

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA

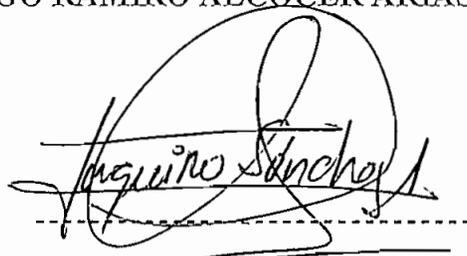
**DISEÑO Y CONSTRUCCIÓN DE UN SINTETIZADOR DIGITAL
DE SEÑALES CONTROLADO POR UN PC.**

**Tesis previa a la obtención del Título de Ingeniero en
Electrónica y Telecomunicaciones**

DIEGO RAMIRO ALCOCER ARIAS

Quito, Mayo, 1996

Certifico que el presente trabajo
de tesis ha sido realizado en su
totalidad por el señor:
DIEGO RAMIRO ALCOCER ARIAS

A handwritten signature in black ink, appearing to read "Tarquino Sánchez A.", is written over a horizontal dashed line. The signature is stylized and cursive.

Ing. Tarquino Sánchez A.

DIRECTOR DE TESIS

DEDICATORIA

*Este trabajo lo realicé por los seres más importantes
en mi vida, mis Padres. Por todo el esfuerzo y cariño que
de ellos recibí constantemente para mi preparación
personal y profesional.*

*A Mirian, ya que sin su ayuda, difícil hubiera
sido el camino.*

AGRADECIMIENTO

Varias han sido las personas que de una u otra manera han colaborado para el desarrollo y culminación de este trabajo, entre quienes merece destacarse el señor Ing. Wagner Alcócer.

Es preciso hacer extensivo el agradecimiento al ingeniero Tarquino Sánchez, director de tesis, quien prestó su contingente en la elaboración de este trabajo.

INDICE GENERAL

i) Introducción

Capítulo I

CONSIDERACIONES GENERALES

NÚMERO	PÁGINA
1.1	Técnicas analógicas. 3
1.1.1	Diseño de circuitos analógicos que permiten obtener formas de onda. 4
1.1.2	Circuitos osciladores 4
1.1.2	Circuitos osciladores sinusoidales 6
1.1.3	Sumario de osciladores sinusoidales 14
1.1.4	Generadores de señales 15
1.1.4.1	Señal cuadrada 15
1.1.5	Generadores de señales, usando circuitos integrados especiales 16
1.2	El procesamiento digital de señales (DPS) 20
1.2.1	Generadores de radio frecuencia (RF) 20
1.2.2	Clasificación de los generadores de radio radio frecuencia (RF) 22
1.2.3	Lazo asegurador de fase (PLL) 24
1.2.4	El TG2000 25
1.3	La síntesis digital directa (DDS) 28
1.4	Comparación entre el procesamiento digital señales y la síntesis digital directa. 32
1.4.1	Diferencias básicas 32
1.4.2	Procesos de análisis y síntesis 34
1.5	Parámetros que definen el alcance y aplicación de un sintetizador digital directo 38
1.5.1	Ancho de banda. 38
1.5.2	Resolución de frecuencia. 40
1.5.3	El reloj base de tiempo 42
1.5.4	El acumulador digital. 43
1.5.5	Manejo de memorias. 50
1.5.5.1	Circuitos con memoria ROM 50
1.5.5.2	Circuitos con memoria PROM, múltiple 51
1.5.5.3	Circuitos RAM 51
1.5.6	Formas de onda. 53
1.5.6.1	Período 54
1.5.6.2	Amplitud 56
1.5.6.3	DDS de 8 bits 57
1.5.7	La conversión digital - analógica de alta velocidad 62
1.5.7.1	Tiempo de conmutación 63
1.5.8	Diseño de filtros pasabajos empleados en el sintetizador digital directo. 64

Capítulo II

DISEÑO Y CONSTRUCCIÓN DEL EQUIPO

2.1	Descripción general del equipo	72
2.1.1	Alternativas de diseño del equipo	77
2.2	Especificaciones técnicas del diseño	84
2.2.1	Acumulador digital	84
2.2.1.1	Sumadores binarios de 4 bits.	85
2.2.1.2	Acumulador digital de 16 bits.	86
2.2.2	Circuito de memoria	90
2.2.3	Convertor digital-análogo	90
2.2.3.1	Polarización del DAC0801 para niveles TTL, DTL, CMOS, PMOS, HTL	92
2.2.4	Filtro de salida	94
2.3	Diseño y construcción de los circuitos de control e interface digital.	99
2.3.1	El reloj base de tiempo	99
2.3.2	El puerto serial del computador	100
2.3.3	El puerto paralelo del computador.	100
2.3.3.1	Descripción del puerto paralelo	101
2.3.3.2	Entradas y salidas	102
2.3.3.3	Velocidad del puerto paralelo	105
2.3.3.4	Cables del puerto paralelo	106
2.3.3.5	Interface del puerto	108
2.3.4	Grabación en memorias externas, parte del sintetizador digital directo.	110
2.3.4.1	Grabado de datos en memoria RAM externa	111
2.3.4.2	Selección de frecuencia de generación	113
2.3.4.3	Control de interface	114
2.4.	Diseño y construcción de los circuitos de alimentación.	117
2.4.1	Fuente D.C.	117
2.4.2	Alimentación momentánea a memoria RAM	117

Capítulo III

DESARROLLO DE PROGRAMAS Y PRESENTACIÓN DE RESULTADOS

3.1	Descripción general de los programas a realizarse.	122
3.1.1	Presentación visual.	122
3.2	Programa del microcontrolador para el ingreso y procesamiento de datos.	126
3.3	Control y presentación de datos en el computador personal.	136
3.3.1	Control de información mediante el puerto paralelo.	136

3.3.2	Programa en Borland C++ para manejo del puerto paralelo bajo Windows	137
3.4	Manejo visual de formas de onda por programa.	140

Capítulo IV

PRUEBAS EXPERIMENTALES, CONCLUSIONES Y RECOMENDACIONES

4.1.1	Medidas experimentales.	156
4.1.2.	Operación del DDS en circuito abierto.	159
4.1.3	Operación del DDS para circuito con carga.	162
4.1.4	Medición de Frecuencia.	167
4.2.	Calidad de la Señal del DDS	170
4.3	Análisis de resultados	187
4.3.1	Respuesta de frecuencia	188
4.3.2	Rangos de frecuencia	189
4.3.3	Características técnicas del equipo	189
4.4.1	Conclusiones	195
4.4.2	Recomendaciones	199

BIBLIOGRAFIA

ANEXOS

I.- INTRODUCCIÓN.-

Las actuales tendencias mundiales en el diseño y construcción de aparatos electrónicos de cualquier clase, se orientan en la utilización del computador personal como herramienta de trabajo, en conjunto con estos aparatos. Además existe la tendencia a la sustitución de técnicas de diseño analógicas por técnicas de diseño digitales, es así que para la generación de señales que en un pasado se las hacía en base a circuitos osciladores, hoy se las está cambiando, utilizando dos técnicas que son muy conocidas en estos días, estas son: el procesamiento digital de señales y la síntesis digital directa.

La síntesis digital directa se basa en la generación de formas de onda mediante la utilización de la expresión matemática que define la señal en el tiempo. Cabe mencionar que la generación de señales de forma directa es una técnica en desarrollo, debido a lo cual no existen muchos sistemas o equipos que la utilicen y que además el adelanto de esta depende del progreso continuo que se realiza hoy en día en la fabricación de convertidores digitales - analógicas, memorias con tiempo de acceso de alta velocidad, y circuitos integrados que trabajen a frecuencias altas.

El presente trabajo pretende analizar con toda profundidad y en detalle los aspectos de diseño básico de la síntesis digital directa, orientada a la construcción del sintetizador digital de señales, con el empleo del computador personal.

En el capítulo I se realiza un análisis teórico acerca de la naturaleza de la síntesis digital, en las cuales puede operar el equipo a construir.

En el capítulo II se realiza el dimensionamiento de los elementos del equipo.

En el capítulo III se realiza el diseño del programa de control y de envío de datos al sintetizador.

En el capítulo IV se presentan los resultados de las pruebas de operación del sintetizador, tanto en circuito abierto y circuito con carga. Finalmente se presentan los resultados encontrados y se determina las bondades y limitaciones del equipo; para llegar a establecer conclusiones y recomendaciones que podrán servir como soporte para la realización de trabajos similares.

CAPITULO I

1.1. Técnicas Analógicas

1.1.1 Diseño de circuitos analógicos que permiten obtener formas de onda.

Introducción al diseño analógico.

1.1.2 Circuitos Osciladores.

Estudio de circuitos osciladores analógicos en frecuencias de hasta los 100MHz.

1.1.3 Sumario de Circuitos Osciladores.

Resumen de los osciladores analógicos estudiados.

1.1.4 Generadores de Señales.

Método para construir un generador de onda cuadrada.

1.1.5 Generadores de Señales usando circuitos integrados especiales

Breve revisión de los circuitos generadores más utilizados.

1.1 TECNICAS ANALOGICAS

El objetivo de estudiar las técnicas analógicas de diseño para la obtención de formas de onda de diverso tipo, es mostrar una visión orientada a definir una teoría conjunta entre la síntesis digital directa y el diseño tradicional. Que partiendo de los conceptos analógicos de diseño, permite establecer comparaciones afines y encontrar diferencias entre los métodos que son parte del estudio de este capítulo.

Por esta razón todo el análisis está orientado a definir bases teóricas desde el punto de vista de diseño analógico, no tratándose de un estudio de generación de señales mediante la utilización de métodos análogos, sino de definir el campo de acción y sus diferencias entre las técnicas de estudio.

Las técnicas analógicas para la obtención de formas de onda de distinto tipo, como por ejemplo señales sinusoidales, triangulares, señales tipo pulso, etc; son generadas por diversos y muy variados métodos, todos ellos muy distintos en su concepción de construcción.

Se pueden mencionar los métodos que utilizan elementos pasivos, elementos activos, circuitos integrados; es decir circuitos que no utilicen variables digitales en su proceso de síntesis¹.

Un proceso de diseño por medios analógicos implica el requerimiento de los siguientes parámetros que definen el tipo y modo de proceso: forma de onda, frecuencia de operación, niveles de voltaje, ancho de banda. Las cuatro variables anteriormente mencionadas definen las diferencias básicas entre las técnicas análogas y digitales para la generación de formas de onda.

¹ Síntesis.- Formación de un todo por reunión de sus partes.

1.1.1 DISEÑO DE CIRCUITOS ANALOGICOS QUE PERMITEN OBTENER FORMAS DE ONDA.

La utilización de elementos pasivos y activos para construir circuitos generadores de señales es tan variado, existiendo una gran cantidad de métodos, cuyo análisis sería tedioso y extremadamente largo, el cual además no es el objetivo del presente trabajo.

El principio general para fabricar un generador de formas de onda mediante técnicas análogas es el de obtener una señal básica, y mediante la manipulación de la misma, obtener señales de distinto tipo. En los métodos analógicos, el manipular señales para la obtención de nuevas señales implica un aumento de elementos o a veces resulta un cambio en la planificación original. A manera de información los generadores de señales comerciales tienen por lo general, señales sinusoidales, triangulares, cuadradas, y TTL. El obtener otra señal distinta a las ofrecidas, implica un pedido especial o la compra de otro generador.

Por esta razón el estudio es orientado a los circuitos que generen señales sinusoidales, dejando su tratamiento más adelante en el capítulo 1 parte 1.1.3

1.1.2 CIRCUITOS OSCILADORES

Un circuito oscilador es aquel en el que se obtiene señal de salida, sin existir ninguna señal de entrada; su estructura básica es la de un amplificador que cuenta con un lazo de realimentación positiva, tal como lo indica la figura 1.1.1.

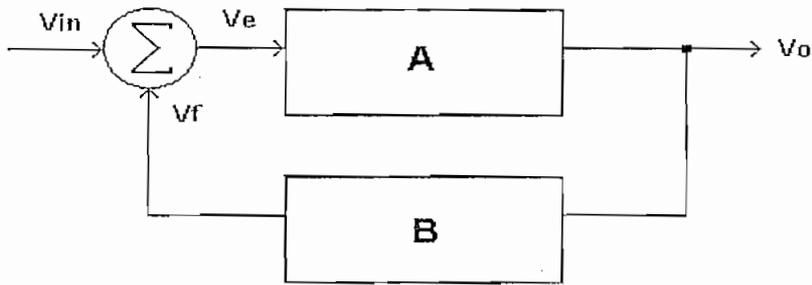


Figura 1.1.1 Amplificador con realimentación positiva.

Parte del voltaje de salida V_o es realimentado por una etapa de amplificación con ganancia B , obteniéndose un voltaje V_f , el cual su vez es sumado con una señal de entrada V_{in} . Del proceso descrito anteriormente se obtiene la ecuación (1.1), que define la ganancia del circuito oscilador respecto a la señal de entrada.

$$Ganancia = \frac{V_o}{V_{in}} = \frac{A}{1 - AB} \quad \text{ecu(1.1)}$$

donde: $A \rightarrow$ ganancia del bloque A, se define como V_o/V_e .

$B \rightarrow$ ganancia del bloque B, se define como V_o/V_f .

Para que se cumpla la condición básica de que un circuito oscilador no tenga señal de entrada ($V_{in} \rightarrow 0$), el término $A \cdot B$ deberá tener un valor cercano a 1. Tanto A y B son términos complejos, por lo cual es conveniente escribir la condición de oscilación por la ecuación (1.2), el cual se refiere al valor real de ambos términos.

$$Real(A \cdot B) = 1 \quad \text{ecu(1.2)}$$

Un circuito oscilador convierta la energía DC en energía AC, utilizando el ruido presente en el proceso de oscilación. La señal de salida debe ser de una frecuencia determinada y en lo posible esta se mantenga estable. Si la frecuencia de la señal de

salida tiene una variación grande, se dice que el oscilador es inestable, caso contrario si la frecuencia de la señal de salida se mantiene estable se dice un oscilador estable.

1.1.2 CIRCUITOS OSCILADORES SINUSOIDALES

Dentro de los circuitos osciladores sinusoidales se tienen los siguientes:

- a) Oscilador de fase.
- b) Oscilador puente de Wien.
- c) Oscilador LC resonante.
- d) Oscilador Colpitts.
- e) Oscilador Hartley.
- f) Oscilador a cristal.
- g) Oscilador de resistencia negativa.

a) OSCILADOR DE FASE

La red de realimentación es obtenida con la combinación de las resistencias y capacitores. La ganancia del amplificador A es determinada por V_o/V_i y el factor B es determinado por V_f/V_o .

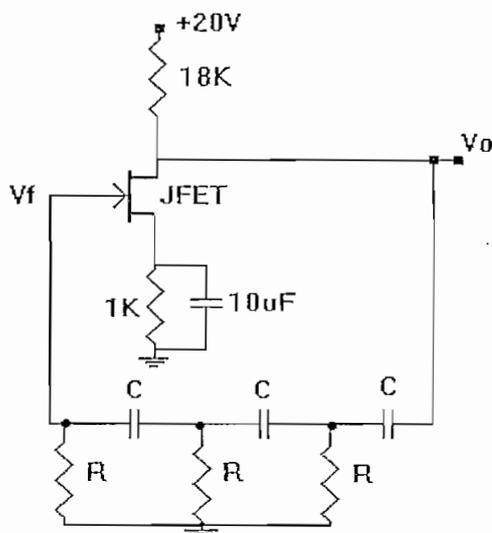


Figura 1.1.2 Oscilador de fase con JFET.

V_o corresponde al voltaje de salida y V_f al voltaje realimentado a la entrada del JFET.

La frecuencia de oscilación es:

$$f_o = \frac{1}{2\pi\sqrt{6RC}} \quad \text{ecu(1.3)}$$

El oscilador de fase no es muy estable debido a que cambios pequeños en los parámetros del circuito dan como resultado variaciones significantes de frecuencia. Para sintonía se usa más comunmente la variación al mismo tiempo de las tres resistencias. Los osciladores de fase operan en frecuencias en el rango de unos pocos hertz hasta varios cientos de kilohertz. En configuraciones distintas, como por ejemplo la construcción mediante parámetros distribuidos son adecuados para rangos de frecuencia en el orden de los megahertz.

b) OSCILADOR PUENTE DE WIEN

La selección de la frecuencia de trabajo se realiza por medio de una combinación de resistencias y capacitores. El oscilador puente de Wien es mucho más estable que el oscilador de fase.

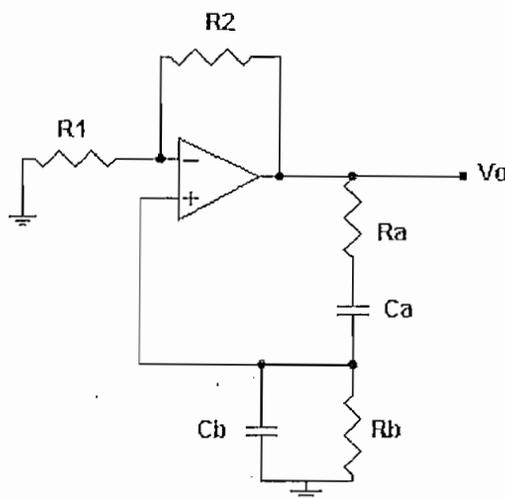


Figura 1.1.3 Oscilador puente de Wien con amplificador operacional.

Para que la oscilación ocurra debe cumplirse la ecuación (1.4)

$$1 + R_2/R_1 > 3 \quad \text{ecu(1.4)}$$

La frecuencia de oscilación es:

$$f_0 = \frac{1}{2\pi(R_a.R_b.C_a.C_b)^{1/2}} \quad \text{ecu(1.5)}$$

Las resistencias R_a , R_b y los condensadores C_a , C_b forman los elementos de ajuste de frecuencia, mientras que las resistencias R_1 , R_2 forman parte de la ruta de realimentación.

El rango de frecuencias de generación de este tipo de osciladores va desde los pocos hertzios hasta valores cercanos a los 100 Megahertzios; siempre y cuando se utilicen amplificadores que respondan al intervalo de frecuencias definido.

c) OSCILADOR LC RESONANTE

El circuito es desarrollado utilizando combinaciones de inductores y capacitores. La frecuencia se determina por la combinación paralelo de LC.

la frecuencia de oscilación es:

$$f_0 = \frac{1}{2\pi\sqrt{LC}} \quad \text{ecu(1.6)}$$

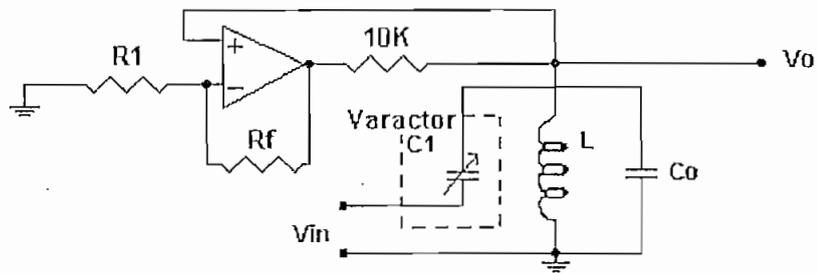


Figura 1.1.4 Circuito LC resonante con amplificador operacional y diodo varactor.

El oscilador LC puede ser usado como un oscilador controlado por voltaje (VCO), tal como se muestra en la figura 1.1.4. Un diodo varactor se coloca en paralelo con la combinación LC, posibilitando la selección de frecuencia mediante un voltaje de entrada V_{in} . Las resistencias R_1 y R_f dan la ganancia del amplificador operacional, generalmente se toma una ganancia mayor que 3. El oscilador LC puede generar frecuencias en el rango de los 10Khz hasta valores cercanos a los 100Mhz sin presentar muchos problemas de sintonización. El circuito presenta muy buena estabilidad.

d) OSCILADOR COLPITTS

Esta basado en una combinación LC, presenta buena estabilidad en la frecuencia de salida para rangos de frecuencias altos, en el orden de los 10Mhz a 100Mhz.

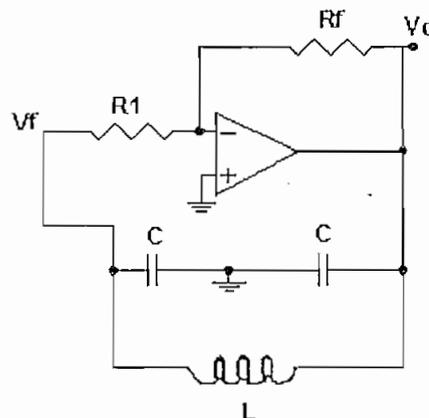


Figura 1.1.5 Oscilador Colpitts con amplificador operacional.

La oscilación ocurre cuando se cumple la ecuación (1.7)

$$R_f / R_1 > 1 \quad \text{ecu(1.7)}$$

La frecuencia de oscilación es:

$$f_0 = \frac{\sqrt{2}}{2\pi\sqrt{LC}} \quad \text{ecu(1.8)}$$

e) OSCILADOR HARTLEY

El oscilador Hartley tiene un diseño similar al oscilador Colpitts, por esta razón presenta características similares en rangos de frecuencias a generar, pero presenta una mejor estabilidad en la señal de salida. La variación de frecuencia se la realiza variando los capacitores con un control común.

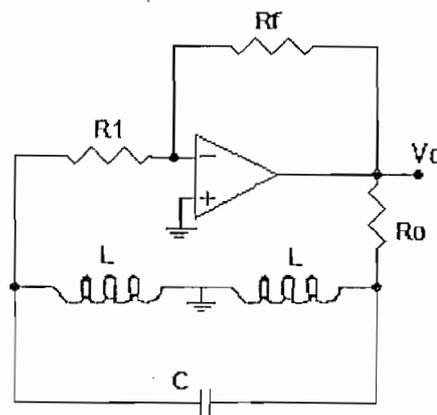


Figura 1.1.6 Oscilador Hartley con amplificador operacional.

Para que ocurra la oscilación se debe cumplir la ecuación (1.9)

$$R_f / R_1 > 1 \quad \text{ecu(1.9)}$$

La frecuencia de oscilación es:

$$f_0 = \frac{1}{2\pi\sqrt{2LC}} \quad \text{ecu(1.10)}$$

El rango de frecuencias que puede generar este tipo de circuito depende de la respuesta del amplificador. Cuando se usa un amplificador operacional la frecuencia generada puede llegar a valores cercanos a los 10Mhz, en cambio con amplificadores con un solo transistor puede llegar a obtenerse valores cercanos a los 100Mhz, dando muy buena respuesta.

f) OSCILADOR A CRISTAL

Un circuito oscilador con cristal es básicamente un oscilador sintonizado que utiliza un cristal piezo-eléctrico² como circuito resonante. El cristal tiene una gran estabilidad para mantener la frecuencia de generación constante a la salida. Cuando un voltaje alterno se aplica al cristal, se producen vibraciones mecánicas que generan una frecuencia natural resonante. Electricamente se representa un cristal, mediante su circuito equivalente, tal como se muestra en la figura 1.1.7.

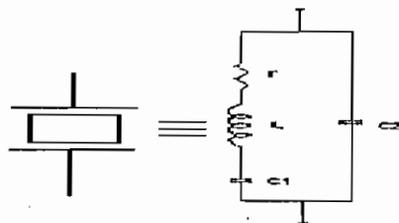


Figura 1.1.7 Circuito equivalente de cristal.

La bobina L y el condensador C_1 representan los equivalentes eléctricos de la masa del cristal, r es el equivalente eléctrico de la fricción contra la estructura interna y

² **Cristal piezo-eléctrico.**- Pieza de material de cuarzo que al actuar un voltaje produce vibraciones mecánicas.

C_2 representa la capacitancia debida al montaje mecánico del cristal. En la figura 1.1.8 se muestra un circuito oscilador con cristal.

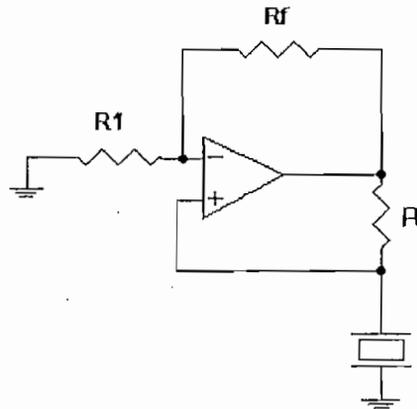


Figura 1.1.8 Oscilador a cristal con amplificador operacional.

En el cual se cumple la ecuación (1.11):

$$A.B = \left(1 + \frac{R_f}{R1}\right) * \frac{Z_c}{Z_c + R} \quad \text{ecu(1.11)}$$

donde Z_c se encuentra en los datos que el fabricante da acerca del cristal.

$R1$, R_f definen la ganancia del amplificador, generalmente se toma una ganancia mayor que 3.

El factor de calidad³ de este tipo de circuitos es alto, además de presentar una buena estabilidad en la frecuencia escogida. Dependiendo del cristal utilizado y la respuesta de frecuencia del amplificador operacional se pueden obtener osciladores en el rango de 1KHz a 50MHz..

g) OSCILADOR DE RESISTENCIA NEGATIVA

³ **Factor de calidad.**- Se define como $Q = \text{frecuencia central} / \text{ancho de banda}$. Representa el rango de frecuencias que genera el oscilador una muy buena señal.

Es posible crear un oscilador con un elemento de dos terminales que tenga resistencia negativa sobre su curva característica. El diodo túnel es uno de aquellos elementos; en la figura 1.1.9 se muestra la curva característica del mismo.

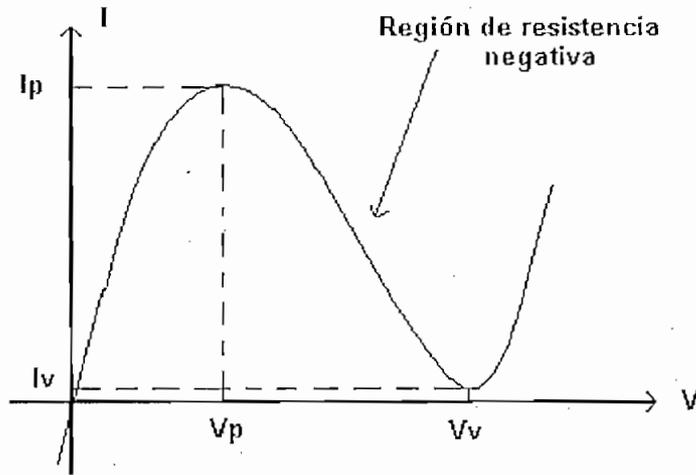


Figura 1.1.9 Características del diodo túnel.

En la región de resistencia negativa cuando se incrementa el voltaje terminal V_p a un voltaje V_v , se produce una reducción en la corriente del diodo, tal como se observa en la figura 1.1.9. El circuito equivalente (a) y un oscilador con diodo túnel (b) se presenta en la figura 1.1.10

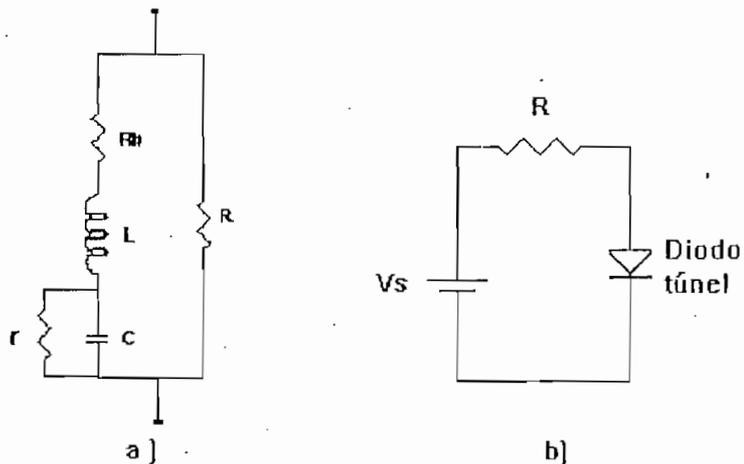


Figura 1.1.10 Circuito equivalente de oscilador de resistencia negativa.

L y R_b se deben a los terminales del diodo, C es la capacitancia de difusión de la

unión en el diodo y $-r$ es la resistencia negativa de la región.

La frecuencia de oscilación es:

$$\omega_0^2 = \frac{1}{LC} \left(1 - \frac{1}{r^2} + \frac{1}{C} \right) \quad \text{ecu(1.12)}$$

L , C y r son valores dados por el fabricante.

1.1.3 SUMARIO DE OSCILADORES SINUSOIDALES

En el cuadro 1.1.1 se muestra un resumen de las frecuencias de resonancia y condiciones de oscilación de los circuitos estudiados. Es importante no exceder los requerimientos de ganancia indicados en el cuadro, debido a que esto puede causar distorsiones a la salida del circuito bajo estudio. Cuando se menciona que la frecuencia de generación depende del amplificador, se refiere a la respuesta de frecuencia del mismo.

Tipo	Frecuencia de Resonancia	Condición de Oscilación	Ventajas	Desventajas
Oscilador de Fase	$f_0 = \frac{1}{2\pi\sqrt{6RC}}$	$A_v > 29$	Frecuencia altas de generación	Difícil sintonización
Oscilador LC	$f_0 = \frac{1}{2\pi\sqrt{LC}}$	$\frac{Rf}{Rl} > 1$	Sintonización automática, estable	Frecuencia depende del transistor.
Oscilador Colpitts	$f_0 = \frac{\sqrt{2}}{2\pi\sqrt{LC}}$	$\frac{Rf}{Rl} > 1$	Estable	Frecuencia depende del transistor
Oscilador Hartley	$f_0 = \frac{1}{2\pi\sqrt{2LC}}$	$\frac{Rf}{Rl} > 1$	Estable	Frecuencia depende del transistor
Oscilador a Cristal	Depende de variables mecánicas	$1 + \frac{Rf}{Rl} > 1$	Gran estabilidad	No permite variación de frecuencias
Resistencia Negativa	$f_0 = \frac{1}{2\pi} \sqrt{\frac{1 - (R_1 + R_2)/R_3}{LC}}$	$\frac{Rb + R}{r} < \frac{L}{rC} < 1$	Alta estabilidad Sintonización fácil	Bajo rango de frecuencias a generar.

Cuadro 1.1.1 Sumario de osciladores

1.1.4 GENERADORES DE SEÑALES.

Un generador de señales por lo general cuenta con varias formas de onda básicas: sinusoidal, triangular, cuadrada, TTL, CMOS. Hasta el momento se ha obtenido una señal sinusoidal a una frecuencia determinada con diseños que utilizan redes RC, L y cristales. Cuando se desea obtener una señal cuadrada, se debe manipular a la señal sinusoidal, obteniéndose se esta forma señales distintas. Los medios más conocidos para obtener señales, sean estas triangulares, cuadradas, sinusoidales rectificadas, tipo pulso, etc; son los siguientes:

1.1.4.1 SEÑAL CUADRADA

Comparador:

La forma más sencilla de obtener una señal cuadrada es mediante la utilización de un amplificador operacional que este funcionando como comparador. El cual cumple con los siguientes principios:

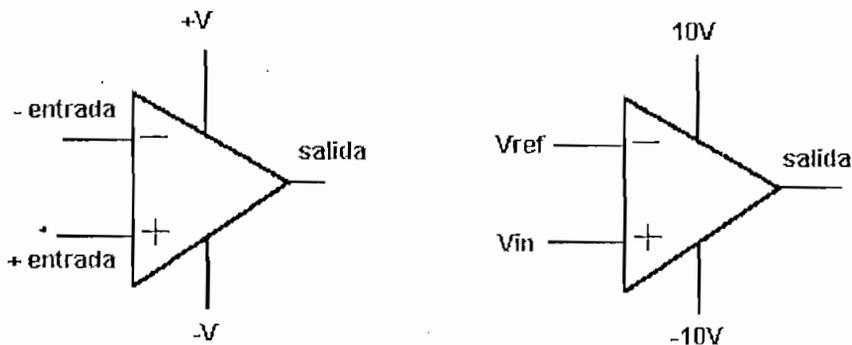


Figura 1.1.11 Amplificador operacional usado como comparador.

La salida es una señal digital que permanece en nivel alto, siempre y cuando la entrada no inversora (+) sea mayor que la entrada inversora (-), y conmuta a un voltaje bajo cuando el voltaje de la entrada no inversora esta por debajo del voltaje de referencia de la entrada inversora. Los niveles altos y bajos de salida dependen de los voltajes de referencia del comparador, así como los voltajes de polarización.

Si se usan amplificadores operacionales con un rango de respuesta de frecuencia alto, se pueden obtener señales cuadradas de muy buena estabilidad en un rango de frecuencias de hasta los 50Mhz.

Una señal TTL, CMOS, puede ser obtenida mediante la polarización correcta de los niveles de referencia del amplificador operacional.

1.1.5 GENERADORES DE SEÑALES, USANDO CIRCUITOS INTEGRADOS ESPECIALES.

Conforme la tecnología se ha ido desarrollando, en el mercado se puede obtener una gran y variada gama de circuitos integrados que nos dan cierto tipo de señales a sus salidas, entre los cuales se tiene por ejemplo : el 566, Exar 2206, Exar 2207, etc .

a) IC 566.-

El 566 es un VCO (oscilador controlado por voltaje), que se utiliza para la generación de señales tanto triangulares como cuadradas a una frecuencia impuesta por elementos pasivos externos. Nos da una buena respuesta de frecuencia en el rango de 1Khz a 100Khz. En la figura 1.1.12 se muestra un circuito en el cual se usa un 566, en donde:

R1, C1, y V determinan la frecuencia de trabajo.

La frecuencia de operación cumple la ecuación (1.13):

$$f_o = \left(\frac{2}{R} * C_1\right) \left(1 - \frac{V_c}{V}\right) \quad (1.13)$$

De la ecuación (1.13) se puede comprobar la facilidad para la generación de señales triangulares, cuadradas, así como escoger la frecuencia deseada.

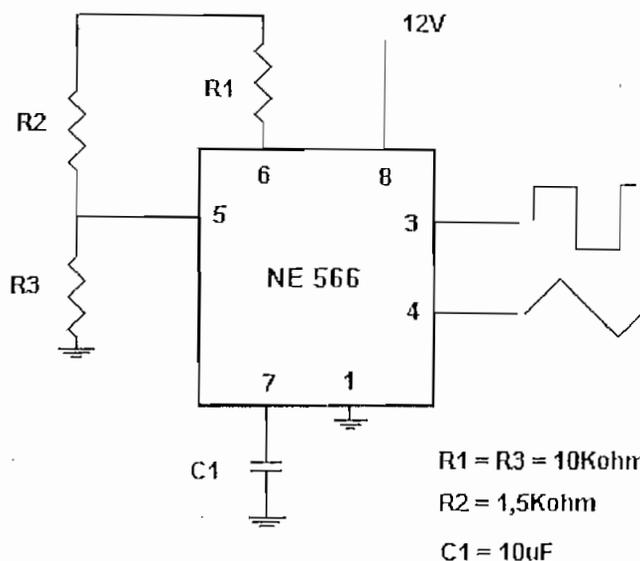


Figura 1.1.12 Oscilador con circuito integrado IC566.

b) EXAR 2206.-

Es un generador de funciones capaz de producir señal: sinusoidal, cuadrada, triangular, rampa, y señales pulso de alta estabilidad y exactitud. Las formas de onda pueden además ser moduladas en amplitud y frecuencia mediante un voltaje externo. Tiene un rango de operación de frecuencia en el orden de 0,01Hz hasta 1MHz.

Se lo usa generalmente en comunicaciones, instrumentación y generadores de funciones. En la figura 1.1.13 se muestra el diagrama de bloques del circuito. El Exar 2206 se compone de cuatro bloques funcionales: un oscilador controlado por voltaje, un multiplicador análogo, un formador de onda sinusoidal⁴ y un amplificador de corriente de ganancia unitaria.

⁴ **Formador de onda sinusoidal.**- Es un filtro para generar señales sinusoidales de alta calidad.

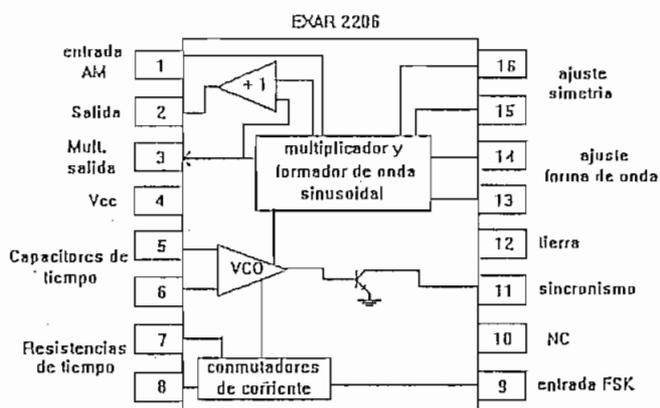


Figura 1.1.13 Diagrama de bloques del EXAR 2206

El VCO produce una salida de frecuencia proporcional a la corriente de entrada, la cual es producida por una resistencia externa colocada en los terminales de resistencias de tiempo. El multiplicador de frecuencias y el formador de onda sinusoidal permiten la obtención de los otras formas de onda mencionadas anteriormente.

Los generados de señales construidos utilizando circuitos integrados son simples de construir, solo necesitan de resistencias y capacitores así como su calibración es sencilla. Su principal desventaja se constituye en la limitación de formas de onda que pueda generar, ya que obtener cualquier otra señal implica que se modifiquen o se añadan nuevos circuitos.

1.2. El Procesamiento Digital de Señales

1.2.1 Generadores de radio frecuencia.

Espectro real de radio frecuencias.

1.2.2 Clasificación de los generadores de radio frecuencia (RF).

Breve descripción de generadores RF.

1.2.3 Lazo Asegurador de Fase(PLL).

Descripción de un PLL para frecuencias de hasta 2GHz.

1.2.4 El TG2000.

Generador comercial de frecuencia en el rango de 4MHz a 2GHz.

1.2 EL PROCESAMIENTO DIGITAL DE SEÑALES (DPS)

El procesamiento digital de señales es una de las técnicas digitales más utilizadas hoy en día para la generación y proceso de señales. Las aplicaciones del DPS son por ejemplo: generadores de radio frecuencia, comprobadores de dispositivos, etc.

El procesamiento digital de señales parte de conceptos análogos de generación, pero que en su proceso de síntesis sufre un cambio radical en comparación a las técnicas análogas, ya que hace uso de medios digitales/análogos para la obtención final del proceso. Debido a que el objetivo del presente trabajo es generar formas de onda mediante la utilización del computador, se analizará más adelante al TG2000, el cual es un generador de señales de radio frecuencia que utiliza la técnica DPS.

1.2.1 GENERADORES DE RADIO FRECUENCIA (RF)

La figura 1.2.1 muestra el espectro RF de un generador ideal que produce una sola frecuencia o tono , con un grado infinito de pureza.

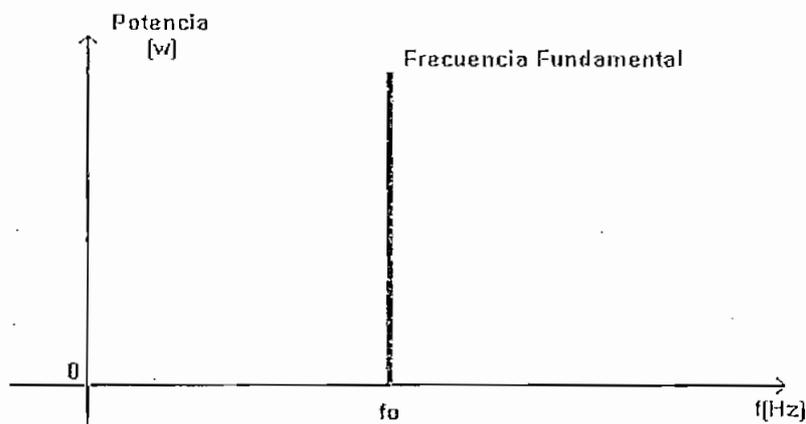


Figura 1.2.1 Espectro de un generador ideal.

El mundo real es mucho más complejo que un simple tono, aparecen armónicos como se muestran en la figura 1.2.2:

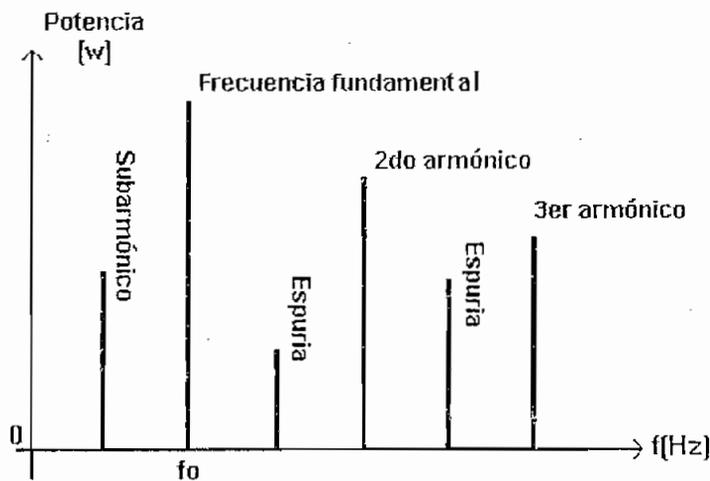


Figura 1.2.2 Espectro de un generador real.

En la realidad el espectro de frecuencias de un generador de radio frecuencia presenta la señal fundamental f_0 , constituida de componentes armónicas y señales no deseadas. Las mismas se clasifican en tres grupos: señales armónicas, espurias y modulación residual.

a) Señales armónicas.-

Debidas al proceso de transporte de banda base a altas frecuencias.

b) Señales espurias (no armónicas).-

Se debe a las inductancias y capacitancias distribuídas entre los puntos de enlace (cables) y superficies de conducción, produciendo efectos oscilatorios.

c) Modulación residual.-

Se debe al ruido de fase y al proceso de aseguración del lazo de realimentación que usa esta técnica. Para entender que se refiere la modulación residual, observemos la figura 1.2.3

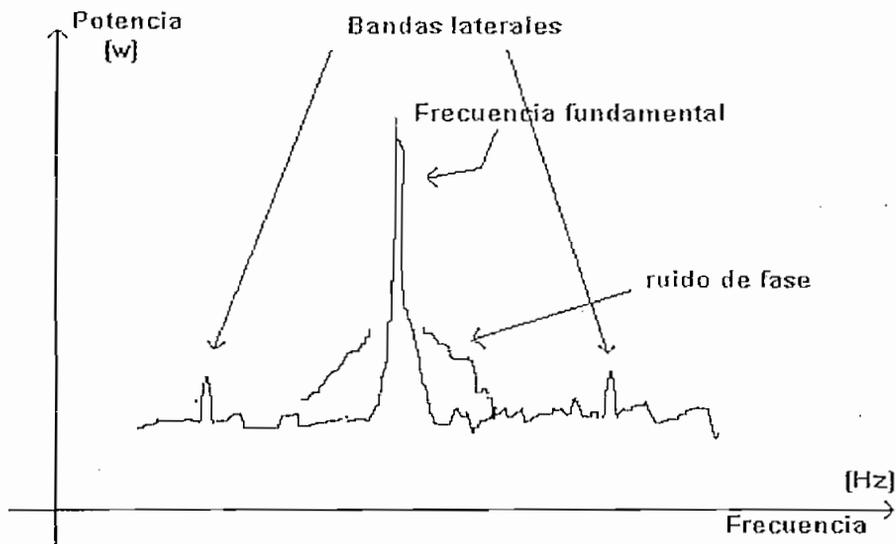


Figura 1.2.3 Acercamiento a la señal-fundamental.

El ruido de fase es causado por las fluctuaciones al azar de ruido y no puede ser completamente eliminado. Para comprender los efectos de la modulación residual, imaginemos un generador de frecuencia modulada (FM) ideal del cual se alimenta a un receptor FM⁶ ideal, el resultado de esta operación es la no existencia de ruido en el parlante del receptor FM. En cambio si se alimenta con un generador FM que contiene modulación residual, el ruido de fase es oído como un chasquido y el proceso de aseguración de lazo es oído como un tono. Esto se cumple siempre y cuando se asuma que las frecuencias de trabajo puedan ser escuchadas.

1.2.2 CLASIFICACION DE LOS GENERADORES DE RADIO

RADIO FRECUENCIA (RF).

⁶ F.M. .- Frecuencia modulada.

Los generadores de RF se clasifican en 2 grupos: a) generadores no sintetizados y b) generadores sintetizados.

a) Generadores no sintetizados o de lazo abierto.-

Son simples y de fácil construcción, por esta razón pueden ser implementados en la práctica sin contar con equipo especializado, un ejemplo de este tipo de generadores es un oscilador controlado por voltaje (VCO).

b) Generadores sintetizados.-

Los generadores sintetizados son más exactos y estables en la frecuencia que un generador no sintetizado. Un VCO puede ser convertido en un generador sintetizado mediante la adición de un lazo asegurador de fase (PLL), tal como se muestra en la figura 1.2.4.

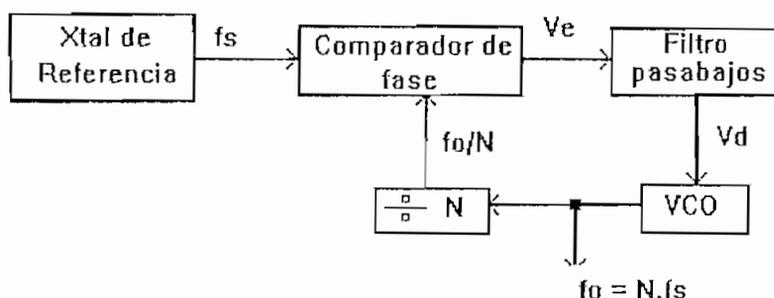


Figura 1.2.4 Generador Sintetizado.

El VCO genera una señal de frecuencia f_0 , la cual pasa por un bloque divisor de razón N para poder comparar la fase de la señal de salida y la fase del cristal de referencia. Si ambas fases son distintas, el comparador a su salida genera un voltaje V_e el cual luego de pasar por un filtro pasabajos se convierte en un voltaje V_d , alimentando entonces al VCO. Este proceso se repite continuamente hasta que las fases de las señales comparadas sean iguales.

El acoplamiento del PLL al VCO genera pasos de frecuencia a diferencia del VCO sólo, en el cual la frecuencia de resolución es infinita. Un beneficio de un control PLL a un oscilador controlado por voltaje es que el ruido de fase declina dramáticamente sobre lo que representaba en el VCO de lazo abierto.

1.2.3 LAZO ASEGURADOR DE FASE (PLL).

La ventaja de usar un PLL basado en un sintetizador, es el de generar señales en el rango de 1MHz a 1GHz.

El mínimo número de componentes para un PLL basado en un sintetizador son: un VCO, un detector de fase / frecuencia, un filtro de lazo y un cristal de referencia; tal como se indica en la figura 1.2.5

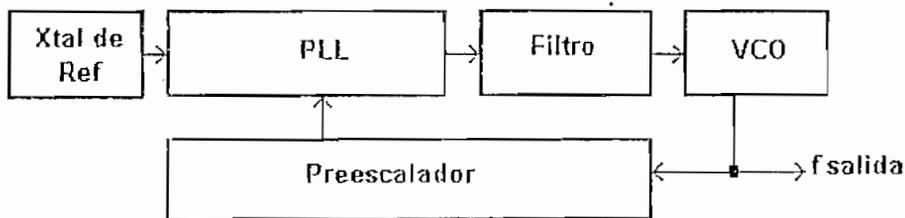


Figura 1.2.5 Diagrama de bloques de un sintetizador con PLL.

Un constituyente importante de un sintetizador con PLL es el módulo de Pre-escala, el cual tiene como función reducir mediante divisiones programables la frecuencia del VCO a una frecuencia manejable para el PLL. Los divisores programables son registros binarios de N bits; en la práctica se cuenta con módulos pre-escala de 2 razones de división, siendo estos generalmente N y $N+1$. La razón de división activa depende de un control separado.

Al dividir la frecuencia generada por el VCO para un número entero (definido por el registro programable), se crean pasos de generación de frecuencia.

En general un VCO permite frecuencias de operación del doble de la frecuencia más baja, es decir un VCO que genera hasta una frecuencia mínima de operación de 1Ghz podrá generar solo hasta una frecuencia de 2Ghz. Si se desea la construcción de un VCO en el rango de los 4 a 2048mhz el usar un solo VCO no soluciona el rango de frecuencia limitado. Para superar este inconveniente se usa redes divisoras binarias para crear frecuencias entre 4 y 1Mhz. Por ejemplo si se desea generar señales en el rango de frecuencias de 4 Mhz a 8 Mhz se usa una red divisora binaria de palabra de 8 bits, esto es, la frecuencia mínima del VCO dividida para 2^8 y la frecuencia máxima para 2^8 , obteniéndose:

$$\frac{1024}{256} = 4\text{MHZ} \qquad \frac{2048}{256} = 8\text{MHZ}$$

Una desventaja de usar este método es el alto contenido de armónicos causados por los divisores.

1.2.4 EL TG2000.

El TG2000 es una tarjeta que se instala en un slot de 8 bits del BUS de un computador, alternativamente se puede obtener la versión externa que usa el puerto paralelo para control y comunicación. La generación de frecuencias del TG2000 va en el rango de los 4 a 2048Mhz, en pasos de resolución y ancho de banda progresivos, tal como se muestra en el cuadro 1.2.1.

El TG2000 utiliza un PLL Fujitsu MB1501⁷ con las siguientes características:

- Posee un interfase serial por el cual se envía información al registro de la red divisora binaria.
- Tiene módulos divisores en la razón de $N/N+1$, siendo estos;

64/65

y

128/129

⁷ PLL- MB1501.- Ver anexos.

Frecuencia de resolución	Banda
250Khz	1024 a 2048 Mhz
125Khz	512 a 1024 Mhz
62.5Khz	256 a 512 Mhz
31.25Khz	128 a 256 Mhz
15.625Khz	64 a 128 Mhz
7.8125Khz	32 a 64 Mhz
3.90625Khz	16 a 32 Mhz
1.95312Khz	8 a 16 Mhz
0.97656Khz	4 a 8 Mhz

Cuadro 1.2.1 Frecuencias de generación del TG2000.

- Detector de fase y frecuencia.
- Dos registros contadores A y N para seleccionar la frecuencia de generación del PLL.

La frecuencia de generación del PLL se obtiene usando la ecuación (2.1)

$$f_{PLL} = \left[(P^N) + A \right]^2 * \left[\frac{f_{osc}}{R} \right] \quad \text{ecu(2.1)}$$

donde:

f_{osc} = frecuencia del oscilador de referencia.

R = registro divisor para la frecuencia del reloj de referencia.

A, N = Registros binarios para selección de frecuencia.

P = módulos divisores.

El control de registros y componentes binarios se realiza mediante la utilización del computador, en el cual se guardan en archivos cambios en el proceso.

1.3 La Síntesis Digital Directa

1.3.1 Introducción a la técnica, definición de parámetros.

1.3 LA SINTESIS DIGITAL DIRECTA (DDS).

A diferencia de las técnicas anteriormente estudiadas la síntesis digital directa (DDS = Digital Direct Synthesis) es un procedimiento mediante el cual se generan formas de onda partiendo de la expresión matemática que define la función en el tiempo y de valores numéricos que definen variables representativas de una señal en particular, como por ejemplo: frecuencia, amplitud, fase, etc.

El principal objetivo de la síntesis digital directa es el obtener resoluciones finas de frecuencia lo más bajas posibles, altas velocidades de conmutación y bajo ruido de fase, distinguiendola de las técnicas análogas y del procesamiento digital de señales.

A la síntesis digital directa, también conocida como síntesis digital de frecuencia directa (DDFS = Digital Direct Frequency Synthesis), se le ha dado un espacio de interés especial de estudio, tanto de la técnica propiamente dicha, así como de los elementos que esta hace uso. Debido a que es una técnica reciente, se la considera como un término nuevo que dentro de poco estará en la voz de desarrollo de ingeniería. Principalmente se la usará en el campo de desarrollo de hardware para la generación de señales y formas de onda que se predestinan previamente o que pueden ser formadas mediante un software especializado.

A causa de las tolerancias de los componentes, variaciones de los valores con el tiempo, e inconsistencias en la manufacturación, las tradicionales técnicas análogas pueden solamente aproximarse a la señal deseada. En contraste con la DDS que calcula la señal directamente.

DDS es conocido como un método numérico, antes que una técnica digital. Para esta categorización hay dos razones: la primera una razón técnica y la segunda una razón más orientada al marketing. El elemento técnico proviene de los cálculos directos que se hacen, de donde la señal actualmente es generada por la manipulación de números.

Aunque el termino "digital" puede tener el mismo significado que el término "numérico", el término digital, comúnmente se refiere a tener dos niveles de amplitud. DDS usa técnicas que son digitales en ambos sentidos, pero los dos aspectos, que tienen que ver con el campo numérico del método DDS, es la representación de cantidades y la precisión inherente a esos resultados, del uso de números.

El elemento marketing es orientado para llamar a DDS una técnica numérica, antes que una técnica digital, esto se relaciona con técnicas impuestas en el tradicional diseño análogo.

Por lo general para mantener formas de onda con una precisión del 0,1% o menos (60 dB en el rango dinámico), los diseñadores tiene que evitar los circuitos digitales. Muchos circuitos tiene la reputación de generar corrientes de ruido que degradan la señal, que se obtiene en circuitos análogos.

En este punto es conveniente enunciar el teorema del muestreo que nos dará una idea básica de cual es el principio teórico de la DDS; este dice: "**Una señal limitada en banda que no contiene componentes espectrales mayores que la frecuencia f_m (Hz) está determinada en forma única por sus valores en intervalos uniformes menores de $1/(2f_m)$ segundos**".

Este sirve para tomar como primer parámetro que una señal análoga puede ser convertida de digital a análoga si es que se tiene el número de parámetros básico que la puedan describir, se puede notar que se sigue insistiendo en el término numérico, referido anteriormente.

La síntesis digital directa nos da una alta exactitud en la señal de salida, todo esto a un costo moderado, siempre y cuando se encuentre en un rango de frecuencias adecuado. Por ejemplo en circuitos análogos donde se requiere una exactitud del 0,1%, se incurre en altos costos de mantenimiento. En cambio si se maximiza el uso de técnicas digitales, la síntesis digital genera señales donde el mantenimiento se hace inapropiado.

Estas son razones que dan la pauta del porque implementar la síntesis digital directa. Es conveniente aclarar que a bajas frecuencias es innecesario construir un aparato que utilice esta técnica, ya que el sistema se lo puede construir usando ciertas características y funciones generales de los microprocesadores. Por ejemplo, modems de baja y media velocidad han sido construídos por esta vía por años. Como el requerimiento es que la frecuencia de la señal se incremente sobre el rango de audio, el computo sobre la señal se incrementa proporcionalmente. Para generar señales en el rango de frecuencia del orden de 1 a 4Mhz el uso del microprocesador no puede ser implementado, a menos que se construyan microcontroladores de muy alta velocidad.

Para frecuencia del orden de 0 a 4Mhz se considera la implementación un sintetizador digital con un hardware dedicado a los efectos de las altas frecuencias. Hoy en día existen elementos que son optimizados para la síntesis directa de una señal.

Debidamente enunciados y explicados loas principales diferencias entre el procesamiento digital de señales y la síntesis digital directa, los parámetros necesarios para la DDS son:

- parámetros de entrada
 - selección de frecuencia.
 - forma de onda a generar.
 - reloj de referencia.

- parámetros de salida
 - la señal generada.

El sintetizador digital en esencia actúa como un periférico, dejando libre el procesador central para otros menesteres. Es decir en un principio se necesita de la existencia de una fuente de información que nos de ciertas variables de entrada, en nuestro caso será un computador que usando el software adecuado se pueda tener la información requerida para iniciar el proceso de síntesis.

1.4. Comparación entre la DPS y DDS

1.4.1 Diferencias básicas.

Definición de los campos de estudio de ambas técnicas.

1.4.2 Procesos de Análisis y Síntesis.

Diferencias teóricas de ambas técnicas.

1.4 COMPARACION ENTRE EL PROCESAMIENTO DIGITAL DE SEÑALES Y LA SINTESIS DIGITAL DIRECTA.

1.4.1 DIFERENCIAS BASICAS.

La síntesis digital directa se constituye en una alternativa derivada del procesamiento digital de señales para que circuitos digitales actualmente generen señales análogas, tal como se indica en la figura 1.4.1

En la DPS el análisis y la síntesis son procesos constitutivos que deben desarrollarse para cualquier diseño y aplicación de la técnica. En la síntesis se usa dos alternativas básicas: la síntesis digital directa (OMCN⁸) y los algoritmos, de tal la DDS se constituye en una extensión de la DPS.

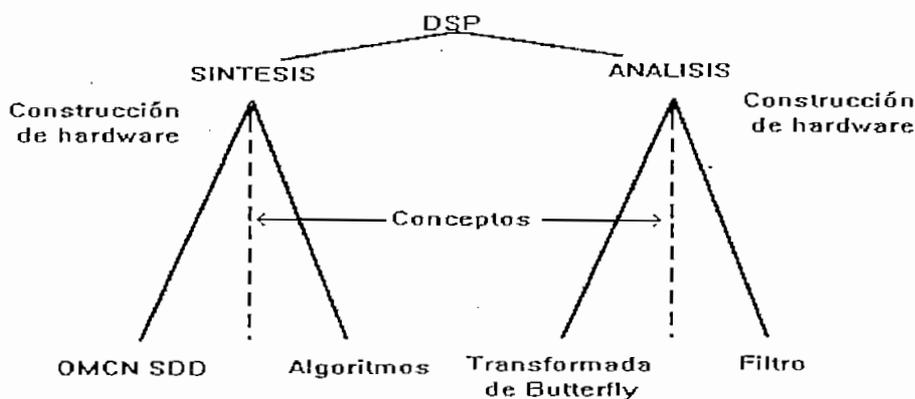


Figura 1.4.1 Procesos de síntesis y análisis en la DPS

El concepto básico diferencial entre las dos técnicas se constituye en el proceso para la obtención de una forma de onda. La DPS manipula señales previamente generadas por procesos digitales/análogos, alcanzando con esta mayores rangos de generación de frecuencia. En cambio la DDS parte del algoritmo matemático que define la señal

⁸ OMCN.- A un DDS se le conoce también como oscilador controlado por número

periódicamente en el tiempo y la genera de modo directo realizando la conversión de datos digitales a análogos.

El procesamiento digital de señales se lo a usado por muchos años en el filtrado de señales cuando se realiza un proceso de conversión análoga a digital. Además el DPS se usa para transformar señales digitalizadas, transformando una señal en el dominio del tiempo en un equivalente en el dominio de la frecuencia. La DPS usa procesos analíticos, es decir toma una señal existente, la procesa, y modifica, hasta obtener la señal deseada.

La síntesis digital directa toma un grupo de parámetros (números) que describen la señal que se desea obtener y entonces genera una secuencia de números que representan la señal. Esta secuencia de números usualmente ingresan a la etapa de conversión digital/análoga para finalmente producir una señal análoga.

Las dos técnicas tienen la limitación de los pasos de frecuencia que se pueden generar. En la DPS para la generación de pasos más finos de frecuencia de resolución hace uso de la DDS, la cual reemplaza el cristal de frecuencia de referencia, para habilitar a la salida pasos de resolución de hasta 2HZ, debido a que la DDS puede generar frecuencias de referencia del orden de 0,1 a 1Mhz con solo cambiar el reloj de referencia, se puede realizar un proyecto de trabajo conjunto entre las dos técnicas sin que esto implique problemas con el ruido de fase, ya que el DDS solo opera como etapa de sincronismo y referencia.

En el cuadro 1.4.1 se puede observar diferencias básicas entre las dos técnicas, donde se muestran parámetros económicos y técnicos. Del mismo se observa que la diferencia entre las dos técnicas se constituye en el rango de frecuencia que puede cubrir tanto la DPS y DDS.

En cambio la DDS presenta costos muy reducidos, el hardware así como el software no son tan complicados y costosos como la DPS.

Parámetros	DPS	SDD
Rango de frecuencia	0 a 2GHz	0 - 4MHz
Frecuencia de resolución	2Hz	0 - 2MHz
Calidad de Señal	Alta	Alta
Tipos de señales	Sinusoidal	Varias
Circuitos Usados	Análogos/Digitales	Digitales
Costo	USD 1200	USD 250

Cuadro 1.4.1 Diferencias básicas entre la SDD y DPS.

1.4.2 PROCESOS DE ANALISIS Y SINTESIS.

De acuerdo a la figura 1.4.1, el análisis y la síntesis son requisitos básicos para generar aplicaciones DPS. La síntesis es aquel proceso que define la obtención de una forma de onda, se la realiza mediante el uso de algoritmos matemáticos o haciendo uso de la DDS, es decir la necesidad es obtener una forma de onda generada a baja frecuencia. Los algoritmos matemáticos para la obtención de formas de onda se mencionaran en el punto 1.5.6 del capítulo 1.

Ambas técnicas utilizan señales que son muestreadas previamente por procesos digitales para su posterior conversión analógica, por lo que es claro notar que se trabaja con el teorema del muestreo usado de modo inverso, donde se parte de las muestras de la señal realizadas a una señal análoga para su posterior conversión.

En el lado izquierdo de la figura 1.4.2 se muestra el proceso de muestreo convencional que se aplica en el procesamiento digital de señales, este proceso utiliza un filtro limitador de la señal de entrada, luego de lo cual la señal es muestreada y digitalizada. El digitalizador posee una secuencia de números que pueden ser procesados para identificar parámetros característicos.

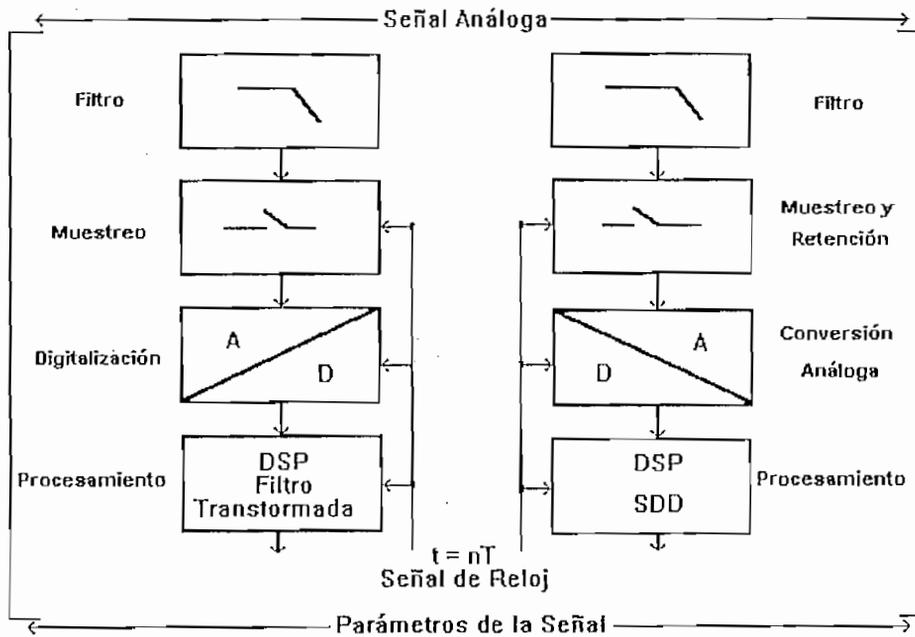


Figura 1.4.2 Análisis usando DPS y Síntesis usando DDS.

Con la DDS los parámetros característicos existen primero, por ejemplo: frecuencia de la señal, forma de onda, etc. La secuencia de números es generada a partir de estos parámetros y luego convertida a la señal analógica. Los ruidos característicos de la señal son removidos usando un filtro limitador de banda. El proceso matemático en esencia es el mismo; el orden en el cual se utilizan las operaciones características es diferente.

Un buen ejemplo para entender el uso de la matemática en la generación de formas de onda, es recordar las aplicaciones de los amplificadores operacionales y los operadores de derivación o integración, así como los sumandos y multiplicadores. Matemáticamente una señal eléctrica puede ser definida como la suma de muchas señales, por lo cual puede ser generada.

Una vez generada la forma de onda, sea esta triangular, sinusoidal, etc, se ejecutan procesos matemáticos de suma y multiplicación. La multiplicación de señales es obtener señales de frecuencias diferentes a la original. Generalmente en un proceso de multiplicación de señales eléctricas se producen los armónicos de señales.

La DPS hace uso de los mismos para generar señales a mayor frecuencia. Cualquier proceso de multiplicación de señales generan ruidos, las mismas que deben ser eliminadas mediante un proceso de filtrado. El proceso descrito anteriormente se lo describe punto a punto en el literal 1.2, correspondiente al estudio del TG2000(ver punto 1.2.4), el cual usa el procesamiento digital de señales.

El proceso de síntesis en la DDS hace uso de tres etapas: acumulador digital, mapa de la forma de onda y conversor digital/análogo. No se producen procesos de suma y multiplicación.

1.5. Parámetros que definen la DDS

1.5.1 Ancho de Banda.

Máximo ancho de banda a generar.

1.5.2. Frecuencia de Resolución.

Frecuencias posibles de generar.

1.5.3 El Reloj base de tiempo.

Importancia del tiempo de referencia.

1.5.4 Acumulador Digital.

Muestras necesarias para realizar una conversión exitosa de señal.

1.5.5 Manejo de memorias.

Circuitos DDS con memorias.

1.5.6 Formas de Onda.

Formato para los datos a grabar en memoria.

1.5.7 Conversión Digital/Analógica.

Consideraciones especiales para escoger el conversor adecuado.

1.5.8 Filtros Pasabajos.

Filtros activos de primer y segundo orden.

1.5 PARÁMETROS QUE DEFINEN EL ALCANCE Y APLICACION DE UN SINTETIZADOR DIGITAL DIRECTO.

1.5.1 ANCHO DE BANDA.

A causa de la digitalización de las señales análogas y del sistema de muestreo, un DDS posee un múltiple espectro de frecuencias en la señal de salida. En suma, a la señal deseada de frecuencia f_0 se obtienen también armónicos, relacionadas con dicha frecuencia f_0 .

El espectro de salida será de la forma :

$$f_0, f_0 + f_{osc}, f_0 + 2 * f_{osc}, f_0 + 3 * f_{osc}, \text{ etc;}$$

donde f_{osc} : frecuencia del oscilador de referencia.

tal como se muestra en la figura 1.5.1

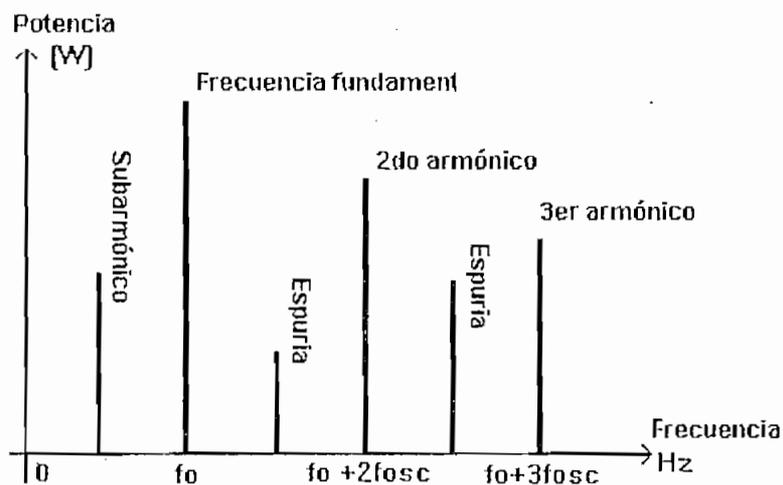


Figura 1.5.1 Espectro de la señal de salida de un DDS.

Estas señales se deben a la mezcla de los productos de la señal de salida con la señal de reloj de referencia, y sus armónicos. Las señales no deseadas deben

ser eliminadas usando un filtro pasabajos, para obtener a la salida la señal de frecuencia f_0 .

La máxima frecuencia de salida de un DDS corresponde a la mitad de la frecuencia del reloj de oscilación o referencia, tal como lo define la ecuación (1.14)

$$f_{max} = \frac{f_{osc}}{2} \quad \text{ecu(1.14)}$$

donde, f_{max} = frecuencia máxima de la señal a generar

f_{osc} = frecuencia del reloj de oscilación

Para aplicaciones prácticas la frecuencia máxima de generación está alrededor del 40 y 45% de la frecuencia de reloj. El ancho de banda del DDS estará definido para aplicaciones prácticas por la ecuación (1.15):

$$AB = 0,40 * f_{osc} \quad \text{ecu(1.15)}$$

La ecuación (1.11) define además el ancho de banda del filtro pasabajos de salida. Debido a que la frecuencia máxima se aproxima por debajo a la frecuencia de Nyquist y que la frecuencia $f_0 + f_{osc}$ se aproxima por encima de la frecuencia de Nyquist por encima, la complejidad del filtro se vuelve más compleja. Por ejemplo si se desea obtener una señal a frecuencias por encima del 40% de la frecuencia de reloj, el filtro de salida empieza a ser impracticable. En este caso para solucionar el problema se necesita la construcción de un filtro Chebysev de por lo menos 20 secciones de 60 dB, el cual rechazará las señales no deseadas.

Producir un filtro de por lo menos 20 secciones es muy costoso

1.5.2 RESOLUCION DE FRECUENCIA.

Una vez definida la frecuencia de oscilación del reloj de referencia y el ancho de banda, el obtener la frecuencia de resolución del DDS se constituye en el siguiente paso a seguir. La resolución de frecuencia es el parámetro que define el intervalo de resolución que el DDS genera, el cual a su vez define las frecuencias a generar y la exactitud de la mismas. La frecuencia de salida f_o de un DDS está definida por la ecuación (1.16)

$$f_o = \left(\frac{f_{osc}}{K}\right) * M \quad \text{ecu(1.16)}$$

donde: f_o = frecuencia de la señal generada por la DDS.

f_{osc} = frecuencia del reloj de referencia en Hz.

K = módulo de operación del acumulador digital.

M = número aplicado al DDS.

La frecuencia del reloj de oscilación es el rango de muestreo del DDS, constituyéndose en el tiempo en el cual el DDS toma la muestra de la amplitud de la señal y realiza la conversión. En muchos casos el rango de muestreo es igual a la frecuencia del reloj de referencia aplicado.

El diseño del DDS determina el módulo de operación K , el cual es definido como el número de estados que el acumulador directo puede tomar.

En muchos equipos DDS se utilizan circuitos binarios, donde K puede tomar los valores en potencia de 2, como por ejemplo: 2^{24} o 2^{32} .

Cuando un DDS usa circuitos decimales, K es una potencia de 10, como por ejemplo 10^6 o 10^8 . Existe hoy en día una técnica nueva llamada Resolución de variable (VR), en la cual se posibilita al operador colocar un número entre 1 y el número máximo de estados del acumulador digital.

