

54 2ej



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

DISEÑO Y CONSTRUCCION DE UNA INTERFAZ PARA MANEJO DE SEÑALES ANALOGICAS Y DIGITALES IBM/PC - COMPATIBLE

TESIS PROFESIONAL
QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
P R E S E N T A :
JOSE GABRIEL GARCIA CRUZ

DIRECTOR: M. EN I. ROLANDO CARRERA MENDEZ

CIUDAD UNIVERSITARIA, D.F. OCTUBRE 1990

FALLA DE ORIGEN



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

INDICE

1. INTRODUCCION.	1
2. EL DISEÑO DE LA INTERFAZ.	4
3. DESCRIPCIÓN DE LA INTERFAZ.	
3.1 Descripción general.	5
3.2 La tarjeta D/A y entrada/salida de señales binarias.	7
3.3 Operación de la tarjeta interfaz.	9
3.3-1 Decodificación de direcciones	10
3.3-2 Bloque de memoria	10
3.3-3 Bloque de conversión	12
3.3-4 Bloque multicanal	12
3.3-5 Secuenciador	12
4. EL BUS DE LA COMPUTADORA PERSONAL IBM COMPATIBLE Y LA INTERFAZ.	
4.1 Introducción.	15
4.2 Mapa de memoria.	21
4.3 Lectura o escritura en la interfaz.	22
5. DESCRIPCIÓN DE LOS CIRCUITOS ELECTRONICOS.	
5.1 Introducción	25
5.2 Circuito de decodificación de direcciones y señales de control.	26
5.3 Circuito de lectura y escritura.	29
5.4 Conversión Digital/Analógica.	37
5.5 Secuenciador.	46

5.6	Circuito de retención.	49
5.7	Circuito integrador y puerto paralelo.	55
5.8	Módulo de conexiones.	59
6. PROGRAMACION.		
6.1	Introducción.	61
6.2	Iniciación de los canales analógicos de salida.	62
6.3	Programación del contador de Eventos externos.	63
6.3-1	Formato de la palabra de control.	65
6.4	Programación del puerto paralelo.	67
6.4-1	Formato de la palabra de control.	68
6.5	Descripción del programa.	70
7.	RESULTADOS.	75
8.	BIBLIOGRAFIA.	77
ANEXO A		
	Listado del programa de presentación para el manejo de la interfaz Digital/Analógica.	79
ANEXO B		
	Diagramas electrónicos del módulo de conexiones.	114
ANEXO C		
	Mapa de componentes de la interfaz.	119
ANEXO D		
	Lista de componentes.	126
ANEXO E		
	Especificaciones técnicas.	130

1. INTRODUCCION.

En la actualidad la computadora es una herramienta fundamental para el desarrollo de diferentes actividades en las que se desenvuelve el ser humano, estas pueden ser la docencia, la investigación, la administración y la industria entre otras.

Por otro lado existen en el mercado diferentes dispositivos que junto con computadoras personales y sus componentes periféricos permiten la automatización de diversos procesos entre los cuales se pueden mencionar:

- La adquisición de datos.
- El procesamiento de señales.
- El control de procesos industriales.
- El monitoreo de procesos.
- Simulación de procesos.

Los dispositivos arriba mencionados actualmente se pueden conseguir a precios muy altos, lo que ha ocasionado que la automatización de diferentes procesos se torne casi imposible, como consecuencia crece día con día la necesidad de crear y construir en nuestro país los dispositivos que resuelven los problemas que presenta la creciente industrialización del país.

Entre éstos dispositivos se encuentran difundidas ampliamente las tarjetas interfaces. Estas interfaces permiten que las computadoras puedan intercambiar señales con el medio externo.

La simbiosis Computadora Personal-Tarjeta Interfaz de uso específico ha permitido que los límites de aplicación de las

computadoras personales se extiendan ilimitadamente.

Como ejemplo de estas tarjetas interfaces, existen actualmente en el mercado las siguientes tarjetas:

-Tarjetas de procesamiento de señales (PCL - 714 Super Lab Card, PCL - 712 Multi Lab Card, INST 74, INST 83).

-Tarjetas de comunicación entre computadoras personales (PCL -712).

-Tarjetas para manejo de relevadores (PCLD - 785).

-Tarjetas para control de motores de pasos (PCL - 738).

Qué es una interfaz ? la interfaz es un elemento que permite la comunicación electrónica entre dos dispositivos cuyas señales a intercambiar no son compatibles (en este caso se trata de interfaces para computadora personales con dispositivos externos). La interfaz está construida de tal manera que puede aceptar o emular señales externas para hacerlas compatibles con el sistema de la computadora.

Las características que debe presentar una interfaz son las siguientes :

- 1.- La interpretación de las señales de dirección y control, provenientes del bus de la computadora, cuando un dispositivo ha sido seleccionado.
- 2.- La sincronía en el tiempo para que los datos digitales puedan ser recibidos o enviados sobre el bus de datos.
- 3.- La capacidad de decodificación de comandos provenientes

de la computadora para realizar una tarea específica.

El objetivo del trabajo aquí presentado es desarrollar una interfaz que satisfaga las características arriba mencionadas y que sea posible de ser reproducido en forma única o en serie.

El escrito se ha dividido en 8 capítulos y 5 anexos. Los capítulos se refieren al diseño de la interfaz, mientras los anexos contienen tanto información técnica como el programa de presentación de la interfaz, diagramas electrónicos e información técnica.

El capítulo dos describe el objetivo de la construcción de la interfaz y sus principales características. En el tercero se explican las diferentes partes en que está constituida y se plantea, de manera simplificada, el problema que presenta el diseño de la interfaz. El capítulo cuatro se refiere a las características del bus de la computadora y se describen cada una de ellas, se presenta un esquema general de las partes que constituyen la interfaz. Dentro del quinto capítulo se describe en forma más detallada cada una de las partes de la interfaz y se relacionan directamente con sus circuitos electrónicos. El sexto capítulo presenta el desarrollo de la programación de cada uno de los dispositivos y hace una presentación del funcionamiento de un programa que permite operar la interfaz. El séptimo capítulo presenta las conclusiones e alcances que tuvo en el presente trabajo. El octavo capítulo presenta las diferentes fuentes informativas que apoyaron al desarrollo del trabajo.

Con respecto a los anexos, el anexo A presenta un listado del programa que se desarrolló para operar la interfaz. El anexo B, C y D presentan la constitución y distribución de los componentes activos y pasivos que componen la interfaz. El anexo E presenta las especificaciones técnicas resumidas.

2. EL DISEÑO DE LA INTERFAZ.

El objetivo del presente trabajo es el diseño y construcción de una interfaz para acoplamiento de señales digitales y analógicas, compatibles con computadoras personales IBM-COMPATIBLES.

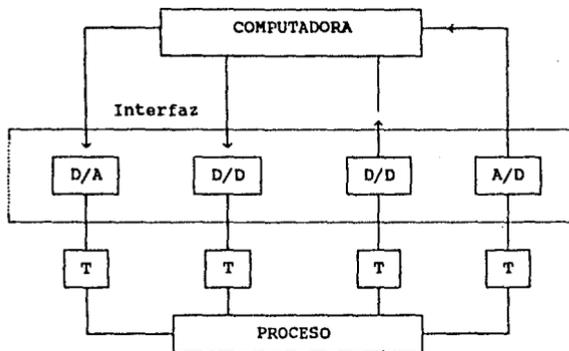
La finalidad, al desarrollar esta tarjeta interfaz, es crear un dispositivo que pueda ser adquirido por la industria o laboratorios de investigación para automatizar o simular sus procesos con las siguientes ventajas:

- I.- Disponibilidad desde una computadora personal, de señales analógicas y digitales suficientes para la automatización de cualquier proceso de control o manufactura.
- II.- Precisión y rapidéz semejantes a las que existen en el mercado internacional, para este tipo de interfaces.
- III.- Adquirible en el mercado nacional.
- IV.- Bajo costo de la tarjeta.
- V.- Fácil programación y uso.

3. DESCRIPCIÓN DE LA INTERFAZ

3.1 Descripción general.

Para que la computadora personal pueda comunicarse con un proceso externo requiere de una interfaz capaz de transferir las señales provenientes del proceso en variables que maneja la computadora. Por eso, en el diagrama funcional de la figura 3.1 se pueden observar los diferentes bloques que componen una interfaz para control de procesos o adquisición de datos.



D - Digital.
A - Analógico.
T - Transductor.

FIG. 3.1 Diagrama general de una interfaz que maneja señales analógicas y digitales.

Las funciones que desarrolla la interfaz son aceptar o enviar datos analógicos o digitales; tales funciones se llevan a cabo por medio de los siguientes dispositivos.

Convertidor analógico/digital (A/D).
Convertidor digital/analógico (D/A).
Acopladores de señales digitales (D/D).

La idea fundamental en el diseño de esta interfaz es tener la capacidad de poder conectar una computadora PC-IBM compatible con una variedad amplia de procesos, cuyas variables puedan ser leídas o controladas por medio de señales eléctricas.

En el presente desarrollo la interfaz ha sido concebida para realizar sólo una parte de estas funciones, estas se reducen al acoplamiento digital de entrada/salida y al manejo de señales analógicas de salida.

Para que la PC y la interfaz puedan comunicarse se requiere que haya un intercambio adecuado de señales, como son: datos, direcciones y control. En la figura 3.2 se muestra en forma esquemática el intercambio de estas señales.

Para que la interfaz pueda ser acoplada con la PC, se requiere que las señales a intercambiar cumplan con ciertas restricciones impuestas por la arquitectura de los buses de la PC. A continuación se enumeran las características necesarias y restricciones que debe presentar la interfaz:

- i) Compatibilidad de nivel lógico.
- ii) Compatibilidad de velocidad de las señales.
- iii) Consumo de corriente.
- iv) Compatibilidad de tecnología (TTL-LS).

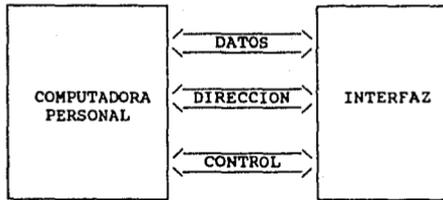


FIG 3.2. Intercambio de señales PC-Interfaz.

3.2. La tarjeta D/A y entrada/salida de señales binarias.

Para su funcionamiento adecuado, la interfaz requiere de otros dispositivos, como se observa en la figura 3.3. Estos dispositivos usualmente son transductores que permiten acondicionar una señal para realizar sobre ella una actividad determinada.

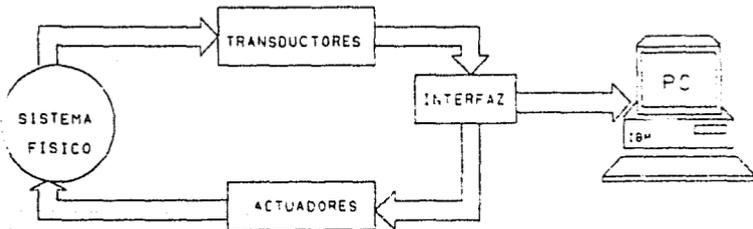


Fig 3.3. Diagrama de interrelación PC - Mundo real

La tarjeta ha sido diseñada en dos partes, la primera parte consiste en un módulo externo para conexiones, la segunda parte es el módulo de conversión y acoplamiento con la computadora, como se indica en la figura 3.4.

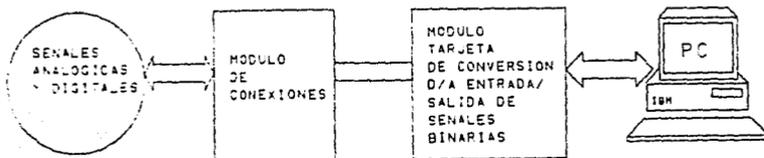


Fig. 3.4 diagrama de acoplamiento PC - señales eléctricas

Esta interfaz está constituida por tres secciones. A continuación se describe cada una de estas.

- 1) Sección analógica. Consta de 16 canales analógicos de salida. Estos permiten una variación de voltaje entre -5.000 V y $+4.997\text{V}$, este voltaje puede ser incrementado o decrementado en valores mínimos de 2.5 mV . Estos voltajes se actualizan por medio de comandos provenientes de un programa escrito exprefeso para controlar la interfaz.
- 2) Sección de señales binarias. Contiene 16 canales digitales de salida y 8 canales digitales de entrada. Estos canales digitales manejan cargas TTL. y son controlados por medio de un programa que se encuentra

operando en la computadora, el que permite leer o escribir estados en los diferentes canales digitales.

- 3) Sección contadores de eventos externos. A partir de 3 contadores programables de 16 bites se puede lograr contabilizar hasta 65536 eventos en cada contador. Estos contadores son independientes y son iniciados por programa.

3.3 Operación de la tarjeta interfaz.

Como se mencionó en el inciso anterior la interfaz contiene tres secciones básicas y cada una de ellas está compuesta por diferentes bloques, los cuales realizan diferentes funciones.

La primera sección corresponde a los canales analógicos de salida, consiste de los siguientes bloques:

- a) Decodificación.
- b) Memoria.
- c) Bloque de conversión.
- d) Multicanal.
- c) Secuenciador.

La segunda sección corresponde a los canales digitales de entrada y/o salida, con los siguientes bloques:

- a) Decodificación.
- b) Puerto paralelo de 24 bites.

La tercera sección corresponde a los contadores de eventos externos, esta sección contiene los siguientes bloques:

- a) Decodificación.
- b) Contadores de eventos externos.

A continuación se describe cada uno de estos bloques, su diagrama funcional se encuentra en la figura 3.5.

3.3-1. Decodificación de direcciones.

Este bloque se encuentra en las tres secciones de que consta la tarjeta y permite discernir que sección ha sido seleccionada. La selección se efectúa cuando un grupo de direcciones presentados por el programa correspondan a las direcciones de uno de los dispositivos en la tarjeta. La selección de un dispositivo permite la realización de actividades de lectura/escritura, estos dispositivos son: convertidor D/A, puerto paralelo y contadores programables.

Los siguientes bloques que se describen corresponden a la sección analógica.

3.3-2 Bloque de memoria.

Se tiene un registro de memoria para cada canal analógico; cada registro almacena el valor binario del voltaje de salida requerido.

La lectura/escritura en la memoria se realiza por medio de un programa que permite al usuario leer o escribir en cada registro. Este proceso es llevado a cabo con la instrucción de lectura o escritura a un puerto de entrada/salida. Los programas de uso para esta interfaz se pueden realizar con lenguajes de alto nivel como Pascal o ensamblador.

Después de que el usuario haya indicado los cambios en cada uno de los canales analógicos de salida, las memorias van a tener

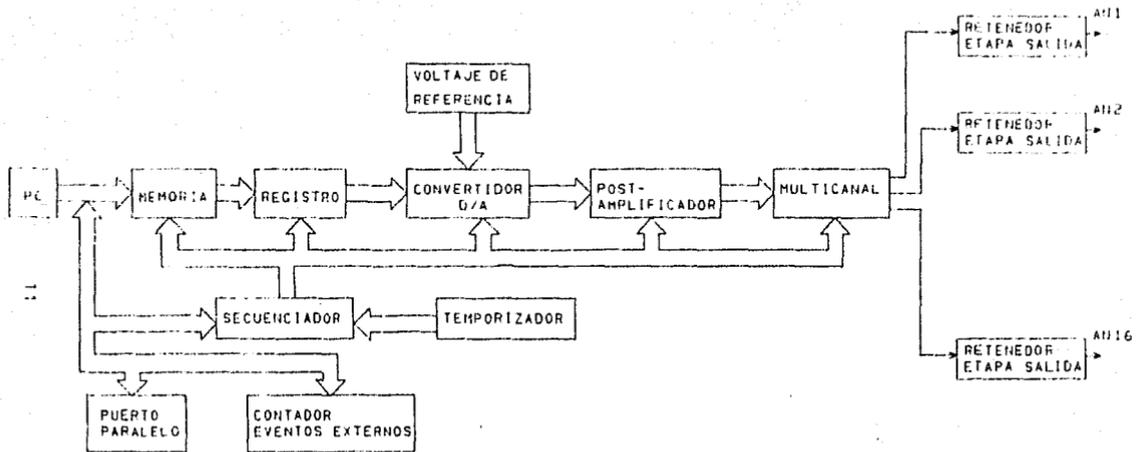


FIG 3.5 DIAGRAMA FUNCIONAL DE LA INTERFAZ D/A

una cierta información, la cual van a mantener hasta que se realice un nuevo cambio.

3.3-3 Bloque de conversión.

En este bloque se realiza la conversión de un dato digital a un voltaje analógico. La conversión es realizada por medio de un convertidor Digital-Analógico de 12 bites. Los datos son tomados de la memoria (sección 3.3-2) por un secuenciador (sección 3.3-5) y almacenados en el registro del convertidor para su posterior conversión.

Para acoplar y amplificar el voltaje que se encuentra a la salida del convertidor se tiene una etapa de conversión de corriente a voltaje y una etapa de amplificación de voltaje.

3.3-4 Bloque multicanal.

En esta parte se realiza la selección del canal analógico de salida que va a ser actualizado. El circuito multicanal se activa por medio de las señales provenientes del secuenciador (sección 3.3-5).

Una vez que se ha seleccionado un canal analógico, la señal de voltaje proveniente del convertidor se presenta a un circuito retenedor de orden cero. Este retenedor realiza la función de mantener estable el voltaje analógico de salida. Se tiene un circuito de retención para cada uno de los diferentes canales analógicos.

3.3-5 Secuenciador.

La secuencia de actividades de conversión y selección del canal analógico es realizada por el circuito secuenciador. La figura 3.6

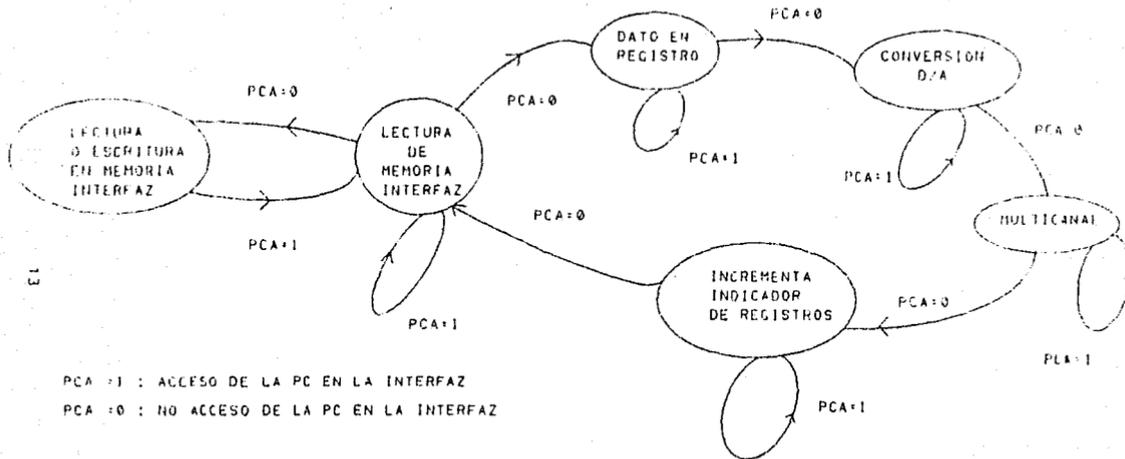


FIG 3.6 DIAGRAMA DE ESTADOS DEL SECUENCIADOR

muestra los diferentes estados que sigue el secuenciador. A continuación se enuncian las actividades que realiza el secuenciador:

- i) Lectura de memoria y carga en el registro del convertidor.
- ii) Inicio de conversión.
- iii) Selección de canal de salida.
- iv) Incremento del apuntador de dirección.

El circuito secuencial inicia su operación realizando la lectura de un registro en la memoria de la tarjeta; después el secuenciador enviará una secuencia de pulsos de control que iniciaran la conversión. Al fin de la conversión, el secuenciador genera la dirección para el circuito multicanal, con la que indica que canal es seleccionado y el valor de voltaje es actualizado, después incrementa el apuntador de direcciones y comienza la secuencia nuevamente.

Este proceso es cíclico cuando la tarjeta no es accesada por la PC. El secuenciador es inhibido cuando la PC está escribiendo o leyendo en los diferentes registros de los canales analógicos (PCA=1).

En la segunda sección se realizan la actividades de obtención de las señales digitales de entrada/salida. Estas señales se dividen en dos partes : una de entrada (8 canales) y una de salida (16 canales). Cada uno de ellas tiene una etapa de acoplamiento realizada con circuitos TTL-LS.

En la tercera sección se contabilizan eventos externos. Esto se realiza por medio de tres contadores programables. Cada uno de ellos es independiente y se encuentran localizados en el mismo circuito integrado.

4. EL BUS DE LA COMPUTADORA PERSONAL IBM-COMPATIBLE Y LA INTERFAZ

4.1 Introducción.

En este capítulo se describen las características del bus de la computadora personal tipo IBM y los atributos de las señales del mismo.

Las señales del bus se agrupan de la siguiente manera:

- a) Datos.
- b) Direcciones.
- c) Control.
- d) Fuente de Energía.

En la figura 4.1 se muestra el conector que se haya situado sobre la tarjeta madre y que permite conectar dispositivos externos al bus de la computadora. Este conector contiene las señales del bus. A continuación se muestran las características de cada grupo.

a) DATOS.

El bus de datos se encuentra constituido por ocho líneas bidireccionales (D0-D7), las cuales pueden transmitir información entre el procesador 8088, memorias y dispositivos externos. El bit menos significativo se encuentra indicado con D0 y el más significativo con D7. Cuando sucede un ciclo de escritura los

datos aparecen sobre este bus para ser captados por la memoria o algún dispositivo externo. En cambio en un ciclo de lectura, la dirección se encuentra estable mientras el procesador realiza la transferencia de información del dispositivo direccionado al procesador de la computadora.

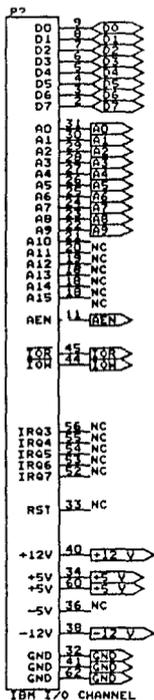


Fig 4.1 Diagrama esquemático del bus de la PC.

b) DIRECCIONES.

Este bus se encuentra constituido con 20 líneas de salida (A0 - A19) unidireccionales, A0 corresponde al bit menos significativo y A19 al bit más significativo. Estas señales son manejadas por el procesador 8088 durante los ciclos de lectura o escritura en memoria o dispositivos externos. Con 20 líneas de dirección es posible direccionar 1 megabyte de memoria, pero no todas las direcciones se encuentran disponibles para el uso de dispositivos entrada/salida (E/S).

Por medio del uso de las instrucciones de lectura o escritura a un puerto E/S se pueden direccionar solo 64K bytes. Las direcciones que se utilizan para poder seleccionar un dispositivo E/S corresponden a las líneas A0 - A9, es decir, la combinación de cada una de estas nos permitirá el direccionamiento de un dispositivo entrada/salida externo.

c) CONTROL.

