

**EFFECTOS DE LA VARIACION DE LA FRECUENCIA
DE MUESTREO Y BITS DE CUANTIFICACION EN
LA DIGITALIZACION DE SEÑALES ANALOGICAS**

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

Departamento de Ingeniería Electrónica

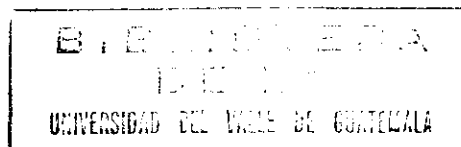
**EFFECTOS DE LA VARIACION DE LA FRECUENCIA
DE MUESTREO Y BITS DE CUANTIFICACION EN
LA DIGITALIZACION DE SEÑALES ANALOGICAS**

GONZALO PALAREA

Trabajo de graduación presentado para optar al grado
académico de Licenciado en Ingeniería Electrónica

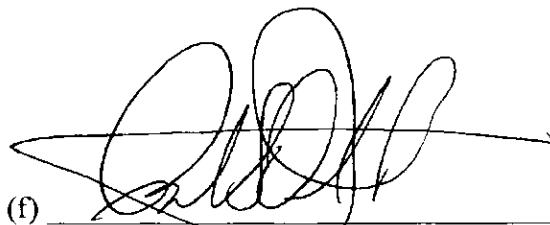
Guatemala

1995



Vo. Bo. :

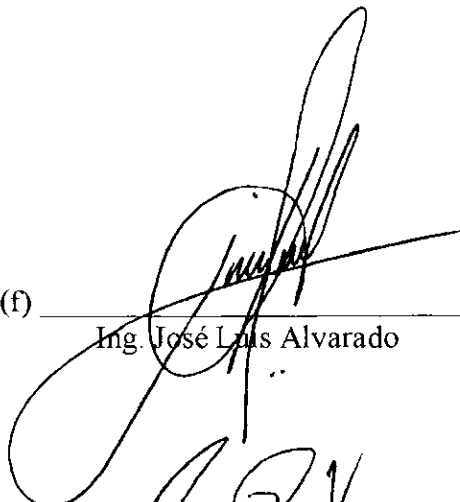
(f)



Ing. Robert Duke
Asesor

Tribunal:

(f)



Ing. José Luis Alvarado

(f)



Ing. Juan Pablo Rodríguez

(f)



Ing. Roberto Tejada

Fecha de aprobación: 31 de octubre de 1995

CONTENIDO

	Páginas
PREFACIO	VII
I. INTRODUCCION	1
II. MUESTREO	3
A. Teorema del muestreo	3
1. Señales pasabajos	4
2. Señales pasa banda	6
B. Señales no limitadas en banda, submuestreo	8
C. Reconstrucción de la señal	11
III. CUANTIFICACION	13
A. Cuantificación uniforme	16
B. Cuantificación no uniforme	18
1. Implementación	19
2. Ley A	20
3. Ley μ	20
IV. EL PCM	21
A. El sistema PCM	23
B. PCM normal	25
C. PCM diferencial (DPCM)	25
D. Modulación Delta Adaptativa (ADM)	27

	Páginas
V. ELABORACION DEL CIRCUITO	31
A. Elección de las características de operación	31
B. Elección de los circuitos integrados	32
C. Conexión de los circuitos integrados	34
1. Circuitería digital	34
2. Circuitería analógica	35
3. Fuente de alimentación	38
D. Interfaz con el usuario	38
1. Panel de control digital	38
2. Panel de control analógico	40
3. Control de alimentación	41
E. Programación	41
1. Selección del puntero de pila (stack pointer)	41
2. Elección de variables	41
3. Rutinas de inicialización	43
4. Caracteres para los 7 segmentos	44
5. Cálculo de los divisores de frecuencia para el CTC	44
6. Valores para la compresión digital	45
7. Rutinas de soporte	46
8. Rutinas de interrupción del CTC	47
9. Cálculo de la frecuencia máxima de muestreo	51

	Páginas
10. Rutinas de selección de opciones	51
11. Rutina de corrida según opciones	53
12. Programa principal	53
13. Compilación del programa y escritura en ROM	53
VI. RESULTADOS	55
VII. CONCLUSIONES Y RECOMENDACIONES	73
A. Conclusiones	73
B. Recomendaciones	74
VIII. BIBLIOGRAFIA	75
APENDICES	
A. Diagrama esquemático de la fuente de alimentación	77
B. Diagrama esquemático del circuito	79
C. Diagrama esquemático de los filtros	81
D. Listado del programa en BASIC para calcular divisores	83
E. Tabla de divisores de frecuencia obtenidos con el programa del apéndice D	85
F. Tabla de valores asignados en la compresión digital	87
G. Listado del programa en lenguaje ensamblador grabado en ROM	89
H. Cálculo del tiempo que dilatan las rutinas de interrupción	107
I. Diagrama esquemático del circuito para programar la EEPROM 2816	121

	Páginas
J. Listado del programa en BASIC para programar la ROM	123
ANEXO: Manual de operación del dispositivo	

LISTA DE FIGURAS

Figura		Página
2.1	Interruptor que conmuta con frecuencia f_m para muestrear una señal	3
2.2	Señal de entrada y señal muestreada	3
2.3	Señal de entrada, tren de pulsos periódico y su producto	4
2.4	Densidad espectral de señal de entrada y señal muestreada	5
2.5	Transformadas de Fourier de señal de entrada y señal muestreada	7
2.6	Condiciones necesarias de f_m para que no ocurra traslape	8
2.7	Frecuencia mínima de muestreo, dependiente de k	9
2.8	Muestreo de una señal no limitada en banda	9
2.9	Traslape de las componentes debido al filtro no ideal	10
2.10	Espectro de $e_m(t)$ utilizando filtro ideal y no ideal	12
3.1	Señal de entrada, de salida y límites de captura para algunos niveles de cuantificación	14
3.2	Nivel a_r , intervalo cuántico E , y el rango de captura	15
3.3	Tres métodos para lograr la cuantificación no uniforme	19
4.1	Señal muestreada, mostrando los niveles de cuantificación y los códigos	22
4.2	Representación binaria por medio de pulsos y de voltajes	22
4.3	Sistema de comunicaciones PCM	23
4.4	Señal de entrada y valor acumulado para un sistema con modulación delta	26

Figura		Página
4.5	Comparación entre las modulaciones delta y delta adaptativa	28
5.1	Panel de control digital	39
5.2	Panel de control analógico	40
5.3	Control de alimentación	41
5.4	Valores utilizados en la compresión digital	45
6.1	Señal senoidal de frecuencia 1 kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando PCM a 8 bits, sin filtro de salida	57
6.2	Señal senoidal de frecuencia 1 kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando PCM a 8 bits, con filtro de salida a 1.5 kHz	58
6.3	Señal senoidal de frecuencia 5 kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando PCM a 8 bits, con filtro de salida a 4 kHz	59
6.4	Señal senoidal de frecuencia 11 kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando PCM a 8 bits, con filtro de salida a 4 kHz	60
6.5	Señal senoidal de frecuencia 3.5 kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando PCM a 8 bits, sin filtro de salida	61
6.6	Señal senoidal de frecuencia 3.5 kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando PCM a 8 bits, con filtro de salida a 4 kHz	62
6.7	Señal triangular de frecuencia 500 Hz, 4Vpp, muestreada con $f_m=12\text{kHz}$, utilizando PCM a 3 bits, sin filtro de salida	63
6.8	Señal triangular de frecuencia 500 Hz, 4Vpp, muestreada con $f_m=12\text{kHz}$, utilizando PCM a 6 bits, sin filtro de salida	64
6.9	Señal triangular de frecuencia 400 Hz, 4Vpp, muestreada con $f_m=12\text{kHz}$, utilizando PCM a 8 bits, sin filtro de salida	65
6.10	Señal cuadrada de frecuencia 500 Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando DPCM a 1 bit, sin filtro de salida	66

Figura		Página
6.11	Señal cuadrada de frecuencia 500 Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando DPCM a 4 bits, sin filtro de salida	67
6.12	Señal cuadrada de frecuencia 500 Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando ADPCM a 1 bit, sin filtro de salida	68
6.13	Señal senoidal de frecuencia 400 Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando ADPCM a 1 bit, sin filtro de salida	69
6.14	Señal senoidal de frecuencia 400 Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando ADPCM a 1 bit, con filtro de salida a 400 Hz	70
6.15	Señal triangular de frecuencia 300 Hz, 4.4Vpp, muestreada con $f_m=20\text{kHz}$, utilizando PCM con expansión a 5 bits, sin filtro de salida	71

LISTA DE TABLAS

Tabla		Página
5.1	Direccionamiento de los puertos	35
5.2	Agrupación de las características de operación en opciones	39
5.3	Bits asignados a los segmentos de los dígitos	40
5.4	Descripción de las variables	42
5.5	Rutinas de inicialización	43
5.6	Tabla de caracteres	44
5.7	Rutinas de soporte	46
5.8	Rutinas de interrupción del CTC	47
5.9	Rutinas de selección de opciones	52

PREFACIO

El objetivo principal del presente trabajo fue diseñar e implementar un circuito que permita estudiar la digitalización de señales de una forma práctica, y poder observar el efecto que causa en la digitalización la variación de los distintos parámetros que este proceso involucra, principalmente cantidad de bits que se utiliza y frecuencia de muestreo.

En los capítulos II, III y IV se presenta, en forma breve, la teoría relacionada con el muestreo, la reconstrucción, la cuantificación y la transmisión de las señales digitalizadas.

En el capítulo V se explican los pasos seguidos en el diseño del circuito y al final del trabajo se adjunta el manual para la operación del aparato que se construyó.

El dispositivo diseñado, además de permitir estudiar los efectos de la variación de la frecuencia de muestreo y los bits de cuantificación, permite implementar distintos tipos de codificación por modulación de pulsos. Por este motivo se implementaron PCM, DPCM, ADPCM y PCM con compresión digital de 8 a 5 bits.

Para poder evaluar el efecto de la variación de un parámetro, se consideró necesario tener la capacidad de grabar la muestra de una señal y después poder reproducirla con cualquiera de los parámetros posibles. Así, la misma muestra se puede reproducir infinidad de veces, variando cada uno de los parámetros cada vez y apreciar el efecto.



I. INTRODUCCION

Las comunicaciones digitales han cobrado gran importancia en el campo de las telecomunicaciones, pues presenta grandes ventajas sobre las comunicaciones analógicas. Sin embargo, las señales con las que el humano interacciona son todas analógicas, con lo cual para poder ser transmitidas en un canal de comunicaciones digital, deben primero ser digitalizadas, y al final del canal, ser reconstruidas. El proceso de la digitalización, transmisión y reconstrucción de la señal involucra varios parámetros, los cuales afectan cada uno de distinta forma el resultado obtenido.

Cuando se estudia la digitalización de señales, se hace de una forma meramente teórica, sin poder observar prácticamente los efectos que causan la variación de los parámetros que ésta involucra. En el presente trabajo se diseñó un circuito tal que permita probar en el laboratorio lo que se estudia en la teoría.



II. MUESTREO

El proceso del muestreo se puede ilustrar de forma simple con un interruptor (figura 2.1) que se mueve periódicamente entre dos contactos con un período T_m (y una

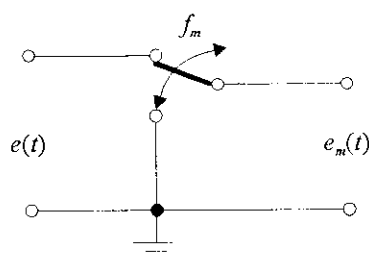


Figura 2.1. Interruptor que conmuta con frecuencia f_m para muestrear una señal.

frecuencia $f_m=1/T_m$). El interruptor permanece en la terminal de entrada durante T segundos y se conecta a tierra durante el resto del período T_m . De esta forma, la salida $e_m(t)$ consiste de cortos segmentos de la señal de entrada $e(t)$ (figura 2.2).

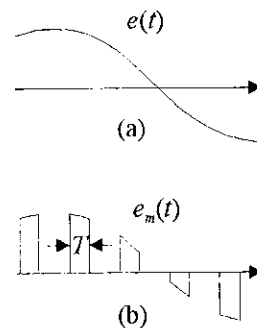


Figura 2.2. (a) Señal de entrada. (b) La misma señal, muestreada.

Esta operación se denomina corte unipolar y la salida recibe el nombre de señal muestreada. Se puede notar que este proceso es equivalente a modular en amplitud un tren de pulsos, por lo cual recibe también el nombre de modulación por amplitud de pulsos (PAM, por sus siglas en inglés). La relación T/T_m se denomina coeficiente de actividad.

A. Teorema del muestreo

Si de una señal analógica se toman periódicamente muestras de su valor, se puede obtener suficiente información para poder reconstruir la señal a partir de ellas. Esto es, de forma bastante simple, el teorema del muestreo. Sin embargo, para poder reconstruir la señal, se tiene que cumplir con algunos requisitos, los cuales dependen básicamente de las frecuencias que comprende la señal a muestrear. Todas aquellas señales limitadas en banda pueden ser definidas con un número finito de muestras. Para demostrar el teorema

del muestreo, es necesario probar que la señal puede ser reconstruida a partir de sus muestras.

1. Señales pasabajos. Una señal pasabajos es aquella cuyas componentes de frecuencia están todas comprendidas entre 0 y una frecuencia máxima f_A .

La señal $e(t)$ (supuesta pasabajos) se muestra en la figura 2.3a. Se tiene un tren de pulsos periódico $M(t)$, de amplitud unitaria y frecuencia f_m , que se muestra en la figura 2.3b. Los pulsos son arbitrariamente angostos, teniendo un ancho dt . Las dos señales se multiplican, de tal forma que se obtiene $e_m(t)=e(t)M(t)$. Con este producto (figura 2.3c) se obtiene la señal $e(t)$ muestreada durante la ocurrencia de cada pulso. Este proceso es equivalente al del interruptor que se había discutido con anterioridad.

La señal $M(t)$ es periódica con una frecuencia f_m , y por lo tanto tiene la expansión de Fourier siguiente:

$$M(t) = f_m dt + 2f_m dt (\cos 2\pi f_m t + \cos 2 \times 2\pi f_m t + \dots) \quad (\text{II-1})$$

y la señal $e_m(t)=e(t)M(t)$ se puede escribir como:

$$e_m(t) = e(t)f_m dt + 2f_m dt (e(t)\cos 2\pi f_m t + e(t)\cos 2 \times 2\pi f_m t + \dots) \quad (\text{II-2})$$

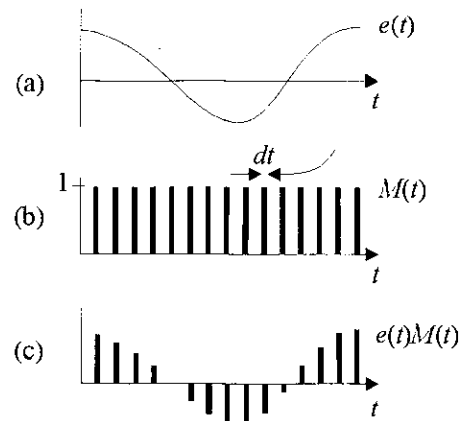


Figura 2.3. (a) Señal de entrada, $e(t)$.
 (b) Tren de pulsos periódico, $M(t)$.
 (c) Producto $e_m(t)=e(t)M(t)$.

De la ecuación (II-2) se puede observar que el primer término de la serie es, aparte de un factor constante, la señal de entrada $e(t)$ misma. El segundo término es, también aparte de un factor constante, el producto de $e(t)$ por una sinusoidal de frecuencia f_m . El tercer término contiene el producto de $e(t)$ por una sinusoidal de frecuencia $2f_m$ (también multiplicado por un factor constante), y así sucesivamente con $3f_m, 4f_m \dots$

Si suponemos que la señal $e(t)$ tiene una densidad espectral $E(j\omega) = \mathcal{F}[e(t)]$ como la que se muestra en la figura 2.4a (obsérvese que la señal es pasabajos con una frecuencia máxima f_A), la señal muestreada tendrá un espectro como el que se presenta en la figura 2.4b. En esta figura se puede observar que el espectro del primer término de la ecuación (II-2) se extiende desde 0 hasta f_A . El espectro del segundo

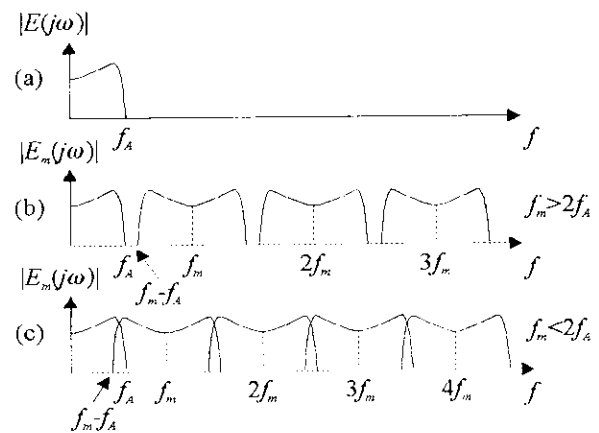


Figura 2.4. (a) Densidad espectral de $e(t)$.
(b) Densidad espectral de $e_m(t)$ para $f_m > 2f_A$.
(c) Densidad espectral de $e_m(t)$ para $f_m < 2f_A$.

término es simétrico respecto a f_m y se extiende desde $f_m - f_A$ hasta $f_m + f_A$; el del tercero es simétrico respecto a $2f_m$ y se extiende desde $2f_m - f_A$ hasta $2f_m + f_A$, y así sucesivamente. Con esta figura se puede observar que los espectros de los términos de la ecuación (II-2) se pueden espaciar o acercar mediante la variación de la frecuencia de muestreo f_m .

Mientras f_m sea mayor o igual que $2f_A$, los espectros de los términos se mantendrán separados. Entonces, si la señal muestreada $e_m(t)$ fuera filtrada con un filtro ideal pasabajos con una frecuencia de corte entre f_A y $f_m/2$, a la salida del filtro se obtendría la señal original $e(t)$ (multiplicada por un factor constante) y nada más. Sin embargo, si f_m fuera menor que $2f_A$, los espectros de los términos se traslaparían (figura 2.4c), imposibilitando así la recuperación de la señal original. De esta forma se ha encontrado que la condición

para poder reconstruir una señal pasabajos muestreada radica en la frecuencia de muestreo, la cual debe ser mayor que el doble de la frecuencia más alta de la señal $e(t)$:

$$f_m \geq 2f_A \quad (\text{II-3})$$

Dicho de otra manera, la frecuencia de muestreo mínima necesaria para poder reconstruir la señal es $2f_A$, la cual se conoce como *frecuencia de Nyquist*.

Un caso especialmente interesante es el de muestrear una señal sinusoidal de frecuencia f_A . Utilizando un filtro ideal con el corte en f_A , existirá cierta ambigüedad sobre si la frecuencia de la señal está dentro o fuera de la banda de paso del filtro, dado que toda la potencia de la señal está concentrada precisamente en la frecuencia de corte del mismo. Para eliminar esta ambigüedad, se requiere que la frecuencia de corte del filtro sea mayor que f_A , y $f_m > 2f_A$ en vez de $f_m \geq 2f_A$. Para ilustrar la necesidad de esta condición, se toma $f_m = 2f_A$, y la primera muestra se hace en un cruce por cero de la sinusoidal. De esta forma, todas las muestras tomadas posteriormente tendrán amplitud cero. Esta situación se evita requiriendo que $f_m > 2f_A$.

2. Señales pasa banda. Para una señal $e(t)$ cuya máxima frecuencia es f_A , la frecuencia de muestreo f_m debe ser mayor que $2f_A$, únicamente si la menor frecuencia de la señal, f_B , es cero. Un caso más general es aquél en el cual $f_B \neq 0$, y por lo tanto la señal tiene un ancho de banda $B = f_A - f_B$. En la figura 2.5 se puede observar la transformada de Fourier de una señal $e(t)$ de este tipo. Llamaremos *LP* a la parte de frecuencias positivas y *LN* a la parte de frecuencias negativas. Al muestrear la señal, obtendremos patrones corridos de este espectro centrados en los múltiplos de la frecuencia de muestreo f_m , además del espectro original. Hay que tener cuidado entonces que los patrones corridos no se traslapen, y esto se logra eligiendo una frecuencia de muestreo f_m adecuada. Si se elige un valor mínimo de f_m como $f_m = 2(f_A - f_B) = 2B$, los patrones corridos de *LP* no se van

a traslapar con LP mismo. Los patrones corridos hacia la izquierda de LN tampoco pueden traslaparse con LP . Sin embargo, los patrones corridos hacia la derecha de LN sí pueden ocasionar un traslape sobre LP . En la figura 2.5b se muestran los corrimientos hacia la derecha de LN debidos a la N y $(N-1)$ armónicas de la señal de muestreo $M(t)$. De esta figura se puede observar que para que no haya traslapes es necesario que:

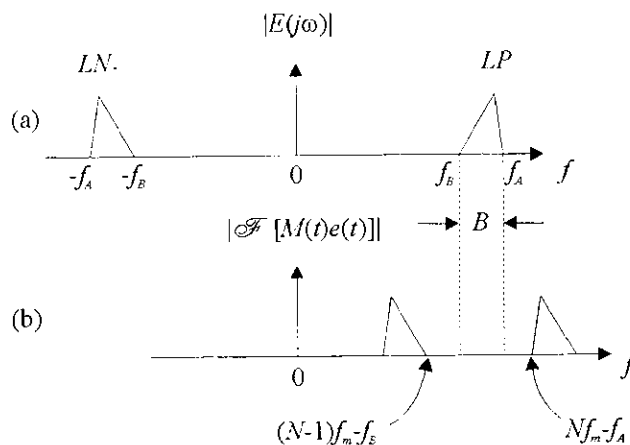


Figura 2.5. (a) Transformada de Fourier de $e(t)$.

(b) Transformada de Fourier de $e_m(t)$.

$$(N-1)f_m - f_B \leq f_B \quad (\text{II-4})$$

$$y \quad Nf_m - f_A \geq f_A \quad (\text{II-5})$$

Sustituyendo $B = f_A - f_B$,

$$(N-1)f_m \leq 2(f_A - B) \quad (\text{II-6})$$

$$y \quad f_m \geq 2 \frac{f_A}{N} \quad (\text{II-7})$$

Si se define $k = f_A/B$, las ecuaciones (II-6) y (II-7) quedan así:

$$f_m \leq 2B \frac{(k-1)}{(N-1)} \quad (\text{II-8})$$

$$y \quad f_m \geq 2B \frac{k}{N} \quad (\text{II-9})$$

en las cuales $k \geq N$ pues $f_m \geq 2B$. Las ecuaciones (II-8) y (II-9) dan las condiciones necesarias para evitar un traslape sobre LP . Dada la simetría del espectro inicial y de los corrimientos, estas condiciones nos aseguran también que no habrá traslape sobre LN .

Las ecuaciones (II-8) y (II-9) se han graficado en la figura 2.6 para varios valores de N . Las regiones sombreadas son aquellas donde las condiciones se satisfacen, mientras

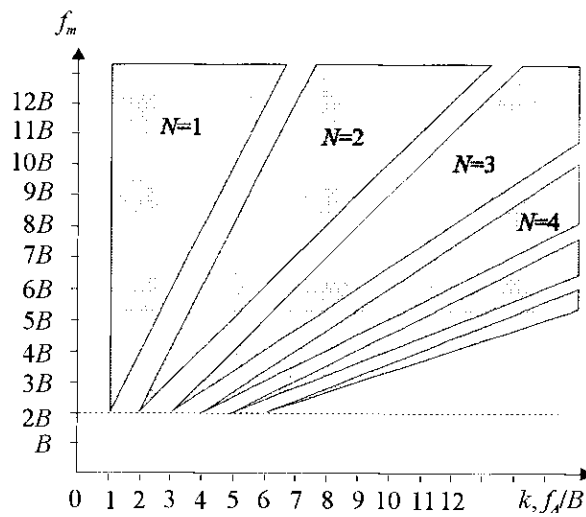


Figura 2.6. En las regiones sombreadas se cumplen las condiciones de f_m para que no ocurra traslape.

que en las no sombreadas no se satisfacen y ocurre traslape. De esta gráfica se puede observar que la frecuencia mínima de muestreo viene dada por:

$$f_m = 2 \frac{f_A}{n} \quad (\text{II-10})$$

donde n es la parte entera de k . En la figura 2.7 se observa la representación gráfica de (II-10); de esta gráfica se puede observar que a medida que aumenta k (la relación entre f_A y B) la frecuencia mínima de muestreo tiende a $2B$.

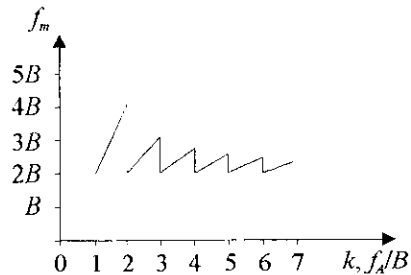


Figura 2.7. Frecuencia mínima de muestreo, dependiente de k .

B. Señales no limitadas en banda, submuestreo

Hasta aquí se ha supuesto que las señales a muestrear están limitadas en banda. Sin embargo esto no se cumple para señales reales. Una señal con un espectro como el que se muestra en la figura 2.8a puede considerarse prácticamente limitada en banda (pasabajos) hasta W si la información contenida en las componentes arriba de W carece de importancia con relación a las componentes abajo de W . Sin embargo, si se utiliza una frecuencia de muestreo $f_m=2W$, se producirá un traslape de los espectros produciendo una distorsión alíneaal, con lo cual, al reconstruir la señal, aparecerán

componentes de frecuencia que originalmente no existían (figura 2.8b). Este tipo de distorsión aparece siempre que alguna componente del espectro de la señal es submuestreada y recibe el nombre de solapamiento espectral.

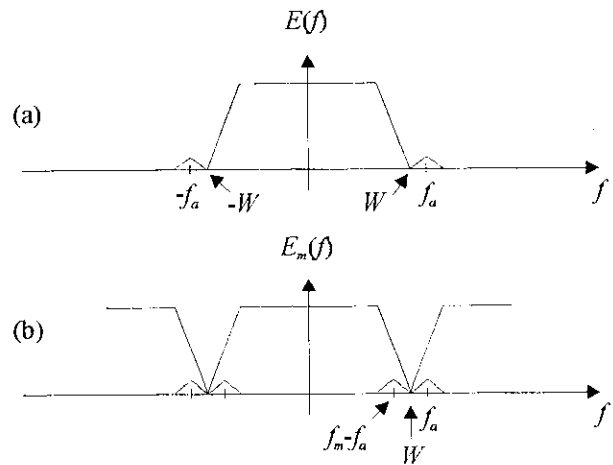


Figura 2.8. (a) Señal no limitada en banda (W). (b) La señal muestreada a $f_m=2W$. Nótese el traslape que provoca una distorsión alíneaal.

Para eliminar el solapamiento espectral se pueden hacer dos cosas: limitar en banda la señal, eliminando aquellas componentes de frecuencia mayores que $f_m/2$ (filtrar la señal) y/o aumentar la frecuencia de muestreo más allá de la frecuencia de Nyquist. Es interesante ver cómo afectan las dos soluciones, para distintos filtros (es decir, con filtros de distinta pendiente). Para esto, se toma un filtro con la característica H siguiente:

$$H(f) = \begin{cases} 1 & \vee |f| \leq W \\ \left(\frac{W}{f}\right)^n & \vee |f| > W \end{cases} \quad (\text{II-11})$$

donde n es un número entero que denota el orden del filtro. La frecuencia de muestreo está dada por

$$f_m = 2kW \ ; \ k > 1 \quad (\text{II-12})$$

Como el filtro no es ideal, habrá distorsión por adyacencia si la señal de entrada presenta frecuencias mayores que $f_m - W$. En la figura 2.9 se puede observar el traslape de componentes debido a la no idealidad del filtro. Utilizando esta gráfica, resulta fácil calcular las potencias de la señal $E(t)$ en la banda de 0 a W (P_b) y de las componentes que producen la distorsión (P_d).

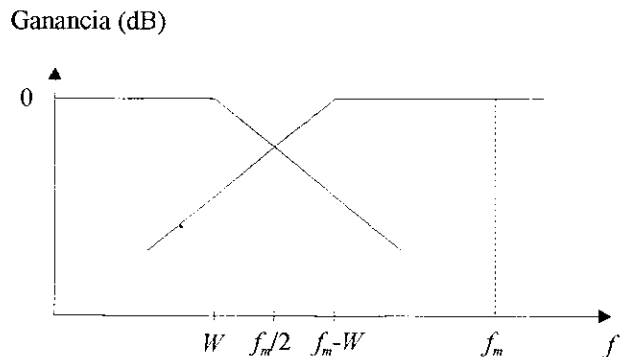


Figura 2.9. Traslape de las componentes debido al filtro no ideal.

$$P_b = \int_0^W |E(f)|^2 df \quad (\text{II-13})$$

$$P_d = \int_{f_m - W}^{f_m} |E(f)|^2 \left(\frac{W}{f}\right)^{2n} df \quad (\text{II-14})$$

Entonces, la relación señal a ruido (S/N) debido a la distorsión por adyacencia, que depende de n y de k es:

$$\left(\frac{S}{N_{DA}}\right)_{n,k} = 10 \log_{10} \frac{\int_0^W |E(f)|^2 df}{\int_{(2k-1)W}^{2kW} |E(f)|^2 \left(\frac{W}{f}\right)^{2n} df} \quad (\text{II-15})$$

Conviene analizar los efectos de variar n y k en esta ecuación. Al aumentar k , el denominador disminuye, pues al trabajar con frecuencias mayores el término W/f disminuye. Por lo tanto al aumentar k , la relación S/N aumenta. Al aumentar n , el denominador también disminuye, pues el término W/f es menor que la unidad, con lo cual aumenta la relación S/N .

Dependiendo de las características de la señal y de la forma en que se va a transmitir, debe tomarse la decisión sobre qué es más conveniente; si mejorar la calidad del filtro o aumentar la frecuencia de muestreo.

C. Reconstrucción de la señal

Como ya se ha mencionado, la reconstrucción de la señal se logra filtrando la señal muestreada. Para ejemplificar la reconstrucción, tomaremos una señal $e(t)$ pasabajos, con un espectro de ésta muestreada como el que se presenta en la figura 2.10a. Observaremos

lo que sucede al filtrarla con un filtro baso ideal con una frecuencia de corte $W=f_A$ (figura 2.10b) y el filtro no ideal de la ecuación (II-11) (figura 2.10c).

Con el filtro ideal no hay ningún problema, pues a la salida de éste se obtendrá la señal original. El problema ocurre con el filtro no ideal, el cual dejará pasar componentes de los espectros corridos debido al muestreo, resultando en una obtención de componentes a la salida que no

estaban presentes en la señal original. Para evaluar la potencia de estas componentes P_c se procede de manera similar como se hizo para calcular la distorsión por solapamiento espectral:

$$P_c = \int_{(2k-1)W}^{\infty} |E_m(f)|^2 \left(\frac{W}{f}\right)^{2n} df \quad (\text{II-16})$$

Si se toma en cuenta que para frecuencias mayores que la de muestreo f_m la atenuación del filtro es muy grande, la integral de la ecuación II-16 se puede reducir desde ∞ hasta $2kW$, y por lo tanto la relación señal a ruido resulta ser igual a la ecuación II-15, con lo que las observaciones que se hicieron de k y n son válidas aquí también.