La variable definida como M , corresponde a un número entero entre 0 y $K/2$. El límite superior de esta variable es llamado como límite Nyquist, la cual garantiza el cumplimiento del teorema de Nyquist. Cuando $M=0$, la ecuación (1.16) da como resultado que la frecuencia de salida es 0, este determina que en el DDS también se pueda obtener señales DC en su banda de trabajo.

La frecuencia de resolución se define como la derivada de la ecuación (1.16) con respecto a la variable M , obteniéndose la frecuencia de resolución como:

$$f_{res} = \frac{f_{osc}}{K} \quad \text{ecu(1.17)}$$

La frecuencia de resolución es idéntica a la frecuencia de salida cuando $M=1$. A causa de que M debe ser un entero, todas las frecuencias de salida podrían tener armónicos en la resolución. Por esta razón, esta resolución es ocasionalmente llamada la cuantización de frecuencia DDS.

La frecuencia de resolución DDS puede ser muy pequeña, como por ejemplo, considerando un aparato pequeño DDS de 24 bits binarios, operando con un reloj a 10MHz. De la ecuación (1.17), se obtiene:

$$f_{res} = \frac{10000000 \text{ Hz}}{2^{24}}$$

$$f_{res} = 0,59 \text{ Hz}$$

Obtener una fina frecuencia de resolución es fundamental para conseguir las características deseadas del DDS. Muchos bits de entrada en el acumulador digital posibilitan pasos finos de secuencia.

En aplicaciones más exactas, estas requieren una frecuencia de resolución exacta, y tienen una singular frecuencia de referencia para el reloj DDS.

1.5.3 EL RELOJ BASE DE TIEMPO

El reloj base de tiempo se constituye en la señal principal que actúa en el sintetizador digital, la cual posibilita el apareamiento de una señal a la salida. Por trabajar con circuitos de nivel TTL (74LS374), el reloj base de tiempo o reloj de referencia utiliza señales eléctricas a nivel de 0 y 5V. El cambiar los niveles de referencia de voltaje para definir un 0 y 1 lógico implicará la utilización de otro tipo de tecnología.

En el mercado se pueden conseguir osciladores a nivel TTL, de implementación muy sencilla, sólo necesitan que sean polarizados, y a la salida se obtendrá la señal oscilante a nivel TTL.

El proceso de diseño de la DDS, implica que la técnica satisfaga necesidades particulares del operador, esto es que genere a pasos de frecuencia preestablecidos. De la ecuación (1.17) se deduce que la frecuencia de resolución puede ser variada usando un valor distinto de K , de donde se obtiene la ecuación (1.18) siguiente:

$$K = \frac{f_{osc}}{f_{res}} \quad \text{ecu(1.18)}$$

El usar la ecuación (1.18) implica la implementación de un DDS de selección binaria o decimal distinta, con el resultado de que el esquema circuital cambiará, es por esta razón que esta solución no es la más aconsejable para obtener frecuencias de resolución exactas.

Despejando f_{osc} de la ecuación (1.18) se obtiene:

$$f_{osc} = \frac{K}{f_{res}} \quad \text{ecu(1.19)}$$

La ecuación (1.19) define el método más sencillo y poderoso para obtener una frecuencia de resolución exacta, además de definir el reloj base de tiempo que se necesita como referencia.

Esta técnica se la llama como técnica de variable de resolución (VR). Por ejemplo, si un diseño requiere precisión absoluta, como 2.85714Hz o 0.100000 Hz, es mejor usar la ecuación (1.19) y suplir al DDS de un reloj especial, usando la tecnología VR.

Ejemplo 1

* Se desea obtener pasos de frecuencia de resolución de 2Hz, con un sintetizador cuya palabra de entrada es de 16 bits. Calcular la frecuencia del reloj de oscilación para cumplir este objetivo mediante la técnica VR.

Datos:

$$f_{res} = 2\text{Hz}$$

$$K = 2^{16}$$

Desarrollo:

Usando la ecuación (1.19) se obtiene:

$$f_{osc} = \frac{2^{16}}{2\text{Hz}}$$

$$f_{osc} = 32768\text{Hz}$$

Se necesita de un oscilador TTL de frecuencia exacta a 32768Hz.

1.5.4 EL ACUMULADOR DIGITAL.

Una señal análoga es digitalizada cuando se establece el tiempo de muestreo y la misma por lo general es recuperada en el receptor. Es por esta razón

que tanto para la conversión digital a análoga, como para la conversión análoga - digital, la definición del teorema de Nyquist es única y fundamental.

El teorema de Nyquist dice: " **Una señal limitada en banda que no contiene componentes espectrales mayores que la frecuencia f_m Hz, está determinada en forma única por sus valores en intervalos uniformes menores de $1/(2f_m)$ segundos**".

Para recuperar una señal análoga a partir de sus muestras digitales, es importante la presencia de una señal de muestreo, que defina el número de muestras que se necesitan a la frecuencia de la señal.

Consideremos una señal sinusoidal de frecuencia f_m . La frecuencia de la señal de muestreo según el teorema de Nyquist, será el doble que f_m , esto es:

$$f_{osc} \geq 2 * f_m \quad \text{ecu(1.20)}$$

donde: f_{osc} = frecuencia de reloj de muestreo.

f_m = frecuencia de la señal a generar.

Es importante considerar que si se mantiene la frecuencia de reloj, y el valor de frecuencia de la señal es menor a la considerada inicialmente, el teorema de Nyquist se cumple. Si utilizamos la relación general de que $f = 1 / T$ (T = período), la ecuación (1.20) queda:

$$T_{osc} \leq \frac{T_m}{2} \quad \text{ecu(1.21)}$$

La ecuación (1.21) es de suma importancia, ya que define de manera directa el número de muestras que existen cuando se digitaliza una señal en el límite del teorema de Nyquist. Para comprender mejor esta definición, consideremos un ejemplo numérico:

Ejemplo 2

* Sea una señal de frecuencia 1000Hz, se desea conocer el número de muestras necesarias por período en la que debe ser digitalizada la señal para realizar la repurecación de la señal en el receptor.

Desarrollo:

De la ecuación (1.21), se obtiene la relación:

$$T_{osc} \leq \left(\frac{1}{1000\text{Hz}} \right) / 2$$

$$T_{osc} = 0,5 \text{ ms}$$

Este resultado nos indica que las muestras que se toman de la señal análoga a digitalizar son cada 0,5ms. Si dibujamos este procedimiento se obtiene la figura 1.5.2. Del mismo se observa que el número de muestras necesarias para realizar la conversión a una señal análoga según el teorema de Nyquist es de 2. Es de notar que este valor es cuando nos encontramos en el límite del teorema de Nyquist.

Para frecuencias más bajas que la señal tomada como referencia superior (fm), el teorema de Nyquist se cumple y la señal puede ser recuperada sin problemas.

Si la frecuencia de reloj se mantiene sin variación y la frecuencia de la señal a muestrear es disminuida, el número de muestras se incrementan.

Por simple inspección se obtiene que el número de muestras se incrementan en proporción directa al período de la señal a digitalizar, manteniendo constante el período de oscilación del reloj base de tiempo.

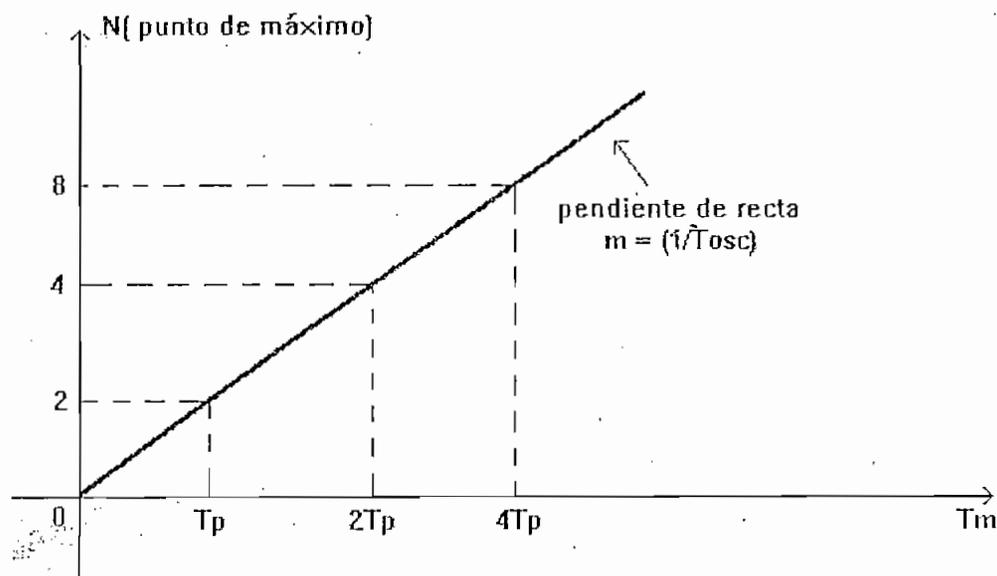


Figura 1.5.3 Número de muestras generadas de una señal de frecuencia f_m .

De la figura 1.5.3, se obtienen las siguientes conclusiones:

- El punto definido como $(1/T_p)$ corresponde al valor máximo de frecuencia que puede generar un Sintetizador Digital Directo, cumpliendo el teorema de Nyquits. En este punto el número de muestras necesarias para generar la señal análoga es de dos, que es definición general en base a Nyquist para cualquier diseño.
- La técnica de la síntesis digital directa define un número llamado punto de máximo, el cual indica el número de muestras totales de información que se pueden proveer para realizar una correcta conversión digital/análoga, cuando se mantiene constante la frecuencia del oscilador de referencia. Este valor esta en estrecha relación con el manejo circuital del mismo.
- En el eje correspondiente a período de la señal, el crecimiento de la variable T_m , denota que la frecuencia de la señal a generar baja, así como el número de muestras aumenta.

Es claro reafirmar que todo este análisis corresponde a la definición previa de que poseemos una señal de reloj invariante en su frecuencia, del cual entonces depende la frecuencia que podamos generar con el sintetizador.

Realizando una adecuada manipulación de la ecuación (1.23), y aplicando la definición de que la frecuencia de una señal es el inverso del período, se obtiene:

$$N = \frac{f_{osc}}{f_m} \quad \text{ecu(1.24)}$$

y graficando esta ecuación se tiene:

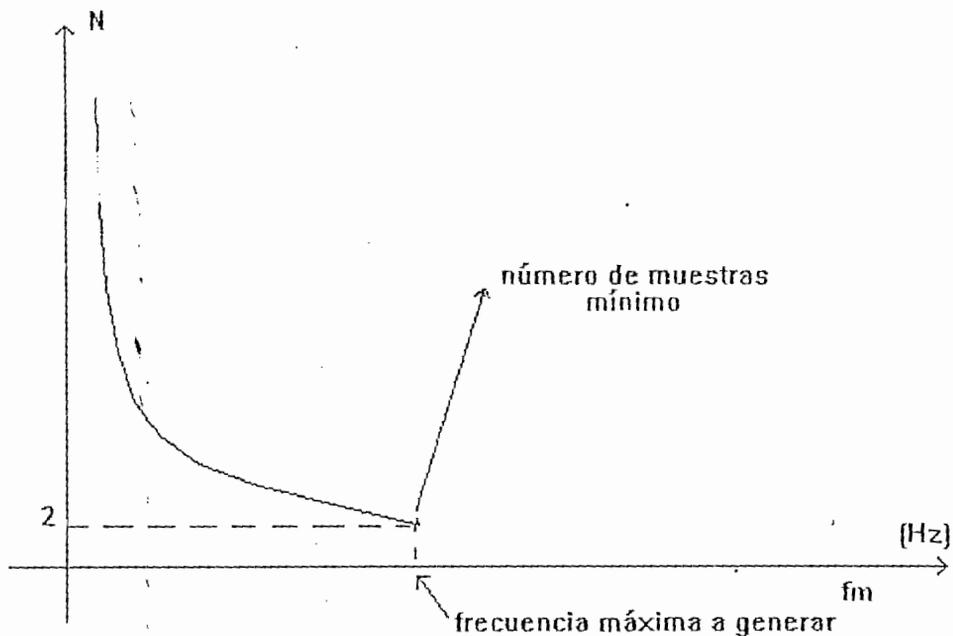


Figura 1.5.4 Número de muestras necesarias para una señal de frecuencia fm.

De la figura 1.5.4 se nota más claramente que existe una frecuencia máxima de generación y que además el número de muestras necesarias para la conversión digital / analógica es de 2. Conforme bajamos el rango de la frecuencia de la señal que se desea obtener, el número de muestras necesarias aumenta de acuerdo a la ecuación (1.24).

Estos resultados dan la idea de que un acumulador digital, debe ser aquella etapa que defina cuantas muestras son necesarias para realizar la conversión a la

Es claro reafirmar que todo este análisis corresponde a la definición previa de que poseemos una señal de reloj invariante en su frecuencia, del cual entonces depende la frecuencia que podamos generar con el sintetizador.

Realizando una adecuada manipulación de la ecuación (1.23), y aplicando la definición de que la frecuencia de una señal es el inverso del período, se obtiene:

$$N = \frac{f_{osc}}{f_m} \quad \text{ecu(1.24)}$$

y graficando esta ecuación se tiene:

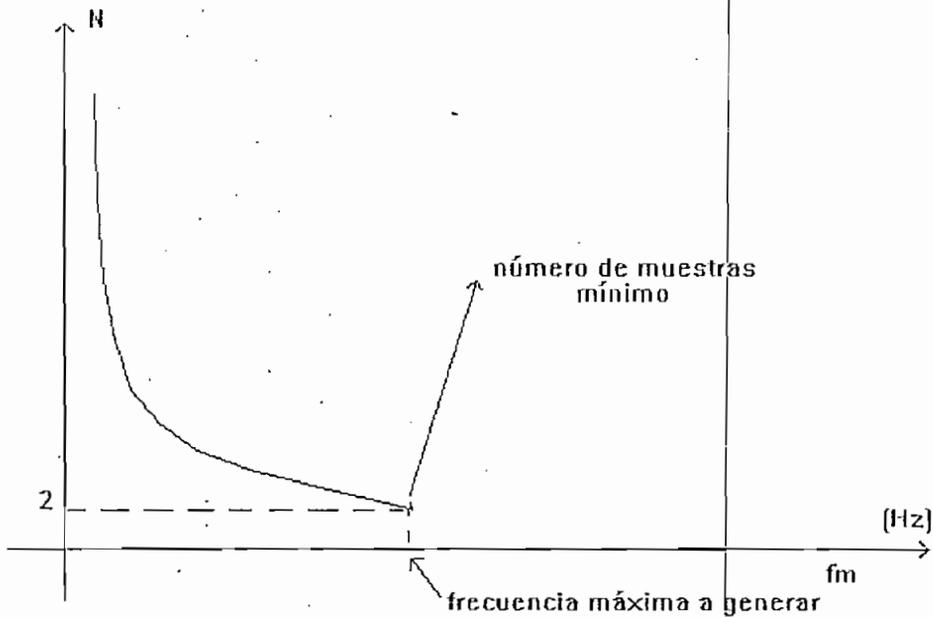


Figura 1.5.4 Número de muestras necesarias para una señal de frecuencia f_m .

De la figura 1.5.4 se nota más claramente que existe una frecuencia máxima de generación y que además el número de muestras necesarias para la conversión digital / analógica es de 2. Conforme bajamos el rango de la frecuencia de la señal que se desea obtener, el número de muestras necesarias aumenta de acuerdo a la ecuación (1.24).

Estos resultados dan la idea de que un acumulador digital, debe ser aquella etapa que defina cuantas muestras son necesarias para realizar la conversión a la

frecuencia requerida. Es claro reafirmar que en este punto se considera un limitante propio del método de la síntesis digital directa, el cual obliga a que la señal del reloj de referencia se mantenga constante.

Es así que se define la función del acumulador digital, la cual se sintetiza en lo siguiente :

" El acumulador digital es aquella etapa que a cada cambio de nivel del reloj de referencia cambia la fase de la señal, definiendo el número de muestras necesarias".

Ejemplo 3

* Cuantas muestras digitales son necesarias por período para realizar una correcta conversión digital/análoga mediante la técnica DDS si se cuenta con un oscilador de referencia de 500khz y la señal que se desea generar es de 100khz.

Datos: $f_{osc} = 500\text{kHz}$.

$f_m = 100\text{kHz}$.

Desarrollo:

De la ecuación (1.24), se obtiene:

$$N = \frac{500\text{Khz}}{100\text{Khz}} = 5$$

se necesitan de 5 muestras.

Un contador digital por lo general el tamaño del paso de conteo generalmente es arreglado a la unidad, a diferencia del acumulador digital en el que el número de muestras necesarias que se necesita, el paso de conteo es modificado.

Para un acumulador digital, el tamaño del paso corresponde al número que define la frecuencia de la señal, f . Esta definición deberá ser tomada muy en cuenta en las relaciones que necesita un sintetizador digital de señales.

1.5.5 MANEJO DE MEMORIAS.

El manejo de memorias corresponde al dispositivo electrónico que almacena la información de la señal a generarse; el mismo tiene estrecha relación con los diferentes tipos de memorias que se consiguen en el mercado, entre los cuales se menciona a: memorias tipo EPROM, PROM, RAM, UVEPROM. La capacidad de almacenamiento de la misma define la estructura del sintetizador.

1.5.5.1 CIRCUITOS CON MEMORIA ROM.

Presentan la condición de fabricación de sintetizadores de muy bajo costo y de buenas características. El diagrama de bloques del sintetizador con memoria ROM se muestra en la figura 1.5.5

El sintetizador está constituido por: el acumulador digital, ROM, DAC y el filtro pasabajos. La forma de onda que se seleccione para generar es grabada previamente mediante las técnicas conocidas para grabar datos ROM.

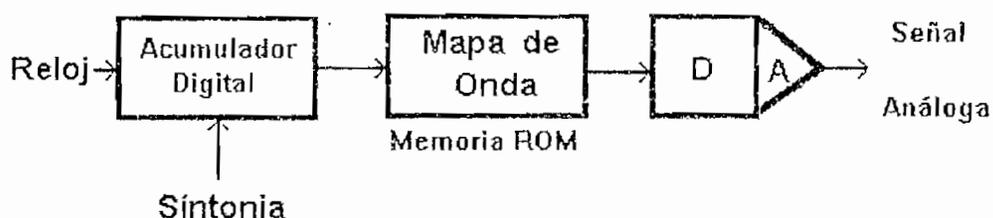


Figura 1.5.5 DDS con memoria ROM.

La principal dificultad de generar formas de onda por este método es el algoritmo de ingreso de datos. El programa para grabar en ROM, constituye de un lector de datos de archivos que previamente han sido ensamblados y compilados en

programas de aplicación, que por lo general son de uso de microprocesadores. Es así que para grabar una señal sinusoidal, los datos ingresados son realizados de uno en uno. En memorias de 8Kbytes, los datos ingresados serán 8192.

Para otra señal, el trabajo es el mismo, además existe la necesidad de sacar la memoria y ubicarla en el programador de EPROMS.

1.5.5.2 CIRCUITOS CON MEMORIA PROM, MULTIPLE.

La solución para obtener sintetizadores que posean ciertas señales previamente grabadas es usar una memoria PROM de mayor capacidad de almacenamiento de datos. Lo cual determina la presencia de un circuito selector de señales desde la PROM. El esquema presentado en la figura 1.5.6 presenta una clara competencia a los osciladores tradicionales de uso en laboratorio.

El sintetizador con PROM de alta capacidad de datos presenta los siguientes bloques: acumulador digital, reloj de referencia, selector de forma de onda, PROM, DAC y filtro pasabajos.

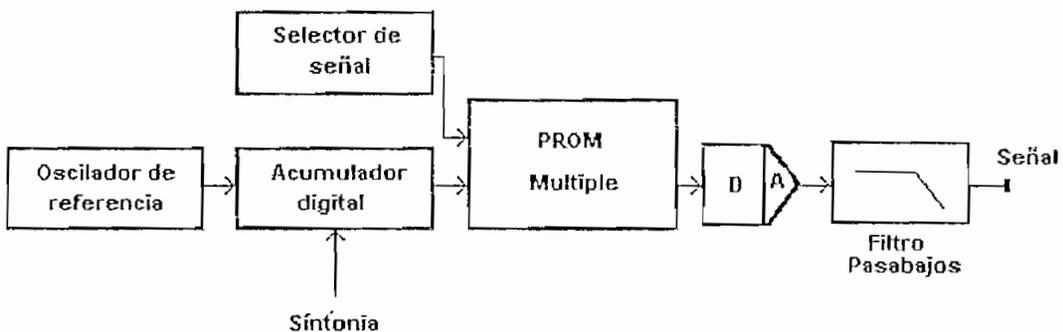


Figura 1.5.6 DDS con PROM múltiple.

1.5.5.3 CIRCUITOS RAM.

Los circuitos RAM poseen la más clara versatilidad en comparación a las estructuras previamente mencionadas. Un circuito RAM nos genera la posibilidad de grabar forma de onda distintas que se desee mediante el ingreso correcto de

datos por formas distintas. El diagrama general del sintetizador con memoria RAM es presentado en la figura 1.5.7

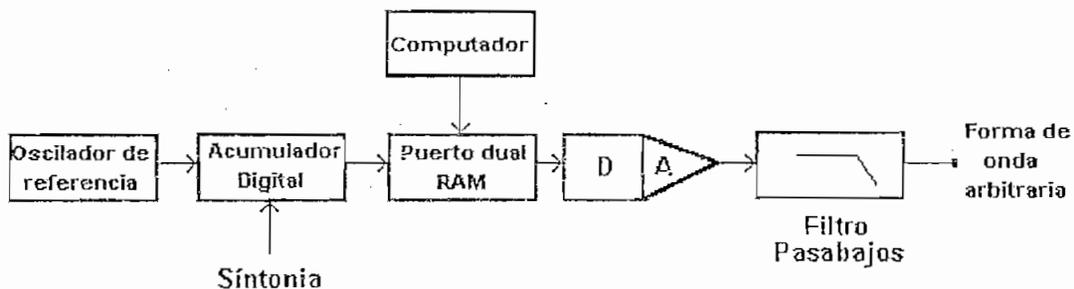


Figura 1.5.7 DDS con Puerto Dual RAM.

Para grabar los datos de la señal en la memoria se tienen varias alternativas: microprocesadores, circuitos especiales, computador personal. En cuanto al uso de microprocesadores se plantean varios esquemas: a) microprocesador con programa b) microprocesador conectado al computador.

El microprocesador con programa se refiere a usar el dispositivo con un programa que genere los algoritmos matemáticos necesarios y los grabe en memoria RAM.

El microprocesador conectado a computador es aquel que se encuentra en constante comunicación serial con el computador, ejecutando solo órdenes, y grabando los datos en memoria RAM.

La opción del computador, es usar el puerto paralelo del computador como un generador de órdenes y datos. Cabe notar que esta opción es la que presenta mayor versatilidad en el proceso de grabar datos en memoria. Una clara desventaja de los circuitos RAM es la condición volátil de los datos grabados.

Cualquier problema en la alimentación de memoria obliga a una retransmisión desde el computador de los datos perdidos. Un circuito PROM tiene los datos grabados sin necesidad de retransmisión de datos.

1.5.6 FORMAS DE ONDA.

Desarrollados pasos de diseño como son: ancho de banda, frecuencia de resolución, reloj base de tiempo, acumulador digital. Definimos el proceso teórico para graficar formas de onda en la DDS.

Para el mismo se parte del análisis matemático que define expresiones generales, obteniéndose como resultado su aplicación para diversas formas de onda en particular, variando solo la definición de la misma (amplitudes en el período).

Una señal eléctrica puede ser representada matemáticamente mediante expresiones que la definen en el tiempo, es decir expresando para cada instante de tiempo correspondiente al dominio, una variable dependiente (amplitud), se logra obtener la definición matemática de la señal.

De la implementación práctica de la DDS, se define el dominio como aquella variable que define el tiempo en que sucede un acontecimiento eléctrico, de tal forma que la variable dependiente se transforma en la información de la amplitud a la salida del circuito.

Realizando una sumatoria de definiciones matemáticas en el tiempo se logra obtener la representación de una forma de onda en particular. El dominio que se utiliza en la DDS es limitante y particular de cada diseño. Es claro notar en este punto que la síntesis define la señal a generar de manera periódica, por lo que a la salida del circuito se obtendrá una señal continua, definida periódicamente.

En la síntesis digital directa para cualquier diseño en particular, se parte de la existencia de un dominio de tiempo, la cual especifica el número de direcciones de memoria en las que se guardará la información de las amplitudes de la señal en el tiempo.

Además se define la amplitud máxima y mínima a la que el DDS puede generar la señal. Esto lo observamos en la figura 1.5.8:

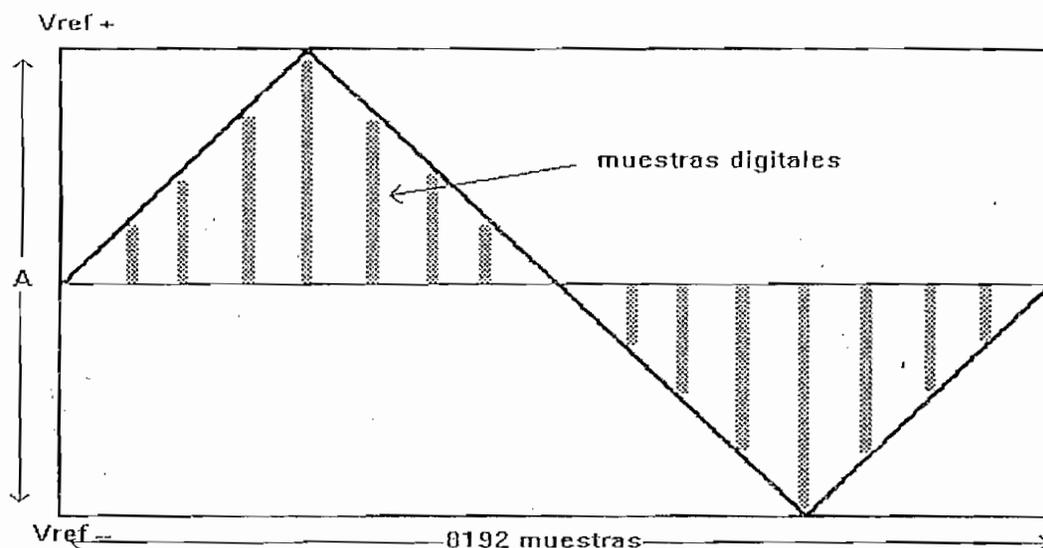


Figura 1.5.8 Mapa de forma de onda, se definen parámetros eléctricos.

T ----> período de la señal.

A ----> amplitud correspondiente al instante de tiempo.

V_{ref+} -> voltaje de referencia positivo, correspondiente al límite superior.

V_{ref-} -> voltaje de referencia negativo, correspondiente al límite negativo.

1.5.6.1 PERIODO.

El dominio que define el período de la señal se relaciona con el dispositivo electrónico que define las amplitudes en el tiempo, este es la memoria.

El dominio del tiempo se lo define:

C_m = número de direcciones accesibles de memoria.

el mismo que representa el intervalo del período del barrido total.

La memoria es direccionable en el intervalo de:

$$i, 1, \dots, C_m \quad \text{tal que: } i = 1, \dots, C_m$$

La función normalizadora, que define todas las direcciones posibles que se graben y lean en memoria, como una función de valor mínimo y máximo, es definida de la forma:

$$\text{Funcion... Normalizadora} = \frac{i}{C_m} \quad \text{ecu(1.25)}$$

donde : $i = 1, 2, \dots, C_m$

de la misma se obtiene los valores máximo y mínimo referidos anteriormente

$$\text{valor máximo} = 1$$

$$\text{valor mínimo} = 1 / C_m$$

Es claro notar que el valor mínimo se acerca a cero, mientras mayor sea el número de direcciones posibles, más se acercará al valor a cero. Relacionando una función matemática normalizada que contenga los límites superiores e inferiores anteriormente mencionados, se obtiene:

$$\frac{k}{C_m} \quad \text{ecu (1.26)}$$

donde: $k = 0, 1, 2, \dots, C_m$

La expresión es de suma importancia ya que nos define la amplitud normalizada en el dominio para cualquier función en particular.

Ejemplo 4:

* Se tiene una memoria RAM de 8Kbytes de capacidad, obtener la función normalizadora para grabar los datos de amplitud ?

Datos:

Capacidad de la memoria: 8Kbytes = 8192 direcciones.

Desarrollo:

$$\text{función normalizador} = k / 8192$$

tal que: $K = 0, 1, 2, \dots, 8192$.

1.5.6.2 AMPLITUD.

Debido a que el DDS necesita la definición de variables eléctricas y a la vez matemáticas, la amplitud es representada por 2^n niveles en la región comprendida entre el límite inferior y superior de amplitud mostrado en la figura 1.5.8 El número de niveles en la región positiva es de $\frac{2^n}{2}$ y $\frac{2^n}{2}$ niveles en la región de amplitud negativa. El número de bits efectivos que serán utilizados a la entrada del conversor digital - análogo, por lo general son para los valores de $n = 8, 16, 32$.

Si $n = 8$, entonces

$$8 \text{ bits} = 1 \text{ byte} \Rightarrow 256 \text{ niveles}$$

El número de niveles efectivos que se utilizarán por región (ver figura 1.5.9), se disminuye en uno. La razón para esta decisión es la de guardarnos un nivel de seguridad, ya que eléctricamente no es recomendable trabajar al límite.

Con esto se asegura la existencia del nivel máximo y mínimo y su consiguiente lectura de memoria. Es así el límite superior e inferior por región queda especificado como:

$$\text{límite práctico} = \# \text{ de nivel teórico por región} - 1 \quad \text{ecu(1.27)}$$

utilizando la ecuación (1.27), se obtiene:

$$\text{límite práctico} = 128 - 1 = 127$$

La ecuación (1.27) es una expresión general que define rangos de trabajo eléctricos y matemáticos para diseños DDS.

1.5.6.3 DDS DE 8 BITS.

Un DDS práctico de construir y que cumpla las especificaciones de un generador de señales comercial, es el que posee una palabra de 8 bits de salida. Para el mismo se define la existencia de una memoria de 8 Kbytes. Refiriendonos a la teoría antes desarrollada se obtiene:

$$C_m = 8192 \text{ -----} \rightarrow \text{número de direcciones}$$

$$\# \text{ niveles} = 2^8 = 256 \text{ niveles.}$$

$$\text{límite práctico} = 256/2 - 1 = 127$$

de las expresiones expuestas se obtiene las condiciones mostradas en la figura 1.5.9.

En el dominio de la amplitud queda definido 255 intervalos y en el dominio del tiempo quedan definidos $C_m = 8192$ intervalos.

Cada intervalo de muestreo en el eje del tiempo corresponde a:

$$t = \frac{T}{C_m} \quad \text{ecu(1.28)}$$

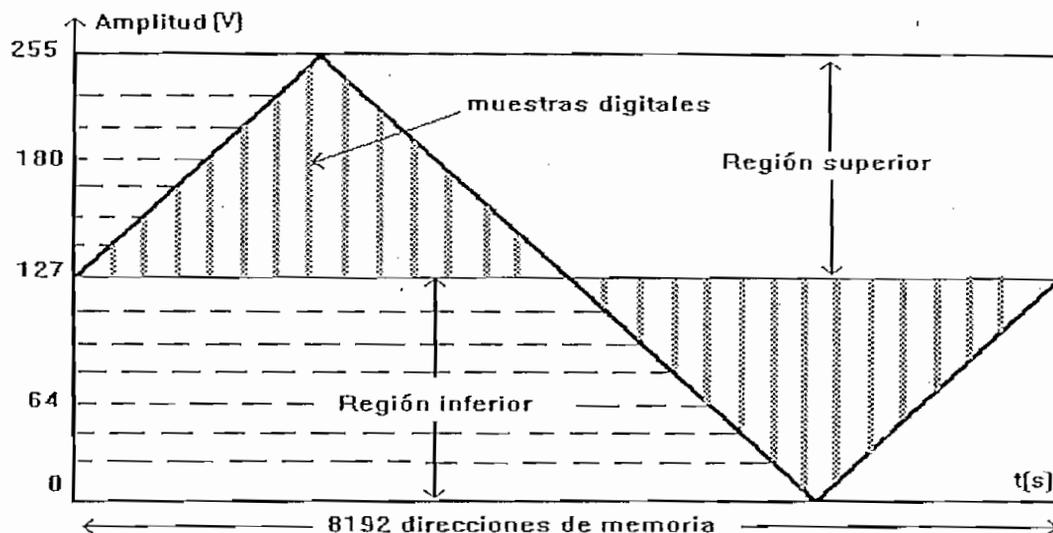


Figura 1.5.9 DDS con memoria de 8Kbytes RAM y palabra de salida de 1 byte.

Restando por definir la amplitud correspondiente a t . Considerando que se busca definir un valor digital y de ese valor hacer la conversión a analógica, entra en este punto el concepto del DDS, el cual es la generación de valores digitales que denotan la función o forma de onda en el tiempo (periódica). En base al número de intervalos, es necesario definir el valor decimal que denota un valor de referencia, que en este caso es definir el potencial 0 de la misma. Definiremos el origen con el valor de 127.

$$127 = 7FH = 0111\ 1111$$

y este el valor de referencia que se aplica en los diseños DDS, en el caso de que la palabra de conversión sea de 8 bits.

Para el caso general la expresión matemática que denota el nivel de referencia es:

$$\text{límite de referencia} = \frac{\text{límite superior}}{2} \quad \text{ecu(1.29)}$$

donde: **límite superior**: corresponde al número máximo en formato binario

que se pueda generar.

límite de referencia: es el número binario que equivale al valor de 0 voltios.

desarrollando la expresión générica:

$$\text{dato digital} = \text{límite de referencia} * \text{función normalizada} \quad \text{ecu(1.30)}$$

Utilizando este proceso de análisis se buscará las expresiones que generen dos tipos de funciones en particular.

* Ejemplo 5

Señal sinusoidal.

La función matemática que define la señal sinusoidal en un período de 2π , es:

$$f(x) = \sin (x)$$

donde: $x \in [0 , 2\pi]$

normalizando la señal de tal forma que para cada amplitud en un período de $C_m = 8192$, exista una función se tiene:

$$f(k) = \sin (2 * \pi * k / 8192)$$

donde: $k = 1, \dots, 8192$ direcciones.

es importante mencionar la transformación de variable que se realizó debido a conveniencias de normalizar la función generadora de la función sinusoidal, está es la siguiente:

$$x = \frac{k}{8192}$$

tomando el valor referencial de nivel de 127, se obtiene:

$$f(k) = 127 * (\sin (2 * \pi * k / 8192) + 1)$$

el término correspondiente a +1 situa el nivel de referencia al valor de 127, es decir cuando K toma el valor de 0, el término entre paréntesis da como resultado el valor de 1; si no se pusiera este número el nivel de referencia escogido no fuera el adecuado.

La función expresada en último término es la que define matemáticamente y eléctricamente a la señal para ser grabada en memoria.

* Ejemplo 6

Señal Triangular

En el caso de una señal triangular, se define una función del tipo que se muestra en la figura 1.5.10:

En la figura 1.5.10 se han definido tres intervalos de período a ser representados matemáticamente, tomando en cuenta los parámetros eléctricos para un diseño de un DDS de 8 bits.

Es así que para un intervalo comprendido en las coordenadas de una recta: (x_1, y_1) y (x_2, y_2) , se define a la ecuación de la recta como solución al problema, obteniendo :

$$y = \left(\frac{y_2 - y_1}{x_2 - x_1} \right) (x - x_1) + y_1 \quad \text{ecu(1.31)}$$

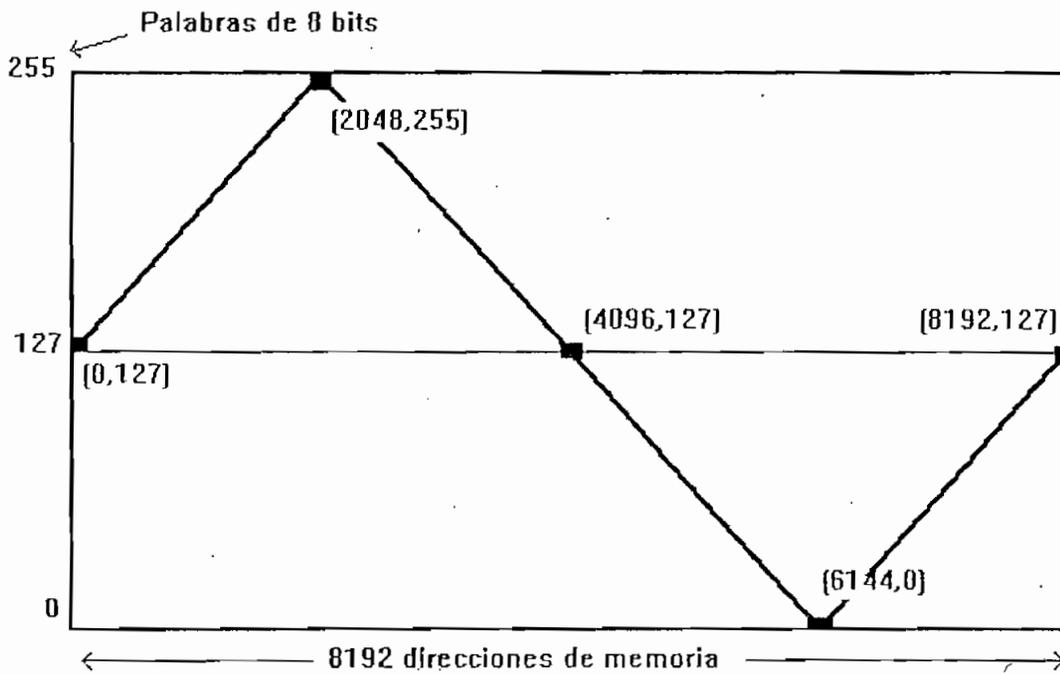


Figura 1.5.10 Procedimiento para buscar función normalizadora de señal triangular.

a) Intervalo de [0 a 2048], direcciones:

$$(x_1, y_1) = (0, 127) \qquad (x_2, y_2) = (2048, 255)$$

$$y = \frac{(255 - 127)}{(2048 - 0)} (x - 0) + 127$$

tal que $x = 1, 2, \dots, 2048$.

b) Intervalo de [2048 a 6144], direcciones:

$$(x_1, y_1) = (6144, 0) \qquad (x_2, y_2) = (2048, 255)$$

$$y = \frac{(255 - 0)}{(2048 - 6144)} (x - 6144) + 0$$

tal que $x = 2048, 2049, \dots, 6144$.

c) Intervalo de [6144 a 8192], direcciones:

$$(x_1, y_1) = (6144, 0) \qquad (x_2, y_2) = (8192, 127)$$

$$y = \frac{(127 - 0)}{(8192 - 6144)} (x - 6144) + 0$$

tal que $x = 6144, 6145, \dots, 8192$.

1.5.7 LA CONVERSION DIGITAL - ANALOGICA DE ALTA VELOCIDAD.

El bloque de conversión digital - análoga toma de la salida del circuito de memoria el número del mapa de la forma de onda que representa la amplitud de la señal, y lo convierte a una señal análoga, el trabajo lo realiza un circuito integrado conversor digital/analógico (DAC).

A causa de la secuencia de números, obtenida a la salida del circuito de memoria, la cual a su vez está sincronizada con la emisión de direcciones del acumulador digital directo, se consigue la representación de la amplitud de la señal.

Debido a la ejecución del muestreo en tiempo real, es posible conseguir a la salida del conversor digital - análogo la señal reconstruida, que representa la señal escogida.

En general, los circuitos integrados conversores no son diseñados para usarse en la síntesis directa de señales; por lo que para obtener salidas de alta calidad en un DDS, se demanda que el conversor no solamente tenga buena estabilidad estática (referido a mantener el valor análogo a la salida sin variaciones)

, sino además que las características dinámicas (cambio de nivel y conmutación) puedan ser emparejadas y controladas.

Debido a que los fabricantes de conversores tipo DAC se encuentran en franca competencia, sus productos tienden a satisfacer las necesidades de la DDS, logrando entonces salidas de alta calidad, con la consiguiente calidad de los sintetizadores digitales directos.

Es por esta razón que se ha desarrollado la teoría que soporte y facilite reglas claras para el escogimiento del DAC correcto.

1.5.7.1 TIEMPO DE CONMUTACION.

Considerando que no existe información acerca del período de trabajo del reloj de referencia, pero que a la vez se conoce información acerca de los tiempos de conmutación de las etapas respectivas en el DDS.

El período del reloj de referencia será la suma de los períodos de conmutación de los retenedores de datos (latches) del DDS más el período de conmutación del circuito de memoria, es decir:

$$\text{Período de reloj} = \text{tiempo conmutación latches} + \text{tiempo conmutación en memoria}$$

El tiempo de conmutación en los retenedores corresponde a los utilizados en el acumulador digital y del circuito de memoria; notado mediante fórmulas se obtiene:

$$T_c = t_{cy} + t_{su} + t_{pd} \quad \text{ecu(1.32)}$$

donde:

T_c = período del reloj de referencia.

t_{cy} = período de conmutación de la memoria.

tpd = período de conmutación de los retenedores en el acumulador digital.

tsu = período de conmutación en el retenedor a la salida de la memoria.

Es de suma importancia recordar la ecuación (1.32), ya que la misma define tanto la calidad de la señal de salida, así como la frecuencia máxima de trabajo.

El DAC que satisface la condición de correcta operación es aquel en que el tiempo de conmutación es menor que el período del reloj de referencia del DDS.

Esta condición guía a la regla: "**Dado un diseño SDD, la más baja frecuencia a la que se puede generar la decide la frecuencia de salida, la cual genera el menor nivel de salidas espurias**".

Una señal de salida limpia es un resultado directo de la forma rectangular de la muestra a la salida del conversor. En otras palabras, se podrían obtener mejores resultados si se usa un reloj de referencia lento para generar pasos de alta calidad, si es que se usa un reloj rápido se obtendrán señales de baja calidad.

La rapidez en la DDS no significa obtener una señal de calidad.

En resumen, el tiempo de conmutación deberá ser mucho menor que el período del reloj de referencia.

Conmutaciones mucho más rápidas producen pasos a la salida que son mucho menos cuadráticas, y no dan un rectángulo. Dando como resultado la obtención de una señal no deseada a la salida del DDS.

1.5.8 DISEÑO DE FILTROS PASABAJOS EMPLEADOS EN EL SINTETIZADOR DIGITAL DIRECTO.

Un filtro elimina porciones no deseadas del espectro de frecuencia. Los tipos de filtros más comunes son: pasabajos, pasa-banda, pasa-altos y rechaza-banda.

En la construcción de un filtro pasabajos y pasabanda, el conocimiento de los datos de partida del problema nos permite definir una plantilla semejante a la de la figura 1.5.11., en cuyo interior debe alojarse la curva de respuesta del filtro.

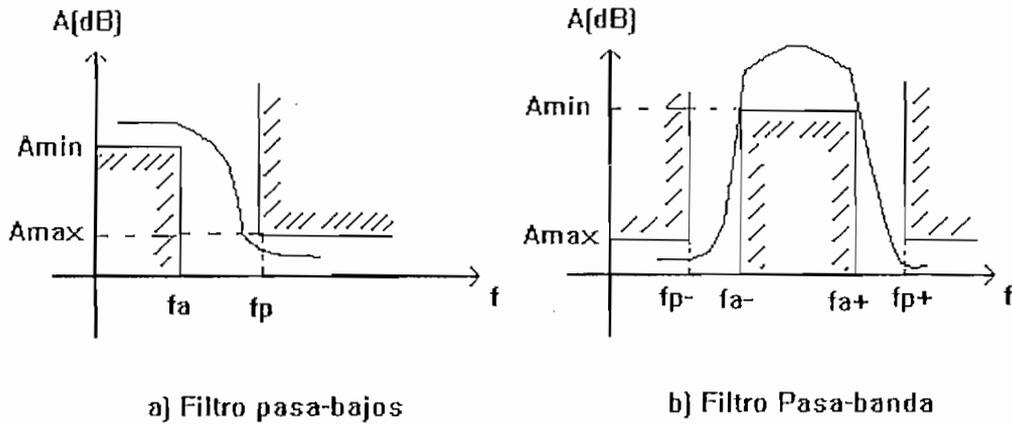


Figura 1.5.11. Plantilla de filtro pasabajos y pasa-banda.

Esta plantilla define lo siguiente:

- La banda de paso o margen de frecuencias en el que el amortiguamiento debe ser inferior a un cierto valor A_{max} , expresado en dB.
- La banda atenuada o margen de frecuencias en el que el amortiguamiento debe ser superior a un cierto valor A_{min} , también expresado en dB.

La banda de paso se halla delimitada por una frecuencia de corte f_p en el caso de los filtros pasa bajo y pasa alto, y por dos frecuencias de corte f_{p+} y f_{p-} en los filtros de paso de banda y de corte de banda.

De la misma forma, la banda atenuada queda delimitada por una o dos frecuencias de corte, f_a o f_{a+} y f_{a-} , según los casos de que se trate.

La frecuencia f_0 es la central:

Los parámetros característicos de una plantilla son tres en los casos de filtros pasabajos y cuatro en los de paso de banda. Estos parámetros son:

- a) La atenuación máxima en la banda de paso, A_{\max} .
- b) La atenuación mínima en la banda atenuada, A_{\min} .
- c) La selectividad k , que expresa la pendiente más o menos abrupta de los cortes y que se halla definida en función de las frecuencias de corte de la forma siguiente:

Filtro de paso bajo:

$$k = \frac{f_p}{f_a} = \frac{W_p}{W_a} \quad \text{ecu(1.33)}$$

Filtro de paso de banda:

$$k = \frac{f_{p+} - f_{p-}}{f_{a+} - f_{a-}} = \frac{W_{p+} - W_{p-}}{W_{a+} - W_{a-}} \quad \text{ecu(1.34)}$$

El ancho de banda B , exclusivamente en los filtros de paso de banda es:

$$B = \frac{f_{p+} - f_{p-}}{f_o} \quad \text{ecu(1.35)}$$

Existe un cierto número de filtros con los cuales se puede obtener una curva de respuesta que quede situada en el interior de la plantilla establecida en cada caso. No obstante, las cualidades y complejidad de cada filtro pueden ser diferentes.

Entre los más eficaces se tiene a los filtros activos, los cuales además producen ganancia, por lo general consisten en redes de resistencias y condensadores, junto con circuitos integrados (generalmente son amplificadores operacionales).

VENTAJAS:

- Tienen una alta impedancia de entrada y baja impedancia de salida.
- Los filtros activos no usan bobinas necesariamente.
- Son fáciles de ajustar a la frecuencia de corte ya que se los hace a través de resistencias variables.

DESVENTAJAS:

- Se necesita de fuentes de alimentación para la polarización del circuito integrado.
- La respuesta de frecuencia del operacional es limitada.

1.5.8.1 FILTROS ACTIVOS DE PRIMER ORDEN (Butterworth).

a) FILTRO RC PASA-BAJOS:

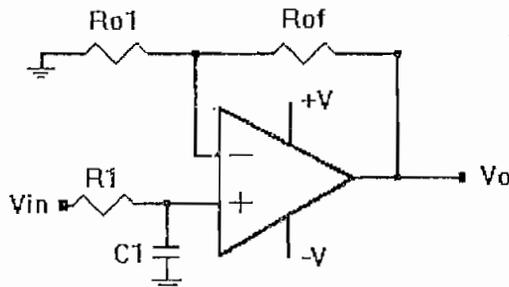


Figura 1.5.12. Filtro activo de primer orden.

El filtro activo de primer orden tiene un polo y la pendiente en la respuesta de frecuencia es de 20dB/dc.

La frecuencia de corte se define según la ecuación (1.36):

$$\omega_p = \frac{1}{RC} \rightarrow f_p = \frac{1}{2\pi RC} \quad \text{ecu(1.36)}$$

En función del amplificador operacional utilizado, la frecuencia de corte puede llegar a valores cercanos al 1Mhz, siempre y cuando se revise la función característica de ganancia versus frecuencia que proporciona el fabricante para el amplificador.

Para señales de amplitud alta, la respuesta de frecuencia del amplificador puede llegar solo a nivel de pocos kilohertzios; en cambio si la amplitud de la señal de entrada se reduce trasdicamente, la respuesta de frecuencia del amplificador se aumenta al nivel mencionado anteriormente.

b) FILTRO DE SEGUNDO ORDEN PASA BAJOS CON UN SOLO AMPLIFICADOR

Los filtros de segundo orden pueden modelar las características de un flitro ideal en forma más cercana que un filtro de primer orden, ya que en la respuesta de frecuencia la pendiente es de 40dB/dc y esta es mayor que la que ofrecen los filtros de primer orden. En la figura 1.5.12 se muestra un filtro de segundo orden.

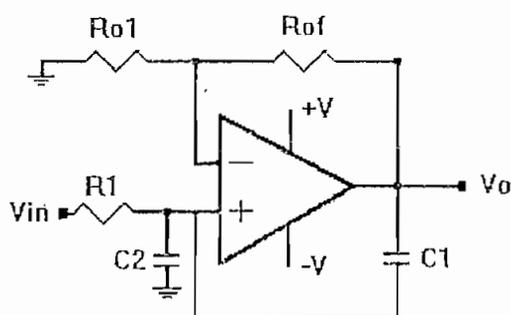


Figura 1.5.12 Filtro pasabajos de segundo orden.

la frecuencia de corte esta definida por la ecuación (1.37)

$$W_o = \sqrt{\frac{1}{C_1 C_2 R_1 R_2}} \quad \text{ecu(1.37)}$$

además se define ϵ como la razón de amortiguamiento, tal como se muestra en la ecuación (1.38).

$$\xi = \sqrt{\frac{C_2}{C_1}}$$

ecu(1.38)

ξ toma valores entre 0,7 y 1.

c) FILTRO PASA BANDA DE SEGUNDO ORDEN

Este filtro es el que presenta mayor versatilidad en el diseño y trabajo. Puede llegarse a construir filtros pasabanda en el rango de 0Hz a 1,2mhz. En la figura 1.5.13 se indica el filtro pasabanda.

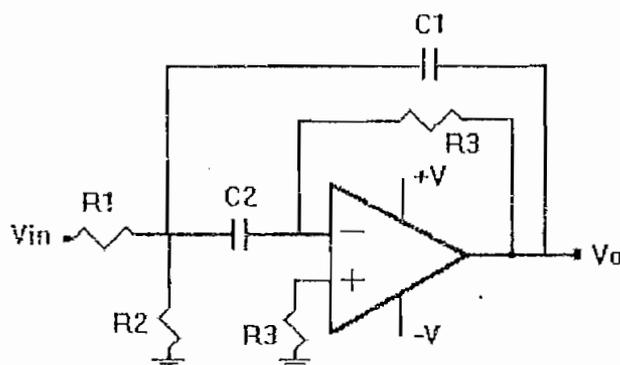


Figura 1.5.13 Filtro pasabanda de segundo orden.

La frecuencia central f_0 en la banda de paso esta definida de acuerdo a la ecuación (1.38).

$$\omega_0^2 = \frac{(R1 + R2)}{R1 \cdot R2 \cdot C1 \cdot C2 \cdot R3} \quad \text{ecu(1.38)}$$

Si el factor de calidad aumenta el ancho de banda disminuye y si el factor de calidad, disminuye el ancho de banda aumenta. En filtros activos de este tipo el factor de calidad máximo a alcanzar esta en el rango de 1 a 10.

$$B = \frac{f_0}{Q} \quad \text{ecu(1.39)}$$

CAPITULO II

2.1. Descripción General del Equipo

2.1.1 Alternativas de diseño.

Tipos de sintetizadores que pueden fabricarse. Definición de parámetros.

2.1 DESCRIPCION GENERAL DEL EQUIPO.

Mediante la técnica de la Síntesis Digital Directa se construirá un sintetizador digital de señales controlado desde el computador. En diagrama de bloques en la figura 2.1.1 se indica al equipo a construirse:

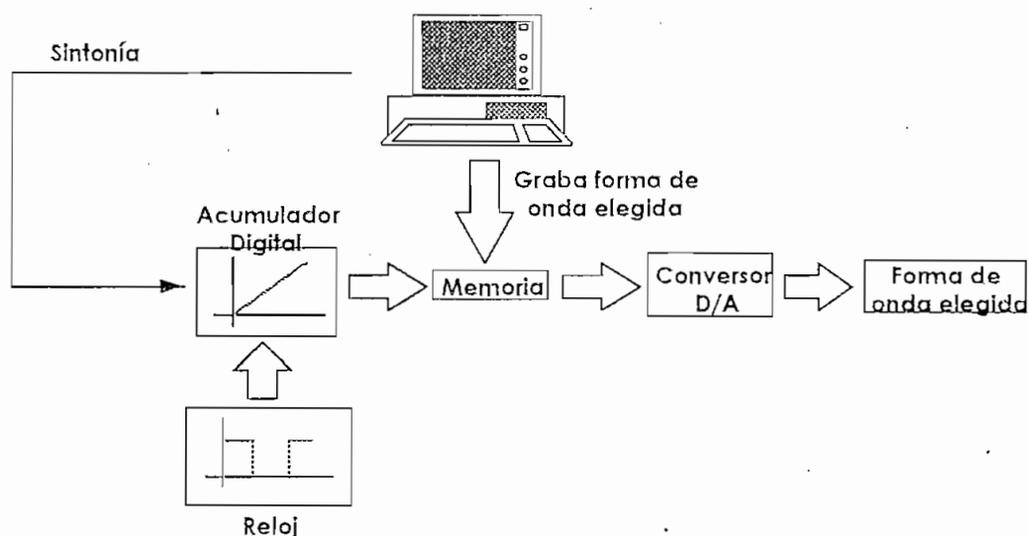


Figura 2.1.1 Diagrama de bloques del Sintetizador.

El equipo está compuesto por el acumulador digital, reloj de referencia, sintonizador, memoria, conversor digital/análogo y circuito amplificador a la salida para manejo de carga.

El acumulador digital es el circuito que genera las direcciones de acceso a memoria, de tal forma que a la salida de la memoria se obtenga un dato digital. El acumulador digital es capaz de sintonizar la frecuencia de generación y a la vez definirla. Es decir tiene dos propósitos: selección de frecuencia y sintonización.

En la memoria se graban los datos de la forma de onda que se desea generar. Estos datos son grabados desde el computador mediante la lectura de un archivo binario seleccionado convenientemente. El proceso descrito anteriormente se lo ejecuta mediante la utilización del puerto paralelo.

El conversor digital/análogo se encarga de cambiar el dato digital generado por memoria a una señal análoga. No posee parámetros para manejo de cargas pequeñas. Debido a esto se implementa un circuito amplificador que responda al rango de frecuencias que se desea generar, para el manejo de cargas.

Debido a las muchas aplicaciones de la técnica aplicada en este trabajo se define el campo en la que ella actúe; este es el de obtener señales de radio frecuencia en valores no mayores a 3MHz.

De este limitante y como consecuencia del mismo, el sintetizador digital de señales tiene las siguientes especificaciones técnicas de acuerdo a los bloques constitutivos del mismo.

• **Frecuencia del oscilador:**

Para el reloj de referencia se utilizan osciladores TTL de distintos valores. Los osciladores utilizados se indican en el cuadro 2.1.1.

Frecuencia del oscilador = 1000000 Hz
Frecuencia del oscilador = 2000000 Hz
Frecuencia del oscilador = 4915200 Hz
Frecuencia del oscilador = 6000000 Hz

Cuadro 2.1.1 Frecuencia del reloj de referencia del DDS.

Se escogieron estos valores debido a que la frecuencia del reloj define a que frecuencias puede generar señales el sintetizador. Además el reloj de referencia indica la máxima frecuencia de la señal.

De la ecuación (1.14); para la máxima frecuencia a generar se necesita de un reloj del doble de la frecuencia máxima, por está razón se utiliza el oscilador de 6 Mhz, limitando el rango de frecuencias a 3MHz.

Todo proceso digital implica que se definan niveles de generación de señales. Es decir en un circuito análogo se puede generar frecuencias continuas, en cambio en un circuito que implique procesos digitales las frecuencias a generar ya no son continuas, sino que se generan cada ciertos valores, como por ejemplo:

Ejemplo 7:

* Calcular la frecuencia de resolución de un sintetizador digital que tiene una palabra de sintonización de 8 bits y el reloj de referencia de 1MHz.

Desarrollo:

Datos: $f_{osc} = 1'000.000 \text{ Mhz.}$

$M = 8$

De la ecuación (1.17) se obtiene:

$$f_{res} = \frac{1000000}{2^8} = 3906,25 \text{ Hz}$$

Se pueden generar señales múltiplos de 3906,25Hz.

La frecuencia del reloj de referencia define la frecuencia de resolución estándar del sintetizador; esto se observa más claramente usando la ecuación (1.17). Los resultados se muestran en el cuadro 2.1.2.

- **Rango de Frecuencia de salida:**

El rango de frecuencias de salida que el DDS genera ondas sinusoidales, sin que se produzcan distorsiones. No se toma en cuenta para definir este parámetro la presencia de señales espurias y armónicas, propias de sistemas digitales a estas frecuencias.

La frecuencia máxima de generación esta definida por el oscilador de referencia de más alto valor, tal como se indico anteriormente.

de la ecuación (1.14) se tiene:

$$f_o = \frac{f_{osc}}{2} = \frac{6000000}{2}$$

$$\text{rango de frecuencia} = 3\text{Mhz}$$

- **Bits de la palabra de control:**

La sintonización de frecuencia es controlada por los bits de la palabra de control. Los bits de la palabra de control son los necesarios para seleccionar las frecuencias posibles dentro del parámetro de Nyquist. De la ecuación (1.17) se tiene que la frecuencia de resolución depende de dos parámetros específicos: la frecuencia del oscilador y el módulo del acumulador digital. Debido a que el módulo del acumulador debe tener un número de bits determinado se necesita definir cual será el módulo del acumulador digital a construirse.

Para el módulo del acumulador digital se tienen tres opciones : módulo de 8 bits, 16 bits. Se pueden tener mayor cantidad de bits pero el sistema se hace cada vez más complejo.

a) módulo de 8 bits:

De la ecuación (1.17) y escogiendo el reloj de referencia de 6MHz, se tiene que la frecuencia de resolución es:

$$f_{res} = \frac{6000000}{2^8} = 23437,5\text{Hz}$$

b) módulo de 16 bits:

De la ecuación (1.17) y escogiendo el reloj de referencia de 6MHz, se tiene que la frecuencia de resolución es:

$$f_{res} = \frac{6000000}{2^{16}} = 91,55 \text{ Hz}$$

De lo expuesto anteriormente se deduce que la mejor opción es escoger el módulo de 16 bits, ya que nos da una frecuencia de resolución menor en comparación al de 8 bits.

$$\text{número de bits de control} = 16$$

- **Resolución de Frecuencia:**

La frecuencia de resolución esta definida por la ecuación (1.17), siendo seleccionable a cuatro valores. Se provee además de una entrada externa para conectar un oscilador de referencia a nivel TTL que tenga la capacidad de variar su frecuencia.

Frecuencia del oscilador (Hz)	Frecuencia de resolución(Hz)
1000000	15.25878906
2000000	30.51757813
4915200	75.00000000
6000000	91.55273438
externa, oscilador TTL	0 - 91.55273438

Cuadro 2.1.2 Frecuencia de resolución del DDS.

- **Impedancia de salida:**

Es la impedancia que se presenta en el toma de salida. Generalmente se lo representa como Z_o . Será definida de acuerdo a resultados experimentales.

2.1.1 ALTERNATIVAS DE DISEÑO DEL EQUIPO.

En el capítulo 1.5.5 se presentan varias alternativas de sintetizadores que usan memorias ROM y RAM, entre los que se tiene:

- a) Sintetizador con memoria EPROM
- b) Sintetizador con memoria EPROM Múltiple.
- c) Sintetizador con memoria RAM.

En esta sección se indican los diagramas de construcción circuital para las respectivas alternativas, de modo que cualquier constructor de circuitos electrónicos lo pueda fabricar, dentro del rango de generación de señales de hasta 3MHz.

2.1.1.1 DIAGRAMA GENERAL.

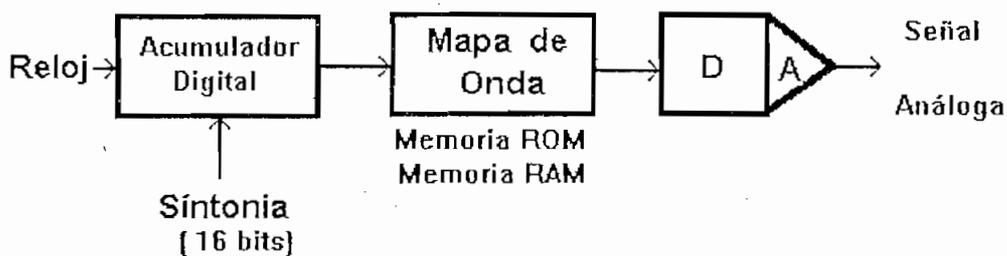


Figura 2.1.2 Diagrama de bloques del DDS.

Cualquier sintetizador digital directo basado en la técnica bajo estudio está constituido por tres bloques: acumulador digital, circuito de memoria y conversor digital-análogo. A su vez cada uno de estos bloques tienen señales de control y datos, entre las señales de control está la palabra de control de frecuencia (16bits)

y la de selección de operación de circuitos⁹. Las señales de datos corresponden a la señal de reloj y a las señales para grabar en memorias.

a) SINTETIZADOR CON MEMORIA EPROM SIMPLE

El Sintetizador de EPROM Simple consiste de una memoria CI2764 de 8KBytes en la que se almacena una sola forma de onda. Se almacenan los datos que se desean grabar de la señal generalmente en forma manual (de dato en dato). Este proceso es tedioso y complicado, pero una vez que se han almacenado todos los datos se crea un archivo y para volver a utilizar los datos en otra memoria, solo se recurre a grabarlos.

Este sintetizador presenta buenas características y es de bajo costo. En la figura 2.1.3 se presenta el circuito puede ser implementado en la práctica. Las entradas de datos digitales S1, S2,.....S16 corresponden a la palabra de control del acumulador digital directo y sirven para selección de frecuencia. Además se provee la entrada del oscilador de referencia.

b) SINTETIZADOR CON EPROM MULTIPLE.

Se utiliza una memoria CI 27010 de 64Kbytes. Permite el almacenamiento de 8 formas de onda previamente seleccionadas y grabadas en el mapa de memoria de acuerdo al cuadro 2.1.3

dirección inicial	dirección final	señal grabada
0000H	1FFFH	señal 1
2000H	3FFFH	señal 2
4000H	5FFFH	señal 3
6000H	7FFFH	señal 4
8000H	9FFFH	señal 5
A000H	BFFFH	señal 6
C000H	D000H	señal 7
E000H	FFFFH	señal 8

Cuadro 2.1.3 Intervalos de direcciones para EPROM múltiple.

⁹ Se refiere a la selección de operación, no operación, habilitación de escritura, estado de alta impedancia, etc.

Una vez grabados los datos en memoria, la señal que se obtiene a la salida se escoge mediante la utilización de 3 selectores que segmentan a la memoria en 8 bloques de 8KBytes cada una respectivamente. La selección es realizado de acuerdo al cuadro 2.1.4:

Selector 1	Selector 2	Selector 3	Selección
0	0	0	Señal 1
0	0	1	Señal 2
0	1	0	Señal 3
0	1	1	Señal 4
1	0	0	Señal 5
1	0	1	Señal 6
1	1	0	Señal 7
1	1	1	Señal 8

Cuadro 2.1.4 Selección de formas de onda para EPROM múltiple.

En la figura 2.1.4 se muestra el circuito implementado. Este circuito presenta una mayor capacidad de generación de formas de onda. Es un poco más costoso que el sintetizador con memoria EPROM simple. Las entradas de datos digitales S1, S2,.....S16 corresponden a la palabra de control del acumulador digital. Además se provee la entrada del oscilador de referencia.

c) EQUIPO CON MEMORIA RAM

Presenta alta versatilidad respecto a los circuitos con memoria EPROM, pero necesita de un circuito de interfase para control y grabado de datos en memoria, encareciendo su costo. Se pueden generar tantas formas de onda como sean posibles ser generadas en computador y transmitidas por el puerto serial o paralelo, hasta la memoria.

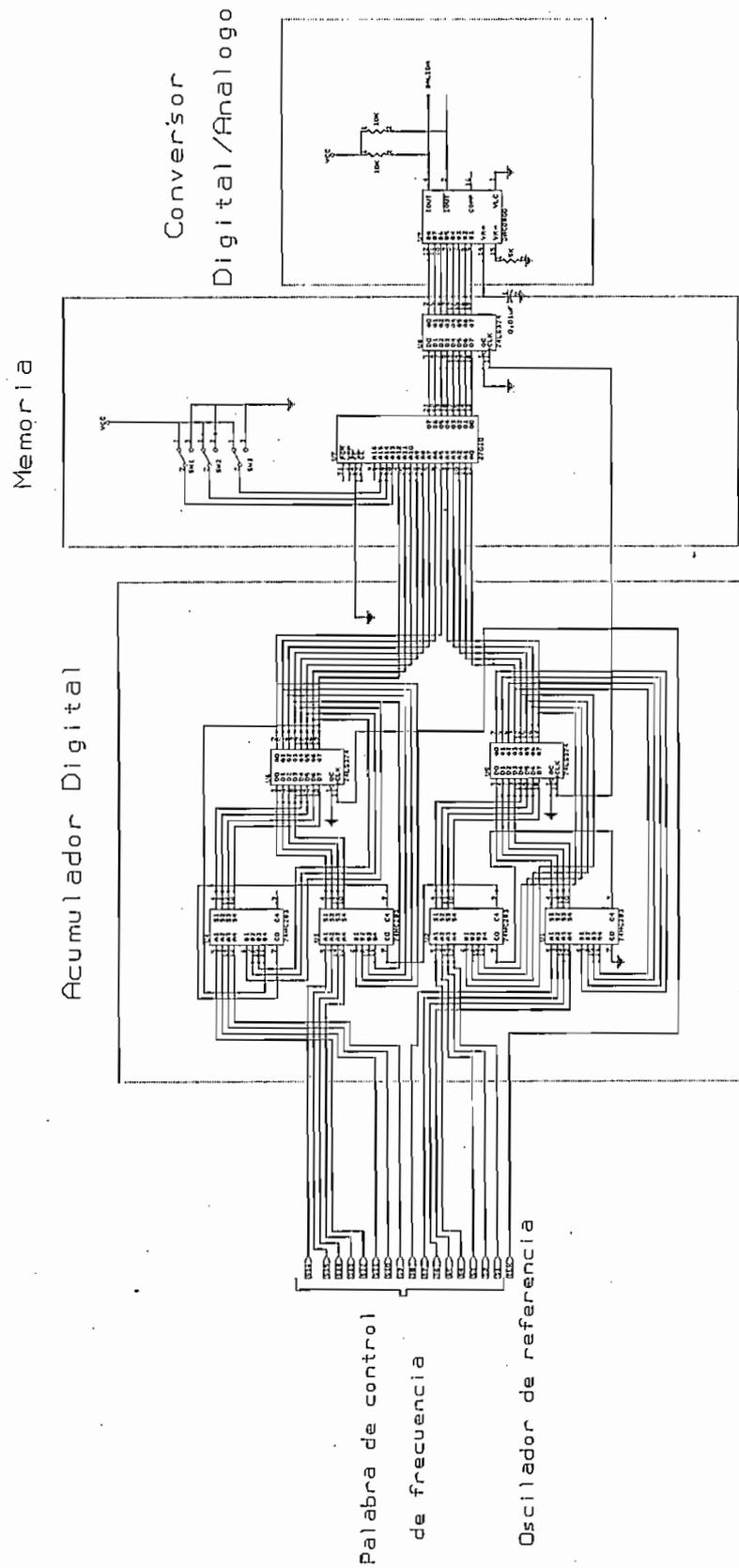


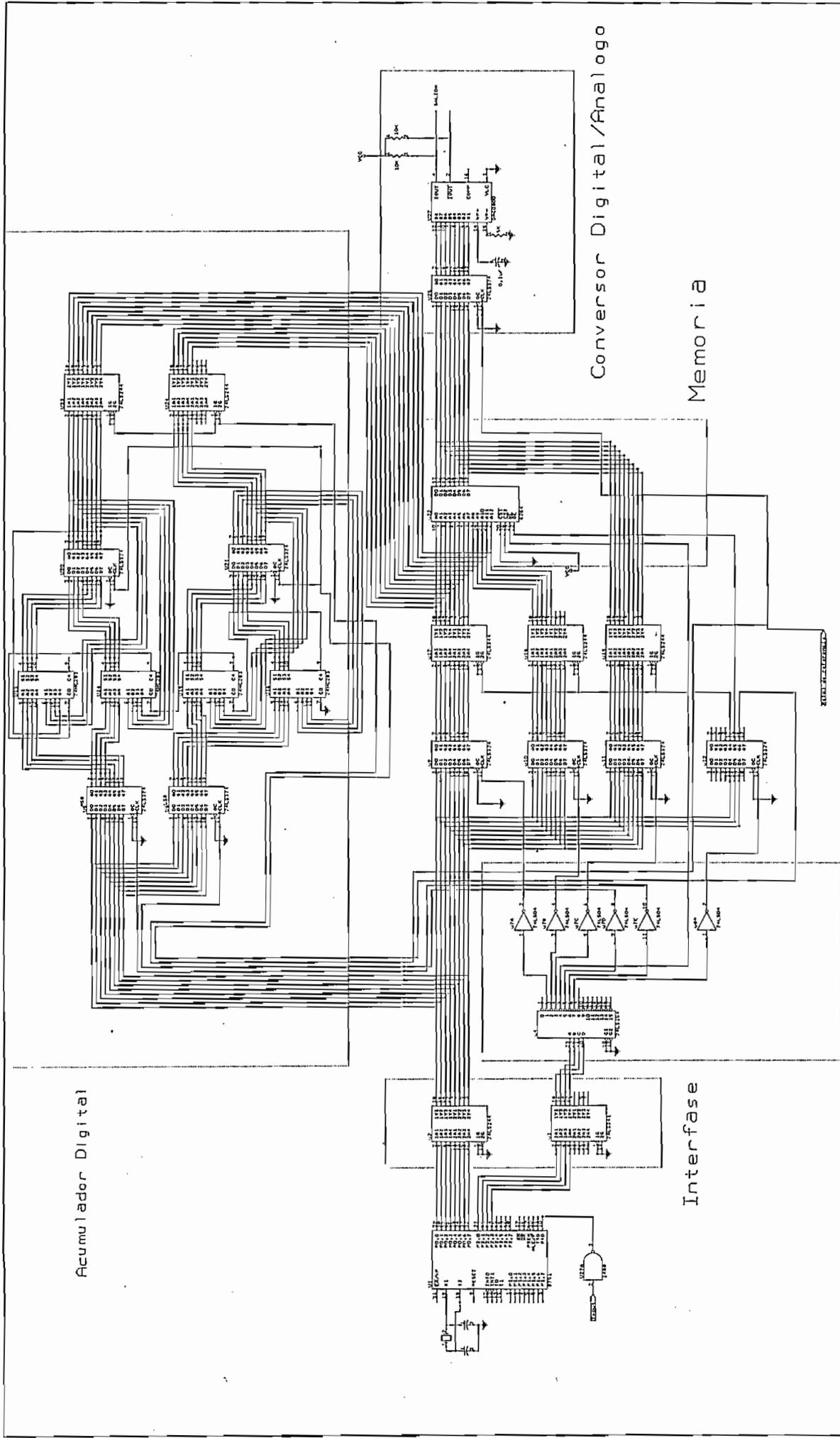
Figura 2.1.4

En el gráfico 2.1.5 se presenta el circuito de grabado en memoria RAM, usando un microprocesador 8751 el cual cuenta con memoria para grabar programas en él.

