

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades
Departamento de Ingeniería Electrónica



**SISTEMA DIGITAL PARA
PROCESAMIENTO Y GENERACION
DE SEÑALES ANALOGICAS**

JUAN CARLOS MAYORGA JUAREZ



Trabajo de graduación presentado para optar
al grado académico de

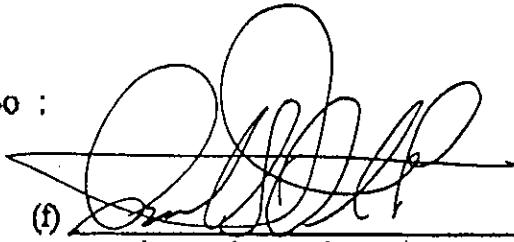
Licenciado en Ingeniería Electrónica

Guatemala

1995

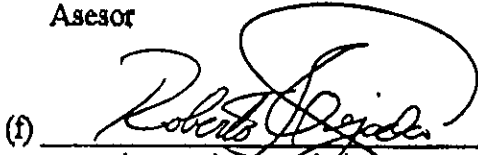
**SISTEMA DIGITAL PARA
PROCESAMIENTO Y GENERACION
DE SEÑALES ANALOGICAS**

Vo. Bo :



(f)

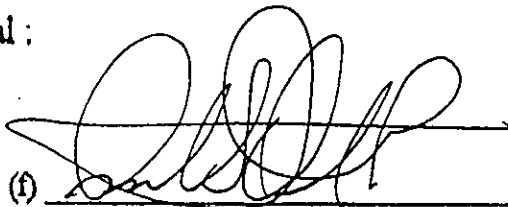
Ingeniero Robert Duke
Asesor



(f)

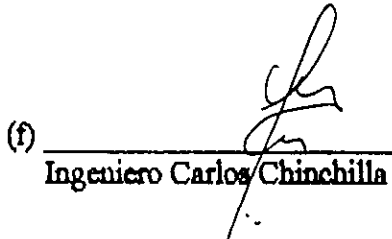
Ingeniero Roberto Fejada
Director del Departamento de Ingeniería Electrónica

Tribunal :



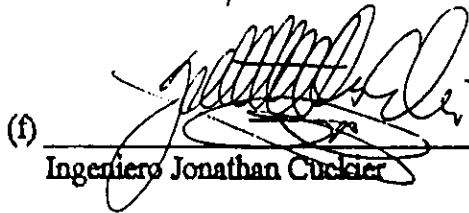
(f)

Ingeniero Robert Duke



(f)

Ingeniero Carlos Chinchilla



(f)

Ingeniero Jonathan Cuelier

A Dios, por darme fuerza, fe y esperanza
A mis padres, Carlos y Gisela
A mis hermanos, Ana Cristina, Luz Elena y Pablo Enrique
A mi novia, Vivian Scarlet
A mis amigos

INDICE

	Página
Introducción	1
I. LA UNIDAD DE CONTROL DEL SDPG	3
A. El CPU de la Unidad de Control	3
B. Interface del Z80 con los dispositivos de Memoria	3
1. Señales de la Interface	3
2. El Decodificador de Memoria	4
3. Interface con la ROM	5
4. Interface con la RAM	5
C. Interface del Z80 con los dispositivos de Entrada/Salida	6
1. Señales de la Interface	6
2. El Decodificador de Puertos	7
3. Interface con el Z80PIO	8
4. Interface con el Z80CTC	9
5. Interface con el Z80DART	10
6. El Generador de Reloj Maestro	12
II. LA UNIDAD DE GENERACION DE SEÑALES ANALOGICAS	13
A. Descripción General de la UGSA	13
B. Circuiteria de Conversión Digital a Analógico	13
C. Circuito Generador de Señales Analógicas	15
D. Circuito Selector de Señal	16

E.	Circuitería de Selección de Amplitud	17
III.	LA UNIDAD DE PROCESAMIENTO DE SEÑALES ANALÓGICAS	19
A.	Descripción General	19
B.	El Circuito de Entrada	20
C.	Circuito de Selección de Entrada	21
D.	Circuito de Selección de Ganancia	22
E.	Circuito Acondicionador de Amplitud	23
F.	Circuito Medidor de Frecuencia	24
G.	Convertidores Analógico-Digitales	24
IV.	UNIDAD DE INTERFACE CON LA PC	26
A.	Descripción General	26
B.	Señales utilizadas para la interface	26
V.	EL PROGRAMA DE OPERACION DEL SDPG	27
A.	Rutina Principal del Programa de Operación	28
B.	Rutina para el muestreo de una entrada	30
C.	Rutina para medición de la frecuencia	32
D.	Rutina para medición de las entradas 1 y 2	32
E.	Rutina para la transmisión de datos a la PC	33
F.	Rutina para la transmisión de la frecuencia a la PC	33

VI.	PROGRAMA DE USUARIO DEL SDPG	34
A.	Funciones que realiza el Programa de Usuario	34
B.	Requerimientos del Programa de Usuario	35
C.	Instalación del Programa de Usuario	35
D.	Ejecución del Programa de Usuario	36
E.	Opciones disponibles al usuario	36
1.	Almacenamiento en Disco	37
2.	Recuperar Datos del Disco	38
3.	Impresión de Datos	38
4.	Modificar la Forma de Onda de la Salida 1	39
5.	Modificar la Amplitud de la Salida 1	39
6.	Modificar la Frecuencia de la Salida 1	40
7.	Modificar la Forma de Onda de la Salida 2	40
8.	Modificar la Amplitud de la Salida 2	41
9.	Modificar la Frecuencia de la Salida 2	41
10.	Solicitar Muestras de Ambas Entradas	42
11.	Modificar la Ganancia de la Entrada 1	42
12.	Modificar la Ganancia de la Entrada 2	43
VII.	CONCLUSIONES	44
VIII.	BIBLIOGRAFIA	47

APENDICES

- A. Diagramas de bloques de las Unidades del SDPG
- B. Diagramas de los circuitos del SDPG
- C. Listado del Programa de Operación del SDPG
- D. Diagramas de flujo del Programa de Operación
- E. Listado del Programa de Usuario del SDPG
- F. Pruebas de Operación del SDPG

Introducción

El Sistema Digital para Procesamiento y Generación de Señales Analógicas (SDPG) es una microcomputadora diseñada para la medición y generación de señales analógicas.

El SDPG se comunica con una computadora personal (PC) por medio de una interfaz serial, con objeto de enviarle los resultados de las mediciones efectuadas y para recibir comandos que modifican los parámetros de su funcionamiento.

El SDPG está comprendido por varias unidades especializadas como lo son la Unidad de Control, la Unidad de Procesamiento de Señales Analógicas, la Unidad de Generación de Señales Analógicas y la Unidad de Interfaz con la PC .

La operación en conjunto de todas las unidades del SDPG, permite que éste realice varias funciones de procesamiento de señales. Estas comprenden procesar dos entradas analógicas para obtener su representación en formato digital. Determinar las características de cada entrada; generar independientemente dos salidas analógicas básicas las cuales pueden ser senoidal, triangular o cuadrada. Controlar las características de las salidas como lo son la amplitud, la forma de onda y la frecuencia. Enviar los resultados de las mediciones a la PC para que ésta realice el tratamiento adecuado y recibir comandos de control de la PC que indican cómo debe modificarse la operación del sistema.

En el presente trabajo, se examinan cada uno de los componentes y los programas que conforman el SDPG. En el capítulo 1 se describe el microprocesador y sus circuitos periféricos, utilizados para implementar la unidad de control del SDPG. En el capítulo 2, se describen los circuitos utilizados para la generación de dos salidas analógicas de amplitud, frecuencia y forma de onda controladas desde la PC. En el capítulo 3, se describen los circuitos utilizados para procesar las dos entradas del SDPG en lo referente a muestreo y medición de frecuencia. En el capítulo 4, se describen los circuitos utilizados para transferir información con la PC. En el capítulo 5, se describen las rutinas en ensamblador, que permiten la operación de cada una de las unidades del SDPG. Finalmente, en el capítulo 6, se describe el programa que permite al usuario controlar la operación del SDPG desde la PC y las opciones que se tienen disponibles. En los apéndices, se incluyen los diagramas de los circuitos de las unidades del SDPG, el listado de los programas, y la impresión de las pruebas de operación del proyecto.

I. La Unidad de Control del SDPG

La Unidad de Control del SDPG, es la encargada de coordinar el funcionamiento de todas las unidades que conforman el sistema. Esta comprende el microprocesador Z80, la circuitería de mapeo de memoria y puertos, los dispositivos de memoria, los dispositivos de entrada/salida y el generador de reloj maestro .

A. *El CPU de la Unidad de Control:*

El CPU de la Unidad de Control es el microprocesador Z80A (denominado simplemente Z80) de 8 bits que opera a una frecuencia de reloj de 4MHz.

B. *Interface del Z80 con los dispositivos de Memoria:*

I. Señales de la Interface

Realizar la interfaz del Z80, con un dispositivo de memoria, es establecer la conexión eléctrica adecuada entre ambos. Esta conexión, se realiza por medio de un grupo de señales del bus del sistema, que permiten al CPU acceder los dispositivos de memoria cuando sea necesario. Las señales involucradas en esta interfaz son: el bus de direcciones, el bus de datos, la señal de acceso de memoria (MREQ), la señal de escritura (WR), y la señal de lectura (RD) .

2. El Decodificador de Memoria

El Decodificador de Memoria, es el circuito que ejecuta el mapeo de memoria y permite que el CPU seleccione los distintos dispositivos de memoria.

El Decodificador de Memoria, consiste en un decodificador de 3 a 8 (74LS138) habilitado cuando el CPU realiza un acceso a memoria, en los 32K de memoria disponibles (MREQ y A15 en estado lógico 0). La memoria que puede accederse (32K) está dividida en 8 segmentos de 4K, correspondiendo cada uno a una salida (Y0-Y7) del Decodificador de Memoria. La combinación de los estados lógicos de las líneas A14, A13, y A12, determinan el segmento que se accesa. Las líneas restantes de A11 a A0 determinan la posición de memoria que se accesa en el dispositivo seleccionado. La tabla 1, muestra el mapeo de memoria implementado en el SDPG.

Tabla 1. Mapeo de Memoria del SDPG

SALIDA DE L. DDM	SEGMENTO DE MEMORIA	USO
Y0	0000H-0FFFFH	ROM
Y1	1000H-1FFFFH	ROM
Y2	2000H-2FFFFH	
Y3	3000H-3FFFFH	RAM
Y4	4000H-4FFFFH	
Y5	5000H-5FFFFH	
Y6	6000H-6FFFFH	
Y7	7000H-7FFFFH	

3. Interface con la ROM

Una ROM almacena información en forma no volátil y ésta no puede modificarse una vez se ha grabado. La ROM de la Unidad de Control es una EPROM del tipo 2764 (8K x 8bits) y tiene almacenado el programa en lenguaje ensamblador (Programa de Operación) que controla el funcionamiento del SDPG.

Para que el Z80 pudiera leer los datos almacenados en la ROM, sus pines se conectaron de la siguiente forma (referirse al diagrama I del Apéndice B):

- las líneas de dirección A0-A12 a las líneas respectivas del bus de direcciones del Z80;
- las líneas de datos O0-O7 a las líneas respectivas del bus de datos del Z80;
- el pin de habilitación de salida OE' se conectó a tierra;
- el pin de selección de la pastilla a la salida respectiva del Decodificador de Memoria.

La EPROM, es accesada cuando una de las salidas Y0 o Y1 del Decodificador de Memoria, está en estado lógico 0 y opera en modo de lectura. Si la EPROM no es seleccionada, está en modo de alta impedancia.

4. Interface con la RAM

Una RAM almacena información en forma volátil y, además, ésta puede ser modificada en cualquier instante. La Unidad de Control utiliza una RAM estática del tipo 6116 (2K x 8bits). Esta permite al CPU almacenar las variables definidas en el Programa de Operación del SPDG y disponer de una pila (stack) cuando existan llamadas a subrutinas.

Para que el CPU pudiera acceder la RAM , los pines de la misma se conectaron de la siguiente forma (referirse al diagrama 1 Apéndice B):

- las líneas de dirección A0-A10 a las líneas respectivas del bus de direcciones del Z80;
- las líneas de datos D0-D7 a las líneas respectivas del bus de datos del Z80;
- el pin de habilitación de salida OE se conectó a RD;
- el pin de selección de la pastilla CE se conecto a la salida respectiva del DDM;.
- el pin de escritura WE se conectó a la señal WR.

La RAM es accesada cuando la salida Y3 del DDM está en el estado lógico 0. Esta RAM, es seleccionada para lectura cuando la señal RD está en estado lógico 0 y para escritura cuando la señal WR está en estado lógico 0.

C. *Interface del Z80 con los dispositivos de Entrada/Salida*

I. Señales de la Interface:

Realizar la interface del Z80 con un dispositivo de Entrada/Salida es establecer la conexión eléctrica adecuada entre ambos. Esta conexión, se realiza por medio de un grupo de señales del bus del sistema que permiten al CPU acceder los dispositivos de Entrada/Salida cuando sea necesario. Las señales involucradas en esta interface son: el bus de direcciones, el bus de datos, la señal de acceso de puertos (IORQ) , la señal de escritura (WR) , y la señal de lectura (RD) .

2- El Decodificador de Puertos

El Decodificador de Puertos es el circuito que ejecuta el mapeo de puertos, y permite que el CPU seleccione cada uno de los dispositivos de Entrada/Salida del sistema.

El Decodificador de Puertos, consiste en un decodificador de 3 a 8 (74LS138) habilitado cuando el CPU realiza un acceso a puertos (IORQ en estado lógico 0). El Decodificador de Puertos, permite acceder 32 direcciones de Entrada/Salida de las 256 disponibles en el rango 00H-1FH, las cuales se dividen en 8 grupos de 4 direcciones por dispositivo accedido. Cada grupo corresponde a una de las salidas (Y0-Y7) del Decodificador de Puertos. Las líneas de direcciones A5-A2, determinan el grupo de direcciones de Entrada/Salida que se habilita y las líneas A1-A0 determinan la dirección o puerto que se accesa. El mapeo de puertos obtenido se muestra en la tabla 2.

Tabla 2. Mapeo de Puertos del SPDG

SALIDA DEL DDP	DIRECCION E/S	DISPOSITIVO SELECCIONADO	PUERTOS ASOCIADOS
Y0	00,01,02,03H	PIO1	PAD1,PAC1,PBD1,PBC1
Y1	04,05,06,07H	PIO2	PAD2,PAC2,PBD2,PBC2
Y2	08,09,0A,0BH	PIO3	PAD3,PAC3,PBD3,PBC3
Y3	0C,0D,0E,0FH	PIO4	PAD4,PAC4,PBD4,PBC4
Y4	10,11,12,13H	CTC	CN0,CN1,CN2,CN3
Y5	14,15,16,17H	DART	CAC,CAD,CBC,CBD
Y6	18,19,1A,1BH		
Y7	1C,1D,1E,1FH		

3. Interface con el Z80PIO

El Z80PIO es un dispositivo de Entrada/Salida paralelo que sirve de interfase entre el microprocesador y los dispositivos periféricos que éste controle. El Z80PIO posee dos puertos A y B de 8 bits, los cuales pueden ser programados para operar en uno de cuatro modos posibles (entrada, salida, bidireccional y de control de bit). Internamente el PIO posee dos registros adicionales de control, que almacenan el modo de operación de cada uno de los puertos.

Para interfazar el CPU con el Z80PIO, los pines del mismo se conectaron de la siguiente forma (referirse al diagrama 2 del Apéndice B):

- las líneas M1, IORQ, RD a las líneas respectivas del Z80;
- las líneas A/B, C/D a las líneas A0 y A1 del bus de direcciones respectivas;
- las líneas D0-D7 a las líneas respectivas del Z80;
- la línea de habilitación CE a la salida correspondiente del Decodificador de Puertos.

Los modos de operación de los puertos del Z80PIO que se utilizaron, fueron el modo de salida (modo 0) y el modo de entrada (modo 1). La asignación de los modos de operación para cada uno de los puertos de los Z80PIO utilizados, se muestra en la tabla 3.

Tabla 3. Modos de Programacion de los Z80PIO

Z80PIO	PUERTO A	PUERTO B
1	SALIDA	SALIDA
2	SALIDA	SALIDA
3	SALIDA	ENTRADA
4	SALIDA	ENTRADA

La Unidad de Control, maneja las unidades de procesamiento y generación de señales por medio de estos cuatro Z80PIO. La tabla siguiente, muestra la función de cada uno de los puertos de los Z80PIO de la Unidad de Control.

Tabla 4 . Asignacion de funciones de los puertos de los PIOS

PIO	PUERTO A	PUERTO B
1	FRECUENCIA SALIDA 1	FRECUENCIA SALIDA 2
2	AMPLITUD/TIPO SEÑAL DE LAS SALIDAS	AMPLITUD/SELECCION DE ENTRADA
3	CONTROL DEL ADC PARA MUESTRAS	VALOR DE VOLTAJE EN FORMATO DIGITAL
4	CONTROL DEL ADC PARA FRECUENCIA	VALOR DE FRECUENCIA EN FORMATO DIGITAL

4. Interface con el Z80CTC

El Z80CTC es un dispositivo programable que posee cuatro circuitos contadores/temporizadores independientes denominados canales (CN0, CN1, CN2, CN3).

Cada canal, excepto el CN3, tiene una línea de salida denominada ZC/TO (Zero Count/Time Out) y una entrada denominada CLK/TRG (Clock/Trigger). El canal CN3 sólo tiene la señal de entrada, debido a limitaciones de pines de la pastilla. Cada canal es accesado como un puerto de Entrada/Salida. La combinación de los estados de las líneas C0 y C1 determinan el puerto (canal) que se accesa.

Para interfazar el CPU con el Z80CTC, los pines del mismo se conectaron de la siguiente forma (referirse al diagrama 3 del Apéndice B):

- las líneas M1, IORQ, RD a las líneas respectivas del Z80;
- las líneas C0, C1 a las líneas A0 y A1 del bus de direcciones respectivas;
- las líneas D0-D7 a las líneas respectivas del Z80;
- la línea de habilitación CE a la salida Y4 del Decodificador de Puertos.

El Z80CTC se programó para que todos sus canales operen en forma de temporizador. No se habilitan las interrupciones (acceso por polcado) y el factor de escala utilizado es 16.

El Z80CTC utilizado en el modo de temporizador, permite obtener señales con frecuencias derivadas del reloj del Z80. En el SDPG, el canal 0 (CN0) genera una señal de reloj de frecuencia de 19.2 kHz la cual es utilizada por la Unidad de Interface con la PC para derivar la velocidad en bits por segundo, necesaria para la comunicación serial con la PC. El canal 1 (CN1) genera una señal de 1200 Hz la cual se aplica directamente a un LED para indicar al usuario del SDPG que el circuito está en modo de operación normal. El valor de la frecuencia de salida del canal 1, (1200 Hz) se eligió por conveniencia para efectos visuales.

5. Interface con el Z80DART

El Z80DART es un dispositivo que comprende dos canales independientes asíncronos para comunicación serial. Cada uno de los canales consiste en un receptor y en un transmisor asíncronos con los que es posible el establecimiento de una comunicación full-duplex, con caracteres de 5 a 8 bits, bit de paridad (par/impar) opcional, y bits de parada de varios tipos (1, 1 1/2 y 2).

El Z80DART posee 3 registros internos de lectura (RR0-RR2) en los que están almacenados, el estado de operación del transmisor, del receptor y los errores que han ocurrido durante la comunicación. Los 6 registros internos de escritura (WR0-WR5) se utilizan para almacenar los valores seleccionados para la operación de cada canal.

Para interfazar el CPU con el Z80DART, los pines del mismo se conectaron de la siguiente forma (referirse al diagrama 3 del Apendice B):

- las líneas M1, IORQ, RD a las líneas respectivas del Z80;
- las líneas B/A, C/D a las líneas A0 y A1 del bus de direcciones respectivas;
- las líneas D0-D7 a las líneas respectivas del Z80;
- la línea de habilitación CE a la salida Y5 del Decodificador de Puertos .

La Unidad de Control, programa el canal A del Z80DART para que éste establezca con la PC una comunicación serial asíncrona, a una velocidad de transmisión de 9600bps, con una longitud de 8 bits por caracter, sin bit de paridad par, un bit de inicio y un bit de parada. El Z80DART es accesado por poleado (lectura de los registros RR0 a RR2) ya que no se habilitaron las interrupciones.

El Z80DART se utiliza para interfazar la Unidad de Control del SDPG con la PC. El Z80DART es el circuito responsable de la transmisión de los datos a la PC en el formato serial, así como de la recepción de datos de la PC y su conversión a formato paralelo.

6. El Generador de Reloj Maestro

El Generador de Reloj Maestro, comprende todos los circuitos necesarios para generar las señales de sincronización para el funcionamiento del SDPG.

El Generador de Reloj Maestro consiste en un oscilador de cristal de cuarzo y en una compuerta AND. El oscilador de cristal de cuarzo produce una señal TTL de 4.00MHz. La compuerta AND sirve de manejador para que la señal de reloj sea distribuida sin distorsión a todos los periféricos.

II. La Unidad de Generación de Señales Analógicas

A. *Descripción General de la UGSA*

La Unidad de Generación de Señales Analógicas es la unidad del SDPG que genera dos salidas analógicas independientes, de forma de onda, amplitud y frecuencia ajustables desde la PC.

Cada salida de la Unidad de Generación de Señales Analógicas, comprende los circuitos de conversión digital analógica, un circuito generador de funciones y un circuito selector de ganancia.

