



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE ADQUISICIÓN Y DESPLIEGUE
DE SEÑALES BIOMÉDICAS CARDÍACAS

TESIS PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACION
P R E S E N T A :
RAUL RAUDRY DIAZ DE LEON

ASESOR ING. RAUL ROSETTE URIBE



NOVIEMBRE 1991

FALLA

GEN



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.

TESIS CON FALLA DE ORIGEN

INDICE

PRESENTACION.....	1
1 PRINCIPIOS BASICOS DE ELECTROCARDIOGRAFIA.....	6
1.1 Fisiologia del corazón.....	6
1.2. Actividad eléctrica del corazón.....	8
1.3. Electrocardiograma.....	10
1.4. Interpretación.....	13
2. ELEMENTOS DE LA COMPUTADORA EMPLEADOS EN EL PROYECTO.....	18
2.1. Hardware de soporte para la realización de gráficos.....	18
2.1.1. Tarjetas de video.....	21
2.1.2. Funcionamiento de las tarjetas de video.....	22
2.1.3. Mapa de memoria para despliegue.....	25
2.1.4. Direccionamiento directo a memoria.....	27
2.1.5. Monitores.....	30
2.2. Slot de entrada/salida.....	33
2.2.1. Señales que constituyen al slot de entrada/salida.....	34
3. DISEÑO DE LA TARJETA DIGITALIZADORA DE SEÑALES.....	42
3.1. Sección de decodificación de direcciones.....	43
3.1.1. Diseño.....	45

3.2	Sección de conversión de datos digitales a analógicos	50
3.2.1	Descripción del circuito convertidor	51
3.2.2	Configuración	53
3.2.3	Conexión del ADC084 con la computadora	56
3.3	Sección de cálculo de frecuencia	59
3.3.1	Circuito básico de cálculo de frecuencia	59
3.3.2	Adaptación de la señal de entrada	69
3.3.3	Indicador de sobreflujo	72
3.3.4	Conexión con el computador	73
3.4	Sección de conversión de datos analógicos a digitales	77
3.4.1	Descripción del circuito convertidor	78
3.4.2	Configuración	79
3.4.3	Conexión del DAC080 con la computadora	81
3.5	Sección de salida de datos digitales	84
3.5.1	Diseño	84
3.6	Integración de secciones	85
3.7	Interfaz con el usuario	87
4	ANÁLISIS DE LA CAPACIDAD DE DIGITALIZACIÓN DE LA TARJETA	91
4.1	Teorema del muestreo de Nyquist	92
4.2	Consideraciones prácticas para la recuperación de señales muestreadas	98
4.3	Capacidad de la tarjeta para la digitalización de señales	100

5	DISEÑO DEL SISTEMA DE ELECTROCARDIOGRAFÍA	102
5.1	Sistema básico de graficación en tiempo real	103
5.2	Desarrollo del sistema de electrocardiografía	106
5.2.1	Diagramas de flujo	107
5.2.2	Listado del sistema de electrocardiografía	116
5.3	Guía del usuario	135
6	ANÁLISIS DE RESULTADOS	147
6.1	Sección de cálculo de frecuencia	148
6.2	Sección de conversión de datos analógicos a digitales	151
6.3	Sección de conversión de datos digitales a analógicos	154
6.4	Pruebas IN-VIVO	156
7	OTRAS APLICACIONES	161
7.1	Control de velocidad para un motor de D.C.	161
7.2	Medidor de capacitancias	163
7.3	Generador programable de señales	165
7.4	Interfase de voz	166
7.5	reconocimiento de voz	167
	CONCLUSIONES	171
	APÉNDICES	174
A	Convertidor de datos digitales a analógicos ADC804	175
B	Convertidor de datos analógicos a digitales DAC830	180

C. Temporizador LM555	185
D. Coeficiente de correlación lineal	189
E. Diagramas de terminales y tablas de verdad de los circuitos con baja y mediana escala de integración	190
BIBLIOGRAFIA	199

PRESENTACION

PRESENTACION

Actualmente, el empleo de las computadoras personales (PCs) no se limita tan sólo al desarrollo de aplicaciones que requieren de la captura manual de información (posibles bases de datos, etc). Hoy en día, es posible alimentar automáticamente a una computadora con imágenes sonoras, topográficas y en general con todo tipo de señales o variables físicas, las cuales, una vez grabadas en memoria, pueden ser analizadas sin que para ello tenga que intervenir la presencia humana.

Una vez en memoria, el tratamiento dado a los datos puede ser muy variado dependiendo de la aplicación. Si se trata de voz o imágenes, por ejemplo, es posible reconocerlos y realizar algún tipo de acción de acuerdo con el resultado obtenido en este proceso. Asimismo, en aplicaciones de control, la computadora puede examinar los datos o realizar cálculos con ellos para determinar las salidas a generar.

Sin embargo, las aplicaciones de almacenamiento y procesamiento de variables físicas no se limitan tan sólo a su reconocimiento o control. En el campo de la medicina, el análisis de las señales cardíacas puede suministrar información muy valiosa acerca del estado de salud de un individuo.

Viendo la importancia que tiene el procesamiento de señales por computadora se decidió llevar a cabo un proyecto en donde se dotaría a una PC con la capacidad de leer, almacenar y desplegar cualquier tipo de información o variable física.

Esto obedeció a la intención de demostrar que con una computadora personal, un circuito digital y un programa, es posible sustituir con iguales o mejores resultados a los equipos comerciales de despliegue de variables físicas de entre las cuales escogimos una de origen biológico como lo es el electrocardiograma.

De acuerdo con lo anterior, se plantearon los siguientes objetivos particulares de este proyecto:

- Diseñar y elaborar una tarjeta que permita a un computador personal el almacenamiento y procesamiento de señales o variables físicas.
- Diseñar un sistema en software capaz de analizar y graficar en tiempo real, las señales obtenidas mediante el empleo de la tarjeta, y finalmente
- Demostrar el funcionamiento del sistema para el caso específico de una señal cardíaca, proporcionada por un amplificador analógico conectado a un sujeto de experimentación.

Para cumplir con estos objetivos, el trabajo se divide en 7 capítulos en el primero de ellos se estudian los conceptos necesarios para saber que es y como se interpreta un electrocardiograma, para lo cual es indispensable contar además, con los conocimientos básicos acerca de la fisiología del corazón. En el segundo se hace una descripción de aquellos recursos del computador (tarjeta de video, monitor y sí t. de entrada/salida) cuyo funcionamiento es necesario conocer para comprender el tercer capítulo en el cual se describe detalladamente la manera como se diseña cada una de las partes que constituyen a la tarjeta de digitalización de señales que son:

- a) Sección de conversión de datos analógicos a digitales
- b) Sección de conversión de datos digitales a analógicos
- c) Sección de cálculo de frecuencia (frecuencímetro) y
- d) Sección de salida de datos digitales

En el cuarto capítulo, se muestra un análisis realizado para determinar las capacidades y limitaciones del circuito. El quinto explica la forma como se desarrolló el programa encargado de la graficación en tiempo real de señales cardíacas, así como la manera en que este debe ser usado. En el sexto se muestran los resultados de las pruebas de funcionamiento realizadas tanto a la tarjeta digitalizadora como al programa de graficación.

Finalmente, en el capítulo siete se presenta un conjunto de aplicaciones alternativas que pueden ser desarrolladas mediante el empleo del mismo circuito de digitalización de señales.

CAPITULO PRIMERO

1. PRINCIPIOS BASICOS DE ELECTROCARDIOGRAFIA

1.1 FISILOGIA DEL CORAZON

La sangre es el liquido encargado de la transeportación de oxigeno y anhídrido carbonico. El oxigeno se difunde en la sangre y se distribuye a todas las células del cuerpo. Las células a cambio, liberan el anhídrido carbónico que se transporta a los pulmones para ser expulsado. La sangre, se desplace por dos tipos de conductos (venas y arterias) requiriendo para ello de algun tipo especial de bomba.

El corazón humano consta de dos bombas. Una genera fuerza para espular la sangre enriquecida con oxigeno hacia todas las células del cuerpo la otra la recibe una vez que ha recorrido el organismo, y la conduce a los pulmones donde se realiza el intercambio gaseoso. Cada una de las bombas está constituida por dos cavidades elásticas con paredes contráctiles, las cuales impulsan la sangre cada vez que se comprimen. Para evitar que la sangre se licite a avanzar y retroceder cuando el corazón se contrae, existe una serie de válvulas que dejan pasar el fluido en una sola dirección.

El diagrama de la figura 1.1 muestra en un corazón abierto, las partes principales que lo constituyen. A es la

aurícula derecha, B la válvula tricúspide, C el ventrículo derecho, D la válvula pulmonar, E la aurícula izquierda, F la válvula mitral, G el ventrículo izquierdo y H la válvula aórtica.

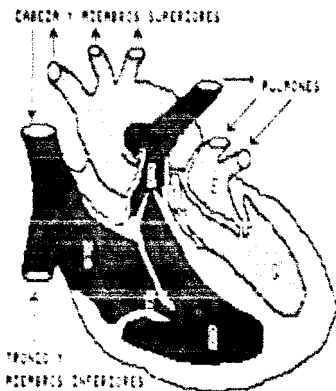


FIGURA 1.1

El lado izquierdo del corazón, la primera bomba y la más potente, recibe la sangre oxigenada de los pulmones y la pasa a la cámara principal llamada ventrículo izquierdo a través de la válvula mitral. Cuando el ventrículo está lleno, el músculo de la pared cardíaca se contrae empujando la sangre por la válvula aórtica, con la fuerza suficiente para enviarla a las partes más lejanas del cuerpo.

Cuando abandona el corazón, la sangre viaja a una velocidad aproximada de 1 m/s, lo que significa que recorre el

cuero en senso de un minuto. En determinadas arterias, la velocidad es mayor, pero conforme se aleja del corazón, disminuye su velocidad casi hasta detenerse. Pese a todo, a los 40 segundos de haber abandonado el corazón la sangre se encuentra de regreso, aunque ahora entra por la parte derecha.

Al igual que la traba izquierda, la derecha tiene una cámara superior de paredes más delgadas que recibe la sangre de un solo istmo; el fluido sale de la primera cámara por la válvula tricúspide, entra en el ventrículo derecho y sale por la válvula semilunar en dirección a los pulmones. El ventrículo derecho no bombea con tanta fuerza como el izquierdo, ya que la resistencia de los pulmones es menor y la presión necesaria para mover la sangre es más baja.

Una vez que la sangre está en los pulmones se intercambia los desechos gaseosos, entra aire nuevo y se carga más oxígeno; la sangre vuelve entonces al lado izquierdo del corazón para iniciar nuevamente su recorrido. Todos estos fenómenos ocurren casi simultáneamente. Mientras el lado izquierdo del corazón bombea sangre hacia los tejidos, el derecho bombea hacia los pulmones la sangre que ha pasado por el lado izquierdo aproximadamente medio minuto antes.

12. ACTIVIDAD ELÉCTRICA DEL CORAZÓN

El corazón posee la capacidad innata de contraerse. Cada contracción va precedida de pulsos de excitación eléctrica. El

corazón puede separarse por completo de su ritmo normal y seguir latiendo durante cierto tiempo.

Para coordinar el bombeo existe un sistema de señales eléctricas repartidas por todo el musculo y que, de hecho, se pueden detectar en todo el organismo. Una serie de electrodos situados en distintas partes del cuerpo revelan que la actividad cardiaca sigue un esquema eléctrico uniforme llamado electrocardiograma (ECG).

El impulso eléctrico en el corazón normal se origina en el nodo sinuauricular o sinusal (SA) situado en la aurícula derecha y se desplaza por ambas aurículas haciendo que estas se contraigan (figura 12). Después, llega a una porción de tejido muscular llamado nodo auriculoventricular (AV) y posteriormente pasa al haz de His y continúa por sus ramas principales derecha e izquierda hasta las fibras o sistema de Purkinje. Sin embargo, por factores que no se han dilucidado, el nodo auriculoventricular retarda el impulso unas centésimas de segundo adicionales antes de permitir que pase al haz de His. Este retardo da tiempo a que las aurículas expulsen su sangre hacia los ventriculos. Después de esto el sistema de Purkinje expande el impulso por los ventriculos, haciendo que todas las porciones se contraigan simultáneamente y tengan un esfuerzo propulsor coordinado.

De no ser por el sistema de Purkinje, el impulso viajaría a menor velocidad por los ventriculos permitiendo que algunas fibras musculares se contrajeran antes que otras, esto

causaría una menor compresión de la sangre y por ello, disminución del poder espulsivo

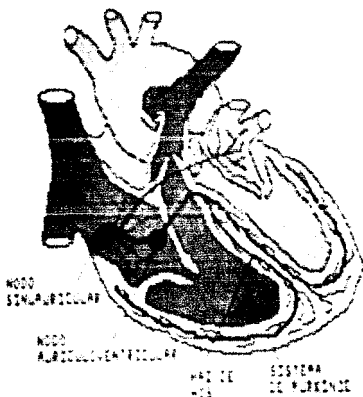


FIGURA 1.2

13 ELECTROCARDIOGRAMA

El electrocardiograma es la representación gráfica de las fuerzas eléctricas que coordinan la actividad cardíaca. cuando un impulso se transmite por las fibras musculares la corriente eléctrica generada se difunde en los líquidos que rodean al corazón y una pequesísima parte llega a la superficie del cuerpo. Colocando electrodos en la piel y conectándolos a

instrumentos adecuados de registro, puede obtenerse un trazo de los impulsos que suscitan los latidos cardiacos (figura 1.3).

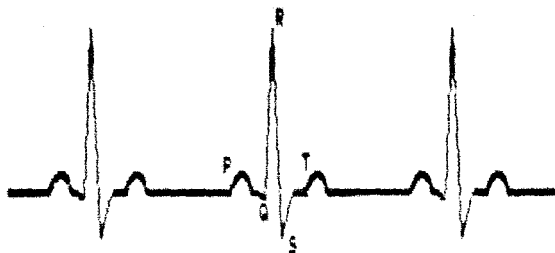
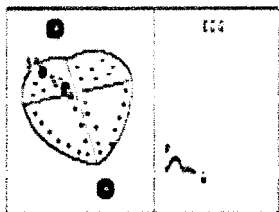


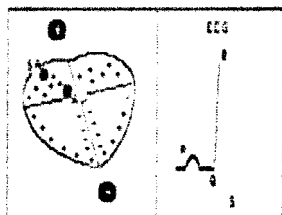
FIGURA 1.3

Desde el punto de vista electrocardiográfico, cada contracción mecánica auricular y ventricular, guarda relación con dos fenómenos eléctricos. El primero es la despolarización o contracción, durante el cual las cargas eléctricas en la superficie de las células musculares se tornan de positivas (+) a negativas (-); el segundo, es llamado repolarización (vuelta del corazón a su estado de reposo) y consiste en la reparación de las cargas positivas en la superficie. La despolarización ventricular es rápida y corresponde al complejo "QRS" (figura 1.3), mientras que la repolarización de los ventriculos es lenta y equivale a la onda "T". El segmento S-T representa el periodo en el cual todas las partes de los ventriculos están despolarizadas. En las aurículas la despolarización se manifiesta por las ondas "P"; también hay repolarización, aunque el registro eléctrico suele ser

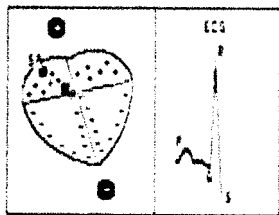
disminuido por el complejo "QRS". En la figura 1 se muestran las porciones del electrocardiograma que se van presentando para cada uno de los eventos de despolarización y repolarización del miocardio para obtener este registro el electrodo positivo debe de estar por arriba y el negativo por debajo del corazón.



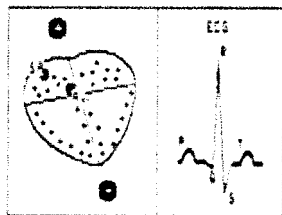
(a)



(b)



(c)



(d)

FIGURA 1 A

El intervalo P-Q, medido desde el comienzo de la onda "P" hasta el comienzo de la "Q", equivale al tiempo que necesita

un impulso que sale del nodo sinusal para llegar a los ventrículos y corresponde al paso de la onda de excitación desde el nodo sinocrisular por las aurículas hasta el nodo aurículoventricular (figura 1-4 A). Como ya se ha mencionado en el nodo aurículoventricular hay normalmente un retardo al que corresponde la mayor parte del tiempo que transcurre en el intervalo P-Q.

El complejo "QRS" equivale a la despolarización de los ventrículos. Durante este tiempo el impulso cardíaco recorre primero el tabique interventricular (figura 1-4 B), e inmediatamente después la pared libre de los dos ventrículos (figura 1-4 C). La onda "T" representa la repolarización de los ventrículos. Una contracción ventricular completa, incluye las fases eléctricas de despolarización y repolarización y se manifiesta en el electrocardiograma por el intervalo Q-T (figura 1-4 D).

14 INTERPRETACION

El electrocardiograma es una arma clínica muy importante en la valoración de la actividad eléctrica cardíaca. Sin embargo, para su interpretación es importante percatarse de que existe una serie de factores extrínsecos que no guardan relación con el corazón en sí y que pueden modificar el trazado final. Algunos de

ellos son los siguientes: resistencia de la piel, grosor de la pared torácica, distancia entre la pared del tórax y el corazón, posición del corazón dentro de la cavidad torácica, interferencia eléctrica, problemas relacionados con el método adecuado para tomar los electrocardiogramas y, finalmente, no apropiado del electrocardiografo. También modifica el electrocardiograma estos factores: fatiga, infecciones, embolias pulmonares, dolor, temor y ejercicio.

Cuando diversas enfermedades causan anomalías cardíacas, el electrocardiograma se altera, en la figura 1.5 (a) por ejemplo, ha ocurrido una lesión del eje del ventricular por lo que la parte del electrocardiograma contenido entre las ondas "S" y "T" se encuentra deprimida. Esto es ocasionado por la difusión anormal de la corriente eléctrica y por lo general indica lesión de las fibras musculares de los ventriculos, frecuente cuando se ha padecido de un ataque cardíaco.

La figura 1.5 (b) muestra una curva "QRS" anormal con alargamiento de la onda "S" y disminución de la "R". Esto suele ocurrir por atrofia unilateral del corazón, por lo cual fluye más corriente de un lado que del otro. Este tipo de electrocardiograma es frecuente en los casos de hipertensión arterial.

Por último, la figura 1.5 (c) muestra lo que se observa en un paciente con bloqueos de la corriente eléctrica en el sistema de Purkinje, en este caso el impulso es transmitido por gran

parte de los ventrículos con tanta lentitud que el complejo "QRS" dura mucho tiempo y tiene forma anormal.

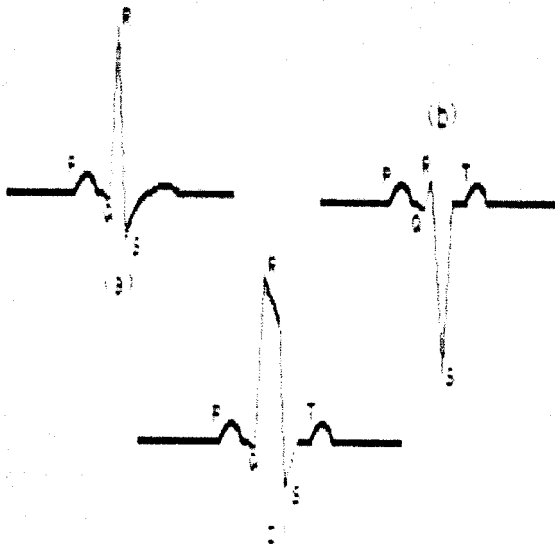


FIGURA 1.5

Las figuras anteriores muestran cómo pueden descubrirse en el trazado electrocardiográfico diversas anomalías de la función cardíaca. Sin embargo, es importante recordar que las fuerzas eléctricas del corazón que se registran son, en el mejor

de los casos, un índice indirecto del estado funcional y anatómico del corazón, y que pueden ser afectadas por factores extracardíacos diversos.

CAPITULO SEGUNDO

2. ELEMENTOS DE LA COMPUTADORA EMPLEADOS EN EL PROYECTO

En este capítulo se dará una breve descripción de los elementos que forman parte de la computadora y cuyo funcionamiento es indispensable conocer para comprender la manera como fueron diseñadas tanto el hardware como el software necesarios para lograr la graficación de señales en tiempo real.

2.1. HARDWARE DE SOPORTE PARA LA REALIZACIÓN DE GRAFICAS

A diferencia de otras formas de programación, la elaboración de gráficas requiere de un conocimiento general de las capacidades del hardware. Por tal motivo, es necesario comprender el potencial y las limitaciones del computador para lograr una correcta visualización de imágenes.

Cada PC cuenta con una tarjeta de video que contiene una memoria de despliegue o de video, que se encuentra físicamente separada de la memoria principal pero que, sin embargo, forma parte del espacio de direcciones de la CPU. El tamaño de esta memoria, así como su división en páginas determina la manera en que debe llevarse a cabo la programación del sistema para lograr una adecuada elaboración de gráficas. Las tarjetas más simples cuentan con 4KB mientras que los modelos más sofisticados llegan a tener hasta 256KB.

La unidad que contiene el tubo de rayos catódicos (CRT) o monitor, es alimentada con la información almacenada en la memoria de despliegue mediante el controlador de video 6845. Este circuito es el encargado de tomar un bloque de memoria llamado "página visual" y trasladarlo al monitor como un conjunto de caracteres o "símbolos" gráficos.

Cuando se dispone de más de una página, es posible la carga independiente de información en otro bloque de memoria llamado "página activa" la cual siempre está preparada para ser desplegada casi instantáneamente cuando se solicita. Los efectos de animación se realizan mediante el intercambio constante de páginas activas a visualizar.

La transferencia de información al monitor se lleva a cabo mediante el proceso de barrido llamado "Raster Scan" mostrado en la figura 2.1. En esta técnica, los disparos emitidos por el cátodo son continuamente enviados de izquierda a derecha y de arriba a abajo a la pantalla de video. Cambiando de propósito a apagado el haz de electrones, el fósforo fotosensitivo del tubo solo brillará en los sitios donde se requiera. Antes de que los puntos excitados se apaguen, el cátodo volverá a disparar y puesto que este proceso (llamado ciclo de refresco) se repite 60

¹ Un pixel es la mínima unidad visual que puede ser controlada en la pantalla de video.

veces por segundo, el monitor dará la impresión de desplegar una imagen fija.

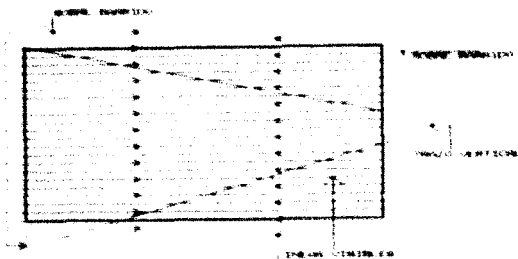


FIGURA 2.1

En la figura anterior, los trazos continuos representan la parte visible del monitor mientras que las líneas punteadas muestran el sobre-barrido que realiza el cátodo en la pantalla.

La memoria de despliegue, se encuentra conectada tanto con la CPU como con el circuito controlador de video. Si ambos dispositivos intentan un acceso simultáneo, entonces se producirá un efecto de distorsión en la imagen conocida como snow (nieve).

Entre cada ciclo de refresco, el haz de electrones es apagado y dirigido desde la parte inferior derecha del monitor hacia la esquina superior izquierda. Este "retroceso vertical" da la oportunidad para que el usuario actualice la memoria de despliegue sin causar distorsión. Puesto que la tarjeta (o adaptador) de video así como el monitor son los dos factores que determinan la calidad de graficación de una PC, se dará un breve

descripción de algunos tipos y características que presentan estos elementos

2.11 TARJETAS DE VIDEO

Este circuito es el que determina la máxima capacidad gráfica que posee una microcomputadora. De manera general, en las PCs se utilizan cuatro adaptadores de video diferentes, estos son: VGA, EGA, MCGA y CGA. Con ellos es posible generar una amplia gama de modos gráficos que van desde el empleo de 2 a 256 colores y una resolución que puede variar de 320x200 a 640x480 pixels.

Tarjeta CGA. El adaptador CGA (Color Graphics Adapter) es, sin duda, el más comúnmente empleado en las computadoras personales. Su resolución puede ser de 320x200 o de 640x200 con una capacidad para generar 4 o 2 colores respectivamente. Cada uno de los pixels está constituido por un punto rojo, uno verde y otro azul; la combinación de estos tres tonos genera el color aparente desplegado en pantalla.

Tarjeta MCGA. El adaptador MCGA (Multicolor Graphics Adapter) es el más utilizado en computadoras basadas en el procesador 80286, como los modelos 30 y 25 de la serie de Sistemas Personales 2 de IBM. Además de presentar los modos de la CGA, esta tarjeta cuenta con la capacidad de desplegar 640x480 ó 320x200 pixels, haciendo uso de 2 y 256 colores respectivamente.

Tarjeta EGA: Al igual que la MCGA, la tarjeta EGA (Enhanced Graphics Adapter) puede generar los mismos modos gráficos presentes en la CGA, además de los siguientes: 320x200/16, 640x200/16, 640x350/16 y 640x350/2 (el último número de esta notación indica la cantidad de colores que pueden ser desplegados).

Tarjeta VGA: El adaptador VGA (Video Graphics Array) es utilizado en computadores que emplean los procesadores 80286 ó 80386, como en los modelos 50, 60 y 80 de la serie de Sistemas 2 de IBM. Esta tarjeta puede generar todos los modos presentes en CGA, MCGA y EGA, así como uno adicional: el 640x480 con capacidad para desplegar 16 colores.

2.12 FUNCIONAMIENTO DE LAS TARJETAS DE VIDEO

La manera en que se encuentra organizada la memoria, así como la forma en que su contenido es enviado al monitor, son los dos factores que determinan el funcionamiento de las tarjetas de video. El adaptador CGA, por ejemplo, emplea una memoria de despliegue "multibit por pixel" (spp), mientras que en VGA y EGA se utiliza una estrategia denominada "multipiano por pixel" (ppp).

Memoria de despliegue multipiano por pixel

En esta técnica la memoria de despliegue del adaptador se encuentra dividida en cuatro zonas o "pianos" de bits en los

colores se graba una imagen completa en alguno de los colores rojo, verde o azul, dejando el cuarto plano para almacenar el nivel de intensidad. La figura 2.3 muestra esta disposición.