El microprocesador funciona como un interfase de control y escritura de datos en memoria RAM. El computador manda los datos de la forma de onda a grabar de manera serial; mediante instrucciones propias del equipo, este graba los datos en memoria.

Debido a que la implementación final del sintetizador se hizo uso del puerto paralelo, el diseño se lo realizó a nivel de bus de datos y control. Es decir, al circuito final construido se puede añadir fácilmente un microprocesador, acoplándolo al bus del sintetizador. En la figura 2.1.5 se muestra el diagrama circuital final con el microprocesador y el sintetizador. Con esta opción se podría usar el pòrtico serial para transmisión, con la consecuente desventaja que el tiempo que se demorará en grabar los datos en memoria se incrementará.

Debido a que el objetivo del presente trabajo no es usar el pòrtico serial, solo se indicará el circuito que debe utilizarse sin definir programación al nivel del microprocesador.



Acumulador Digital

Interfase

Memoria

Convertor Digital/Analogo

Circuito de Control

Figura 2.1.5

2.2. Especificaciones Técnicas del diseño

2.2.1 Acumulador Digital.

Descripción detallada y definición de parámetros.

2.2.2 Circuito de memoria.

Memoria utilizada en el DDS.

2.2.3 Conversor Digital-Analogo.

Estudio del conversor utilizado.

2.2.4 Filtro de Salida.

Filtro pasabanda de segundo orden.

2.2 ESPECIFICACIONES TECNICAS DEL DISEÑO.

2.2.1 ACUMULADOR DIGITAL.

Del diagrama de bloques de la figura 2.1.1 el acumulador digital corresponde al sintonizador y selector de frecuencia. Su función es la de generar direcciones a memoria en la cual se han grabado datos de la forma de onda. De acuerdo a la frecuencia a generar y del reloj de referencia escogido se utiliza la ecuación (1.24) para determinar cuantas muestras son necesarias. De la figura 1.5.3 se deduce que el circuito electrónico a construirse debe direccionar a memoria en forma lineal dependiendo de la frecuencia que se haya escogido.

En la figura 2.1.6 se muestra el diagrama de bloques de un acumulador digital.

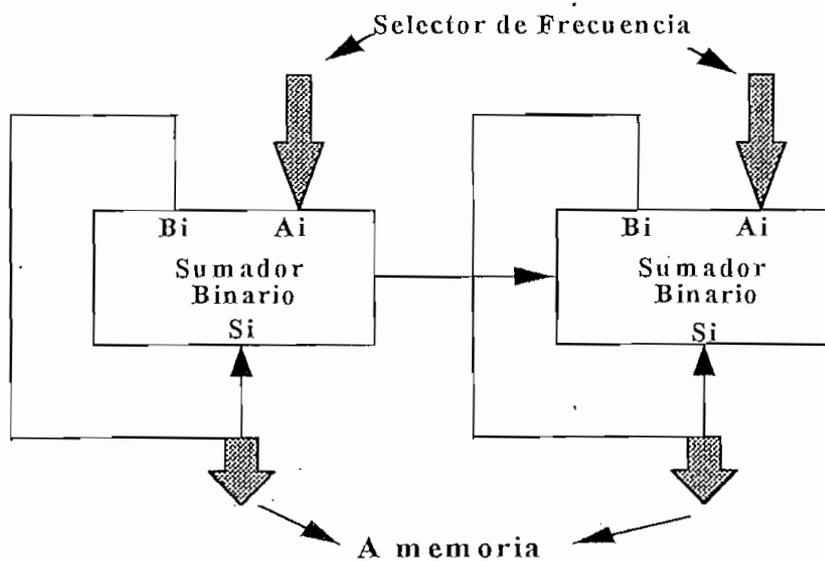


Figura 2.1.6 Diagrama de bloques del Acumulador Digital.

El acumulador tiene una palabra de control de frecuencia que genera pasos de muestras determinados, el cual a su vez es definido por el usuario.

El número de muestras en el límite inferior es de 8192, y el número de muestras en el límite superior de la frecuencia es de 2. El circuito que genera la

rampa lineal es un sumador de 2 operandos (A_i y B_i), que se encuentran realimentados. En la figura 2.1.6 se indica un acumulador digital en el cual se puede observar la palabra de control de frecuencia, los sumandos, el proceso de realimentación de datos y la salida a memoria.

El proceso de realimentación es muy importante para que el circuito genere constantemente y en períodos exactos las direcciones de muestras a memoria. El sumador de 16 bits usado en la DDS consiste de 4 sumadores (74HC283), y de dos retenedores de dato tipo D (74LS374). Para comprender mejor el funcionamiento de los sumadores binarios se explicará el sumador usado en la síntesis digital.

2.2.1.1 SUMADORES BINARIOS DE 4 BITS.

Se utilizan 4 sumadores binarios de alta calidad de conmutación. Uno es el 74HC283 de la familia TTL. Estos tienen la función de sumar 2 números binarios de 4 bits, con un bit de bandera que indica que se ha lleva uno. Operando por si solos tienen la capacidad de realizar sumas binarias cuyo resultado sea el número 15, pero utilizándolos en cascada tienen la capacidad de ampliar sus sumas en el factor de 2^M , siendo M el número de bits que definen la palabra de entrada de control de frecuencia.

Para el circuito acumulador digital directo se deben utilizar 4 sumadores de 4 bits cada uno, llegando a obtener una palabra de control de 16 bits. La construcción en cascada de los 4 sumadores mediante la conexión del bit de bandera (carry) al bit de entrada de recibo, posibilita contar con un sumador de palabras de 16 bits.

Los retenedores 74LS374 son los encargados de producir la realimentación hacia los sumadores del resultado de las sumas producidas, generando pasos de direccionamiento hacia la memoria, obteniendo muestras distintas hacia el conversor. Es decir si el usuario ingresa en la palabra de control el número 3, se producirán sumas de 3 en 3, con los resultados de 3, 6, 9, 12,65535, lo que

significa que se extraera los datos cada 3 localizaciones (direcciones) en la memoria.

2.2.1.2 ACUMULADOR DIGITAL DE 16 BITS.

El principio básico para la construcción del acumulador digital en base a sumadores binarios consiste en definir la palabra de control y las direcciones a memoria requeridas.

En base a la definición de que 16 son los bits de la palabra de control, se deduce que se necesitan 4 sumadores binarios, cada uno aporta con 4 bits a la palabra de control.

Es importante realizar la conexión de los bits de bandera entre cada uno de los sumadores para implementar en la práctica las sumas reales.

El bus de datos que sale de los sumadores se dirigen hacia los retenedores para realizar el proceso de realimentación hacia los sumadores y para direccionar a memoria el dato requerido a su salida.

Cabe notar que el acumulador digital directo no es contador binario, las señales eléctricas generadas en los líneas de direccionamiento a memoria son aperiódicas en ciertos casos como el que se muestra en el cuadro 2.2.1, donde se nota que para sumas de 3 en la línea C, la señal producida es 001011, la cual no es periódica.

El acumulador presenta características mucho más versátiles, las cuales a su vez son seleccionables.

El acumulador puede funcionar como un contador binario, si es que a la entrada de la palabra de control de sintonía ingresa el número 1.

D	C	B	A	Suma de 3 en C
0	0	0	0	0
0	0	0	1	
0	0	1	0	
0	0	1	1	0
0	1	0	0	
0	1	0	1	
0	1	1	0	1
0	1	1	1	
1	0	0	0	
1	0	0	1	0
1	0	1	0	
1	0	1	1	
1	1	0	0	1
1	1	0	1	
1	1	1	0	
1	1	1	1	1

Cuadro 2.1.5 Señales aperiódicas generada en el acumulador digital.

El circuito del acumulador digital, implementado se presenta en la figura 2.2.2; los circuitos integrados U1 y U2 (74LS374) son usados para seleccionar la palabra de control de sintonía. Los circuitos integrados denominados como U4, U5, U6 y U7 (74HC283) son los sumadores binarios en cascada convirtiéndose en conjunto en un sumador de 2 palabras de 16 bits.

Los circuitos integrados U8, U9 (74LS374) son los encargados de realizar el proceso de realimentación y el direccionamiento a memoria. Los circuitos integrados U10, U11 (74LS244) son incrementadores de corriente y presentan alta impedancia según la señal de control definida en la figura 2.2.2 como habilitador del acumulador digital.

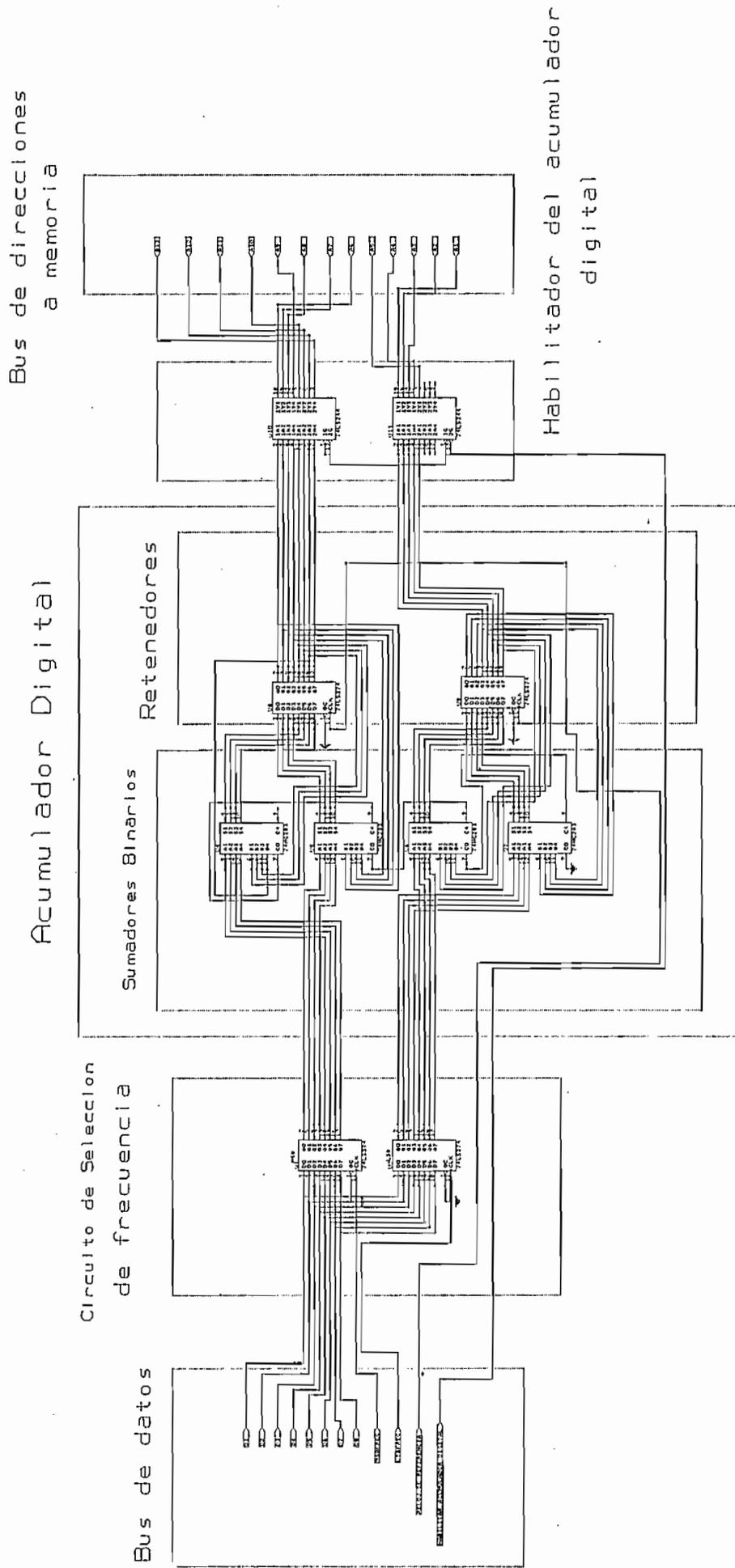


Figura 2.2.2

La palabra de control de frecuencia tiene 16 bits, los cuales han sido segmentados en dos bloques de 8 bits cada uno, en la figura 2.2.3 se muestra como se dividieron y nombraron cada uno de estos bloques.

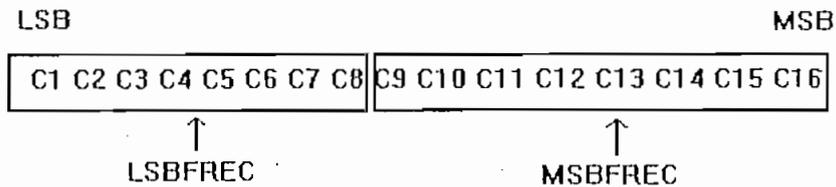


Figura 2.2.3 División de palabra de control.

En la figura 2.2.2 se tiene:

a) **MSBFREC.**

El MSBFREC es un pulso eléctrico que cuando se ejecuta se graba el dato del byte MSBFREC que corresponde a la palabra de control de frecuencia.

b) **LSBFREC.**

El LSBFREC es un pulso eléctrico que cuando se ejecuta se graba el dato del byte LSBFREC que corresponde a la palabra de control de frecuencia.

c) **Reloj de referencia.**

Es la señal del reloj de referencia. La selección del mismo se realiza en forma análoga mediante conmutadores.

d) **Bus de datos.**

Son los datos que se mandan para seleccionar la frecuencia a generar.

Se hacen uso de capacitores eliminadores de ruido (0,1 μ F) a la fuente para aumentar la confiabilidad del circuito. Debido a la gran cantidad de conexiones se realizo una tarjeta de doble lado, los negativos del acumulador digital se presentan en Anexos.

2.2.2 CIRCUITO DE MEMORIA

El circuito de memoria corresponde a una memoria RAM de 8Kbytes de capacidad. Para el sintetizador digital a construir se escogio la memoria 6264 que presenta las siguientes características:

- Tiempo de acceso a datos escritos en memoria de 100ns en el rango de temperatura de 0 a 70°C.
- Bajo consumo de potencia
 - 100 mA en plena operación.
 - 30 mA en standby.
- Organización 8K x 8.

El circuito de memoria esta compuesto por la memoria RAM, las líneas del acumulador digital y un circuito integrado 74LS374 para proveer sincronización total al circuito con el reloj de referencia. En el presente punto no se explicara el procedimiento mediante el cual se graban los datos en memoria ya que se desarrollara más ampliamente en el punto 2.3.4, correspondiente al circuito externo para grabado de datos en memoria.

2.2.3 CONVERTOR DIGITAL-ANALOGO

El convertor utilizado en el sintetizador digital es el DAC0801 del fabricante National Semiconductor, es un convertor de 8 bits digital a análogo de

alta velocidad y alta corriente de salida. El tiempo de conmutación es cercano a los 100ns. El DAC0801 además permite obtener voltajes diferenciales de salida cercanos a los 20Vpp con resistencias de carga, como se muestra la figura 2.2.4.

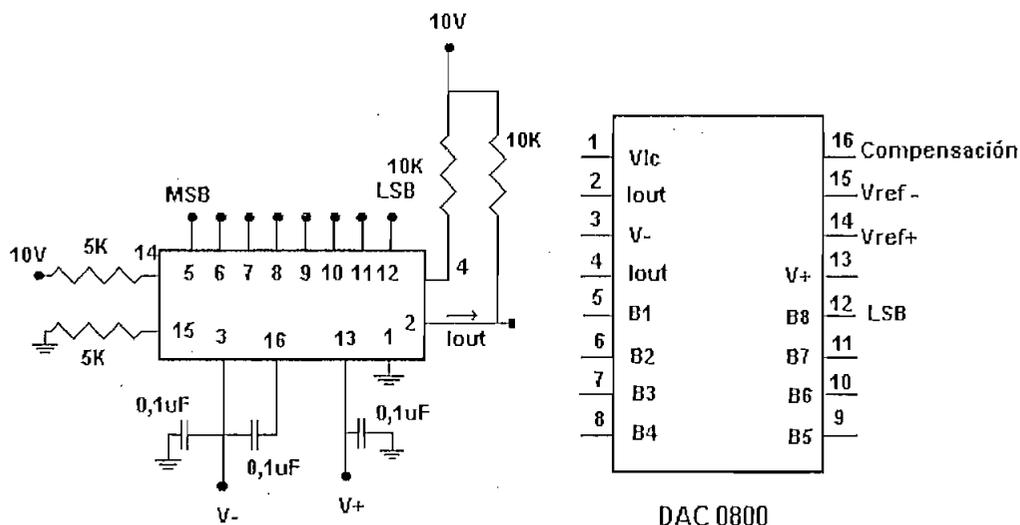


Figura 2.2.4 Conversor DAC0800, conexión típica y distribución de pines.

Las entradas del convertidor son inmunes al ruido y pueden aceptar niveles TTL, mediante la adecuada polarización del pin correspondiente al estado lógico V_{LC} , como se indica en la figura 2.2.4 El simple ajuste del potencial del pin V_{LC} posibilita el interfase directo con todas las familias lógicas. Las características del convertidor permanecen invariantes cuando el rango de la fuente de poder esta en el orden de los +/-4,5V a +/-18V.

La potencia de disipación es de solamente 33mW con una fuente de +/-5V, y es independiente de los estados lógicos escogidos.

Características generales

- Alto tiempo de conmutación de corriente 100ns
- error a plena escala +/-1 LSB
- No linealidad sobre temperatura +/-0,1%
- corriente a plena escala +/-10 ppm/°C

- salidas complementadas
- Interfase directo con TTL, CMOS, PMOS y otras
- Bajo consumo de potencia 33mW a +/-5V
- Bajo costo.

2.2.3.1 POLARIZACION DEL DAC 0801 PARA NIVELES TTL, DTL, CMOS, PMOS, HTL.

- TTL, DTL

Se polariza el pin V_{LC} a nivel de tierra.

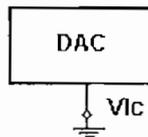


Figura 2.2.5 Polarización TTL y HTL

- 15VCMOS, HTL

Para un nivel CMOS de 12 a 15V se usa un diodo Zener con un voltaje de polarización cercano a 5V, conectando este punto entonces como referencia a la polarización del conversor.

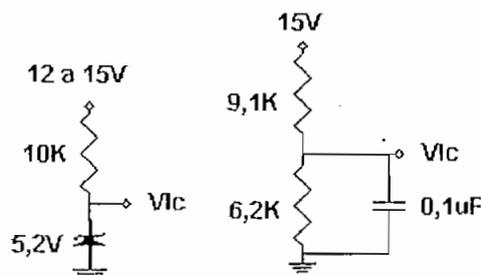


Figura 2.2.6 Polarización CMOS, HTL.

- 5VCMOS

La polarización de diodos posibilita obtener una referencia al pin V_{LC} cercanos a los 2V.

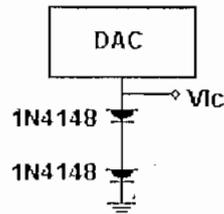


Figura 2.2.7 Polarización CMOS.

- 10VCMOS

Para este proposito es usa una red divisora de voltaje y un capacitor para eliminar cualquier ruido generado. Del punto de la red divisora se conecta al punto de referencia de polarización del conversor.

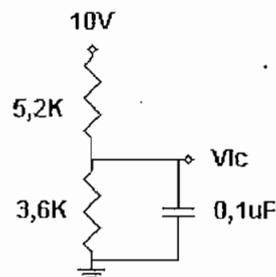


Figura 2.2.8 Polarización 5V CMOS.

- PMOS

El sistema de polarización PMOS utiliza una red en serie de diodos y resistencias. La polarización es negativa y la referencia que provee al conversor es negativa.

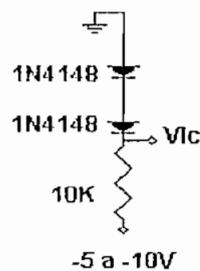


Figura 2.2.9 Polarización PMOS.

2.2.4 FILTRO DE SALIDA.

El filtro de salida se hace necesario en el sistema debido a la presencia de señales espurias propias de la conversión digital/análoga. Un filtro es una red que proporciona una modificación en la amplitud o la fase de las componentes del espectro de frecuencia de una señal, esto implica que ciertas componentes de la señal de entrada, aquellas situadas en la banda de paso serán transmitidas a través del filtro y el resto serán atenuadas o eliminadas.

Un filtro activo está compuesto por resistencias, condensadores y amplificadores operacionales y de acuerdo a su característica de respuesta de frecuencia tiene configuraciones y funciones de transferencia bien definidas.

Para el sistema el interés es obtener en lo posible una sola frecuencia de paso, que es el tono generado por el sintetizador, se elige un filtro pasa banda de segundo orden con filtro activo.

La configuración de este filtro se indica en la figura 2.2.10.

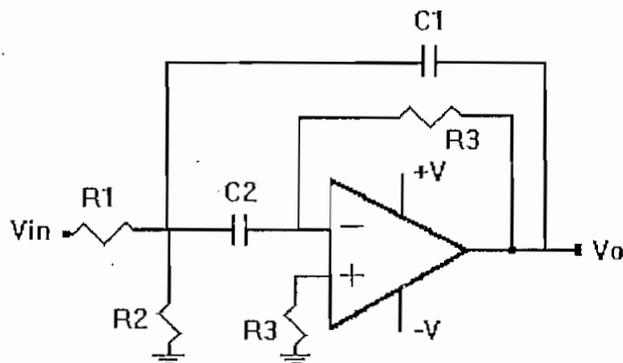


Figura 2.2.10 Filtro pasabanda de segundo orden.

y su función de transferencia se indica en la ecuación (2.1):

$$A_v(S) = \frac{A_0 \frac{W_0}{Q} S}{S^2 + \frac{W_0}{Q} S + W_0^2} \quad \text{ecu(2.1)}$$

Para encontrar los valores de cada uno de los elementos se iguala cada uno de los términos de la función anterior con los coeficientes de la función de transferencia de un circuito LC de segundo orden.

Igualando coeficientes se tiene:

$$R1.C1 = \frac{Q}{W_0.A} \quad \text{ecu(2.2)}$$

$$R3 \frac{C1.C2}{C1+C2} = \frac{Q}{W_0} \quad \text{ecu(2.3)}$$

$$R.R3.C1.C2 = \frac{1}{W_0^2} \quad \text{ecu(2.4)}$$

Para el diseño de filtros activos pasa banda se necesitan tres datos: la frecuencia central, ancho de banda (B) y ganancia. Con estos datos se calcula $Q = f_0/B$ y se reemplaza en las igualdades anteriores, se tiene 3 ecuaciones con 5 incógnitas por lo que se debe asumir el valor de 2 de ellas, siendo lo más recomendado dar un valor fijo a los capacitores. Se tiene:

Ejemplo 8:

* Diseñar un filtro activo pasabanda de frecuencia central 1MHz y factor de calidad 5.

Datos:

$f_0 = 1\text{MHz}$.

$Q = 5$.

Desarrollo:

Se asume que $C1 = C2 = 460\text{pF}$.

Al reemplazar en la ecuaciones (2.2), (2.3) y (2.4) se obtiene:

$$R1 = 12,27\text{Kohm.}$$

$$R2 = 1,2\text{Kohm.}$$

$$R3 = 5.2\text{Kohm.}$$

Para un ajuste de sintonía se coloca en lugar de R3 una resistencia fija más un potenciómetro.

2.3 Diseño y Construcción de Circuitos

2.3.1 El reloj base de tiempo.

Descripción de los osciladores TTL usados.

2.3.2 El puerto Serial del Computador.

2.3.3 El puerto paralelo del Computador.

Estudio descriptivo del puerto paralelo.

2.3.4 Grabación en memorias externas.

Descripción de los procesos de control para grabar los datos en memoria.

2.3 DISEÑO Y CONSTRUCCION DE LOS CIRCUITOS DE CONTROL E INTERFACE DIGITAL.

2.3.1 EL RELOJ BASE DE TIEMPO.

Los osciladores utilizados en el sintetizador digital directo son de la compañía ECS Inc. Son de altas características de generación, oscilan a niveles TTL, mediante la polarización de +5VDC. En la figura 2.3.1 se muestra la forma de onda a niveles TTL que presentan estos osciladores.

En la figura 2.3.1. se indica la distribución de pines y forma de conexión.

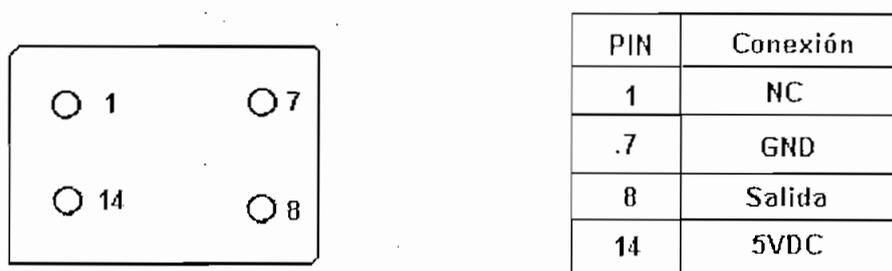


Figura 2.3.1 Distribución de pines para los osciladores de referencia.

Los osciladores a niveles TTL pueden conseguirse de las siguientes frecuencias:

Frecuencia(MHz) Frecuencia de Resolución

1,0000	15,2587 Hz
1,2288	19,6533 Hz
1,8432	28,1250 Hz
2,0000	30,5176 Hz
2,4576	37,5000 Hz
3,6864	56,2500 Hz
4,0000	61,0351 Hz
4,9152	75,0000 Hz
5,0000	76,2939 Hz
5,0688	77,3437 Hz
6,0000	91,5527 Hz
6,1440	93,2922 Hz

El reloj base de tiempo es el elemento circuital más importante en el diseño del sintetizador digital directo, puesto de que el depende la frecuencia de salida que se pueda obtener. Para la construcción del reloj base de tiempo se toman las consideraciones generales de aislar los circuitos de reloj del resto de tarjetas implementadas.

2.3.2 EL PUERTO SERIAL DEL COMPUTADOR.

Debido a que el sintetizador hace uso de opciones automáticas de selección de frecuencia, selección del reloj de oscilación, grabado en memoria, el estudio del puerto serial del computador pierde importancia dando origen a un estudio profundo del circuito e interfase construido en el equipo.

2.3.3 EL PUERTO PARALELO DEL COMPUTADOR.

Todas las computadoras personales IBM compatibles tienen un puerto paralelo generalmente se encuentra en la parte posterior del computador. En un principio el puerto fue diseñado para proveer un interfase a impresora, el cual todavía es de uso común.

Circuitos de prueba, son programados por el pórtilo paralelo, voltímetros, osciloscopios y analizadores lógicos. Para computadores portátiles o cualquier computador las cuales no tienen un slot libre o un manejar de drive, se usa el puerto paralelo para añadir aparatos estándar como respaldos en cinta magnética, adaptadores de red y sistemas de sonido.

Otros productos incluyen herramientas para diagnóstico, de autos y controladores. Debido a que el puerto paralelo se encuentra en la parte posterior del computador, no se necesita hacer ninguna adaptación al computador y fácilmente se conecta cualquier equipo al mismo.

2.3.3.1 DESCRIPCION DEL PUERTO PARALELO.

El sistema operativo denomina al p rtico paralelo como LPT1 (l nea de impresora 1), o LPT2 y LPT3 para p rticos adicionales. Otra denominaci n para el p rtico paralelo es p rtico de impresi n, reflejando de esta manera su aplicaci n m s com n. El conector posterior del p rtico en el computador es un conector tipo hembra de 25 D pines (no debe ser confundido con el conector tipo macho 25D que es usado para el p rtico serial).

Cada p rtico paralelo tiene una de tres posibles direcciones base en memoria: 3BCh, 378h o 278h. Cuando una computadora se energiza, una rutina tipo del sistema operativo busca de un p rtico paralelo en cada una de las tres direcciones respectivas. El sistema operativo determina si existe o no p rtico paralelo por la escritura y luego por la devoluci n de esa escritura, verificando entonces lo que se escribi .

En el cuadro 2.3.1 se indica la distribuci n de pines del puerto paralelo.

Conector Tipo D Pin	Se�al	Funci�n	In/Out	Registro	Registro Invertido	N�mero de bit en el registro
1	-STB	Habilitar D0 a D7	I/O	control	s�	0
2	D0	Dato bit 0	O	datos	no	0
3	D1	Dato bit 1	O	datos	no	1
4	D2	Dato bit 2	O	datos	no	2
5	D3	Dato bit 3	O	datos	no	3
6	D4	Dato bit 4	O	datos	no	4
7	D5	Dato bit 5	O	datos	no	5
8	D6	Dato bit 6	O	datos	no	6
9	D7	Dato bit 7	O	datos	no	7
10	-ACK	Reconocimiento	I	estado	no	6
11	BSY	Impresora Ocupada	I	estado	s�	7
12	PE	Fin de papel	I	estado	no	5
13	SEL	On line	I	estado	no	4
14	-AUTO	Alimentaci�n autom�tica	I/O	control	s�	1
15	-ERR	Error	I	estado	no	3
16	-INIT	Inicializar impresora	I/O	control	no	2
17	-SELIN	Seleccionar impresora	I/O	control	s�	3
18-25	GND	Tierra	I		no	-

Cuadro 2.3.1 Distribuci n de pines en el puerto paralelo.

2.3.3.2 ENTRADAS Y SALIDAS.

Del cuadro 2.3.1 se puede observar que el puerto paralelo se constituye de líneas de entrada y salida de datos. Las cuales son agrupadas en 3 registros, estos son: registro de control, registro de datos y registro de estado. A cada registro se accede mediante la asignación de la dirección a memoria; generalmente las direcciones más comunes son las siguientes:

Registro	Dirección
datos	378h
estado	379h
control	37Ah

Cuadro 2.3.2 Direcciones a registros del puerto paralelo.

a) REGISTRO DE DATOS

El registro de datos esta compuesto por 8 líneas de datos D0 a D7, son ocho salidas que llevan los datos a la impresora para ser impresos. Para otras aplicaciones, se puede usar la línea de datos como salidas de propósito general. Para controlar los estados de los pines 2 a 9 sobre el conector del pórtico paralelo, se escribe el dato seleccionado al registro de datos.. Por ejemplo para poner 1 lógico (estado lógico alto) en los pines D4 a D7 y 0 lógico en los pines D0 a D3 se escribe el dato F0h al registro de datos. En VisualBasic, se usa la sentencia OUT, como sigue:

OUT Dirección del registro de datos, Dato

, por ejemplo usando la dirección del registro de datos 378h y el dato F0h,

OUT 3BCh,F0h

Algunos pÓrticos paralelos tienen líneas de datos bidireccionales, las cuales pueden ser usadas como entradas o bien como salidas.

b) REGISTRO DE ESTADO

Las líneas de estado son cinco entradas, las cuales pueden leer datos digitales en los pines respectivos. La dirección del registro de estado es la dirección del registro de datos aumentado una. Por ejemplo si el registro de datos tiene la dirección en memoria de 378h, la dirección del registro de estado será de 379h

El registro de estado es solamente para lectura, la escritura en el mismo no tiene efecto. Las cinco líneas de estado usan los bits del 3 al 7 en el registro, correspondiente a los pines 10 a 13 y 15 del conector. Los bits 0, 1 y 2 no son usados. Para leer el estado de las entradas, basta con leer el registro. En Visual Basic se usa la función:

INP (Estado de la dirección del pÓrtico)

por ejemplo:

INP (379h)

Sin embargo, el valor leído no es exactamente el que se encuentra en los pines respectivos. Los bits 3 a 6 se leen normalmente, los bits en el registro de estado capturan el correspondiente nivel lógico de sus pines respectivos. El bit 7, sin embargo, contiene el complemento del estado lógico del pin 11 (Ocupado). Para encontrar el verdadero nivel lógico al conector, se debe complementar o invertir el bit 7.

La función booleana OR- exclusivo(XOR), opera muy fácilmente para invertir uno o más bits dentro de un byte, dejando a los demás bits sin sufrir cambio alguno. Como lo muestra el cuadro 2.3.2, el resultado de la operación XOR es solamente 1 cuando la entrada consiste de 1 y 0.

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

Cuadro 2.3.2 Tabla de verdad operando XOR.

Para invertir el bit seleccionado dentro de un byte, primero se crea una máscara al byte, en el cual los bits que se desean invertir son 1 y los bits que ignore son 0. Por ejemplo para invertir el bit 7, la máscara debería ser 10000000, o 80h. Si se desea usar la función XOR a este byte con el byte leído del registro de estado, se debe conectar al respectivo pórtilo. El bit de máscara da su nombre debido a que se encuentran ceros, ocultos, los bits que no se desea cambiar. Por ejemplo:

10101XXX Entradas al pórtilo de estado, bits 3 a 7 al conector (X = no disponibles)

00101XXX Resultado cuando se lee el registro de estado (bit 7 invertido)

10000000 Byte de máscara.

10101XXX XOR de dos bytes previos, da como resultado los niveles lógicos verdades al conector.

La instrucción en VisualBASIC que muestra los niveles lógicos verdaderos a los pines, para un pórtilo con una dirección de 3BCh es:

INP (3BDh XOR 80h)

c) REGISTRO DE CONTROL

A más de los registros de datos y de control, el p \acute{o} rtico paralelo posee un p \acute{o} rtico de control bidireccional. Se usan las cuatro como entradas o salidas, en cualquier combinaci \acute{o} n. La direcci \acute{o} n del registro de control es la direcci \acute{o} n del registro de datos sumado en 2. Por ejemplo para un registro de datos de direcci \acute{o} n 378h, el registro de control tendr \acute{i} a la direcci \acute{o} n 37Ah.

Las cuatro l \acute{i} neas de control usan los bits 0 a 3 en el registro, los cuales corresponden a los pines 1, 14, 16 y 17 sobre el conector. Los cuatro bits habilitan al p \acute{o} rtico paralelo para interrupciones de hardware. En p \acute{o} rticos con l \acute{i} neas de datos bidireccionales, bit 5 o 7 debe configurar el p \acute{o} rtico como entrada o salida. Para escribir a las l \acute{i} neas de control, se escribe al registro de control:

OUT Direcci \acute{o} n del registro de control, Dato de control

por ejemplo para escribir 0Ah al registro de control con direcci \acute{o} n de registro (37Ah), la sentencia es:

```
OUT 3BEh, 0Ah
```

Igual que el registro de estado, el p \acute{o} rtico de control tiene bits invertidos. En el conector los bits 0, 1 y 3 est \acute{a} n complementados en su nivel l \acute{o} gico en el registro de control. Solamente el bit 2 se lee normalmente. Para asegurarse de que el valor escrito es igual al que se encuentra en el conector, se implementa el valor XOR con 0Bh (00001011), como se muestra en el ejemplo que sigue:

```
OUT 3BEh,0Ah XOR 0Bh
```

2.3.3.3 VELOCIDAD DEL PUERTO PARALELO.

La velocidad a la que el p \acute{o} rtico paralelo puede leer o escribir, depende de la velocidad del computador y la velocidad del software, el cual adem \acute{a} s depende del lenguaje de programaci \acute{o} n y el programa que se escribe. Usando programaci \acute{o} n tipo ensamblador, se puede leer el puerto paralelo y grabar los valores en memoria o escribir el valor de memoria al p \acute{o} rtico, usando unas peque \acute{n} as instrucciones.

Como regla en general, es posible transferir datos a unos 40K o 80Kbytes por segundo sobre computadoras muy rápidas. En computadoras de 8MHz, se puede llegar a transmitir a 10Kbytes o 15Kbytes por segundo.

En diseños de computadoras recientes, que incluyen el puerto paralelo de Intel SL y los puertos que siguen el estándar 1284 de IEEE pueden transferir datos a 1Mbytes por segundo. Algunos adaptadores de red, manejadores y otros aparatos de puerto paralelo usan métodos más rápidos de transmisión cuando se detecta un puerto de alta velocidad.

2.3.3.4 CABLES DEL PUERTO PARALELO.

Para poder experimentar con un puerto paralelo se necesita un cable que permita fácilmente acceder a las líneas de señales. Los conectores estándar de impresión generalmente tienen un conector macho de 25 pines para conectar a la computadora y en el otro extremo cuenta con un conector macho tipo Centronics, con 11 contactos que no son usados.

Una solución es usar el cable de impresión y usar solo los alambres que tienen conexión directa al puerto. Si se usa un diferente tipo de cable o conector, para obtener mejores resultados, se tomará en cuenta la siguiente consideración:

- Un cable ideal, es aquel en el cual la señal presente a la entrada es exactamente igual a la salida.

En la realidad las señales que viajan a través de los cables se degradan y sufren problema de ruido, debido a la resistencia del conductor, cruces por cero de los conductores adyacentes, señales reflejadas, mala tierra y otros problemas. En el interfase serial RS232 se puede usar cables de hasta 100 pies de largo y no se presentan problemas de consideración.

El pÓrtico paralelo fue diseÑado para transferencia en distancias cortas. Esta es la raz3n del porque la longitud de un cable de impresi3n tiene 15 pies de longitud como mÁximo, e inclusive menos. Existen algunas formas de minimizar los problemas del enlace paralelo, por ejemplo: recubrimiento de metal para evitar las interferencias de frecuencias de radio y prevenir el acoplamiento capacitivo entre los conductores sobre lados opuesto, en el momento de producirse conexi3n.

Muchos cables de impresora cuentan con un metal (tipo papel) que envuelve a los conductores a lo largo del cable. En algunos cables de impresi3n el recubrimiento se extiende a los conectores. El recubrimiento de papel metÁlico no previene el acoplamiento magn3tico, inductivo entre los conductores en el cable, pero esto no se constituye en un gran problema.

Un m3todo simple para minimizar el acoplamiento es usar cable de par trenzado en cada conductor, logrando as3 reducir el efecto de acoplamiento.

Idealmente, en un cable de par trenzado, cada seÑal del conductor es usada con un conductor a tierra que provee el camino de retorno. Esto minimiza el Área de la corriente de lazo formada por la seÑal y los conductores de retorno, el que reduce la habilidad de las seÑales a acoplarse magn3ticamente entre conductores adyacentes.

AdemÁs, el par trenzado causa una reducci3n de ruido, ya que los picos entre uno y otro que se producen tienden a suprimirse con el mÁs pr3ximo. Desafortunadamente los conectores de puerto paralelo no tienen suficientes pins para dar a cada seÑal de las 17 seÑales una seÑal de tierra para retorno. Se necesita el contar con por lo menos 8 conectores mÁs.

Ordinariamente el cable estÁndar no tiene pares trenzados en el cable de retorno con D3 a D7 y las cuatro l3neas de estado. Esto ayuda a reducir el ruido. En el mercado se puede encontrar cable de par trenzado, el cual tiene 18 secciones de cables de par trenzado, con 2 secciones mÁs, que permiten conexi3n a los IDC (insulation-displacement connectors).

2.3.3.5 INTERFASE DE PUERTO.

El puerto paralelo usa de ordinario 5 voltios, y cables de longitud de 8 pies a 15 pies de largo. El correcto interfase para estos componentes pueden reducir los problemas de ruido o degeneración de las señales. Para obtener mejores resultados en el proceso de transmisión se sigue los siguientes recomendaciones:

- a) Para eliminar el desacoplamiento, conectar un capacitor de la línea de 5 voltios a tierra a cada integrado que tenga conexión directa con los conectores del cable. Se usa generalmente un capacitor que tenga una buena respuesta a la alta frecuencia, como por ejemplo el de cerámica, mica o politireno.

Los contactos de los capacitores a la línea de 5 voltios y tierra deben ser en lo posible pequeños. Un valor aceptable para este propósito es un valor de $0,1\mu\text{F}$, pero este valor exacto no es crítico. Además coloca un capacitor electrostático de $10\mu\text{F}$ en el lugar del tablero que corresponde a la conexión de la fuente de 5 voltios y tierra. Los capacitores de desacoplamiento guardan la energía necesitada por las compuertas lógicas para que ellas conmuten. Todas las compuertas lógicas necesitan de corriente para responder a los cambios a sus entradas.

Cuando la corriente esta disponible en un capacitor, la compuerta puede conmutar fácilmente, sin causar picos en la línea de la fuente de poder o las líneas de tierra. El capacitor debe ser localizado en lo posible cerca del circuito integrado para así minimizar la inductancia del lazo formado por las conexiones entre el capacitor y el circuito integrado. Una baja inductancia permite una operación más rápida.

Los manejadores o receptores que se puede conectar al puerto paralelo afectan la calidad de la señal. Para aumentar la inmunidad al ruido de las entradas de datos de las líneas de datos del puerto use equipos con entradas Schmitt-trigger, como el 74LS14. Para una compuerta lógica ordinaria, si el cambio a la entrada es lento, la conmutación a la salida podría producir un jitter, pudiendo entonces dificultarse la decisión si la señal era alto o bajo.

El Schmitt Triggers ayuda a resolver este problema, debido a que se cuenta con dos conmutadores umbral, uno para la condición de bajo a alto y otro para la condición de alto a bajo. Por ejemplo la salida de un inversor 74LS14 no va a bajo mientras no llegue al límite de la entrada en 1,6V. Después la salida conmuta a bajo, y al ir a alto esto no lo realiza hasta que se llegue a un voltaje de 0,8V.

Para usar las salidas del puerto paralelo, líneas de estado, datos y control se debe usar buffers o líneas de drives con habilitación de sus correspondientes salidas. Por ejemplo se puede usar el 74LS240, como salidas que pueden típicamente dar 24mA a sus salidas y pueden manejar entradas de hasta 12mA, comparadas con las 8 y 0,4mA de las salidas ordinarias LSTTL.

- b) No conectar señales de reloj o de control sin haberlas pasado por un buffer, incluyendo las entradas y salidas de los flip-flops, contadores y registros, directamente al cable. Estos aparatos son sensitivos a las señales reflejadas. Pero si se usa buffers se logra aislar estas señales.

Algunos chips como por ejemplo el 73LS374 tienen salidas buffers. Si no se encuentra en uso la línea de estado, asegura a 5 voltios o tierra una resistencia de 4700 ohmios para asegurarlo a un nivel lógico válido. Otro método importante para obtener buena calidad de las señales es tomar en cuenta los valores que varían en el conductor, como son el diámetro, aislamiento y distancia entre uno y otro conductor en el cable. Los productores de cable dan ciertas características de sus conductores, por ejemplo un valor típico para cables de par trenzado es de 100 ohmios.

Cuando la carga y el conductor están conectados toda la energía es transferida a la carga. Si no existe conexión perfecta alguna energía se refleja de regreso a la fuente, y la energía que se refleja puede causar que se reduzca la vida útil del equipo, además como causa que los niveles lógicos de lectura no sean los correctos en el receptor.

El diseño de los terminadores para el puerto paralelo es especialmente dificultoso porque cada circuito de la computadora, varía con el tipo de máquina

con el que se cuenta. Usando la terminación incorrecta se reduce la velocidad de transmisión o también existe un excesivo consumo de potencia.

- c) Para los drives, una serie de resistencias proveen acoplamiento de impedancias. El valor de las resistencias es de valor igual a la impedancia característica de la línea, menos la resistencia de la entrada. Algunas tarjetas de puerto paralelo incluyen una resistencia de 30 ohmios en serie con cada dato a la salida. En el receptor, una red de resistencia y capacitor se encuentra al finalizar la línea.

El valor de la resistencia es igual a la impedancia característica del conductor y el capacitor provee una baja impedancia durante la conmutación.

2.3.4 GRABACION EN MEMORIAS EXTERNAS, PARTE DEL SINTETIZADOR DIGITAL DIRECTO.

Los datos de la forma de onda que se encuentran almacenados en el computador en un archivo tipo ASCII, son mandados secuencialmente por el puerto paralelo a escribirse en una memoria RAM externa. El posibilitar que este proceso se cumpla correctamente involucra la utilización de señales de control y datos separadamente. Por esta razón se ha dividido el estudio de la grabación de datos en memoria externa en bloques funcionales de datos y control.

Del puerto paralelo se utilizan el pórtico de datos y el pórtico de control. Son 13 líneas, divididas de la siguiente manera:

pórtico de datos--> 8 líneas de datos --> D1, D2, D3, D4, D5, D6, D7, D8.

 pines--> 2 3 4 5 6 7 8 9.

pórtico de control --> 4 líneas de control --> A, B, C, D.

 pines --> 2 14 16 17.

señal de tierra --> 1 línea --> GND.
pin --> 18.

En la figura 2.3.2 se muestra la distribución de pines utilizados del puerto paralelo.

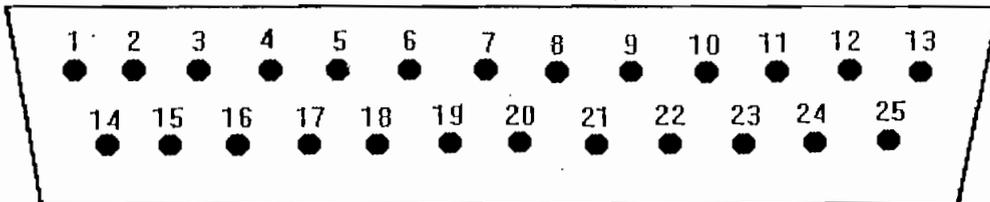


Figura 2.3.2 Distribución pines, puerto paralelo.

Para el presente trabajo se ha utilizado buffers 74LS244 y 74LS241. Estos circuitos además tienen una señal de control que inhabilita o habilita sus salidas. En la práctica se han conectado ambos controles a tierra, dejando en operación total. El circuito implementado se muestra en la figura 2.3.3.

De esta manera se cuenta con 2 bloques de señales (datos y control). Debido a que el número de ordenes de control necesarias excede el valor de 4, se amplía el número de señales de control mediante un decodificador de 4 a 16.

2.3.4.1 GRABADO DE DATOS EN MEMORIA RAM EXTERNA.

La memoria que usa la síntesis directa es de 8Kbytes de capacidad, cuenta con 13 líneas de direccionamiento y con palabras para grabar de un byte. Debido a que el bus de datos del pòrtico solo cuenta con palabra de 8 bits, se ha segmentado la palabra de dirección en 2 palabras de 8 bits, generando una palabra de 16 bits.

Para lograr este propósito se hace uso de los circuitos integrados U8 y U9 (74LS274) que se muestran en la figura 2.3.5.

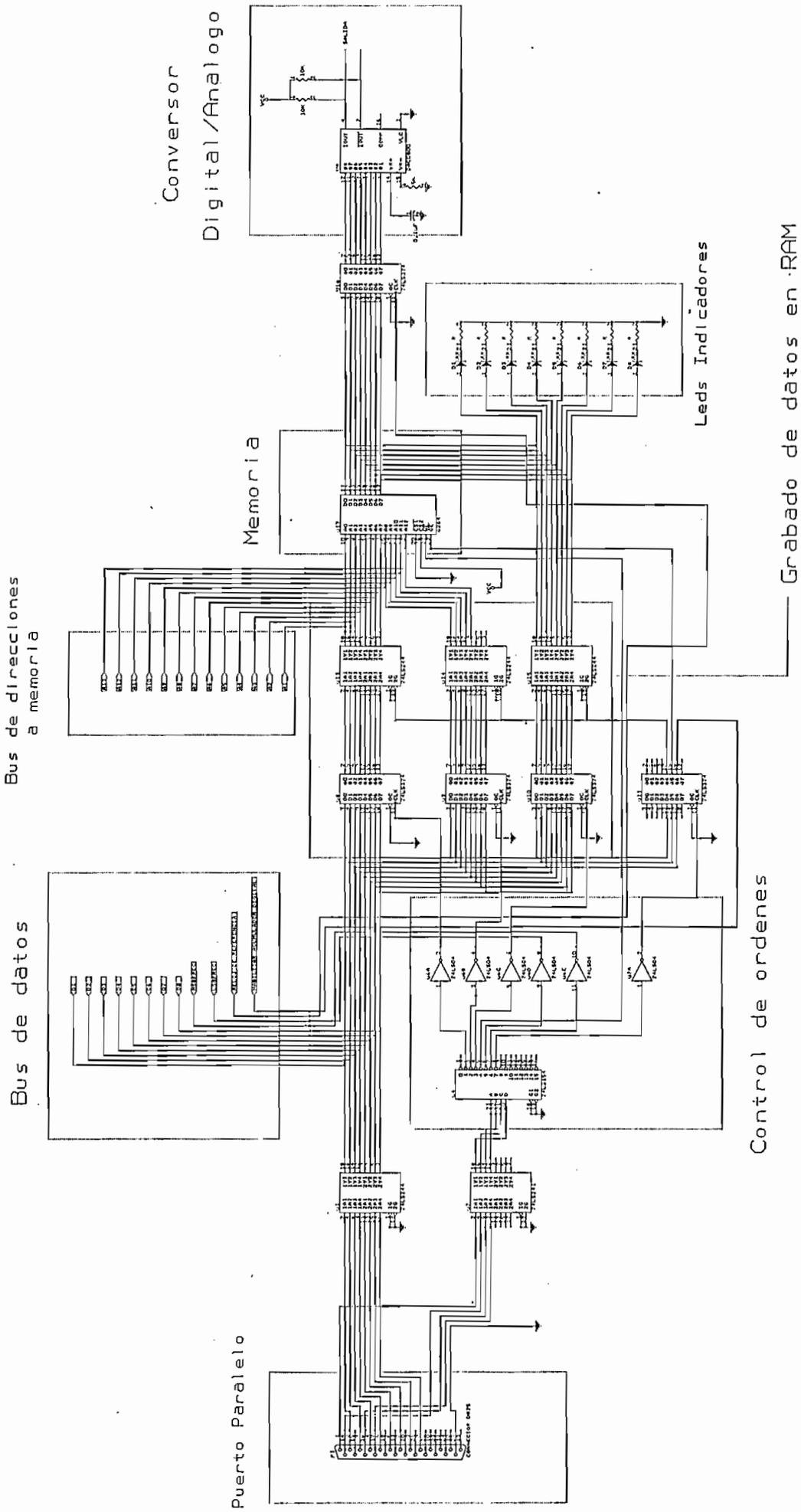


Figura 2.3.3

Los circuitos integrados U8 y U9 tienen la capacidad de ser controlados por solo una señal, la cual hace que los datos que se encuentran a la entrada pasen a la salida y sean retenidos.

Para este fin se usa una palabra de control desde el puerto paralelo que indica a cual circuito se habilita el paso del dato que se encuentra en el bus de datos respectivo. Este proceso es repetido cuantas veces sean necesarios para direccionar localidades diferentes.

Para grabar el dato digital de 8 bits en la memoria se utiliza el circuito integrado integrado U10 (74LS374) de similares características que U8 y U9. Se energiza de nuevo el bus de datos del puerto con el dato a grabar y mediante los bits de control se manda la orden de pasar los datos del bus al latch respectivo de grabado en memoria. Se localiza en memoria el dato a grabar, luego de lo cual del p rtico de control manda un pulso de escritura a memoria. Mediante este procedimiento se ha guardado el dato en memoria.

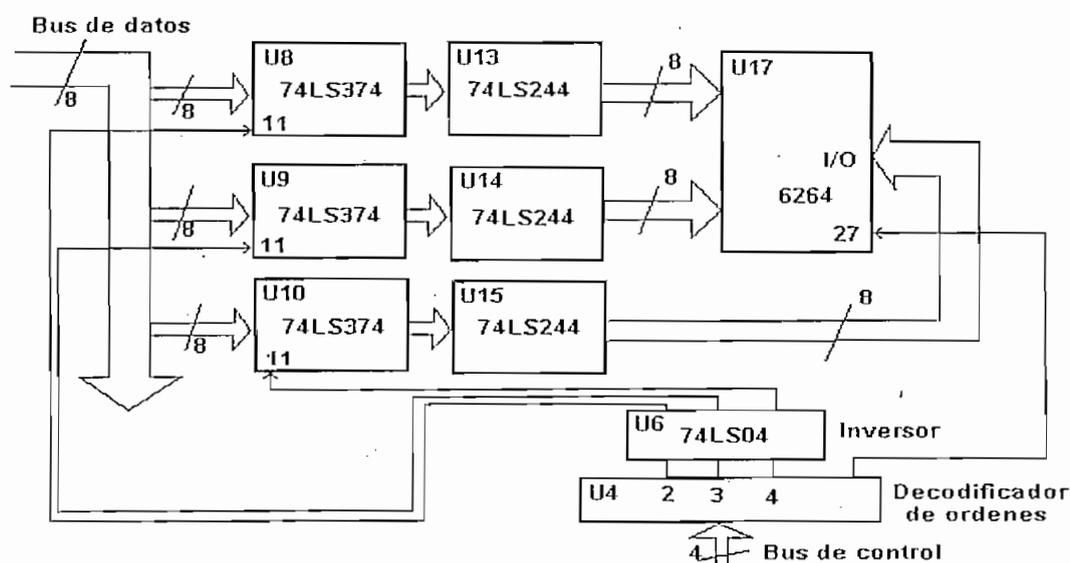


Figura 2.3.4 Circuito de grabado de datos en RAM.

2.3.4.2 SELECCION DE FRECUENCIA DE GENERACION.

Para el control de la frecuencia de generaci n al acumulador digital necesita de una palabra de control de dos bytes. De la misma manera como se

resolvió el problema para el direccionamiento de la localidad escogida en memoria, se procede en este caso. A la palabra de control de frecuencia se la segmenta en dos palabras de 8 bits respectivamente (MSBFREC y LSBFREC).

El bus de datos se energiza primero con el valor digital de LSBFREC a enviar, luego de lo cual el pórtilo de control se encargará de mandar el dato requerido para que el dato digital pase al retenedor U1 (tarjeta del acumulador) y mantenga el dato de selección escogido. Luego por el bus de datos manda el valor digital correspondiente al MSBFREC y se procede a mandar la señal de paso de datos al retenedor U2 (tarjeta del acumulador).

Realizado este procedimiento el dato digital de 16 bits ya se encuentra posicionado en el acumulador y la frecuencia de generación se ha escogido.

2.3.4.3 CONTROL DE INTERFASE.

Debido a que los procesos implementados en el sintetizador son dos: grabado de datos y generador de señales, debe existir una palabra de 3 bits que controle que cuando se graben los datos en memoria la operación del acumulador se vea deshabilitada, y una vez que se han grabado los datos en memoria la misma palabra defina que el acumulador entre en operación y que el circuito de grabado se deshabilite.

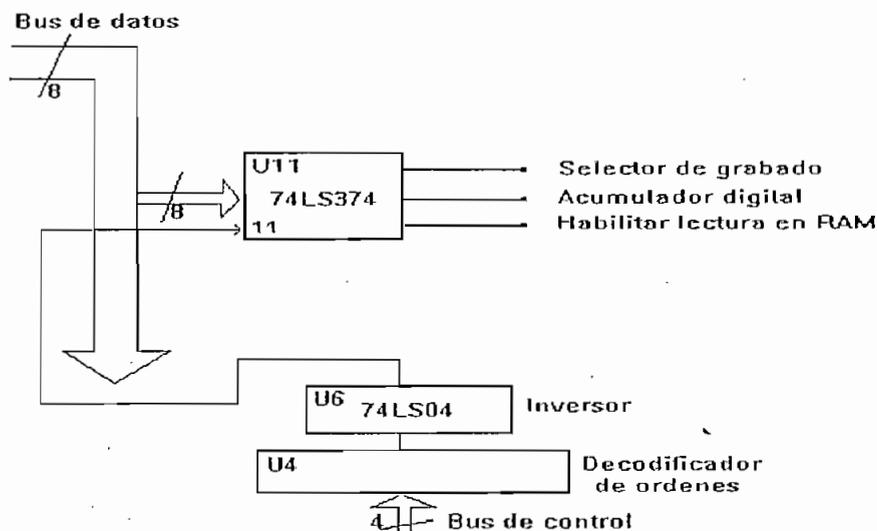


Figura 2.3.5 Selector de operaciones.

El circuito implementado se muestra en la figura 2.3.5.

Se han descrito todas las operaciones electrónicas involucradas en el proceso de control y procesamiento de información. En el capítulo 3 se detallará su funcionamiento integral, debido a que todo este procedimiento necesita de órdenes (software).

2.4. Diseño de circuitos de alimentación

2.4.1 Fuente D.C.

Fuente D.C. utilizada.

2.4.2 Alimentación momentánea a memoria RAM.

Circuito de alimentación a memoria cuando falle la alimentación de energía eléctrica.

2.4. DISEÑO Y CONSTRUCCION DE LOS CIRCUITOS DE ALIMENTACION.

2.4.1 FUENTE D.C.

Para la polarización del conversor digital - analógico utilizado, se necesita de una fuente que provea salida de voltaje +/- 18V, 1A. Además debe poseer la capacidad de variar este voltaje en un rango de -/+ 10V. Esto se debe a posibles variaciones de la amplitud de la señal en el conversor.

El circuito implementado se muestra en el gráfico 2.4.1

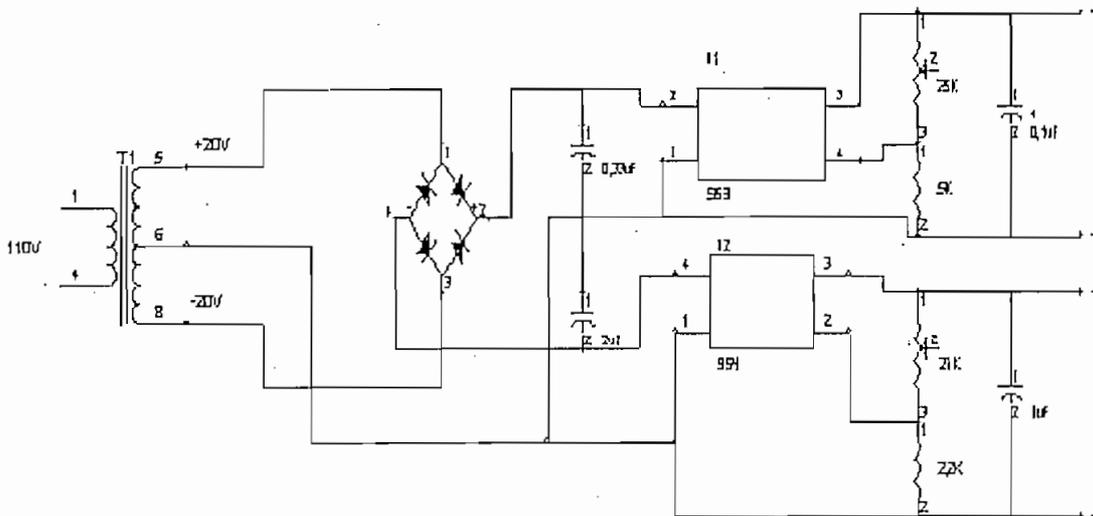


Figura 2.4.1 Fuente D.C. variable.

2.4.2 ALIMENTACION MOMENTANEA A MEMORIA RAM.

Se implementa la alimentación a memoria debido a que los datos binarios guardados se pueden perder si es que se produce la desconexión del equipo,, con

lo que el equipo pierde versatilidad. El propósito es que mediante baterías instaladas en el sintetizador se guarden los datos aunque se haya perdido la energía principal.

El circuito utilizado, además de su sencillez en la construcción presenta muy buenas características. En la figura 2.4.2 se muestra el circuito de conmutación implementado.

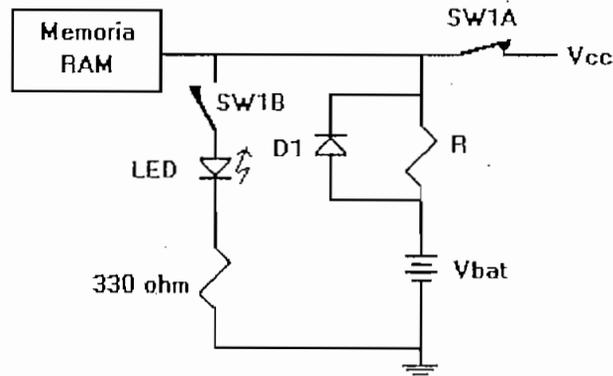


Figura 2.4.1 Circuito de batería a memoria RAM.

SW1A y SW1B funcionan en condiciones de abierto y cerrado. Mientras SW1A este cerrado la fuente Vcc alimenta a memoria y a su vez carga la batería siguiendo el camino de la resistencia; a su vez SW1B esta abierto. Cuando entra en funcionamiento el circuito de batería SW1B se cierra para energizar el led que indica funcionamiento, en cambio SW1A se abre para desabilitar la energía de la fuente principal.

Las baterías utilizadas son recargables (Niquel - Cadmio), el fabricante recomienda la carga a una corriente de 70mA durante 14 horas. Las pilas a plena carga pueden alimentar un circuito digital que consuma 700mA durante una hora.

El valor de la resistencia se la calcula a partir de la ecuación(2.5)

$$R = \frac{V_{cc} - V_{bat}}{0,1 * I_n} \quad \text{ecu(2.5)}$$

donde: V_{cc} = voltaje de la fuente.

V_{bat} = voltaje de la batería.

I_n = corriente nominal de la batería.

Para nuestro caso se tiene:

Datos:

$V_{cc} = 5V$

$V_{bat} = 4,3V$

$I_n = 700mA$

aplicando la ecuación (2.5) se obtiene:

$$R = \frac{5 - 4,3V}{0,1 * 0,7} = 10 \text{ ohm}$$

La resistencia es de 10 ohmios; se recomienda usar una resistencia de $\frac{1}{2}$ watio.

CAPITULO III

Programa del Sintetizador

3.1. Descripción general de los programas.

Diagramas de flujo.

3.2 Ingreso y Procesamiento de datos.

Formato de archivos y datos.

3.3 Control y Presentación de Datos..

El puerto paralelo como control del DDS.

3.4 Manejo Visual de Formas de Onda.

Descripción del programa.

3.1 DESCRIPCIÓN GENERAL DE LOS PROGRAMAS A REALIZARSE.

De acuerdo al esquema planteado el desarrollo del programa de control e ingreso de datos a memoria bajo un formato visual, se usa Visual Basic 3.0 como el entorno de trabajo de la aplicación. El programa tiene dos campo de acción definidos de acuerdo a la tecnología utilizada, estos son:

- a) Presentación visual y manejo de información en el ambiente Windows 3.1.
- b) Control del sintetizador mediante el interface al pòrtico paralelo.

3.1.1 PRESENTACION VISUAL.

Manejar recursos informáticas asociados a una circuiteria electrónica implica la existencia de controles visuales accesibles al usuario. El programa que se realizó cuenta con las siguientes funciones visuales, los cuales se definen de acuerdo al orden lógico secuencial de utilización:

- a) Presentación de una pantalla principal, en la que se encuentran definidos todos los recursos con el que cuenta el sistema.
- b) Mostrar el gráfico de los datos de la forma de onda a grabar, el cual previamente se ha seleccionado de un menú tipo administrador de archivos. Además se presenta una señal visual que indica que porcentaje del archivo esta siendo leído.
- c) Indicar el gráfico de la señal a generar y escribir los datos simultáneamente a memoria externa. También emplear un indicador visual del porcentaje de información grabada al circuito del sintetizador digital.

- d) Dirigir la operación circuital del sintetizador, habilitando y deshabilitando funciones de grabado y generación de señal.
- e) Seleccionar la frecuencia de resolución desde el computador utilizando solo el mouse.
- f) Ayudas generales acerca del funcionamiento del programa.

Las funciones mencionadas tienen ejecución mediante botones de acceso rápido¹⁰ y menús de selección, todo esto referido al modo de trabajo en Windows.

a) Ejecución de pantalla principal:

Cuando se ejecuta el programa se muestra en pantalla la figura 3.1.1.

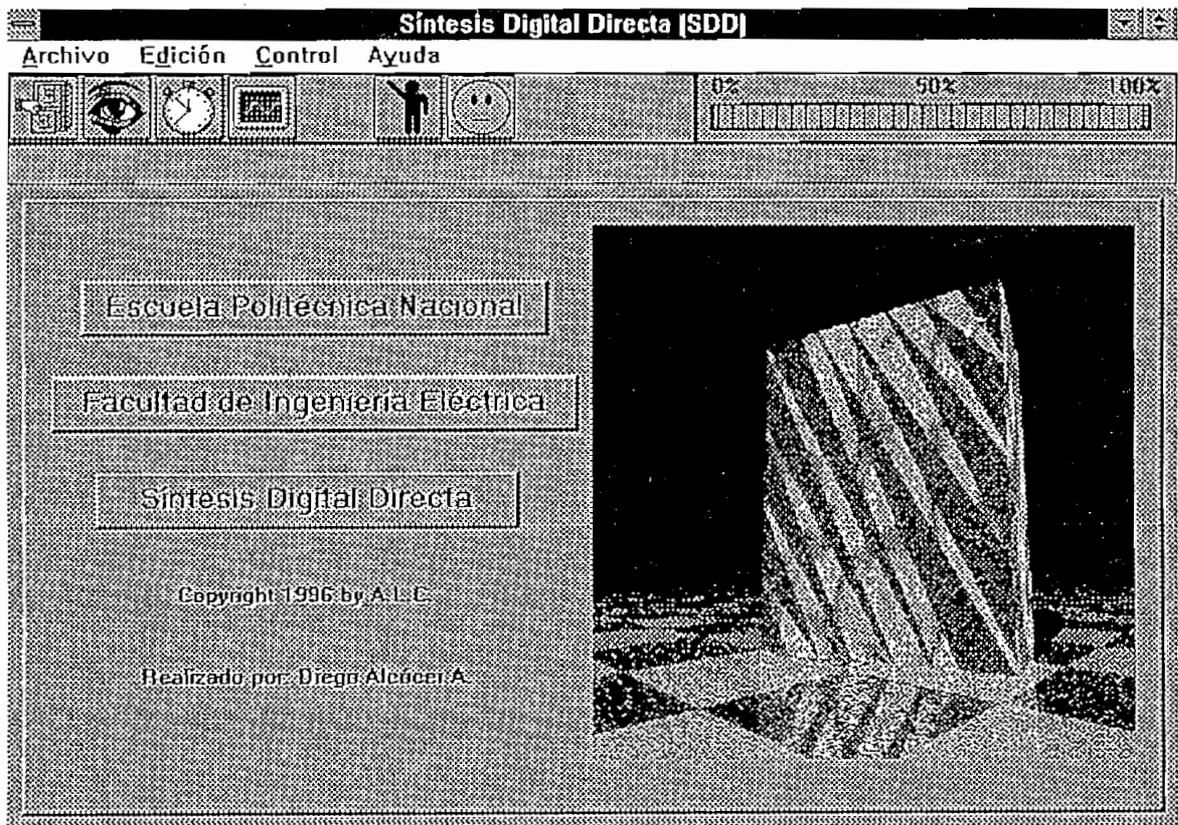


Figura 3.1.1 Menu principal de subrutinas.

¹⁰ Permiten acceder a funciones de forma rápida y sencilla. Son parte del trabajo en el entorno Windows.

En Visual Basic los procesos de órdenes se ejecutan a partir de eventos o acontecimientos, es decir el momento que exista una condición establecida se ejecutan subrutinas de proceso, como por ejemplo mostrar la pantalla de ayuda. Se definen las subrutinas existentes necesarias para controlar el sintetizador.

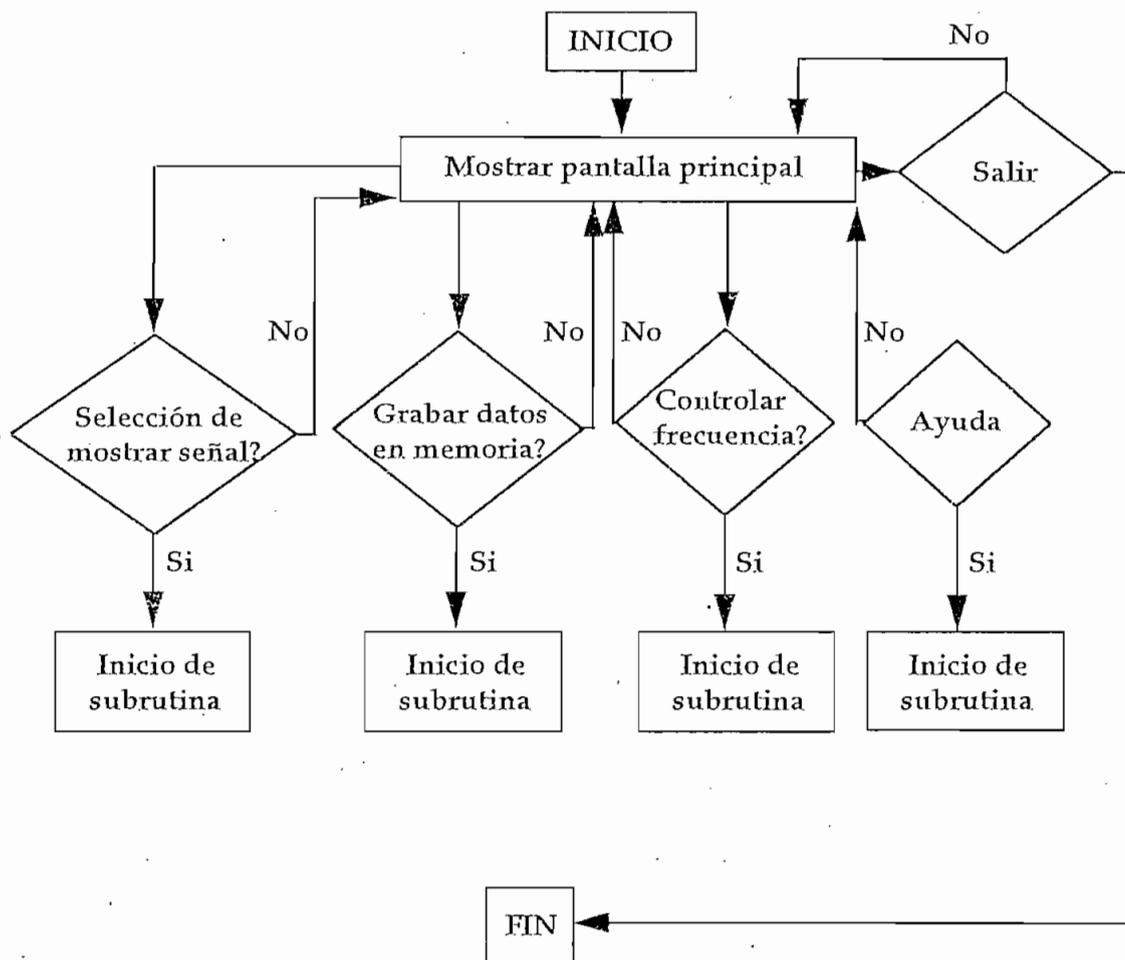


Figura 3.1.2 Diagrama de flujo de eventos.