De entre las señales que componen el bus de control, la interfaz utiliza las siguientes:

Reloj (CLK). Esta es una señal cuadrada de 4.77 Mhz. la cual es derivada de un circuito oscilador a cristal. Tiene un ciclo de trabajo del 33.3% , un período de 210 nanosegundos con un tiempo activo de 70 nanosegundos y un tiempo inactivo de 140 nanosegundos. Esta señal se encuentra sincronizada con las líneas de control de escritura o lectura a memoria. Un ciclo típico de lectura o escritura es de 4 períodos de reloj, o aproximadamente 840 nanosegundos como se indica en la figura 4.2.

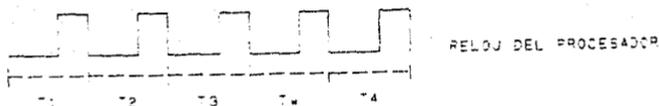


Fig. 4.2 Diagramas de tiempos del reloj.

HABILITACION DEL LATCH DE DIRECCIONES (ALE).-Esta es una señal de salida , indica cuando las direcciones del bus son válidas; se activa con el primer pulso de reloj en un ciclo de lectura o escritura. Cuando es activa alta , las direcciones son válidas.

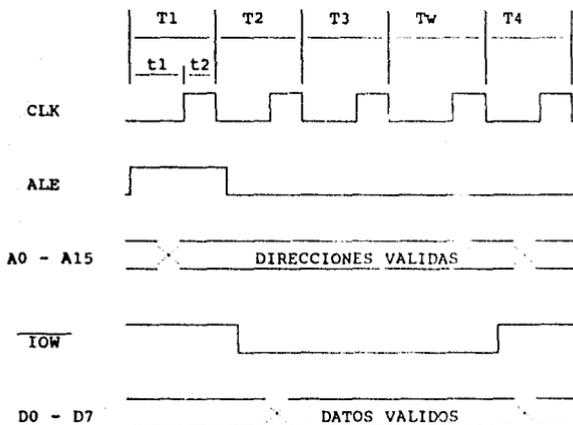
HABILITACION DE REFRESCAMIENTO A MEMORIA (AEN).-Esta señal es activa alta e indica que se encuentra presente un ciclo de acceso directo a memoria (DMA), con AEN activa se deshabilitan las líneas de dirección, datos y control del procesador 8088 para cualquier dispositivo entrada/salida.

ESCRITURA A UN PUERTO (IOW).- Esta señal es activa baja y se encuentra manejada por el circuito controlador 8288, ella indica, que los datos se encuentran sobre el bus para que puedan ser cargados en el dispositivo seleccionado. En la fig 4.3 se muestra el diagrama de tiempos en un ciclo de escritura a puerto entrada/salida.

LECTURA A UN PUERTO (IOR).- Esta señal es activa baja, sincroniza la lectura de un dispositivo entrada/salida. En la figura 4.4 se muestra su diagrama de tiempos.

d) FUENTES DE ENERGIA.

El suministro de energia se encuentra contituido por diferentes voltajes de corriente directa, en la siguiente tabla se muestran dichos voltajes y sus posibles variaciones.

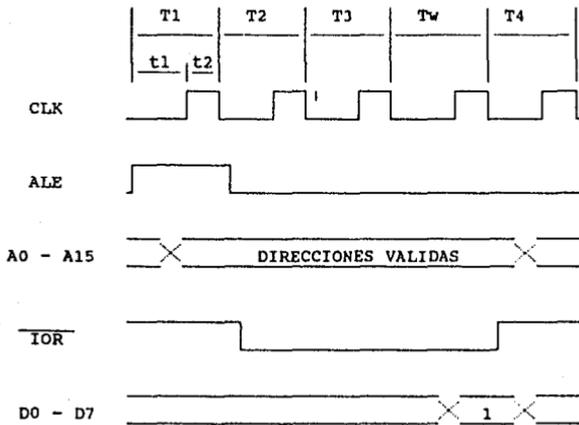


$T_1 = T_2 = T_3 = T_4 = 210$ ns.

$t_1 = 140$ ns.

$t_2 = 70$ ns.

FIG 4.3 DIAGRAMA DE TIEMPOS DE UN CICLO DE ESCRITURA A UN PUERTO ENTRADA/SALIDA.



$T1 = T2 = T3 = T4 = 210 \text{ ns.}$

$t1 = 140 \text{ ns.}$

$t2 = 70 \text{ ns.}$

1 = DATOS VALIDOS

FIG. 4.4 DIAGRAMA DE TIEMPOS EN UN CICLO DE LECTURA A UN PUERTO ENTRADA/SALIDA.

VOLTAJE	VARIACION
a) 5 VDC	+4.75 A +5.25 VDC
b) +12 VDC	+11.4 A +12.6 VDC
c) -12 VDC	-10.8 A -13.2 VDC
d) -5 VDC	-4.5 A -5.5 VDC
e) GND	NO CONTIENE VARIACION.

4.2 Mapa de memoria.

La interfaz requiere de un cierto número de direcciones para ser asignadas a sus diferentes opciones, estas opciones son:

- 1) Señales analógicas de salida.
- 2) Señales digitales de entrada - salida.
- 3) Conteo de eventos externos.

El número de localidades necesarias para cada opción es :

- 32 localidades para los canales analógicos.
- 4 localidades para canales digitales.
- 4 localidades para el conteo de eventos externos.

La figura 4.5 muestra el mapa direcciones que utiliza la computadora; en éste podemos observar que las localidades 0000H - 01FFH se encuentran reservadas para el sistema operativo de la computadora y las localidades 0200H - 03FFH están disponibles para el usuario.

0000H		DISPONIBLE PARA LA TARJETA MADRE.
01FFH	512	
0200H		DISPONIBLE PARA EL USUARIO.
03FFH	512	
0400H		
	64512	SIN USO.
FFFFH		

FIG. 4.5 ESPACIO DE DIRECCIONES E/S DE LA COMPUTADORA.

Entre las localidades 0200H - 03FFH algunas se encuentran reservadas para los diversos dispositivos de la computadora, sólo ciertas direcciones se encuentran disponibles para el uso de dispositivos externos, como se muestra en la fig 4.6. De estas localidades se han escogido las que indica la figura 4.7.

En este diseño la decodificación de direcciones se ha realizado de tal forma que no sólo se restringe a un solo rango de direcciones, es decir, éstas se pueden variar siempre y cuando se encuentren disponibles para el usuario según la tabla de la figura 4.6.

4.3 Lectura o escritura en la interfaz.

La lectura y escritura de la tarjeta se realiza por medio de instrucciones en lenguajes Pascal ó Ensamblador. Estas manejan palabras de 16 bites, compuestas por dos octetos; es decir, primero leen o cargan los 8 bites menos significativos y en seguida los 8 bites más significativos.

0200H	1	LIBRE
0201H	1	ADATADOR DE JUEGO DE CONTROL
0202H - 0277H	118	LIBRES
0278H - 027FH	8	SEGUNDO PUERTO PARA IMPRESORA
0280H - 02F7H	120	LIBRES
02F8H - 02FFH	8	SEGUNDO PUERTO PARA IMPRESORA
0300H - 0377H	120	LIBRES
0378H - 037FH	8	PUERTO PARA IMPRESORA
0380H - 03AFH	48	LIBRES
03B0H - 03BFH	16	MONITOR MONOCROMATICO
03C0H - 03CFH	16	LIBRES
03D0H - 03DFH	16	GRAFICAS A COLOR
03E0H - 03EFH	16	LIBRES
03F0H - 03F7H	8	DRIVER PARA DISKETTE DE 5 1/4
03F8H - 03FFH	8	PUERTO SERIE

FIG. 4.6 Localidades disponibles en la computadora

DIRECCION	USO
0300H - 031FH	CANALES ANALOGICOS
0320H - 0324H	CANALES DIGITALES
0328H - 032BH	CONTABILIZACION DE EVENTOS

fig 4.7 Localidades asignadas en la interfaz.

La interfaz contiene la característica de poder almacenar información para cada uno de los dieciséis canales analógicos de salida, veinticuatro canales digitales, dieciseis para escritura y ocho canales para lectura; el conteo de eventos externos se realiza por medio de tres contadores (Timer) programables.

5 DESCRIPCION DE LOS CIRCUITOS ELECTRONICOS.

5.1 Introducción

En el presente capítulo se presenta el diseño de los circuitos electrónicos de la interfaz y se indica la interrelación entre ellos. La interfaz esta compuesta por los siguientes circuitos:

- 1) Decodificación de direcciones y señales de control.
- 2) Lectura y escritura en memoria local.
- 3) Conversión Digital/Analógica.
- 4) Secuenciador.
- 5) Retención.
- 6) Integrador y puertos paralelos.
- 7) Acoplamiento para señales analógicas.
- 8) Acoplamiento para señales digitales.

La interfaz se compone de dos módulos. El primero (módulo de conversión) se encuentra localizado dentro de una computadora personal, insertado en una ranura; éste módulo constituye la tarjeta de conversión. El segundo módulo es el de conexiones de la tarjeta de conversión con dispositivos externos y se encuentra fuera de la computadora, se conecta al módulo de conversión a través de un cable de listón.

El primer módulo se encuentra constituido por los primeros seis circuitos eléctricos que arriba se mencionan, el segundo módulo lo constituyen los dos últimos circuitos.

A continuación se describe el funcionamiento de cada uno de los circuitos que constituyen la tarjeta interfaz.

5.2 Circuito de decodificación de direcciones y señales de control.

Esta etapa se encuentra dividida en tres partes: la que corresponde a la selección de los canales analógicos de salida, selección de los canales digitales de Entrada/Salida, y la selección del contador de eventos externos.

En la figura 5.1 se muestra el diagrama eléctrico correspondiente a la decodificación, indicando las señales que ésta involucra y la interrelación entre ellas.

La decodificación se realiza comparando las direcciones del bus de la Computadora Personal con la programada en la interfaz; esto lo realiza el circuito integrado 74LS85 (U5). Este circuito es un comparador de 4 bits que compara las direcciones A6-A9 con las de un microinterruptor programable (SW1). El resultado de la comparación es una señal que indica que uno de los tres dispositivos ha sido seleccionado, de tal forma que con ella se permite recibir o mandar información a la interfaz; esta señal se indica como DS ($DS = (A=B) * \overline{AEN}$) *direccion seleccionada*, con A5 alta indica que se han seleccionado los canales digitales y con A5 baja se seleccionan los canales analógicos. Cuando se selecciona los canales digitales se activa la señal DIS ($DIS = \overline{A5} + \overline{A=B} + AEN$) *seleccion digital* y para los canales analógicos se activa la señal AS ($AS = A5 + (\overline{A=B}) + AEN$) *seleccion analógica*.

Estos dispositivos son seleccionados siempre y cuando la señal AEN (*access enable*) se encuentre inactiva (cero lógico).

Si se presenta la señal de selección para los canales analógicos, esto indica que se ha presentado una señal de escritura por medio de la línea \overline{IOW} (*input output write*) o una lectura por medio de la línea \overline{IOR} (*input output read*). Estas

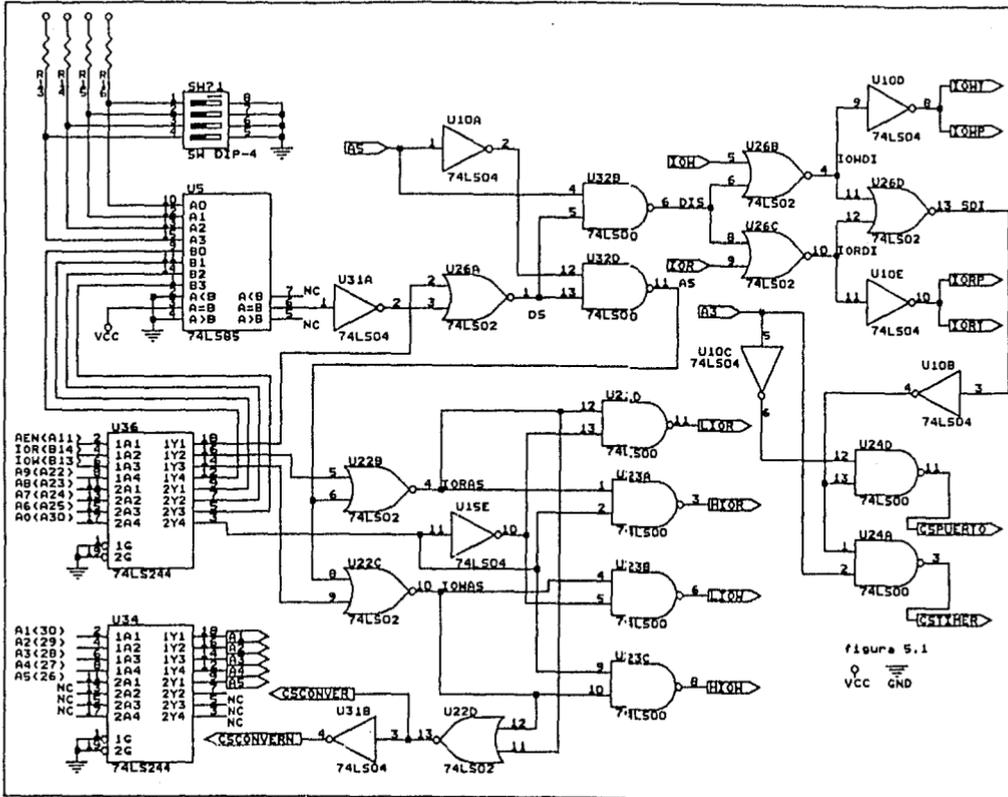


figure 5.1
 □ VCC
 □ GND

líneas de lectura o escritura se presentan sólo una a la vez. Cuando se presenta una escritura para un canal analógico, la línea \overline{IOW} se encuentra activa y se presenta con dos pulsos, el primer pulso indica que se envían los ocho bits menos significativos y el segundo los ocho bits más significativos, con lo cual se logra tener una palabra de 16 bits. Lo mismo sucede cuando se presenta una lectura de la interfaz.

La combinación booleana de las señales $A0$ y \overline{IOW} permite obtener un pulso de sincronía para los bits menos significativos y otro para los bits más significativos. Estos se indican con las señales $LIOW$ ($LIOW = A5 + (\overline{A=B}) * AEN + \overline{IOW} + A0$) y $HIOW$ ($HIOW = A5 + \overline{A=B} + AEN + \overline{IOW} + \overline{A0}$). La combinación de las señales $A0$ y \overline{IOR} permite obtener las señales $LIOR$ ($LIOR = A5 + (\overline{A=B}) + AEN + \overline{IOR} + A0$) y $HIOR$ ($HIOR = A5 + (\overline{A=B}) + AEN + \overline{IOR} + \overline{A0}$).

Si se presenta la señal de selección DIS ($DIS = \overline{A5} + (\overline{A=B}) + AEN$) para los canales digitales, ésta indica que hay un ciclo de lectura o escritura en ellos. Para seleccionar los canales Digitales Entrada/Salida o los Contadores de Eventos Externos se hace uso de la línea de dirección $A3$ la cual indica qué dispositivo ha sido seleccionado; con la combinación de ella se obtienen las señales $CSPUERTO$ ($CSPUERTO = A3 + \overline{A5} + (\overline{A=B}) + AEN + (\overline{IOW} * \overline{IOR})$) selección al puerto y la señal $CSTIMER$ ($CSTIMER = \overline{A3} + \overline{A5} + (\overline{A=B}) + AEN + (\overline{IOW} * \overline{IOR})$) selección al timer. Por medio de la combinación de \overline{IOR} y DIS se realiza la lectura a los canales digitales Entrada/Salida, esta señal se indica con $IORP$ ($IORP = \overline{IOR} + \overline{A5} + (\overline{A=B}) + AEN$) lectura al puerto y la del contador de eventos externos se indica con $IORT$ ($IORT = \overline{IOR} + \overline{A5} + (\overline{A=B}) + AEN$) lectura al timer. De la combinación de \overline{IOW} y DIS se obtiene la señal de escritura para los canales digitales Entrada/Salida, indicada con $IOWP$ ($IOWP = \overline{IOW} + \overline{A5} + (\overline{A=B}) + AEN$) escritura al puerto, o al contador de eventos externos indicada con $IOWT$ ($IOWT = \overline{IOW} + \overline{A5} + (\overline{A=B}) + AEN$) escritura al timer.

Todas las líneas de datos, dirección y control pasan por un circuito que impide sobrecargar las señales eléctricas del CPU, este circuito es 74LS244 el que es un buffer de tres estados.

Para los datos se ocupa el circuito integrado U1, que se encuentra localizado en la figura 5.7, para las direcciones el circuito integrado U34, para las señales de control el circuito integrado U36 estos se encuentra ubicados en la figura 5.1.

A la interfaz se han asignado las siguientes localidades de memoria:

Para los canales analógicos las localidades 0300H hasta 031FH. Las direcciones A0 a la A4 seleccionan los diferentes registros.

A los canales digitales las localidades 0320H - 0323H, las direcciones A0 y A1 seleccionan los registros internos del puerto.

Corresponden al contador de eventos externos las localidades 0328H - 032CH; las direcciones A0 y A1 seleccionan los registros del contador.

Las figuras 5.2 y 5.3 muestran los diagramas de tiempo para generación de señales de lectura y escritura en el puerto y timer respectivamente, las figuras 5.4 y 5.5 muestra la generación de señales de sincronía para la escritura y lectura del convertidor.

5.3. Circuito de Lectura y Escritura en memoria local.

Esta parte de la interfaz realiza el intercambio de información entre la computadora personal y la interfaz es decir, permite la lectura o escritura en la memoria local. En la figura 5.6 se muestra un diagrama de bloques de la memoria local.

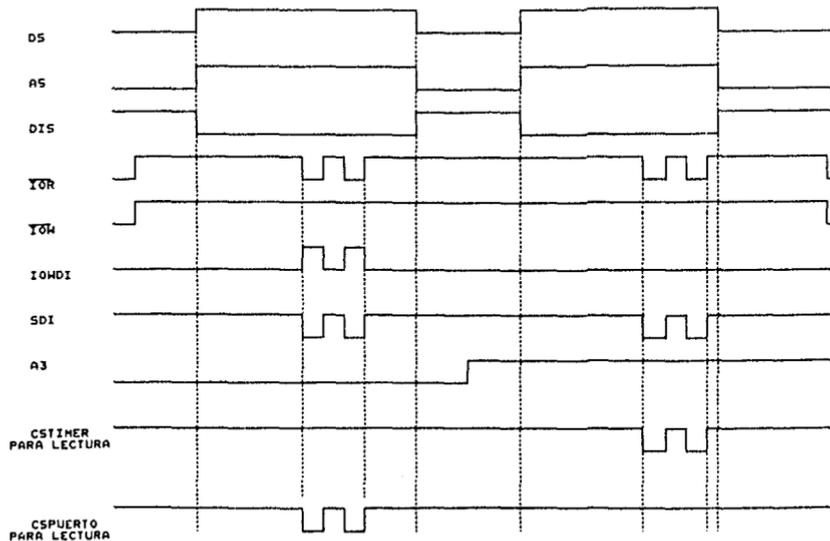


FIGURA 5.2

DIAGRAMA DE TIEMPO SOBRE SELECCION DE LOS DISPOSITIVOS TIMER Y PUERTO DE LECTURA			
INSTITUTO DE INGENIERIA TAMPARO		NO.	REV.
UNAH		A	1.1
Proy. 0120 Fecha: 13 / 1 / 1989 Hoja 1 de 4			

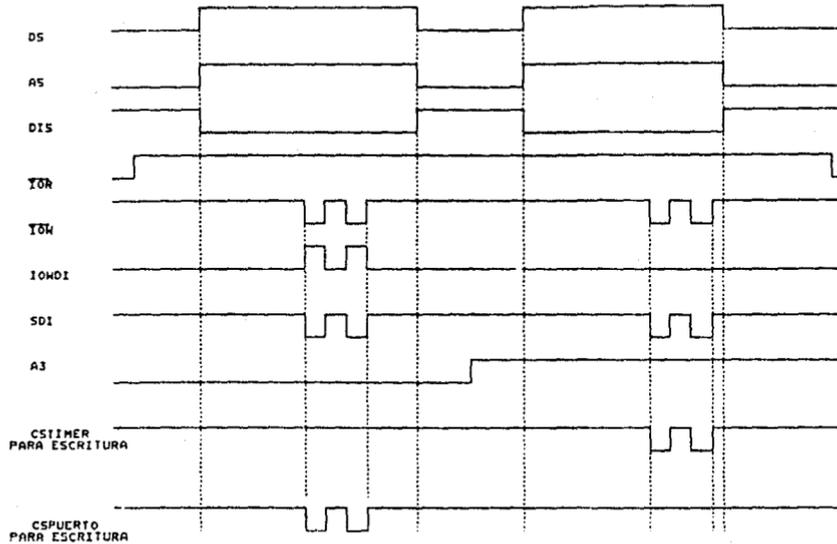


FIGURA 5.3

DIAGRAMA DE TIEMPO SOBRE SELECCION DE LOS DISPOSITIVOS TIMER Y PUERTO (ESCRITURA)			
INSTITUTO DE INGENIERIA	Tamaño	NO.	REV
UNAM	A	1	1.1
Proy. 0120 Fecha: 13 / 3 / 1989 Hoja 2 de 4			

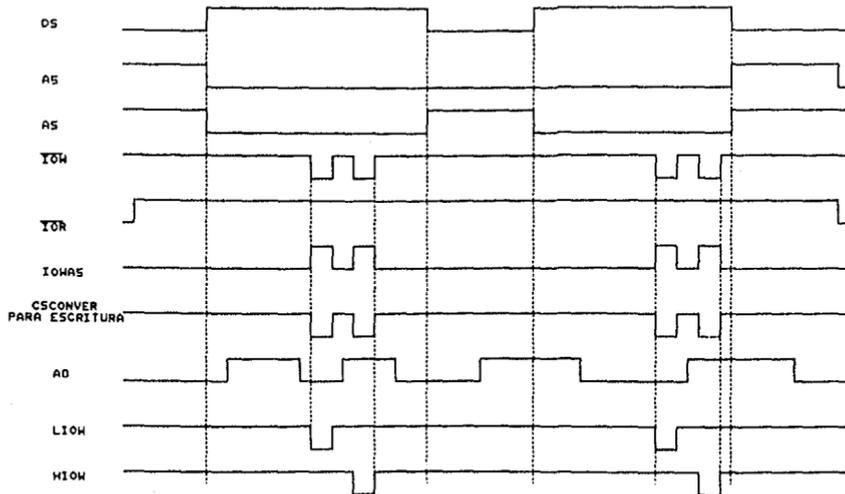


FIGURA 5.4

DIAGRAMA DE TIEMPO SOBRE SELECCION DE EL CONVERTIDOR (ESCRITURA)			
INSTITUTO DE INGENIERIA	Tamaño	NO.	REV
UNAM	A	1	1.1
PROM. #120 / mschalln / 3 / 1989 / Hoja 4 de 4			

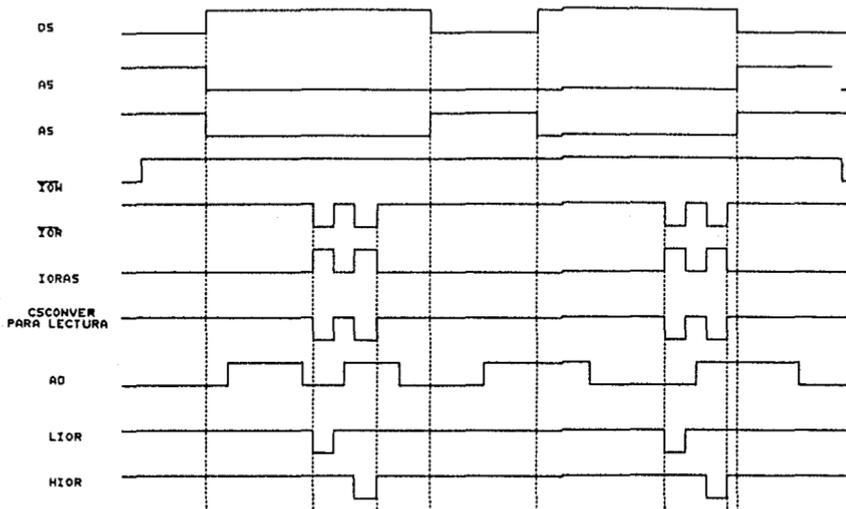
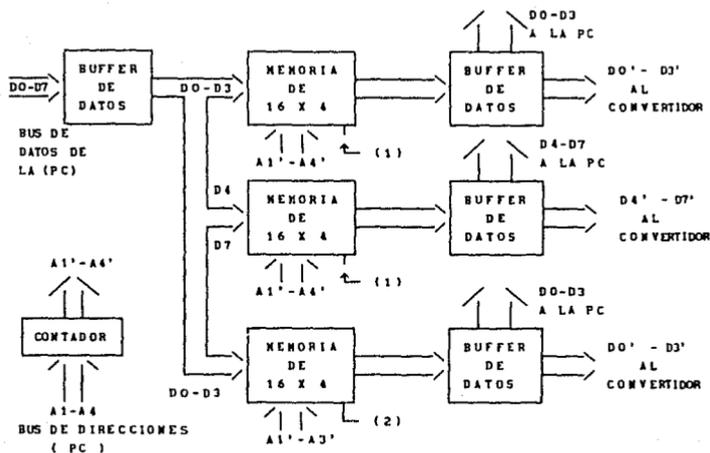


FIGURA 5.5

DIAGRAMA DE TIEMPO SOBRE SELECCIÓN DE EL CONVERTIDOR (LECTURA)			
INSTITUTO DE INGENIERIA UNAM	Tamaño A	Nº. 1	REV 1.1
Prov. 0120 Fecha: 13 / 3 / 1989 Hoja 3 de 4			



- (1) SEÑAL DE ESCRITURA BITS MENOS SIGNIFICATIVOS
 (2) SEÑAL DE ESCRITURA BITS MAS SIGNIFICATIVOS

FIG 5.6 Diagrama general para una escritura o lectura en la memoria local.