B. *Circuitería de Conversión Digital a Analógico*

Debido a que la Unidad de Control opera con señales digitales y a que las salidas de la Unidad de Generación de Señales Analógicas son señales analógicas, es indispensable establecer una conversión digital a analógica para permitir que la Unidad de Control pueda modificar las características de estas señales. Esta función, la realiza la Circuitería de Conversión Digital Analógica, la cual se muestra en el diagrama 4 del Apéndice B. La Circuitería de Conversión Digital Analógica de cada salida consiste en un convertidor digital analógico, en un convertidor de corriente a voltaje y en un circuito desplazador de nivel.

El convertidor digital analógico utilizado es el DAC0808 de 8 bits con una precisión de 0.19% y un tiempo de establecimiento de 150ns. El convertidor digital analógico, recibe una palabra de 8 bits del puerto respectivo del Z80PIO1 y lo convierte a un valor de corriente, el cual es convertido a voltaje por el circuito convertidor de corriente a voltaje.

El circuito convertidor de corriente a voltaje, consiste en un amplificador operacional en la configuración de amplificador inversor (referirse al diagrama 4 del Apéndice B). La resistencia de entrada, del amplificador operacional, es infinita, con lo que la corriente de salida del convertidor digital analógico, fluye en su totalidad, por la resistencia de retroalimentación R_{IVC} , generando un voltaje de salida dado por la siguiente expresión:

$$V_{OIVC} = R_{IVC} * (V_{REF}/R_{14}) * (A_1/2 + A_2/4 + A_3/8 + A_4/16 + A_5/32 + A_6/64 + A_7/128 + A_8/256)$$

donde

$A_N = '1'$ si la salida A_N del convertidor digital analógico está en nivel lógico 1

$A_N = '0'$ si la salida A_N del convertidor digital analógico está en nivel lógico 0

El voltaje de salida del convertidor de corriente a voltaje, está comprendido en el rango entre 0.01 y 2.10V con rango de variación de 0.008V.

El circuito desplazador de nivel, consiste en un amplificador operacional en la configuración de sumador inversor de ganancia unitaria. La salida del circuito desplazador de nivel, es la suma de un voltaje constante (7.5V) y del voltaje de salida del circuito

convertidor de corriente a voltaje. El rango del voltaje de control, generado por la Circuitería de Conversión Digital Analógica va de -7.5V a -9.5V.

C. *Circuito Generador de Señales Analógicas*

El Circuito Integrado Generador de Señales XR-2206 produce señales senoidales, triangulares y cuadradas de alta calidad, estabilidad y precisión. La frecuencia de la señal de salida, puede seleccionarse entre 0.01Hz y 1MHz. El Circuito Generador de Señales, se caracteriza por producir señales con una distorsión armónica baja (0.5%) y de una alta estabilidad (20ppm/°C). Por permitir un amplio rango de variación de frecuencia (2000:1) y por presentar una baja sensibilidad a cambios en el voltaje de alimentación (0.01%/V).

La frecuencia de oscilación de la señal de salida del Circuito Generador de Señales, es proporcional a la corriente I_r drenada del pin 7. Para variar la frecuencia de salida, se aplica el voltaje de salida del Circuito Desplazador de Nivel (V_{ODCN}) al pin de temporización numero 7. Esto permite obtener 256 valores de frecuencia entre 3.0 kHz y 95 kHz espaciados por saltos discretos de 0.3 kHz. Los valores máximo y mínimo de frecuencia, generadas por el XR2206 corresponden a los valores respectivos del voltaje V_{ODCN}:

El Circuito Generador de Señales, posee una salida para una señal cuadrada y una salida común para las señales senoidales y triangulares. La selección del tipo de la señal en la salida común, se realiza conectando/desconectando los pines 13 y 14, por medio de un interruptor analógico. Este interruptor, es controlado por medio de la señal SS2 de la Unidad de Control (línea PA1 para la salida 1 y PA5 para la salida 2 ambas del PIO2). Cuando SS2 está en nivel lógico 0, el interruptor analógico está abierto y la señal de salida es una señal senoidal. Cuando SS2 está en nivel lógico 1, el interruptor analógico está cerrado, por lo que la señal de salida es una señal triangular.

D. Circuito Selector de Señal

El Circuito Selector de Señal, comprende en el circuito lineal LF398 y sus dispositivos externos conectados en la configuración de interruptor de dos canales. Este circuito, presenta mejores características en lo que respecta a ganancia unitaria, ancho de banda, resistencia de entrada, voltaje de desplazamiento, número de conexiones externas y número de señales de control que los que pueden ser obtenidos con un multiplexor analógico, implementado utilizando interruptores analógicos del tipo LF13201, resistencias con 5% de tolerancia y un amplificador operacional.

En esta configuración, según las especificaciones del fabricante, la ganancia es unitaria para ambas entradas con un porcentaje muy bajo de variación (0.02% y 0.2%). El voltaje de desplazamiento es bajo (6mV y 75mV), la impedancia de entrada es alta ($10^{10}\Omega$ y $47k\Omega$), y el ancho de banda es amplio (de 1MHz para la entrada A y de 400kHz para la entrada B).

La función del Circuito Selector de Señal, consiste en seleccionar una de las señales producidas por el Circuito Generador de Señales. La salida triangular/senoidal del Circuito Generador de Señales, se aplica a la entrada A del LF398 y la salida cuadrada se aplica a la entrada B. La selección de la forma de onda de la salida, se realiza por medio de la señal digital SSI (línea PA0 para la salida 1 y PA4 para la salida 2 ambas del Z80PIO2). La señal SSI se aplica negada (por medio de una compuerta inversora) al pin 3 del LF398. Si la señal SSI está en estado lógico 0 la señal seleccionada es la señal senoidal/triangular, mientras que si la señal SSI está en estado lógico 1 la señal seleccionada es la señal cuadrada.

E. Circuitería de Selección de Amplitud

La Circuitería de Selección de Amplitud consiste en un amplificador operacional en la configuración inversora y un conjunto de interruptores analógicos que permiten variar la ganancia del amplificador.

El estado abierto/cerrado de los interruptores analógicos, es controlado por medio de señales de control digitales generadas por un decodificador de 2 a 4 (74LS139). Sólo un interruptor, puede estar activado (cerrado) a la vez. La selección del interruptor que debe cerrarse, la realiza la Unidad de Control por medio de las líneas SG1 y SG2 (PA1/PA2 del Z80PIO2 para la salida 1 y PA5/PA6 del Z80PIO2 para la salida 2).

El interruptor que se cierra, determina la resistencia de entrada del amplificador operacional (resistencia conectada a la entrada inversora) y por lo tanto, modifica la ganancia del amplificador, debido a que la relación $R_{\text{RETROALIMENTACION}}/R_{\text{ENTRADA}}$ varía. Los 4 valores de ganancia disponibles se muestran en la tabla 5.

Tabla 5. Ganancias del Circuito Selector de Ganancia

SG2	SG1	GANANCIA
0	0	1
0	1	1/2
1	0	1/3
1	1	1/5

Los 4 diferentes valores de ganancia, permiten obtener señales cuadradas, triangulares y senoidales de 4 amplitudes pico distintas. Estas se muestran en la tabla 6.

Tabla 6. Amplitudes de las señales generadas por la UGSA

GANANCIA	AMPLITUD DE LA SEÑAL SENOIDAL (Vp)	AMPLITUD DE LA SEÑAL TRIANGULAR (Vp)	AMPLITUD DE LA SEÑAL CUADRADA (Vp)
1	5	8	5
1/2	2.5	4	2.8
1/3	1.5	3	2
1/5	1.0	2	1.4

III. La Unidad de Procesamiento de Señales Analógicas

A. *Descripción General*

La Unidad de Procesamiento de Señales Analógicas, es la unidad del SDPG responsable de muestrear las señales presentes en las entradas del SDPG y de determinar la frecuencia de la señal presente en cada entrada.

Los circuitos de la Unidad de Procesamiento de Señales Analógicas comprenden un circuito de entrada para cada una de las entradas del SDPG, un multiplexor analógico, un circuito de selección de ganancia, un circuito acondicionador de amplitud, un circuito medidor de frecuencia y dos convertidores analógico-digitales. El procesamiento de cada entrada, se realiza seleccionando una de las dos entradas del SDPG por medio del multiplexor analógico, ajustando la amplitud por medio del circuito de selección de ganancia, y obteniendo las muestras correspondientes por medio de un convertidor analógico-digital. La Unidad de Procesamiento de Señales Analógicas obtiene 256 muestras de 8 bits por medición de la entrada seleccionada en 5428.5 μ s.

La PC determina por "software" el valor máximo y mínimo en base a los muestras obtenidas. El circuito medidor de frecuencia y un segundo convertidor analógico-digital permiten obtener el valor correspondiente a la frecuencia (1 dato de 8 bits).

B. El Circuito de Entrada

El Circuito de Entrada consiste en un conjunto de cinco resistencias, dos diodos de silicio y dos diodos zener (referirse al diagrama 5 del Apéndice B). El circuito de entrada, debe presentar una alta resistencia de entrada y además debe limitar la señal presente en la entrada, a valores tolerables por los circuitos de procesamiento y de muestreo.

Las resistencias R_{CE11} y R_{CE12} son las que determinan la resistencia de entrada y la fijan a un valor de $1M\Omega$, característico de los instrumentos de medición. Los diodos de silicio del tipo 1N914 (D_{CE11} , D_{CE21} , D_{CE12} , D_{CE22}) están en polarización inversa, por lo que entran en conducción solamente cuando el voltaje de entrada sobrepasa el valor máximo permitido. Los diodos zener del tipo NTE140A con voltaje de ruptura de 10V (D_{CE31} , D_{CE41} , D_{CE32} , D_{CE42}), establecen el valor máximo de voltaje permitido. Este valor está dado por la suma del voltaje de ruptura del diodo zener y del voltaje de polarización directa del diodo de silicio. Los valores máximos que se pueden aplicar a la entrada 1 del SDPG son de -11.02V y de +11.11V y los que se pueden aplicar a la entrada 2 son de +10.94V y -11.01V.

Limitar el voltaje de entrada a estos valores evita la saturación de los amplificadores operacionales utilizados en los circuitos de la Unidad de Procesamiento de Señales Analógicas al mismo tiempo que los protege de sobrevoltajes que pueden dañarlos.

C. *Circuito de Selección de Entrada*

El Circuito de Selección de Entrada, es un multiplexor analógico de dos entradas, al que se le aplican las señales de salida de los dos circuitos de entrada de la Unidad de Procesamiento de Señales Analógicas, que permite elegir la señal que será muestreada. Se utiliza el circuito lineal LF398 (referirse al diagrama 5 del Apéndice B) en la configuración de interruptor de dos canales, debido a que presenta mejores características en lo que respecta a ganancia unitaria, ancho de banda, resistencia de entrada, voltaje de desplazamiento, número de conexiones externas y número de señales de control, que los que pueden ser obtenidos con un multiplexor analógico, implementado utilizando interruptores analógicos del tipo LF13201, resistencias con 5% de tolerancia y un amplificador operacional. De esta forma, la ganancia es unitaria para ambas entradas con un porcentaje muy bajo de variación (0.02% y 0.2%), el voltaje de desplazamiento es bajo (6mV y 75mV), la impedancia de entrada es alta ($10^{10}\Omega$ y $47k\Omega$), y el ancho de banda es amplio (1 MHz para la entrada A y 400kHz para la entrada B), según especificaciones del fabricante.

La entrada A del LF398 es la señal obtenida del circuito de entrada 1 y la entrada del LF398 es la señal obtenida del circuito de entrada 2. La selección de la entrada a muestrear, se realiza por medio de la señal digital PB3 del Z80PIO2 de la Unidad de Control que denominaremos SIN.

La señal SIN se aplica negada (por medio de una compuerta Inversora) al pin 3 del LF398. Si la señal SIN está en estado lógico 0, la señal seleccionada es la señal de la entrada 1 de la Unidad de Procesamiento de Señales Analógicas. Si la señal SIN está en estado lógico 1, la señal seleccionada es la de la entrada 2 de la Unidad de Procesamiento de Señales Analógicas.

D. Circuito de Selección de Ganancia

El Circuito de Selección de Ganancia, consiste en un amplificador operacional del tipo TL074 en la configuración inversora y cuatro interruptores analógicos del tipo LF13201 que permiten variar el valor de la resistencia que se conecta a la entrada inversora del amplificador operacional (referirse al diagrama 6 del Apéndice B). De esta forma, los cuatro posibles valores de ganancia, están determinados por el interruptor analógico que se cierra y por la relación correspondiente $R_{\text{RETROALIMENTACION}}/R_{\text{ENTRADA}}$. La selección del interruptor analógico que debe cerrarse, se realiza al activar una de las salidas del decodificador de 2 a 4. La salida se selecciona por medio de las señales AI1 y AI2 que corresponden a las líneas PA0 y PA1 del Z80PIO2 de la Unidad de Control. La ganancia del Circuito de Selección de Ganancia, en función de los valores de las señales AI1 y AI2 se muestran en la tabla 7.

Tabla 7. Ganancias del Circuito Selector de Ganancia

A12	A11	GANANCIA
0	0	1/2
0	1	1
1	0	4
1	1	8

El Circuito de Selección de Ganancia, permite la variación de la amplitud de la señal seleccionada por el LF398 y por lo tanto, permite amplificar pequeñas señales, o bien atenuar señales de gran amplitud presentes en las entradas de la Unidad de Procesamiento de Señales Analógicas, para que éstas puedan procesarse con mayor facilidad.

E. Circuito Acondicionador de Amplitud

El Circuito Acondicionador de Amplitud, es el responsable de convertir la señal obtenida del Circuito de Selección de Ganancia, a una señal cuyos valores máximo y mínimo, estén en el rango de 0 a 5V. Esto, se debe a que el convertidor analógico-digital de la Unidad de Procesamiento de Señales Analógicas, sólo puede convertir voltajes que estén en este rango. El Circuito Acondicionador de Amplitud, consiste en una red de resistencias utilizadas para realizar una división de voltaje y un amplificador operacional, utilizado como seguidor de voltaje. Los valores de la resistencias del Circuito Acondicionador de Amplitud, fueron determinadas considerando que la señal de entrada, está en el rango $\pm 12V$ (lo cual incluye el voltaje de saturación de los amplificadores operacionales).

F. Circuito Medidor de Frecuencia

El Circuito Medidor de Frecuencia, consiste en el circuito convertidor de frecuencia a voltaje LM2907-8 junto con las resistencias y capacitores necesarios para su operación (referirse al diagrama 6 del Apéndice B). Este circuito, genera un voltaje DC proporcional a la frecuencia aplicada a su entrada. La versión utilizada del circuito, es la versión de 8 pines con entrada referida directamente a tierra. La señal aplicada a la entrada del Circuito Medidor de Frecuencia debe poseer valores positivos y negativos para que el circuito genere un voltaje DC de salida. Los valores de los capacitores y las resistencias, se seleccionaron para minimizar el rizado en la salida y para establecer el rango del voltaje de salida. De esta forma, el rango de frecuencias que puede medirse es entre 0 y 12 kHz, obteniéndose un voltaje de salida de 0V y 5V respectivamente.

G. Convertidores Analógico-Digitales

La Unidad de Procesamiento de Señales Analógicas, utiliza dos convertidores analógico-digitales del tipo ADC0820 para la obtención de las 256 muestras del voltaje de la entrada seleccionada y del valor de la frecuencia de la señal presente en la entrada seleccionada.

El ADC0820 es un convertidor analógico-digital de 8 bits, con un tiempo de conversión de 1.5 μ s, que utiliza la técnica de conversión paralela denominada half-flash, y que tiene entradas diferenciales. El ADC0820 no requiere de una señal de reloj externa para realizar la conversión y no requiere de un circuito adicional de muestreo y retención, para señales con un ritmo de variación menor que 100mV/ μ s.

El modo en el que se utilizan ambos convertidores analógico-digitales, es el modo de escritura-lectura en el cual el pin de modo del convertidor (pin 7) está conectado a +5V, con lo que la conversión se inicia cada vez que la entrada de escritura WR pasa al nivel lógico 0. El tiempo de conversión se minimiza, forzando la entrada de lectura RD al estado lógico 0, después de 1 μ s del cambio al estado lógico 1 de la señal de escritura WR, con lo que los datos de la conversión, pueden ser leídos inmediatamente.

El primer convertidor analógico-digital, obtiene las 256 muestras de la entrada seleccionada. La señal de escritura, corresponde a la línea PA0 del Z80PIO3 y la señal de lectura corresponde a la línea PA1 del Z80PIO3 de la Unidad de Control. Las líneas de datos de este convertidor D0-D7, se conectan a las líneas PB0-PB7 del Z80PIO3.

El segundo convertidor analógico-digital, obtiene una muestra correspondiente al valor en formato digital, de la frecuencia de la señal presente en la entrada seleccionada. La señal de escritura, corresponde a la línea PA0 del Z80PIO4 y la señal de lectura corresponde a la línea PA1 del Z80PIO4 de la Unidad de Control. Las líneas de datos de este convertidor D0-D7, se conectan a las líneas PB0-PB7 del Z80PIO4.

IV. UNIDAD DE INTERFACE CON LA PC

A. *Descripción General*

La Unidad de Interface con la PC comprende el Z80DART, los circuitos de conversión de niveles TTL a niveles RS-232C (Line Driver MC1488) y de niveles RS-232C a niveles TTL (Line Receiver MC1489). El diagrama de esta unidad, se muestra en el diagrama 2 del Apéndice B.

La Unidad de Interface con la PC, es la encargada establecer la comunicación serial con la PC con los parámetros adecuados (9600bps, 8 bits por caracter, sin paridad y un bit de parada), de enviar los datos relativos a las mediciones efectuadas por la UPSA a la PC y de recibir los comandos de modificación de la operación del SDPG, enviados por la PC. Todas las funciones de transferencia de datos y de determinación del estatus de la comunicación con la PC, las realiza el Z80DART por medio de la Unidad de Control. La traducción de niveles, la realizan los circuitos de interface.

B. *Señales utilizadas para la interface*

Las señales utilizadas para la comunicación con la PC, son únicamente la señal de transmisión de datos (TXD), la de recepción de datos (RXD) y la tierra. La señal DCD del Z80DART, se conecta a la señal RXD para que éste pueda determinar en que momento está recibiendo datos y poder así muestrearlos sincronizados con la señal de reloj aplicada a los pines (TXC y TXD). Las señales de RTS y CTS no se habilitan en el Z80DART y por lo tanto, no son necesarias para la comunicación entre PC y SDPG.

V. El Programa de operación del SDPG

El Programa de Operación del SDPG, es el programa que gobierna el funcionamiento de la Unidad de Control del SDPG. Este programa, está escrito en lenguaje ensamblador para el microprocesador Z80 y está almacenado en la EPROM de la Unidad de Control. El programa fuente SPDG64.ASM fue compilado utilizando el compilador para Z80 AS.EXE (versión 1.5) para obtener el programa objeto SDPG62.OBJ que reside en la EPROM de la Unidad de Control.

En el Programa de Operación, se definen todos los dispositivos que conforman el SDPG y también todas las rutinas que debe ejecutar el Z80 para que el SDPG pueda realizar el procesamiento y generación de las señales analógicas. El Programa de Operación, está orientado únicamente al control de las unidades que conforman el SDPG, es decir que el SDPG no realiza ninguna función, a menos que la PC se lo indique. Por lo tanto, el SDPG es un dispositivo esclavo de la PC.

El Programa de Operación, consiste en una rutina principal, en la que a partir del dato enviado por la PC, se selecciona el tipo de procesos o rutinas que deben ejecutarse. Los diagramas de flujo de la rutina principal y de las sub-rutinas del Programa de Operación, se muestran en el Apéndice C. El listado compilado del programa, se muestra en el Apéndice D.

A. *Rutina Principal del Programa de Operación*

La rutina principal del Programa de Operación, es la rutina responsable de definir todos los dispositivos del SDPG, de enviarles los valores iniciales y de esperar el comando enviado por la PC para activar los dispositivos correspondientes.

La rutina principal, contiene la definición de las direcciones de los dispositivos del SDPG como lo son las direcciones de los Z80PIO, del Z80CTC, del Z80DART, de la memoria para muestras, del puntero de pila y de las variables de memoria.

La primera acción ejecutada, consiste en definir los parámetros de operación de cada uno de los puertos de los Z80PIO y de los canales del Z80CTC y del Z80DART. La segunda acción, consiste en enviar a los puertos de los Z80PIOs los valores por default, de las salidas y los valores de deshabilitación de los circuitos de procesamiento de las entradas. La tercera acción, consiste en iniciar a cero todas las variables de memoria.

Luego de ejecutar la tercera acción, el SDPG espera el envío por la PC de un carácter entre 00H y 08H. Este carácter define la acción que debe ejecutar el SDPG, es decir ejecutar las rutinas, para modificar las características de las salidas, o bien realizar el procesamiento de las entradas.

Si el carácter enviado por la PC es 00H, entonces el SDPG espera la recepción del valor de la frecuencia de la salida 1 para su envío al puerto A del Z80PIO1.

Si el carácter enviado por la PC es 01H entonces el SDPG espera la recepción del valor de la amplitudes y forma de onda de ambas salidas para su envío al puerto A del Z80PIO2.

Si el carácter enviado por la PC es 02H, entonces el SDPG espera la recepción del valor de la frecuencia de la salida 2 para su envío al puerto B del Z80PIO1.

Si el carácter enviado por la PC es 03H, entonces el SDPG espera la recepción del valor de la ganancia a utilizar, cuando se realice el muestreo de la entrada 1. Este valor se almacena en la variable de memoria COMA, y se envía al puerto B del Z80PIO2 únicamente cuando se ejecuta la rutina para muestrear la entrada 1 (MX_IN1).

Si el carácter enviado por la PC es 04H, entonces el SDPG espera la recepción del valor de la ganancia a utilizar, cuando se realice el muestreo de la entrada 2. Este valor se almacena en la variable de memoria COMA y se envía al puerto B del Z80PIO2 únicamente cuando se ejecuta la rutina para muestrear la entrada 2 (MX_IN2).