En un inicio se muestra la pantalla principal, tal como se indica en el diagrama de flujo de la figura 3.1.2., en la cual se presentan varias subrutinas definidas por las opciones que requiere el sintetizador.

Todas las subrutinas se encuentran al mismo tiempo habilitadas, de tal forma que el usuario pueda acceder a las misma usando los menús de selección.

El proceso de programar en Visual Basic se basa en el diagrama de la figura 3.1.2. Además de obedecer a eventos del mouse, también obedece a eventos

de teclado, existen teclas de manejo en caso que no se disponga de un mouse.

b) Mostrar gráfico de forma de onda a grabar:

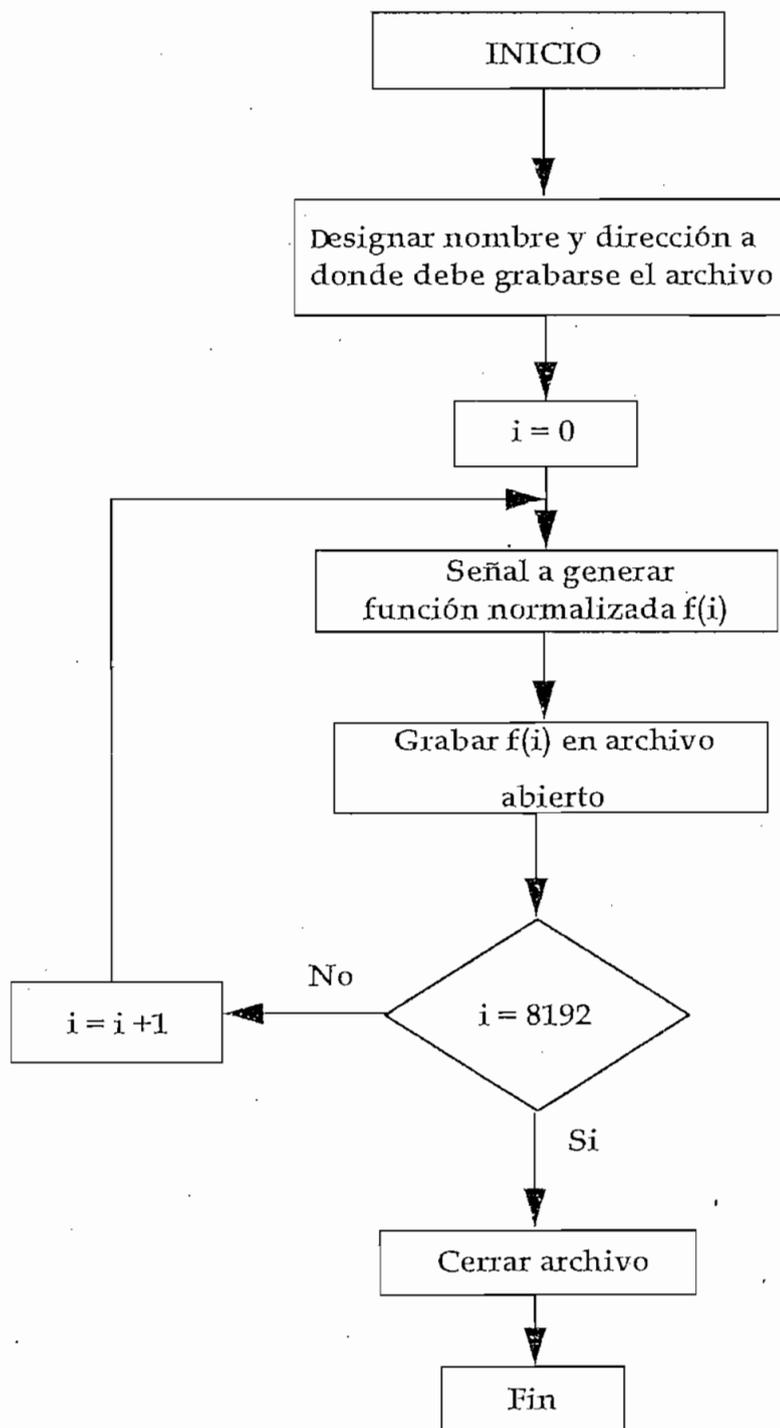


Diagrama 3.1.1 Diagrama de flujo para archivos a generar.

El programa implementado realiza la lectura de archivos grabados previamente, no los genera. Para crear estos archivos se recomienda hacer uso de programas de menor capacidad que Visual Basic, entre los cuales se puede mencionar: Qbasic 4.5, Pascal, etc.

Los archivos disponibles para la aplicación fueron desarrollados en Qbasic 4.5; de acuerdo al diagrama de flujo que se presenta en el diagrama 3.1.1.

Para diferenciarlos de los múltiples archivos existentes en un computador se ha sugerido grabar estos archivos con la extensión **DDS**, de tal forma que para una función sinusoidal se la puede encontrar con el nombre:

sin.sdd

Se puede grabar al archivo creado con un nombre de máximo ocho letras y 3 letras de extensión. No se permiten caracteres gráficos.

3.2 PROGRAMA DEL MICROCONTROLADOR PARA EL INGRESO Y PROCESAMIENTO DE DATOS.

Debido a que el control y grabado de datos se lo realiza con el puerto paralelo, el programa para el ingreso y procesamiento de datos se referirá al mismo. No se hará referencia al tratamiento de información usando el microcontrolador 8751.

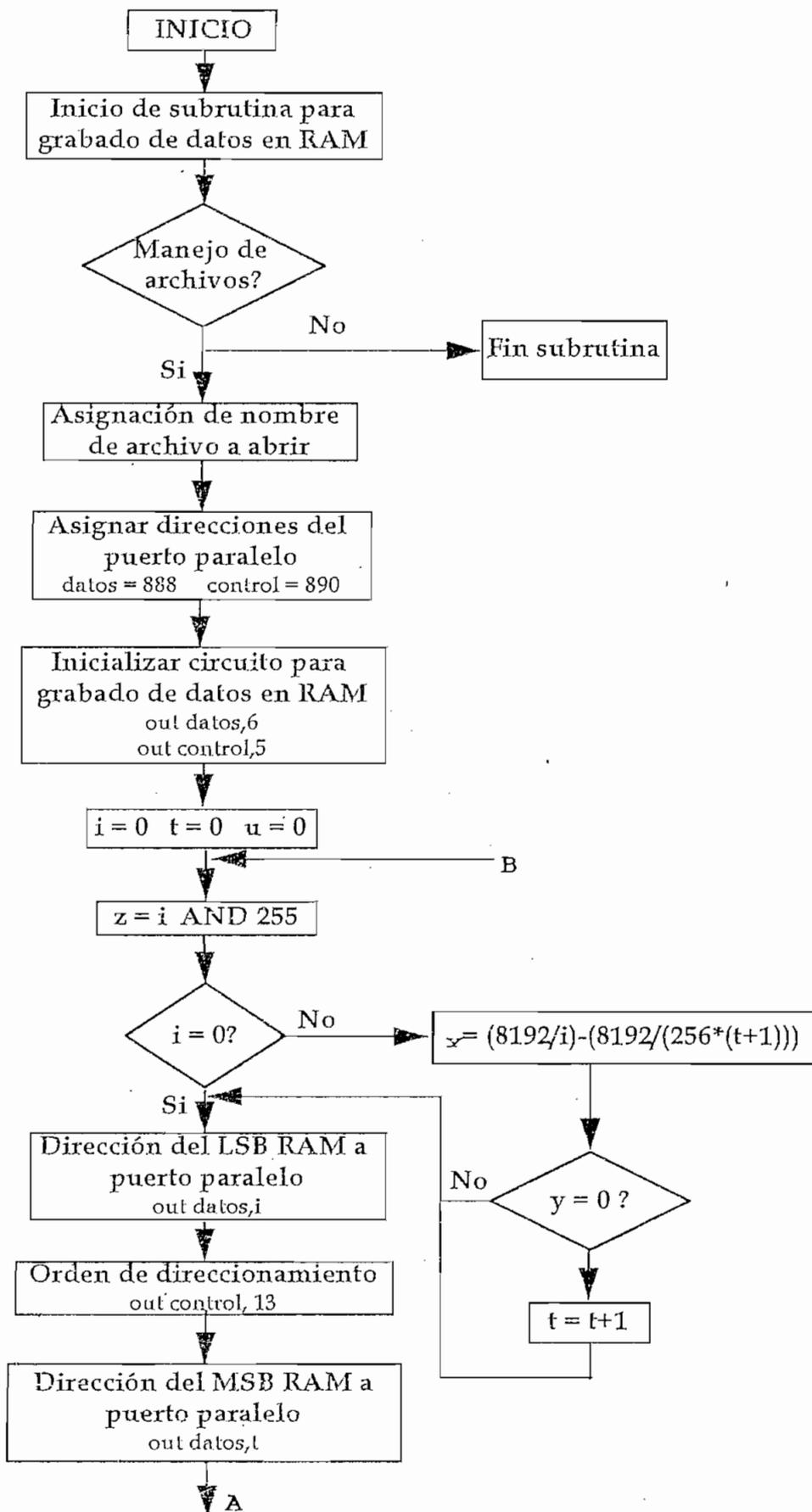
c) Indicar en pantalla datos leídos de archivo SDD y escritura en memoria RAM.

Tomando en cuenta la construcción del circuito de grabado de datos, el cual se mostró en el capítulo 2, sección 2.3.4.1; se define el diagrama de flujo de programa de acuerdo al diagrama 3.2.1.

El manejo de archivos se realiza mediante la llamada al administrador de archivos de Visual Basic, mediante la sentencia:

Abrir.Filter = "archivos SDD (*.sdd)|*.sdd|"

ejecutando este comando se muestra en pantalla la figura 3.2.1.



Continúa siguiente hoja.

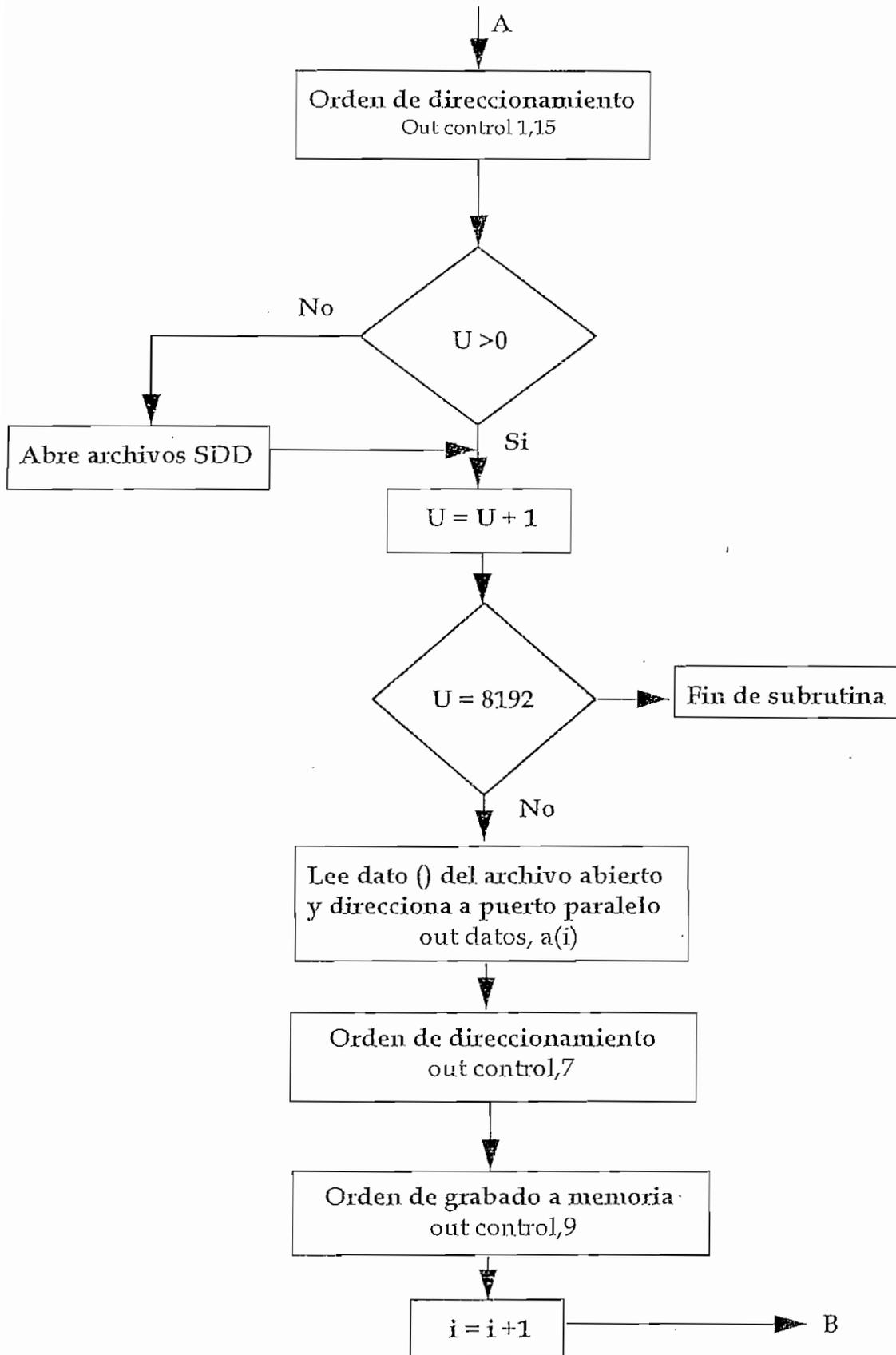


Diagrama 3.2.1 Subrutina de grabado de datos en memoria RAM

d) Dirigir la operación circuital del sintetizador.

El control de información se define a partir de las funciones necesarias que el computador ejecuta para que el sintetizador digital pueda realizar todas sus operaciones. Debido a que en el capítulo 2, sección 2.3.4 se definieron todas las funciones de control que se ejecutarán, es necesario definir cuales serán las palabras que se enviarán por el pórtico de control para que el sintetizador puede ejecutar esas ordenes. Para eso se realiza un pequeño estudio del pórtico de control del puerto paralelo.

Conector D Pin	Registro	Bit en el registro	Bit invertido
1	Control	0	Si
14	Control	1	Si
16	Control	3	No
17	Control	2	Si

Cuadro 3.2.1 Pórtico de control del puerto paralelo.

Tal como se muestra en el cuadro 3.2.1, las señales en los pines 1, 14 y 17 son invertidas, es decir si se escribe la sentencia:

OUT 380h,Fh

los verdaderos valores que se obtienen en los pines de salida son los datos que se muestran en el cuadro 3.2.2.

Pin	1	14	16	17
Dato a la salida	0	0	1	0

Cuadro 3.2.2 Valores a la salida del pórtico de control.

Debido a este limitante y para no usar ningún otro tipo de circuito electrónico tal como se recomienda en el capítulo 2.3.4, se encuentra todas las 16 combinaciones reales a la salida del pórtico para los 16 datos que se pueden mandar, es decir se enviarán los datos al pórtico, se realizara la lectura a la salida y se obtienen los resultados que se muestran en el cuadro 3.2.3

Dato escrito en programa	Lectura Pin 1	Lectura Pin 14	Lectura Pin 16	Lectura Pin 17	Orden asignada
11	0	0	0	0	Reset
3	0	0	0	1	LSB RAM
15	0	0	1	0	MSB RAM
7	0	0	1	1	DATO
9	0	1	0	0	ESCRIBIR RAM
1	0	1	0	1	LSBFREC
13	0	1	1	0	MSBFREC
5	0	1	1	1	ORDENES
10	1	0	0	0	LIBRE
2	1	0	0	1	LIBRE
14	1	0	1	0	LIBRE
6	1	0	1	1	LIBRE
8	1	1	0	0	LIBRE
0	1	1	0	1	LIBRE
12	1	1	1	0	LIBRE
4	1	1	1	1	LIBRE

Cuadro 3.2.3 Ordenes de control.

Es decir, si en el programa se escribe la sentencia:

Out 378h, 11

se ejecutará lo orden correspondiente a limpiar cualquier dato que se encuentre almacenado en el circuito de control y grabado de memoria.

Si se ejecuta la sentencia:

Out 378h, 9

en ese instante el dato se graba en memoria RAM; siempre y cuando ya se haya direccionado la localidad escogida de memoria para grabar y puesto el dato a la entrada de la memoria. El número 378h corresponde a la dirección del pórtico de control del puerto paralelo.

En el cuadro 3.2.3 se definen los términos:

RESET

Orden para inicializar funciones del sintetizador.

LSB RAM

Orden para direccionar los primeros 8 bits de direccionamiento a memoria.

MSB RAM

Orden para direccionar los 8 bits siguientes de la palabra de direccionamiento a memoria. La palabra de direccionamiento a memoria consta de 13 bits.

DATO

Orden para mandar el dato de la forma de onda a la entrada de datos en la memoria.

ESCRIBIR RAM

Orden para escribir el dato en memoria, el cual ya se encontraba direccionado a memoria.

LSBFREC

Orden para seleccionar los primeros 8 bits de la palabra de control de frecuencia.

MSBFREC

Orden para seleccionar los segundos 8 bits de la palabra de control de frecuencia. La palabra de control de frecuencia tiene 16 bits.

LIBRE

Son sentencias que se encuentran disponibles al usuario. Se pueden utilizar para futuras aplicaciones.

ORDENES

Orden para seleccionar los dos modos de operación del puerto paralelo. Estos son: grabar los datos en el sintetizador y controlar el funcionamiento del sintetizador, deshabilitando el circuito de grabado de datos en memoria. Refiriendonos al capítulo 2, parte 2.3.4 los valores que se deben mandar por el pórtico paralelo son los que se muestran en el cuadro 3.2.4.

	Buffers del circuito de grabado en memoria	Habilitación de lectura o escritura en RAM	Habilitación del acumulador digital.
Bit de control	1	1	0
Bit de control	0	0	1

Cuadro 3.2.4 Ordenes al sintetizador.

Es decir, cuando se deseen mandar datos a ser grabados en la memoria, el circuito del acumulador deberá estar deshabilitado, esto se logra poniendo un 0 lógico en el bit de control, al mismo tiempo que se habilita el acceso a escribir en

memoria con el bit respectivo puesto en 1 lógico. Una vez que se han acabado de grabar los datos, el circuito por orden del usuario deshabilita el circuito de grabado y pone los valores de 1L y 0L en el acumulador digital y memoria respectivamente.

Este proceso se logrará mandando las ordenes por el bus de datos del pòrtico y seleccionando la orden específica para que el circuito opere en el modo escogido.

e) Selección de Frecuencia de Generación del Sintetizador.

La palabra de control de frecuencia es de 16 bits, por esta razón se fracciona a la misma en dos palabras de 8 bits para mandar la información por el pòrtico de datos del puertò paralelo en forma secuencial. El control de frecuencia del sintetizador esta definido de acuerdo a la teoría de la síntesis digital como un número entre 0 y 32768. La selección de frecuencia se lo realiza de manera visual de acuerdo a la figura 3.2.3. El manejo de frecuencia se realiza mediante la utilización del ratón usando la barra de direccionamiento inferior, al mismo tiempo que se muestra la frecuencia a la que se esta generando la señal escogida.

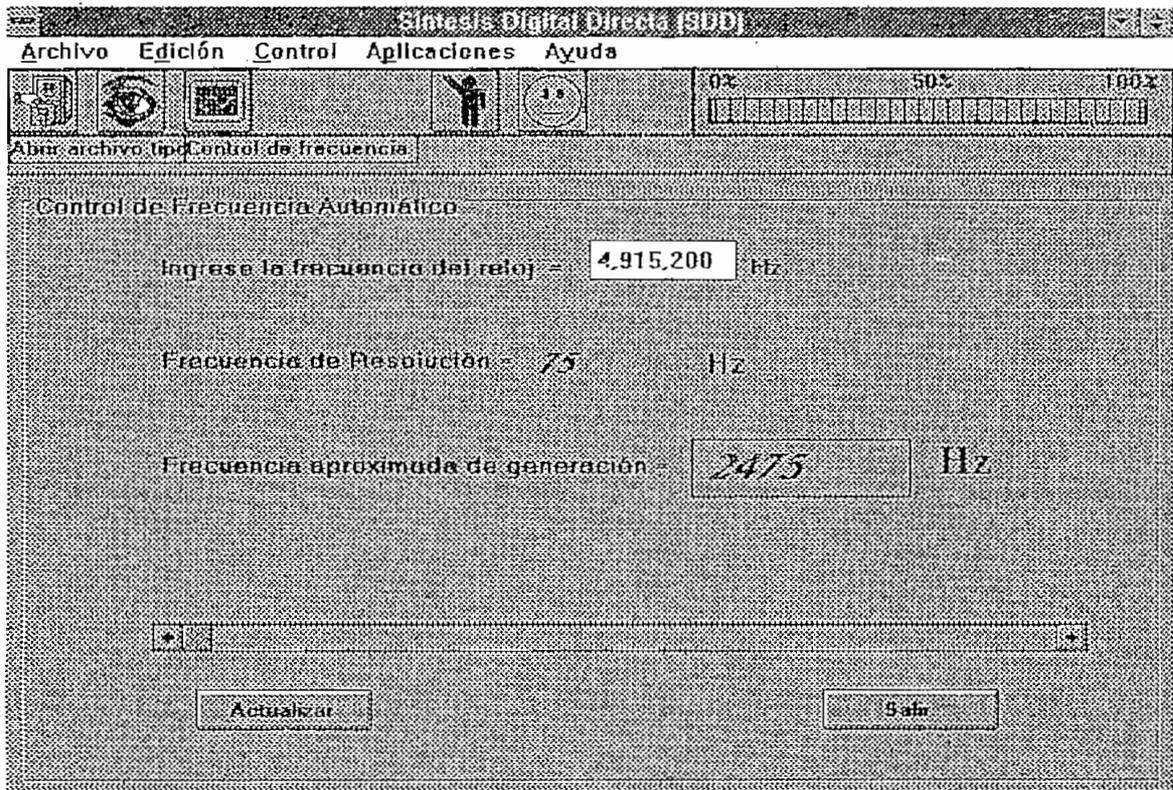


Figura 3.2.3 Selección de frecuencias en el sintetizador

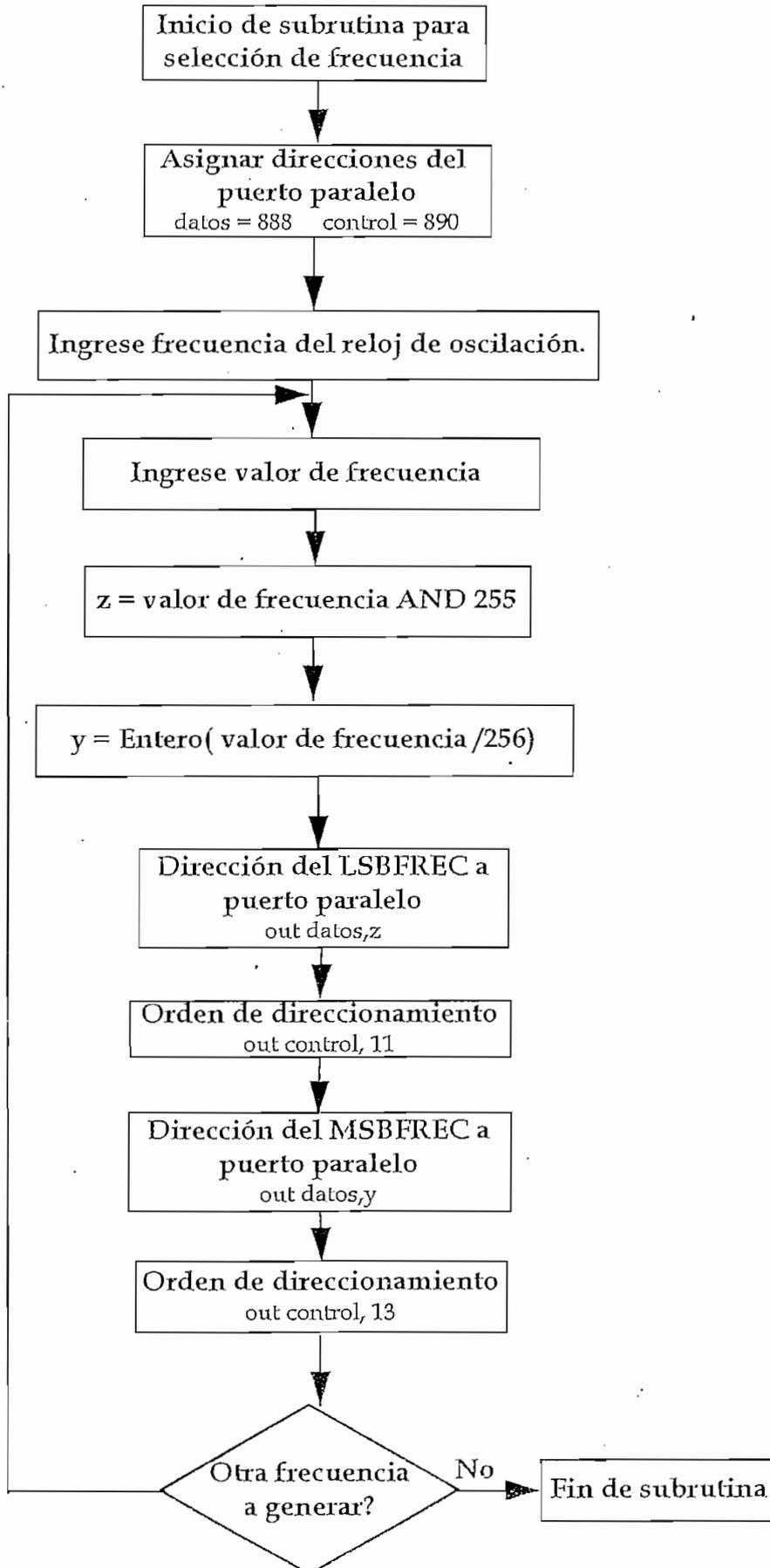


Diagrama 3.2.2 Diagrama del flujo del control de frecuencia.

3.3 CONTROL Y PRESENTACION DE DATOS EN EL COMPUTADOR PERSONAL.

3.3.1 CONTROL DE INFORMACION MEDIANTE EL PUERTO PARALELO.

El puerto paralelo presenta la mayor versatilidad para el control y manejo del sintetizador digital; dentro del cual las ordenes que se envíen tienen un tratamiento previo mediante el manejo visual. El funcionamiento del pórtico es realizado mediante algoritmos que no tiene acceso el usuario, de tal modo que no existan manipulaciones y daño a elementos externos y al mismo pórtico. Se hará uso de un archivo tipo DLL (Librería de Enlace Dinámico) conocido como CUSER2.DLL que permite la operación directa al pórtico usando los comandos conocidos para manejo del puerto paralelo.

Los comandos de escritura y lectura del puerto paralelo en programas anteriores a Visual Basic hacen uso del formato de comandos descritos en el capítulo 2, sección 2.3.3. Si se escriben las mismas sentencias en el compilador de VisualBasic, el programa generara un error de sintaxis, tal como se muestra en la figura 3.3.1; esto se debe a que no existen los comandos Inp y Out dentro de VisualBasic.

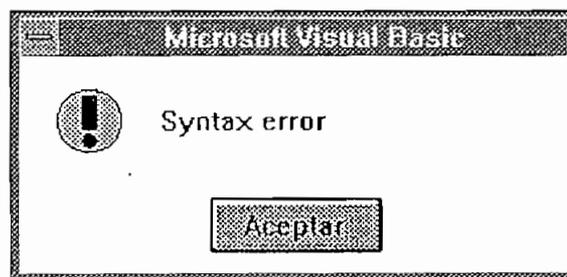


Figura 3.3.1 Error de sintaxis.

La filosofía de trabajo dentro de Windows es crear archivos individualizados, de tal modo que cualquier aplicación dentro de Windows puede encontrar las herramientas necesarias para ejecutarse. Por ejemplo, si en un procesador de palabras bajo Windows se desea añadir nuevos tipos de letras, solo tiene que actualizar las fuentes¹¹ del sistema, sin necesidad de hacer cambios al programa.

Para el caso de crear un programa autónomo en Visual Basic, se necesita acceder a librerías de enlace dinámico (DLL), donde existen funciones especializadas que un programa Windows puede llamar cuando las necesite.

Windows incluye más de 600 funciones en tres bibliotecas de enlace dinámicas, se las denomina funciones de interfaz de programación de aplicaciones (Application Programming Interface API). Las funciones DLL que se utilizará frecuentemente se encuentran en tres bibliotecas de enlace: KERNEL.EXE, GDI.EXE, y USER.EXE. Pero dentro de estas tres librerías no existe ningún comando de control al puerto paralelo, por lo que es necesario crear una librería de enlace dinámico individualizada para manejar el puerto.

Estas librerías individualizadas se pueden crear a partir de lenguajes de programación que acceden directamente a funciones del microprocesador interno del computador, como Borland C++, de tal forma que se pueda acceder al pórtilo sin problemas.

El archivo CUSER2.DLL permite hacer operaciones de entrada/salida como INP y OUT que operan de la misma manera que lo hacen en QuickBasic. El programa hecho en Borland C++, como se muestra en el punto 3.3.2.

3.3.2 PROGRAMA EN BORLAND C++ PARA MANEJO DEL PUERTO PARALELO BAJO WINDOWS.

¹¹ Tipos de letras que el procesador posee.

El listado mostrado a continuación se recogió de Internet, en un artículo referente al manejo del p \acute{o} rtico paralelo desde Visual Basic.

LISTADO DE ARCHIVO CUSER2.DLL, ANTES DE SER COMPILADO

```
#include <c:\borlandc\include\windows.h>
#include <c:\borlandc\include\dos.h>

extern "C" {int FAR PASCAL _export Test (int) ;
void FAR PASCAL _export Out (int, int) ;
int FAR PASCAL _export Inp (int) ;}
int FAR PASCAL LibMain (HANDLE hInstance, WORD wDataSeg,
        WORD wHeapSize, LPSTR lpCmdLine)

{
    hInstance = hInstance;
    wDataSeg = wDataSeg;
    wHeapSize = wHeapSize;
    lpCmdLine = lpCmdLine;
    if(wHeapSize > 0)
        UnlockData (0);
    return 1 ;
}

int FAR PASCAL _export Test (int arg1)

{
    return (arg1 + 1);
}

void FAR PASCAL _export Out (int portaddr, int portdata)
```

```

{
    unsigned char data;
        outportb(portaddr,portdata);
}
int FAR PASCAL _export Inp (int portaddr)
{
    int portdata;
    portdata = inportb(portaddr);
    return (portdata);
}

```

Para usar Inp y Out en Visual Basic se debe decirle al programa que estas operaciones existen. Se debe recordar que la operación OUT es una subrutina que envía un byte a un puerto de salida, no regresando el valor al programa de VisualBasic. La operación INP es una función, debido que permite regresa un byte desde un puerto de entrada.

Para que Visual Basic entienda la existencia de la nueva librería, se deberá grabar al archivo CUSER2.DLL bajo el siguiente camino en el disco duro del computador:

C:\windows\system\cuser2.dll

Y mediante las declaraciones que se muestran en el cuadro 3.3.1 grabadas dentro de la forma general de Visual Basic, se tendrá acceso a nuevos comandos, en este caso INP y OUT para control del puerto paralelo.

```

Declare Function Test Lib "c:\windows\system\cuser2.dll" (ByVal Numb%) As Integer
    Declare Sub Out Lib "c:\windows\system\cuser2.dll" (ByVal Addr%, ByVal Byte%)
    Declare Function Inp Lib "c:\windows\system\cuser2.dll" (ByVal Addr%) As Integer

```

Cuadro 3.3.1 Declaraciones para librería Cuser2.dll

En la figura 3.3.2 se muestra en que forma¹² de Visual Basic se deberá poner los comandos respectivos.

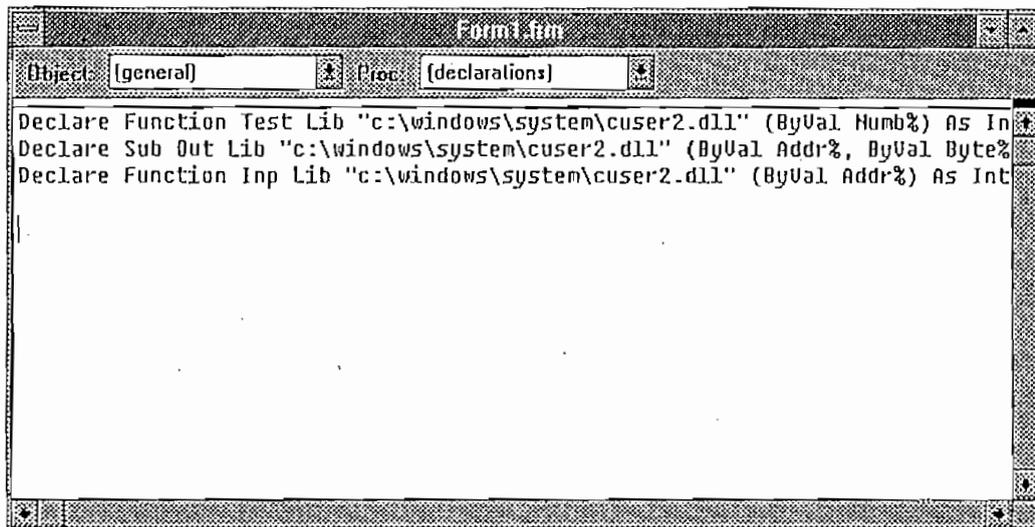


Figura 3.3.2 Listado de ordenes al puerto paralelo.

3.4 MANEJO VISUAL DE FORMAS DE ONDA POR PROGRAMA.

En el capítulo 3 parte 3.3, se definieron todas las subrutinas¹³ que deberán ser ejecutas para realizar el control y manejo del sintetizador digital desde el computador. Es el objetivo de este manual describir cual fue el proceso de programación de la aplicación gráfica que controla al equipo, así como de permitir a cualquier otro programador realizar modificaciones al mismo.

¹² Se denomina a una forma en Visual Basic aquel entorno de trabajo en el cual se definirán componentes del programa, como por ejemplo: menus de selección, botones, etc.

¹³ Subrutina.- Se refiere al código de programa que es llamado a ejecutar ciertas acciones específicas del programa principal

Definición de necesidades.-

a) El programa se ejecutará en el ambiente Windows 3.1. Contará con menus de selección.

b) Se necesita de controles visuales para las siguientes funciones:

- Lectura de archivos y presentación visual de datos.

Consiste en leer los datos de los archivos con extension **SDD**, mostrando en pantalla la forma de la señal deseada. Además se permite al usuario escoger dos opciones de lectura. La primera consiste en leer datos y mostrar en pantalla y la segunda se refiere a ejecutar el procedimiento de lectura de datos, pero a la vez se graban los datos en el sintetizador digital.

- Control de frecuencia del sintetizador.

Se seleccionara la frecuencia del sintetizador desde el computador mediante un interfase gráfica con el usuario. Permitirá realizar los cambios de la frecuencia sin mayores dificultades.

- Barrido de frecuencia del sintetizador.

Se presenta como una aplicación de la síntesis digital. En lo posible el barrido ocupa controles de frecuencia utilizados anteriormente. Se presentará el control de frecuencia en forma visual.

- Manejo de ayudas

Se presentan referencias del programa y de la aplicación.

- Control de errores.

Controlar en lo posible los errores que se puedan generar en la ejecución

del programa.

a) Manejo de menus.-

La barra de menú aparece inmediatamente debajo de la barra de títulos, tal como se observa en la figura 3.4.1; en la cual se definen ciertos contenidos de acuerdo a las necesidades planteadas. Para acceder a estas funciones se tiene dos vías: usar el mouse o usar el teclado. El proceso que se sigue para crear un menu de este tipo, es el siguiente:

- a) Diseñar y planificar cuales funciones deberán aparecer en la barra de menus.
- b) Agrupar funciones afines y definir prioridades.
- c) Usar el Diseñador de Menus que provee Visual Basic.

En la aplicación creada se definieron y agruparon las funciones de acuerdo a la figura 3.4.1.

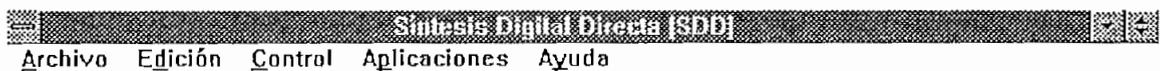


Figura 3.4.1. Barra de Menús.

Para crear una barra de menús de Visual Basic se hace uso del Diseñador de Menús siguiendo el procedimiento siguiente:

- 1) En el menú de Visual Basic escoger Window y luego Menu Design, aparecerá la figura 3.4.2
Se le conoce como Diseñador de Menús en el cual se puede crear los menús planificados anteriormente.
- 2) En la caja que dice Caption, escribir el texto que se desea que aparesca en la barra de menú.

- 3) En la caja que dice Name, escribir el nombre que se usará como referencia en el control del menú. Recordar este nombre es de suma importancia, ya que el llamado a subrutina se lo realiza con esta designación.

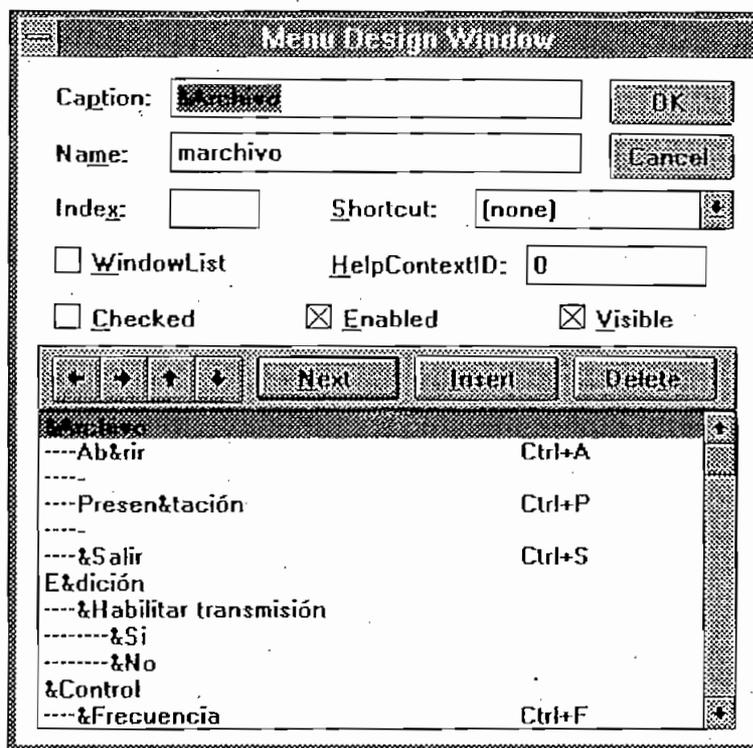


Figura 3.4.2. Diseñador de Menús.

- 4) Usar las flechas de izquierda o derecha para cambiar el nivel de función, de acuerdo a la planificación realizada anteriormente.
- 5) Definir las propiedades, si es que se necesita. Se refiere a escoger de Checked, Enabled, Visible, Index.
- 6) Escoger Next para crear otra función de Menú.
- 7) Repetir el procedimiento desde 2, hasta que se hayan creado todas las funciones.
- 8) Presionar OK para cerrar el diseñador de Menús.

Siguiendo este procedimiento se puede crear fácilmente una barra de menús, con sus respectivos funciones asignadas a cada una.

b) Pantalla de presentación.

Para la pantalla de presentación se hizo uso de los controles que se indican en la figura 3.4.3, definiendo además el **Nombre** de programa para cada botón de control; con el cual se le conocerán en la ejecución del código . Es decir se designa un nombre cualquiera al botón (objeto), de tal forma que cuando se ejecute el diagrama de flujo respectivo se haga la llamada respectiva a cada objeto. Por ejemplo el botón designado como 1A ejecuta la rutina designada a leer datos de archivo y graficarlos en pantalla.

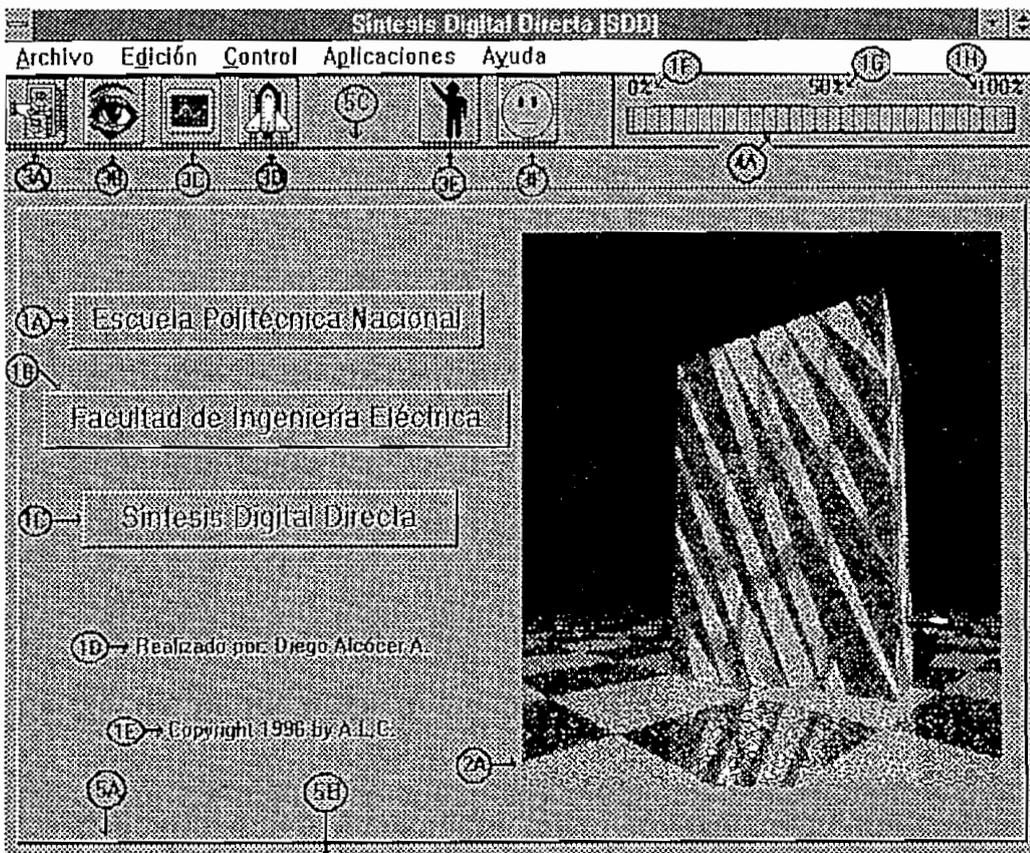


Figura 3.4.3 Pantalla de presentación.

Es importante indicar los nombres escogidos para cada objeto. En el cuadro 3.4.1, se señala el nombre escogido, así como la función asignada. En la figura

3.4.3. además de indican a cuales controles se les ha designado el nombre respectivo indicado en el cuadro 3.4.1.

La pantalla de presentación cuando se inicia el programa siempre se encuentra activa. Cuando se deja de ejecutar una subrutina determinado el programa siempre vuelve a mostrar la pantalla principal. Los menús y los controles visuales solo tienen acceso mediante teclado y el ratón.

Referencia	Control	Nombre	Función
1A	Label	escuela	Muestra texto "Escuela Poli....."
1B	Label	facultad	muestra texto "Facultad de....."
1C	Label	sisntesis	muestra texto "Síntesis Digí....."
1D	Label	diego	muestra texto " Realizado por"
1E	Label	version	muestra texto " Copyright....."
1F	Label	textopor(2)	muestra texto 0%
1G	Label	textopor(0)	muestra texto 50%
1H	Label	textopor(1)	muestra texto 100%
2A	Caja de imágenes	novell	muestra gráfico en pantalla principal
3A	botón de comando	boton1	ejecuta subrutina de lectura de datos
3B	botón de comando	boton2	ejecuta subrutina de grabado en memoria
3C	botón de comando	boton4	ejecuta subrutina de control de frecuencia
3D	botón de comando	boton9	ejecuta subrutina de barrido de frecuencia
3E	botón de comando	boton7	ejecuta subrutina de pantalla principal
3F	botón de comando	boton8	salir del programa
4A	Control de medida	indica	muestra cuando datos leídos del archivo
5A	Frame	panel2	forma sobre la cual se colocan los botones
5B	Frame	panel1	forma sobre la cual se coloca el panel 2

Cuadro 3.4.1. Controles en la pantalla de presentación.

Para usar el control de dialogo común se sigue el siguiente procedimiento:

- a) Seleccionar el botón de diálogo común de la caja de herramientas.
- b) Escoger que tipo de ventana de dialogo se usará. Se tienen las siguientes opciones:

Propiedad escogida	Dialogo mostrado
0	No hay acción
1	Abrir archivo
2	Grabar como
3	Escoger colores
4	Escoger tipo de letra
5	Imprimir
6	Invocar archivos de ayuda

Cuadro 3.4.2 Funciones del botón de dialogo común.

La ventana de diálogo permite especificar un drive¹⁴, un directorio, un archivo con una extensión, y un nombre de archivo. Además permite operaciones de lectura de datos de archivos. El nombre que se ha escogido para designar a la ventana de diálogo en el programa desarrollado es el de **Abrir**.

La ventana de dialogo para abrir archivo se presenta en la figura 3.4.4

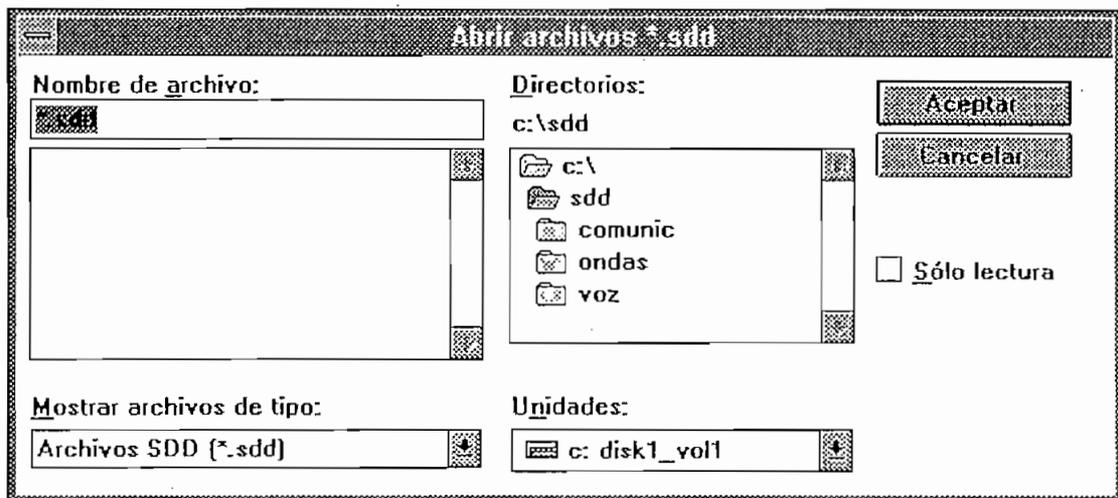


Figura 3.4.4. Ventana de Diálogo.

Una vez que se ha empezado a leer los datos del archivo abierto estos son mostrados en pantalla sobre una forma preestablecida. Es importante mencionar

¹⁴ Drive.- Unidad para leer o grabar datos de una unidad de disco flexible.

que para realizar este proceso se debe abrir una nueva forma y sobre la misma graficar los datos que se sigan leyendo. Para graficar los datos se hace uso del comando PSET, que es el mismo que se ha usado en versiones inferiores a Visual Basic, como por ejemplo GwBasic, Qbasic.

Para graficar los datos con PSET se tiene la instrucción siguiente:

PSET (x,y)

donde X es el valor de la dirección a memoria donde se escribirán los datos

Y es el valor de la amplitud de la forma de onda.

En el programa se ha hechos dos distinciones precisas: la una se refiere a solo leer los datos del archivo y mostrarlos graficamente y la otra tiene que ver que a mas de mostrar los datos en pantalla se graban los datos en el circuito del sintetizador digital. Para ejecutar estas subrutinas se hace uso de los botones de acceso rápido descritos en el cuadro 3.4.1, correspondiente al menú principal.

En la figura 3.4.5. se muestra la pantalla que muestra la forma de la onda que se esta leyendo del archivo escogido en la ventana de dialogo común.

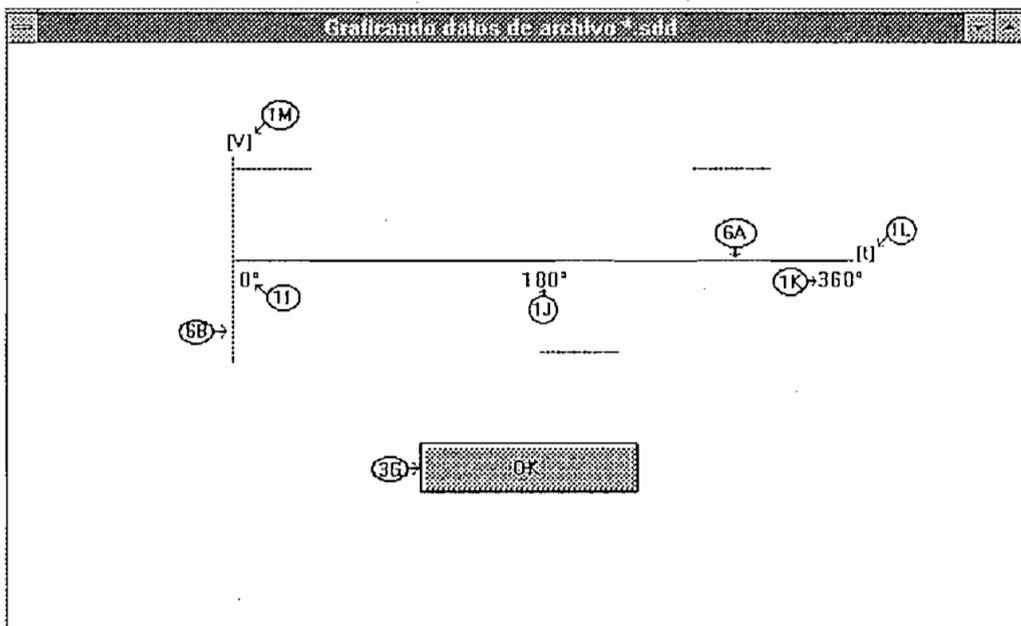


Figura 3.4.5. Pantalla de la forma de onda escogida.

Además se ejecuta el comando PSET se utiliza otras herramientas para dibujar que posee Visual Basic, entre las cuales tenemos: dibujar líneas y texto. En el cuadro 3.4.3 se muestra el nombre asignado al botón respectivo de acuerdo a la figura 3.4.5.

Referencia	Boton	Nombre Asignado	Función
1I	label	grado(0)	Muestra "0 grados"
1J	label	grado(1)	muestra "180 grados"
1K	label	grado(3)	muestra "360 grados"
1L	label	grado(4)	muestra "t(s)" en pantalla
1M	label	grado(2)	muestra "Vm" en pantalla
3G	Botón de comando	okgrafico	Botón para salir
6A	Línea	línea	muestra línea horizontal
6B	Línea	línea	muestra línea vertical

Cuadro 3.4.3 Controles de la pantalla de gráfico de onda.

C) Control de frecuencias.

El control de frecuencias se lo realiza mediante el llamado del botón asignado para esta función tal como se describió en la sección 2. Para el control de frecuencias se construye la pantalla que se muestra en la figura 3.4.6. Se ha utilizado el control referido como Barra de desplazamiento horizontal, el cual tiene la característica que cuando se mueve a la derecha o a la izquierda genera un valor entre el rango de 0 a 64532. En el programa se la ha utilizado para escoger la frecuencia de generación del sintetizador. La misma solo tiene acceso mediante el ratón, no permite controles de otro tipo.

El botón referido a Control de máscara permite el ingreso de datos numéricos en un total de 7 dígitos. El manejo del mismo es posible con el ratón o desde teclado.

El manejo de cada control es explicado en el Manual de referencia de Visual Basic 3.0. Es de importancia especificar el nombre con el cual se ejecuta las instrucciones una vez que se ejecuta el código.

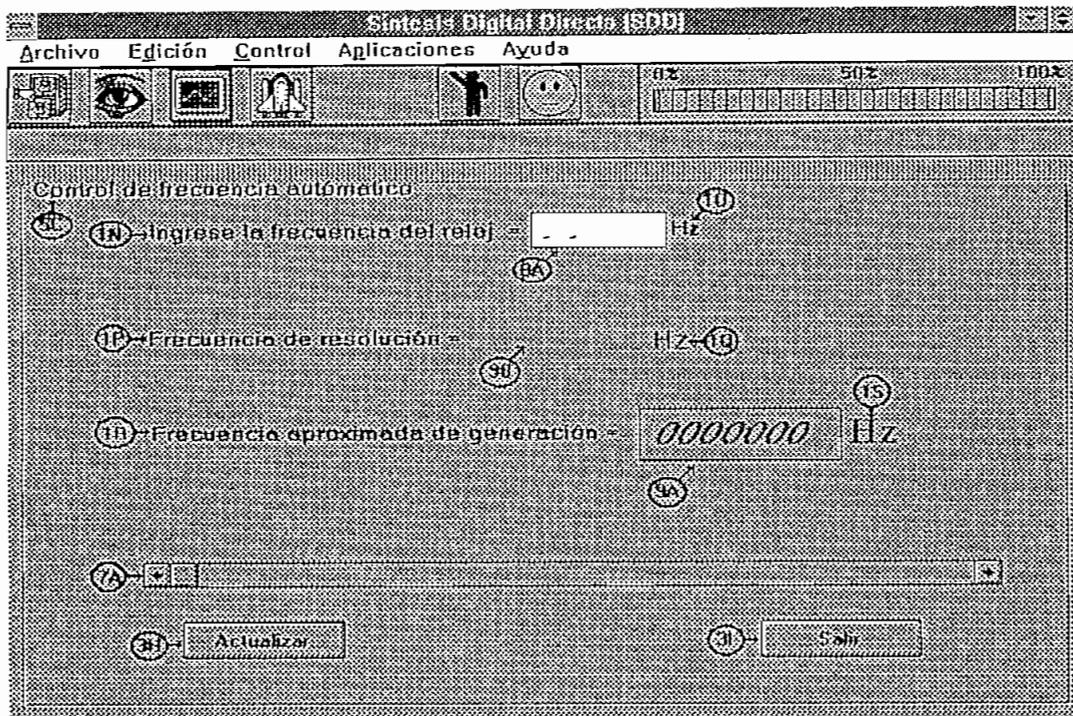


Figura 3.4.6 Control de frecuencias.

De acuerdo a la figura 3.4.6, se designaron los nombres y funciones respectivas de acuerdo al cuadro 3.4.4

Referencia	Botón	Nombre	Función
1N	label	ventana(0)	Muestra "Ingresa valor de frec....."
1O	label	ventana(1)	muestra la palabra Hz
1P	label	ventana(4)	muestra en pantalla: frecuencia de reso....
1Q	label	ventana(2)	muestra la palabra Hz
1R	label	ventana(5)	muestra " Frecuencia generada"
1S	label	ventana(3)	muestra la palabra Hz
3H	control	actualizar	ejecuta subrutina de actualizar valores
3I	control	command 3	sale de la pantalla de control de frecuencia
5C	frame	frecuencia	pantalla sobre el cual se colocan los botones
7A	barra de desplaz.	selector	selección de la frecuencia de generación
8A	máscara	maskauno	ingresar dato del reloj de referencia escogido
9A	text box	textouno	muestra la frecuencia de resolución
9B	text box	texto	muestra la frecuencia generada de la señal

Cuadro 3.4.4. Controles de frecuencias.

d) Barrido de Frecuencias

El barrido de frecuencias se constituye en una aplicación de la síntesis digital directa. Se ha usado la pantalla base del control de frecuencia, adisionando algunas controles visuales, además de eliminar algunos. El pantalla de barrido de frecuencias se lo muestra en la figura 3.4.7.

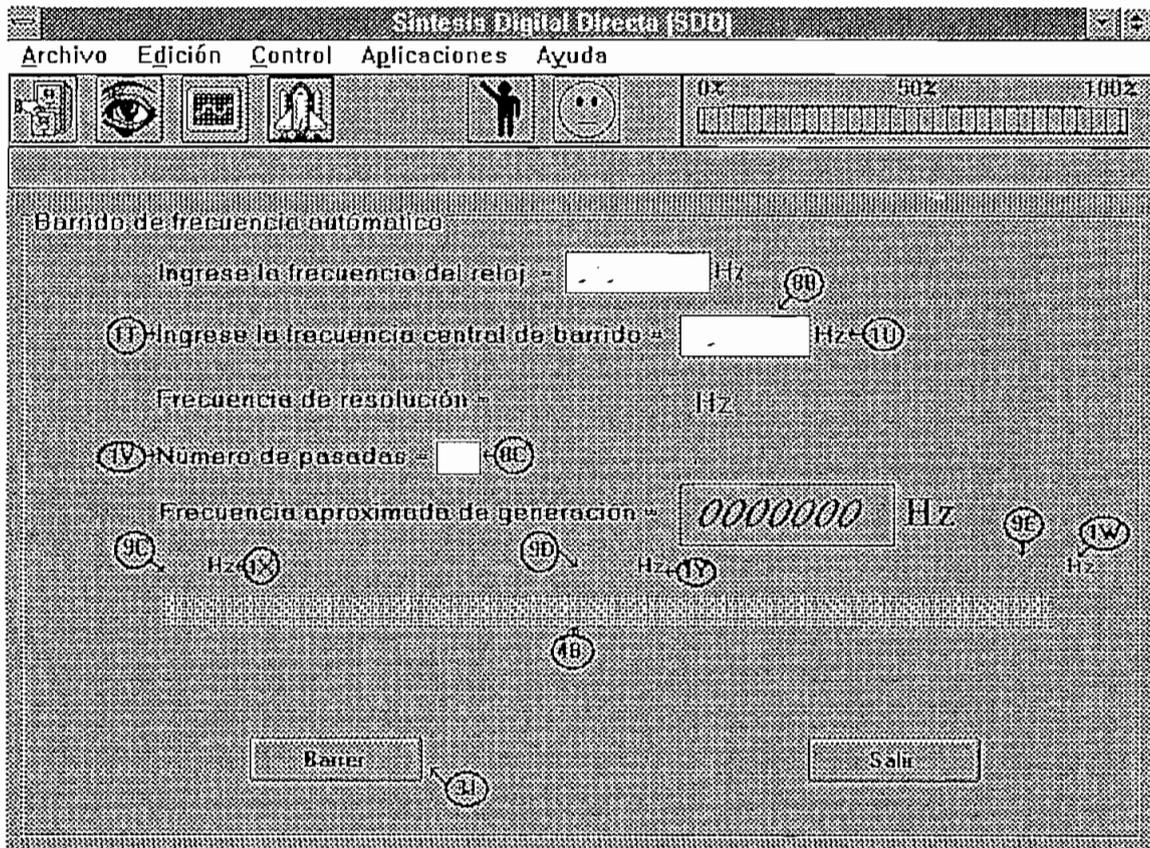


Figura 3.4.7. Barrido de frecuencias.

De acuerdo a la figura 3.4.7, se designan los nombres asociados al código, tal como se indica en el cuadro 3.4.5.

En el mismo se establecen los nombres asignados para la ejecución del código del programa.

e) Manejo de Ayudas

Referencia	Boton	Nombre	Función
1T	label	ventana(6)	muestra en el mensaje: "frecuencia central..
1U	label	ventana(7)	muestra en pantalla la palabra Hz
1V	label	ventana(8)	muestra en pantalla: numero de pasadas
1W	label	ventana(11)	muestra en pantalla Hz.
1X	label	ventana(10)	muestra en pantalla Hz.
1Y	label	ventana(9)	muestra en pantalla Hz.
3I	botón de control	barrer	sale de la pantalla
4B	control de medida	medir	muestra la frecuencia generada en el barrido
8B	máscara	maskarados	para ingresar dato de frecuencia central
8C	máscara	pasadas	dato de cuantas veces se ejecuta el barrido
9C	text box	eje(0)	indica la frecuencia mas baja de barrido
9D	text box	eje(1)	indica la frecuencia central de barrido
9E	text box	eje(2)	indica la frecuencia más alta de barrido.

Cuadro 3.4.5. Control de barrido de frecuencias.

Para manejar ayudas en Visual Basic existe muchas posibilidades, dependiendo cada una de la complejidad que se requiera. Se ha usado el control denominado caja de diálogo, dentro del cual se han escrito ciertos mensajes aclaratorios. En la figura 3.4.8 se muestra el control de ayuda, referido a : Acerca de, que contiene cierta información del programa y del tema desarrollado.

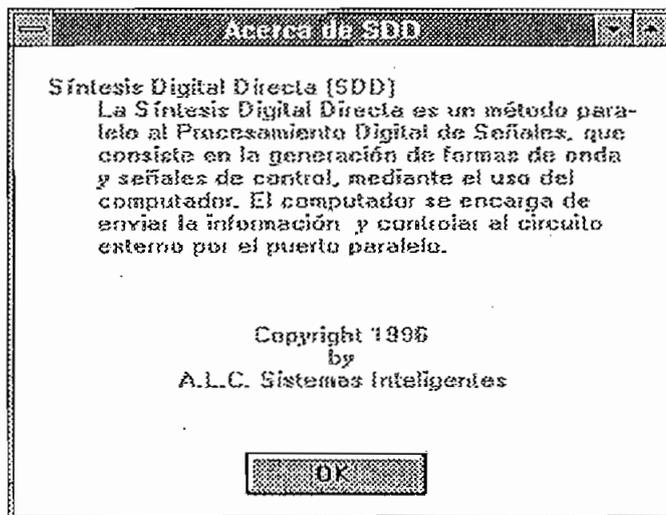


Figura 3.4.8. Ayuda de programa.

e) Control de Errores

El control de errores se ejecuta el momento que la aplicación a encontrado un error generado. Generalmente los errores se deben por ejemplo a: división de un número para cero, no ingreso de datos, ingreso de datos incorrecto, etc. En Visual Basic se puede centralizar los errores y controlarlos, es decir el momento que ocurre un error, el programa lo controla y no permite que el programa sea abortado. Además de posibilitar ese control, se puede mostrar un mensaje que alerte al usuario que esta cometiendo un error. Esto se realiza mediante comando **On error goto** y **Msg** que se lo escribe en el código de programa.

On error goto --> Cuando suceda un error vaya a la subrutina indicada.

Msg --> muestre el siguiente mensaje de alerta.

Ejemplo 9:

On Error goto Manejador de errores:

Manejador de errores:

Msg " No ha ingresado datos de frecuencia central y de reloj"

y el resultado de este código ejecutado se muestra en la figura 3.4.9.

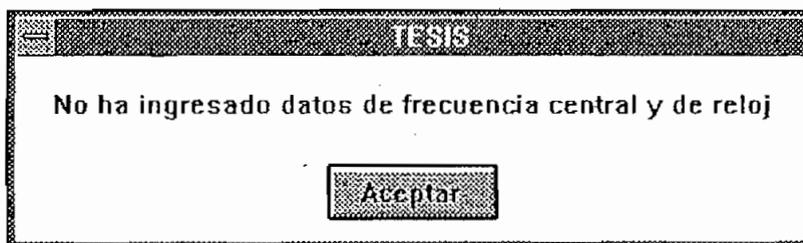


Figura 3.4.9. Pantalla de alerta de error.

f) Ejecucion de archivo con extensión exe

Una vez que se han realizado todos los pasos de diseño y la aplicación se encuentra completa, el archivo o proyecto se lo puede hacer ejecutable sin necesitar de llamar necesariamente a Visual Basic, para esto se sigue el siguiente procedimiento:

a) Escoger Archivo en Visual Basic.

b) En el menú de archivo escoger Make Exe File. Se mostrara la figura 3.4.9.

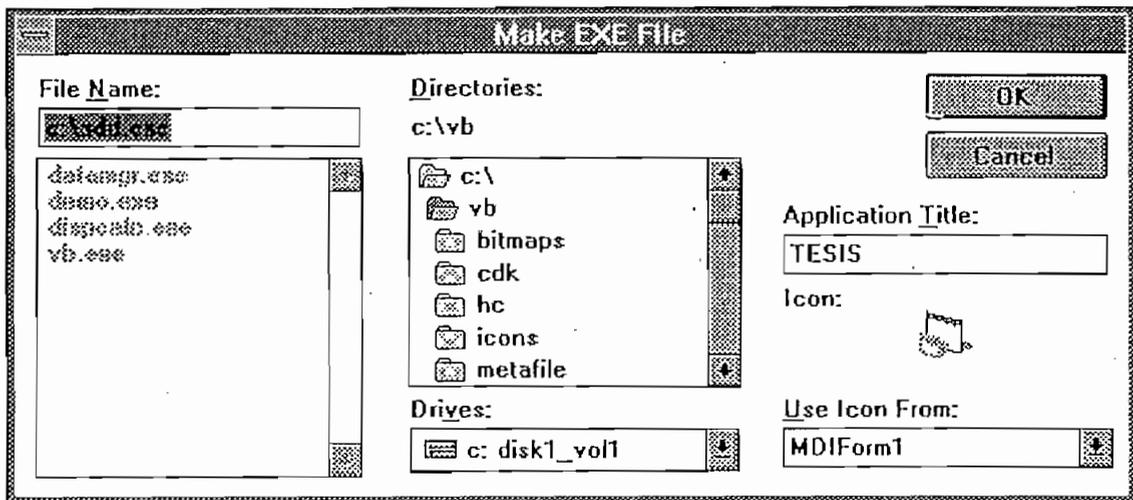


Figura 3.4.9. Para realizar archivo .EXE

En la figura 3.4.9. se especifica el nombre con el que se desea conocer la aplicación y el icono con el cual se desea asociar cuando se minimize la ventana.

Luego de lo cual se escoge OK y el archivo se encuentra listo a ser utilizado.

CAPITULO IV

4.1. Medidas Experimentales

4.1.1 Pruebas de operación del DDS.

Descripción fotográfica y de pruebas.

4.1.2 Operación del DDS en circuito abierto.

Determinación de respuesta Voltaje vs. Frecuencia.

4.1.3 Operación del DDS en circuito con carga resistiva.

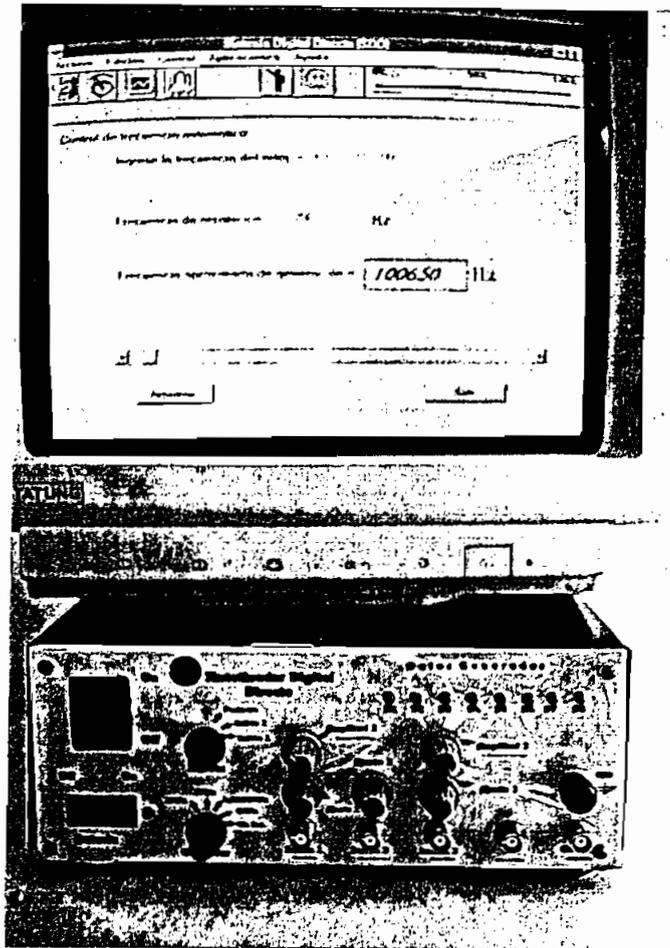
Determinación de respuesta Voltaje vs. Frecuencia con diferentes cargas.

4.1.4 Medición de Frecuencia.

Comparación de resultados.

4.1.1 MEDIDAS EXPERIMENTALES.

El sintetizador fue montado en una caja metálica tipo rectangular como el mostrado en la fotografía 4.1.1. En ella puede verse la parte frontal, en la que constan los controles de amplitud, controles de ajuste, control del reloj de referencia, habilitación de salidas y leds que indican que señal esta siendo grabada. Así como el conmutador de la batería de respaldo.



Fotografía 4.1.1 Vista frontal del sintetizador.

Los control de amplitud de las salidas uno y tres tienen la función de variar la amplitud de la señal de salida en un rango máximo de 24Vpp. No se dispone de señales de mayor voltaje debido a limitaciones en la polarización del circuito amplificador a la salida. No debe excederse el rango de variación a 0 (posición mínima) del control de amplitud ya que el transistor utilizado no soporta corrientes mayores que 1 amperio.

El control de ajuste de las salidas uno y tres tiene como función el variar la polarización del amplificador de salida. Cuando se desea obtener mayor ganancia el variar este control permite obtener señales de muy diferentes amplitudes. Cuando se varia este control lo que se esta haciendo es variar el punto de polarización eliminando en algunos casos la componente de señal continua que se pueda obtener a la salida para algunas formas de onda especiales, por ejemplo: señales de comunicaciones.

El control de ajuste de la salida dos tiene como función el mejorar la señal triangular a frecuencias medias (20 Khz a 400 Khz). El control de ajuste esta constituido de un amplificador operacional con ganancia mayor que 1.