En la figura 5.7 se muestra el diagrama eléctrico del arreglo de la memoria junto con los *buffer* del bus de datos. Está constituido por las memorias RAM (*random access memory*) U3, U8, U13; cada una de ellas contiene 16 registros de 4 bites. Estas memorias son circuitos 7489, RAM, tienen cuatro líneas de datos (D0 - D3), cuatro líneas de dirección (A0 - A3), cuatro líneas de

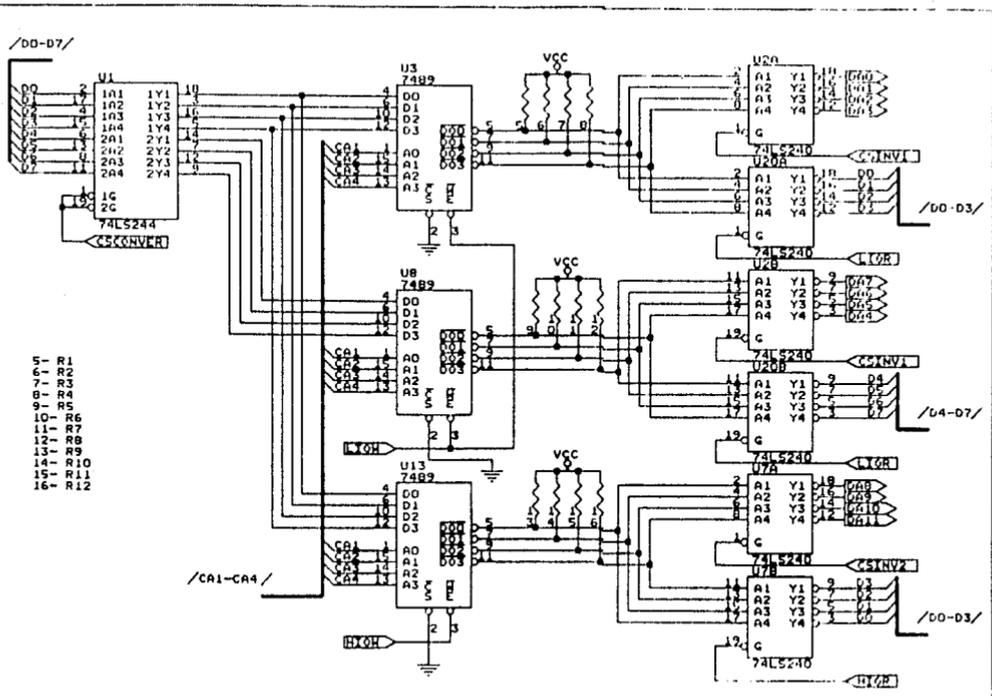


FIGURA 5.7

CHD VCC
 0

APAREJO DE LA MEMORIA			
INSTITUTO DE INGENIERIA	Tamaño	NO.	REV
UNAM	a	1	1.1
From 012 Fecha: 11 / 5 / 1989 2 de 10			

salida de datos (DO0 - DO3). Las salidas son de lógica negativa con colector abierto, por lo que se colocan resistencias pull-up para cada una de las salidas (R1-R12). Las memorias se controlan con las líneas \overline{CS} y \overline{WE} ; la señal \overline{CS} (habilitación del chip) indica que el dispositivo ha sido seleccionado cuando es activa baja, en caso contrario el dispositivo no es seleccionado y las salidas se encuentran en estado lógico 'uno'. La señal \overline{WR} (write/read) activa la escritura en la memoria, es activa baja. La lectura se realiza cuando \overline{WR} está en un 'uno' lógico. Las memorias se conectan en serie para formar palabras de 12 bites de longitud.

Para escribir en la interfaz se emplean las señales LIOW y HIOW generadas en el circuito de decodificación. La señal LIOW que permite escribir los bites menos significativos, esta señal activa la señal de escritura \overline{WR} en las memorias U3 y U8. Después se presenta la señal HIOW que activa \overline{WR} en la memoria U13, cargando los bites más significativos.

Para realizar una lectura se generan las señales LIOR y HIOR, que permiten obtener información de las memorias; activando la señal de control G activa el tercer estado de los circuitos U20A, U20B y U7A (74LS240).

LIOR activa la señal de habilitación G en los circuitos U20A y U20B, de donde se obtienen los primeros 8 bites menos significativos, y la señal HIOR activa la señal de habilitación G del circuito U7B para obtener los 4 bits más significativos. Las salidas de los circuitos U20, U20B y U7B (D0 - D7) van directamente al bus de la computadora. Cuando la señal de habilitación G no es seleccionada en los circuitos U2A, U20B y U7B, sus salidas se encuentran en alta impedancia.

Las memorias siempre se encuentran habilitadas, es decir \overline{CS} se encuentra en 'cero' lógico y la señal \overline{WR} se encuentra en estado

alto excepto cuando se realiza una lectura en la memoria.

Cuando no se realiza una lectura o escritura en los registros de los canales analógicos, el circuito secuenciador realiza la lectura de cada uno de los registros de la memoria de la interfaz para realizar la conversión de este dato y obtener un voltaje analógico de salida. De esta forma se leen los 16 registros y se repite la secuencia. Este circuito secuenciador se describirá posteriormente (sección 5.5).

En las figuras 5.4 y 5.5 muestran los diagramas de tiempos especificando en que momento se puede realizar una lectura o escritura en cada uno de los registros. A0 indica cuando se tiene en el bus el octecto más significativo (A0=1) o el menos significativo (A0=0), cuando A0 es igual a cero todo el tiempo, la longitud de la palabra es de ocho bites.

5.4 Conversión analógica.

Esta sección está formada por un convertidor digital-analógico y dos amplificadores operacionales que realizan la conversión de corriente a un voltaje analógico a la salida.

El convertidor digital-analógico (U2 de la figura 5.8) es el circuito integrado DAC1230. Este convertidor es de 12 bits de resolución, contiene ocho líneas de entrada de datos, cinco líneas de control y cuatro líneas de salida para la construcción del voltaje analógico.

A continuación se describirá el funcionamiento de cada señal de salida o entrada de éste circuito (figura 5.9)

a) Líneas de entrada de datos. Estas llevan el dato a convertir a los registros temporales 1 y 2, el primero almacena

los bits D4 - D11 y el segundo almacena los bits D0 - D3 completando así los 12 bits necesarios para realizar la conversión del dato.

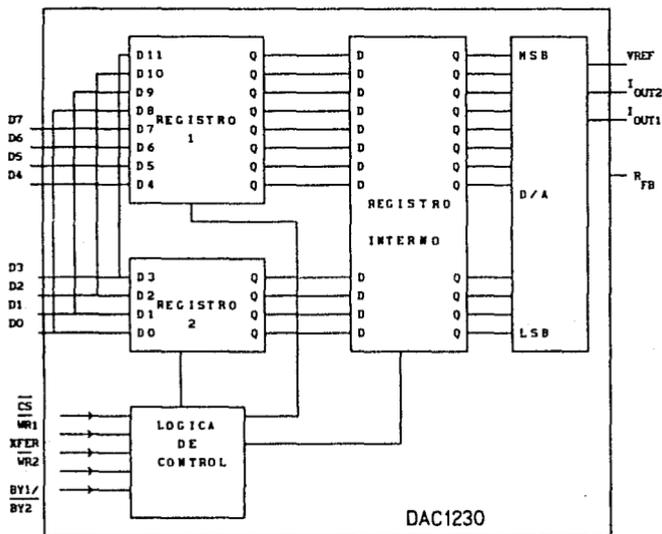


FIG. 5.9 Diagrama interno del convertidor.

b) Las líneas de control son:

CS: Habilita al convertidor, ésta señal es activa con un 'cero' lógico e inactiva con 'uno' lógico; cuando se encuentra inactiva no se realiza conversión alguna.

$\overline{WR1}$: Permite la escritura en el primer registro, correspondiente a D4 - D11. Esta señal es activa baja.

$\overline{WR2}$.- Permite la escritura en el segundo registro, correspondiente a D0 - D3. Esta señal es activa baja.

$\overline{BY1}/\overline{BY2}$: Selección de los registros internos del convertidor; cuando es alta, habilita el primer registro y cuando es baja, habilita el segundo registro.

XFER: Realiza la transmisión del dato capturado en los registros al registro interno de 12 bits. Al terminar la captura en el registro interno, se realiza automáticamente la conversión.

c) Las líneas que intervienen en la construcción de voltaje analógico a la salida de convertidor son:

VREF: Voltaje de referencia, necesario para polarizar al árbol de resistencias interno (R - 2R) del convertidor.

I_{out2} : Referencia de corriente del convertidor, se conecta a la tierra del sistema.

I_{out1} : Salida de corriente del convertidor, esta salida indica un valor de corriente proporcional al dato convertido. El valor de la corriente se rige por la siguiente fórmula:

$$I_{out1} = \frac{(V_{ref} (1 - 1/4096))}{r_{fb}} \quad (5.1)$$

r_{fb} = resistencia de retroalimentación.

V_{ref} = Voltaje de referencia, en el cual puede contener una

variación máxima entre 10 volt a -10 volt .

En la figura 5.8 muestra el diagrama electrónico del convertidor digital - analógico junto con los amplificadores operacionales.

La conversión se realiza bajo la siguiente secuencia:

La señal \overline{CS} es activada con $CSCONVERN$, $\overline{WR1}$ del convertidor (U2) se activa con la señal $WR1$ que presenta un pulso activo bajo mientras que $BYTE-XFER$ se encuentra en estado alto para la realización de la escritura en el primer registro; $\overline{WR2}$ del convertidor se activa con la señal $\overline{WR2}$ y presenta un pulso activo bajo mientras que $BYTE-XFER$ se encuentra en estado bajo para que se realice la escritura en el segundo registro, al momento que se realiza esta escritura la señal \overline{XFER} es activada con un nivel bajo lo que indica que los datos han sido transferidos al registro interno del convertidor, después de lo cual se realiza la conversión.

Como resultado de la conversión se obtiene una corriente proporcional al número binario, esta corriente se obtiene de la línea I_{out1} . Esta señal se encuentra conectada a la entrada no inversora del circuito amplificador U14 de la figura 5.8. La configuración de U14 realiza la conversión de corriente a voltaje (0 V. a $-V_{ref}$). El corrimiento (*offset*) de la señal se controla por medio del potenciómetro POT1. De la segunda parte obtiene un voltaje bipolar comprendido entre $+V_{ref}$ a $-V_{ref}$ por medio de la configuración sumadora del amplificador U19 de la figura 5.8.

A continuación se presenta un análisis de esta configuración:

$$I_1 = \frac{V_o'}{R14 + POT2}$$

$$I_2 = \frac{V_{ref}}{R20 + POT1}$$

si: $I_o = I_1 + I_2$

$$V_{sal} = \frac{(V_o') R17}{R14 + POT2} + \frac{(V_{ref}) R17}{R20 + POT1}$$

De tal forma que:

$$R14 + POT2 = 5K$$

$$R17 = 10K$$

$$R20 + POT1 = 10K$$

se obtiene:

$$V_o = -(2 V_o' + V_{ref}) \dots\dots\dots(5.2)$$

Donde

V_o = Voltaje de salida.

V_o' = Voltaje de salida monopolar.

V_{ref} = Voltaje de referencia.

Como se observa en la formula 5.2, las resistencias deben ser iguales, de ahí la necesidad de incluir los potenciómetros POT1 y POT2 para obtener la precisión requerida.

Una parte esencial para la construcción del voltaje analógico de salida es el voltaje de referencia (V_{ref}), cuyo circuito (figura 5.10) tiene las siguientes características:

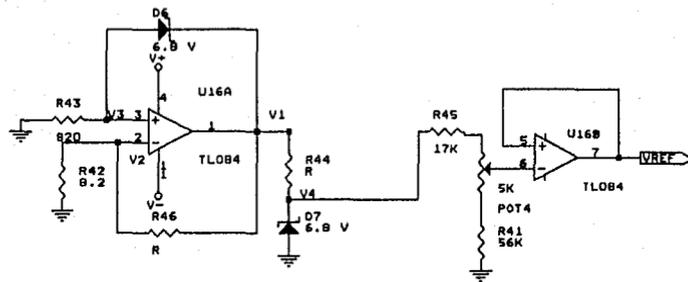


FIGURA 5.10

VCC GND

- i) Regulación de línea de 0.08%
- ii) Regulación de carga de 0.06%

El voltaje de referencia se obtiene por medio de dos amplificadores operacionales (TL084). El primer amplificador tiene la función de regular el voltaje de la línea, el cual funciona de la siguiente manera:

$$V_1 = A (V_3 - V_2)$$

$$V_3 = V_1 - V_z$$

$$V_z = \frac{R_{42}}{R_{42} + R_{41}} V_1$$

$$\text{Si } V_3 = V_z$$

$$V_1 - V_z = 1 - \frac{R_{42}}{R_{42} + R_{41}} * V_1$$

$$V_1 = V_z * \frac{R_{42} + R_{41}}{R_{41}}$$

Como V_1 es aproximadamente V_4

El voltaje de referencia queda dado por la formula:

$$V_{ref} = \frac{POT4 + R_{46}}{POT4 + R_{46} + R_{45}} * V_1$$

Primeramente presenta una configuración de un regulador de voltaje estable, permitiendonos obtener un voltaje aproximado de 6.8 Volts.

Esta configuración puede obtener variaciones entre 10 Volts y 14 Volts en la alimentación del circuito.

El segundo amplificador está en configuración de seguidor. Esta configuración proporciona una corriente de 0 mA. hasta 20 mA. con un voltaje de referencia fijo. El potenciómetro POT4 sirve para ajustar el voltaje de referencia con un error de .1 milivolts con lo que se logra la precisión deseada. El voltaje de referencia para el convertidor tiene un valor nominal de 5.000 Volts.

Al voltaje convertido (salida del amplificador U19 de la figura 5.9) se le denomina ANALOGICA. Esta señal tiene una variación determinada por la fórmula siguiente:

$$V_{out} = V_{ref} ((D - 2048) / 2048)$$

Donde

D es el valor del voltaje en una escala de 0 a 4095 del voltaje. De esta manera se realiza la conversión para obtener el voltaje analógico de salida correspondiente. D toma valores en el siguiente rango:

$$0 < D < 4095 \text{ siendo } \begin{cases} -5V, & D = 0 \\ 0V, & D = 2047 \\ +5V, & D = 4095 \end{cases}$$

5.5 Secuenciador.

En este circuito se generan las señales de control. Estas señales activan los diferentes dispositivos de la interfaz. El circuito está formado por cuatro secciones a saber:

- a) Temporizador.
- b) Decodificador.
- c) Contador.
- d) Lógica de control.

El circuito electrónico del secuenciador se muestra en la figura 5.11. La sección del temporizador se encuentra formada por un divisor de frecuencia (tres biestables JK en configuración toggle U6A, U6B y U11B), la entrada al divisor de frecuencia es la señal de reloj de la computadora.

El secuenciador genera ocho pulsos para la sincronía de actividades de la interfaz, para ello los decodificadores del circuito U9 se conectan en serie para generar esos pulsos.

Las salidas del divisor de frecuencia (Q1, Q2 y Q3) se conectan a las entradas de U9, (éste circuito está formado por dos decodificadores de cuatro salidas seleccionadas por dos líneas).

Las salidas de U9 son ocho pulsos bajos que se generan en forma secuencial (1Y0, 1Y1, , 2Y3), de allí el nombre de secuenciador.

Los circuitos correspondientes al divisor de frecuencia y decodificación son habilitados cuando la señal CSCONVERN es activa baja; esto sucede cuando no hay, por parte de la PC, un

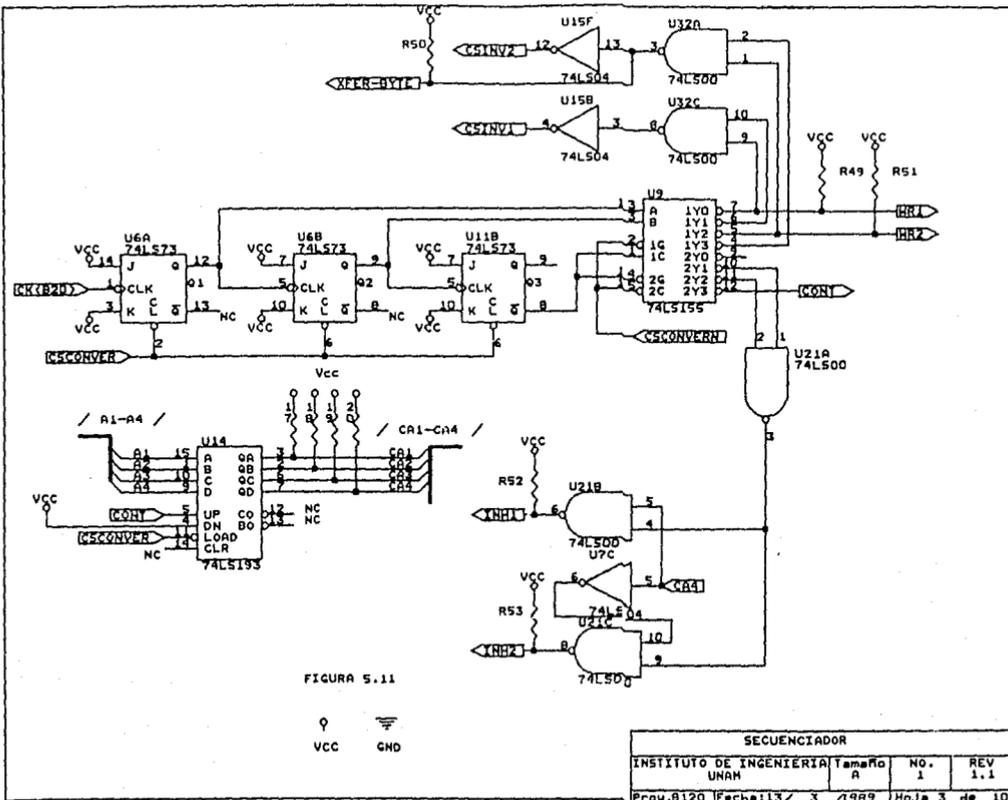


FIGURA 5.11

 VCC
 GND

SECUENCIADOR			
INSTITUTO DE INGENIERIA UNAH	Tamaño A	NO. 1	REV 1.1
Proy. B.120 Fecha: 13 / 3 / 1989 Hoja 3 de 10			

acceso en los registros de los canales analógicos de salida.

Al momento en que se realiza una escritura o lectura en uno de los canales analógicos de salida, la pata 11 (load) del contador U14 se encuentra habilitado con la señal CCONVER para realizar una carga en paralelo. CCONVER presenta dos pulsos activos bajos: el primer pulso habilita la señal de escritura en las memorias U3 y U8 de la figura 5.7, el segundo pulso habilita la señal de escritura en la memoria U13 de la misma figura. Las salidas del contador U14 se conectan directamente a las líneas de dirección de cada una de las memorias U3, U8 y U13 de la figura 5.7.

Cuando la PC no accesa las memorias del convertidor, el secuenciador se encuentra en un ciclo continuo de refrescamiento de voltaje en la etapa de salida de la interfaz.

A continuación se presentan la secuencia de señales generadas en el secuenciador.

Primero se genera la señal CSINV1, figura 5.11, ésta es activa baja con una duración de 200 nanosegundos, habilita la señal G de los buffer inversores U2A y U2B de la figura 5.1, con lo que se activa el paso de la información de las memorias a los registros del convertidor; se lee la parte menos significativa del dato, al mismo tiempo se genera la señal $\overline{WR1}$ que activa la escritura en el circuito DAC1230 ($\overline{WR1}$ y $\overline{WR2}$) de la figura 5.9 $\overline{WR1}$ activa la escritura, la que se realiza cuando se presenta el flanco de subida. Con el segundo pulso de Q1 se genera la señal CSINV2; ésta es activa baja con una duración de 200 nanosegundos, habilita la señal G del buffer inversor U7 de la figura 5.1. De esta manera se lee la parte más significativa del dato a convertir, al mismo tiempo se genera la señal $\overline{WR2}$, que activa $\overline{WR2}$ del convertidor U2. $\overline{WR2}$ activa la escritura cuando se presenta el

flanco de subida. La señal XFER-BYTE realiza la transferencia de información de los registros internos del convertidor al registro de conversión para luego dar inicio a la conversión.

En seguida se presenta la señal CONT, que es activa baja, la que incrementa el contador U14 de la figura 5.11, indicando que se ha terminado la secuencia de conversión de un canal analógico de salida y con ello se direcciona el siguiente canal analógico, El incremento se genera con el flanco de subida y así se inicia la conversión de un nuevo dato.

INH1 y INH2 del secuenciador habilitan los circuitos multicanal de la etapa de salida.

La figura 5.12 se muestra el diagrama de tiempos del secuenciador.