Si el carácter enviado por la PC es 05H entonces el SDPG realiza el muestreo de la señal presente en la entrada 1, así como la medición de la frecuencia y envía las muestras obtenidas a la PC (se ejecutan las rutinas MX_IN1 y TMT_DAT).

Si el carácter enviado por la PC es 06H, entonces el SDPG realiza el muestreo de la señal presente en la entrada 2 y envía la muestras obtenidas a la PC (se ejecutan las subrutinas MX_IN2 y TMT_DAT) .

Si el carácter enviado por la PC es 07H entonces el SDPG envía el valor medido de la frecuencia de la entrada 1 a la PC (se ejecuta la rutina TMT_FRC).

Si el carácter enviado por la PC es 08H, entonces el SDPG envía el valor medido de la frecuencia de la entrada 2 a la PC (se ejecuta la rutina TMT_FRC).

En caso de que no se reciba uno de los caracteres anteriores, el SDPG seguirá esperando a que la PC envíe un carácter en el rango válido que le indique la operación que debe efectuar.

B. Rutina para el muestreo de una entrada

La rutina para el muestreo de la entrada seleccionada M_IN, es la responsable de la obtención y almacenamiento en memoria de 256 muestras de la señal presente en la entrada.

Esta rutina, realiza un bucle cuyo contador de control es el registro B, el cual indica el número de muestras pendientes (se hace uso de la instrucción de ramificación DJNZ). Se utiliza el puntero IX para direccionar la posición en la RAM de la Unidad de Control, en la que se almacena cada muestra. La rutina activa el convertidor analógico digital en cada ciclo del bucle, al enviar las señales WR y RD al convertidor, en la secuencia correcta, para que éste realice la conversión y entregue el dato de 8 bits, que corresponde al valor de voltaje presente en la entrada seleccionada.

La rutina, permite realizar el muestreo de la señal obtenida del Circuito de Selección de Señal de la Unidad de Procesamiento de Señales Analógicas y se utiliza por lo tanto para, muestrear ambas entradas. Las rutinas MX_IN1 y MX_IN2, son las que seleccionan la entrada 1 y 2 respectivamente, enviando la combinación adecuada al puerto B del Z80PIO2.

Implementar la conversión directamente en el bucle, permite minimizar el tiempo que transcurre entre la obtención de muestras consecutivas, ya que se elimina el tiempo de ejecución de las instrucciones CALL y RET, necesarias para ejecutar una sub-rutina que únicamente active el convertidor analógico-digital. En versiones anteriores del Programa de Operación, la rutina M_IN realizaba la llamada a una sub-rutina denominada ADC_ACT la cual activaba el convertidor. Sin embargo, al realizar cálculos de tiempo total de obtención de muestras, se determinó que este esquema limitaba el rendimiento del SDPG.

Se prefirió optar por no utilizar esta rutina, e incluir la conversión analógica digital en el bucle, debido a que de esta forma, se puede muestrear y reproducir una señal de mayor frecuencia. La obtención de las 256 muestras y su almacenamiento en memoria, son ejecutados en un tiempo de 5428.5 μ s, lo que corresponde a 47158 muestras por segundo. Por lo tanto, la máxima frecuencia que puede muestrearse basándose en el teorema del muestreo de Nyquist es de 23.58 kHz.

C. Rutina para medición de la frecuencia

La rutina para medición de la frecuencia, es la responsable de obtener la representación en formato digital del valor del voltaje DC generado por el Circuito Medidor de Frecuencia. Esta rutina, efectúa la activación del convertidor analógico digital para medición de frecuencia y almacena el dato en la variable de memoria FRC1. Esta rutina, es común para ambas entradas, ya que son las rutinas MX_IN1 y MX_IN2 las responsables de enviar la señal adecuada al Circuito de Selección de Entrada, para que se determine la frecuencia de la señal en las entradas 1 y 2 respectivamente.

D. Rutinas para medición de las entrada 1 y 2

Las rutinas MX_IN1 y MX_IN2 son las rutinas responsables de la obtención de las muestras y de la determinación de la frecuencia de las entradas 1 y 2 respectivamente. Cada una de estas rutinas, envía las señales de control adecuadas al puerto B del Z80PIO2. Estas señales de control, determinan la ganancia del Circuito de Selección de Ganancia de la Unidad de Procesamiento de Señales Analógicas y la señal de salida del Circuito de Selección de Entrada (entrada 1 ó 2). Estas rutinas, hacen uso de la rutina M_IN para obtener y almacenar las muestras y M_FRC para determinar la frecuencia de la entrada seleccionada.

E. Rutina para la transmisión de datos a la PC

La rutina para la transmisión de datos a la PC, TMT_DAT, es la responsable de acceder los datos de las muestras obtenidas por la rutina M_IN y enviarlos consecutivamente a la PC. Al igual que la rutina M_IN se utilizan los registros B como contador de número de muestras e IX como puntero de memoria.

F. Rutina para la transmisión de la frecuencia a la PC

La rutina para la transmisión de la frecuencia a la PC, TMT_FRC, es la responsable de recuperar el valor medido de la frecuencia de la entrada muestreada (almacenado en la variable FRC1) y enviarlo a la PC.

VI. Programa de Usuario del SDPG

El programa de Usuario del SDPG permite al usuario controlar la operación del SDPG para ajustar las características de las salidas y de las entradas a sus necesidades. Este programa está almacenado en la PC la cual actúa como unidad maestra. El principal objetivo seguido durante el desarrollo del programa, fue el de presentar al usuario un fácil acceso a todas las opciones para el control del SDPG. Este programa fue desarrollado en Turbo Pascal versión 7.

En las siguientes secciones, se describen las opciones disponibles al usuario, la lógica general del programa y los requerimientos para la instalación.

A. *Funciones que realiza el Programa de Usuario*

Las funciones específicas que realiza el Programa de Usuario, con los datos enviados por el SDPG son las siguientes:

- interpretar los datos relativos a una entrada particular;
- obtener el valor real para una entrada dada del voltaje máximo y del mínimo;
- desplegar el valor de la frecuencia de la señal presente en la entrada;
- calcular el valor del voltaje promedio;
- enviar comandos de control para cambio de amplitud , de frecuencia y de tipo de la señal de salida;

B. Requerimientos del Programa de Usuario

El Programa de Usuario, requiere para su ejecución, de la siguiente configuración:

- una computadora personal con 1MB de RAM;
- monitor VGA;
- 1 MB de espacio disponible en disco;
- disco flexible de 3.5 " de 1.44MB;
- sistema operativo DOS 6.0 .

C. Instalación del Programu de Usuario

Para instalar el Programa de Usuario en una computadora personal, el usuario debe digitar el comando 'a:instalar'. Esto activará la rutina de instalación, la cual se encarga de grabar todos los archivos utilizados por el Programa de Usuario en el disco duro, en el directorio SDPG .

Al terminar la rutina de instalación, aparecerá en pantalla un mensaje, indicando que la instalación ha sido concluida. Los programas instalados en el directorio SDPG comprenden los archivos de los tipos de letra con extensión '.CHR' , las unidades TPU , los archivos de datos con extensión '.DAT' y el programa ejecutable SDPG.EXE.

D. Ejecución del Programa de Usuario

El Programa de Usuario, es ejecutado cuando en el indicador del DOS se digita el comando 'c:\SDPG'. El Programa de Usuario, inicia con una pantalla de presentación y luego presenta la pantalla de trabajo. En este punto, el usuario puede elegir entre las diferentes opciones presentadas, para modificar la operación del SDPG.

La pantalla del Programa de Usuario, comprende las áreas para visualizar la forma de onda de las entradas 1 y 2, las áreas en las que muestran las características de las dos entradas, el menú de opciones o menú principal y el área de mensajes (área al lado del menú principal).

E. Opciones disponibles al usuario

El Programa de Usuario permite controlar directamente los parámetros de operación del SPDG. El usuario puede por lo tanto, modificar la ganancia de ambas entradas, modificar la amplitud, frecuencia y forma de onda de cada salida. Almacenar y recuperar archivos de las mediciones efectuadas; imprimir los datos de mediciones realizadas; y solicitar el muestreo de las entradas.

Todas estas operaciones, fueron agrupadas en el menú principal, el cual aparece en la parte inferior de la pantalla y presenta 12 opciones diferentes (cada una correspondiente a una operación específica). La opción que puede seleccionarse, es la que presenta un

fondo blanco. Para desplazarse entre las diferentes opciones, se deben utilizar las teclas ↑, ↓, ←, y →. Se selecciona la opción al presionar la tecla <Enter>. La tecla <Esc> permite regresar al DOS, cancelando así la operación del programa. A continuación, se describe cada una de las opciones presentadas en el menú principal.

1. Almacenamiento en Disco

Los datos de las muestras de las entradas, enviados por el SDPG, pueden ser almacenados en el directorio, en el cual está grabado el programa de usuario. Para almacenar en disco, el usuario debe elegir la opción 'Grabar' del Menú principal. La información de las entradas, enviada por el SDPG, se almacena en el disco, como un archivo de 514 bytes con extensión '.DAT'. Se puede elegir el nombre del archivo en el que se almacenará la información, presionando las teclas ↑ y ↓. Al presionar la tecla <Enter> se selecciona el nombre del archivo actual, se graban los datos en el disco duro y se retorna al menú principal. La tecla <Esc> cancela la selección de archivo y retorna al menú principal.

2. Recuperar Datos del Disco

Los datos relativos a las señales presentes en las entradas del SDPG que hayan sido almacenados en el disco con anterioridad, pueden ser mostrados en pantalla por medio de la opción 'Recuperar' del Menú principal. El nombre del archivo a recuperar, se elige por medio de las teclas \uparrow y \downarrow y al presionar a tecla <Enter> se recuperan los datos de ambas entradas y se presentan en pantalla. Al presionar la tecla <Esc> la selección del archivo es cancelada y se retorna al menú principal.

3. Impresión de datos

Los datos obtenidos del SDPG relativos a las señales, pueden ser enviados a la impresora, por medio de la opción 'Imprimir' del Menú Principal. Esta opción, envía a la impresora las características de cada entrada (máximo, mínimo, promedio y frecuencia) y la forma de onda de la señal. Pueden imprimirse los datos actuales y los datos de mediciones anteriores que hayan sido almacenadas con anterioridad. La selección de la información a imprimir, se realiza eligiendo el archivo de muestras correspondiente por medio de la opción 'Recuperar'. Al presionar 'enter' y seleccionar la opción 'Imprimir', la información presentada en pantalla se envía a la impresora. La tecla <Esc> cancela la impresión y permite el retorno al menú principal.

4. Modificar la Forma de Onda de la Salida 1

El usuario, puede elegir la forma de onda de la salida 1 al seleccionar la opción 'Forma 1' del menú principal. Las formas de onda que pueden elegirse son senoidal, triangular y cuadrada. Para poder acceder todos los tipos de forma de onda, se deben presionar las teclas \uparrow y \downarrow . Al presionar la tecla <Enter>, la forma que aparece en pantalla se selecciona. Al presionar la tecla <Esc> se cancela la selección y se retorna al menú principal.

5. Modificar de la Amplitud de la Salida 1

El usuario puede elegir la amplitud de la señal de la salida 1, al seleccionar la opción 'Amplitud 1' del menú principal. El usuario modifica la amplitud, al seleccionar la ganancia del Circuito de Selección de Amplitud de la salida 1, descrito en el capítulo 3. Los valores de ganancia que pueden elegirse son 1, 1/2, 1/3 y 1/5. Para cambiar el valor mostrado en pantalla, se utilizan las teclas \uparrow y \downarrow . Al presionar la tecla <Enter>, el valor que aparece en pantalla se selecciona. Al presionar la tecla <Esc> se cancela la selección y se retorna al menú principal.

6. Modificar la Frecuencia de la señal de la salida 1

El usuario, puede elegir la frecuencia de la señal de la salida 1 al seleccionar la opción 'Frecuencia 1' del menú principal. Se tienen disponibles 256 valores de frecuencia entre 3.0kHz y 95kHz. Para cambiar el valor mostrado en pantalla, se utilizan las teclas ↑ y ↓. Al presionar la tecla <Enter> la frecuencia actual es seleccionada. Al presionar la tecla <Esc> se cancela la selección y se retorna al menú principal.

7. Modificar la Forma de Onda de la Salida 2

El usuario puede elegir la forma de onda de la salida 2, al seleccionar la opción 'Forma 2' del menú principal. Las formas de onda que pueden elegirse, son senoidal, triangular y cuadrada. Para poder acceder todos los tipos de forma de onda, se deben presionar las teclas ↑ y ↓. Al presionar la tecla <Enter> la forma que aparece en pantalla se selecciona. Al presionar la tecla <Esc>, se cancela la selección y se retorna al menú principal.

8. Modificar la Amplitud de la Salida 2

El usuario, puede elegir la amplitud de la señal de la salida 2 al seleccionar la opción 'Amplitud 2' del menú principal. El usuario, modifica la amplitud al seleccionar la ganancia del Circuito de Selección de Amplitud de la salida 2 descrito en el capítulo 3. Los valores de ganancia que pueden elegirse son 1, 1/2, 1/3 y 1/5. Para cambiar el valor mostrado en pantalla, se utilizan las teclas \uparrow y \downarrow . Al presionar la tecla <Enter>, el valor que aparece en pantalla se selecciona. Al presionar la tecla <Esc>, se cancela la selección y se retorna al menú principal.

9. Modificar la Frecuencia de la señal de la Salida 2

El usuario puede elegir la frecuencia de la señal de la salida 2 al seleccionar la opción 'Frecuencia 2' del menú principal. Se tienen disponibles 256 valores de frecuencia entre 3.0kHz y 95kHz. Para cambiar el valor mostrado en pantalla, se utilizan las teclas \uparrow y \downarrow . Al presionar la tecla <Enter>, la frecuencia actual es seleccionada. Al presionar la tecla <Esc> se cancela la selección y se retorna al menú principal.

10. Solicitar Muestras de Ambas Entradas

Para la obtención de las muestras de las señales presentes en las entradas del SDPG, el usuario debe elegir la opción 'Muestreo' en el menú principal. Esta opción, le permitirá actualizar los datos presentados en pantalla en lo referente a forma de onda de la señal y características (valor máximo, valor mínimo, valor promedio y frecuencia).

Al elegir esta opción, el mensaje 'Muestreo=No' aparece en la pantalla. Si el usuario desea recibir datos del SDPG, deberá presionar la tecla \uparrow de tal forma que aparezca el mensaje 'Muestreo=Si' en pantalla y presionar la tecla <Enter>. Si el usuario no desea recibir datos, puede presionar la tecla <Esc> o bien modificar el mensaje de 'Muestreo=Si' a 'Muestreo=No' con la tecla \downarrow y luego presionar <Enter>.

11. Modificar la Ganancia de la Entrada 1

El usuario, puede elegir la ganancia del Circuito de Selección de Ganancia de la UPSA seleccionando la opción 'Ganancia 1' en el menú principal. El valor que el usuario elija, determina la ganancia que debe poseer este circuito, al realizar el muestreo de la entrada 1. Los valores de ganancia que pueden elegirse son 1/2, 1, 4 y 8. Para cambiar el valor mostrado en pantalla, se utilizan las teclas \uparrow y \downarrow . Al presionar la tecla <Enter>, el valor actual es seleccionado. Al presionar la tecla <Esc> se cancela la selección y se retorna al menú principal.

12. Modificar la Ganancia de la Entrada 2

El usuario, puede elegir la ganancia del Circuito de Selección de Ganancia de la UPSA seleccionando la opción 'Ganancia 1' en el menú principal. El valor que el usuario elija, determina la ganancia que debe poseer este circuito al realizar el muestreo de la entrada 1. Los valores de ganancia que pueden elegirse son 1/2, 1, 4 y 8. Para cambiar el valor mostrado en pantalla, se utilizan las teclas \uparrow y \downarrow . Al presionar la tecla <Enter>, el valor actual es seleccionado. Al presionar la tecla <Esc> se cancela la selección y se retorna al menú principal.

VII. CONCLUSIONES Y RECOMENDACIONES

1. El SDPG es una microcomputadora especializada, dedicada al muestreo de las señales presentes en sus dos entradas y a la generación de señales analógicas básicas, bajo el control de la Computadora Personal.
2. El SDPG genera dos salidas analógicas independientes, que pueden ser señales de forma de onda triangular, senoidal o cuadrada, de frecuencia variable entre 3.0kHz y 95kHz y con cuatro valores posibles de ganancia, que permiten ajustar la amplitud de salida. Todas estas características pueden ser modificadas por el usuario desde el programa almacenado en la computadora personal.
3. En el diseño de microcomputadoras especializadas, como el SDPG es preferible utilizar los dispositivos periféricos diseñados para el microprocesador, ya que esto no sólo simplifica el circuito y el programa en ensamblador, sino que permite actualizar la velocidad de operación del circuito, al aparecer en el mercado los circuitos integrados para una velocidad mayor.

4. La creciente capacidad de cómputo de las computadoras personales, permite relegar a microcomputadoras especializadas como el SDPG las funciones de muestreo y medición, con objeto de aprovechar al máximo toda la capacidad de cómputo y de procesamiento de datos de las computadoras.

5. Los componentes utilizados en la implementación del SDPG, permiten que éste sea un circuito de bajo costo y fácilmente reproducible, ya que utiliza componentes de fácil adquisición en el mercado. Además de que se obtiene con ello, un circuito con un rendimiento aceptable y un sistema de control sencillo desde la computadora personal.

6. El SDPG puede ser tomado como base para el desarrollo de sistemas digitales para procesamiento de señales con interface con una computadora personal, ya que su diseño puede ser modificado, para incluir un mayor número de entradas, o bien sólo generar señales, utilizando siempre los circuitos presentados.

7. El SDPG puede ser utilizado en aplicaciones en las que se requiera el monitoreo y observación de la forma de onda de una señal analógica, en aplicaciones donde sea necesario realizar un análisis de señal. Lo mismo, en aplicaciones en las que sea necesaria la aplicación de señales de características controladas a un circuito y la observación de las respuestas obtenidas.

8. El SDPG puede llegar a ser incluido como apoyo en los laboratorios, en alguno de los cursos de la la Universidad, ya que reúne muchos conocimientos generales y los integra en un proyecto donde todos los principios para el desarrollo de sistemas digitales de procesamiento y generación de señales, son abarcados. Además que es un proyecto de fácil construcción y bajo costo.

VIII.- BIBLIOGRAFIA

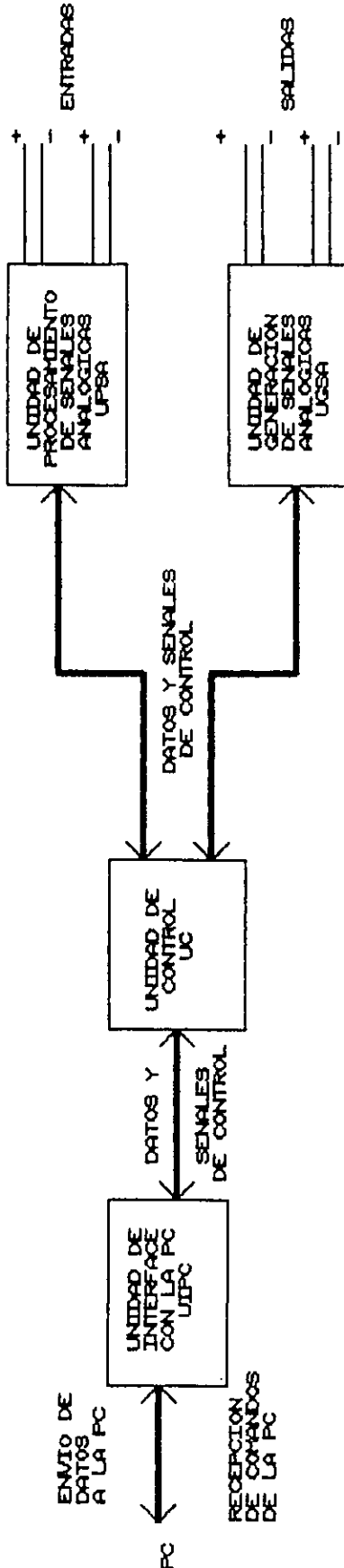
- (1) Coffron, J. Practical Interfacing Techniques for Microprocessor Systems. Englewood Cliffs. Prentice Hall Inc. 1983. 401p.
- (2) Coffron, J. Z80 Applications. Berkeley. Sybex Inc. 1984. 295p.
- (3) Coughlin, R. Driscoll, F. Circuitos Integrados Lineales y Amplificadores Operacionales. México. Prentice-Hall Hispanoamericana S.A. 1987. 394p. 2da Ed. Prentice Hall Inc. 1987.
- (4) Daugherty, K. Analog to Digital Conversion- A practical approach. New York. McGraw Hill Inc. 1994. 213p.
- (5) Fleming, T. Analog/Digital and Digital/Analog data converters. EDN, May 29. 1986. 22p.
- (6) Galan, P. Pascal y Turbopascal. Fundamentos de Programación. Madrid. Paraninfo S.A. 1988. 210p.
- (7) Goldstein, L. Turbo Pascal-Introducción a la programación orientada a objetos. México. Prentice-Hall Hispanoamericana. 1993. 502p.
- (8) Hayes, J. Diseño de Sistemas Digitales y Microprocesadores. México. MC-Graw Hill Interamericana de México S.A. 1988. 873p.
- (9) Holdworth, B. Microprocessor Engineering. Tiptree. Butterworth & Co Publishers Ltd. 1986. 340p.
- (10) Jung, W. IC Op-Amp Cookbook Third Edition. Carmel. Prentice Hall Computer Publishing. 1993. 581p.
- (11) Mano, M. Diseño Digital. México. Prentice Hall Hispanoamericana S.A. 1987. 491p.
- (12) Millman, J. Halkias, C. Electrónica Integrada. Circuitos Analógicos y digitales. Barcelona. Editorial Hispano Europea S.A. 1986. 917p.
- (13) National Semiconductor Corporation. Data Acquisition Linear Devices Databook. Santa Clara. National Semiconductor Corporation. 1989 Edition. 924p.
- (14) National Semiconductor Corporation. Operational Amplifiers Databook. Santa Clara. National Semiconductor Corporation. 1993 Edition. 1234p.

