to Output Float	$\overline{CE} = V_{IL}$	0	60	0	60	0	105	ns

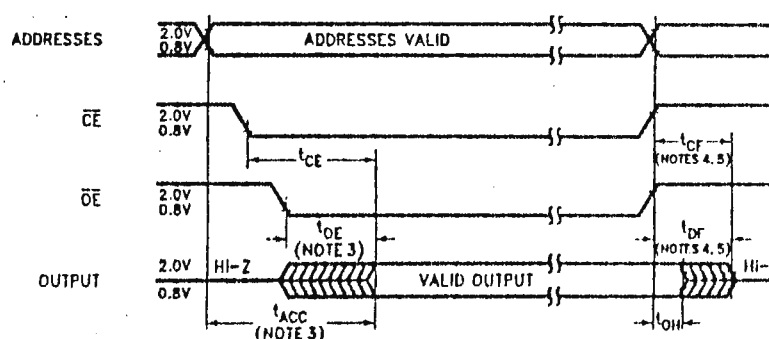
Capacitance $T_A = +25^\circ\text{C}$, $f = 1\text{ MHz}$ (Note 2)

Symbol	Parameter	Conditions	Typ	Max	Units
C_{IN}	Input Capacitance	$V_{IN} = 0\text{V}$	6	12	pF
C_{OUT}	Output Capacitance	$V_{OUT} = 0\text{V}$	9	12	pF

AC Test Conditions

Output Load	1 TTL Gate and $C_L = 100\text{ pF}$ (Note 8)	Timing Measurement Reference Level	
Input Rise and Fall Times	$\leq 5\text{ ns}$	Inputs	0.8V and 2V
Input Pulse Levels	0.45V to 2.4V	Outputs	0.8V and 2V

AC Waveforms (Notes 6, 7 & 9)



TL/D7512-3

Note 1: 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.

Note 2: This parameter is only sampled and is not 100% tested.

Note 3: OE may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of CE without impacting t_{ACC} .

Note 4: The t_{DF} and t_{CF} compare level is determined as follows:

High to TRI-STATE, the measured $V_{OH}(DC) - 0.10\text{V}$
Low to TRI-STATE, the measured $V_{OL}(DC) + 0.10\text{V}$

Note 5: TRI-STATE may be attained using OE or CE.

Note 6: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that at least a $0.1\text{ }\mu\text{F}$ ceramic capacitor be used on every device between V_{CC} and GND.

Note 7: The outputs must be restricted to $V_{CC} + 1.0\text{V}$ to avoid latch-up and device damage.

Note 8: 1 TTL Gate $I_{OH} = 1.6\text{ mA}$, $I_{OH} = -400\text{ }\mu\text{A}$

C_L 100 pF includes fixture capacitance.

Note 9: V_{PP} may be connected to V_{CC} except during pre-programming.

Note 10: Inputs and outputs can undershoot to -2.0V for 20 ns Max.

Programming Characteristics (Notes 1, 2, 3 & 4)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
t_{AS}	Address Setup Time		2			μs
t_{OES}	\overline{OE} Setup Time		2			μs
t_{VPS}	V_{PP} Setup Time		2			μs
t_{VCS}	V_{CC} Setup Time		2			μs
t_{DS}	Data Setup Time		2			μs
t_{AH}	Address Hold Time		0			μs
t_{DH}	Data Hold Time		2			μs
t_{DF}	Output Enable to Output Float Delay	$\overline{OE} = V_{IL}$	0		130	ns
t_{PW}	Program Pulse Width		0.5	0.5	10	ms
t_{OE}	Data Valid from \overline{OE}	$\overline{OE} = V_{IL}$			150	ns
I_{PP}	V_{PP} Supply Current During Programming Pulse	$\overline{OE} = V_{IL}$ $PGM = V_{IL}$			30	mA
I_{CC}	V_{CC} Supply Current				10	mA
T_A	Temperature Ambient		20	25	30	$^{\circ}C$
V_{CC}	Power Supply Voltage		5.75	6.0	6.25	V
V_{PP}	Programming Supply Voltage		12.2	13.0	13.3	V
t_{FR}	Input Rise, Fall Time		5			ns
V_{IL}	Input Low Voltage			0.0	0.45	V
V_{IH}	Input High Voltage		2.4	4.0		V
I_{IN}	Input Timing Reference Voltage		0.8	1.5	2.0	V
I_{OUT}	Output Timing Reference Voltage		0.8	1.5	2.0	V

Note 1: National's standard product warranty applies only to devices programmed to specifications described herein.

Note 2: V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP} . The EPROM must not be inserted into or removed from a board with voltage applied to V_{PP} or V_{CC} .