5.6 Circuito de retención.

Realiza la retención de los 16 canales analógicos de salida (CANALO - CANAL15); esto se lleva a cabo por medio de una multicanalización y circuitos retenedores de orden cero. El multicanal está constituido con los circuitos U17 y U18 de las figuras 5.13 y 5.14 respectivamente; estos circuitos contienen ocho canales analógicos con una sola línea de entrada/salida, 3 líneas de dirección para indicar el canal seleccionado y una línea de control (la que indica la habilitación del circuito). Contiene dos líneas de polarización V+, V- y la referencia GND.

La línea de alimentación V+ se configuró por medio de un regulador de voltaje con un diodo zener, este voltaje es de 6.8 Volts. De igual manera se configuró la polarización V- a -6.8 Volts.

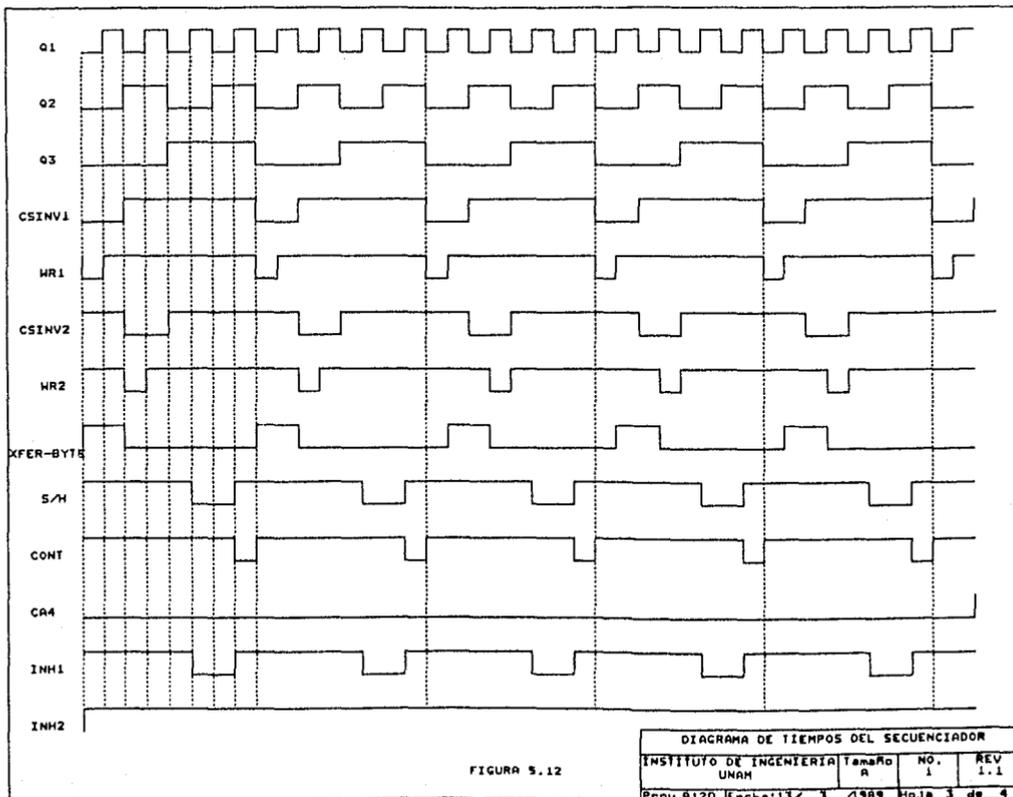


FIGURA 5.12

DIAGRAMA DE TIEMPOS DEL SECUENCIADOR				
INSTITUTO DE INGENIERIA UNAH	Tamaño A	Nº. 1	REV 1.1	
Proy. B120 Fecha: 13/3/1989 Hoja 3 de 4				

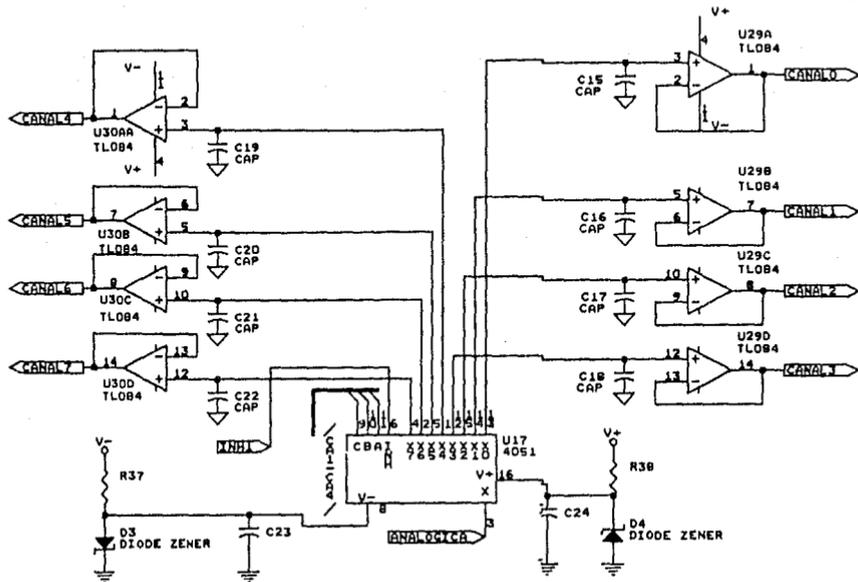
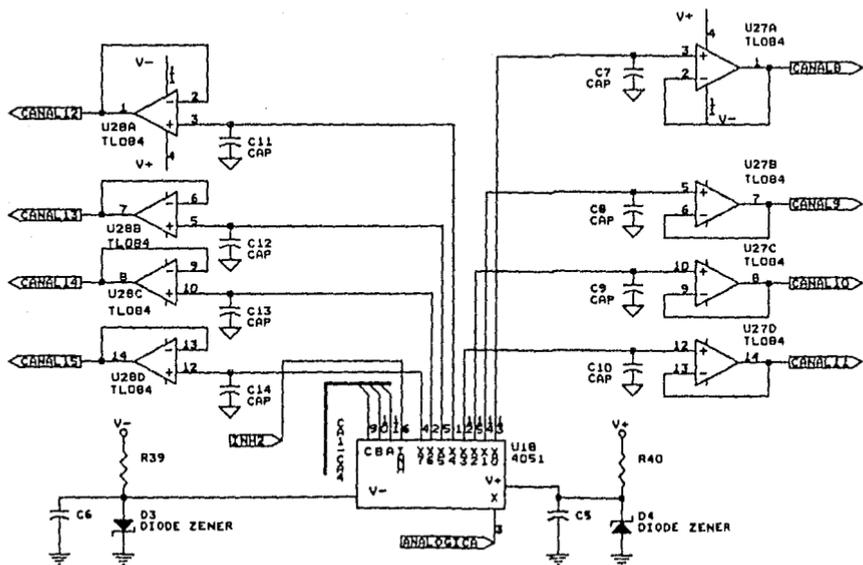


FIGURA 5.13

CANALES DE RETENCION 0 - 7			
INSTITUTO DE INGENIERIA	Tamaño	NO.	REV
UNAH	A	1	1.1
Proy. 0120 Fecha: 11/3 / 1999 Hoja 6 de 10			



CANALES DE RETENCION B - 15

INSTITUTO DE INGENIERIA UNAM	Tamaño A	NO. 1	REV 1.1
PRM.B12 Fecha: 13 / 3 / 1983 Hoja 7 de 10			

Las líneas de dirección (CA1-CA4) están conectadas al contador del secuenciador (U14 de la figura 5.11), estas líneas permanecen fijas mientras se realiza el muestreo.

La otra parte está constituida por un retén de orden cero que permite mantener constante el voltaje analógico para cada canal durante todo el ciclo del secuenciador; se encuentra formado por un amplificador operacional en configuración de seguidor y un capacitor en la entrada.

Una vez convertido el dato binario en una señal analógica equivalente, ésta pasa al retén, el cual mantiene ese valor hasta que se tome una nueva muestra para el mismo canal.

A continuación se muestra el cálculo del valor del capacitor optimo para el retenedor.

Considerando la gráfica 1 se busca un capacitor que permita encontrar una constante de tiempo igual, tanto para la carga del capacitor como la descarga.

Si la constante de tiempo para la carga es:

$$t_c = R_c * C$$

$$t_c = 3.3 \text{ useg.}$$

$$R_c = 50 \text{ ohms.}$$

El valor del capacitor optimo para la carga es :

$$C = 66 \text{ nf.}$$

Para la constante de descarga se tienen las siguientes condiciones:

$$t_d = R_d \cdot C$$

$$t_d = 49.5 \text{ useg.}$$

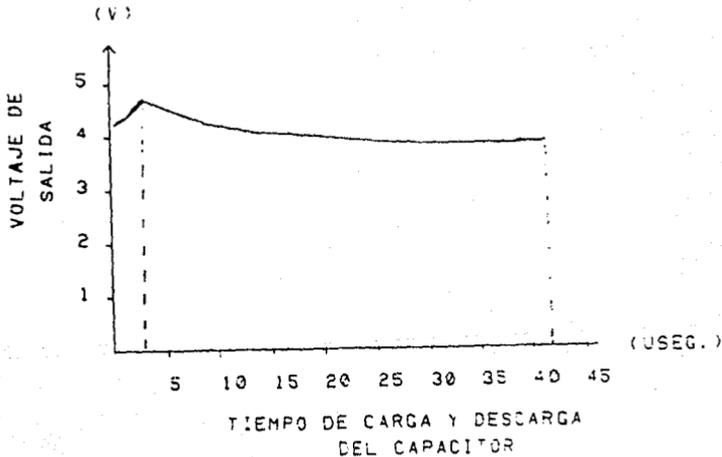
$$R_d = 10^{12} \text{ ohms}$$

El valor del capacitor optimo para la descarga es:

$$C = 4.95 \times 10^{-18} \text{ farads}$$

De tal forma que por medio de un proceso interactivo se logra obtener el valor optimo del capacitor que permite cumplir las condiciones de carga y descarga del voltaje, obteniendo el siguiente valor como el más cercano a estas condiciones:

$$C = 3.3 \text{ nf.}$$



GRAFICA 1

5.7 Circuito Integrador y Puerto Paralelo.

Esta parte se encuentra formada por un dispositivo de entrada/salida binaria en paralelo y un contador de eventos externos.

El dispositivo de entrada/salida está realizado con el puerto U39 de la figura 5.15, que es el circuito integrado 82C55 (PPI Programable Paralell Interface). Con él se lleva a cabo la comunicación de la interfaz con dispositivos entrada/salida que manejen señales binarias.

El 82C55 es un circuito integrado realizado con tecnología CMOS. Contiene 24 líneas de entrada/salida para las que se pueden definir diferentes configuraciones. Hay tres modos de transferencia de datos:

Modo 0 - Entrada/Salida básica.

Modo 1 - Entrada/Salida habilitada.

Modo 2 - Bus bidireccional

Las 24 líneas se agrupan en dos grupos de 8 bites y dos de 4 bites. En el modo 0 se tienen opciones para programar las 24 salidas. Los datos se envían o reciben por ellas y no hay control de hand shake. En el modo 1, los dos grupos de 4 bites sirven de control mientras los otros dos octetos sirven para intercambio de datos. En el modo 2 se pueden enviar o recibir el octeto de datos en función de las líneas de control.

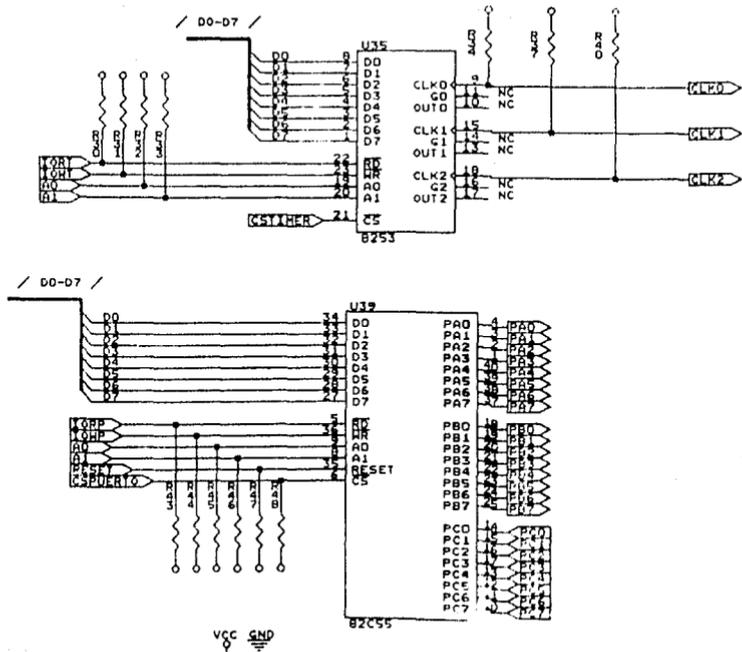


FIGURA 5.15

DIAGRAMA ELECTRONICO DEL PPURT Y EL CONTADOR			
INSTITUTO DE INGENIERIA	Tema No.	NO.	REV
UNAM	A	1	1-1
TECNOLOGIA FACULTAD DE INGENIERIA Y 1989 HOJA 2 de 10			

La palabra de control permite establecer la configuración del puerto Entrada/Salida, esta palabra de control indica las siguientes características:

Agrupamiento Entrada/Salida.
Dirección de los puertos.
Asignaciones de control.
Asignaciones de estado.

Las señales que manejan el circuito 82C55 son las siguientes:

D7 - D0	Bus de Datos (Bidireccional).
RESET	Línea que inicia los puertos.
CS	Habilitación del chip.
RD	Lectura al puerto.
WR	Escritura al puerto.
A0,A1	Líneas de control.
PA7 - PA0	Salida/Entrada, puerto A.
PB7 - PB0	Salida/Entrada, puerto B.
PC7 - PC0	Salida/Entrada, puerto C.

En la interfaz los puertos A y B se han configurado como salidas, el puerto C como entrada. A la salida o entrada de cada uno de ellos se colocaron buffers inversores con la finalidad de proteger al circuito de las señales externas.

En la figura 5.16 se muestra el diagrama eléctrico de los circuitos de amplificación de potencia (buffers).

La interfaz contiene 2 conectores que sirven de comunicación entre ella y el módulo de conexiones.

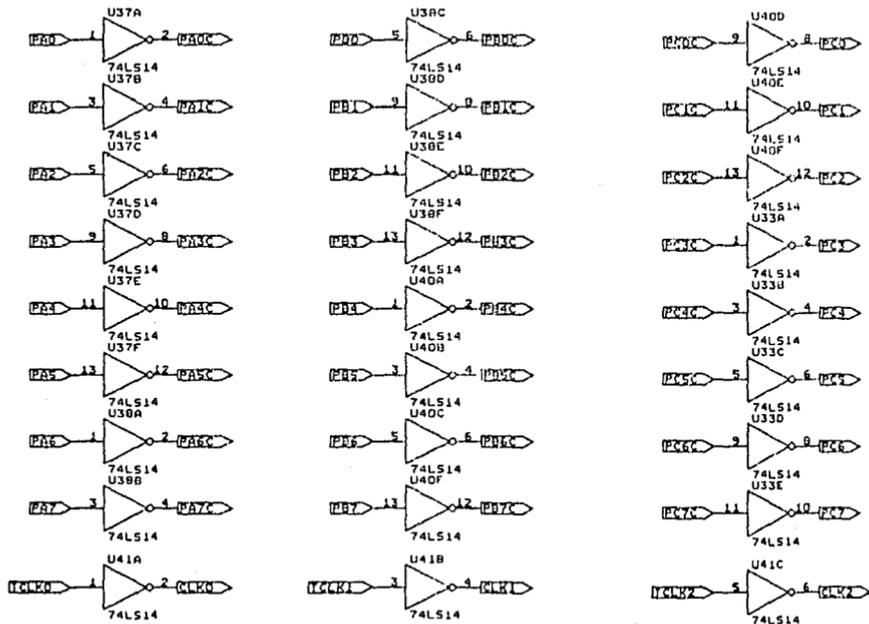


FIGURA 5.16

ACOMPLIMIENTO DIGITAL			
INGENIERIA			
INSTITUTO DE INGENIERIA	UNAH	NO. 1	REV 1.1
Proy. 0120	Fecha: 11/3/1999	Hoja 9	de 10

La segunda parte se encuentra formada por tres contadores, integrados en un solo *chip* (U35), de eventos externos como se indica en la figura 5.15.

U35 es un Contador/Timer programable (82C53) diseñado específicamente para su uso en sistemas con microcomputadoras.

A continuación se describen las señales que presenta dicho circuito.

D7 - D0	Bus de Datos (Bidireccional).
CS	Habilitación del <i>chip</i> .
RD	Lectura al puerto.
WR	Escritura al puerto.
A0,A1	Líneas de control.
CLK0,CLK1,CLK2	Reloj para cada contador.
OUT0,OUT1,OUT2	Salida del contador.

Este circuito es activado bajo ordenes de un programa de aplicación específica para esta interfaz; primero se programa el registro de control contituido por las líneas A0, A1 y el bus de datos. Estas líneas establecen el modo de operación del contador y éstos son programados en forma independiente, permitiendo el funcionamiento para cada contador. Con la señal \overline{WR} y el bus de datos permite programar la palabra de control, con la señal \overline{RD} y el bus de datos permite la lectura para cada contador, esta lectura se puede realizar cuando el usuario lo necesite.

5.8 Módulo de conexiones.

Esta parte consiste en el enlace de la tarjeta interfaz con el modulo de conexiones , este modulo permite el acoplamiento de la interfaz con las señales del mundo exterior.

Este módulo contiene 16 buffers, uno para cada uno de los canales analógicos (estos buffers se encuentran especificados en el anexo C con su diagrama eléctrico); 26 buffers digitales pertenecientes a cada una de las señales de entrada/salida y de los contadores de eventos externos.

6. PROGRAMACION

6.1 Introducción.

La interfaz contiene únicamente el hardware que permite desarrollar las funciones que se mencionan en capítulos anteriores; para complementar su funcionamiento se requiere de un programa adecuado para ello, este software se realiza por medio de un lenguaje de programación de alto nivel. En este capítulo se describe la programación de los diferentes dispositivos que constituyen la interfaz como son: canales analógicos de salida, el contador de eventos externos y los puertos digitales entrada/salida. A continuación se describe el programa de presentación y uso desarrollado para esta interfaz.

Las tareas que realiza el programa son:

- Establecer la iniciación de los canales analógicos de salida.
- Programar el contador de eventos externos (palabra de control).
- Programar el puerto paralelo (palabra de control).

Los diferentes componentes de la tarjeta se programan a través de la escritura o lectura en un registro de memoria, cuya dirección han sido asignada previamente a la interfaz. A continuación se detallan las direcciones de los registros que contiene la interfaz y se especifica que función contiene cada uno de ellos.

0300H - 031FH	Canales Analógicos de salida.
0320H - 0324H	Canales Digitales Entrada/Salida.
0328H - 032BH	Contadores de Eventos Externos.

6.2 Iniciación de los canales analógicos de salida.

Los bits A0-A9 del bus de direcciones se utilizan para acceder los diferentes dispositivos de Entrada/Salida, una vez que se ha seleccionado el segmento en que se instala la tarjeta; de estos bits, los 5 más significativos (A5-A9) se utilizan para direccionar la tarjeta dentro de este segmento. Los cinco bits menos significativos (A0-A4) se utilizan para direccionar los diferentes canales analógicos de salida con que cuenta la interfaz.

La tabla 6.1 muestra la asignación de direcciones (A0-A4) para cada uno de los canales analógicos de salida.

La programación consiste en cargar un dato con el valor binario correspondiente al canal analógico seleccionado. Los datos tienen una longitud de 12 bits. Como se observa en la tabla, se emplean dos registros para la programación de cada canal analógico; en el primer registro se programan los bits menos significativos (D0-D7) y el segundo registro los bits más significativos (D8-D11).

A continuación se muestra el seguimiento que presenta el programa para realizar la programación de lectura de datos; iniciamos obteniendo por medio del uno de los puertos asignados el dato, como el dato viene en complemento a dos se invierte y se aplica la fórmula de conversión a voltaje, teniendo la siguiente

expresión:

$$\text{Voltaje a programar} = ((\text{dato}-2048)/2048)*5$$

Para la escritura de datos se sigue el siguiente procedimiento; se inicia con la conversión del voltaje a programar a el número binario por medio de la fórmula siguiente:

$$\text{Dato previo} = 2048 * (\text{voltaje a programar}/\text{voltaje de referencia})$$

$$\text{Dato a programar} = (2048 + \text{parte entera del dato previo})$$

6.3 Programación del Contador de Eventos Externos.

El contador de eventos externos utilizado en la interfaz está constituido por 3 contadores independientes y un registro de control. La función del contador de eventos externos es integrar los cambios externos binarios aleatorios o periódicos que hayan ocurrido durante un periodo de tiempo de una cierta señal discreta. El valor inicial de cada contador es cero.

Las actividades con estos contadores se realizan básicamente en dos partes:

- 1.- Se inician los contadores con cero y se establece el modo de operación.
- 2.-La lectura del valor del contador en el momento que se desee.

Las funciones anteriores se especifican utilizando las líneas de control de acuerdo a la Tabla 6.2.

DIRECCION					ASIGNACION DEL CANAL
A4	A3	A2	A1	A0	
0	0	0	0	0	CANAL 0 PARTE MENOS SIGNIFICATIVA.
0	0	0	0	1	CANAL 0 PARTE MAS SIGNIFICATIVA.
0	0	0	1	0	CANAL 1 PARTE MENOS SIGNIFICATIVA.
0	0	0	1	1	CANAL 1 PARTE MAS SIGNIFICATIVA.
0	0	1	0	0	CANAL 2 PARTE MENOS SIGNIFICATIVA.
0	0	1	0	1	CANAL 2 PARTE MAS SIGNIFICATIVA.
0	0	1	1	0	CANAL 3 PARTE MENOS SIGNIFICATIVA.
0	0	1	1	1	CANAL 3 PARTE MAS SIGNIFICATIVA.
0	1	0	0	0	CANAL 4 PARTE MENOS SIGNIFICATIVA.
0	1	0	0	1	CANAL 4 PARTE MAS SIGNIFICATIVA.
0	1	0	1	0	CANAL 5 PARTE MENOS SIGNIFICATIVA.
0	1	0	1	1	CANAL 5 PARTE MAS SIGNIFICATIVA.
0	1	1	0	0	CANAL 6 PARTE MENOS SIGNIFICATIVA.
0	1	1	0	1	CANAL 6 PARTE MAS SIGNIFICATIVA.
0	1	1	1	0	CANAL 7 PARTE MENOS SIGNIFICATIVA.
0	1	1	1	1	CANAL 7 PARTE MAS SIGNIFICATIVA.
1	0	0	0	0	CANAL 8 PARTE MENOS SIGNIFICATIVA.
1	0	0	0	1	CANAL 8 PARTE MAS SIGNIFICATIVA.
1	0	0	1	0	CANAL 9 PARTE MENOS SIGNIFICATIVA.
1	0	0	1	1	CANAL 9 PARTE MAS SIGNIFICATIVA.
1	0	1	0	0	CANAL 10 PARTE MENOS SIGNIFICATIVA.
1	0	1	0	1	CANAL 10 PARTE MAS SIGNIFICATIVA.
1	0	1	1	0	CANAL 11 PARTE MENOS SIGNIFICATIVA.
1	0	1	1	1	CANAL 11 PARTE MAS SIGNIFICATIVA.
1	1	0	0	0	CANAL 12 PARTE MENOS SIGNIFICATIVA.
1	1	0	0	1	CANAL 12 PARTE MAS SIGNIFICATIVA.
1	1	0	1	0	CANAL 13 PARTE MENOS SIGNIFICATIVA.
1	1	0	1	1	CANAL 13 PARTE MAS SIGNIFICATIVA.
1	1	1	0	0	CANAL 14 PARTE MENOS SIGNIFICATIVA.
1	1	1	0	1	CANAL 14 PARTE MAS SIGNIFICATIVA.
1	1	1	1	0	CANAL 15 PARTE MENOS SIGNIFICATIVA.
1	1	1	1	1	CANAL 15 PARTE MAS SIGNIFICATIVA.