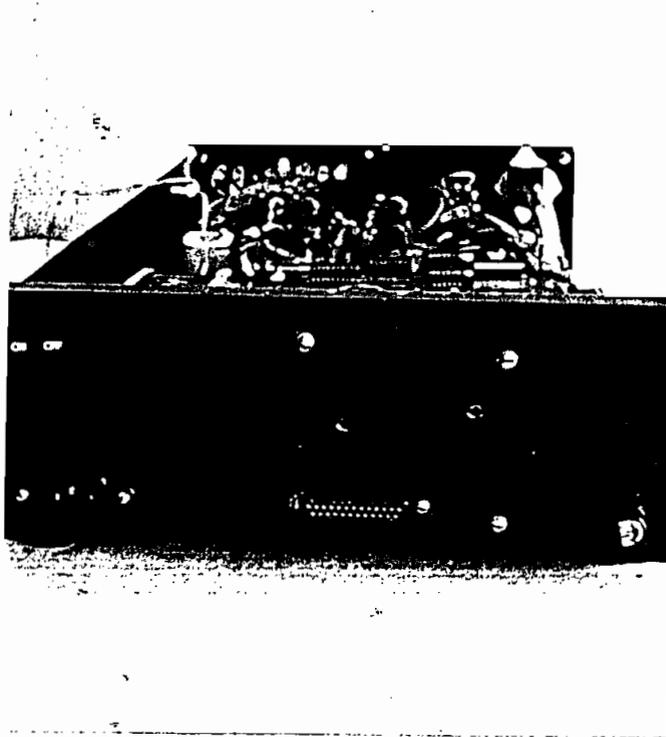
El selector del reloj de oscilación define cual será la frecuencias a la que el sintetizador podrá generar. En el cuadro 2.1.2 del capítulo 2 se muestra las frecuencia posibles a ser generadas con los osciladores respectivos. Se provee de una entrada externa a nivel TTL de un oscilador de otra frecuencia hasta el rango máximo de 4 Mhz.

Debido a que el sintetizador utiliza una memoria RAM en la que los datos digitales pueden ser perdidos si es que falla la alimentación de energía, se dispone de una batería de respaldo que provee 700mA de corriente durante una hora.

Los leds indican que datos están siendo grabados en memoria y una vez que el sintetizador entra en operación muestra los datos generados; aunque el ojo humano no está en capacidad de observar esta condición por la alta velocidad a la que se están siendo generandos los datos.

En la parte posterior del sintetizador tal como se observa en la fotografia 4.1.2 se encuentra: el interface paralelo de comunicaciones, el fusible de seguridad (2 Amp), la toma de alimentación 110/60Hz y el conmutador para cuando no se utilice el

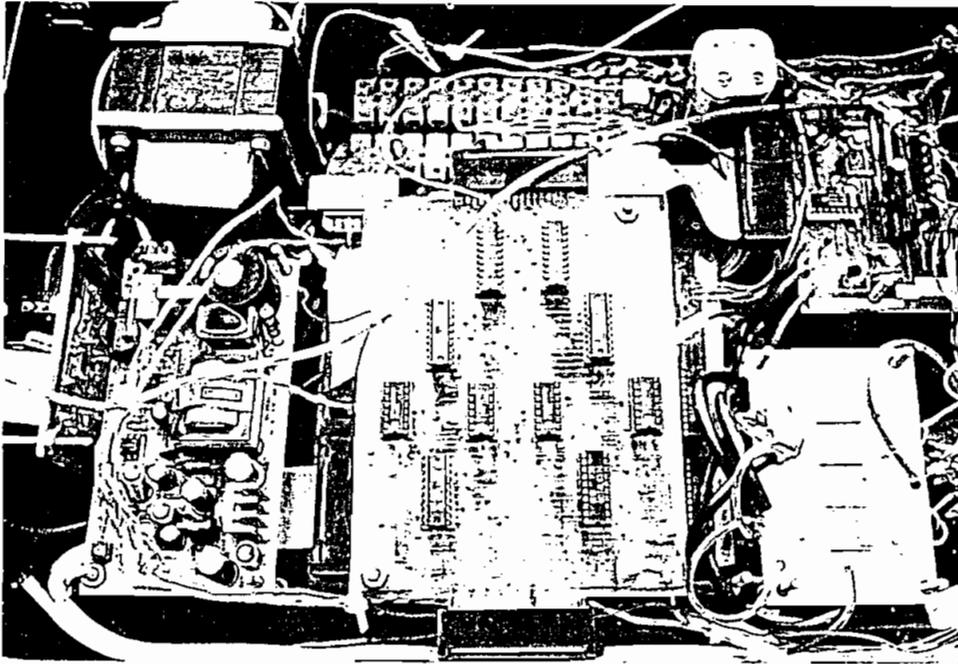
sintetizador por largo tiempo, su función es para conservar las baterías en óptimas condiciones.



Fotografía 4.1.2 Vista posterior del sintetizador.

El interior del sintetizador puede ser observado en la fotografía 4.1.3. Observando de arriba hacia abajo se tiene: primero: (1) las fuentes de alimentación, (2) la tarjeta del acumulador digital y (3) bajo esta la tarjeta de control e interface. (4) A un lado se encuentra las borneras que envía señales a los leds indicadores. A continuación se ven los osciladores de referencia (5) y la tarjeta de salida (6) en la que se encuentra el convertor digital/análogo. A un lado en la cubierta superior del sintetizador se puede observar el parlante (7) en el cual son escuchados los tonos generados por el equipo.

Los diagramas circuitales correspondientes: a) al acumulador directo se encuentra en la sección 2.2.1.2, figura 2.2.2. b) tarjeta de control e interfase, sección 2.3.4.1, figura 2.3.3 c) la tarjeta de los osciladores y la tarjeta de salida se muestra en la figura 2.3.6 de la sección 2.3.4.3.



Fotografía 4.1.3. Vista superior del sintetizador.

4.1.2. Operación del DDS en circuito abierto.

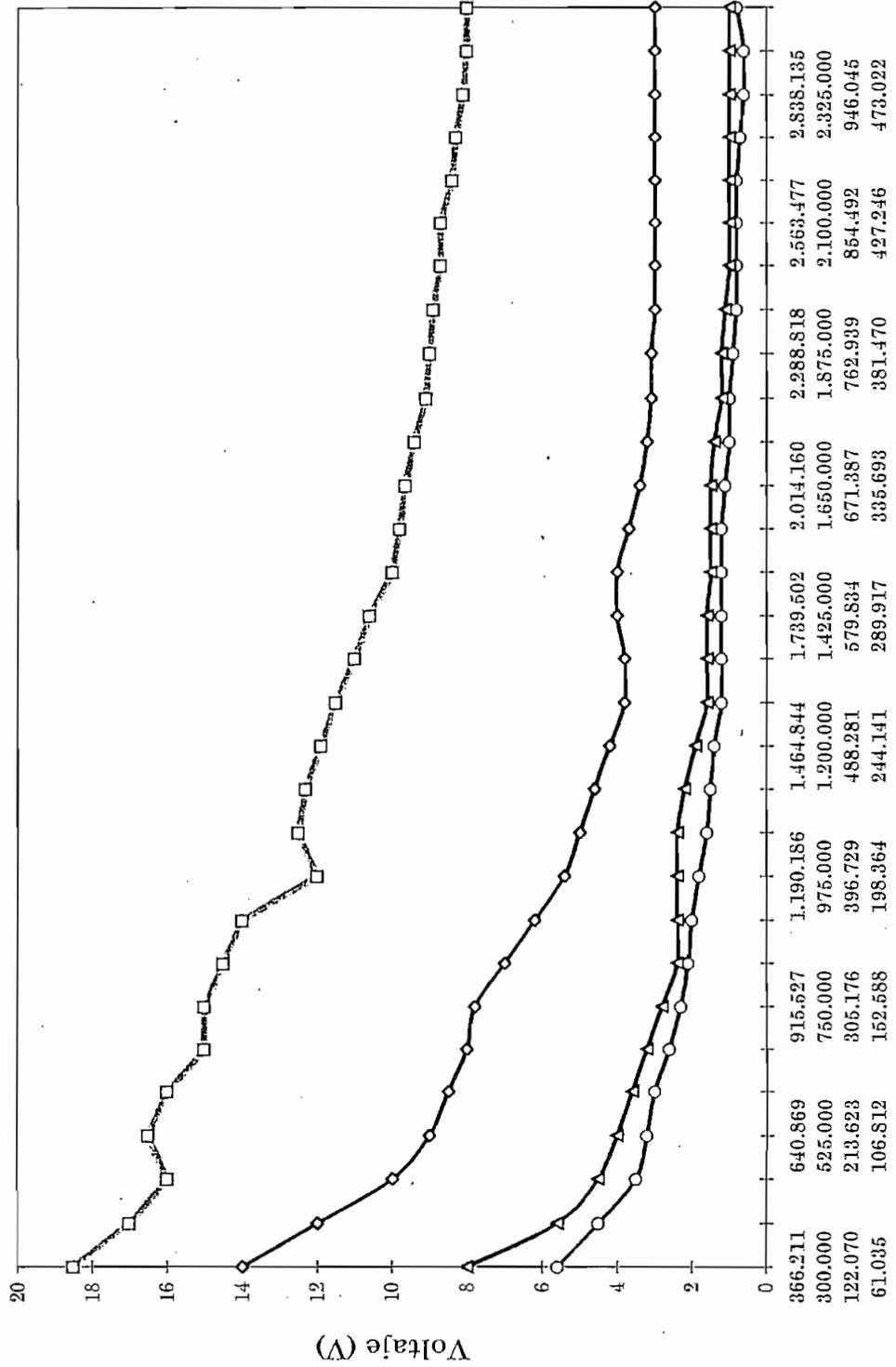
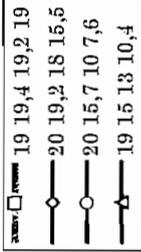
Dentro de este apartado se analizará el funcionamiento del sintetizador en circuito abierto (sin carga a la salida) para diferentes valores de frecuencia. Se utilizó una señal sinusoidal para la prueba, así como la salida uno.

Para frecuencias superiores a 1Mhz, la señal de salida presenta gran cantidad de ruido. En el cuadro 4.1.1 se muestra los voltajes obtenidos a la salida versus la frecuencia generada (depende del reloj de referencia). En la figura 4.1.1, además se indica la respuesta de frecuencia versus voltaje para cada uno de los osciladores con el que cuenta el sintetizador.

FRECUENCIA VS (VOLTAJE PICO - PICO Y VOLTAJE EFICAZ)

Reloj (Hz)	FRECUENCIAS GENERADAS					RELOJ DE REFERENCIA					RELOJ DE REFERENCIA					
	1000000	2000000	4915200	6000000	1000000	2000000	4915200	6000000	1000000	2000000	4915200	6000000	1000000	2000000	4915200	6000000
frec (Hz)	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344
n	frec(Hz)	frec(Hz)	frec(Hz)	frec(Hz)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vef(V)	Vef(V)	Vef(V)	Vef(V)
1	15,26	30,52	75	91,55	19	20	20	19	6,718	7,071	7,071	6,718	7,071	7,071	7,071	6,718
1000	15258,79	30517,58	75000	91552,78	19,4	19,2	15,7	15	6,859	6,788	5,551	6,859	6,788	5,551	6,788	5,308
2000	30517,58	61035,16	150000	183105,47	19,2	18	10	13	6,788	6,364	3,536	6,788	6,364	3,536	6,364	4,596
3000	45776,37	91552,78	225000	274658,20	19	15,5	7,6	10,4	6,718	5,480	2,687	6,718	5,480	2,687	5,480	3,677
4000	61035,16	122070,31	300000	369210,94	18,5	14	5,6	8	6,541	4,950	1,980	6,541	4,950	1,980	4,950	2,828
5000	76293,95	152587,89	375000	457763,67	17	12	4,5	5,6	6,010	4,243	1,591	6,010	4,243	1,591	4,243	1,980
6000	91552,73	183105,47	450000	549316,41	16	10	3,5	4,5	5,657	3,536	1,237	5,657	3,536	1,237	3,536	1,591
7000	106811,52	213628,05	525000	640869,14	16,5	9	3,2	4	5,834	3,182	1,131	5,834	3,182	1,131	3,182	1,414
8000	122070,31	244140,63	600000	732421,88	16	8,5	3	3,6	5,657	3,006	1,061	5,657	3,006	1,061	3,006	1,273
9000	137329,10	274658,20	675000	823974,61	15	8	2,6	3,2	5,303	2,828	0,919	5,303	2,828	0,919	2,828	1,131
10000	152587,89	305175,78	750000	915527,34	15	7,8	2,3	2,8	5,303	2,758	0,813	5,303	2,758	0,813	2,758	0,990
11000	167846,68	335693,36	825000	1007080,08	14,5	7	2,1	2,4	5,127	2,475	0,742	5,127	2,475	0,742	2,475	0,849
12000	183105,47	366210,94	900000	1098632,81	14	6,2	2	2,4	4,950	2,192	0,707	4,950	2,192	0,707	2,192	0,849
13000	198364,26	396728,52	975000	1190185,55	12	5,4	1,8	2,4	4,243	1,909	0,636	4,243	1,909	0,636	1,909	0,849
14000	213623,05	427246,09	1050000	1281738,28	12,5	5	1,6	2,4	4,419	1,768	0,566	4,419	1,768	0,566	1,768	0,849
15000	228881,84	457763,67	1125000	1373291,02	12,3	4,6	1,5	2,2	4,349	1,626	0,530	4,349	1,626	0,530	1,626	0,778
16000	244140,63	488281,25	1200000	1464843,75	11,9	4,2	1,4	1,9	4,207	1,485	0,495	4,207	1,485	0,495	1,485	0,672
17000	259399,41	518798,83	1275000	1556396,48	11,5	3,8	1,2	1,6	4,066	1,344	0,424	4,066	1,344	0,424	1,344	0,566
18000	274658,20	549316,41	1350000	1647949,22	11	3,8	1,2	1,6	3,889	1,344	0,424	3,889	1,344	0,424	1,344	0,566
19000	289916,99	579833,98	1425000	1739501,95	10,6	4	1,2	1,6	3,748	1,414	0,424	3,748	1,414	0,424	1,414	0,566
20000	305175,78	610351,56	1500000	1831054,69	10	4	1,2	1,5	3,536	1,414	0,424	3,536	1,414	0,424	1,414	0,530
21000	320434,57	640869,14	1575000	1922607,42	9,8	3,7	1,2	1,5	3,465	1,308	0,424	3,465	1,308	0,424	1,308	0,530
22000	335693,36	671386,72	1650000	2014160,16	9,65	3,4	1,1	1,5	3,412	1,202	0,389	3,412	1,202	0,389	1,202	0,530
23000	350952,15	701904,30	1725000	2105712,89	9,4	3,2	1	1,4	3,323	1,131	0,354	3,323	1,131	0,354	1,131	0,495
24000	366210,94	732421,88	1800000	2197265,63	9,1	3,1	1	1,2	3,217	1,096	0,354	3,217	1,096	0,354	1,096	0,424
25000	381469,73	762939,45	1875000	2288818,36	9	3,1	0,9	1,2	3,182	1,096	0,318	3,182	1,096	0,318	1,096	0,424
26000	396728,52	793457,03	1950000	2380371,09	8,9	3	0,8	1,1	3,147	1,061	0,283	3,147	1,061	0,283	1,061	0,389
27000	411987,30	823974,61	2025000	2471923,83	8,7	3	0,8	1	3,076	1,061	0,283	3,076	1,061	0,283	1,061	0,354
28000	427246,09	854492,19	2100000	2563476,56	8,7	3	0,8	1	3,076	1,061	0,283	3,076	1,061	0,283	1,061	0,354
29000	442504,88	885009,77	2175000	2655029,30	8,4	3	0,8	1	2,970	1,061	0,283	2,970	1,061	0,283	1,061	0,354
30000	457763,67	915527,34	2250000	2746582,03	8,3	3	0,7	1	2,934	1,061	0,247	2,934	1,061	0,247	1,061	0,354
31000	473022,46	946044,92	2325000	2838134,77	8,1	3	0,6	1	2,864	1,061	0,212	2,864	1,061	0,212	1,061	0,354
32000	488281,25	976562,50	2400000	2929687,50	8	3	0,6	1	2,828	1,061	0,212	2,828	1,061	0,212	1,061	0,354
32768	500000,00	1000000,00	2457600	3000000,00	8	3	0,8	1	2,828	1,061	0,283	2,828	1,061	0,283	1,061	0,354

Frecuencia vs Voltaje Sin Carga



La lectura de la figura 4.1.1 se la realiza del siguiente modo: En el eje de las abcisas en la primera línea cerca la palabra "frecuencia (Hz)" se leen las frecuencias generadas correspondientes cuando el reloj de referencia es de 1Mhz. La línea superior corresponde para el reloj de referencia de 2Mhz, y así correspondientemente para el reloj de 4,9152 y 6 MHz.

4.1.3 Operación del DDS para circuito con carga.

Dentro de este apartado se analizará el funcionamiento del sintetizador con carga resistiva a la salida para diferentes valores de frecuencia. Se utilizó una señal sinusoidal para la prueba, así como la salida número uno del sintetizador.

Se han realizado las pruebas con cargas resistivas de 10 KOhm y 1KOhm. Para cargas resistivas de menor valor de 1KOhm el circuito presenta distorsiones.

Para frecuencias superiores a 1MHz, la señal de salida presenta gran cantidad de ruido. En el cuadro 4.1.2 se muestran los voltajes pico- pico y la potencia eficaz obtenidos a la salida versus la frecuencia generada para una resistencia de carga de 10Kohm.

En la figura 4.1.2 además se indica la respuesta de frecuencia versus el voltaje pico-pico para cada uno de los osciladores de referencia con el que cuenta el sintetizador.

En el cuadro 4.1.3 se muestran los voltajes pico- pico y la potencia eficaz obtenidos a la salida versus la frecuencia generada para una resistencia de carga de 1KOhm.

FRECUENCIA VS VOLTAJE(RL=10 kohm) Y POTENCIA EFICAZ

FRECUENCIAS GENERADAS				RELOJ DE REFERENCIA				POTENCIA EFICAZ				
Reloj (Hz)	1000000	2000000	4915200	6000000	10000000	20000000	4915200	60000000	100000000	200000000	4915200	600000000
frec (Hz)	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344
n	frec(Hz)	frec(Hz)	frec(Hz)	frec(Hz)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	P(mW)	P(mW)	P(mW)	P(mW)
1	15	31	75	92	15	15	15	15	2,81	2,81	2,81	2,81
1000	15.259	30.518	75.000	91.553	15	14	14	14,5	2,81	2,45	2,45	2,63
2000	30.518	61.035	150.000	183.105	15	13	13	12	2,81	2,11	2,11	1,80
3000	45.776	91.553	225.000	274.658	14	12,5	10	10	2,45	1,95	1,25	1,25
4000	61.035	122.070	300.000	366.211	14	12	8	8	2,45	1,80	0,80	0,80
5000	76.294	152.588	375.000	457.764	14	12	7	7	2,45	1,80	0,61	0,61
6000	91.553	183.105	450.000	549.316	13	10	6	6	2,11	1,25	0,45	0,45
7000	106.812	213.623	525.000	640.869	13	10	5	5,5	2,11	1,25	0,31	0,38
8000	122.070	244.141	600.000	732.422	13	8	4,5	4	2,11	0,80	0,25	0,20
9000	137.329	274.658	675.000	823.975	12,5	8	4	3,6	1,95	0,80	0,20	0,16
10000	152.588	305.176	750.000	915.527	12,5	8	4	3,2	1,95	0,80	0,20	0,13
11000	167.847	335.693	825.000	1.007.080	12	8	3,4	2,8	1,80	0,80	0,14	0,10
12000	183.105	366.211	900.000	1.098.633	12	8	3	2,4	1,80	0,80	0,11	0,07
13000	198.364	396.729	975.000	1.190.186	11,5	7	2,8	2,4	1,65	0,61	0,10	0,07
14000	213.623	427.246	1.050.000	1.281.738	11	7	2,5	2	1,51	0,61	0,08	0,05
15000	228.882	457.764	1.125.000	1.373.291	10,5	6	2,5	2	1,38	0,45	0,08	0,05
16000	244.141	488.281	1.200.000	1.464.844	10,2	6	2,2	1,9	1,30	0,45	0,06	0,05
17000	259.399	518.799	1.275.000	1.556.396	10,2	6	2	1,7	1,30	0,45	0,05	0,04
18000	274.658	549.316	1.350.000	1.647.949	10	5,5	2	1,5	1,25	0,38	0,05	0,03
19000	289.917	579.834	1.425.000	1.739.502	10	5	1,8	1,5	1,25	0,31	0,04	0,03
20000	305.176	610.352	1.500.000	1.831.055	9	5	1,8	1,5	1,01	0,31	0,04	0,03
21000	320.435	640.869	1.575.000	1.922.607	9	5	1,8	1,5	1,01	0,31	0,04	0,03
22000	335.693	671.387	1.650.000	2.014.160	8,5	4,7	1,6	1,5	0,90	0,28	0,03	0,03
23000	350.952	701.904	1.725.000	2.105.713	8,5	4,7	1,5	1,5	0,90	0,28	0,03	0,03
24000	366.211	732.422	1.800.000	2.197.266	8,5	4,5	1,5	1,5	0,90	0,25	0,03	0,03
25000	381.470	762.939	1.875.000	2.288.818	8	4,5	1,4	1	0,80	0,25	0,02	0,01
26000	396.729	793.457	1.950.000	2.380.371	8	4,5	1,4	1	0,80	0,25	0,02	0,01
27000	411.987	823.975	2.025.000	2.471.924	8	4,4	1,3	1	0,80	0,24	0,02	0,01
28000	427.246	854.492	2.100.000	2.563.477	8	4,3	1,3	1	0,80	0,23	0,02	0,01
29000	442.505	885.010	2.175.000	2.655.029	8	4,3	1,3	1	0,80	0,23	0,02	0,01
30000	457.764	915.527	2.250.000	2.746.582	8	4,2	1,2	1	0,80	0,22	0,02	0,01
31000	473.022	946.045	2.325.000	2.838.135	8	4,1	1,2	1	0,80	0,21	0,02	0,01
32000	488.281	976.563	2.400.000	2.929.688	8	4,1	1,2	1	0,80	0,21	0,02	0,01
32768	500.000	1.000.000	2.457.600	3.000.000	8	4	1,2	1	0,80	0,20	0,02	0,01

Frecuencia vs Voltaje(Vpp)
Resistencia = 10 KOhm

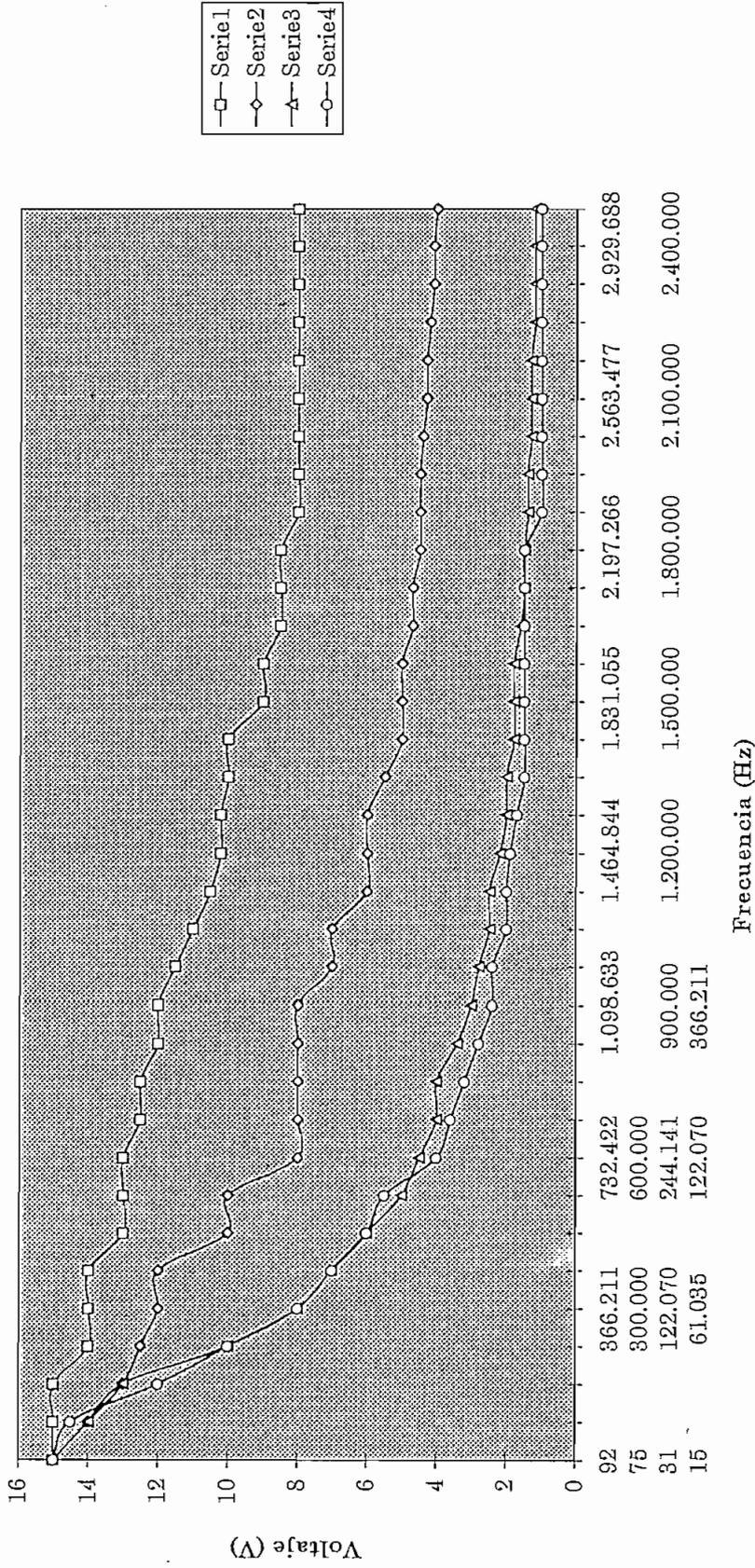


Figura 4.1.2

FRECUENCIA VS VOLTAJE Y POTENCIA EFICAZ (RL = 1kohm)

FRECUENCIAS GENERADAS										RELOJ DE REFERENCIA					RELOJ DE REFERENCIA				
Reloj (Hz)	1000000	2000000	4915200	6000000	1000000	2000000	4915200	6000000	1000000	2000000	4915200	6000000	1000000	2000000	4915200	6000000			
frec res (Hz)	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344	15,2587891	30,5175781	75	91,5527344			
n	frec(Hz)	frec(Hz)	frec(Hz)	frec(Hz)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Vpp(V)	Pef(mW)	Pef(mW)	Pef(mW)	Pef(mW)			
1	15	31	75	92	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	2,5	2,5	2,5	2,5			
1000	15.259	30.518	75.000	91.553	4,5	4,5	4,5	4,3	4,5	4,5	4,5	4,3	2,5	2,5	2,5	2,3			
2000	30.518	61.035	150.000	183.105	4,5	4,5	4,5	4,3	4,5	4,5	4,5	4,3	2,5	2,5	2,5	2,3			
3000	45.776	91.553	225.000	274.658	4,5	4,5	4,4	4,1	4,5	4,5	4,4	4,1	2,5	2,5	2,4	2,1			
4000	61.035	122.070	300.000	366.211	4,5	4,5	4,4	3,7	4,5	4,5	4,4	3,7	2,5	2,5	2,4	1,7			
5000	76.294	152.588	375.000	457.764	4,5	4,5	4,2	3,7	4,5	4,5	4,2	3,7	2,5	2,5	2,2	1,7			
6000	91.553	183.105	450.000	549.316	4,4	4,4	4	3,4	4,4	4,4	4	3,4	2,4	2,5	2,0	1,4			
7000	106.812	213.623	525.000	640.869	4,4	4,4	4	3,4	4,4	4,4	4	3,4	2,4	2,5	2,0	1,4			
8000	122.070	244.141	600.000	732.422	4,4	4,4	3,9	3,2	4,4	4,4	3,9	3,2	2,4	2,4	2,0	1,4			
9000	137.329	274.658	675.000	823.975	4,4	4,4	3,8	3	4,4	4,4	3,8	3	2,4	2,4	1,8	1,3			
10000	152.588	305.176	750.000	915.527	4,4	4,4	3,8	3	4,4	4,4	3,8	3	2,4	2,4	1,8	1,1			
11000	167.847	335.693	825.000	1.007.080	4,4	4,4	3,5	3	4,4	4,4	3,5	3	2,4	2,4	1,5	1,1			
12000	183.105	366.211	900.000	1.098.633	4,4	4,4	3,5	2,8	4,4	4,4	3,5	2,8	2,4	2,4	1,5	1,0			
13000	198.364	396.729	975.000	1.190.186	4,4	4,4	3,2	2,5	4,4	4,4	3,2	2,5	2,4	2,4	1,3	0,8			
14000	213.623	427.246	1.050.000	1.281.738	4,4	4,4	3	2,2	4,4	4,4	3	2,2	2,4	2,4	1,1	0,6			
15000	228.882	457.764	1.125.000	1.373.291	4,4	4,4	3	2,2	4,4	4,4	3	2,2	2,4	2,4	1,1	0,6			
16000	244.141	488.281	1.200.000	1.464.844	4,4	4,2	2,7	2	4,4	4,2	2,7	2	2,4	2,2	0,9	0,5			
17000	259.399	518.799	1.275.000	1.556.396	4,4	4,2	2,5	2	4,4	4,2	2,5	2	2,4	2,2	0,8	0,5			
18000	274.658	549.316	1.350.000	1.647.949	4,2	4,2	2,4	2	4,2	4,2	2,4	2	2,2	2,2	0,7	0,5			
19000	289.917	579.834	1.425.000	1.739.502	4,2	4,2	2,4	1,8	4,2	4,2	2,4	1,8	2,2	2,2	0,7	0,4			
20000	305.176	610.352	1.500.000	1.831.055	4,2	4,2	2,3	1,8	4,2	4,2	2,3	1,8	2,2	2,2	0,7	0,4			
21000	320.435	640.869	1.575.000	1.922.607	4,2	4	2,2	1,8	4,2	4	2,2	1,8	2,2	2,2	0,6	0,4			
22000	335.693	671.387	1.650.000	2.014.160	4,2	4	2,2	1,5	4,2	4	2,2	1,5	2,2	2,2	0,6	0,3			
23000	350.952	701.904	1.725.000	2.105.713	4,2	4	2	1,5	4,2	4	2	1,5	2,2	2,2	0,5	0,3			
24000	366.211	732.422	1.800.000	2.197.266	4,2	4	2	1,5	4,2	4	2	1,5	2,2	2,2	0,5	0,3			
25000	381.470	762.939	1.875.000	2.288.818	4,2	4	2	1,4	4,2	4	2	1,4	2,2	2,2	0,5	0,2			
26000	396.729	793.457	1.950.000	2.380.371	4,2	4	1,8	1,4	4,2	4	1,8	1,4	2,2	2,2	0,4	0,2			
27000	411.987	823.975	2.025.000	2.471.924	4,2	4	1,8	1,3	4,2	4	1,8	1,3	2,2	2,2	0,4	0,2			
28000	427.246	854.492	2.100.000	2.563.477	4,2	4	1,7	1,2	4,2	4	1,7	1,2	2,2	2,2	0,4	0,2			
29000	442.505	885.010	2.175.000	2.655.029	4,2	4	1,7	1,2	4,2	4	1,7	1,2	2,2	2,2	0,4	0,2			
30000	457.764	915.527	2.250.000	2.746.582	4,2	4	1,5	1,2	4,2	4	1,5	1,2	2,2	2,2	0,3	0,2			
31000	473.022	946.045	2.325.000	2.838.135	4,2	4	1,5	1,2	4,2	4	1,5	1,2	2,2	2,2	0,3	0,2			
32000	488.281	976.563	2.400.000	2.929.688	4,2	4	1,5	1,2	4,2	4	1,5	1,2	2,2	2,2	0,3	0,2			
32768	500.000	1.000.000	2.457.600	3.000.000	4,2	4	1,5	1,2	4,2	4	1,5	1,2	2,2	2,2	0,3	0,2			

Frecuencia vs Voltaje (Vpp)

Resistencia = 1KOhm

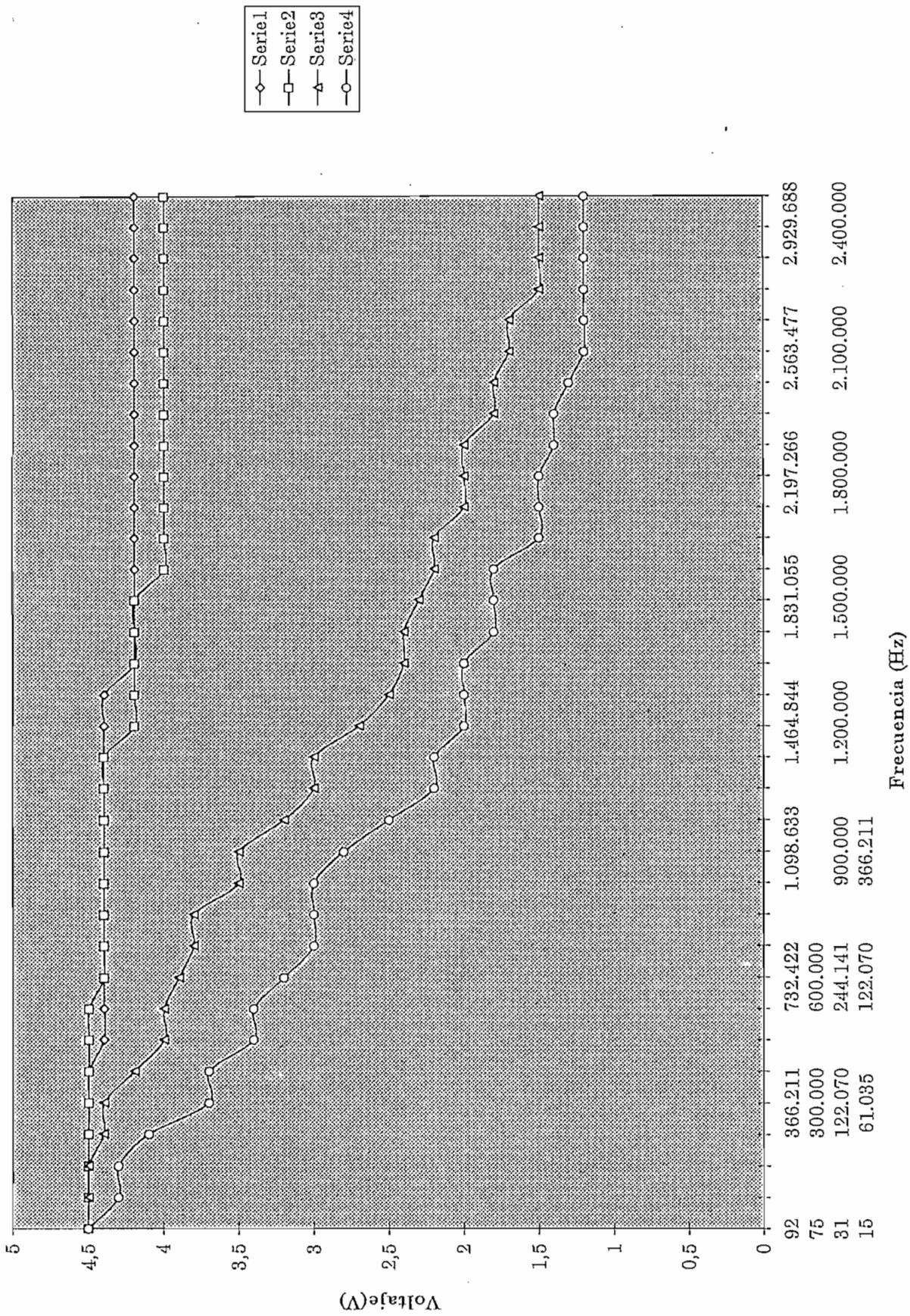


Figura 4.1.3

En la figura 4.13. además se indica la respuesta de frecuencia versus el voltaje pico-pico para cada uno de los osciladores de referencia con el que cuenta el sintetizador.

4.1.4 Medición de Frecuencia.

Para la medición de frecuencia en el rango de los 0Hz a 500KHz se utilizó un multímetro Fluke 72; presenta un error del 0,1% en la lectura. Para el rango de medición de frecuencia de 1MHz a 3Mhz se utilizó un contador de frecuencia FLUKE 1910A. Se hizo uso de la señal TTL generada por el sintetizador, debido a que es señal que da mejor respuesta a altas frecuencias. Las características técnicas de ambos equipos de medición se encuentran en el anexo 5.

En el cuadro 4.1.4 se muestra los resultados prácticos de la frecuencias generadas y las medidas experimentales. Las mediciones de frecuencia fueron tomadas en circuito abierto a condiciones de temperatura de 25 °C.

Reloj=6,000,000MHz			Reloj=4,915,200MHz			Reloj = 2,000,000MHz			Reloj = 1,000,000MHz		
SDD	FLUKE 1910A	SDD	FLUKE 1910A	SDD	FLUKE 1910A	FLUKE 72	SDD	FLUKE 1910A	FLUKE 72	SDD	FLUKE 1910A
frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)	frec (Hz)
2.999.908	3.000.207	2.457.525	2.458.129	1.000.000	979.256	475.000	499.985	475.000	475.000	499.985	500.000
2.909.362	2.909.475	2.409.825	2.409.318	915.527	300.589	475.000	488.129	475.000	475.000	488.129	488.000
2.883.655	2.883.656	2.394.000	2.394.077	885.010	154.896	475.000	469.818	475.000	475.000	469.818	470.000
2.851.226	2.851.337	2.383.350	2.383.426	871.796	444.542	475.000	450.424	475.000	475.000	450.424	451.000
2.792.999	2.798.505	2.362.200	2.362.274	837.708	111.533	475.000	427.795	475.000	475.000	427.795	428.000
2.754.180	2.754.287	2.309.250	2.309.323	808.838	458.961	475.000	404.883	475.000	475.000	404.883	404.000
2.734.863	2.734.968	2.272.125	2.281.610	762.939	245.865	475.000	392.227	475.000	475.000	392.227	392.000
2.508.544	2.508.644	2.213.925	2.218.801	732.422	214.796	475.000	354.507	475.000	475.000	354.507	355.000
2.489.135	2.491.084	2.171.550	2.171.619	701.904	323.655	475.000	340.500	475.000	475.000	340.500	341.000
2.392.181	2.392.366	2.102.700	2.128.462	671.387	201.440	475.000	301.712	475.000	475.000	301.712	302.000
2.379.272	2.379.364	2.060.325	2.105.986	640.869	214.892	475.000	267.227	475.000	475.000	267.227	267.000
2.353.363	2.353.454	2.028.525	2.060.011	549.316	124.654	475.000	231.674	475.000	475.000	231.674	232.000
2.314.544	2.314.635	1.991.475	2.041.042	518.799	478.962	475.000	205.811	475.000	475.000	205.811	206.000
2.282.226	2.282.315	1.927.875	1.927.503	457.764	458.961	457.700	185.333	457.700	457.700	185.333	185.000
2.249.908	2.250.745	1.885.500	1.887.057	427.246	245.865	427.200	139.761	427.200	427.200	139.761	139.700
2.169.616	2.193.720	1.874.925	1.879.018	396.729	214.796	396.700	108.703	396.700	396.700	108.703	108.700
2.163.024	2.172.404	1.837.875	1.874.907	366.211	323.655	366.200	93.174	366.200	366.200	93.174	93.170
2.120.635	2.148.523	1.747.800	1.806.105	335.693	201.440	335.600	77.645	335.600	335.600	77.645	77.640
2.115.875	2.129.740	1.678.950	1.762.545	305.176	214.892	305.100	69.880,5	305.100	305.100	69.880,5	69.880
2.049.499	2.097.000	1.641.900	1.664.082	274.658	458.961	274.600	62.116,0	274.600	274.600	62.116,0	66.110
2.004.272	2.004.528	1.535.925	1.537.848	244.141	245.865	244.100	45.777,0	244.100	244.100	45.777,0	44.770
1.984.863	1.984.942	1.382.325	1.384.442	213.623	214.796	213.600	38.148,0	213.600	213.600	38.148,0	38.140
1.939.636	1.940.984	1.308.225	1.308.266	91.553	323.655	91.550	30.518,2	91.550	91.550	30.518,2	30.510
1.920.227	1.920.302	1.292.325	1.292.967	61.035	201.440	61.030	22.888,5	61.030	61.030	22.888,5	22.880
1.758.544	1.758.708	1.275.000	1.275.658	45.776	201.440	45.770	15.259,0	45.770	45.770	15.259,0	15.250
1.325.408	1.327.856	1.228.800	1.228.839	38.147	214.892	38.140	7.629,5	38.140	38.140	7.629,5	7.629
1.170.227	1.193.057	1.144.050	1.287.806	14.038	458.961	14.030	3.051,8	14.030	14.030	3.051,8	3.051
1.034.454	1.039.051	773.250	828.771	4.425	245.865	4.425	1.525,9	4.425	4.425	1.525,9	1.525
452.545	201.624	683.250	941.802	305	214.796	305	152,6	305	305	152,6	153

Cuadro 4.1.4

4.2. Calidad de la Señal del DDS

4.2.1. Aplicaciones.-

Generador de señales, modulador de frecuencia, comunicaciones, etc.

Debido a la gran versatilidad del sintetizador digital, se presentan las siguientes aplicaciones:

- a) Generador de ondas sinusoidales, triangulares y cuadradas.
- b) Modulador de frecuencia.
- c) Generador de códigos de comunicaciones.
- d) Generador de señales de control para SCR.
- e) Generador de Onda TTL.
- f) Generador de fónemas.
- g) Generador de ondas especiales.

a) **Generador de ondas sinusoidales, triangulares y cuadradas.**

Una vez grabados los datos de los archivos correspondientes a las ondas sinusoidales, triangulares y cuadradas en el sintetizador digital se obtuvo la respuesta de frecuencia que se muestra en el cuadro 4.2.1.; en el que se detalla la frecuencia máxima y mínima de generación para las señales respectivas.

	Frecuencia mínima (Hz)	Frecuencia máxima (Hz)
señal sinusoidal	0	1'000.000
señal triangular	0	300.000
señal cuadrada	0	50.000

Cuadro 4.2.1 Rango de frecuencia mínima y máxima de generación.

En la figura 4.2.1 y 4.2.2 se muestran resultados experimentales obtenidos a la salida del sintetizador. En las fotografías 4.2.1 y 4.2.2 se indica la señal

sinusoidal a frecuencia máxima y la señal sinusoidal a frecuencia cercana a los 3 MHz respectivamente.

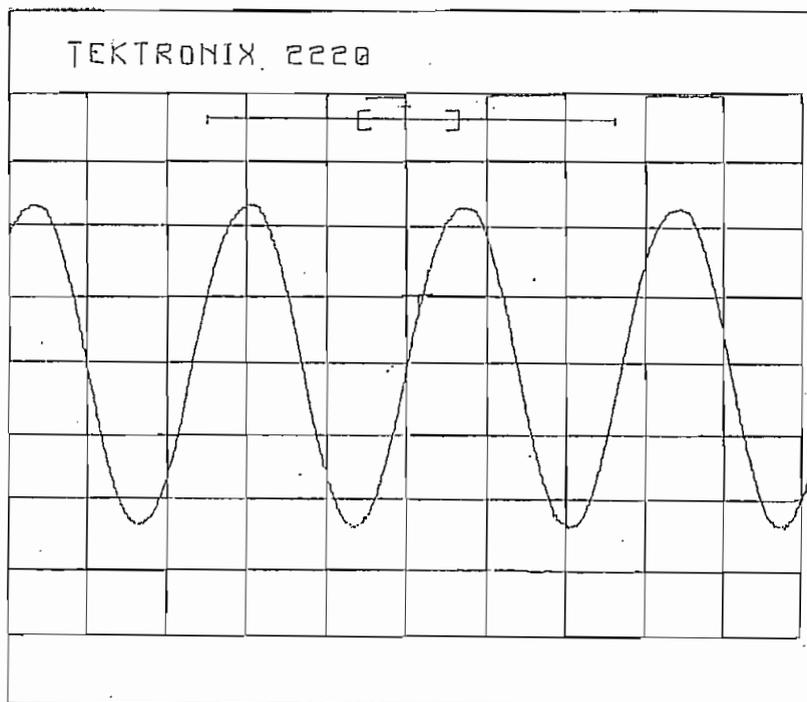


Figura 4.2.1 Frecuencia = 74250Hz/ Amplitud = 2V/div.

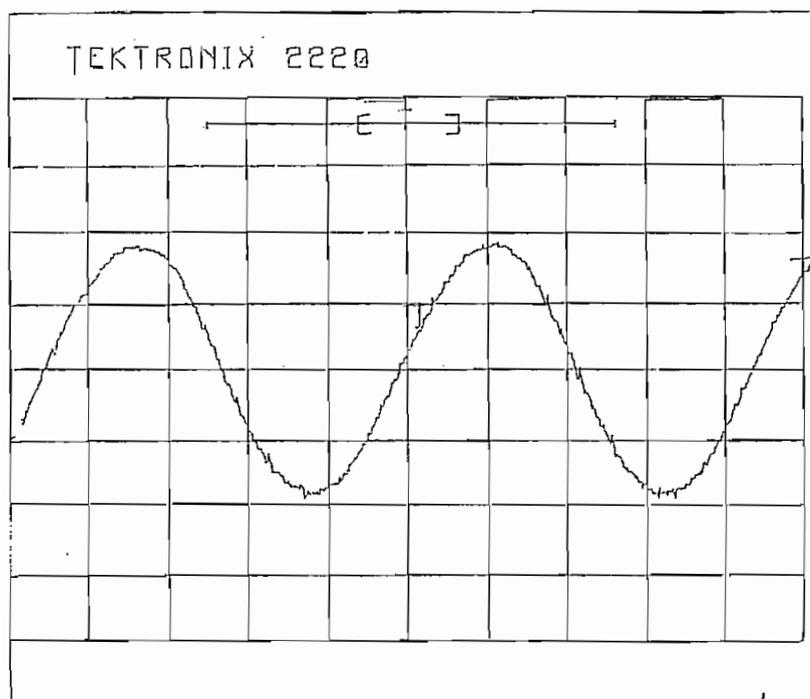
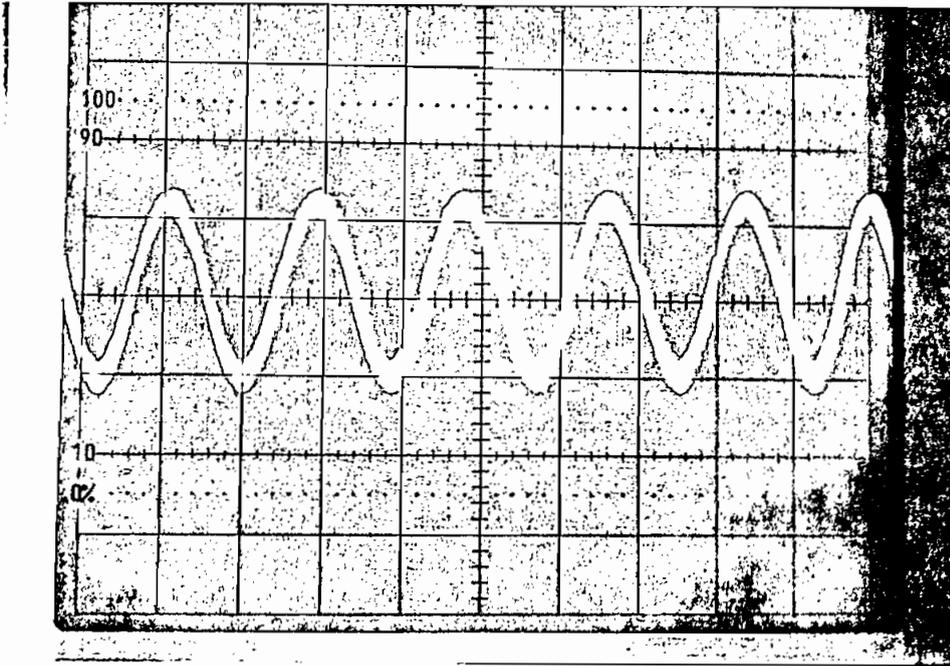
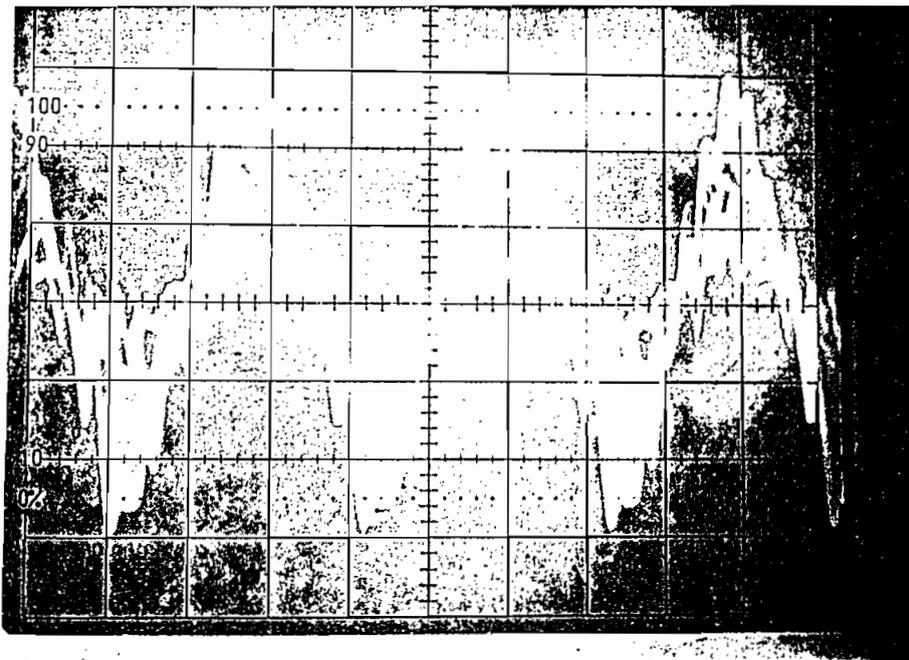


Figura 4.2.2 Frecuencia = 450150Hz / Amplitud = 2V/div.



Fotografía 4.2.1 Señal Sinusoidal: Frecuencia = 1 MHz / Amplitud = 0,5V/div.



Fotografía 4.2.2 Señal sinusoidal: Frecuencia= 3Mhz / Amplitud = 0,2V/div.

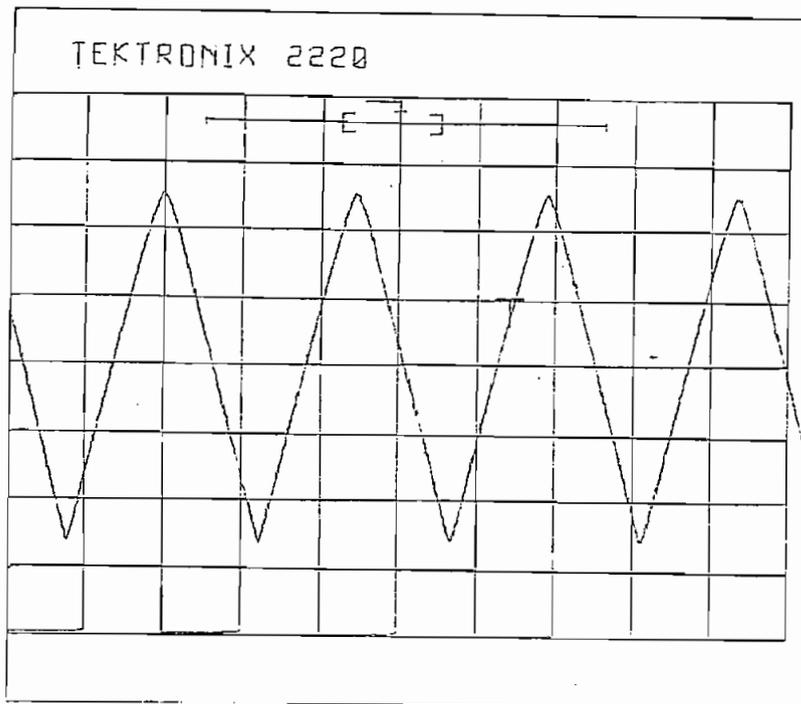
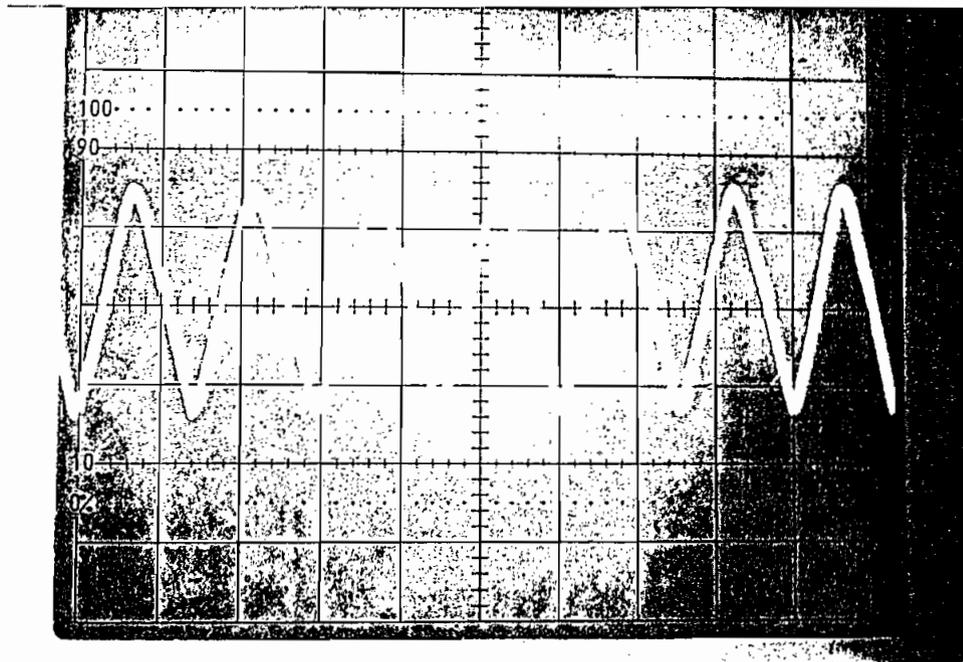


Figura 4.2.3 Señal triangular: Frecuencia = 3750 Hz / Amplitud = 2V/div.



Fotografía 4.2.3 Señal Triangular: Frecuencia= 300khz / Amplitud = 1V/div.

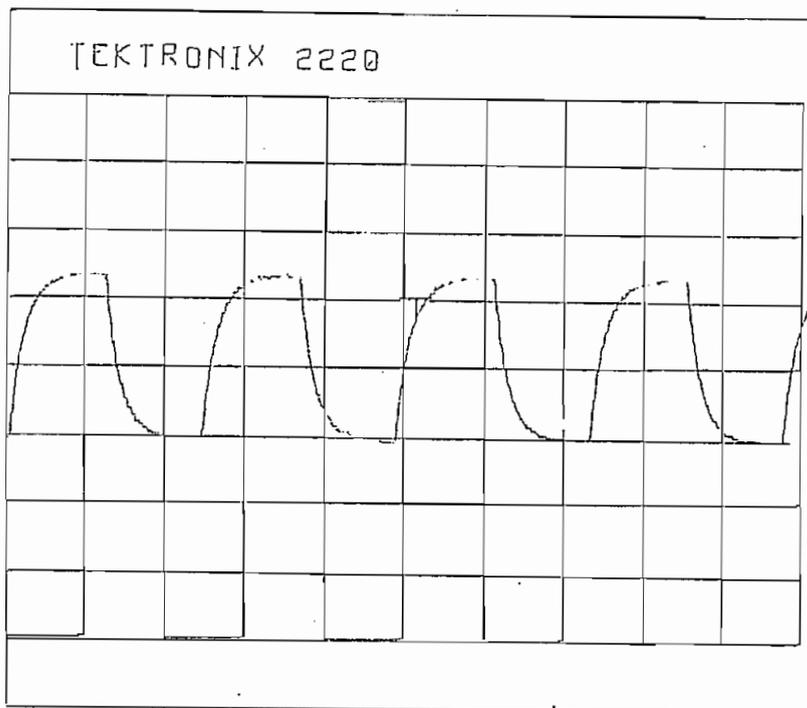


Figura 4.2.4 Señal cuadrada: Frecuencia = 51450 Hz / Amplitud = 2V/div.

b) Modulador de frecuencia.

Debido a que el control de frecuencia del sintetizador digital se realiza mediante la emisión de números (M , tal como lo define la ecuación (1.16) en el capítulo 1) en el rango de 0 a 32768, se puede modular fácilmente desde el computador mediante la emisión controlada de números. Por ejemplo:

Ejemplo 10.-

Se tiene la señal en banda base definida por la expresión:

$$y = x$$

donde $x = 0, 1, \dots, 600$.

Desarrollar la modulación en frecuencia, si la portadora es una señal sinusoidal de 300Khz. Además indicar la expresión matemática que representa la modulación y compararla con la ecuación (1.16) del capítulo 1.

Desarrollo:

Datos : frecuencia de portadora (f_0) = 300khz.

señal modulante (f_{mod}) = $f(x) = x$.

Se escoge el reloj de referencia de 4'915.200 Mhz como señal que define la frecuencia de resolución, de la ecuación (1.17) del capítulo 1 se tiene:

$$f_{res} = 75Hz$$

usando la ecuación (1.16), se encuentra el valor de M: obteniendo:

$$M = \frac{300.000}{75} = 4000$$

M = 4000, es el número que se debe mandar al acumulador directo, vía puerto paralelo para fijar la frecuencia de la portadora exactamente a 300Khz.

Se procede a realizar la modulación en frecuencia variando el valor de M en el rango especificado por la señal modulante ($f(x) = x$). La expresión que se debe escribir en programa es la que se indica a continuación.

$$f_{mod\ ulada} = f_{res} * (i) \quad \text{ecu(4.1)}$$

donde: $y = 3700, 3701, \dots, 4000, 4001, \dots, 4300$.

La ecuación (4.1) no es más que la ecuación (1.16) modificada en el término correspondiente a M, y escrita como una variable; en este caso la variable es i.

En el programa del Sintetizador Digital se ha implementado una demostración de la modulación de frecuencia, en la que la señal modulante es $f(x) = x$; se la encuentra con el nombre de Generador de Barrido.

c) Generador de códigos de comunicaciones.-

Para crear un código de comunicaciones se escribe el archivo de la forma de onda que representa el código. En anexos 4 se encuentra el listado de grabado de señal para el código Manchester y NRZ. Además se provee de ejemplos para otro tipo de códigos de comunicaciones en forma de archivo listo a ser leído.

En la figura 4.2.5 y figura 4.2.6 se muestra el código NRZ y Manchester respectivamente, para la palabra de transmisión : 0101000101010001.

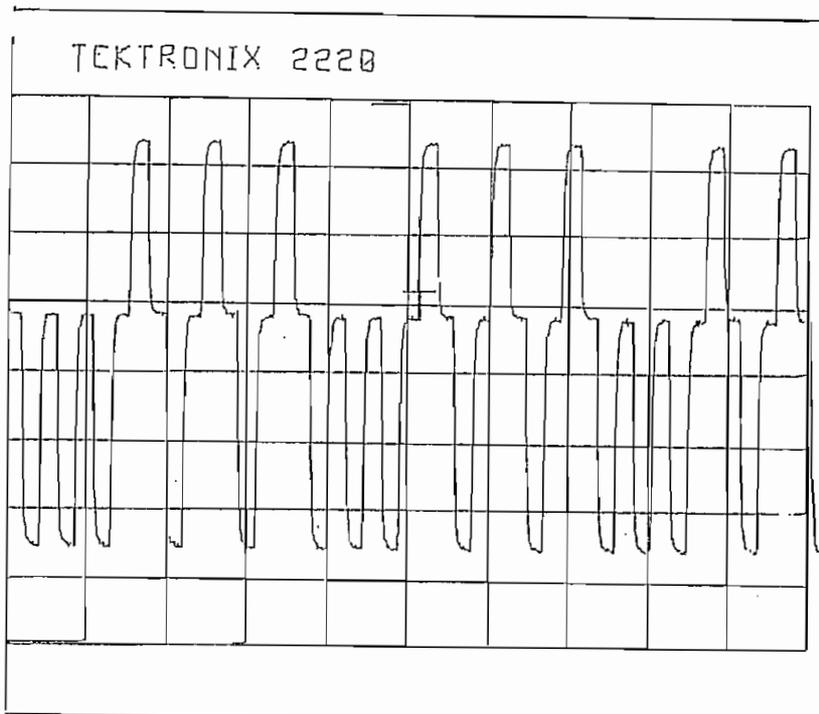


Figura 4.2.5 Código NRZ. Frecuencia = 10KHz / Amplitud = 2V/div.

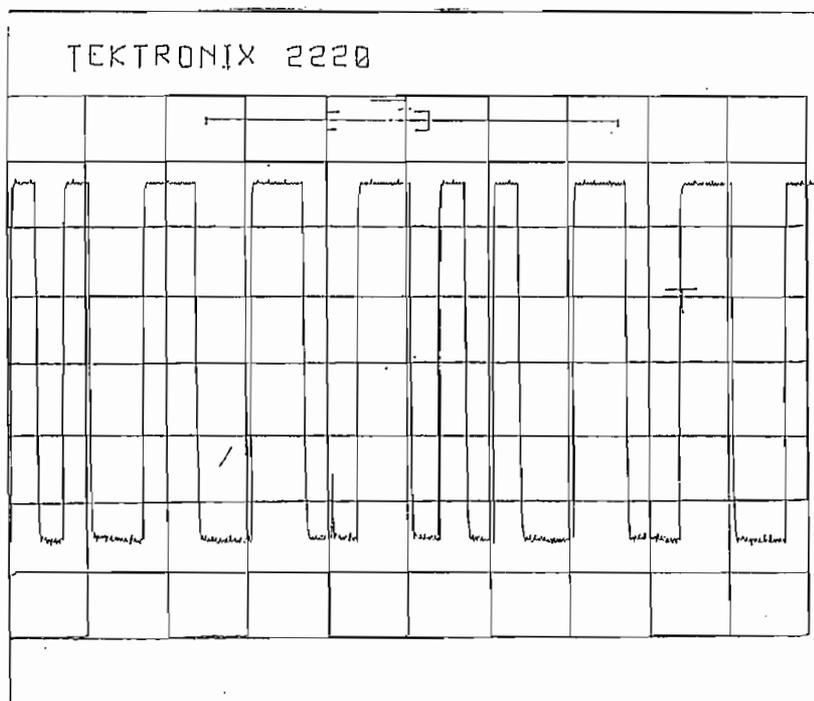


Figura 4.2.6 Código Manchester. Frecuencia = 10KHz / Amplitud = 2V/div.

d) Generador de señales de control para SCR.-

Para generar señales de control para SCR (rectificador controlado de silicio) se utiliza la mínima frecuencia de generación del sintetizador, esto es, la frecuencia de resolución para cada uno de los osciladores de referencia. Cuando el sintetizador se encuentra en esta condición direcciona todas las localidades de memoria RAM, por lo que entonces es posible grabar en una localidad específica un valor de amplitud, obteniéndose a la salida un pulso.

Si se graba datos en diversas localidades de memoria no contiguas el resultado será un tren de pulsos.

El ancho del pulso mínimo que puede generar el sintetizador está en estrecha relación con la frecuencia de resolución y la capacidad de la memoria.

Ejemplo 11:

Encontrar el ancho mínimo del pulso posible de generar usando el sintetizador digital con el reloj de referencia de 1Mhz.

Datos: Frecuencia del reloj de referencia = 1mhz.

Del cuadro 2.1.2. en el capítulo 2, se obtiene la frecuencia de resolución:

$$f_{res} = 15,25878 \text{ Hz}$$

Obteniendo el período de la frecuencia de resolución se tiene:

$$\text{Periodo } (f_{res}) = 65,536 \text{ ms.}$$

En el tiempo de 65,536 mseg se direccionan a las 8192 localidades de memoria. Se deduce entonces que el pulso mínimo tendrá una duración de :

$$\text{Pulso minimo} = \frac{65,536}{8192} = 8 \text{ useg.}$$

Se puede obtener un pulso que tenga una duración cercana a los 8 useg, si se usa el reloj de referencia de 1Mhz.

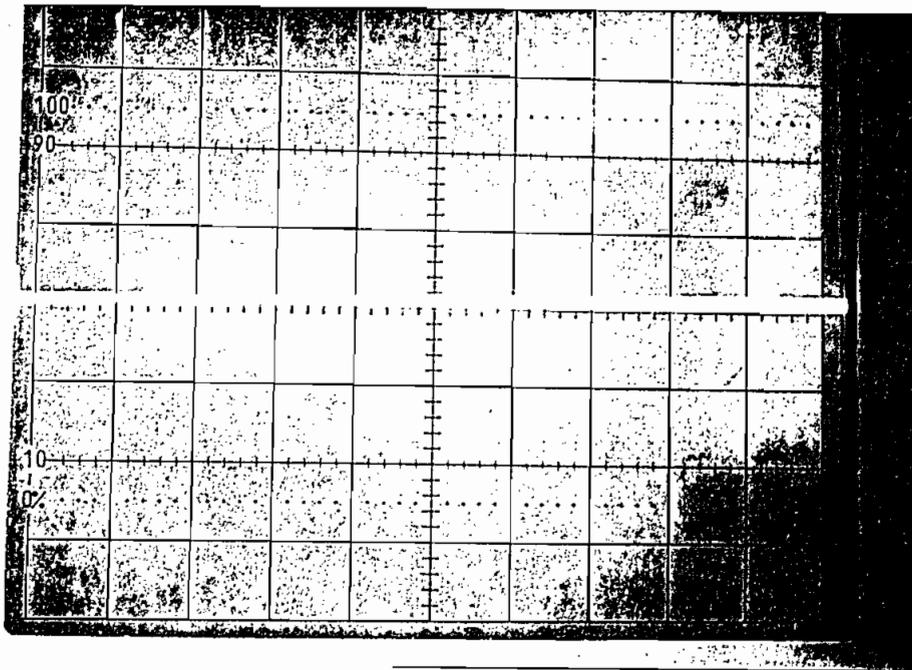
Reloj de referencia (MHz)	Tiempo del pulso mínimo (useg)
1	8
2	4
4,9152	1,6276
6	1,33

Cuadro 4.2.2 Tiempos de pulso mínimos.

Usando el ejemplo 11, se obtiene el cuadro 4.2.2 en el que se indica el tiempo que dura un pulso mínimo para los diferentes osciladores de referencia con los que cuenta el sintetizador.

Como se mencionó anteriormente, si se graba un solo dato en una localidad de memoria se reproduce a la salida un pulso de la duración indicada en el cuadro 4.2.2. Si se graban dos localidades contiguas con un valor de amplitud determinado se puede duplicar o triplicar la duración del pulso mínimo, es así que se puede tener pulsos que duren desde 1,3useg hasta pulsos que duren 65,536ms; que ese el caso de cuando se graban todas las localidades de memoria con un dato determinado.

En la fotografía 4.2.4 se muestra una aplicación práctica de un pulso que dura 1,6276useg y que se activa cuando el angulo de fase es de 30 grados. En anexo 6 se indica además el programa para grabar el pulso, de tal forma que puede esta información ser enviada al sintetizador para su posterior generación.



Fotografía 4.2.4 Tren de pulsos para disparar un SCR.

e) Generador de Onda TTL.-

De las señales de direccionamiento a memoria que genera el acumulador digital, la señal del bit más significativo corresponde a una señal TTL de la misma frecuencia que la señal que se está generando en el sintetizador. Es así que la señal del acumulador a más de definir cual es la frecuencia a la que genera el sintetizador, también se utiliza como un generador TTL. Para el equipo construido está señal alcanza valores cercanos a los 3 Mhz. En la figura 4.2.7 se indica el diagrama de bloques en el que se observa lo explicado anteriormente.

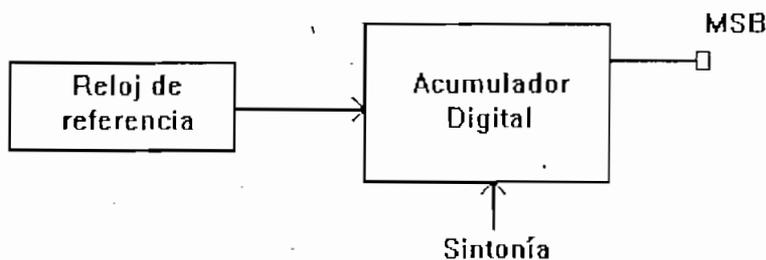


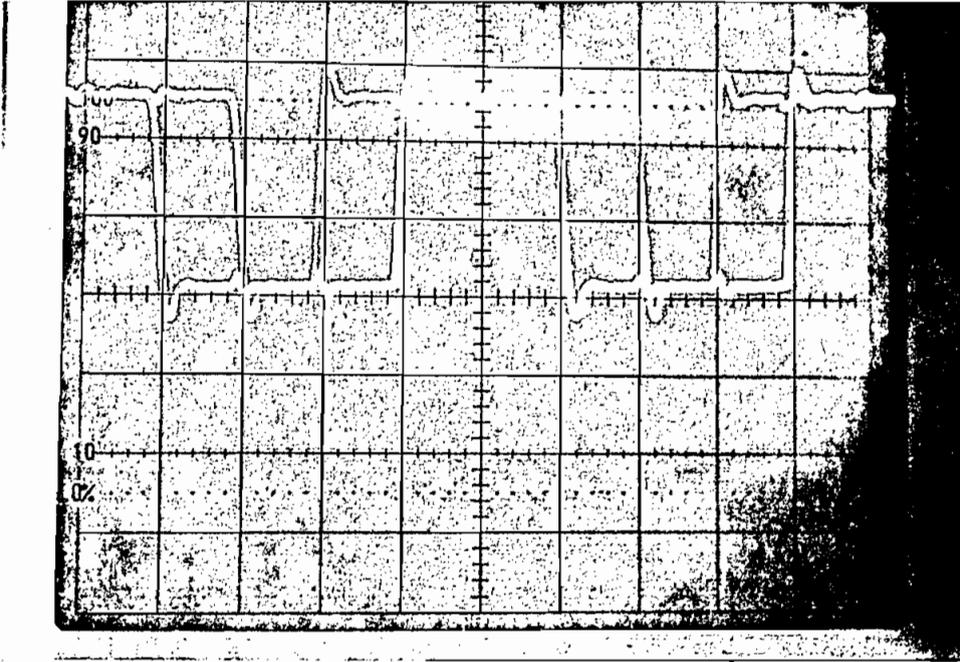
Figura 4.2.7 Generador TTL usando la DDS.

En la foto 4.2.5 y 4.2.6 se muestra las ondas TTL a frecuencias de 1Mhz y 3 Mhz respectivamente obtenidas en el sintetizador.

