

107
2ej



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA
U. N. A. M.

FALLA DE ORIGEN

DISEÑO Y REALIZACION DE UN SISTEMA
DE CONVERSION A/D - D/A PARA
COMPRESION Y TRANSMISION DE
SEÑALES DE VOZ

FALLA DE ORIGEN

T E S I S

QUE PARA OBTENER EL TITULO DE

INGENIERO MECANICO ELECTRICISTA

P R E S E N T A N

DONALDO ROJAS REYES

DEMETRIO TREJO BECERRIL

Director de Tesis DR. ROGELIO ALCANTARA SILVA

MEXICO, D. F.

FALLA DE ORIGEN

1989



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

INDICE

CAPITULO I. INTRODUCCION.	1
CAPITULO II. SISTEMA DE CONVERSION A/D - D/A.	9
II.1. INTRODUCCION.	9
II.2. CONVERTIDORES DIGITAL - ANALOGICO (D/A).	11
II.2.1 Convertidor D/A de Resistencias Ponderadas.	11
II.2.2 Convertidor D/A de Escalera R-2R.	12
II.2.3 Convertidor D/A Serie.	14
II.2.4 Parámetros Utilizados en los Convertidores D/A.	14
II.3. CONVERTIDORES ANALOGICO - DIGITAL (A/D).	15
II.3.1 Convertidores que Emplean Técnicas de Carga y Descarga de un capacitor.	16
II.3.1.1 Convertidor de integración simple.	16
II.3.1.2 Convertidor de doble integración.	19
II.3.1.3 Convertidor de voltaje a frecuencia.	21
II.3.2 Convertidor A/D por Comparación de Voltaje.	22
II.3.2.1 Convertidor de rampa discreta.	22
II.3.2.2 Convertidor de balance continuo.	23
II.3.2.3 Convertidor de aproximaciones sucesivas.	25
II.3.2.4 Convertidor paralelo.	26
II.3.3 Ventajas y Desventajas de los Diferentes Convertidores.	29
II.3.4 Parámetros Utilizados en los Convertidores A/D.	30

CAPITULO III. DIGITALIZACION DE VOZ.	34
III.1 INTRODUCCION.	34
III.2 MUESTREO.	35
II.2.1 Teorema del Muestreo.	35
II.2.2 Muestreo con un Tren de Impulsos.	36
III.3. CUANTIZACION.	40
III.3.1 Cuantización Uniforme.	45
III.3.2 Cuantización No Uniforme.	47
III.3.2.1 Compansión.	49
III.3.2.1.1 Ley μ 255 de compansión.	50
III.3.2.1.2 Ley A de compansión.	56
III.3.3 Comparación de la Relación Señal a Ruido entre la Cuantización Uniforme y No uniforme.	60
III.3.4 Cuantización Adaptiva.	62
III.3.5 Cuantización Diferencial.	63
CAPITULO IV. DISEÑO Y REALIZACION DEL SISTEMA CODEC.	66
IV. 1 INTRODUCCION.	66
IV. 2 ETAPA DE FILTRADO.	69
IV.2.1 Filtrado de Entrada.	70
IV.2.2 Filtrado de Salida.	76
IV. 3 CONVERSION A/D - D/A.	77
IV. 3.1 Codec.	77
IV. 3.2 Codec TPS116.	78
IV. 4 BASE DE TIEMPO.	80

IV.5 CONVERSION SERIE/PARALELO - PARALELO/SERIE.	83
IV.5.1 Conversión Serie/Paralelo.	83
IV.5.2 Conversión Paralelo/Serie.	85
IV.6 EXPANSION Y COMPRESION.	85
CAPITULO V. DISEÑO DE LAS INTERFASES PARA LA PC Y EL TMS32010.	88
V.1 INTRODUCCION.	88
V.2 INTERFASE SISTEMA CODEC - PC.	88
V.2.1 Hardware de Entrada/Salida de la PC.	88
V.2.2 Acceso Directo a Memoria (DMA).	93
V.2.3 Diseño y Realización de la Interfase Sistema CODEC - PC.	97
V.2.4 Programación de la Interfase Sistema CODEC - PC.	98
V.2.4.1 Programación para la conversión A/D.	98
V.2.4.2 Programación para la conversión D/A.	100
V.3 INTERFASE SISTEMA CODEC - TMS32010.	103
V.3.1 El Microprocesador TMS32010.	103
V.3.2 Características del microprocesador TMS32010.	103
V.3.3 Arquitectura de un TMS32010.	108
V.3.4 Diseño y Realización de la Interfase Sistema CODEC - TMS32010.	110
V.4 APLICACIONES	113
CAPITULO VI. CONCLUSIONES Y PERSPECTIVAS .	116
BIBLIOGRAFIA.	119
APENDICE.	

1. INTRODUCCION.

Los descubrimientos científicos y tecnológicos obtenidos en los últimos años en áreas como la electrónica, la computación, las comunicaciones y el control, han generado nuevas técnicas y metodologías para la solución de problemas a los cuales se enfrenta el hombre diariamente. Esta afluencia de resultados dió origen a una nueva disciplina de la ingeniería, "El Procesamiento Digital de Señales".

Se puede decir que el Procesamiento Digital de Señales (PDS) nos permite estimar parámetros característicos de una señal para transformarla a una forma más conveniente o deseable para una aplicación determinada. Es por ello que el desarrollo de sistemas para el PDS, a despertado un gran interés en áreas como la transmisión digital de señales, sistemas de radar, imágenes biomédicas, análisis, síntesis y reconocimiento de voz entre otras, ya que su uso representa una herramienta muy útil para poder generar como resultado señales con una mejor calidad con respecto a las señales originales de entrada.

También se puede mencionar que hoy en día el tratamiento digital de imágenes es un campo de gran interés en el PDS. Las degradaciones existentes en una imagen, debidas fundamentalmente a diferencias en los dispositivos de captación, constituyen una motivación para intentar mejorar el contenido informativo de la señal, así como también la de reducir la alta capacidad de memoria necesaria para almacenar una sucesión de escenas con degradación mínima y el de disminuir el elevado tiempo de proceso de las mismas. Es en este tipo de problemas donde entran los dispositivos PDS.

En la figura 1.1 se muestra un diagrama de un sistema de PDS donde una parte importante del sistema es su respectivo procesador, que podría ser un microprocesador (μP) de propósito general o en aplicaciones más sofisticadas un μP de propósito particular, cuya arquitectura permite efectuar operaciones conjuntas de los múltiples procesos con mayor rapidez que un microprocesador ordinario, el cual está diseñado especialmente para el PDS. Así mismo, podemos observar que otra de las partes fundamentales del sistema es la conversión o adquisición de la señal, ya que de esta etapa depende que la señal que se procese se represente con una gran calidad con respecto a la señal de entrada. Una vez que la señal ha sido procesada, esta se puede almacenar o sacar al mundo real para alguna aplicación particular.

Dado que el objetivo del PDS es el de tratar de mejorar el contenido informativo de las señales procesándolas digitalmente, fue necesario enfrentarse al problema desde la parte de su adquisición, que es una de las más importantes dentro del sistema PDS, es por ello que para nuestro caso se diseñó y realizó un sistema de conversión analógica-digital (A/D) y digital analógica (D/A) para compresión y transmisión de señales de voz, de tal manera que se tuviera una interfase a una computadora personal (PC) y al procesador de señales TMS32010.

Antes de tratar de proponer una arquitectura de PDS para aplicaciones de señales de voz, se tuvieron que estudiar las características de esta señal. Haciendo un análisis del modelo estadístico de la voz, se pudo encontrar que la mayor cantidad de información se encuentra en las amplitudes pequeñas, lo que motivó a la investigación sobre las técnicas de cuantización más eficientes que existen, de tal manera que nos

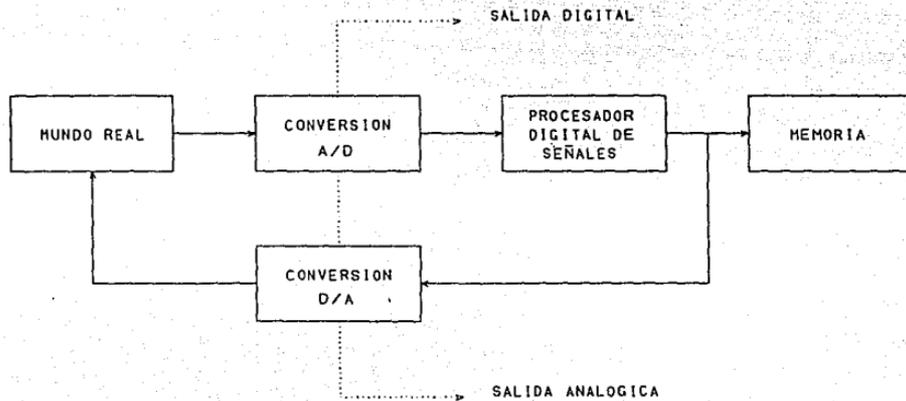


FIG. 1.1 DIAGRAMA DE UN SISTEMA DE PROCESAMIENTO DIGITAL DE SEÑALES

proporcionara un criterio para poder escoger un circuito que permita hacer la conversión A/D y D/A lo mejor posible. Dentro del estudio de todas las técnicas de cuantización, las técnicas de cuantización uniforme y no uniforme son las que proporcionan el hardware necesario para la realización del sistema, lo cual no quiere decir que no haya hardware implantado en otras técnicas de cuantización, la razón es que las técnicas implantadas con la de cuantización uniforme y no uniforme son las más accesibles en cuanto a facilidad de adquisición, además de proporcionarnos un bajo costo con respecto a las otras.

Ahora bien, la información que nos proporcionó el modelo estadístico de la voz, hizo que nos inclináramos por la cuantización no uniforme, debido a que la digitalización de la señal de voz es más eficiente si la amplitud de la señal es comprimida (reduciendo el número de bits requeridos en su representación) a una escala logarítmica y después es transmitida para posteriormente, en la recepción, ser expandida y eventualmente procesada. La conversión a escala logarítmica asegura que las bajas amplitudes de la señal estén digitalizadas con una mínima pérdida de fidelidad. Esta técnica de COMPRESIÓN y luego de EXPANSIÓN se conoce como COMPANSION, la cual se encuentra implantado en hardware, en circuitos conocidos como CODECS.

Analizando las ventajas en hardware que nos proporciona la cuantización no uniforme, se diseñó el sistema conversión A/D y D/A, el cual consta principalmente de tres partes: Sistema CODEC, Interfase Sistema CODEC-PC e Interfase Sistema CODEC-TMS32010 (figura 1.2).

En la etapa del Sistema CODEC, la señal de voz se acondicionó a un rango de frecuencias de 300-3400 Hz, que es el rango empleado en una línea

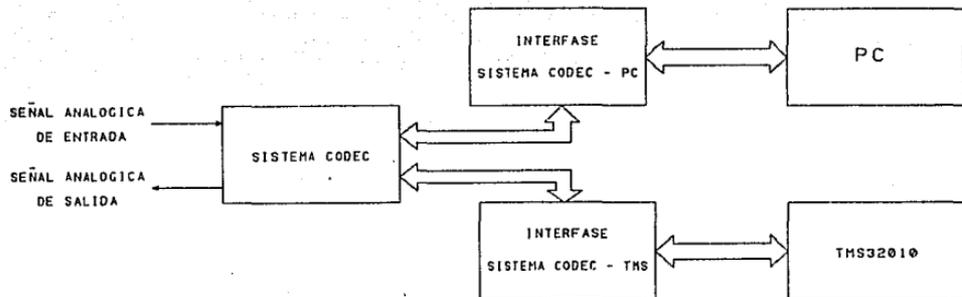


FIG. 1.2 SISTEMA DE CONVERSION A/D - D/A PARA COMPRESION Y TRANSHISION DE SEÑALES DE VOZ

telefónica. En esta parte se requería de un filtro paso banda el cual se realizó utilizando un filtro paso bajas (FPB) en serie con un filtro paso altas (FPA) con un ancho de banda similar al de la línea telefónica. Así mismo, también se utilizó un filtro que se encuentra implantado en el circuito TP3040 el cual se emplea en sistemas PCM (Modulación por Codificación de Pulsos). Por otro lado, una parte esencial de esta etapa, es el circuito CODEC, que es un dispositivo que tiene implantada la ley $\mu 255$ de compansión que permite hacer la conversión, además de ser un circuito diseñado especialmente para trabajar con señales de voz. Esta etapa es la parte medular de toda el sistema de conversión, ya que nos proporciona en su salida digital el dato correspondiente a una señal analógica de entrada, y en su salida analógica una señal proporcional al dato presentado en su entrada digital.

Dentro del sistema CODEC una parte importante es la Base de Tiempo, la cual proporciona todas las señales de control tanto para el sistema CODEC, como para las interfases con la PC y el TMS32010. Dicha base de tiempo tiene la característica de ser muy estable y precisa, ya que es generada por un oscilador, además de proporcionar gran flexibilidad al permitir variar la frecuencia de muestreo de la señal de voz.

La Interfase Sistema CODEC-PC se diseñó de tal manera que el dato ya digitalizado (el cual es proporcionado por el sistema CODEC) se almacene en la memoria de la PC para posteriormente ser procesado. Aquí la transferencia de datos del sistema CODEC hacia la PC se hizo utilizando la técnica de DMA (Acceso Directo a Memoria), permitiendo hacer una transferencia muy rápida de la información y de manera eficiente.