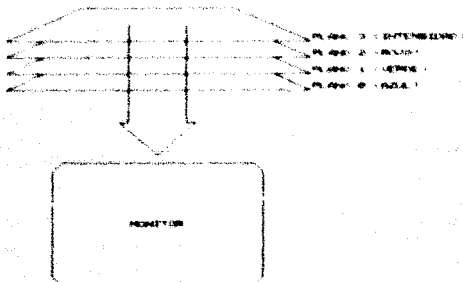


FIGURA 2.3

El circuito controlador de las tarjetas VGA y EGA lee simultáneamente los correspondientes cuatro bits de cada una de las imágenes para determinar cuál de los 16 colores disponibles es el que debe ser impreso en el monitor. Puesto que cada plano puede tener un valor de 0 o de 1 en un punto en particular, la combinación de estos cuatro bits genera una cantidad contenida entre 0000 y 1111, con lo que se tiene un total de 16 números binarios cuyo color correspondiente es determinado por el circuito controlador de video.

Los adaptadores EGA emplean en el proceso de despliegue una técnica llamada "sistema digital", en la cual se "pandeare" gráfica cada rojo, verde y azul como un punto en el monitor con

intensidad alta, mediana o nula. Por otro lado, en las tarjetas VGA con "sistemas analógicos" cada punto de la imagen es enviado al monitor con una brillantez que varía desde totalmente nula hasta completamente alta, pasando por todos los valores intermedios. La mezcla visual de estos niveles es lo que produce el rango de colores desplegado en la pantalla.

Memoria de despliegue multibit por píxel

La tarjeta VGA emplea una estrategia de despliegue llamada "multibit por píxel". Como se muestra en la figura 2.3, este adaptador utiliza un plano único conocido como "mapa de bits" (o bit map).

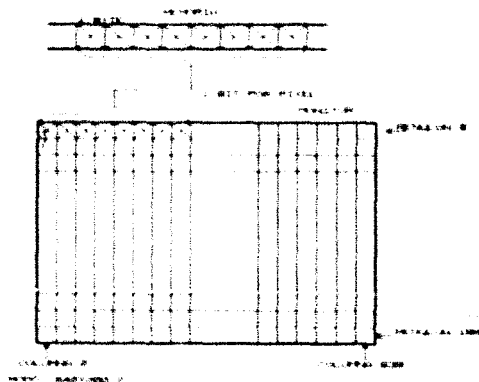


FIGURA 2.3

Para el modo 320x200 cada pixel requiere de dos bits para poder ser graficado ya que con ellos es posible representar cuatro atributos diferentes correspondientes a las combinaciones 00, 01, 10 y 11. De esta manera se tiene que un byte puede controlar cuatro pixels desplegadas en el monitor. Por otro lado, en la representación de un pixel para el modo 640x200/2, es necesario el empleo de solo 1 bit, esto es debido a que con él es posible representar los dos colores requeridos 0 y 1 (equivalentes a pixel encendido y apagado respectivamente).

2.13 MAPA DE MEMORIA PARA DESPLIEGUE

En los computadores personales IBM y compatibles, han sido asignados 256K disponibles con fines de despliegue de información². Como se observa en la figura 2.6, su localización comienza en la dirección A0000h. Es interesante notar que este valor equivale a 640K, el cual es el límite de la memoria RAM destinada al usuario para los modelos XT, AT y compatibles. En los Sistemas Personales 2 de IBM en donde el tamaño de la RAM

² Aunque una tarjeta en particular use o no el total de memoria destinada a despliegue, los 256K siempre están disponibles para este propósito.

es de 16000, los bytes destinados al despliegue de información comienzan en la dirección FFFFFM del mapa de memoria

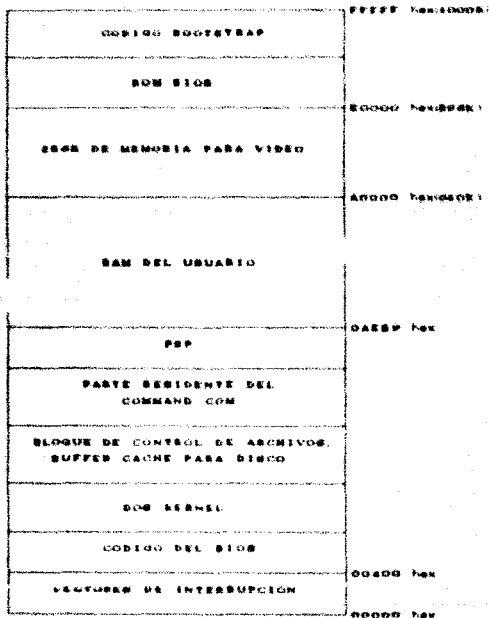


FIGURA 2.6

Los adaptadores tipo CGA cuentan con una memoria de 16K, la cual se encuentra separada en dos bloques de 8000 bytes cada

uno (figura 2.5), el primero de ellos inicia en la dirección B8000h y es empleado para almacenar las líneas PARES de la imagen, el segundo comienza inmediatamente después del anterior (dirección B8004h) y contiene la información gráfica de las líneas NONES. Esta estructura es una característica del hardware, realizada en un esfuerzo por reducir la lentitud en el despliegue de imágenes causada por la baja velocidad de operación del adaptador. Esta metodología no es seguida por la tarjeta VGA ni por la EGA, ya que estas emplean un mapa de memoria secuencial que va de izquierda a derecha y de arriba a abajo (estas localidades corresponden directamente a la línea del monitor).

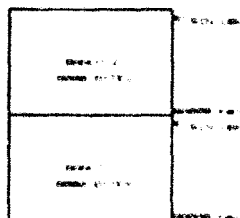


FIGURA 2.5

2.14 DIRECCIONAMIENTO A MEMORIA DE VIDEO

Para almacenar direcciones, la computadora usa dos palabras binarias por una localización de 16 bits, con cada una ellas es posible almacenar números contenidos entre 0 y 65535. Para establecer la dirección de una localidad, el 8088 suma las

memoria de video, correspondientes a los píxeles desplegados en pantalla

Como se ha mencionado, la memoria del adaptador tipo CGA está constituida por dos bancos de 8000 bytes cada uno, el primero de ellos almacena la información de las líneas PARES y su dirección de inicio es la B8000 mientras que el segundo guarda los datos referentes a las líneas NONES⁵ dirección B8000.

Es evidente pues, que antes de graficar un punto es indispensable saber si se encuentra en una línea PAR o NON para de esta forma, determinar el segmento de la dirección. Una vez que se ha obtenido este valor es necesario conocer además su desplazamiento y para ello debe aplicarse la siguiente fórmula:

$$\text{Desplazamiento} = (Y \text{ DIV } 2) * 80 + X \text{ DIV } 8$$

La operación que se encuentra adentro del paréntesis, calcula la línea en la que se encuentra el píxel que se quiere graficar (dentro de su correspondiente banco de memoria). Ahora bien, puesto que cada línea está compuesta por 80 bytes, al multiplicar $(Y \text{ DIV } 2) * 80$ se obtiene el número de bytes necesarios para alcanzar la línea donde se encuentra el píxel. Para

⁵ Recuerdese que los valores de segmento para las direcciones B8000 y B8008 son B8000 y B8008 respectivamente

determinar el byte exacto que contiene el punto a ser graficado es necesario sumarle al valor anterior el resultado obtenido de dividir el contenido de la coordenada X entre 8.

Para ilustrar esto, se determinaran a manera de ejemplo los valores del segmento y desplazamiento del byte que contiene al píxel localizado en las coordenadas X=56 y Y=111 de la pantalla de video. Como se puede observar, el contenido de Y es un número IMPAR por lo que el valor del segmento correspondiente debe ser BA00H. Al realizar la operación $Y \text{ DIV } 8$, se tiene que el píxel se encuentra en la línea 68 cuyo byte de inicio (a partir de la dirección BA000H) es el $60 \cdot 80 + 4800$. Por último, al calcular $60 \text{ DIV } 8$, tenemos que el punto que se desea graficar está en el séptimo byte de la línea 68, es decir, en la localidad 4807 (o 1207H) del segundo banco de memoria. Así, se tiene que los valores de segmento y desplazamiento calculados son el BA00H y el 1207H respectivamente. La manera de modificar la localidad de memoria indicada por estos resultados, se realiza mediante software y varía dependiendo del lenguaje de programación que se utilice. En el capítulo 5, se describe la forma de acceder la memoria de video empleando para ello la instrucción MEM del lenguaje Pascal.

2.15 MONITORES

El monitor es el segundo factor en importancia que determina la calidad de las imágenes que se pueden producir, ya

que la falta de una unidad de despliegue adecuada puede ser una seria limitante para la salida del adaptador gráfico.

Existen cuatro tipos principales de monitores usados en computadores compatibles con IBM. Estos son el de Color Estándar (SCD), el de Color Realizado (ECC), el Analógico y el de Frecuencia Variable. Todos ellos emplean para su funcionamiento la técnica *Master Scan* o barrido del monitor (ver figura 2.1).

Monitor de Color Estándar

El SCD (*Standard Color Display*) cuenta con una resolución máxima de 640×200 *pixels* y capacidad para desplegar simultáneamente los 16 colores mostrados en la tabla 2.1. Este tipo de monitores sólo puede aceptar los modos $320 \times 200/4$ y $640 \times 200/2$ de la tarjeta CGA, así como el $320 \times 200/16$ y $640 \times 200/16$ del adaptador EGA.

0 NEGRO	8 GRIS
1 AZUL	9 AZUL INT.
2 VERDE	10 VERDE INT.
3 CYAN	11 CYAN INT.
4 ROJO	12 ROJO INT.
5 MAGENTA	13 MAGENTA INT.
6 CAFE	14 AMARILLO
7 BLANCO	15 BLANCO INT.

TABLA 2.1

Monitor de Color Realizado

El ECC (*Enhanced Color Display*) tiene una resolución de 640×350 *pixels* y puede desplegar al mismo tiempo 16 colores, los

cuales (a diferencia del SCD), son seleccionados de un total de 64 tonalidades disponibles cuando se emplea el modo 640x350/16. Este monitor puede aceptar todas las configuraciones presentes en los adaptadores CGA y EGA.

Monitor Analógico

A diferencia de los monitores anteriores que emplean tecnología digital, los analógicos (como los de la serie de Sistemas Personales 2 de IBM) utilizan continuamente una señal variable capaz de generar más colores y una gran cantidad de subgraduaciones entre estos.

Las unidades de despliegue analógicas aceptan todos los modos gráficos presentes en los adaptadores VGA, sin embargo no pueden ser conectadas a tarjetas CGA ni a EGA. Los 256 colores que puede generar simultáneamente, son seleccionados de un total de 256,000 tonalidades diferentes. Un ejemplo de ello es el presente en la PS/2 modelo 8512 de IBM.

Monitor de Frecuencia Variable o Multiscanning

Este tipo de monitor puede desplegar tanto señales digitales de video como analógicas. El prefijo "multi" se refiere a la habilidad para analizar la salida del circuito adaptador y determinar el modo gráfico que debe ser usado. Las unidades Multiscanning pueden aceptar cualquier tipo de tarjeta, aun las no muy comerciales como la QuadEGA, que cuenta con el modo 752x410 y capacidad para generar 16 colores simultáneamente.

2.2 SLOT DE ENTRADA/SALIDA

Para comunicarse con dispositivos periféricos (como lo es la tarjeta de digitalización y reproducción de señales), el procesador utiliza los slots de entrada/salida. Cada uno de ellos cuenta con un bus de datos de 8 bits, 20 líneas de direcciones, seis niveles de interrupción, líneas de control para lectura y escritura a memoria o puertos, señal de reloj, líneas de control para el manejo de tres canales de direccionamiento directo a memoria (DMA), así como con cuatro niveles de voltaje (5 y 12 Vdc).

La señal llamada $IO \#$ permite la operación con memorias y dispositivos de entrada/salida de acceso lento. Asimismo existe una línea de validación (check line) que reporta condiciones de error al procesador; su activación genera una interrupción no-esperable en el 8088, lo cual es útil por ejemplo cuando una tarjeta de expansión de memoria indica que se ha habido un error de paridad, en cuyo caso el control del programa es transferido a la rutina de servicio correspondiente para que resuelva el conflicto.

La siguiente figura muestra la localización y el orden de los conectores disponibles en el slot de entrada/salida (constituido por 62 terminales en total). Todas las líneas son compatibles con dispositivos de lógica TTL.

CL

El reloj del sistema (Clock) opera a un tercio de la frecuencia de oscilación, teniendo un periodo de 210 nanosegundos (4.77 MHz) a velocidad normal y de 100 nanosegundos (10 MHz) a velocidad turbo.

RESET DRIVE

Esta línea es empleada para restablecer o inicializar el sistema, está sincronizada con el flanco negativo del reloj y se activa con un voltaje alto.

A0 - A19

Los bits de direcciones son empleados para acceder tanto localidades de memoria como dispositivos de entrada/salida. Las 20 terminales permiten el direccionamiento de 1 Megabyte de memoria, siendo A0 el bit menos significativo (LSB) y A19 el más significativo (MSB). Estas líneas emplean lógica positiva y pueden ser generadas por el microprocesador o por un controlador de DMA.

D0 - D7

Estas terminales constituyen el bus de datos del sistema. Permiten el intercambio de información entre el microprocesador, memoria y los dispositivos de entrada/salida. Al igual que el de direcciones, este bus emplea lógica positiva.

ALE

La señal llamada *Address latch enable* es generada por el microprocesador (en modo mínimo), o por el circuito controlador de bus 8288 (en modo máximo). Habilita a los chips 8281 para almacenar y mantener una dirección válida.

I/O CH CK

La línea de validación del canal de entrada/salida (I/O Channel Check) proporciona al microprocesador información sobre errores de paridad generados al tratar de realizar una operación de lectura/escritura a memoria o a algún dispositivo periférico. Un valor de voltaje bajo indica que se ha cometido un error de paridad.

I/O CH RDY

Esta línea (I/O Channel Ready) debe ser activada con un voltaje bajo por una memoria o por un dispositivo de entrada/salida, con el fin de alargar los ciclos de lectura o escritura. Esto permite a los periféricos de baja velocidad "ajustarse" al canal con un mínimo de dificultad. Las memorias o periféricos lentos deben activar esta señal inmediatamente después de detectar una dirección válida. Esto originará un ciclo de lectura o escritura lo suficientemente grande (entre 10 ciclos de reloj mínimo) como para dar tiempo al dispositivo de leer o escribir su información en el bus de datos.

IRQ2-IRQ7

Las terminales de solicitud de interrupción (*Interrupt Request*) son empleadas para indicarle al microprocesador que un dispositivo periférico requiere de su atención. Cada línea tiene asociada una prioridad, siendo IRQ2 la de valor más alto y IRQ7 la de más baja prioridad. Las interrupciones son generadas cuando por alguna de las líneas IRQ se presenta y mantiene un voltaje alto hasta que la interrupción es reconocida por el microprocesador. En ese momento, el flujo del programa es llevado a la rutina de servicio de interrupción correspondiente (*Interrupt service routine*).

IOR

Esta línea (*Input Output Read*) es empleada para indicarle a un dispositivo de entrada/salida que puede dejar su información en el bus de datos para que sea leída por el procesador o por un circuito controlador de acceso directo a memoria. IOR se activa con un voltaje bajo.

IOW

La señal IOW (*Input Output Write*) le indica al dispositivo periférico seleccionado que debe leer la información presente en ese momento en el bus de datos. Al igual que la anterior, esta señal emplea lógica negativa y puede ser activada por el procesador o por un circuito DMA.

MEMR

MEMR (Memory Read) habilita la salida de la memoria seleccionada para que deposite en el bus de datos la información de la localidad indicada en sus entradas de direcciones.

MEMW

MEMW (Memory Write) habilita a la memoria seleccionada para almacenar en la localidad indicada en sus entradas de direcciones, la información que se encuentra en ese momento en el bus de datos. Ambas señales (MEMR y MEMW) pueden ser generadas por el microprocesador o por un controlador DMA y funcionan con lógica negativa.

DRQ0-DRQ3

Estas líneas (DMA Request) son empleadas por dispositivos periféricos para solicitar al microprocesador los buses del sistema así como las líneas de control MEMR, MEMW, IOR e IOW, con el fin de llevar a cabo una operación de acceso directo a memoria. Al igual que en las líneas de interrupción, estas tienen niveles prioritarios de acceso, siendo DRQ0 la de mayor nivel y DRQ3 la de menor prioridad. Una solicitud es generada cuando en alguna de estas líneas se presenta y mantiene un voltaje alto hasta que la correspondiente terminal de reconocimiento (DACK) es activada.

DACK1-DACK3

Estas señales (DMA Acknowledges) son generadas por el microprocesador para indicarle a los dispositivos periféricos que su solicitud de acceso directo a memoria ha sido aceptada por lo que pueden empezar su operación. Esto se realiza gracias a que el procesador pone en alta impedancia los buses del sistema así como las líneas de control que intervienen en el acceso.

AEN

Esta línea (Address Enable) se emplea para indicarle al procesador o a otros dispositivos periféricos que se está llevando a cabo una operación de acceso directo a memoria. La presencia de un voltaje alto en esta línea indica que el DMA ha tomado control del bus de datos, de direcciones, así como de las líneas de control IOK, IOW, MEMR y MEMW.

TC

La línea de terminación de conteo (Terminal Count) genera un pulso (en lógica positiva) cuando una operación de acceso directo a memoria ha sido terminada.

CARD SELECT

Esta señal es generada por alguna de las tarjetas acopladas a los slots de expansión para indicar que ha sido seleccionada por el sistema.

De las terminales anteriores debe tenerse presente el funcionamiento de las líneas de direcciones, de datos, así como las señales de control IOR, IOW y AEN, ya que son estas las que se emplearán en la elaboración de la tarjeta digitalizadora.

CAPITULO TERCERO

3. DISEÑO DE LA TARJETA DIGITALIZADORA DE SEÑALES

En el capítulo uno se revisaron los conceptos básicos referentes a la actividad eléctrica cardíaca así como su representación gráfica (electrocardiogramas) para el diagnóstico de enfermedades. Ahora bien, si se desea que la señal de voltaje procedente del corazón sea desplegada y analizada por una computadora es necesario, convertir la señal analógica a una serie de valores binarios mediante un proceso conocido como digitalización con el fin de que ésta pueda ser fácilmente enviada y registrada en la memoria de la computadora.

Sin embargo, una PC por sí misma, es incapaz de realizar este proceso, para ello es indispensable la conexión (a través de alguno de los slots de expansión) de un circuito encargado de registrar la amplitud que va tomando la señal cardíaca a lo largo del tiempo y convertirla en una serie de datos binarios, por tal motivo fue necesario diseñar una tarjeta que cumpliera con esta función y que además contara con circuitos adicionales que permitieran conocer la frecuencia de la señal de entrada, así como reproducir los datos digitales almacenados en la memoria del computador.

Para su diseño, el circuito fue dividido en secciones o módulos, cada uno de ellos, contribuye con la realización de tareas específicas y en conjunto, constituyen una tarjeta de digitalización y reproducción de señales. La forma en que estas secciones fueron diseñadas se presenta a continuación.

3.1 SECCION DE DECODIFICACION DE DIRECCIONES

El decodificador de direcciones es el circuito encargado de indicar a la tarjeta el tipo de operación que debe realizar así como el momento en que dichas operaciones deben ejecutarse.

Como se mencionó anteriormente, la tarjeta está constituida por secciones que realizan tareas específicas. Cada sección cuenta con al menos un canal de comunicación con la PC, el cual puede ser de entrada o salida.

Dado que son varios los tipos de datos transferidos entre el computador y la tarjeta, es necesario que a cada una de las entradas y salidas del circuito acopladas a la PC, se le asocie una dirección específica dentro del mapa de puertos. De esta manera, con base en el valor presente en el bus de direcciones el circuito "sabrà" si el computador se está o no refiriendo a él. En caso afirmativo, podrá ejecutar exactamente la operación de lectura o escritura de datos que se le demande.

Cada una de las secciones que componen al circuito tiene asociado un canal único de comunicación con la PC, salvo la sección de cálculo de frecuencias que cuenta con 3 entradas y 2 salidas de datos, por lo que requiere de 5 direcciones para operar (la explicación de esto se dará más adelante) por el momento basta con saber que son necesarias 5 direcciones en total para acceder las secciones de la tarjeta. Ahora bien, para evitar conflictos de flujo de información en el computador es necesario

seleccionar un rango de direcciones que no sea ocupado por el sistema operativo o por algún dispositivo periférico.

El rango de valores seleccionado que cumple con este requerimiento es el que va de la dirección 300h a la 307h del mapa de puertos. La siguiente tabla muestra el valor hexadecimal de cada una de estas direcciones así como su equivalente en código binario.

HEX	BINARIO															
300h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
301h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
302h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
303h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
304h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
305h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
306h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
307h	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLA 3.1

Además de identificar las direcciones anteriores, la tarjeta debe reconocer si el microprocesador intenta un acceso a memoria o a puertos de entrada/salida; de no ser así, se producirían verticircuitos en el bus de datos del computador. El llamado a puertos se lleva a cabo cuando alguna de las líneas IOW o IOB es activada por el microprocesador.

Con el fin de que la tarjeta reconozca cada uno de estos valores binarios se adoptó el bus de direcciones del μP de entrada/salida en circuito combinatorial con ocho salidas, cada una de las cuales activa el canal de datos requerido por el

en los que se desea reconocer un "0" (todas excepto el 9 y el 8), se encuentran conectados a las entradas de las compuertas NOR. Por tal motivo, si en todas ellas aparecen ceros, las salidas marcadas con la letra "S" serán iguales a "1".

Asimismo es necesario que cuando se este llevando a cabo un acceso a puertos la señal IORQ sea activada, esto se logra mediante una compuerta NAND con las entradas conectadas a IOW e IOR. De este modo, si alguna de las dos señales se activa (con un cero), a la salida de la compuerta aparecerá un valor de "1".

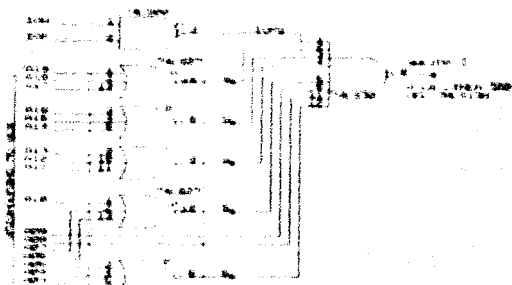


FIGURA 3.2

Las salidas de las compuertas NOR, así como las líneas ADD, AOE e IORQ deben ser conectadas a una compuerta NAND. De esta manera, si el valor presente en el bus de direcciones es el mismo que el que aparece en la tabla 3.3, y el acceso que se lleva a cabo es a puertos, entonces en cada una de

las 8 entradas de esta respuesta aparecerá un "1" y en consecuencia se obtendrá en su salida un valor de "0" que habilitará al circuito 74LS139. Este a su vez seleccionará la salida correspondiente al valor presente en los restantes 3 bits menos significativos (A₁A₂).

Si por el contrario alguno o varios bits no concuerdan con los mostrados en la tabla 2.3 o si el acceso que se lleva a cabo es a memoria entonces a la salida de la respuesta NAND de 8 entradas habrá un "1", por lo que el decodificador quedará deshabilitado y ninguna de sus 8 salidas será seleccionada.

El circuito cuyos valores de salida corresponden a los mostrados en la tabla 2.3 se obtiene al unir la terminal \bar{G} del circuito anterior con la línea de habilitación $\bar{G}2A$ del 74LS139.

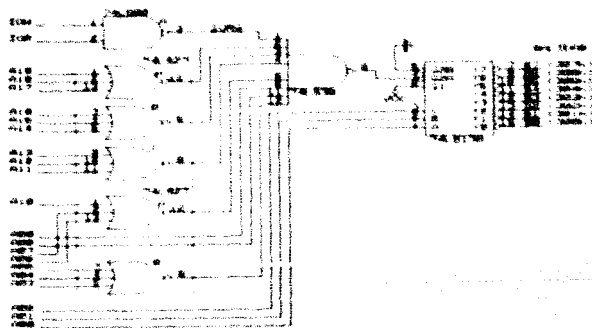


FIGURA 2.4

Por último, para hacer que la tarjeta solo sea accedida por el CPU y no por algún circuito de Acceso Directo a Memoria (DMA), la señal AEN se conecta a la línea de habilitación $\overline{CS2}$ del 74LS138 (figura 3-4). De esta manera el decodificador solo puede ser activado cuando el acceso lo realiza el microprocesador, es decir cuando AEN es igual a cero.

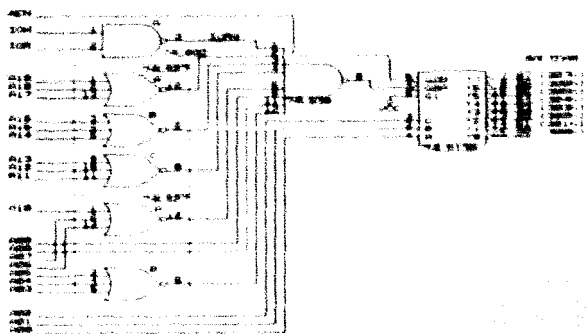


FIGURA 3-4

El diagrama de la figura anterior constituye la sección decodificadora de direcciones de la tarjeta.

3.2 SECCION DE CONVERSION DE DATOS ANALOGICOS A DIGITALES

Los sistemas digitales realizan todas sus operaciones internas en binario o en algun tipo de código binario. Por tal motivo, cualquier información que vaya a introducirse a un computador debe convertirse a forma binaria antes de ser procesada.

La figura 3.5 muestra la manera como una microcomputadora se conecta a un convertidor analógico-digital con el fin de realizar la adquisición de datos.

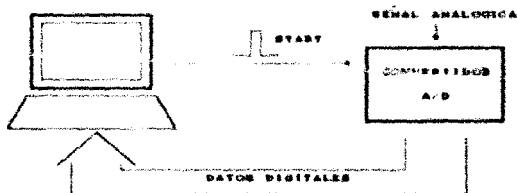


FIGURA 3.5

La señal "start" es generada por la microcomputadora para iniciar una conversión analógico-digital, después de esperar el tiempo necesario para que esta operación sea completada (que por lo general es aproximadamente igual a 100µs), el computador transfiere los datos digitales de la salida del convertidor a su memoria.

3.21 DESCRIPCIÓN DEL CIRCUITO CONVERTIDOR

En el diseño de esta sección se utilizó el circuito ADC 804 que es un convertidor de aproximaciones sucesivas con resolución de 8 bits y tiempo de conversión aproximado de 10µs; el esquema del circuito es el siguiente:

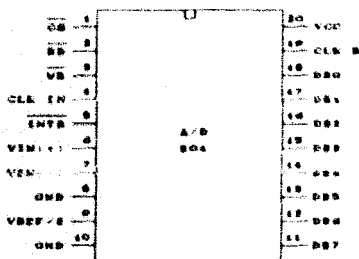


FIGURA 3.4

Con el fin de comprender mejor la forma en que se realizó el diseño de esta sección, se dará una breve descripción de cada una de las señales del convertidor⁴.