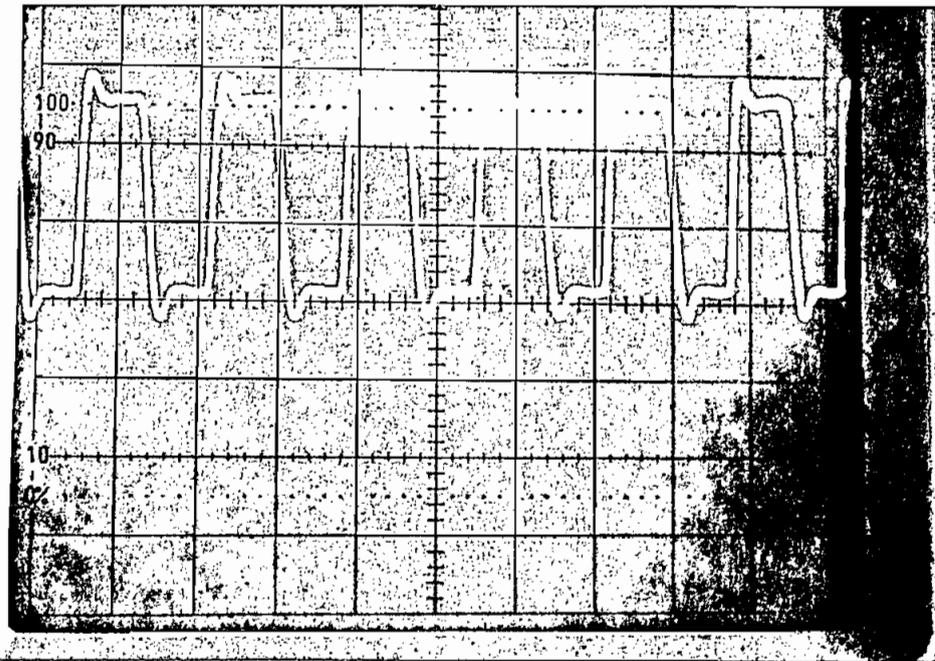
f) Generador de fónemas.-

Para realizar está aplicación se utilizaron señales gráficas de fónemas de voz que fueron proveídos por el Dr. Gualberto Hidalgo, el cual trabaja con estás aplicaciones.

Una vez obtenida está información y debido a la imposibilidad de realizar una conversión de información entre el archivo donde se encontra-



Fotografía 4.2.5 Onda TTL a 1Mhz.



Fotografía 4.2.6 Onda TTL a 3MHz.

ban estos datos y el archivo de manejo del sintetizador, se realizó una aproximación mediante señales sinusoidales. En anexos 7 se indica el archivo que fue utilizado para generar este tipo de señales.

En la figura 4.2.8 y 4.2.9 se muestran los fónemas correspondientes a la palabra IS e IN, respectivamente.

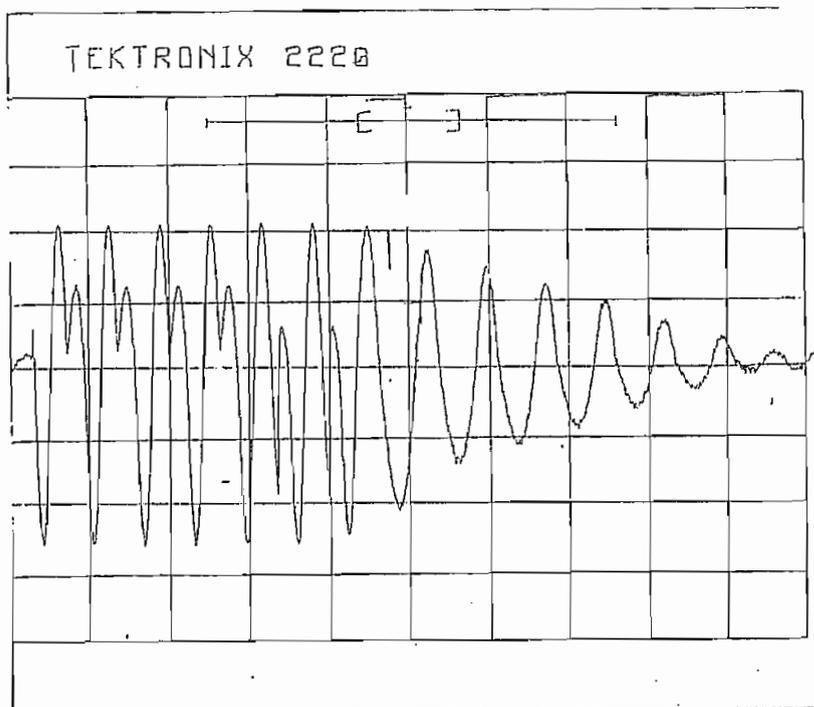


Figura 4.2.8 Fónema IS, frecuencia = 675Hz, Amplitud = 2V/div.

g) Generador de ondas especiales.

Debido a las innumerables formas de onda que el sintetizador puede generar, en este apartado se mostrará algunas señales especiales que fueron grabadas para comprobar la eficacia y versatilidad del generador de señales digital.

En la figura 4. 2.10 se muestra una señal eléctrica que en el semiciclo positivo está compuesta por una señal sinusoidal y en el semiciclo negativo tiene una señal triangular.

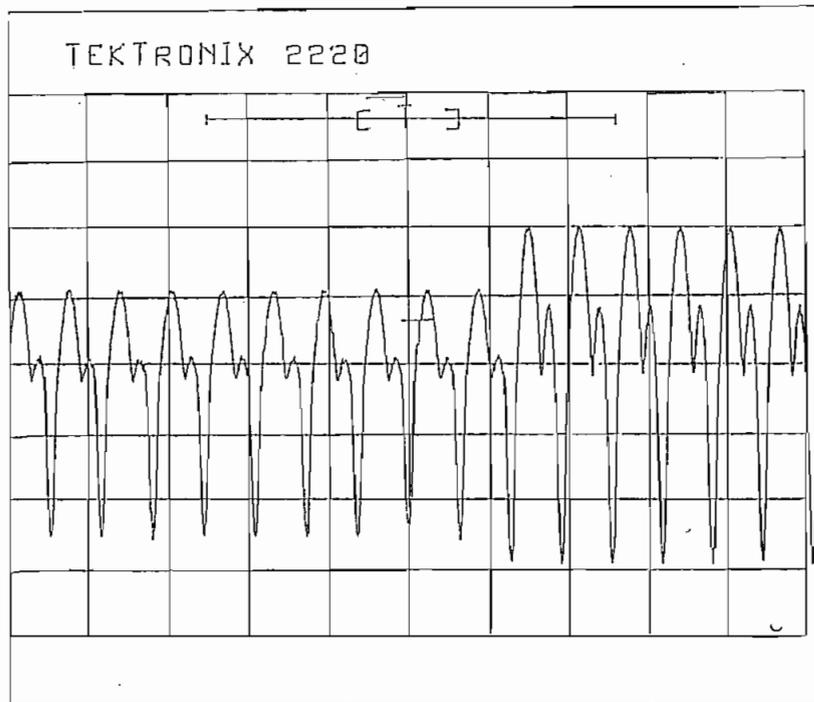


Figura 4.2.9 Fónema IN, frecuencia = 750Hz/Amplitud = 2V/div.

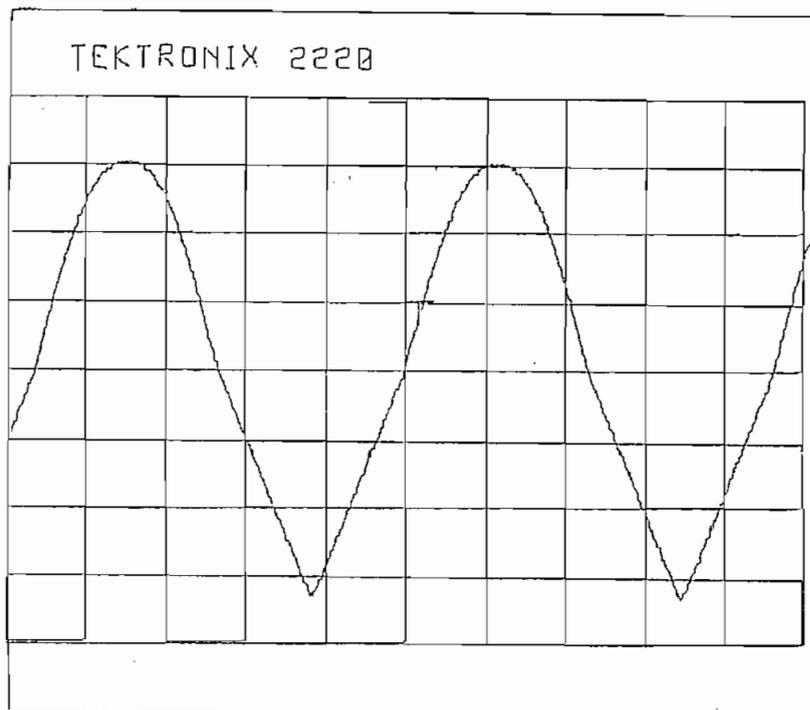


Figura 4.2.10 Onda especial: Frecuencia= 5400hz / Amplitud = 2V/div.

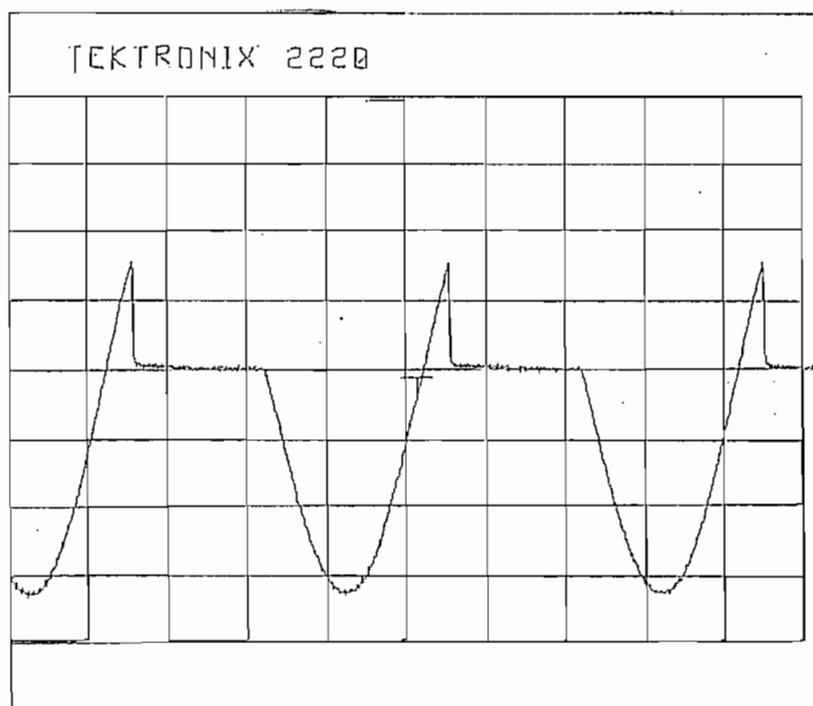


Figura 4.2.11 Onda especial, frecuencia = 3375Hz/Amplitud = 2V/div

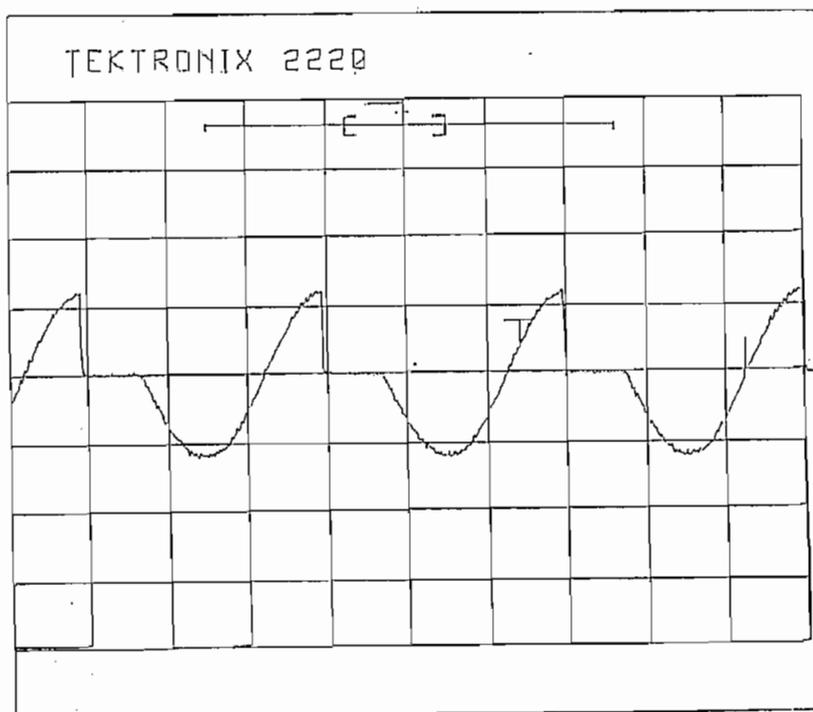


Figura 4.2.12 Onda especial: Frecuencia= 3154hz / Amplitud = 2V/div.

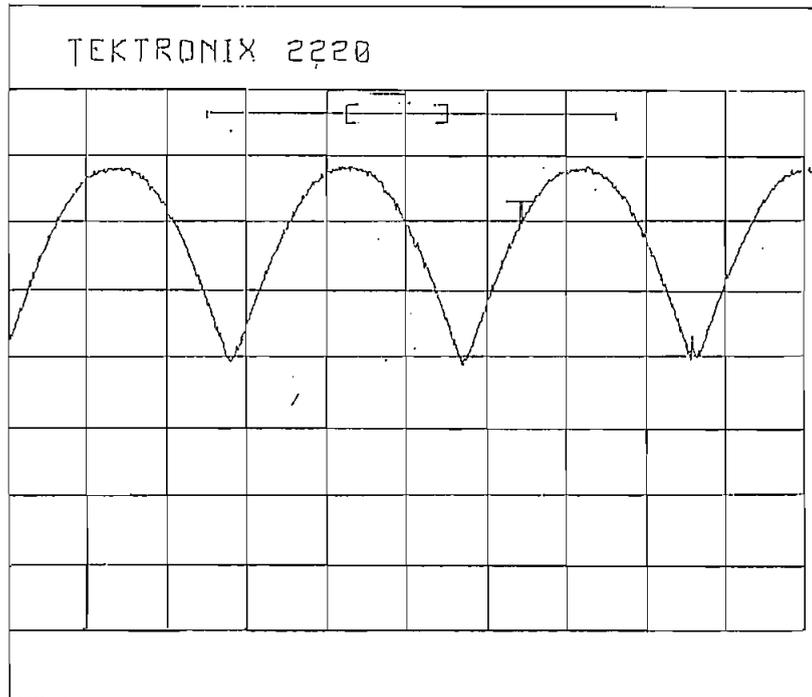


Figura 4.2.12 Onda especial, frecuencia = 3475Hz/Amplitud = 2V/div

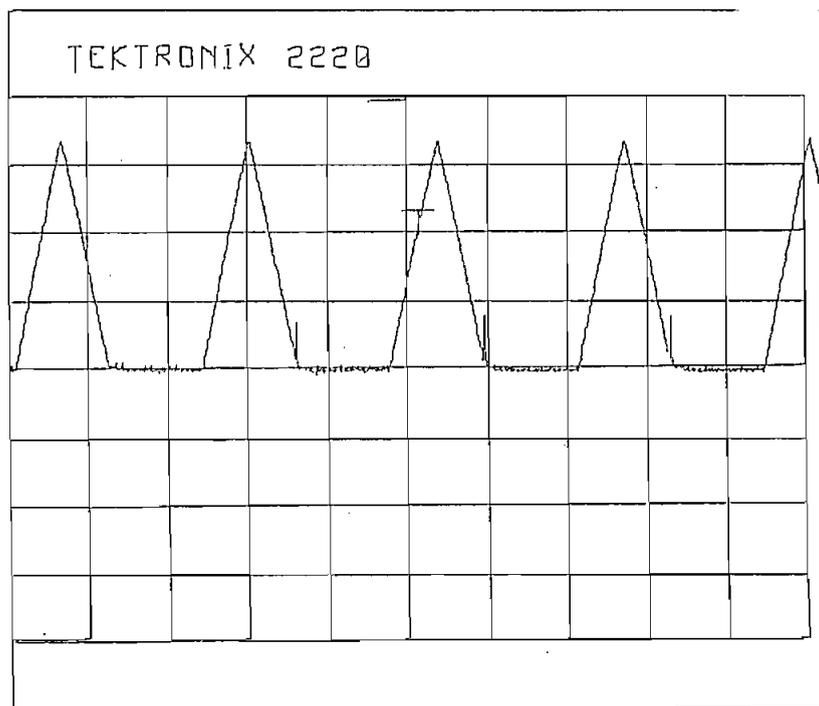
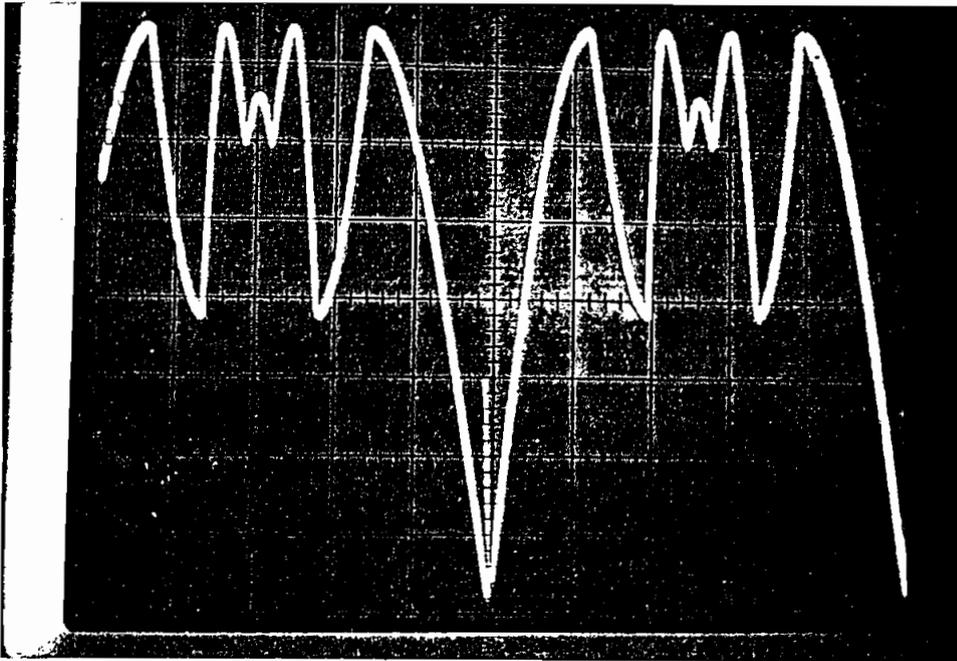


Figura 4.2.14 Onda especial: Frecuencia= 5325hz / Amplitud = 2V/div.

En la fotografía 4.2.7. se puede observar una forma de onda especial que obtenerla de otra forma es muy difícil y casi imposible. La misma corresponde al signo de Batman el cual es muy conocido.



Fotografía 2.4.7. Onda Batman, frecuencia = 3400Hz, Amplitud = 2V/div.

4.3. Análisis de Resultados

4.3.1 Respuesta de frecuencia.

Comparación entre el valor medido y generado. Error relativo.

4.3.2 Rangos de frecuencia.

Rangos de frecuencia máximos por señal; posible de generar.

4.3.3 Características técnicas del equipo.

Corriente consumida, impedancia de salida, etc. Modo de empleo, precauciones de uso.

4.3.1 Respuesta de frecuencia

De los datos obtenidos en la sección 4.1 del capítulo 4, se ha procedido a calcular el error relativo de las mediciones. Se considerará al valor medido por el contador de frecuencia (FLUKE 1910A) como el valor real y al valor generado por el sintetizador como el valor medido. Usando la ecuación (4.2) que define el error relativo, se obtienen los resultados que se muestran en el cuadro 4.3.1

$$\text{Error relativo} = \frac{\text{Valor medido} - \text{valor real}}{\text{valor real}} \times 100 \quad \text{ecu (4.2)}$$

Para el cuadro 4.3.1 se ha tomado una muestra de 16 mediciones con el reloj de referencia de 6Mhz. Como se puede observar los resultados obtenidos son muy satisfactorios, se alcanzan errores con valores cercanos a 0,00001%.

DDS	Fluke 1910A	Error Relativo
Frecuencia (Hz)	Frecuencia (Hz)	(%)
2,999,908	3,000,007	0.0100
2,909,362	2,909,475	0.0039
2,883,655	2,883,656	0.00001
2,851,226	2,851,337	0.0039
2,792,999	2,798,505	0.1967
2,754,180	2,754,287	0.0039
2,734,863	2,734,968	0.0038
2,508,544	2,508,644	0.0040
2,489,135	2,491,084	0.0782
2,392,181	2,392,366	0.0077
2,379,272	2,379,364	0.0039
2,353,363	2,353,454	0.0039
2,314,544	2,314,635	0.0039
2,282,600	2,282,315	0.0039
2,249,908	2,250,745	0.0372
2,169,616	2,193,720	1.0988

Cuadro 4.3.1 Error relativo, muestra = 16 datos.

4.3.2 Rangos de frecuencia

En el cuadro 4.3.2 se indican los rangos máximos de frecuencias por grupo de señales que el sintetizador puede generar sin que se presenten distorsión. Estos resultados son consecuencia directa de la experimentación con muchas señales de distintas formas.

Señal	Frecuencia mínima (Hz)	Frecuencia máxima (Hz)
Sinusoidal	0	1,000,000
Triangular	0	300,000
Cuadrada	0	50,000
TTL	0	3,000,000
Fónemas	0	20,000
Control de SCR	0	100
Especiales	0	20,000
Comunicaciones	0	35,000

Cuadro 4.3.2 Rangos de generación de frecuencia.

4.3.3 Características técnicas del equipo

- La corriente consumida en forma total es de 400mA.
- El voltaje máximo de salida es de 24Vpp a frecuencias en el orden de 0Hz a 50khz.
- La mínima carga a conectar a la salida es de 1Kohm, se puede conectar cargas de menor valor, pero por seguridad del equipo se recomienda no pasar este límite.
- Se pueden generar señales TTL en el rango de 0Hz a 3Mhz.
- Utiliza tres pilas de Niquel-Cadmio de 1,8V cada una, de 700mA/hora.

- Cuando todas las salidas están operando tiene un consumo de potencia de 44W.

Modo de empleo

La forma de obtener la señal requerida a la salida del sintetizador se la considera siguiendo los pasos descritos a continuación:

1. Colocar el equipo horizontalmente, y conectar el cable de alimentación de 110V/60Hz.
2. Conectar el puerto de comunicación paralelo entre el sintetizador y el computador.
3. Antes de prender el sintetizador, asegurarse de las siguientes condiciones :
 - a) El conmutador de batería para largo tiempo de uso debe estar en On.
 - b) El conmutador de batería que se encuentra en la parte frontal del sintetizador debe estar en Off.
 - c) El conmutador del selector del reloj de referencia debe estar en oscilador externo.
4. Encender el computador, se ingresa al programa de operación del equipo, para proceder a la transmisión de datos.
5. En primer término aparece la pantalla de presentación del trabajo, luego se puede continuar con el programa. En la pantalla se tiene todas las opciones para abrir, leer, grabar un archivo de datos del equipo.

6. Encender el equipo.
7. Escoger la opción de grabar en el programa, luego aparecerá una pantalla de navegación de archivos. Una vez que se ha escogido la señal a grabar presionando OK, la transmisión de datos empieza. Al mismo tiempo se muestra en pantalla la forma de onda que se está grabando.
8. Cuando se ha termina de grabar los datos, presionar la tecla OK que se muestra en pantalla, de no hacerlo se pueden generar errores.
9. Escoger la opción de control de frecuencia
10. Aparece una opción en la que primero se debe escribir con cual reloj de referencia se trabajara.
11. Presionar la tecla Actualizar.
12. Escoger en el sintetizador la frecuencia que se ha seleccionado en el paso 10.
13. En la pantalla de control de frecuencia usando el dispositivo que se encuentra encima de los botones actualizar y salir se controla la frecuencia a generar por el sintetizador.
14. Si se desea cambiar la frecuencia del reloj de oscilación, cambiar en pantalla el valor escogido, actualizar y cambiar en el sintetizador a la frecuencia del reloj de oscilación.

15. Para grabar otra forma de onda, la perrilla correspondiente al reloj de oscilación debe estar en la posición de **oscilador externo**. Se continua con el paso 7.
16. Una vez efectuadas las operaciones indicadas, se puede salir de la ejecución del programa, con ayuda de la opción Salir, mostrada en pantalla.

Precauciones de uso

En el manejo del sintetizador digital, deben observarse ciertas precauciones técnicas y de uso, para evitar daños y averías, el objetivo de estas precauciones es prolongar la vida útil del equipo, así como para la protección personal del que lo utiliza.

Las normas que se exponen deben estudiarse con detenimiento, cuyo objetivo también es el óptimo uso del equipo:

1. El personal que vaya a operar el equipo deberá tener un adiestramiento técnico sobre el funcionamiento del equipo, de tal manera que no haya errores y daños en el equipo.
2. El equipo por ser portátil está alimentado por baterías, las cuales deben ser recargadas o cambiadas, si se detecta un funcionamiento erróneo.
3. Cuando se realiza la transmisión de datos asegurarse que la posición del conmutador del reloj de referencia este en Oscilador Externo, de no hacerlo se pueden causar daños irreparables.

4. No exceder el rango de carga especificado en las características técnicas del equipo.
5. Evitar cortocircuitos a las salidas del sintetizador.
6. Las perrillas de control de amplitud correspondientes a salida 1 y 3, nunca deben estar en posición cero, ya que esto causa daño a los circuitos de salida. No se debe manipular constantemente estos controles.
7. El oscilador externo TTL que se desea conectar debe tener una frecuencia máxima de hasta 8Mhz. No exceder este rango.
8. Evite los golpes y las vibraciones mecánicas para preservar de averías al equipo.
9. Se deberá asegurar que, una vez terminado el trabajo, el equipo este en la posición OFF, manteniendo de esta forma la energía de las baterías.

De esta manera, si se cumple a cabalidad las precauciones básicas mencionadas se podrá mantener y preservar en la mejor condición posible el sintetizador digital.

- **Accesorios del Sintetizador Digital**

1. Un computador portátil con su respectivo puerto paralelo. El computador debe tener el sistema operativo Windows.

2. Un cable paralelo para comunicación, el cual se incluye en el equipo.
3. Cable de alimentación 110V/60Hz.
4. Conector BNC-BNC.

4.4 Conclusiones y Recomendaciones

4.4.1 Conclusiones

En base a los resultados obtenidos en pruebas experimentales realizadas con el equipo construido, se concluye con los siguientes aspectos:

• Conclusiones Generales

- A. Uno de los objetivos del presente trabajo fue dotar de un equipo práctico y manejable para la utilización en Laboratorios de la Facultad de Ingeniería Eléctrica, el cual fue completado en su totalidad.
- B. Se expuso una nueva técnica en el procesamiento digital de señales.
- C. El rango de frecuencia máxima de 3 MHz para señal sinusoidal no pudo ser alcanzado. Entre las principales razones se concluyen las siguientes:
 - a) El convertor utilizado no es el adecuado para la técnica de la síntesis digital directa, ya que la respuesta Voltaje-Frecuencia limita las posibilidades de trabajo con el mismo.
 - b) El número de muestras a altas frecuencias es pequeño, mejorar la señal implica la utilización de un mayor número de muestras, es decir la memoria ya no será de 8Kbytes de capacidad, sino que deberá tener una capacidad mayor.

- c) Se probó la construcción de filtros activos y pasivos (pasabanda y pasabajos), pero la señal no pudo ser mejorada.
- D. La impedancia de salida del equipo por seguridad se la fijo en $1\text{K}\Omega$. No pude ser alcanzado el valor fijado de $50\ \Omega$, debido a que el conversor digital/análogo tiene una respuesta Voltaje-Frecuencia variable, es decir para diferentes cargas resistivas a la salida; la curva de respuesta (voltaje-frecuencia) obedece a una característica no constante. Esto se puede observar claramente en las figuras 4.1.1, 4.1.2 y 4.1.3.
- E. En cuanto al parámetro de la potencia de salida fijada en $0,5\ \text{mW}$ en los objetivos propuestos; se concluye que a la salida del sintetizador se obtiene una respuesta Potencia vs Frecuencia, tal como se observa en el cuadro 4.1.3.

Del mismo se puede indicar que se obtienen potencias cercanas al rango de $0,2\text{mW}$ - $4,5\text{mW}$. Es importante notar que este valor de potencia a la salida corresponde cuando la carga conectada al sintetizador es de $1\text{K}\Omega$.

- F. Cuando se estaban realizando las pruebas de transmisión de información del computador al sintetizador utilizando el microcontrolador, se observó que el tiempo de transmisión de los datos de la forma de onda se demoraban aproximadamente 5 minutos, generándose muchos errores. La función del microcontrolador era la de recibir los datos que mandaba el computador, procesarlos y escribirlos en memoria RAM. Es decir estaba cumpliendo la función de un puente entre el computador y el sintetizador. Para optimizar el tiempo de transmisión se probó la utilización del puerto paralelo del computador; el tiempo de transmisión como es lógico disminuyó (en computadores de 100MHz el tiempo de transmisión es de

30 segundos). Se concluyó entonces que el microcontrolador estaba realizando una función que podría ser ejecutada directamente desde el computador; por lo cual se consideró que la mejor opción sería la de prescindir del microcontrolador y utilizar el pòrtico paralelo del computador.

G. Conforme se iban construyendo las etapas constitutivas del sintetizador se encontraban muchos parámetros que en el diseño no fueron tomados en cuenta. Es una técnica nueva y que está en su etapa de desarrollo. Todas estas variables son mencionados en las recomendaciones para futuras referencias.

H. Es preciso indicar que el presente trabajo participó a nombre de la Escuela Politécnica Nacional, en el concurso Nacional "Generación 2000", para inventores, donde se obtuvo el Tercer Lugar, ganando el premio FUNDACYT. Este evento se realizó del 13 al 21 de Abril de 1996.

- **Utilización del equipo**

A. El equipo fué construido con el propósito de ser utilizado en el laboratorio, donde se determinó la necesidad de construir prótotipos sencillos y económicos.

B. El Sintetizador Digital es un equipo de multipropósito. En un inicio el objetivo fue construir un generador de señales. Realizando pruebas experimentales se lo pudo utilizar como un modulador de frecuencia, generador de fónemas, etc.

- C. El equipo tiene la facilidad de almacenar los datos de la señal previamente grabada, el mismo que puede ser trasladado y se lo opera en cualquier condición ambiental existente.

- D. Si se desean mejorar los resultados obtenidos en el rango de frecuencias que puede generar el sintetizador se deberá considerar en lo posible cambiar el conversor digital/análogo a la salida. Esto no presenta ningún problema debido a que el diseño tomo en consideración futuros cambios y optimizaciones del circuito.

- E. Se comprobó por primera vez la efectividad de la técnica de la Síntesis Digital Directa como herramienta para la generación de señales de formas muy variadas.

- F. Es posible implementar muchas otras aplicaciones, con sólo modificar el programa de control del sintetizador.

- **Construcción del Equipo**

Al referirme a la construcción del equipo es preciso mencionar las dificultades por las que se atravesó en el transcurso de su realización. La ausencia de diversos dispositivos y materiales en el mercado nacional, la falta de equipo necesario para las pruebas de laboratorio significó una gran inversión de tiempo, económica, pero sin embargo el interés puesto en mi persona permitieron culminar el equipo, que ya es una realidad.

- A. El Sintetizador Digital esta compuesto por 4 tarjetas importantes que son:

- Tarjeta de control, en la cual se tiene los circuitos de control y la memoria RAM. Está tarjeta es de doble lado, construida en el país.
- Tarjeta del Acumulador digital, en la cual se tiene los circuitos de selección de frecuencia y generación de datos.
- Tarjeta de Salida, en la cual se tiene los circuitos de manejo de la señal a la salida.
- Tarjeta de osciladores, en la cual están los osciladores de referencia.

4.4.2 Recomendaciones.

Como consecuencia y fruto del presente trabajo surgen inquietudes y sugerencias, las mismas que presentamos a continuación:

- A. La realización de otro trabajo que optimice el presente, queda como sugerencia para próximos trabajos de tesis, los puntos adicionales que debería cumplir otra versión del Sintetizador Digital son los siguientes:
- Optimizar el programa de control, permitiendo que el usuario pueda dibujar la forma de onda usando herramientas gráficas. Con esto se lograría una versatilidad superior al obtenido.

- Se recomienda realizar un estudio completo y profundo de filtros activos, que pueden mejorar la señal obtenida. Es necesario contar con un Analizador de Espectros en el rango de frecuencia de 1MHz a 3MHz si es que se desea realizar un buen trabajo.
- Es posible generar fónemas de voz, pero el trabajo que se realizará deberá ser ampliado en su totalidad.
- Diseñar nuevas aplicaciones con el equipo.

B. La señal obtenida puede ser mejorada si es que se cambia el conversor Digital/Análogo a la salida. Se recomienda en una futura aplicación usar un conversor de mejores características al utilizado. Debido a que es una técnica nueva y en procesos de desarrollo el acceso a circuitos integrados de muy alta calidad todavía es restringida. Hoy en día existen empresas que venden prototipos de conversores adaptados especialmente para la técnica de síntesis digital. Entre los conversores diseñados para este fin se tienen: DAC 812 (Burr Brown), BT104 (Brooktree), DAC330 (Datel), TDC1018 (TRW-LSI).

C. Se recomienda incluir en el Pensum de estudio de la Facultad el tema de la Síntesis Digital es importante; debido a que es un método de reciente creación y permite aplicaciones muy prácticas.

D. Si se construye un nuevo prototipo se recomienda lo siguiente:

- Todos los circuitos integrados deben estar colocados en una sola placa. Usar una placa de doble lado con unión metálica entre ambos lados.

- Si se utilizan zócalos, estos deberán tener sus terminales lo más pequeñas posibles.
- La señal del reloj de referencia en la placa a construirse deberá contar con apantallamiento en todo su recorrido por la misma.

Esto se debe a que se pueden presentar problemas de radiación de energía.

D. Se recomienda tomar muy en cuenta las siguientes precauciones:

- Evitar en lo posible cortocircuitar los terminales de salida del sintetizador.
- El equipo fue diseñado para cargas resistivas. Evitar utilizarlo para cargas capacitivas e inductivas de muy bajo valor.
- La impedancia de salida fue fijada en $1\text{K}\Omega$; para que el equipo tenga una vida útil mayor, no sobrepasar este límite.
- En cuanto al control de ajuste de las salidas uno y tres, estos no deben ser manipulados constantemente. Una vez que se fija el ajuste para conseguir estabilidad y mayor ganancia; evitar mover el mismo constantemente.

REFERENCIAS BIBLIOGRAFICAS

- MITCHELL F.H., Introduction to Electronics Desing, Ed. Prentice Hall , 1988. Pags: 548-574.
- HIDALGO P., “ Comunicación Digital”, Quito, Escuela Politécnica Nacional, 1986, Pags: 13-15.
- McCUNE, “Create Signal with DDS”, Electronics Design News, Marzo 1991, Pags: 95- 107.
- McCUNE, “Data modulation using direct digital synthesis”, Proceedings of de RF technology expo, 1990, pag: 413.
- McCUNE, “Numeric synthesis of square waves”, Digital RF Solutinos Corp, AN1008.
- MICROSOFT, “Microsoft Visual Basic 3.0”, 1993.
- VELARDE J.”Microcontroladores”, Quito, Superintendencia de Telccomunicaciones del Ecuador, INCAITEL, Pag: 30-80.
- TEXAS INSTRUMENTS, “Linears Circuits”, 1989.Pag 2-181.
- ELECTRONICS NOW, “PC-Based Function Generator”, Marzo 1994, Pag: 35-44.
- SANCHEZ T. ,Electrónica Básica, Escuela Politécnica Nacional, 1994.

ANEXO No. 1

*MANUAL DE REFERENCIA
VISUAL BASIC 3.0*

1.- Introducción

Breve descripción de la programación en objetos.

2.- Iniciando Visual Basic 3.0

Definición de la estructura de programación.

3.- Primeros Pasos

Explicación del entorno de trabajo.

4.- Propiedades, Eventos y Métodos

Definición de parámetros.

1. INTRODUCCIÓN.

Visual Basic es una herramienta de programación bajo el ambiente Windows que permite crear aplicaciones, en la que exista un interfase¹ gráfico con el usuario. Los gráficos de la aplicación se diseñan mediante creación de objetos², es decir se crean controles gráficos que el usuario pueda acceder o utilizar fácilmente. Una vez definido el objeto se asocian una serie de propiedades, como por ejemplo su tamaño, apariencia, etc.

Definido el objeto en conjunto se procede a escribir procedimientos en un código de programación, el que responderá a los requerimientos que el usuario haya definido.

Trabajar en el ambiente Windows implica tener acceso a funciones como :

MDI: (multiple document interface): Se refiere a trabajar con varias aplicaciones al mismo tiempo.

OLE: (Object linking and embedding): Enlace dinámico de objetos, es decir que variables de una aplicación puedan ser reconocidas por otra aplicación distinta.

DDE: (Dynamic Data Exchanged): Cuando se alteran el valor de una variable en una aplicación que ha sido relacionada con otra previamente, el valor de la variable relacionada también es alterada automáticamente en la otra aplicación.

2. INICIANDO VISUAL BASIC

¹ **Interfase.**- Medio que permite la comunicación en tres ambientes.

² **Objeto.**- Todo evento a ser procesado.

Los tres pasos principales para crear una aplicación Windows en Visual Basic son:

- a) Crear el interfase.
- b) Definir propiedades.
- c) Escribir el código.
- d)
- a) Creación de la interfaz.

El primer paso para desarrollar una aplicación en Visual Basic es planear la interface con el usuario, es decir que es lo que el usuario va a mirar en el monitor, y cuales son las acciones a las que el programa deberá responder. Esta primera fase es de absoluta y pura creatividad del programador y se llama fase de diseño.

- b) Propiedades.

Un segundo paso es establecer las *propiedades* de todos y cada uno de los objetos colocados en su interface.

- c) Código.

A este nivel interviene la programación donde se escriben los códigos para cada uno de los *eventos* o *sucesos* que Visual Basic puede reconocer.

Los programas en los lenguajes convencionales se ejecutan secuencialmente, la ejecución comienza en la primera línea y se desplaza de acuerdo al flujo del programa. Un programa en Visual Basic trabaja de un modo totalmente diferente, es un conjunto de diferentes partes de código que son activadas por los sucesos que se le han indicado que reconozcan. Es decir el control lo tiene el usuario, el código responder a un determinado

suceso; como el click del ratón, la presión de un tecla, el desbordamiento de un tiempo, etc.

Al ejecutar la aplicación, Visual Basic realiza lo siguiente.

1. Supervisa las acciones que el usuario pueda ejecutar sobre la aplicación y sus diferentes controles.
2. Cuando reconoce un suceso, ejecuta el procedimiento, que ha sido escrito previamente.

3. PRIMEROS PASOS EN VISUAL BASIC.

Cuando se inicia Visual Basic automáticamente crea un nuevo grupo de hojas de trabajo (forma o formulario) y añade funciones dentro de la ventana Windows, tal como se muestra en la figura 1:

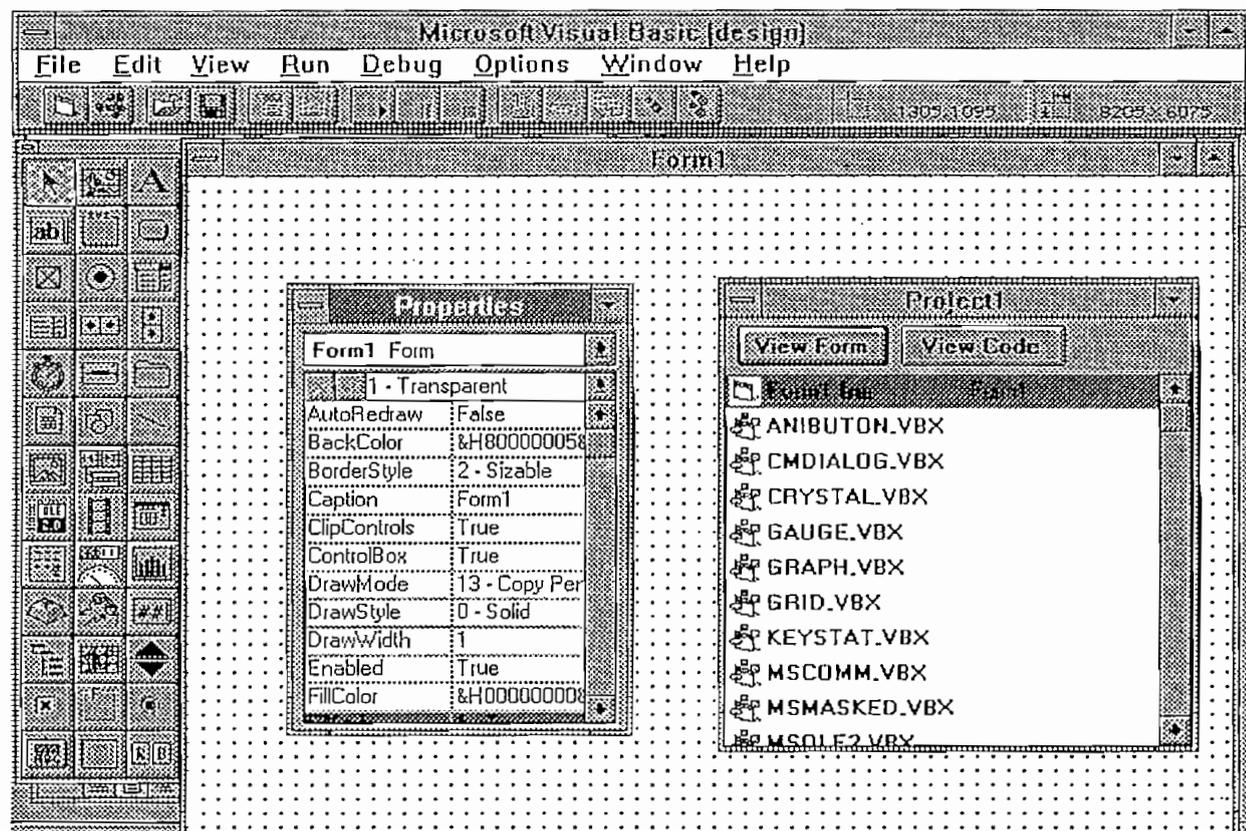


Figura 1. Pantalla principal de Visual Basic

donde,

- Barra de Menú



Permite al usuario acceder a las funciones que posee Visual Basic. Muestra los comandos que se usan para construir una aplicación.

- Barra de Herramientas.



La barra de herramientas está debajo de la barra del menú y sirve para utilizar las funciones más comunes de la barra de menús. En la barra de herramientas se tiene los siguientes iconos³ de izquierda a derecha.

- Nuevo Formulario.
- Nuevo módulo.
- Abrir Proyecto.
- Archivar Proyecto.
- Diseño de menú.
- Ventana de propiedades
- Ejecutar
- Pausa
- Fin
- Punto de pausa .- Detener ejecución del programa.
- Observación Instantánea .- Observar código ejecutado.
- Orden de llamada .- Modificar código.
- Ejecución paso a pasó .- Ejecutar una línea de código a la vez
- Paso de procedimiento.

- Forma

Sirve como una ventana para construir un interfase de la aplicación. Se puede añadir gráficos, controles, imágenes sobre la forma.

³ **Icono.**- Presentación gráfica de un control.

- Ventana de proyecto

Lista las diferentes formas, módulos de códigos y controles que se encuentran sobre el proyecto. Un proyecto es la colección de archivos que se usan para construir una aplicación.

- Propiedades de la Ventana

Muestra una lista de propiedades para la forma selecciona o control.

- Caja de herramientas

Provee un conjunto de herramientas que se usan en el diseño colocando controles sobre una forma. En la cual debido al especial interés de los mismos se describirán brevemente las funciones de los botones que fueron utilizados en el presente trabajo, tal como se observa en el cuadro 1

Referencia	Gráfico	Botón	Descripción
1		Text box	Se usa para poner texto, que el usuario puede ingresar o cambiar en cualquier momento
2		Botón de comando	Activa una acción cuando se escoge.
3		Barra desplazamiento horizontal	Permite al usuario seleccionar un valor de un rango de valores.
4		Línea	Añade un segmento de línea sobre una forma
5		Caja de imágenes	Se usa para desplegar imágenes gráficas
6		Dialogo común	Usado para desplegar una lista de directorios en forma de árbol, en el sistema.
7		Control de mascara	Provee un control para el ingreso de datos, sean estos números o letras
8		Frame	Usado para crear un grupo de controles gráficos o funcionales
9		Control de medida	Muestra gráficamente valores medidos.
10		Label	Se usa para poner texto que no es manipulable por el usuario
11		Puntero	Provee una forma para mover, cambiar el tamaño de las formas y controles

Cuadro 1. Botones de control.

Cada uno de estos botones tienen asociados propiedades y eventos a los cuales responden con una acción definida por el código escrito por el programador.

4. PROPIEDADES , EVENTOS Y MÉTODOS

A medida que se crea la aplicación el formulario es un lienzo en donde se van colocando los diferentes controles. No hay ninguna regla para la creación de la apariencia de un formulario, este es un arte, y es parte del programador hacer que esta se vea como lo planificó.

4.1 Propiedades de Formularios y controles en Visual Basic.

Las propiedades de los formularios y controles se deben a variables que están definidas propiamente en Visual Basic, dejando al programador su selección, más no pueden ser cambiadas. A continuación se presenta todas las propiedades a las que el usuario tiene acceso:

- **ActiveControl Property.**

Especifica que el formulario o un control tiene el foco⁴; Cuando el formulario es empleado, no es válido si todos los controles no son visibles en la ejecución de la interface.

Uso:

```
{form.|mdiform.|Screen.}ActiveControl
```

- **ActiveForm Property.**

Especifica el formulario que esta activo

Uso:

```
{mdiform|Screen}.ActiveForm
```

⁴ **Foco.**- Es el ventana o el control que esta activa.

- **AutoRedraw Property.**

Determina si el formulario se redibuja o no cuando es tapado por otro.

Uso:

```
[form.][picturebox.]AutoRedraw[ = {True|False}]
```

Esta propiedad puede tomar el valor de verdadero(true) o falso(false). Cuando es verdad activa la propiedad de repintar inmediatamente el formulario. En falso desactiva esta opción.

- **BackColor.**

Determina el color de fondo del formulario, usando la paleta de colores.

- **BorderStyle.**

Determina el estilo de borde del formulario. Se puede escoger entre cuatro propiedades, Simple, Doble, Simple Fijo, Doble Fijo.

- **Caption.**

Esta propiedad determina el texto que es mostrado en la barra de títulos del formulario.

- **ControlBox.**

Especifica si el cuadro de control que se encuentra en la esquina superior izquierda de todas las aplicaciones *Windows* está activa o no. Este cuadro de control permite a un usuario de aplicaciones *Windows* minimizar, maximizar, mover o cerrar las aplicaciones.

- **Enabled. (False/true).**

Determina que el formulario reaccione o no a eventos. Si se pone en False, el formulario no reaccionara a los eventos que ocurran sobre los controles puestos en él.

- **FillColor.**

Determina el color que se usará para rellenar círculos o cuadros creados con las instrucciones para graficar círculos y líneas.

- **FontName.**

Escoge el tipo de letra (fuente) para la presentación de texto sobre el formulario.

- **FontSize.**

Establece el tamaño de la fuente.

- **FontBold.**

Establece si el texto es en negrita.

- **FontItalic.**

Cuando es true el texto aparece inclinado.

- **FontUnderline.**

Cuando esta propiedad es True el texto aparece subrayado.

- **FontStrikethru.**

Cuando se desea que el texto sea tachado.

- **FontTransparent.**

Determina si se muestra o no el fondo a través de los caracteres.

- **ForeColor.**

Establece el color del texto que aparece impreso sobre el formulario.

- **Height, Width.**

Determina el alto y ancho del formulario o el control.

- **Left, Top.**

Con estas dos propiedades se establece el lugar en donde se encuentra la esquina superior izquierda del formulario con respecto a la esquina superior izquierda del monitor.

- Icon.

Determina el icono que es mostrado en tiempo de ejecución cuando el formulario es reducido de tamaño.

- MaxButton.

Esta propiedad junto con MinButton permiten mostrar o no los botones de maximizar y/o minimizar la ventana.

- MinButton.

Esta propiedad junto con MaxButton permiten mostrar o no los botones de maximizar y/o minimizar la ventana.

- MousePointer.

Indica la forma que tendrá el puntero del ratón cuando se ubique encima del formulario.

- Name.

Esta propiedad es útil en todos los formularios y controles de Visual Basic. Define el Nombre del formulario dentro del código del programa y no puede cambiarse en ejecución de la aplicación.

- Picture.

Con esta propiedad se muestra en el formulario un gráfico .El gráfico puede ser realizado en cualquier editor gráfico para *Windows* o se puede utilizar uno de los gráficos que Visual Basic trae en sus librerías.

- Visible.

Establece si el formulario será visible o no cuando se ejecuta la aplicación.

- WindowState.

Determina si la ventana aparecerá en su estado normal (0), en un estado minimizado, o maximizado, cuando se corre la aplicación.

4.2 Eventos de los controles en Visual Basic

Los eventos⁵ que responde un botón de control son los siguientes:

- Activate.

Cuando el formulario se activa o ha empezado la ejecución de la aplicación.

- Click.

Cuando se realiza un Click⁶ del mouse sobre cualquier parte del formulario

- DblClick.

Cuando se hace un Doble-Click (dos golpes seguidos de mouse) sobre cualquier parte del formulario.

- Deactivate.

Cuando el formulario se pone en el fondo de la pantalla o detrás de un formulario activo.

- DragDrop.

Ocurre cuando finaliza la acción conocida como "arrastras y soltar". Cuando se arrastra un control sobre el formulario, o cuando se arrastra el formulario.

- DragOver.

Cuando una operación de arrastrar y soltar está en progreso.

- GotFocus.

⁵ Eventos.- Acción realizada por el usuario en la aplicación.

⁶ Click.- Cuando se presiona el botón izquierdo del mouse.

Ocurre cuando un objeto recibe el enfoque. Esto ocurre cuando el objeto o formulario reacciona ante un evento realizado sobre él. Se produce con el Tabulador o con el mouse, o en tiempo de ejecución.

- **KeyDown.KeyUp.**

Cuando el usuario presiona (en ese momento KeyDown) o suelta (en ese momento KeyUp) una tecla, mientras el formulario tiene el enfoque.

- **KeyPress.**

Ocurre cuando el usuario presiona y suelta una tecla.

- **Load.**

Ocurre cuando un formulario es llamado por primera vez. Por lo general cuando se corre un programa se llama a por lo menos un formulario la primera vez. Luego se utiliza la sentencia Load para llamar a los demás formularios de la aplicación.

- **LostFocus.**

Cuando el Formulario o un control deja de ser enfocado por la acción del Tabulador o alguna sentencia en el código del programa.

- **MouseDown, MouseMove, MouseUp.**

Eventos asociados al ratón, cuando se presiona una tecla (mouse down), se suelta la tecla (mouse up) o se mueve el puntero del ratón sobre el formulario.

- **Resize.**

Ocurre cuando el tamaño del formulario cambia.

- **Unload.**

Ocurre cuando un formulario es removido de la pantalla. Cuando se vuelve a colocar el formulario en la pantalla, el contenido de todos los controles es reinicializado. Este evento es disparado mediante las acciones de cerrar el formulario desde el cuadro de control, o una sentencia Unload.

Las propiedades de los controles y formularios se listan en el cuadro 2. En la fila que dice: "Referencia" se refiere al número que aparece en la columna uno del cuadro 1.

PROPIEDADES	TB	C	HS	LI	PB	D	M	F	ME	LB	PUN
	B				B				D		
Referencia	1	2	3	4	5	6	7	8	9	10	11
Align				x							
Alignment	x									x	
Archive											
AutoRedraw				x							
AutoSize				x						x	
BackColor	x	x		x	x		x	x	x		
BackStyle										x	
BackStyle											
BorderColor				x							
BorderStyle	x			x				x	x		
BorderStyle				x							
BorderStyle							x				
BorderWidth				x							
Cancel		x									
Caption		x					x		x	x	
ClipControls				x			x				
Columns											
CurrentX				x							
CurrentY				x							
DataField	x			x		x				x	
DataSource	x			x		x				x	
Default		x									
DragIcon	x	x	x	x	x	x	x	x	x	x	
DragMode	x	x	x	x	x	x	x	x	x	x	
DrawMode				x	x						
DrawStyle				x							
DrawWidth				x							
Drive											
Enabled	x	x	x	x	x	x	x	x	x	x	
FileName											
FillColor				x							
FillStyle				x				x			
FontBold	x	x		x	x		x	x	x		
FontItalic	x	x		x	x		x	x	x		
FontName	x	x		x	x		x	x	x		
FontSize	x	x		x	x		x	x	x		
FontStrikthru	x	x		x	x		x	x	x		
FontTransparent				x							
FontUnderline	x	x		x	x		x	x	x		
Forecolor	x			x	x		x	x	x		
hDC				x							
Height	x	x	x	x	x	x	x	x	x		
HelpContextID	x	x	x	x	x		x	x			
Hidden											
HideSelection	x										
hWnd	x	x	x	x	x		x	x			
Image				x							

Value		x	x								
Visible	x	x	x	x	x	x	x	x	x	x	
Width	x	x	x	x	x	x	x	x	x	x	
WordWrap										x	
X1,X2,Y1,Y2				x							

Cuadro 2. Propiedades de los controles

De la misma forma en cuadro 3 se indican los eventos a los cuales los botones de control responden

EVENTOS	TB	C	HS	LI	PB	D	IM	F	M	LB	PU
	B	B			B	B			E		N
									D		
Referencia	1	2	3	4	5	6	7	8	9	10	11
Change	x		x		x	x				x	
Click	x	x			x	x	x	x	x	x	x
DblClick	x				x		x	x	x	x	x
DragDrop	x	x	x		x	x	x	x	x	x	x
DragOver	x	x	x		x	x	x	x	x	x	x
DropDown											
Error											
GotFocus	x	x	x		x	x			x		x
KeyDown	x	x	x		x	x			x		x
KeyPress	x	x	x		x	x			x		x
KeyUp	x	x	x		x	x			x		x
LinkClose	x				x					x	
LinkError	x				x					x	
LinkNotify	x				x					x	
LinkOpen	x				x					x	
LostFocus	x	x	x		x	x			x		x
MouseDown	x	x			x	x	x	x	x	x	x
MouseMove	x	x			x	x	x	x	x	x	x
MouseUp	x	x			x	x	x	x	x	x	x
Paint					x						
PathChange											
PatternChange											
Resize					x						x
Reposition											
RowColChange									x		
Scroll			x								
SelChange									x		
Timer											
Updated											x
Validate											

Cuadro 3. Eventos asociados a los controles

ANEXO No. 2

FORM1.FRM - 1

```
Declare Function Test Lib "c:\windows\system\cuser2.dll" (ByVal Numb%) As Integer
Declare Sub out Lib "c:\windows\system\cuser2.dll" (ByVal Addr%, ByVal Byte%)
Declare Function Inp Lib "c:\windows\system\cuser2.dll" (ByVal Addr%) As Integer

Sub Command1_Click ()
    form1.Visible = False
End Sub

Sub okgrabar_Click ()
End Sub

Sub okgrafico_Click ()
    Unload form1

    form1.Visible = False
End Sub
```

FORM3.FRM - 1

```
Declare Function Test Lib "c:\windows\system\cuser2.dll" (ByVal Numb%) As Integer
Declare Sub out Lib "c:\windows\system\cuser2.dll" (ByVal Addr%, ByVal Byte%)
Declare Function Inp Lib "c:\windows\system\cuser2.dll" (ByVal Addr%) As Integer

Sub Command1_Click ()
    Load mdiform1

    mdiform1.Show
End Sub
```

MDIFORM1.FRM - 1

```
Declare Function Test Lib "c:\windows\system\cuser2.dll" (ByVal Numb%) As Integer
Declare Sub out Lib "c:\windows\system\cuser2.dll" (ByVal Addr%, ByVal Byte%)
Declare Function Inp Lib "c:\windows\system\cuser2.dll" (ByVal Addr%) As Integer

Sub boton3_MouseMove (Button As Integer, Shift As Integer, X As Single, y As Single)
    ayuda3.Visible = True
End Sub

Sub confrec_Change ()

End Sub

Sub nuevo_MouseMove (Button As Integer, Shift As Integer, X As Single, y As Single)

End Sub
```

```
Sub reloj_Click ()
```

```
' es la selección de la frecuencia de resolución, mediante el menú.
```

```
panell.Visible = True  
panel2.Visible = True  
escuela.Visible = False ' no muestra en pantalla el logotipo de EPW  
facultad.Visible = False ' no muestra en pantalla el logotipo de facu  
ltad  
sintesis.Visible = False ' no muestra en pantalla el logotipo de SDD  
version.Visible = False ' no muestra en pantalla el logotipo de la v  
ersión  
novell.Visible = False ' no muestra el gráfico de novell  
diego.Visible = False ' no muestra en pantalla el logotipo del nom  
bre  
frecuencia.Visible = False ' no muestra en pantalla la ventana de escog  
er la frecuencia de trabajo  
  
datos = 888 ' da en decimal la dirección del puerto para  
lelo, bus de datos  
Control = 890 ' da en decimal la dirección del puerto para  
lelo, bus de control  
out Control, 11 ' señal de inicialización de ordenes.  
out datos, 6 ' inicializa al circuito para que se encuent  
re listo a recibir datos y grabarlos  
out Control, 5 ' se ejecuta la orden anterior
```

```
End Sub
```

```
Sub okreloj_Click ()
```

```
End Sub
```

```
Sub reloj1_Click (Value As Integer)
```

```
' la subrutina siguiente, posibilita la selección automática para escoger  
' una frecuencia de resolución a definirla mediante un oscilador externo
```

```
datos = 888 ' direcciones del puerto paralelo  
Control = 890 ' direcciones del puerto paralelo  
out Control, 11 ' reset de ordenes del circuito  
out datos, 15 ' es el dato que se manda para seleccionar el re  
loj de oscilación escogido  
out Control, 10 ' pasa la orden al MUX para seleccionar la frecu
```

```
MDIFORM1.FRM - 2
```

```
encia escogida  
out Control, 11 ' reset de ordenes al circuito
```

```
End Sub
```

```
Sub reloj2_Click (Value As Integer)
```

```
' la subrutina siguiente, posibilita la selección automática para escoger
```

' una frecuencia de resolución de 15,258789 Hz

```
datos = 888           ' direcciones del puerto paralelo
Control = 890         ' direcciones del puerto paralelo
out Control, 11      ' reset de ordenes del circuito
out datos, 3         ' es el dato que se manda para seleccionar el re
loj de oscilación escogido
out Control, 10      ' pasa la orden al MUX para seleccionar la frecu
encia escogida
out Control, 11      ' reset de ordenes al circuito
End Sub
```

Sub reloj3_Click (Value As Integer)

' la subrutina siguiente, posibilita la selección automática para escoger
' una frecuencia de resolución de 30,517578 Hz

```
datos = 888           ' direcciones del puerto paralelo
Control = 890         ' direcciones del puerto paralelo
out Control, 11      ' reset de ordenes del circuito
out datos, 2         ' es el dato que se manda para seleccionar el re
loj de oscilación escogido
out Control, 10      ' pasa la orden al MUX para seleccionar la frecu
encia escogida
out Control, 11      ' reset de ordenes al circuito
```

End Sub

Sub reloj3_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)

End Sub

Sub reloj4_Click (Value As Integer)

' la subrutina siguiente, posibilita la selección automática para escoger
' una frecuencia de resolución de 75,000000 Hz

```
datos = 888           ' direcciones del puerto paralelo
Control = 890         ' direcciones del puerto paralelo
out Control, 11      ' reset de ordenes del circuito
out datos, 1         ' es el dato que se manda para seleccionar el re
loj de oscilación escogido
out Control, 10      ' pasa la orden al MUX para seleccionar la frecu
encia escogida
out Control, 11      ' reset de ordenes al circuito
```

End Sub

Sub reloj4_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)

End Sub

Sub reloj5_Click (Value As Integer)

' la subrutina siguiente, posibilita la selección automática para escoger
M01FORM1.FRM - 3

' una frecuencia de resolución de 91,552734 Hz

```
datos = 888           ' direcciones del puerto paralelo
Control = 890        ' direcciones del puerto paralelo
out Control, 11      ' reset de ordenes del circuito
out datos, 0         ' es el dato que se manda para seleccionar el re
loj de oscilación escogido
out Control, 10      ' pasa la orden al MUX para seleccionar la frec
encia escogida
out Control, 11      ' reset de ordenes al circuito
```

End Sub

Sub reloj5_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)

End Sub

Sub actualizar_Click ()

t = 0

textouno.Text = ((mascarauno) / 65536)

'[texto.Text = texto.Text + textouno.Text

texto.Text = ((mascarauno / 2) / (65536))

End Sub

Sub barrer_Click ()

On Error GoTo fin

pasa = pasadas

textouno.Text = ((mascarauno) / 65536)

frecres = ((mascarauno) / 65536)

valor = Int(mascarados / frecres)

eje(0).Text = (valor - 300) * frecres

eje(1).Text = (valor) * frecres

eje(2).Text = (valor + 300) * frecres

medir.Max = 600

valor = valor - 300

barrer:

texto.Text = valor * frecres

For i = 0 To 600

p = valor + i

```
medir.Value = 1
texto.Text = p * frecres
```

```
z = p And 255
y = Int(p / 256) ' corresponde al MSB del selector de frecuencia
```

```
datos = 888 ' da en decimal la dirección del puerto para
lelo, bus de datos
Control = 890 ' da en decimal la dirección del puerto para
lelo, bus de control
out Control, 11 ' señal de inicialización de ordenes.
out datos, 1 ' inicializa al circuito sintetizador digita
l directo
out Control, 5 ' se ejecuta la orden anterior
out Control, 11
```

MDIPORM1.FRM - 4

```
out Control, 11
out datos, y ' corresponde al MSB del selector de frecuen
cia
out Control, 13
out Control, 11
out datos, z ' corresponde al LSB del selector de frecuen
cia
out Control, 1
Control, 1
out Control, 11
```

```
Next i
valor = valor + 600
For k = 0 To 600
```

```
v = valor - k
medir.Value = 600 - k
texto.Text = v * frecres
z1 = v And 255
Y1 = Int(v / 256) ' corresponde al MSB del selector de frecuencia
```

```
datos = 888 ' da en decimal la dirección del puerto para
lelo, bus de datos
Control = 890 ' da en decimal la dirección del puerto para
lelo, bus de control
out Control, 11 ' señal de inicialización de ordenes.
out datos, 1 ' inicializa al circuito sintetizador digita
l directo
out Control, 5 ' se ejecuta la orden anterior
out Control, 11
```

```
out Control, 11
```

```

    out datos, Y1          ' corresponde al MSB del selector de frecue
ncia
    out Control, 13
    out Control, 11
    out datos, z1        ' corresponde al LSB del selector de frecue
ncia
    out Control, 1
    out Control, 11

```

```
Next k
```

```
valor = Int(mascarados / frecres)
```

```
valor = valor - 300
```

```
pasa = pasa - 1
```

```
If pasa = 0 Then GoTo finalizar
```

```
GoTo barrer
```

```
fin:
```

```
MsgBox "No ha ingresado datos de frecuencia central y de reloj"
```

```
Exit Sub
```

```
finalizar:
```

```
End Sub
```

```
Sub boton1_Click ()
```

```
    boton7.Visible = False
```

```
    boton8.Visible = False
```

```
    cuadro.Visible = True
```

```
    panell.Visible = False
```

```
    On Error GoTo salida
```

```
MDIFORM1.FRM - 5
```

```
abrir.Filter = "Archivos SDD (*.sdd)|*.sdd|"
```

```
abrir.Action = 0
```

```
abrir.Action = 1
```

```
u = 0
```

```
file = abrir.FileName
```

```
indica.Visible = True
```

```
For i = 0 To 8191
```

```
    i
```

```
    = 0 To 8191
```

```
    indica.Value = i
```

```
    If u > 0 Then GoTo Piri
```

```
    Close #1
```

```
    Open file For Input As #1
```

```
Piri:
```

```
    u = u + 1
```

```
    If u = 8191 Then GoTo salida
```

```
    Input #1, a
```

```

X = (1 * .7) + 2100
y = -(7 * a) + 2985
Load form1
form1.Visible = True
form1.Line (2100, 1089)-(2100, 3089), RGB(255, 0, 0)
form1.Line (2100, 2089)-(7900, 2089), RGB(255, 0, 0)
form1.Visible = True
form1.PSet (X, y), RGB(0, 0, 255)

```

```
Next i
```

```
salida:
```

```

cuadro.Visible = False
boton7.Visible = True
boton8.Visible = True

```

```
Exit Sub
```

```
indica.Visible = False
```

```
End Sub
```

```
Sub boton1_MouseMove (Button As Integer, Shift As Integer, X As Single, y As Single)
```

```
ayudal.Visible = True
```

```
End Sub
```

```
Sub boton2_Click ()
```

```

' es el botón de acceso rápido a la subrutina que posibilita observar la
' señal que se esta grabando y al mismo tiempo grabar en circuito externo
' la misma señal.