Finalmente, en la Interfase Sistema CODEC-TMS32010 el dato ya

digitalizado (el cual es proporcionado por el sistema CODEC) se puede procesar directamente para alguna aplicación específica tal como : síntesis de voz, el reconocimiento de patrones, etc. En esta parte la interfase tiene la opción de elegir la técnica de transferencia de datos, ya sea por interrupciones o por "polling" (búsqueda).

Es importante mencionar que el principal uso de este sistema es el de poder realizar el procesamiento de voz, por lo que las características que se presentaran corresponden a esta aplicación en particular, sin embargo, con algunas modificaciones en la etapa de filtrado se podría lograr tener una tarjeta aplicable a cualquiera de las otras áreas citadas, lo cual nos permitiría utilizarla en todas aquellas señales que no tienen una variación uniforme o lineal.

Otro factor importante dentro del sistema de conversión , es que el circuito CODEC nos permite obtener una determinada relación señal a ruido (SNR), lo que implica que para obtener estas mismas características para la misma SNR , se hubiese necesitado un convertidor A/D de 12 bits (cuantización uniforme), mientras que con este CODEC solo se necesitan 8 bits (ver figura III.15).

Aun cuando existen en el mercado tarjetas capaces de realizar la misma función, no son muy accesibles en cuanto a costo, mientras que la tarjeta que aquí se propone, tiene entre otros objetivos el de realizar exactamente la misma tarea, pero a un costo por debajo de las comerciales.

Durante el desarrollo del presente trabajo, en el capítulo 1 se presenta un panorama general del origen del Procesamiento Digital de Señales (PDS) y del sistema de conversión dentro del mismo contexto. En el capítulo 2 se mencionan las técnicas tradicionales de conversión tanto en

los sistemas A/D, como en los D/A. En el capítulo 3 se estudian los diferentes procesos de digitalización de voz, así como las técnicas de cuantización más comunes, profundizando principalmente en la cuantización no uniforme y en la técnica de compansión que se utiliza en los circuitos CODECS. En los capítulos 4 y 5 se presenta el diseño y realización tanto de los sistemas de conversión A/D y D/A (sistema CODEC), como de sus interfases con una PC y una arquitectura en base a un TMS32010. También se mencionan dentro de éstos capítulos algunas de las muchas aplicaciones en el contexto de los PDS. Por último en el capítulo 6 se trata sobre las diferentes perspectivas que tiene la tarjeta de conversión y las conclusiones a las que se llegaron.

II. SISTEMAS DE CONVERSION A/D, D/A.

II.1 INTRODUCCION.

Debido a que gran parte de las variables disponibles en el mundo real son analógicas, y ya que son muchos los transductores que poseen una salida eléctrica analógica proporcional a la magnitud medida, como pueden ser las señales de audio, video, presión, temperatura, etc, resulta más conveniente procesar las señales en forma digital. Además, con el gran avance de las computadoras, la información en forma digital se puede procesar, almacenar, comunicar y presentar con facilidad en forma repetitiva y sin errores.

Ahora bien, hay diferentes ventajas que los sistemas digitales tienen sobre los sistemas analógicos, y estas son:

- 1.- Poca sensibilidad hacia los cambios de parámetros digitales.
- 2.- Precisión predeterminada.
- 3.- Son más predecibles y tienen poca variación en su precisión debido a variaciones del medio ambiente (ruido, interferencia, temperatura, tiempo).
- 4.- Facilidad de realización.
- 5.- Mejor rango dinámico.
- 6.- Se pueden hacer réplicas exactas (cada sistema es exactamente similar al original).

Así pues, conviene hacer una interfase para convertir estas señales analógicas a digitales.

Por otra parte, esta señal tratada o procesada digitalmente se puede

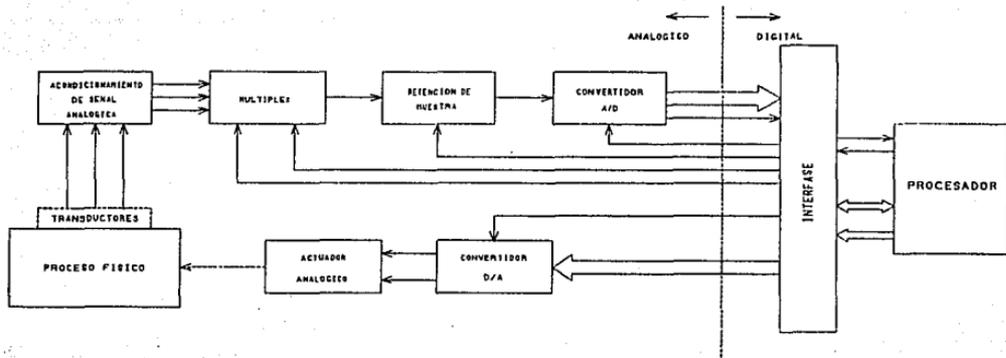


FIG. 11.1 APLICACION DE CONVERTIDORES EN UN SISTEMA DE PROCESAMIENTO DIGITAL

requerir en forma analógica, ya sea para su uso en monitoreo, sobre un controlador o actuador, o una representación analógica sobre un registrador, una pantalla y para diversos procesos de control e instrumentación, lo cual obliga a realizar el proceso inverso, la conversión digital-analógica (D/A). En la figura 11.1 se muestra la aplicación de los convertidores en un sistema de procesamiento digital. Por lo tanto es necesario disponer de elementos capaces de efectuar esta conversión en uno u otro sentido.

11.2 CONVERTIDORES DIGITAL-ANALÓGICO (D/A).

Un convertidor digital analógico (DAC) es un dispositivo que recibe una información digital (D) en forma de una palabra de n bits, y la transforma en una señal analógica (A) proporcional a la magnitud expresada por la entrada digital (D), es decir:

$$A = KD \quad (2.1)$$

Estos convertidores pueden construirse para aceptar la palabra digital en una variedad de códigos tales como el BCD o bien el código binario que es el más común.

11.2.1 Convertidor D/A de Resistencias Ponderadas.

Es el convertidor más simple y consiste de resistencias de valores precisos y proporcionales, siguiendo potencias de dos, en la que los

coeficientes a_i pueden ser ceros o unos y constituyen los n bits de la palabra de entrada. Esta operación puede efectuarse mediante un sumador de n entradas cuyos pesos sean precisamente las n potencias de 2. (ver figura II.2), en donde el voltaje de salida será V_o .

$$I_t = a_1 \frac{V_R}{2R} + a_2 \frac{V_R}{4R} + \dots + a_n \frac{V_R}{2^n R}$$

$$I_t = I_r$$

$$I_r = - \frac{V_o}{r}$$

$$a_1 \frac{V_R}{2R} + a_2 \frac{V_R}{4R} + \dots + a_n \frac{V_R}{2^n R} = - \frac{V_o}{2^n R}$$

$$V_o = \frac{r}{2^n R} + \left[a_1 2^{n-1} + a_2 2^{n-2} + \dots + a_n \right] \quad (2.2)$$

II.2.2 Convertidor D/A de Escalera R-2R.

Este convertidor trabaja básicamente igual al convertidor denominado de resistencias ponderadas, pero evita el rango amplio de valores de resistencias empleando solamente 2 valores (ver figura II.3).

En este caso las corrientes que circulan por las resistencias $2R$ están en progresión geométrica de razón de $1/2$, ya que después de cualquier resistencia R , la impedancia que presenta la red de resistencias es también R , lo que produce una división sucesiva por dos de las corrientes a través de cada una de las resistencias R y $2R$.

$$V_o = -V_R \left[\frac{a_1}{2} + \frac{a_2}{4} + \dots + \frac{a_n}{2^n} \right]$$

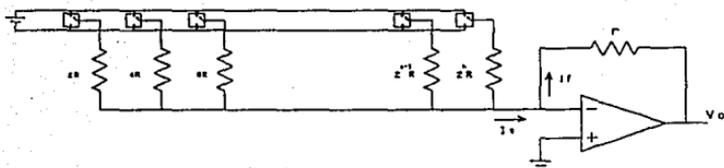


Fig. II.2 Convertidor D/A de resistencias ponderadas.

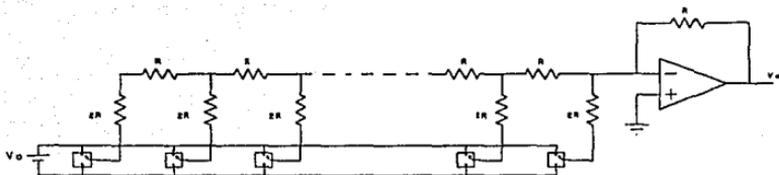


Fig. II.3 Convertidor D/A de escalera R-2R.

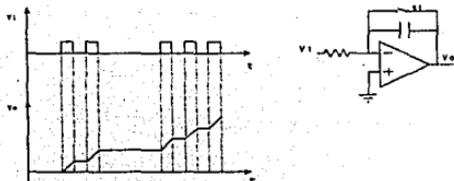


Fig. II.4 Convertidor D/A serie.

$$V_o = - \frac{V_R}{2^n} \left[a_1 2^{n-1} + a_2 2^{n-2} + \dots + a_n \right] \quad (2.3)$$

II.2.3 Convertidor de D/A Serie.

Cuando se requiere usar una entrada consistente en una sucesión de pulsos en el tiempo, se requiere un DAC serie.

Este DAC recibe una sucesión de pulsos en el tiempo donde cada posición tiene un peso.

Si analizamos el circuito mostrado en la figura II.4 vemos que:

$$V_o = - \frac{N}{R_1 C} \int V_1 dt = - \frac{A t}{R_1 C} N \quad (2.4)$$

donde

N : es el número de pulsos en el intervalo de integración.

A : es la amplitud de los pulsos.

t : intervalo de integración.

La constante de proporcionalidad entre el número de pulsos que entran (N) y la salida analógica V_o es:

$$K = \frac{At}{RC} \quad (2.5)$$

II.2.4 Parámetros Utilizados en los Convertidores D/A.

A continuación se presentan algunas definiciones que nos permiten poder cuantificar la calidad del DAC, éstas son:

- **Precisión Absoluta.**- Se refiere al porcentaje de error que presenta el DAC, a plena escala.
- **Resolución.**- Se refiere a la longitud de la palabra que puede manejar el DAC, es decir el número de bits cuyo bit menos significativo (LSB) puede distinguirse a la salida analógica.
- **Linealidad Diferencial.**- Dos códigos digitales adyacentes cualesquiera pueden dar como resultado valores medidos de salida que tengan exactamente un LSB.
- **Linealidad absoluta.**- Es el grado de relación lineal entre las entradas binarias de salida analógica producidas. Esta cantidad se expresa generalmente en porcentaje.
- **Sensitividad a la fuente de alimentación.**- Se expresa normalmente en función del cambio de porcentaje de escala completa del valor de la salida analógica, para un cambio de 1% de corriente continua (c.c) en la fuente de alimentación.
- **Coefficiente de temperatura.**- Es la medida del efecto del cambio de temperatura en la exactitud de la salida.
- **Tiempo de respuesta.**- Expresa el intervalo del tiempo requerido para representar una cantidad digital en un voltaje analógico.

11.3 CONVERTIDORES ANALOGICO-DIGITAL (A/D).

El proceso de digitalización de una señal analógica, es un proceso de cuantización, y consiste en dividir el rango de la señal continua entre varios valores discretos. A la diferencia entre dos valores discretos contiguos cualesquiera de la señal se le denomina "cuanto". Si el valor

del cuanto en todo el rango de la señal es constante, el proceso de cuantización es uniforme o lineal (ver capítulo 3).

Un convertidor analógico digital (A/D) es un dispositivo que recibe una señal de entrada y la transforma en una palabra digital.

Los convertidores A/D se pueden clasificar de la siguiente manera :

II.3.1 Convertidores A/D que Emplean Técnicas de Carga y Descarga de un Capacitor.

II.3.1.1 Convertidor de Integración simple.

Este convertidor también llamado convertidor A/D modulador de ancho de pulso o de rampa simple, se deriva del hecho de que el nivel de la señal analógica es convertida en un pulso cuya duración de tiempo es una función del valor de la señal de entrada. El ancho del pulso es convertido a un formato digital, al contar el número de ciclos de la frecuencia de referencia que ocurren entre el inicio y el fin del pulso. (ver figura II.5b).

El funcionamiento de este convertidor es el siguiente, (ver figura II.5a).

El interruptor S_1 permanece cerrado hasta que se inicia la conversión, en el instante que S_1 se abre, el voltaje del capacitor aumenta en forma lineal. Cuando el voltaje del capacitor excede el valor de entrada (V_{in}) el comparador cambia de estado, evitando que el contador continúe su operación, dando por terminada la conversión. En este momento el contador tendrá almacenada la información de entrada V_{in} convertida en digital.