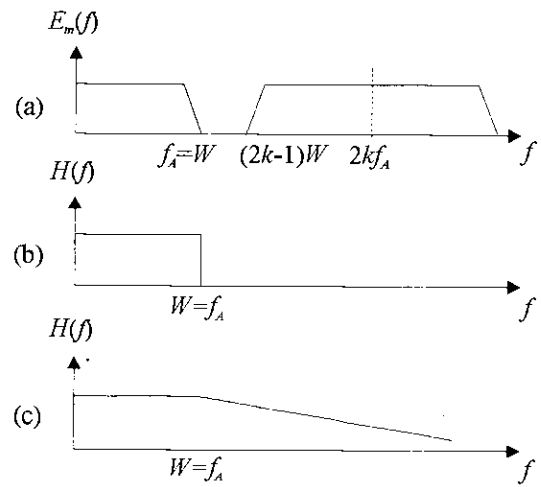


Figura 2.10. (a) Espectro de la señal $e_m(t)$. (b) Filtro ideal. (c) Filtro no ideal.

III. CUANTIFICACION

El siguiente paso para digitalizar una señal es cuantizarla. Cada una de las muestras de la señal muestreada, como se vio en el capítulo 2, podía tomar infinitos valores, o dicho de otra forma, los posibles valores de amplitud eran continuos. Para cuantificar una señal muestreada es necesario tener previamente establecidos los niveles a los cuales se va a aproximar cada muestra.

Al permitir que las muestras tomen únicamente ciertos valores, se introduce una variación a la señal, la cual se manifiesta como una distorsión y se denomina "Ruido de Cuantificación", "Error de Cuantificación", o más precisamente, "Distorsión de Cuantificación". Si se denomina $e(t)$ a la señal de entrada y $c(t)$ a la señal cuantificada, la distorsión $\epsilon(t)$ será:

$$\epsilon(t) = c(t) - e(t) \quad (\text{III-1})$$

Esta distorsión es irreversible, es decir, al insertarla en el proceso de cuantificación, ya no se puede extraer de la señal. Sin embargo, es preferible insertarle esta distorsión a la señal para poder transmitirla de forma digital, que permitir las degradaciones que sufre una señal al transmitirla de forma analógica, pues estas son incontrolables.

La distorsión de cuantificación sí se puede controlar, y de hecho el tema central del proceso de cuantificación es la evaluación y minimización de ésta. Esta distorsión se puede expresar como potencia o valor RMS. Esta medida es de mucha utilidad para el diseño y la especificación de los sistemas; sin embargo, el valor recomendado debe ser determinado subjetivamente escuchando la palabra u observando la imagen cuantificada, y no por analogía con el ruido aleatorio ordinario.

Existen otras dos degradaciones propias del proceso de cuantificación: la distorsión por recorte y el ruido del canal en reposo.

El análisis espectral de la distorsión de cuantificación es demasiado difícil de generalizar, por lo que resulta más conveniente el tratamiento probabilístico. Lo primero que hay que encontrar es una ecuación generalizada para la potencia de error de cuantificación en términos de la función de densidad de probabilidad $p(e)$ de la señal y de un cuantificador arbitrario. Se supondrá que la señal a cuantificar $e(t)$ está comprendida en un rango de variación entre $-A$ y $+A$, y por ningún motivo saldrá de estos límites. Esto se puede expresar de la siguiente forma:

$$\int_{-\infty}^{-A} p(e) de = \int_{+A}^{+\infty} p(e) de = 0 \quad (\text{III-2})$$

La cuantificación se realizará sobre un conjunto de niveles

$$-a_N; -a_{N+1}; \dots; -a_1; a_0; a_1; \dots; a_N$$

espaciados entre sí arbitrariamente. En la figura 3.1 se pueden observar ciertos niveles de cuantificación, así como los límites de captura con una línea en trazos y las señales de entrada y cuantificada. Los saltos escalonados de la señal cuantificada se producen cuando $e(t)$ cambia del rango de captura

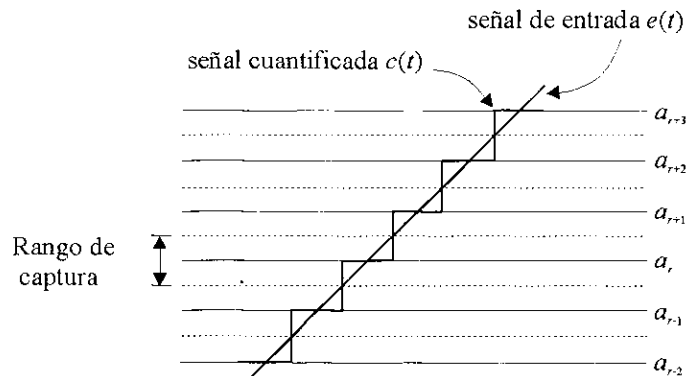


Figura 3.1. Señal de entrada, señal de salida y límites de captura para algunos niveles de cuantificación.

de un intervalo al otro. En la figura 3.2 se representa solamente el nivel a_r . En ésta se

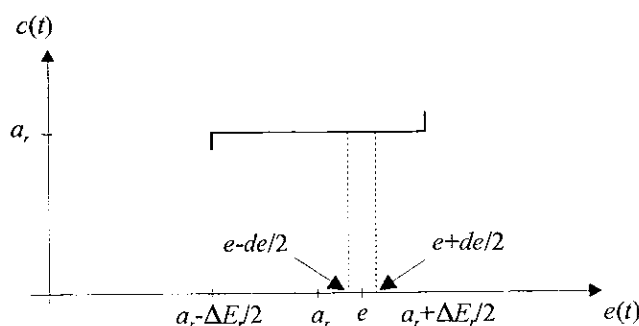


Figura 3.2. El nivel a_r , el intervalo cuántico E_r y el rango de captura $a_r - \Delta E_r/2; a_r + \Delta E_r/2$.

puede observar el rango de captura:

$$a_r - \frac{\Delta E_r}{2} ; a_r + \frac{\Delta E_r}{2}$$

donde E_r es el intervalo cuántico del nivel r .

Si $e(t)$ se encuentra en el rango de captura de a_r , el error de cuantificación instantáneo será:

$$\epsilon = a_r - e \quad (\text{III-3})$$

y la potencia de error instantáneo

$$\epsilon^2 = (a_r - e)^2 \quad (\text{III-4})$$

La potencia de error para el intervalo $(e - de/2)$ a $(e + de/2)$ (capturado por a_r) es

$$(a_r - e)^2 p(e) de \quad (\text{III-5})$$

con lo cual, la potencia media de distorsión de cuantificación de todo el rango de captura del nivel cuántico es

$$\overline{\epsilon_r^2} = \int_{a_r - \frac{E_r}{2}}^{a_r + \frac{E_r}{2}} (a_r - e)^2 p(e) de \quad (\text{III-6})$$

Por lo tanto, la potencia total de distorsión de cuantificación N_Q se obtiene con la suma de todos los $\overline{\epsilon_r^2}$

$$N_Q = \sum_{r=-N}^N \overline{\epsilon_r^2} = \sum_{r=-N}^N \int_{a_r - \frac{E_r}{2}}^{a_r + \frac{E_r}{2}} (a_r - e)^2 p(e) de \quad (\text{III-7})$$

La potencia de la señal sin cuantificar es

$$S = \int_{-A}^{+A} e^2 p(e) de \quad (\text{III-8})$$

resultando así que la relación señal a ruido debido a la distorsión de cuantificación es

$$\frac{S}{N_Q} = \frac{\int_{-A}^{+A} e^2 p(e) de}{\sum_{r=-N}^N \int_{a_r - \frac{E_r}{2}}^{a_r + \frac{E_r}{2}} (a_r - e)^2 p(e) de} \quad (\text{III-9})$$

La elección de los niveles a_r puede hacerse de dos formas: fijando todos los E_r iguales, o asignando distintos tamaños a cada E_r según sea conveniente. Esto da cabida a dos tipos de cuantificación: uniforme y no uniforme, respectivamente, las cuales se detallan a continuación.

A. Cuantificación uniforme

Para este caso, los intervalos cuánticos son todos iguales, es decir $E_r = \delta = \text{constante}$. Si se tiene un alto número de niveles de cuantificación, se puede suponer que la función de densidad de probabilidad varía muy poco en un intervalo cuántico, pudiendo considerarla constante en todo el intervalo. Así, la potencia de distorsión de cuantificación resulta ser

$$N_Q = \sum_{r=-N}^N p(a_r) \int_{a_r - \frac{\delta}{2}}^{a_r + \frac{\delta}{2}} (a_r - e)^2 de = \sum_{r=-N}^N \frac{p(a_r) \delta^3}{12} \quad (\text{III-10})$$

Como $p(a_r)\delta$ es la probabilidad que e se encuentre en el intervalo cuántico r , y la suma de las probabilidades que e se encuentre en cada uno de los intervalos debe ser la unidad (pues se debe encontrar en alguno), resulta

$$N_Q = \frac{\delta^2}{12} \quad (\text{III-11})$$

Para calcular la relación señal a ruido por distorsión de cuantificación sería necesario calcular la potencia de la señal, la cual depende de $p(e)$. Si se llama V_e al valor RMS de la señal de entrada, resulta

$$\frac{S}{N_Q} = \frac{V_e^2}{\delta^2/12} \quad (\text{III-12})$$

y

$$\left[\frac{S}{N_Q} \right]_{dB} = 10 \log \frac{V_e^2}{\delta^2/12} = 10.8 + 20 \log \left(\frac{V_e}{\delta} \right) \quad (\text{III-13})$$

En ocasiones es preferible referir la distorsión al valor pico de la señal de entrada, el cual es $V = N\delta$, con lo que la relación potencia de cresta a distorsión de cuantificación es

$$\frac{S_c}{N_Q} = \frac{N^2 \delta^2}{\delta^2 / 12} = 12N^2 \quad (\text{III-14})$$

o

$$\left[\frac{S_c}{N_Q} \right]_{\text{dB}} = 10.79 + 20 \log N \quad (\text{III-15})$$

Se puede observar que la relación crece en 6dB cuando se duplica el número de niveles. Al utilizar una codificación digital de la muestra cuantificada, duplicar el número de niveles equivale a aumentar un bit en la cuantificación. El número total de niveles $2N$ viene dado por 2^n , donde n es el número de bits a utilizar. Sustituyendo esto resulta

$$\left[\frac{S_c}{N_Q} \right]_{\text{dB}} = 4.77 + 6.02n \quad (\text{III-16})$$

por lo que la relación aumenta 6dB por cada bit que se agrega.

B. Cuantificación no uniforme

La cuantificación uniforme funcionaría siempre bien si las señales tuvieran una función de densidad de probabilidad uniforme. Sin embargo, éste no es el caso más general. Si se conoce dicha función de la señal a digitalizar, no es necesario asignar muchos niveles de cuantificación a los rangos en donde se sabe que la señal va a estar sólo durante pequeños instantes. Entonces, si se cuenta con un número de niveles cuánticos específico, es mejor asignar más niveles a los rangos donde se sabe que la señal se encuentra la mayor parte del

tiempo. Esto se logra separando los niveles de manera no uniforme, juntándolos donde hay más y separándolos donde hay menos probabilidad de ocurrencia de la señal. Según la naturaleza de la señal, se debe encontrar la ley compansión (compresión y expansión) que mejor se le adapte.

En telefonía, donde las señales a procesar son de voz, se han adoptado mayormente dos leyes, conocidas como ley μ y ley A , las cuales se describirán posteriormente.

1. Implementación. La cuantificación no uniforme se puede implementar de distintas formas. Estos métodos se ilustran en la figura 3.3 y son los siguientes:

Directa: mediante un cuantificador donde los niveles cuánticos ya están espaciados según la ley adoptada. La dificultad de este método radica en lo complicado de mantener estables los umbrales de decisión.

Compresión digital: el mensaje se cuantifica utilizando un cuantificador de intervalos muy finos y luego se procesa digitalmente (usualmente con una ROM), según la ley establecida para llevarlo a la cantidad de bits deseados.

Compansión analógica: este fue el primer método utilizado. Consiste en pasar la

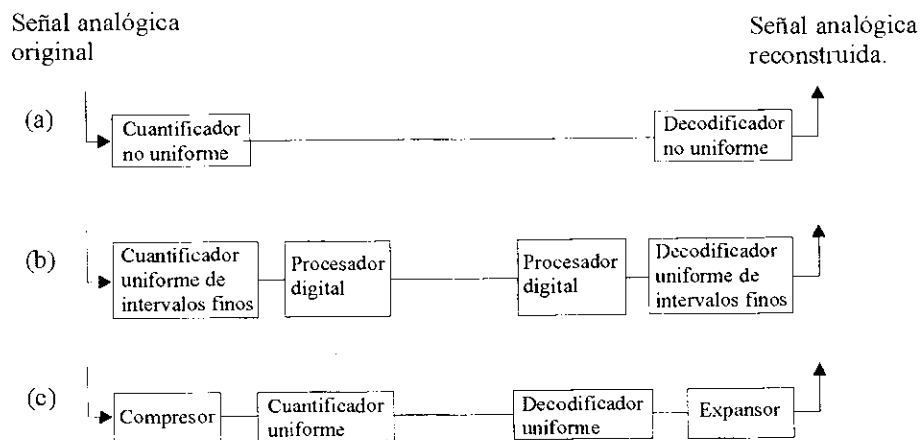


Figura 3.3. Cuantificación no uniforme. (a) directa (b) compresión digital (c) compansión analógica.

señal de entrada a través de un cuádrupolo alineal (compansor), el cual le modifica su función de densidad de probabilidad para luego cuantificarlo uniformemente. El compansor utilizado para señales de voz expande los niveles más bajos de tal modo que abarquen un mayor número de intervalos cuánticos, y comprime los niveles más altos cubriendo un menor número de intervalos. En el receptor se utiliza un cuádrupolo cuya función de transferencia es inversa a la del transmisor, para restaurar la linealidad del proceso.

2. Ley A . Esta ley de compansión se utiliza en Europa, Centro y Sur América y en todas las rutas telefónicas internacionales. Los países que no la utilizan deben adaptarse a ellas para sus comunicaciones internacionales. Su curva se caracteriza matemáticamente por:

$$x = \begin{cases} \frac{Aw}{1 + \ln A} & \vee 0 \leq w \leq 1/A \\ \frac{1 + \ln Aw}{1 + \ln A} & \vee 1/A \leq w \leq 1 \end{cases} \quad (\text{III-17})$$

3. Ley μ . Esta ley se utiliza en Estados Unidos y en Japón y deben "traducirla" a ley A para sus comunicaciones internacionales. Se define por la relación:

$$x = \frac{\ln(1 + \mu w)}{\ln(1 + \mu)} \quad \vee 0 \leq w \leq 1 \quad (\text{III-18})$$

IV. EL PCM

Generalmente, una señal que se cuantifica antes de ser transmitida, también se muestrea. La cuantificación se utiliza para reducir los efectos del ruido, y el muestreo permite multiplexar varios mensajes por división de tiempo. Las operaciones combinadas de muestrear y cuantificar generan una forma de onda PAM (modulación por amplitud de pulso) cuantificada, es decir un tren de pulsos cuyas amplitudes están restringidas a una cantidad discreta de magnitudes. Estas muestras cuantificadas se podrían transmitir directamente, o representar cada nivel de cuantificación por un código y transmitir el código correspondiente al valor en vez del valor mismo. Frecuentemente es utilizada la representación binaria para denominar los niveles cuantificados, pues la transmisión de los dígitos binarios es muy simple, transmitiéndolos como pulsos. Se puede comprender de donde toma su nombre este sistema de transmisión: PCM (por las siglas en inglés de Modulación por Codificación de Pulsos).

Para ejemplificar, se va a considerar el siguiente caso: Se tiene una señal analógica $e(t)$ cuyos valores están limitados entre -4 y $+4V$. Los niveles de cuantificación estarán separados $1V$, por lo cual se necesitarán ocho niveles. Los niveles estarán localizados en $-3.5, -2.5, \dots, +3.5V$. Se le asigna el número 0 al nivel $-3.5V$, el número 1 a $-2.5V$, y así sucesivamente hasta asignar el 7 a $+3.5V$. Cada número tiene su representación en binario, desde el 000 para el 0 hasta 111 para el 7. En la figura 4.1 se puede observar la señal de entrada $e(t)$, los instantes de muestreo, el valor de las muestras, el nivel cuántico más próximo, el número correspondiente al nivel y la representación binaria. Se podría transmitir el valor de las muestras, el valor cuántico de la muestra o, en PCM, la representación binaria.

La simplicidad en la transmisión y recepción en PCM consiste en que únicamente se trabajan dos niveles: 1 y 0. Por lo tanto, en el receptor, para reconstruir la señal

cuantificada, todo lo que se requiere es, dentro de la ranura de tiempo de cada pulso, determinar si éste representa un 1 ó un 0. La amplitud exacta del pulso no es importante.

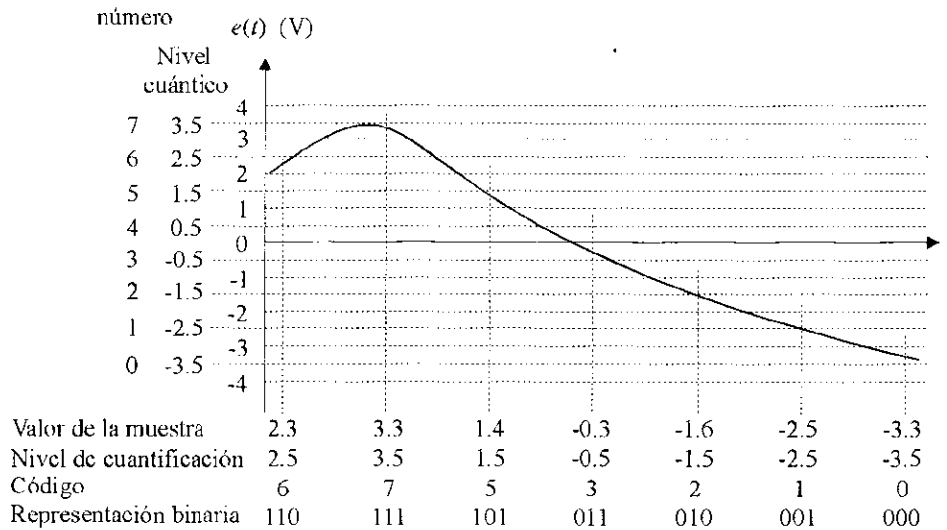


Figura 4.1. La señal $e(t)$ se muestrea regularmente; se indican el valor de la muestra, el nivel de cuantificación, el código y la representación binaria.

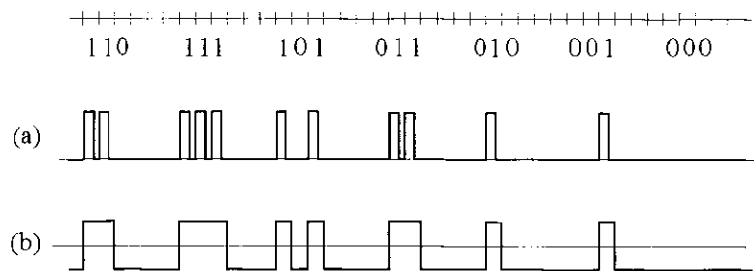


Figura 4.2. (a) Representación por medio de pulsos. (b) Representación por medio de voltajes.

La transmisión de los códigos binarios puede realizarse de varias formas. En la figura 4.2 se ejemplifican dos formas, utilizando los mismos valores de la figura 4.1. A cada grupo de dígitos binarios que conforman un código se les denomina 'palabra'. Al tiempo que toma transmitir una palabra se le denomina ranura de tiempo de palabra. Se pueden

intercalar palabras conteniendo códigos de distintas fuentes de señal, logrando así la multiplexación de señales por división de tiempo.

A. El sistema PCM

Un sistema de comunicaciones PCM se representa en la figura 4.3. Una señal de entrada $e(t)$ se muestrea, y estas muestras se cuantifican. Las muestras cuantificadas se le aplican al codificador, el cual responde a cada una de éstas generando un patrón único e identificable de pulsos binarios. En el ejemplo de las figuras 4.1 y 4.2, cada patrón de pulsos tiene un significado numérico que corresponde al orden asignado de los niveles cuantificados. Sin embargo, esta característica no es esencial y se le puede asignar cualquier patrón de pulsos a cualquier nivel, siempre y cuando en el receptor se pueda identificar el nivel a partir del patrón de pulsos.

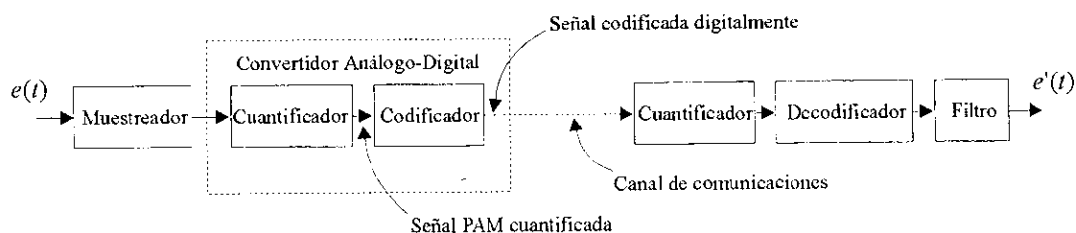


Figura 4.3. Sistema de comunicaciones PCM.

La combinación del cuantificador y el codificador en la caja punteada de la figura 4.3 se conoce como "convertidor análogo-digital", abreviado como "convertidor A/D". En los convertidores A/D disponibles comercialmente, no hay una distinción clara entre la parte de circuitería utilizada para cuantificar y la utilizada para codificar. En algunos casos, incluso el muestreador está incluido en el mismo circuito integrado. Los convertidores A/D pueden entregar el resultado de la conversión en formato serial o en formato paralelo; en este último caso, es necesario convertir los datos de paralelo a serial previo a transmitirlos.

Cuando los datos llegan al receptor, éste debe decidir, pulso por pulso, si la señal es un alto o un bajo. Este procedimiento se conoce como "recuantificación", y por eso el primer bloque en el receptor es un cuantificador; sin embargo es un cuantificador relativamente simple, pues únicamente debe decidir entre dos niveles. Este cuantificador pasa los datos al decodificador, conocido también como convertidor digital/análogo o convertidor D/A. En muchos de los convertidores D/A disponibles, la entrada debe ser en formato paralelo, en cuyo caso es necesario convertir los datos de formato serial a este otro. La salida del decodificador o convertidor D/A, es la secuencia de pulsos muestreados multinivel cuantificados. Esta salida, que constituye una señal PAM cuantificada, se filtra para eliminar cualquier componente de frecuencia que quede fuera de la banda base. La salida final, $e'(t)$, es idéntica a la señal $e(t)$ de entrada, excepto por el ruido de cuantificación y por algún error ocasional en la decisión del nivel de un pulso en la recepción.

El sistema PCM se utiliza ampliamente en las redes telefónicas. Una de las principales razones de esto es que la calidad de la transmisión es casi independiente de la distancia, pues las señales digitales pueden ser regeneradas casi perfectamente en puntos intermedios de los extremos del enlace. Otra característica importante es que las señales provenientes de distintas fuentes pueden ser multiplexadas en el tiempo, permitiendo un aumento en la capacidad de los pares de cable utilizados para la transmisión. Además, la transmisión en PCM que se ha estandarizado posee velocidades de transmisión de 56 ó 64Kbit/seg, lo cual permite transmitir en el canal no solamente señales vocales, sino también cualquier otro tipo de datos.

Debido a que lo que se transmite en PCM no son más que datos que permiten reconstruir la señal analógica original, y en sí cómo se transmita la información para efectuar la reconstrucción no es relevante, pueden idearse varios métodos para la transmisión de dicha información. Un sistema eficiente sería aquel que permita reconstruir la señal original transmitiendo la menor cantidad de datos posible (es decir, utilizando el

canal de transmisión eficientemente). Así es como han surgido distintos tipos de PCM, los cuales se explican a continuación.

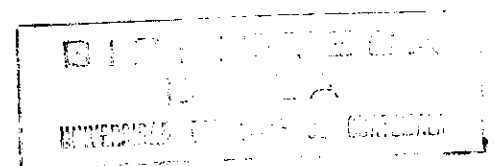
B. PCM Normal.

El PCM normal es el que ya se ha ejemplificado anteriormente en las figuras 4.1 y 4.2. Es decir, el rango de distintos valores posibles de la señal de entrada se divide en cierta cantidad de intervalos, y éstos se codifican a un código binario único para cada nivel cuantificado. Comercialmente se ha adoptado un código de 8 bits (256 niveles de cuantificación) y 8000 muestras por segundo (la mayor parte de la potencia de la voz se encuentra entre los 300 y 3500Hz, con lo cual 8000 muestras por segundo es bastante holgado), con lo cual la velocidad de transmisión requerida para acomodar un canal PCM es de 64kb/s.

C. PCM Diferencial (DPCM).

Un sistema en el cual la señal $e(t)$ a ser transmitida es primero muestreada, tiene cabida a un esquema de transmisión alternativo a transmitir los valores muestreados (cuantificados o no) en cada momento de muestreo. Se puede, para cada instante de muestreo, transmitir la diferencia entre el valor de la muestra presente y la anterior. Si se transmiten las diferencias, entonces simplemente sumando estos cambios en el receptor, es posible regenerar la señal original.

Este esquema diferencial es especialmente útil cuando las diferencias se transmiten en PCM. Es de suponerse que las diferencias $e(k) - e(k-1)$ serán menores que las muestras mismas. Por lo tanto, son necesarios menos niveles para cuantificar las diferencias que los requeridos para cuantificar las muestras, y se requerirán menos bits para cuantificar los



niveles. Por ejemplo, si la señal de entrada $e(t)$ se extiende en el rango V_H-V_L , y utilizando PCM las muestras $e(k)$ se codifican utilizando 256 niveles, la diferencia entre niveles es $S=(V_H-V_L)/256$, o lo que es lo mismo, $V_H-V_L = 256S$. Si las diferencias $e(k)-e(k-1)$ se extienden sólo en el rango $\pm 2S$, los niveles de cuantificación necesarios estarán en $\pm 0.5S$ y $\pm 1.5S$. De esta forma hay únicamente 4 niveles, y 2 bits por muestra son suficientes. La variación en la señal $e(t)$ entre una muestra y otra depende de la relación entre su frecuencia y la frecuencia de muestreo.

Dependiendo de las características del sistema, se debe elegir la cantidad de bits por muestra que se van a utilizar. Cuando se utiliza un sólo bit, la modulación recibe el nombre de Modulación Delta, DM ó M.

Dado que en este sistema se transmiten únicamente las diferencias, es necesario tener en el receptor un acumulador que vaya sumando las diferencias respectivas. Se vuelve necesario tener en el transmisor un acumulador igual al del receptor, de manera que se

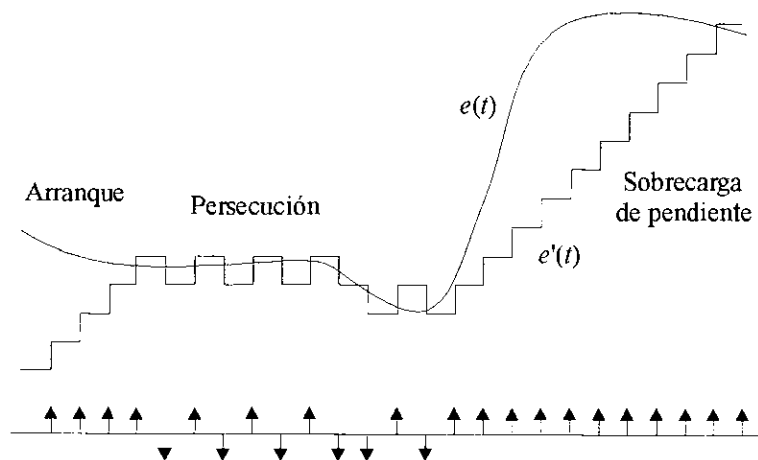


Figura 4.4. La señal de entrada $e(t)$ y el valor acumulado $e'(t)$ para un sistema con modulación delta.

pueda saber el valor que se tiene acumulado en el receptor y transmitir así la diferencia entre el valor de $e(t)$ al momento de muestrear y el valor acumulado.

El sistema diferencial puede presentar dos problemas. Primero, si la señal de entrada $e(t)$ varía más rápidamente de lo que el sistema puede detectar, va a existir una diferencia entre la salida $e'(t)$ y la entrada $e(t)$. Esto se puede solucionar aumentando la frecuencia de muestreo o la cantidad de bits utilizados para codificar las diferencias. A este fenómeno se le conoce como *ruido de sobrecarga*. Segundo, si la señal de entrada varía muy lentamente, el sistema va a oscilar alrededor del valor de $e(t)$, pues éste no es capaz de transmitir una diferencia de '0'; de tal forma, con una muestra se le va a decir al acumulador que suba, y como con ésta se va a pasar del valor $e(t)$, con la siguiente se le va a decir que baje. Esto provoca un ruido que puede eliminarse mediante un filtro, pues la frecuencia de muestreo en PCM diferencial es considerablemente mayor que la frecuencia de Nyquist. En la figura 4.4 se muestra la señal de entrada $e(t)$ y el valor acumulado para un sistema con modulación delta, y se pueden observar estos dos problemas.

D. Modulación Delta Adaptativa (ADM)

Se considerará ahora una modificación de la DM llamada Modulación Delta Adaptativa (ADM por sus siglas en inglés), en la cual el valor que se suma o resta del acumulador no se mantiene fijo, sino que se incrementa progresivamente cuando ocurre error de sobrecarga, logrando que el valor acumulado llegue más rápidamente al valor de la señal. Esto se puede lograr de muchas maneras y con variados algoritmos. Aquí se describirá únicamente uno.

Al valor que se sumará (o restará) del acumulador le llamaremos paso, P . Este P , como ya se mencionó, no será fijo, sino que podrá variar con cada intervalo de muestreo. El valor de P será siempre un múltiplo de un paso básico P_0 . El algoritmo con el cual se genera P es el siguiente: en el k -ésimo instante de muestreo, se genera un paso igual en magnitud al generado en el $(k-1)$ -ésimo instante. Este paso se suma o resta del acumulador, según sea necesario, para acercar el valor acumulado al valor de la señal de

entrada. Además, si la dirección del paso en el instante k es la misma que en el instante $(k-1)$, el paso se incrementa en P_o . Si la dirección es contraria, el paso se decrece en P_o . Utilizando este algoritmo, hay ocasiones en las cuales el paso total $P=0$. En un caso así, en el siguiente instante el paso será P_o en la dirección adecuada para acercar el acumulado a la señal. Para poder escribir el algoritmo en forma de ecuación, llamaremos $\hat{e}(t)$ al valor acumulado de $e(t)$; de esta forma definiremos:

$$d(k) = \begin{cases} +1 & \text{si } e(t) > \hat{e}(t) \text{ en el instante } k \\ -1 & \text{si } e(t) < \hat{e}(t) \text{ en el instante } k \end{cases} \quad (\text{IV-1})$$

Así, en el instante k el paso P será dado por:

$$P(k) = |P(k-1)|d(k) + P_o d(k-1) \quad (\text{IV-2})$$

Las características importantes de este esquema adaptativo se muestran en la figura 4.5. Se puede observar que mientras la condición $e(t) > \hat{e}(t)$ persista, los saltos en $\hat{e}(t)$ se vuelven progresivamente mayores. Se logra así que $\hat{e}(t)$ alcance a $e(t)$ mucho antes de lo que lo haría la modulación delta. Por otro lado, cuando, en respuesta a una pendiente pronunciada en $e(t)$ el paso se haya vuelto muy grande, se requieren varios intervalos de muestreo para que el paso decaiga. Se puede notar también que cuando la señal $e(t)$ permanece constante (dentro de un intervalo de amplitud P_o), $\hat{e}(t)$ oscila alrededor de $e(t)$ con una frecuencia de la mitad de la que tiene en DM.