```

```
boton7.Visible = False
```

```
boton8.Visible = False
```

```
panell.Visible = False ' borra de pantalla el pane
```

```
1 1
```

```
cuadro.Visib cuadro.Visible = True
```

```
On Error GoTo finaldos
```

```
panel2.Visible = False ' borra de pantalla el pane
```

```
1 2
```

```

abrir.Filter = "Archivos SDD (*.sdd)|*.sdd|" ' solo selecciona los archi
vos con la extensión tipo SDD

```

```
abrir.Action = 0 ' censa que se presione ENT
```

```
ER
```

```
abrir.Action = 1 ' censa que se presione CAN
```

```
CEL
```

```
file = abrir.FileName ' una vez presionado enter
```

```
o hecho doble click, asigna el archivo seleccionado como file
```

```
indica.Visible = True ' habilita el indicador del
```

```
porcentaje de información grabada
```

```
datos = 888 ' da en decimal la direcció
```

```
MDIFORM1.PRM - 6
```

```
n del puerto paralelo, bus de datos
```

```
Control = 890 ' da en decimal la direcció
```

```
n del puerto paralelo, bus de control
```

```
out Control, 11 ' señal de inicialización d
```

```

e ordenes.
  out datos, 6 ' inicializa al circuito pa
ra que se encuentre listo a recibir datos y grabarlos
  out Control, 5 ' se ejecuta la orden anter
ior
  out Control, 11
inicializandolos: ' inicializa las variables
de control en el software
t = 0 ' variable de conteo del MS
B, una vez cumplido los 256 direcciones del LSB(Address)
u = 0 ' variable de conteo de información grabada y de abrir el ar
chivo

```

Iniciados:

```

  For i = 0 To 8191 ' variable que define el nú
mero de direcciones de la memoria RAM a ser grabadas
  indica.Value = i ' se ejecuta el comando de
mostrar el porcentaje de la información centaje de la información grabada
  z = i And 255 ' filtra la variable al LSB
  If i = 0 Then GoTo grabados ' condición para que no se
produzca una división para cero.
  y = (8192 / i) - (8192 / [256 * (t + 1)]) ' algoritmo que define grab
ado de 256 direcciones y aumento en uno del MSB
  If y = 0 Then GoTo altados ' en este momento se produc
e la condición de última dirección del LSB grabado
grabados:
  out datos, i ' pone en el puerto paralel
o la dirección del LSB
  out Control, 3 ' pone en memoria RAM la di
rección del LSB, es una señal de control
  out datos, t ' pone en el puerto paralel
o la dirección del MSB
  out Control, 15 ' pone en memoria RAM la
dirección del MSB, es una señal de control
  If u > 0 Then GoTo leedos ' contador de datos grabado
s
Close #1

```

```

  Open file For Input As #1 ' lee del archivo abierto lo
s datos correspondiente
leedos:
  u = u + 1 ' datos grabados, actualizac
ión de la variable
  If u = 8192 Then GoTo finaldos ' indica que se ha acabado d
e grabar en memoria
  Input #1, a ' lee del archivo el dato
  out datos, a ' pone el dato en el puerto
paralelo
  out Control, 7 ' habilita el dato a la entr
ada de la memoria RAM
  out Control, 9 ' graba en la memoria RAM
  X = (i * .7) + 2100 ' escala de gráfico en el e
je X
  y = -(7 * a) + 2985 ' escala de gráfico en el ej
e Y
  Load form1
  form1.Line (2100, 1089)-(2100, 3089), RGB(255, 0, 0) ' dibuja la li

```

```
boton8.Visible = True
```

```
Exit Sub
```

```
indica.Visible = False ' deja de mostrar la indicación gráfica del porcenta  
je de información grabada
```

```
End Sub
```

```
Sub boton2_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Sin  
gle)
```

```
ayuda2.Visible = True$ ayuda2.Visible = True
```

```
End Sub
```

```
Sub boton4_Click ()
```

```
' es el botón de acceso rápido para cargar el menú de control de
```

```
' frecuencia del circuito
```

```
panell.Visible = True
```

```
panel2.Visible = True
```

```
ventana(8).Visible = False
```

```
pasadas.Visible = False
```

```
ventana(6).Visible = False
```

```
ventana(7).Visible = False
```

```
ventana(8).Visible = False
```

```
ventana(9).Visible = False
```

```
ventana(10).Visible = False
```

```
ventana(11).Visible = False
```

```
frecuencia.Visible = True ' muestra la pantalla de control de frecuenc  
ia de salida
```

```
escuela.Visible = False ' no muestra en pantalla el logotipo de EPN
```

```
facultad.Visible = False ' no muestra en pantalla el logotipo de facu
```

```
ltad
```

```
sisntesis.Visible = False ' no muestra en pantalla el logotipo de SDD
```

```
version.Visible = False ' no muestra en pantalla el logotipo de la v
```

```
ersión
```

```
novell.Visible = False ' no muestra el gráfico de novell
```

```
diego.Visible = False ' no muestra en pantalla el logotipo del nom
```

```
bre
```

```
frecuencia.Caption = "Control de frecuencia automático"
```

```
ventana(6).Visible = False
```

```
eje(0).Visible = False
```

```
eje(1).Visible = False
```

```
eje(2).Visible = False
```

```
ventana(7).Visible = False
```

```
ventana(3).Visible = True
```

```
ventana(5).Visible = True
```

```
texto.Visible = True
```

```
selector.Visible = True
```

```
mascarados.Visible = False
```

```
actualizar.Visible = True
```

```
barrer.Visible = False
```

```
medir.Visible = False
```

```
End Sub
```

```
Sub boton4_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Sin
```

gle)

ayuda4.Visible = True

End Sub

Sub boton7_Click ()

' subrutina para mostrar la pantalla de presentación

panell.Visible = True ' muestra el panel de fondo 1

panel2.Visible = True ' muestra el panel de fondo 2

escuela.Visible = True ' muestra el mensaje de EPN

facultad.Visible = True ' muestra el mensaje de facultad

diego.Visible = True ' muestra el mensaje del autor del pr

ograma

version.Visible = True ' muestra el mensaje de la versión de

l software

novell.Visible = True ' muestra el gráfico de Novell

sisntesis.Visible = True ' muestra el mensaje de síntesis

frecuencia.Visible = False
de frecuencia

' deshabilita la ventana del control

End Sub

Sub boton7_MouseMove (Button As Integer, Shift As Integer, X As Single, y As Single)

ayuda7.Visible = True

End Sub

Sub boton8_Click ()

End

End Sub

Sub boton8_MouseMove (Button As Integer, Shift As Integer, X As Single, y As Single)

ayuda8.Visible = True

End Sub

Sub boton9_Click ()

panell.Visible = True

panel2.Visible = True

frecuencia.Visible = True ' muestra la pantalla de control de frecuencia de salida

escuela.Visible = False ' no muestra en pantalla el logotipo de EPN

facultad.Visible = False ' no muestra en pantalla el logotipo de facultad

ltad

sisntesis.Visible = False ' no muestra en pantalla el logotipo de SDD

version.Visible = False ' no muestra en pantalla el logotipo de la versión

ersión

novell.Visible = False ' no muestra el gráfico de novell

diego.Visible = False ' no muestra en pantalla el logotipo del nombre

bre

frecuencia.Caption = "Barrido de frecuencia automático"

selector.Visible = False

ventana(6).Visible = True

ventana(7).Visible = True

```
ventana(8).Visible = True
ventana(9).Visible = True
ventana(10).Visible = True
ventana(11).Visible = True
```

```
pasadas.Visible = True
eje(0).Visible = True
eje(1).Visible = True
eje(2).Visible = True
```

MDIFORM1.FRM - 9

```
ventana(3).Visible = True
ventana(5).Visible = True
texto.Visible = True
mascarados.Visible = True
actualizar.Visible = False
barrer.Visible = True
medir.Visible = True
```

End Sub

```
Sub boton9_MouseMove (Button As Integer, Shift As Integer, X As Single, y As Single)
ayuda9.Visible = True
```

End Sub

```
Sub Command3_Click ()
panel1.Visible = False
panel2.Visible = False
```

End Sub

```
Sub mabrir_Click ()
boton7.Visible = False
boton8.Visible = False
cuadro.Visible = True
panel1.Visible = False
On Error GoTo salidali
abrir.Filter = "Archivos SDD (*.sdd)|*.sdd|"
abrir.Action = 0
abrir.Action = 1
```

```
u = 0
file = abrir.FileName
```

```
indica.Visible = True
For i = 0 To 8191
indica.Value = i
If u > 0 Then GoTo Pirili
Close #1
Open file For Input As #1
```

Pirili:

```
u = u + 1
If u = 8191 Then GoTo salidali
Input #1, a
```

```
X = (i * .7) + 2100
y = -(7 * a) + 2985
Load form1
form1.Visible = True
form1.Line (2100, 1089)-(2100, 3089), RGB(255, 0, 0)
form1.Line (2100, 2089)-(7900, 2089), RGB(255, 0, 0)
form1.Visible = True
form1.PSet (X, y), RGB(0, 0, 255)
```

Next I

salidali:

```
cuadro.Visible = False
boton7.Visible = True
boton8.Visible = True
```

Exit Sub

```
Indlca.Visible = False
```

End Sub

Sub macerca_Click ()

MDIFORM1.PRM - 10

```
panell.Visible = False
panel2.Visible = True
Load form3
form3.Show
```

End Sub

Sub mbarrido_Click ()

```
panell.Visible = True
panel2.Visible = True
frecuencia.Visible = True ' muestra la pantalla de control de frecuencia de salida
escuela.Visible = False ' no muestra en pantalla el logotipo de EPM
facultad.Visible = False ' no muestra en pantalla el logotipo de facultad
datadacu
ltad
sintesis.Visible = False ' no muestra en pantalla el logotipo de SDD
version.Visible = False ' no muestra en pantalla el logotipo de la versión
novell.Visible = False ' no muestra el gráfico de novell
diego.Visible = False ' no muestra en pantalla el logotipo del nombre
frecuencia.Caption = "Barrido de frecuencia automático"
selector.Visible = False
ventana(6).Visible = True
ventana(7).Visible = True
ventana(8).Visible = True
ventana(9).Visible = True
ventana(10).Visible = True
ventana(11).Visible = True
```

```

pasadas.Visible = True
eje(0).Visible = True
eje(1).Visible = True
eje(2).Visible = True

ventana(3).Visible = True
ventana(5).Visible = True
texto.Visible = True
mascarados.Visible = True
actualizar.Visible = False
barrer.Visible = True
medir.Visible = True

```

End Sub

```

Sub mdatos_Click ()
' es el botón de acceso rápido a la subrutina que posibilita observar la
' señal que se esta grabando y al mismo tiempo grabar en circuito externo
' la misma señal.
boton7.Visible = False
boton8.Visible = False
cuadro.Visible = True

panel1.Visible = False           ' borra de pantalla el pane
1 1
On Error GoTo finaldosa
panel2.Visible = False         ' borra de pantalla el pane
1 2
abrir.Filter = "Archivos SDD (*.sdd)|*.sdd|" ' solo s" ' solo selecciona los archi
vos con la extensión tipo SDD
abrir.Action = 0               ' censa que se presione ENT
ER
abrir.Action = 1              ' censa que se presione CAN
CEL
file = abrir.FileName         ' una vez presionado enter
o hecho doble click, asigna el archivo seleccionado como File
indica.Visible = True        ' habilita el indicador del
porcentaje de información grabada

```

MDIFORM1.FRM - 11

```

datos = 888                   ' da en decimal la direcció
n del puerto paralelo, bus de datos
Control = 890                 ' da en decimal la direcció
n del puerto paralelo, bus de control
out Control, 11              ' señal de inicialización d
e ordenes.
out datos, 6                  ' inicializa al circuito pa
ra que se encuentre listo a recibir datos y grabarlos
out Control, 5               ' se ejecuta la orden anter
ior
inicializaciondosa:         ' inicializa las variables
de control en el software
t = 0                         ' variable de conteo del MS
B, una vez cumplido los 256 direcciones del LSB(Address)

```

```

u = 0 ' variable de conteo de inf
ormación grabada y de abrir el archivo
iniciadosa:
  For i = 0 To 8191 ' variable que define el nú
mero de direcciones de la memoria RAM a ser grabadas
  indica.Value = i ' se ejecuta el comando de
mostrar el porcentaje de la información grabada ' filtra la variable al LSB
  z = i And 255 ' condición para que no se
  If i = 0 Then GoTo grabadosa ' algoritmo que define grab
  produzca una división para cero ' en este momento se produ
  y = (8192 / i) - (8192 / (256 * (t + 1))) '
ado de 256 direcciones y aumento en uno del MSB
  If y = 0 Then GoTo altadosa '
ce la condición de última dirección del LSB grabado
grabadosa:
  out datos, i ' pone en el puerto paralelo
o la dirección del LSB
  out Control, 3 ' pone en memoria RAM la di
rección del LSB, es una señal de control
  out datos, t ' pone en el puerto paralelo
o la dirección del MSB
  out Control, 15 ' pone en memoria RAM la
dirección del MSB, es una señal de control
  If u > 0 Then GoTo leedosa ' contador de datos grabad
os
Close #1

  Open file For Input As #1 ' lee del archivo abierto lo
s datos correspondiente:
leedosa:
  u = u + 1 ' datos grabados, actualizac
ión de la variable
  If u = 8192 Then GoTo finaldosa ' indica que se ha acabado
de grabar en memoria
  Input #1, a ' lee del archivo el dato
  out datos, a ' pone el dato en el puerto
paralelo
  out Control, 7 ' habilita el dato a la entr
ada de la memoria RAM
  out Control, 9 ' graba en la memoria RAM
  X = (i * .7) + 2100 ' escala de gráfico en el e
je X
  y = -(7 * a) + 2985 ' escala de gráfico en el ej
e Y
  Load form1

  form1.Line (2100, 1089)-(2100, 3089), RGB(255, 0, 0) ' dibuja la li
nea vertical
  form1.Line (2100, 2089)-(7900, 2089), RGB(255, 0, 0) ' dibuja la li
nea horizontal
  form1.Visible = True ' carga la forma que soporta
el gráfico que se realiza
  form1.PSet (X, y), RGB(0, 0, 255) ' dibuja el punto en las coo
rdenadas correspondientes y de color azul

```

Next 1

altadosa:

t = t + 1

s periodos de 256 que se han producido

GoTo grabadosa

Finaldosa:

cuadro.Visible = False

boton7.Visible = True

boton8.Visible = True

' contador que informa de lo

Exit Sub

indica.Visible = False

je de información grabada

' deja de mostrar la indicación gráfica del porcenta

End Sub

Sub MDIForm_Load ()

panel1.Visible = True

panel2.Visible = True

End Sub

Sub frecuencia_Click ()

panel1.Visible = True

panel2.Visible = True

ventana(8).Visible = False

pasadas.Visible = False

ventana(6).Visible = False

ventana(7).Visible = False

ventana(8).Visible = False

ventana(9).Visible = False

ventana(10).Visible = False

ventana(11).Visible = False

ventana(6).Visible = False

eje(0).Visible = False

eje(1).Visible = False

eje(2).Visible = False

■ascarados.Visible = False

actualizar.Visible = True

barrer.Visible = False

■edir.Visible = False

frecuencia.Visible = True

ia de salida

escuela.Visible = False

facultad.Visible = False

ltad

■ntesis.Visible = False

version.Visible = False

ersi"n

novell.Visible = False

diego.Visible = False

bre

' muestra la pantalla de control de frecuencia

' no muestra en pantalla el logotipo de BPM

' no muestra en pantalla el logotipo de facu

' no muestra en pantalla el logotipo de SDD

' no muestra en pantalla el logotipo de la v

' no muestra el gráfico de novell

' no muestra en pantalla el logotipo del no

' habilita el ■enl para realizar el control de frecuencia directo al

5

```
    mbarrido.Enabled = False
    boton2.Visible = False ' deshabilita el bot"n de acceso r pido
para control de frecuencia de resoluci"n
    boton9.Visible = False
    boton4.Visible = False ' deshabilita la transmissi"n de datos
```

End Sub

```
Sub mpresentacion_Click ()
' subrutina para mostrar la pantalla de presentaci"n
```

```
    panell.Visible = True ' muestra el panel de fondo 1
    panel2.V panel2.Visible = True ' muestra el panel de fondo 2
    escuela.Visible = True ' muestra el mensaje de EPN
    facultad.Visible = True ' muestra el mensaje de facultad
    diego.Visible = True ' muestra el mensaje del autor del pr
ograma
    version.Visible = True ' muestra el mensaje de la versi"n de
l software
    novell.Visible = True ' muestra el gr fico de Novell
    sintesis.Visible = True ' muestra el mensaje de sintesis

    frecuencia.Visible = False ' deshabilita la ventana del control
de frecuencia
```

End Sub

```
Sub msalir_Click ()
End
```

End Sub

```
Sub msi_Click ()
    msi.Checked = True ' pone la marca de habilitaci"n de transm
isi"n en SI
    mno.Checked = False ' pone la marca de habilitaci"n de transm
isi"n en SI
```

```
    mfrecuencia.Enabled = True ' habilita del men" la opci"n de control
de frecuencia
    mdatos.Enabled = True ' habilita del men" la opci"n de datos
    mbarrido.Enabled = True
    boton2.Visible = True ' habilita el bot"n de acceso r pido para
control de frecuencia de resoluci"n
    boton9.Visible = True
    boton4.Visible = True ' habillta la transmissi"n de datos
End Sub
```

```
Sub Picture1_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As S
ingle)
```

```
    ayuda1.Visible = False ' ayuda que indica que se abra archivo
tipo SDD
    ayuda2.Visible = False ' ayuda que indica opci"n de leer y grab
ar en circuito
```

```

ayuda3.Visible = False      ' ayuda que indica selecci"n de frecuenc
ia de resoluci"n
ayuda4.Visible = False      ' ayuda que indica selecci"n de frecuenc
ia de trabajo
ayuda7.Visible = False      ' ayuda que indica pantalla de presentac

```

```

i"n
ayuda8.Visible = False      ' ayuda de salida de programa
ayuda9.Visible = False      ' ayuda del barrido de frecuencias
End Sub

```

```

Sub selector_Change ()
frecres = (mascarauno) / 65536      ' calcula la frecuencia de resoluci"n
texto.Text = selector.Value * frecres      ' muestra en pantalla la frecuenci
a de resoluci"n, segun el cursor de selecci"n

```

```

z = selector.Value And 255
y = Int(selector.Value / 256)      ' corresponde al MSB del selector de frecue
ncia

```

```

datos = 888      ' da en decimal la direcci"n del puerto para
lelo, bus de datos
Control

```

```

Control = 890      ' da en decimal la direcci"n del puerto para

```

```

lelo, bus de control
out Control, 11      ' se"al de inicializaci"n de ordenes.
out datos, 1      ' inicializa al circuito sintetizador digita

```

```

1 directo
out Control, 5      ' se ejecuta la orden anterior
out Control, 11

```

```

out Control, 11
out datos, y      ' corresponde al MSB del selector de frecuen

```

```

cia
out Control, 13
out Control, 11
out datos, z      ' corresponde al LSB del selector de frecuen

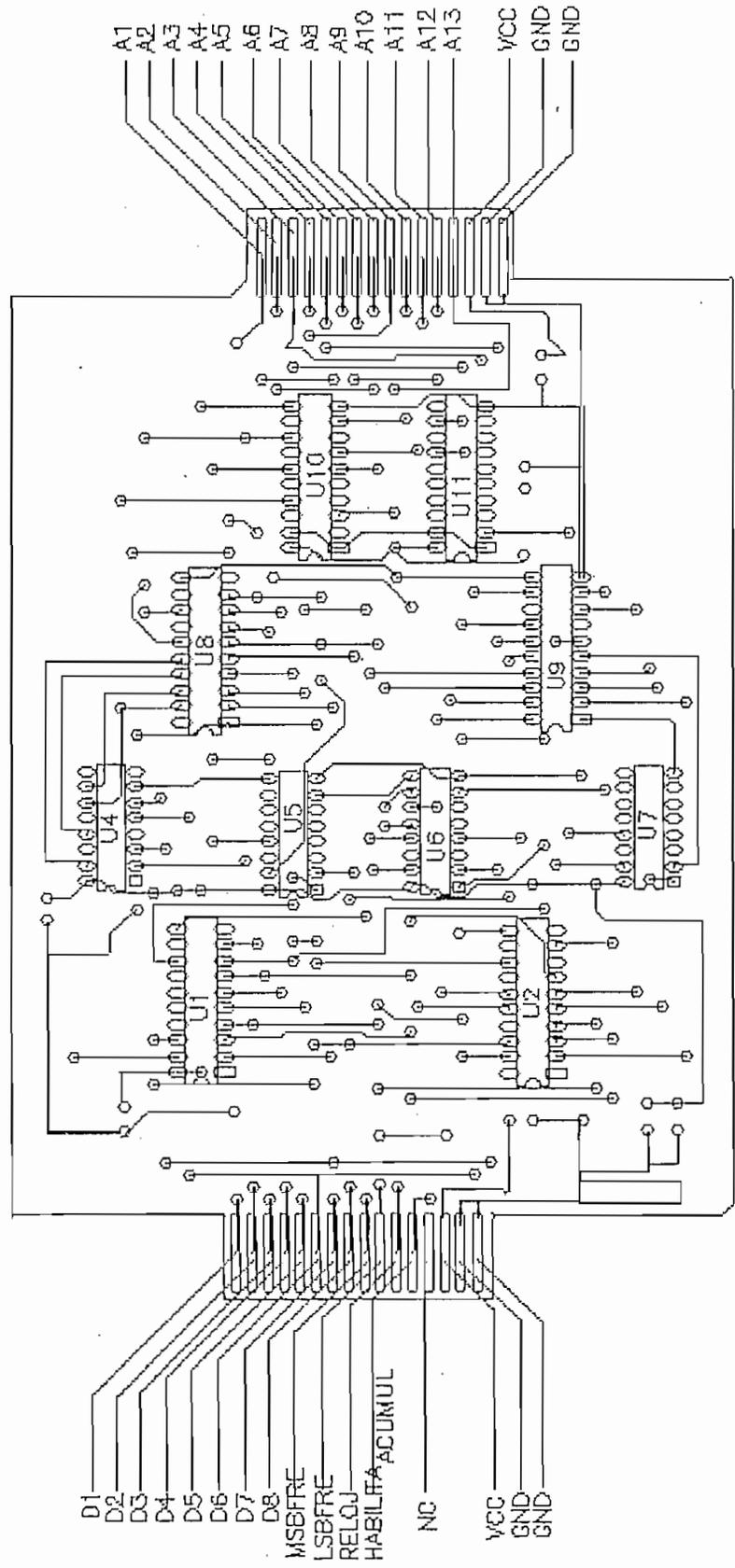
```

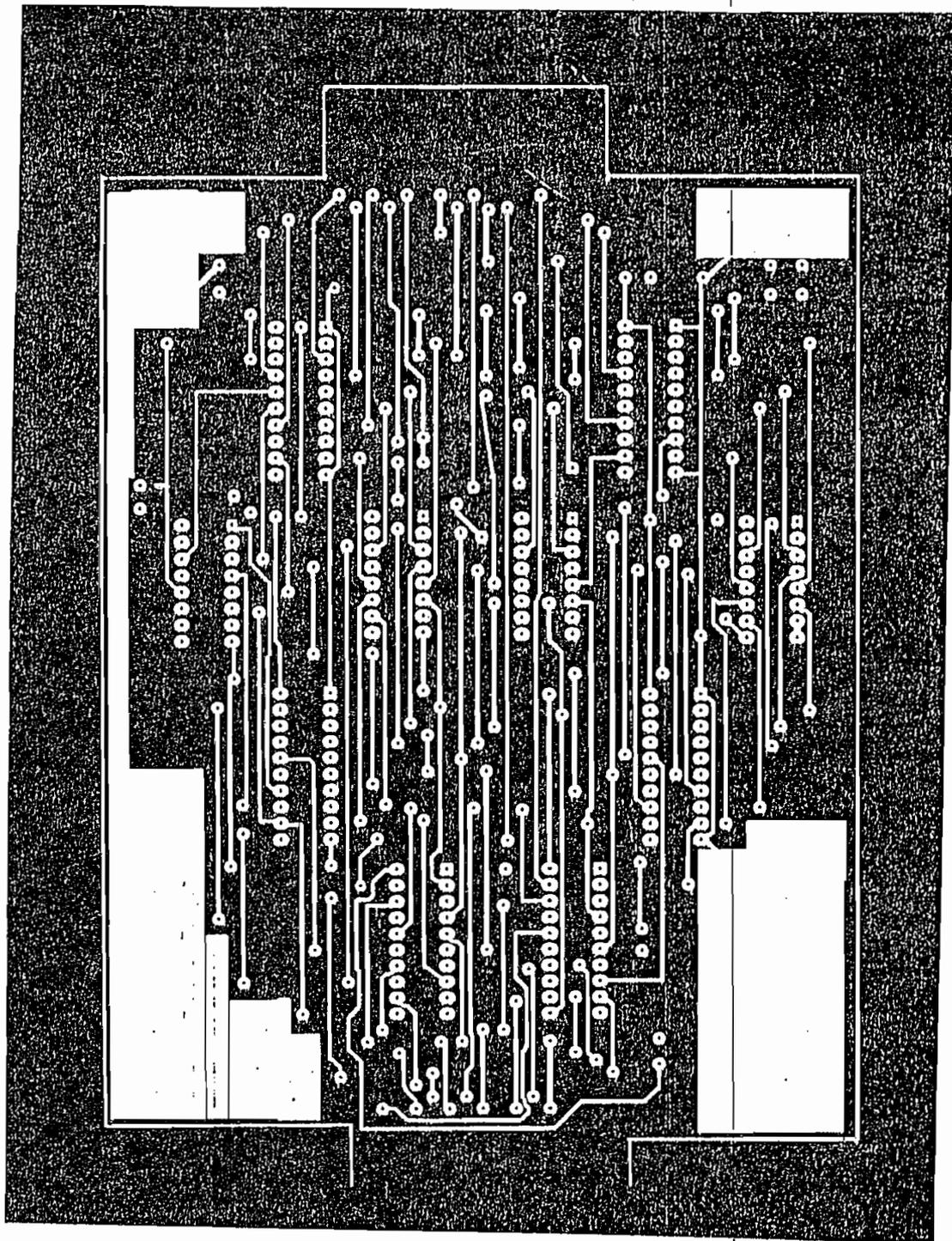
```

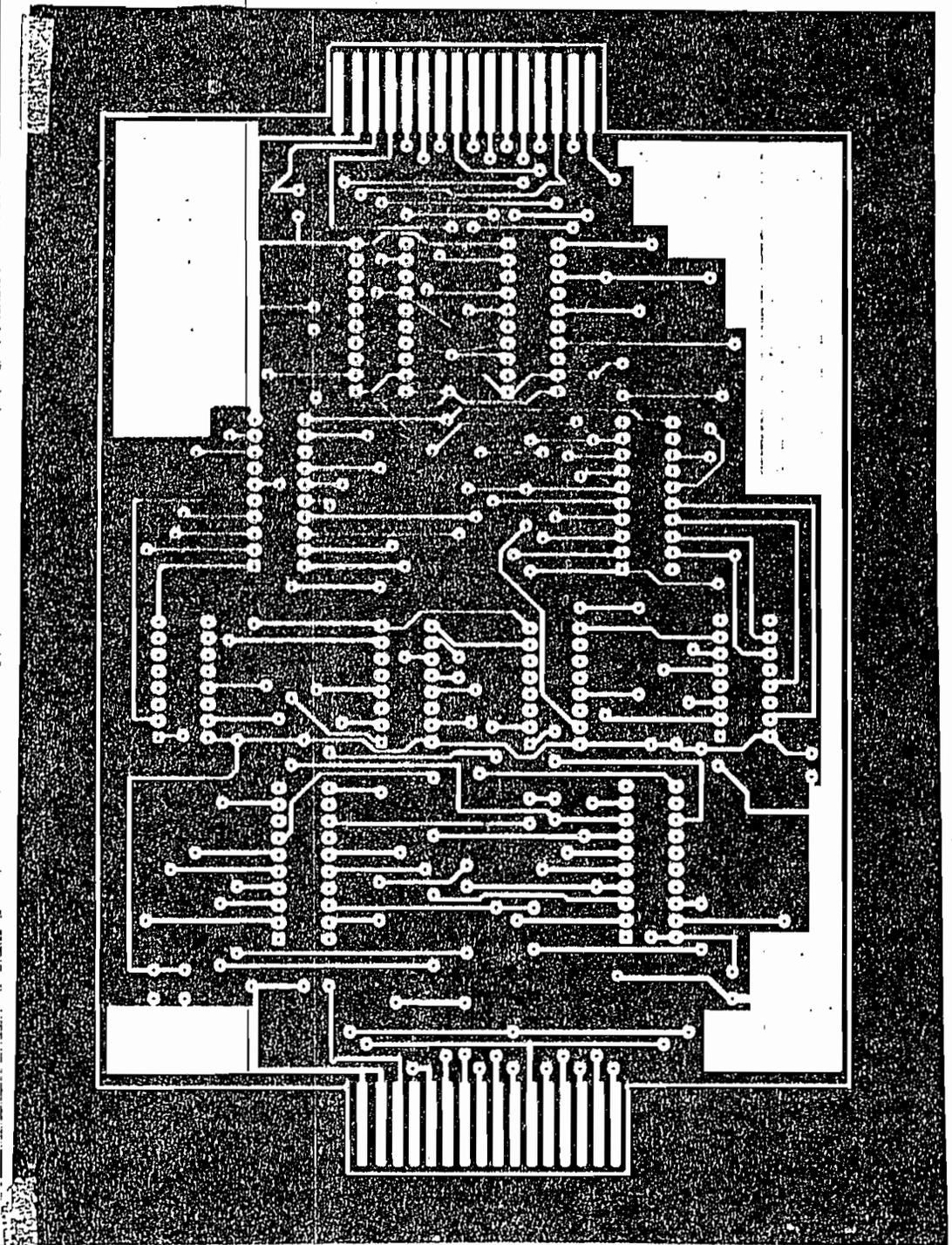
cia
out Control, 1
out Control, 11
End Sub

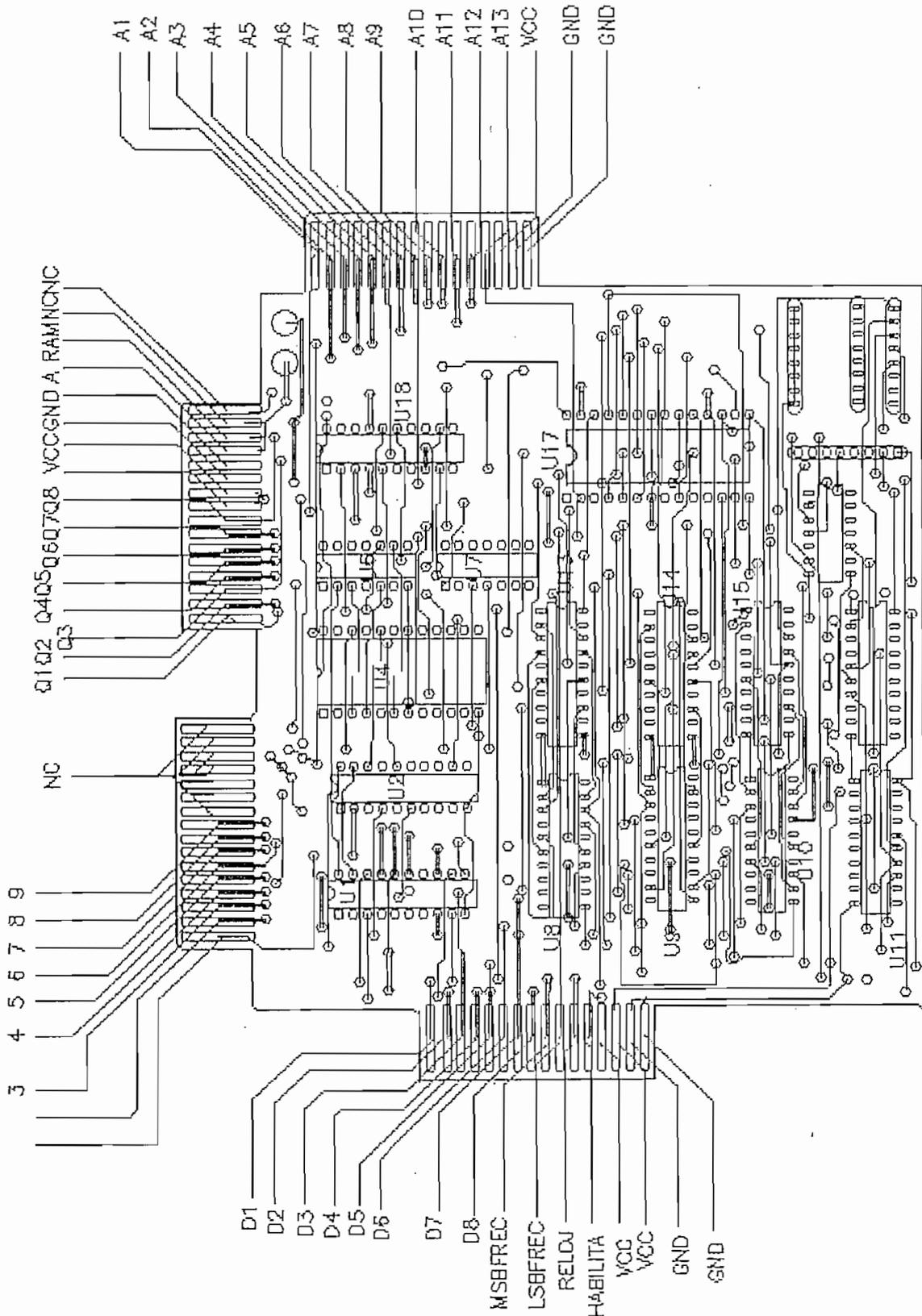
```

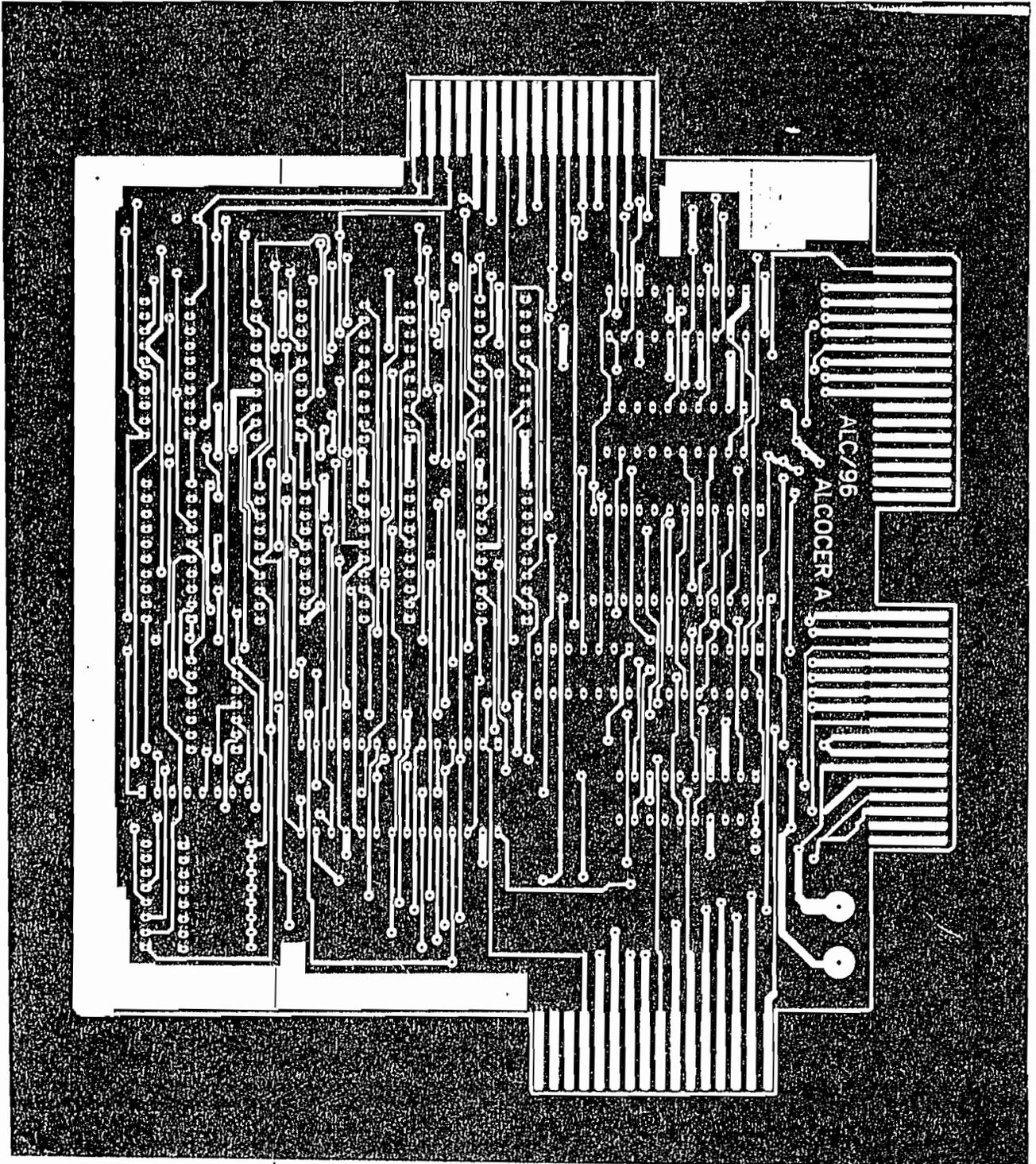
ANEXO No. 3

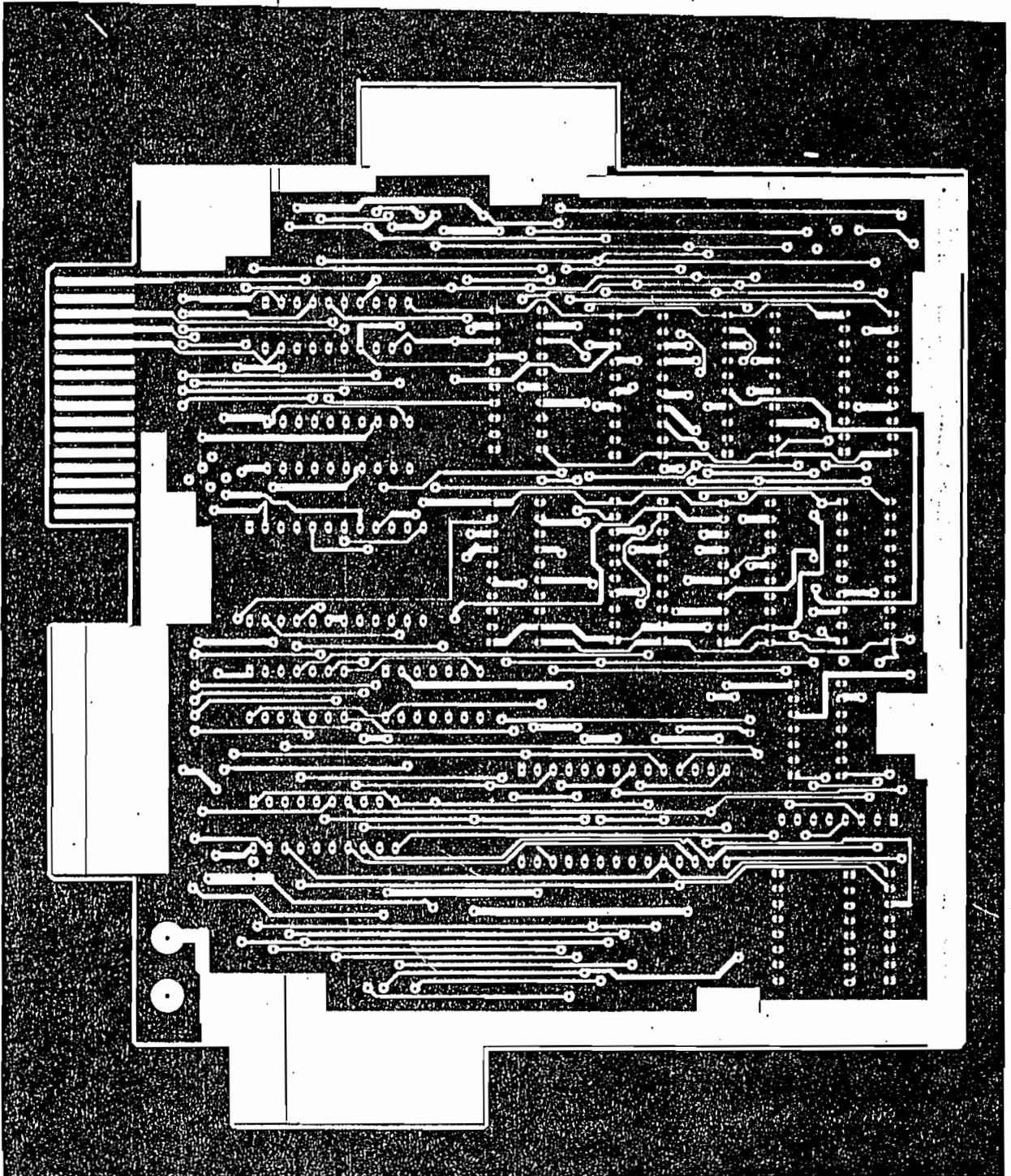












ANEXO No. 4

```

OPEN "c:\dos\nrz001.sdd" FOR OUTPUT AS #1 ' Define un nombre al archivo
' abre el mismo para grabar d
' en el.

      FOR i = 0 TO 8191 ' Define el numero de localid
' a grabar

IF i >= 1023 THEN GOTO uno ' Primer intervalo a grabar
y = 255 ' Funcion que se desea grabar
' en el intervalo predefinido
      GOTO graba ' va ha subrutina de graba
uno: '
IF i >= 2046 THEN GOTO dos ' Segundo intervalo a grabar.
y = 0 ' Funcion a grabar en
' segundo intervalo.
      GOTO graba ' Va a subrutina graba
dos: '
IF i >= 4092 THEN GOTO tres ' Tercer Intervalo
y = 255
      GOTO graba
tres: '
IF i >= 5115 THEN GOTO cuatro ' Cuarto Intervalo
y = 0
      GOTO graba
cuatro: '
IF i >= 6138 THEN GOTO cinco ' Quinto intervalo
y = 255
      GOTO graba
cinco: '
IF i = 8191 THEN GOTO fin ' Noveno Intervalo
y = 0
      GOTO graba
graba: ' subrutina de grabar el dato en archi

```

```
z = INT(y)           ' guarda solo el valor entero.
PRINT #1, z         ' en este momento guarda el dato.
PRINT z, i          ' muestra en pantalla en dato y la
                    ' direccion.
NEXT i              ' va ha la siguiente direccion.
fin:                ' fin de programa
END
```

ANEXO No. 5

RF (RADIO FREQUENCY) SIGNAL GENERATORS ARE ESSENTIAL equipment for radio-frequency design. The TG2000 described in this article is a PC-based synthesized signal generator. It can also function as a tracking generator for DKD Instruments' Models 810B and 1800C/H PC-based spectrum analyzers. The model 810 spectrum analyzer was the subject of an article in *Radio-Electronics*, August and September 1991.

The TG2000 is a IBM-PC compatible card that can be installed in an 8-bit slot on the PC bus. (Alternatively, it can be interfaced to a parallel port, although an external power supply is needed for that.) Software for the TG2000 is a stand-alone package that is available on the *Electronics Now* BBS (516-293-2283, 9600; V.32, V.42bis).

By reviewing the key specifications shown in Table 1, it can be seen that the TG2000 covers the band from 4 to more than 2048 MHz in progressive octave bandwidths. Its output is phase-locked to a 4-MHz crystal reference and its output power ranges from -5 dBm to 0 dBm. Output impedance is nominally 50 ohms and is provided via an F-connector. An optional output connector provides an attenuated output. External FM modulation is supported via an AC-coupled RCA jack input. A sync pulse is provided at another RCA jack for triggering scopes and other equipment. System requirements are very modest for stand-alone operation: a PC/XT or better with CGA or better monitor, 512K RAM, and one floppy drive are all you need.

RF generator basics

Figure 1 shows the RF output spectrum from an "ideal" generator that produces a single frequency or tone with infinite purity. The real world is more complex than Fig. 2 shows. An actual spectrum consists of the fundamental frequency, as well as other undesirable frequency components.

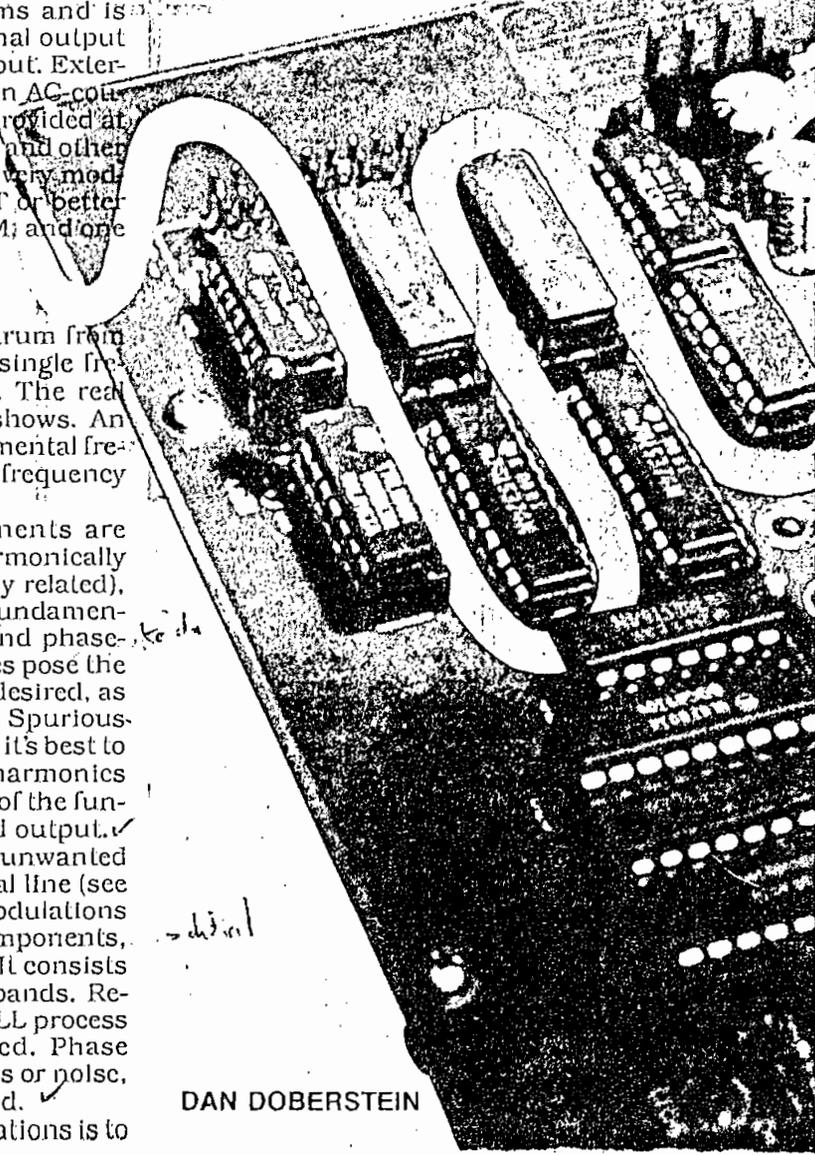
The unwanted frequency components are broken down into three categories: Harmonically related, spurious (or non-harmonically related), and residual modulation close to the fundamental frequency that are due to noise and phase-locked loop (PLL) processes. Harmonics pose the least problems, and at times could be desired, as in frequency multiplication or mixing. Spurious response is generally never useful, and it's best to minimize its content and power. Subharmonics or outputs at whole number fractions of the fundamental are almost always undesired output.

To understand the third group of unwanted outputs "zoom in" on the fundamental line (see Fig. 3), where the small residual modulations can be seen. Of all the unwanted components, this one is the toughest to eliminate. It consists of phase noise and residual FM sidebands. Residual FM is generally caused by the PLL process and cannot be completely eliminated. Phase noise is caused by random fluctuations or noise, and it cannot be completely eliminated.

One way to visualize the residual modulations is to

A look at the design of a 2-GHz RF signal generator.

PC-BASED RF SIGNAL GENERATOR



DAN DOBERSTEIN

TABLE 1
MODEL TG2000 SPECS

Frequency coverage: 4 to 2100 MHz, Single Output, F/SMA Connector.
 Power output: 0 dBm to -10dBm
 w/Option P-01 ; +7 dBm to -10dBm,
 Total power Flatness; ± 2 dB over the band 10 to 2000 MHz

Frequency step sizes:
 Step Band
 250.00000 Khz 1024 to 2048 MHz
 125.00000 KHZ 512 to 1024 MHz
 62.50000 KHZ 256 to 512 MHz
 31.25000 KHZ 128 to 256 MHz
 15.62500 KHZ 64 to 128 MHz
 7.81250 KHZ 32 to 64 MHz
 3.90625 KHZ 16 to 32 MHz
 1.95312 KHZ 8 to 16 MHz
 0.97656 KHZ 4 to 8 MHz

With option DDS-01 step sizes are reduced by:
 Max step size (@ top of band); Take above step sizes divide by 238
 Min step size (@ bot of band); Take above step sizes divide by 476

VSWR of Output (50 ohm): < 1.3 all bands.

Frequency Output Spec's:
 Primary Oscillator Phase locked to Crystal or DDS reference.
 Reference Sidebands; > -50 dBc typ. @ 1000 MHz
 Phase Noise; > -70 dBc/Hz @ 10KHz offset @ 1000 MHz
 Harmonics;
 2nd; > -10dBc 68 to 2200 MHz
 > -5dBc 5 to 68 MHz
 All others > -10dBc 5 to 2200 MHz
 Spurious; > -50dBc TYP to 2200 MHz
 Subharmonics, LO/2; > -20dBc, -30dBc typ. 1100 to 2200 MHz
 5 to 1100 MHz no subharmonic content.

Frequency Accuracy; +/- 30KHz All bands @ room temp.
 +/- 200Hz w/ Optional TCXO reference.

EXT. FM Modulation Input: (Optional) Ext. AM Pulse Modulation:
 Impedance; >10K ohm Impedance; TTL
 Bandwidth; >5KHz Bandwidth; 5KHz
 Max Sensitivity; 1v/100MHz ON/OFF Ratio; >30db
 Coupling; AC Copuling; TTL Levels

Power Requirements: -5 VDC @ 0.5 amps, +12 VDC @ 0.2 amps, -12VDC @ 0.1 amps

Interfaces: PC Bus or Parallel Interface (Centronics).

System Requirements: 360K Disk Drive, VGA, EGA or CGA graphics adapter, 512k Ram, DOS 3.0 or higher.

imagine a perfect FM radio receiver. If the output of a perfect RF generator were fed into the perfect radio, nothing would be heard from the speaker. If a signal containing phase noise and residual FM were fed into our perfect radio receiver, the phase noise would be heard as a hiss, and the PLL sidebands would be heard as a single tone (assuming that the sidebands were audio frequencies within the human ear's frequency response range).

Synthesized vs. non-synthesized

RF signal generators can be classified into two categories: synthesized and non-synthesized, or open-loop. Non-synthesized generators tend to be less expensive, and for that reason you will still find them in consumer receivers. An example of a non-synthesized RF generator is a voltage-controlled oscillator (VCO). Synthesized generators such as the TG2000 are more accurate and frequen-

cy-stable than non-synthesized generators.

A VCO can be converted into a synthesizer with the addition of a PLL. This turns the open-loop VCO into a closed-loop control system in which frequency can be controlled with much greater precision. There is a price for this accuracy though: VCOs essentially have infinite frequency resolution and fast settling

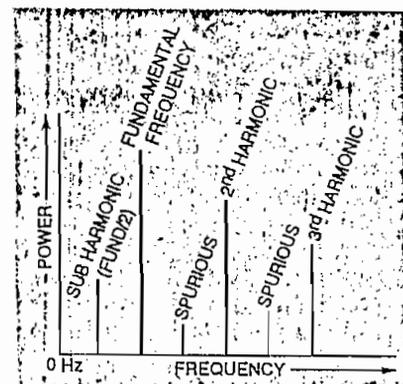


FIG. 1—IDEAL RF OUTPUT SPECTRUM from an ideal generator. The ideal generator produces a single frequency or tone with infinite purity.

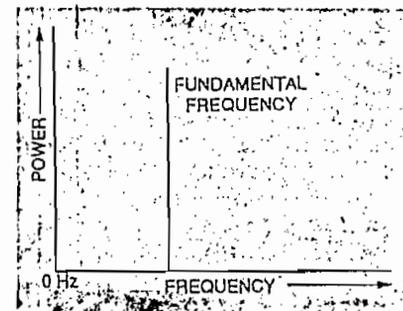


FIG. 2—THE PRACTICAL GENERATOR is considerably more complex than an ideal one. The actual spectrum consists of the fundamental frequency that we want, and other frequency components which we don't want.

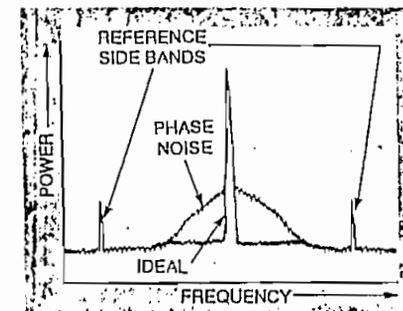


FIG. 3—TO SEE THE THIRD GROUP of unwanted outputs you can "zoom in" on the fundamental line.

times. Once a PLL is coupled to the VCO, the oscillator has fixed-frequency steps and, in general, slower settling time. A benefit to PLL control of a VCO is that the phase noise declines dramatically over that of the open-loop VCO. ✓

The PLL

The minimum number of components for a common PLL-based synthesizer are a VCO, a phase/frequency detector, a loop filter, and a crystal reference (see Fig. 4). The PLL divides the crystal reference down to form the reference frequency, typically on the order of 10 to 500 kilohertz (kHz). The output of the VCO is also divided down and then compared with the reference frequency at the phase/frequency detector. This generates an error voltage that, when filtered and fed to the VCO, forces the VCO frequency to be an exact multiple of the reference frequency. The filter for the error voltage is called the loop filter. The time constants of this filter and other characteristics will greatly effect PLL performance. Phase noise, reference sideband levels, and settling time will all be influenced by the loop filter.

As already mentioned, the reference frequency is found in the RF output as sidebands that are symmetrical about the fundamental. This is the approach used in the TG2000.

Another common element found in RF PLL synthesizers is the prescaler, which reduces the frequency of a VCO to one that can be handled by the programmable dividers. Prescalers come in many divide ratios, and some have multiple divide ratios. Dual-modulus prescalers are a special class of prescalers. These dividers have two divide ratios, typically N and $N + 1$. The active divide ratio depends on a separate control input. Dual-modulus prescalers allow finer step sizes for a given divide ratio than fixed prescalers. ✓

A trade off

Ideally, a synthesizer has infinitely small step sizes and no phase noise. Unfortunately as step size is reduced, phase

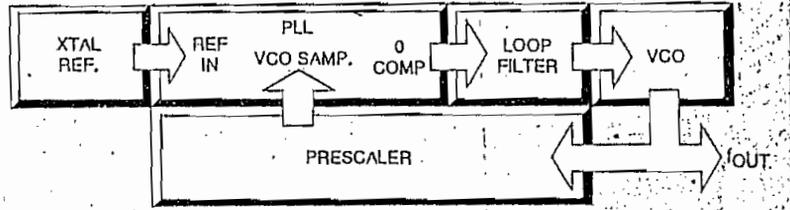


FIG. 4—THE COMPONENTS for common PLL-based synthesizer are a VCO, a phase/frequency detector, a loop filter, and a crystal reference.

noise increases. There are some exceptions to this rule but, in general, to obtain the cleanest possible output in terms of phase noise, larger step sizes or equivalently larger reference frequencies are needed. As a rule of thumb, every time the reference frequency is reduced by a factor of 2, the phase noise increases by 6 dB. A technique that's just starting to be widely used, called fractional synthesis or arithmetically locked loops, offers some relief in the phase noise/step size trade off.

The 125-kHz reference frequency is the highest that can be accepted by the TG2000 while providing its complete frequency range. This limitation on frequency range is caused by the dual modulus characteristic of the MB1501 prescaler. ✓

Multiple octaves

A typical VCO can span, at best, a frequency range roughly double that of the VCO's lowest operating frequency. This is called its octave bandwidth. For example, a VCO with a 0- to 28-volt control voltage range that outputs 1 GHz with zero volts at its control point will output 2 GHz with 28 volts of control voltage. It becomes obvious that to cover 4 to 2048 MHz with one VCO is not practical. To overcome this limitation, many strategies are used in RF generators. They usually involve mixing a VCO octave band down or up.

The TG2000 uses a progressive-division technique. A 1024- to 2048-MHz VCO is phase locked, and then a series of divide-by-2 prescaler ICs creates the frequencies between 4 and 1024 MHz in progressively smaller octave bandwidths. To

cover the 4- to 8-MHz band, for example, the 1024- to 2048-MHz VCO output is divided by 256, or 2^8 . One advantage of progressive division is that phase noise improves with each increase in divide ratio. The power of two relation also applies to the minimum step size available in each octave band.

Each divide-by-2 divides this minimum step size in half. A disadvantage of this method is the relatively high harmonic content caused by the dividers and higher frequency bleed-through due to limitations in high frequency RF isolation.

Another disadvantage of the progressive-division technique is a relatively large step size. The TG2000 uses a reference frequency of 125 kHz, which the divide-by-2 prescaler translates as a 250-kHz step for the 1024- to 2048-MHz bands. Provision has been made in the design of the TG2000 for the addition of an optional direct digital synthesis (DDS) reference. The DDS reference, which can replace the fixed crystal reference, has the ability to select frequency with approximately 2-hertz (Hz) resolution. This would permit much finer steps for the combined system with virtually no loss in phase-noise performance. ✓

The MB1501 PLL

The Fujitsu MB1501 PLL IC is the heart of the TG2000 generator. The block diagram of the 1501 is shown in Fig. 5. The MB1501 incorporates a built in serial interface for loading the programmable reference divider, two counter registers (A and N), a high-speed, dual-modulus divider (64/65 or 128/129), and two phase/frequency detectors. ✓

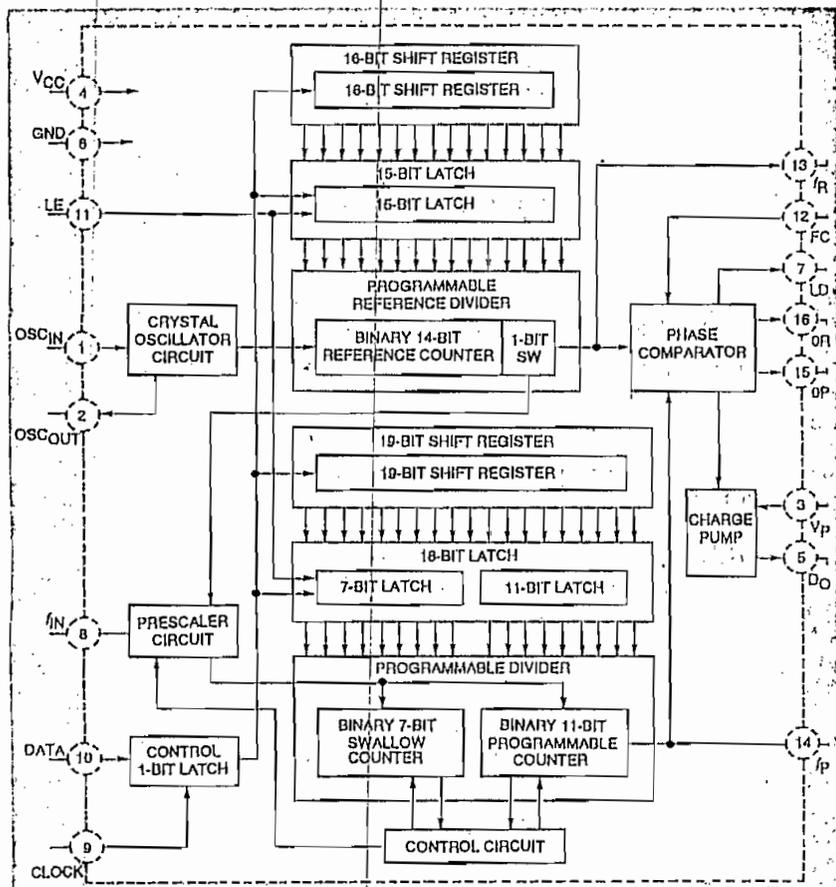


FIG. 5—BLOCK DIAGRAM OF THE FUJITSU MB1501. This PLL chip is the heart of the TG2000 generator.

The output frequency of the VCO is determined by the integers loaded into the R, A and N registers by the following formula:

$$f_{VCO} = [(PN) + A]^2 f_{OSC} / R$$

where $P = 64$ or 128

The TG2000 uses the dual-modulus prescaler in the divide by 64/65 mode, or $P=64$. The VCO can oscillate only at frequencies that are integer multiples of the reference frequency, f_{OSC} . The factor of two accounts for the fixed divide-by-2 operation (done by an NEC584 prescaler) that the VCO output carries out before it reaches the VCO input which has a maximum frequency of 1100 MHz (refer to Fig. 4).

The MB1501's internal dual-modulus prescaler is needed to reduce the input frequency of the VCO to the point where the N and A counters can operate. This dual-modulus ability overcomes the effect of a fixed-mod-

ulus prescaler—step-size multiplication. For example, for a fixed-modulus prescaler of 64, the minimum step sizes would be multiplied by 64. That would translate to a 16-MHz step size in the 1024 to 2048 MHz band for the TG2000. The N/A counter in combination with the dual-modulus divider avoids this problem with the limitation that N must always be greater than or equal to A.

The restriction that N be greater than or equal to A has some not-so-obvious ramifications, such as placing limits on frequency range versus reference frequency, as discussed earlier. Essentially, for a given reference frequency (f_{OSC}/R), the generator is limited to a minimum frequency that can be synthesized while still providing full coverage at minimum step size. For $P=64$, $R=32$, and $f_{OSC}=4$ MHz, the minimum frequency is 512 MHz. Since the generator has a

fixed divide-by-2 prescaler, that results in 1024 MHz at the VCO. Below that frequency, not all integer multiples of the reference frequency are possible due to the $N \geq A$ restriction. In other words, you could not sweep the VCO frequency from 900 to 1024 MHz at step sizes of 250 kHz.

While all this might seem confusing, a general rule is that if you want a large step size for improved phase noise, and your PLL uses a dual-modulus approach, the dual-modulus divide ratio should be as small as possible.

All fixed divide ratios, step sizes, and prescalers are a power or multiple of two. Other numbers could be used, but when it is time to compute A, N, and band switching points for a given output frequency, the power/multiple-of-two relationships pay off.

Two phase/frequency detectors are included with the MB1501 PLL. The bipolar on-chip charge pump permits a minimum number external components for the loop filter. The differential phase-comparator outputs require an external charge pump, usually an op-amp. In the TG2000, an external LM358A op-amp (IC1-a) boosts the bipolar charge pump voltage from 0 to 5 volts to -0.5 to 28 volts. In addition, the op-amp and its associated feedback network acts as an active loop filter. Polarity of the phase-detector output can be inverted by the PLL's FC input. This input is required to compensate for op-amp inversions or VCOs with a negative voltage-versus-frequency slope.

The digital interface to the MB1501 is serial—a block diagram of it is shown in Fig. 6. Three lines are used; CLOCK, DATA, and LOAD ENABLE. Data is clocked in serially, and after an appropriate number of bits (determined by what you are loading) a LOAD ENABLE is sent. A 15-bit word serves as the reference divider, and a 19-bit word holds both the A- and the N-count values. The C source code for the routines that send the N-, A-, and R-register values to the

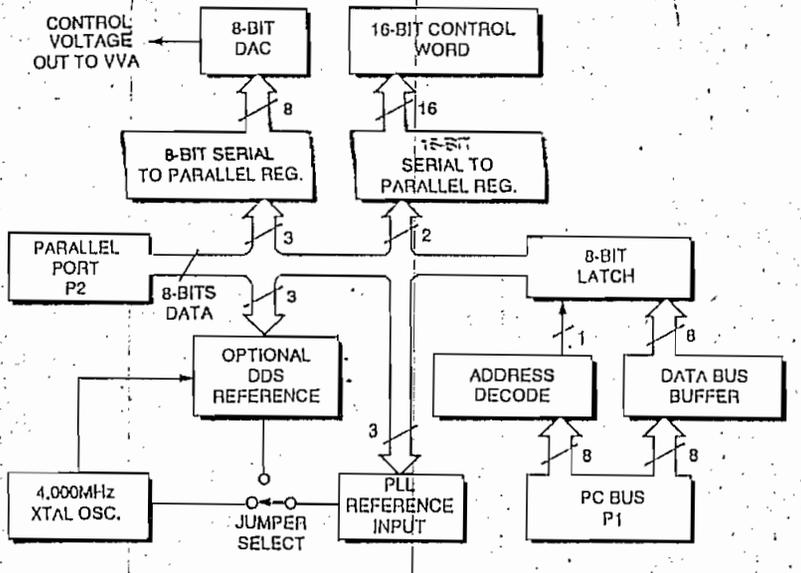


FIG. 6—MB1501 DIGITAL INTERFACE block diagram.

MB1501 are provided on the *Electronics Now BBS* as part of a file called TG2000.ZIP.

Overall operation

As already discussed, the synthesis technique used in the TG2000 is successive division. Every band except the one from

1024—2048 is the output of a divide-by-2 divider. The computer determines the band being used and turns on the appropriate dividers and a single output amplifier path is activated; the other amplifiers are turned off. Five paths cover the 4—2048 MHz band. The band

from 256 to 2048 is covered in three paths, with each path covering one octave. The band from 4 to 256 MHz is covered with two paths, each covering 3 octaves. A total of nine octaves are covered in this fashion from 4 to 2048 MHz. The three octave paths are possible due to the divider used which has selectable 2/4/8 divide ratios available. See the block diagram in Fig. 7.

The summed signal is passed through a voltage-variable attenuator (VVA) pin-diode array (a Seimens BAR60), which can be seen in Fig. 8 as D12. The VVA corrects output power variations and provides variable output power. An 8-bit DAC controls the VVA. The signal then passes through a final amplification stage and out to the RF output connector. The VVA requires a 0- to 10-volt control signal and bias voltage for proper operation. The device has a non-linear attenuation versus control-voltage response. These nonlinearities do not affect attenuation control because the computer uses a look-up table to control it.

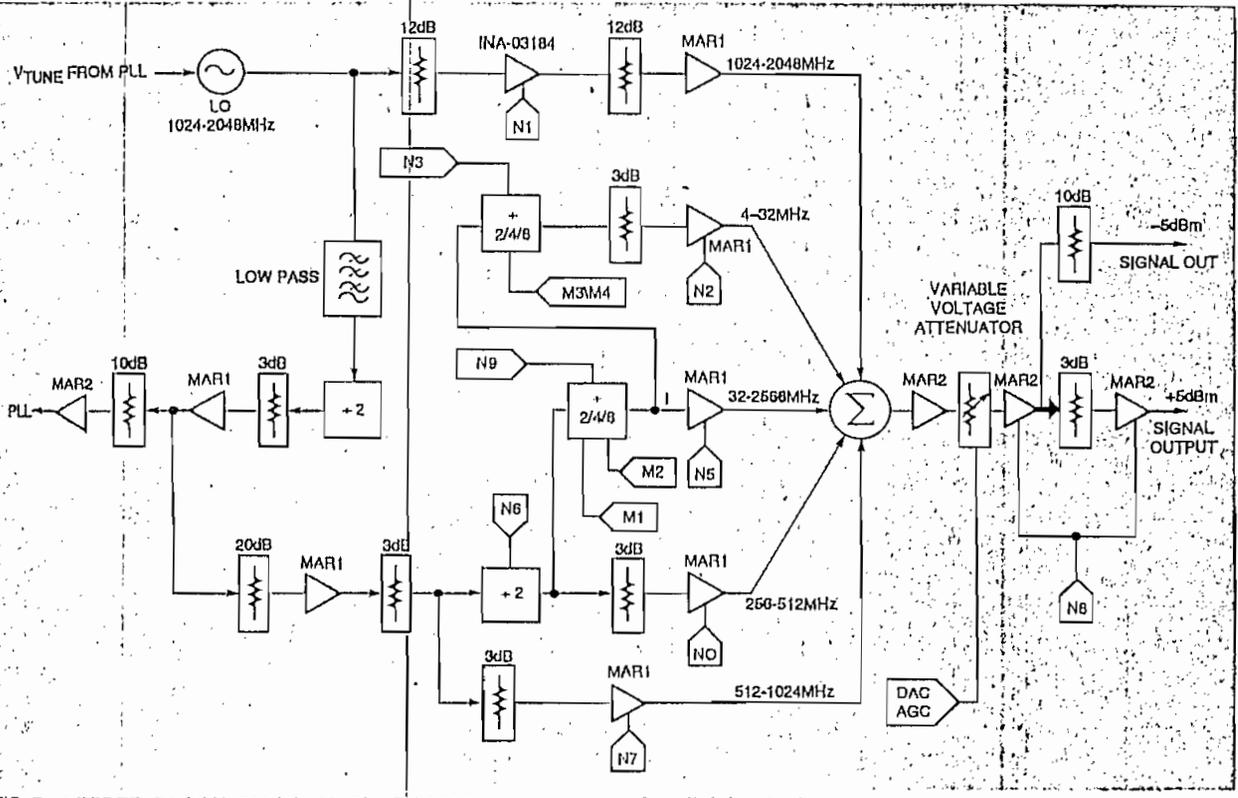


FIG. 7—DIVIDER BLOCK DIAGRAM. The TG2000 uses a successive division technique for synthesis.

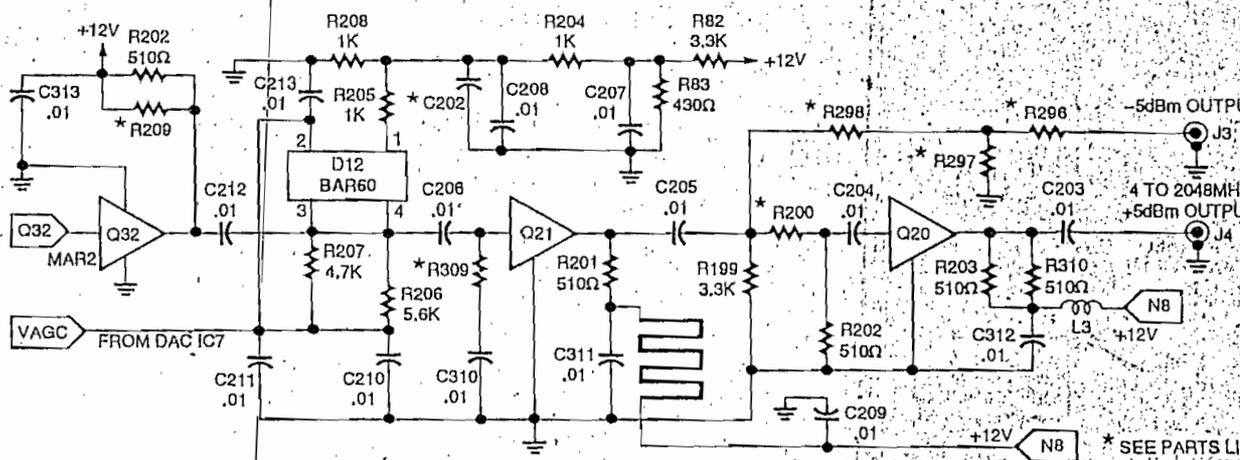


FIG. 8—THE SUMMED SIGNAL is passed through a voltage-variable attenuator pin-diode array (D12), which corrects output power variations and provides variable output power.

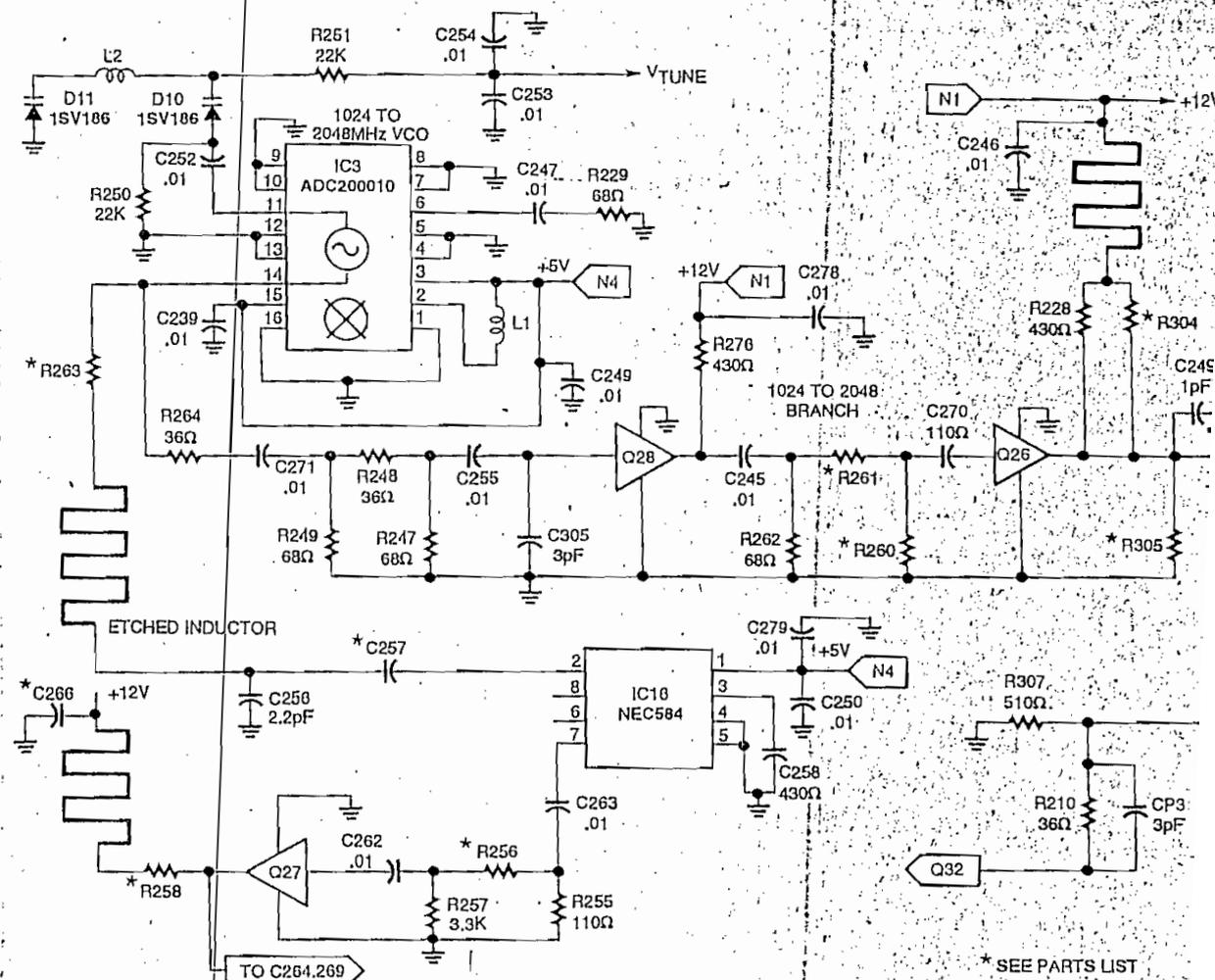


FIG. 9—THE FIRST NEC548G (IC16) acts as a prescaler, reducing the output frequency of the VCO by a factor of 2. It becomes the 512-to-1024 divider when this band amplifier is switched on.