- (15) Texas Instruments Incorporated. Linear Circuits Databook Volume 3. Voltage Regulators, Supervisors, Comparators, Special Functions, and Building Blocks. Dallas. Texas Instruments Incorporated. 1992. 927p.
- (16) Texas Instruments Incorporated. Linear Circuits Databook Volume 1. Operational Amplifiers. Dallas. Texas Instruments Incorporated. 1992. 1404p.
- (17) Texas Instruments Incorporated. TTL Logic Databook. Standard TTL, Schottky, Low Power Schottky. Dallas. Texas Instruments Incorporated. 1988. 1240p.
- (18) Tocci, R. Sistemas Digitales: Principios y Aplicaciones. 5ta Edición. México. Prentice Hall Hispanoamericana S.A. 1993. 823p
- (19) Wakerly, J. Diseño Digital: Principios y Practicas. México. Prentice Hall Hispanoamericana S.A. 1992. 734p.
- (20) Welsh, J. Elder, J. Pascal Introducción. México. Prentice Hall Hispanoamericana S.A. 1983. 268p.
- (21) Zilog. Z80 Family Databook, Discrete Devices and Embedded Controllers. Campbell. Zilog Inc. 1994. 425p.

APENDICE A

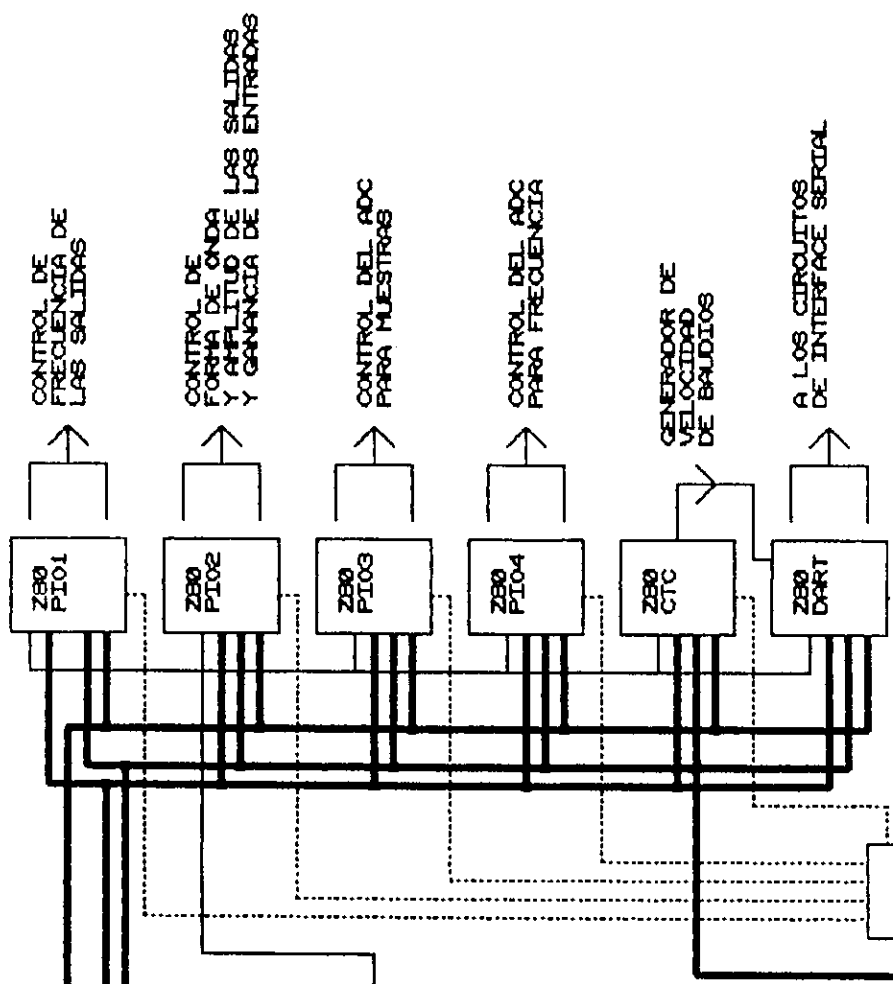
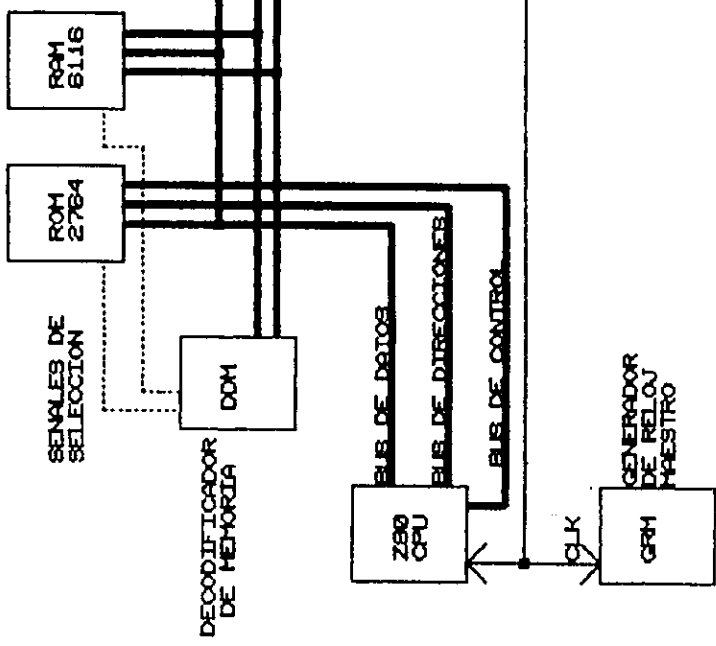
Diagramas de bloques de
las Unidades del SDPG

DIAGRAMA DE BLOQUES DEL SISTEMA DIGITAL PARA PROCESAMIENTO Y GENERACION DE SEÑALES ANALOGICAS



TRABAJO DE GRADUACION	
Title	DIAGRAMA 1 APENDICE A
Size Document Number	REV
A	JUAN CARLOS MAYORGA J
Date:	January 11, 1996 Sheet 1 of 4

UNIDAD DE CONTROL



CONTROL DE FRECUENCIA DE LAS SALIDAS

CONTROL DE FORMA DE ONDA Y ANCHURAS DE LAS SALIDAS Y GANANCIA DE LAS ENTRADAS

CONTROL DEL ADC PARA MUESTRAS

CONTROL DEL ADC PARA FRECUENCIA

GENERADOR DE VELOCIDAD DE BAUDIOS

A LOS CIRCUITOS DE INTERFACE SERIAL

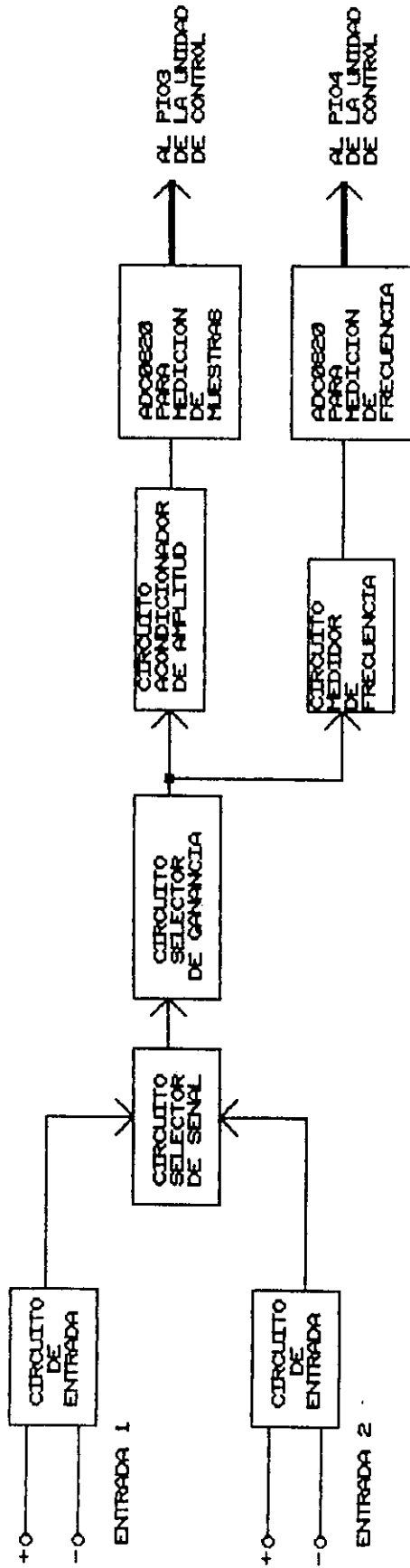
TRABAJO DE GRADUACION

Title	DIAGRAMA 2 APENDICE A
Size Document Number	A
Date:	January 8, 1996
Sheet	2 of 4

JUAN CARLOS MAYORCA J

DECODIFICADOR DE PUERTOS

UNIDAD DE PROCESAMIENTO DE SEÑALES ANALÓGICAS



TREBAJO DE GRADUACION

Title

DIAGRAMA 3 APENDICE A

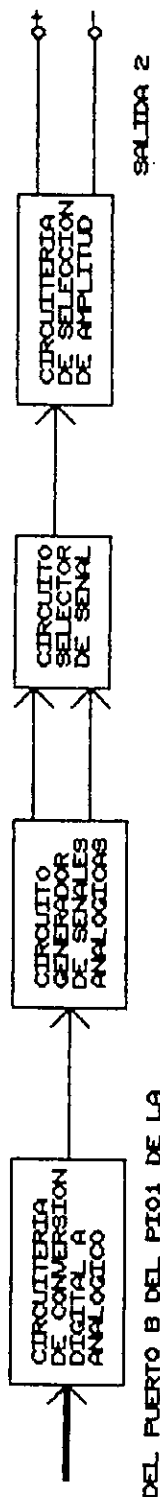
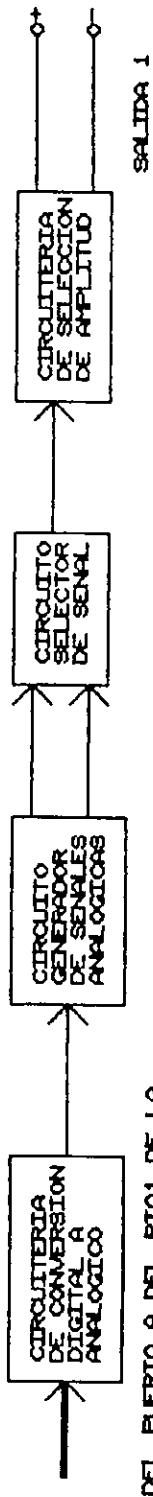
Size Document Number

A JUAN CARLOS MAYORGA J

Date: January 8, 1995

Sheet 3 of 4

UNIDAD DE GENERACION DE SEÑALES ANALOGICAS



TRABAJO DE GRADUACION

Title

DIAGRAMA 4 APENDICE A

Size

A

Date:

January 8, 1996

Sheet

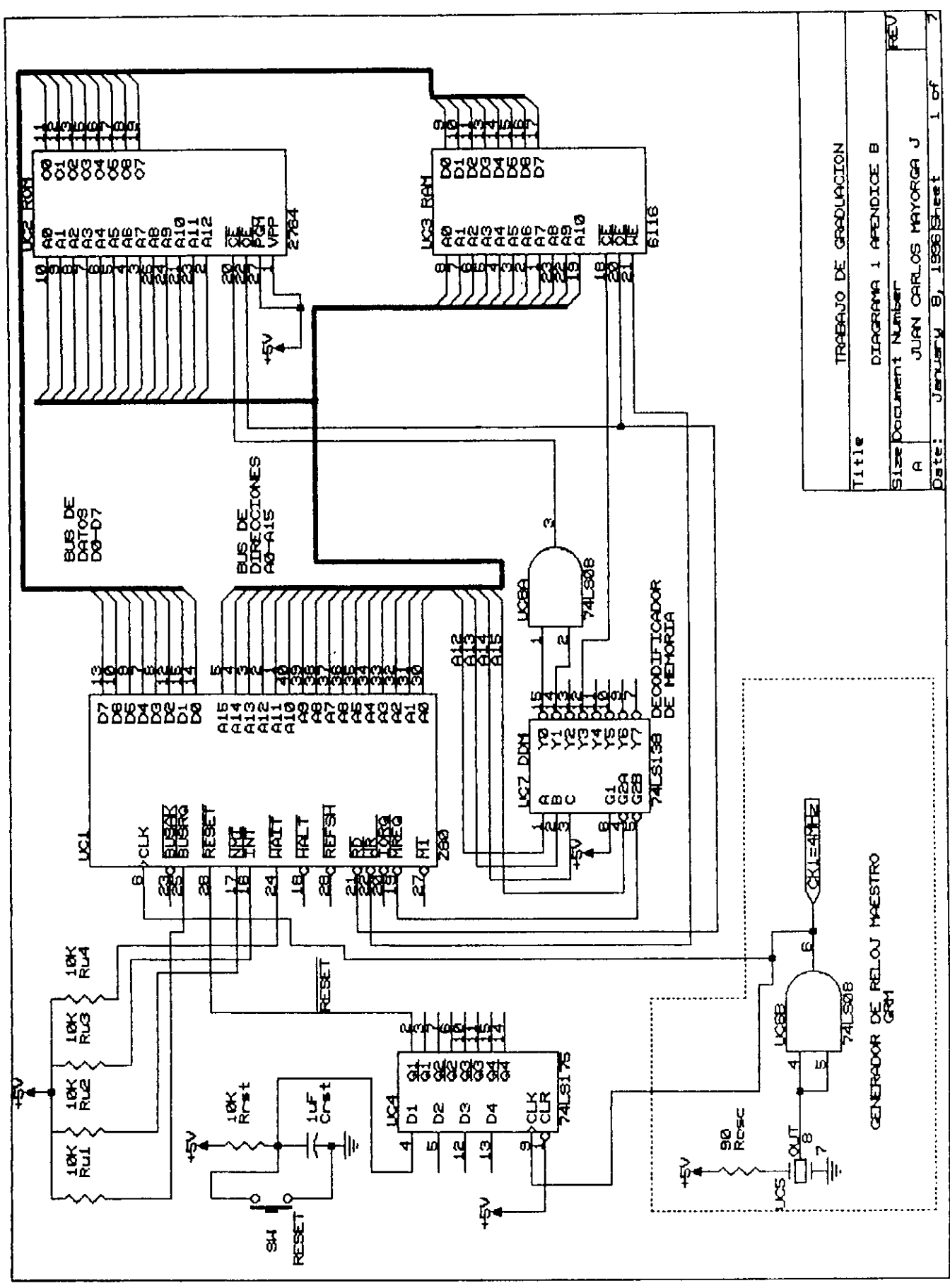
4 of 4

REV

JUAN CARLOS MAYORGA JUAREZ

APENDICE B

Diagramas de los circuitos
del SDPG



Title	TRABAJO DE GRADUACION
Size Document Number	DIAGRAMA 1 APENDICE B
Date:	JANUARY 8, 1996
Sheet	1 of 7