TABLA 6.1 Mapeo de los canales analógicos.

RD	WR	A1	A0	ACCION
1	0	0	0	CARGAR EL CONTADOR 0
1	0	0	1	CARGAR EL CONTADOR 1
1	0	1	0	CARGAR EL CONTADOR 2
1	0	1	1	ESCRIBIR EL MODO DE
OPERACION AL REGISTRO DE CONTROL				
0	1	0	0	LEER EL CONTADOR 0
0	1	0	1	LEER EL CONTADOR 1
0	1	1	0	LEER EL CONTADOR 2
0	1	1	1	NO OPERACION

TABLA 6.2 FUNCIONES DEL CONTADOR.

La programación del contador de eventos externos se debe efectuar con los siguientes pasos:

1.- Se indica el modo de acceso al contador, como es lectura aleatoria o lectura de un solo octeto o dos octetos.

2.- Se inicia el valor del contador con cero.

El contador se incrementará cuando se presente un flanco de subida.

6.3-1 Formato de la palabra de control.

La palabra de control enviada al contador de eventos externos consiste en un octeto de datos constituido de la siguiente forma:

D7	D6	D5	D4	D3	D2	D1	D0
SC1	CO	RL1	RLO	M2	M1	M0	BCD

A continuación se describen las funciones de cada bit de la palabra de control.

Los bites D7 y D6 permiten la selección del contador (SC1-SC0) como se muestra.

SC1	SC0	FUNCION
0	0	SELECCION CONTADOR 0.
0	1	SELECCION CONTADOR 1.
1	0	SELECCION CONTADOR 2.
1	1	ILEGAL.

Los bites D5 y D4 son utilizados para la operación de escritura - lectura (RL1-RL0) del registro del contador; abajo se detallan las funciones.

RC1	RL0	FUNCION
0	0	LECTURA AL VUELO.
0	1	ESCRIBIR/LEER EL OCTETO MAS SIGNIFICATIVO.
1	0	ESCRIBIR/ LEER EL OCTETO MENOS SIGNIFICATIVO.
1	1	ESCRIBIR/LEER PRIMERO EL OCTETO MENOS SIGNIFICATIVO Y DESPUES EL MAS SIGNIFICATIVO.

El contador de eventos externos tiene 5 modos de operación, que se seleccionan a través de los bits (M0-M2) de acuerdo a:

M2	M1	M0	MODO	DESCRIPCION
0	0	0	0	conteo por interrupcion
0	0	1	1	disparo programable
0	1	0	2	conteo por valor promedio
0	1	1	3	generador de onda cuadrada
1	0	0	4	habilitación de conteo Software
1	0	1	5	habilitación de conteo Hardware

Por último, el bit D0 especifica si el número del contador está en código binario (BCD=0) o código BCD (BCD=1).

Una vez que se ha programado el contador de eventos externos se puede leer el estado de cada uno de los contadores en cualquier momento.

A continuación se muestra un ejemplo de la programación de uno de los contadores; se inicia con la lectura del puerto, posteriormente se realiza un corrimiento de ocho bits y después como el dato viene en complemento se niega y se obtiene el evento que se está realizando.

6.4 Programación del puerto paralelo.

El puerto utilizado en la tarjeta está constituido por tres puertos independientes y un registro de control.

Este puerto se utiliza para la escritura de 16 canales digitales de salida y la lectura de 8 canales digitales de entrada, los cuales pueden ser programados o leídos al momento que el usuario lo desee.

Con el puerto se realizan dos actividades básicas.

- 1.- Iniciar la palabra de control.
- 2.- Leer el estado o escribir un dato en cada uno de los puertos.

Las funciones anteriores se especifican utilizando las líneas A0-A1 del bus de direcciones, junto con las líneas \overline{RD} y \overline{WR} de acuerdo a la tabla 6.3.

RD	WR	A1	A0	ACCION
1	0	0	0	CARGAR EL PUERTO A
1	0	0	1	CARGAR EL PUERTO B
1	0	1	0	CARGAR EL PUERTO C
1	0	1	1	ESCRIBIR EL MODO DE
OPERACION AL REGISTRO DE CONTROL				
0	1	0	0	LEER EL PUERTO A
0	1	0	1	LEER EL PUERTO B
0	1	1	0	LEER EL PUERTO C
0	1	1	1	NO OPERACION

TABLA 6.3 FUNCIONES DEL CONTADOR.

6.4.1 Formato de la palabra de control del puerto.

La palabra de control enviada al puerto consiste en un octeto de datos constituido de la siguiente forma:

D7	D6	D5	D4	D3	D2	D1	D0
F	M0	M1	PA	PCU	MS	PB	PCL

A continuación se describen las funciones de cada bit de la palabra de control.

El bit 7 indica la activación de la bandera uno lógico activa.

Los bits 6 y 5 el modo de selección.

M0	M1	MODO
0	0	MODO 0
0	1	MODO 1
1	X	MODO 2

Modo 0.- Entrada/salida básica.

Modo 1.- Entrada/salida habilitada.

Modo 2.- Entrada/salida bidireccional.

El bit 4 indica la configuración del puerto A, configurado como entrada (PA=1) y como salida (PA=0).

El bit 3 indica la configuración del puerto C parte alta, configurado como entrada (PCU=1) y como salida (PCU=0).

El bit 2 indica el modo de operación para el grupo B, configurado como entrada (MS=1) y como salida (MS=0).

El bit 1 indica la configuración del puerto B, configurado como entrada (PB=1) y como salida (PB=0).

El bit 0 indica la configuración del puerto C parte baja, configurado como entrada (PCL=1) y como salida (PCL=0).

Para la programación de los puertos se programaron de la siguiente manera; el puerto A y B como salidas y el puerto C como entradas. A continuación se muestra un ejemplo de la programación de uno de los puertos; únicamente el programa pide la información para 8 canales analógicos, posteriormente se aplica la siguiente

formula:

$$\begin{aligned} \text{Puerto} = & (\text{puerto } 0 * 1) + (\text{puerto } 1 * 2) + (\text{puerto } 2 * 4) + \\ & (\text{puerto } 3 * 8) + (\text{puerto } 4 * 16) + (\text{puerto } 5 * 32) + \\ & (\text{puerto } 6 * 64) + (\text{puerto } 7 * 128) \end{aligned}$$

6.5 Descripción del Programa de presentación.

El programa permite realizar la lectura o escritura para cada uno de los canales analógicos de salida, para los canales digitales Entrada/Salida y obtener el estado de cada uno de los contadores en la tarjeta.

El programa está comprendido en tres partes. La primera consta del manejo de los registros de los canales analógicos.

En esta parte se permite la escritura o lectura en cada uno de los canales analógicos, esto se indica en el diagrama de flujo I.

A continuación se presenta el menú para los canales analógicos

```
TECLEE LA OPCION DESEADA .....
VER EL ESTADO DEL CONTADOR
                                     A ..... A
                                     E ..... B
                                     C ..... C
LEER O ESCRIBIR EN UN PUERTO
LEER PUERTO C ..... D
ESCRIBIR AL PUERTO A ..... E
ESCRIBIR AL PUERTO B ..... F
MENU PRINCIPAL.....G
```

La segunda parte esta formada por la lectura de los canales digitales (8) o la escritura de 16 canales digitales y la lectura del estado de cada uno de los contadores, esto se indica en el diagrama de flujo II.

A continuación se presenta el menu para los canales digitales.

TECLEE LA OPCION DESEADA.....
PROGRAMAR CANALES
PROGRAMAR TODOS LOS CANALES.....A
PROGRAMAR UN CANAL DETERMINADO.....B
LEER TODOS LOS CANALES.....C
LEER UN CANAL DETERMINADO.....D

MENU PRINCIPAL.....E

La tercera parte esta indicada como la terminación de uso de la tarjeta y se indica en cada uno de los menus de operación de la tarjeta.

Este programa solo permite asignar voltajes analógicos para cada uno de los canales analógicos de salida y la asignación de estados lógicos para los canales digitales de salida, así como la lectura de los estados lógicos a la entrada de los canales digitales.

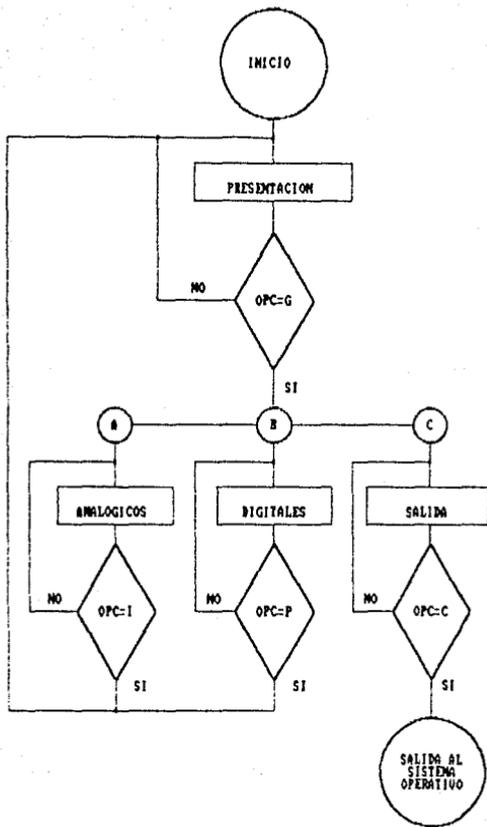


FIGURA 1 70

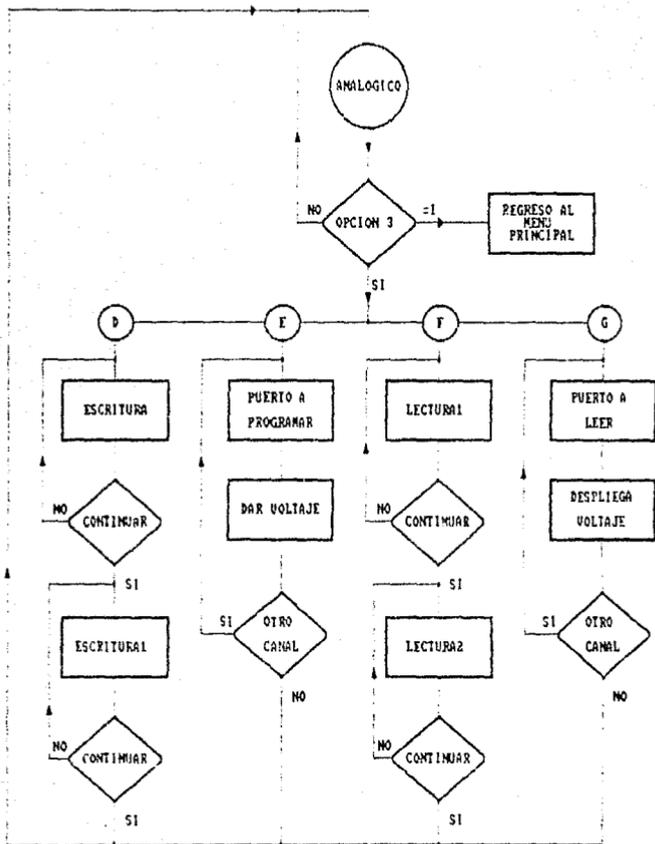


FIGURA 11

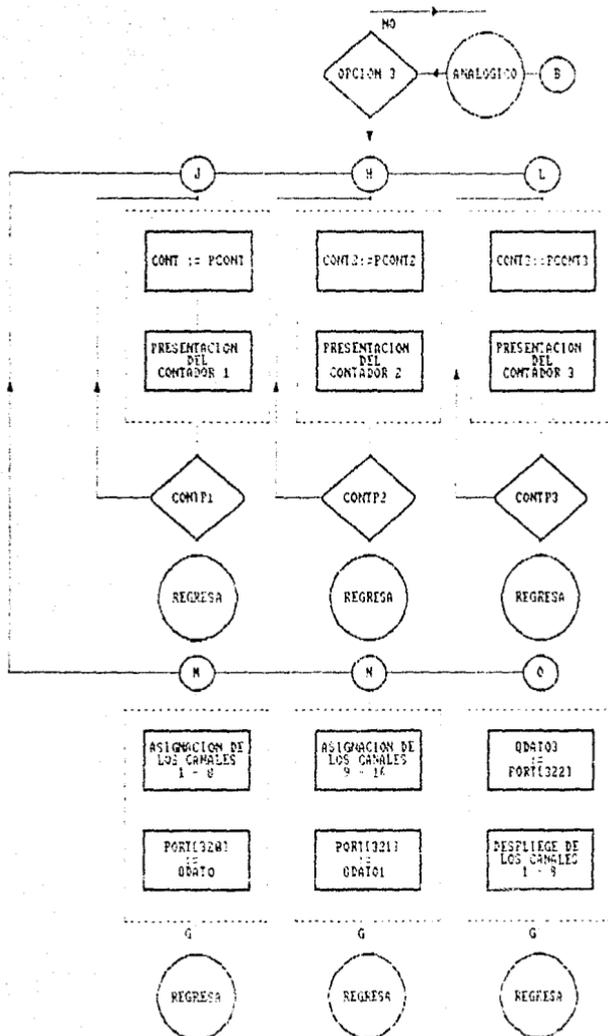


FIGURA 111

7 RESULTADOS

En la interfaz de conversión Digital-Analógica desarrollada en el presente trabajo se permite el manejo de 16 canales analógicos de salida, 16 canales digitales de salida y 8 canales digitales de entrada y tres contadores independientes para el conteo de eventos externos, todas estas actividades son llevadas a cabo por un programa desarrollado específicamente para esta tarjeta. Este programa se escribió en lenguaje de alto nivel (Pascal). La interfaz puede ser conectada con las diferentes computadoras compatibles IBM-PC-XT, por las características similares de hardware que presentan.

Cada canal analógico puede ser programado dentro de una variación de voltaje que va de -5.000 Volts y 4.997 Volts. Se tiene una resolución de 12 bites, dado el convertidor que se utilizó, y la variación mínima de voltaje que permite esta circuitería es de 2.5 milivolts.

En el diseño de la interfaz se minimizaron los componentes, es decir, se utilizó un solo convertidor Digital-Analógico para la generación de 16 señales analógicas de salida, este convertidor se encuentra integrado a las etapas de multicanalización y retención. Esto permitió la disminución del costo de la tarjeta interfaz y la facilidad de poder ser construida con una mayoría de componentes que existen en el mercado nacional (salvo el convertidor).

Se logró un consumo de corriente bajo, ya que los componentes ocupados en la interfaz son de bajo consumo LS (low power schottly) y CMOS.

La interfaz permite escoger diferentes segmentos de memorias de trabajo, por medio de un micro-swicht, estos segmentos deben estar disponibles para el usuario.

El error en el voltaje analógico de salida es de ± 0.7 milivolts/bit. Se tiene un periodo de refrescamiento de 6.5 microsegundos para cada canal analógico de salida.

El módulo de conexiones permite el acoplamiento de la interfaz y algun proceso exterior. Este módulo acopla las señales de la interfaz con la señales eléctricas externas.

Para cada señal analógica se tiene un conector BNC con su respectiva tierra; para las señales digitales se tiene un conector DB37.

Las aplicaciones posibles son acoplar los diferentes procesos en los cuales la interfaz puede controlar dichos dispositivos, estos se enfocan a la industria en general (manejo de actuadores , relevadores etc.) y la investigación como el control de procesos.

Esto es muy versatil, ya que la interfaz se programa desde una computadora personal para iniciar las condiciones idóneas para dicho proceso.

Con el desarrollo de esta interfaz Digital/Analógica se logra aportar a la industria e investigación un dispositivo que sea adquirido directamente en el mercado nacional, permitiendo tener una independencia tecnológica.

8. BIBLIOGRAFIA

- 1.- ANALOG DEVICES ' PC BUS COMPATIBLE PRODUCTS, Catalog and Technical Notes 1987.
- 2.- CMOS DATABOOK, National Semiconductor Corp. Santa Clara, 1984.
- 3.- Data Conversion/Adquisition Databook, National Semiconductor Corp., Santa Clara, 1984.
- 4.- Eggebrecht, L. ' Interfacing to the IBM Personal Computer', Howard W. Sams and Co., Indianapolis, 1983.
- 5.- Hall, D ' Microprocessors and Interfacing, Programming and Hardware', Mc Graw-Hill International Editions, 1986.
- 6.-Hayes ' Digital System Design and Microprocessors' McGraw - Hill in computer organization and architecture, 1985.
- 7.- Hill, R Peterson ' Digital System Hardware Organization and Design', Jhon Wiley and Sons, Inc., 1978.
- 8.- IBM, ' Personal Computer PC jr. Hardware Reference Library. Technical Reference', International Bussiness Machines Corporation, 1983.
- 9.- Intel Component Data Catalog. Intel Corporation, 1987.
- 10- Rollins, D. ' IBM-PC 8088 MACRO Assembler Programming', McMillan Publishing Company, 1985.

11.- Streitmatter ' Microprocessors Theory and Aplications' Reston
1979.

12.- Turbo Pascal Owners Handbook, Borland International, 1987.

13.-Wiatrowski,H. ' Logic Circuit and Microcomputer Systems',
International Student Edition, 1984.

14.-Wobschall ' Circuit Design for Electronic
Instrumentacion', McGraw - Hill , 1979.

ANEXO A

Listado del programa y presentación para el manejo de la interfaz Digital/analógica.

Este programa permite operar la interfaz de manera sencilla; este programa consta dos partes, la primera parte corresponde a los canales analógicos de salida y la segunda parte a los canales digitales entrada/salida y contadores. Cada parte contiene diferentes opciones que permiten el manejo de la interfaz

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

```

Program convertidor
(input, output);
Uses Dos, Crt, Printer, Turbo3;
{N+}
{F+}
{R+}
    ( Etiqueta para salir al sistema operativo )

Label exit;
    ( Etiqueta de regreso al menu )

Label regresar;
Label uno;
Label doss;
Label tres;

VAR
    ( variables para iniciar graficas )

Opcion          : String[10];
Opcion2         : String[10];
Digitale        : Array [1..8] of real;
Digital         : Array [1..16] of String[10];
Gital           : Array [1..16] of Integer;
Y              : Array [1..16] of Integer;
X              : Array [1..10] of Integer;
Gd, gm, a, vari : Integer;
Puerton, vari2, vari3 : Integer;
N, m           : Integer;
Opciona        : Char;
Opcion2a       : Char;

    ( variables para la lectura de cada canal )

Dato0, dato1, dato2, dato3, dato4, dato5      : Integer;
Dato6, dato7, dato8, dato9, dato10, dato11, dato12 : Integer;
Dato13, dato14, dato15, puertore             : Integer;

    ( variales que niegan la lectura de un )
    ( canal analogico )

Ddato0, ddato1, ddato2, ddato3, ddato4, ddato5 : Integer;
Ddato6, ddato7, ddato8, ddato9, ddato10, ddato11 : Integer;
Ddato12, ddato13, ddato14, ddato15             : Integer;
Ddatorr                                       : Word;

    ( variables que se asignan a cada )

```

```

( uno de los puertos )

Puerto0,puerto1,puerto2,puerto3,puerto4 : Integer;
Puerto5,puerto6,puerto7,puerto8,puerto9 : Integer;
Puerto10,puerto11,puerto12,puerto13,puerto14 : Integer;
Puerto15,puertori,ddatorr : Integer;

( variables de ayuda en el programa )
( principal )

Puerto,dato,puertor,vref : Integer;

( variables para la escritura de cada )
( uno de los canales en forma analogica )

Vol0,vol1,vol2,vol3,vol4,vol5,vol6,vol7,vol8 : Integer;
Vol9,vol10,vol11,vol12,vol13,vol14,vol15 : Integer;
Voltajei : Integer;

Pcont1,pcont2,pcont3 : Integer;
Gital1,gital2,gital3,gital4,gital5,gital6 : Integer;
Gital7,gital8,gital9,gital10,gital11 : Integer;
Gital12,gital13,gital14,gital15,gital16 : Integer;
Gitale1,gitale2,gitale3,gitale4,gitale5 : Integer;
Gitale6,gitale7,gitale8 : Integer;

Qdato1,Qdato2,Qpuera,Qpuerb,Qpuerc,QdatoP : Integer;
Qpuertop,Ipuerto : Integer;

( variables para los diferentes )
( contadores )

Cont1,cont2,cont3,Qdato : Word;

Qdatoe1,qdatoe11,qdato11,qdatob1,cont22 : Real;
Dator,dato00,dato01,dato02,dato03,dato04 : Real;
Dato05,dato06,dato07,dato08,dato09,dato010 : Real;
Dato011,dato012,dato013,dato014,dato015 : Real;

Voltaje0,voltaje1,voltaje2,voltaje3,dato0 : Real;
Voltaje4,voltaje5,voltaje6,voltaje7,voltaje8 : Real;
Voltaje9,voltaje10,voltaje11,voltaje12 : Real;
Voltaje13,voltaje14,voltaje15,voltaje : Real;

Vvoltaje0,vvoltaje1,vvoltaje2,vvoltaje3 : Real;
Vvoltaje4,vvoltaje5,vvoltaje6,vvoltaje7 : Real;
Vvoltaje9,vvoltaje10,vvoltaje11,vvoltaje12 : Real;
Vvoltaje13,vvoltaje14,vvoltaje15,vvoltaje : Real;
Vvoltaje8 : Real;

```

```

        { variables para cada uno de los }
        { 16 canales digitales de salida }
Oprime,opf                                     : Char;
Digital1,digital2,digital3,digital4           : Char;
Digital5,digital6,digital7,digital8           : Char;
Digital10,digital11,digital12,digital13       : Char;
Digital14,digital15,digital16,digital9       : Char;

        { variables para cada uno de los }
        { 8 canales de entrada }

Digitale1,digitale2,digitale3,digitale4       : Char;
Digitale5,digitale6,digitale7,digitale8       : Char;

        { variable que nos permite continuar }
        { despues de observar la presentacion }

D,contp1,contp2,contp3,opcion1               : Char;
opcion1                                       : String [10];

Canal,qcont,Lpuertoc,Lpuertob                 : Char;
Continuar,qpot,qpuerto,Lpuertoa             : Char;
Opc                                           : String[10];

        { Esta subrutina nos presenta la portada de inicio }

```

```

Procedure presentacion;