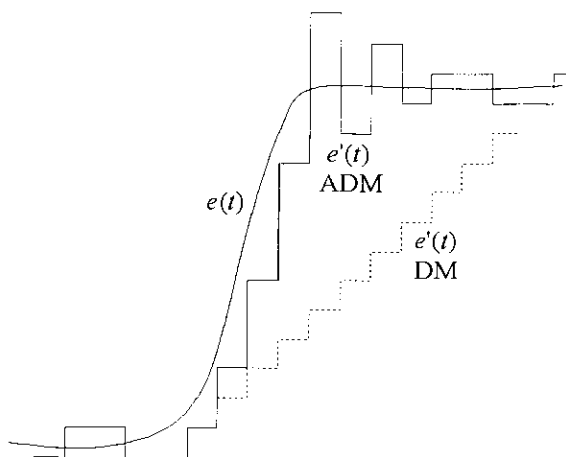


Figura 4.5. Comparación entre las modulaciones delta (DM) y delta adaptativa (ADM).

En resumen, el sistema ADM reduce el error de sobrecarga, pero lo hace a costa de aumentar el error de cuantificación. En la transmisión de voz, la reducción del error de

sobrecarga produce una ventaja neta, aun con el error de cuantificación que provoca, y un modulador de delta adaptativa puede trabajar con velocidades de transmisión de 32kb/s, con una calidad comparable con la de PCM a 64kb/s. Más aún, el ADM puede operar a 16kb/s sufriendo únicamente una pequeña degradación de su rendimiento. La frecuencia de muestreo en ADM es también mayor que la frecuencia de Nyquist, por lo que el error de cuantificación se puede disminuir bastante utilizando un filtro pasa-bajos en la salida, sin degradar la señal original.



V. ELABORACION DEL CIRCUITO

A. Elección de las características de operación

El primer paso para el diseño del circuito fue fijar las características de operación deseadas del mismo. Estas se listan a continuación:

- Entradas: micrófono y señal a través de un conector BNC (línea), seleccionable.
- Salidas: una para bocina (audio) y otra con un conector BNC, siendo la bocina desactivable. Además, otro conector BNC para monitorear la entrada al convertidor análogo-digital.
- Filtros: de entrada y salida, seleccionables y sintonizables, con un rango de frecuencias que depende de la frecuencia de muestreo máxima permitida.
- Frecuencia de muestreo: variable, en pasos de 200Hz, hasta el máximo que el circuito permita (por lo menos 16 kHz).
- Bits de cuantificación: de 1 a 8
- Capacidad para grabar muestras en RAM, para luego ser reproducidas.
- Poder trabajar con PCM, DPCM, ADPCM y compansión.
- Despliegue de opciones seleccionadas por medio de LEDs y 7 segmentos.
- Selección de las opciones por medio de botones.

B. Elección de los circuitos integrados

Tomando en cuenta las características fijadas, el siguiente paso fue seleccionar los circuitos integrados a utilizar.

Debido a que no se iba a trabajar con más de 8 bits, se eligió el microprocesador Z80. La velocidad de reloj del sistema sería la única limitante para la frecuencia de muestreo, así que se eligió el Z80B, el cual tiene una frecuencia de reloj máxima de 6 MHz.

Luego se decidió utilizar el convertidor análogo-digital ADC0820, debido a su corto tiempo de conversión de 600ns en el modo de escritura-lectura (WR-RD). Este integrado permite además una fácil conexión al BUS del CPU.

Para la conversión digital-análoga se eligió el DAC0808.

Como era necesario seleccionar varias opciones de las señales analógicas, se optó por utilizar conmutadores analógicos de circuito integrado LF13201, los cuales permiten ser activados con señales de niveles TTL.

Debido a que el DAC0808 no se puede conectar directamente al bus, fue necesario utilizar un puerto paralelo para proveerle un interfaz hacia el CPU. Se escogió el Z80B-PIO, el cual tiene dos puertos paralelos de 8 bits. El puerto restante se utilizaría para controlar los conmutadores LF13201. Se utilizarían además 2 bits de este puerto para elegir una RAM a utilizar al grabar una muestra, permitiendo así conectar hasta 4 circuitos integrados de RAM, decodificando los 2 bits con un integrado 74LS139.

Teniendo la capacidad de elegir la RAM a utilizar, fue posible utilizar más de los 64kB que permite direccionar el Z80, así que se trabajaría con 4 RAMs de 32kB cada una. Se eligió la RAM 62256.

Para almacenar el programa se escogió un circuito integrado de ROM de 2kB, y para poder efectuar cambios fácilmente durante la fase de desarrollo del programa se eligió una EEPROM, siendo ésta la 2816.

Debido a que la familia Z80 carece de un controlador de despliegue y/o teclado, se tuvo que utilizar uno de la familia 8200. Se eligió el 8279-5, cuya frecuencia máxima de reloj es de 5 MHz. Este sería el circuito que limite la frecuencia del reloj de sistema, el cual se fijó en 5 MHz.

A manera de simplificar la programación de la frecuencia de muestreo, se decidió utilizar el temporizador/contador de la familia Z80, el Z80B-CTC, el cual debería ser programado en el modo de contador, con una señal de entrada de frecuencia de la mitad de la del reloj del sistema (es decir, 2.5 Mhz). Para agilizar la toma de valores y obtener una mayor frecuencia de muestreo, se utilizaría la facilidad que dan los integrados de la familia Z80 para manejar interrupciones y se eligió el modo de interrupción 2. Para poder dividir la frecuencia hasta valores pequeños, se conectarían dos de los cuatro contadores de este integrado en cascada.

Se seleccionó un cristal TTL de 10.0 MHz, y esta frecuencia se dividió dos veces con un 74LS74, para obtener los 5.0 MHz de reloj del sistema, y los 2.5 MHz para entrada el CTC.

Se eligió el despliegue de 7 segmentos MAN72, que es de ánodo común. El LED a utilizar sería el XC209RB.

El amplificador operacional LM348 se escogio para efectuar todas las operaciones analógicas.

El amplificador de audio elegido fue el LM386. Se eligió el micrófono electret de condensador WM-034CX.

Se utilizaría un LM555 para generar la señal de reinicio. Dado que el 8279 tiene una entrada de reinicio invertida a la de los demás circuitos integrados, fue necesario asignar un 74LS04 para invertir esta señal.

C. Conexión de los circuitos integrados

Al trabajar en el soporte físico, se comenzó con la parte digital y luego se diseñó la parte analógica.

1. Circuitería digital. Teniendo los circuitos integrados digitales que se van a utilizar ya elegidos, se procedió a asignar las direcciones a la memoria y a los puertos.

Se le asignó direcciones comenzando desde el 0000_h a la ROM, pues el Z80 comienza a leer instrucciones en esa dirección después de ser inicializado; por lo tanto la ROM utilizaría las direcciones 0000_h-07FF_h . A la RAM se le asignó los 32kB superiores (8000_h-FFFF_h), para hacer así la elección entre ROM y RAM muy simple, utilizando únicamente la línea de direcciones A15. Así, con $A15=1$, se selecciona RAM, y con $A15=0$, ROM. Habiendo decidido el direccionamiento de esta forma, queda lugar para direccionar 32kB de ROM, los cuales podrían ser utilizados si los 2kB asignados resultaran insuficientes para el programa. Se implementó un circuito combinacional con el 74LS00 para generar la señal de selección de RAM y ROM, utilizando A15 y \overline{MREQ} . Como se colocarían 4 RAMs en las mismas direcciones, para diferenciarlas se aplicó la señal resultante de este circuito combinacional a un 74LS138 en la pata G, y las entradas A y B provendrían de los bits más significativos del puerto B del Z80B-PIO.

Las direcciones de los puertos no tienen mayor importancia, siempre y cuando no ocurran traslapes. Se decidió comenzar sus direcciones desde el 00_h . Como ninguno de los periféricos que se direccionaría contenía más de 4 puertos, se realizó la decodificación

de las direcciones con el decodificador restante en el 74LS138, con sus entradas A y B conectadas a las direcciones A2 y A3, respectivamente. Sus salidas Y0-Y3 se conectaron a la entrada habilitadora del CTC, PIO, ADC y 8279, respectivamente. De esta forma, el direccionamiento quedó como se detalla en la tabla 5.1.

Tabla 5.1
Direccionamiento de los Puertos

Dirección	Descripción
00 _h	Canal 0 del CTC
01 _h	Canal 1 del CTC
02 _h	Canal 2 del CTC
03 _h	Canal 3 del CTC
04 _h	Datos del canal A del PIO
05 _h	Control del canal A del PIO
06 _h	Datos del canal B del PIO
07 _h	Control del canal B del PIO
08 _h , 09 _h , 0A _h ó 0B _h	Convertidor análogo digital
0C _h ó 0E _h	Datos del 8279
0D _h ó 0F _h	Control del 8279

Las señales digitales de entrada al convertidor DAC0808 se conectaron al puerto A del Z80B-PIO.

2. Circuitería analógica. Para la parte analógica se comenzó trabajando con el micrófono. Se polarizó con una resistencia de 10K, entre tierra y -12V. Se hicieron pruebas observando la señal en el osciloscopio mientras se hablaba. De esta forma se determinó que para llevar la señal a los 5Vpp necesarios para cubrir todo el rango del convertidor, era necesario amplificarla aproximadamente 5 veces. Se diseñó entonces un

amplificador inversor con el LM348 con ganancia de 4.7, para evitar que la señal saliera del rango. A continuación se colocó un seguidor.

Para la entrada de señal (línea), se implementó un circuito igual al del amplificador, modificando la ganancia a 1.

Tocaba ahora diseñar la parte que seleccionaría cuál de estas dos señales se tomaría. Se implementó un multiplexor analógico utilizando dos conmutadores del LF13201, conectando la salida de los dos entre sí, y haciendo que la entrada habilitadora de uno fuera invertida de la del otro. Para este efecto se utilizaría uno de los inversores libres del 74LS04.

El filtro pasa-bajos sintonizable fue la última parte del circuito que se diseñó, pues era necesario saber la frecuencia de muestreo máxima que se iba a tener, para así calcular la frecuencia de corte máxima en la mitad de la de muestreo. Cuando se hicieron los cálculos de la frecuencia máxima de muestreo, se encontró que esta iba a ser de 20.0 kHz. De esta forma, se diseñó el filtro para cortar a frecuencias entre 100 Hz y 10 kHz. El diseño del filtro se tomó del libro “Manual para Ingenieros y técnicos en electrónica” (ver bibliografía). Debido a que el filtro de entrada y el de salida son iguales, y para utilizar únicamente un integrado para los dos, se decidió utilizar 2 amplificadores operacionales por cada filtro (es decir, medio LM348 por filtro). Con 2 operacionales, un filtro de orden 5 es el de mayor orden que se puede implementar. Para hacer sintonizable el circuito, se colocaron potenciómetros en serie con las resistencias del filtro. Los 5 potenciómetros se mueven con el mismo eje, de forma que los 5 polos del filtro se ajusten al mismo tiempo. Para calcular el valor de los potenciómetros, se calculó primero el filtro para la frecuencia mayor (resistencia menor) y luego para la frecuencia menor (resistencia mayor), y con la diferencia entre las resistencias se obtuvo el valor. Los filtros se implementaron en un circuito impreso.

Para elegir entre la señal filtrada o no filtrada, se implementó otro multiplexor analógico con los dos conmutadores libres del LF13201; uno alimentado con la señal filtrada y el otro con la señal sin filtrar.

Ya se tenía en este punto diseñada toda la circuitería necesaria para elegir la fuente de la señal de entrada, y si ésta iba a ser filtrada o no. Sólo restaba acoplar esta señal al convertidor análogo-digital. Se implementó un amplificador inversor sumador, el cual sumaría a la señal de entrada un voltaje DC, a manera de subir el nivel DC de la entrada a 2.5V, para que quedara centrada en la escala del convertidor. Se utilizó el voltaje de alimentación negativo (-12V) para subir el nivel, y se utilizó una relación de resistencias en el sumador de 10K/48K, con lo cual se obtiene exactamente 2.5V. Se agregó un potenciómetro de 1K (ajuste) en serie a los 48K, para poder ajustar el nivel de reposo de la señal de entrada, en caso que quedara sobre el umbral de decisión de algún nivel del convertidor. Esta señal se conectó al convertidor análogo-digital. Para proteger el convertidor se conectó en su entrada un diodo Zener de 5.1V (1N4733).

Los convertidores análogo-digital y digital-análogo se trabajaron con una escala de 5V.

Para pasar la salida del convertidor digital-análogo de corriente a voltaje, se aplicó la corriente de salida a la entrada inversora de un operacional, con realimentación de la salida a la entrada inversora por medio de una resistencia del mismo valor utilizado en la referencia de escala del convertidor. El nivel DC de la salida de este operacional se eliminó utilizando un capacitor.

Se implementó otro multiplexor analógico igual a los anteriores para elegir entre la señal de salida filtrada o no. La salida de este multiplexor se aplicó a un seguidor de voltaje, y la salida de éste se alambro al conector BNC de salida.

El amplificador de audio se alimentó con la señal proveniente de otro multiplexor analógico que elegía entre la señal de salida o tierra. Este amplificador se diseñó para una ganancia de 20, con un potenciómetro de entrada de 10K para controlar el volumen.

3. Fuente de alimentación. Para la fuente se utilizó un transformador de 26VAC con derivación central, con rectificador de onda completa. Se utilizaron capacitores de 4700 μ F, y los reguladores integrados LM7812, LM7912 y LM7805. Del último se utilizaron 2; uno para alimentar los circuitos integrados y el otro para alimentar los LEDs y los 7 segmentos.

D. Interfaz con el usuario

El siguiente paso fue diseñar el interfaz que se tendría con el exterior; es decir el despliegue de la información de los parámetros elegidos y la forma de elegirlos, la ubicación de los conectores, interruptores, potenciómetros, etc. Este interfaz se puede dividir en tres partes; el panel de control digital, con el cual se eligen y despliegan las características de operación de la unidad; el panel de control analógico, con el cual se sintonizan los filtros, se controla el volumen, se ajusta el nivel de entrada al convertidor análogo-digital y se conectan las señales externas; y los controles de alimentación, en el cual se conecta la energía a la unidad, se enciende y se apaga y se restauran las condiciones iniciales el circuito.

1. Panel de control digital. Se agruparon las características de operación en opciones, quedando como se muestra en la tabla 5.2.

Tabla 5.2

Agrupación de las características de operación en opciones.

Opción	Descripción	Valores posibles	Despliegue
1	Función	directo, grabar, reproducir	LED
2	Sistema	PCM, DPCM, ADPCM, PCM con compansión	LED
3	Entrada	Micrófono, Línea	LED
4	Filtro de entrada	si, no	LED
5	Audio	si, no	LED
6	Filtro de salida	si, no	LED
7	Frecuencia de muestreo	0.2 - XX.X kHz	7 segmentos x 3
8	Bits de cuantificación	1 - 8	7 segmentos x 1
9	Número de muestra	1 - 4	7 segmentos x 1

Se agregó un dígito de 7 segmentos para indicar la opción que se estaba modificando. Se pudo notar que con cuatro botones era posible seleccionar y modificar las opciones: uno para cambiar de opción (acepta), dos para modificar la opción (+, -) y otro para aceptar todas las opciones y comenzar a trabajar (comenzar, terminar). Se agregó un LED para indicar si el circuito se encontraba trabajando. Este fue el momento en el cual se tuvo que escoger una caja para almacenar el circuito completo, pues el interfaz debería de caber en ella. Se eligió una caja de 6.5 cm de alto,

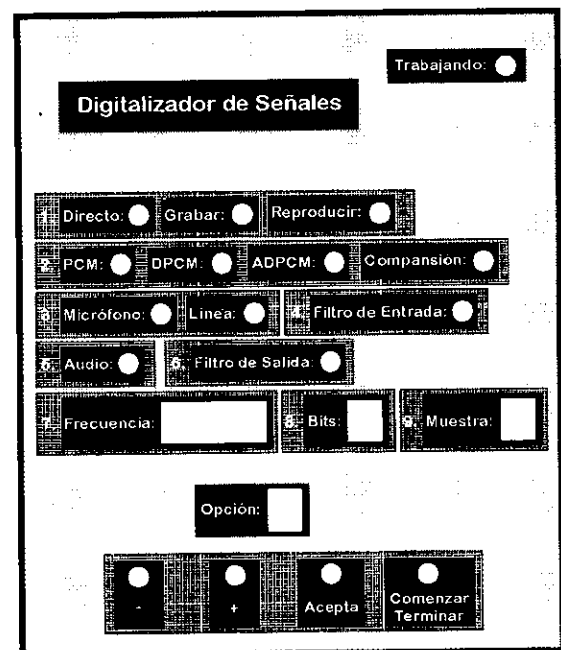


Figura 5.1. Panel de control digital.

Se eligió una caja de 6.5 cm de alto,

15.5 cm de ancho y 17.5 cm de fondo. El panel de control digital se colocaría en la parte superior de la caja; conociendo las medidas, se diseñó el panel y quedó como se muestra en la figura 5.1 (reducido a 46%). A los segmentos de los dígitos se les asignó el bit según se indica en la tabla 5.3.

Tabla 5.3

Bits asignados a los segmentos de los dígitos.

Bit	Segmento
7	a
6	b
5	c
4	d
3	e
2	f
1	g
0	no se usa

2. Panel de control analógico. El panel de control analógico se colocó en la parte frontal de la caja. Este quedó como se muestra en la figura 5.2 (reducido al 46%).

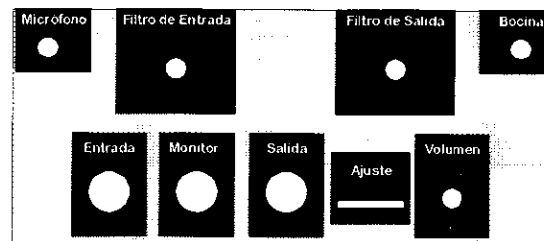


Figura 5.2. Panel de control analógico.

3. Control de alimentación. El control de alimentación se colocó en la parte posterior de la caja y quedó como se muestra en la figura 5.3 (reducido al 46%).

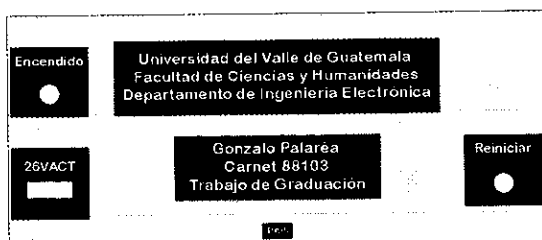


Figura 5.3. Control de alimentación.

E. Programación

Al tener toda la unidad ya alambrada (excepto los filtros, pues era necesario saber la frecuencia máxima de muestreo, la cual se iba a conocer hasta tener hecho el programa), se comenzó a trabajar con el programa, el cual se escribió en lenguaje ensamblador para el Z80, en el editor de textos del MS-DOS 5.0. Los pasos que se siguieron para el desarrollo se detallan a continuación. El listado del programa se incluye en el apéndice G.

1. Selección del puntero de pila (stack pointer). Se fijó el puntero de pila inicial en la dirección 8010_h . Como la RAM comienza en 8000_h , se tienen 16 bytes para usar en la pila.

2. Elección de variables. En la tabla 5.4 se detallan las variables que se utilizaron, su dirección de memoria y su contenido.

Tabla 5.4

Descripción de las variables

Nombre	Dirección	Contenido																		
OPC	0801B _h	OPCION: indica la opción que se está seleccionando.																		
BITS	0801C _h	BITS: la cantidad de bits que se van a utilizar en la reproducción de la señal.																		
MUES	0801D _h	MUESTRA: indica cual de las RAMS se va a utilizar para grabar la muestra																		
FREC	08011 _h	FRECUENCIA: la frecuencia de muestreo que se va a utilizar. El contenido de la variable, multiplicado por 100 da el valor de la frecuencia de muestreo a utilizar en Hz.																		
FREC1	08012 _h	FRECUENCIA DE LA MUESTRA 1: Frecuencia que se utilizará para grabar o reproducir la muestra 1.																		
FREC2	08013 _h	FRECUENCIA DE LA MUESTRA 2: Frecuencia que se utilizará para grabar o reproducir la muestra 2.																		
FREC3	08014 _h	FRECUENCIA DE LA MUESTRA 3: Frecuencia que se utilizará para grabar o reproducir la muestra 3.																		
FREC4	08015 _h	FRECUENCIA DE LA MUESTRA 4: Frecuencia que se utilizará para grabar o reproducir la muestra 4.																		
COND1	08016 _h	CONDICIONES 1: Los bits de esta variable significan lo siguiente (0=activado): <table border="1" data-bbox="551 1297 884 1716"> <thead> <tr> <th>Bit</th> <th>Contenido</th> </tr> </thead> <tbody> <tr> <td>7</td> <td>Directo</td> </tr> <tr> <td>6</td> <td>Grabar</td> </tr> <tr> <td>5</td> <td>Reproducir</td> </tr> <tr> <td>4</td> <td>PCM</td> </tr> <tr> <td>3</td> <td>DPCM</td> </tr> <tr> <td>2</td> <td>ADPCM</td> </tr> <tr> <td>1</td> <td>Compansión</td> </tr> <tr> <td>0</td> <td>No se usa</td> </tr> </tbody> </table>	Bit	Contenido	7	Directo	6	Grabar	5	Reproducir	4	PCM	3	DPCM	2	ADPCM	1	Compansión	0	No se usa
Bit	Contenido																			
7	Directo																			
6	Grabar																			
5	Reproducir																			
4	PCM																			
3	DPCM																			
2	ADPCM																			
1	Compansión																			
0	No se usa																			
COND2	08017 _h	CONDICIONES 2: Los bits de esta variable significan lo siguiente (0=activado): <table border="1" data-bbox="551 1808 839 1847"> <thead> <tr> <th>Bit</th> <th>Contenido</th> </tr> </thead> </table>	Bit	Contenido																
Bit	Contenido																			

	7	Micrófono
	6	Línea
	5	Filtro de Entrada
	4	Audio
	3	Filtro de Salida
	2	No se usa
	1	Trabajando
	0	No se usa
RUTINT	08018 _h	Dirección de rutina de interrupción del CTC.
FRECD	08020 _h	Frecuencia utilizada al grabar.
FRECD1	08021 _h	Frecuencia utilizada al grabar muestra 1.
FRECD2	08022 _h	Frecuencia utilizada al grabar muestra 2.
FRECD3	08023 _h	Frecuencia utilizada al grabar muestra 3.
FRECD4	08024 _h	Frecuencia utilizada al grabar muestra 4.

3. Rutinas de inicialización. A continuación se escribieron las rutinas de inicialización, las cuales se listan y explican en la tabla 5.5.

Tabla 5.5

Rutinas de inicialización

Nombre	Descripción
CERO	Deshabilita interrupciones, inicializa puntero de pila, fija parte alta del vector de interrupciones (registro I), fija modo de interrupción 2.
IPIO	Inicializa PIO: en ambos puertos fija modo 2, deshabilita interrupciones y pone salidas a 0.
INVAL	Inicializa valores de variables: 80 (8 kHz) en todas las frecuencias, 8 bits, muestra 1, opción 1, directo, PCM, micrófono.
IKD	Inicializa 8279: 8 caracteres con entrada por la izquierda, teclado codificado, divisor 20, borra RAM de caracteres.
ICTC	Inicializa CTC: Fija vector de interrupción.

4. Caracteres para los 7 segmentos. Ya teniendo los dígitos de 7 segmentos alambrados, se pudo asignar los valores correspondientes para los números que se iban a desplegar. Así, la tabla de caracteres quedó como se indica en la tabla 5.6, y se escribió en el programa bajo el nombre de NUMEROS.

Tabla 5.6

Tabla de caracteres

Caracter	Valor
0	00000011 _b
1	10011111 _b
2	00100101 _b
3	00001101 _b
4	10011001 _b
5	01001001 _b
6	01000001 _b
7	00011111 _b
8	00000001 _b
9	00001001 _b
Blanco	11111111 _b

5. Cálculo de los divisores de frecuencia para el CTC. Sabiendo que la frecuencia de muestreo vendría dada por el contenido de la variable FREC (de 1 byte) multiplicado por 100, la frecuencia máxima que se podría tener era de 25.4 kHz. Así, se escribió un programa en BASIC para calcular los divisores de frecuencia que, al dividir los 2.5 MHz daría las frecuencias de 200 a 25400 Hz en pasos de 200Hz. El programa se lista en el apéndice D, y los divisores que se utilizaron en el apéndice E. Los divisores se incluyeron en el programa en la tabla DIVISOR.

6. Valores para la compresión digital. Como la compansión se trabajaría con el método de compresión digital, comprimiendo de 8 a 5 bits, sería necesario asignar 32 niveles. Así, los 16 niveles centrales se separaron por 4 números, los 8 niveles siguientes (4 hacia arriba y 4 hacia abajo) se separaron por 8 números y los restantes 8 niveles se separaron por 16 números. En la figura 5.4 se puede apreciar la forma en que se asignaron los valores, teniendo en el eje 'X' el valor a 8 bits, y en el eje 'Y' el valor comprimido.

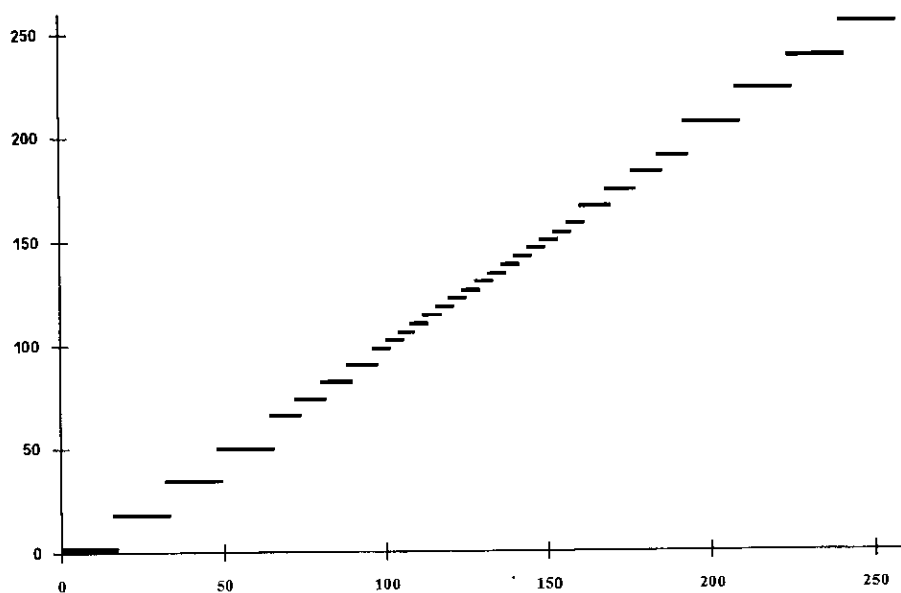


Figura 5.4. Valores utilizados en la compresión digital para lograr la compansión.

En el apéndice F se listan los valores utilizados para la compansión, a partir de los cuales se obtiene la gráfica de la figura 5.4.

7. Rutinas de soporte. Luego se escribieron las rutinas de soporte, las cuales sirven para leer el teclado y desplegar caracteres en los dígitos de 7 segmentos. Estas rutinas se listan y explican en la tabla 5.7.

Tabla 5.7

Rutinas de soporte

Nombre	Descripción
ESCNUME	Escribe en la posición del 8279 dada por el registro C, el caracter según la tabla NUMEROS dado por el registro D.
LEETEC	Borra la cola de teclado del 8279, hace un bucle hasta que se recibe un teclazo, y lo ajusta para que quede en los 2 bits inferiores.
ACTDISP	Actualiza los LEDs, los 7 segmentos y las salidas del PIO que controlan los switches analógicos, según opciones elegidas. Si está en DIRECTO, no despliega MUESTRA. Si está en GRABAR, forza a 8 bits y no enciendo ninguno de los LEDS de la opción 2 ni el del Filtro de Salida ni el de Audio.. Si está REPRODUCIR, no enciende el LED del Filtro de Entrada ni los de la opción 3. Si está en ADPCM, forza a 1 bit. Si está en COMPANSION, forza a 5 bits.
ICTCI	Inicializa el CTC como contador con interrupciones, con las constantes que obtiene de la tabla DIVISOR según el contenido de la variable FREC
ICTCN	Inicializa el CTC como contador sin interrupciones, hace una reinicialización del canal y carga 0 en el registro C, lo cual se utiliza para avisar que se usó esta rutina y se debe terminar de trabajar.

8. Rutinas de interrupción del CTC. Se prosiguió a escribir las rutinas de interrupción del CTC. Estas son las rutinas que hacen el trabajo que se elige con el panel de control digital. Se listan y explican en la tabla 5.8.

Tabla 5.8

Rutinas de interrupción del CTC

Nombre	Descripción
DIRPCM1	PCM directo a 1 bit. Pide conversión al ADC, la lee, copia el bit más significativo a los demás, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM2	PCM directo a 2 bits. Pide conversión al ADC, la lee, copia los bits 7-6 a los bits 5-4, 3-2, 1-0, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM3	PCM directo a 3 bits. Pide conversión al ADC, la lee, copia los bits 7-6-5 a los bits 4-3-2, y los bits 7-6 a los bits 1-0, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM4	PCM directo a 4 bits. Pide conversión al ADC, la lee, copia los bits 7-6-5-4 a los bits 3-2-1-0, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM5	PCM directo a 5 bits. Pide conversión al ADC, la lee, copia los bits 7-6-5 a los bits 2-1-0, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM6	PCM directo a 6 bits. Pide conversión al ADC, la lee, copia los bits 7-6 a los bits 1-0, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM7	PCM directo a 7 bits. Pide conversión al ADC, la lee, copia el bit 7 al bit 0, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRPCM8	PCM directo a 8 bits. Pide conversión al ADC, la lee, saca el resultado al DAC y chequea si se ha presionado una tecla para terminar.
DIRDPCM1	DPCM directo a 1 bit. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta 1 al acumulado (sin permitir pasar de 255 ni

- bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM2** DPCM directo a 2 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta 1 o 2 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM3** DPCM directo a 3 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 4 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM4** DPCM directo a 4 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 8 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM5** DPCM directo a 5 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 16 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM6** DPCM directo a 6 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 32 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM7** DPCM directo a 7 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 64 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRDPCM8** DPCM directo a 8 bits. Pide conversión al ADC, la lee, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 128 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC y chequea si se ha presionado una tecla para terminar.
- DIRADPCM** PCM Delta Adaptativo directo. Pide conversión al ADC, la lee, compara con el valor acumulao (registro D) y dependiendo de la comparación anterior, ejecuta lo siguiente:

Valor actual > acumulado y valor anterior > acumulado anterior:
incrementa sumando y lo suma

Valor actual < acumulado y valor anterior < acumulado anterior:
incrementa sumando y lo resta

Valor actual < acumulado y valor anterior > acumulado anterior:
pone sumando a 0

Valor actual > acumulado y valor anterior < acumulado anterior:
pone sumando a 0

Chequea si se ha presionado una tecla para terminar.