Este tipo de convertidor tiene una rapidez de conversión R_c pequeña,

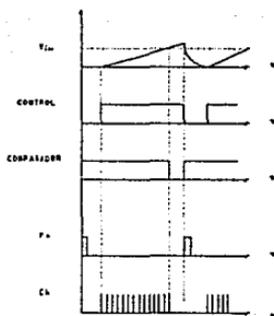
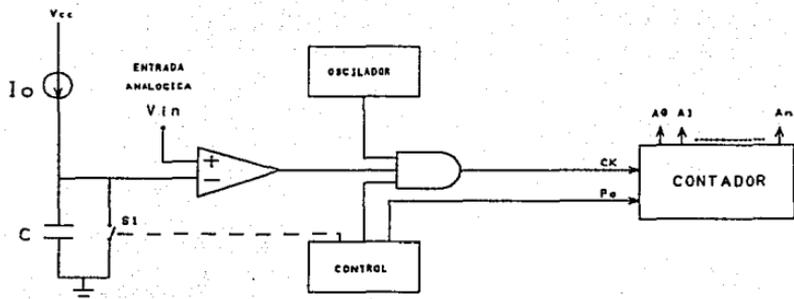


Fig. II.5 Convertidor de integración simple.

pues se necesita $2^n - 1$ pulsos de reloj para completar una palabra de n bits con n unos en un contador binario, por lo que:

$$R_c = - \frac{R}{2^n - 1} \text{ [palabras/seg]} \quad (2.6)$$

$$V_c = - \frac{1}{C} \int i dt = - \frac{I_c}{C} t \quad (2.7)$$

Si V_{in} es igual al valor máximo que el convertidor debe manejar, entonces el tiempo de conversión máximo (t_{CM}) es :

$$t_{CM} = \frac{V_{inM} C}{I_0} \quad (2.8)$$

y la frecuencia de reloj es:

$$f_R = \frac{2^n - 1}{t_{CM}} = \frac{2^n - 1}{V_{inM} C} I_0 \quad (2.9)$$

En otras palabras, antes de iniciar cada ciclo de integración el contador es puesto a cero (P_0) y la magnitud alcanzada por el contador al final de cada ciclo será:

$$N = f T \quad (2.10)$$

sustituyendo la E_c (2.8) en (2.10)

$$N = \frac{f C V_{in}}{I_0} \quad (2.11)$$

donde:

N : palabra digital, es función de V_{in}

f: frecuencia del oscilador

t: tiempo empleado por la rampa en alcanzar el valor de voltaje de V_{in}

II.3.1.2 Convertidor de doble integración.

Este convertidor minimiza grandemente las inexactitudes propias del convertidor de integración simple. En este convertidor se genera un pulso cuya duración es proporcional al voltaje que se requiere convertir, haciendo una comparación entre dos integraciones.

En la figura II.6 se muestra el diagrama de bloques de este convertidor.

La señal de inicio (P_0) coloca a los contadores a cero, logrando que el bloque de control conecte la señal analógica de entrada al integrador. Si V_{in} se mantiene aproximadamente constante, la salida del integrador (V_c) será una rampa negativa. Mientras esta salida sea negativa, la salida de comparador será un voltaje alto que habilita el paso del reloj al contador. Cuando el bloque de control recibe la señal de "fin de cuenta" (L) del contador, el bloque de control conectará la señal $-V_{ref}$ al integrador y pondrá en ceros otra vez al contador, la cual proporcionará a la salida del integrador una rampa de pendiente positiva que alcanzará el valor de cero volts, formando así un voltaje bajo a la salida del comparador que impedirá la entrada de la señal de reloj al contador deteniendo con esto la cuenta. (ver figura II.6a).

Como t_1 y V_{ref} son constantes, t_2 resulta directamente proporcional a V_{in} (ver figura II.6b). Las ventajas de este convertidor son su precisión

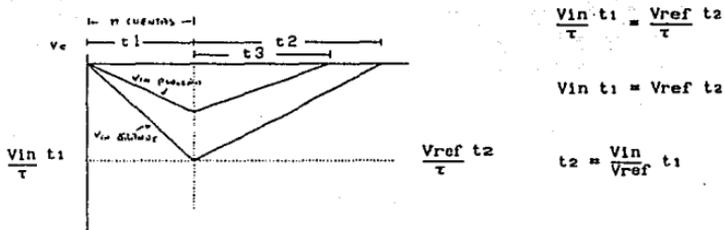
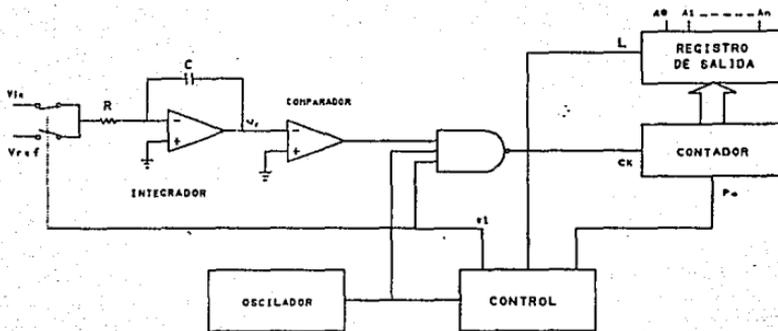


Fig. II.8 Convertidor de doble integración.

y su costo reducido, lo mismo que su inmunidad a efectos de temperatura o variaciones en los parámetros del integrador. Entre sus desventajas tenemos que tiene un tiempo de conversión grande.

11.3.1.3. Convertidor de voltaje a frecuencia.

Un convertidor de voltaje a frecuencia (V/F) consta esencialmente de una fuente de corriente de precisión, conmutada por un comparador (figura 11.7). Estos pulsos son generados por un circuito monoestable que es activado cuando el voltaje de entrada es mayor que la del condensador cargado por los pulsos de corriente.

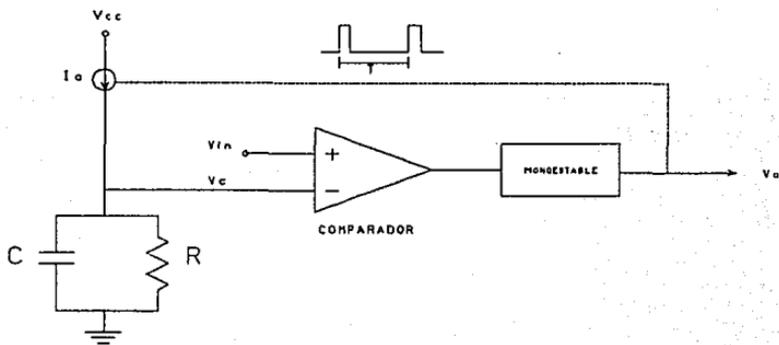


Fig. 11.7 Convertidor de F/V.

De esta forma, cuando $V_{in} > V_c$ el comparador permite que se generen pulsos de amplitud constante t , que producen la carga del condensador. Cada nuevo pulso de corriente I_0 aumenta el potencial del condensador, si $V_c > V_{in}$, interrumpe la generación de nuevos pulsos hasta que la corriente de descarga producida por R , haga que $V_c < V_{in}$.

Cuando el convertidor de V/F alcanza el régimen permanente, V_c oscilará alrededor de V_{in} , y la corriente media de descarga I_d será :

$$I_d = \frac{V_c}{R} = \frac{V_{in}}{R} = f t I_0 \quad (2.12)$$

en donde

$$f = \frac{V_{in}}{t I_0 R} = K V_{in} \quad (2.13)$$

Obteniéndose a la salida una sucesión de pulsos de amplitud t y frecuencia " f ", proporcional al voltaje de entrada.

II.3.2 Convertidores A/D por Comparación de Voltaje.

II.3.2.1 Convertidor de rampa discreta.

Este convertidor es muy parecido al convertidor de integración simple (rampa simple), aquí para generar una rampa, se usa un DAC, quedando el convertidor con esto dividido en escalones (ver figura II.8).

Una vez que se aplica la señal: V_{in} a la entrada del comparador, la señal de RESET pone en ceros al contador, cuyas salidas alimentan al convertidor D/A, la salida del comparador tendrá un voltaje alto, que

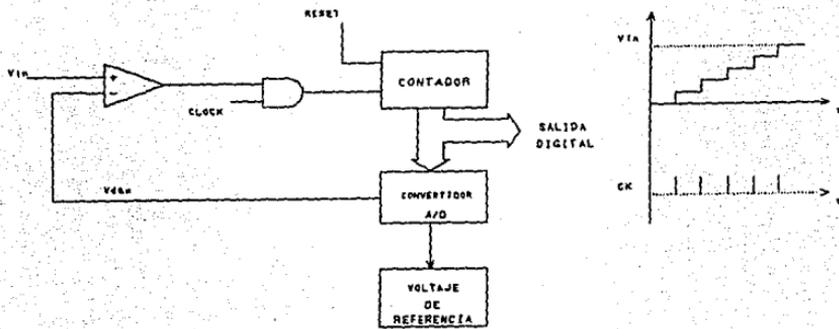


Fig. 11.8 Convertidor de rampa discreta.

habilita el paso de la señal de reloj al contador. Conforme este aumenta su cuenta, el voltaje a la salida del convertidor A/D (V_{dac}) también lo hará, hasta que alcance el voltaje de entrada V_{in} , para que el conteo siempre parta de cero.

11.3.2.2 Convertidor de balance continuo.

El objetivo de este convertidor es reducir el tiempo de conversión. Se emplea para ello, un contador que tenga la capacidad de incrementar o decrementar su cuenta.

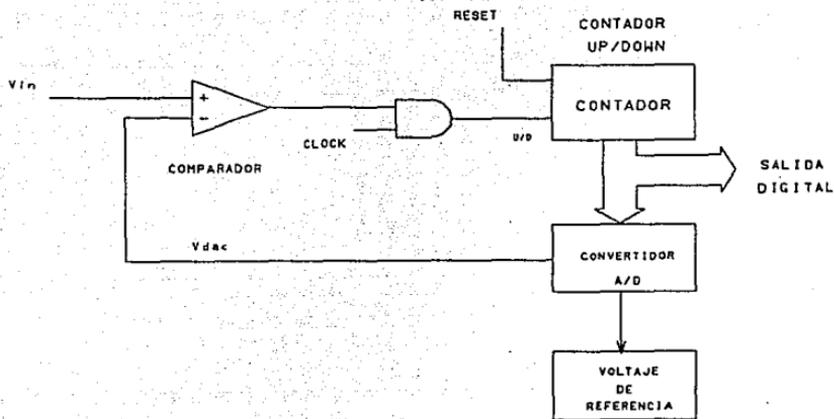


Fig. II.9 Convertidor de balance continuo.

En la figura II.9 al aplicar V_{in} al comparador, si se supone inicialmente que el voltaje del convertidor (V_{dac}) es igual a cero, entonces a su salida aparecerá un nivel alto, que si se aplica al contador, se logra que este cuente en forma ascendente. La salida digital del contador es aplicada al convertidor D/A, dando a la salida un voltaje V_{dac} , cuando el conteo alcance un valor tal que haga que V_{dac} sea mayor que V_{in} , la salida del comparador cambia a un nivel bajo, logrando con esto que se invierta la dirección del conteo del contador, es decir en forma descendente, esto hace que nuevamente V_{dac} sea menor que V_{in} y por

lo tanto el conteo de nuevo sea ascendente y al llegar a ese punto se establece un balance alrededor de V_{in} con una oscilación de *mas-menos* una cuenta en el contador; de esta forma el valor que aparece en el contador corresponde a la señal analógica.

II.3.2.3 Convertidor de aproximaciones sucesivas.

La operación de este convertidor, que es uno de los más usados, se basa en n comparaciones sucesivas entre la entrada analógica V_{in} , y el voltaje de realimentación V_{dac} (ver figura II.10).

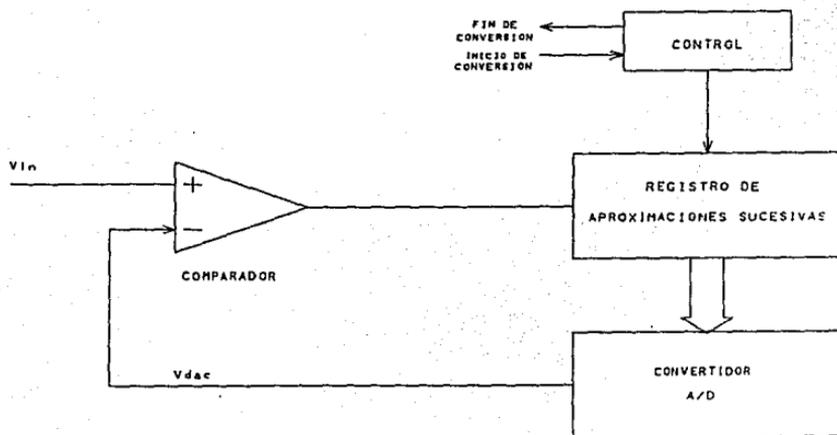


Fig. II.10 Convertidor de aproximaciones sucesivas.

Nótese que el tiempo de conversión de estos convertidores está determinado por el número de bits del convertidor. En éste sistema el elemento principal es el registro de aproximaciones sucesivas (RAS) cuya función es la siguiente:

al inicio del ciclo de conversión, el RAS en el primer pulso de reloj presenta el patrón binario 1000...00 al convertidor D/A; dependiendo de la respuesta del comparador, si el voltaje de salida del comparador es alto (esto es $V_{dac} < V_{in}$) se confirma a b_n como "1" lógico y prueba b_{n-1} , presentando el patrón binario 1100...00 al convertidor D/A. Si la salida del comparador al patrón binario 1000...00 es un voltaje bajo ($V_{dac} > V_{in}$), será b_n un "0" lógico y prueba a b_{n-1} , presentando al patrón binario 0100...00 al convertidor D/A. Este ciclo se repite con todos los bits, en el orden $b_n, b_{n-1}, b_{n-2}, \dots, b_1, b_0$. Haciéndolos "1" lógico para la prueba y confirmándolos como "1" o cambiándolos a "0" según la respuesta del comparador, solo se necesita un pulso de reloj para cada bit. Una vez barrido todos los bits, el convertidor manda la señal *fin de conversión*.

El convertidor A/D de aproximaciones sucesivas tiene la gran ventaja de tener una gran velocidad de ejecución con excelente resolución.