GENERADOR DE RELOJ MAESTRO

DECODIFICADOR DE MEMORIA

BUS DE DATOS D0-D7

BUS DE DIRECCIONES A0-A15

LC2 ROM

LC3 RAM

6118

74LS08

74LS138

74LS175

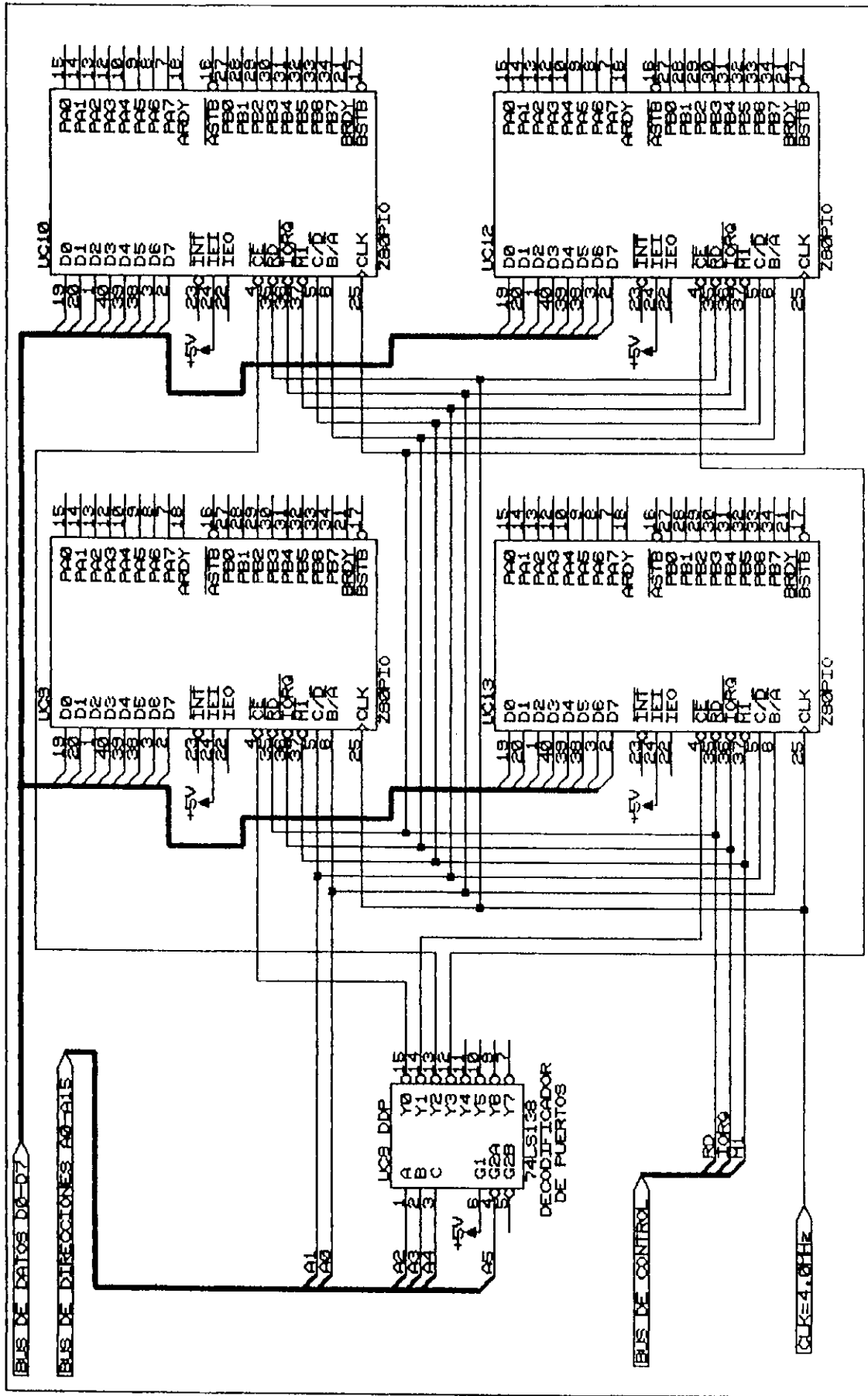
LC4

LC5

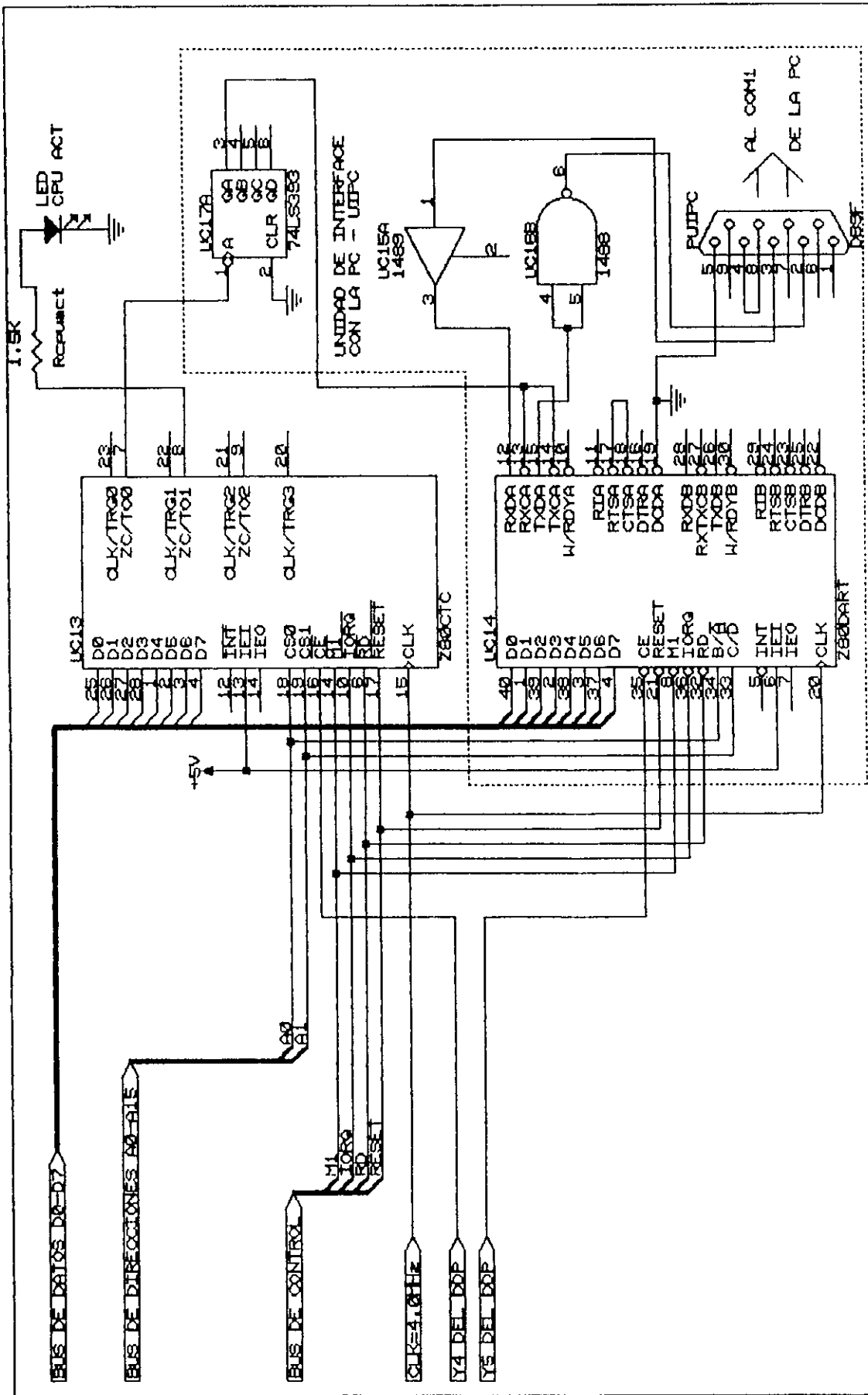
LC6

LC7

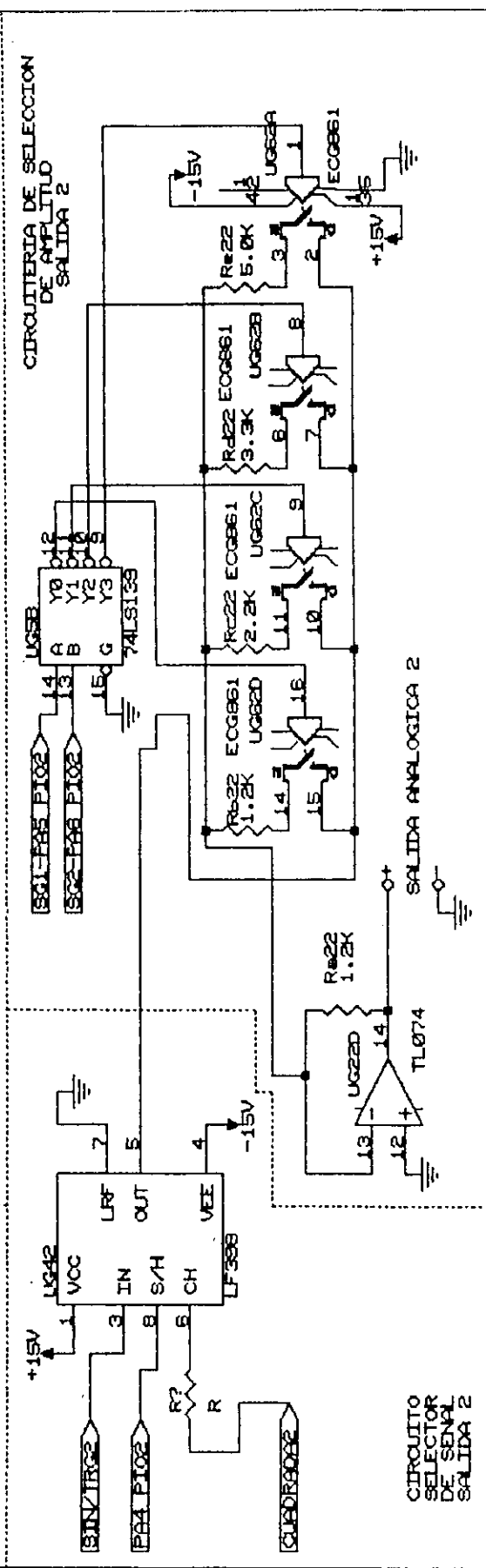
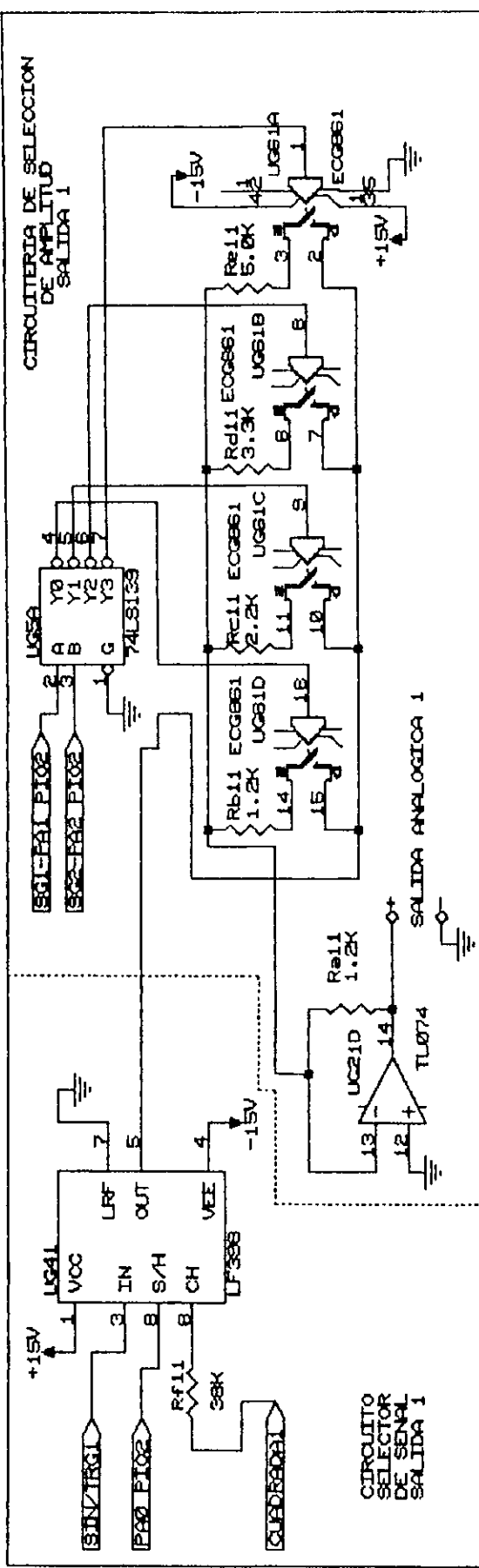
LC8



Título		TREBAJO DE GRADUACION
Size Document Number		FIGURA 2 APENDICE B
Date:		JUAN CARLOS MAYORGA J
REV		January 8, 1996 Sheet 2 of 7

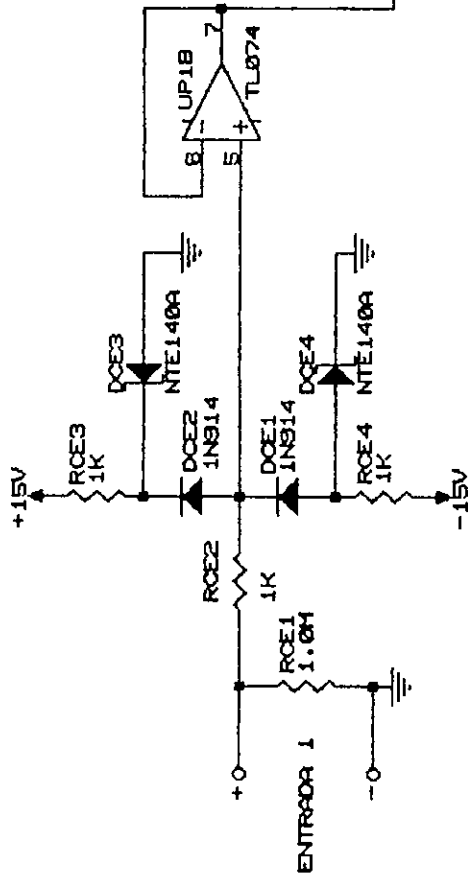


Title: TRABAJO DE GRADUACION
 Diagrama 3 APRENDICE B
 Size: Document Number
 A JUAN CARLOS MAYORGA J
 Date: January 8, 1996 Sheet 3 of 7



TRABAJO DE GRADUACION	
Title	DIAGRAMA 5 APENDICE B
Size Document Number	JUAN CARLOS MAYORGA J
REV	REV
Date:	January 9, 1995 Sheet 5 of 7

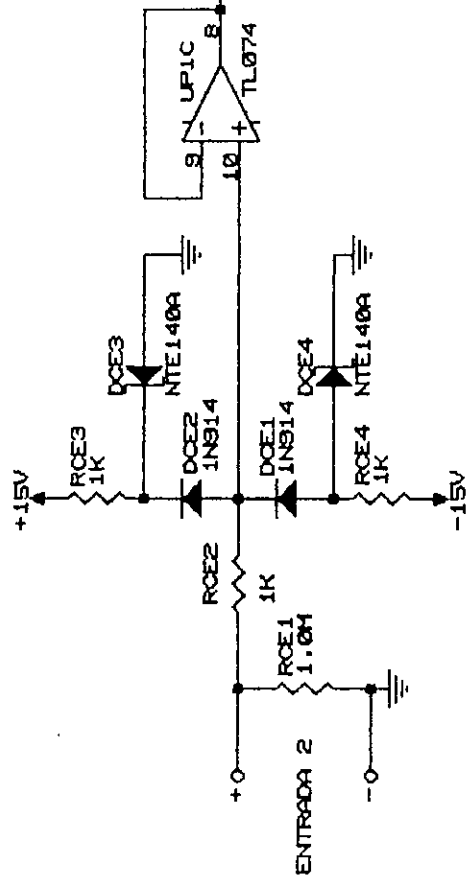
CIRCUITO DE ENTRADA PARA LA ENTRADA 1 DEL SOPG



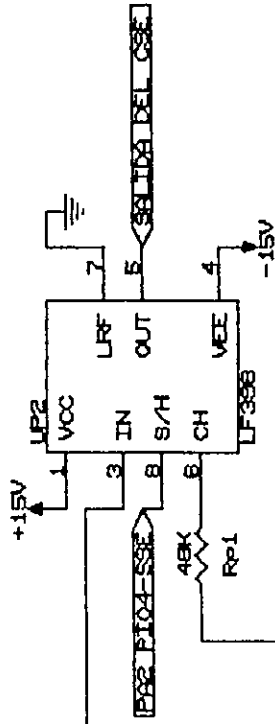
ENTRADA 1

ENTRADA 2

CIRCUITO DE ENTRADA PARA LA ENTRADA 2 DEL SOPG

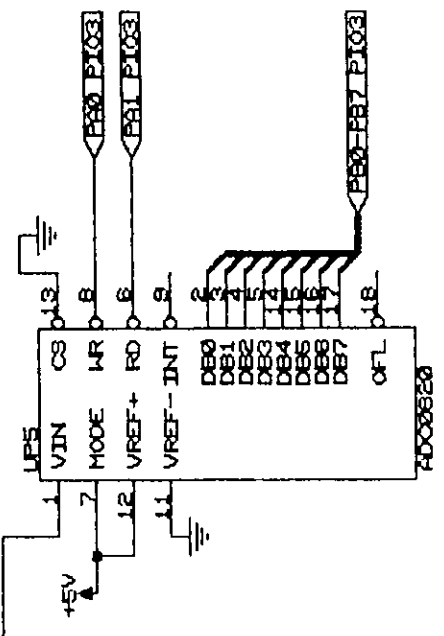


CIRCUITO DE SELECCION DE ENTRADA CSE

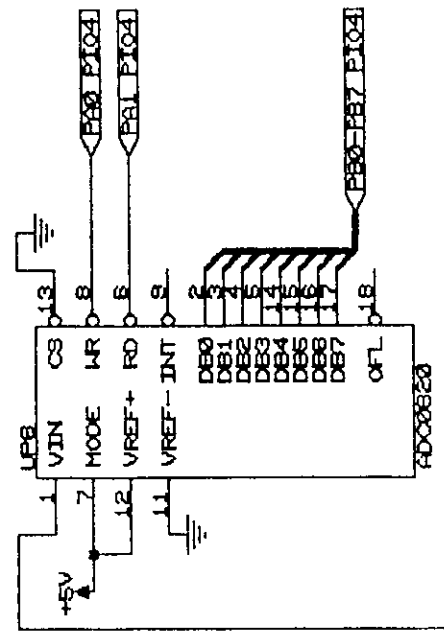


TRABAJO DE GRADUACION	
Title	DIAGRAMA 6 APENDICE B
Size	Document Number
A	JUAN CARLOS MAYORGA J
Date:	January 9, 1996
Sheet	6 of 7

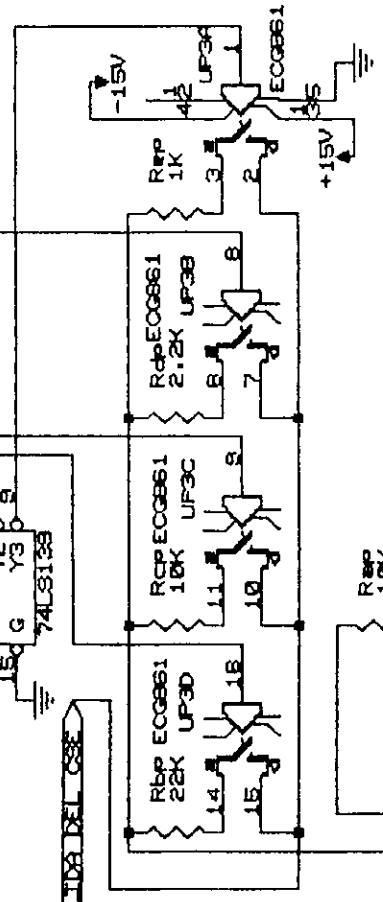
ADC PARA MUESTREO DE LA ENTRADA SELECCIONADA



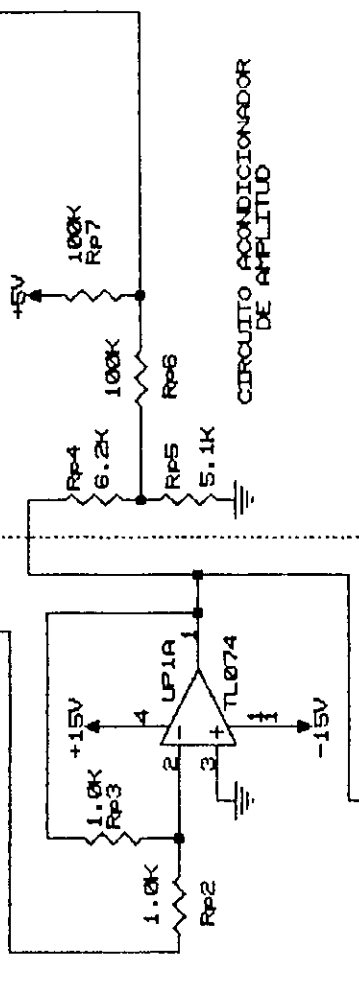
ADC PARA OBTENER EL VALOR DE LA FRECUENCIA DE LA SEÑAL EN LA ENTRADA SELECCIONADA



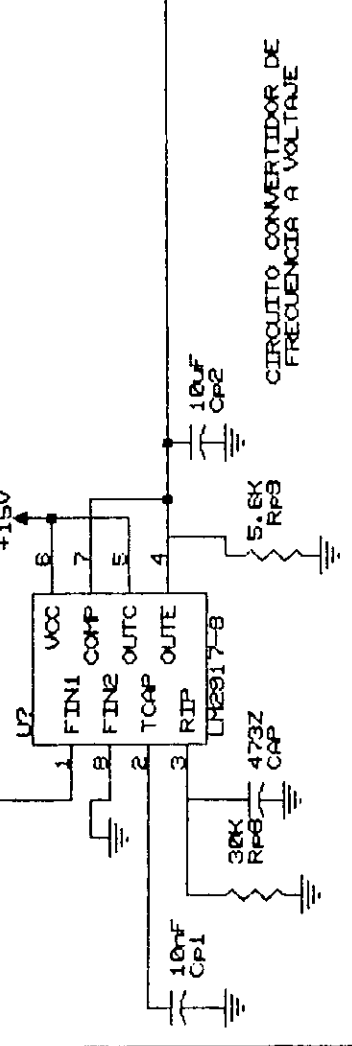
CIRCUITO DE SELECCION DE GANANCIA CSB



CIRCUITO ACONDICIONADOR DE AMPLITUD



CIRCUITO CONVERTIDOR DE FRECUENCIA A VOLTAJE



Title	TRABAJO DE GRADUACION
Size	DIAGRAMA 7 AFENDICE B
Document Number	JUAN CARLOS MAYORGA J
Date:	January 9, 1996 Sheet 7 of 7

APENDICE C

Listado del Programa de
Operación del SDPG

```

;Programa de Operación del SDFG
;Trabajo de Graduación
;Juan Carlos Mayorga Juarez

```

```

-----;
; DEFINICION DE LAS DIRECCIONES DE LOS PUERTOS DE LA UC
-----;

```

```

;Direcciones asignadas al P101

```

```

0000: ! 8 PAD1 EQU 00H ;Dirección del puerto A (datos)
0002: ! 9 PAC1 EQU 02H ;Dirección del puerto A (control)
0001: ! 10 PBD1 EQU 01H ;Dirección del puerto B (datos)
0003: ! 11 PBC1 EQU 03H ;Dirección del puerto B (control)

```

```

;Direcciones asignadas al P102

```

```

0004: ! 13 PAD2 EQU 04H ;Dirección del puerto A (datos)
0006: ! 14 PAC2 EQU 06H ;Dirección del puerto A (control)
0005: ! 15 PBD2 EQU 05H ;Dirección del puerto B (datos)
0007: ! 16 PBC2 EQU 07H ;Dirección del puerto B (control)

```

```

;Direcciones asignadas al P103

```

```

0008: ! 18 PAD3 EQU 08H ;Dirección del puerto A (datos)
000A: ! 19 PAC3 EQU 0AH ;Dirección del puerto A (control)
0009: ! 20 PBD3 EQU 09H ;Dirección del puerto B (datos)
000B: ! 21 PBC3 EQU 0BH ;Dirección del puerto B (control)

```

```

;Direcciones asignadas al P104

```

```

000C: ! 23 PAD4 EQU 0CH ;Dirección del puerto A (datos)
000E: ! 24 PAC4 EQU 0EH ;Dirección del puerto A (control)
000D: ! 25 PBD4 EQU 0DH ;Dirección del puerto B (datos)
000F: ! 26 PBC4 EQU 0FH ;Dirección del puerto B (control)

```

```

;Direcciones asignadas al CTC

```

```

0010: ! 28 CND EQU 08H ;Dirección del canal B
0011: ! 29 CH1 EQU 09H ;Dirección del canal 1
0012: ! 30 CN2 EQU 0AH ;Dirección del canal 2
0013: ! 31 CN3 EQU 0BH ;Dirección del canal 3

```

```

;Direcciones asignadas al DART

```

```

0014: ! 33 CAD EQU 04H ;Dirección del canal A (datos)
0015: ! 34 CBD EQU 05H ;Dirección del canal B (datos)
0016: ! 35 CAC EQU 06H ;Dirección del canal A (control)
0017: ! 36 CBC EQU 07H ;Dirección del canal B (control)

```

```

-----;
; DEFINICION DE LAS VARIABLES DE MEMORIA
-----;

```

```

3000: ! 40 TCAR EQU 3000H ; Variable para almacenar el caracter a enviar
3002: ! 41 RCAR EQU 3002H ; Variable para almacenar caracter recibido
3004: ! 42 RERR EQU 3004H ; Variable para almacenar errores
3006: ! 43 CBMA EQU 3006H ; Variable para almacenar el comando 1 enviado por la PC
3008: ! 44 COMB EQU 3008H ; Variable para almacenar el comando 2 enviado por la PC
300A: ! 45 COMC EQU 300AH ; Variable para almacenar el comando 3 enviado por la PC
300C: ! 46 COMD EQU 300CH ; Variable para almacenar el comando 4 enviado por la PC
300E: ! 47 COME EQU 300EH ; Variable para almacenar el comando 5 enviado por la PC
3010: ! 48 COMF EQU 3010H ; Variable para almacenar el comando 6 enviado por la PC
3012: ! 49 FRC1 EQU 3012H ; Valor de frecuencia medida en la entrada seleccionada
3014: ! 50 VART EQU 3014H ; Variable para selección de rutinas a ejecutar
3100: ! 51 MRAM EQU 3100H ; Dirección de Memoria para almacenar muestras

```

```

-----;
;INICIO DEL PD
-----;