⁴ Información adicional acerca del ADC 804 puede consultarse en el apéndice A.

- **CS** Habilita el circuito para realizar una operación de lectura de datos o de inicio de conversión
- **RD** Habilita las salidas del circuito para realizar una lectura de datos
- **WR** Inicia una operación de conversión de datos (este línea equivale a la señal "start" de la figura 3.5)
- **INTR** Indica que una operación de conversión ha sido terminada
- **VIN(+)** y **VIN(-)** Son las entradas diferenciales de voltajes analógicos
- **VREF/2** Es la entrada que permite ajustar el voltaje de referencia (voltaje para el cual se obtiene la máxima cantidad digital). Si $VREF/2 = 2.5$, entonces $VREF = 5$ volts, cuyos valores digitales de salida correspondientes son 01111111 y 11111111 respectivamente
- **CLK N** y **CLK R** Proporcionan al convertidor la señal de reloj con ayuda de un circuito RC externo. La entrada **CLK IN** hace uso del inversor mostrado a continuación.

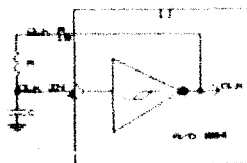


FIGURA 3.7

La frecuencia de salida de este circuito está dada por la fórmula:

$$f_{\text{CLK}} = \frac{1}{4 \cdot 100} \quad \text{para} \quad n = 2 \cdot 1000$$

a 800-807. Son salidas 3 estados por las que se obtiene el valor binario correspondiente al voltaje analógico de entrada.

3.2.2 CONFIGURACION

Para la configuración del ADC 804, se estableció primeramente el reloj del convertidor conectando un circuito RC a las terminales CLK IN y CLK R de la tarjeta como se muestra en la figura 3.7. Puesto que la resistencia tiene un valor de 10K Ω y el capacitor de 150pF (valores recomendados por el fabricante), la frecuencia de la señal obtenida a la salida del inversor es de

$$f_{\text{CLK}} = \frac{1}{4 \cdot 10000} = 2.5 \text{ KHz} \approx 2.5 \text{ KHz}$$

Posteriormente, se fijó el valor de referencia del convertidor a 5V implementando el divisor de voltaje mostrado en la figura 3.8 el cual suministra 2.5V a la entrada VREF/2. De esta manera el ADC 804 queda habilitado para recibir por la entrada VREF+ señales que varíen entre 0 y +5V, ya que como se mencionó anteriormente, el voltaje en la línea VREF/2 debe ser

igual a la mitad del valor de referencia (indiferente que V_{REF} indique el máximo voltaje de entrada al convertidor).

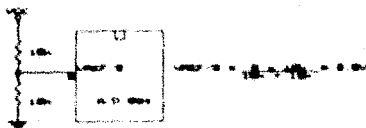


FIGURA 3.8

Sin embargo, se desea que el convertidor también aceptara valores negativos de voltaje. Por tal motivo, se diseñó el circuito divisor mostrado en la figura 3.9, del cual puede obtenerse una señal variable entre 0 y 5V, para entradas con amplitud de 15V.

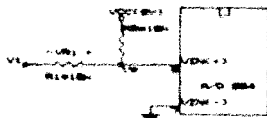


FIGURA 3.9

En la figura anterior, el valor en el punto V_s está dado por la suma del voltaje en la resistencia R_1 (V_{R1}) y el voltaje de entrada (V_i), es decir:

$$V_s = V_{R1} + V_i \quad (11)$$

Por otro lado

$$V_{as} = \left(\frac{5V - V_i}{20k\Omega} \right) 10k\Omega$$

Simplificando:

$$V_{as} = 2.5V - \frac{V_i}{2} \quad (2)$$

Sustituyendo 2 en 1 se tiene que el voltaje a la salida del circuito está dado por la fórmula:

$$V_o = \frac{V_i}{2} + 2.5V$$

lo cual indica que el circuito de la figura 3.9 divide la entrada entre dos y le agrega un offset de 2.5V. El resultado de aplicar esta fórmula a distintos valores de entrada es el siguiente:

ENTRADA (V _i)	SALIDA (V _o)
-5.00	0.00
-3.50	1.25
-2.00	2.50
-0.50	3.75
1.00	5.00

TABLA 3.4

Como puede observarse, el circuito divisor adapta la señal de entrada (variable entre -5 y 5 volts), para que pueda ser

introducida a la terminal VFM-1, la cual como se recordara, sólo puede recibir voltajes contenidos entre 0 y 5V.

3.2.3 CONEXION DEL ADC 804 CON LA COMPUTADORA

La dirección seleccionada para realizar las operaciones de lectura de datos fue la 306h, por tal motivo, la salida número 6 del circuito decodificador se conectó tanto a la entrada RD como a la línea DR del ADC 804 (figura 3.10), con esto las salidas del convertidor (conectadas al bus de datos del CPU) se habilitan permitiendo a la computadora el acceso a la información digital, al momento en que se inicia una nueva conversión.

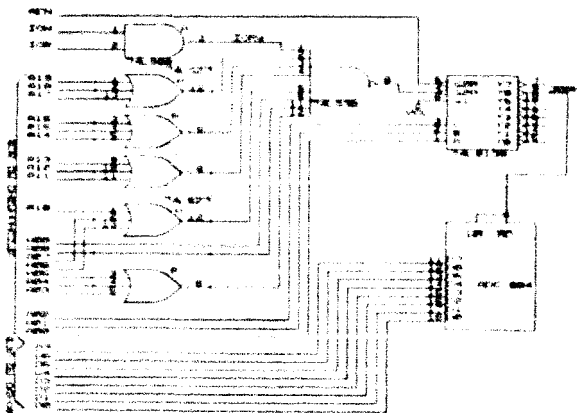


FIGURA 3.10

La figura 3 il muestra el diagrama de tiempos con los pasos necesarios para realizar una lectura de datos

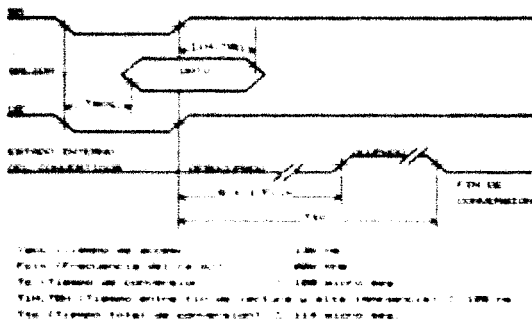


FIGURA 3-15

Como se puede ver la señal \overline{RD} (generada por el decodificador de direcciones cuando el computador lo solicita) habilita al convertidor para que deposite su información en el bus de datos de donde es tomada por la computadora. Asimismo, una nueva conversión es llevada a cabo 8 ciclos de reloj después de que la señal \overline{WR} realice la transición de bajo a alto. Una vez que finaliza el cálculo, el nuevo valor es almacenado en el latch de salida en espera de ser leído. Cabe aclarar que entre lectura y lectura debe transcurrir el tiempo necesario para que el ADC 804 termine la conversión (al menos 114 μ s), y que de no ser así, el circuito iniciaría el cálculo de un nuevo valor dejando inconclusa la conversión anterior.

A pesar de que las señales **RD** y **WR** son activadas simultáneamente, no existe riesgo de que el valor de salida se actualice durante su lectura debido a que el tiempo de acceso a los datos (13ns) es menor al tiempo total de conversión (140ns).

La sección de adquisición de datos diseñada con base en las consideraciones anteriores se muestra a continuación:



FIGURA 5.12

Este circuito cuenta con las siguientes características:

- Tiempo de conversión total de aproximadamente 140ns
- Capacidad para recibir señales variables entre -5 y +5V
- Resolución de 8 bits y
- Acceso a los datos de salida mediante la dirección 306h.

3.3 SECCION DE CALCULO DE FRECUENCIA

Una señal analógica consiste de un voltaje continuo que varía a través del tiempo. Cuando los valores que toma esta señal se van repitiendo, el tiempo necesario para completar un ciclo se denomina periodo T , mientras que la cantidad inversa es decir $1/T$ es llamada frecuencia. Su medición suministra información de gran importancia según la aplicación. En medicina, por ejemplo, el conocimiento del número de latidos por minuto que presenta un paciente puede proporcionar una idea del estado en que se encuentra. Asimismo, en instrumentación es posible calcular los valores de capacitancia, resistencia o inductancia de algún circuito oscilador mediante el conocimiento de la frecuencia que este genera.

3.3.1 CIRCUITO BASICO DE CALCULO DE FRECUENCIA

A pesar de que la determinación de la frecuencia pudo haberse realizado mediante un osciloscopio, se consideró conveniente elaborar un frecuencímetro en hardware ya que de esta forma, los cálculos pueden ser realizados sin la intervención del microprocesador lo que permite un incremento considerable en la velocidad de grabación. La manera como se realizó esta elección se describe asequida.

Para determinar la frecuencia, fue necesario conocer el número de ciclos que presentaba la señal de entrada en un tiempo de un segundo; para esto se empleó el circuito 4040, el cual es un contador de 12 bits elaborado con tecnología CMOS. Asimismo, para almacenar la cuenta final se utilizaron dos latches (74LS27a) de ocho bits cada uno; el circuito también encargado de determinar la frecuencia se muestra a continuación:

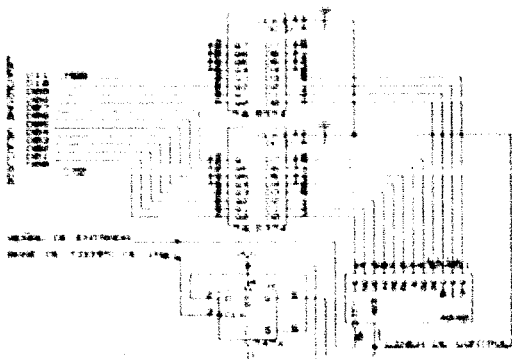


FIGURA B.10

Como se puede observar, la señal de entrada (que debe ser un tren de pulsos cuadrado) es introducida por la línea de reloj (CLK) del 4040; esto permite que la cuenta se incremente cada vez que un ciclo es completado; de esta manera, si la entrada

tiene una frecuencia de 1000Hz, en un segundo el contador contendrá un valor de 1000 binario es decir 00001101000. Este ocurre solo en el caso de que la cuenta comienza desde el comienzo, debe introducirse por la terminal CLK del flip-flop tipo D una tasa de tiempo de 1Hz con el fin de obtener una salida de control con un periodo de 1 segundo (figura 3.14), encargada de habilitar tanto los latches como el contador, para poder llevar a cabo las operaciones de cálculo de frecuencia.

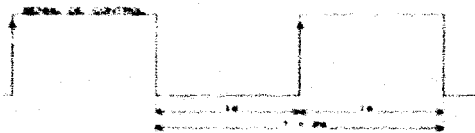


FIGURA 3.14

Al enviar esta señal un valor de "0" por la entrada RST, el 4040 queda habilitado para contar los flancos negativos (cambio de un voltaje alto a uno bajo) presentes en la señal de entrada, y puesto que el "0" permanece durante todo un segundo el valor obtenido en el contador al finalizar este tiempo es el equivalente a la frecuencia introducida por la terminal CLK del 4040. Ahora bien, cuando se lleva a cabo el cambio de bajo a alto (indicado en la figura 3.14 por el signo \uparrow) el valor presente a la salida del 4040 es transferido a los latches donde es almacenado. Posteriormente cuando la señal de control es igual a "1"

el contador se restablece quedando preparado para realizar un nuevo cálculo. Esto se muestra de manera gráfica a continuación.

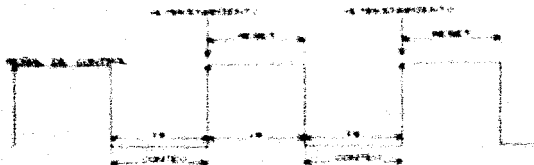


FIGURA 3-19

Cabe señalar que a pesar de que las líneas CLK de los latches y RST del contador están conectadas juntas, no existe riesgo de que el $\overline{\text{RST}}$ se restablezca antes de que los datos sean almacenados, esto es debido a que la retención de información en los latches es inmediata, mientras que el contador requiere de 320ns para restablecer su estado.

Por lo general el circuito básico de cálculo de frecuencia mostrado en la figura 3-13 proporciona resultados correctos sin embargo, cuando la frecuencia de entrada es cercana o inferior a 1 MHz se generan errores de medición como los

* El fabricante asegura que el tiempo de retención de datos (DATA HOLD TIME) es igual a 0ns

mostrados en la figura 3.16, en donde a pesar de que las señales (a), (b) y (c) son diferentes, el valor de frecuencia registrado para las tres entradas es de 1Hz. Asimismo, para la figura (d) ninguno de los flancos negativos se presenta cuando la línea de control es igual a "0" por lo que el valor calculado es de 0 Hz, lo cual es incorrecto.

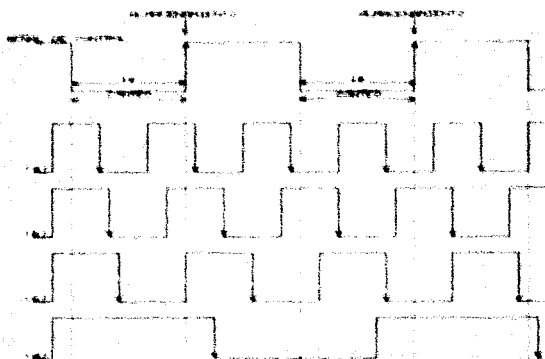


FIGURA 3.16

Estos errores pueden ser corregidos suministrando la frecuencia que se desea calcular a la entrada del flip-flop tipo "D" (para generar la señal de control) y alimentando la terminal

CLK del 4040 con una base de tiempo de 60 Hz, de la manera como se muestra enseguida.

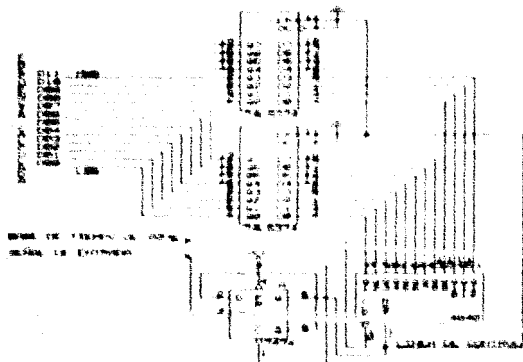


FIGURA 8-17

En esta configuración, el valor contenido en los latches equivale al número de veces que 1/60 de segundo (el inverso de la frecuencia base) cabe en cada uno de los ciclos de la señal de entrada, cuyo periodo puede calcularse de la siguiente manera:

$$T = \left[\text{CONTENIDO DE LOS LATCHES} \right] \left[\frac{1}{60} \right]$$

Puesto que la frecuencia es igual al inverso del periodo, su valor puede obtenerse mediante la fórmula

$$\text{FRECUENCIA DE ENTRADA} = \left[\frac{\text{CONVENIDO DE LOS LATCHES}}{60} \right]^{-1}$$

El ejemplo de la figura 3 18 muestra lo anterior

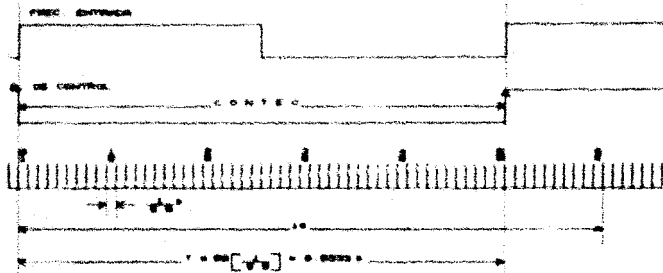


FIGURA 3 18

Como se puede ver, el flip-flop "D" divide la frecuencia de entrada entre 2 con el fin de generar la señal de control, la cual durante el instante en que toma un valor de "0" (que es igual a la duración del periodo de la señal que se quiere medir), habilita al 4040 para que cuente los ciclos de la base de tiempo de 60Hz. De esta manera, si fueran 50 los pulsos contados, entonces el periodo de la señal de entrada sería de:

$$T = \frac{50}{60} = 0.8333 \text{ s}$$

Al obtener la inversa del periodo se tiene que la frecuencia calculada es

$$f = \frac{1}{0.0001} = 10 \text{ kHz}$$

Este valor nunca pudo haberse obtenido mediante el circuito original, sin embargo con esta última configuración es imposible medir frecuencias superiores a 10kHz

Para obtener en todo momento mediciones exactas, es necesario sustituir las entradas como se muestra en la figura 3.17, si se van a realizar mediciones de baja frecuencia e intercambiarlas cuando se desean mediciones de frecuencias altas. Este intercambio de entradas se logra mediante el circuito de la figura 3.19

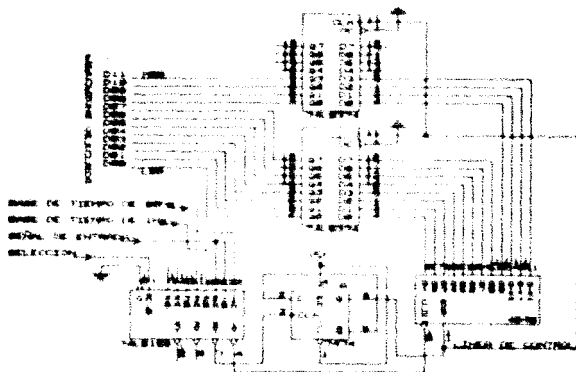


FIGURA 3.19

El circuito integrado 74LS158 está constituido por cuatro multiplexores de 2 a 1 línea. Cada uno de ellos cuenta con una entrada llamada "B", otra denominada "A" y una salida "Y". Cuando la terminal de selección (común a todos los multiplexores) presenta un valor de "0", las señales conectadas a las líneas "A" se devían la frecuencia de la señal de entrada y la base de tiempo de 60Hz, son enviadas por las salidas Y2 y Y1 respectivamente, con lo que el circuito queda habilitado para medir frecuencias bajas. Por otro lado, si la selección adquiere un "1", entonces la señal de entrada será llevada a la terminal Y1 y la base de 10z a Y2, lo que permite realizar mediciones de frecuencias altas. Para el diagrama de la figura anterior, tanto la base de tiempo de 1Hz como la de 60Hz, fueron generadas por dos circuitos integrados LM555 configurados para operar como estables (osciladores).³

Ahora bien, con el fin de que el computador pueda controlar la línea de selección del 74LS158, se agregó otro

³ Información adicional sobre la estructura y configuración del circuito LM555 como monoestable y estable, puede consultarse en el apéndice c.

flip-flop tipo "D" al frecuencímetro conectados de la manera como se muestra en la figura 3.20.

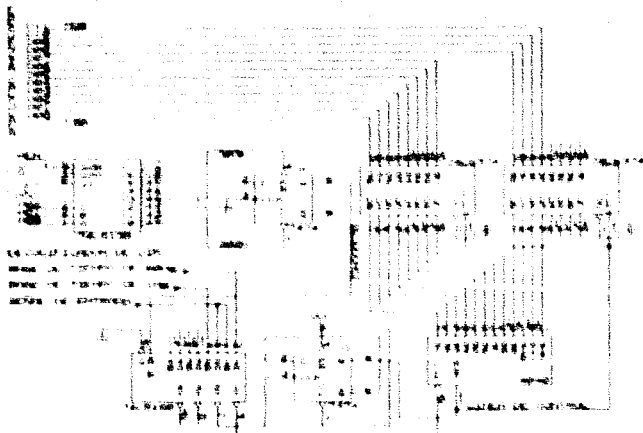


FIGURA 3.20

Como se observa, la salida 2 del circuito decodificador de direcciones se conectó a la línea **CLEAR** y la 3 a la terminal **PRESET** del flip-flop; esto permite enviar un "0" o un "1" a la selección del multiplexor cuando la computadora activa respectivamente las direcciones 302W y 303W.

3.3.2. ADAPTACION DE LA SEÑAL DE ENTRADA

A pesar de que el circuito que se ha diseñado está habilitado para medir tanto frecuencias altas como bajas, presenta aún el problema de que la señal de entrada debe ser un tren de pulsos cuadrado. Sin embargo, por lo regular las variables físicas no poseen esta característica, por tal motivo es necesario convertir cualquier voltaje de entrada en una onda cuadrada con el fin de que su frecuencia pueda ser medida. Para tal efecto se emplea el circuito LM311, el cual es un comparador de voltaje con salida compatible con lógica TTL. La figura 3.21 (a) muestra el modelo simplificado del LM311.

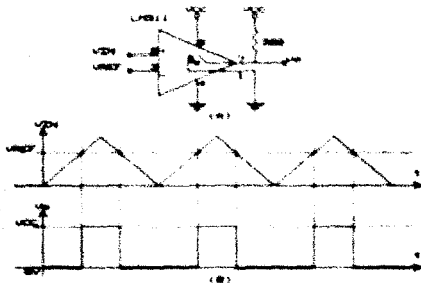


FIGURA 3.21

La salida de este circuito se comporta como un interruptor conectado entre las terminales 7 y 1. Cuando el voltaje en la entrada "+" es mayor que el de la "-" (referencia o umbral),

el interruptor equivalente se abre por lo que la salida V_o es igual a $V_{cc}/5V$ de igual manera cuando la línea "1" es menor que $V_{cc}/5V$ el interruptor se cierra con lo que se obtiene a la salida un voltaje de 0V. De esta forma el en cada uno de los niveles existe un lapso en el que la señal de entrada rebasa el voltaje establecido por la línea "1" entonces se obtendrá una forma de onda similar a la mostrada en la figura 3.11. Su frecuencia puede ser fácilmente leída por el circuito que se ha diseñado.

Ahora bien, si el voltaje de referencia permanece constante entonces puede presentarse el caso de que la señal de entrada sea siempre menor que éste, por lo que a pesar de que exista una frecuencia factible de ser leída, el resultado a la salida de los flashes será en todo momento igual a cero. Para corregir esto, se pensó que la computadora debía controlar el nivel de referencia o umbral del comparador, a fin de que pudiera encontrar un valor de voltaje en el que en ciertos intervalos de tiempo la terminal "1" sea más positiva que la "0" y en otros sea más negativa. Para tal efecto se incorporó al frecuencímetro un DAC810 de la manera como se muestra en la figura 3.22. El circuito se implementó de manera sencilla al emplearse en la sección de conversión de datos digitales a analógicos cuyo diseño se explicará más adelante con la única diferencia de que en éste las información binaria es enviada por la dirección \overline{LD} y no por la \overline{CS} . Las cuatro puertas NAND con histéresis se agregaron al diseño con el fin de eliminar el ruido proveniente de la entrada

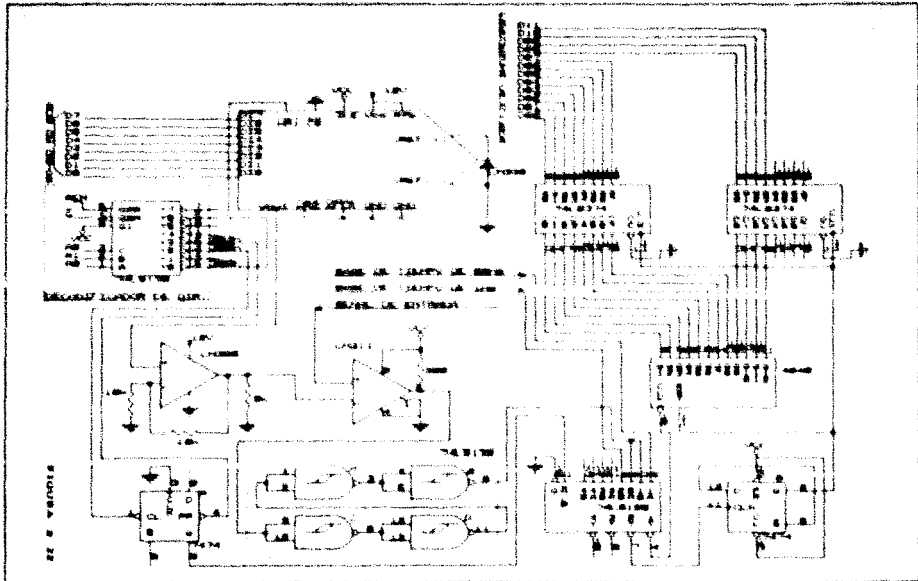


FIGURE 2-22

3.3.3 INDICADOR DE SOBREFLUJO

Como se mencionó anteriormente, el circuito 4040 empleado en la elaboración del frecuencímetro es de 12 Bits, esto significa que solo puede contar del 0 al 4095 ($2^{12}-1$). Si la frecuencia sobrepasa esta cantidad entonces se producirán errores de medición debido a que cuando el 4040 alcanza el valor final, al siguiente ciclo de reloj vuelve a comenzar la cuenta desde cero. De esta manera si la frecuencia de entrada fuera de 4100 Hz el valor obtenido sería de 4 Hz.

Para evitar esto se diseñó el circuito mostrado en la figura 3.23, el cual está constituido por dos flip-flops tipo JK. El primero de ellos (llamado A1) está diseñado para que su comportamiento sea como el de un tipo "T" con un valor de "1" a la entrada, de esta forma para cada pulso introducido por su línea CLK, la salida adquiere un valor contrario al de su estado anterior.

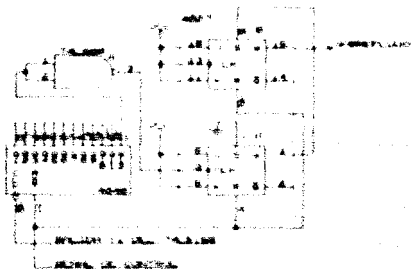


FIGURA 3.23

Quando el contador pase del valor 4095 al 0, el bit número 12 del 4040 realiza una transición de alto a bajo, por lo que a la salida de la compuerta NAND (configurada como inversor) se presenta un flanco positivo. En ese momento el primer flip-flop I cambia de estado y se obtiene a su salida un "1" que activará a su vez la línea Set del segundo flip-flop, con esto, la terminal de SOBREFLUJO (SF) adoptara un valor también igual a "1", indicando que la cuenta ha sido rebasada. Si el 4040 vuelve a pasar el 4095, el flip-flop I cambiará de estado, sin embargo la salida del segundo permanecerá constante mientras su línea de Reset no sea activada lo cual sucede cuando se inicia una nueva medición de frecuencia.