II.3.2.4 Convertidor paralelo.

Es el más rápido y más simple de los convertidores A/D. Ese tipo de convertidor consiste básicamente de un grupo de comparadores analógicos que identifican el nivel de la señal analógica V_{in} en niveles discretos determinados por voltajes de referencia, según se muestra en la figura II.11a. Si las salidas de los comparadores se decodifican de tal manera

que se tenga una representación binaria, a la salida del convertidor de código (codificador) se tendrá un dato proporcional a la señal analógica de entrada. Esta decodificación se muestra en la figura II.b valiéndose para ello de un convertidor de código.

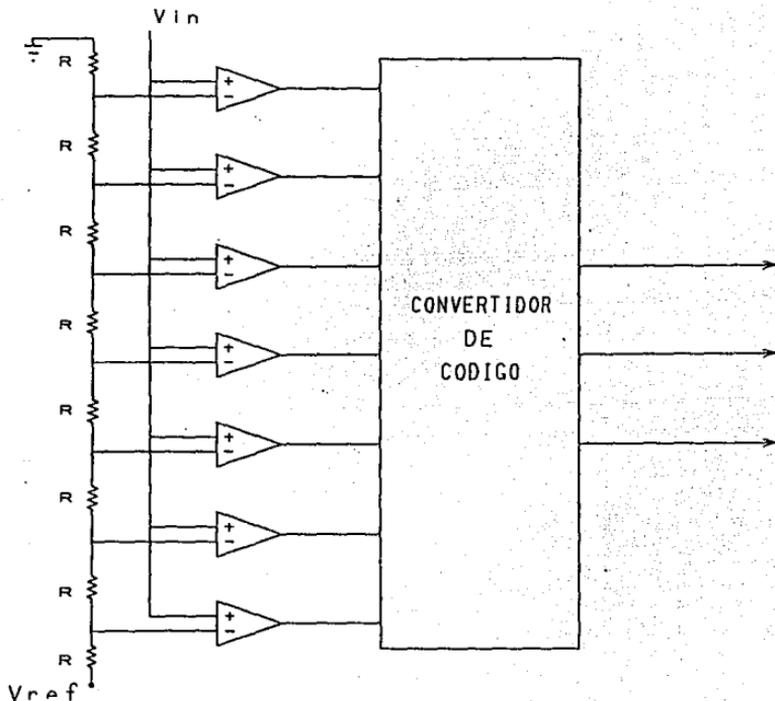


Fig. II.11a Convertidor paralelo

A1	A2	A3	A _{n-2}	A _{n-1}
0	0	0		0	0
1	0	0		0	0
1	1	0		0	0
⋮					⋮
1	1	1		1	0
1	1	1		1	1



A1	A2	A3	A _{n-2}	A _{n-1}
0	0	0		0	0
0	0	0		0	1
0	0	0		1	0
⋮					⋮
1	1	1		1	0
1	1	1	1	1

Fig. II.11b Decodificación de un convertidor paralelo.

11.3.3 Ventajas y Desventajas de los Diferentes Convertidores.

Convertidores basados en técnicas de integración

Ventajas

- a. Excelente linealidad
- b. No requieren técnicas de muestreo.
- c. Bajo costo y simplicidad.
- d. La sección contadora puede estar alejada del circuito de entrada, permitiendo enviar información por dos cables.
- e. La respuesta puede hacerse independiente de los elementos que la constituyen.
- f. Buen rechazo a señales externas.

Desventajas

- a. El tiempo de conversión es largo.
- b. Normalmente funciona con una polaridad pero con circuitos más complejos tienen que operar en forma bipolar.

Convertidores de balance continuo

Ventajas

- a. No requieren del uso de muestreo y retención.
- b. La salida permanentemente esta relacionada con la entrada.
- c. Conversión muy rápida para variaciones pequeñas de la señal analógica.

Desventajas

- a. Tiempos de conversión largos para cambios en la entrada.

- b. No puede obtenerse la información de salida en serie.

Convertidores de aproximaciones sucesivas

Ventajas.

- a. Tiempo de conversión muy alto.
- b. Alta resolución .
- c. Información de salida en serie o paralelo.
- d. Buena linealidad.
- e. La exactitud no depende de la estabilidad del reloj.

Desventajas

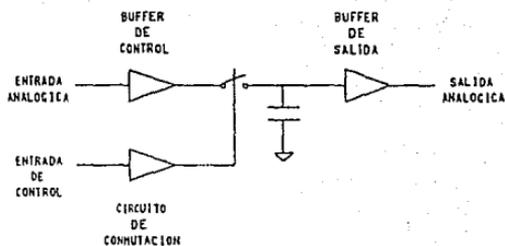
- a. Se requiere de circuitos de muestreo y retención.
- b. Complejidad en el circuito.

II.3.4 Parámetros Utilizados en los Convertidores A/D.

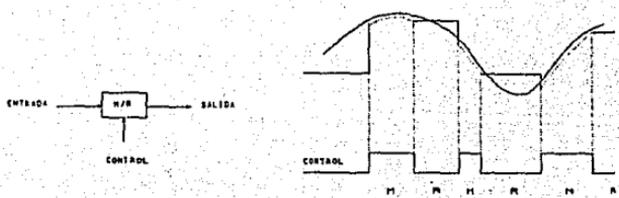
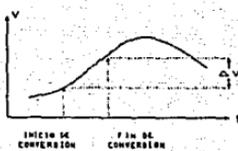
Cuando se desea digitalizar una señal que cambia rápidamente de valor, puede ocurrir un cambio significativo en el valor de dicha señal antes de que termine la conversión, siendo ésta una fuente de error que puede generar otros errores.

Para reducir los efectos de este error se emplea un circuito denominado " *Sample and hold* " (muestra y retén). Se trata de un interruptor electrónico de alta velocidad usado para tomar muestras de la señal y un circuito capacitivo seguido de un amplificador de ganancia unitaria, usado para almacenar el valor muestreado de la señal. (ver figura II.12). Dos parámetros de este circuito son muy importantes:

Tiempo de Adquisición.- Es el tiempo necesario para cerrar el



(a)



(b)

Fig. II.12 Muestreo y Reten. (a) circuito de muestreo y reten. (b) señal de salida de un muestreador-retenedor.

Interruptor y cargar el voltaje de la señal de entrada en el capacitor.

Tiempo de Apertura.- Es el tiempo que se requiere para que el interruptor se abra completamente.

A continuación se presentan algunos parámetros usados en los convertidores A/D :

Tiempo de Conversión.- Es el tiempo que tarda un convertidor en realizar una conversión completa, desde el instante que se aplica una señal de entrada, hasta que aparece la salida.

Resolución.- Se refiere al valor mínimo de voltaje que se puede convertir, se acostumbra a asociarle la letra Q y se define como:

$$Q = \frac{\text{Rango de escala completa}}{2^n}$$

Monotonicidad.- Un convertidor A/D o D/A es monótonico si su salida se incrementa o permanece constante cuando la entrada se incrementa para todo rango de escala (Fig. II.13).

Linealidad.- Es el grado de relación lineal entre el voltaje de entrada requerida para producir un cambio unitario en la salida digital (figura II.14).

Exactitud.- Es la medida del error esperado en la representación de un voltaje analógico en uno digital.

Rango del Voltaje de Entrada.- Es el rango del voltaje de entrada que el convertidor puede convertir.

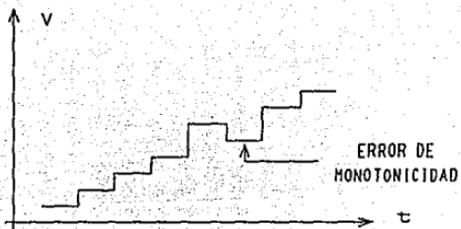


Fig. II.13 Error de monotonidad.

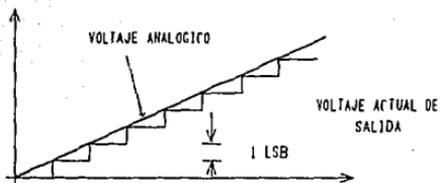


Fig. II.14 Relación lineal entrada/salida.

III. DIGITALIZACION DE VOZ.

III.1 INTRODUCCION.

Las técnicas de digitalización de las señales de voz las podemos clasificar en dos: aquellas formas de onda analógicas codificadas digitalmente lo más fielmente posible a la señal de entrada, y las formas de onda procesadas, codificadas solo en algunos aspectos significativos, como en procesos de habla y escucha. Dentro de la primera categoría, el problema fundamental es el de los convertidores A/D y D/A al estar restringidos a las características requeridas.

El primer paso en la digitalización de una señal analógica es establecer el tiempo discreto con el que se va a muestrear la señal de entrada, para posteriormente pasar a la etapa de cuantización, como se muestra en la figura III.1.

Durante el presente capítulo se realizará un seguimiento del proceso para digitalizar la voz, de acuerdo a la secuencia mostrada en la figura III.1.

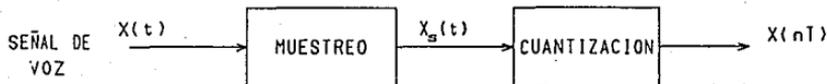


Fig. III.1 Proceso de Cuantización de una señal analógica.

III.2 MUESTREO.

Las señales que portan información deben ser accesibles para su procesamiento digital, por lo que las condiciones necesarias para convertir una señal analógica en discreta, sin perder información, son que deben primero ser muestreadas en forma periódica y posteriormente convertidas a muestras de amplitud discreta por medio de la cuantización.

III.2.1 Teorema del Muestreo.

La representación de señales continuas en el tiempo a través de sus muestras se conoce como el teorema de muestreo de SHANNON (NYQUIST), el cual se enuncia :

Séa $X(t)$ una señal a banda limitada con $X(\omega)=0$ para $|\omega|>W_m$.

Entonces $X(t)$ esta determinada por sus muestras $X(nT)$,

$$n = 0, \pm 1, \pm 2, \pm 3 \quad (3.1)$$

$$\text{Si } W_s > 2W_m$$

$$\text{donde } W_s = \frac{2\pi}{T}$$

El principio de muestreo puede ser explicado usando un interruptor el cual cambia periódicamente de posición entre los dos contactos a una velocidad de $f_s = 1/T$ Hz permaneciendo en el contacto de entrada por "d" segundos y en el contacto de tierra el resto del periodo de cada muestra. La salida $Y(t)$ de un muestreador, puede ser representado como (Ver figura III.2) :

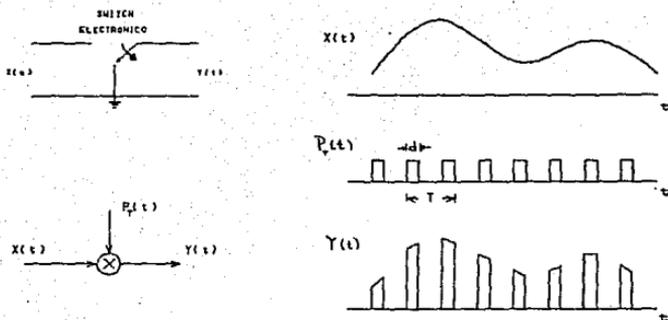


Fig. III.2 Muestreo de una señal analógica.

$$Y(t) = X(t) Pr(T) \quad (3.2)$$

Si $d \rightarrow 0$, $Pr(t)$ se aproxima a un tren de impulsos.

Es importante que "d" sea pequeño para mantener una potencia promedio constante.

III.2.2 Muestreo con un Tren de Impulsos.

Sea el tren de impulsos, o función de muestreo $\delta_T(t)$ donde :

$$\delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT) \quad (3.3)$$

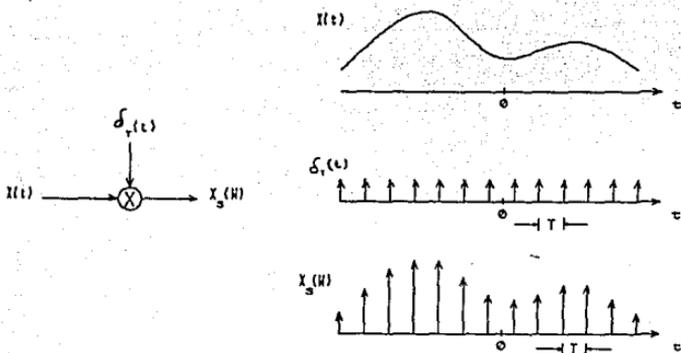


Fig. III.3 Muestreo de una señal analógica con un tren de impulsos

T es el periodo de muestreo, y la frecuencia fundamental de $\delta(t)$ es $\omega_s = 2\pi/T$, o frecuencia de muestreo. En el dominio de tiempo tenemos: (Ver figura III.3)

$$x_s(t) = x(t) \delta_T(t) \quad (3.4)$$

$x_s(t)$ es un tren de impulsos con las amplitudes de los impulsos iguales a las muestras de $x(t)$ en los intervalos espaciados por T , eso es:

$$x_s(t) = \sum_{n=-\infty}^{\infty} x(nT) \delta(t-nT) \quad (3.5)$$

sabemos que por el teorema de modulación

$$X_s(t) = \frac{1}{2} \mathcal{F} \{ X(t) \} * \mathcal{F} \{ \delta_T(t) \}$$

además

$$\mathcal{F} \{ X_s(t) \} = \int_{-\infty}^{\infty} e^{-j2\pi t} X_s(t) dt = X_s(W)$$

$$\mathcal{F} \{ X(t) \} = X(W)$$

$$\mathcal{F} \{ \delta_T(t) \} = W_s \delta_{W_s}(W) = \frac{2}{T} \sum_{n=-\infty}^{\infty} \delta(W-nW_s)$$

$$X_s(W) = \frac{1}{T} \sum_{n=-\infty}^{\infty} X(W-nW_s) \quad (3.6)$$

$X_s(W)$ es una función periódica en el dominio de la frecuencia compuesta por réplicas desplazadas de $X(W)$ ponderadas por $1/T$.