```

```

0000: ! 55 ORG 0000H

```

```

0000:          ! 56 INICIO:
0000: 31F037 ! 57      LD  SP,37F0H  ;Definición del Puntero de Pila
;-----;
;INICIALIZACION DE PERIFERICOS
;-----;
; Inicialización del P101
0003: 3E0F ! 63      LD  A,0FH    ;Palabra de control para programar puertos
0005: D302 ! 64      OUT (PAC1),A ;Puerto A programado en Modo 0 (salida)
0007: D303 ! 65      OUT (PBC1),A ;Puerto B programado en Modo 0 (salida)
0009: D300 ! 66      OUT (PAD1),A ;Envío valor inicial al puerto A
000B: D301 ! 67      OUT (PBD1),A ;Envío valor inicial al puerto B
;-----;
; Inicialización del P102
000D: 3E0F ! 70      LD  A,0FH    ;Palabra de control para programar puertos
000F: D306 ! 71      OUT (PAC2),A ;Puerto A programado en Modo 0 (salida)
0011: D307 ! 72      OUT (PBC2),A ;Puerto B programado en Modo 0 (salida)
0013: 3E33 ! 73      LD  A,33H    ;Valor default ganancias/forma de onda salidas
0015: D304 ! 74      OUT (PAD2),A ;Envío valor inicial al puerto A
0017: AF ! 75      XOR  A       ;Valor default de ganancias de las entrada
0018: D305 ! 76      OUT (PBD2),A ;Envío valor inicial al puerto B
;-----;
; Inicialización del P103
001A: 3E0F ! 79      LD  A,0FH    ;Palabra de control para el puerto A
001C: D30A ! 80      OUT (PAC3),A ;Puerto A programado en Modo 1 (SALIDA)
001E: 3E03 ! 81      LD  A,03H    ;Deshabilitación del ADC para muestras
0020: D300 ! 82      OUT (PAD3),A ;Envío del valor inicial al puerto A
0022: 3E4F ! 83      LD  A,4FH    ;Palabra de control para el puerto B
0024: D30B ! 84      OUT (PBC3),A ;Puerto B programado en Modo 1 (Entrada)
;-----;
; Inicialización del P104
0026: 3E0F ! 87      LD  A,0FH    ;Palabra de control para el puerto A
0028: D30E ! 88      OUT (PAC4),A ;Puerto A programado en Modo 0 (salida)
002A: 3E03 ! 89      LD  A,03H    ;Deshabilitación del ADC para frecuencia
002C: D30C ! 90      OUT (PAD4),A ;Envío del valor inicial al puerto A
002E: 3E4F ! 91      LD  A,4FH    ;Palabra de control para el puerto B
0030: D30F ! 92      OUT (PBC4),A ;Puerto B programado en Modo 1 (Entrada)
;-----;
; Inicialización del CTC
0032: 3E07 ! 95      LD  A,07H    ;Envío de palabra de control
0034: D310 ! 96      OUT (CN0),A  ;al canal 0
0036: 3E0D ! 97      LD  A,0DH    ;Envío de la constante de tiempo
0038: D310 ! 98      OUT (CN0),A  ;al canal 0 /velocidad para el DART
003A: 3E07 ! 99      LD  A,07H    ;Envío de palabra de control
003C: D311 ! 100     OUT (CN1),A  ;al canal 1
003E: 3E68 ! 101     LD  A,68H    ;Envío de la constante de tiempo
0040: D311 ! 102     OUT (CN1),A  ;al canal 1
0042: 3E07 ! 103     LD  A,07H    ;Envío de palabra de control
0044: D312 ! 104     OUT (CN2),A  ;al canal 2
0046: 3E1A ! 105     LD  A,1AH    ;Envío de la constante de tiempo
0048: D312 ! 106     OUT (CN2),A  ;al canal 2
004A: 3E07 ! 107     LD  A,07H    ;Envío de palabra de control
004C: D313 ! 108     OUT (CN3),A  ;al canal 3
004E: 3E1A ! 109     LD  A,1AH    ;Envío de la constante de tiempo
0050: D313 ! 110     OUT (CN3),A  ;al canal 3

```

```

-----;
;Iniciación del DART
; 8 bits/caracter - 1 stop bit - sin paridad - clk x 1
0052: 3E10 ! 114 LD A,10H ;Se envía la palabra de reset al canal A
0054: D316 ! 115 OUT (CAC),A
0056: 3E04 ! 116 LD A,04H ;Se direcciona WR4
0058: D316 ! 117 OUT (CAC),A
005A: 3E04 ! 118 LD A,04H ;Se programa el reloj x1, 1 stop bits
005C: D316 ! 119 OUT (CAC),A ;sin paridad
005E: 3E03 ! 120 LD A,03H ;Se direcciona WR3
0060: D316 ! 121 OUT (CAC),A
0062: 3EC1 ! 122 LD A,0C1H ;8 bits/caracter recibido, sin autoenable
0064: D316 ! 123 OUT (CAC),A ;receptor habilitado
0066: 3E05 ! 124 LD A,05H ;Se direcciona WR5
0068: D316 ! 125 OUT (CAC),A
006A: 3E6A ! 126 LD A,06AH ;DTR=0, RTS=0, 8 bits/caracter
006C: D316 ! 127 OUT (CAC),A ;transmisor habilitado
006E: 3E01 ! 128 LD A,01H ;Se direcciona WR1
0070: D316 ! 129 OUT (CAC),A
0072: 3E00 ! 130 LD A,00H ;Se deshabilitan todas las interrupciones
0074: D316 ! 131 OUT (CAC),A
;+++++;
; RUTINA PRINCIPAL
;+++++;
;Iniciación de variables de memoria
0076: AF ! 136 XOR A ;Iniciación a 0 del Acumulador
0077: 320030 ! 137 LD (TCAR),A ;Iniciación a 0 de la variable TCAR
007A: 320230 ! 138 LD (RCAR),A ;Iniciación a 0 de la variable RCAR
007D: 320430 ! 139 LD (RERR),A ;Iniciación a 0 de la variable RERR
0080: 321230 ! 140 LD (FRC1),A ;Iniciación a 0 de la variable FRC1
0083: 3E0F ! 141 LD A,0FH
0085: D300 ! 142 OUT (PAD1),A ;Valor de frecuencia de la salida 1
0087: D301 ! 143 OUT (PBD1),A ;Valor de frecuencia de la salida 2
0089: AF ! 144 XOR A ;Borra el acumulador
;Rutina Principal
008A: ! 146 NE1
008A: CD4301 ! 147 CALL RCV_CAR ;Recepción de datos de la PC
008D: 3A0230 ! 148 LD A,(RCAR) ;Almacena en A el valor recibido
0090: 321430 ! 149 LD (VART),A ;Almacena en VART el valor recibido
0093: FE00 ! 150 CP 00H ;Compara valor recibido con 00H
0095: C2A300 ! 151 JP NZ,NE2 ;Si no es 00H bifurca a etiqueta NE2
0098: CD4301 ! 152 CALL RCV_CAR ;Recibe valor de frecuencia de la PC
009B: 3A0230 ! 153 LD A,(RCAR) ;Se almacena en A el valor
009E: D300 ! 154 OUT (PAD1),A ;Se modifica la frecuencia de la salida 1
00A0: C32C01 ! 155 JP NEF ;Bifurca a etiqueta NEF
00A3: ! 156 NE2
00A3: 321430 ! 157 LD (VART),A ;Almacena en VART el valor recibido
00A6: FE01 ! 158 CP 01H ;Compara valor recibido con 01H
00A8: C2B600 ! 159 JP NZ,NE3 ;Si no es 01H bifurca a etiqueta NE3
00AB: CD4301 ! 160 CALL RCV_CAR ;Recibe valor de amplitud de la PC
00AE: 3A0230 ! 161 LB A,(RCAR) ;Se almacena el valor en A
00B1: D304 ! 162 OUT (PAD2),A ;envía Amplitud y Forma de las Salidas
00B3: C32C01 ! 163 JP NEF ;Bifurca a etiqueta NEF
00B6: ! 164 NE3
00B6: 321430 ! 165 LD (VART),A ;Almacena en VART el valor recibido

```

```

0009: FED2 ! 166 CP 02H ;Compara valor recibido con 03H
000B: C2C900 ! 167 JP NZ,NE4 ;Si no es 03H bifurca a etiqueta NE5
000E: CD4301 ! 168 CALL RCV_CAR ;Recibe valor de frecuencia
00C1: 3A0230 ! 169 LD A,(RCAR) ;Almacena el valor en A
00C4: D301 ! 170 OUT (PBD1),A ;Valor de frecuencia de la salida 2
00C6: C32C01 ! 171 JP NEF ;Bifurca a etiqueta NEF
00C9: ! 172 NE4
00C9: 321430 ! 173 LD (VART),A ;Almacena en VART el valor recibido
00CC: FE03 ! 174 CP 03H ;
00CE: C2DD00 ! 175 JP NZ,NE5 ;
00D1: CD4301 ! 176 CALL RCV_CAR ;
00D4: 3A0230 ! 177 LD A,(RCAR) ;
00D7: 320630 ! 178 LD (COMA),A ;Almacena Ganancia Entrada 1 (00,01,02,03)
00DA: C32C01 ! 179 JP NEF ;Bifurca a etiqueta NEF
00DD: ! 180 NES
00DD: 321430 ! 181 LD (VART),A ;Almacena en VART el valor recibido
00EB: FE04 ! 182 CP 04H ;
00E2: C2F100 ! 183 JP NZ,NE6 ;
00E5: CD4301 ! 184 CALL RCV_CAR ;
00E8: 3A0230 ! 185 LD A,(RCAR) ;
00EB: 320630 ! 186 LD (COMB),A ;Almacena Ganancia Entrada 2 (00,01,02,03)
00EE: C32C01 ! 187 JP NEF ;Bifurca a etiqueta NEF
00F1: ! 188 NE6
00F1: 321430 ! 189 LD (VART),A ;Almacena en VART el valor recibido
00F4: FE05 ! 190 CP 05H ;
00F6: C20201 ! 191 JP NZ,NEA ;
00F9: CD9101 ! 192 CALL MX_IN1 ;Muestreo de la Entrada 1
00FC: CD0501 ! 193 CALL TMT_DAT ;Envío de mediciones a la PC
00FF: C32C01 ! 194 JP NEF ;Bifurca a etiqueta NEF
0102: ! 195 NEA
0102: 321430 ! 196 LD (VART),A ;Almacena en VART el valor recibido
0105: FE06 ! 197 CP 06H ;
0107: C21301 ! 198 JP NZ,NEB ;
010A: CDA201 ! 199 CALL MX_IN2 ;Muestreo de la Entrada 2
010D: CD0501 ! 200 CALL TMT_DAT ;Envío de mediciones a la PC
0110: C32C01 ! 201 JP NEF ;Bifurca a etiqueta NEF
0113: ! 202 NEB
0113: 321430 ! 203 LD (VART),A ;Almacena en VART el valor recibido
0116: FE07 ! 204 CP 07H ;
0118: C22101 ! 205 JP NZ,NEC ;
011B: C0C901 ! 206 CALL TMT_FRC ;Envío de frecuencia de la entrada 1
011E: C32C01 ! 207 JP NEF ;Bifurca a etiqueta NEF
0121: ! 208 NEC
0121: 321430 ! 209 LD (VART),A ;Almacena en VART el valor recibido
0124: FE08 ! 210 CP 08H ;Compara valor recibido con 08H
0126: C22C01 ! 211 JP NZ,NEF ;Si no es 08H bifurca a etiqueta NEF
0129: C0C901 ! 212 CALL TMT_FRC ;Envío de frecuencia de la entrada 2
012C: ! 213 NEF
012C: C30A00 ! 214 JP NE1 ;Bifurca a-etiqueta NE1

```

```

;+++++;
;
; SUB-RUTINAS
;+++++;
;=====;
;DELAY : RUTINA PARA GENERACION DE UN RETARDO DE 256us

```

```

;=====;
012F:      ! 222 DELAY
012F:  06FF ! 223      LD  B,0FFH  ;Almacenar en B duraci3n del retardo
0131:      ! 224 NEXT2
0131:  00      ! 225      NOP          ;No operaci3n
0132:  10FD ! 226      DJNZ NEXT2 ;Decrementar B y si no es cero ir a Next2
0134:  C9      ! 227      RET
;=====;
;TMT_CAR: RUTINA PARA TRANSMISION DE UN CARACTER
;=====;
0135:      ! 231 TMT_CAR
0135:  AF      ! 232      XOR  A
0136:      ! 233 TE1
0136:  D016 ! 234      IN   A,(CAC) ;Lectura de RR0
0138:  E604 ! 235      AND  04H  ;Chequeo del bit 2 de RR0 ( TX buffer vacio)
013A:  CA3601 ! 236     JP   Z,TE1 ;Si D2=0 entonces TX buffer no esta vacio
013D:  3A0030 ! 237     LD   A,(TCAR) ;Cargar en A caracter a transmitir
0140:  D314 ! 238     OUT (CAD),A ;Envio de caracter
0142:  C9      ! 239     RET      ;Retorno
;=====;
;RCV_CAR: RUTINA PARA RECEPCION DE UN CARACTER
;=====;
0143:      ! 243 RCV_CAR
0143:  AF      ! 244     XOR  A
0144:      ! 245 RE1
0144:  D016 ! 246     IN   A,(CAC) ;Lectura del registro RR0
0146:  E601 ! 247     AND  01H  ;Chequeo de bit D0 para ver si
0148:  CA4401 ! 248     JP   Z,RE1 ;se ha recibido un caracter
014B:  D014 ! 249     IN   A,(CAD) ;lectura del caracter recibido
014D:  3E0230 ! 250     LD   (RCAR),A ;almacenamiento en memoria
0150:  3E01 ! 251     LD   A,01H ;WR0 direcciona el registro RR1
0152:  D316 ! 252     OUT (CAC),A
0154:  D016 ! 253     IN   A,(CAC) ;Lectura del registro RR1
0156:  3E0430 ! 254     LD   (RERR),A ;Almacenar error
0159:  E670 ! 255     AND  70H  ;Chequeo de bits D6,D5,D4
015B:  CA6201 ! 256     JP   Z,RE2 ;Si no hay error retorna
015E:  3E30 ! 257     LD   A,30H ;Comando de reset de errores
0160:  D316 ! 258     OUT (CAC),A ;Envio del comando
0162:      ! 259 RE2
0162:  C9      ! 260     RET      ;Retorno
;=====;
;M_IN: RUTINA PARA MUESTREO DE LA ENTRADA SELECCIONADA
;=====;
0163:      ! 264 M_IN
0163:  06FF ! 265     LD   B,0FFH ;B contiene el numero de muestras a tomar
0165:  D0210031 ! 266     LD   IX,MRAM ;IX es el apuntador de memoria
0169:      ! 267 AX1
0169:  3E03 ! 268     LD   A,03H ;Inicializaci3n del ADC (WR=1 & RD=1)
016B:  D308 ! 269     OUT (PAD3),A ;Envio de comando al ADC
016D:  3D      ! 270     DEC  A      ;Se decrementa el valor de A (SOC=02h)
016E:  D308 ! 271     OUT (PAD3),A ;Activa la se1al de escritura WR (WR=0)
0170:  3D      ! 272     DEC  A      ;Se decrementa el valor de A (EOC=01H)
0171:  D308 ! 273     OUT (PAD3),A ;Activa la se1al de lectura RD
0173:  D009 ! 274     IN   A,(PBD3) ;Lectura de datos de la conversi3n
0175:  D07700 ! 275     LD   (IX),A ;Almacena la muestra en la posici3n IX

```

```

0170: DD23 ! 276      INC  IX      ;Incrementa el puntero de memoria
017A: 18ED ! 277      DJNZ AX1    ;Si B(0)0 bifurca a AX1 (continua muestreo)
017C: 3E03 ! 278      LD   A,03H  ;Desactivación del ADC (WR=1 &RD=1)
017E: D300 ! 279      DUT  (PAD3),A ;Envío de comando al ADC
0180: C9      ! 280      RET

;=====;
;M_FRC: Rutina para medicion de la frecuencia de la entrada activada ;
;=====;

0181: ! 284 M_FRC
0181: 3ED3 ! 285      LD   A,03H  ;Inicialización del ADC
0183: D30C ! 286      DUT  (PAD4),A ;Envío de comando al ADC
0185: 3D      ! 287      DEC  A      ;Se decrementa el valor de A (SOC=02h)
0186: D30C ! 288      OUT  (PAD4),A ;Activa la señal de escritura WR
0188: 3D      ! 289      DEC  A      ;Se decrementa el valor de A (EDC=01H)
0189: D30C ! 290      OUT  (PAD4),A ;Activa la señal de lectura RD
018B: D80D ! 291      IN   A,(PBD4) ;Lectura de datos de la conversión
018D: 321230 ! 292     LD   (FRC1),A ;Almacena el valor del voltaje en FRC1
0190: C9      ! 293      RET      ;Retorno

;=====;
;MX_IN1: Rutina para medicion de la entrada 1 ;
;=====;

0191: ! 297 MX_IN1
0191: 3A0630 ! 298     LD   A,(COMA) ;Lectura de la ganancia para la entrada 1
0194: E603 ! 299     AND  03H     ;Se obtiene el valor de la ganancia
0196: D305 ! 300     OUT  (PBD2),A ;Envío de comando de selección de ganancia
0198: CD2F01 ! 301     CALL DELAY   ;Retardo
019B: CD6301 ! 302     CALL M_IN    ;Obtener muestras de la entrada
019E: CD0101 ! 303     CALL M_FRC   ;Determinar frecuencia
01A1: C9      ! 304     RET      ;Retorno

;=====;
;MX_IN2: Rutina para medicion de la entrada 2 ;
;=====;

01A2: ! 308 MX_IN2
01A2: 3A0830 ! 309     LD   A,(COMB) ;Lectura de la ganancia para la entrada 2
01A5: E603 ! 310     AND  03H     ;Se obtiene el valor de la ganancia
01A7: F604 ! 311     OR   04H     ;Se selecciona la entrada 2
01A9: D305 ! 312     OUT  (PBD2),A ;Envío de comando de selección de ganancia
01AB: CD2F01 ! 313     CALL DELAY   ;Retardo
01AE: CD6301 ! 314     CALL M_IN    ;Obtener muestras de la entrada
01B1: CD0101 ! 315     CALL M_FRC   ;Determinar frecuencia
01B4: C9      ! 316     RET      ;Retorno

;=====;
;TMT_DAT: Rutina para transmision de los datos a la PC ;
;=====;

01B5: ! 320 TMT_DAT
01B5: 06FF ! 321     LD   B,0FFH ;Se almacena el numero de valores a enviar
01B7: D0210031 ! 322     LD   IX,MRAM ;IX es el apuntador de memoria
01B8: ! 323 TD2
01BB: D07E00 ! 324     LD   A,(IX) ;Se lee la muestra almacenada
01BE: 320030 ! 325     LD   (TCAR),A ;Se almacena en la variable de transmisión
01C1: CD3501 ! 326     CALL TMT_CAR ;Se envia la muestra
01C4: DD23 ! 327     INC  IX      ;Se incrementa el apuntador de memoria
01C6: 18F3 ! 328     DJNZ TD2    ;Si B(0)0 continua el Envío de muestras
01C8: C9      ! 329     RET      ;Retorno

;=====;

```

```
      ;TMT_FRC: RUTINA PARA TRANSMISION DEL VALOR DE LA FRECUENCIA MEDIDA ;  
      ;=====;  
01C9:      ! 333 TMT_FRC  
01C9:      AF      ! 334      XOR      A  
01CA:      3A1230 ! 335      LD      A,(FRC1) ;Se lee el valor de la frecuencia  
01CB:      320030 ! 336      LD      (TCAR),A ;Se almacena en la variable de transmisión  
01D0:      CD3501 ! 337      CALL   TMT_CAR ;Se envía el valor de la frecuencia  
01D3:      C9      ! 338      RET  
01D4:      ! 339      END
```

End of Assembly
No Errors

Symbol Table

Symbol	Value	Kind	Defined	References
AX1	0169	N	267	277
CAC	0016	=	35	115 117 119 121 123 125 127 129 131 234 246 252 253 258
CAD	0014	=	33	238 249
CBC	0017	=	36	
CBD	0015	=	34	
CN0	0010	=	28	96 98
CN1	0011	=	29	100 102
CN2	0012	=	30	104 106
CN3	0013	=	31	108 110
COMA	3006	=	43	178 298
COMB	3008	=	44	186 309
COMC	300A	=	45	
COMD	300C	=	46	
COME	300E	=	47	
COMF	3010	=	48	
DELAY	012F	N	222	301 313
FRC1	3012	=	49	140 292 335
INICIO:	0000	N	56	
MRAM	3100	=	51	266 322
MX_IN1	0191	N	297	192
MX_IN2	01A2	N	300	199
M_FRC	0101	N	284	303 315
M_IN	0163	N	264	302 314
NE1	000A	N	146	214
NE2	00A3	N	156	151
NE3	00B6	N	164	159
NE4	00C9	N	172	167
NE5	00DD	N	180	175
NE6	00F1	N	188	183
NEA	0102	N	195	191
NEB	0113	N	202	198
NEC	0121	N	208	205
NEF	012C	N	213	155 163 171 179 187 194 201 207 211
NEXT2	0131	N	224	226
PAC1	0002	=	9	64
PAC2	0006	=	14	71
PAC3	000A	=	19	80
PAC4	000E	=	24	88
PAD1	0000	=	8	66 142 154
PAD2	0004	=	13	74 162
PAD3	0008	=	18	82 269 271 273 279
PAD4	000C	=	23	90 286 288 290
PBC1	0003	=	11	65
PBC2	0007	=	16	72
PBC3	000B	=	21	84
PBC4	000F	=	26	92
PBD1	0001	=	10	67 143 170
PBD2	0005	=	15	76 300 312
PBD3	0009	=	20	274
PBD4	000D	=	25	291

Symbol	Value	Kind	Defined	References
RCAR	3002	=	41	138 148 153 161 169 177 185 250
KCV_CAR	0143	N	243	147 152 160 168 176 184
RE1	0144	N	245	248
RE2	0162	N	259	256
RERR	3004	=	42	139 254
TCAR	3000	=	40	137 237 325 336
TD2	0188	N	323	328
TE1	0136	N	233	236
TMT_CAR	0135	N	231	326 337
TMT_DAT	0185	N	320	193 200
TMT_FRC	0109	N	333	206 212
VART	3014	=	50	149 157 165 173 181 189 196 203 209