3.3.4 CONEXIÓN CON EL COMPUTADOR

Puesto que son 13 los bits de salida del frecuencimetro (doce de datos y el indicador de SF), la transferencia debe realizarse en dos ciclos de lectura, ya que la computadora sólo puede recibir 8 bits a la vez. La figura 3.14 muestra las conexiones realizadas para lograr el envío de información.

La señal de SOBREFLUJO fue conectada a la entrada D₁₂ de uno de los Interc. junto con los bits más significativos del valor

de la frecuencia, con el fin de que pudiera ser enviada y analizada por la computadora

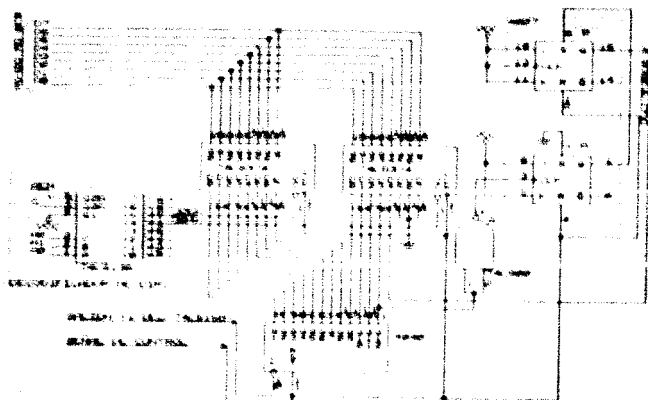


FIGURA 3.10

Las salidas 4 y 5 del circuito decodificador se unieron a las terminales Output Control de los *latches*; de esta manera los 6 bits menos significativos pueden ser accedidos mediante la dirección 305W, mientras que con la 304W se leen los 5 bits restantes. Los valores obtenidos presentan el siguiente formato

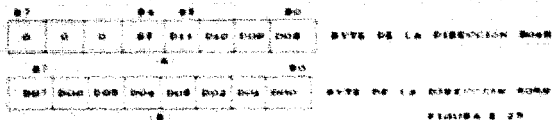


FIGURA 1 20

Si el indicador de SOBREFLUJO presenta un "1" entonces la cantidad obtenida es inválida y a tener la señal de entrada una frecuencia superior a 4.95 Hz. Sin embargo, si no se presenta un error en la lectura, el valor de frecuencia puede ser calculado moviendo un carrileto de 8 posiciones a la izquierda del primer bit, para poderle sumar el resultado el byte (b₇) de la muestra como se muestra a continuación:

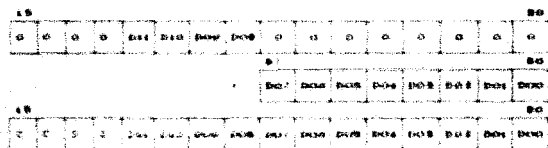


FIGURA 1 21

El valor en decimal obtenido en esta operación es igual a la frecuencia en Hz de la fuente de onda de entrada.

Por último, para eliminar un poco de ruido inducido a la tarjeta por la misma señal de entrada, se usó con optoacopladores el circuito básico, el contador y los dos lámparas del resto de los componentes. Asimismo, se agregaron compuertas NAND con histéresis a las entradas y salidas de los acopladores. La sección de cálculo de frecuencia terminada, se presenta en la figura 1 22.

3.4 SECCION DE CONVERSION DE DATOS DIGITALES A ANALOGICOS

La función de un convertidor de datos digitales a analógicos (DAC) es la de tomar un valor representado en algún tipo de código binario y convertirlo a un voltaje o corriente proporcional al valor digital de entrada.

La figura 3.28 muestra la manera en que un DAC debe ser conectado a la computadora con el fin de obtener una forma de onda analógica a su salida.

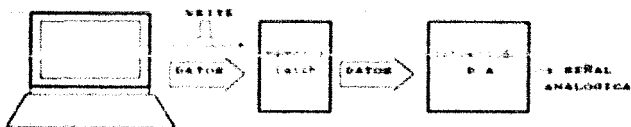


FIGURA 3.28

La señal **WRITE** activa el **latch** de entrada para que éste almacene la información binaria que le envía la computadora a través del bus de datos. Posteriormente el convertidor toma la información de la memoria y la convierte a un valor analógico equivalente.

El **latch** mostrado en la figura anterior sirve para guardar sólo la información que se quiere convertir. Sin él, el DAC transformaría cualquier valor presente en el bus de datos.

Cabe señalar que en este proyecto, el convertidor no será empleado para la graficación en tiempo real de señales cardíacas, sin embargo, permite el desarrollo una gran variedad de aplicaciones como las descritas en el capítulo 7.

3.41 DESCRIPCIÓN DEL CIRCUITO CONVERTIDOR

En la elaboración de esta sección se empleó el circuito DAC 830 configurado para obtener a su salida una señal analógica de voltaje, su esquema se muestra enseguida⁴

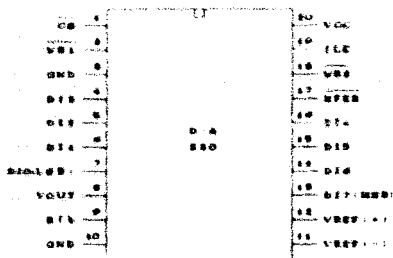


FIGURA 3.30

Este convertidor tiene una resolución de 8 bits (256 niveles de voltaje) y tiempo de conversión de los μ s, además cuenta con dos latidos de entrada que permiten la actualización simultánea de cualquier número de DACs

⁴ Debido a que este convertidor puede ser configurado para obtener una salida de voltaje o corriente, el nombre y función de algunas de las señales varían dependiendo del tipo de salida que se requiera. Información adicional de este circuito puede consultarse en el apéndice B.

LA SALUD DE LA BIBLIOTECA

La descripción de cada una de las señales de este circuito es la siguiente:

- **WR1**: Almacena los bits que provienen del bus de datos en la memoria /ata de entrada.
- **CS** y **LE**: Habilitan la línea de escritura **WR1**.
- **WR2**: Almacena la entrada en la segunda memoria /ata iniciándose con esta una operación de conversión de datos.
- **XFER**: habilita la línea **WR2**.
- **DIO-D7**: Son las líneas de entrada para el valor digital.
- **Vout**: Es la salida del voltaje analógico correspondiente al valor digital de entrada.
- **Rfb**: Es una resistencia interna con valor de 15k Ω empleada en la configuración de salida de voltaje para conectarse en serie con un diodo zener, el cual suministra el valor de referencia **VREF(-)** constante e igual al voltaje zener.
- **VREF(+)**, **VREF(-)**: Indican los voltajes de salida máximo y mínimo correspondientes a los valores digitales de entrada 11111111 y 00000000 respectivamente.

3.4.2 CONFIGURACION

Para el diseño de esta sección se estableció primeramente que la salida de voltaje variara entre 0 y +5V, por tal motivo la entrada **VREF(-)** se conectó a tierra y a la terminal **VREF(+)** se le suministraron 2.5V mediante el diodo zener de la

Figura 3.30. Con esto, el DAC 814 qued6 habilitado para obtener salidas variables entre 0 y 2.5V, posteriormente, la se~al obtenida por la linea VOUT se aplic6 a un amplificador operacional con una ganancia de dos, obteniendose de esta forma el voltaje variable entre 0 y 5V deseado.

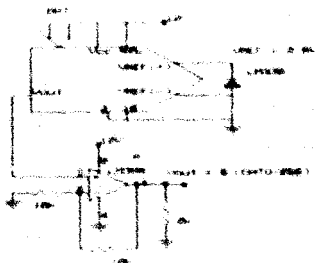


FIGURA 3.30

La salida generada por el circuito de la figura anterior pudo haberse obtenido suministrando un voltaje de referencia de 5V, evitando el uso del amplificador operacional, sin embargo el fabricante se~ala que para conseguir un 0.1% de error de linealidad⁵ es necesario que VREF sea menor a 5V y que el voltaje de polarizaci6n sea al menos 9 volts mayor que el valor de referencia. Para cumplir con esta ~ltima condici6n, el circuito

⁵ El error de linealidad es la desviaci6n m6xima entre la curva formada por los voltajes reales de salida y la linea recta con los valores comprendidos entre 0V y VREF

El diagrama de la sección completa diseñada con base en las especificaciones anteriores es el siguiente.



FIGURA 5.02

Las características de este circuito son

- Resolución de 8 bits (256 niveles de voltaje a la salida)
- tiempo de conversión de 1 μ s
- Voltaje de salida variable entre 0 y 5 volts y.
- Escritura de los datos binarios que se desean convertir a través de la dirección 201K

3.5 SECCION DE SALIDA DE DATOS DIGITALES

Esta sección pone a disposición del usuario un byte de información digital. Con él es posible adaptar un multiplexor analógico permitiendo la digitalización de varias señales. Asimismo, puede ser emplead en sistemas de control de dos posiciones (ON-OFF) o para adaptar un segundo convertidor de datos digitales a analógico.

3.5.1 DISEÑO:

Para el diseño de esta sección se empleó un latch 74LS373 así como una compuerta NAND configurada como inversor, con sus entradas conectadas a la terminal número 7 del circuito decodificador de direcciones y su salida unida a la línea **G** del latch (figura 3.4a).

Si la computadora realiza un ciclo de escritura a la dirección **30th** entonces a la salida de la compuerta NAND se obtendrá un "1" que habilitará al circuito 74LS373 con lo que la información presente en ese momento en el bus de datos será transferida a la salida de latch donde se mantendrá mientras no se lleve a cabo una nueva operación de acceso.

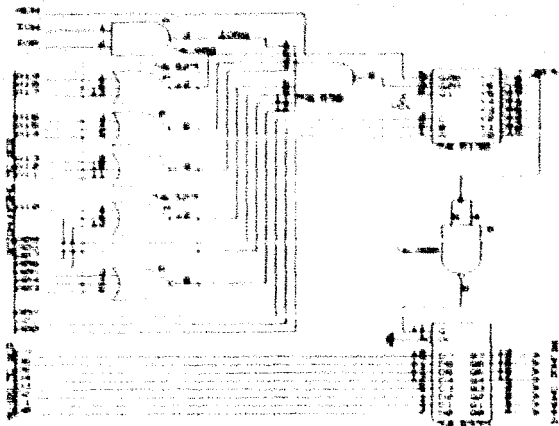


FIGURA 3-34

Este circuito constituye la sección de salida digital de la tarjeta.

3.8 INTEGRACION DE SECCIONES

Para integrar las secciones en una tarjeta única, se conectaron las líneas de escritura y lectura de cada una de ellas, a sus correspondientes terminales de salida del circuito decodificador de direcciones, así como todos los pines de datos se unieron entre sí para formar un bus único. Ahora bien, puesto

que a través de él deba fluir la información desde y hacia la computadora. Fue necesario agregar un TRANSECTOR (74LS245) con el fin de controlar la dirección de los datos. La figura 3.35 presenta un diagrama a bloques de la tarjeta que muestra la forma en que se conectarán las señales de control, los buses y las salidas del decodificador de direcciones.

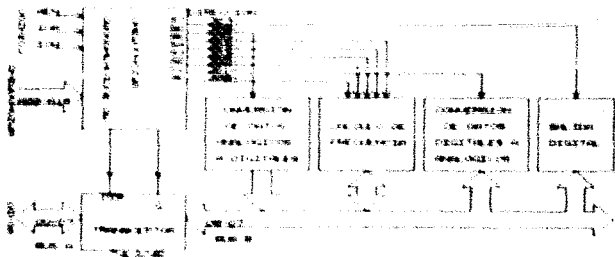


FIGURA 3.35

El 74LS245 permite la transmisión de datos del bus "A" al "B" o del "B" al "A" cuando en la línea \overline{DIR} aparece un valor de "1" o "0" respectivamente. Si la computadora accede alguna de las direcciones de la tarjeta, la salida de la compuerta NAND de 8 entradas (de la sección decodificadora de direcciones) habilita el transceptor mediante su línea \overline{G} para realizar el envío de información.

Cuando la señal \overline{IOW} es activada, en la terminal \overline{DIR} aparece un valor de "1" con lo que los datos son transferidos de

la computadora a la tarjeta. De igual manera, la señal **KOR** debe presentar un valor de "00" para enviar información a la PC.

La siguiente tabla muestra a manera de resumen, las tareas que se llevan a cabo para cada una de las direcciones establecidas para acceder a la tarjeta.

BISS:	TAREA
800H	FIJA EL NIVEL DE REFERENCIA UMbral DE LA REGION DE CALCULO DE FRECUENCIA
801H	CONVIERTE LA INFORMACION DIGITAL DEL BUS DE DATOS EN VALORES DE VOLTAGE ANALOGICOS
802H	HABILITA A LA TARJETA PARA REALIZAR MEDICIONES DE FRECUENCIAS BAJAS
803H	PERMITE LA MEDICION DE FRECUENCIAS ALTAS
804H	LEE LOS 4 BITS MAS SIGNIFICATIVOS DEL VALOR DE FRECUENCIA ASI COMO LA SEÑAL DE SOBREPUNZO
805H	LEE LOS BITS MENOS SIGNIFICATIVOS DEL VALOR DE FREQ.
806H	CONVIERTE LOS VOLTAGES ANALOGICOS DE ENTRADA A VALORES BINARIOS DE 8 BITS
807H	ESCRIBE UN BYTE EN LA MEMORIA LOCAL DE LA REGION DE SALIDA DE DATOS DIGITALES

3.7 INTERFASE CON EL USUARIO

Además de contar con 8 canales de comunicación con la PC, la tarjeta debe tener una línea de entrada a través de la cual se aplique la señal analógica que se desea digitalizar. De igual manera, son necesarias 8 salidas de datos digitales, así

como una terminal por la que se pueda obtener la forma de onda analógica generada por el computador.

Estas terminales se pusieron a disposición del usuario en una interfase (cable) la sustrata en la figura 3 se conectada a la tarjeta a través de un cable plano de 16 hilos.

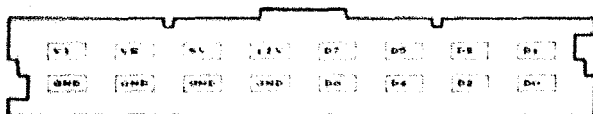


FIGURA 3 5d

La descripción de cada una de estas líneas es la siguiente:

- VI Es la entrada de la señal analógica que se desea digitalizar, su amplitud debe estar contenida entre -1 y +5V.
- VS Es la salida de la forma de onda analógica generada por la computadora a través del convertidor D/A, esta señal puede variar entre 0 y 5V.
- 5V: Suministra un voltaje de corriente directa de 5V.
- 12V: Proporciona un voltaje de corriente directa de 12V, esta terminal y la anterior pueden ser empleadas para polarizar circuitos externos.
- D0-D7: son las 8 líneas de salida de los datos digitales.
- GND: Son las terminales de tierra.

Una vez terminado el diseño de esta tarjeta, realizaremos un análisis tendiente a conocer sus capacidades y limitaciones para la digitalización de señales

CAPITULO CUARTO

4. ANALISIS DE LA CAPACIDAD DE DIGITALIZACION DE LA TARJETA

Aunque algunas señales son digitales por naturaleza (como datos de teletipo, salidas de computadores, señales pulsantes de radar, sonar, etc.) muchas otras son analógicas o funciones suaves del tiempo, como ya se ha mencionado. Si se desea que estas señales sean aceptadas por un computador es necesario convertirlas a alguna forma de código binario mediante un proceso conocido como digitalización, el cual consiste en tomar una serie de muestras de voltaje de la señal analógica que se desea convertir, para posteriormente asociarle a cada una de ellas un código que por lo general es de 8 bits y que corresponde a la amplitud que tenía la señal en el momento de la muestra. De esta manera el resultado obtenido es una serie de valores binarios que describen de manera exacta, tanto la forma como la frecuencia de la señal analógica y que además pueden ser fácilmente almacenados, analizados e incluso reproducidos por una computadora.

La digitalización presenta sin embargo el problema de que a partir de un valor de frecuencia, el proceso de muestreo destruye parte de la información de la señal de entrada. Esto significa que si la forma de onda que se desea digitalizar contiene componentes de frecuencia superiores a este valor límite,

resultará imposible realizar a partir de los resultados obtenidos el análisis y la reproducción exacta de la señal analógica original.

4.1 TEOREMA DEL MUESTREO DE NYQUIST

Este valor de frecuencia "límite" del que heco hablado, está relacionado con el número de muestras tomadas en un segundo (velocidad de muestreo) mediante el Teorema de Nyquist, el cual establece que si la señal $x(t)$ tiene componentes de frecuencia no mayores a Δ Hz, el proceso de muestreo no destruye ningún contenido de información de la misma siempre y cuando la velocidad de muestreo f_s sea mayor o igual al valor de 2Δ Hz⁶, es decir:

$$f_s \geq 2\Delta$$

Para mostrar el resultado anterior, considérese una señal $x(t)$ definida para todo tiempo como se indica en la figura 4.1.1a), si $x(t)$ es estrictamente limitada en banda, con componentes de frecuencia no mayores a Δ Hz, entonces su transformada de Fourier $X(f)$ que representa a dicha señal en el

⁶ La razón de muestreo de 2Δ muestras por segundo, para un ancho de banda de Δ Hz, es llamado a menudo índice de Nyquist.

dominio de la frecuencia, tiene la propiedad de ser igual a cero para $|f| > B$ (figura 4.1 (b)).

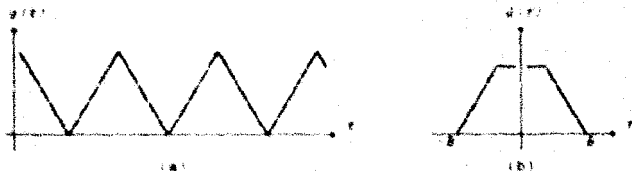


FIGURA 4.1

Ahora bien, supongamos que $g(t)$ se muestra a una velocidad de f_s veces por segundo, como consecuencia de este proceso, se obtiene una señal $g_s(t)$ formada por una sucesión infinita de números espaciados cada $T_s = (1/f_s)$ segundos (figura 4.2).

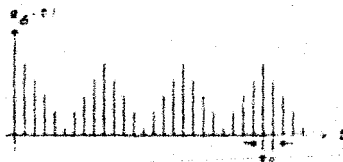


FIGURA 4.2

La figura 4.3, muestra la gráfica de la señal $g_s(f)$ en el dominio de la frecuencia, como se puede observar $g_s(f)$

represente un espectro continuo con un periodo igual a la velocidad de muestreo f_s , en donde la forma de cada uno de ellos es igual a $2B$.



FIGURA 3

Si en la gráfica anterior hacemos $f_s = 2B$ entonces el espectro quedaría como el mostrado a continuación.



FIGURA 4

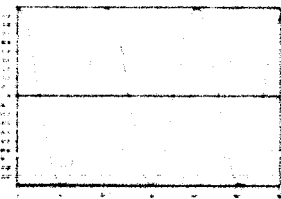
Como se puede ver, ninguna de las repeticiones del espectro de $2B$ se superpone con otra por lo que la forma de $2B$ se preserva pudiéndose recuperar nuevamente y si haciendo pasar la señal muestreada a través de un filtro ideal hayo para con un ancho de banda de B Hz. Esto se presenta de manera gráfica

en la figura 4.5 en donde el cuadro que encierra al periodo central de $\hat{u}_0(f)$ representa la respuesta de amplitud del filtro paso bajas ideal.

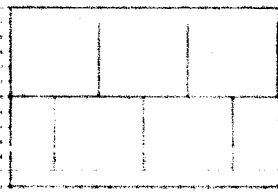


FIGURA 4.5

Para comprobar lo anterior, se simuló una forma de onda senoidal con frecuencia igual a 2 Hz (figura 4.6(a)). Al muestrear la señal a una velocidad de 20 Hz (equivalente al índice de Nyquist) se obtuvo la sucesión de números mostrada en la figura 4.6(b).



(a)



(b)

FIGURA 4.6

Ahora bien, considerando que la respuesta de un filtro ideal paso bajas a una entrada de tipo impulso es una función sinc

de fórmula $\frac{2\pi \cdot 2\pi \cdot 2\pi}{2\pi \cdot 2\pi}$, se generaron en computadora las respuestas a cada uno de los impulsos mostrados en la figura anterior obteniéndose los siguientes resultados

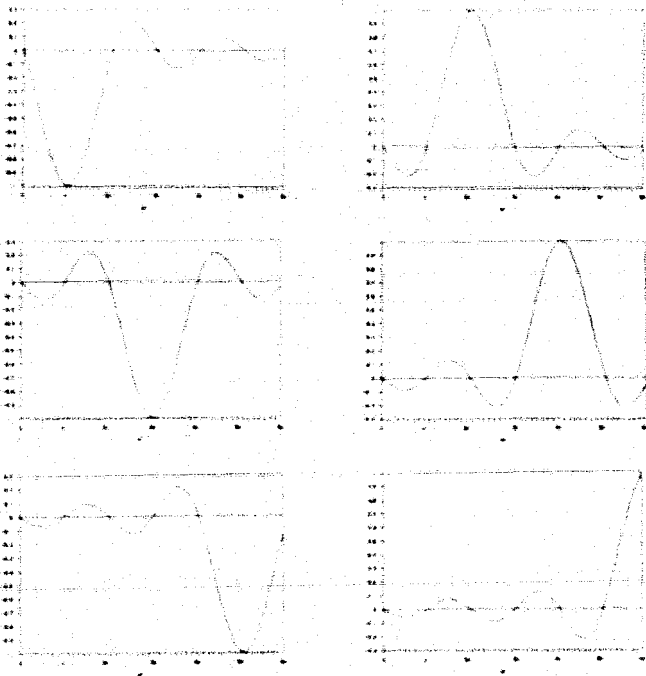


FIGURA 4.7

Al sumar los valores de cada una de estas funciones se obtiene la siguiente gráfica, que como se puede observar, es muy semejante a la señal original:

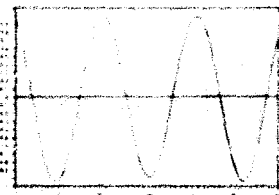


FIGURA 6.6

De esta manera vemos que en teoría es posible implementar un filtro paso bajo mediante software, al simular su respuesta para cada una de las muestras y sumar los resultados. Esto puede realizarse siempre y cuando la velocidad de muestreo sea al menos dos veces mayor a la máxima componente de frecuencia de la señal de entrada.

Si consideramos ahora el caso para cuando se hace $f_s = 2B$, entonces se encuentra que al construir el espectro $\hat{Y}_A(f)$ de la señal muestreada, las réplicas desplazadas en frecuencia de $\Omega_s/2$ se traslapan, de la forma como se ilustra en la siguiente figura:



FIGURA 4.9

En este caso las frecuencias altas son reflejadas en las bajas produciéndose un efecto conocido como *aliasing* o interferencia. Es por lo tanto evidente que si la razón de muestreo es menor a 25, la señal original esta no podrá ser recuperada exactamente de su versión muestreada $e_{\Delta t}^n$ y la información se perderá así en el proceso de muestreo.

4.2 CONSIDERACIONES PRACTICAS PARA LA RECUPERACION DE SEÑALES MUESTREADAS

En teoría 25 muestras tomadas en un segundo, especifican completamente una señal, la cual puede ser recuperada si se hace pasar a través de un filtro con 25dB de ancho de banda de 5 Hz. Ahora bien, si las muestras son tomadas exactamente a una velocidad de 25 Hz entonces el filtro requerido deberá tener una característica de corte infinito. Esto significa que el filtro deberá ser ideal, algo imposible de lograr en la práctica. Asimismo un filtro real con una característica de corte muy abrupta podría ser utilizado para este propósito sin embargo,

resultaría muy complejo y además, parte de la banda lateral inferior sería transmitida a la sección de la señal que se desea recuperar, produciéndose por este motivo problemas de pseudointerferencia.

Esta situación puede evitarse hasta cierto punto, muestreando a una velocidad mayor a $2B$ (figura 4.10). De esta manera se crea una banda de protección, con lo que las exigencias para el diseño del filtro serían menos severas.



FIGURA 4.10

Otra consideración práctica que debe tomarse en cuenta para el proceso de recuperación es que conforme el espectro de la señal muestreada se aleja del origen, su amplitud va decreciendo, de esta forma, si la velocidad de muestreo f_s es muy grande, los periodos de $0.5 \Delta f$ se alejarán tanto que la amplitud de las repeticiones de $0.5 \Delta f$ podrá despreciarse, por lo que el espectro de la señal muestreada será prácticamente igual al de la señal original. Así, es posible recuperar la señal $x(t)$ sin hacer uso de un filtro paso bajas cuando sus componentes de frecuencia no son demasiado altas o cuando el valor f_s es muy grande.

4.3 CAPACIDAD DE LA TARJETA PARA LA DIGITALIZACION DE SEÑALES

Como se recordará, el circuito encargado de la digitalización de señales (DAD 804) está configurado para tomar 1 muestra cada 116 ns por lo que su velocidad de muestreo, equivalente al índice de Nyquist es de 8770 Hz (116ns). Esto significa que en "condiciones ideales" es posible la digitalización de señales cuya componente máxima de frecuencia sea de 4386 Hz. Sin embargo, como se mencionó anteriormente, el proceso de muestreo realizado a esta velocidad, implica la implementación de un "filtro ideal" lo cual es imposible de realizar. Es por lo tanto indispensable que la frecuencia de la forma de onda de entrada sea menor a 4386 Hz. De esta manera, si se cuenta con un filtro con frecuencia de corte entre f y $f/2$, la señal original podrá ser recuperada sin pérdida de información. Un valor recomendable como límite de frecuencia es el de 3500 Hz, de esta forma se tiene un índice de Nyquist de 7000 Hz y por consiguiente una banda de protección de 1750 Hz suficiente para eliminar gradualmente al fenoem de alias interferencia.

Asimismo, si las componentes de la señal de entrada son de Hz, baja frecuencia (como suelen ser aquellas de origen biomédico), ésta podrá digitalizarse y reproducirse sin la necesidad de un filtro obteniéndose (como se verá en el capítulo de pruebas al sistema) resultados satisfactorios.

CAPITULO QUINTO

5. DISEÑO DEL SISTEMA DE ELECTROCARDIOGRAFÍA

Como se comentó en el capítulo tres la tarjeta implementada en hardware ha sido diseñada para realizar un conjunto de tareas específicas las cuales no pueden ser ejecutadas sin la intervención de un programa desarrollado en software que indique el momento y el tipo de tarea que debe llevarse a cabo.

Además de coordinar las acciones del hardware, el software sirve como interfase entre el usuario y la tarjeta, ya que gracias a él es posible convertir los datos binarios en gráficas o valores decimales cuya interpretación es mucho más fácil de realizar. Asimismo, a través del teclado, el software capta los comandos proporcionados por el usuario y los transmite en forma de pules binarios a la sección de la tarjeta encargada de ejecutar la tarea solicitada.