```

```

Begin
  Clrscr;
  Gotoxy(16,9);
  Writeln('*****');
  ');
  Gotoxy(16,10);
  Writeln('*',' ':49,'*');
  Gotoxy(16,11);
  Writeln('*',' ':13,'INSTITUTO DE INGENIERIA',' ':13,'*');
  Gotoxy(16,12);
  Writeln('*','COORDINACION DE AUTOMATIZACION',' ':19,'*');
  Gotoxy(16,13);
  Writeln('*',' ':49,'*');
  Gotoxy(16,14);
  Writeln('*',' ':14,'PAQUETE DE CONVERSION',' ':14,'*');

```

```

Gotoxy(16,15);
Writeln('*',' ':17,'PARA SENALES',' ':20,'*');
Gotoxy(16,16);
Writeln('*',' ':14,'DIGITALES - ANALOGICAS',' ':13,'*');
Gotoxy(16,17);
Writeln('*','ELABORADO POR: JOSE GABRIEL GARCIA CRUZ':10,'*');
Gotoxy(16,18);
Writeln('*****
);

```

```

End;
{ Esta subrutina nos presenta el menu de opciones para }
{ los canales analogicos y digitales y la respuesta se }
{ almacena en la variable que se asigna posteriormente }

```

Procedure mascara;

Begin

```

Clrscr;
Gotoxy(15,7);Writeln('          MENU DE OPCIONES          ');
Gotoxy(15,9);Writeln('TECLEE LA OPCION          DESEADA.....');
Gotoxy(15,11);Writeln('OPCION PARA CANALES ANALOGICOS
          .....A');
Gotoxy(15,13);Writeln('OPCION PARA CANALES DIGITALES
          .....B');
Gotoxy(15,15);Writeln('TERMINAR LA SECCION.....C');
Write(Chr(7));

```

End;

```

{Procedimiento que nos permite escoger una de las }
{funciones de los canales analogicos escritura o }
{lectura }

```

Procedure analogicos;

Begin

```

Clrscr;
Gotoxy(15,5);Writeln('TECLEE LA OPCION DESEADA.....');
Gotoxy(15,7);Writeln('PROGRAMAR CANALES');
Gotoxy(15,9);Writeln('PROGRAMAR TODOS LOS CANALES.....A');
Gotoxy(15,11);Writeln('PROGRAMAR UN CANAL DETERMINADO.....B');
Gotoxy(15,13);Writeln('LEER TODOS LOS CANALES.....C');
Gotoxy(15,15);Writeln('LEER UN CANAL DETERMINADO.....D');
Gotoxy(15,21);Writeln('MENU PRINCIPAL.....E');

```

```

Write(Chr(7));

End;

      {Procedimiento que nos permite escoger una de las }
      {funciones de los canales digitales                }

Procedure Digitales;

Begin

  Clrscr;
  Gotoxy(15,5);Writeln('TECLEE LA OPCION DESEADA .....');
  Gotoxy(15,7);Writeln('VER EL ESTADO DEL CONTADOR ');
  Gotoxy(15,9);Writeln(' ':25,'A ..... A');
  Gotoxy(15,11);Writeln(' ':25,'B ..... B');
  Gotoxy(15,13);Writeln(' ':25,'C ..... C');
  Gotoxy(15,15);Writeln('LEER O ESCRIBIR EN UN PUERTO');
  Gotoxy(15,17);Writeln('LEER PUERTO C ..... D');
  Gotoxy(15,19);Writeln('ESCRIBIR AL PUERTO A ..... E');
  Gotoxy(15,21);Writeln('ESCRIBIR AL PUERTO B ..... F');
  Gotoxy(15,23);Writeln('MENU PRINCIPAL.....G');
  Write(Chr(7));

End;

      { Procedimiento que nos permite obtener el valor }
      { de voltaje analogico para cada canal. Esto se }
      { realiza por medio de lectura de un puerto con }
      { una instruccion en pascal. Se realiza una con- }
      { version del dato obtenido ( entero ) a un dato }
      { real. Esta contiene los primeros 8 canales }

Procedure lectural;

Begin

  Clrscr;
  Begin
    dato0 := portw[$300];
    ddato0 := not(dato0);
    dato0 := 4095 - ddato0 ;
    dato0 := dato0 and ($0FFF);
    voltaje0 := ((dato0 - 2048)/2048)*5;

  End;

End;

```

```

Begin
  dato1      := portw[$302];
  ddato1     := not(dato1);
  dato1      := 4095 - ddato1 ;
  dato1      := dato1 and ($0FFF);
  vvoltaje1  := ((dato1 - 2048)/2048)*5;
End;

```

```

Begin
  dato2      := portw[$304];
  ddato2     := not(dato2);
  dato2      := 4095 - ddato2 ;
  dato2      := dato2 and ($0FFF);
  vvoltaje2  := ((dato2 - 2048)/2048)*5;

```

End;

```

Begin
  dato3      := portw[$306];
  ddato3     := not(dato3);
  dato3      := 4095 - ddato3 ;
  dato3      := dato3 and ($0FFF);
  vvoltaje3  := ((dato3 - 2048)/2048)*5;

```

End;

```

Begin
  dato4      := portw[$308];
  ddato4     := not(dato4);
  dato4      := 4095 - ddato4 ;
  dato4      := dato4 and ($0FFF);
  vvoltaje4  := ((dato4 - 2048)/2048)*5;

```

End;

```

Begin
  dato5      := portw[$30A];
  ddato5     := not(dato5);
  dato5      := 4095 - ddato5 ;
  dato5      := dato5 and ($0FFF);
  vvoltaje5  := ((dato5 - 2048)/2048)*5;

```

End;

```

Begin
  dato6      := portw[$30C];
  ddato6     := not(dato6);
  dato6      := 4095 - ddato6 ;
  dato6      := dato6 and ($0FFF);

```

```

vvoltaje6 := ((dato6 - 2048)/2048)*5;

End;

Begin
dato7      := portw[$30E];
ddato7     := not(dato7);
dato7      := 4095 - ddato7 ;
dato7      := dato7 and ($0FFF);
vvoltaje7 := ((dato7 - 2048)/2048)*5;

End;

End;
Procedure lecl;
Begin
Gotoxy(5,5);Writeln('CANAL 0      VOLTAJE = ',VVOLTAJE0:6:4 );
Gotoxy(5,7);Writeln('CANAL 1      VOLTAJE = ',VVOLTAJE1:6:4 );
Gotoxy(5,9);Writeln('CANAL 2      VOLTAJE = ',VVOLTAJE2:6:4 );
Gotoxy(5,11);Writeln('CANAL 3     VOLTAJE = ',VVOLTAJE3:6:4 );
Gotoxy(5,13);Writeln('CANAL 4     VOLTAJE = ',VVOLTAJE4:6:4 );
Gotoxy(5,15);Writeln('CANAL 5     VOLTAJE = ',VVOLTAJE5:6:4 );
Gotoxy(5,17);Writeln('CANAL 6     VOLTAJE = ',VVOLTAJE6:6:4 );
Gotoxy(5,19);Writeln('CANAL 7     VOLTAJE = ',VVOLTAJE7:6:4 );

End;

      { Procedimiento que nos permite obtener el valor }
      { de voltaje analogico para cada canal. Esto se }
      { realiza por medio de lectura de un puerto con }
      { una instruccion en pascal. Se realiza una con- }
      { version del dato obtenido ( entero ) a un dato }
      { real. Esta contiene los ultimos 8 canales }

Procedure lectura2;

Begin
Clrscr;

Begin
dato8      := portw[$310];
ddato8     := not(dato8);
dato8      := 4095 - ddato8 ;
dato8      := dato8 and ($0FFF);
vvoltaje8 := ((dato8 - 2048)/2048)*5;

End;

End;

```

```

Begin
  dato9      := portw[$312];
  ddato9     := not(dato9);
  dato9      := 4095 - ddato9 ;
  dato9      := dato9 and ($0FFF);
  vvoltaje9  := ((dato9 - 2048)/2048)*5;

End;

Begin
  dato10     := portw[$314];
  ddato10    := not(dato10);
  dato10     := 4095 - ddato10 ;
  dato10     := dato10 and ($0FFF);
  vvoltaje10 := ((dato10 - 2048)/2048)*5;

End;

Begin
  dato11     := portw[$316];
  ddato11    := not(dato11);
  dato11     := 4095 - ddato11 ;
  dato11     := dato11 and ($0FFF);
  vvoltaje11 := ((dato11 - 2048)/2048)*5;

End;

Begin
  dato12     := portw[$318];
  ddato12    := not(dato12);
  dato12     := 4095 - ddato12 ;
  dato12     := dato12 and ($0FFF);
  vvoltaje12 := ((dato12 - 2048)/2048)*5;

End;

Begin
  dato13     := portw[$31A];
  ddato13    := not(dato13);
  dato13     := 4095 - ddato13 ;
  dato13     := dato13 and ($0FFF);
  vvoltaje13 := ((dato13 - 2048)/2048)*5;

End;

```

```

Begin
  dato14      := portw[$31C];
  ddato14    := not(dato14);
  dato14     := 4095 - ddato14 ;
  dato14     := dato14 and ($0FFF);
  vvoltaje14 := ((dato14 - 2048)/2048)*5;

```

End;

```

Begin
  dato15      := portw[$31E];
  ddato15    := not(dato15);
  dato15     := 4095 - ddato15 ;
  dato15     := dato15 and ($0FFF);
  vvoltaje15 := ((dato15 - 2048)/2048)*5;

```

End;

End;

Procedure llec1:

```

Begin
  Gotoxy(3,5);
  Writeln('CANAL 0      VOLTAJE = ',VVOLTAJE0:6:4 );
  Gotoxy(3,7);
  Writeln('CANAL 1      VOLTAJE = ',VVOLTAJE1:6:4 );
  Gotoxy(3,9);
  Writeln('CANAL 2      VOLTAJE = ',VVOLTAJE2:6:4 );
  Gotoxy(3,11);
  Writeln('CANAL 3      VOLTAJE = ',VVOLTAJE3:6:4 );
  Gotoxy(3,13);
  Writeln('CANAL 4      VOLTAJE = ',VVOLTAJE4:6:4 );
  Gotoxy(3,15);
  Writeln('CANAL 5      VOLTAJE = ',VVOLTAJE5:6:4 );
  Gotoxy(3,17);
  Writeln('CANAL 6      VOLTAJE = ',VVOLTAJE6:6:4 );
  Gotoxy(3,19);
  Writeln('CANAL 7      VOLTAJE = ',VVOLTAJE7:6:4 );
  Gotoxy(40,7);
  Writeln('CANAL 8      VOLTAJE = ',VVOLTAJE8:6:4 );
  Gotoxy(40,5);
  Writeln('CANAL 9      VOLTAJE = ',VVOLTAJE9:6:4 );
  Gotoxy(40,9);
  Writeln('CANAL 10     VOLTAJE = ',VVOLTAJE10:6:4 );
  Gotoxy(40,11);
  Writeln('CANAL 11     VOLTAJE = ',VVOLTAJE11:6:4 );
  Gotoxy(40,13);
  Writeln('CANAL 12     VOLTAJE = ',VVOLTAJE12:6:4 );

```

```

Gotoxy(40,15);
Writeln('CANAL 13      VOLTAJE = ',VVOLTAJE13:6:4 );
Gotoxy(40,17);
Writeln('CANAL 14      VOLTAJE = ',VVOLTAJE14:6:4 );
Gotoxy(40,19);
Writeln('CANAL 15      VOLTAJE = ',VVOLTAJE15:6:4 );
Gotoxy(16,22);
Writeln('OPRIMA LA TECLA G Y <RETURN>  PARA CONTINUAR');

Repeat
  Gotoxy(64,22);Writeln(' ':10);
  Gotoxy(64,22); Readln(continuar);
Until ((continuar = 'G') or (continuar = 'g'));

End;
      { Procedimiento que nos indica que el }
      { dato es incorrecto }

```

Procedure ayuda;

```

begin
  Lectural;
  Lectura2;
  Gotoxy(2,2);
  Writeln('CANAL',':5,'VOLTAJE',':5,'VOLTAJE');
  Gotoxy(43,2);
  Writeln('CANAL',':5,'VOLTAJE',':7,'VOLTAJE');
  Gotoxy(12,3);
  Writeln('ANTERIOR',':4,'ACTUAL');
  Gotoxy(53,3);
  Writeln('ANTERIOR',':6,'ACTUAL');
  Gotoxy(3,5);
  Writeln('0',':7,',VVOLTAJE0:6:4,' V');
  Gotoxy(3,7);
  Writeln('1',':7,',VVOLTAJE1:6:4,' V');
  Gotoxy(3,9);
  Writeln('2',':7,',VVOLTAJE2:6:4,' V');
  Gotoxy(3,11);
  Writeln('3',':7,',VVOLTAJE3:6:4,' V');
  Gotoxy(3,13);
  Writeln('4',':7,',VVOLTAJE4:6:4,' V');
  Gotoxy(3,15);
  Writeln('5',':7,',VVOLTAJE5:6:4,' V');
  Gotoxy(3,17);
  Writeln('6',':7,',VVOLTAJE6:6:4,' V');
  Gotoxy(3,19);
  Writeln('7',':7,',VVOLTAJE7:6:4,' V');
  Gotoxy(45,5);
  Writeln('8',':7,' ',VVOLTAJE8:6:4,' V');

```

```

Gotoxy(45,7);
Writeln('9','':7,' ',VVOLTAJE9:6:4,' V');
Gotoxy(45,9);
Writeln('10','':7,' ',VVOLTAJE10:6:4,' V');
Gotoxy(45,11);
Writeln('11','':7,' ',VVOLTAJE11:6:4,' V');
Gotoxy(45,13);
Writeln('12','':7,' ',VVOLTAJE12:6:4,' V');
Gotoxy(45,15);
Writeln('13','':7,' ',VVOLTAJE13:6:4,' V');
Gotoxy(45,17);
Writeln('14','':7,' ',VVOLTAJE14:6:4,' V');
Gotoxy(45,19);
Writeln('15','':7,' ',VVOLTAJE15:6:4,' V');
End;

```

Procedure mensaje;

```

Begin
  Clrscr;
  ayuda;
  Gotoxy(20,21);Writeln('LOS DATOS DEBEN ESTAR COMPRENDIDOS ');
  Gotoxy(23,22);Writeln('ENTRE -5.0000 V Y 4.9975 V. ');
  Writeln(Chr(7));
  Delay(2000);
  Gotoxy(20,21);Writeln(' ':40);
  Gotoxy(20,21);Writeln(' ':40);
  Gotoxy(22,22);Writeln(' ':30);
  Gotoxy(20,23);Writeln(' ':30);
End;

```

Procedure escritura;

```

      { Este procedimiento nos permite escribir un }
      { voltaje en uno de los 16 canales analogicos. }
      { Este voltaje tiene un valor maximo de 4.9974 V. }
      { y un minimo de -5.000 V. ; si el voltaje no se }
      { encuentra en este rango, se indicara un mensaje }
      { de error y el programa se quedara esperando un }
      { voltaje dentro del rango; Esto se realiza para }
      { los primeros canales ( 0 - 7 ). }

Begin
  Clrscr;
  ayuda;
  begin
    {$I-}
    Repeat
      Gotoxy(24,5);Writeln('':8);
      Gotoxy(24,5);Readln(voltaje0);
    
```

```

vol0 := ioreult ;
If (vol0 <> 0) then mensaje;
If ((voltaje0 > -5.0) and (voltaje0 > 4.9975 )) then mensaje;
Until ((vol0 = 0 ) and (voltaje0 >= -5.000)
and (voltaje0 <= 4.9975));
{$I+}
dato00      := (2048*voltaje0/vref);
dato0       := 2048 + trunc(dato00) ;
Portw[$300] := dato0;
end;

{$I-}
Repeat
Gotoxy(24,7);Writeln('':8);
Gotoxy(24,7);Readln(voltaje1);
vol1 := ioreult ;
If (vol1 <> 0) then mensaje;
If ((voltaje1 > -5.0) and (voltaje1 > 4.9975 )) then mensaje;
Until ((VOL1 = 0 ) and (voltaje1 >= -5.000)
and (voltaje1 <= 4.9975));
{$I+}
dato01      :=(2048*(voltaje1/vref));
dato1       := (2048 + trunc (dato01));
Portw[$302] := dato1;

{$I-}
Repeat
Gotoxy(24,9);Writeln('':8);
Gotoxy(24,9);Readln(voltaje2);
vol2 := ioreult ;
If (vol2 <> 0) then mensaje;
If ((voltaje2 > -5.0) and (voltaje2 > 4.9975 )) then mensaje;
Until ((vol2 = 0 ) and (voltaje2 >= -5.000)
and (voltaje2 <= 4.9975));
{$I+}

dato02      := (2048*(voltaje2/vref));
dato2       := (2048 + trunc (dato02));
Portw[$304] := dato2;

{$I-}
Repeat
Gotoxy(24,11);Writeln('':8);
Gotoxy(24,11);Readln(voltaje3);
vol3 := ioreult ;
If (vol3 <> 0) then mensaje;
If ((voltaje3 > -5.0) and (voltaje3 > 4.9975 )) then mensaje;
Until ((vol3 = 0 ) and (voltaje3 >= -5.000)
and (voltaje3 <= 4.9975));

```

```

{$I+}
dato03      := (2048*(voltaje3/vref));
dato3       := (2048 + trunc(dato03));
Portw[$306] := dato3;

{$I-}
Repeat
Gotoxy(24,13);Writeln('':8);
Gotoxy(24,13);Readln(voltaje4);
vol4 := ioreult ;
If (vol4 <> 0) then mensaje;
If ((voltaje4 > -5.0) and (voltaje4 > 4.9975 )) then mensaje;
Until ((vol4 = 0) and (voltaje4 >= -5.000)
and (voltaje4 <= 4.9975));
{$I+}
dato04      := (2048*(voltaje4/vref));
dato4       := (2048 + trunc(dato04));
Portw[$308] := dato4;

{$I-}
Repeat
Gotoxy(24,15);Writeln('':8);
Gotoxy(24,15);Readln(voltaje5);
vol5 := ioreult ;
If (vol5 <> 0) then mensaje;
If ((voltaje5 > -5.0) and (voltaje5 > 4.9975 )) then mensaje;
Until ((vol5 = 0) and (voltaje5 >= -5.000)
and (voltaje5 <= 4.9975));
{$I+}
dato05      := (2048*(voltaje5/vref));
dato5       := (2048 + trunc(dato05));
Portw[$30A] := dato5;

{$I-}
Repeat
Gotoxy(24,17);Writeln('':8);
Gotoxy(24,17);Readln(voltaje6);
vol6 := ioreult ;
If (vol6 <> 0) then mensaje;
If ((voltaje6 > -5.0) and (voltaje6 > 4.9975 )) then mensaje;
Until ((vol6 = 0) and (voltaje6 >= -5.000)
and (voltaje6 <= 4.9975));
{$I+}
dato06      := (2048*(voltaje6/vref));
dato6       := (2048 + trunc(dato06));
Portw[$30C] := dato6;

{$I-}
Repeat
Gotoxy(24,19);Writeln('':8);

```

```

Gotoxy(24,19);Readln(voltaje7);
vol7 := ioreult ;
If (vol7 <> 0) then mensaje;
If ((voltaje7 > -5.0) and (voltaje7 > 4.9975 )) then mensaje;
Until ((vol7 = 0 ) and (voltaje7 >= -5.000)
and (voltaje7 <= 4.9975));
{$I+}

dato07      := (2048*(voltaje7/vref));
dato7       := (2048 + trunc(dato07));
Portw[$30E] := dato7;

```

End;

```

{ Este procedimiento nos permite escribir un }
{ voltaje en uno de los 16 canales analogicos. }
{ Este voltaje tiene un valor maximo de 4.9974 V. }
{ y un minimo de -5.000 V. ; si el voltaje no se }
{ encuentra en este rango, se indicara un mensaje }
{ de error y el programa se quedara esperando un }
{ voltaje dentro del rango; Esto se realiza para }
{ los ultimos canales ( 8 - 15 ). }

```

Procedure escritural;

Begin

Begin

(\$I-)

Repeat

Gotoxy(67,5);Writeln('':15);

Gotoxy(67,5);Readln (voltaje8);

vol8 := ioreult ;

If (vol8 <> 0) then mensaje;

If ((voltaje8 > -5.0) and (voltaje8 > 4.9975)) then mensaje;

Until ((vol8 = 0) and (voltaje8 >= -5.000)

and (voltaje8 <= 4.9975));

(\$I+)

dato08 := (2048*(voltaje8/vref));

dato8 := (2048 + trunc(dato08));

Portw[\$310] := dato8;

End;

Begin

(\$I-)

Repeat

Gotoxy(67,7);Writeln('':15);

Gotoxy(67,7);Readln(voltaje9);

```

vol9 := ioreult ;
If (vol9 <> 0) then mensaje;
If ((voltaje9 > -5.0) and (voltaje9 > 4.9975 )) then mensaje;
Until ((vol9 = 0 ) and(voltaje9 >= -5.000)
and (voltaje9 <= 4.9975));
($I+)

dato09      := (2048*(voltaje9/vref));
dato9       := (2048 + trunc(dato09));
Portw[$312] := dato9;
End;

Begin
($I-)

Repeat
Gotoxy(67,9);Writeln('':15);
Gotoxy(67,9);Readln(voltaje10);
VOL10 := ioreult ;
If (vol10 <> 0) then mensaje;
If ((voltaje10 > -5.0) and (voltaje10 > 4.9975 )) then mensaje;
Until ((vol10 = 0) and (voltaje10 >= -5.000)
and (voltaje10 <= 4.9975));
($I+)

dato010     := (2048*(voltaje10/vref));
dato10      := (2048 + trunc(dato010));
Portw[$314] := dato10;
End;

Begin
($I-)

Repeat
Gotoxy(67,11);Writeln('':15);
Gotoxy(67,11);Readln(voltaje11);
vol11 := ioreult ;
If (vol11 <> 0) then mensaje;
If ((voltaje11 > -5.0) and (voltaje11 > 4.9975 )) then mensaje;
Until ((VOL11 = 0 ) and (voltaje11 >= -5.000)
and (voltaje11 <= 4.9975));
($I+)

dato011     := (2048*(voltaje11/vref));
dato11      := (2048 + trunc(dato011));
Portw[$316] := dato11;
End;

Begin
($I-)

```

```

Repeat
Gotoxy(67,13);Writeln('':15);
Gotoxy(67,13);Readln(voltaje12);
voll2 := ioresult ;
If (VOL12 <> 0) then mensaje;
If ((voltaje12 > -5.0) and (voltaje12 > 4.9975 )) then mensaje;
Until ((voll2 = 0 ) and (voltaje12 >= -5.000)
and (voltaje12 <= 4.9975));
{$I+}

dato012 := (2048*(voltaje12/vref));
dato12 := (2048 + trunc(dato012));
Portw[$318] := dato12;
End;

Begin

{$I-}
Repeat
Gotoxy(67,15);Writeln('':15);
Gotoxy(67,15);Readln(voltaje13);
voll3 := ioresult ;
If (voll3 <> 0) then mensaje;
If ((voltaje13 > -5.0) and (voltaje13 > 4.9975 )) then mensaje;
Until ((voll3 = 0 ) and (voltaje13 >= -5.000)
and (voltaje13 <= 4.9975));
{$I+}

dato013 := (2048*(voltaje13/vref));
dato13 := (2048 + trunc(dato013));
Portw[$31A] := dato13;
End;

Begin

{$I-}
Repeat
Gotoxy(67,17);Writeln('':15);
Gotoxy(67,17);Readln(voltaje14);
voll4 := ioresult ;
If (voll4 <> 0) then mensaje;
If ((voltaje14 > -5.0) and (voltaje14 > 4.9975 )) then mensaje;
Until ((voll4 = 0) and (voltaje14 >= -5.000)
and (voltaje14 <= 4.9975));
{$I+}

dato014 := (2048*(voltaje14/vref));
dato14 := (2048 + trunc(dato014));
Portw[$31C] := dato14;