62 symbols defined

130 symbol references

APENDICE D

Diagramas de Flujo del
Programa de Operación
del SDPG

DIAGRAMA DE FLUJO DE LA RUTINA PRINCIPAL DEL PROGRAMA DE OPERACION DEL SDPG

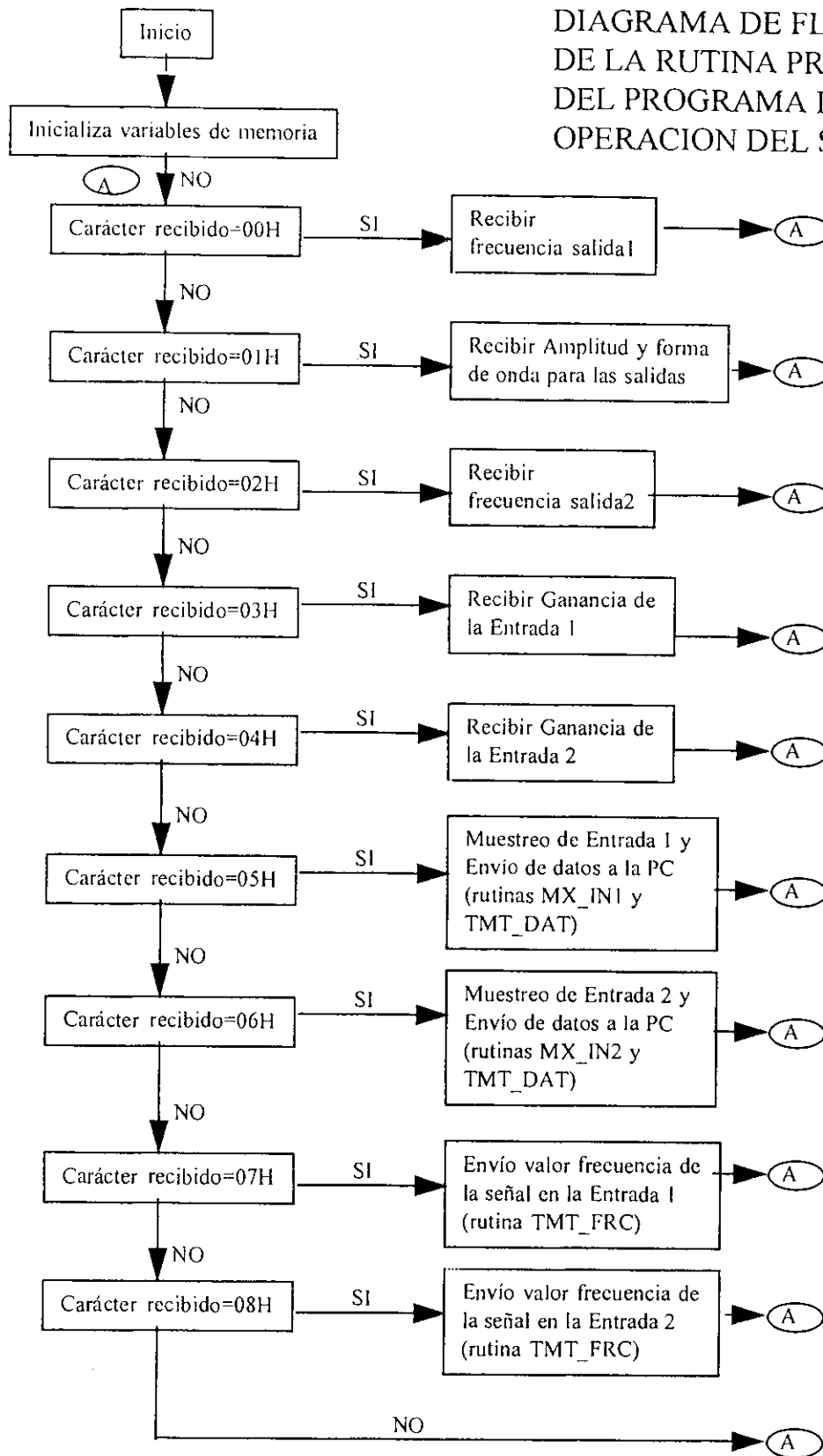


DIAGRAMA DE FLUJO DE LA
RUTINA DE MEDICION DE
LA ENTRADA SELECCIONADA
(M_IN)

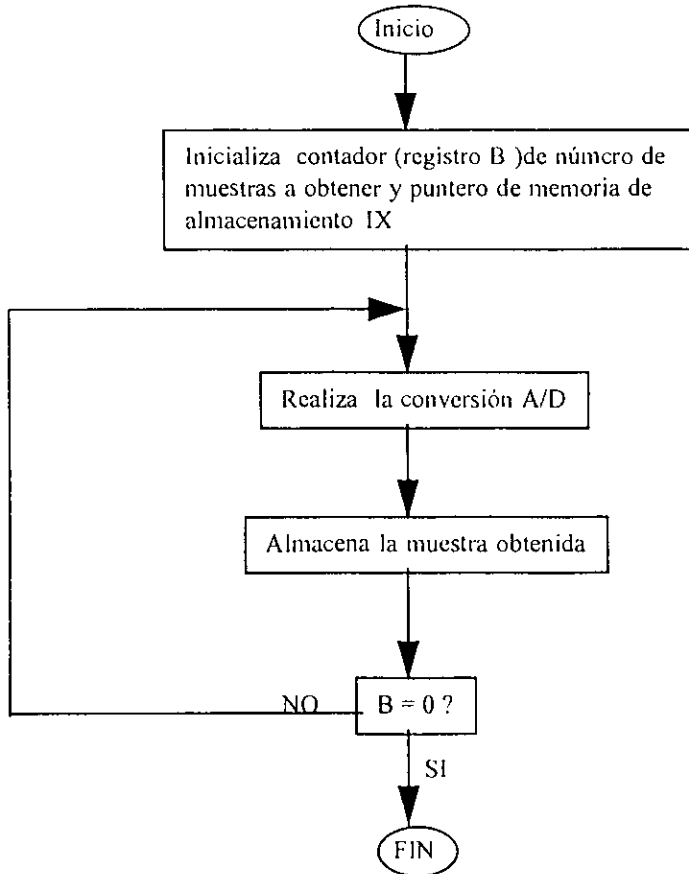


DIAGRAMA DE FLUJO DE LA
RUTINA DE MEDICION DE
FRECUENCIA DE LA ENTRADA
SELECCIONADA (M_FRC)

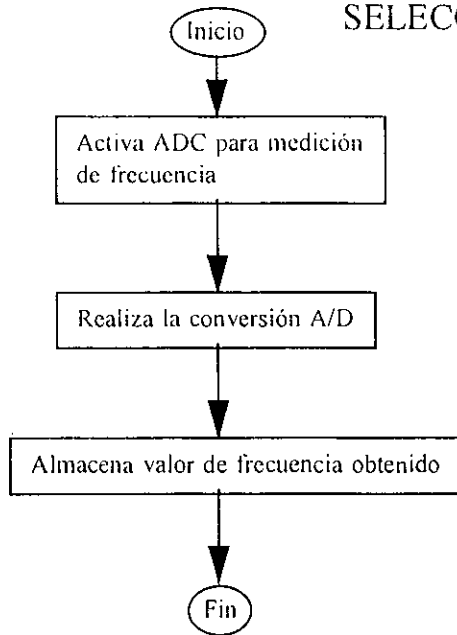


DIAGRAMA DE FLUJO DE
LAS RUTINAS DE MUESTREO
DE LAS ENTRADAS 1 & 2
(MX_IN1 & MX_IN2)

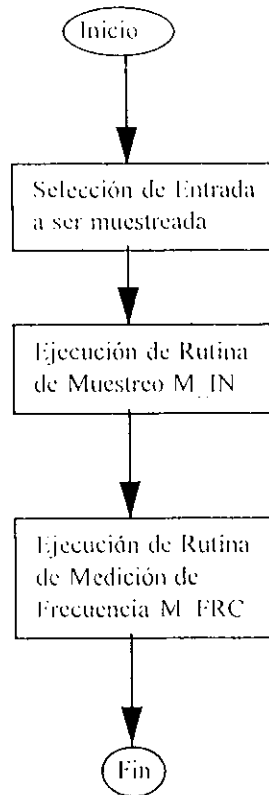


DIAGRAMA DE FLUJO DE LA RUTINA DE TRANSMISION DE DATOS A LA PC (TMT_DAT)

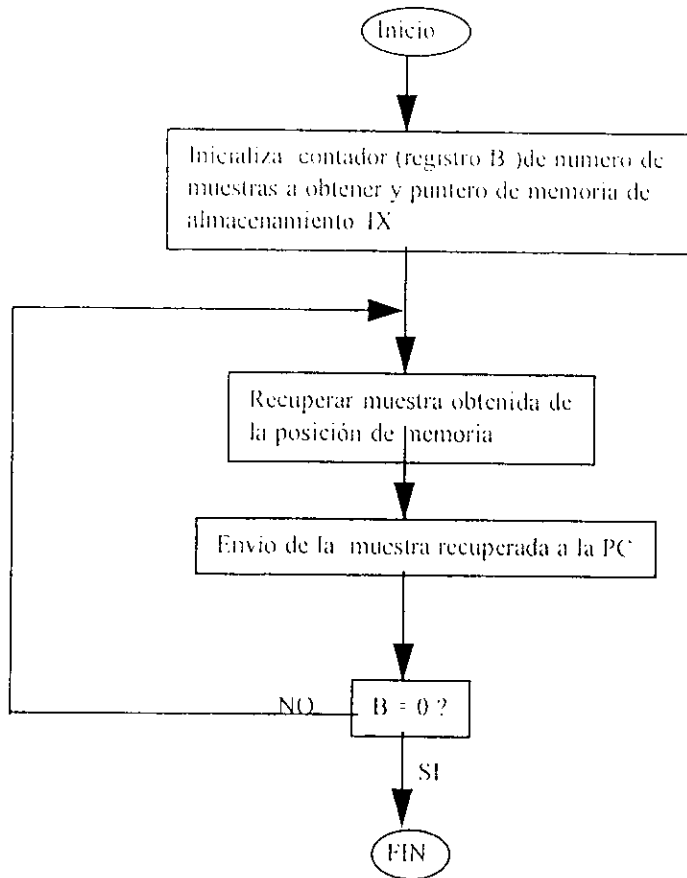
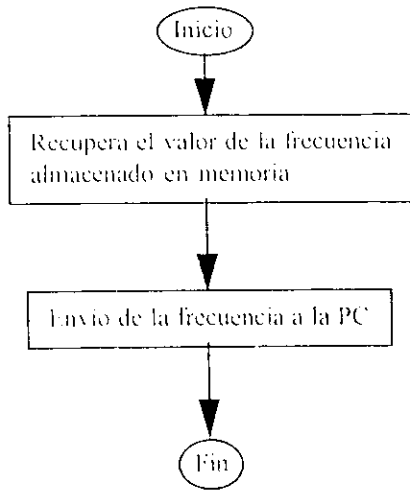


DIAGRAMA DE FLUJO DE LA
RUTINA DE TRANSMISION DEL
VALOR DE LA FRECUENCIA
MEDIDA (TMT_FRC)



APENDICE E

Listado del Programa de
Usuario del SDPG

```
{Universidad del Valle de Guatemala}
{Trabajo de Graduacion}
{Programa de Usuario}
{Autor: Juan Carlos Mayorga Juarez. @1995}
```

```
PROGRAM SDPG,
USES
CRT,DOS,GRAPH,PUERTOS3,IMPRIME;
{$V-}
```

```
{***** Definición de constantes del programa *****}
```

```
CONST
```

```
{Definición de caracteres a ser usados en menú}
```

```
Cr=#13;
Esc=#27;
Up=#72;
Dw=#80;
Rt=#77;
Lt=#75;
```

```
{Definición de strings de mensajes del menú}
```

```
Out1S:string[15]='Salida1: ';
Out2S:string[15]='Salida2: ';
In1S:string[20]='Ganancia Entrada 1: ';
In2S:string[20]='Ganancia Entrada 2: ';
OutGainS:string[15]='Ganancia ';
OutAmpS:string[15]='Amplitud ';
OutTipoS:string[15]='Forma de Onda ';
OutFreeS:string[15]='Frecuencia';
```

```
{Definición de los colores de las areas de visualización}
```

```
ColorIn1: Byte=White;
ColorIn2: Byte=White;
ColorEjes:Byte=Green;
ColorGrid:Byte=Blue;
ColorMenu:Byte=Green;
ColorDisp:Byte=White;
ColorNoDisp:Byte=Black;
```

```
{***** Definición de tipos de datos *****}
```

```
TYPE
```

```
ValOpMenu= record
```

```
  x1mn:integer;
  y1mn:integer;
  x2mn:integer;
  y2mn:integer;
  x3mn:integer;
  y3mn:integer;
  txt1mn:string[11];
  txt2mn:string[12];
  End;
```

```
xyset = set of 1..12;
```

```
ArMuestras = array [1..256] of integer;
```

```
ArComH = array [1..4] of byte;
```

```
ArComS = array [1..4] of string[10];
```

```
{***** Definición de variables *****}
```

```
VAR
```

```
GrafDriver,GrafMode,GrafErr,menupoint,frcl,frc2,z:integer;
frer1,frer2,maxr1,maxr2,minr1,minr2,promr1,promr2,gain:real;
key:char;
Muestreo,Continua:Boolean;
Opcion:array[1..12] of Boolean;
Archivo1,Archivo2: file of byte;
Arx1,Arx2: ArMuestras;
Max1,Max2,Min1,Min2,prom1,prom2,sum1,sum2:longint;
Max1s,Max2s,Min1s,Min2s,Prom1s,Prom2s,frcl,frc2s:string[12];
forma1,forma2,ampli1,ampli2,salidas,ampin1,ampin2:byte;
MatOpMenu:array[1..12] of ValOpMenu;
```

```

ax1mn,ax2mn,ax3mn,ay1mn,ay2mn,ay3mn:array[1..12] of integer;
atxt1mn,atxt2mn:array[1..12] of string[11];
xset1,xset2,xset3,yset1,yset2,yset3,yset4,tset1,tset2,tset3,tset4:xyset;
AForm1H,AForm2H,AGainH,AAmp1H,AAmp2H:ArComH;
AFormS,AAmpS,AGainS:ArComS;
AFrecH: array [1..256] of byte;
AFrecR: array [1..256] of real;

```

```

{***** Declaración de funciones y procedimientos *****}

```

```

FUNCTION add_commands(data1,data2,data3,data4:byte):byte;assembler:

```

```

asm;
mov al,data1
mov bl,data2
mov cl,data3
mov dl,data4
or cl,dl
or bl,cl
or al,bl
End;

```

```

PROCEDURE ModoGraf;

```

```

Begin
DirectVideo:=false;
SetGraphMode(GrafMode);
SetPalette(1,ColorGrid);
End;

```

```

PROCEDURE ModoTexto;

```

```

Begin
ClearDevice;
RestoreCrtMode;
DirectVideo:=True;
TextAttr:=7;
Clrscr;
Delay(100);
End;

```

```

PROCEDURE IniciaVarMenu; {Inicializa variables del menu de opciones}

```

```

Var
inx:integer;
Begin
xset1:={1,4,7,10};
xset2:={2,5,8,11};
xset3:={3,6,9,12};
yset1:={1,2,3};
yset2:={4,5,6};
yset3:={7,8,9};
yset4:={10,11,12};
tset1:={4,7};
tset2:={1,10};
tset3:={2,3,5,8,11,12};
tset4:={6,9};
{inicializa matrices}
for inx:=1 to 12 do
Begin
if (inx in xset1) then Begin
ax1mn[inx]:=30;
ax2mn[inx]:=100;
ax3mn[inx]:=34;
End;
if (inx in xset2) then Begin
ax1mn[inx]:=105;
ax2mn[inx]:=180;
ax3mn[inx]:=106;
End;
if (inx in xset3) then Begin
ax1mn[inx]:=185;
ax2mn[inx]:=275;

```

```

        ax3mn[inx]:=186;
    End;
if (inx in yset1) then Begin
    ay1mn[inx]:=400;
    ay2mn[inx]:=412;
    ay3mn[inx]:=402;
    End;
if (inx in yset2) then Begin
    ay1mn[inx]:=415;
    ay2mn[inx]:=427;
    ay3mn[inx]:=417;
    End;
if (inx in yset3) then Begin
    ay1mn[inx]:=432;
    ay2mn[inx]:=444;
    ay3mn[inx]:=434;
    End;
if (inx in yset4) then Begin
    ay1mn[inx]:=449;
    ay2mn[inx]:=461;
    ay3mn[inx]:=451;
    End;
if (inx in tset1) then atxt1mn[inx]:='00000000';
if (inx in tset2) then atxt1mn[inx]:='0000000000';
if (inx in tset3) then atxt1mn[inx]:='000000000000';
if (inx in tset4) then atxt1mn[inx]:='00000000000000';
End; {End for inx}
atxt2mn[1]:=' Grabar';
atxt2mn[2]:='Recuperar';
atxt2mn[3]:='Imprimir';
atxt2mn[4]:=' Format';
atxt2mn[5]:='Amplitud1';
atxt2mn[6]:='Frecuencia1';
atxt2mn[7]:=' Forma2';
atxt2mn[8]:='Amplitud2';
atxt2mn[9]:='Frecuencia2';
atxt2mn[10]:='Muestrea';
atxt2mn[11]:='Ganancia1';
atxt2mn[12]:='Ganancia2';
for inx:=1 to 12 do
    Begin
        with MatOpMenu[inx] do
            Begin
                x1mn:= ax1mn[inx];
                y1mn:= ay1mn[inx];
                x2mn:= ax2mn[inx];
                y2mn:= ay2mn[inx];
                x3mn:= ax3mn[inx];
                y3mn:= ay3mn[inx];
                tx1mn:=atxt1mn[inx];
                tx2mn:=atxt2mn[inx];
            End;
        End;
    End;
End;

```

End;

PROCEDURE IniciaVarGen; {Inicializa variables generales}

Var

q:integer;

Begin

AForm1H[1]:=\$03;AForm1H[2]:=\$01;AForm1H[3]:=\$02;

AForm2H[1]:=\$30;AForm2H[2]:=\$10;AForm2H[3]:=\$20;

AGainH[1]:=\$00;AGainH[2]:=\$01;AGainH[3]:=\$02;AGainH[4]:=\$03;

AAmp1H[1]:=\$00;AAmp1H[2]:=\$04;AAmp1H[3]:=\$08;AAmp1H[4]:=\$0C;

AAmp2H[1]:=\$00;AAmp2H[2]:=\$40;AAmp2H[3]:=\$80;AAmp2H[4]:=\$C0;

AFormS[1]:='Triangular';AFormS[2]:='Senoidal';AFormS[3]:='Cuadrada';

AAmpS[1]:='1';AAmpS[2]:='1/2';AAmpS[3]:='1/3';AAmpS[4]:='1/5';

AGainS[1]:='1/2';AGainS[2]:='1';AGainS[3]:='4';AGainS[4]:='8';

AFrecH[1]:=\$01;

```

AFreeR[1] := 3.0;
for q:=2 to 256 do
  Begin
    AFreeR[q] := AFreeR[q-1] + 0.3;
    AFreeI[q] := AFreeI[q-1] + $01;
  End;
End;

```

```

PROCEDURE IniciaGraf; { Inicializa modo grafico }
Begin
  GrafDriver := VGA;
  GrafMode := VGAFI;
  InitGraph(GrafDriver, GrafMode, 'c:\tempo');
End;

```

```

PROCEDURE InFrame(x1f,y1f,x2f,y2f:integer); { Grafica Marco para Entradas }
Begin
  SetColor(ColorEjes);
  SetLineStyle(SolidLn,0, NormWidth);
  Rectangle(x1f,y1f,x2f,y2f);
End;

```

```

PROCEDURE InGrid(xmg0,ymg0,ying1,ying2,ying3:integer); { Grafica Grid para Entradas }
var
  k,x,y,yinc:word;
Begin
  SetColor(ColorGrid);
  for k:=1 to 5 do
    Begin
      y:=ymg0+30*k;
      Line(21,y,534,y);
    End;
  for k:=1 to 15 do
    Begin
      x:=20+35*(k-1);
      Line((20+x),ying1,(20+x),ying2);
    End;
  SetLineStyle(UserbitLn,$8888, NormWidth);
  for k:=1 to 6 do
    Begin
      y:=ying3+30*(k-1);
      Line(21,y,534,y);
    End;
  for k:=1 to 14 do
    Begin
      x:=37+35*(k-1);
      Line((21+x),ying1,(21+x),ying2);
    End;
  SetLineStyle(SolidLn,0, NormWidth);
End;

```

```

PROCEDURE GrafVal(OrgY,Muestra:integer); { Grafica valor muestreado }
Var
  x,y:integer;
Begin
  x:=Getx;
  inc(x,2);
  y:=OrgY+90-round(Muestra/1.5);
  Lineto(x,y);
End;

```

```

PROCEDURE GrafIn(xg0,yg0:integer;Armtx:ArMuestras); { grafica muestras de la Entrada }
Var
  x0,y0,I:integer;
  a_in:array[1..256] of integer;
Begin
  SetColor(ColorIn2);
  x0:=xg0;

```

```

y0:=yg0;
for l:=1 to 254 do
  Begin
    a_in[l]:=Armr[l];
  End;
Moveto(x0,y0);
for l:=1 to 254 do
  Begin
    GrafVal(y0,a_in[l]);
  End;
End;

```

```

PROCEDURE DisplInfoCom(xdc0,ydc0:integer;dcs:string;decolor:integer);

```

```

Begin
case decolor of
  0:Begin
    setcolor(Black);
    Moveto(xdc0,ydc0);
    outtext(dcs);
  End;
  1:Begin
    setcolor(White);
    Moveto(xdc0,ydc0);
    outtext(dcs);
  End;
End;
End;

```

```

PROCEDURE ValMedIn(yvm1,yvm2,yvm3,yvm4,yvm5:integer;txtvm:string);

```

```

Begin
DisplInfoCom(550,yvm1,txtvm,1);
DisplInfoCom(542,yvm2,'Max (V):',1);
DisplInfoCom(542,yvm3,'Min (V):',1);
DisplInfoCom(542,yvm4,'Prom (V):',1);
DisplInfoCom(542,yvm5,'Frec (Hz):',1);
End;