Note 3: The maximum absolute allowable voltage which may be applied to the V_{PP} pin during programming is 14V. Care must be taken when switching the V_{PP} supply to prevent any overshoot from exceeding this 14V maximum specification. At least a 0.1 μF capacitor is required across V_{PP} to GND to suppress spurious voltage transients which may damage the device.

Note 4: Programming and program verify are tested with the Interactive Program Algorithm, at typical power supply voltages and timing.

Programming Waveforms (Note 3)

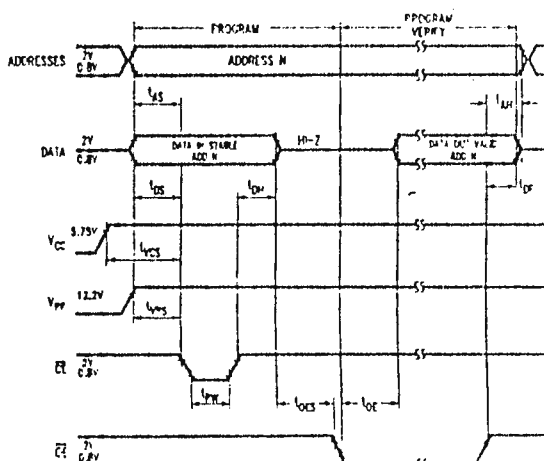


FIGURE 4

Functional Description

DEVICE OPERATION

The six modes of operation of the NMC27C256 are listed in Table I. It should be noted that all inputs for the six modes are at TTL levels. The power supplies required are V_{CC} and V_{PP} . The V_{PP} power supply must be at 13.0V during the three programming modes, and must be at 5V in the other three modes. The V_{CC} power supply must be at 6V during the three programming modes, and at 5V in the other three modes.

Read Mode

The NMC27C256 has two control functions, both of which must be logically active in order to obtain data at the outputs. Chip Enable (\overline{CE}) is the power control and should be used for device selection. Output Enable (\overline{OE}) is the output control and should be used to gate data to the output pins, independent of device selection. Assuming that addresses are stable, address access time (t_{ACC}) is equal to the delay from \overline{CE} to output (t_{CE}). Data is available at the outputs t_{OE} after the falling edge of \overline{OE} , assuming that \overline{CE} has been low and addresses have been stable for at least $t_{ACC} - t_{OE}$.

The sense amps are clocked for fast access time. V_{CC} should therefore be maintained at operating voltage during read and verify. If V_{CC} temporarily drops below the spec. voltage (but not to ground) an address transition must be performed after the drop to ensure proper output data.

Standby Mode

The NMC27C256 has a standby mode which reduces the active power dissipation by 99%, from 55 mW to 0.55 mW. The NMC27C256 is placed in the standby mode by applying a CMOS high signal to the \overline{CE} input. When in standby mode, the outputs are in a high impedance state, independent of the \overline{OE} input.

Output OR-Tying

Because NMC27C256s are usually used in larger memory arrays, National has provided a 2-line control function that accommodates this use of multiple memory connections. The 2-line control function allows for:

- a) the lowest possible memory power dissipation, and
- b) complete assurance that output bus contention will not occur.

To most efficiently use these two control lines, it is recommended that \overline{CE} (pin 20) be decoded and used as the primary device selecting function, while \overline{OE} (pin 22) be made a common connection to all devices in the array and connected to the READ line from the system control bus. This assures that all deselected memory devices are in their low power standby modes and that the output pins are active only when data is desired from a particular memory device.

Programming

CAUTION: Exceeding 14V on pin 1 (V_{PP}) will damage the NMC27C256.

Initially, and after each erasure, all bits of the NMC27C256 are in the "1" state. Data is introduced by selectively programming "0s" into the desired bit locations. Although only "0s" will be programmed, both "1s" and "0s" can be presented in the data word. The only way to change a "0" to a "1" is by ultraviolet light erasure.

The NMC27C256 is in the programming mode when the V_{PP} power supply is at 13.0V and \overline{OE} is at V_{IH} . It is required that at least a 0.1 μ F capacitor be placed across V_{PP} , V_{CC} to ground to suppress spurious voltage transients which may damage the device. The data to be programmed is applied 8 bits in parallel to the data output pins. The levels required for the address and data inputs are TTL.

When the address and data are stable, an active low TTL program pulse is applied to the $\overline{CE}/\overline{PGM}$ input. A program pulse must be applied at each address location to be programmed. Any location may be programmed at any time - either individually, sequentially, or at random. The NMC27C256 is designed to be programmed with interactive programming where each address is programmed with a series of 0.5 ms pulses until it verifies (up to a maximum of 20 pulses or 10 ms). The NMC27C256 must not be programmed with a DC signal applied to the $\overline{CE}/\overline{PGM}$ input.