El software además permite variar las aplicaciones sin necesidad de modificar la estructura original del hardware. De esta manera es posible elaborar sistemas de análisis de señal ex. de control de variables físicas o de reconocimiento y síntesis de voz diseñando los programas que permitan el empleo de la nueva aplicación.

En este capítulo se explicará el desarrollo de una posible aplicación de la tarjeta dada mediante el empleo de un programa de graficación en tiempo real.

51 SISTEMA BASICO DE GRAFICACION EN TIEMPO REAL

Hasta el momento solo se ha cumplido uno de los tres objetivos mencionados al inicio de este proyecto, como se recordará, la elaboración de la tarjeta fue el primero de ellos. Ahora es necesario realizar un sistema que permita controlar las actividades de la tarjeta, así como graficar en tiempo real una señal analógica cardíaca.

Para el diseño de este sistema fue necesario seleccionar entre los lenguajes Pascal y C, el que mejor cubriera las exigencias del diseño. Se decidió emplear el lenguaje Pascal por presentar la más alta velocidad en la impresión de gráficas, así como el menor tiempo de compilación. Además Pascal cuenta con las instrucciones de bajo nivel necesarias para permitir el acceso directo tanto a puertos como a las localidades de memoria de la tarjeta de video.

Una vez seleccionado el lenguaje, se procedió a diseñar el programa basado en el diagrama de la figura 5.1, cuyo algoritmo muestra la forma de graficar en tiempo real una señal analógica de baja frecuencia.

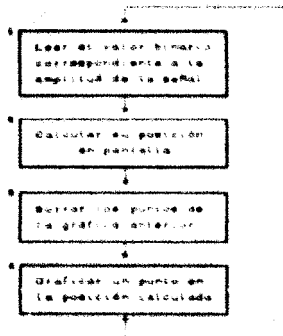


FIGURA 4.1

Para la elaboración del primer bloque se usó la instrucción PORT de Pascal cuyo formato es el siguiente:

DatoDigitalizado = PORT [\$306]

Esta sentencia de la en la variable DatoDigitalizado, el valor que en ese momento aparece en el puerto con dirección 306h, que como se recordará es el que se encuentra conectado a la salida del convertidor de datos analógicos a digitales. Así, cuando el programa ejecute la instrucción anterior, la variable DatoDigitalizado contendrá el valor equivalente a la amplitud de la señal analógica.

Una vez que se ha obtenido esta información es indispensable dibujar su posición en la pantalla. Para obtener el valor en el eje de las Y's fue necesario cambiar el origen del sistema de coordenadas localizado en la esquina superior

izquierda, a la parte inferior izquierda. Asimismo el contenido de la variable `DatoDigitalizado` se redujo a la mitad debido a que el monitor cuenta con tan sólo 200 `bits` en el eje de las `Y`'s, mientras que el valor obtenido del puerto puede variar entre 0 y 255. De este modo si los datos no fueran reducidos, las amplitudes con valor superior a 200 no podrían ser graficadas en la pantalla.

Para obtener las posiciones en el eje de las `X`'s fue necesario tan sólo incrementar en 1 el valor de la coordenada `X` para cada nuevo valor de `Y` calculado. Cabe aclarar que cuando `X` excedía la longitud de la pantalla (639 `bits`) su contenido era fijado con un "0", por lo que el siguiente dato era graficado al inicio de la pantalla.

De esta manera, los valores de `X` y `Y` contenían la posición correspondiente al valor de amplitud de la señal analógica, que se representó de manera gráfica al encender un `bit` en el lugar indicado por estas coordenadas.

Por lo anterior es evidente que conforme se van graficando los valores de la señal, es indispensable borrar los que se desplegaron en el barrido anterior, para realizar esta tarea se elaboró el procedimiento correspondiente al bloque 3 de la figura 5.1, el cual borra los siguientes 8 `bits` a partir del valor actual de `X` (`X`, `X+1`, `X+2`, ..., `X+7`) siempre que `X` sea múltiplo de 8 (0, 8, 16, ..., 632). Así, el procedimiento va "limpiando" la pantalla conforme los valores de la señal van siendo graficados.

Este procedimiento direcciona directamente la memoria de video mediante la instrucción:

MEM(Seg Deep) : 0

la cual almacena el valor 000000 en la direccion indicada por el segmento y desplazamiento almacenados en las variables Seg y Deep respectivamente. De esta manera, al grabar unos ceros en la memoria de video, se borran instantaneamente los pixels controlados por el byte que a sido accedido.

Con esto queda terminado el programa básico de graficación en tiempo real. Sin embargo se le incorporaron procedimientos y funciones tendientes a realizar una aplicación de despliegue de señales cardiacas cuyo desarrollo se explica a continuación.

5.2 DESARROLLO DEL SISTEMA DE ELECTROCARDIOGRAFIA

El Sistema de Electrocardiografía se desarrolló tratando de igualar o mejorar a los equipos ya existentes en el mercado. Opciones de cálculo de frecuencia, incremento y decremento de la amplitud de la señal cardiaca, alarmas y muchas otras funciones más, fueron incorporadas al sistema básico de graficación en tiempo real con el objeto de elaborar un producto que por su eficiencia y bajo costo, pudiera competir con los equipos de electrocardiografía comerciales.

Este sistema presenta además la ventaja (por estar desarrollado en software) de poder ser actualizado con muy pocas o ninguna modificación del hardware. Por ejemplo, la opción de desplegar 1 o más canales puede ser fácilmente incorporada conectando un solo circuito integrado a la tarjeta digitalizadora. Asimismo es posible agregar una base de datos con información de pacientes o bien incorporar un sistema experto para el diagnóstico de enfermedades cardíacas.

Por ahora las opciones con las que cuenta este nuevo sistema de electrocardiografía son suficientes para darnos una buena idea de la actividad eléctrica del corazón.

A continuación, se dará una descripción técnica del sistema comenzando por explicar los procedimientos y diagramas de flujo generales para posteriormente analizar los aspectos particulares del programa.

5.21 DIAGRAMAS DE FLUJO

El diagrama de flujo que proporciona una idea general del funcionamiento del sistema es el elaborado con base en el programa principal. En él se muestran los procedimientos más importantes así como la secuencia en que éstos son ejecutados, cada uno de ellos contribuye en la realización de tareas específicas y en conjunto integran el sistema de electrocardiografía completo.

El diagrama de flujo que representa al programa principal del sistema es el que se muestra en la figura 5.2.

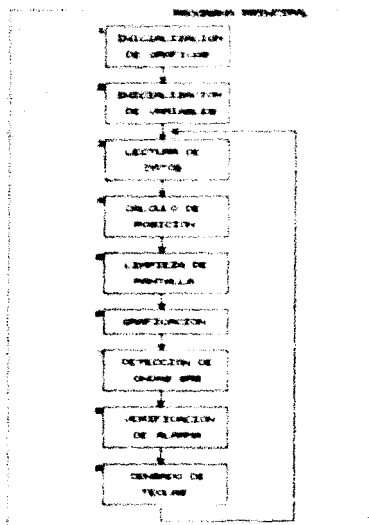


FIGURA 5.2

La descripción de cada uno de los procedimientos de este diagrama es la siguiente:

• El primero de los bloques es empleado para permitir la realización de gráficas en la pantalla de despliegue. Este procedimiento detecta la existencia de la tarjeta de video y verifica que los programas de graficación se encuentren disponibles. Si alguna de las dos condiciones anteriores no se cumple, el programa no podrá ser ejecutado.

• El segundo procedimiento prepara a las variables globales los valores requeridos para poner en marcha el sistema. Sin este bloque, el contenido de las variables podría provocar el funcionamiento no adecuado del programa.

Como puede observarse en el diagrama 5.2, los dos primeros bloques son ejecutados una sola vez debido a que sólo establecen las condiciones iniciales del sistema.

• Los 4 procedimientos siguientes (lectura de datos, cálculo de posición, limpieza de pantalla y graficación) son los encargados de recibir los valores binarios correspondientes a la señal analógica e imprimirlas en la pantalla en forma gráfica. El funcionamiento de estos bloques ya ha sido explicado en la sección del Sistema Básico de Graficación en Tiempo Real al inicio de este capítulo.

• El séptimo procedimiento identifica el momento en el que se presenta un complejo QRS en la señal cardíaca, lo cual se logra

simplemente detectando variaciones repentinas y relativamente grandes de voltaje. Cuando esto sucede, el flujo del programa es enviado a los subprocedimientos del siguiente diagrama.

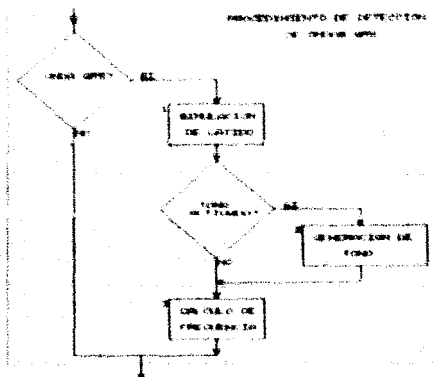


FIGURA 6.3

- o El primero de estos bloques imprime en la parte superior izquierda de la pantalla, un símbolo que simula el latido del corazón del paciente
- o Si el segundo subprocedimiento se encuentra activado, este generará un sonido mientras el símbolo descrito en el punto

anterior aparecer en la pantalla, está en condiciones de que el usuario del sistema tenga una idea del ritmo con que late el corazón del paciente sin que tenga que dirigir su vista al monitor. En este bloque está inhabilitado el flujo del programa para el bloque 1 al 3.

o El último de estos subprocedimientos obtiene la frecuencia de la señal cardíaca y el promedio de los valores calculados en los últimos veinte golpes detectados. El resultado es impreso cada vez que se completa un barrido del monitor.

o El octavo procedimiento del diagrama de flujo del programa principal, es el encargado de comparar en todo momento el valor promedio de frecuencia cardíaca con las variables *LimiteInferior* y *LimiteSuperior* cuyo contenido es determinado por el usuario, si la frecuencia es menor que la primera variable o mayor que la segunda entonces sonará una alarma indicando una situación de posible peligro para el paciente.

o El último bloque de este diagrama es empleado para ejecutar las opciones que el usuario solicita al sistema. Cuando se presiona una tecla, el programa la identifica y de acuerdo con

el valor del código ASCII del carácter leído, el flujo es transferido a alguno de los subprocesamientos mostrados en la figura 5.4

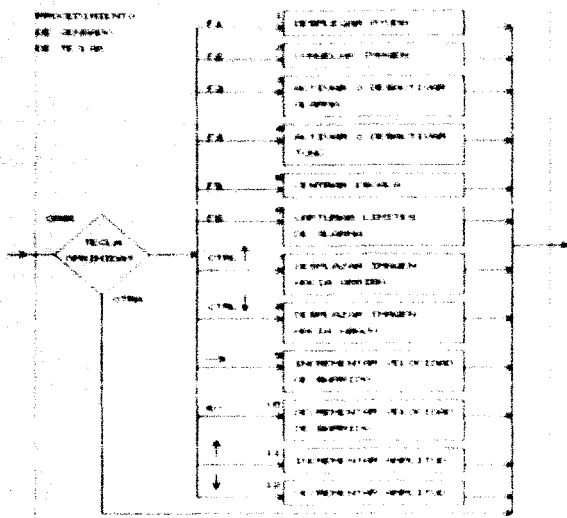


FIGURA 5.4

En este nuevo diagrama, el subprocedimiento del primer bloque despliega en la pantalla mediante la tecla F1, los menús de ayuda con la información necesaria para que el

usuario pueda ejecutar todas las opciones que ofrece el sistema. Si se presiona nuevamente esta tecla, los menús desaparecerán de la pantalla.

□ El segundo bloque es el encargado de congelar la imagen cuando el usuario considera que algo de interés ha ocurrido. Si se presiona la tecla F2, el programa continúa graficando hasta llegar al final de la pantalla, en ese momento, se interrumpe la lectura de nuevos datos con lo que la imagen permanece estática. Al presionar nuevamente F2, el proceso de graficación continúa con los valores que en ese instante presenta la señal cardiaca perdiéndose de este forma, los datos recibidos durante el tiempo en que la imagen se encuentra congelada.

□ El tercer subprocedimiento es empleado para deshabilitar mediante la tecla F3, la señal de alarma activada por una posible situación de peligro del paciente. Para volver a habilitarla, es necesario teclear nuevamente F3.

□ Como ya se ha mencionado, cuando se detecta un complejo QRS, el programa genera una señal audible con el fin de que el usuario del sistema tenga una idea del ritmo con que late el corazón del paciente. Si por alguna razón se desea

deshabilitar este sonido se deberá presionar la tecla F6. con esto el procedimiento de activación y desactivación de tono suspenderá la generación del sonido. Basta oprimir la misma tecla para habilitarlo otra vez.

o Cuando se presiona F6 el flujo del programa es enviado al sexto subprocedimiento el cual dará la posibilidad al usuario de modificar los límites de alarma superior e inferior.

o El séptimo bloque se activa al presionar Ctrl y ↑ al mismo tiempo y es empleado para recorrer la gráfica hacia arriba. Asimismo, tecleando Ctrl y ↓ se obtiene un corrimiento de la gráfica hacia abajo (octavo bloque). Para regresar la imagen a su posición original, el usuario deberá presionar la tecla F5.

o Con los bloques 9 y 10, es posible incrementar o decrementar la velocidad de barrido de la pantalla al presionar, en el caso respectivamente

* Los símbolos ↑ ↓ ← y → indican las teclas de los cursores hacia arriba, hacia abajo, a la izquierda y a la derecha respectivamente.

o Los dos últimos subprocedimientos permiten aumentar o disminuir el tamaño de la gráfica al presionar ↑ o ↓. El funcionamiento de las teclas anteriores (←, →, ↑ y ↓) se explicará con mayor detalle en la siguiente sección (guía del usuario).

o Si ninguna de las teclas anteriores es presionada, el procedimiento de censado de teclas transfiere el control al programa principal sin ejecutar ninguna de las tareas arriba mencionadas.

Cuando el último procedimiento ha terminado, el programa da un salto incondicional hacia el bloque número 3 para leer y graficar un nuevo valor. Cabe aclarar que en cada iteración del programa principal es graficado un solo dato por lo que para borrar toda la pantalla es necesario realizar 638 iteraciones.

A continuación, se presenta un listado del sistema de electrocardiografía en donde puede observarse con mayor detalle la manera en que fueron desarrollados los procedimientos y subprocedimientos que constituyen al programa.

-----DRAWBORDER-----

PROCEDURE DrawBorders: (Dibuja un marco en el Viewport activo)

VAR

Viewport: ViewPortType;

BEGIN

SetColor(1);

SetLineStyle(Solid, 0, NewWidth);

GetViewSettings(Viewport);

WITH Viewport DO

Rectangle(x1, y1, x2, y2);

END;

-----VENTANA-----

PROCEDURE Ventana(a,b,c,d: Integer); (Define un ViewPort con coordenadas
(a,b),(c,d).)

BEGIN

SetViewport(a, b, c, d, ClipOff);

DrawBorders;

END;

-----INICIALIZACION DE GRAFICAS-----

PROCEDURE Inicializaciondegraficas;

(Inicializa graficas y reporta cualquier error que
ocurra)

VAR

InGraphicsMode: boolean; (Bandera de inicializacion de modo grafico)

PathToDriver: string; (Almacena la trayectoria a *.BIN y *.CHR.)

BEGIN

InDirectVideo := False;

(Escritura a video mediante BIOS)

PathToDriver := 'a:\bin';

REPEAT

(\$IFDEF User814

(Validacion para User814 #DEFINE)

GraphDriver := IBM814;

GraphMode := IBM814-1;

(ELSE)

GraphDriver := Detect;

(Autodeteccion)

(ENDIF)

InitGraph(GraphDriver, GraphMode, PathToDriver);

ErrorCode := GraphResult;

IF ErrorCode < 0 THEN BEGIN

(ERROR)

```

writeLn "ERROR DE GRAFICACION:"
IF ErrorCode = gFileNotFound THEN BEGIN
  writeLn "Indique la trayectoria para los archivos Bul (*)
  write (") a presione <Ctrl>Break para terminar:"
  ReadLn PathToServer;
  writeLn
  END;
ELSE
  Halt(1);
Error: Fin de programat
END;
UNTIL ErrorCode = gOk;
MaxX := GetMaxX;
MaxY := GetMaxY;
MaxColor := GetMaxColor;
SetTextStyle(StyleAll, Horizontal, 0);
END;

```

(-----IMPRESION DE ESCALA-----)

```

Procedura ImpresionEscala: (Este procedimiento imprime los
                          valores de escala en el lado
                          izquierdo de la ventana de
                          despliegue.)
Var
  MasDinco, Mas2PDinco : Integer;
  MenDinco, Men2PDinco : Integer;
  Cero : Integer;

```

```

Begin
  SetViewPort(0, 63, 29, 192, 0, 0);
  ClearViewPort;
  SetTextStyle(StyleAll, Horizontal, 2);
  MasDinco := Bottom - 255 Div Amplitud;
  Mas2PDinco := Bottom - 191 Div Amplitud;
  Cero := Offset;
  Men2PDinco := Bottom - 63 Div Amplitud;
  MenDinco := Bottom - 0 Div Amplitud;
  OutTextXY(1, MasDinco, 0, 5.0);
  OutTextXY(1, Mas2PDinco, 0, 2.5);
  OutTextXY(1, Cero, 0, 0.0);
  OutTextXY(1, Men2PDinco, 0, 2.5);
  OutTextXY(1, MenDinco, 0, 5.0);
  SetViewPort(0, 63, 29, 192, 0, 0);
  y:=bottom-Port(0,061 Div Amplitud;
  Move(x1, y);
End;

```

(-----AFILIAR Y/O ZOOM-----)

```

Procedure Factoracion(x, y: Integer; Valors: Integer);

```

```
Const
    Longitud=31
    (Capture los valores de los
    (limites de alarma. )
```

```
Var
    ValorSt : String;
    ArrValor : Array[1..4] of char;
    Cont.I : Byte;
    Car : Char;
    BLANCO : Array[1..20] of integer;
    Error : Integer;
```

```
BEGIN
```

```
Cont:=0;
Car:='';
GETIMAGE(0,1,20,17,BLANCO);
WHILE Car<>#13 DO BEGIN
    repeat
        Car:=ReadKey;
    until (Car<>#0) And (Car<>#9) or (Car=#13) or (Car=#27);
    IF Car=#8 THEN BEGIN
        IF Cont=0 THEN BEGIN
            PUTIMAGE((X*Cont)+8-80,Y,BLANCO,3);
            Cont:=Cont+1;
        END;
    END;
    ELSE
        IF (Cont<Longitud) And (Car<>#13) THEN BEGIN
            Cont:=Cont+1;
            ArrValor[Cont]:=Car;
            OutText(X*Cont+8-80,Y,Car);
        END;
        IF (Cont=Longitud) and (Car=#13) Then BEGIN
            Sound(600);
            Delay(60);
            NoSound;
        END;
    END;
    ValorSt:='';
    FOR I:=1 TO Cont DO
        ValorSt:=concat(ValorSt,ArrValor[I]);
    Val(ValorSt,Valor,Error);
```

```
END;
```

```
..... IMPRIME LIMITES.....
```

```
Procedure ImprimeLimites; (Imprime en pantalla los valores
    (de los limites de alarma. )
```

```
Var
```


Begin

```
XX:=20;YY:=15;
if beep then Sound(200);
SetViewPort(XX,YY,XX+40,YY+10,ClipOn);
PutImage(8,8,CORAZON,XX,YY);
SetViewPort(8,8,63,140,ClearOn);
YY:=0;
moveTo(X,YY);
```

End;

----- [BORRA CORAZON] -----

Procedure BorraCorazon; (Borra el corazn cuando termina el juego.)

Var
XX,YY: Integer;

Begin

```
XX:=20;YY:=15;
if beep then Sound(200);
SetViewPort(XX,YY,XX+40,YY+10,ClipOn);
ClearViewPort;
SetViewPort(8,8,63,140,ClearOn);
YY:=0;
moveTo(X,YY);
X:=10;
```

End;

----- [IMPRIME ALARMA] -----

Procedure ImprimeAlarma; (Imprime un simbolo que indica si la alarma se encuentra o no activada.)

Var

XX,YY,YC,XC: Integer;

Begin

```
XC:=440;
YY:=30;
XC:=0;
YC:=17;
SetViewPort(XX,YY,XX+36,YY+21,ClearOn);
ClearViewPort;
Line(XC,YC,XC+36,YC);
Arc(XC+18,YC,180,360,5);
```



```
Ends
SetViewPort(1,63,639,192,0)
VtYar
move(0,0,0)
```

```
Ends
```

```
-----CAPTURA LIMITES DE ALARMA-----
```

```
Procedure CapturarLimites:
```

```
var
```

```
    X: Integer;
```

```
Begin
```

```
    NoSound;
    SetViewPort(242,2,637,59,0)
    SetImage(0,0,59,59,0)
    ClearViewPort;
    SetTextStyle(5611,Horizontal,5);
    OutTextXY(0,17, "PROPORCIONE LOS LIMITES DE ALARMA");
    OutTextXY(0,18, "PROPORCIONE LOS LIMITES DE ALARMA");
    SetTextStyle(5611,Horizontal,4);
    OutTextXY(0,14,20, "PRESIONE [ENTER] DESPUES DE CADA VALOR");
    SetTextStyle(5611,Horizontal,5);
    OutTextXY(0,25, "INFERIOR : ");
    OutTextXY(0,26, "INFERIOR : ");
    CapturarValor(X,17,36,0,0);
    OutTextXY(0,25, "SUPERIOR : ");
    OutTextXY(0,26, "SUPERIOR : ");
    CapturarValor(X,17,36,0,0);
    Delay(1000);
    ClearViewPort;
    SetImage(0,0,59,59,0);
    IncrImage(1);
```

```
Ends
```

```
-----IMPRIME AYUDA 2-----
```

```
Procedure ImprimeAyuda2; (Imprime el menu 2 de ayuda)
```

```
var
```

```
    X,Y,Esq: Integer;
```

Begin

```

LIMITES=0000
K04=377*Vx+200*Esp+130
NOCOND=0
SetViewPort=190,64,299,191,C1200000
ClearViewPort
Preamble=
SetTextStyle=Small,HEXIZONTAL,0
OutText=XXXXX"2" R E N O 2111
Line=0,16,287,1600
OutText=XXXXXX"1" IF(1) QUITAR MENA DE AYUDA 11
Line=0,1,XX+Esp+3,XX+12,XX+Esp+500
Line=XX+12,XX+Esp+5,XX+5,XX+Esp+200
Line=XX+12,XX+Esp+5,XX+5,XX+Esp+800
OutText=XXXXXX"3" XX+Esp, 2400,VELOCIDAD DE BARRIDO 11
Line=00,XX+2*Esp+5,XX+12,XX+2*Esp+500
Line=00,XX+2*Esp+5,XX+12,XX+2*Esp+200
Line=00,XX+2*Esp+5,XX+12,XX+2*Esp+800
OutText=XXXXXX"30,XX+2*Esp,17600,VELOCIDAD DE BARRIDO 11
Line=00,1,XX+3*Esp+XX+6,XX+3*Esp+210
Line=00,6,XX+3*Esp+2,XX+12,XX+3*Esp+0
Line=00,6,XX+3*Esp+2,XX+6,XX+3*Esp+0
Line=00,5,XX+3*Esp+2,XX+5,XX+3*Esp+0
OutText=XXXXXX"30,XX+3*Esp, INCREMENTAR AMPLITUD 11
Line=00,1,XX+3*4*Esp+XX+6,XX+3*4*Esp+0
Line=00,6,XX+3*4*Esp+XX+12,XX+3*4*Esp+0
Line=00,6,XX+3*4*Esp+2,XX+6,XX+3*4*Esp+0
Line=00,5,XX+3*4*Esp+2,XX+5,XX+3*4*Esp+0
OutText=XXXXXX"30,XX+4*Esp, DECREMENTAR AMPLITUD 11
Line=00,1,XX+3*8*Esp+XX+6,XX+3*8*Esp+0
Line=00,6,XX+3*8*Esp+XX+12,XX+3*8*Esp+0
Line=00,6,XX+3*8*Esp+2,XX+6,XX+3*8*Esp+0
Line=00,5,XX+3*8*Esp+2,XX+5,XX+3*8*Esp+0
OutText=XXXXXX"30,XX+5*Esp, BARRIR IMAGEN 11
Line=00,1,XX+3*8*Esp+XX+6,XX+3*8*Esp+0
Line=00,6,XX+3*8*Esp+XX+12,XX+3*8*Esp+0
Line=XX+5*Esp,XX+2*4*Esp+XX+30,XX+8*Esp+0
Line=XX+5*3*XX+2*4*Esp+XX+30,XX+8*Esp+0
OutText=XXXXXX"30,XX+6*Esp,10000 BARRIR IMAGEN 11
OutText=XXXXXX"30,XX+7*Esp,10000 DECREMENTAR MENA 11
SetViewPort=190,64,299,190,C1110000
Viewport=Port=190,64,299,190,C1110000
XX=240
moveof,XX

```

End:

..... [IMPRIME AYUDA]

Procedura ImprimeAyuda: (Imprime el menu de ayuda)

Var

X,Y, Esp: Integer;

Begin

```
Limite:=200;
X:=20;Y:=24;Esp:=14;

NoSound;
SetViewPort(352,64,636,192,ClipeOnly);
ClearViewPort;
DrawBorder;
SetTextStyle(Small,HORIZONTAL,5);
OutText(X+5,Y,2,' M E N U ');
Line(0,16,227,16);
OutText(X+5,Y,' (F1): 2. OTAR MENU DE AYUDA ');
OutText(X+5,Y,' (F2): CONGELAR IMAGEN ');
OutText(X+5,Y,' (F3): ACTIVAR/DESACTIVAR ALARMA ');
OutText(X+5,Y,' (F4): ACTIVAR/DESACTIVAR TONO ');
OutText(X+5,Y,' (F5): CENTRAR ESCALA ');
OutText(X+5,Y,' (F6): FIJAR LÍMITES DE ALARMA ');
OutText(X+5,Y,' (F7): DESPLEGAR MENU 2 ');
SetViewPort(0,63,636,192,ClipeOnly);
Y:=bottom-Port(0,63,636,192,AmpLitud);
X:=24;
MoveTo(X,Y);
```

End;

..... ELIMINA MENU.....