- DIRCOMN Directo con expansión. Pide conversión al ADC, la lee, divide dentro de 4 y con este resultado obtiene el valor adecuado de la tabla COMVAL, el cual saca al DAC. Chequea si se ha presionado una tecla para terminar.
- REPPCM1 PCM reproduciendo a 1 bit. Lee valor de la RAM, copia el bit más significativo a los demás, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM2 PCM reproduciendo a 2 bits. Lee valor de la RAM, copia los bits 7-6 a los bits 5-4, 3-2, 1-0, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM3 PCM reproduciendo a 3 bits. Lee valor de la RAM, copia los bits 7-6-5 a los bits 4-3-2, y los bits 7-6 a los bits 1-0, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM4 PCM reproduciendo a 4 bits. Lee valor de la RAM, copia los bits 7-6-5-4 a los bits 3-2-1-0, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM5 PCM reproduciendo a 5 bits. Lee valor de la RAM, copia los bits 7-6-5 a los bits 2-1-0, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM6 PCM reproduciendo a 6 bits. Lee valor de la RAM, copia los bits 7-6 a los bits 1-0, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM7 PCM reproduciendo a 7 bits. Lee valor de la RAM, copia el bit 7 al bit 0, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPPCM8 PCM reproduciendo a 8 bits. Lee valor de la RAM, saca el resultado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.

- REPDPCM1 DPCM reproduciendo a 1 bit. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta 1 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM2 DPCM reproduciendo a 2 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta 1 o 2 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM3 DPCM reproduciendo a 3 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 4 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM4 DPCM reproduciendo a 4 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 8 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM5 DPCM reproduciendo a 5 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 16 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM6 DPCM reproduciendo a 6 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 32 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM7 DPCM reproduciendo a 7 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 64 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
- REPDPCM8 DPCM reproduciendo a 8 bits. Lee valor de la RAM, obtiene diferencia con el valor acumulado (registro D) y dependiendo de la diferencia suma o resta de 1 a 128 al acumulado (sin permitir pasar de 255 ni bajar de 0); saca el acumulado al DAC, incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.

REPADPCM	PCM Delta Adaptativo reproduciendo. Lee valor de la RAM, compara con el valor acumulado (registro D) y dependiendo de la comparación anterior, ejecuta lo siguiente: Valor actual > acumulado y valor anterior > acumulado anterior: incrementa sumando y lo suma Valor actual < acumulado y valor anterior < acumulado anterior: incrementa sumando y lo resta Valor actual < acumulado y valor anterior > acumulado anterior: pone sumando a 0 Valor actual > acumulado y valor anterior < acumulado anterior: pone sumando a 0 Incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
REPCOMN	Reproduciendo con compansión. Lee valor de la RAM, divide dentro de 4 y con este resultado obtiene el valor adecuado de la tabla COMVAL, el cual saca al DAC. Incrementa el puntero (HL) y chequea si llegó al final de la RAM para terminar.
GRABAR	Graba muestra. Pide conversión al ADC, la lee, la guarda en posición indicada por HL, incrementa puntero (HL), chequea si llegó al final de la RAM para terminar.

9. Cálculo de la frecuencia máxima de muestreo. Teniendo las rutinas de interrupción ya escritas, se pudo calcular cuánto tiempo demora darles servicio, y encontrar la más tardada, ya que ésta es la que limita la frecuencia de muestreo. Las rutinas más tardadas fueron las de PCM diferencial, las cuales se podrían repetir a una frecuencia de 20.2 kHz; para dar un poco de holgura se fijó la frecuencia máxima en 20.0 kHz. En el apéndice H se muestran los cálculos de los tiempos que tardan las rutinas. Fue en este momento cuando se pudo calcular los filtros y terminar de alambrar el circuito.

10. Rutinas de selección de opciones. Las rutinas de selección de opciones se listan y explican en la tabla 5.9.

Tabla 5.9

Rutinas de selección de opciones

Nombre	Descripción
OP1	Opción 1. Permite elegir los parámetros de la opción 1 (Directo, Grabar, Reproducir). Cambia la selección con las teclas “-” y “+”. Con la tecla “Acepta” salta a otra rutina, donde llama a las demás opciones según la selección hecha (Directo: opciones 2, 3, 4, 5, 6, 7, y 8 si no es ADPCM; Grabar: opciones 9, 3, 4 y 7; Reproducir: opciones 9, 2, 5, 6, 7 y 8 si no es ADPCM. Con “Comenzar/Terminar” comienza a trabajar.
OP2	Opción 2. Permite elegir los parámetros de la opción 2 (PCM, DPCM, ADPCM, Compansión). Cambia la selección con las teclas “-” y “+”. Con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP3	Opción 3. Permite elegir los parámetros de la opción 3 (Micrófono, Línea). Cambia la selección con las teclas “-” y “+”. Con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP4	Opción 4. Permite elegir entre utilizar filtro de entrada o no. Cambia la selección con las teclas “-” y “+”. Con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP5	Opción 5. Permite elegir entre utilizar audio o no. Cambia la selección con las teclas “-” y “+”. Con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP6	Opción 6. Permite elegir entre utilizar filtro de salida o no. Cambia la selección con las teclas “-” y “+”. Con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP7	Opción 7. Fijar la frecuencia. Con la tecla “+” se incrementa el valor en 2 (200Hz), sin pasar de 200 (20kHz); con la tecla “-” se decrementa el valor en 2, sin bajar de 2; con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP8	Opción 8. Fijar cantidad de Bits. Con la tecla “+” se incrementa el valor en , sin pasar de 8; con la tecla “-” se decrementa el valor en 1, sin bajar de 1; con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.
OP9	Opción 9. Fijar memoria a utilizar. Con la tecla “+” se incrementa el valor en , sin pasar de 4; con la tecla “-” se decrementa el valor en 1, sin bajar de 1; con “Acepta” regresa y con “Comenzar/Terminar” comienza a trabajar.

11. Rutina de corrida según opciones. Al presionar la tecla “Comenzar/Terminar” en cualquiera de las opciones, el programa salta a la rutina FINOPC, la cual busca la dirección de la rutina de interrupción a utilizar según las condiciones que se eligieron y la graba en la variable que se asignó para ese fin y la graba en las 4 RAMs, así no importará cuál de ellas se utilice. Para facilitar la búsqueda de las direcciones de las rutinas, se agregó una tabla al final del programa, conteniéndolas. Estas tablas se nombraron “RDIRPCM” (rutinas PCM directo), “RDIRDPCM” (rutinas DPCM directo), “RREPPCM” (rutinas PCM reproduciendo) y “RREPDPCM” (rutinas DPCM reproduciendo). Luego enciende el LED de “Trabajando” y llama a la rutina de soporte ICTCI. A continuación elige la RAM a utilizar según la variable MUES y fija los valores iniciales a utilizar en las rutinas de interrupción.

Después de fijar todas las condiciones para trabajar, el programa entra en un bucle (con la etiqueta LOOP:) en el cual habilita las interrupciones, chequea si ya se utilizó la rutina ICTCN (con el registro C), y si ya se utilizó, sale del bucle. Si no, hace un HALT, con lo cual espera hasta que llegue otra interrupción. Al terminar de darle servicio a la interrupción, comienza el bucle de nuevo.

Al salir del bucle, selecciona la RAM 1 nuevamente (en la cual se guardan todas las variables), revisa si la acción recién terminada fue “Grabar”, y si lo fue, guarda la frecuencia que se utilizó. Luego de esto, el programa regresa a la opción 1.

12. Programa principal. El programa principal consta únicamente de un bucle que llama a la rutina OPC1, ya que ésta llama a las demás.

13. Compilación del programa y escritura en ROM. La compilación se realizó con el programa “William Comput Z80 Assembler”, el cual al compilar genera 2 archivos:

uno conteniendo el programa fuente con los datos compilados en una columna a la par (“TE.PRN”), y otro conteniendo únicamente los datos compilados, en formato Intel HEX (“TE.HEX”). Ya teniendo el programa compilado, se constató que la dirección más alta que se utilizó fue la $07C2_h$ (1986), con lo cual quedaron libres 62 bytes de la ROM.

Para pasar los datos a la ROM, se diseñó un circuito para ser conectado al puerto paralelo de la PC. Este circuito copia los bytes que se sacan por el puerto, secuencialmente en la ROM. El diagrama de este circuito se adjunta en el apéndice I. Fue necesario, además, escribir un programa en BASIC para tomar los datos del archivo “TE.HEX” y pasarlos al puerto paralelo. Dicho programa se lista en el apéndice J.

VI. RESULTADOS

Al tener el dispositivo terminado se procedió a observar con el osciloscopio la salida obtenida al conectarle distintas señales de entrada y fijarle distintos parámetros de digitalización. En el presente capítulo se presentan algunas de estas observaciones, las cuales fueron tomadas con un osciloscopio Fluke 97 y trasladadas por el puerto serial a la computadora personal. En todas las figuras, la señal de entrada se muestra en la parte superior, y la de salida en la inferior.

En la figura 6.1 se muestra una señal senoidal de 1 kHz, digitalizada con $f_m=8\text{kHz}$, utilizando el método PCM a 8 bits y sin filtro de salida. Se pueden observar los niveles de cuantificación. En la figura 6.2 se observa la misma señal, pero con el filtro de salida activado y sintonizado a 1.5 kHz. En esta figura se puede notar cómo el filtro de salida hace que la señal sea reconstruida.

En las figuras 6.3 y 6.4 se puede observar el fenómeno del submuestreo, con el cual se obtiene en la salida una señal de frecuencia distinta a la de la señal de entrada.

En las figuras 6.5 se observa una señal senoidal de 3.5kHz siendo muestreada con $f_m=8\text{ kHz}$, con el filtro de salida desactivado. En la figura 6.6 se tiene la misma señal, con el filtro de salida sintonizado a 4 kHz. Se puede observar el efecto que provoca utilizar un filtro no ideal, pues la señal de salida no contiene únicamente la frecuencia de la de entrada, sino que además tiene otra frecuencia, la cual proviene del proceso de muestreo.

En las figuras 6.7, VI-8 y VI-9 se presentan señales triangulares siendo muestreadas a $f_m=12\text{kHz}$, utilizando PCM a 3, 6 y 8 bits, respectivamente. Se pueden observar los niveles de cuantificación.

En las figuras 6.10, 6.11 y 6.12 se hace una comparación del sistema DPCM y ADPCM, muestreando una señal cuadrada de frecuencia 500Hz con $f_m=12\text{kHz}$. En la figura 6.10 se tiene el sistema DPCM a 1 bit, y se observa que la salida no sigue satisfactoriamente a la entrada. En la figura 6.11 se presenta el sistema DPCM a 4 bits, el cual puede seguir un poco mejor a la entrada. En la figura 6.12 se observa el método ADPCM a 1 bit, con el cual se obtiene un resultado comparable al de la figura 6.11, utilizando este método un ancho de banda 4 veces menor que el anterior.

En las figuras 6.13 y 6.14 se muestra el método ADPCM con $f_m=16\text{kHz}$ para una señal de entrada senoidal de frecuencia 400Hz, con el filtro de desactivado y con éste sintonizado a 400Hz, respectivamente. Se puede apreciar que el método es capaz de reproducir satisfactoriamente la señal de entrada.

En la figura 6.15 se muestra el sistema PCM con compansión a 5 bits y $f_m=20\text{kHz}$, para una señal de entrada triangular de frecuencia 300Hz. Se puede apreciar que la separación entre los niveles para valores cercanos al centro es pequeña, mientras que dicha separación en los niveles de mayor amplitud es mayor.

Además de las pruebas que se hicieron con el generador de funciones, se hicieron otras con el micrófono como entrada. Es interesante escuchar como la voz se va degradando mientras se baja la frecuencia de muestreo y la cantidad de bits con el método PCM. También resulta claro cómo mejora la voz al activar el filtro de salida. El método DPCM se probó para distintas frecuencias de muestreo y bits de cuantificación, obteniendo mejores resultados para las frecuencias de muestreo mayores y mayor cantidad de bits. El método ADPCM fue capaz de reproducir la voz claramente, utilizando $f_m=16\text{kHz}$ o mayor, y el filtro de salida sintonizado a 4 kHz. Fue también interesante comparar el método PCM normal a 5 bits y el PCM con compansión a 5 bits, resultando más clara la voz con el segundo.

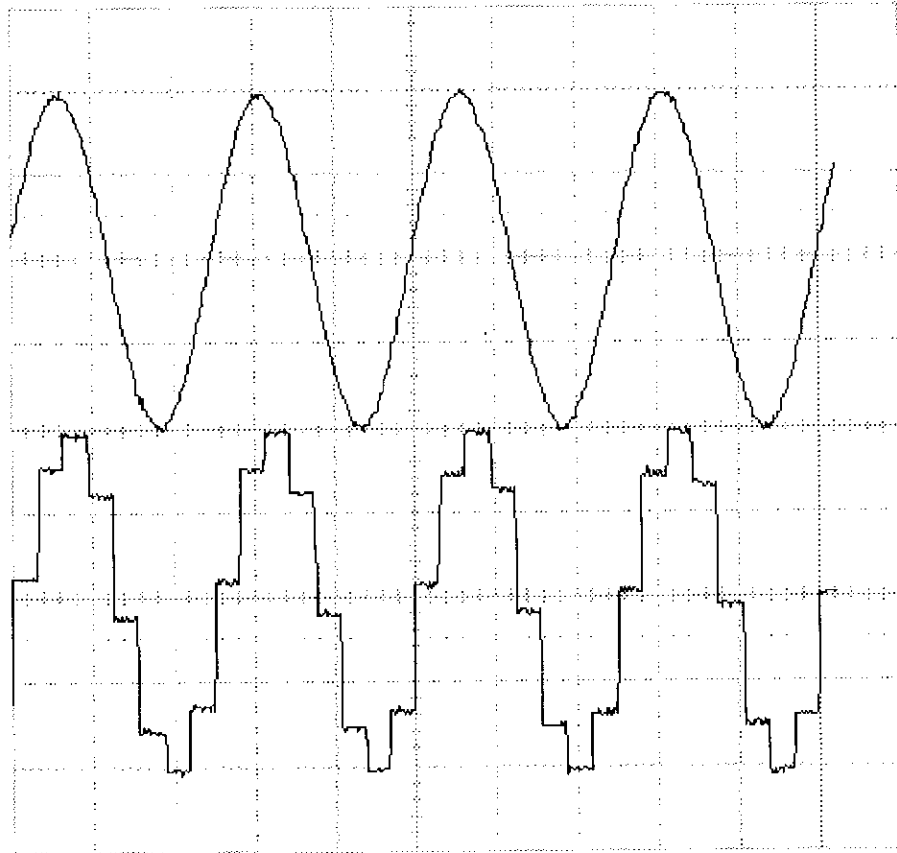


Figura 6.1. Señal senoidal de frecuencia 1kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando el método PCM a 8 bits, sin filtro de salida.

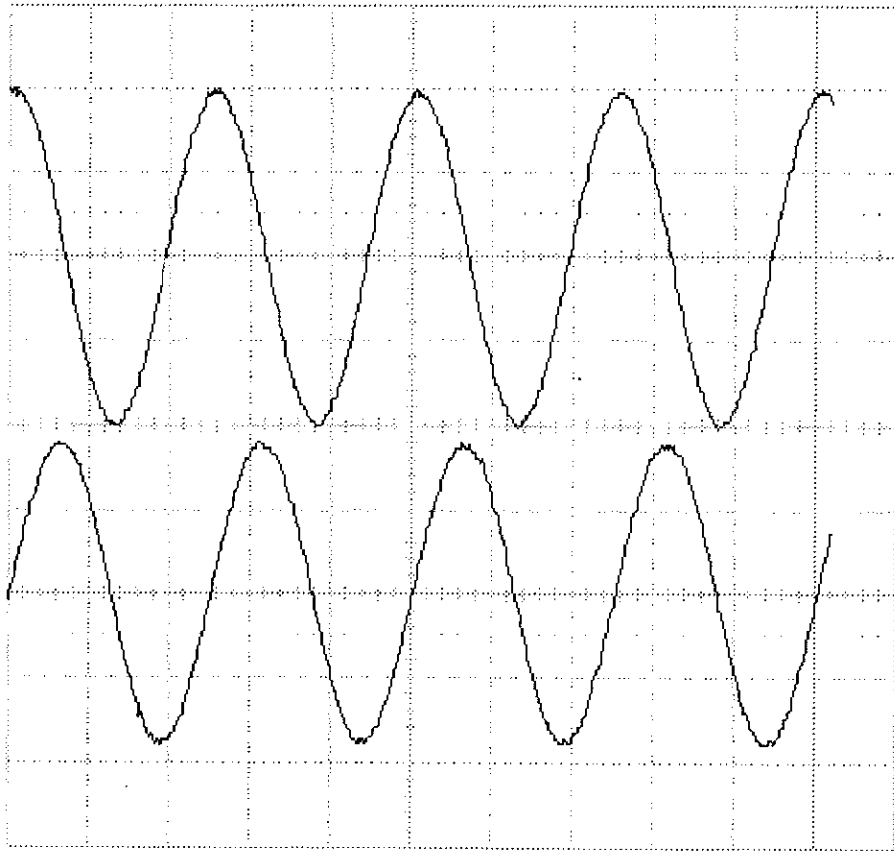


Figura 6.2. Señal senoidal de frecuencia 1kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando el método PCM a 8 bits, con filtro de salida sintonizado a 1.5kHz.

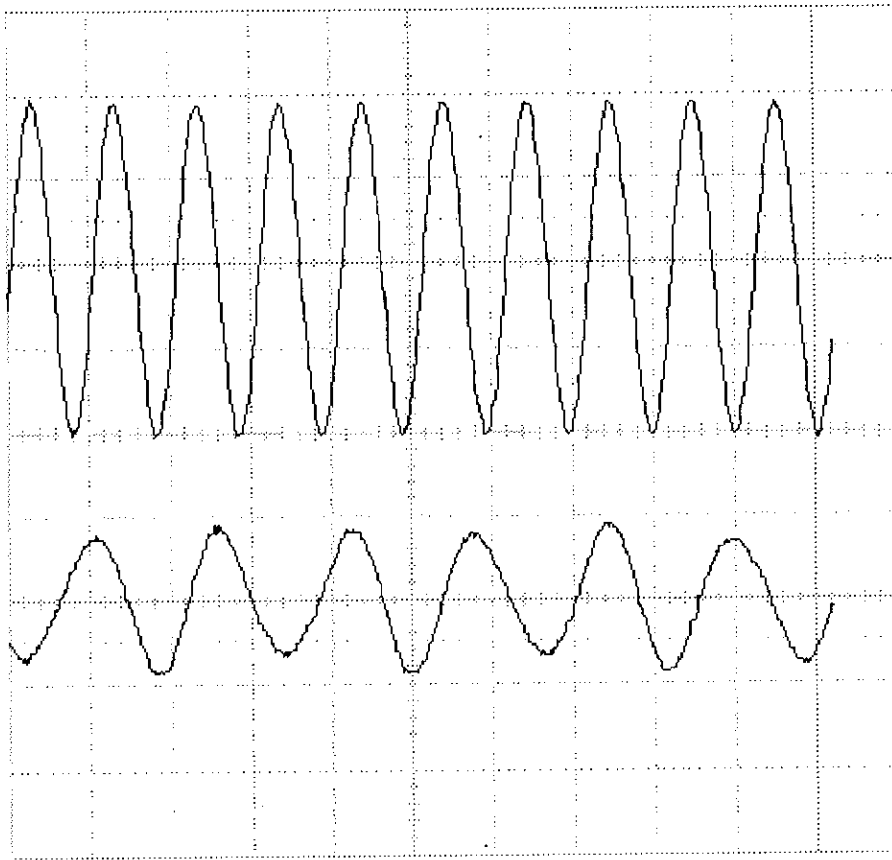


Figura 6.3. Señal senoidal de frecuencia 5kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando el método PCM a 8 bits, con filtro de salida sintonizado a 4kHz.

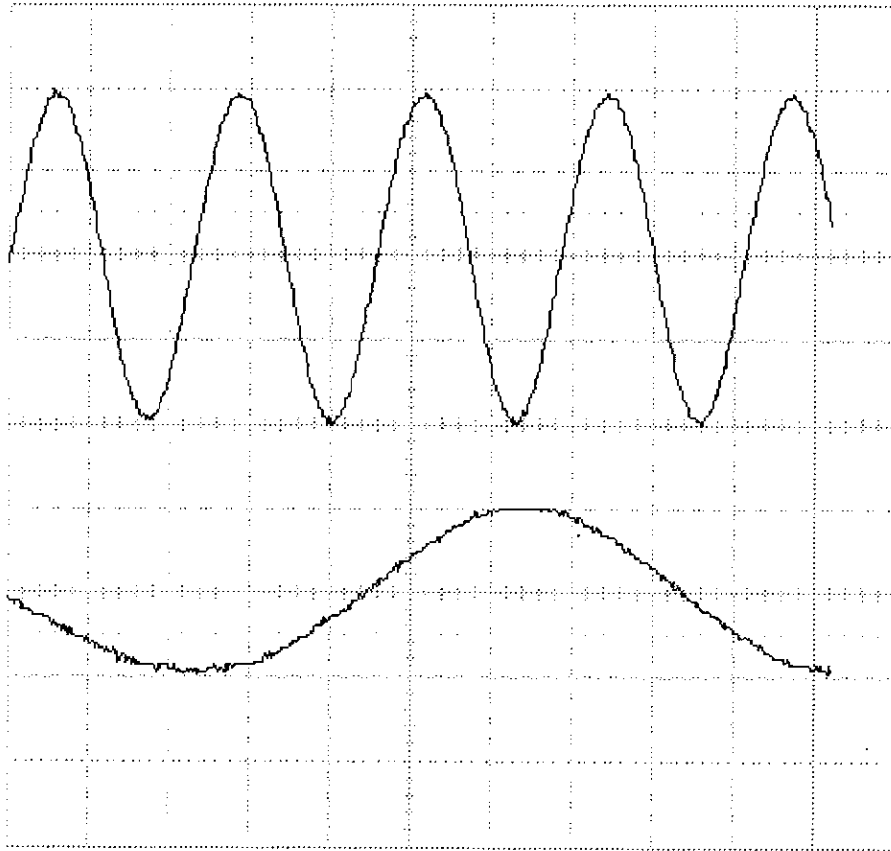


Figura 6.4. Señal senoidal de frecuencia 11kHz, 4Vpp, muestreada con $f_m=8$ kHz, utilizando el método PCM a 8 bits, con filtro de salida sintonizado a 4kHz.

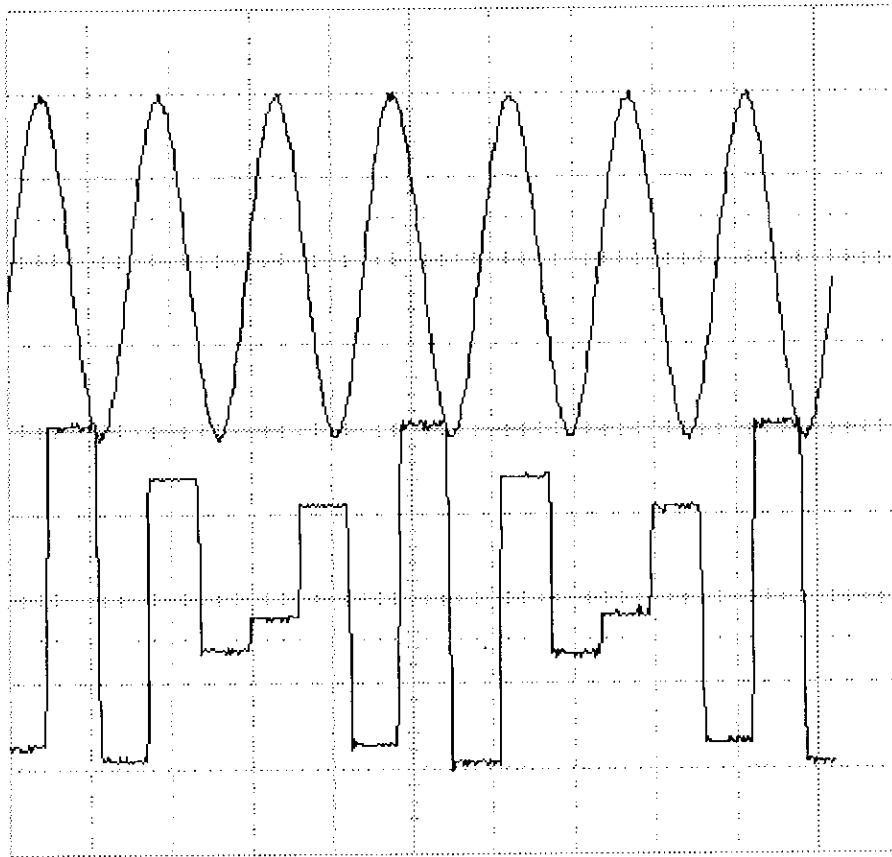


Figura 6.5. Señal senoidal de frecuencia 3.5kHz, 4Vpp, muestreada con $f_m=8$ kHz, utilizando el método PCM a 8 bits, con filtro de salida desactivado.

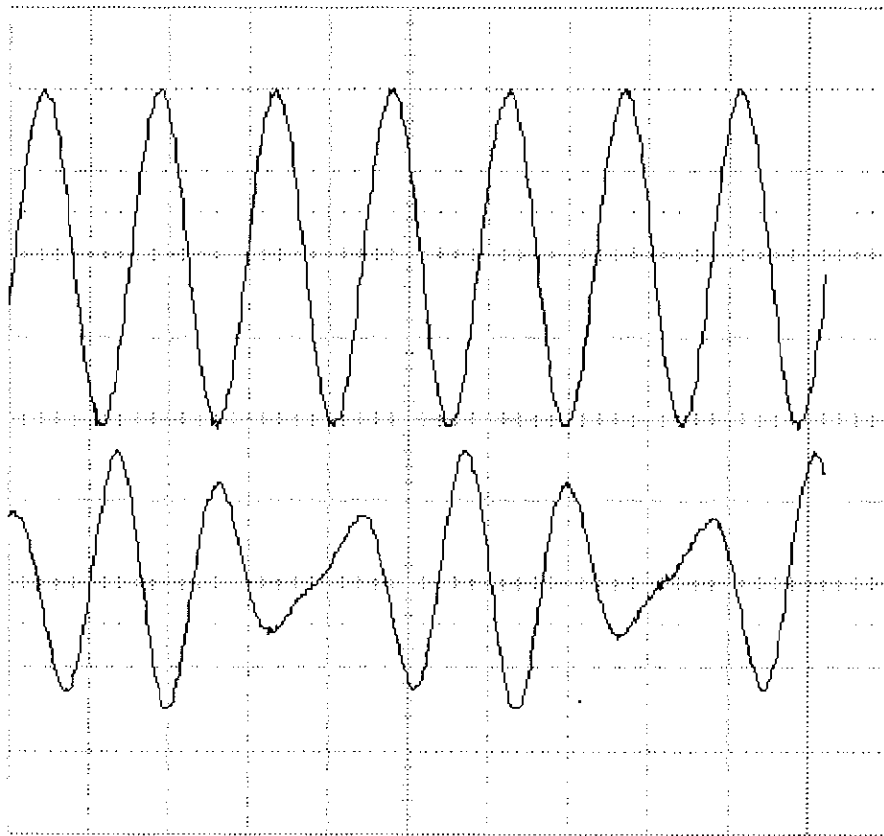


Figura 6.6. Señal senoidal de frecuencia 3.5kHz, 4Vpp, muestreada con $f_m=8\text{kHz}$, utilizando el método PCM a 8 bits, con filtro de salida sintonizado a 4kHz.

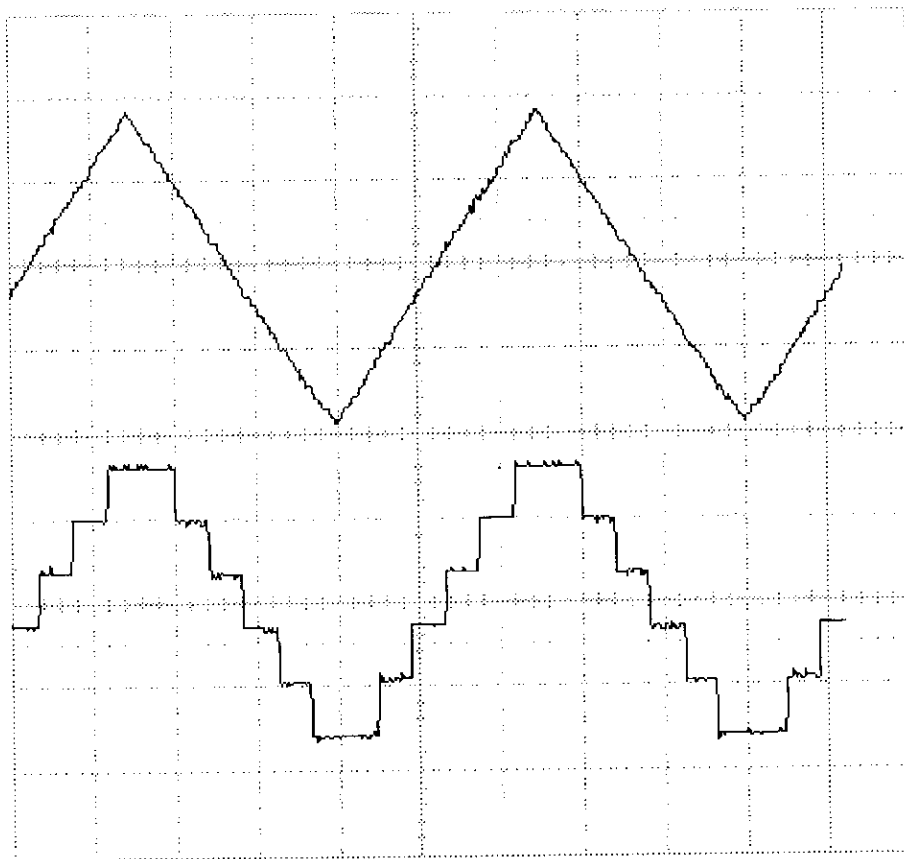


Figura 6.7. Señal triangular de frecuencia 500Hz, 4Vpp, muestreada con $f_m=12\text{kHz}$, utilizando el método PCM a 3 bits, con filtro de salida desactivado.