Programming multiple NMC27C256s in parallel with the same data can be easily accomplished due to the simplicity of the programming requirements. Like inputs of the parallel NMC27C256s may be connected together when they are programmed with the same data. A low level TTL pulse applied to the $\overline{CE}/\overline{PGM}$ input programs the parallel NMC27C256s.

TABLE I. Mode Selection

Mode	Pins $\overline{CE}/\overline{PGM}$ (20)	\overline{OE} (22)	V_{PP} (1)	V_{CC} (28)	Outputs (11-13, 15-19)
Read	V_{IL}	V_{IL}	5V	5V	D_{OUT}
Standby	V_{IH}	Don't Care	5V	5V	Hi-Z
Program	V_{IL}	V_{IH}	13.0V	6V	D_{IN}
Program Verify	V_{IH}	V_{IL}	13.0V	6V	D_{OUT}
Program Inhibit	V_{IH}	V_{IH}	13.0V	6V	Hi-Z
Output Disable	Don't Care	V_{IH}	5V	5V	Hi-Z

Functional Description (Continued)

Program Inhibit

Programming multiple NMC27C256s in parallel with different data is also easily accomplished. Except for \overline{CE} all like inputs (including \overline{OE}) of the parallel NMC27C256s may be common. A TTL low level program pulse applied to an NMC27C256's $\overline{CE}/\overline{PGM}$ input with V_{pp} at 13.0V will program that NMC27C256. A TTL high level \overline{CE} input inhibits the other NMC27C256s from being programmed.

Program Verify

A verify should be performed on the programmed bits to determine whether they were correctly programmed. The verify may be performed with V_{pp} at 13.0V. V_{pp} must be at V_{CC} , except during programming and program verify.

ERASURE CHARACTERISTICS

The erasure characteristics of the NMC27C256 are such that erasure begins to occur when exposed to light with wavelengths shorter than approximately 4000 Angstroms (\AA). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000 \AA -4000 \AA range.

After programming, opaque labels should be placed over the NMC27C256's window to prevent unintentional erasure. Covering the window will also prevent temporary functional failure due to the generation of photo currents.

The recommended erasure procedure for the NMC27C256 is exposure to short wave ultraviolet light which has a wavelength of 2537 Angstroms (\AA). The integrated dose (i.e., UV intensity \times exposure time) for erasure should be a minimum of 15W-sec/cm².

The NMC27C256 should be placed within 1 inch of the lamp tubes during erasure. Some lamps have a filter on their tubes which should be removed before erasure. Table II

shows the minimum NMC27C256 erasure time for various light intensities.

An erasure system should be calibrated periodically. The distance from lamp to unit should be maintained at one inch. The erasure time increases as the square of the distance. (If distance is doubled the erasure time increases by a factor of 4.) Lamps lose intensity as they age. When a lamp is changed, the distance has changed or the lamp has aged, the system should be checked to make certain full erasure is occurring. Incomplete erasure will cause symptoms that can be misleading. Programmers, components, and even system designs have been erroneously suspected when incomplete erasure was the problem.

SYSTEM CONSIDERATION

The power switching characteristics of EPROMs require careful decoupling of the devices. The supply current, I_{CC} , has three segments that are of interest to the system designer—the standby current level, the active current level, and the transient current peaks that are produced by voltage transitions on input pins. The magnitude of these transient current peaks is dependent on the output capacitance loading of the device. The associated V_{CC} transient voltage peaks can be suppressed by properly selected decoupling capacitors. It is recommended that at least a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND. This should be a high frequency capacitor of low inherent inductance. In addition, at least a 4.7 μF bulk electrolytic capacitor should be used between V_{CC} and GND for each eight devices. The bulk capacitor should be located near where the power supply is connected to the array. The purpose of the bulk capacitor is to overcome the voltage drop caused by the inductive effects of the PC board traces.

TABLE II. Minimum NMC27C256 Erasure Time

Light Intensity (Micro-Watts/cm ²)	Erasure Time (Minutes)
15,000	20
10,000	25
5,000	50



PRELIMINARY

NMC27C512A

524,288-Bit (64k x 8) UV Erasable CMOS PROM

General Description

The NMC27C512A is a high-speed 512k UV erasable and electrically reprogrammable CMOS EPROM, ideally suited for applications where fast turnaround, pattern experimentation and low power consumption are important requirements.

The NMC27C512A is designed to operate with a single +5V power supply with $\pm 5\%$ or $\pm 10\%$ tolerance. The CMOS design allows the part to operate over extended and military temperature ranges.