En la figura III.4a se muestra el caso en que $W_s > 2W_m$, aquí la representación de la transformada de Fourier no se traslapa en la banda base. La figura III.4b muestra una situación en la que $W_s < 2W_m$, en este caso la representación de su transformada de Fourier se traslapa en parte en su banda base, a este efecto se le conoce como efecto de *Aliasing* (o efecto de *traslape*). Este traslape puede ser evitado si se cumple el teorema del muestreo.

Si se tiene $W_s > 2W_m$, $X(t)$ puede ser recuperado de $X_s(t)$ con un filtro paso bajas (FPB).

Por lo tanto se puede concluir que dada una señal $X(t)$, ésta se puede reconstruir a través de sus muestras generando un tren de impulsos periódico cuyas amplitudes son los valores de las muestras sucesivas. Este tren de impulsos será procesado a través de un filtro ideal paso bajas con ganancia T y frecuencia de corte más grande que W_m y menor que

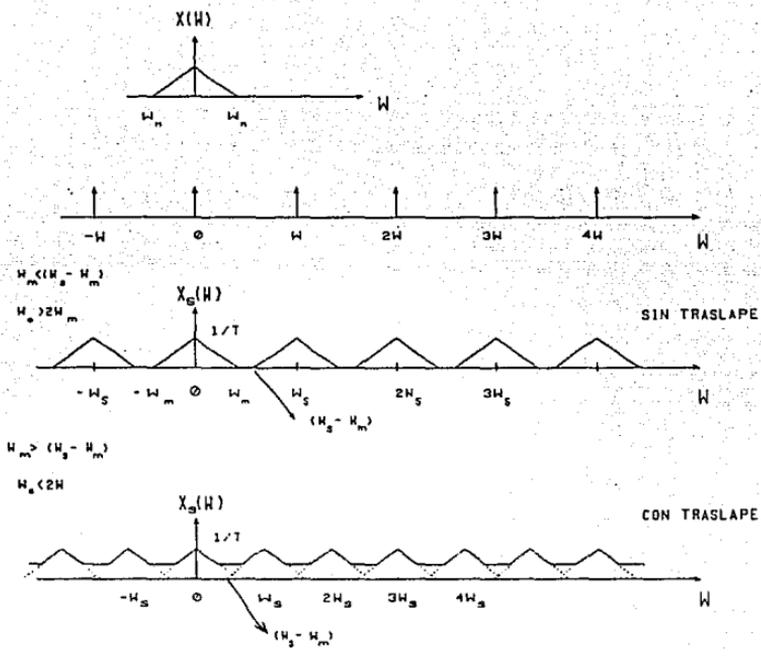


Fig. III.4 Interpretación del muestreo en el dominio de la frecuencia
 a) Para $W_s > 2 W_m$ b) Para $W_s < 2 W_m$.

$(W_s - W_m)$. La salida resultante será exactamente igual a $X(t)$ (Ver figura III.5).

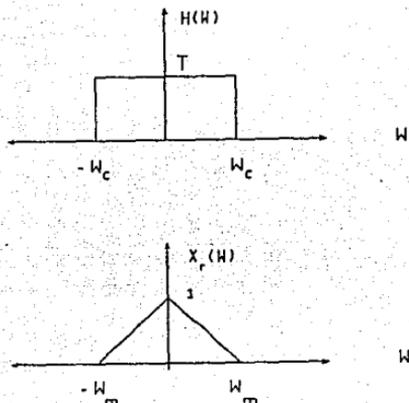


Fig. III.5 Reconstrucción de una señal a través de un filtro paso bajas ideal.

III.3 CUANTIZACION.

Las señales como las de voz o video tienen un rango de amplitud continua. Cuando ésta amplitud es transmitida sobre un canal ruidoso, el receptor no puede discernir la secuencia exacta de valores transmitidos. Sin embargo, el efecto del ruido puede ser minimizado mediante la representación de la muestra a través de un número finito de niveles

discretos. Este esquema de señalización discreto es conocido como Modulación por Amplitud de Pulsos (PAM).

Una vez que la señal ha sido muestreada de acuerdo al Teorema del Muestreo, a la salida se obtendrá una señal PAM discreta en tiempo, cuya amplitud podrá tomar valores dependiendo del voltaje de la información de la señal en el instante de muestreo.

La cuantización consiste en aproximar la amplitud de los pulsos de una señal PAM a niveles discretos específicos. Esto es, mientras que con un proceso de muestreo se discretiza la señal en tiempo, con el proceso de cuantización se discretiza la señal en amplitud. La secuencia de niveles puede ser transmitida, usando uno o varios esquemas de señalización digital. Un ejemplo de un cuantizador se muestra en la figura III.6.

En esta figura, la entrada del cuantizador es un proceso aleatorio $X(t)$ que representa la salida de una fuente de información analógica. La forma de onda aleatoria $X(t)$ es muestreada a una razón apropiada y los valores muestreados $X(kT)$ son convertidos en Q niveles m_1, m_2, \dots, m_Q , previamente determinados, de manera que:

$$X_q(kT) = m_i \quad (3.7)$$

para $i = 1, 2, \dots, Q$

si $X_{i-1} \leq X(kT) < X_i$ $X_0 = -\infty, X_Q = +\infty$

La salida del cuantizador es la secuencia de niveles $X_q(t)$, que se muestra en la figura III.6, donde:

$$X_q(t) = X_q(kT), \quad kT \leq t < (k+1)T \quad (3.8)$$

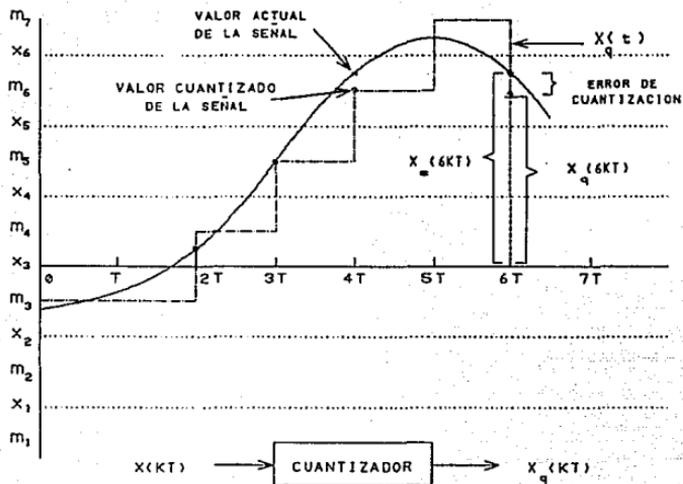


Fig. III.6 Operación de cuantización : m_1, m_2, \dots, m_7 son los siete niveles de salida del cuantizador.

Observamos que la señal cuantizada es una buena aproximación a la señal original. La cantidad de la aproximación puede ser dada de acuerdo a X_i 's y m_i 's, que son los parámetros que nos permiten optimizar el proceso.

Una medida que es comunmente usada para evaluar los efectos del ruido en un esquema de cuantización es :

$$\frac{S_q}{N_q} \approx \frac{E \{ [X_q(kT)]^2 \}}{E \{ [X(kT) - X_q(kT)]^2 \}} \quad (3.9)$$

La ecuación anterior nos representa la relación de la energía de la señal de salida del cuantizador a la energía de la señal de ruido de una fuente de transmisión para un proceso aleatorio $X(t)$.

Otra medida que también nos permite evaluar el efecto del ruido se muestra en la siguiente ecuación

$$e(kT) = X(kT) - X_q(kT) \quad (3.10)$$

donde $e(kT)$ representa el error de cuantización .

En lo sucesivo, X nos representará $X(kT)$ y X_q a $X_q(kT)$.

Por lo general, se puede considerar que el conjunto de muestras $X(kT)$ pertenece a un conjunto finito de amplitudes tal que :

$$|X(kT)| \leq X_{max} \quad (3.11)$$

Así también en esta parte se va a considerar que una señal de voz tiene una densidad de probabilidad de Laplace (figura III.7). En esta misma figura se muestra una comparación entre las densidades de probabilidad Gama , Laplace y de una señal de voz, normalizada de tal manera que la media sea cero y la variancia σ_x^2 sea igual a uno, en la figura III.7 se observa que la función de densidad Gama es una mejor

aproximación a la de Laplace.

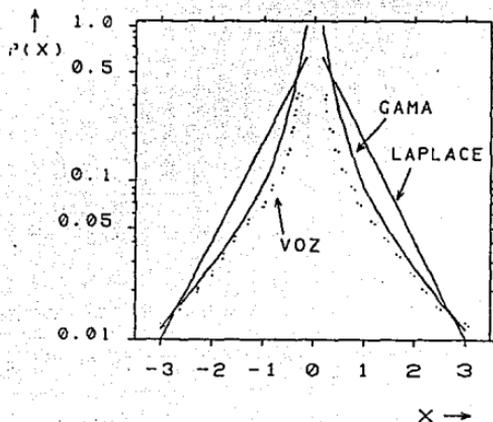


Fig. III.7 Voz real y densidades de probabilidad Gama y Laplace.

Por otro lado se puede demostrar que solo el 0.35% de las muestras de la señal de voz están fuera del intervalo

$$-4 \sigma_x < X(kT) \leq 4 \sigma_x \quad (3.12)$$

por lo que se puede asumir que el intervalo pico a pico de la señal de voz es proporcional a su desviación estándar.

III.3.1. Cuantización Uniforme.

En esta técnica de cuantización, el rango de la variable aleatoria continua X está dividido en Q intervalos de igual longitud, es decir, que tiene intervalos de tamaño (Δ) uniforme. Si el valor de X disminuye en el i -ésimo intervalo de cuantización, entonces el valor cuantizado de X es tomado por el punto medio del intervalo. (figura III.8).

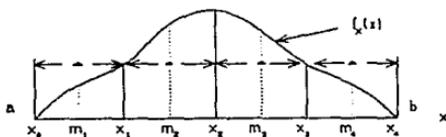


Fig. III.8 Ejemplo de un cuantizador uniforme con un tamaño de intervalo = Δ y una $Q = 4$.

Si a y b son los valores máximo y mínimo de los valores de X , respectivamente, entonces el tamaño del intervalo Δ es :

$$\Delta = \frac{b - a}{Q} \quad (3.13)$$

La salida cuantizada X_q es representada como

$$X_q = m_i \quad \text{si} \quad X_{i-1} \leq X \leq X_i$$

donde

$$X_i = a + i\Delta \quad (3.14)$$

y

$$m_i = \frac{X_{i-1} + X_i}{2} \quad , \quad i = 1, 2, 3, \dots, Q \quad (3.15)$$

La energía de la señal del ruido de cuantización, N_q , para un cuantizador uniforme es :

$$\begin{aligned} N_q &= E \{ (X - X_q)^2 \} \\ &= \int_a^b (X - X_q)^2 f(X) dX \\ &= \sum_{i=1}^Q \int_{X_{i-1}}^{X_i} (X - m_i)^2 f(X) dX \end{aligned} \quad (3.16)$$

donde $X_i = a + i\Delta$ y $m_i = a + i\Delta - \Delta/2$.

La energía de la señal X_q a la salida del cuantizador será :

$$\begin{aligned} S_q &= E \{ (X_q)^2 \} \\ &= \sum_{i=1}^Q (m_i)^2 \int_{X_{i-1}}^{X_i} f(X) dX \end{aligned} \quad (3.17)$$

por lo que la relación S_q/N_q nos dará una medida de la fidelidad en un cuantizador uniforme.

La ecuación anterior nos indica que la fidelidad de un cuantizador se incrementa a medida que aumenta el número de niveles Q del cuantizador, si el número de niveles de cuantización es grande, entonces la salida X_q puede ser lo más cercano a la entrada X . El número de intervalos Q es determinado dependiendo de la fidelidad de la transmisión deseada.

Por otro lado se tiene que la representación del número de niveles de cuantización en función de la longitud B (# de bits) de una palabra se obtiene a partir de:

$$Q = 2^B \quad (3.18)$$

También se puede demostrar que la relación señal a ruido (SNR) para un cuantizador uniforme, la cual está expresado en dB está dada por

$$\text{SNR(dB)} = 6B + 4.77 - 20 \log_{10} \left[\frac{X_{\max}}{\sigma_x} \right] \quad (3.19)$$

de donde evaluando la expresión anterior con la Ec.(3.12) se obtiene que

$$\text{SNR(dB)} = 6B - 7.2 \quad (3.20)$$

en otras palabras, por cada bit que se agregue (lo cual duplica el número de niveles de cuantización) se obtiene una ganancia de 6 dB.

III.3.2 Cuantización no uniforme.

Un problema que se tiene al utilizar un cuantizador uniforme , es que una señal de voz cambia con el tiempo y su variancia σ_x^2 puede variar de un segmento de voz a otro, por lo que si el cuantizador está diseñado para manejar señales de gran magnitud, el intervalo de cuantización será grande, dando como resultado un error de cuantización en las amplitudes pequeñas.

Una forma de corregir este problema es utilizar un intervalo de cuantización que se incremente de la misma forma que la amplitud de la señal, esto es, utilizar un intervalo de cuantización variable. A esta técnica se le conoce como cuantización no uniforme.