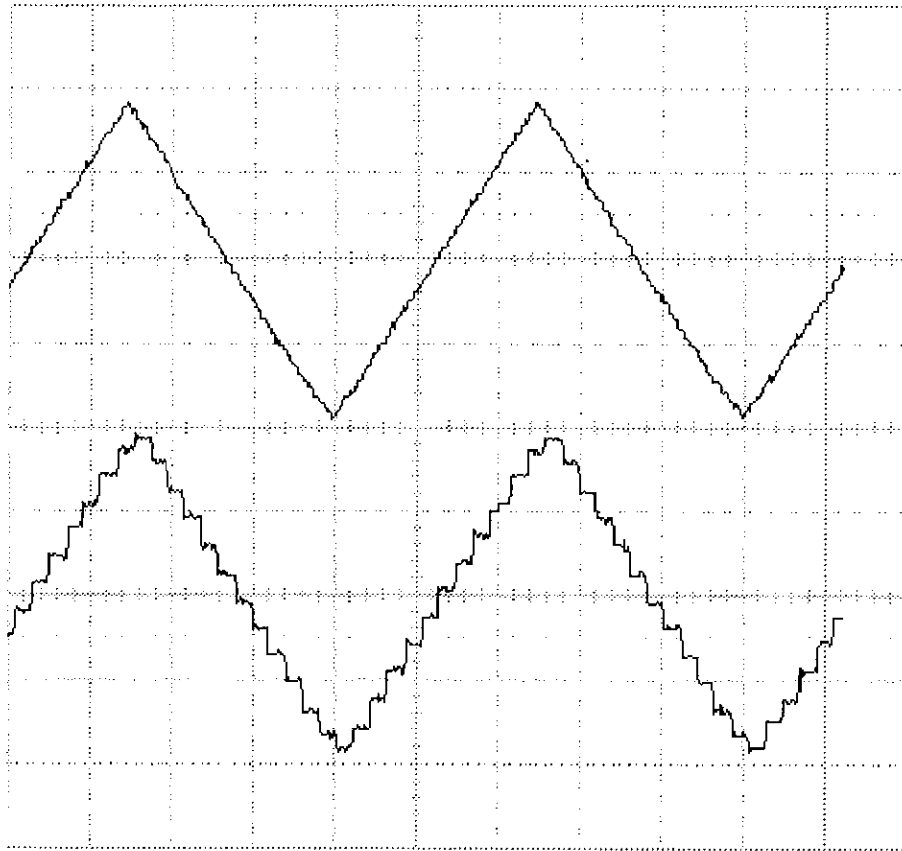


Figura 6.8. Señal triangular de frecuencia 500Hz, 4Vpp, muestreada con $f_m=12\text{kHz}$, utilizando el método PCM a 6 bits, con filtro de salida desactivado.

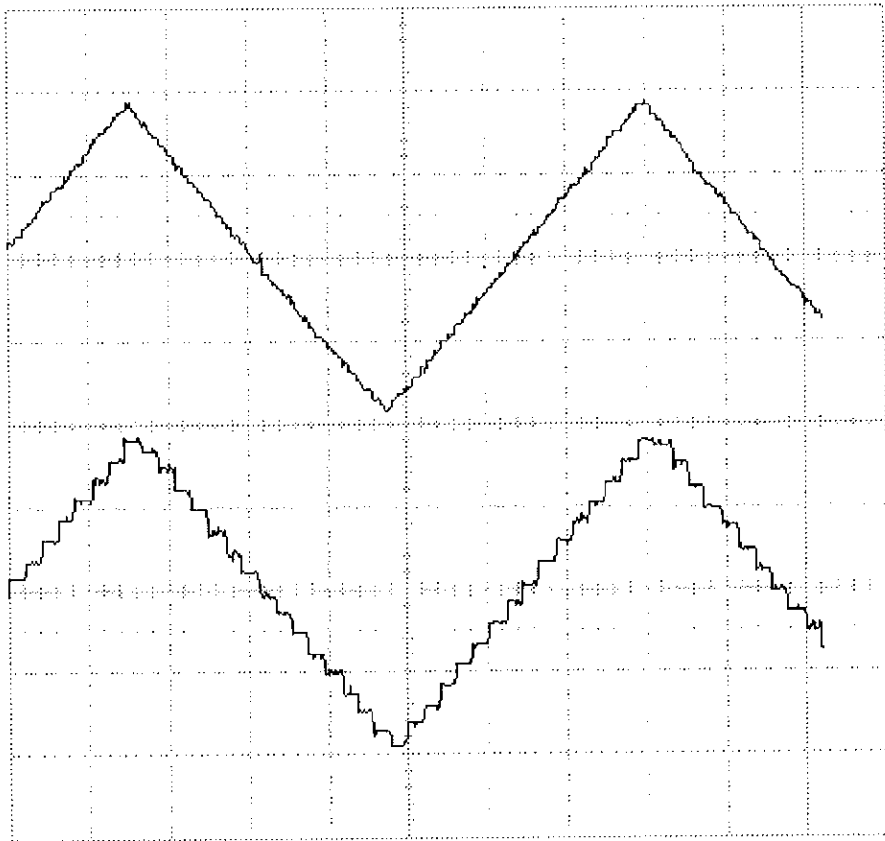


Figura 6.9. Señal triangular de frecuencia 400Hz, 4Vpp, muestreada con $f_m=12\text{kHz}$, utilizando el método PCM a 8 bits, con filtro de salida desactivado.

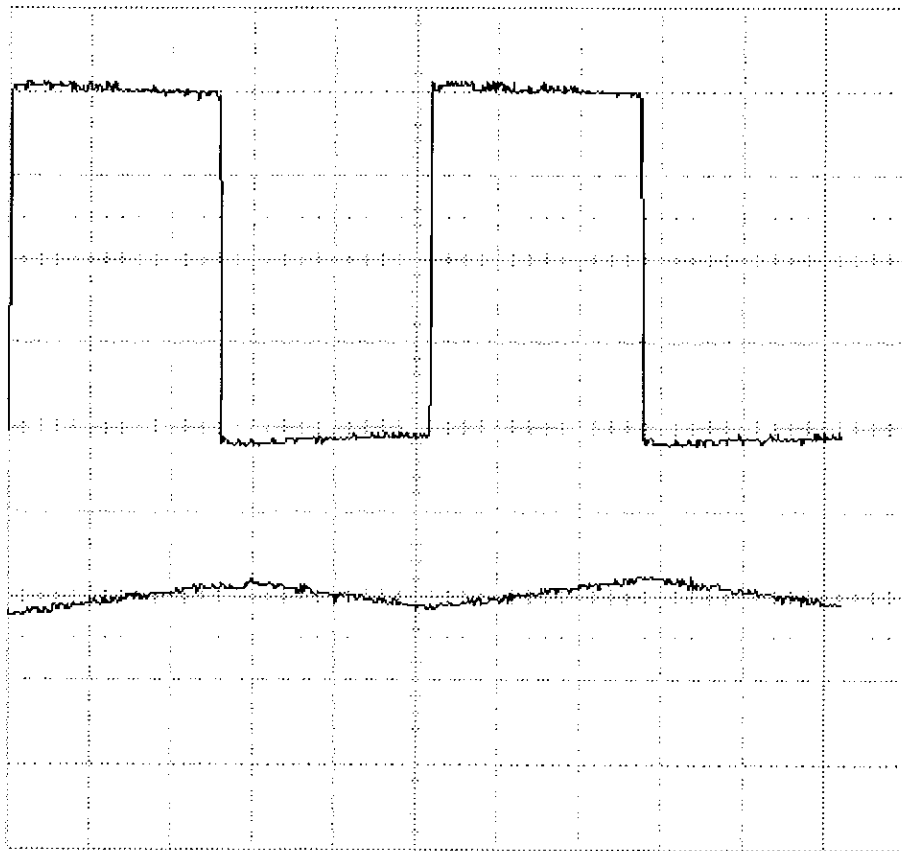


Figura 6.10. Señal cuadrada de frecuencia 500Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando el método DPCM a 1 bit, con filtro de salida desactivado.

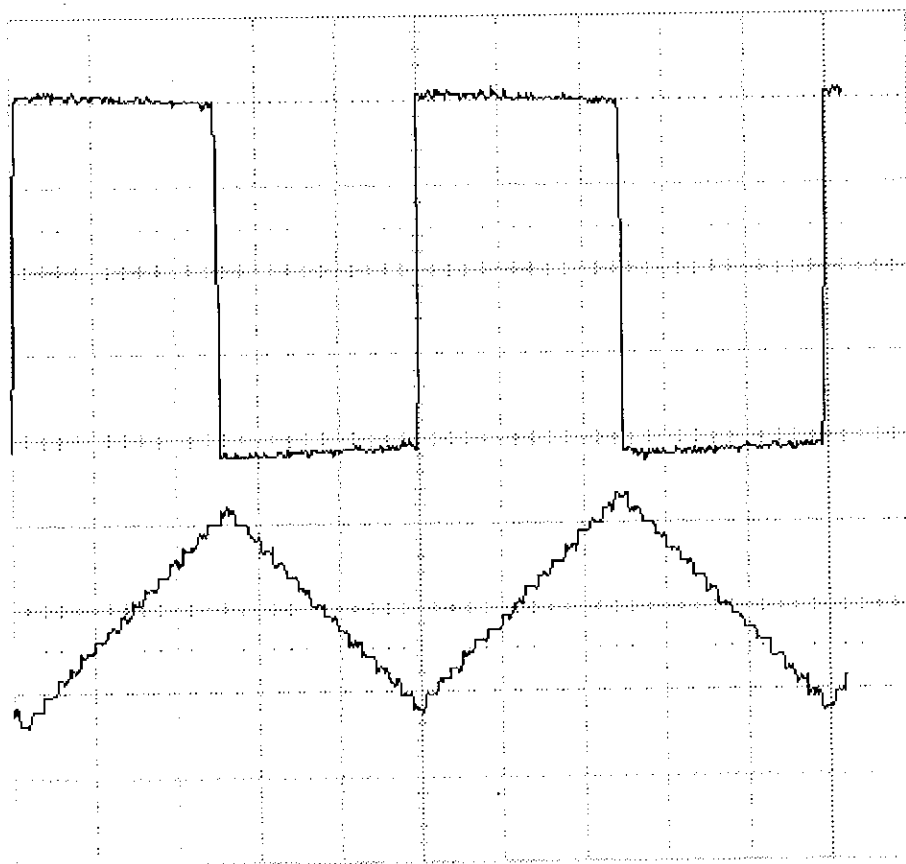


Figura 6.11. Señal cuadrada de frecuencia 500Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando el método DPCM a 4 bits, con filtro de salida desactivado.

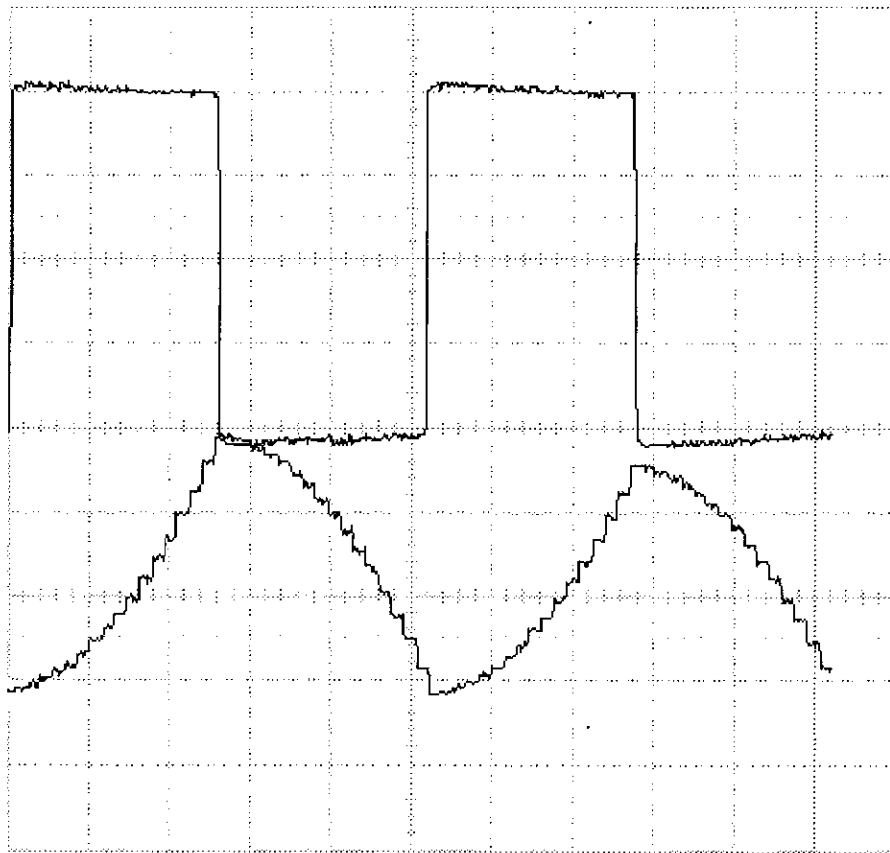


Figura VI-12. Señal cuadrada de frecuencia 500Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando el método ADPCM (a 1 bit), con filtro de salida desactivado.

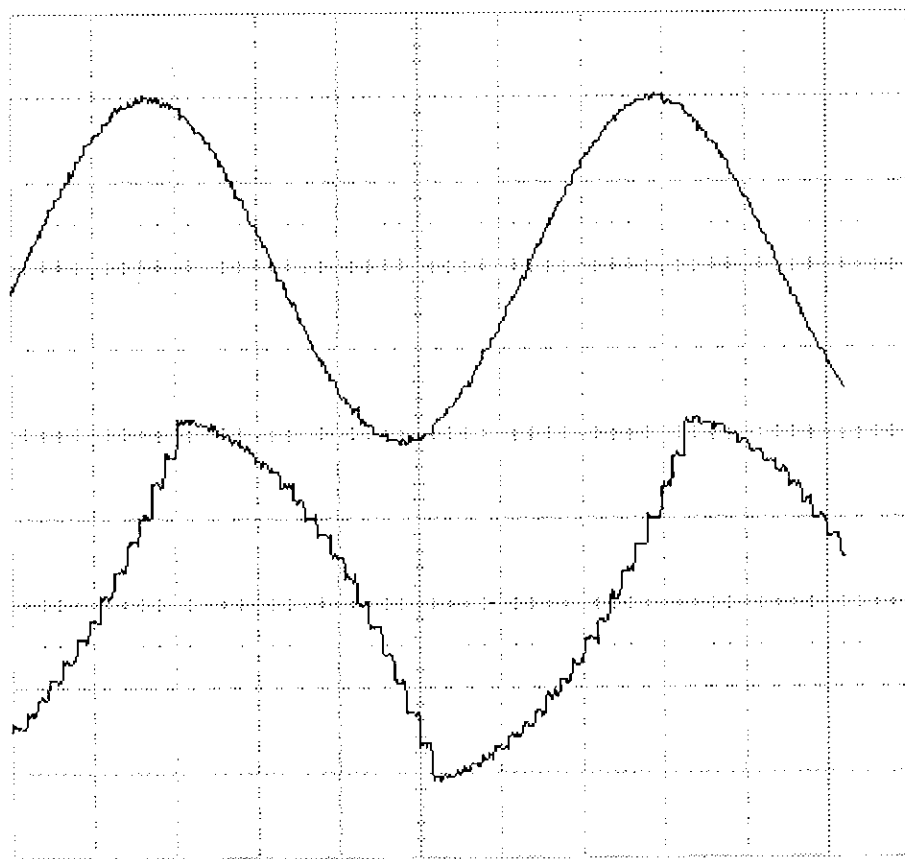


Figura 6.13. Señal senoidal de frecuencia 400Hz, 4Vpp, muestreada con $f_m=16\text{kHz}$, utilizando el método ADPCM (a 1 bit), con filtro de salida desactivado.

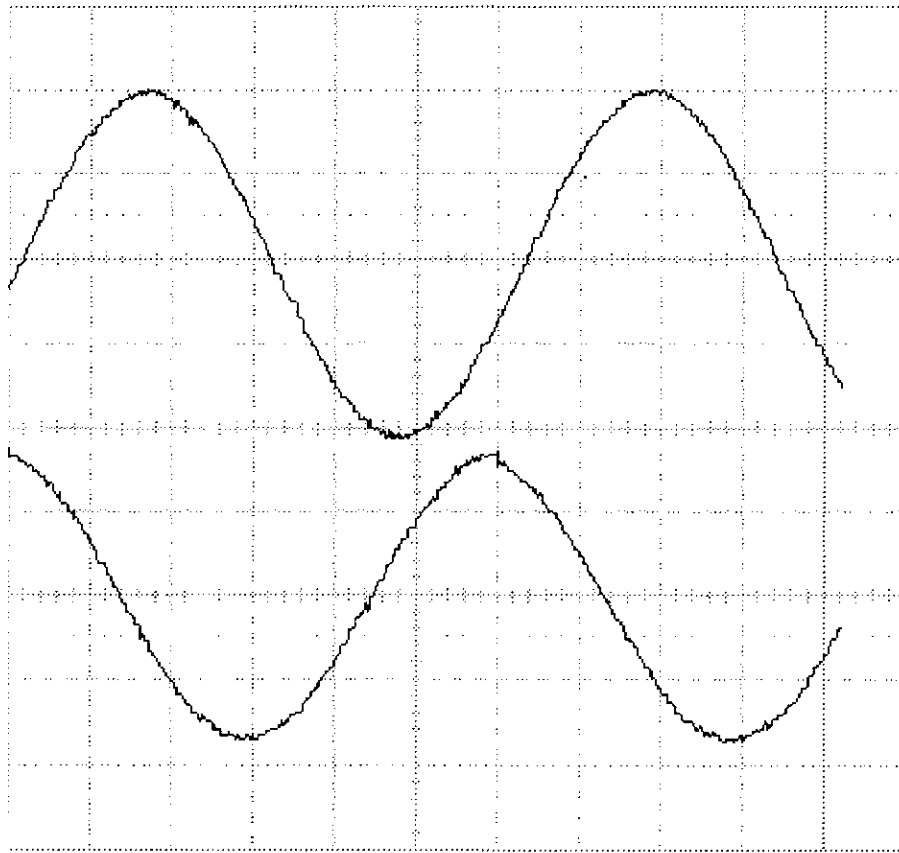


Figura 6.14. Señal senoidal de frecuencia 400Hz, $4V_{pp}$, muestreada con $f_m=16\text{kHz}$, utilizando el método ADPCM (a 1 bit), con filtro de salida sintonizado a 400Hz.

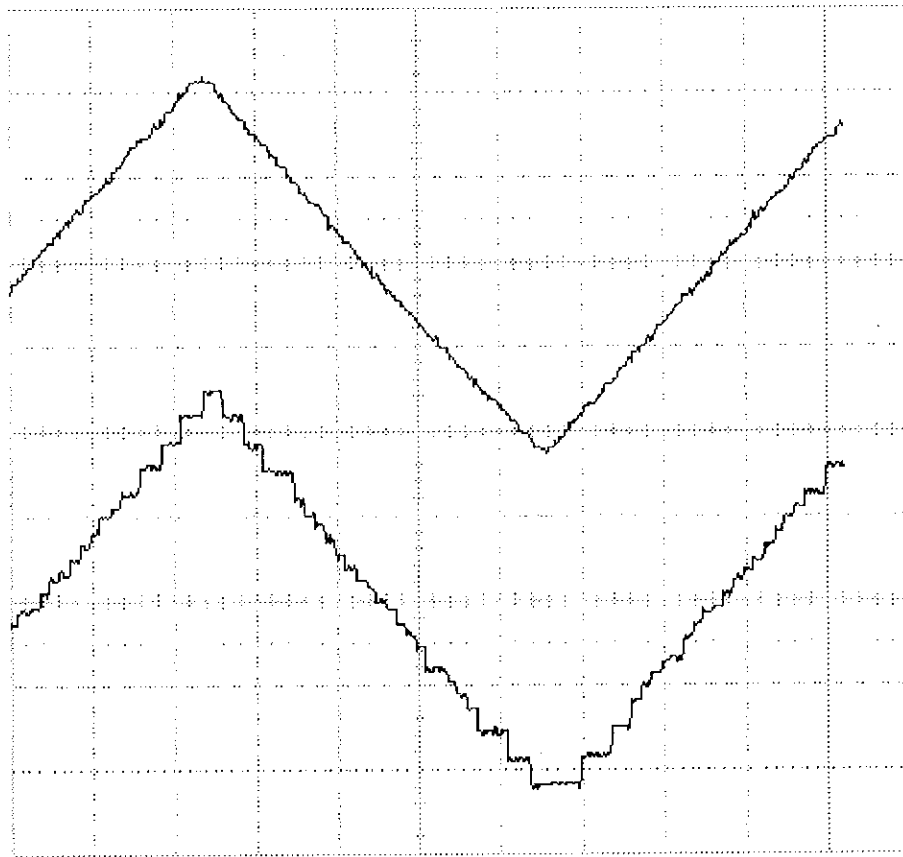


Figura 6.15. Señal triangular de frecuencia 300Hz, 4.4Vpp, muestreada con $f_m=20\text{kHz}$, utilizando el método PCM con compansión (a 5 bits), con filtro de salida desactivado.

VII. CONCLUSIONES Y RECOMENDACIONES

A. Conclusiones

- Debido a que el filtro de salida no es ideal, para lograr la reconstrucción perfecta de una señal se debe utilizar una frecuencia de muestreo mayor que la de Nyquist cuando se utiliza el sistema PCM.
- Trabajando con voz, el método PCM con frecuencia de muestreo de 8 kHz utilizando 8 bits es el que más claro se escucha; sin embargo, a 7 y 6 bits la calidad es aún bastante buena, y con 4 bits la claridad puede ser aún suficiente para algunas aplicaciones.
- Trabajando con el método PCM a 8 bits, con el filtro de salida activado y sintonizado a 4 kHz, no se notó mejoría en la calidad de la voz para frecuencias de muestreo mayores a 8 kHz; a frecuencias de muestreo menores sí se escucha alterada la voz. Con el filtro de salida desactivado, sí se percibió mejoría en la calidad de la voz para frecuencias de muestreo mayores a 8 kHz.
- Trabajando con voz, el método ADPCM con frecuencia de muestreo de 16 kHz y filtro de salida sintonizado a 4 kHz presentó bastante claridad. De los métodos estudiados aquí, éste es el indicado a utilizar cuando se cuenta con poco ancho de banda para transmitir la señal.
- El método DPCM mostró claridad en la voz para frecuencias de muestreo de 12 kHz o mayores y 3 ó más bits de cuantificación.
- Utilizando el método PCM con expansión a 5 bits, con frecuencia de muestreo de 8 kHz, se logró una mejoría en la calidad de la voz, comparándolo con el PCM normal a

5 bits. Con la compansión, se logró una calidad comparable a la del PCM normal a 6 bits.

- Como trabajo de graduación, y a un costo razonable, se logró construir un aparato muy didáctico para el estudio de la digitalización de señales, el cual puede servir a aquellos estudiantes de telecomunicaciones que lo deseen emplear. El aparato no demostró tener algún defecto en su operación.

B. Recomendaciones

- El digitalizador de señales desarrollado para el presente trabajo de graduación, puede ser utilizado para fines didácticos en el laboratorio de aquellas clases relacionadas con las comunicaciones. Con él resulta muy simple realizar observaciones prácticas de los resultados obtenidos en la digitalización de una señal, y se puede decidir cuál método y qué parámetros son los adecuados, dependiendo de la aplicación que se tenga.
- Al digitalizador de señales resultaría muy simple efectuarle modificaciones de programación, a manera de aprovechar el soporte físico que se diseñó para otras aplicaciones. Bastaría con reemplazar las rutinas de los métodos de transmisión, con lo cual se aprovecharían incluso las rutinas de soporte que ya se tienen.

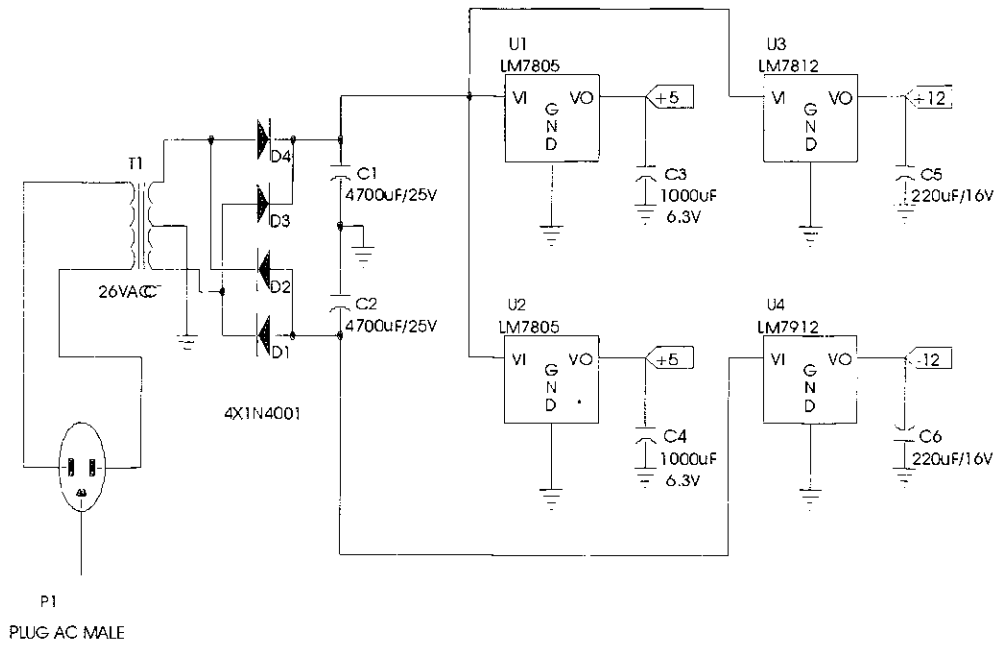
VIII. BIBLIOGRAFIA

- Kustra, R. O. y O. O. Tujsnaider. Principios de transmisión de señales digitales.
1984 Argentina, Empresa Nacional de Telecomunicaciones. 477pp.
- Taub, H. y Schilling, D. L. Principles of communication systems. 2da ed. McGraw-Hill
1986 Book Company. 749pp.
- Kaufman, M. y A. Seidman. Manual para ingenieros y técnicos en electrónica. México,
1988 McGraw-Hill. 692pp.

APENDICE A

Diagrama esquemático de la fuente de alimentación

(ver siguiente página)



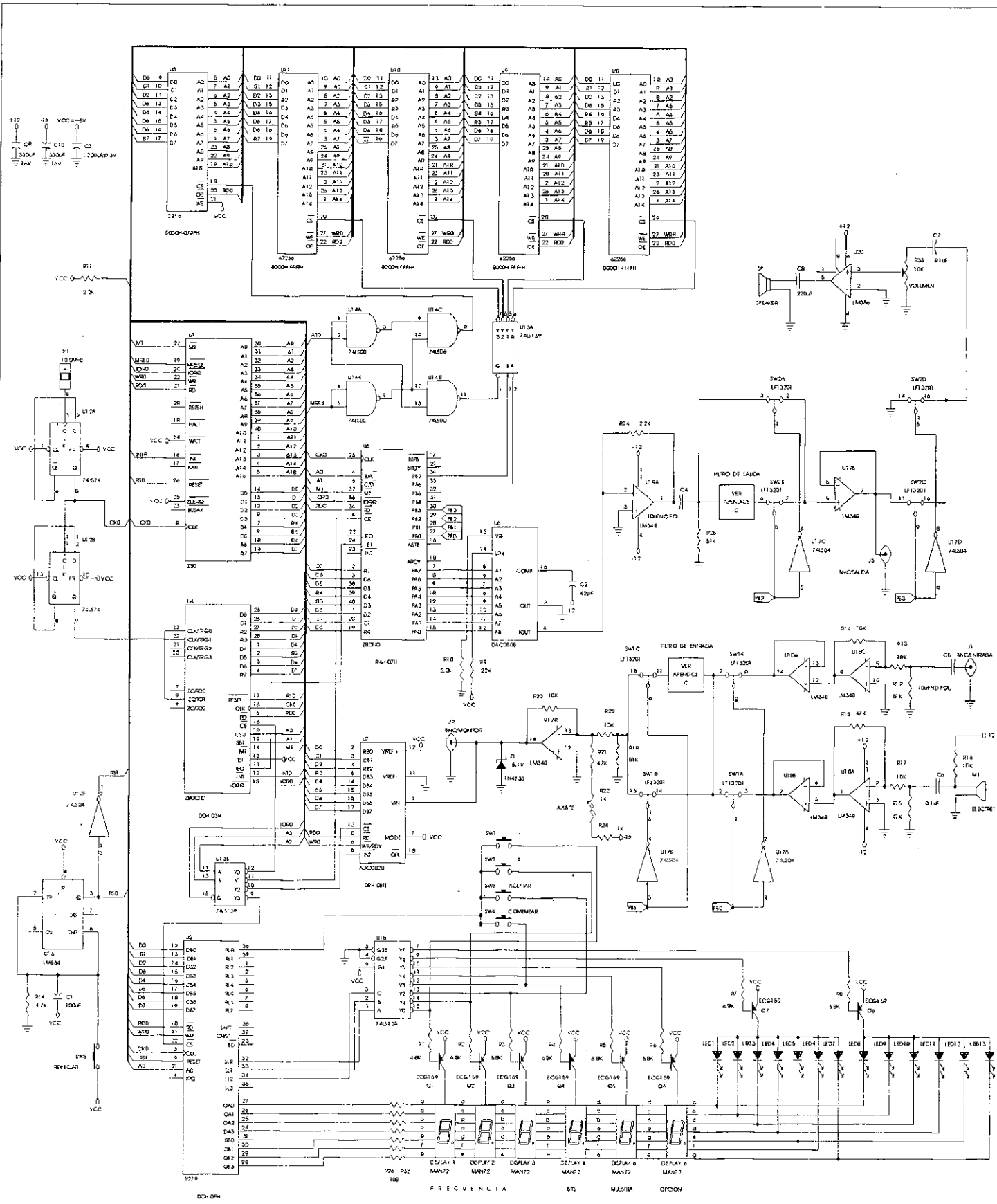
UNIVERSIDAD DEL VALLE
 GONZALO PALAREA
 CARNET 88103
 TRABAJO DE GRADUACION

Size		REV
A	FUENTE DE PODER	
Date:	August 20, 1995	Sheet 3 of 3

APENDICE B

Diagrama esquemático del circuito

(ver siguiente página)