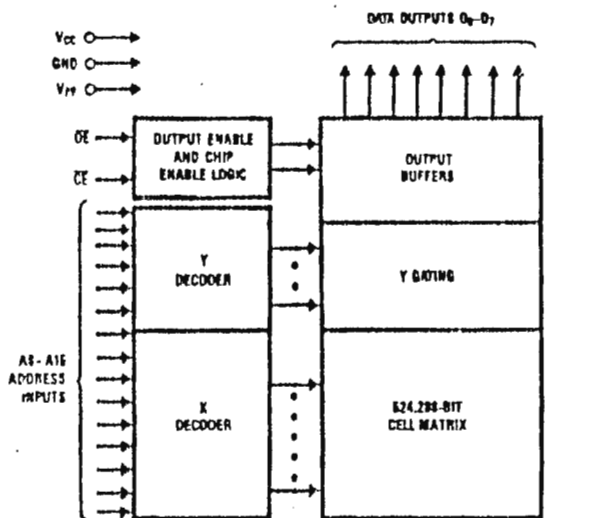
The NMC27C512A is packaged in a 28-pin dual in-line package with transparent lid. The transparent lid allows the user to expose the chip to ultraviolet light to erase the bit pattern. A new pattern can then be written electrically into the device by following the programming procedure.

This EPROM is fabricated with National's proprietary, time proven CMOS double-poly silicon gate technology which combines high performance and high density with low power consumption and excellent reliability.

Features

- Clocked sense amps for fast access time down to 150 ns
- Low CMOS power consumption
 - Active Power: 110 mW max
 - Standby Power: 0.55 mW max
- Optimum EPROM for total CMOS system
- Extended temperature range (NMC27C512AQE), -40°C to 85°C , and military temperature range (NMC27C512AQM), -55°C to 125°C , available
- Pin compatible with NMOS 512k EPROM
- Fast and reliable programming—100 μs typical/byte
- Static operation—no clocks required
- TTL, CMOS compatible inputs/outputs
- TRI-STATE* output
- Manufacturer's identification code for automatic programming control.
- High current CMOS level output drivers

Block Diagram



Pin Names

A0-A15	Addresses
CE	Chip Enable
OE/Vpp	Output Enable/Programming Voltage
O0-O7	Outputs
PM	Program

COMMERCIAL TEMPERATURE RANGE

Absolute Maximum Ratings (Note 1)

Temperature Under Bias	-10°C to +80°C
Storage Temperature	-65°C to +150°C
All Input Voltages except A9 & \overline{OE}/V_{PP} with Respect to Ground (Note 9)	+6.5V to -0.6V
V_{CC} Supply Voltage with Respect to Ground	+7.0V to -0.6V
ESD Rating (Mil. Std. 883C, Method 3015.2)	2000V
All Output Voltages with Respect to Ground (Note 9)	$V_{CC} + 1.0V$ to GND -0.6V

\overline{OE}/V_{PP} Supply Voltage & A9 with Respect to Ground	+14.0V to -0.6V
Power Dissipation	1 W
Lead Temperature (Soldering, 10 sec.)	300°C

Operating Conditions (Note 6)

Temperature Range	0°C to +70°C
V_{CC} Power Supply	
NMC27C512AQ15, 17, 20, 25	5V ± 5%
NMC27C512AQ150, 170, 200, 250	5V ± 10%

READ OPERATION

DC Electrical Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{LI}	Input Load Current	$V_{IN} = V_{CC}$ or GND		0.01	1	μA
I_{LO}	Output Leakage Current	$V_{OUT} = V_{CC}$ or GND, $\overline{CE} = V_{IH}$		0.01	1	μA
I_{PP}	V_{PP} Load Current	$\overline{OE}/V_{PP} = V_{CC}$ or GND			10	μA
I_{CC1}	V_{CC} Current (Active) TTL Inputs	$\overline{CE} = V_{IL}$, $f = 5$ MHz Inputs = V_{IH} or V_{IL} , $I/O = 0$ mA		15	30	mA
I_{CC2}	V_{CC} Current (Active) CMOS Inputs	$\overline{CE} = GND$, $f = 5$ MHz Inputs = V_{CC} or GND, $I/O = 0$ mA		10	20	mA
I_{CCS1}	V_{CC} Current (Standby) TTL Inputs	$\overline{CE} = V_{IH}$		0.1	1	mA
I_{CCS2}	V_{CC} Current (Standby) CMOS Inputs	$\overline{CE} = V_{CC}$		0.5	100	μA
V_{IL}	Input Low Voltage		-0.2		0.8	V
V_{IH}	Input High Voltage		2.0		$V_{CC} + 1$	V
V_{OL1}	Output Low Voltage	$I_{OL} = 2.1$ mA			0.40	V
V_{OH1}	Output High Voltage	$I_{OH} = -2.5$ mA	3.5			V
V_{OL2}	Output Low Voltage	$I_{OL} = 10$ μA			0.1	V
V_{OH2}	Output High Voltage	$I_{OH} = -10$ μA	$V_{CC} - 0.1$			V