```

```

PROCEDURE BorraMuestras;

```

```

var
co:integer;
Begin
for co:=1 to 254 do
  Begin
    Arx1[co]:=130;
    Arx2[co]:=130;
  End;
End;

```

```

PROCEDURE DisplInfoM(ydi1,ydi2,ydi3,ydi4:integer;vardi1,vardi2,vardi3,vardi4:string;dicolor:integer);

```

```

Begin
DisplInfoCom(540,ydi1,vardi1,dicolor);
DisplInfoCom(540,ydi2,vardi2,dicolor);
DisplInfoCom(540,ydi3,vardi3,dicolor);
DisplInfoCom(540,ydi4,vardi4,dicolor);
End;

```

```

PROCEDURE CaleValM(Arcal:ArMuestras;var Maxcal,Mincal,Sumcal:longint);

```

```

var
Cocal:integer;
Begin
Maxcal:=0;
Mincal:=255;
Sumcal:=0;
for cocal:=1 to 254 do
  Begin
    if Maxcal<Arcal[cocal] then Maxcal:=Arcal[cocal];
    if Mincal>Arcal[cocal] then Mincal:=Arcal[cocal];
    sumcal:=sumcal+Arcal[cocal];
  End;

```

```
End;  
End;
```

```
PROCEDURE DetGainIn(gainin:byte;var igainn:real);  
Begin  
case gainin of  
$00:igainn:=2;  
$01:igainn:=1;  
$02:igainn:=1/4;  
$03:igainn:=1/8;  
End;  
End;
```

```
PROCEDURE ProcesaMuestras1;  
var  
co:integer;  
Begin  
DispInfoM(50,85,120,155,Max1s,Min1s,Prom1s,Fre1s,0);  
CalcValM(Arx1,Max1,Min1,Sum1);  
Fre1:=round((48.35*fre1)+1);  
prom1:=round(sum1/254);  
gain:=1;  
DetGainIn(ampin1,gain);  
if prom1>=130 then prom1:=((prom1*0.223)-29)*gain  
else prom1:=((prom1*0.212)-27.6)*gain;  
if max1>=130 then max1:=((max1*0.223)-29)*gain  
else max1:=((max1*0.212)-27.6)*gain;  
if min1>=130 then min1:=((min1*0.223)-29)*gain  
else min1:=((min1*0.212)-27.6)*gain;  
str(Maxr1:10:2,max1s);  
str(Minr1:10:2,min1s);  
str(Promr1:10:2,prom1s);  
str(Frer1:10:0,fre1s);  
DispInfoM(50,85,120,155,Max1s,Min1s,Prom1s,Fre1s,1);  
End;
```

```
PROCEDURE ProcesaMuestras2;  
var  
co:integer;  
Begin  
DispInfoM(265,300,335,370,Max2s,Min2s,Prom2s,Fre2s,0);  
CalcValM(Arx2,Max2,Min2,Sum2);  
Fre2:=(48.35*fre2)+1;  
prom2:=round(sum2/254);  
gain:=1;  
DetGainIn(ampin2,gain);  
if prom2>=130 then prom2:=((prom2*0.223)-29)*gain  
else prom2:=((prom2*0.212)-27.6)*gain;  
if max2>=130 then max2:=((max2*0.223)-29)*gain  
else max2:=((max2*0.212)-27.6)*gain;  
if min2>=130 then min2:=((min2*0.223)-29)*gain  
else min2:=((min2*0.212)-27.6)*gain;  
str(Maxr2:10:2,max2s);  
str(Minr2:10:2,min2s);  
str(Promr2:10:2,prom2s);  
str(Frer2:10:0,fre2s);  
DispInfoM(265,300,335,370,Max2s,Min2s,Prom2s,Fre2s,1);  
End;
```

```
PROCEDURE OpSel(iops:integer);  
Var  
x1sl,y1sl,x2sl,y2sl,x3sl,y3sl:integer;  
txt1sl,txt2sl:string[11];  
Begin  
with MatOpMenu[iops] do  
Begin  
x1sl:=x1mn;  
y1sl:=y1mn;
```

```

x2sl:=x2mn;
y2sl:=y2mn;
x3sl:=x3mn;
y3sl:=y3mn;
txt1sl:=txt1mn;
txt2sl:=txt2mn;
End;
SetColor(White);
Rectangle(x1sl,y1sl,x2sl,y2sl);
DispInfoCom(x3sl,y3sl,txt1sl,1);
DispInfoCom(x3sl,y3sl,txt2sl,0);
End;

```

```

PROCEDURE OpNorm(iopn:integer);
Var
x1on,y1on,x2on,y2on,x3on,y3on:integer;
txt1on,txt2on:string[11];
Begin
with MatOpMenu[iopn] do
  Begin
    x1on:=x1mn;
    y1on:=y1mn;
    x2on:=x2mn;
    y2on:=y2mn;
    x3on:=x3mn;
    y3on:=y3mn;
    txt1on:=txt1mn;
    txt2on:=txt2mn;
  End;
SetColor(White);
Rectangle(x1on,y1on,x2on,y2on);
DispInfoCom(x3on,y3on,txt1on,0);
DispInfoCom(x3on,y3on,txt2on,1);
End;

```

```

PROCEDURE DrawMenu;
Begin
if Opcion[1]=true then OpSel(1)
  else OpNorm(1);
if Opcion[2]=true then OpSel(2)
  else OpNorm(2);
if Opcion[3]=true then OpSel(3)
  else OpNorm(3);
if Opcion[4]=true then OpSel(4)
  else OpNorm(4);
if Opcion[5]=true then OpSel(5)
  else OpNorm(5);
if Opcion[6]=true then OpSel(6)
  else OpNorm(6);
if Opcion[7]=true then OpSel(7)
  else OpNorm(7);
if Opcion[8]=true then OpSel(8)
  else OpNorm(8);
if Opcion[9]=true then OpSel(9)
  else OpNorm(9);
if Opcion[10]=true then OpSel(10)
  else OpNorm(10);
if Opcion[11]=true then OpSel(11)
  else OpNorm(11);
if Opcion[12]=true then OpSel(12)
  else OpNorm(12);
End;

```

```

PROCEDURE ValuaOp;
var
zp:integer;
Begin
Case MenuPoint of

```

```
1: Begin
  Opcion[1]:=True;
  For zp:=2 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
  End;
2: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[2]:=True;
  End;
3: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[3]:=True;
  End;
4: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[4]:=True;
  End;
5: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[5]:=True;
  End;
6: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[6]:=True;
  End;
7: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[7]:=True;
  End;
8: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[8]:=True;
  End;
9: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[9]:=True;
  End;
10: Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
```

```

    Opcion[10]=True
  End;
11:Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[11]:=True
  End;
12:Begin
  For zp:=1 to 12 do
    Begin
      Opcion[zp]:=False;
    End;
    Opcion[12]:=True
  End;
End; {case}
End;

```

```

PROCEDURE SetFrecVal(fout:integer;OFrecS:string);
var
  seg3:boolean;
  q:integer;
  hval:byte;
  sval:string[15];
Begin
  seg3:=true;
  DispInfoCom(300,400,OFrecS,1);
  q:=1;
  hval:=AFrech[q];
  str(AFrecR[q]:12:2,sval);
  DispInfoCom(450,400,sval,1);
  DispInfoCom(570,400,'kHz',1);
  repeat
  case readkey of
    Up:if q<256 then Begin
      q:=q+1;
      DispInfoCom(450,400,sval,0);
      hval:=AFrech[q];
      str(AFrecR[q]:12:2,sval);
      DispInfoCom(450,400,sval,1);
      End;
    Dv:if q>1 then Begin
      q:=q-1;
      DispInfoCom(450,400,sval,0);
      hval:=AFrech[q];
      str(AFrecR[q]:12:2,sval);
      DispInfoCom(450,400,sval,1);
      End;
    Cr: Begin
      case fout of
        1:Begin {envio frecuencia salida 1}
          writecom($00);
          writecom(hval);
          seg3:=false;
          End;
        2:Begin {envio frecuencia salida 2}
          writecom($02);
          writecom(hval);
          seg3:=false;
          End;
      End;
    End;
  Esc: seg3:=false
  End {case readkey}
  until seg3=false;
  DispInfoCom(300,400,OFrecS,0);
  DispInfoCom(450,400,sval,0);

```

```
DispInfoCom(570,400,'kHz',0);
End;
```

```
PROCEDURE SetComVal(ArI:ArComH;ArS:ArComS;ComS:string;imax,lop:integer);
```

```
var
seg:boolean;
q,m:integer;
hval:byte;
sval:string[10];
Begin
seg:=true;
DispInfoCom(300,400,ComS,1);
q:=1;
hval:=ArI[q];
sval:=ArS[q];
DispInfoCom(480,400,sval,1);
repeat
case readkey of
Up:if q<Imax then Begin
q:=q+1;
DispInfoCom(480,400,sval,0);
hval:=ArI[q];
sval:=ArS[q];
DispInfoCom(480,400,sval,1);
End;
Dw:if q>1 then Begin
q:=q-1;
DispInfoCom(480,400,sval,0);
hval:=ArI[q];
sval:=ArS[q];
DispInfoCom(480,400,sval,1);
End;
Cr:Begin
case lop of
1: Begin
forma1:=hval;
salidas:=add_commands(forma1,ampli1,forma2,ampli2);
writecom($01);
writecom(salidas);
seg:=false;
End;
2: Begin
ampli1:=hval;
salidas:=add_commands(forma1,ampli1,forma2,ampli2);
writecom($01);
writecom(salidas);
seg:=false;
End;
3: Begin
writecom($03);
writecom(hval);
ampli1:=hval;
seg:=false;
End;
4: Begin
forma2:=hval;
salidas:=add_commands(forma1,ampli1,forma2,ampli2);
writecom($01);
writecom(salidas);
seg:=false;
End;
5: Begin
ampli2:=hval;
salidas:=add_commands(forma1,ampli1,forma2,ampli2);
writecom($01);
writecom(salidas);
seg:=false;
End;
End;
```

```

6 Begin
  writecom($04);
  writecom(hval);
  ampin2:=hval;
  seg:=false;
End;
End;
Ese: seg:=false
End {case readkey}
until seg=false;
DispInfoCom(300,400,ComS,0);
DispInfoCom(480,400,sval,0);
End;

```

PROCEDURE Borrar;

```

Begin
Cleardevice;
InFrame(20,5,535,185);      {Marco visualizacion entrada 1}
InGrid(20,5,6,184,20);      {Grid entrada 1}
InFrame(20,210,535,390);    {Marco area visualizacion entrada 2}
InGrid(225,210,211,389,225); {Grid Entrada 2}
ValMedIn(10,35,70,105,140,'Entrada 1:');
ValMedIn(215,250,285,320,355,'Entrada 2:');
DrawMenu;
End;

```

PROCEDURE Salvar;

```

var
  cont,co:integer;
  dato:byte;
  Atem:array[1..514] of byte;
  NomArc:string[12];
Begin
  DispInfoCom(300,400,'Grabar en:',1);
  NomArc:=GphReadLn(300,420,'filedat.dat');
  If Length(NomArc)= 0 Then
  Begin
    DispInfoCom(300,400,'Grabar en:',0);
    Exit;
  End;
  SetColor(white);
  for cont:=1 to 256 do
    Atem[cont]:=Arx1[cont];
  for cont:=1 to 256 do
    Atem[cont+256]:=Arx2[cont];
  {$+}
  ChDir('c:\temp');
  assign(archivo1,NomArc);
  rewrite(archivo1);
  for cont:=1 to 514 do
    write(archivo1,Atem[cont]);
  close(archivo1);
  {$+}
  DispInfoCom(300,400,'Grabar en:',0);
End;

```

PROCEDURE Recuperar;

```

var
  cont:integer;
  dato:byte;
  NomArc:string[12];
  Atem:array [1..514] of byte;
Begin
  DispInfoCom(300,400,'Recuperar Datos de:',1);
  NomArc:=GphReadLn(300,420,'filedat.daf');
  If Length(NomArc)= 0 Then

```

```

Begin
  DispInfoCom(300,400,'Recuperar Datos de',0);
  Exit;
End;
ChDir('C:\tempo');
assign(archivo1,NomArc);
{$I-}
reset(archivo1);
{$I+}
if IOResult=0 then
  Begin
    for cont:=1 to 514 do
      Begin
        read(archivo1,dato);
        ATem[cont]:=dato;
      End;
    close(archivo1);
    for cont:=1 to 256 do
      Arx1[cont]:=ATem[cont];
      fre1:=Arx1[255];
      ampin1:=Arx1[256];
    for cont:=257 to 514 do
      Arx2[cont-256]:=ATem[cont];
      fre2:=Arx2[255];
      ampin2:=Arx2[256];
    End
  else Begin
    DispInfoCom(300,430,'No encontrado',1);
    delay(2000);
    DispInfoCom(300,430,'No encontrado',0);
    End;
  DispInfoCom(300,400,'Recuperar Datos de',0);
  Borrar;
  GrafIn(21,90,Arx1);
  GrafIn(21,300,Arx2);
  ProcesaMuestras1;
  ProcesaMuestras2;
End;

```

PROCEDURE RxMuestras;

```

var
ke,ix,jx:integer;
Begin
setcolor(white);
  BorraMuestras;
  DispInfoCom(300,430,'Recibiendo Datos',1);
  writecom($05);
  jx:=-1;
  ke:=0;
  REPEAT
  ix:=LeerCom(jx);
  arx1[ke]:=ix;
  ke:=ke+1;
  UNTIL ix=-1;
  writecom($07);
  ix:=Leercom(jx);
  if ix=-1 then fre1:=0
    else fre1:=ix;
  Arx1[255]:=fre1;
  Arx1[256]:=ampin1;
  writecom($06);
  jx:=-1;
  ke:=0;
  REPEAT
  ix:=LeerCom(jx);
  arx2[ke]:=ix;
  ke:=ke+1;
  UNTIL ix=-1;

```

```

writecom($08);
ix:=Leecom(jx);
if ix=-1 then fre2:=0
  else fre2:=ix;
Arx2[255]:=fre2;
Arx2[256]:=ampin2;
DispInfoCom(300,430,'RecibiEndo Datos',0);
SetColor(ColorGrid);
GrafIn(21,90,Arx1);
ProcesaMuestras1;
GrafIn(21,300,Arx2);
ProcesaMuestras2;
End;

```

PROCEDURE SetMuestreo;

```

Var
sig:boolean;
co integer;
Mxval:array[1..2] of string[12];
Mxrel:array[1..2] of boolean;
Begin
sig:=true;
Mxrel[1]:=True;
Mxrel[2]:=False;
MxVal[1]:='Muestreo=On';
MxVal[2]:='Muestreo=Off';
if Muestreo=True then Begin
  co:=1;
  DispInfoCom(300,400,MxVal[co],1);
  End;
if Muestreo=False then Begin
  co:=2;
  DispInfoCom(300,400,MxVal[co],1);
  End;
Repeat
Case Readkey of
Up: Begin
  if co<2 then Begin
    DispInfoCom(300,400,MxVal[co],0);
    co:=co+1;
    DispInfoCom(300,400,MxVal[co],1);
    End;
  End;
Dw: Begin
  if co>1 then Begin
    DispInfoCom(300,400,MxVal[co],0);
    co:=co-1;
    DispInfoCom(300,400,MxVal[co],1);
    End;
  End;
Cr: Begin
  Muestreo:=MxRel[co];
  sig:=false;
  End;
End;
until sig=false;
DispInfoCom(300,400,MxVal[co],0);
{Borrar;}
if Muestreo=True then RxMuestras;
SetColor(white);
End;

```

PROCEDURE Imprimir;

```

Begin
ImprimeGrafica;
End;

```

PROCEDURE ExecOp;

```

Begin
  Case MenuPoint of
    1: Salvar;
    2: Recuperar;
    3: Imprimir;
    4: SetComVal(AForm1H,AFormS.OutTipoS+Out1S,3,1);
    5: SetComVal(AAmp1H,AAmpS.OutGainS+Out1S,4,2);
    6: SetFreeVal(1,Out1recS+''+Out1S);
    7: SetComVal(AForm2H,AFormS.OutTipoS+Out2S,3,4);
    8: SetComVal(AAmp2H,AAmpS.OutGainS+Out2S,4,5);
    9: SetFreeVal(2,Out1recS+''+Out2S);
    10: SetMuestreo;
    11: SetComVal(AGam1H,AGainS.In1S,4,3);
    12: SetComVal(AGam1H,AGainS.In2S,4,6);
  End;{case}
End.

```

PROCEDURE ChMenu;

```

Begin
  case Readkey of
    Rt:Begin
      If MenuPoint<12 then
        Begin
          MenuPoint:=MenuPoint+1;
          ValuaOp;
          DrawMenu;
        End
      else
        Begin
          MenuPoint:=12;
          ValuaOp;
          DrawMenu;
        End;
      End;
    Lt:Begin
      If MenuPoint>1 then MenuPoint:=MenuPoint-1;
      ValuaOp;
      DrawMenu;
      End;
    Up:Begin
      If MenuPoint>3 then MenuPoint:=MenuPoint-3;
      ValuaOp;
      DrawMenu;
      End;
    Dw:Begin
      If MenuPoint<10 then MenuPoint:=MenuPoint+3;
      ValuaOp;
      DrawMenu;
      End;
    Cr: ExceOp;
    Esc: continua:=false;
  End;
End;

```

PROCEDURE MenuVent;

```

Var
  tecla.char;
Begin
  Muestreo:=false;
  continua:=true;
  MenuPoint:=1;
  ValuaOp;
  DrawMenu;
  Repeat
  if keypressed then ChMenu;
  until continua=false;
End;

```

```

PROCEDURE TextoInicio;
Begin
  rectangle(10,10,630,420);
  settextstyle(triplexfont,horizdir,4);
  outtextxy(200,40,'SISTEMA DIGITAL');
  outtextxy(150,70,'PARA PROCESAMIENTO');
  outtextxy(180,100,'Y GENERACION DE');
  outtextxy(170,133,'SEYALES ANALOGICAS');
  settextstyle(triplexfont,horizdir,3);
  outtextxy(175,200,'TRABAJO DE GRADUACION');
  outtextxy(220,230,'PRESENTADO POR');
  outtextxy(150,260,'JUAN CARLOS MAYORGA JUAREZ');
  outtextxy(265,300,'SIDPG 1 0');
  outtextxy(175,350,'© Copyright JCMJ 1995');
  outtextxy(295,385,'UDV');
End;

```

```

PROCEDURE Presenta;
Begin
  setcolor(white);
  TextoInicio;
  repeat
  until keypressed;
  setcolor(black);
  TextoInicio;
  setcolor(white);
End;

```

```
{***** PROGRAMA PRINCIPAL *****}
```

```

BEGIN
  IniciaVarMenu;
  IniciaVarGen;
  IniciaGraf;
  ModoGraf;
  Presenta;
  ModoGraf;
  InFrame(20,5,535,185);
  InGrid(20,5,6,184,20); {Grid entrada 1}
  InFrame(20,210,535,390);
  InGrid(225,210,211,389,225); {Grid Entrada 2}
  BorraMuestras;
  ValMedIn(10,35,70,105,140,'Entrada 1');
  ValMedIn(215,250,285,320,355,'Entrada 2');
  OpenCom;
  GrafIn(21,90,Arx1);
  ProcesaMuestras1;
  GrafIn(21,300,Arx2);
  ProcesaMuestras2;
  MenuVent;
  CloseCom;
  ModoTexto;
  Clrscr;
END;

```

APENDICE F

Pruebas de Operación del
SDPG

SISTEMA DIGITAL
PARA PROCESAMIENTO
Y GENERACION DE
SEÑALES ANALOGICAS

TRABAJO DE GRADUACION
PRESENTADO POR
JUAN CARLOS MAYORGA JUAREZ

SDPG 1.0

@ Copyright JCMJ 1995
UVG

Entrada 1:

Max (V):
-0.01
Min (V):
-0.01
Prom (V):
-0.01
Frec (Hz):
1

Entrada 2:

Max (V):
-0.01
Min (V):
-0.01
Prom (V):
-0.01
Frec (Hz):
1

Grabar	Recuperar	Imprimir
Forma1	Amplitud1	Frecuencia1
Forma2	Amplitud2	Frecuencia2
Muestra	Ganancia1	Ganancia2

Entrada 1:

Max (V): 5.46
Min (V): -4.68
Prom (V): 0.11
Frec (Hz): 2177



Entrada 2:

Max (V): 2.34
Min (V): -2.25
Prom (V): 0.22
Frec (Hz): 2660



- | | | |
|---------|-----------|-------------|
| Grabar | Recuperar | Imprimir |
| Forma1 | Amplitud1 | Frecuencia1 |
| Forma2 | Amplitud2 | Frecuencia2 |
| Muestra | Ganancia1 | Ganancia2 |

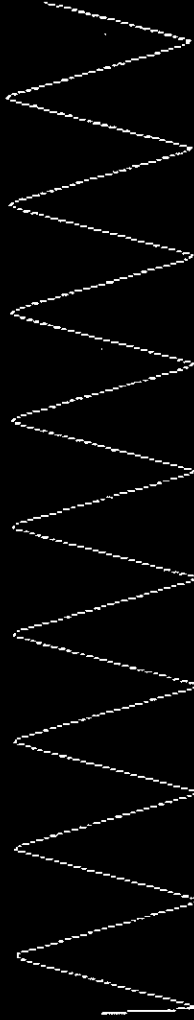
Entrada 1:

Max (V): 10.69
Min (V): -9.16
Prom (V): 0.88
Frec (Hz): 12330



Entrada 2:

Max (V): 16.05
Min (V): -14.88
Prom (V): 0.21
Frec (Hz): 12330



Grabar	Recuperar	Imprimir
Forma1	Amplitud1	Frecuencia1
Forma2	Amplitud2	Frecuencia2
Muestra	Ganancia1	Ganancia2