Un ejemplo del cuantizador no uniforme se muestra en la figura III.9, donde la entrada del cuantizador Gaussiano aleatorio y la salida del cuantizador esta determinada por:

$$X_q = m_i \quad (3.21)$$

si $X_{i-1} < X \leq X_i$, $i = 1, 2, \dots, Q$

$$X_0 = -\infty \quad , \quad X_Q = \infty$$

El tamaño del incremento $\Delta_i = X_i - X_{i-1}$ es variable. El punto final del cuantizador X_i y los niveles de salida m_i son maximizados con una relación señal a ruido diferente.

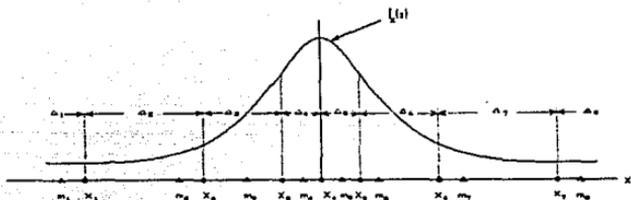


Fig. III.9 Un cuantizador no uniforme para una variable gaussiana.

$$X_0 = -\infty, \quad X_Q = 8 \quad \text{y} \quad \Delta_i = \Delta_{Q+1-i} \quad (i = 1, 2, 3, 4).$$

En la práctica un cuantizador no uniforme es realizado mediante la compresión de un cuantizador uniforme.

III.3.2.1 Compansión.

Es bien conocido que en cualquier sistema de adquisición de señales, el proceso de conversión A/D introduce ruido de cuantización, por lo que la digitalización de un código de palabra es una representación binaria truncada de la muestra analógica. Este efecto de truncamiento es más pronunciado para señales pequeñas cuando se tiene un esquema de cuantización uniforme, lo cual es indeseable para transmisión de voz, debido a que la mayor cantidad de información reside en las amplitudes pequeñas, por otro lado, encontramos personas que tienen diferente potencia de voz y distintos timbres de voz, o que algunas personas gritan y otras hablan muy suave, entonces si se tuviese una cuantización uniforme, la reproducción de todos los tipos de voces de personas no sería de modo alguno posible. Entonces una forma de poder reproducir todos estos tipos de voces con una buena calidad, consiste en tomar intervalos de cuantización no uniforme, ajustando el tamaño del intervalo de cuantización de tal forma que sea proporcional al nivel de la señal de entrada. En este caso el intervalo de cuantización es más pequeño para señales de amplitud pequeña y grandes para señales de amplitud grande, por consiguiente, las amplitudes pequeñas son representadas con mas bits y por lo tanto con una gran resolución.

En la práctica en vez de usar niveles de cuantización no uniforme para adaptar las características de la señal, estas son antes que nada

comprimidas en amplitud en forma no lineal para obligar a que todas caigan dentro de un intervalo específico. Esta característica de compresión es de forma logarítmica.

El efecto que produce la conversión a escala logarítmica es que asegura que las amplitudes pequeñas sean digitalizadas con una mínima pérdida de fidelidad. Este proceso de COMPANSIÓN y el de exPANSION de señales se conoce como COMPANSION .

El proceso básico de *compansión* es mostrado en la figura III.10 en donde, una muestra de entrada analógica es primero comprimida y entonces cuantizada con intervalos de cuantización uniforme.



Fig. III.10 Proceso de Compansión de una señal.

La compansión esta definida por dos estándares internacionales : ley μ 255 y la ley A.

III.3.2.1.1 Ley μ 255 de Compansión.

Es un estándar empleado en los Estados Unidos y Japón, y sus características de compresión estan dados por la ecuación:

$$F(x) = \text{sgn}(x) \frac{\ln(1 + \mu|x|)}{\ln(1 + \mu)} \quad (3.22)$$

de donde :

- $F(x)$: es el valor de salida comprimido
 x : es la señal de entrada normalizada $-1 < x < 1$
 μ : es el parámetro de compresión ($\mu = 255$ en EE UU)
 $\text{sgn}(x)$: es el signo o polaridad de x .

La representación codificada de la ley $\mu 255$ utiliza un código de palabra en un formato de 8 bits de magnitud signada (incluyendo el bit de signo), esta permite hacer una compresión de 13 a 7 bits (sin tomar en cuenta el bit de signo) y una expansión para el mismo número de bits.

En este formato de 8 bits como se ve en la figura III.11, el primer bit nos representa la polaridad y los otros bits restantes especifican la magnitud de la muestra. Estos 7 bits son convenientemente particionados en 3 bits que identifican el segmento identificado como S y 4 bits para el paso de cuantización identificado como Q. Este mismo formato se emplea en la ley A.

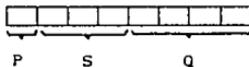


fig. III.11 Formato de 8 bits

Polaridad de $P = 0$ para un valor de muestra positiva

Polaridad de $P = 1$ " " " " " negativa

3 bits número de segmento SSS

4 bits para el paso de cuantización QQQQ

Esta característica de compansión tiene la propiedad de ser aproximadamente cerrada para un conjunto de 8 segmentos de línea recta como se muestra en la figura III.12. En esta figura podemos observar como los valores de la muestra de entrada sucesivas de intervalos grandes son comprimidos en intervalos de tamaño uniforme. La pendiente de cada segmento es exactamente la mitad de la siguiente. El tamaño de paso entre dos códigos de palabras adyacentes es el doble en cada segmento sucesivo. Esta propiedad permite hacer la conversión hacia un formato lineal más eficiente. La aproximación de una línea recta de la curva de compansión $\mu 255$ es algunas veces referido como una aproximación a 15 segmentos, donde hay 8 segmentos positivos y 8 segmentos negativos, donde los 2 segmentos mas cercanos al origen son colineales y son considerados como uno.

1	0	0	0	0	0	0	0
1	0	0	0	1	1	1	1
1	0	0	1	1	1	1	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	1
1	1	0	0	1	1	1	1
1	1	0	1	1	1	1	1
1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1

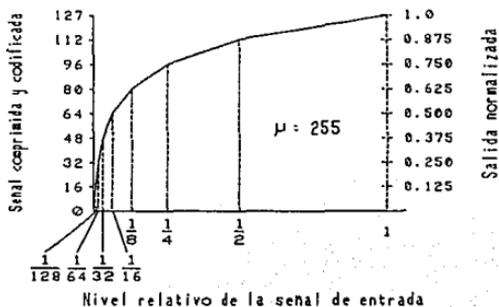


Fig. III.12 Ley $\mu 255$ de compansión.

A partir de la figura III.12 y el formato de codificación de la figura III.11 se obtuvo la tabla I, de donde a partir de ella y con la ayuda de las tablas II y III se generó el algoritmo para la realización de la compresión y expansión.

Lo que se hace es tomar los puntos finales de cada segmento donde sus valores son 31, 95, 223, 4063. Se notará que :

$$\begin{aligned} 31 &= 2^6 - 33 \\ 95 &= 2^7 - 33 \\ 223 &= 2^8 - 33 \\ &\vdots \\ &\vdots \\ &\vdots \\ 4063 &= 2^{12} - 33 \end{aligned}$$

De aquí se puede observar que se suma un valor constante de 33 a cada valor de la tabla. Esto significa que el número de segmento corresponde a un número N (el cual es codificado) que puede ser determinado al encontrar el bit " 1 " más significativo de la representación binaria del número $N + 33$.

En esta parte, se muestra mediante un par de ejemplos las cuales se encuentran abajo de las tablas II y III, el proceso de compresión y el de expansión para los casos positivo y negativo, donde la tablas mencionadas anteriormente nos ayudaran para nuestro propósito.

TABLA I LEY μ 255 CODIFICACION/DECODIFICACION

Rango de Amplitud de Entrada	Tamaño de Paso	Código de Segmento S	Código de Cuantización Q	Valor del código	Amplitud Decodificada
0 - 1	1		0000	0	0
1 - 3			0001	1	2
3 - 5	2	000	0002	2	4
29 - 31			1111	15	30
31 - 35	4	001	0000	16	33
91 - 95			1111	31	93
95 - 103	8	010	0000	32	99
215 - 223			1111	47	219
223 - 239	16	011	0000	48	231
463 - 479			1111	63	471
479 - 511	32	100	0000	64	485
959 - 991			1111	79	975
991 - 1055	64	101	0000	80	1023
1951 - 2015			1111	95	1983
2015 - 2143	128	110	0000	96	2079
3935 - 4063			1111	111	3999
4063 - 4139	256	111	0000	112	4191
7903 - 8159			1111	127	8031

* Esta tabla despliega solamente la magnitud codificada. El bit de polaridad es asignado " 0 " para el " + " y " 1 " para el " - ".

TABLA II $\mu 255$ DE CODIFICACION BINARIA

Valor de Entrada													Código de palabra comprimida								
Bit	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	0	0	0	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	0	1	0	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	0	1	1	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	x	1	0	0	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	x	x	1	0	1	Q ₃	Q ₂	Q ₁	Q ₀	
0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	x	x	x	1	1	0	Q ₃	Q ₂	Q ₁	Q ₀	
1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	x	x	x	x	1	1	1	Q ₃	Q ₂	Q ₁	Q ₀	

* NOTA : La polaridad no se muestra en esta tabla y el bit de carga es el bit de signo

$$(1) +865_{10} \xrightarrow{\text{BIAS}} +865_{10} + 33_{10} = +382_{16} = (0)0\ 0011\ 1000\ 0010_2 \rightarrow (0)100\ 1100_2$$

$$(2) -2513_{16} \xrightarrow{\text{BIAS}} -2513_{16} - 33_{10} = -962_{16} = (1)0\ 1001\ 1111\ 0010_2 \rightarrow (1)110\ 0011_2$$

TABLA III $\mu 255$ DECODIFICACION BINARIA

Código de palabra comprimida										Valor de Salida											
Bit	6	5	4	3	2	1	0	Bit	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	1
0	0	1	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	
0	1	0	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	0	
0	1	1	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	0	0	
1	0	0	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	0	0	0	
1	0	1	Q ₃	Q ₂	Q ₁	Q ₀		0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	0	0	0	0	
1	1	0	Q ₃	Q ₂	Q ₁	Q ₀		0	1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	0	0	0	0	0	
1	1	1	Q ₃	Q ₂	Q ₁	Q ₀		1	Q ₃	Q ₂	Q ₁	Q ₀	0	0	0	0	0	0	0	0	

* NOTA : La polaridad no se muestra en esta tabla y el bit de carga es el bit de signo

$$(1) 3C_{16} = (0)011\ 1100_2 \rightarrow (0)0001\ 1100\ 1000_2 = +01c8_{16} = +456_{10} \xrightarrow{\text{REMOVE BIAS}} +456_{10} - 33_{10} = +423_{10}$$

$$(2) E3_{16} = (1)110\ 0011_2 \rightarrow (1)0\ 1001\ 1100\ 0000_2 = -09c0_{16} = -2496_{10} \xrightarrow{\text{REMOVE BIAS}} -2496_{10} + 33_{10} = -2463_{10}$$

Una expresión que nos permite cuantificar la relación señal a ruido de un proceso de compansión (ley μ 255 para nuestro caso) esta dado por :

$$\text{SNR(dB)} = 6B + 4.77 - 20 \log[\ln(1 + \mu)] - 10 \log \left[1 + \frac{X_{\max}}{\mu \sigma_x} \right]^2 + \sqrt{2} \left[\frac{X_{\max}}{\mu \sigma_x} \right]^2 \quad (3.23)$$

III.3.2.1.2 Ley A de Compansión.

Este estándar es utilizado en Europa y sus características de compresión estan dados por la ecuación :

$$F(x) = \begin{cases} \text{sgn}(x) \frac{A|x|}{1 + \ln(A)} & \text{para } 0 \leq |x| \leq \frac{1}{A} \\ \text{sgn}(x) \frac{(1 + \ln A|x|)}{1 + \ln(A)} & \text{para } \frac{1}{A} \leq |x| \leq 1 \end{cases} \quad (3.24)$$

$F(x)$: es el valor de salida comprimido.

x : es la señal de entrada normalizada $-1 < x < 1$

A : es el parámetro de compresión ($A = 87.6$ en Europa).

$\text{sgn}(x)$: es el signo o polaridad de x .

Este estándar nos permite hacer una compresión de 12 a 7 (bits sin incluir el bit de signo) y una expansión para el mismo número de bits.

La figura III.13 muestra una aproximación práctica de las características de entrada/salida para una curva que tiene una ley A de compansión. En la ley A hay ocho segmentos positivos y ocho segmentos negativos, los dos primeros segmentos de cada polaridad (cuatro en total) son

colineales y por lo tanto algunas veces es considerado como una sola línea, lo que motiva que algunas veces lo conozcan como aproximación de 13 segmentos.

1	1	1	1	1	1	1	1
1	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0
1	1	0	1	0	0	0	0
1	1	0	0	0	0	0	0
1	0	1	1	0	0	0	0
1	0	1	0	0	0	0	0
1	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0

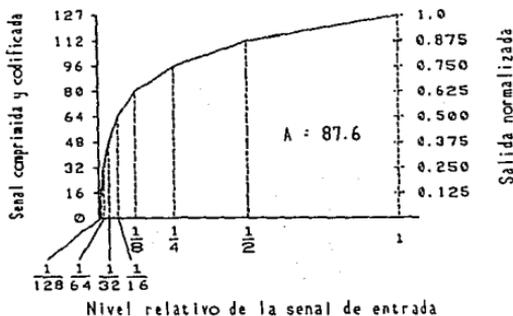


Fig. III.13 Ley A de Compansión.