AC Electrical Characteristics

Symbol	Parameter	Conditions	NMC27C512A								Units
			Q15, Q150		Q17, Q170		Q20, Q200		Q25, Q250		
			Min	Max	Min	Max	Min	Max	Min	Max	
t _{ACC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$		150		170		200		250	ns
t _{CE}	\overline{CE} to Output Delay	$\overline{OE} = V_{IL}$		150		170		200		250	ns
t _{OE}	\overline{OE} to Output Delay	$\overline{CE} = V_{IL}$		60		75		75		100	ns
t _{DF}	\overline{OE} High to Output Float	$\overline{CE} = V_{IL}$	0	50	0	55	0	55	0	60	ns
t _{CF}	\overline{CE} High to Output Float	$\overline{OE} = V_{IL}$	0	50	0	55	0	55	0	60	ns
t _{QW}	Output Hold After Addresses, \overline{CE} or \overline{OE} Whichever Occurs First	$\overline{CE} = \overline{OE} = V_{IL}$			0		0		0		ns

Capacitance $T_A = +25^\circ\text{C}$, $f = 1\text{ MHz}$ (Note 2)

Symbol	Parameter	Conditions	Typ	Max	Units
C_{IN1}	Input Capacitance except $\overline{\text{OE}}/\text{V}_{PP}$	$V_{IN} = 0\text{V}$	6	12	pF
C_{OUT}	Output Capacitance	$V_{OUT} = 0\text{V}$	9	12	pF
C_{IN2}	$\overline{\text{OE}}/\text{V}_{PP}$ Input Capacitance	$V_{IN} = 0\text{V}$	20	25	pF

AC Test Conditions

Output Load

1 TTL Gate and
 $C_L = 100\text{ pF}$ (Note 8)

Timing Measurement Reference Level

Inputs

0.8V and 2V

Input Rise and Fall Times

 $\leq 5\text{ ns}$

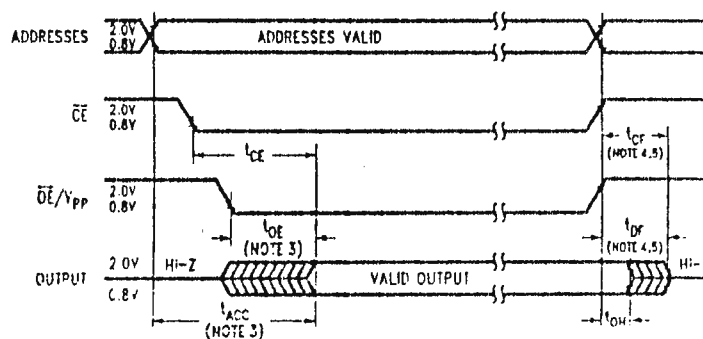
Outputs

0.8V and 2V

Input Pulse Levels

0.45V to 2.4V

AC Waveforms (Notes 6, 7)



Note 1: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and should not be used for 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 maximum rating conditions for extended periods may affect device reliability.

Note 2: This parameter is only sampled and is not 100% tested.

Note 3: $\overline{\text{OE}}$ may be delayed up to $t_{ACC} - t_{CE}$ after the falling edge of $\overline{\text{OE}}$ without impacting t_{ACC} .

Note 4: The t_{OE} and t_{OF} compare level is determined as follows:

High to TRI-STATE, the measured $V_{OL}(\text{NO}) = 0.10\text{V}$;

Low to TRI-STATE, the measured $V_{OH}(\text{NO}) = 0.10\text{V}$.

Note 5: TRI-STATE may be attained using $\overline{\text{OE}}$ or $\overline{\text{CE}}$.

Note 6: The power switching characteristics of EPROMs require careful device decoupling. It is recommended that at least a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND.

Note 7: The outputs must be restricted to $V_{CC} + 1.0\text{V}$ to avoid latch-up and device damage.

Note 8: 1 TTL Gate, $I_{OL} = 1.6\text{ mA}$, $I_{OH} = -400\text{ }\mu\text{A}$.

C_L 100 pF includes fixture capacitance.

Note 9: Inputs and outputs can withstand to $\pm 2.0\text{V}$ for 20 ns Max.