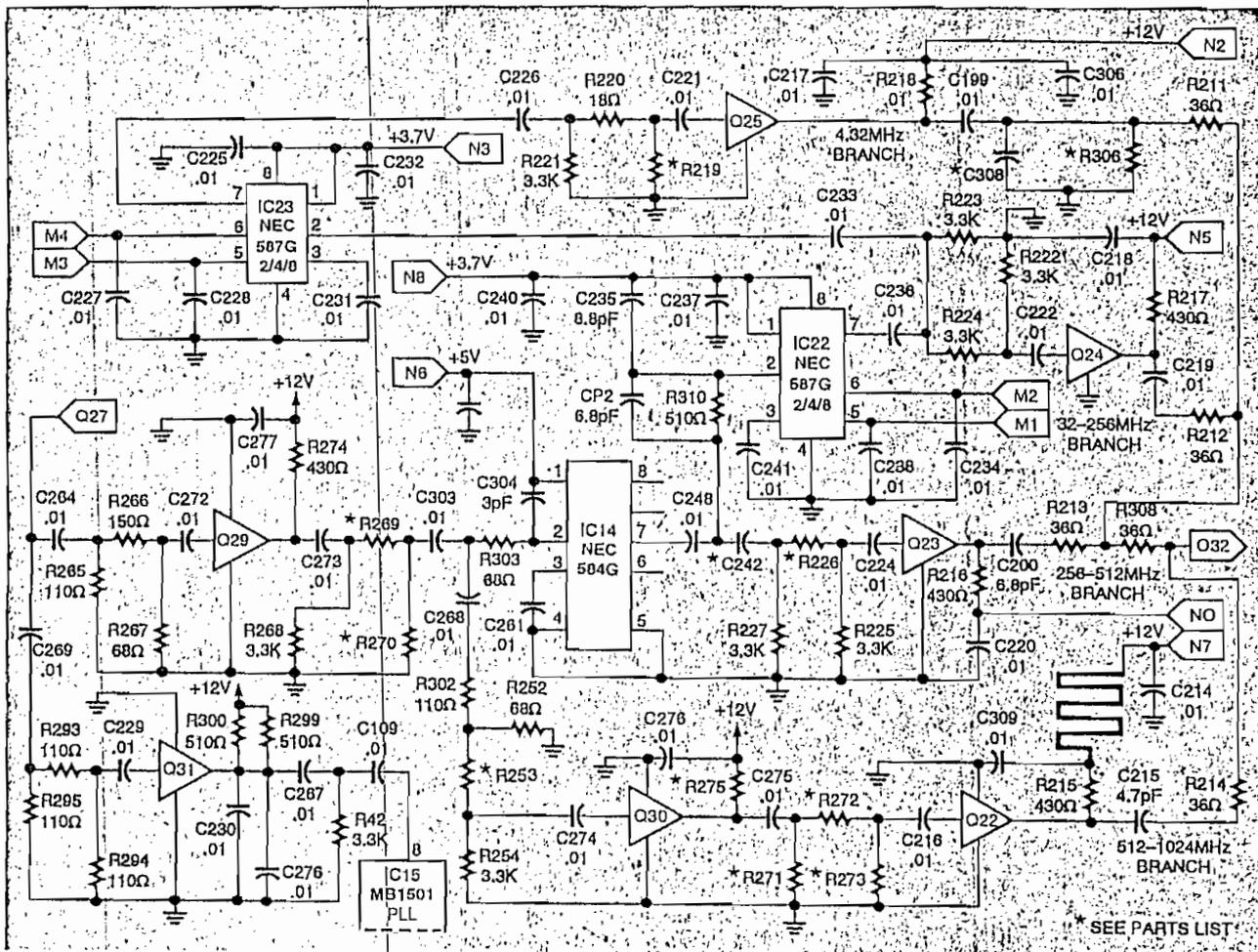


FIG. 10—AFTER THE SIGNAL IS SPLIT, one half is sent to the 512-to-1024 path. Splitting again occurs with half going to the 512 to 1024 amplifier and the other half to the 256 to 512 MHz divider.

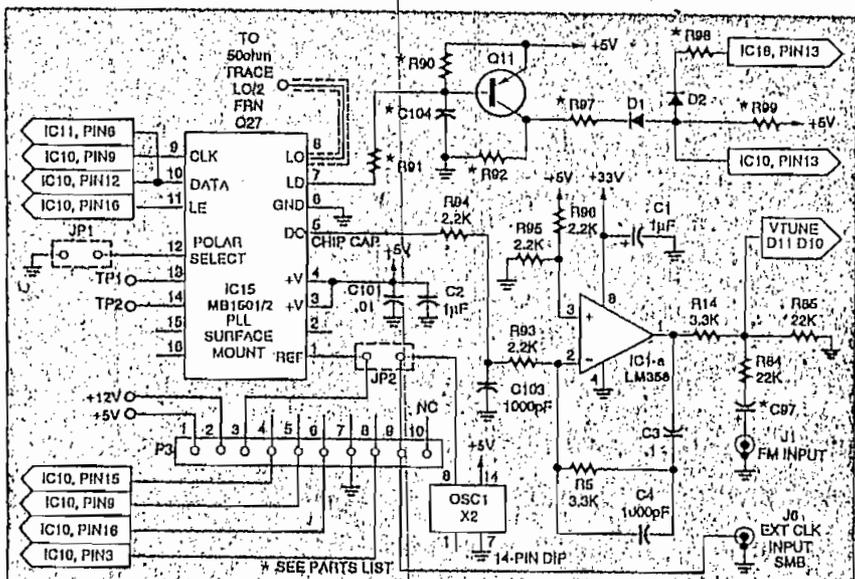


FIG. 11—THE FM INPUT (J1) IS AC-coupled and directly modulates the VCO control point.

The dividers

The two divide-by-2 dividers

are NEC584Gs, and the two divide-by-2/4/8 dividers are

NEC587Gs (see Figs. 9 and 10). The first NEC587G (IC16) acts as a prescaler, reducing the output frequency of the VCO by two. It becomes the 512-to-1024 divider when this band amplifier is switched on. This signal is split; one half is sent to the PLL and the other half is sent to the 512-to-1024 path. Splitting again occurs with half going to the 512-1024 amplifier and the other half to the 256-512 MHz divider. The output the 256-512 MHz divider is sent to the 256-512 MHz amplifier and the 32-256 MHz 2/4/8 divider. The 4-32 MHz band is handled by the last 2/4/8 divider.

The VCO

Three different voltage-controlled oscillators can be installed in the TG2000. The

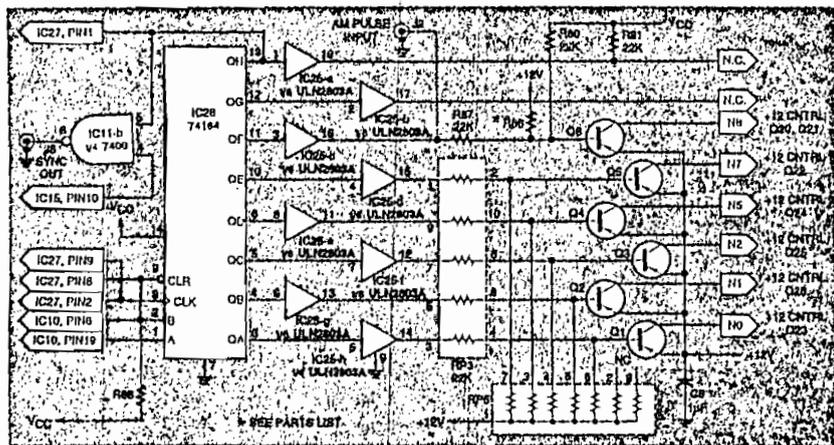


FIG. 12—THE AM PULSE MODULATION OPTION IS DC-coupled and accepts TTL control signals.

standard VCO is an ADC20010 made by Anadigics. The device is actually both a VCO and mixer combined in a single module, but just the VCO is used here.

Power/frequency calibration

The combination DAC/VVA automatically adjusts output power so that it's flat within ± 2 dB over the 10 to 2000 MHz band. The computer uses calibration data from the TGCAL0.DAT, TGCAL5.DAT, and TGCAL10.DAT files, which are sent to the DAC to linearize power output. Those files represent 0dBm, -5dBm, and -10dBm, respectively.

Readers will not be able to achieve the same calibration accuracy obtained with a factory-assembled product; a slight degradation will be seen because kit builders must depend on generic calibration files.

Frequency accuracy is determined primarily by the 4-MHz crystal TTL oscillator (OSC1 in Fig. 11). The standard oscillator is accurate to ± 50 ppm. More accurate frequency references improve the accuracy of the frequency output.

External modulation inputs

FM and optional pulse-AM modulation is obtained with RCA jacks. The AC-coupled FM Input (J1 in Fig. 11) modulates the VCO control point. Sensitivity decreases as DC control voltage increases. The maximum sensitivity is about

1V/100MHz. Both the amplitude and frequency content must be limited for this input to prevent the PLL from breaking lock.

The AM pulse modulation option (see Fig. 12) is DC-coupled and accepts TTL control signals. Maximum frequency of on/off toggling is limited by the response time of the amplifier voltage supply. A TTL sync-pulse output is available at RCA jack (J5). The sync pulse is produced at the end of a sweep and can be the external trigger for an oscilloscope. A wide-band oscilloscope (or an RF detector probe and a lower performing scope) can display these frequency response curves.

Digital interface

PC-bus and parallel-port interfaces are supported (see Fig. 13). The PC-bus interface comprises an address/strobe decoder made up of a 74688 (IC20) and two 74138s (IC18 and IC19). The PC data bus is bidirectionally buffered by the 74245 (IC21). Two eight-bit 74374 latches (IC9 and IC10) hold two eight-bit PC-bus bytes for control of the TG2000.

The simpler parallel interface has two octal 74244 buffers (IC4 and IC5) and part of a hex 7404 inverter (IC6). The connection to the parallel interface is provided by header P2. A cable terminated with a 26-pin IDC connector on one end and a DB-25 on the other is all that's needed to complete the interface.

When the TG2000 is connected to the parallel interface and external to the host PC, an external source of power is needed. A supply of +5 volts at 1 ampere, +12 volts at 0.5 ampere, and -12 volts at 0.1 ampere is recommended.

The remainder of the digital interface is made up of the 16-bit serial-to-parallel control register and the 8-bit serial-to-parallel converter for the DAC. The 16-bit control register switches the various dividers and RF amplifiers for the different bands on and off. Two 74164s (IC27 in Fig. 14 and IC28 in Fig. 12) convert a serial data stream to 16-bit parallel words for the control register. Two UNL2803A open-collector octal buffers (IC25 and IC26) provide the control voltages or currents needed for proper switching. Transistors Q1 to Q7 and Q9 and Q10 provide additional level and buffer current for the +5- and +12-volt switching. Transistor Q8 is not used so it is jumpered across its emitter and collector.

A 74164 (IC8) provides the serial-to-parallel conversion for the 8-bit DAC (IC7). The 8-bit DAC provides a 0- to 10-volt control signal to the VVA for AGC purposes, as already discussed.

An MC34063A +12- to 28-volt converter (IC3) provides the higher voltage needed to drive the VCO across its full range (see Fig. 15). It is essentially a small switching power supply capable of delivering a few milliamperes at 28 volts. The -1-volt bias for the LM358A (IC1-a) is derived from a voltage divider across or connected to the -12-volt supply. This bias allows the LM358A to operate at or near ground potential.

Software

Two programs provide software support for the TG2000. Version 4.0 of the Series-800 spectrum analyzers provides the tracking-generator capability. A separate DOS-based program called TG2000.EXE operates the TG2000 as a general-purpose signal generator. Only the TG2000.EXE software will be discussed here. It is

Continued on page 88

RF SIGNAL GENERATOR

continued from page 42

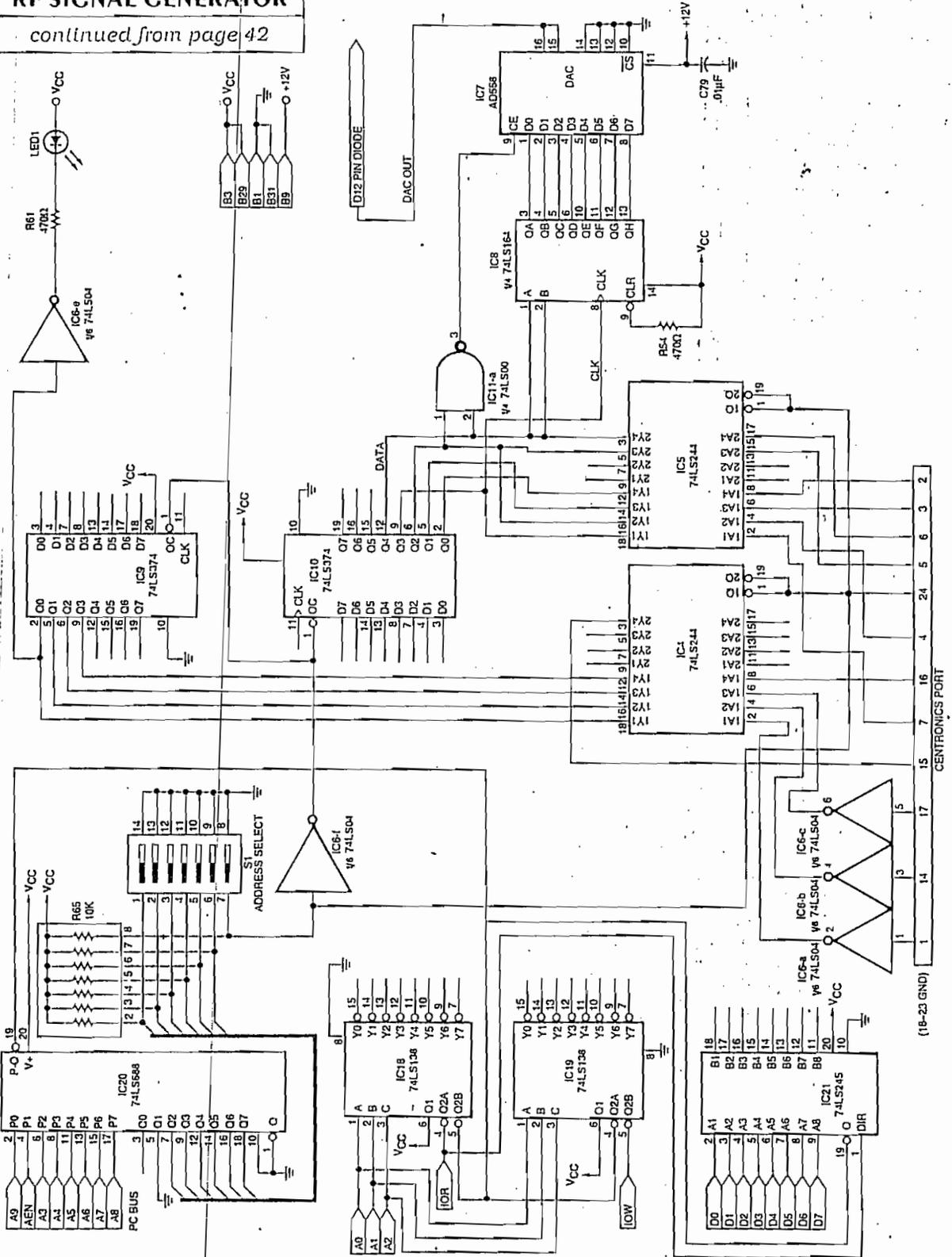


FIG. 13—THE PC-BUS INTERFACE comprises an address/strobe decoder made up of a 74688 (IC20) and two 74138s (IC18 and IC19). The PC data bus is bidirectionally buffered by the 74245 (IC21).

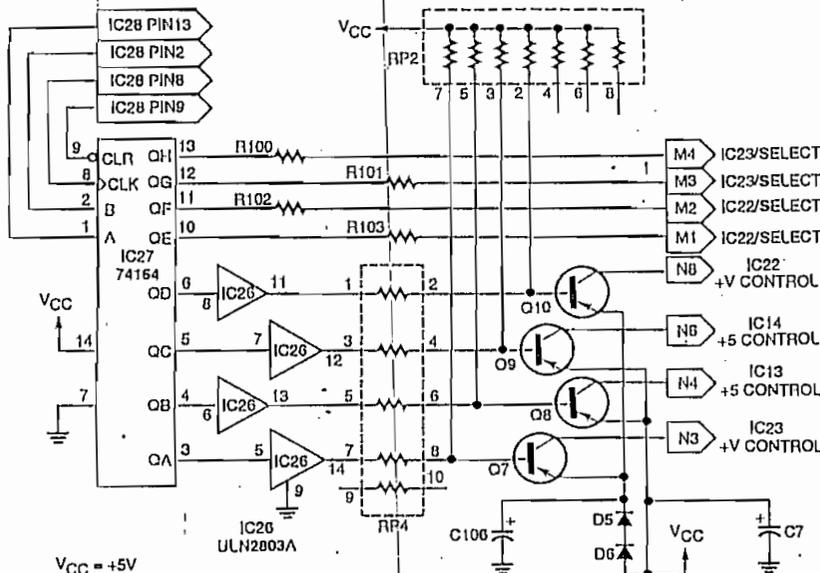


FIG. 14—TWO 74164s (IC27 AND IC28) convert a serial data stream to 16-bit parallel words for the control register.

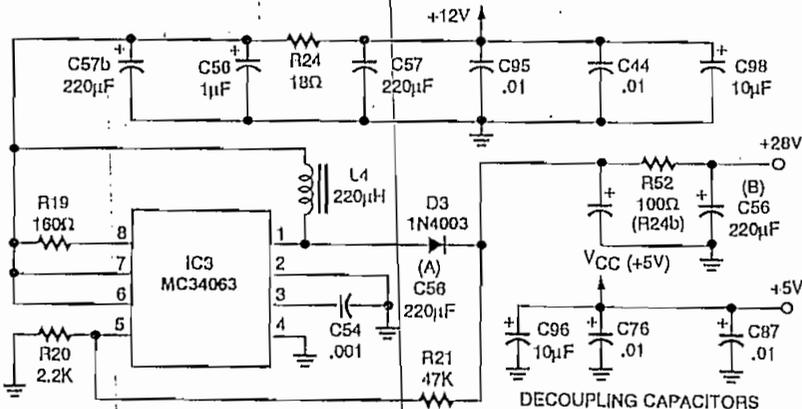


FIG. 15—THE MC34063A +12- TO 28-VOLT CONVERTER provides the higher voltage needed to drive the VCO across its full range.

available on the *Electronics Now BBS*, as part of a ZIP file called TG2000.ZIP.

The TG2000.EXE program contains a menu that allows the user to configure and control the generator. A complete user manual is also contained in the TG2000.ZIP file, so only the highlights will be covered here.

Before using the TG2000 software, you must choose the interface, and possibly the address where the generator will reside on the PC bus. Select the SETUP item from the main menu to configure the software for the chosen address/interface. The setup information is stored in a configuration.

Once the setup operation is complete, you can select either linear or log sweeps and the modulation function. The linear sweep has a constant step size for the entire band of the instrument. The log sweep increases its step size by a factor of two every time it crosses a band boundary. Both sweep functions sweep from a user-de-

ORDERING INFORMATION

Note: The following items are available from DKD Instruments, 1406 Parkhurst St., SImi Valley, CA 93065. (805) 581-5771:

- TG2000 kit—\$775.00
- TG2000 assembled and tested—\$1100.00

defined starting frequency to a user-defined stopping frequency. Sweep rate can be controlled by the DELAY entry.

The other modulation function is a frequency toggle. Two user-defined frequencies are toggled back and forth at a user-defined rate controlled by DELAY. This function can simulate FSK modulation, commonly used for digital data transmission.

Three diagnostic functions are available: Blink, Bitroll, and DAC Ramp. Blink will verify interface operation by toggling all output lines at a 1-Hz rate. Bitroll will debug the 16-bit control register. An alternating pattern of ones and zeros is sent to the 74164s. This results in a square wave at each control bit. DAC Ramp commands the DAC to produce a ramp voltage that will permit any missing codes to be spotted with a scope.

Next month's article will cover the building and test of the TG2000.

"YOUR FREE CATALOG KNOCKED MY SOCKS OFF"

We get that sort of comment all the time. People are impressed that our free Consumer Information Catalog lists so many free and low-cost government booklets. There are more than 200 in all, containing a wealth of valuable information.

Our free Catalog will very likely impress you, too. But first you have to get it. Just send your name and address to:

Consumer Information Center
Department KO
Pueblo,
Colorado
81009



A public service of this publication and the Consumer Information Center of the U. S. General Services Administration

ANEXO No. 6

```
OPEN "c:\dos\scri.sdd" FOR OUTPUT AS #1
```

```
FOR i = 0 TO 8191  
  IF i >= 2000 THEN GOTO uno  
  y = 127  
  GOTO graba
```

```
uno:   IF i <= 2011 THEN GOTO dos  
y = 127  
GOTO graba
```

```
dos: y = 255  
GOTO graba
```

```
graba:  
  z = INT(y)  
  PRINT #1, z  
  PRINT z, i
```

```
NEXT i
```

ANEXO No. 7

```

OPEN "c:\dos\vozis.sdd" FOR OUTPUT AS #1
CLS
k = 0
p = 0
q = 0
qa = 0
qad = 0
qadf = 0
qafa = 0
qafg = 0
qafc = 0
regresa:
    k = k + 1
    IF k >= 5 THEN GOTO segundo

    FOR i = 0 TO 351
        IF i >= 117 THEN GOTO uno
        Y = 127 * (SIN(2 * 3.1415926# * (i + 117) / 234) + 1)
        GOTO graba

uno:
    IF i >= 234 THEN GOTO doS
    Y = 127 * (.81 * SIN(2 * 3.1415926# * (i - 117) / 234) + 1)
    GOTO graba

doS:
    IF i >= 352 THEN GOTO fin
    Y = 127 * (.45 * SIN(2 * 3.1415926# * (i - 234) / 234) + 1)
    GOTO graba

graba:
    z = INT(Y)
        PRINT #1, z
        PRINT z,
    NEXT i

fin:
    GOTO regreso

segundo:
    p = p + 1
    IF p = 3 THEN GOTO tercero
    FOR w = 0 TO 351
        IF w >= 87 THEN GOTO unoa
        Y = 127 * (SIN(2 * 3.1415926# * (w + 117) / 234) + 1)
        GOTO grabados

unoa:
    IF w >= 283 THEN GOTO dosa
    Y = 127 * (.81 * SIN(2 * 3.1415926# * (w - 117) / 234) + 1)
    GOTO grabados

dosa:
    IF w >= 391 THEN GOTO fin
    Y = 127 * (.23 * SIN(2 * 3.1415926# * (w - 234) / 234) + 1)
    GOTO grabados

grabados:
    z = INT(Y)
        PRINT #1, z
        PRINT z,
    NEXT w

fina:
    GOTO segundo

tercero:
    q = q + 1
    IF q = 2 THEN GOTO cuarto
    FOR t = 0 TO 410
        IF t >= 118 THEN GOTO unoad

```

```

        Y = 127 * (.95 * SIN(2 * 3.1415926# * (t + 117) / 234) + 1)
        GOTO grabatres
unoad:  IF t >= 273 THEN GOTO dosad
        Y = 127 * (.79 * SIN(2 * 3.1415926# * (t - 118) / 310) + 1)
        GOTO grabatres
dosad:  IF t >= 411 THEN GOTO tercero
        Y = 127 * (.8 * SIN(2 * 3.1415926# * ((.25 / 137) * (t - 273) + .5)) + 1)
        GOTO grabatres
grabatres:
        z = INT(Y)

        PRINT #1, z

        PRINT z,

        NEXT t

        GOTO tercero

cuarto:
        qa = qa + 1
        IF qa = 2 THEN GOTO quinto
        FOR tq = 0 TO 410
            IF tq >= 118 THEN GOTO unoada
            Y = 127 * (.8 * SIN(2 * 3.1415926# * ((.25 / 117) * (tq) + .75)) + 1)
            GOTO grabacuatro

unoad:  IF tq >= 273 THEN GOTO dosada
        Y = 127 * (.65 * SIN(2 * 3.1415926# * ((.5 / 155) * (tq - 118))) + 1)
        GOTO grabacuatro
dosada: IF tq >= 411 THEN GOTO final
        Y = 127 * (.55 * SIN(2 * 3.1415926# * ((.25 / 137) * (tq - 273) + .5)) + 1)
        GOTO grabacuatro
grabacuatro:
        z = INT(Y)
        PRINT #1, z
        PRINT z,
        NEXT tq
        GOTO cuarto

quinto:
        qad = qad + 1
        IF qad = 2 THEN GOTO seis
        FOR tqa = 0 TO 410
            IF tqa >= 118 THEN GOTO unoadad
            Y = 127 * (.55 * SIN(2 * 3.1415926# * ((.25 / 117) * (tqa) + .75)) + 1)
            GOTO grabacinco

unoadad: IF tqa >= 273 THEN GOTO dosadad
        Y = 127 * (.55 * SIN(2 * 3.1415926# * ((.5 / 155) * (tqa - 118))) + 1)
        GOTO grabacinco

```

dosadad:

```
IF tqa >= 410 THEN GOTO quinto
Y = 127 * (.45 * SIN(2 * 3.1415926# * ((.25 / 137) * (tqa - 273) + .5)) + 1)
GOTO grabacinco
```

grabacinco:

```
z = INT(Y)
PRINT #1, z
PRINT z,
NEXT tqa
GOTO quinto
```

seis:

```
qaf = qaf + 1
IF qaf = 2 THEN GOTO siete
FOR tqf = 0 TO 410
  IF tqf >= 118 THEN GOTO unoadaf
Y = 127 * (.45 * SIN(2 * 3.1415926# * ((.25 / 117) * (tqf) + .75)) + 1)
GOTO grabaseis
```

unoadaf:

```
IF tqf >= 273 THEN GOTO dosadaf
Y = 127 * (.45 * SIN(2 * 3.1415926# * ((.5 / 155) * (tqf - 118))) + 1)
GOTO grabaseis
```

dosadaf:

```
IF tqf >= 410 THEN GOTO seis
Y = 127 * (.35 * SIN(2 * 3.1415926# * ((.25 / 137) * (tqf - 273) + .5)) + 1)
GOTO grabaseis
```

grabaseis:

```
z = INT(Y)
PRINT #1, z
PRINT z,
NEXT tqf
GOTO seis
```

siete:

```
qafa = qafa + 1
IF qafa = 2 THEN GOTO ocho
FOR tqfa = 0 TO 410
  IF tqfa >= 118 THEN GOTO unoadafa
Y = 127 * (.35 * SIN(2 * 3.1415926# * ((.25 / 117) * (tqfa) + .75)) + 1)
GOTO grabasiete
```

unoadafa:

```
IF tqfa >= 273 THEN GOTO dosadafa
Y = 127 * (.35 * SIN(2 * 3.1415926# * ((.5 / 155) * (tqfa - 118))) + 1)
GOTO grabasiete
```

dosadafa:

```
IF tqfa >= 410 THEN GOTO siete
Y = 127 * (.25 * SIN(2 * 3.1415926# * ((.25 / 137) * (tqfa - 273) + .5)) + 1)
GOTO grabasiete
```

grabasiete:

```
z = INT(Y)
PRINT #1, z
PRINT z,
NEXT tqfa
```

ocho:

```
    qafg = qafg + 1
    IF qafg = 2 THEN GOTO nueve
    FOR tqfg = 0 TO 410
        IF tqfg >= 118 THEN GOTO unoadafg
    Y = 127 * (.25 * SIN(2 * 3.1415926# * ((.25 / 117) * (tqfg) + .75)) + 1)
    GOTO grabaocho
```

unoadafg:

```
    IF tqfg >= 273 THEN GOTO dosadafg
    Y = 127 * (.25 * SIN(2 * 3.1415926# * ((.5 / 155) * (tqfg - 118))) + 1)
    GOTO grabaocho
```

dosadafg:

```
    IF tqfg >= 410 THEN GOTO ocho
    Y = 127 * (.15 * SIN(2 * 3.1415926# * ((.25 / 137) * (tqfg - 273) + .5)) + 1)
    GOTO grabaocho
```

grabaocho:

```
    z = INT(Y)
    PRINT #1, z
    PRINT z,
    NEXT tqfg
    GOTO ocho
```

nueve:

```
    qafc = qafc + 1
    IF qafc = 2 THEN GOTO diez
    FOR tqfc = 0 TO 410
        IF tqfc >= 118 THEN GOTO unoadafc
    Y = 127 * (.15 * SIN(2 * 3.1415926# * ((.25 / 117) * (tqfc) + .75)) + 1)
    GOTO grabanueve
```

unoadafc: IF tqfc >= 273 THEN GOTO dosadafc

```
    Y = 127 * (.15 * SIN(2 * 3.1415926# * ((.5 / 155) * (tqfc - 118))) + 1)
    GOTO grabanueve
```

dosadafc:

```
    IF tqfc >= 410 THEN GOTO nueve
    Y = 127 * (.05 * SIN(2 * 3.1415926# * ((.25 / 137) * (tqfc - 273) + .5)) + 1)
    GOTO grabanueve
```

grabanueve:

```
    z = INT(Y)
    PRINT #1, z
    PRINT z,
    NEXT tqfc
    GOTO nueve
```

diez:

```
    FOR yu = 0 TO 3216
    Y = 127 * (.05 * SIN(2 * 3.1415926# * ((yu + 225) / 300)) + 1)
    GOTO grabadiez
```

grabadiez:

```
    z = INT(Y)
    PRINT #1, z
    PRINT z,
```

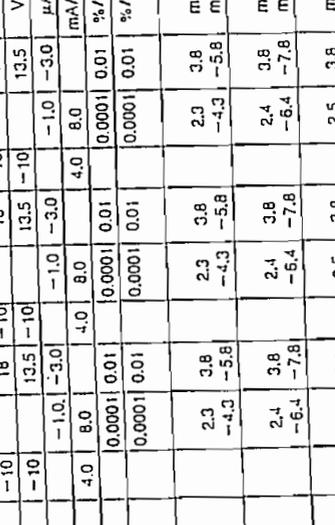
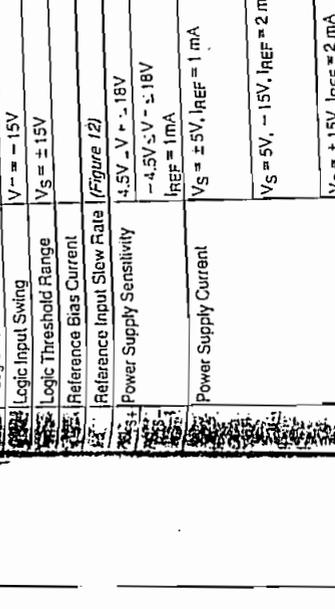
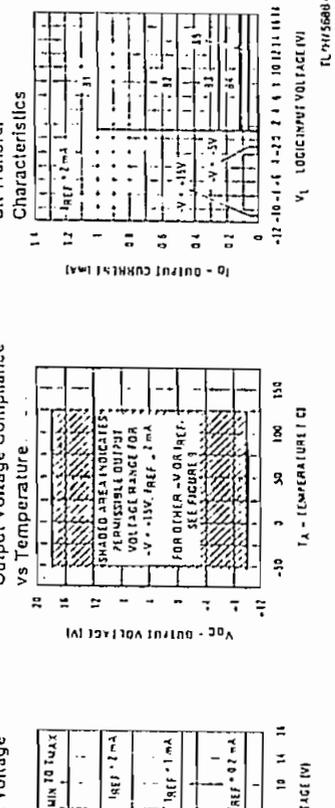
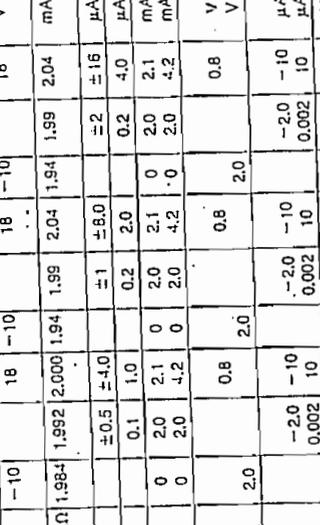
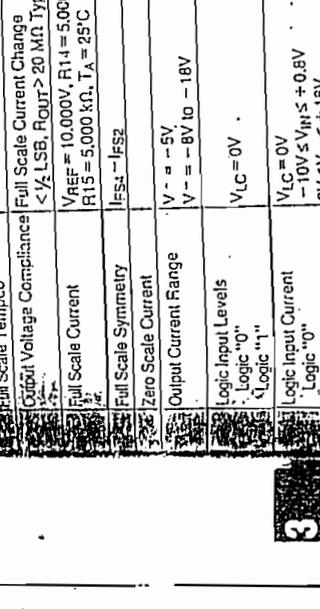
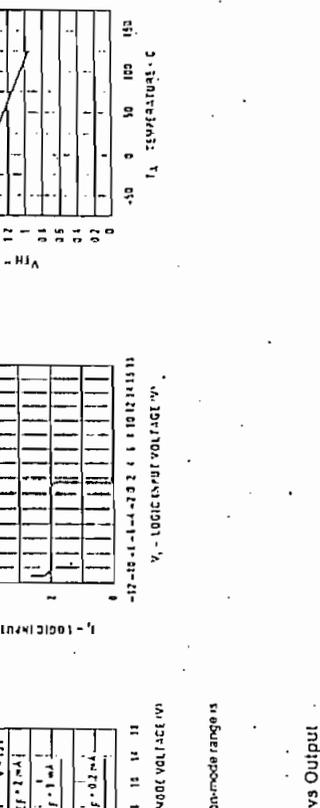
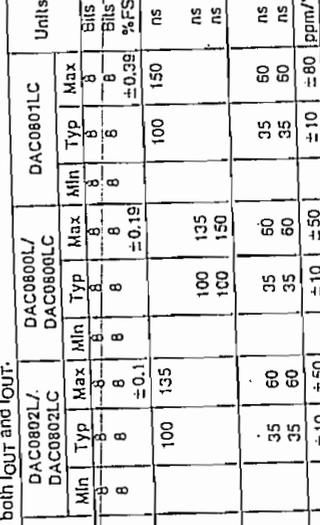
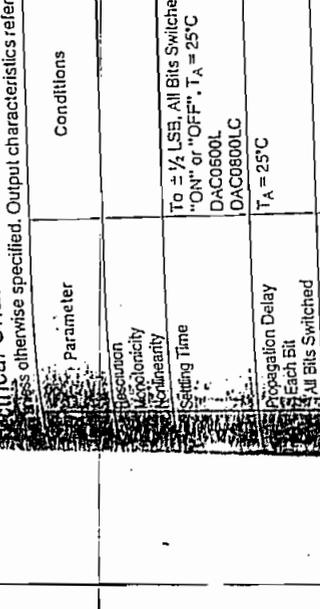
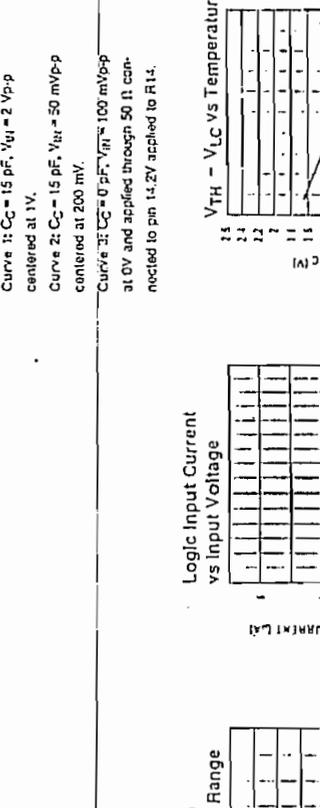
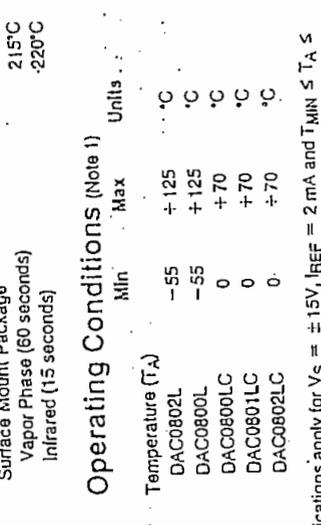
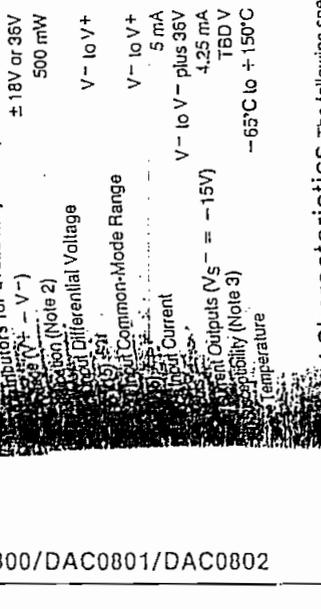
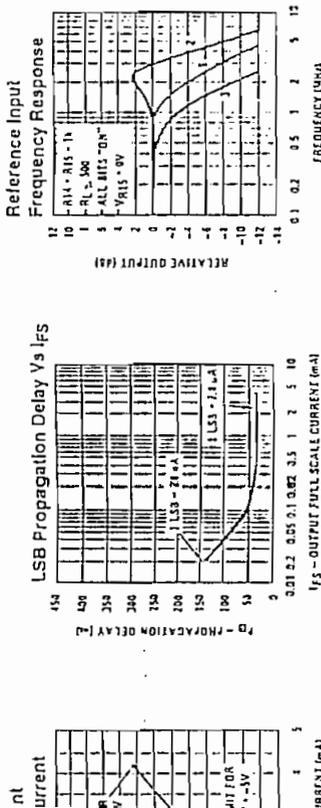
NEXT yu
GOTO final

final:

END

ANEXO No. 8

ance Characteristics



C0800/DAC0801/DAC0802

DAC0801/DAC0802

300 V

215°C

220°C

Dual-In-Line Package (ceramic)

Surface Mount Package

Vapor Phase (60 seconds)

Infrared (15 seconds)

Operating Conditions (Note 1)

Min	Max	Units
-55	+125	°C
-55	+125	°C
0	+70	°C
0	+70	°C
0	+70	°C

Temperature (T_A)

DAC0802L -55 +125 °C

DAC0800L -55 +125 °C

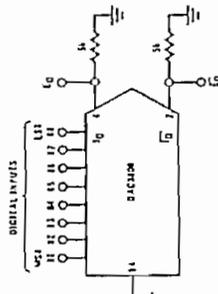
DAC0800LC 0 +70 °C

DAC0801LC 0 +70 °C

DAC0802LC 0 +70 °C

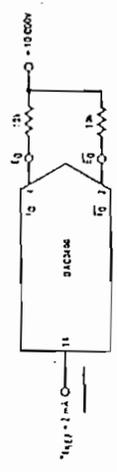
Electrical Characteristics The following specifications apply for V_S = ±15V, I_{REF} = 2 mA and T_{MIN} ≤ T_A ≤ T_{MAX} unless otherwise specified. Output characteristics refer to both I_{OUT} and I_{OUTR}.

Parameter	Conditions	DAC0802L/DAC0802LC			DAC0800L/DAC0800LC			DAC0801LC		
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max
Resolution	To ±1/2 LSB, All Bits Switched "ON" or "OFF", T _A = 25°C	8	8	8	8	8	8	8	8	8
Monotonicity	DAC0800LC	±0.1			±0.19					
Settling Time		100	135					100	150	ns
Propagation Delay	T _A = 25°C	35	60		35	60		35	60	ns
Delay per Bit		35	60		35	60		35	60	ns
Full Scale Tempco		±10	±50		±10	±50		±10	±50	ppm/°C
Output Voltage Compliance	Full Scale Current Change < 1/2 LSB, R _{OUT} > 20 M Ω , Typ	-10	18	-10	18	-10	18	-10	18	V
Full Scale Current	V _{REF} = 10,000V, R ₁₄ = 5,000 k Ω , R ₁₅ = 5,000 k Ω , T _A = 25°C	1,992	2,000	1,94	1,99	2,04	1,94	1,99	2,04	mA
Full Scale Symmetry	I _{FS4} - I _{FS2}	±0.5	±4.0		±1	±8.0		±2	±15	µA
Zero Scale Current		0.1	1.0		0.2	2.0		0.2	4.0	µA
Output Current Range	V _S = -5V V _S = -8V I _D = -18V	0	2.0	2.1	0	2.0	2.1	0	2.0	2.1
Logic Input Levels	V _{LC} = 0V		0.8	2.0		0.8	2.0		0.8	2.0
Logic "0"	V _{LC} = 0V	-2.0	-10		-2.0	-10		-2.0	-10	µA
Logic "1"	2V ≤ V _{IN} ≤ 5 + 0.8V	0.002	10		0.002	10		0.002	10	µA
Logic Input Swing	V _S = -15V	-10	18	-10	18	-10	18	-10	18	V
Logic Threshold Range	V _S = ±15V	-10	13.5	-10	13.5	-10	13.5	-10	13.5	V
Reference Bias Current		-1.0	-3.0		-1.0	-3.0		-1.0	-3.0	µA
Reference Input Slow Rate	(Figure 12)	4.0	8.0		4.0	8.0		4.0	8.0	mA/µs
Power Supply Sensitivity	4.5V ≤ V _S ≤ 18V	0.0001	0.01		0.00001	0.01		0.00001	0.01	%/%
	-4.5V ≤ V _S ≤ -18V	0.0001	0.01		0.00001	0.01		0.00001	0.01	%/%
Power Supply Current	I _{REF} = 1 mA	2.3	3.8		2.3	3.8		2.3	3.8	mA
	V _S = ±5V, I _{REF} = 1 mA	-4.3	-5.8		-4.3	-5.8		-4.3	-5.8	mA
	V _S = 5V, -15V, I _{REF} = 2 mA	2.4	3.8		2.4	3.8		2.4	3.8	mA
	V _S = ±15V, I _{REF} = 2 mA	-6.4	-7.8		-6.4	-7.8		-6.4	-7.8	mA
		2.5	3.8		2.5	3.8		2.5	3.8	mA
		-6.5	-7.8		-6.5	-7.8		-6.5	-7.8	mA



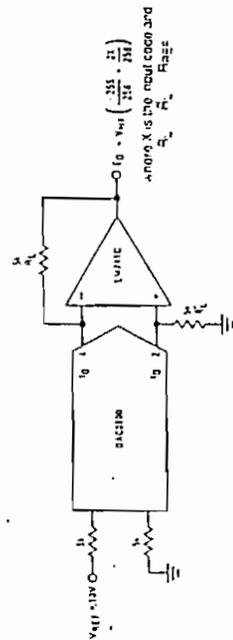
	B1	B2	B3	B4	B5	B6	B7	B8	I_0 mA	I_5 mA	E_0
Full Scale	1	1	1	1	1	1	1	1	1.992	0.000	-9.960
Half Scale	1	1	1	1	1	1	0	0	1.984	0.008	-9.920
Zero Scale	1	1	1	0	0	0	0	0	1.008	0.984	-9.920
Half Scale	1	0	0	0	0	0	0	0	1.000	0.992	-9.960
Half Scale	0	1	1	1	1	1	1	1	0.992	1.000	-5.000
Zero Scale	0	0	0	0	0	0	0	0	0.008	1.984	-0.040
Zero Scale	0	0	0	0	0	0	0	0	0.000	1.992	-0.000

FIGURE 6. Basic Unipolar Negative Operation (Note 4)



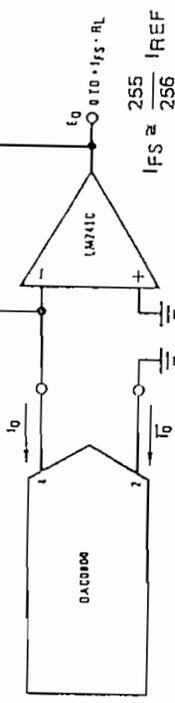
	B1	B2	B3	B4	B5	B6	B7	B8	E_0
Pos. Full Scale	1	1	1	1	1	1	1	1	-9.920
Pos. Full Scale - LSB	1	1	1	1	1	1	1	0	-9.840
Zero Scale + LSB	1	0	0	0	0	0	0	1	-0.080
Zero Scale	1	0	0	0	0	0	0	0	0.000
Zero Scale - LSB	0	1	1	1	1	1	1	1	-0.080
Neg. Full Scale + LSB	0	0	0	0	0	0	0	1	-9.920
Neg. Full Scale	0	0	0	0	0	0	0	0	-10.000

FIGURE 7. Basic Bipolar Output Operation (Note 4)



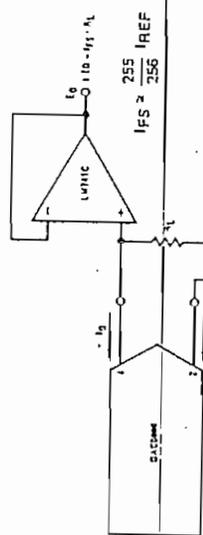
If $R_L = R_T$, within $\pm 0.05\%$, output is symmetrical about ground

	B1	B2	B3	B4	B5	B6	B7	B8	E_0
Pos. Full Scale	1	1	1	1	1	1	1	1	+9.960
Pos. Full Scale - LSB	1	1	1	1	1	1	1	0	+9.880
(+)Zero Scale	1	0	0	0	0	0	0	0	+0.040
(-)Zero Scale	0	1	1	1	1	1	1	1	-0.040
Neg. Full Scale + LSB	0	0	0	0	0	0	0	1	-9.880
Neg. Full Scale	0	0	0	0	0	0	0	0	-9.960



For complementary output (operation as negative logic DAC), connect inverting input of op amp to I_0 (pin 2), connect I_5 (pin 4) to ground.

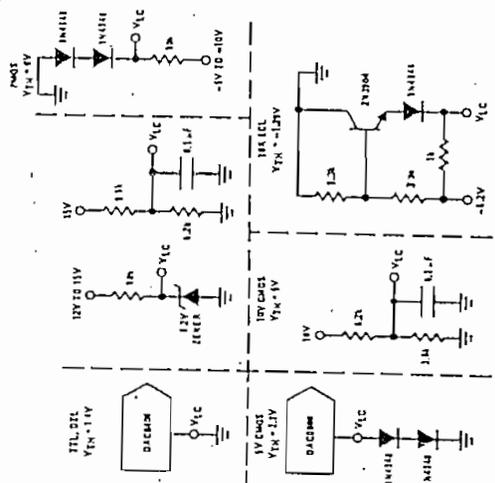
FIGURE 9. Positive Low Impedance Output Operation (Note 4)



For complementary output (operation as a negative logic DAC) connect non-inverting input of op amp to I_5 (pin 2); connect I_0 (pin 4) to ground.

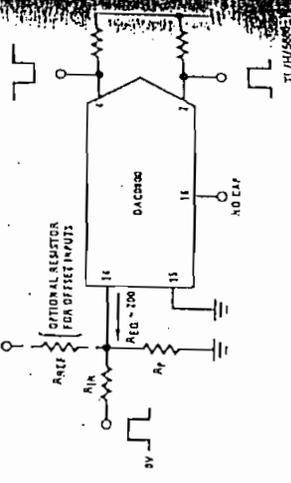
FIGURE 10. Negative Low Impedance Output Operation (Note 4)

$V_{TH} = V_{LC} + 1.4V$
15V CMOS, HTL, HNHL
 $V_{TH} = 7.5V$



Note. Do not exceed negative logic input range of DAC.

FIGURE 11. Interfacing with Various Logic Families

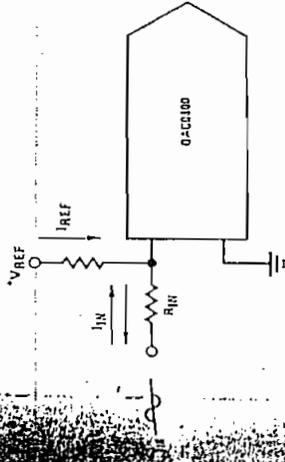


Typical values: $R_{IN} = 5k$, $V_{IN} = 10V$

FIGURE 12. Pulsed Reference Operation (Note 4)

Typical Applications (Continued)

(a) $|I_{REF}| \geq$ peak negative swing of I_{IN}



(b) $+V_{REF}$ must be above peak positive swing of V_{IN}

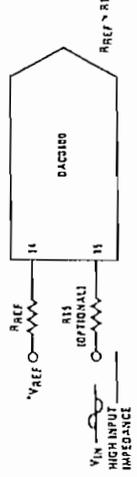


FIGURE 13: Accommodating Bipolar References (Note 4)

TUH/5866-11

TUH/5866-12

Note: For 1 μ s conversion time with 8-bit resolution and 7-bit accuracy, LM561 comparator replaces the LM519 and the reference current is doubled by reducing R1, R2 and R3 to 2.5 k Ω and R4 to 2 k Ω .

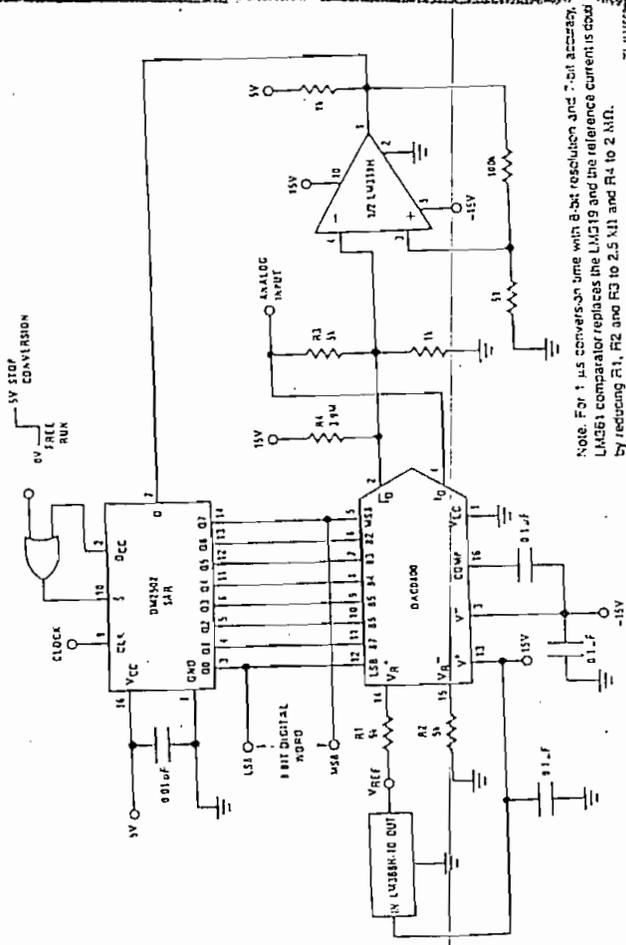


FIGURE 15: A Complete 2 μ s Conversion Time, 9-Bit A/D Converter (Note 4)

TUH/5866-11

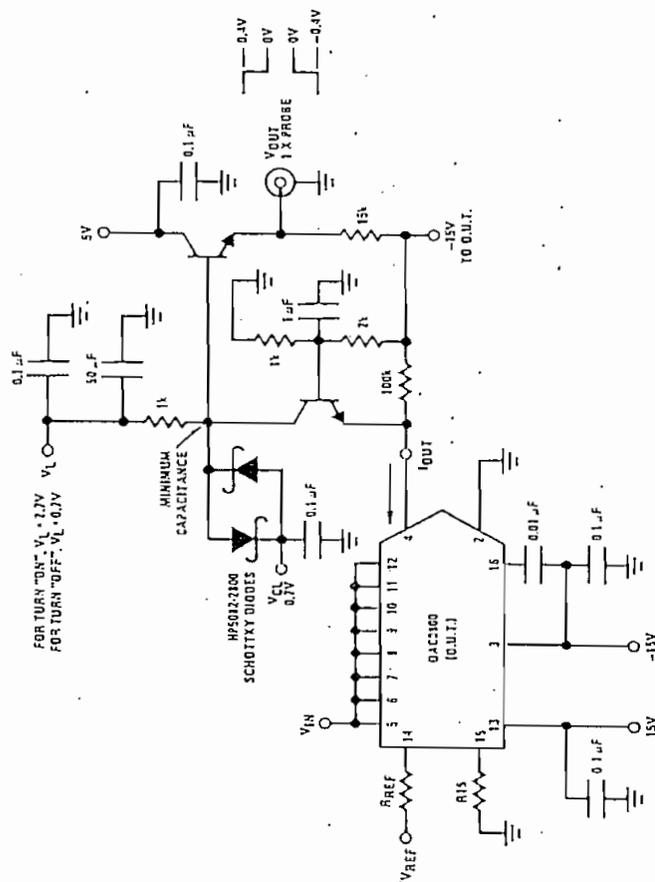


FIGURE 14: Settling Time Measurement (Note 4)

TUH/5866-7

Table 1: Signals and their Functions on PC Parallel (Printer) Port

D-Shell Connector Pin	Signal	Function	I/O at D-Shell Connector	Register	Register Bit	Inverted at Register	Centronics Connector Pin
1	-STB	Strobe D0 Thru D7	I/O	Control	0	Y	1
2	D0	Data Bit 0	O*	Data	0	N	2
3	D1	Data Bit 1	O*	Data	1	N	3
4	D2	Data Bit 2	O*	Data	2	N	4
5	D3	Data Bit 3	O*	Data	3	N	5
6	D4	Data Bit 4	O*	Data	4	N	6
7	D5	Data Bit 5	O*	Data	5	N	7
8	D6	Data Bit 6	O*	Data	6	N	8
9	D7	Data Bit 7	O*	Data	7	N	9
10	-ACK	Acknowledge (Triggers interrupt)	I	Status	6	N	10
11	BSY	Printer Busy	I	Status	7	Y	11
12	PE	Paper End (Out of Paper)	I	Status	5	N	12
13	SEL	Printer Selected (On-Line)	I	Status	4	N	13
14	-AUTOLF	Automatic Line Feed After CR	I/O	Control	1	Y	14
15	-ERR	Error	I	Status	3	N	32
16	-INIT	Initialize Printer	I/O	Control	2	N	31
17	-SELIN	Select Printer (Pulse On-Line)	I/O	Control	3	Y	36
18-25	GND	Ground	I	—	—	N	18, 19-30, 33
—	NC	No Connection at PC	—	—	—	—	15, 17, 18, 34, 35

*Some data ports are bidirectional (I/O).

disk uses interrupt 5, it's not available for the parallel port.

Although interrupt-driven software is fast, most parallel-port printer drivers don't use interrupts. This is partly due to a problem on the original parallel port and many of its imitators. On these ports, the interrupt-request line isn't latched. So if the pulse is short, the computer may not see it at all.

Table 1 summarizes the signals at the parallel port connector. These signals more or less follow the printer interface popularized by the Centronics Data Computer Corporation, although the 25-pin connector doesn't use all of the 36 lines in the original interface. Centronics was an early manufacturer of low-cost dot-matrix printers, and, although Centronics printers are no longer made, the name lives on in the connector, signals and pinout that the company made popular.

Although each signal has a name that suggests a particular function, you don't have to use the signals for their intended purposes. For example, you can use the paper-end signal for any type of input, not just to announce that a printer is out of paper.

Inputs and Outputs. You can access the parallel port through both MS-DOS and the ROM BIOS. Services 00, 01, and 02 of BIOS interrupt 17h

enable you to send a byte to the printer, initialize the printer and get printer status. DOS interrupt 21h, function 05, also writes a byte to the printer, and function 40h can direct a block of data to a parallel port. But for full access to all of the port's 17 signals, you need to read and write directly to the port. To do this, you ignore the DOS and BIOS functions and instead read from and write directly to the port's data, status and control registers shown in Fig. 1.

Data Lines. Data lines D0 through D7 are eight outputs that carry the data to be printed to the printer. For other applications, you can use the data lines as general-purpose outputs. To control the states of pins 2 through 9 on the parallel connector, you just write the desired data to the data register, whose address is the base address of the port. For example, to bring high D4 through D7 and low D0 through D3, you'd write F0h to the data register. In BASIC, you use the OUT statement, as follows:

```
OUT DataRegisterAddress, DataPortData
or, using the base address 3BCCh and data F0h,
OUT &h3BC,&hF0
```

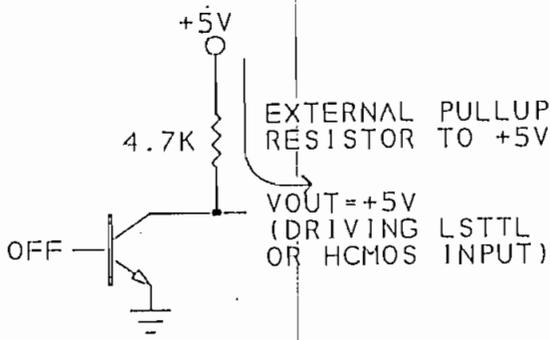
Listing 1 is a QuickBASIC program that brings each of a data-port pins high, one at a time.

Some parallel ports have bidirectional data lines, which you can use as inputs as well as outputs. Later, I'll explain how to determine if you have a bidirectional port and how to use a bidirectional data port for input.

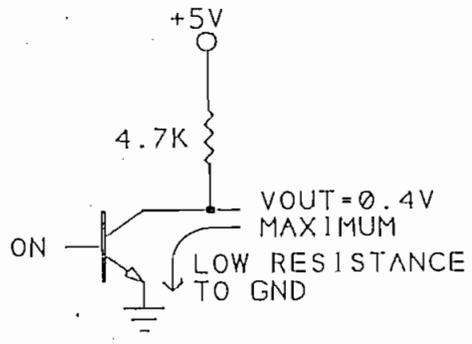
Status Lines. The status lines are five inputs that you read at a status register, which has an address of base address +1, or 3BDh for a port with a base address of 3BCh. The status register is read-only; writing to it has no effect. The five status lines use Bits 3 through 7 in the register, corresponding to pins 10 through 13 and 15 at the connector. Bits 0, 1 and 2 aren't used. To read the status inputs, you read the status port. In BASIC, you use the INP function:

```
PRINT INP(StatusPortAddress)
or instead use
PRINT INP(&h3BD)
```

However, the value you read doesn't exactly match the logic states at the connector. Bits 3 through 6 read normally. The bits in the status register match the logic states of their corresponding pins. Bit 7, however, contains the complement of the logic



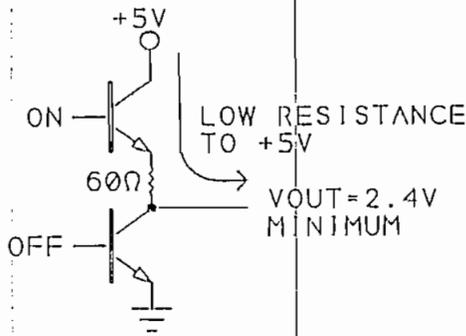
LOGIC HIGH (1)



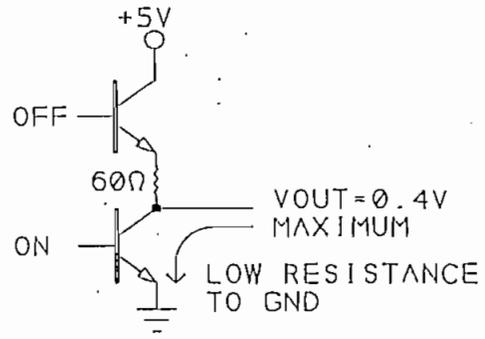
LOGIC LOW (0)

OK TO CONNECT TWO OPEN-COLLECTOR OUTPUTS TOGETHER.
ANY LOW OUTPUT CAUSES COMBINED OUTPUT TO BE LOW.

TOTEM-POLE OUTPUT



LOGIC HIGH (1)



LOGIC LOW (0)

DO NOT TIE TOTEM-POLE OUTPUTS TOGETHER.
IF ONE OUTPUT IS HIGH, AND THE OTHER LOW, DAMAGING
CURRENTS RESULT AS THE OUTPUTS FIGHT FOR CONTROL.

Fig. 2. At parallel ports that have open-collector or similar outputs, you can drive the input buffer with an external source. At ports that have totem-pole outputs, driving the input buffer with an external source has unpredictable results and may damage the circuits.

external signals at the data port. In fact, connecting outputs to the data port may destroy the port's circuits.

Beginning with its PS/2 model in 1987, IBM began to offer a bidirectional parallel port, the data lines of which can function as inputs as well as outputs. Other computer vendors have also added bidirectional ports. For example, computers that use Intel's 386SL or 486SL Superset have an I/O Subsystem chip that includes

bidirectional parallel port.

Typically, to use a bidirectional port for input, you must first run a setup program or move a jumper to configure the port for bidirectional operation. You then select input or output by setting or clearing a bit in the control register—usually Bit 5, but some ports use Bit 7.

If your computer doesn't have a bidirectional port, you can get one by adding an inexpensive expansion

card. Essential Data carries several, including STB's DSP550 (\$37), with one parallel port and two high-speed serial ports. Other suppliers have similar products. But many cards and computers, even newer ones, still don't have bidirectional data lines. Also, because all parallel ports have status and control inputs, any vendor can call its ports bidirectional, even if the eight-bit data port is output-only. So be sure you know what you're

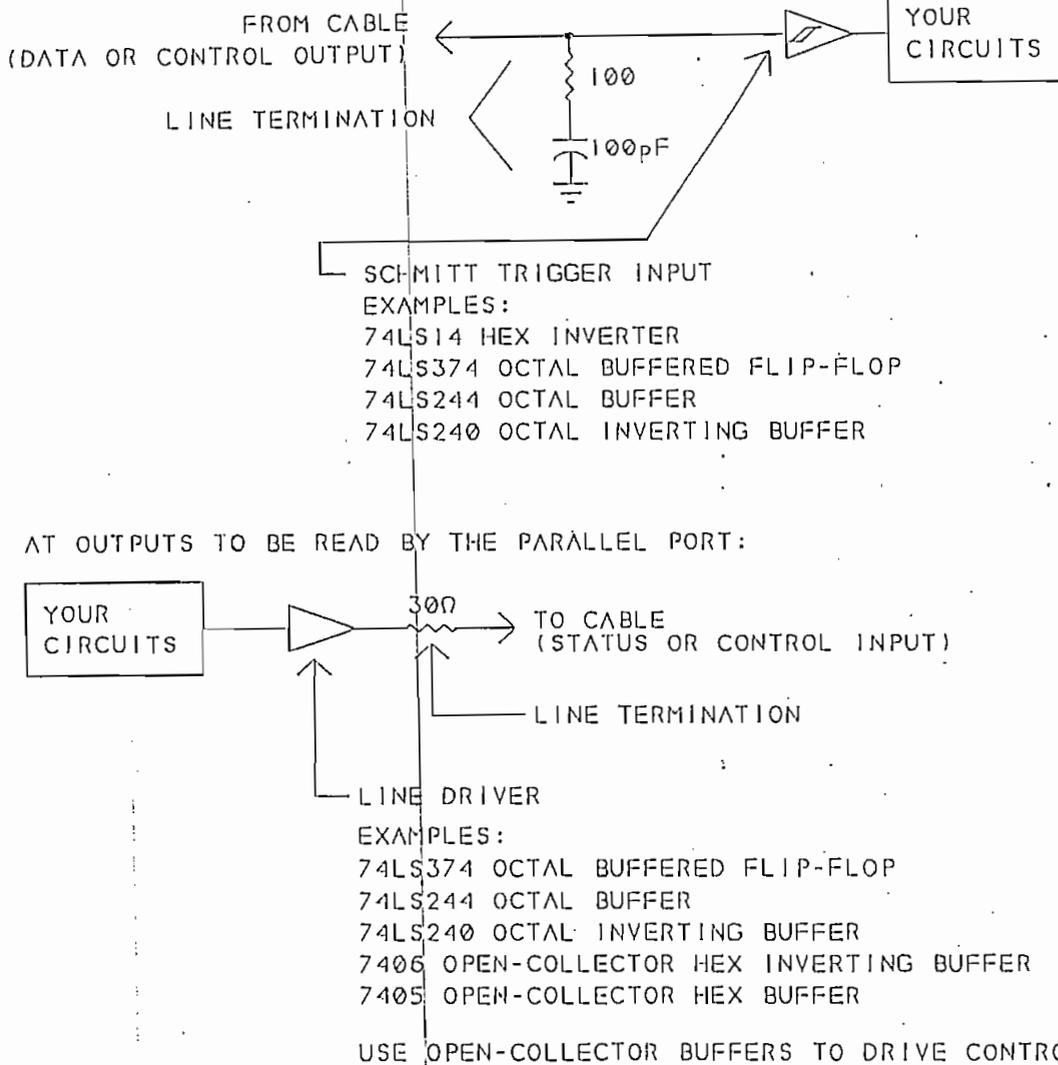


Fig. 3. Parallel-port interfaces.

such as the 74LS14 hex inverter. Figure 3 lists other options. At an ordinary logic gate, if a slowly changing input approaches the switching threshold, the output will jitter as the gate tries to decide whether the input is high or low. Schmitt triggers solve the problem by having two switching thresholds, one for low-to-high transitions and another, lower threshold for high-to-low transitions. For example, the output of a 74LS14 inverter won't go low until the input reaches 1.6 volts. After the output switches low, it won't go high again until the input drops to 0.8 volt. The 0.8-volt difference between the two thresholds is

called hysteresis and prevents the output from jittering if the input is changing slowly or if it has some noise or ringing on it. At outputs that drive parallel-port status, control or data inputs, use buffers or line drivers with strong output-drive ability. For example, on the 74LS240, low outputs can typically sink 24 mA per output, and high outputs can source 12 mA, compared with 8 and 0.4 mA for ordinary LS TTL. Figure 3 lists other line drivers. In general, CMOS drivers, including HCMOS, aren't as strong as LS TTL. Remember to use open-collector

devices to drive the control inputs and "output-only" data ports that you've determined can be used for input. Don't connect clock or control signals that aren't buffered—including the inputs and outputs of flip-flops, counters and shift registers—directly to the cable. These devices are sensitive to signal reflections. Instead, use buffers to isolate the signals. Some chips, like the 74LS374, have buffered outputs. If you aren't using the status inputs, tie them to +5 volts or ground through a 4,700-ohm resistor to hold them at a valid logic level. Another important issue for good signal quality is proper electrical ter-

Essential Data
PO Box 640963
San Jose, CA 95134
Tel.: 1-800-795-4756 or 408-955-0440;
fax: 408-955-0821

CIRCLE NO. 120 ON FREE INFORMATION CARD

Misco
One Misco Plaza
Holmdel, NJ 07733
1-800-333-5640
Tel.: 908-264-8324; fax: 908-888-9449

CIRCLE NO. 121 ON FREE INFORMATION CARD

National Semiconductor
2900 Semiconductor Dr.
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel.: 1-800-272-9959 or 408-721-5000

CIRCLE NO. 122 ON FREE INFORMATION CARD

Parallel Technologies
PO Box 7
Redmond, WA 98073-0007
Tel.: 206-869-1136; fax: 206-869-9767

CIRCLE NO. 123 ON FREE INFORMATION CARD

Personal Computing Tools
90 Industrial Park Rd.
Hingham, MA 02043
1-800-767-6728
Tel.: 617-740-0120; fax: 617-740-2728

CIRCLE NO. 124 ON FREE INFORMATION CARD

TAB Books
Blue Ridge Summit, PA 17294-0214
Tel.: 1-800-233-1128

CIRCLE NO. 125 ON FREE INFORMATION CARD

mination of the conductors in the cable. To understand termination, you again have to think of the cable as more than a simple series of connections between logic inputs and outputs. When a long conductor carries high-frequency signals, it has characteristics of a transmission line, which is a circuit that transfers energy from a source to a load. Because the sharp edges of digital transitions contain high-frequency components, digital circuits are considered high-frequency, even if the transmission rate (bits-per-second) is slow.

For the most efficient energy transfer from the source (output) to the load (input), the input impedance should match the characteristic impedance of the wire. Measuring the characteristic impedance of a conduc-

ture varies with the conductor's diameter, insulation type and distance between it and other conductors in the cable. It doesn't, however, change with the length of the conductor.

Cable manufacturers often specify the characteristic impedance of their products. A typical value for twisted-pair and ribbon cable is 100 ohms.

When load and conductor impedances match, all of the energy is transferred to the load. If they don't match, some of the energy reflects back to the source, and the reflections may bounce back and forth several times before they die out. This can cause delays or invalid logic levels at the receiver.

Designing terminations for the parallel port is especially difficult because the exact port circuits vary with the computer. Using the wrong termination can slow the data or consume excess power. Figure 3 shows typical examples of terminations at inputs and outputs.

At the driver, a series resistor provides impedance matching. The resis-

resistance of the output. Some parallel-port cards include a 30-ohm resistor in series with each data output. You can do the same for outputs that drive the status or control inputs.

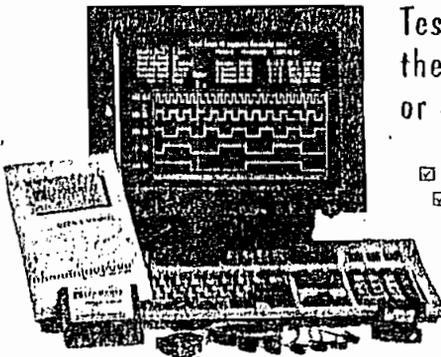
At the receiver, a resistor/capacitor combination network terminates the line. The resistor's value equals the characteristic impedance of the conductor, and the capacitor provides a low impedance during switching. Unlike some other terminations, this one is usable in both LSTTL and CMOS circuits.

Transmission-line theory, including line terminations, is a field of study unto itself. If you want to know more, *National Semiconductor's Interface: Line Drivers and Receivers* data book (#40032) has several application notes on the topic.

Transmitting Over Long Distances

If the parallel port's 15-foot limit isn't long enough for what you want to do,

The Real Logic Analyzer



Test your Logic circuits with the printer port of your IBM or compatible computer!

- 5 Input capture channels via printer port
- High Speed 64K input capture buffer
- Glitch capture and display
- Full triggering on any input pattern
- Automatic time base calibration
- 4 cursors measure time and frequency
- Save, print or export waveforms (PCX)

The Real Logic Analyzer is a software package that converts an IBM or compatible computer into a fully functional logic analyzer. Up to 5 waveforms can be monitored through the standard PC parallel printer port. The user connects a circuit to the port by making a simple cable or by using our optional cable with universal test clips. The software can capture 64K samples of data at speeds of up to 1.2uS (Depending on computer). The waveforms are displayed graphically and can be viewed at several zoom levels. The triggering may be set to any combination of high, low or Don't Care values and allows for adjustable pre and post trigger viewing. An automatic calibration routine assures accurate time and frequency measurements using 4 independent cursors. A continuous display mode along with our high speed graphics drivers, provide for an "Oscilloscope-type" of real time display. An optional Buffer which plugs directly to the printer port is available for monitoring high voltage signals.

LOGIXELL
ELECTRONICS
61 Piper Cr.
Kanata, Ontario
Canada K2K 2S9

Requires 286, or higher with EGA or VGA display.

LA20 Software Only \$79.95us
Software With Test Cable \$99.95us
BUFF05 Buffer \$39.95us

Tel: (613)599-7088
Fax: (613)599-7089



CIRCLE NO. 69 ON FREE INFORMATION CARD