```

```

End;

Begin
  {$I-}
  Repeat
    Gotoxy(67,19);WriteLn('':15);
    Gotoxy(67,19);ReadLn(voltaje15);
    voll5 := ioreult ;
    If (voll5 <> 0) then mensaje;
    If ((voltaje15 > -5.0) and (voltaje15 > 4.9975 )) then mensaje;
    Until ((voll5 = 0) and (voltaje15 >= -5.000)
    and (voltaje15 <= 4.9975));
  {$I+}

  dato015      := (2048*(voltaje15/vref));
  dato15       := (2048 + trunc(dato015));
  Portw[$31E] := dato15;
End;

End;

```

```

      { Procedimiento que nos permite conocer el estado }
      { de contador 1 }

```

```

Procedure contador1;

```

```

Begin
  vari := 100;
  Clrscr;
  repeat
    cont1 := Port[$328];
    cont1 := (Port[$328] SHL 8) or cont1 ;
    cont1 := not(cont1);
    Gotoxy(15,15);
    WriteLn('ES ESTADO DEL CONTADOR 1 ES : ',CONT1 );
    if ( cont1 = vari ) then vari := 100 + vari;
    if (cont1 < 35000) then portw[$300] := $0000;
    if (cont1 > 35000) then portw[$300] := $0fff;
  until keypressed;
End;

```

```

      { Procedimiento que nos permite conocer el estado }
      { de contador 1. }

```

```

Procedure contador2;

```

```

Begin
  Clrscr;
  repeat

```

```

cont2 := Port[$329];
cont2 := (Port[$329] SHL 8) or cont2 ;
cont2 := not(cont2);
Gotoxy(15,15);
cont2 := trunc(cont2/10);
Writeln('ES ESTADO DEL CONTADOR 2 ES : ',CONT2);
if (cont2 < 40960) then portw[$302] := (cont2);
until keypressed;
End;

      { Procedimiento que nos permite conocer el estado }
      { de contador 1. }

Procedure contador3;
Begin
  Clrscr;
  repeat
    cont3 := Port[$32A];
    cont3 := (Port[$32A] SHL 8) or cont3 ;
    cont3 := not(cont3);
    Gotoxy(15,15);
    Writeln('ES ESTADO DEL CONTADOR 3 ES : ',CONT3);
    cont22 := 65535 * sin(cont3);
    writeln(cont22);
    cont3 := trunc(cont22);
    cont3 := 4 * cont3;
    writeln(cont3);
    if (cont3 < 65000) then portw[$304] := (cont3);
    until keypressed;
  End;

Procedure mensaje1;
Begin
  gotoxy(5,25); writeln(' DATO NO VALIDO ');
  Delay(500);
End;

Procedure conteo;
Begin
qdato := ((1 * gital[1]) + (2 * gital[2]) + (4 * gital[3])
+ (8 * gital[4]) + (16 * gital[5]) + (32 * gital[6])
+ (64 * gital[7]) + (128 * gital[8]));
Port[$320] := qdato;
End;

Procedure puertoa;
Begin
  Clrscr;
  Gotoxy(20,2);
  Writeln('ACTIVO = 1','':7,'DESACTIVO = 0');
  Gotoxy(16,21);

```

```
Writeln('SE PROGRAMA ESTOS CANALES CUANDO SE ASIGNAN');
Gotoxy(21,22);
Writeln('LOS PRIMEROS 8 CANALES DIGITALES');
```

```
Begin
```

```
Repeat
```

```
  Gotoxy(54,5);Writeln(' ':10);
  Gotoxy(15,5);
  Writeln('CANAL DIGITAL 1 (1/0).....');
  Gotoxy(54,5);Readln(digital[1]);
  y[1] := length (digital[1]);
  If (digital[1] = '1' ) then gital[1] := 1;
  If (digital[1] = '0' ) then gital[1] := 0;
  Until (( y[1] = 1 ) and (digital[1] = '1')
  or (digital[1] = '0'));
  conteo;
```

```
End;
```

```
Begin
```

```
Repeat
```

```
  Gotoxy(54,7);Writeln(' ':10);
  Gotoxy(15,7);
  Writeln('CANAL DIGITAL 2 (1/0).....');
  Gotoxy(54,7);Readln(digital[2]);
  y[2] := length ( digital [2] );
  if (digital[2] = '1' ) then gital[2] := 1;
  if (digital[2] = '0' ) then gital[2] := 0;
  Until ((y[2] = 1) and (digital[2] = '1')
  or (digital[2] = '0'));
```

```
  conteo;
```

```
End;
```

```
Begin
```

```
Repeat
```

```
  Gotoxy(54,9);Writeln(' ':10);
  Gotoxy(15,9);
  Writeln('CANAL DIGITAL 3 (1/0).....');
  Gotoxy(54,9);Readln(digital[3]);
  y[3] := length ( digital [3] );
  if (digital[3] = '1' ) then gital[3] := 1;
  if (digital[3] = '0' ) then gital[3] := 0;
  Until ((y[3] = 1) and (digital[3] = '1')
  or (digital[3] = '0'));
```

```
  conteo;
```

```
End;
```

```
Begin
```

```
Repeat
```

```
  Gotoxy(54,11);Writeln(' ':10);
  Gotoxy(15,11);
```

```

Writeln('CANAL DIGITAL 4 (1/0).....');
Gotoxy(54,11);Readln(digital[4]);
y[4] := length ( digital [4] );
if (digital[4] = '1' ) then gital[4] := 1;
if (digital[4] = '0' ) then gital[4] := 0;
Until ((y[4] = 1) and (digital[4] = '1')
or (digital[4] = '0'));
conteo;
End;

Begin
Repeat
Gotoxy(54,13);Writeln(' ':10);
Gotoxy(15,13);
Writeln('CANAL DIGITAL 5 (1/0).....');
Gotoxy(54,13);Readln(digital[5]);
y[5] := length ( digital {5} );
if (digital[5] = '1' ) then gital[5] := 1;
if (digital[5] = '0' ) then gital[5] := 0;
Until ((y[5] = 1) and (digital[5] = '1')
or (digital[5] = '0'));
conteo;
End;

Begin
Repeat
Gotoxy(54,15);Writeln(' ':10);
Gotoxy(15,15);
Writeln('CANAL DIGITAL 6 (1/0).....');
Gotoxy(54,15);Readln(digital[6]);
y[6] := length ( digital [6] );
if (digital[6] = '1' ) then gital[6] := 1;
if (digital[6] = '0' ) then gital[6] := 0;
Until ((y[6] = 1) and (digital[6] = '1')
or (digital[6] = '0'));
conteo;
End;

Begin
Repeat
Gotoxy(54,17);Writeln(' ':10);
Gotoxy(15,17);
Writeln('CANAL DIGITAL 7 (1/0).....');
Gotoxy(54,17);Readln(digital[7]);
y[7] := length ( digital [7] );
if (digital[7] = '1' ) then gital[7] := 1;
if (digital[7] = '0' ) then gital[7] := 0;
Until ((y[7] = 1) and (digital[7] = '1')
or (digital[7] = '0'));
conteo;

```

```

End;

Begin
Repeat
  Gotoxy(54,19);Writeln(' ':10);
  Gotoxy(15,19);
  Writeln('CANAL DIGITAL 8 (1/0).....');
  Gotoxy(54,19);Readln(digital[8]);
  y[8] := length ( digital [8] );
  if (digital[8] = '1' ) then gital[8] := 1;
  if (digital[8] = '0' ) then gital[8] := 0;
  Until ((y[8] = 1) and (digital[8] = '1')
  or (digital[8] = '0'));
  conteo;
end;
end;

Procedure conteol;
Begin
qdatol := ((1 * gital[9]) + (2 * gital[10]) + (4 * gital[11])
+ (8 * gital[12]) + (16 * gital[13]) + (32 * gital[14])
+ (64 * gital[15]) + (128 * gital[16]));
Port[$321] := qdatol;
End;

Procedure puertob;
Begin
Clrscr;
Gotoxy(20,2);
Writeln('ACTIVO = 1','':7,'DESACTIVO = 0');
Gotoxy(16,21);
Writeln('SE PROGRAMA ESTOS CANALES CUANDO SE ASIGNAN');
Gotoxy(21,22);
Writeln('LOS PRIMEROS 8 CANALES DIGITALES');

Begin
Repeat
  Gotoxy(54,5);Writeln(' ':10);
  Gotoxy(15,5);
  Writeln('CANAL DIGITAL 9 (1/0).....');
  Gotoxy(54,5);Readln(digital[9]);
  y[9] := length ( digital [9] );
  if (digital[9] = '1' ) then gital[9] := 1;
  if (digital[9] = '0' ) then gital[9] := 0;
  Until ((y[9] = 1) and (digital[9] = '1')
  or (digital[9] = '0'));
  conteol;
End;

Begin

```

```

Repeat
  Gotoxy(54,7);Writeln(' ':10);
  Gotoxy(15,7);
  Writeln('CANAL DIGITAL 10 (1/0).....');
  Gotoxy(54,7);Readln(digital[10]);
  y[10] := length ( digital [10] );
  if (digital[10] = '1' ) then gital[10] := 1;
  if (digital[10] = '0' ) then gital[10] := 0;
  Until ((y[10] = 1) and (digital[10] = '1')
  or (digital[10] = '0'));
  conteol;
End;

Begin;
Repeat
  Gotoxy(54,9);Writeln(' ':10);
  Gotoxy(15,9);
  Writeln('CANAL DIGITAL 11 (1/0).....');
  Gotoxy(54,9);Readln(digital[11]);
  y[11] := length ( digital [11] );
  if (digital[11] = '1' ) then gital[11] := 1;
  if (digital[11] = '0' ) then gital[11] := 0;
  Until ((y[11] = 1) and (digital[11] = '1')
  or (digital[11] = '0'));
  conteol;
End;

Begin
Repeat
  Gotoxy(54,11);Writeln(' ':10);
  Gotoxy(15,11);
  Writeln('CANAL DIGITAL 12 (1/0).....');
  Gotoxy(54,11);Readln(digital[12]);
  y[12] := length ( digital [12] );
  if (digital[12] = '1' ) then gital[12] := 1;
  if (digital[12] = '0' ) then gital[12] := 0;
  Until ((y[12] = 1) and (digital[12] = '1')
  or (digital[12] = '0'));
  conteol;
End;

Begin
Repeat
  Gotoxy(54,13);Writeln(' ':10);
  Gotoxy(15,13);
  Writeln('CANAL DIGITAL 13 (1/0).....');
  Gotoxy(54,13);Readln(digital[13]);
  y[13] := length ( digital [13] );
  if (digital[13] = '1' ) then gital[13] := 1;
  if (digital[13] = '0' ) then gital[13] := 0;

```

```

Until ((y[13] = 1) and (digital[13] = '1')
or (digital[13] = '0'));
conteol;
End;

Begin
Repeat
Gotoxy(54,15);Writeln(' ':10);
Gotoxy(15,15);
Writeln('CANAL DIGITAL 14 (1/0).....');
Gotoxy(54,15);Readln(digital[14]);
y[14] := length ( digital [14] );
if (digital[14] = '1' ) then gital[14] := 1;
if (digital[14] = '0' ) then gital[14] := 0;
Until ((y[14] = 1) and (digital[14] = '1')
or (digital[14] = '0'));
conteol;
End;

Begin
Repeat
Gotoxy(54,17);Writeln(' ':10);
Gotoxy(15,17);
Writeln('CANAL DIGITAL 15 (1/0).....');
Gotoxy(54,17);Readln(digital[15]);
y[15] := length ( digital [15] );
if (digital[15] = '1' ) then gital[15] := 1;
if (digital[15] = '0' ) then gital[15] := 0;
Until ((y[15] = 1) and (digital[15] = '1')
or (digital[15] = '0'));
conteol;
End;

Begin
Repeat
Gotoxy(54,19);Writeln(' ':10);
Gotoxy(15,19);
Writeln('CANAL DIGITAL 16 (1/0).....');
Gotoxy(54,19);Readln(digital[16]);
y[16] := length ( digital [16] );
if (digital[16] = '1' ) then gital[16] := 1;
if (digital[16] = '0' ) then gital[16] := 0;
Until ((y[16] = 1) and (digital[16] = '1')
or (digital[16] = '0'));
conteol;
End;
End;

Procedure puertocen;

```

```

Begin
  n := 1 ;
repeat
  digitale[n] := 0 ;
  n := n+1 ;
until (n = 9) ;

  N := 8;
  Qdatoel := (Qdato2/2);
  Qdatoel := (Qdatoel * 2) ;
  Repeat
  Qdatobl := qdatoel;
  Qdatoel := (Qdatobl/2);
  Qdatoell := Trunc(qdatoel);
  Qdatoll := (Qdatoell * 2);
  Digitale[n] := (Qdatobl - Qdatoll);
  Qdatoel := Qdatoell;
  N := n-1;
  If ((qdatobl = 1) or (qdatobl = 0))
  then digitale[1] := qdatobl ;
  Until (qdatobl = 1) or (qdatobl = 0);

```

end;

Procedure puertoc;

```

Begin
  Clrscr;
  Gotoxy(20,2);Writeln('ACTIVO = 1','':7,'DESACTIVO = 0');

  Puertocen;
  Gotoxy(54,5);Writeln('':10);
  If (digitale[1] = 1) then digitale1 := '1';
  If (digitale[1] = 0) then digitale1 := '0';
  Gotoxy(15,5);
  Writeln('CANAL DIGITAL 1 (1/0).....',digitale1);

  Gotoxy(54,7);Writeln('':10);
  If (digitale[2] = 1) then digitale2 := '1';
  If (digitale[2] = 0) then digitale2 := '0';
  Gotoxy(15,7);
  Writeln('CANAL DIGITAL 2 (1/0).....',digitale2);

  Gotoxy(54,9);Writeln('':10);
  If (digitale[3] = 1) then digitale3 := '1';
  If (digitale[3] = 0) then digitale3 := '0';
  Gotoxy(15,9);
  Writeln('CANAL DIGITAL 3 (1/0).....',digitale3);

  Gotoxy(54,11);Writeln('':10);

```

```

If (digitale[4] = 1) then digitale4 := '1';
If (digitale[4] = 0) then digitale4 := '0';
Gotoxy(15,11);
Writeln('CANAL DIGITAL 4 (1/0).....',digitale4);

Gotoxy(54,13);Writeln(' ':10);
If (digitale[5] = 1) then digitale5 := '1';
If (digitale[5] = 0) then digitale5 := '0';
Gotoxy(15,13);
Writeln('CANAL DIGITAL 5 (1/0).....',digitale5);

Gotoxy(54,15);Writeln(' ':10);
If (digitale[6] = 1) then digitale6 := '1';
If (digitale[6] = 0) then digitale6 := '0';
Gotoxy(15,15);
Writeln('CANAL DIGITAL 6 (1/0).....',digitale6);

Gotoxy(54,17);Writeln(' ':10);
If (digitale[7] = 1) then digitale7 := '1';
If (digitale[7] = 0) then digitale7 := '0';
Gotoxy(15,17);
Writeln('CANAL DIGITAL 7 (1/0).....',digitale7);

Gotoxy(54,19);Writeln(' ':10);
If (digitale[8] = 1) then digitale8 := '1';
If (digitale[8] = 0) then digitale8 := '0';
Gotoxy(15,19);
Writeln('CANAL DIGITAL 8 (1/0).....',digitale8);

end;

      { Procedimiento que nos permite limpiar la pantalla }
      { y salir al sistema operativo. }
Procedure salir;
Begin
  Clrscr;
End;

Begin
      { localidades que se asignaron a cada puerto }

puerto0 := $300;
puerto1 := $302;
puerto2 := $304;
puerto3 := $306;
puerto4 := $308;
puerto5 := $30A;
puerto6 := $30C;
puerto7 := $30E;

```

```

puerto8 := $310;
puerto9 := $312;
puerto10 := $314;
puerto11 := $316;
puerto12 := $318;
puerto13 := $31A;
puerto14 := $31C;
puerto15 := $31E;
pcont1 := $320;
pcont2 := $321;
pcont3 := $322;
vref := 5;

```

```

      { inicializacion del puerto }

```

```

Port[$323] := $89; { 1 0 0 0 1 0 0 1 }

```

```

      { palabra de control del contador 1 }

```

```

Port[$32B] := $78; { 0 1 1 1 1 0 0 0 }

```

```

      { escritura al contador 1 con valor cero }
      { para el byte menos significativo }

```

```

Port[$329] := $FF;

```

```

Port[$329] := $FF;

```

```

      { palabra de control del contador 2 }

```

```

Port[$32B] := $B8; { 1 0 1 1 1 0 0 0 }

```

```

      { escritura al contador 2 con valor cero }
      { para el byte menos significativo }

```

```

Port[$32A] := $FF;

```

```

Port[$32A] := $FF;

```

```

      { palabra de control del contador 0 }

```

```

Port[$32B] := $38; { 0 0 1 1 1 0 0 0 }

```

```

      { escritura al contador 0 con valor cero }
      { para el byte menos significativo }

```

```

Port[$328] := $FF;

```

```

Port[$328] := $FF;

```

```

M :=1;

```

```

repeat

```

```

  gital[m] := 0;

```

```

  m := m+1;

```

```

  until (m =17);

```

```

Presentacion;

```

```

repeat

```

```

  Gotoxy(16,22);Writeln('OPRIMA G Y <RETURN> PARA CONTINUAR');

```

```

Gotoxy(58,22);Writeln('':10);
Gotoxy(58,22);Readln(opc);
x[1] := length ( opc );
Until (( x[1] = 1 ) and ((opc = 'G' ) or (opc = 'g')));
Gotoxy(55,9);

```

Regresar:

Mascara;

Repeat

```

Gotoxy(55,9);Writeln('':10);
Gotoxy(55,9);Readln (OPCION);
x[2] := length (OPCION);
if ((opcion = 'a') or ( opcion = 'A')) then opciona := 'A';
if ((opcion = 'b') or ( opcion = 'B')) then opciona := 'B';
if ((opcion = 'c') or ( opcion = 'C')) then opciona := 'C';
Until (( x[2] = 1 ) and ((opcion = 'a') or (opcion = 'A') or
(opcion = 'b') or (opcion = 'B') or ( opcion = 'c') or
(opcion = 'C')));

```

doss:

Case Opciona of

'A', 'a' :

Begin

Analogicos;

Repeat

Gotoxy(53,5);Writeln('':10);

Gotoxy(53,5);Readln(opcion1);

x[3] := length (opcion1);

if ((opcion1 = 'a') or (opcion1 = 'A')) then opcionla := 'A';

if ((opcion1 = 'b') or (opcion1 = 'B')) then opcionla := 'B';

if ((opcion1 = 'c') or (opcion1 = 'C')) then opcionla := 'C';

if ((opcion1 = 'd') or (opcion1 = 'D')) then opcionla := 'D';

if ((opcion1 = 'e') or (opcion1 = 'E')) then opcionla := 'E';

Case opcionla of

'a', 'A':

Begin

Escritura;

Escritural;

Goto doss;

End;

'B', 'b':

Begin

```

Repeat
  Clrscr;
  uno:
  {$I-}
  Repeat
    Gotoxy(41,10);Writeln(' ':10);
    Gotoxy(15,10);Writeln('EL CANAL A PROGRAMAR ES ');
    Gotoxy(41,10);Readln(puerto);
    ipuerto := ioreult ;
    If (( ipuerto <> 0 ) or (puerto >17 ))then
      Begin
        Gotoxy(15,12);Writeln(' EL DATO NO ES CORRECTO' );
        Delay(1000);
        Gotoxy(15,12);writeln( ' ':80);
      end;
  Until ((ipuerto = 0 ) and (puerto < 16));

```

```

{$I+}
{$I-}
Repeat
  Gotoxy(15,15);Writeln('EL VOLTAJE A PROGRAMAR ES ');
  Gotoxy(44,15);Readln(voltaje);
  voltajei := ioreult ;
  If (voltajei < 0) then mensaje;
  If ((voltaje > -5.0) and (voltaje > 4.9975 )) then
    mensaje;
  Until ((voltajei = 0 ) and (voltaje >= -5.000) and
    (voltaje <= 4.9975));
  {$I+}
  dato0 := (2048*(voltaje/vref));
  dato := (2048 + trunc(dato0));
  if (puerto = 1) then Portw[$300] := dato;
  if (puerto = 2) then Portw[$302] := dato;
  if (puerto = 3) then Portw[$304] := dato;
  if (puerto = 4) then Portw[$306] := dato;
  if (puerto = 5) then Portw[$308] := dato;
  if (puerto = 6) then Portw[$30A] := dato;
  if (puerto = 7) then Portw[$30C] := dato;
  if (puerto = 8) then Portw[$30E] := dato;
  if (puerto = 9) then Portw[$310] := dato;
  if (puerto = 10) then Portw[$312] := dato;
  if (puerto = 11) then Portw[$314] := dato;
  if (puerto = 12) then Portw[$316] := dato;
  if (puerto = 13) then Portw[$318] := dato;
  if (puerto = 14) then Portw[$31A] := dato;
  if (puerto = 15) then Portw[$31C] := dato;
  if (puerto = 16) then Portw[$31E] := dato;

```

```

        Gotoxy(15,17);WriteLn('DESEAS PROGRAMAR OTRO CANAL
        (S/N)');
        Gotoxy(49,17);ReadLn(CANAL);
    Until ((CANAL = 'N') or (CANAL = 'n'));
    Goto doss;
End;
'C', 'c':
Begin

    Lectural;
    Lectura2;
    Llecl;
    Goto doss;

End;
'd', 'D' :
Begin
    Repeat
        Clrscr;
        {$I-}
        Repeat
            Gotoxy(15,10);WriteLn('EL CANAL A LEER ES ');
            Gotoxy(34,10);WriteLn(' ':10);
            Gotoxy(41,10);ReadLn(puertor);
            puertori := ioresult ;
            If (( puertori <> 0 ) or (puertor > 17 ))then
                Begin
                    Gotoxy(15,12);WriteLn(' EL DATO NO ES CORRECTO' )
                    Delay(1000);
                    Gotoxy(15,12);writeLn( ' ':80);
                    end;
                Until (( puertori = 0 ) and (puertor < 17 ));
            {$I+}
            if (puertor = 1) then puertore := $300;
            if (puertor = 2) then puertore := $302;
            if (puertor = 3) then puertore := $304;
            if (puertor = 4) then puertore := $306;
            if (puertor = 5) then puertore := $308;
            if (puertor = 6) then puertore := $30A;
            if (puertor = 7) then puertore := $30C;
            if (puertor = 8) then puertore := $30E;
            if (puertor = 9) then puertore := $310;
            if (puertor = 10) then puertore := $312;
            if (puertor = 11) then puertore := $314;
            if (puertor = 12) then puertore := $316;
            if (puertor = 13) then puertore := $318;
            if (puertor = 14) then puertore := $31A;
            if (puertor = 15) then puertore := $31C;
            if (puertor = 16) then puertore := $31E;
            datorr := Portw[puertori];