Al igual que en la ley $\mu 255$, a partir de la figura III.13 y del formato de codificación de la figura III.11, se obtuvo la tabla V, de donde a partir de ella y con la ayuda de las tablas VI y VII generó el algoritmo de compresión y expansión. En la ley A se busca el bit " 1 " más significativo (de más a la izquierda) y se reemplaza por su correspondiente código de palabra comprimida, valiendose para ello de las tablas VI y VII, y sumando y restando un valor constante que se vera en los ejemplos que se muestran de abajo de las tablas mencionadas.

TABLA V LEY A CODIFICACION/DECODIFICACION

Rango de Amplitud de Entrada	Tamaño de Paso	Código de Segmento S	Código de Cuantización Q	Valor del código	Amplitud Decodificada
0 - 2			0000	0	1
2 - 4		000	0001	1	3
30 - 32	2		1111	15	31
32 - 34			0000	16	33
		001			
62 - 64			1111	31	63
64 - 68	4		0000	32	66
		010			
124 - 128			1111	47	126
128 - 136	8		0000	48	132
		011			
248 - 256			1111	63	252
256 - 272	16		0000	64	264
		100			
496 - 512			1111	79	504
512 - 544	32		0000	80	528
		101			
992 - 1024			1111	95	1008
1024 - 1088	64		0000	96	1056
		110			
1984 - 2048			1111	111	2016
2048 - 2176	128		0000	112	2112
		111			
3968 - 4096			1111	127	4032

* Esta tabla despliega solamente la magnitud codificada. El bit de polaridad es asignado " 0 " para el " + " y " 1 " para el " - "

TABLA VI LEY A DECODIFICACION BINARIA

Valor de Entrada											Código de Palabra Comprimido									
Bit	11	10	9	8	7	6	5	4	3	2	1	0	Bit	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	Q ₃	Q ₂	Q ₁	Q ₀	x	0	0	0	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x		0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x		0	1	0	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x		0	1	1	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x		1	0	0	Q ₃	Q ₂	Q ₁	Q ₀	
0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x		1	0	1	Q ₃	Q ₂	Q ₁	Q ₀	
0	1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	x		1	1	0	Q ₃	Q ₂	Q ₁	Q ₀	
1	Q ₃	Q ₂	Q ₁	Q ₀	x	x	x	x	x	x	x		1	1	1	Q ₃	Q ₂	Q ₁	Q ₀	

* NOTA : La polaridad no se muestra en esta tabla y el bit de carga es el bit de signo

$$(1) + 3221_{10} = + C95_{16} = (0)1100\ 1001\ 0101_2 + (0)111\ 1001_2$$

$$(2) - 199_{10} = - C7_{16} = (1)0000\ 1100\ 0111_2 - (1)011\ 1000_2$$

TABLA VII LEY A DECODIFICACION BINARIA

Código de Palabra Comprimido								Valor de Salida												
Bit	6	5	4	3	2	1	0	Bit	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	0	0	0	0	Q ₃	Q ₂	Q ₁	Q ₀	1
0	0	1	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	1	
0	1	0	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	1	0	
0	1	1	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	1	0	0	
1	0	0	Q ₃	Q ₂	Q ₁	Q ₀		0	0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	1	0	0	0	
1	0	1	Q ₃	Q ₂	Q ₁	Q ₀		0	0	1	Q ₃	Q ₂	Q ₁	Q ₀	1	0	0	0	0	
1	1	0	Q ₃	Q ₂	Q ₁	Q ₀		0	1	Q ₃	Q ₂	Q ₁	Q ₀	1	0	0	0	0	0	
1	1	1	Q ₃	Q ₂	Q ₁	Q ₀		1	Q ₃	Q ₂	Q ₁	Q ₀	1	0	0	0	0	0	0	

* NOTA : La polaridad no se muestra en esta tabla y el bit de carga es el bit de signo

$$(1) (0)001\ 1101_2 + (0)0000\ 0011\ 1011_2 = + 3B_{16} = + 59_{10}$$

$$(2) (1)110\ 0100_2 + (1)0101\ 0010\ 0000_2 = - 520_{16} = - 1312_{10}$$

III.3.3 Comparación de la relación señal a ruido entre la cuantización uniforme y no uniforme.

En todo sistema, el ruido siempre se hará presente al ser inherente a éste, sin embargo, el utilizar una cuantización uniforme o no uniforme nos llevará a obtener una mejor relación señal a ruido para un determinado número de bits.

En esta parte se hará una comparación de la ley $\mu 255$ de compansión con la cuantización uniforme. En donde evaluando las ecuaciones (3.19) y (3.23) se obtienen las gráficas que se muestran en las figuras III.14, III.15 y III.16.

En las gráficas antes mencionadas se observan un conjunto de curvas que representan el número de bits empleados para transmitir la señal con el menor ruido posible, también se puede ver las curvas y los bits necesarios para una determinada SNR, así como las curvas para una cuantización uniforme.

Las curvas miden la relación señal a ruido (SNR) contra $\left[\frac{X_{max}}{\sigma_x} \right]$ y en ellas se muestra una clara reducción en SNR a medida que $\left[\frac{X_{max}}{\sigma_x} \right] \rightarrow 1$. A través de las ecuaciones (3.19) y (3.23) se puede demostrar que un buen valor de SNR, se obtiene cuanto $\left[\frac{X_{max}}{\sigma_x} \right] > 8$.

De las gráficas se puede ver que para una $\mu = 100$ se tiene un intervalo de

$$8 < \left[\frac{X_{max}}{\sigma_x} \right] < 30 \quad (a)$$

para $\mu = 255$ se tiene un intervalo de

$$8 < \left[\frac{X_{max}}{\sigma_x} \right] < 100 \quad (b)$$

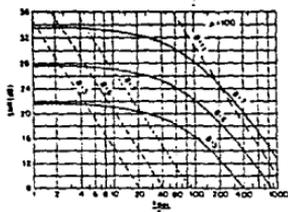


Fig. III.14 Para $\mu = 100$

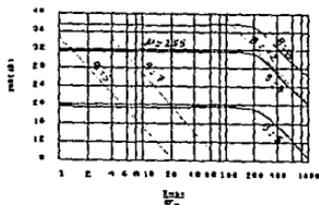


Fig. III.15 Para $\mu = 255$

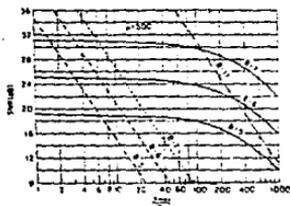


Fig. III.16 Para $\mu = 500$

SNR para la ley μ y una cuantización uniforme como una función de $\left(\frac{X_{max}}{\sigma_x}\right)$ para una μ determinada y diferentes números de bits (B) del cuantizador.

de igual manera para un $\mu = 500$ se tiene

$$8 < \left[\frac{X_{max}}{\sigma_x} \right] < 150 \quad (c)$$

De las gráficas se observa que con $B = 7$, la SNR esta sobre 30 dB, manteniendose en el intervalo marcado por (b) para una señal de entrada dada, es por esta razón que se emplea en los sistemas PCM.

III.3.4 Cuantización Adaptiva.

La amplitud de una señal de voz puede variar dependiendo de la persona, del medio de comunicación y de su naturaleza no estacionaria. Una alternativa para uniformizar la amplitud de la señal de entrada es mediante un cuantizador no uniforme o un cuantizador por niveles de señal de entrada, adaptándola a las características deseadas. Cuando una cuantización es usada directamente sobre un sistema de entrada, es llamada PCM adaptiva o simplemente, APCM.

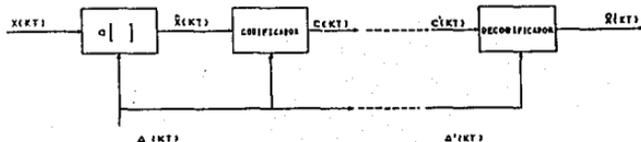


Fig. III.17 Proceso de cuantización Adaptiva.

La idea básica de la cuantización adaptiva es dar un paso Δ o nivel de cuantización y un rango que varíe de acuerdo a las variaciones de la señal de entrada (Figura III.17).

En este esquema de cuantización, la amplitud o variación de la señal de entrada, es estimada de la entrada misma, es decir se ajusta el paso Δ dependiendo de la entrada. En general el tiempo de adaptación del cuantizador puede ser instantáneo.

III.3.5 Cuantización Diferencial.

Debido a la naturaleza de la señal de voz, ésta no cambia rápidamente de muestra a muestra, lo cual quiere decir que la diferencia entre muestras adyacentes están correlacionadas. Como la diferencia entre muestras adyacentes tienen una variación más pequeña con respecto a la variación de la señal misma, esto motivó la creación de un esquema de cuantización diferencial (figura III.18). En este sistema la entrada al cuantizador es una señal,

$$d(kT) = X(kT) - \hat{X}(kT) \quad (3.25)$$

la cual es la diferencia $X(kT)$ y un estimador o predictor de la muestra de entrada, la que se denotará por $\hat{X}(kT)$. Este estimador es la salida de un sistema predictor P , cuya entrada es una versión cuantizada de la señal de entrada $X(kT)$. la señal $d(kT)$ es llamada señal de error de predicción. En este esquema el cuantizador puede ser fijo o variable, uniforme o no uniforme, pero en cualquiera de los casos, sus parámetros

deben ser ajustados a la variancia de $d(kT)$.

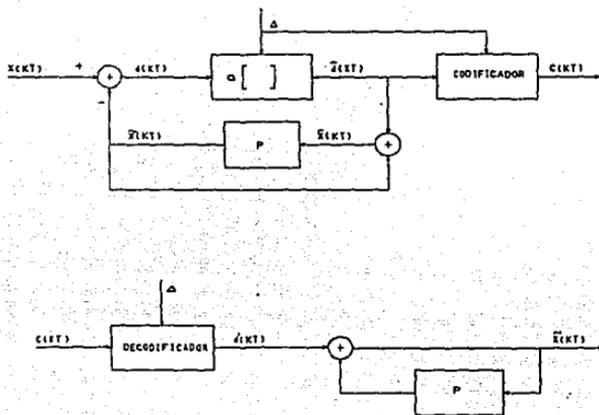


Fig. III.18 Cuantización diferencial. a) Codificador b) Decodificador

La señal de diferencia cuantizada se puede expresar como:

$$\hat{d}(kT) = d(kT) + e(kT) \quad (3.26)$$

donde $e(kT)$ es el error de cuantización. De acuerdo a la figura III.17, la señal de diferencia cuantizada sumada con el valor del predictor $\tilde{x}(kT)$ produce una versión de la señal de entrada, es decir

$$\hat{x}(kT) = \tilde{x}(kT) + \hat{d}(kT) \quad (3.27)$$

de donde sustituyendo (3.25) y (3.26) en (3.27) se obtiene

$$\hat{X}(kT) = X(kT) + e(kT) \quad (3.28)$$

Esto es, independientemente de las propiedades del sistema predictor, la cuantización de la muestra de voz difiere de la entrada solamente por el error de cuantización de la señal de diferencia. Por lo tanto, si el predictor es bueno, la variancia de $d(kT)$ sera más pequeña que la variancia de $X(kT)$, de tal manera que el cuantizador diferencial con un número de niveles dados, puede ser ajustado para dar un menor error de cuantización, comparado con el mínimo error de cuantización que se tuviera al cuantizar la señal original.

IV. DISEÑO Y REALIZACION DEL SISTEMA CODEC .

IV.1 INTRODUCCION.

La parte del sistema de conversión A/D - D/A para compresión y transmisión de señales de voz, a la que llamaremos sistema CODEC, es la que nos permite hacer la conversión propiamente dicha, en donde a base de un circuito CODEC (TPS116) se diseñó toda la arquitectura necesaria para realizar la adquisición de dicha señal. Aquí se comentará el diseño y realización de este sistema, así como de las partes que lo constituyen, y del empleo del TPS116 en la realización de este.

Es importante mencionar que el sistema CODEC se diseñó de tal manera que se pudiera conectar a una PC para almacenar el dato digitalizado, el cual esta comprimido en ley $\mu 255$, logrando con ello un ahorro de memoria de la PC. Así mismo también se contempló la necesidad de conectar el sistema CODEC a un sistema de PDS, para nuestro caso un sistema a base del μP TMS32010, el cual permite hacer un procesamiento en tiempo real.

En la figura IV.1 se muestra el sistema CODEC, donde se observa que cuando se hace la conversión A/D, la señal analógica de entrada (voz) es acondicionada por el filtro de entrada, el cual esta compuesto por un FFB en serie con un FPA y el filtro de transmisión del TP3040. Esta señal es posteriormente digitalizada a través del CODEC, entregando un dato de 8 bits en serie, que será convertida en paralelo a través del convertidor S/P, para así lograr su conexión con la Interfase Sistema CODEC - PC o con el bloque de memorias de expansión en donde se conectará la Interfase Sistema CODEC - TMS32010. Así mismo, veremos que en el proceso inverso

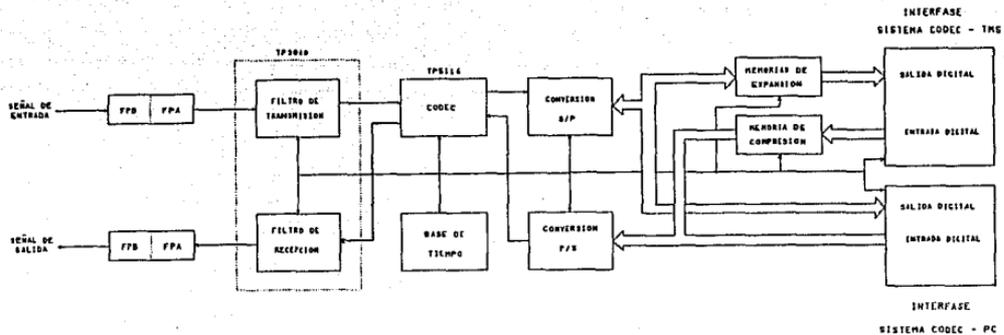


FIG. IV.1 SISTEMA CODEC

(conversión D/A), el dato proporcionado por la Interfase Sistema CODEC-PC o por la Interfase Sistema CODEC - TMS32010 es convertido en P/S para que pueda ser tomado por el circuito CODEC, y posteriormente sea filtrada entregando una señal analógica a la salida.