APENDICE E

MANUAL DEL USUARIO

E.1 DESCRIPCION GENERAL DEL PROGRAMADOR

El programador de memorias EPROM por computadora (PEXPC), programa las memorias 2616, 2732, 2764, 27128, 27256, 27512. La unidad puede ser operada en cualquier sistema equipado con slot disponible, para tarjetas E/S de 8 bits. Para los sistemas IBM PC\XT\AT y compatibles, un programa de menus puede controlar el PEXPC. Corriendo el programa PEXPC56.RDR desde un archivo por lotes (PEXPC.BAT), se puede controlar el PEXPC. Debido a que se ha trabajado en un modo de pantalla específico de QBASIC, este programa correrá adecuadamente en una computadora equipada con tarjeta de video VGA (color) o superior (color).

E.2 SUMARIO TECNICO DEL PEXPC.

Voltaje de operación: 110 VAC.

Conector: DB37M (MACHO)

Base de EPROM: ZIF-28

Memorias: 2716; 2732; 2764; 27128; 27256 y 27512.

Numero de bits por dato: 8 Bits

E.3 CONECTANDO EL PEXPC A LA COMPUTADORA.

Siga las instrucciones siguientes para conectar el PEXPC a la computadora.

**** AVISO:** SI NO ESTA USTED SEGURO DE COMO PROCEDER CON LA INSTALACION DE PERIFERICOS OPCIONALES, COMO EL PEXPC, LE RECOMENDAMOS QUE BUSQUE LA ASISTENCIA TECNICA CALIFICADA, YA QUE UNA INSTALACION INCORRECTA PUEDE CAUSAR SERIOS DAÑOS A SU COMPUTADORA O TARJETA PERIFERICA.

a) Abra el gabinete de la computadora, remueva el tornillo de la cubierta de la abertura de expansión, situado en la parte trasera de la computadora y quite la cubierta. Instale la tarjeta de decodificación del PEXPC en uno de los slot E/S disponible y asegúrela al chasis de su PC.

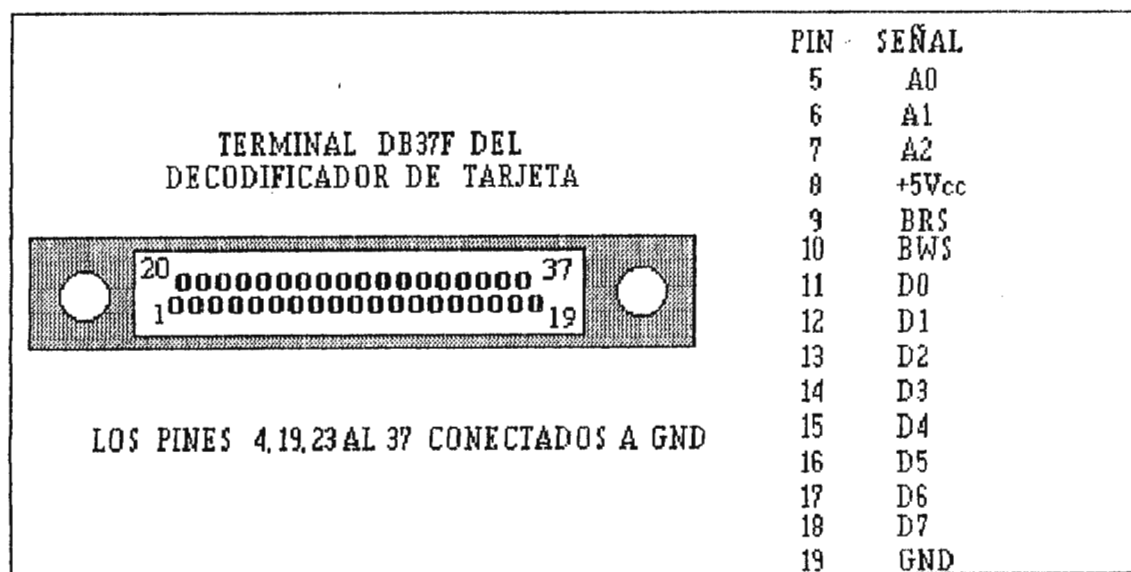


Figura E.1, Muestra las señales en los distintos pines del conector DB37F del decodificador de Tarjeta del PEXPC.

E.3 CONECTANDO EL PEXPC A LA COMPUTADORA.

Siga las instrucciones siguientes para conectar el PEXPC a la computadora.