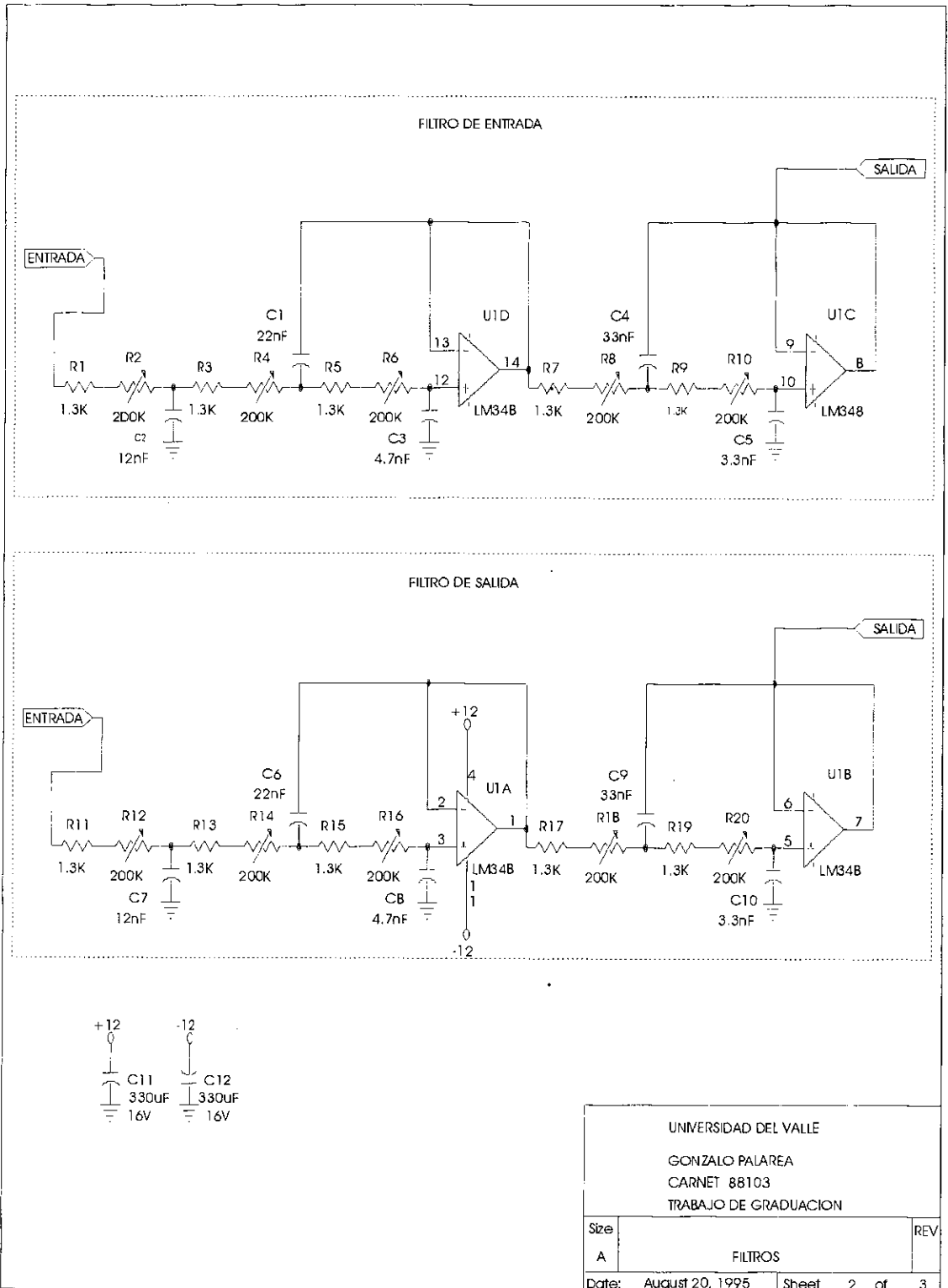
- LED1: DIRECTO
- LED2: GRABAR
- LED3: REPRODUCIR
- LED4: PGM
- LED5: BFM
- LED6: ADPCM
- LED7: COMPANION
- LED8: MICROFONO
- LED9: LINEA
- LED10: AUDIO ENTRADA
- LED11: FILTRO SALIDA
- LED12: TIRAMANDO

UNIVERSIDAD DEL VALLE	REV
CONSEJO PALARA	
CAMPUS BIBES	
TALLER DE GRADUACION	
Doc: _____	
C: _____	
Fecha: August 20, 1995	Sheet: 1 of 3

APENDICE C

Diagrama esquemático de los filtros

(ver siguiente página)



APENDICE D

Listado del programa en BASIC para calcular divisores

```
CLS
INPUT "NOMBRE DEL ARCHIVO DE SALIDA: ", N$
OPEN (N$) FOR OUTPUT AS #1
PRINT #1, " ALTO BAJO DIVISOR FREC. FREC."
PRINT #1, "          REAL DESEADA"
FOR FREC = 200 TO 25400 STEP 200
  LOCATE 8, 8
  PRINT "CALCULANDO DIVISORES PARA OBTENER "; FREC; " Hz."
  DIF = 1000
  ALTOB = 0: BAJOB = 0
  FOR ALTO = 1 TO 50
    FOR BAJO = 1 TO 255
      FRER = 2500000 / ALTO / BAJO
      DIFE = ABS(FRER - FREC)
      IF DIFE < DIF THEN ALTOB = ALTO: BAJOB = BAJO: DIF = DIFE
    NEXT BAJO
  NEXT ALTO
  PRINT #1, USING "#####"; ALTOB; BAJOB; ALTOB * BAJOB; 2500000 / ALTOB / BAJOB; FREC
NEXT FREC
CLOSE #1
END
```


APENDICE E

Tabla de divisores de frecuencia obtenidos con el programa del apéndice D

ALTO	BAJO	DIVISOR	FREC. REAL	FREC. TEORICA	ALTO	BAJO	DIVISOR	FREC. REAL	FREC. TEORICA
50	250	12500	200	200	1	245	245	10204	10200
25	250	6250	400	400	1	240	240	10417	10400
17	245	4165	600	600	1	236	236	10593	10600
25	125	3125	800	800	1	231	231	10823	10800
10	250	2500	1000	1000	1	227	227	11013	11000
15	139	2085	1199	1200	1	223	223	11211	11200
19	94	1786	1400	1400	1	219	219	11416	11400
11	142	1562	1601	1600	1	216	216	11574	11600
10	139	1390	1799	1800	1	212	212	11792	11800
5	250	1250	2000	2000	1	208	208	12019	12000
8	142	1136	2201	2200	1	205	205	12195	12200
7	149	1043	2397	2400	1	202	202	12376	12400
13	74	962	2599	2600	1	198	198	12626	12600
19	47	893	2800	2800	1	195	195	12821	12800
7	119	833	3001	3000	1	192	192	13021	13000
11	71	781	3201	3200	1	189	189	13228	13200
3	245	735	3401	3400	1	187	187	13369	13400
5	139	695	3597	3600	1	184	184	13587	13600
7	94	658	3799	3800	1	181	181	13812	13800
5	125	625	4000	4000	1	179	179	13966	14000
5	119	595	4202	4200	1	176	176	14205	14200
4	142	568	4401	4400	1	174	174	14368	14400
3	181	543	4604	4600	1	171	171	14620	14600
4	130	520	4808	4800	1	169	169	14793	14800
2	250	500	5000	5000	1	167	167	14970	15000
13	37	481	5198	5200	1	164	164	15244	15200
2	231	462	5411	5400	1	162	162	15432	15400
2	223	446	5605	5600	1	160	160	15625	15600
2	216	432	5787	5800	1	158	158	15823	15800
3	139	417	5995	6000	1	156	156	16026	16000
13	31	403	6203	6200	1	154	154	16234	16200
17	23	391	6394	6400	1	152	152	16447	16400
2	189	378	6614	6600	1	151	151	16556	16600
2	184	368	6793	6800	1	149	149	16779	16800
3	119	357	7003	7000	1	147	147	17007	17000
2	174	348	7184	7200	1	145	145	17241	17200
2	169	338	7396	7400	1	144	144	17361	17400
7	47	329	7599	7600	1	142	142	17606	17600
3	107	321	7788	7800	1	140	140	17857	17800
2	156	312	8013	8000	1	139	139	17986	18000
5	61	305	8197	8200	1	137	137	18248	18200
2	149	298	8389	8400	1	136	136	18382	18400
3	97	291	8591	8600	1	134	134	18657	18600
2	142	284	8803	8800	1	133	133	18797	18800
2	139	278	8993	9000	1	132	132	18939	19000
2	136	272	9191	9200	1	130	130	19231	19200
2	133	266	9398	9400	1	129	129	19380	19400
2	130	260	9615	9600	1	128	128	19531	19600
1	255	255	9804	9800	1	126	126	19841	19800
1	250	250	10000	10000	1	125	125	20000	20000

APENDICE F

Tabla de valores asignados en la compresión digital

A: Nivel a 8 bits; B: Nivel asignado

A	B	A	B	A	B	A	B	A	B	A	B	A	B	A	B
0	2	32	34	64	66	96	98	128	130	160	166	192	206	224	238
1	2	33	34	65	66	97	98	129	130	161	166	193	206	225	238
2	2	34	34	66	66	98	98	130	130	162	166	194	206	226	238
3	2	35	34	67	66	99	98	131	130	163	166	195	206	227	238
4	2	36	34	68	66	100	102	132	134	164	166	196	206	228	238
5	2	37	34	69	66	101	102	133	134	165	166	197	206	229	238
6	2	38	34	70	66	102	102	134	134	166	166	198	206	230	238
7	2	39	34	71	66	103	102	135	134	167	166	199	206	231	238
8	2	40	34	72	74	104	106	136	138	168	174	200	206	232	238
9	2	41	34	73	74	105	106	137	138	169	174	201	206	233	238
10	2	42	34	74	74	106	106	138	138	170	174	202	206	234	238
11	2	43	34	75	74	107	106	139	138	171	174	203	206	235	238
12	2	44	34	76	74	108	110	140	142	172	174	204	206	236	238
13	2	45	34	77	74	109	110	141	142	173	174	205	206	237	238
14	2	46	34	78	74	110	110	142	142	174	174	206	206	238	238
15	2	47	34	79	74	111	110	143	142	175	174	207	206	239	238
16	18	48	50	80	82	112	114	144	146	176	182	208	222	240	254
17	18	49	50	81	82	113	114	145	146	177	182	209	222	241	254
18	18	50	50	82	82	114	114	146	146	178	182	210	222	242	254
19	18	51	50	83	82	115	114	147	146	179	182	211	222	243	254
20	18	52	50	84	82	116	118	148	150	180	182	212	222	244	254
21	18	53	50	85	82	117	118	149	150	181	182	213	222	245	254
22	18	54	50	86	82	118	118	150	150	182	182	214	222	246	254
23	18	55	50	87	82	119	118	151	150	183	182	215	222	247	254
24	18	56	50	88	90	120	122	152	154	184	190	216	222	248	254
25	18	57	50	89	90	121	122	153	154	185	190	217	222	249	254
26	18	58	50	90	90	122	122	154	154	186	190	218	222	250	254
27	18	59	50	91	90	123	122	155	154	187	190	219	222	251	254
28	18	60	50	92	90	124	126	156	158	188	190	220	222	252	254
29	18	61	50	93	90	125	126	157	158	189	190	221	222	253	254
30	18	62	50	94	90	126	126	158	158	190	190	222	222	254	254
31	18	63	50	95	90	127	126	159	158	191	190	223	222	255	254

APENDICE G

Listado del programa en lenguaje ensamblador, grabado en ROM

```

PAD EQU 004H           ; DATOS CANAL A PIO (DAC)
DAC EQU 004H           ; DATOS CANAL A PIO (DAC)
PAC EQU 005H           ; CONTROL CANAL A PIO
PBD EQU 006H           ; DATOS CANAL B PIO
PBC EQU 007H           ; CONTROL CANAL B PIO
CTC0 EQU 000H          ; CANAL 0 CTC
CTC1 EQU 001H          ; CANAL 1 CTC
CTC2 EQU 002H          ; CANAL 2 CTC
CTC3 EQU 003H          ; CANAL 3 CTC
KDD EQU 00CH           ; DATOS 8279
KDC EQU 00DH           ; CONTROL 8279
ADC EQU 008H           ; CONVERTIDOR AD
OPC EQU 0801BH         ; VARIABLE: OPCION
BITS EQU 0801CH        ; VARIABLE: BITS
MUES EQU 0801DH        ; VARIABLE: MUESTRA
FREC EQU 08011H        ; VARIABLE: FRECUENCIA
FREC1 EQU 08012H       ; VARIABLE: FREC. MUESTRA 1
FREC2 EQU 08013H       ; VARIABLE: FREC. MUESTRA 2
FREC3 EQU 08014H       ; VARIABLE: FREC. MUESTRA 3
FREC4 EQU 08015H       ; VARIABLE: FREC. MUESTRA 4
COND1 EQU 08016H       ; VARIABLE: CONDICIONES 1 (SOFTWARE)
COND2 EQU 08017H       ; VARIABLE: CONDICIONES 2 (HARDWARE)
RUTINT EQU 08018H      ; VARIABLE: DIR. RUTINA INT. CTC
FRECD EQU 08020H       ; FRECUENCIAS
FRECD1 EQU 08021H      ; DEFINITIVAS
FRECD2 EQU 08022H
FRECD3 EQU 08023H
FRECD4 EQU 08024H
PRINCRAM EQU 008030H   ; COMIENZO DE RAM A USAR PARA MUESTRAS (32720 BYTES)

CERO:                 ORG 0000H
DI                     ; DESHABILITA INTERRUPTS
LD SP,08010H           ; INICIALIZA STACK POINTER
LD A,H
LD I,A                 ; FIJA VECTOR DE INT.
IM2                    ; MODO DE INTERRUPCION 2

IPIO:                 LD A,00001111B           ; INICIALIZACION DE PIO
                     OUT (PAC),A             ; MODO 0 (SALIDA)
                     OUT (PBC),A             ; EN EL PUERTO A
                     LD A,00000011B         ; EN EL PUERTO B
                     OUT (PAC),A             ; DESHABILITAR INT.
                     OUT (PBC),A             ; EN EL A
                     LD A,00000000B         ; Y EN EL B
                     OUT (PAD),A             ; PONER A 0
                     OUT (PBD),A             ; LA SALIDA DE LOS DOS

INVAL:                 LD A,80                 ; VALORES INICIALES
                     LD (FREC),A             ;
                     LD (FREC1),A           ; 8 KHZ PARA FRECUENCIAS
                     LD (FREC2),A
                     LD (FREC3),A
                     LD (FREC4),A
                     LD (FRECD),A

```

```

LD (FRECD1),A
LD (FRECD2),A
LD (FRECD3),A
LD (FRECD4),A
LD A,8
LD (BITS),A           ; 8 BITS
LD A,1
LD (MUES),A           ; MUESTRA 1
LD (OPC),A            ; OPCION 1
LD A,01101111B
LD (COND1),A          ; DIRECTO, PCM
LD A,01111111B
LD (COND2),A          ; MICROFONO
                        ; INICIALIZACION DE 8279
IKD: LD A,00000000B    ; 8 CHAR, ENTRADA IZQ. ENCLAD
      OUT (KDC),A
      LD A,00110100B   ; DIVISOR 20
      OUT (KDC),A
      LD A,11011111B   ; BORRAR DISPLAY RAM
      OUT (KDC),A
      LD B,255
RT:   DJNZ RT          ; RETRASO
      LD A,10100000B
      OUT (KDC),A
      CALL ACTDISP     ; INICIALIZA DISPLAYS
                        ; INICIALIZA VECT. INT. EN CTC
ICTC: LD HL,RUTINT
      LD A,L
      OUT (CTC0),A
                        ;
PRINC: CALL OPI
      JR PRINC
                        ;
OPI:  LD A,1
      LD (OPC),A
      CALL ACTDISP
      CALL LEETEC
                        ; LEE TECLADO
      CP 4
                        ; VER SI SE QUIERE COMENZAR
      JP Z,FINOPC1
                        ; SALTA SI QUIERE
      CP 3
                        ; VER SI ACEPTO
      JR NZ,OP1A
                        ; SALTAR SI NO ACEPTO
      LD A,(COND1)
                        ; VER QUE OPCION SE ELIGIO
      OR 00011111B
      CP 01111111B
                        ; E IRSE A ELLA
      JP Z,DIROPC
                        ; DIRECTO
      CP 10111111B
      JP Z,GRABOPC
                        ; GRABAR
      JP REPROPC
                        ; O REPRODUCIR
OP1A: LD A,(COND1)
      OR 00011111B
                        ; MODIFICA OPCION
      RRC A
                        ; A LA SIGUIENTE
                        ; POSIBLE
      CP 11101111B
                        ; SI LLEGO AL TOPE, REGRESA
      JR NZ,OP1B
      LD A,01111111B
OP1B: AND I1100000B
      LD B,A
                        ; GRABA NUEVAS CONDICIONES
      LD A,(COND1)
      AND 00011111B
      OR B
      LD (COND1),A
      JR OP1

```

```

DIROPC:          CALL OP2
                 CALL OP3
                 CALL OP4
                 CALL OP5
                 CALL OP6
                 CALL OP7
                 LD A,(COND1)
                 BIT 2,A
                 JP Z,OP1
                 BIT 1,A
                 JP Z,OP1
                 CALL OP8
                 JP OP1
                 ; OPCIONES SI SE ELIGIO DIRECTO
                 ; REVISAS SI ES ADPCM
                 ; SI ES, PASA A OPCION 1
                 ; REVISAS SI ES COMPANS
                 ; SI ES, PASA A OPCION 1
                 ; SI NO, PIDE BITS

GRABOPC:        CALL OP9
                 CALL OP3
                 CALL OP4
                 CALL OP7
                 JP OP1
                 ; OPCIONES SI SE ELIGIO GRABAR

REPROPC:        CALL OP9
                 CALL OP2
                 CALL OP5
                 CALL OP6
                 CALL OP7
                 LD A,(COND1)
                 BIT 2,A
                 JP Z,OP1
                 BIT 1,A
                 JP Z,OP1
                 CALL OP8
                 JP OP1
                 ; OPCIONES SI SE ELIGIO REPRODUCIR
                 ; REVISAS SI ES ADPCM
                 ; SI ES, PASA A OPCION 1
                 ; REVISAS SI ES COMPANS
                 ; SI ES, PASA A OPCION 1
                 ; SI NO, PIDE BITS

OP2:            LD A,2
                 LD (OPC),A
                 CALL ACTDISP
                 CALL LEETEC
                 CP 4
                 JP Z,FINOPC
                 CP 3
                 RET Z
                 ; LEE TECLADO
                 ; VER SI SE QUIERE COMENZAR
                 ;
                 ; VER SI ACEPTO
                 ; REGRESAR SI ACEPTO

OP2A:           LD A,(COND1)
                 OR 11100000B
                 RRCA
                 CP 11111110B
                 JR NZ,OP2B
                 LD A,11101111B
                 AND 00011111B
                 LD B,A
                 LD A,(COND1)
                 AND 11100000B
                 OR B
                 LD (COND1),A
                 JR OP2
                 ; MODIFICA OPCION
                 ; A LA SIGUIENTE
                 ; POSIBLE
                 ; GRABA NUEVAS CONDICIONES

OP2B:           LD A,(COND1)
                 AND 11100000B
                 OR B
                 LD (COND1),A
                 JR OP2
                 ; GRABA NUEVAS CONDICIONES

OP3:            LD A,3
                 LD (OPC),A
                 CALL ACTDISP
                 CALL LEETEC
                 CP 4
                 JP Z,FINOPC
                 CP 3
                 RET Z
                 ; LEE TECLADO
                 ; VER SI SE QUIERE COMENZAR
                 ;
                 ; VER SI ACEPTO
                 ; REGRESAR SI ACEPTO

OP3A:           LD A,(COND2)
                 ; MODIFICA OPCION

```



```

OR 00111111B          ; A LA SIGUIENTE
RRCA                   ; POSIBLE
CP 11011111B
JR NZ,OP3B
LD A,01111111B
OP3B: AND 11000000B
LD B,A                ; GRABA NUEVAS CONDICIONES
LD A,(COND2)
AND 00111111B
OR B
LD (COND2),A
JR OP3

OP4: LD A,4
LD E,11011111B       ; MARCA CON 0 EL BIT A MODIFICAR
JP OP456

OP5: LD A,5
LD E,11101111B
JP OP456

OP6: LD A,6
LD E,11110111B
JP OP456

OP456: LD (OPC),A
OP456D: CALL ACTDISP
CALL LEETEC           ; LEE TECLADO
CP 4                  ; VER SI SE QUIERE COMENZAR
JP Z,FINOPC          ;
CP 3                  ; VER SI ACEPTO
RET Z                 ; REGRESAR SI ACEPTO
OP456A: LD A,(COND2) ; MODIFICA OPCION
OR E                  ;
CP E                  ;
JR Z,OP456B          ; SI YA ESTABA EN 0 EL BIT, SALTA
LD A,E                ; SI NO, LO PONE EN 0
JR OP456C            ;
OP456B: LD A,11111111B ;
OP456C: XOR E         ;
LD B,A                ; GRABA NUEVAS CONDICIONES
LD A,(COND2)
AND E
OR B
LD (COND2),A
JR OP456D

OP7: LD A,(MUES)
CP 10
JR Z,OP7B
LD HL,FREC
ADD L
LD L,A                ; GRABAR LA FREC
LD A,(FREC)          ; UTILIZ. EN ESA
LD (HL),A            ; MUESTRA
OP7B: LD A,7
LD (OPC),A
CALL ACTDISP
CALL LEETEC          ; LEE TECLADO
CP 4                  ; VER SI SE QUIERE COMENZAR
JP Z,FINOPC          ;
CP 3                  ; VER SI ACEPTO
RET Z                 ; REGRESAR SI ACEPTO
CP 2                  ; VER SI QUIERE AUMENTAR
JR NZ,OP7A           ; SALTAR SI NO (DISMINUIR)
LD A,(FREC)

```

	ADD A,2	; AUMENTA
	LD (FREC),A	
	CP 202	; FRECUENCIA MAXIMA=200
	JR NZ,OP7	
	LD A,200	
	LD (FREC),A	
	JR OP7	
OP7A:	LD A,(FREC)	
	CP A	; BORRA CARRY FLAG
	SBC A,2	; DISMINUYE
	LD (FREC),A	
	CP 0	; NO PERMITE 0
	JR NZ,OP7	
	LD A,2	
	LD (FREC),A	
	JR OP7	
OP8:	LD A,8	; PASA EN A LA OPCION
	LD E,9	; EN E EL MAXIMO PERMITIDO
	LD HL,BITS	; Y EN HL LA DIRECCION DE VARIABLE
	JP OP89	; A MODIFICAR
OP9:	LD A,9	
	LD E,5	
	LD HL,MUES	
	JP OP89	
OP89:	LD (OPC),A	
OP89B:	PUSH HL	
	LD A,(MUES)	
	CP 10	
	JR NZ,OP89C	
	LD A,0	
OP89C:	LD HL,FREC	
	ADD L	
	LD L,A	
	LD A,(HL)	
	LD (FREC),A	
	POP HL	
	CALL ACTDISP	
	CALL LEETEC	; LEE TECLADO
	CP 4	; VER SI SE QUIERE COMENZAR
	JP Z,FINOPC	
	CP 3	; VER SI ACEPTO
	RET Z	; REGRESAR SI ACEPTO
	CP 2	; VER SI QUIERE AUMENTAR
	JR NZ,OP89A	; SALTAR SI NO (DISMINUIR)
	LD A,(HL)	
	INC A	; AUMENTA
	LD (HL),A	
	CP E	; CHEQ. SI SE PASO
	JR NZ,OP89B	
	LD A,E	
	DEC A	
	LD (HL),A	
	JR OP89B	
OP89A:	LD A,(HL)	
	CP A	; BORRA CARRY FLAG
	DEC A	; DISMINUYE
	LD (HL),A	
	CP 0	; NO PERMITE 0
	JR NZ,OP89B	
	LD A,1	
	LD (HL),A	
	JR OP89B	

FINOPC:	POP HL	; REvisa LAS CONDICIONES ELEGIDAS
FINOPC1:	LD A,(COND1)	; AJUSTA STACK PARA VOLVER A PRINC CON RET
	LD C,A	
	BIT 7,C	; VER SI ES DIRECTO
AJDIR:	JR NZ,AJGRAB	
	BIT 4,C	; PCM DIRECTO?
	JR NZ,AJDIRB	
	LD HL,RDIRPCM	
	JP AJDIRC	
AJDIRB:	BIT 3,C	; DPCM DIRECTO?
	JR NZ,AJDIRD	
	LD HL,RDIRDPCM	
AJDIRC:	LD A,(BITS)	
	LD B,A	
	SLA B	
	DEC HL	
	DEC HL	
AJDIRA:	INC HL	; BUSCA DIRECCION DE RUTINA
	DJNZ AJDIRA	; DE ACUERDO A BITS
	LD DE,RUTINT	
	LDI	
	LDI	
	LD BC,(RUTINT)	; Y LA GUARDA EN SU LUGAR
	JP FINCHEQ	
AJDIRD:	BIT 2,C	; ADPCM DIRECTO?
	JR NZ,AJDIRE	;
	LD BC,DIRADPCM	
	JP FINCHEQ	
AJDIRE:	LD BC,DIRCOMN	; COMPANSION
	JP FINCHEQ	
AJGRAB:	BIT 6,C	; GRABAR?
	JR NZ,AJREP	
	LD BC,GRABAR	
	JP FINCHEQ	
AJREP:	BIT 4,C	; PCM REPRODUCIR?
	JR NZ,AJREPB	
	LD HL,RREPPCM	
	JP AJREPC	
AJREPB:	BIT 3,C	; DPCM REPRODUCIR?
	JR NZ,AJREPD	
	LD HL,RREPDPCM	
AJREPC:	LD A,(BITS)	
	LD B,A	
	SLA B	
	DEC HL	
	DEC HL	
AJREPA:	INC HL	; BUSCA DIRECCION DE RUTINA
	DJNZ AJREPA	; DE ACUERDO A BITS
	LD DE,RUTINT	
	LDI	
	LDI	
	LD BC,(RUTINT)	; Y LA GUARDA EN SU LUGAR
	JP FINCHEQ	
AJREPD:	BIT 2,C	; ADPCM REPRODUCIR?
	JR NZ,AJREPE	;
	LD BC,REPADPCM	
	JP FINCHEQ	
AJREPE:	LD BC,REPCOMN	; COMPANSION REPRODUCIR?
	JP FINCHEQ	
FINCIIEQ:	LD A,11000000B	; MEMORIA 4
	OUT (PBD),A	
	LD (RUTINT),BC	; FIJA DIR. DE RUTINA INT. A USAR

```

LD A,10000000B      ; MEMORIA 3
OUT (PBD),A
LD (RUTINT),BC      ; FIJA DIR. DE RUTINA INT. A USAR
LD A,01000000B      ; MEMORIA 2
OUT (PBD),A
LD (RUTINT),BC      ; FIJA DIR. DE RUTINA INT. A USAR
LD A,00000000B      ; MEMORIA 1
OUT (PBD),A
LD (RUTINT),BC      ; FIJA DIR. DE RUTINA INT. A USAR
LD HL,COND2
RES 1,(HL)          ; TRABAJANDO
CALL ACTDISP
CALL ICTCI          ; INICIALIZAR CTC
IN A,(PBD)          ; LEE CONDICIONES
AND 00111111B      ; Y ELIGE
LD B,A              ; RAM A UTILIZAR
LD A,(MUES)         ; SEGUN MUESTRA
DEC A
RRC A
RRC A
OR B
OUT (PBD),A
LD HL,PRINCRAM
LD D,128            ; VALOR ACUMULADO INICIAL
LD C,1              ; PARA REVISAR SI FINALIZO
LD B,1              ; SUMANDO INICIAL PARA ADPCM
LD E,0              ; DIR. ANTERIOR ADPCM: BAJO
LOOP:
EI
BIT 0,C             ; REvisa si finalizo
JR Z,FINLOOP
HALT
JR LOOP
FINLOOP:
LD A,0              ; REGRESA A UTILIZAR RAMO
OUT (PBD),A
LD A,(COND1)
BIT 6,A             ; CHEQ. SI OPCION FUE GRABAR
JR NZ,FINOPCA
LD A,(MUES)
LD HL,FRECD        ; GUARDA FRECUENCIA USADA
ADD L               ; AL GRABAR
LD L,A
LD A,(FREC)
LD (HL),A
JR FINOPCA
FINOPCA:
LD DE,FREC1        ; RESTAURA FREC.
LD HL,FRECD1       ; UTILIZADAS AL
LD BC,4            ; GRABAR
LDIR
LD HL,COND2
SET 1,(HL)         ; NO TRABAJANDO
RET
DIRPCM1:
OUT (ADC),A
IN A,(ADC)
CALL BTS1
JP FINDIR
DIRPCM2:
OUT (ADC),A
IN A,(ADC)
CALL BTS2
JP FINDIR
DIRPCM3:
OUT (ADC),A
IN A,(ADC)
CALL BTS3
JP FINDIR

```

; PCM DIRECTO (NO GRABA)

```

DIRPCM4:      OUT (ADC),A
              IN A,(ADC)
              CALL BTS4
              JP FINDIR

DIRPCM5:      OUT (ADC),A
              IN A,(ADC)
              CALL BTS5
              JP FINDIR

DIRPCM6:      OUT (ADC),A
              IN A,(ADC)
              CALL BTS6
              JP FINDIR

DIRPCM7:      OUT (ADC),A
              IN A,(ADC)
              CALL BTS7
              JP FINDIR

DIRPCM8:      OUT (ADC),A
              IN A,(ADC)
              JP FINDIR

DIRDPCM1:     OUT (ADC),A
              IN A,(ADC)                ; LEE ADC
              LD E,1                    ; FIJA DIFERENCIA MAXIMA PERMIT.
              CALL BTSDPCM              ; AJUSTA ACUMULADO
              JP FINDIR                 ; SACA

DIRDPCM2:     OUT (ADC),A
              IN A,(ADC)
              LD E,2
              CALL BTSDPCM
              JP FINDIR

DIRDPCM3:     OUT (ADC),A
              IN A,(ADC)
              LD E,4
              CALL BTSDPCM
              JP FINDIR

DIRDPCM4:     OUT (ADC),A
              IN A,(ADC)
              LD E,8
              CALL BTSDPCM
              JP FINDIR

DIRDPCM5:     OUT (ADC),A
              IN A,(ADC)
              LD E,16
              CALL BTSDPCM
              JP FINDIR

DIRDPCM6:     OUT (ADC),A
              IN A,(ADC)
              LD E,32
              CALL BTSDPCM
              JP FINDIR

DIRDPCM7:     OUT (ADC),A
              IN A,(ADC)
              LD E,64
              CALL BTSDPCM
              JP FINDIR

```

DIRDPCM8: OUT (ADC),A
IN A,(ADC)
LD E,128
CALL BTSDPCM
JP FINDIR

DIRADPCM: OUT (ADC),A
IN A,(ADC)
CALL ADPCM
JP FINDIR

DIRCOMN: OUT (ADC),A
IN A,(ADC)
CALL COMPAN
JP FINDIR

FINDIR: OUT (DAC),A
IN A,(KDC)
BIT 0,A
CALL NZ,ICTCN
RETI

; REPRODUCIR PCM

REPPCM1: LD A,(HL)
CALL BTS1
JP FINREP

REPPCM2: LD A,(HL)
CALL BTS2
JP FINREP

REPPCM3: LD A,(HL)
CALL BTS3
JP FINREP

REPPCM4: LD A,(HL)
CALL BTS4
JP FINREP

REPPCM5: LD A,(HL)
CALL BTS5
JP FINREP

REPPCM6: LD A,(HL)
CALL BTS6
JP FINREP

REPPCM7: LD A,(HL)
CALL BTS7
JP FINREP

REPPCM8: LD A,(HL)
JP FINREP

REPDPCM1: LD A,(HL)
LD E,1
CALL BTSDPCM
JP FINREP

REPDPCM2: LD A,(HL)
LD E,2
CALL BTSDPCM
JP FINREP

REPDPCM3: LD A,(HL)
LD E,4

```

CALL BTSDPCM
JP FINREP

REPDCM4:  LD A,(HL)
          LD E,8
          CALL BTSDPCM
          JP FINREP

REPDCM5:  LD A,(HL)
          LD E,16
          CALL BTSDPCM
          JP FINREP

REPDCM6:  LD A,(HL)
          LD E,32
          CALL BTSDPCM
          JP FINREP

REPDCM7:  LD A,(HL)
          LD E,64
          CALL BTSDPCM
          JP FINREP

REPDCM8:  LD A,(HL)
          LD E,128
          CALL BTSDPCM
          JP FINREP

REPADPCM: LD A,(HL)
          CALL ADPCM
          JP FINREP

REPCOMN:  LD A,(HL)
          PUSH HL
          CALL COMPAN
          POP HL
          JP FINREP

FINREP:   OUT (DAC),A
          INC HL           ; INCREMENTA POINTER
          BIT 7,H          ; REVIS A SI LLEGO A FIN
          CALL Z,ICTCN    ; SI LLEGO, ELIMINA INTERR
          RETI            ; REGRESA

BTS1:     SRA A           ; COPIA EL BIT 7
          SRA A           ; A TODOS LOS DEMAS
          SRA A
          SRA A
          SRA A
          SRA A
          SRA A
          SRA A
          RET

BTS2:     AND 11000000B   ; COPIA LOS BITS 7 Y 6
          LD E,A          ; A LOS BITS
          RRCA            ; 5 Y 4
          RRCA
          OR E
          RRCA            ; 3 Y 2
          RRCA
          OR E
          RRCA            ; 1 Y 0
          RRCA
          OR E
          RET