```

```

datorr := NOT(datorr);
Ddatorr := 4095 - Ddato0 ;
datorr := ddatorr and ($OFFF);
dator := ((datoRR- 2048)/2048)*5;
Gotoxy(15,14);Writeln('SU VOLTAJE ES : ',dator:6:4)
Gotoxy(15,17);Writeln('DESEAS LEER OTRO CANAL
(S/N)');

Gotoxy(49,17);Readln(CANAL);
Until ((CANAL = 'N') or (CANAL = 'n'));

Goto doss;

End;

'e', 'E' :

Begin
Goto regresar;
End;
End;
Until (( X[3] = 1 ) and (( opcion1 > 'f' ) OR
(opcion1 > 'F')));

End;

'B', 'b':
tres:
Begin
Digitales;

Repeat
Gotoxy(53,5);Writeln('':10);
Gotoxy(53,5);Readln(opcion2);
X[4] := length ( opcion2 );
if ((opcion2 = 'a') or (opcion2 = 'A')) then opcion2a := 'A';
if ((opcion2 = 'b') or (opcion2 = 'B')) then opcion2a := 'B';
if ((opcion2 = 'c') or (opcion2 = 'C')) then opcion2a := 'C';
if ((opcion2 = 'd') or (opcion2 = 'D')) then opcion2a := 'D';
if ((opcion2 = 'e') or (opcion2 = 'E')) then opcion2a := 'E';
if ((opcion2 = 'f') or (opcion2 = 'F')) then opcion2a := 'F';
if ((opcion2 = 'g') or (opcion2 = 'G')) then opcion2a := 'G';
Case opcion2a of
'a', 'A' :
Begin
Contador1 ;
Gotoxy(15,20);
Writeln('OPRIMA G PARA CONTINUAR');

```

```

Repeat
  Gotoxy(40,20);Writeln(' ');
  Gotoxy(40,20);Readln(CONTP1);
Until ((CONTP1 = 'G' ) OR (CONTP1 = 'g'));

Goto tres;

End;
'b', 'B' :
Begin
  Contador2 ;

  Gotoxy(15,20);

  Writeln('OPRIMA G PARA CONTINUAR');

  Repeat
    Gotoxy(40,20);Writeln(' ');
    Gotoxy(40,20);Readln(CONTP2);
  Until ((CONTP2 = 'G' ) OR ( CONTP2 = 'g' ));

  Goto tres;
End;
'c', 'C' :
Begin
  Contador3 ;

  Gotoxy(15,24);

  Writeln('OPRIMA G PARA CONTINUAR');

  Repeat
    Gotoxy(40,20);Writeln(' ');
    Gotoxy(40,20);Readln(CONTP3);
  Until ((contp3 = 'G' ) OR ( contp3 = 'g'));

  Goto tres;
End;

'D', 'd' :
Begin
  Clrscr;

  Qdato2 := port[$326];
  puertoc;

  Gotoxy(18,24);Writeln('OPRIMA G Y <RETURN> PARA CONTINUAR
Repeat
  Gotoxy(18,25);Writeln(' ');
  Gotoxy(18,25);Readln(lpuertoc);

```

```

Until (( lpuertoc = 'G') or ( lpuertoc = 'g'));
Goto tres;
End;

'E', 'e' :
Begin
  Clrscr;
  puertoa;
  Gotoxy(18,24);Writeln('OPRIMA G Y <RETURN> PARA
CONTINUAR');

  Repeat
    Gotoxy(18,25);Writeln(' ');
    Gotoxy(18,25);Readln(lpuertoa);
  Until (( lpuertoa = 'G') or ( lpuertoa = 'g'));
  Goto tres;
End;

'F', 'F' :
Begin
  Clrscr;
  puertob;

  Gotoxy(18,24);Writeln('OPRIMA G Y <RETURN> PARA
CONTINUAR');

  Repeat
    Gotoxy(18,25);Writeln(' ');
    Gotoxy(18,25);Readln(lpuertob);
  Until (( lpuertob = 'G') or ( lpuertob = 'g'));
  Goto tres;
End;

'G', 'g' : Goto regresar;
End;
Until ((x[4] = 1) and ((opcion2 > 'B') or ( opcion2 > 'g')));
End;

'C', 'c': Begin
  Clrscr;
  Goto exit;
End;

End;
exit:
End.

```

ANEXO B

Diagramas electrónicos del módulo de conexiones

Este anexo contiene los diagramas electrónicos y de disposición de componentes del módulo de conexiones.

Acoplamiento digital.

Acoplamiento analógico.

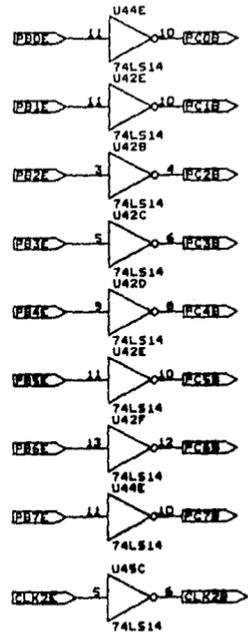
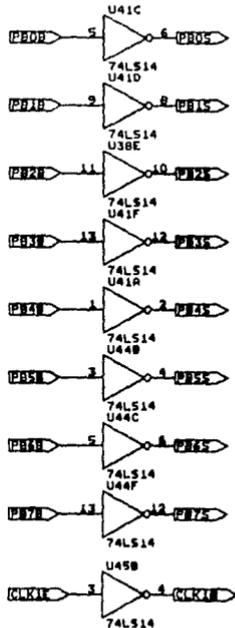
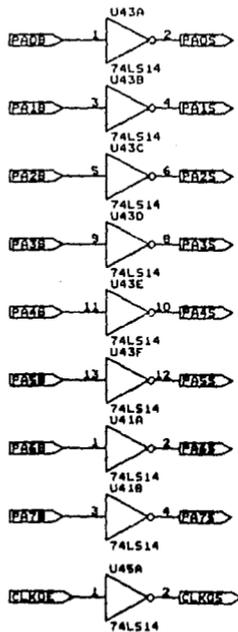
Conectores de la interfaz Digital-Analógico.

Conectores de entrada del módulo de acoplamiento para canales digitales.

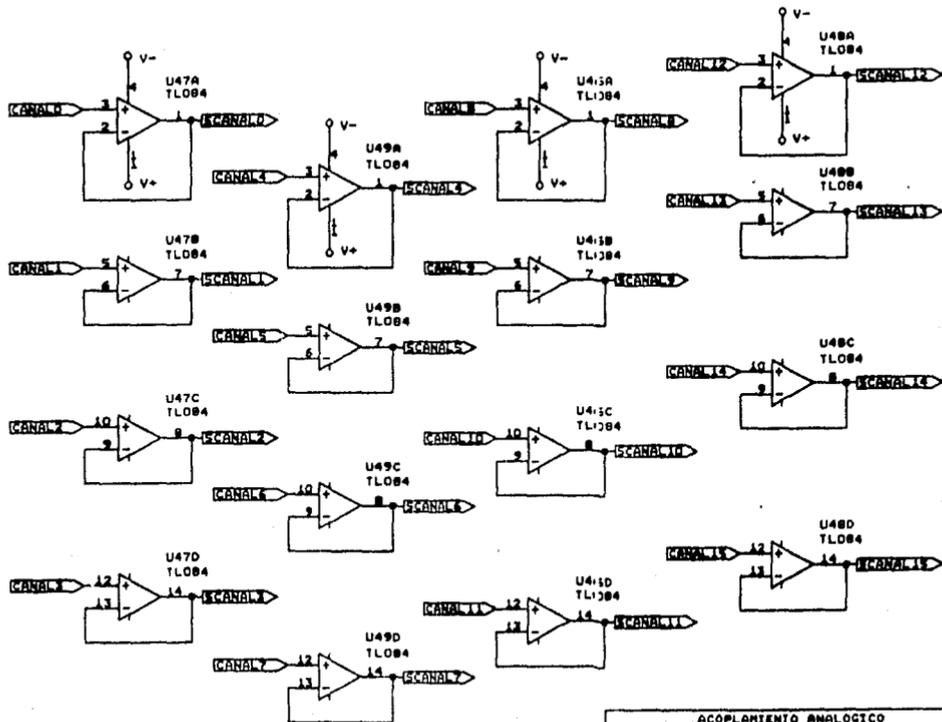
Conectores de entrada del módulo de acoplamiento para canales digitales.

Conectores de salida del módulo de acoplamiento para canales digitales.

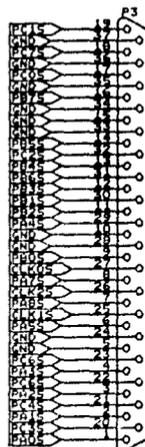
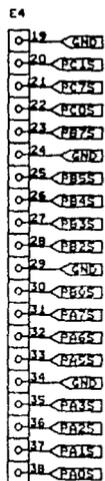
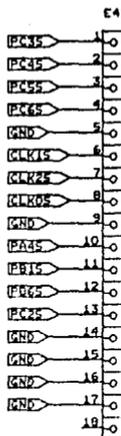
Esquema general.



ACOPLEMIENTO DIGITAL			
MÓDULO EXTERNO			
INSTITUTO DE INGENIERIA	Tamaño	Nº.	REV
UNAM	A	1	1.1
Fron. B120 Facultad / 3 / 1988 Hoja 3 de 10			

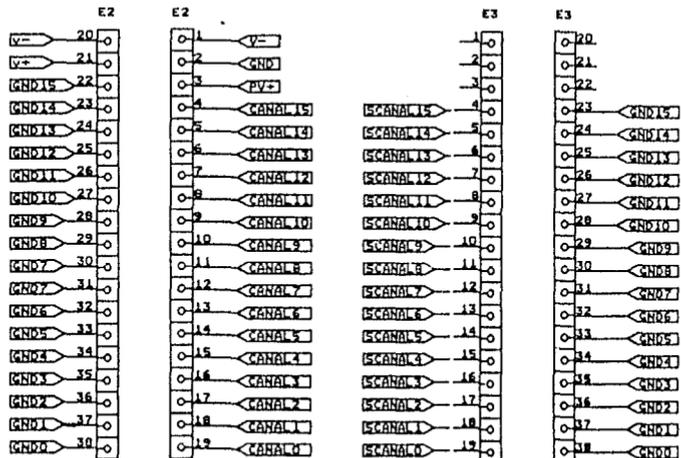


ACOPLANAMIENTO ANALOGICO MODULO EXTERNO			
INSTITUTO DE INGENIERIA	TOMO NO.	NO.	REV
UNAH	A	1	1.1
Proy. 8128 Fecha: 11/3/1989	Hoja 9	de	10

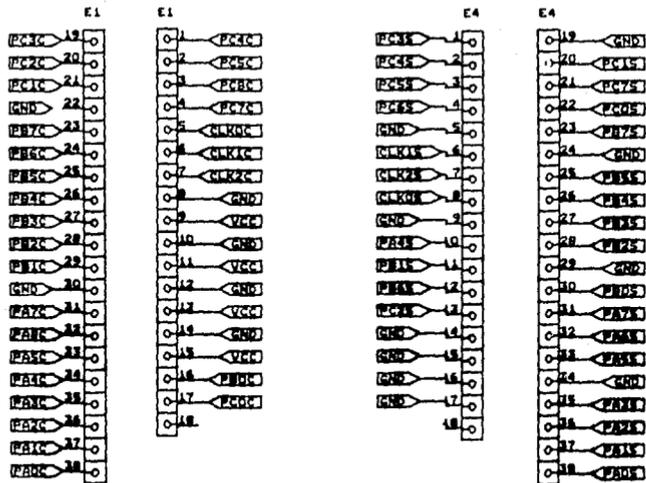


D837

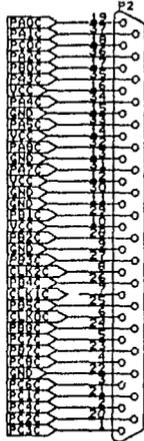
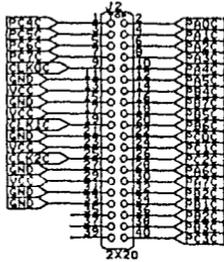
CONECTORES DEL PUERTO Y TIMER			
MODULO EXTERNO			
INSTITUTO DE INGENIERIA	Tamaño	NO.	REV
UNAM	A	1	1.1
Proy. 8120	Fschell/3	2/89	Hoja 9 de 10



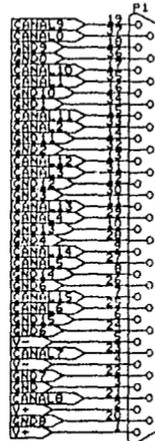
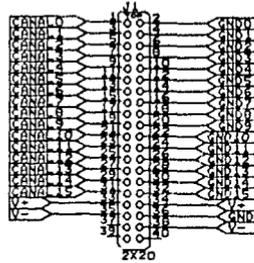
CONECTORES DE LOS AMPLIFICADORES			
MODULO EXTERNO			
INSTITUTO DE INGENIERIA	TOMO NO	NO.	REV
UNAM	A	1	1.1
Cov. 8120 Pichalls / 3 1989 Hoja 9 de 10			



CONECTORES MODULO EXTERNO			
INSTITUTO DE INGENIERIA Y MAN	Tamaño	NO.	REV
	#	1	1.1
Proy. B120 Fecha: 13/3/1985	Mo. 1	9	de 10



DB37



DB37

CONECTORES MODULO EXTERNO			
INSTITUTO DE INGENIERIA UNAH	Tamaño A	NO. 1	REV 1.1
Proy. 8120 Fecha: 13/1/1982 Hoja 9 de 10			

ANEXO C

Distribución Física del módulo de conexiones y mapa de componentes de la interfaz y módulo de conexión.

Este anexo contiene los diagramas de distribución física del módulo de conexiones.

Esquema general del módulo de conexiones.

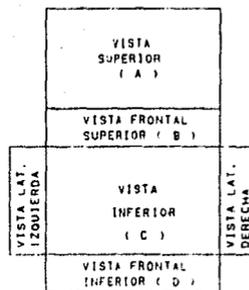
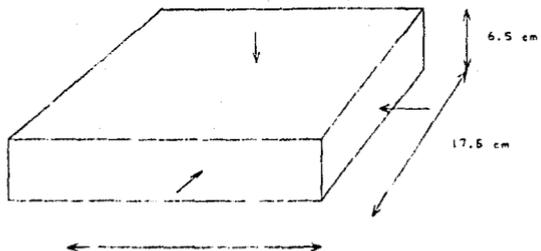
Vista frontal superior.

Vista lateral izquierda y derecha.

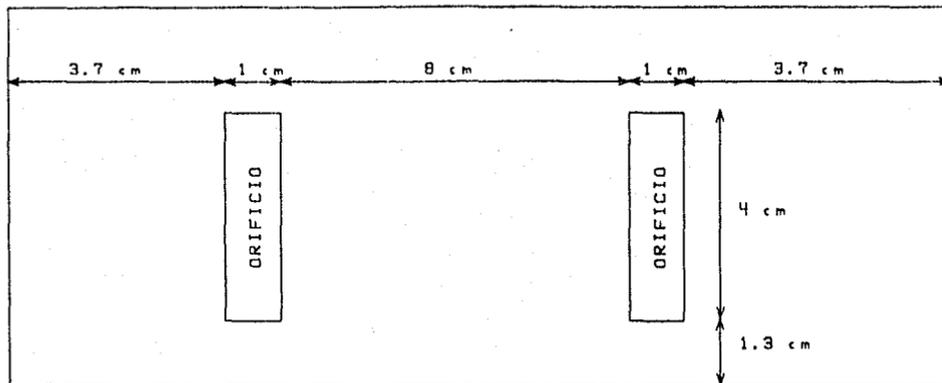
Vista frontal inferior.

Mapa de componentes de la interfaz.

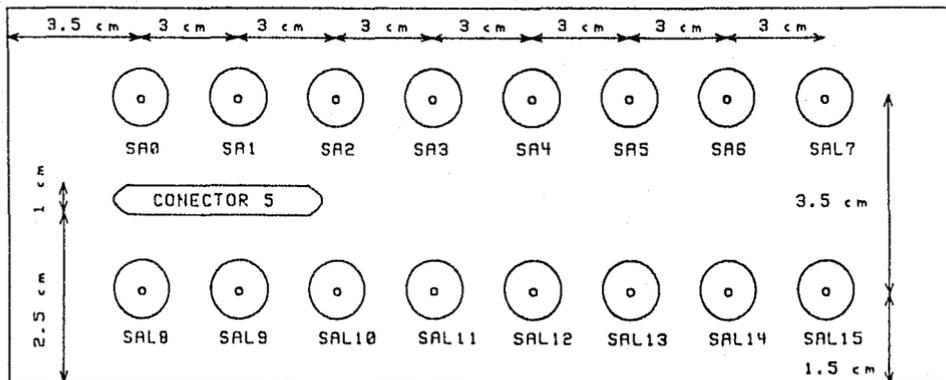
Mapa de componentes del módulo de conexiones.



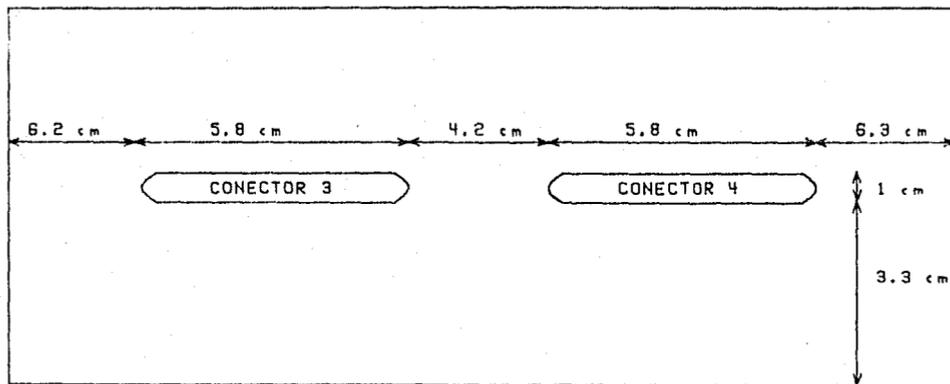
ESQUEMA GENERAL DEL MÓDULO DE CONEXIONES



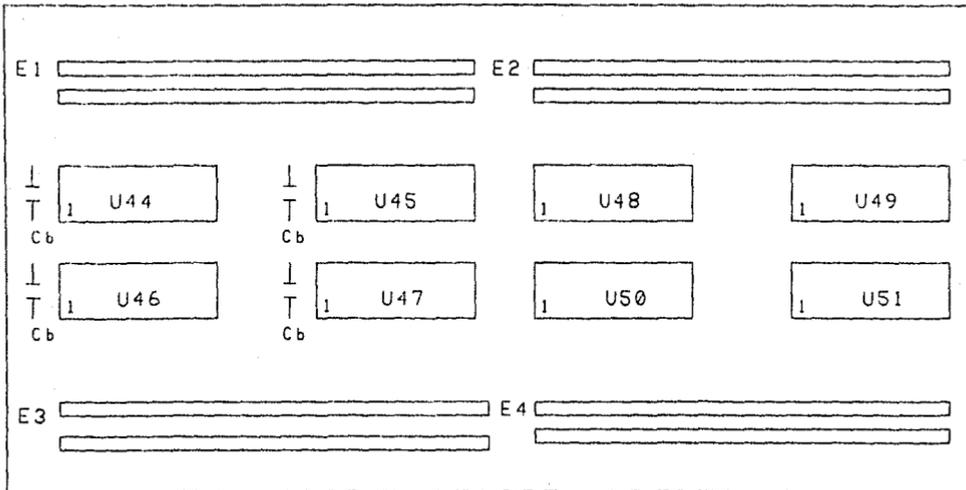
VISTA LATERAL IZQUIERDA (E) , DERECHA (F)
DEL MODULO DE CONEXIONES



VISTA FRONTAL INFERIOR (D)
DEL MODULO DE ACOPLAMIENTO



VISTA FRONTAL SUPERIOR (B)
DEL MODULO DE CONEXIONES



ANEXO D

LISTA DE COMPONENTES

LISTA DE COMPONENTES DE LA TARJETA INTERFAZ

LISTA DE MATERIALES.

CANTIDAD	REFERENCIA	ESPECIFICACION
3	U1, U34, U36,	74LS244
1	U2	DAC1230
1	U5	74LS85
3	U10, U31, U15	74LS04
2	U14, U19	LF355N
2	U22, U26	74LS02
4	U21, U23, U24, U32	74LS00
3	U3, U8, U13	7489
3	U2, U7, U20	74LS240
1	U18, U17	4051
2	U6A, U6B, U11B	74LS73
1	U4	74LS193
1	U9	74LS155
2	U16, U27, U28,	TL084
1	SW1	SWICHT PROGRAMABLE
4	R13, R14, R15 R16	4K7 W
12	R1, R2, R3, R4 R5, R6, R7, R8 R9, R10, R11, R12	6K8 W
2	R9, R43	820 W
4	R13, R14, R15 R16	4K7 W

CANTIDAD	REFERENCIA	ESPECIFICACION
2	R17, POT1	10K W
1	R20	8K2 W
1	R21	2K7 W
1	POT3	20K W
1	R16	1M W
1	R17	10K W
12	R17, R18, R19, R20, R21, R22, R23, R24, R25, R26, R27, R28, R49, R45', POT2	1K W
1	R47	47 W
1	R42	8.2 W
1	R44	560 W
1	R45	15K W
1	R41	56K W
1	POT4	5K W
	U29, U28, U30	
1	C5, C6, C23, C24	22 MF
17	C7, C8, C9, C10, C11, C12 C13, C14, C15 C16, C17, C18 C19, C20, C21, C22, C23	3.3 nf
4	D3, D4, D5 D6	6.8 V

CANTIDAD	REFERENCIA	ESPECIFICACION
2	R37, R38, R39 R40	15K W
9	R17, R18, R19, R20, R49, R50, R51, R52, R53	1K W

ANEXO E

ESPECIFICACIONES TECNICAS

INTERFAZ IICADA -II

Características

<u>Descripción</u>	Interfaz para el manejo de senales analógicas y digitales IBM - PC compatible, con un periodo de muestreo establecido y tres contadores de eventos externos.
<u>Número de canales</u> analógico	16 para salida.
digitales	16 para salida y 8 para entrada.
Acoplamiento	TTL-LS para los canales de E/S baja impedancia para los canales analógicos.
<u>Entrada:</u>	Memoria interna de la interfaz.
<u>Rango de voltaje</u>	+/- 5.000 VDC.
<u>Resolución para cada canal</u>	12 bites.
<u>Tiempo de conversión</u>	3.2 us por canal.
<u>Orden de conversión</u>	Interna (por programa).
<u>Polarización externa</u>	No requiere.
<u>Presentación</u>	Tarjeta de conversión D/A insertada en las ranuras disponibles para el usuario dentro de la PC y módulo de conexiones.

Tecnología

Rapida y de bajo consumo(LS, CMOS).

Polarización

+5 VDC.

+12 VDC.

-12 VDC.

Programa

Programa de presentación desarrollado en los lenguajes Pascal-Ensamblador.