**** AVISO:** SI NO ESTA USTED SEGURO DE COMO PROCEDER CON LA INSTALACION DE PERIFERICOS OPCIONALES, COMO EL PEXPC, LE RECOMENDAMOS QUE BUSQUE LA ASISTENCIA TECNICA CALIFICADA, YA QUE UNA INSTALACION INCORRECTA PUEDE CAUSAR SERIOS DAÑOS A SU COMPUTADORA O TARJETA PERIFERICA.

a) Abra el gabinete de la computadora, remueva el tornillo de la cubierta de la abertura de expansión, situado en la parte trasera de la computadora y quite la cubierta. Instale la tarjeta de decodificación del PEXPC en uno de los slot E/S disponible y asegúrela al chasis de su PC.

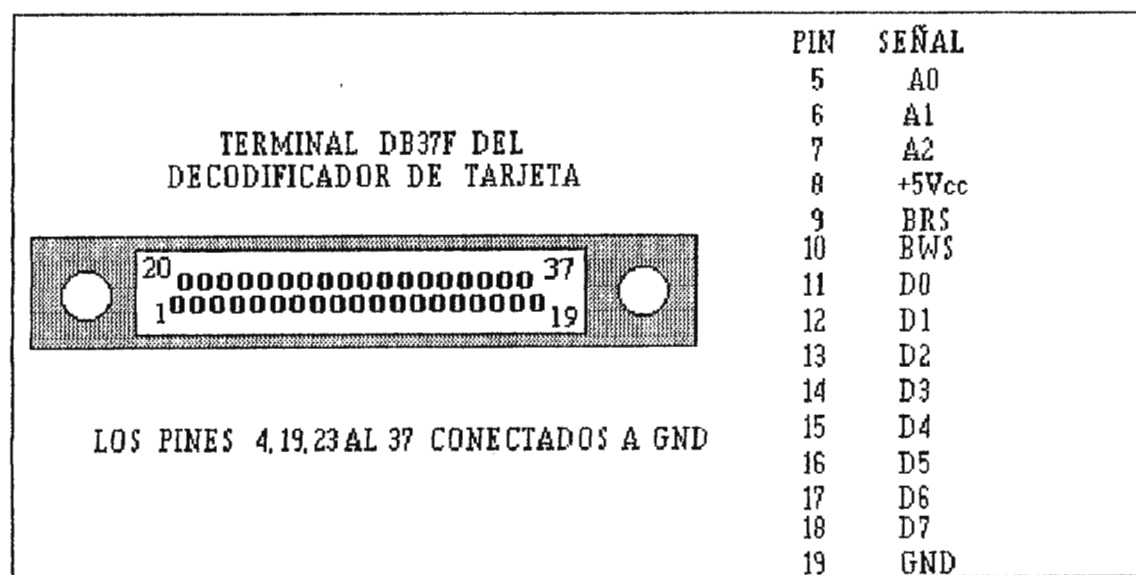


Figura E.1, Muestra las señales en los distintos pines del conector DB37F del decodificador de Tarjeta del PEXPC.

b) Conecte el bus del programador en la terminal DB37F (hembra) del decodificador de tarjeta instalado en el slot E/S de 8 bits, de la computadora. La figura E.1, muestra el conector de 37 pines (DB37F).

c) Encienda la computadora y asegúrese que en el disco se encuentren los siguientes archivos: PEXPC.BAT, PEXPC56.RDR, QBASIC.INI, QBASIC.EXE; y corra el programa a:\PEXPC.BAT.

E.4 COMANDOS DEL PROGRAMADOR

COMANDOS	DESCRIPCION
[ESC]	Cancela sub-menu y retorna al menu principal.
[S]ALIR	Comando para salir del programa hacia el DOS.
[A]RCHIVO	Comando que habilita el sub-menu para ver, crear o imprimir archivos.
[L]EER	Comando que habilita sub-menu para crear un archivo en una unidad de disco y almacenar en él el contenido de un rango de una EPROM seleccionada
[G]RABAR	Comando que habilita sub-menu para tranferir una archivo creado a una EPROM adecuada.
[S]	Confirma salida del comando de salir.
[N]	Niega el comando salir y retorna al menu principal
[C]	<ul style="list-style-type: none"> - Al inicio del programa, permite continuar después de haber removido cualquier EPROM del PEXPC. - En el sub-menu de ver archivo permite continuar viendo el contenido de un archivo con más de 16 líneas de dirección. - En los sub-menu de leer y grabar EPROM permite continuar después de haber colocado la EPROM en el PEXPC.
[1],[2],[3],[4],[5],[6]	En los sub-menu de grabar y leer EPROM permite seleccionar una memoria correlativa a cada uno de los comandos, 2716, 2732, 2764, 27128, 27256, 27512 respectivamente.