```

BTS3:	AND 1110000B LD E,A RRCA RRCA RRCA OR E RLCA RLCA AND 0000011B OR E RET	; COPIA LOS BITS 7, 6 Y 5 ; A LOS BITS ; 4, 3 Y 2 ; Y LOS BITS 7 Y 6 ; A LOS BITS ; 1 Y 0
BTS4:	AND 1111000B LD E,A RRCA RRCA RRCA RRCA OR E RET	; COPIA LOS BITS 7,6,5,4 ; A LOS BITS ; 3,2,1,0
BTS5:	AND 11111000B LD E,A RLCA RLCA RLCA AND 00000111B OR E RET	; COPIA LOS BITS 7, 6 Y 5 ; A LOS BITS ; 2, 1 Y 0
BTS6:	AND 11111100B LD E,A RLCA RLCA AND 00000011B OR E RET	; COPIA LOS BITS 7 Y 6 ; A LOS BITS ; 1 Y 0
BTS7:	AND 11111110B LD E,A RLCA AND 00000001B OR E RET	; COPIA EL BIT 7 ; AL BIT 0
BTSDPCM:	SUB D JR C,BTSDPA JR NZ,BTSDPD INC A	; RESTA ACUMULADO ; SI ES MENOR, SALTA ; SI NO ERA IGUAL, SALTA ; SI LA DIFERENCIA ERA 0, PONER 1
BTSDPD:	CPE JR C,BTSDPB LD A,E	; COMPARAR CON DIF. MAXIMA PERMIT. ; SI MENOR, SALTAR ; SI NO, DIF. MAXIMA
BTSDPB:	ADD D LD D,A RET NC LD D,255 LD A,D RET	; SUMAR DIFERENCIA ; Y GUARDAR ; REGRESAR SI NO PASO DE 255 ; SI PASO, PONER 255
BTSDPA:	NEG CPE JR C,BTSDPC LD A,E	; SACA COMPL. 2 (PASA DE NEG A POS) ; COMP. CON DIF. MAXIMA PERMIT. ; SI MENOR, SALTAR ; SI NO, DIF. MAXIMA
BTSDPC:	LD B,A LD A,D SUB B LD D,A RET NC	; EN B DIFERENCIA ; EN A ACUMULADO ; RESTA DIFERENCIA ; GUARDA ACUMULADO ; REGRESA SI NO BAJO DE 0


```

LD D,0          ; SI PASO, PONE 0
LD A,D
RET            ; Y REGRESA

NOP

ADPCM:         CP D          ; COMPARA CON ACUMULADO
              JR C,ADPCMB    ; SI DISMINUYO, SALTAR
              BIT 0,E        ; REVISAR SI ANTERIOR FUE MENOR
              JR Z,ADPCMA    ; SI FUE MENOR, SALTA
              INC B          ; MAYOR Y ANTERIOR MAYOR
              LD A,D         ; AUMENTA SUMANDO Y
              ADD B          ; LO SUMA
              LD D,A         ; GUARDA EN ACUMULADO
              RET NC        ; REGRESA SI NO PASO DE 255
              LD D,255      ; SI PASO, CARGA 255
              LD A,D         ; Y EN A
              RET           ; REGRESA

ADPCMA:        LD E,1        ; MAYOR Y ANTERIOR MENOR, GUARDA DIR.
              LD B,1        ; AJUSTA SUMANDO
              INC D          ; SUBE 1 A ACUMULADO
              LD A,D         ; LO PASA A A
              RET           ; Y REGRESA

ADPCMB:        BIT 0,E        ; REvisa SI ANTERIOR FUE MENOR
              JR Z,ADPCMC    ; SI FUE MENOR, SALTA
              LD B,1        ; MENOR Y ANTERIOR MAYOR
              LD E,0         ; AJUSTA SUMANDO Y GUARDA DIR.
              DEC D          ; BAJA 1 EN ACUMULADO
              LD A,D         ; LO GUARDA EN A
              RET           ; Y REGRESA

ADPCMC:        INC B          ; MENOR Y ANTERIOR MENOR
              LD A,D         ; AUMENTA SUMANDO
              SUB B          ;
              LD D,A         ; LO RESTA
              RET NC        ; REGRESA SI NO BAJO DE 0
              LD D,0         ; SI BAJO, CARGA 0
              LD A,D         ;
              RET           ; REGRESA

COMPAN:        SRL A          ; DIVIDE
              SRL A          ; ENTRE 4
              LD HL,COMPVAL ; BUSCA
              LD D,0         ; EL
              LD E,A         ; VALOR
              ADD HL,DE      ; CORRESPONDIENTE
              LD A,(HL)      ;
              RET           ; REGRESA

GRABAR:        OUT (ADC),A    ; GRABAR A RAM
              IN A,(ADC)     ; PIDE CONV.
              LD (HL),A      ; LEE CONV.
              INC HL         ; GUARDA EN (HL)
              BIT 7,H        ; INCREMENTA POINTER
              CALL Z,ICTCN   ; REvisa SI LLEGO A FIN
              RETI          ; SI LLEGO, ELIMINA INTERR
              ; REGRESA

; INIC. CTC CON INTERRUPT
; CON CONSTANTES SEGUN FREC

ICTCI:         LD H,0
              LD A,(FREC)
              LD L,A
              LD BC,DIVISOR
              ADD HL,BC
              DEC HL

```

```

LD A,11000101B           ; CANAL 0 COMO CONTADOR CON INT
OUT (CTC0),A             ; SIGUE DATO
LD A,(HL)                ; CONSTANTE 1
OUT (CTC0),A
LD A,01000101B           ; CANAL 1 COMO CONTADOR SIN INT
OUT (CTC1),A             ; SIGUE DATO
DEC HL
LD A,(HL)                ; CONSTANTE 2
OUT (CTC1),A
RET

ICTCN: LD A,01000011B           ; INIC. CTC SIN INTERRUPT
OUT (CTC0),A             ; CANAL 0 COMO CONTADOR SIN INT
LD C,0                   ; RESET DEL CANAL Y NO SIGUE DATO
RET                       ; AVISAR QUE HUBO FIN
;

; ESCRIBE EN POSICION C
; EL DIGITO D
ESCNUME: LD A,C
OR 10010000B
OUT (KDC), A             ; APUNTA A DONDE VA EL DATO
LD HL,NUMEROS-1
LD E,B                   ; GUARDA B EN E
LD B,D
INC B
;
; OBTIENE LA POSICION ADECUADA
; RESTAURA B
; LEE EL DATO DE ROM
; ESCRIBE EL DATO EN 8279
;
POSI: INC HL
DJNZ POSI
LD B,E
LD A,(HL)
OUT (KDD),A
RET

; HACE LOOP HASTA TECLAZO
; VER SI HAY TECLAZOS ANTERIORES
LEETEC: IN A,(KDC)
AND 00000111B
CP 0
JR Z,LEETECA
LD B,A                   ; SI HAY, LOS LEE
; SI HAY, LOS LEE
LTI: LD A,01000000B           ; LEER TECLADO
OUT (KDC),A             ; EN A
IN A,(KDD)              ; HASTA VACIAR COLA
DJNZ LTI
;
; VER SI HAY TECLAZO
; DEJAR SOLO LA CUENTA
; SI NO HAY NADA
; VUELVE A LEER
LEETECA: IN A,(KDC)
AND 00000111B           ; LEER TECLADO
CP 0
JR Z,LEETECA
LD A,01000000B           ; EN A
OUT (KDC),A             ; HASTA VACIAR COLA
IN A,(KDD)              ; LO ACOMODA
RRCA
RRCA
RRCA
;
; BORRA LO NO DESEADO
; INCREMENTA 1 Y REGRESA
; EN A EL NUMERO DE TECLA (1-4)
;
ACTDISP: EXX
LD C,0                   ; PREPARA PARA
LD D,0                   ; ESCRIBIR DIGITO 0
LD A,(FREC)
CP 99
JR C,FINCEN             ; CHEQUEA CUANTAS CENTENAS
; TIENE EL NUMERO

```

```

CP A ; BORRA CARRY FLAG
SBC A,100 ;
INC D ;
JR CENTENA ;
FINCEN: LD B,A ;
CALL ESCNUME ; Y LAS ESCRIBE
LD A,B ;
INC C ; PREPARA PARA ESCR. DIGITO 1
LD D,0 ;
DECENA: CP 9 ; CHEQUEA CUANTAS DECENAS
JR C,FINDEC ; TIENE EL NUMERO
SBC A,10 ;
INC D ;
JR DECENA ;
FINDEC: LD B,A ;
CALL ESCNUME ; Y LAS ESCRIBE
LD A,B ;
INC C ; PREPARA PARA ESCR. DIGITO 2
LD D,A ; CARGA LAS UNIDADES
CALL ESCNUME ; Y LAS ESCRIBE
INC C ; PREPARA PARA ESCR. DIGITO 3
LD A,(COND1)
BIT 6,A ; GRABAR?
JR NZ,ACTDISPA
LD A,8 ; SI ES GRABAR, 8 BITS
JR ACTDISPD
ACTDISPA: BIT 2,A ; ADPCM?
JR NZ,ACTDISPB
LD A,1 ; SI ES ADPCM, 1 BIT
JR ACTDISPD
ACTDISPB: BIT 1,A ; COMPANSION?
JR NZ,ACTDISPC
LD A,5 ; SI ES COMPANSION, 5 BITS
JR ACTDISPD
ACTDISPC: LD A,(BITS) ; CARGA BITS
ACTDISPD: LD D,A ; Y LO ESCRIBE
CALL ESCNUME
INC C ; PREPARA PARA ESCR. DIGITO 4
LD A,(COND1)
BIT 7,A ; REvisa SI ES DIRECTO
JR Z,ACTDISPE ; SI ES. NO DESPLIEGA MUESTRA
LD A,(MUES)
JR ACTDISPF
ACTDISPE: LD A,10 ; LO CARGA
ACTDISPF: LD D,A
CALL ESCNUME ; Y LO ESCRIBE
INC C ; PREPARA PARA ESCR. DIGITO 5
LD A,(COND2)
BIT 1,A ; REvisa SI ESTA TRABAJANDO
JR Z,ACTDISPG ; SI ESTA, NO DESPLIEGA OPCION
LD A,(OPC) ; CARGA OPCION
JR ACTDISPH
ACTDISPG: LD A,10 ; BLANCO
ACTDISPH: LD D,A
CALL ESCNUME ; Y LO ESCRIBE
LD A,10010110B ; ESCRIBIR 8279 CON AI DESDE 6
OUT (KDC),A
LD A,(COND1) ; ACTUALIZAR LEDS COND. SOFTWARE
BIT 6,A ; REvisa SI ES GRABAR
JR NZ,ACTDISPI ; SI NO ES, SALTA
OR 00011111B ; SI ES GRABAR, APAGAR TIPO DE PCM
LD B,00011000B ; APAGAR AUDIO
JR ACTDISPK ; Y FILTRO DE SALIDA
ACTDISPI: BIT 5,A ; REvisa SI ES REPRODUCIR
JR NZ,ACTDISPJ ; SI NO ES, SALTA
LD B,11100000B ; SI SI REPR. APAGAR MICROFONO,
JR ACTDISPK ; LINEA Y FILTRO ENTRADA

```

```

ACTDISPJ: LD B,00000000B ; SI NO, NO APAGAR NADA
ACTDISPK: OUT (KDD),A ; LEDS DE CONDICIONES DE SOFTWARE
LD A,(COND2) ; OTROS LEDS
OR B ; APAGA LEDS QUE NO HAY QUE PRENDER
OUT (KDD),A ; LEDS DE CONDICIONES DE HARDWARE
LD B,A ; GUARDA EN B CONDICIONES DE HARDWARE
IN A,(PBD) ; LEE PUERTO B
AND 11000000B ; BORRA CONDICIONES ANTERIORES
BIT 7,B ; MICROFONO?
JR Z,ACTDISPL
SET 0,A ; APAGA MICROFONO (PRENDE LINEA)
ACTDISPL: BIT 5,B ; FILTRO ENTRADA?
JR Z,ACTDISPM
SET 1,A ; APAGA FILTRO DE ENTRADA
ACTDISPM: BIT 4,B ; AUDIO?
JR Z,ACTDISPN
SET 2,A ; APAGA AUDIO
ACTDISPN: BIT 3,B ; FILTRO SALIDA?
JR Z,ACTDISPO
SET 3,A ; APAGA FILTRO SALIDA
ACTDISPO: OUT (PBD),A ; ACTUALIZA SWITCHES
ENX
RET

NUMEROS: DB 00000111B ; TABLA DE CODIGOS DE CARACTERES
DB 10011111B ; '0'
DB 00100101B ; '1'
DB 00001101B ; '2'
DB 10011001B ; '3'
DB 01001001B ; '4'
DB 01000001B ; '5'
DB 00011111B ; '6'
DB 00000001B ; '7'
DB 00001001B ; '8'
DB 11111111B ; '9'
; BLANCO

DIVISOR: DB 50, 250 ; TABLA DE DIVISORES DE FREC.
DB 25, 250
DB 17, 245
DB 25, 125
DB 10, 250
DB 15, 139
DB 19, 94
DB 11, 142
DB 10, 139
DB 5, 250
DB 8, 142
DB 7, 149
DB 13, 74
DB 19, 47
DB 7, 119
DB 11, 71
DB 3, 245
DB 5, 139
DB 7, 94
DB 5, 125
DB 5, 119
DB 4, 142
DB 3, 181
DB 4, 130
DB 2, 250
DB 13, 37
DB 2, 231
DB 2, 223
DB 2, 216

```

DB 3, 139
DB 13, 31
DB 17, 23
DB 2, 189
DB 2, 184
DB 3, 119
DB 2, 174
DB 2, 169
DB 7, 47
DB 3, 107
DB 2, 156
DB 5, 61
DB 2, 149
DB 3, 97
DB 2, 142
DB 2, 139
DB 2, 136
DB 2, 133
DB 2, 130
DB 1, 255
DB 1, 250
DB 1, 245
DB 1, 240
DB 1, 236
DB 1, 231
DB 1, 227
DB 1, 223
DB 1, 219
DB 1, 216
DB 1, 212
DB 1, 208
DB 1, 205
DB 1, 202
DB 1, 198
DB 1, 195
DB 1, 192
DB 1, 189
DB 1, 187
DB 1, 184
DB 1, 181
DB 1, 179
DB 1, 176
DB 1, 174
DB 1, 171
DB 1, 169
DB 1, 167
DB 1, 164
DB 1, 162
DB 1, 160
DB 1, 158
DB 1, 155
DB 1, 154
DB 1, 152
DB 1, 151
DB 1, 149
DB 1, 147
DB 1, 145
DB 1, 144
DB 1, 142
DB 1, 140
DB 1, 139
DB 1, 137
DB 1, 136
DB 1, 134
DB 1, 133
DB 1, 132
DB 1, 130

DB 1, 129
DB 1, 128
DB 1, 126
DB 1, 125

COMPVAL:

DB 2
DB 2
DB 2
DB 2
DB 18
DB 18
DB 18
DB 18
DB 34
DB 34
DB 34
DB 34
DB 50
DB 50
DB 50
DB 50
DB 66
DB 66
DB 74
DB 74
DB 82
DB 82
DB 90
DB 90
DB 98
DB 102
DB 106
DB 110
DB 114
DB 118
DB 122
DB 126
DB 130
DB 134
DB 138
DB 142
DB 146
DB 150
DB 154
DB 158
DB 166
DB 166
DB 174
DB 174
DB 182
DB 182
DB 190
DB 190
DB 206
DB 206
DB 206
DB 206
DB 222
DB 222
DB 222
DB 222
DB 238
DB 238
DB 238
DB 238
DB 254
DB 254

```
DB 254
DB 254

RDIRPCM:  DW DIRPCM1
           DW DIRPCM2
           DW DIRPCM3
           DW DIRPCM4
           DW DIRPCM5
           DW DIRPCM6
           DW DIRPCM7
           DW DIRPCM8

RDIRDPCM: DW DIRDPCM1
           DW DIRDPCM2
           DW DIRDPCM3
           DW DIRDPCM4
           DW DIRDPCM5
           DW DIRDPCM6
           DW DIRDPCM7
           DW DIRDPCM8

RREPPCM:  DW REPPCM1
           DW REPPCM2
           DW REPPCM3
           DW REPPCM4
           DW REPPCM5
           DW REPPCM6
           DW REPPCM7
           DW REPPCM8

RREPDPCM: DW REPDPCM1
           DW REPDPCM2
           DW REPDPCM3
           DW REPDPCM4
           DW REPDPCM5
           DW REPDPCM6
           DW REPDPCM7
           DW REPDPCM8

NOP
NOP
NOP
NOP
NOP
END;
```

APENDICE H

Cálculo del tiempo que dilatan las rutinas de interrupción

Tabla H.1
Bucle de espera de interrupción

NOMBRE	COMANDO	CICLOS	TOTAL
	INTERRUPCION	24	
LOOP:	EI	4	
	BIT 0,C	8	
	JR Z,FINLOOP	7	
	HALT	4	
	JR LOOP	12	59

Tabla H.2
Subrutina que finaliza rutinas "directo"

NOMBRE	COMANDO	CICLOS	TOTAL
FINDIR:	OUT (DAC),A	11	
	IN A,(KDC)	11	
	BIT 0,A	8	
	CALL NZ,ICTCN	10	
	RETI	14	54

Tabla H.3
Subrutina que finaliza rutinas "reproduciendo"

NOMBRE	COMANDO	CICLOS	TOTAL
FINREP:	OUT (DAC),A	11	
	INC HL	6	
	BIT 7,H	8	
	CALL Z,ICTCN	10	
	RETI	14	49

Tabla H.4
Subrutina que ajusta a 1 bit para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS1:	SRA A	8	
	SRA A	8	
	SRA A	8	
	SRA A	8	
	SRA A	8	
	SRA A	8	
	SRA A	8	
	RET	10	66

Tabla H.5
Subrutina que ajusta a 2 bits para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS2:	AND 11000000B	7	
	LD E,A	4	
	RRCA	4	
	RRCA	4	
	OR E	4	
	RRCA	4	
	RRCA	4	
	OR E	4	
	RRCA	4	
	RRCA	4	
	OR E	4	
	RET	10	57

Tabla H.6
Subrutina que ajusta a 3 bits para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS3:	AND 11100000B	7	
	LD E,A	4	
	RRCA	4	
	RRCA	4	
	RRCA	4	
	OR E	4	
	RLCA	4	
	RLCA	4	
	AND 00000011B	7	
	OR E	4	
	RET	10	56

Tabla H.7
Subrutina que ajusta a 4 bits para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS4:	AND 11110000B	7	
	LD E,A	4	
	RRCA	4	
	RRCA	4	
	RRCA	4	
	RRCA	4	
	OR E	4	
	RET	10	41

Tabla H.8
Subrutina que ajusta a 5 bits para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS5:	AND 11111000B	7	
	LD E,A	4	
	RLCA	4	
	RLCA	4	
	RLCA	4	
	AND 00000111B	7	
	OR E	4	
	RET	10	44

Tabla H.9
Subrutina que ajusta a 6 bits para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS6:	AND 11111100B	7	
	LD E,A	4	
	RLCA	4	
	RLCA	4	
	AND 00000011B	4	
	OR E	4	
	RET	10	37

Tabla H.10
Subrutina que ajusta a 7 bits para PCM

NOMBRE	COMANDO	CICLOS	TOTAL
BTS7:	AND 1111110B	7	
	LD E,A	4	
	RLCA	4	
	AND 0000001B	4	
	OR E	4	
	RET	10	33

Tabla H.11
Subrutina que modifica valor acumulado en DPCM

NOMBRE	COMANDO	MAX	MIN	>	>	<	<
BTSDPCM:	SUB D	4		4	4	4	4
	JR C,BTSDPA	12	7	7	7	12	12
	JR NZ,BTSDPD	12	7	7	12		
	INC A	4		4			
BTSDPD:	CP E	4		4	4		
	JR C,BTSDPB	12	7	7	12		
	LD A,E	4		4			
BTSDPB:	ADD D	7		7	7		
	LD D,A	4		4	4		
	RET NC	11	5	5	5		
	LD D,255	7		7	7		
	LD A,D	4		4	4		
	RET	10		10	10		
BTSDPA:	NEG	4				4	4
	CP E	4				4	4
	JR C,BTSDPC	12	7			7	12
	LD A,E	4				4	
BTSDPC:	LD B,A	4				4	4
	LD A,D	4				4	4
	SUB B	4				4	4
	LD D,A	4				4	4
	RET NC	11	5			5	5
	LD D,0	7				7	7
	LD A,D	4				4	4
	RET	10				10	10
TOTAL				74	76	77	78

Tabla H.12
Subrutina que ajusta valor acumulado en ADPCM

NOMBRE	COMANDO	MAX	MIN	>	>	<	<	
ADPCM:	CP D	4		4	4	4	4	
	JR C,ADPCMB	12	7	7	7	12	12	
	BIT 0,E	8		8	8			
	JR Z,ADPCMA	12	7	7	12			
	INC B	4		4				
	LD A,D	4		4				
	ADD B	4		4				
	LD D,A	4		4				
	RET NC	11	5	5				
	LD D,255	7		7				
	LD A,D	4		4				
	RET	10		10				
	ADPCMA:	LD E,1	7			7		
		LD B,1	7			7		
INC D		4			4			
LD A,D		4			4			
RET		10			10			
ADPCMB:	BIT 0,E	8				8	8	
	JR Z,ADPCMC	12	7			7	12	
	LD B,1	7				7		
	LD E,0	7				7		
	DEC D	4				4		
	LD A,D	4				4		
	RET	10				10		
ADPCMC:	INC B	4					4	
	LD A,D	4					4	
	SUB B	4					4	
	LD D,A	4					4	
	RET NC	11	5				5	
	LD D,0	7					7	
	LD A,D	4					4	
	RET	10					10	
TOTAL				68	63	63	78	

Tabla H.13
Subrutina que elige nivel para compansión

NOMBRE	COMANDO	CICLOS	TOTAL
COMPAN:	SRL	8	
	SRL	8	
	LD HL,COMPVAL	10	
	LD D,0	7	
	LD E,A	4	
	ADD HL,DE	11	
	LD A,(HL)	7	
	RET	10	65

Tabla H.14
Rutina para PCM directo a 1 bit

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM1:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS1	83			
	JP FINDIR	64	169	228	21.9

Tabla H.15
Rutina para PCM directo a 2 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM2:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS2	74			
	JP FINDIR	64	160	219	22.8

Tabla H.16
Rutina para PCM directo a 3 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM3:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS3	73			
	JP FINDIR	64	159	218	22.9

Tabla H.17
Rutina para PCM directo a 4 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM4:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS4	58			
	JP FINDIR	64	144	203	24.6

Tabla H.18
Rutina para PCM directo a 5 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM5:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS5	61			
	JP FINDIR	64	147	206	24.3

Tabla H.19
Rutina para PCM directo a 6 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM6:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS6	54			
	JP FINDIR	64	140	199	25.1

Tabla H.20
Rutina para PCM directo a 7 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM7:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL BTS7	50			
	JP FINDIR	64	136	195	25.6

Tabla H.21
Rutina para PCM directo a 8 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRPCM8:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	JP FINDIR	64	86	145	34.5

Tabla H.22
Rutina para DPCM directo a 1 bit

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM1:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,1	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.23
Rutina para DPCM directo a 2 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM2:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,2	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.24
Rutina para DPCM directo a 3 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM3:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,4	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.25
Rutina para DPCM directo a 4 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM4:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,8	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.26
Rutina para DPCM directo a 5 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM5:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,16	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.27
Rutina para DPCM directo a 6 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM6:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,31	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.28
Rutina para DPCM directo a 7 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM7:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,64	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.29
Rutina para DPCM directo a 8 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRDPCM8:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD E,128	7			
	CALL BTSDPCM	95			
	JP FINDIR	64	188	247	20.2

Tabla H.30
Rutina para ADPCM directo

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRADPCM:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL ADPCM	95			
	JP FINDIR	64	181	240	20.8

Tabla H.31
Rutina para compansión directo

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
DIRCOMN:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	CALL COMPAN	82			
	JP FINDIR	64	168	227	22.0

Tabla H.32
Rutina para PCM reproduciendo a 1 bit

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM1:	LD A,(HL)	7			
	CALL BTS1	83			
	JP FINREP	59	149	208	24.0

Tabla H.33
Rutina para PCM reproduciendo a 2 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM2:	LD A,(HL)	7			
	CALL BTS2	74			
	JP FINREP	59	140	199	25.1

Tabla H.34
Rutina para PCM reproduciendo a 3 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM3:	LD A,(HL)	7			
	CALL BTS3	73			
	JP FINREP	59	139	198	25.3

Tabla H.35
Rutina para PCM reproduciendo a 4 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM4:	LD A,(HL)	7			
	CALL BTS4	58			
	JP FINREP	59	124	183	27.3

Tabla H.36
Rutina para PCM reproduciendo a 5 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM5:	LD A,(HL)	7			
	CALL BTS5	61			
	JP FINREP	59	127	186	26.9

Tabla H.37
Rutina para PCM reproduciendo a 6 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM6:	LD A,(HL)	7			
	CALL BTS6	54			
	JP FINREP	59	120	179	27.9

Tabla H.38
Rutina para PCM reproduciendo a 7 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM7:	LD A,(HL)	7			
	CALL BTS7	50			
	JP FINREP	59	116	175	28.6

Tabla H.39
Rutina para PCM reproduciendo a 8 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPPCM8:	LD A,(HL)	7			
	JP FINREP	59	66	125	40.0

Tabla H.40
Rutina para DPCM reproduciendo a 1 bit

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM1:	LD A,(HL)	7			
	LD E,1	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.41
Rutina para DPCM reproduciendo a 2 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM2:	LD A,(HL)	7			
	LD E,2	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.42
Rutina para DPCM reproduciendo a 3 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM3:	LD A,(HL)	7			
	LD E,3	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.43
Rutina para DPCM reproduciendo a 4 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM4:	LD A,(HL)	7			
	LD E,7	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.44
Rutina para DPCM reproduciendo a 5 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM5:	LD A,(HL)	7			
	LD E,15	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.45
Rutina para DPCM reproduciendo a 6 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM6:	LD A,(HL)	7			
	LD E,31	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.46
Rutina para DPCM reproduciendo a 7 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM7:	LD A,(HL)	7			
	LD E,63	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.47
Rutina para DPCM reproduciendo a 8 bits

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPDPCM8:	LD A,(HL)	7			
	LD E,127	7			
	CALL BTSDPCM	95			
	JP FINREP	59	168	227	22.0

Tabla H.48
Rutina para ADPCM reproduciendo

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPADPCM:	LD A,(HL)	7			
	CALL ADPCM	95			
	JP FINREP	59	161	220	22.7

Tabla H.49
Rutina para compansión reproduciendo

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
REPCOMN:	LD A,(HL)	7			
	PUSH HL	10			
	CALL COMPAN	82			
	POP HL	10			
	JP FINREP	59	168	227	22.0

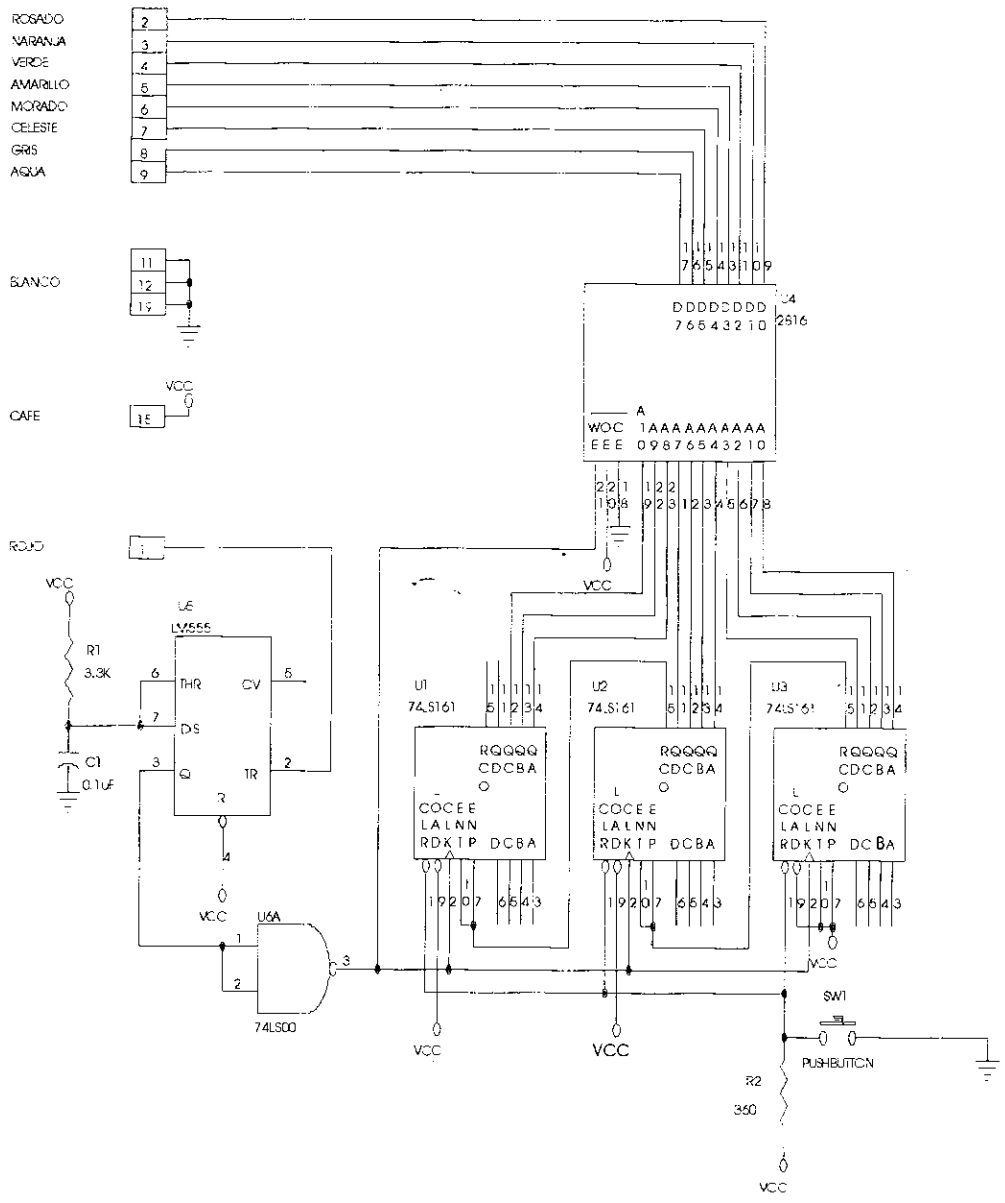
Tabla H.50
Rutina para grabar

NOMBRE	COMANDO	CICLOS	TOTAL	INT.	FREC.
GRABAR:	OUT (ADC),A	11			
	IN A,(ADC)	11			
	LD (HL),A	7			
	INC HL	6			
	BIT 7,H	8			
	CALL Z,ICTCN	10			
	RETI	14	67	126	39.7