Procedura EliminaMenu: (Borra el menu de ayuda activado)

Begin

```
SetViewPort(352,70,636,192,ClipeOnly);
ClearViewPort;
SetViewPort(0,63,636,192,ClipeOnly);
Y:=bottom-Port(0,63,636,192,AmpLitud);
X:=24;
Line(22,66,227,66);
MoveTo(X,Y);
Ayuda := False;
Limite := MaxX+8;
```

End;

-----ESPERA-----

Procedura Esperar (Envia un mensaje para indicar que la imagen esta siendo
complejada.)

```
Begin
  If Not ImageCompletada Then Begin
    SetViewport(14, 2, 37, 36) ClearView
    SetImage(0, 0, 39, 56, Image)
    SetViewport(29, 30, 42, 51) ClearView
    ClearViewPort()
    SetTextStyle(Small, Horizontal, 1)
    OutText(5, 3, "COMPLEJANDO")
    OutText(5, 4, "INGRELANDO")
    SetViewport(3, 47, 67, 192) ClearView
    YText:
    MoveTo(5, 3)
  End
End
```

End

-----INFORME DEL K-----

Procedura InformeFact

```
Var
  Step : Integer
  Lugar : Integer
  Segundo : String
```

```
Begin
  SetViewport(20, 193, 629, 192) ClearView
  ClearViewPort()
  SetTextStyle(Small, Horizontal, 2)
  For Step = 3 to Retardo+3 do Begin
    Lugar = Round(617 / (Retardo+3)*Step)
    Str(Step, Segundo)
    OutText(Lugar, 1, Segundo)
    Line(Lugar, 0, Lugar, 3)
  End
  SetViewport(1, 67, 629, 192) ClearView
  YText:
  MoveTo(5, 1)
```

End

-----LENSADO DE TECLAS-----

Procedura Teclas: Var Retardo, Amplitud, Bottom: Integer

```
Var
  Caracter1, Caracter2 : Char (Cinca el teclado y de acuerdo)
```

```

con el valor leído, transfiere
del flujo del programa a la rutina
correspondiente.
Begin
  Caracter1:=ReadKey;
  Case ord(Caracter1) of
    8:Begin
      Caracter2:=ReadKey;
      Case Caracter2 of
        (F1) : (1) If Not(Ayuda) Then Begin (Activa o desactiva los )
              Ayuda:=True; (señal de ayuda. )
              If sonido Then
                ImprimeAyuda1;
              Else
                ImprimeAyuda2;
              End;
              EliminaMarka;
        (F2) : (2) Begin
              Espera1;
              ImagenCongelada:=True; (Congelamiento de imagen)
              End;
        (F3) : (3) Begin (Activa o desactiva la )
              If Alarma Then (señal de alarma. )
                Alarma:= False;
              Else
                Alarma:= True;
              ImprimeAlarma;
              End;
        (F4) : (4) Begin (Activa o desactiva la )
              If Beep Then (señal de tono )
                Beep:= False;
              Else
                Beep:= True;
              ImprimeBocina;
              End;
        (F5) : (5) Begin (Centra la escala)
              Offset:= 65;
              Bottom := Offset*(255 div Amplitud) div 2;
              ImprimeEscala;
              End;
        (F6) : (6) CapturaLmites;
        CTRL_Z:=Z; (Baja Imagen)
        #145: Begin
              Offset:= Offset + 1;
              If Offset = 128 Then Offset:=127;
              Bottom := Offset*(255 div Amplitud) div 2;
              ImprimeEscala;
              End;
      End;
    End;
  End;

```

```

(CTRL ARRIBA) 840: Begin
                Offset:=Offset+4;  (Subir Ingreso)
                If Offset = 0 Then Offset:=1;
                Bottom:=Offset*255 div Amplitud div 2;
                ImprimeEscala;
            End;
        M1: Begin
                Retardo:=Retardo+1;  (Incrementa la vel de barrido)
                If Retardo = 0 Then Retardo:=40;
                ImprimeRet;
            End;
        A1: Begin
                Retardo:=Retardo+1;  (Incrementa la vel de barrido)
                If Retardo = 4 Then Retardo:=8;
                ImprimeRet;
            End;
        M2: Begin
                IncAmplitud:=  (Incrementa la Amplitud)
                If Amplitud = 1 Then Amplitud:=1;
                Bottom:=Offset*255 div Amplitud div 2;
                ImprimeEscala;
            End;
        P1: Begin
                IncAmplitud:=  (Incrementa la amplitud)
                If Amplitud = 1 Then Amplitud:=1;
                Bottom:=Offset*255 div Amplitud div 2;
                ImprimeEscala;
            End;
        D1: If ayuda Then  (Selecciona el menu de ayuda a 1)
                If Menu1 Then Begin  (se despliega)
                    Menu1:=0;
                    ImprimeAyuda;
                End;
                Else Begin
                    Menu1:=1;
                    ImprimeAyuda;
                End;
            End;
        End;
    End;
22: Begin  (Fin de programa)
        Nosounds;
        CloseGraphs;
        Halt(1);
    End;
End;
End;

```


-----BORRADO DE COLUMNAS-----

Procedure DelCol(X: Integer);

(Borra la ventana de despliegue conforme
va siendo borrada.)

Var
I, K: Byte;
J: Integer;

Begin

for I:=K div 84
to K div 80;

for J:=32 to 63 do Begin
del(\$B800:J);
del(\$BA00:J);
J:=J+80;

End;

del(\$BA00:J);
J:=J+80;

for J:=65 to 95 do Begin
del(\$B800:J);
del(\$BA00:J);
J:=J+80;

End;

End;

-----CONSOLA [IMAGEN]-----

Procedure CongelaImagen; (Congela imagen)

Var

Salida1, Salida2: Char;
XX: Integer;

Begin

XX:=5;

nosound;

setviewport(242,2,637,58,Elip/8);

clearviewport;

settextstyle(Small, Horizontal, 5);

outtextxy(XX, 12, 'IMAGEN CONGELADA');

outtextxy(XX, 12, 'IMAGEN CONGELADA');

outtextxy(XX, 33, 'PRESIONE [F2] PARA CONTINUAR');

```

OutText(17,34)      PRESIONE [F1] PARA CONTINUAR);
Repeat
  Salidas:=ReaPaus;
  If Salidas=0 Then
    Salidas:=ReaPaus;
  Until (Salidas=1) Or (Salidas=2) Or 1;
ImagerConsejada:=false;
ClearViewPort;
For Imager (0) For 1000 Put In;
ImagerConsejada:=false;
End;

(----- BUENA ALARMA -----)

Procedure VerificaAlarma;      (Produce el tono cuando la alarma ha
                               )
                               (sido activada)
Begin
  If Alarma Then
    If (FrecFrecedion<=inf) or (FrecFrecedion>sup) Then
      If FrecAlarma Then Begin
        Sound(200);
        FrecAlarma:=false;
      End
    Else Begin
        Nosound;
        FrecAlarma:=true;
      End
    End
End;

(----- IMPRIME FRECUENCIA -----)

Procedure ImprimeFrecuencia;      (Imprime el valor de frecuencia de
                               )
                               (la señal)
Var
  BarridoFrecuencia : Bytes;
  FrecCalculada : String;
  Num.ms.ss.cc : words;

Begin
  BitViewPort(7);
  SetTextJustify(RightText,TopText);
  SetTextStyle(Small,Horizontal,20);
  ClearViewPort;
  FrecFrecedion:=0;

```

```

For BarridoFrecuencia: 1 to ProcesoFrecuencia do
  FrecFrecuencia:=FrecFrecuencia+AnexoFrecuencia*(BarridoFrecuencia)
FrecFrecuencia:=FrecFrecuencia/ProcesoFrecuencia
Set FrecFrecuencia:=0; FrecCalculada:=1
OutText:=100.9; FrecCalculada:=1
OutText:=199.10; FrecCalculada:=1
SetText Justify (LeftText, TopText);
SetViewPort (col=3, row=192, col=9);
WritePort (Port=3) Div Amplitud;
setViewPort;

```

End;

..... CALCULO FRECUENCIA.....

Procedure CalculaFrecuencia; (Calcula el valor de frecuencia)

VAR

hh,mm,ss,cc: Integer;

Begin

Write;

Picos:=True;

GetTime(hh,mm,ss,cc);

Tiempo1 := hh*60*60+mm*60+cc;

If (Tiempo1-Tiempo2)<0 Then

FrecActual:= Trunc(60000/(Tiempo1-Tiempo2));

Tiempo2:=Tiempo1;

AnexoFrecuencia:=ContadorFrecuencia:=FrecActual;

Inc(ContadorFrecuencia);

If ContadorFrecuencia=ProcesoFrecuencia then ContadorFrecuencia:=1;

End;

..... RETARDIO.....

Procedure Pausa(Ref:Byte);

Var

Cuenta*Pausas: LongInt;

Ciclos := 1; LongInt*;

Begin

Case ref of

0 : Ciclos := 145; (3)

```
1 1 00000000 0000 0000
2 1 00000000 0000 0000
3 1 00000000 0000 0000
4 1 00000000 0000 0000
```

Ends:

For Quantafasare 1 to 0000 do

Ends:

----- INICIALIZA PANTALLA Y VARIABLES-----

Procedure InicializaVariables:

Begin:

```
Retardo      1= 30;
Amplitud     1= 20;
Bottle       1= 128;
offset       1= 65;
Alarma       1= True;
ImapenOrbales 1= False;
Lumf         1= 40;
Lum          1= 140;
Tiemposol   1= 10;
Pico         1= False;
Beep         1= True;
FrecAlarma   1= True;
Limite      1= Max-5;
Menu         1= 1;
Ayuda       1= False;
```

For ContadorFrecuencia 1 to Frecuencia do

ArregloFrecuencia[ContadorFrecuencia]=1;

ContadorFrecuencia=1;

Line(0,60,0,1)Print

Verano(20,0,639,190);

Line(0,offset,632,offset);

ImprimeEscala;

ImprimeEjeX;

Verano(0,0,639,60);

SetTitleViewManagerScreenFont,windowTitle,1;1;

OutText(0,0,75,14, "PPM");

Line(14,0,14,639);

OutText(14,0,14, "LIMITES DE ALARMA");

SetTextStyleSmall,fontTitle,1;1;

OutText(14,0,14, "AYUDA");

OutText(14,14,14, "AYUDA");

```
Impr:meAl:mas;
Impr:meBoc:mat;
Impr:meI:bit:es;
F:ib:re:je:O:ra:z:ont;
End:;
```

```
(-----PRESENTACION-----)
```

```
Procedure Presentacion;
```

```
Begin;
Ventana(9,5,639,198);
SetTextStyle(FreezeFont,Horizontal,4);
OutText(68,50,"SISTEMA DE ELECTROCARDIOGRAFIA");
OutText(95,100,"FACULTAD DE INGENIERIA");
Delay(2000);
ClearViewPort;
```

```
End;
```

```
(-----LECTURA DE DATOS-----)
```

```
Procedure LecturaDeDatos;
```

```
Begin;
YAPico:=YPico;
YPico:=Port($306);
```

```
End;
```

```
(-----CALCULA POSICION-----)
```

```
Procedure CalculaPosicion;
```

```
Begin;
Y:=Bottom-YPico Div Amplitud;
xor:=1;
inc(X);
If X=1 then Begin;
NoSound;
If ImageKongrada Then CongradaImage;
Impr:meF:re:cu:en:cia;
Y:=24;
Y:=Bottom-Port($306) Div Amplitud;
moveto(X,Y);
```

```
End;
```

```
End;
```

```
(-----GRAFICA VALOR-----)
```

```
Procedure Graficacion;
```

```

Begin
  If X mod 100 Then CalcDistrib
  FiniteDistrib
  RankDistrib
Ends

----- DETECTION DE UNBAs GR5 -----

Procedure DetectaGR5;

Begin
  If X mod 100 Then CalcDistrib
  If X mod 100 Then BorrDistrib
  If Verifica100xFico And (Not Fico) Then CalculeFrecuencias
  If Verifica100xFico Then Fico:=Falso;
End;

----- PROGRAMA PRINCIPAL -----

BEGIN ( Programa Principal )

  InicializacionDeGráficos;
  Presentacion;
  InicializaVariables;
  While True do Begin
    LecturaDeDatos;
    CalculaPosicions;
    Graficacion;
    DetectaGR5;
    If X mod 100 Then VerificaAlarma;
    If Keypressed then Exit( Retardc, Amplitud, Bottom );
  End;
END. ( Programa Principal )

```

5.3 GUIA DEL USUARIO

Para ejecutar el Sistema de Electrocardiografía, es necesario un computador IBM PC XT, PCjr o compatible, que cuente con al menos 256 Kb de memoria RAM, una tarjeta de Video CGA o algún simulador en software y la tarjeta de digitalización instalada en alguno de los slots. El usuario debe asegurarse además de que el disquete donde se encuentra grabado el programa, incluya un subdirectorio llamado BGI con los siguientes archivos:

CGA.BGI	TRIP.CHR	GRAPH.TPU
LIT.TSR	SANS.CHR	

Además, es necesario conectar la línea positiva de la señal cardíaca a la primera entrada del adaptador acoplado a la computadora (figura 5.5) y la negativa a cualquiera de las terminales de tierra.

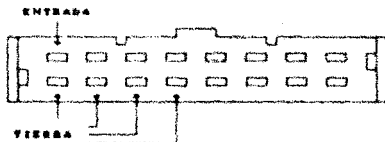


FIGURA 5.5

Una vez que se haya cargado el sistema operativo y aparezca el signo A > en la pantalla, el usuario debe colocar el disco con el Sistema de Electrocardiografía en el drive A y

escribir la palabra **ELECTRO** seguido de la letra **ENTER**, en ese momento empezará la ejecución del sistema.

Si el programa encendió los archivos en el subdirectorio **B01**, desplegará primero la presentación del proyecto y posteriormente una pantalla como la que se muestra en la figura 5 b.

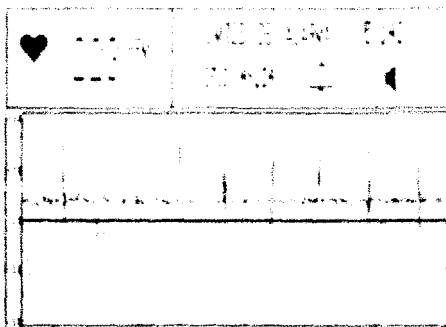


FIGURA 5 b

Como puede observarse, en la parte inferior aparece una ventana donde es graficada la señal cardíaca⁵. En el lado izquierdo de la misma se encuentra una escala que indica el rango

⁵ Para efectos de demostración todas las señales que aparecen en este capítulo fueron simuladas mediante el voltaje generado por un aparato reproductor de audiotapes y por un circuito LM399.

de voltaje que pueda ser desplegado, es decir, si en la parte inferior aparece un -5 y en la superior un $+5$, entonces el sistema se encuentra en ese momento habilitado para recibir señales que varíen entre -5 y 5 voltios. Más adelante se explicará la forma en que este rango puede ser modificado. En la parte inferior de la ventana se encuentra otra escala que indica el tiempo (en segundos) requerido para graficar la sección de la señal cardíaca que va siendo desplegada a lo largo de la pantalla. De esta forma, si el valor máximo de esta escala es de 6 , entonces el programa requerirá de seis segundos para desplegar la señal desde el inicio hasta el final de la pantalla.

En la parte superior izquierda del monitor se encuentra otra ventana con el dibujo de un corazón, el cual se enciende y apaga al mismo tiempo en que se producen los latidos del paciente. Mientras este dibujo se encuentra encendido, el programa genera un sonido que permite tener una idea del ritmo con que se producen los latidos. A un lado del corazón está un número que indica la frecuencia cardíaca del paciente en Pulsos Por Minuto.

En la parte superior de la ventana derecha aparecen los límites de alarma, si la frecuencia de la señal cardíaca es inferior al primer valor o superior al segundo, entonces sonará una alarma para informarle al médico que su paciente podría estar en una situación anormal. La figura de la campana indica que la alarma se encuentra activada. A la derecha de la pantalla aparece la figura de un altavoz que informa si el sonido que genera la

[F2] Permite detener la graficación de la señal si el usuario ha visto algo de interés en la pantalla, en ese momento el programa desplegará la palabra CONGELANDO y detendrá la imagen cuando haya terminado de graficar esa sección de la señal cardiaca. Para continuar con el proceso de graficación, el usuario deberá teclear F2 o ENTER

[F3] Es empleada para inhabilitar la señal de alarma cuando ésta ha sido activada. Si la tecla F3 es presionada la figura de la campana será tachada indicando que ha quedado desactivada (figura 5.8)

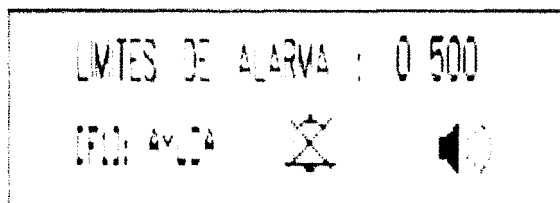


FIGURA 5.8

Para volver a activar la alarma debe presionarse nuevamente la tecla F3

[F4] Se utiliza para desactivar el tono producido al graficar una onda QRS. Al igual que con la señal de alarma, cuando

se presiona F4, la figura del altavoz es tachada (figura 5.9).
Para volver a activar el tono basta con presionar F4.

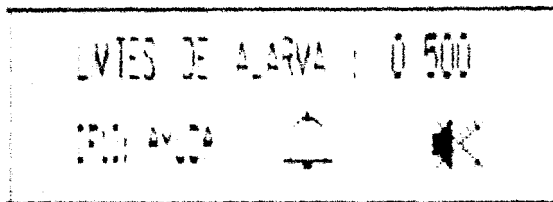


FIGURA 5.9

{F8} Regresa la posición de la escala (situada a la izquierda de la pantalla) a su lugar original cuando ésta ha sido desplazada.

{F6} Se emplea para modificar los límites de alarma. Cuando la tecla F6 es presionada, en la ventana superior derecha aparece el siguiente mensaje:

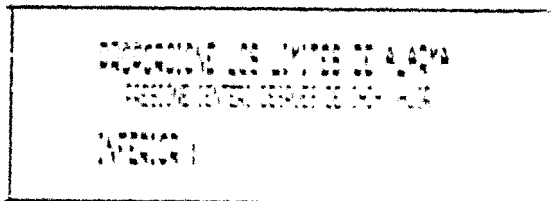


FIGURA 5.10

en ese momento el usuario deberá proporcionar el nuevo valor del límite inferior y posteriormente el del límite superior después de teclear cada cantidad es necesario presionar ENTER.

[PgOn] Despliega el menú mostrado en la figura 5.11 que contiene información adicional para el manejo del sistema

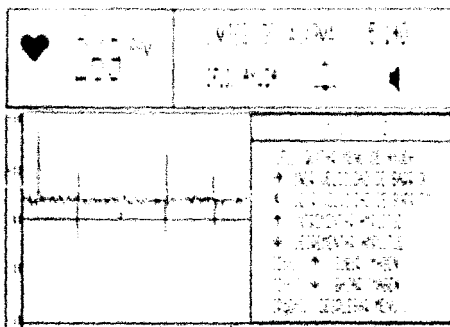


FIGURA 5.11

La descripción de cada una de las opciones de este segundo menú es la siguiente

[F1] Elimina el menú 2 de la pantalla volviendo la ventana de despliegue a su tamaño normal.

→ : Al presionar el cursor con la flecha a la derecha, el tiempo de graficación se reduce, dando a la señal cardíaca un efecto de expansión a lo ancho de la pantalla, esto permite

observar los detalles del electrocardiograma con mayor claridad. Presionando el cursor a la izquierda, el tiempo de graficación se incrementará con lo que la imagen será comprimida. De esta manera, se podrán desplegar porciones relativamente grandes del electrocardiograma, aunque con poco detalle.

Para tener una idea más clara de la forma en que trabajan estas teclas, se presentan a continuación dos gráficas de la misma señal de entrada pero con los efectos de ampliación y compresión de la imagen arriba descritos.

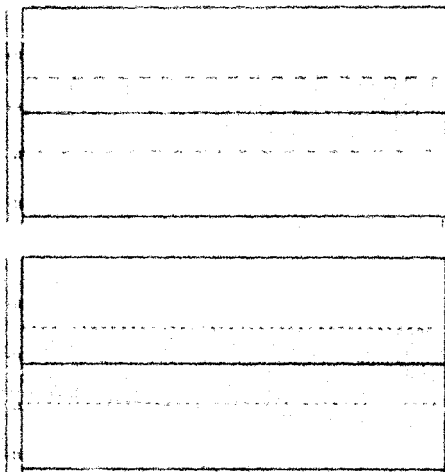


FIGURA 5.12

↑: El cursor hacia arriba incrementa la amplitud de la señal hasta un doble de su tamaño original. De igual manera, la presión de ↓ decremente la amplitud hasta una quinta parte.

El efecto que producen estas teclas se muestra en las siguientes gráficas:

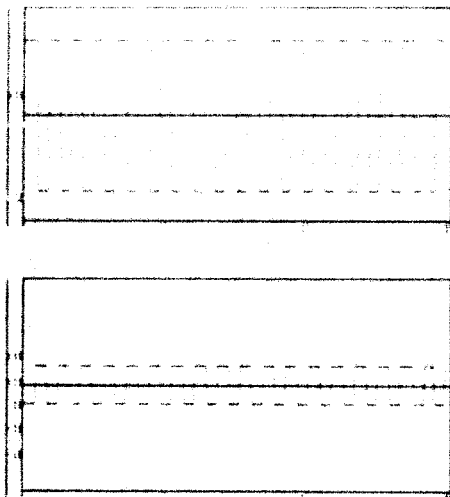


FIGURA 8.28

Ctrl ↑: Presionadas al mismo tiempo, desplazan tanto la gráfica de la señal como los valores de la escala de voltaje hacia

arriba; mientras que con **Ctrl J** se desplazan hacia abajo. Estos comandos resultan útiles cuando se desea colocar la imagen en alguna parte específica de la ventana de despliegue.

Un ejemplo de desplazamiento de gráfica se muestra a continuación:

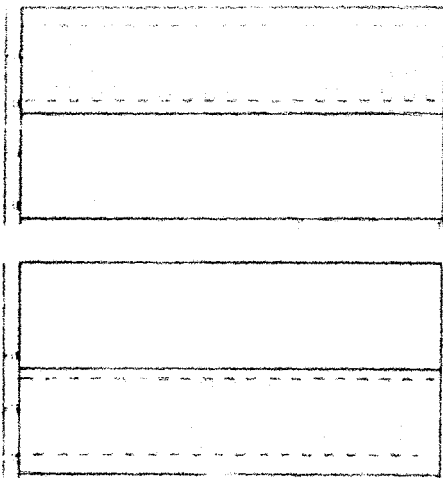


FIGURA 9.44

[PgOn] Despliega nuevamente en pantalla el menú 1 cuyas opciones ya han sido descritas.

Si el usuario desea terminar la ejecución del sistema, bastará con que presione la tecla ESC.

Con la elaboración de este programa, ha sido alcanzado el segundo objetivo del proyecto, lo que resta ahora es probar el funcionamiento de la tarjeta y verificar que tanto las gráficas como los valores presentados por el sistema de electrocardiografía sean correctos, esto se describirá en el siguiente capítulo.

CAPITULO SEXTO

6. ANALISIS DE RESULTADOS

Hasta ahora, no se ha demostrado tanto el circuito de digitalización de señales como el programa de graficación en tiempo real de la señal en tres canales a 250ms por papel y se de funcionar en una computadora personal y de emitir resultados sin embargo esto no basta para afirmar que en verdad funcionan adecuadamente. Para ello es necesario realizar un análisis de resultados para verificar que estos realmente corresponden a los valores esperados.

Para comprobar el funcionamiento de la tarjeta se compararon los datos emitidos por cada una de las secciones con un conjunto de entradas de valores conocidos para posteriormente calcular con esta información los porcentajes de error así como el índice de correlación⁴ que establece la asociación lineal entre los datos teóricos y los obtenidos de manera experimental. Además, para comprobar que las graficas desplegadas por el sistema de electrocardiografía fueran correctas se digitalizó una señal cardiaca real y se comparó con la generada por un equipo comercial. Los resultados obtenidos se presentan a continuación.

⁴ La forma como se calcula este coeficiente de correlación puede ser interpretada de pueden consultarse en el apéndice B.

6.1 SECCION DE CALCULO DE FRECUENCIA

Con el fin de obtener las mediciones de frecuencia realizadas por esta sección, se utilizó mediante un generador de funciones una señal periódica cuya frecuencia fue incrementándose de 100 en 100 ms partiendo de 0 ms. Cada valor fue medido con la computadora y registrado a fin de realizar las comparaciones con los resultados teóricos. El ajuste de las frecuencias se realizó con la ayuda de un osciloscopio digital modelo 35000, cuya resolución no permitía tomar lecturas del tipo exactas por tal motivo fue necesario siempre determinar el valor de incertidumbre de la medición, el cual se obtiene al calcular la mitad de la diferencia entre el valor de las subdivisiones inferiores y superiores más próximas al resultado que se ha leído. De esta manera se tiene que si la frecuencia registrada por la computadora se encuentra dentro del rango de incertidumbre, entonces puede considerarse que las variaciones entre los valores teóricos y los experimentales no son relevantes, por lo tanto, las mediciones realizadas por la computadora son exactas.