E.5 USANDO DIRECCION INICIAL Y DIRECCION FINAL

En los sub-menu para leer y grabar EPROM sirven para indicar la dirección de inicio y fin del rango de direcciones que se grabarán en una EPROM o que se leerán de una EPROM.

E.6 PROBLEMAS QUE SE PUEDEN DAR AL CORRER EL PROGRAMA.

a) Una vez dentro del programa y específicamente dentro del sub-menu de abrir archivo se puede dar el problema de que se interrumpa el programa y caer dentro de QBASIC debido a que se quiera abrir un archivo que no existe. QBASIC presentará una ventana donde indica el error, teniendo que presionarse la tecla [ENTER] y luego SHIFT+F5 para reinicializar todo el programa. Al ocurrir esto y querer salir del programa habrá que salir con los comandos de QBASIC.

b) Si al correr el programa se da un problema de incompatibilidad de video debido a que el programa ha sido elaborado con atributos de color, se puede solucionar cambiando los atributos de color y modo de pantalla (resolución).

E.7 PASOS PARA USAR EL PEXPC, (Ejemplo para grabar una 2716)

1. Antes de encender la computadora asegúrese de que el PEXPC está debidamente conectado y que en el porta EPROM de éste no hay ninguna EPROM.

2. Estando en el menú principal, se selecciona la opción ARCHIVO presionando [A], y aparecerá el sub-menú del cual se selecciona la opción NUEVO presionando [N]. Aparecerá una ventana y se pedirá la ruta y el nombre del archivo que se creará, A:\udb, después aparecerán los mensajes "INICIO FINAL", los cuales se refieren a la dirección inicial y final del rango de direcciones que deseamos grabar. Como el ejemplo es conserniente a una EPROM 2716 las direcciones no pueden ser mayores que 07FFH. Introduciremos como dirección inicial 0000H y como final 0002H teniendo un rango de 3 líneas, el rango aparecerá debajo de las dos direcciones anteriores.

3. Luego de haber indicado el rango, aparecerá la primera dirección (0000H) y deberemos introducir a la derecha de ésta el dato de 8 bits en hexadecimal que se desea, para el ejemplo se introducirá FFH en las tres direcciones hasta llegar a la línea 0002H. Después de este paso se presionará [ESC] dos veces hasta volver al menú principal.

4. Estando en el menú principal se escoge la opción GRABAR presionando [G], y aparecerá una pequeña ventana donde habrá que escribir el nombre del archivo que se quiere que sea grabado en la 2716. Después de indicado el archivo con su ruta (A:\udb), el programa buscará la dirección final del archivo y sugerirá las EPROM que pueden ser utilizadas. Como la dirección final de este archivo es 0002H, el programa sugerirá que se utilicen cualquiera de las 6 EPROM y aparecerá un mensaje que sugiere esto.

5. Después aparece un menú de selección de EPROM donde sólo son habilitadas las que el programa ha sugerido. Se selecciona la 2716 presionando [1] y aparecerá un mensaje que nos indica que en ese momento ya podemos colocar la EPROM en el porta EPROM, presionando [C] después de haber colocado la EPROM, el programa comenzará a leer el archivo udb y a transferirlo al PEXPC. Al finalizar el proceso el programa vuelve al menú principal.

6. Estando en el menú principal se puede seleccionar la opción LEER presionando [L], y aparecerá un sub-menú de selección de EPROM, seleccionamos la 2716 ([1]), se indica en que archivo se quiere que sea almacenado el contenido de la memoria (A:\udbl), y luego el rango que se desea leer (0000H-0002H), presionando [ENTER], el programa comenzará a almacenar el contenido de las tres líneas en el archivo udbl, visualizando rápidamente los datos de esas tres direcciones.

7. Seleccionando del menú principal la opción ARCHIVO y luego ABRIR presionando dos veces [A], se puede ver detenidamente el contenido del archivo udbl y verificar si el archivo udb fue grabado correctamente en la EPROM 2716.

BIBLIOGRAFIA

- TOCCI, Ronald J. Sistemas Digitales/ Principios y Aplicaciones. (Mexico, 1988)
- DRISCOLL, Frederick F. Microprocessor-Microcomputer Technologies. (U.S.A., 1983)
- TEXAS INSTRUMENTS. TTL Data Book For Design Engineers. (Dallas, 1988)
- NATIONAL SEMICONDUCTOR. CMOS EPROMs Data Book. (U.S.A.)
- NTE ELECTRONICS, INC. NTE Semiconductors. (Sexta Edición-New Jersey, 1992)
- PHILLIPS ECG. ECG Semiconductors Guide Reference. (Pensylvania, 1988)