APENDICE I

Diagrama esquemático del circuito para programar la EEPROM 2816

(ver siguiente página)



UNIVERSIDAD DEL VALLE
 GONZALO PALAREA
 CARNET 88103
 TRABAJO DE GRADUACION

Size		REV
A	PROGRAMADOR DE EEPROM 2816	
Date:	August 20, 1995	Sheet 1 of 1

APENDICE J

Listado del programa en BASIC para programar la ROM

```
DECLARE FUNCTION DEC (D$)
CLS
INPUT "Tamaño de la ROM en KB: ", T
INPUT "Inicio de su mapeo en Hexadecimal: ", CERO$
CERO = DEC(LEFT$(CERO$, 1)) * 4096 + DEC(MID$(CERO$, 2, 1)) * 256 + DEC(MID$(CERO$, 3,
1)) * 16 + DEC(RIGHT$(CERO$, 1))
ULT = 0

TAM = 1024 * T - 1

DIM DATO(TAM)

FOR C = 0 TO (TAM)
  LOCATE 8, 8: PRINT (TAM) - C; " ";
  DATO(C) = 255
NEXT C
LOCATE 3, 3
INPUT "Nombre del archivo (completo): ", na$
OPEN (na$) FOR INPUT AS #1
DO
  LINE INPUT #1, L$
  POSS$ = MID$(L$, 4, 4)
  POSI = DEC(LEFT$(POSS$, 1)) * 4096 + DEC(MID$(POSS$, 2, 1)) * 256 + DEC(MID$(POSS$, 3, 1)) *
16 + DEC(RIGHT$(POSS$, 1))
  POSI = POSI - CERO
  DATOSS$ = MID$(L$, 10, (LEN(L$) - 11))
  FOR C = 1 TO LEN(DATOSS$) STEP 2
    NUM = DEC(MID$(DATOSS$, C, 1)) * 16 + DEC(MID$(DATOSS$, C + 1, 1))
    IF DATO(POSI) <> 255 THEN PRINT "ERROR. DOS DATOS EN LA MISMA DIRECCION: ";
  POSI: END
  DATO(POSI) = NUM
  POSI = POSI + 1
  IF POSI > ULT THEN ULT = POSI
NEXT C
LOOP UNTIL (EOF(1))
CLOSE #1

SACAPUERTO:

OPEN "LPT1" FOR OUTPUT AS #1
FOR C = 0 TO ULT + 5
  LOCATE 8, 8:
```



```
PRINT ULT + 5 - C; "  "  
PRINT #1, CHR$(DATO(C));  
FOR DL = 1 TO 50  
NEXT DL  
NEXT C  
PRINT CHR$(7)  
END
```

```
FUNCTION DEC (D$)  
D = ASC(D$)  
D = D - 48  
IF D > 16 THEN D = D - 7  
DEC = D  
END FUNCTION
```

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

DIGITALIZADOR DE SEÑALES:

MANUAL DE OPERACION

GONZALO PALAREA

Guatemala

1995



DIGITALIZADOR DE SEÑALES:

MANUAL DE OPERACION

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ciencias y Humanidades

Departamento de Ingeniería Electrónica

DIGITALIZADOR DE SEÑALES:

MANUAL DE OPERACION

GONZALO PALAREA

Anexo al trabajo de graduación presentado para optar al grado
académico de Licenciado en Ingeniería Electrónica

Guatemala

1995



CONTENIDO

	Páginas
I. INTRODUCCION	1
II. ESPECIFICACIONES	3
A. Formas de operación	3
B. Entradas	3
C. Salidas	3
D. Filtros	4
E. Frecuencia de muestreo	4
F. Resolución	4
G. Métodos de transmisión	4
1. PCM	4
2. DPCM	10
3. ADPCM	10
4. Compansión	10
H. Dimensiones y peso	12
I. Alimentación	12
J. Panel de control digital	12
K. Panel de control analógico	13
III. OPERACION	15
A. Energizar el dispositivo	15

	Páginas
B. Elección de los parámetros de digitalización	16
1. Directo	17
2. Grabar	17
3. Reproducir	18
C. Sintonización de los filtros	19
1. Filtro de entrada	19
2. Filtro de salida	19
D. Ajuste	19
E. Volumen	20
IV. CALCULO DEL ANCHO DE BANDA	21

LISTA DE FIGURAS Y TABLAS

Figura		Página
2.1	Valores utilizados en la compresión digital para lograr la expansión	11
2.2	Control de alimentación	12
2.3	Panel de control digital	13
2.4	Panel de control analógico	13

Tabla		Página
2.1	Frecuencias de muestreo exactas que emplea el dispositivo	5
2.2	Valores asignados a los niveles para el PCM a 1 bit	6
2.3	Valores asignados a los niveles para el PCM a 2 bits	6
2.4	Valores asignados a los niveles para el PCM a 3 bits	6
2.5	Valores asignados a los niveles para el PCM a 4 bits	7
2.6	Valores asignados a los niveles para el PCM a 5 bits	7
2.7	Valores asignados a los niveles para el PCM a 6 bits	8
2.8	Valores asignados a los niveles para el PCM a 7 bits	9
2.9	Valores asignados para la compresión digital	11
3.1	Opciones del panel de control digital y sus posibles valores	16



I. INTRODUCCION

A pesar que el efecto que causa la variación de cada parámetro de la digitalización puede ser calculado matemáticamente con un análisis cuantitativo, el análisis cualitativo resulta muy útil en la determinación de los parámetros que cumplen satisfactoriamente con el objetivo esperado. El digitalizador de señales permite efectuar rápidamente un estudio cualitativo de los parámetros necesarios para digitalizar y reconstruir una señal dada.

Este manual se ha organizado de la siguiente manera: en el capítulo 2 se presentan las características del dispositivo; en el capítulo 3 se explica la forma de operación del mismo; como generalmente el fin de la digitalización de una señal es su transmisión, se incluyó en el capítulo 4 la forma de calcular el ancho de banda necesario para transmitir la señal, dados los parámetros elegidos.

II. ESPECIFICACIONES

A. Formas de operación

El dispositivo tiene dos formas de operación: directo y muestreado. En el modo directo, las señales que se toman de la entrada son procesadas según los parámetros elegidos y se coloca el resultado en la salida. En el modo muestreado, se graba en memoria una muestra de la señal de entrada, para luego ser reproducida según los parámetros que se hayan elegido. Se dispone de 4 áreas de memoria para muestras, las cuales tienen una memoria disponible de 32720 bytes.

B. Entradas

Se cuenta con dos entradas distintas: una para micrófono y otra para señales en general, por ejemplo, provenientes de un generador de funciones. El micrófono debe ser un "electret", y se conecta con un "mini plug" monaural. La entrada para señales en general cuenta con un conector tipo BNC, y su impedancia de entrada es de $50k\Omega$. Esta entrada no debe exceder un voltaje de $5V_{pp}$ para que no sea recortada.

C. Salidas

Hay dos salidas: una para bocina, la cual debe ser conectada utilizando un "mini plug" monaural, con una impedancia de 3 a 16Ω , y otra por un conector BNC, con una impedancia de 75Ω .

D. Filtros

Se cuenta con un filtro de entrada y uno de salida. Ambos son filtros pasa-bajos de orden 5, sintonizables desde 100Hz hasta 10kHz.

E.. Frecuencia de muestreo

La frecuencia de muestreo se puede variar desde 0.2 hasta 20.0kHz, en pasos de 0.2kHz. En la tabla 2.1 se muestra la frecuencia exacta a la cual trabaja el dispositivo, según la frecuencia que se indica en el panel de control digital.

F. Resolución

Para la digitalización se utiliza una resolución de 8 bits (256 niveles) máximo.

G. Métodos de transmisión

Si bien el dispositivo no transmite las señales digitalizadas, si puede simular distintos métodos de transmisión. Estos métodos son: PCM, DPCM, ADPCM y PCM con compansión.

1. PCM. El método PCM se puede trabajar con resoluciones de 1 a 8 bits. La conversión análogo-digital se efectúa siempre a 8 bits, y dependiendo de la resolución elegida, se asignan los niveles a cada muestra. De esta forma, a la señal de salida (no filtrada) se le asignan 2^n niveles, donde n es la cantidad de bits elegida. En las tablas 2.2 a

2.8 se muestran los valores asignados a los niveles, según la cantidad de bits de resolución elegidos, para PCM de 1 a 7 bits. Al PCM de 8 bits se le asigna a cada nivel el valor obtenido en la conversión.

Tabla 2.1

Frecuencias de muestreo exactas que emplea el dispositivo (en kHz)

Indicada	Real	Indicada	Real	Indicada	Real	Indicada	Real
0.2	0.200	5.2	5.198	10.2	10.204	15.2	15.244
0.4	0.400	5.4	5.411	10.4	10.417	15.4	15.432
0.6	0.600	5.6	5.605	10.6	10.593	15.6	15.625
0.8	0.800	5.8	5.787	10.8	10.823	15.8	15.823
1.0	1.000	6.0	5.995	11.0	11.013	16.0	16.026
1.2	1.199	6.2	6.203	11.2	11.211	16.2	16.234
1.4	1.400	6.4	6.394	11.4	11.416	16.4	16.447
1.6	1.601	6.6	6.614	11.6	11.574	16.6	16.556
1.8	1.799	6.8	6.793	11.8	11.792	16.8	16.779
2.0	2.000	7.0	7.003	12.0	12.019	17.0	17.007
2.2	2.201	7.2	7.184	12.2	12.195	17.2	17.241
2.4	2.397	7.4	7.396	12.4	12.376	17.4	17.361
2.6	2.599	7.6	7.599	12.6	12.626	17.6	17.606
2.8	2.800	7.8	7.788	12.8	12.821	17.8	17.857
3.0	3.001	8.0	8.013	13.0	13.021	18.0	17.986
3.2	3.201	8.2	8.197	13.2	13.228	18.2	18.248
3.4	3.401	8.4	8.389	13.4	13.369	18.4	18.382
3.6	3.597	8.6	8.591	13.6	13.587	18.6	18.657
3.8	3.799	8.8	8.803	13.8	13.812	18.8	18.797
4.0	4.000	9.0	8.993	14.0	13.966	19.0	18.939
4.2	4.202	9.2	9.191	14.2	14.205	19.2	19.231
4.4	4.401	9.4	9.398	14.4	14.368	19.4	19.380
4.6	4.604	9.6	9.615	14.6	14.620	19.6	19.531
4.8	4.808	9.8	9.804	14.8	14.793	19.8	19.841
5.0	5.000	10.0	10.000	15.0	14.970	20.0	20.000

Tabla 2.2

Valores asignados a los niveles para el PCM a 1 bit

Nivel	Valor
1	00000000 _b
2	11111111 _b

Tabla 2.3

Valores asignados a los niveles para el PCM a 2 bits

Nivel	Valor
1	00000000 _b
2	01010101 _b
3	10101010 _b
4	11111111 _b

Tabla 2.4

Valores asignados a los niveles para el PCM a 3 bits

Nivel	Valor	Nivel	Valor
1	00000000 _b	5	10010010 _b
2	00100100 _b	6	10110110 _b
3	01001001 _b	7	11011011 _b
4	01101101 _b	8	11111111 _b

Tabla 2.5

Valores asignados a los niveles para el PCM a 4 bits

Nivel	Valor	Nivel	Valor
1	00000000 _b	9	10001000 _b
2	00010001 _b	10	10011001 _b
3	00100010 _b	11	10101010 _b
4	00110011 _b	12	10111011 _b
5	01000100 _b	13	11001100 _b
6	01010101 _b	14	11011101 _b
7	01100110 _b	15	11101110 _b
8	01110111 _b	16	11111111 _b

Tabla 2.6

Valores asignados a los niveles para el PCM a 5 bits

Nivel	Valor	Nivel	Valor
1	00000000 _b	17	10000100 _b
2	00001000 _b	18	10001100 _b
3	00010000 _b	19	10010100 _b
4	00011000 _b	20	10011100 _b
5	00100001 _b	21	10100101 _b
6	00101001 _b	22	10101101 _b
7	00110001 _b	23	10110101 _b
8	00111001 _b	24	10111101 _b
9	01000010 _b	25	11000110 _b
10	01001010 _b	26	11001110 _b
11	01010010 _b	27	11010110 _b
12	01011010 _b	28	11011110 _b
13	01100011 _b	29	11100111 _b
14	01101011 _b	30	11101111 _b
15	01110011 _b	31	11110111 _b
16	01111011 _b	32	11111111 _b

Tabla 2.7

Valores asignados a los niveles para el PCM a 6 bits

Nivel	Valor	Nivel	Valor
1	00000000 _b	33	1000010 _b
2	00000100 _b	34	10000110 _b
3	00001000 _b	35	10001010 _b
4	00001100 _b	36	10001110 _b
5	00010000 _b	37	10010010 _b
6	00010100 _b	38	10010110 _b
7	00011000 _b	39	10011010 _b
8	00011100 _b	40	10011110 _b
9	00100000 _b	41	10100010 _b
10	00100100 _b	42	10100110 _b
11	00101000 _b	43	10101010 _b
12	00101100 _b	44	10101110 _b
13	00110000 _b	45	10110010 _b
14	00110100 _b	46	10110110 _b
15	00111000 _b	47	10111010 _b
16	00111100 _b	48	10111110 _b
17	01000001 _b	49	11000011 _b
18	01000101 _b	50	11000111 _b
19	01001001 _b	51	11001011 _b
20	01001101 _b	52	11001111 _b
21	01010001 _b	53	11010011 _b
22	01010101 _b	54	11010111 _b
23	01011001 _b	55	11011011 _b
24	01011101 _b	56	11011111 _b
25	01100001 _b	57	11100011 _b
26	01100101 _b	58	11100111 _b
27	01101001 _b	59	11101011 _b
28	01101101 _b	60	11101111 _b
29	01110001 _b	61	11110011 _b
30	01110101 _b	62	11110111 _b
31	01111001 _b	63	11111011 _b
32	01111101 _b	64	11111111 _b

Tabla 2.8

Valores asignados a los niveles para el PCM a 7 bits

Nivel	Valor	Nivel	Valor	Nivel	Valor	Nivel	Valor
1	0000000 _b	33	0100000 _b	65	1000001 _b	97	1100001 _b
2	0000010 _b	34	0100010 _b	66	1000011 _b	98	1100011 _b
3	0000100 _b	35	0100100 _b	67	1000101 _b	99	1100101 _b
4	0000110 _b	36	0100110 _b	68	1000111 _b	100	1100111 _b
5	00001000 _b	37	01001000 _b	69	10001001 _b	101	11001001 _b
6	00001010 _b	38	01001010 _b	70	10001011 _b	102	11001011 _b
7	00001100 _b	39	01001100 _b	71	10001101 _b	103	11001101 _b
8	00001110 _b	40	01001110 _b	72	10001111 _b	104	11001111 _b
9	00010000 _b	41	01010000 _b	3	10010001 _b	105	11010001 _b
10	00010010 _b	42	01010010 _b	74	10010011 _b	106	11010011 _b
11	00010100 _b	43	01010100 _b	75	10010101 _b	107	11010101 _b
12	00010110 _b	44	01010110 _b	76	10010111 _b	108	11010111 _b
13	00011000 _b	45	01011000 _b	77	10011001 _b	109	11011001 _b
14	00011010 _b	46	01011010 _b	78	10011011 _b	110	11011011 _b
15	00011100 _b	47	01011100 _b	79	10011101 _b	111	11011101 _b
16	00011110 _b	48	01011110 _b	80	10011111 _b	112	11011111 _b
17	00100000 _b	49	01100000 _b	81	10100001 _b	113	11100001 _b
18	00100010 _b	50	01100010 _b	82	10100011 _b	114	11100011 _b
19	00100100 _b	51	01100100 _b	83	10100101 _b	115	11100101 _b
20	00100110 _b	52	01100110 _b	84	10100111 _b	116	11100111 _b
21	00101000 _b	53	01101000 _b	85	10101001 _b	117	11101001 _b
22	00101010 _b	54	01101010 _b	86	10101011 _b	118	11101011 _b
23	00101100 _b	55	01101100 _b	87	10101101 _b	119	11101101 _b
24	00101110 _b	56	01101110 _b	88	10101111 _b	120	11101111 _b
25	00110000 _b	57	01110000 _b	89	10110001 _b	121	11110001 _b
26	00110010 _b	58	01110010 _b	90	10110011 _b	122	11110011 _b
27	00110100 _b	59	01110100 _b	91	10110101 _b	123	11110101 _b
28	00110110 _b	60	01110110 _b	92	10110111 _b	124	11110111 _b
29	00111000 _b	61	01111000 _b	93	10111001 _b	125	11111001 _b
30	00111010 _b	62	01111010 _b	94	10111011 _b	126	11111011 _b
31	00111100 _b	63	01111100 _b	95	10111101 _b	127	11111101 _b
32	00111110 _b	64	01111110 _b	96	10111111 _b	128	11111111 _b

2. DPCM. La diferencia entre muestras en el método DPCM se puede cuantificar con resoluciones de 1 a 8 bits. La cantidad de niveles que se pueden variar de una muestra a otra (hacia arriba o hacia abajo) viene dado por $2^{(n-1)}$, donde n es el número de bits elegidos para la cuantificación de la diferencia.

3. ADPCM. El método ADPCM se trabaja únicamente a 1 bit, y el algoritmo utilizado es el siguiente:

- Inicializar el valor acumulado a 128 ($A_0=128$)

- Repetir: Tomar la muestra M_n .

Calcular el valor D_n según la ecuación II-1.

Tomar $A_n = A_{n-1} + D_n$

Hasta que se desee terminar.

$$D_n = \begin{cases} D_{n-1} + 1 & \text{si } M_n > A_{n-1} \wedge M_{n-1} > A_{n-2} \\ D_{n-1} - 1 & \text{si } M_n < A_{n-1} \wedge M_{n-1} < A_{n-2} \\ 0 & \text{si } M_n > A_{n-1} \wedge M_{n-1} < A_{n-2} \vee M_n < A_{n-1} \wedge M_{n-1} > A_{n-2} \end{cases} \quad (\text{II-1})$$

4. Compansión. Para la compansión se utiliza el método de compresión digital, comprimiendo las muestras de 8 a 5 bits (es decir de 256 a 32 niveles). Los valores asignados en la compresión se muestran en la tabla 2.9 y se grafican en la figura 2.1.

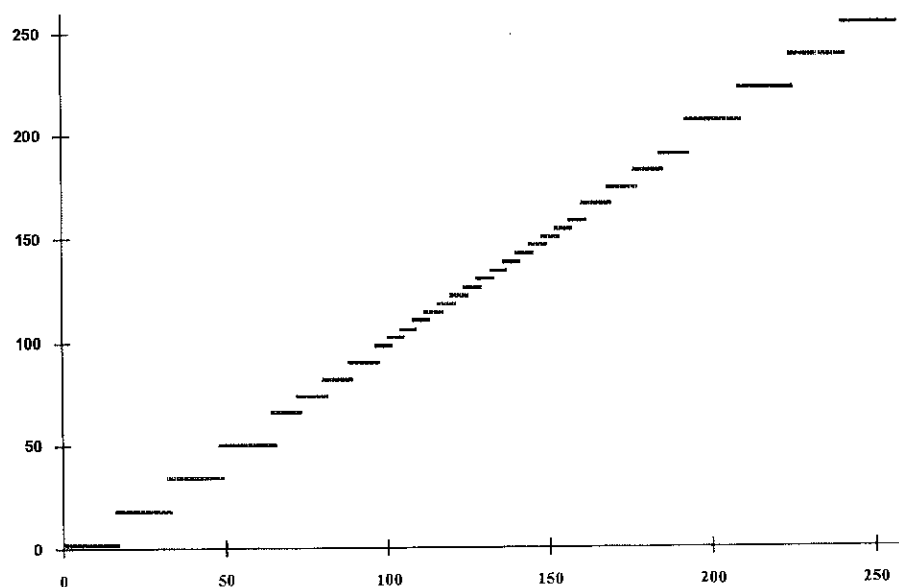


Figura 2.1. Valores utilizados en la compresión digital para lograr la compansión.

Tabla 2.9

Valores asignados para la compresión digital
A=valor a 8 bits; B=valor asignado

A	B	A	B	A	B	A	B
0-15	2	96-99	98	128-131	130	160-167	166
16-31	18	100-103	102	132-135	134	168-175	174
32-47	34	104-107	106	136-139	138	176-183	182
48-63	50	108-111	110	140-143	142	184-191	190
64-71	66	112-115	114	144-147	146	192-207	206
72-79	74	116-119	118	148-151	150	208-223	222
80-87	82	120-123	122	152-155	154	224-239	238
88-95	90	124-127	126	156-159	158	240-255	254

H. Dimensiones y peso

El dispositivo está contenido en una caja metálica de 6.5 cm de alto, 15.5 cm de ancho y 17.5 cm de fondo. Su peso es de 2lb.

I. Alimentación

El dispositivo se alimenta con 24 a 26 VAC con derivación central. Posee un conector de cuatro contactos y un interruptor de doble polo. Se puede observar la ubicación del conector y el interruptor en la figura 2.2, la cual muestra la parte posterior del dispositivo. En dicha figura se muestra también el interruptor de reinicialización.

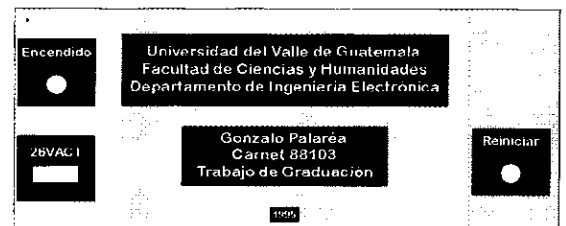


Figura 2.2. Control de alimentación.

J. Panel de control digital

El panel de control digital se encuentra en la parte superior del dispositivo y contiene los indicadores y botones necesarios para elegir los parámetros de la digitalización. En la figura 2.3 se muestra este panel.

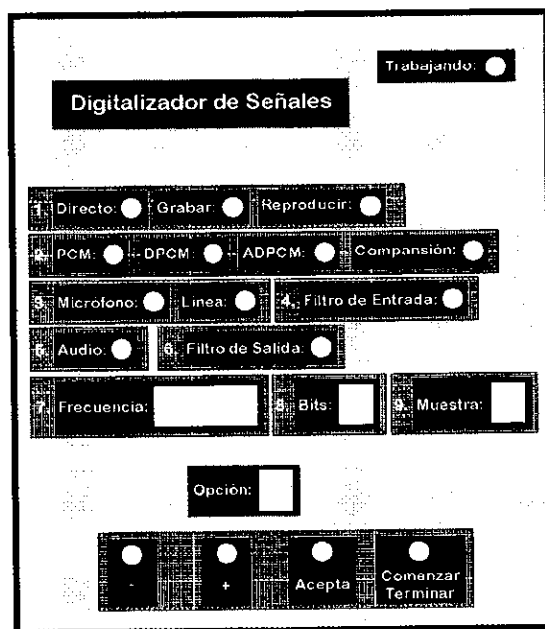


Figura 2.3. Panel de control digital.

K. Panel de control analógico

En este panel se conectan las señales externas, se gradúan los filtros, el volumen y se ajusta el nivel de entrada al convertidor análogo-digital. Se encuentra en la parte frontal del dispositivo y se muestra en la figura 2.4.

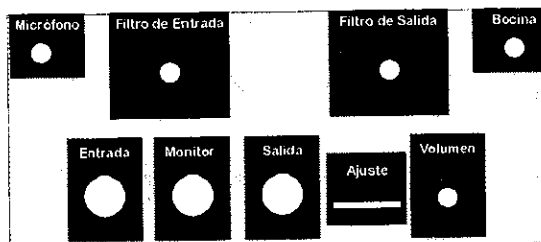


Figura 2.4. Panel de control analógico.

El conector rotulado "monitor" es de tipo BNC, y su función es la de proveer una conexión para monitorear la entrada al convertidor análogo-digital. Su impedancia de salida es de 75Ω . Las señales que se obtengan de este conector, estarán montadas sobre 2.5VDC.

III. OPERACION

A. Energizar el dispositivo

Para energizar el dispositivo se debe conectar el transformador en la parte posterior y colocar el interruptor en la posición de encendido.

Si en algún momento el dispositivo trabaja de manera irregular, se debe presionar el botón de reiniciar, el cual se encuentra ubicado en la parte posterior.

Al momento del encendido o reinicio, el dispositivo toma los siguientes valores iniciales:

- Forma de operación: directo
- Método de transmisión: PCM
- Fuente de señal de entrada: micrófono
- Filtro de entrada: desactivado
- Audio: desactivado
- Filtro de salida: desactivado
- Frecuencia de muestreo: 8.0 kHz
- Bits de resolución: 8
- Muestra: 1
- Opción: 1

B. Elección de los parámetros de digitalización

En la casilla de “opción”, el dispositivo indica cual de las opciones se está modificando en ese momento. Las opciones están numeradas en el panel de control digital, y son las que se indican en la tabla 3.1.

Tabla 3.1

Opciones del panel de control digital y sus posibles valores

Número de Opción	Descripción	Valores posibles
1	Forma de operación	directo, grabar, reproducir
2	Método de transmisión	PCM, DPCM, ADPCM, Compansión
3	Señal de entrada	Micrófono, Línea
4	Filtro de entrada	Activado, Desactivado
5	Audio	Activado, Desactivado
6	Filtro de salida	Activado, Desactivado
7	Frecuencia de muestreo	0.2-20.0 kHz
8	Bits de resolución	1-8
9	Muestra	1-4

Para elegir los parámetros de la digitalización, se utilizan los cuatro botones del panel de control digital. Con los botones “-” y “+” se varía el valor de la opción que se está eligiendo, con el botón “acepta” se le indica al dispositivo que el valor que está desplegando en ese momento es correcto, y con el botón “comenzar/terminar” se le indica que todas las opciones están correctas y se desea comenzar la digitalización. Si se está trabajando en la forma “directo”, este botón sirve también para terminar la digitalización.

1. Directo. Si en la opción 1 se elige la forma de operación "directo", el dispositivo permitirá modificar, en el orden indicado, las siguientes opciones:

- Método de transmisión (opción 2)
- Señal de entrada (opción 3)
- Filtro de entrada (opción 4)
- Audio (opción 5)
- Filtro de salida (opción 6)
- Frecuencia de muestreo (opción 7)
- Bits de resolución (opción 8) Si el método elegido en la opción 2 es "ADPCM", esta opción no se puede modificar y queda fija en 1. Si el método elegido es "Compansión", tampoco se puede modificar y queda fija en 5.

Cuando se presione el botón de "comenzar/terminar", el dispositivo encenderá la luz de "trabajando", y digitalizará según los parámetros indicados hasta que se presione de nuevo el botón de "comenzar/terminar"

2. Grabar. Si en la opción 1 se elige la forma de operación "grabar", el dispositivo permitirá modificar, en el orden indicado, las siguientes opciones:

- Muestra a grabar (opción 9)
- Señal de entrada (opción 3)
- Filtro de entrada (opción 4)
- Frecuencia de muestreo (opción 7)

En esta forma de operación, el audio (opción 5) y el filtro de salida (opción 6) permanecen desactivados. Los bits de resolución no se pueden modificar y permanecen en 8.

Cuando se presione el botón de “comenzar”, el dispositivo encenderá la luz de “trabajando” y grabará las muestras en el area de memoria elegida, hasta llenarla. Así, el tiempo que toma para completar la grabación, T_{GR} , viene dado por la ecuación III-1, donde f_m es la frecuencia de muestreo elegida.

$$T_{GR} = \frac{32720}{f_m} \quad (\text{III-1})$$

3. Reproducir. Si en la opción 1 se elige la forma de operación “reproducir”, el dispositivo permitirá modificar, en el orden indicado, las siguientes opciones:

- Muestra a grabar (opción 9)
- Método de transmisión (opción 2)
- Audio (opción 5)
- Filtro de salida (opción 6)
- Frecuencia de muestreo (opción 7)
- Bits de resolución (opción 8) Si el método elegido en la opción 2 es “ADPCM”, esta opción no se puede modificar y queda fija en 1. Si el método elegido es “Compansión”, tampoco se puede modificar y queda fija en 5.

En esta forma de operación, las señales de entrada (opción 3) y el filtro de entrada (opción 4) permanecen desactivados. La frecuencia de muestreo se puede modificar, pero si luego de completar la reproducción se intenta modificar algún parámetro, ésta se restaurará al valor utilizado al momento de grabar.

Cuando se presione el botón de “comenzar”, el dispositivo encenderá la luz de “trabajando” y reproducirá las muestras almacenadas en el area de memoria elegida, hasta completarla. Así, el tiempo que toma para completar la reproducción, T_{GR} , viene dado por la ecuación III-1, donde f_m es la frecuencia de muestreo elegida.

C. Sintonización de los filtros

Se recomiendan los siguientes procedimientos para sintonizar los filtros:

1. Filtro de entrada. Conectar un generador de señales a la entrada del dispositivo, el cual genere una señal senoidal de la frecuencia a la cual se desea que el filtro de entrada recorte. Conectar un osciloscopio al conector "monitor" del dispositivo. Girar la perilla del filtro de entrada hasta la izquierda, y luego ir girando hacia la derecha hasta que la señal que se observa en el osciloscopio se reduzca su amplitud al 70% (3dB).

2. Filtro de salida. Conectar un generador de señales a la entrada del dispositivo, el cual genere una señal senoidal de la frecuencia a la cual se desea que el filtro de salida recorte. Poner a trabajar el dispositivo en "directo", con el método "PCM" a 8 bits, con la señal de entrada de "línea", con el filtro de entrada desactivado, el filtro de salida activado y la frecuencia de muestreo al máximo (20.0kHz). Conectar un osciloscopio al conector de salida del dispositivo. Girar la perilla del filtro de salida hasta la izquierda, y luego ir girando hacia la derecha hasta que la señal que se observa en el osciloscopio se reduzca su amplitud a 70% (3dB).

D. Ajuste

En ocasiones, es posible que se obtenga un ruido de fondo al estar trabajando el dispositivo. Este ruido se puede eliminar, haciendo girar el botón de "ajuste" ubicado en el panel de control analógico. Dicho ruido se obtiene cuando el nivel de reposo de la señal de entrada al convertidor analógico-digital está muy próximo a un umbral de decisión de nivel; al mover este botón, se hace variar el nivel de reposo, eliminando el ruido.

E. Volumen

Si se desea trabajar con el audio activado, se puede ajustar el volumen en el panel de control analógico, girando la perilla que hay para este efecto. Se debe tener precaución de no poner demasiado volumen, de tal forma que el sonido no se distorsione debido a saturación del amplificador de audio. El volumen se aumenta girando la perilla hacia la derecha, y se disminuye girándola hacia la izquierda.

IV. CALCULO DEL ANCHO DE BANDA

El ancho de banda necesario para transmitir la señal por un canal digital de comunicaciones viene dado por:

$$B = f_m \times n \quad (\text{IV-1})$$

donde f_m es la frecuencia de muestreo y n la cantidad de bits utilizados para la cuantificación, independientemente del método de transmisión que se utilice. La unidad utilizada para medir el ancho de banda es bits/s.