La tabla 6.1 muestra los valores de frecuencia de la señal generada con su respectiva incertidumbre. La lectura

tomada por la computadora, así como los porcentajes de error permitidos y reales obtenidos en cada una de las ediciones

VALOR DE FRECUENCIA LA SEÑAL - Hz	FRECUENCIA CALCULADA - Hz	ERROR PERMISIBLE	ERROR CALCULADO
0000 ± 0	0000	± 0.00 %	± 0.00 %
0200 ± 4	0201	± 2.00 %	± 0.50 %
0400 ± 8	0390	± 2.00 %	0.25 %
0600 ± 8	0597	± 1.66 %	0.50 %
0800 ± 12	0805	± 1.50 %	± 0.50 %
1000 ± 10	1000	± 1.00 %	± 0.00 %
1200 ± 20	1200	± 2.42 %	± 0.00 %
1400 ± 10	1400	± 1.42 %	± 0.00 %
1600 ± 20	1602	± 1.25 %	± 0.12 %
1800 ± 22	1814	± 1.22 %	± 0.22 %
2000 ± 20	2000	± 1.00 %	± 0.00 %
2200 ± 40	2198	± 2.27 %	0.09 %
2400 ± 30	2400	± 1.21 %	± 0.00 %
2600 ± 33	2594	± 1.27 %	0.23 %
2800 ± 30	2784	± 1.30 %	0.30 %
3000 ± 44	2990	± 1.47 %	0.13 %
3200 ± 50	3191	± 1.56 %	0.10 %
3400 ± 51	3400	± 1.47 %	± 0.10 %
3600 ± 60	3615	± 1.66 %	± 0.42 %
3800 ± 71	3809	± 1.87 %	± 0.00 %
4000 ± 70	4008	± 1.75 %	± 0.10 %

TABLA 0.1

Con el propósito de presentar una mejor interpretación de los valores estadísticos se graficaron los porcentajes de error

peristaltica y reales obteniendose el resultado mostrado en la figura 4.1

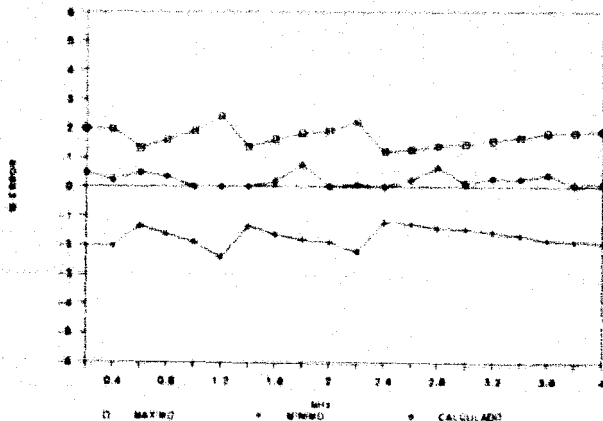


FIGURA 4.1

Como se muestra, todos los valores de frecuencia leídos con la computadora se encuentran dentro del rango de error permisible por lo que puede considerarse que las mediciones fueron correctas.

Además de la representación gráfica de los resultados, se calculó el coeficiente de correlación entre los valores de frecuencia práctica y teórica, el resultado obtenido fue

$$r = 0.99988$$

el cual indica una fuerte relación lineal entre ambas variables.

6.2. SECCION DE CONVERSION DE DATOS ANALOGICOS A DIGITALES

La verificación de los datos digitales generados por esta sección se realizó de manera semejante a la anterior es decir, mediante una fuente de poder se introdujeron a la entrada del convertidor A/D una serie de voltajes conocidos cuyas amplitudes fueron incrementándose de 0.5 en 0.5 Volts. Cada uno de estos valores fue medido con la liculadora y registrado con el fin de realizar las comparaciones con los resultados teóricos. A diferencia de las pruebas anteriores, los valores de voltaje ajustados para llevar a cabo este experimento, no presentan incertidumbre debido a que fueron medidos con un multímetro digital (FLUKE modelo 800DA), por lo que la lectura de los datos se realizó de manera directa.

La tabla 4.2 muestra las amplitudes de voltaje suministradas al convertidor, los valores digitales teóricos y prácticos correspondientes así como el error de linealidad y su porcentaje en ella se puede observar que el error de linealidad máximo encontrado fue de 1 unidades de diferencia entre los valores reales y los calculados lo que corresponde a tan sólo un 0.78% del valor máximo de la escala de salida (128).

VOLTAGE IN ENTRADA	VOLTAGE ENTRADA V _E	VOLTAGE SALIDA V _S	ERROR (%)	WILEY ERROR ANALISIS
0.0V	0.0	0.0	0	0.00%
0.5V	0.5	0.5	0	0.00%
1.0V	1.0	1.0	0	0.00%
1.5V	1.5	1.5	0	0.00%
2.0V	2.0	2.0	0	0.00%
2.5V	2.5	2.5	0	0.00%
3.0V	3.0	3.0	0	0.00%
3.5V	3.5	3.5	0	0.00%
4.0V	4.0	4.0	0	0.00%
4.5V	4.5	4.5	0	0.00%
5.0V	5.0	5.0	0	0.00%
5.5V	5.5	5.5	0	0.00%
6.0V	6.0	6.0	0	0.00%
6.5V	6.5	6.5	0	0.00%
7.0V	7.0	7.0	0	0.00%
7.5V	7.5	7.5	0	0.00%
8.0V	8.0	8.0	0	0.00%
8.5V	8.5	8.5	0	0.00%
9.0V	9.0	9.0	0	0.00%
9.5V	9.5	9.5	0	0.00%
10.0V	10.0	10.0	0	0.00%
10.5V	10.5	10.5	0	0.00%
11.0V	11.0	11.0	0	0.00%
11.5V	11.5	11.5	0	0.00%
12.0V	12.0	12.0	0	0.00%
12.5V	12.5	12.5	0	0.00%
13.0V	13.0	13.0	0	0.00%
13.5V	13.5	13.5	0	0.00%
14.0V	14.0	14.0	0	0.00%
14.5V	14.5	14.5	0	0.00%
15.0V	15.0	15.0	0	0.00%
15.5V	15.5	15.5	0	0.00%
16.0V	16.0	16.0	0	0.00%
16.5V	16.5	16.5	0	0.00%
17.0V	17.0	17.0	0	0.00%
17.5V	17.5	17.5	0	0.00%
18.0V	18.0	18.0	0	0.00%
18.5V	18.5	18.5	0	0.00%
19.0V	19.0	19.0	0	0.00%
19.5V	19.5	19.5	0	0.00%
20.0V	20.0	20.0	0	0.00%
20.5V	20.5	20.5	0	0.00%
21.0V	21.0	21.0	0	0.00%
21.5V	21.5	21.5	0	0.00%
22.0V	22.0	22.0	0	0.00%
22.5V	22.5	22.5	0	0.00%
23.0V	23.0	23.0	0	0.00%
23.5V	23.5	23.5	0	0.00%
24.0V	24.0	24.0	0	0.00%
24.5V	24.5	24.5	0	0.00%
25.0V	25.0	25.0	0	0.00%
25.5V	25.5	25.5	0	0.00%
26.0V	26.0	26.0	0	0.00%
26.5V	26.5	26.5	0	0.00%
27.0V	27.0	27.0	0	0.00%
27.5V	27.5	27.5	0	0.00%
28.0V	28.0	28.0	0	0.00%
28.5V	28.5	28.5	0	0.00%
29.0V	29.0	29.0	0	0.00%
29.5V	29.5	29.5	0	0.00%
30.0V	30.0	30.0	0	0.00%
30.5V	30.5	30.5	0	0.00%
31.0V	31.0	31.0	0	0.00%
31.5V	31.5	31.5	0	0.00%
32.0V	32.0	32.0	0	0.00%
32.5V	32.5	32.5	0	0.00%
33.0V	33.0	33.0	0	0.00%
33.5V	33.5	33.5	0	0.00%
34.0V	34.0	34.0	0	0.00%
34.5V	34.5	34.5	0	0.00%
35.0V	35.0	35.0	0	0.00%
35.5V	35.5	35.5	0	0.00%
36.0V	36.0	36.0	0	0.00%
36.5V	36.5	36.5	0	0.00%
37.0V	37.0	37.0	0	0.00%
37.5V	37.5	37.5	0	0.00%
38.0V	38.0	38.0	0	0.00%
38.5V	38.5	38.5	0	0.00%
39.0V	39.0	39.0	0	0.00%
39.5V	39.5	39.5	0	0.00%
40.0V	40.0	40.0	0	0.00%
40.5V	40.5	40.5	0	0.00%
41.0V	41.0	41.0	0	0.00%
41.5V	41.5	41.5	0	0.00%
42.0V	42.0	42.0	0	0.00%
42.5V	42.5	42.5	0	0.00%
43.0V	43.0	43.0	0	0.00%
43.5V	43.5	43.5	0	0.00%
44.0V	44.0	44.0	0	0.00%
44.5V	44.5	44.5	0	0.00%
45.0V	45.0	45.0	0	0.00%
45.5V	45.5	45.5	0	0.00%
46.0V	46.0	46.0	0	0.00%
46.5V	46.5	46.5	0	0.00%
47.0V	47.0	47.0	0	0.00%
47.5V	47.5	47.5	0	0.00%
48.0V	48.0	48.0	0	0.00%
48.5V	48.5	48.5	0	0.00%
49.0V	49.0	49.0	0	0.00%
49.5V	49.5	49.5	0	0.00%
50.0V	50.0	50.0	0	0.00%
50.5V	50.5	50.5	0	0.00%
51.0V	51.0	51.0	0	0.00%
51.5V	51.5	51.5	0	0.00%
52.0V	52.0	52.0	0	0.00%
52.5V	52.5	52.5	0	0.00%
53.0V	53.0	53.0	0	0.00%
53.5V	53.5	53.5	0	0.00%
54.0V	54.0	54.0	0	0.00%
54.5V	54.5	54.5	0	0.00%
55.0V	55.0	55.0	0	0.00%
55.5V	55.5	55.5	0	0.00%
56.0V	56.0	56.0	0	0.00%
56.5V	56.5	56.5	0	0.00%
57.0V	57.0	57.0	0	0.00%
57.5V	57.5	57.5	0	0.00%
58.0V	58.0	58.0	0	0.00%
58.5V	58.5	58.5	0	0.00%
59.0V	59.0	59.0	0	0.00%
59.5V	59.5	59.5	0	0.00%
60.0V	60.0	60.0	0	0.00%
60.5V	60.5	60.5	0	0.00%
61.0V	61.0	61.0	0	0.00%
61.5V	61.5	61.5	0	0.00%
62.0V	62.0	62.0	0	0.00%
62.5V	62.5	62.5	0	0.00%
63.0V	63.0	63.0	0	0.00%
63.5V	63.5	63.5	0	0.00%
64.0V	64.0	64.0	0	0.00%
64.5V	64.5	64.5	0	0.00%
65.0V	65.0	65.0	0	0.00%
65.5V	65.5	65.5	0	0.00%
66.0V	66.0	66.0	0	0.00%
66.5V	66.5	66.5	0	0.00%
67.0V	67.0	67.0	0	0.00%
67.5V	67.5	67.5	0	0.00%
68.0V	68.0	68.0	0	0.00%
68.5V	68.5	68.5	0	0.00%
69.0V	69.0	69.0	0	0.00%
69.5V	69.5	69.5	0	0.00%
70.0V	70.0	70.0	0	0.00%
70.5V	70.5	70.5	0	0.00%
71.0V	71.0	71.0	0	0.00%
71.5V	71.5	71.5	0	0.00%
72.0V	72.0	72.0	0	0.00%
72.5V	72.5	72.5	0	0.00%
73.0V	73.0	73.0	0	0.00%
73.5V	73.5	73.5	0	0.00%
74.0V	74.0	74.0	0	0.00%
74.5V	74.5	74.5	0	0.00%
75.0V	75.0	75.0	0	0.00%
75.5V	75.5	75.5	0	0.00%
76.0V	76.0	76.0	0	0.00%
76.5V	76.5	76.5	0	0.00%
77.0V	77.0	77.0	0	0.00%
77.5V	77.5	77.5	0	0.00%
78.0V	78.0	78.0	0	0.00%
78.5V	78.5	78.5	0	0.00%
79.0V	79.0	79.0	0	0.00%
79.5V	79.5	79.5	0	0.00%
80.0V	80.0	80.0	0	0.00%
80.5V	80.5	80.5	0	0.00%
81.0V	81.0	81.0	0	0.00%
81.5V	81.5	81.5	0	0.00%
82.0V	82.0	82.0	0	0.00%
82.5V	82.5	82.5	0	0.00%
83.0V	83.0	83.0	0	0.00%
83.5V	83.5	83.5	0	0.00%
84.0V	84.0	84.0	0	0.00%
84.5V	84.5	84.5	0	0.00%
85.0V	85.0	85.0	0	0.00%
85.5V	85.5	85.5	0	0.00%
86.0V	86.0	86.0	0	0.00%
86.5V	86.5	86.5	0	0.00%
87.0V	87.0	87.0	0	0.00%
87.5V	87.5	87.5	0	0.00%
88.0V	88.0	88.0	0	0.00%
88.5V	88.5	88.5	0	0.00%
89.0V	89.0	89.0	0	0.00%
89.5V	89.5	89.5	0	0.00%
90.0V	90.0	90.0	0	0.00%
90.5V	90.5	90.5	0	0.00%
91.0V	91.0	91.0	0	0.00%
91.5V	91.5	91.5	0	0.00%
92.0V	92.0	92.0	0	0.00%
92.5V	92.5	92.5	0	0.00%
93.0V	93.0	93.0	0	0.00%
93.5V	93.5	93.5	0	0.00%
94.0V	94.0	94.0	0	0.00%
94.5V	94.5	94.5	0	0.00%
95.0V	95.0	95.0	0	0.00%
95.5V	95.5	95.5	0	0.00%
96.0V	96.0	96.0	0	0.00%
96.5V	96.5	96.5	0	0.00%
97.0V	97.0	97.0	0	0.00%
97.5V	97.5	97.5	0	0.00%
98.0V	98.0	98.0	0	0.00%
98.5V	98.5	98.5	0	0.00%
99.0V	99.0	99.0	0	0.00%
99.5V	99.5	99.5	0	0.00%
100.0V	100.0	100.0	0	0.00%

TABLE 1

El diagrama de la figura 61 presenta una mejor interpretación de estos datos. En él han sido graficados los

valores digitales esperados y calculados que se muestran en la tabla anterior

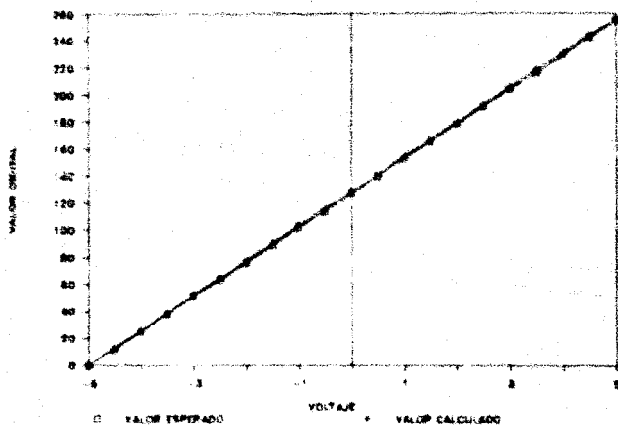


FIGURA # 1

El coeficiente de correlación correspondientes a estas variables es

$$r = 0.999907$$

6.3 SECCION DE CONVERSION DE DATOS DIGITALES A ANALOGICOS

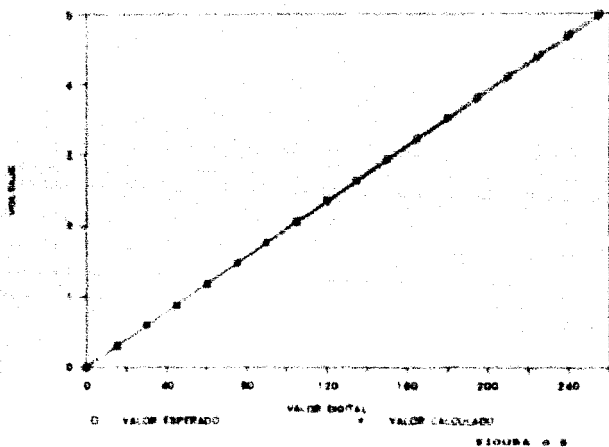
Para probar el funcionamiento de esta seccion se generaron en el computador un conjunto de datos digitales de 8 bits los cuales fueron enviados a la entrada del convertidor. Los valores de voltaje obtenidos a su salida fueron medidos con la ayuda del multímetro digital F1 K11 y comparados con los teóricos.

La tabla 6.3 muestra los valores suministrados al convertidor, los resultados teóricos y prácticos correspondientes así como el error y el porcentaje de error que existe entre ellos.

ENTRADA	V. ESPERADO	V. CALCULADO	ERROR	% DE ERROR
0	0.00V	0.04V	0.04V	0.08
15	0.20V	0.20V	0.00V	0.00
30	0.40V	0.36V	0.04V	0.08
45	0.60V	0.56V	0.04V	0.08
60	0.80V	0.72V	0.08V	0.10
75	1.00V	0.92V	0.08V	0.10
90	1.20V	1.08V	0.12V	0.10
105	1.40V	1.24V	0.16V	0.12
120	1.60V	1.36V	0.24V	0.15
135	1.80V	1.52V	0.28V	0.16
150	2.00V	1.68V	0.32V	0.16
165	2.20V	1.84V	0.36V	0.16
180	2.40V	2.00V	0.40V	0.17
195	2.60V	2.16V	0.44V	0.17
210	2.80V	2.32V	0.48V	0.17
225	3.00V	2.48V	0.52V	0.17
240	3.20V	2.64V	0.56V	0.18
255	3.40V	2.80V	0.60V	0.18

TABLA 6.3

El porcentaje máximo de error encontrado fue del 1.0%. Este es un valor muy pequeño al considerarse que se trata de una diferencia de tan solo 0.25 volts entre los valores esperados y los calculados. Estos datos se muestran gráficamente en la figura 6.3.



Según se muestra los errores de linealidad son tan pequeños que las curvas de los valores teóricos y prácticos aparecen traslapadas en la gráfica. El índice de correlación obtenido para este nuevo análisis fue:

$$r = 0.99995$$

6.4. PRUEBAS IN VIVO

Una vez realizadas las pruebas anteriores, se llevaron a cabo los experimentos tendientes a comprobar el funcionamiento del sistema de electrocardiografía. Para ello, se introdujo a la entrada del convertidor una señal estandarizada proveniente de un perro que en ese momento estaba siendo intervenido quirúrgicamente. La siguiente figura muestra el electrocardiograma obtenido con el equipo Scalet Packard.

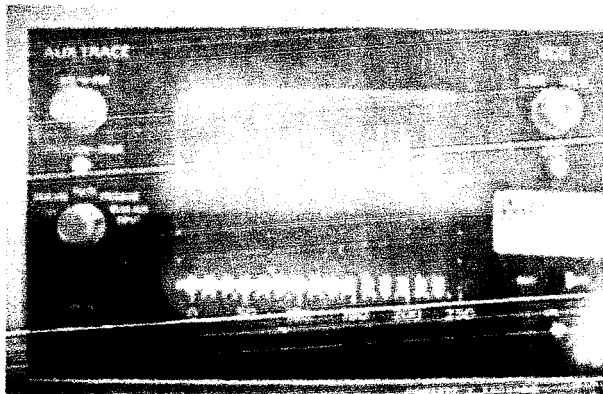
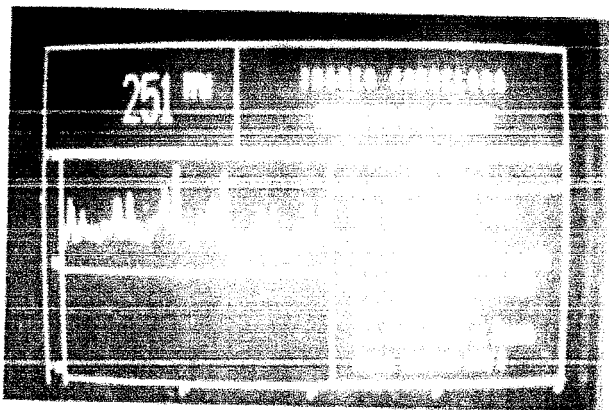


FIGURA 6.4

Al mismo tiempo, la señal cardiaca fue enviada a la computadora con el fin de comparar estas gráficas. El resultado obtenido fue el siguiente:

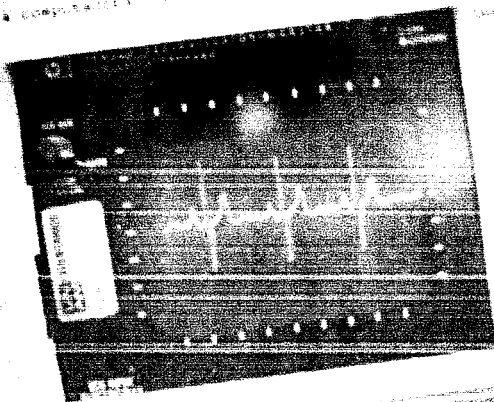


Las dos fotografías anteriores no fueron tomadas simultáneamente, por lo que las señales mostradas en cada una de ellas no corresponden exactamente al mismo intervalo de tiempo. La opción de desplegar la frecuencia cardiaca.

Las dos fotografías anteriores no fueron tomadas simultáneamente, por lo que las señales mostradas en cada una de ellas no corresponden exactamente al mismo intervalo de tiempo.

Otra prueba realizada al sistema de electrocardiografía consistió en graficar ahora una señal proveniente de un corazón

humano los resultados obtenidos en el estudio experimental
como en la comparación con los resultados obtenidos en el estudio



Como se puede ver, al igual que con el equipo comercial, la computadora es capaz de desplegar claramente más de una de las ondas que constituyen los ciclos de actividad cardíaca.

CAPITULO SEPTIMO

7. OTRAS APLICACIONES

El uso de la tarjeta digitalizadora no solo se limita a aplicaciones de conversión de datos para su graficación en tiempo real. Con ella es posible además elaborar sistemas de control, medición, procesamiento de voz, generación de señales etc.

En esta capítulo se describen cinco posibles aplicaciones que se pueden desarrollar sin alterar la estructura original de la tarjeta.

7.1 CONTROL DE VELOCIDAD PARA UN MOTOR DE C.D.

En ocasiones, se tiene la necesidad de mantener constante la velocidad de un motor. En las plantas de producción por ejemplo, existen bandas transportadoras llevando envases para ser llenados con algún producto determinado, si dichas bandas no mantienen su velocidad, es muy probable que el producto se derrame ocasionando pérdidas. Así como en el ejemplo anterior, podemos encontrar un sinnúmero de aplicaciones en donde se requiere de este tipo de accionamiento de control.

A continuación se describirá la manera como esto puede llevarse a cabo con el uso de una computadora personal, la tarjeta de digitalización y el software especialmente diseñado para este propósito.

En esta aplicación es necesario conocer primero el número de revoluciones que da el motor en un segundo, esto puede lograrse mediante el transductor digital de desplazamiento angular mostrado en la figura 7.1.

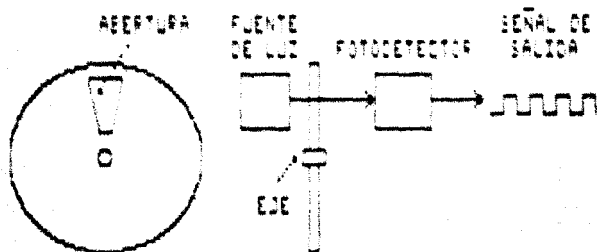


FIGURA 7.1

Conforme la rueda gira, la fuente de luz transmite una serie de pulsos al fotodetector obteniéndose a su salida una señal digital cuya frecuencia es igual al número de revoluciones por segundo. La señal que se obtiene del transductor debe ser introducida a la tarjeta en donde su frecuencia es calculada (figura 7.2). Al leer el resultado, emitido por el frecuencímetro, el computador conocerá la velocidad con la que gira el motor. Ahora bien, si el voltaje de alimentación es suministrado por la f_{AC} mediante la sección de conversión de datos digitales a analógicos y una etapa de potencia, entonces la

computadora podrá aumentar o disminuir las revoluciones hasta alcanzar la velocidad deseada.



FIGURA 7.2

El software de esta aplicación debe primero obtener el valor calculado por el frecuencímetro incorporado a la tarjeta que es igual a la velocidad del motor. Si la lectura es menor a la deseada el programa deberá incrementar paulatinamente el voltaje de alimentación al motor pero si por el contrario el valor leído es mayor el voltaje deberá disminuirse hasta alcanzar la velocidad requerida.

7.2 MEDICIÓN DE CAPACITANCIAS

A diferencia de las resistencias, la medición de los valores de capacitancia no es una tarea fácil de realizar, para ello se requiere de instrumentos de medición poco comunes como pueden ser los puente de impedancias o de desarrrollas matemáticas no tanto complejas cuando se emplean métodos de medición indirectos.

Ahora bien, si las resistencias R_a y R_b son conocidas la computadora podrá calcular fácilmente el valor de capacitancia si despegar el coeficiente C de la fórmula anterior, es decir $C = \frac{R_a R_b}{\dots}$. Una vez obtenido el resultado este podrá ser mostrado al usuario a través de la pantalla de video.

7.3 GENERADOR PROGRAMABLE DE SEÑALES

Para el análisis de circuitos analógicos, en ocasiones se requiere de la salida por de una señal de entrada para probar su funcionamiento. En este tipo de aplicaciones, la computadora puede ser programada para obtener a la salida del convertidor D/A incorporado en la tarjeta, la forma de onda requerida para llevar a cabo dicho análisis.

En la generación de la señal es indispensable almacenar la forma de un ciclo completo en la memoria de la computadora, esto puede realizarse generando mediante ecuaciones matemáticas, una serie de datos cuyos valores contenidos entre 0 y 255 correspondan a la magnitud que va tomando la señal analógica a través del tiempo.

Habiendo quedado grabado en memoria un periodo completo, este debe ser constantemente enviado al convertidor D/A para obtener la señal periódica requerida. De este modo, dependiendo de la forma del periodo que se haya grabado, pueden obtenerse señales cuadradas, triangulares, dientes de sierra etc.

Para incrementar o disminuir la frecuencia, basta con variar la memoria a mayor o menor velocidad.

7.4. INTERFASE DE VOZ

Una aplicación un poco más interesante que las anteriores consiste en digitalizar una señal de voz y almacenar los valores binarios obtenidos en memoria, con el fin de que la computadora lea dichos valores y emita una señal para dar indicaciones habladas a los usuarios.

Lo anterior es posible debido a que la señal de voz está limitada estrictamente en banda a 3.3 kHz, por lo que la velocidad de muestreo de Nyquist es entonces de 6.6 kHz. La tarjeta emplea sin embargo una velocidad de 8.77 kHz, con lo que se tiene que la banda de protección es de 2.17 kHz de sobremuestreo suficiente para eliminar grandemente los efectos de la frecuencia interferencia.

El diagrama de la figura 7.5 muestra la forma en que se lleva a cabo este proceso.

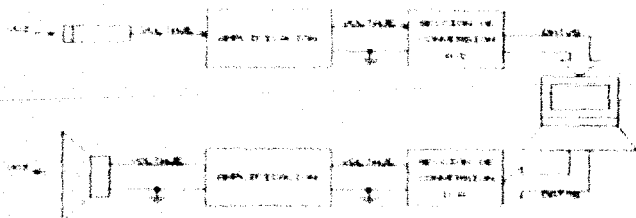


FIGURA 7.5

Para grabar la voz en la computadora, se convierten primero las ondas sonoras en voltaje mediante el empleo de un micrófono, el voltaje obtenido se amplifica, digitaliza y guarda en la memoria de la PC en forma de valores binarios para reproducir la señal. Los datos binarios son transferidos al convertidor D/A cuya señal de salida es amplificada y convertida nuevamente en ondas sonoras a través de un altavoz.

7.5 RECONOCIMIENTO DE VOZ

Cuando una señal de voz se encuentra almacenada en memoria a parte de reproducirla es también posible reconocer las palabras que fueron introducidas. Esto puede realizarse mediante métodos de análisis espectral, inteligencia artificial, redes neuronales etc.

El reconocimiento de voz resulta de gran ayuda por ejemplo para gente discapacitada, ya que con una computadora es posible encender o apagar equipos eléctricos y electrónicos simplemente mencionando su nombre.

Lo que se requiere para esta aplicación es convertir la voz en voltaje mediante un micrófono, la señal obtenida se amplifica, digitaliza y guarda en memoria donde puede ser analizada (figura 7.5). Una vez que se ha reconocido la palabra almacenada se activa alguno de los 8 bits correspondientes en la memoria de salida digital de la tarjeta correspondiente al aparato eléctrico o electrónico que se ha mencionado, el cual puede ser

esencialmente o apagado por acción de un relevo que amplifica rápidamente al computador.

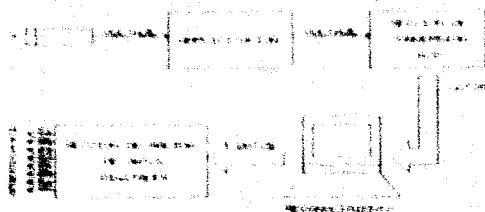


FIGURA 7 a

Como ya se ha dicho existen varios métodos espaciales en el reconocimiento de voz. El análisis espectral es uno de los más utilizados. Sin embargo para una PC la descomposición de señales en sus múltiples frecuencias y su interpretación es una tarea que puede tardar hasta varios minutos. Existen el empleo de redes neuronales artificiales además de eso lento, consume gran parte de la capacidad de memoria técnica que son cientos las neuronas que deben ser simuladas para este tipo de aplicación.

Un método sencillo y que por lo general da buenos resultados es pedir la repetición de las palabras que han sido escuchadas. De esta forma la PC puede reconocer fácil y rápidamente palabras como TELEVISION, RADIO, LAMPARA o COMPUTADORA.

Los ejemplos que se acaban de mencionar son tan sólo unas cuantas de las muchas aplicaciones que se le pueden dar a la tarjeta de digitalización, la cual como se ha podido ver, puede ser empleada tanto en medicina, como en control, medición y procesamiento de voz.

CONCLUSIONES

CONCLUSIONES

La intención de este proyecto ha sido demostrar que con una computadora personal es posible desarrollar un sistema de graficación en tiempo real de señales cardíacas.

Para lograr esto se diseñó y elaboró una tarjeta que permite el amuestreo de señales analógicas en la memoria de la computadora. Previamente, se realizó un análisis encargado de determinar ante todo el teorema de Nyquist, la componente máxima de frecuencia que debe tener una señal para que ésta pueda ser digitalizada y reproducida sin pérdida de información. Mediante dicho análisis pudo conocerse que en teoría es posible la digitalización de voltajes analógicos con componentes de frecuencia inferiores a 1700 Hz. Sin embargo, el proceso de muestreo realizado a esta velocidad requiere del empleo de un filtro ideal "pass bases" que es algo imposible de conseguir en la práctica, por tal motivo, se tomó como límite de frecuencia el valor de 950 Hz para tener de esta forma, una ganancia de protección de 1771 Hz, suficiente para reproducir con eficiencia las señales analógicas digitalizadas. Bajo esta condición.

A parte de elaborar este circuito, se realizó un programa encargado de controlar las acciones del hardware y de desplegar en pantalla la forma de la señal cardíaca conectada a la entrada de la tarjeta, así como su frecuencia en pulsos por minuto. Este programa llamado "Sistema de Electrocardiografía" se

1951

The first part of the report deals with the general situation in the country. It is a very interesting and well-written study. The author has done a great deal of research and has gathered a wealth of material. The report is well organized and easy to read. It is a valuable contribution to the study of the country.

1952

The second part of the report deals with the economic situation. It is a very interesting and well-written study. The author has done a great deal of research and has gathered a wealth of material. The report is well organized and easy to read. It is a valuable contribution to the study of the country.

1953

The third part of the report deals with the social situation. It is a very interesting and well-written study. The author has done a great deal of research and has gathered a wealth of material. The report is well organized and easy to read. It is a valuable contribution to the study of the country.

Las pruebas realizadas al programa consistieron en digitalizar y graficar en la pantalla de la computadora una señal cardiaca real y compararla con la registrada por un equipo de electrocardiografía comercial. Los resultados obtenidos mostraron que las señales desplegadas en el monitor de la PC corresponde efectivamente a las voltajes presentes a la entrada de la tarjeta.

Los resultados de estas pruebas confirmaron que tanto el circuito como el programa diseñados en este proyecto funcionan eficientemente en la digitalización y despliegue de señales cardiacas reales en por ello que estamos en condiciones de afirmar que los objetivos planteados al principio de este proyecto han sido cumplidos en su totalidad.

APENDICES

APENDICE A

CONVERTIDOR DE DATOS ANALÓGICOS A DIGITALES ADC0804

A.1 DESCRIPCIÓN GENERAL

Los circuitos ADC0801, ADC0802, ADC0804, ADC0805 y ADC0805 constituyen una familia de convertidores de aproximaciones sucesivas. Su diseño permite la operación con procesadores tales como el NS3200, el 8080, el Z80 dando la apariencia de ser simples localidades de memoria o puertos. Su conexión no requiere de algún tipo de lógica adicional.

Sus entradas diferenciales permiten incrementar la razón de rechazo al ruido, aun cuando su línea de referencia (V_{ref}) puede ser ajustada para codificar cualquier rango de voltaje por pequeño que este sea.

A.2. CARACTERÍSTICAS

- Es compatible con procesadores 8080 y derivados, sin necesidad de emplear lógica de control adicional.
- Es compatible con circuitos de lógica TTL y CMOS.
- Cuenta con entradas analógicas diferenciales.
- Cuenta con un generador de reloj integrado.
- Acepta señales analógicas que varíen entre 0 y 5V, requiriendo para ello de un voltaje de polarización único (+5V).

- No requiere de ajuste a cero.
- Resolución de 8 Bits.
- Tíemp. de conversión típico de 100 μ s.

A.3 FUNCIONAMIENTO

La serie AD7501 emplea un algoritmo de aproximaciones sucesivas para generar un código de ocho bits a la diferencia de los voltajes de entrada (V_{IN+} vs V_{IN-}); la siguiente figura muestra un diagrama funcional del circuito.

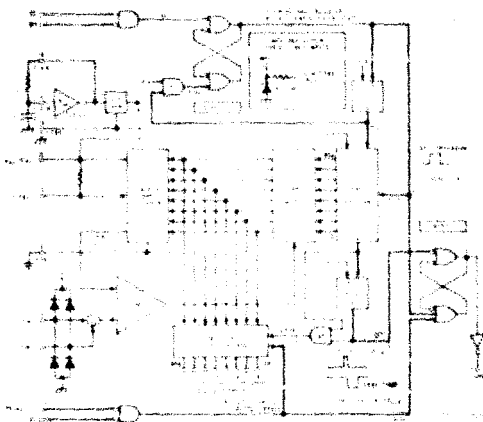


FIGURA A.1

La conversión comienza cuando las líneas CS y WR presentan simultáneamente un voltaje bajo, esto ocasiona que el *flip flop* de reloj (clock flip) sea fijado con un "1" que a su vez borra el contenido del registro de corriente (current register), limpia el *flip flop* de interrupción (int. flip) y fija con un "1" tanto el contenido de un *flip flop* tipo D (D flip) como la entrada del registro de corriente. En ese momento, el bit más significativo de este bit (MSB) es ajustado en "1" mientras que los otros permanecen con un valor de "0", esto produce un voltaje V_{out} a la salida del DAC (diferencia de voltaje correspondiente al valor del MSB). Ahora bien, si V_{out} es mayor que V_{in} o $V_{in} - V_{out}$, en el comparador se obtendrá un "0" que será almacenado en el bit más significativo del registro de expresiones sucesivas (SAR). En caso contrario el MSB del SAR se ajustará con un "1". Un pulso de reloj después, el siguiente bit del registro de corriente es fijado generando un nuevo valor V_{out} que se compara nuevamente por la diferencia de los voltajes de entrada, el resultado obtenido es almacenado ahora en el segundo bit más significativo del SAR. Este proceso se repite para cada uno de los bits del registro de corriente.

Cuando el último bit ha sido fijado (ochos pulsos de reloj después), en la salida Q del registro de corriente aparece un "1", el cual habilita mediante un *flip flop* tipo D (data flip) y una compuerta AND, las entradas del latch de salida para que éste almacene el resultado de la conversión contenido en el SAR. Al mismo tiempo, el latch habilita el fijamiento de

interfazión TTL para que a su salida se obtenga un "0" indicando que la conversión ha sido terminada. Simplemente para transferir el resultado al resto de salida se debe activar los pines de las señales CS y RL.

A.4 ENTRADAS DIGITALES DE CONTROL

Las líneas de control CS, WR y RL pueden ser activadas con niveles de voltaje compatibles con lógica TTL y MOS. Estas señales se habilitan por voltajes bajos para permitir su conexión con el bus de control de algún microprocesador sin necesidad de emplear lógica adicional.

A.5 ENTRADAS DIFERENCIALES DE VOLTAJES ANALÓGICOS

Esta característica del convertidor, permite una gran flexibilidad en el desarrollo de aplicaciones. La línea V_{IN+} puede ser empleada para restar e instantáneamente un valor constante de voltaje a la terminal V_{IN-}. Además, este tipo de entradas permite la reducción del ruido de auto-corriente.

A.6 AJUSTE DE ESCALA

La escala AD7511 ha sido diseñada para ajustar su referencia a 1.25 o a cualquier otro valor de voltaje aplicado a su entrada V_{REF}. 2. Como ejemplo, considérase una señal analógica

con variaciones de 0.5 a 3.5 volts. Si se aplican 0.5 V a la entrada **Vref** para absorber el offset y si el voltaje aplicado a la línea **Vref 2** se hace igual a 1.5 V, entonces el convertidor podrá codificar la señal analógica asociando el valor de 00000000 a los 0.5 V y 11111111 a los 3.5V.

APENDICE B

CONVERTIDOR DE DATOS DIGITALES A ANALÓGICOS (DAC830)

B1 DESCRIPCIÓN GENERAL

El DAC830 es un convertidor multiplicativo de 8 Bits elaborado con tecnología CMOS está diseñado para convertirse directamente con el 80C8, 80C8, y otros microprocesadores populares. Su estructura interna proporciona compatibilidad con lógica TTL. Sus dos pines de entrada permiten obtener el voltaje de salida correspondiente a un byte mientras que el siguiente dato es almacenado, lo cual permite la actualización simultánea de cualquier número de DATA.

B2 CARACTERÍSTICAS

- Puede ser configurado para trabajar con dos, uno o ningún pin de entrada.
- Permite la conexión directa con cualquier microprocesador popular.
- Puede ser configurado como convertidor de voltaje o de corriente.
- Sus entradas de datos y de control son compatibles con niveles de voltaje TTL.

- Opera en modo *stand alone* (sin procesador)
- Cuenta con un tiempo de conversión aproximado de 1µs.

B.3 DIAGRAMA DE TERMINALES

(Para su configuración como convertidor de datos digitales a corriente.)

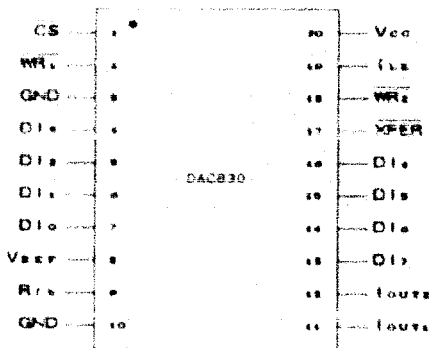


FIGURA B.1

B.4 DESCRIPCIÓN DE SEÑALES

- **WR1**: Almacena los bits que provienen del bus de datos en la memoria *16* bits de entrada.
- **CS = LE**: Habilita la línea de escritura **WR1**.
- **WR2**: Almacena la ecitación en la segunda memoria *latch* iniciándose con esta una operación de conversión de datos.

- **VREF**: Establece la línea **WR2**
- **NO DP**: Son las líneas de entrada para el valor digital
- **VREF**: Fija los valores de corriente suministrados por las salidas **OUT1** e **OUT2**
- **OUT1** e **OUT2**: Son las salidas de corriente del convertidor. Sus valores están determinados por las expresiones:

$$I_{OUT1} = \frac{V_{REF}}{100\Omega} \times \frac{ENTRADA\ DIGITAL}{255}$$

$$I_{OUT2} = \frac{V_{REF}}{100\Omega} \times \frac{255 - ENTRADA\ DIGITAL}{255}$$

- **Rfb**: Es una resistencia interna que puede emplearse como retroalimentación para un amplificador operacional externo, el cual convierte la salida de corriente del DAC08D a una señal de voltaje (figura 8.3).



FIGURA 8.3

• **Vcc:** Es la entrada de polarización del circuito. Su voltaje de entrada puede variar desde +5 a +15V, siendo este último valor el que permite su operación óptica.

8.5 FUNCIONAMIENTO

Para su funcionamiento, el DAC 830 emplea dos memorias latch de entrada, un convertidor D/A, una compuerta AND y 2 compuertas OR (compuertas AND con entradas negadas). Su diagrama funcional es el siguiente.

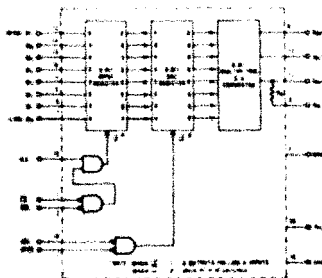


FIGURA 8

Si las señales \overline{CS} y \overline{WR} son activadas con un voltaje bajo y la línea \overline{LE} presenta un valor alto, entonces a la salida de la compuerta AND aparecerá un "0" que habilitará al latch de entrada (input register), en ese momento los datos que se

encuentran a la entrada del DAC830, serán transferidos al ítem de conversión (DAC Register). Posteriormente, cuando las líneas ~~WR1~~ y ~~XFEN~~ presenten simultáneamente un voltaje bajo, los datos contenidos en la entrada del segundo registro pasarán al convertidor iniciándose en ese instante el cálculo del valor analógico de corriente o voltaje correspondiente al dato digital de entrada.

APENDICE C

TEMPORIZADOR LM555

C.1 DESCRIPCIÓN GENERAL

El temporizador LM555 es un circuito altamente estable capaz de producir retardos exactos de tiempo y oscilaciones. Se proporcionan terminales adicionales para disparo o restablecimiento. En el modo de operación de retardo de tiempo, se controla con precisión mediante un capacitor y resistores externos. Para operación oscilatoria en modo astable, la frecuencia de la señal así como el ciclo de trabajo pueden ser controlados mediante el empleo de dos resistores y un capacitor. La estructura de su salida puede impulsar circuitos TTL.

C.2 CARACTERÍSTICAS

- Opera tanto en modo astable como monostable
- Ciclo de trabajo ajustable
- Es compatible con circuitos TTL
- Estabilidad de temperatura de $\pm 0.05\%$ por $^{\circ}\text{C}$

C.3 APLICACIONES

- Temporizador de precisión
- Generación de pulsos

- Generador de retardos de tiempo
- Detector de púlsos perdidos

C.4 ESTRUCTURA INTERNA

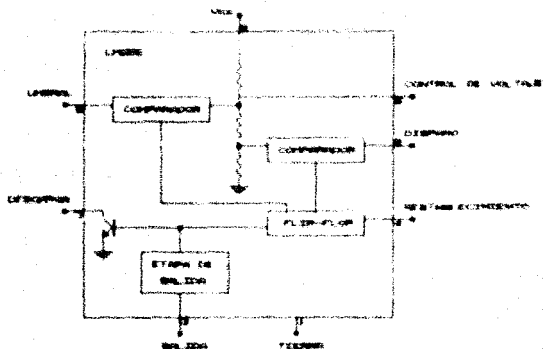


FIGURA C.4

C.5 FUNCIONAMIENTO COMO MONOESTABLE

En este modo de operación, el circuito funciona como un disparador (ver figura C.2). La figura C.2 muestra la configuración.

El capacitor C_1 es inicialmente descargado por el transistor interno del LM555. Cuando un pulso negativo menor a $1/3$

de Vcc es aplicado a través de la entrada 2, el flip-flop será ajustado con un "1" provocando por un lado, un corto circuito entre el capacitor C y por otro, la presencia de un voltaje alto por la salida del circuito (terminal 3). En ese momento, el voltaje del capacitor se incrementará exponencialmente por un periodo $T = 1.1 R_1 C$. Al término de este tiempo, su voltaje será igual a $2/3$ de Vcc. En consecuencia, entonces restablece el flip-flop cuyo contenido hace que se descargue el capacitor y que en la salida del circuito se presenta un "0".

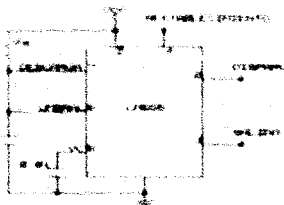


FIGURA C 8

Durante el tiempo en que la salida se encuentra en alto, la aplicación de pulsos adicionales no tendrá efecto, sin embargo el circuito puede ser restablecido mediante la aplicación de un pulso negativo a través de la terminal 4, de esta forma la salida del 555 permanecerá en bajo, hasta que un nuevo pulso vuelva a ser aplicado por la entrada 2.

C6 FUNCIONAMIENTO COMO AESTABLE E (Oscilador)

Si el circuito es configurado como se muestra en la figura C.6, el AIC puede oscilar a su mismo voltaje de funcionamiento libremente como multivibrador. En esta configuración, el capacitor externo C es cargado a través de Ra + Rb y descargado por Rb.

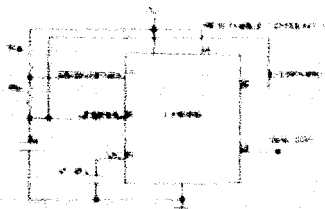


FIGURA C.6

En este modo, la frecuencia de oscilación está dada por

$$f = \frac{1}{T} = \frac{1}{R_a + 2R_b C}$$

Siendo el ciclo de trabajo igual a

$$D = \frac{R_b}{R_a + 2R_b}$$

Cabe señalar que tanto en la configuración monostable como en la astable la duración de los pulsos de salida es independiente del voltaje de polarización.

ANEXOS

COEFICIENTE DE CORRELACION LINEAL

El coeficiente de correlación r se define como una medida de la asociación lineal que existe entre las variables X y Y . Para un total de n valores tiene la siguiente expresión:

$$r = \frac{\sum_{i=1}^n x_i y_i - \frac{(\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n}}{\left[\sum_{i=1}^n x_i^2 - \frac{(\sum_{i=1}^n x_i)^2}{n} \right]^{1/2} \left[\sum_{i=1}^n y_i^2 - \frac{(\sum_{i=1}^n y_i)^2}{n} \right]^{1/2}}$$

Este parámetro se encuentra en el intervalo $-1 \leq r \leq 1$, en donde un valor $r = 1$ indica una relación lineal negativa perfecta entre X y Y , mientras que un valor $r = -1$ señalará una asociación lineal positiva perfecta. Si $r = 0$, entonces no existe ninguna relación lineal entre X y Y .

APENDICE E

DIAGRAMAS DE TERMINALES Y TABELAS DE VERDAD DE LOS CIRCUITOS CON BAJA Y MEDIANA ESCALA DE INTEGRACION

EL CIRCUITO 74LS00

Cuatro compuertas NAND de 2 entradas

Diagrama de terminales

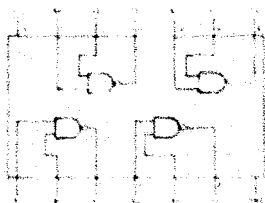


FIGURA E 1

Tabla de verdad:

$$Y = AB$$

ENTRADAS		SALIDA
A	B	Y
L	L	M
L	H	H
H	L	H
H	H	L

E2 CIRCUITO 74LS138

Decodificador de 3 a 8 líneas

Diagrama de terminales

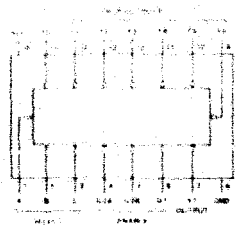


FIGURA 6.2

Tabla de verdad

HABILITACION		SELECCION			SALIDAS							
G1	G2	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
•	H	•	•	•	H	H	H	H	H	H	H	H
L	•	•	•	•	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	L	L	L	H	H	H	H	L	L	H	H
H	L	L	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

E3 CIRCUITO 74LS30

Compuerta NAND de 8 entradas

Diagrama de terminales

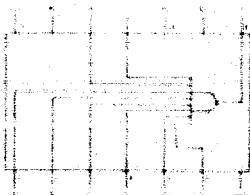


FIGURA E.9

Tabla de verdad

$Y = \overline{A \cdot B \cdot C \cdot D \cdot E \cdot F \cdot G \cdot H}$

ENTRADAS	SALIDAS
A - H	Y
H EN TODAS LAS ENTRADAS	L
L EN UNA O MAS ENTRADAS	H

E.4. CIRCUITO 74LS27

Tres compuertas nor de 3 entradas

Diagrama de terminales

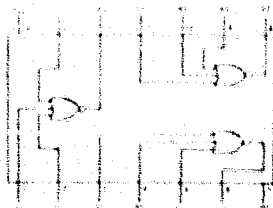


FIGURA E.4

Tabla de verdad:

$$Y = A \cdot B \cdot C$$

ENTRADAS			SALIDAS
A	B	C	Y
L	L	L	H
L	L	H	L
L	H	L	L
L	H	H	L
H	L	L	L
H	L	H	L
H	H	L	L
H	H	H	L

ES. CIRCUITO 74LS132

Cuatro compuertas NAND de 2 entradas tipo
schmitt trigger

Diagrama de Interconexión

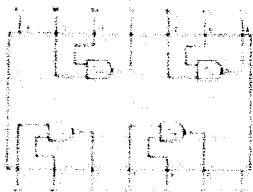


FIGURA 5.5

Tabla de verdad:

Y = AB

ENTRADAS		SALIDA
A	B	Y
L	L	H
L	H	H
H	L	H
H	H	L

E 6 CIRCUITO 74LS158

Cuatro multiplexores de dos líneas a una

Diagrama de terminales:

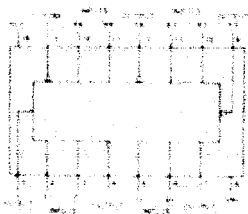


FIGURA E 6

Tabla de verdad

ENTRADAS				SALIDA
HABILITACION (Strobe)	SELECCION (select)	A	B	Y
H	*	*	*	H
L	L	L	*	H
L	L	H	*	L
L	H	*	L	H
L	H	*	H	L

EL CIRCUITO 4027

Dos flip-flops tipo JK

Diagrama de terminales

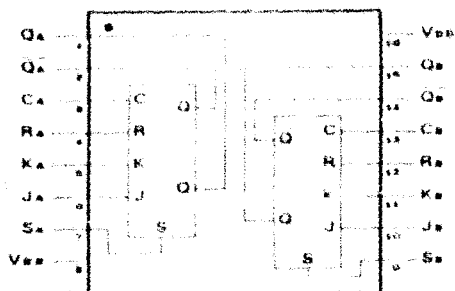


FIGURA 8.7

Tabla de verdad

RELOJ	ENTRADAS					SALIDAS	
	J	K	S	R	Qn	Qn+1	Qn+1
	H	H	L	L	L	H	L
	H	L	L	L	L	H	L
	L	H	L	L	L	L	H
	H	H	L	L	H	L	H
	H	H	L	L	Qn	Qn	Qn
	H	H	L	L	H	L	L
	H	H	L	H	L	L	H
	H	H	H	H	L	H	H

ES CIRCUITO 4040B

Contador binario de 12 bits

Diagrama de terminales

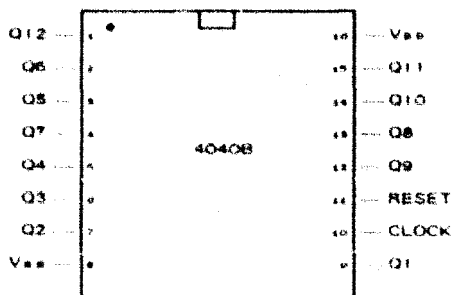


FIGURA 8.9

Tabla de verdad

RELOJ (CLOCK)	RESTABLECIMIENTO (reset)	ESTADO DE LA SALIDA
	L	SIN CAMBIO
	L	AVANZA AL SIGUIENTE ESTADO
	H	TODAS LAS SALIDAS EN BAJO

BIBLIOGRAFIA

BIBLIOGRAFIA

- 1 Adams, Lew
High Performance Graphics in 11.
USA, Windcrest.
- 2 Canaves, George
Análisis y Estadística.
México, Editorial McGraw Hill, 1986
- 3 Cristofoli, Frederico y Coughlin, Robert
Circuitos Integrados Lineales y Amplificadores Operacionales,
2da. Edición, México, Editorial Prentice Hall, 1987
- 4 Gerez, Víctor
Desarrollo y Administración de Empresas de Computadora.
México, Editorial C E T S A , 1984
- 5 Guyton, Arthur
Fisiología humana.
México, Editorial Interamericana 1979
- 6 Haykin, Simon.
Principios de Reconstrucción.
2da. Edición, México, Editorial Interamericana, 1985
- 7 Holman, Jack
Métodos Experimentales para Ingenieros.
4ta. Edición, México, Editorial McGraw Hill, 1987

8. Kelly, Stan
Microelectronics
USA, 1978, 1988
9. Mano, Morris
Digital Design
Mexico, Editorial Prentice Hall, 1984
10. Nise, Forrest
Engineering Software II
USA, Radio Shack, 1982
11. MOTOROLA
CMOS Logic Data
2nd Edition, USA, 1988
12. National Semiconductor Corporation
Linear Databook 2
USA, 1988
13. National Semiconductor Corporation
LSI CMOS Logic Data
USA, 1987
14. RCA Design
Materiales de Referencia para el Diseñador
USA, 1986
15. Ratti, Ger. y Bernardi, Christian
Arithmetic
Italia, Editorial Pisa & Zanichelli, 1984

16. Cottrill, Stella
Transmisión de Información, Modulación y Ruido.
3ra Edición, México, Editorial McGraw Hill, 1985.
17. Ince, Ronald
Sistemas Digitales.
3ra Edición, México, Editorial Prentice Hall, 1989.
18. Takahara, Roger
Principios de Sistemas.
México, Editorial McGraw Hill, Serie Schaum, 1988.