Cabe mencionar que también se desarrollará la base de tiempo, la cual proporcionará las señales de control para el sistema CODEC, así como para todo el hardware a su alrededor (Interfase Sistema CODEC - PC e Interfase Sistema CODEC - TMS32010).

Es importante mencionar que solo puede estar habilitado la Interfase Sistema CODEC - PC o la Interfase Sistema CODEC - TMS32010, esto se logra al deshabilitar las memorias de expansión y compresión para el primer caso, o deshabilitando la propia Interfase Sistema CODEC - PC para el segundo caso.

Por otro lado sabemos que en los sistemas PCM, que son comunmente usados en redes telefónicas públicas y privadas, las señales analógicas de voz son codificadas en palabras binarias y transmitidas usualmente en serie a una razón de 8000 muestras/seg. Esta digitalización de datos es comunicada más eficientemente como se mencionó en el capítulo anterior , si la amplitud de la señal es comprimida a una escala logarítmica y después transmitida para posteriormente ser expandidas en el receptor.

Los detalles del diseño del sistema CODEC se darán en cada una de las partes en que está dividido el presente capítulo.

Las partes siguientes corresponden al diseño de este sistema :

- etapa de filtrado.
- etapa de conversión A/D - D/A.
- base de tiempo.

- conversión S/P y P/S.
- compresión y expansión.

para lo cual se tienen las siguientes características principales :

Conversión A/D:

frecuencia de muestreo programable
entrada : Señal Analógica de voz (200 - 3400 [kHz])
salida : Señal Digital de voz (8 bits comprimidos en ley μ en su interfase con la PC y 14 bits en complemento a 2 para la interfase con el TMS32010).

Conversión D/A :

frecuencia de muestreo programable
entrada : Señal Digital de voz (8 bits comprimidos en ley μ 255 en su interfase con la PC y 14 bits en complemento a 2 para la interfase con el TMS32010).
salida : Señal Analógica de voz (200 - 3400 [kHz])

IV.2 ETAPA DE FILTRADO.

Esta etapa constituye la primera parte del sistema y consiste en acondicionar la señal de voz a los requerimientos de esta, de tal manera que se tenga una señal entre 200-3400 Hz, que es el rango de frecuencias que se tiene en una línea telefónica. Las características de una red telefónica se muestran en la figura IV.2

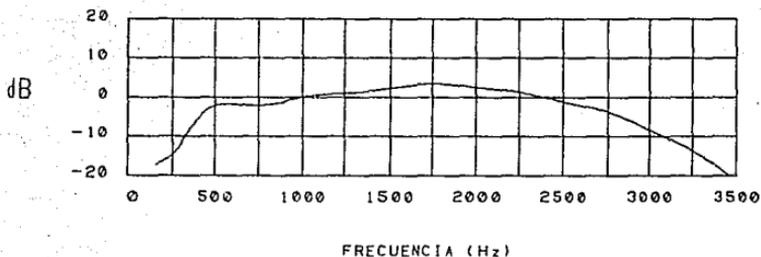


Fig. IV.2 Respuesta en frecuencia de una línea telefónica

El filtrado se realizó en dos etapas las cuales se mencionan a continuación :

IV.2.1 Filtrado de entrada.

Este filtrado consiste en acondicionar la señal de voz para su digitalización (A/D). En ésta etapa como se muestra en la figura IV.3, se usaron dos bloques de filtrado.

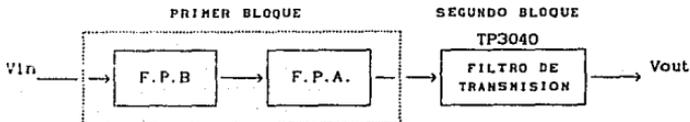


FIG IV.3 Filtrado de entrada

En el primer bloque, se diseñó un Filtro Paso Bajas (F.P.B.) en serie con un filtro paso altas (F.P.A.), el cual nos da una característica de un filtro paso banda (200 - 3400 Hz), restringiendo el ancho de banda a las frecuencias de corte de cada uno de los filtros.

En el diseño de los filtros se utilizó una arreglo de realimentación múltiple (figura IV.4 y V.5), para obtener una mayor estabilidad del circuito, así como una característica de un filtro de Butterworth, de tal manera que nos proporcionará una respuesta plana y ganancia unitaria.

Los cálculos se muestran a continuación :

Diseño del Filtro Paso Bajas

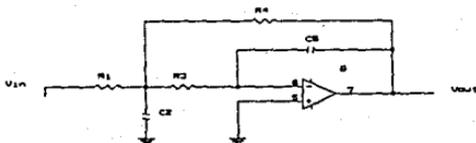


Fig. IV.4 Filtro paso bajas.

Sabemos que un FPB tiene una función de transferencia (F.T.) de :

$$H(S) = \frac{H_0 \omega_n^2}{S^2 + \alpha \omega_n S + \omega_n^2} \quad (4.1)$$

Por otro lado, de la figura IV.4 se obtiene la F.T. siguiente :

$$H(S) = \frac{1}{R_1 R_3 C_2 C_5} \left[\frac{1}{R_1} + \frac{1}{R_3} + \frac{1}{R_4} \right] \frac{1}{C_2} + \frac{1}{R_3 R_4 C_2 C_5} \quad (4.2)$$

de donde comparando (4.1) con (4.2) tenemos que :

$$\omega_n = \sqrt{\frac{1}{R_3 R_4 C_2 C_5}}$$

$$\alpha = \sqrt{\frac{C_5}{C_2}} \left[\sqrt{\frac{R_3 R_4}{R_1}} + \sqrt{\frac{R_4}{R_3}} + \sqrt{\frac{R_3}{R_4}} \right]$$

$$H_0 = \frac{R_4}{R_1}$$

$$Q = \frac{1}{\alpha}$$

Para propósitos de diseño hacemos que :

$$k = \frac{1}{4 Q^2 (H_0 + 1)}$$

$$C_2 = C \quad ; \quad C_5 = K C$$

$$\omega_s = 2 \pi f_c$$

$$R_4 = \frac{1}{2 Q \omega_0 C_2 K}$$

$$R_3 = \frac{R_4}{H_0 + 1}$$

$$R_1 = \frac{R_4}{H_0}$$

Luego entonces para un filtro paso altas con $f_c = 200$ Hz, con

$H_0 = 1$, y $Q = 0.7071$ se tiene que :

$$k = \frac{1}{4 Q^2 (H_0 + 1)} = \frac{1}{4 (0.7071)^2 (2)} = 0.25$$

Si $C_2 = 0.1 \mu\text{f}$ tenemos entonces que :

$$C_5 = 0.25(0.1) = 0.025 \mu\text{f} \approx 0.022 \mu\text{f}$$

$$R_4 = \frac{1}{2(0.7071)(2\pi)(3.4 \times 10^{-3})(0.1 \times 10^{-6})(0.25)} = 1323 \Omega$$

$$R_4 \approx 1.2 \text{ K}\Omega$$

$$R_3 = \frac{1323}{1 + 1} = 661.98 \Omega \approx 680 \Omega$$

$$R_1 = \frac{R_4}{H_0} = R_4 = 1.2 \text{ K}\Omega$$

Diseño del Filtro Paso Altas

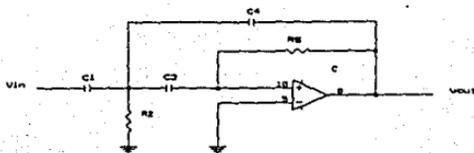


Fig. IV.5 Filtro paso altas

Sabemos que un FPA tiene una función de transferencia (F.T.) de :

$$H(S) = \frac{H_0 S^2}{S^2 + \alpha \omega S + \omega_n^2} \quad (4.3)$$

Por otro lado, de la figura IV.5 se obtiene la F.T. siguiente :

$$H(s) = \frac{-\frac{C_1}{C_4} S^2}{S^2 + \frac{1}{R_5} \left[\frac{C_1}{C_3 C_4} + \frac{1}{C_4} + \frac{1}{C_3} \right] S + \frac{1}{R_2 R_5 C_3 C_4}} \quad (4.4)$$

comparando (4.3) con (4.4) tenemos que :

$$n = \sqrt{\frac{1}{R_2 R_5 C_3 C_4}}$$

$$\alpha = \frac{1}{Q} = \sqrt{\frac{R_2}{R_5}} \left[\frac{C_1}{\sqrt{C_3 C_4}} + \sqrt{\frac{C_3}{C_4}} + \sqrt{\frac{C_4}{C_3}} \right]$$

Para propósitos de diseño tenemos que

$$C_1 = C_3 = C$$

$$R_5 = \frac{Q}{\omega_0 C} (2 H_0 + 1)$$

$$R_2 = \frac{1}{Q \omega_0 C (H_0 + 1)}$$

Luego entonces para un filtro paso altas con $f_c = 200$ Hz, con

$H_0 = 1$, y $Q = 0.7071$ se tiene que :

Si $C = 0.1 \mu\text{f}$

$$R_B = \frac{0.7071 (2 + 1)}{2\pi (200)(0.1 \times 10^{-6})} = 16.880 \text{ K}\Omega \approx 16 \text{ K}\Omega$$

$$R_2 = \frac{1}{2\pi (0.7071)(200)(0.1 \times 10^{-6})(2 + 1)} = 3.751 \text{ K}\Omega \approx 3.8 \text{ K}\Omega$$

El segundo bloque está conformado por el circuito TP3040 (ver apéndice). Este filtro es un circuito monolítico que contiene filtros para recepción y transmisión, diseñados especialmente para aplicaciones de PCM donde la frecuencia de muestreo es por lo general de 8 KHz . Este circuito como se muestra en la figura IV.6 se divide en dos partes:

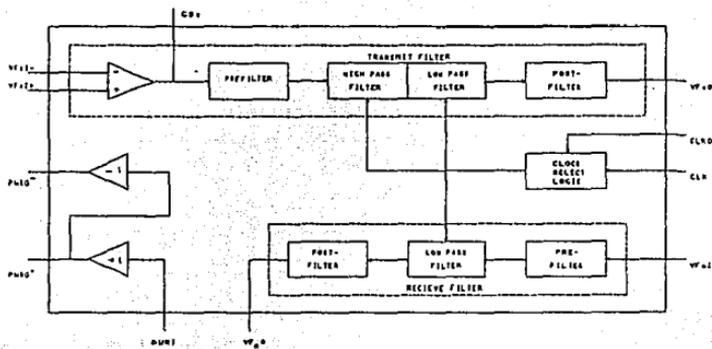


Fig. IV.6 Filtro para PCM (TP3040).

Filtro de Transmisión.- El filtro de transmisión es un filtro paso bajas elíptico de quinto orden en serie con un filtro Chebyshev de cuarto orden, que proporciona una respuesta de tipo paso banda en un rango de 200- 3400 Hz.

Filtro de Recepción.- El filtro de recepción es un filtro paso bajas elíptico de cuarto orden diseñado para reconstruir la señal de voz del decodificador, como resultado del proceso de muestreo. Esta parte se utiliza en la etapa de *Filtrado de Salida*.

IV.2.2 Filtrado de Salida.

Esta etapa nos permite acondicionar la señal de salida que nos proporciona el sistema CODEC.

Esta etapa como se muestra en la figura IV.7 esta compuesta por el filtro de recepción del TP3040, así como de un filtro paso banda (FPB en serie con un FPA) el cual nos sirve para suavizar la señal después de salir de éste.

El diseño del filtro Paso Banda tiene las mismas características que el primer bloque de filtrado de entrada, por lo que ya no es necesario volver a calcularlo.



FIG IV.7 Filtro de salida

IV.3 CONVERSION A/D - D/A.

Debido a que se va a procesar señales de voz , se optó (razones que se explicaron en el capítulo anterior) por utilizar un convertidor con características no lineales, es decir que que tuviese una cuantización no uniforme, logrando así una mínima pérdida de fidelidad de la señal.

Esta conversión se basó en el uso de un CODEC (TPS116), que es un circuito usado especialmente en sistemas PCM.

IV.3.1 Codec.

Debido al gran avance de la telefonía digital, así como la de su gran proliferación, la cual usa 64 KBits/seg , muchas compañías que fabrican circuitos integrados han desarrollado circuitos capaces de digitalizar una señal analógica en camino hacia un transmisor, así como de convertir una señal digital en analógica en camino hacia el receptor, estos circuitos son llamados CODEC's (CODers-DECCoders), normalmente cuando se habla de CODEC's se supone una cuantización no uniforme , mientras que en convertidores A/D y D/A, esto implica una cuantización uniforme . Los CODEC's han sido desarrollados para ley μ y ley A para codificación en PCM, además, en años recientes, algunas compañías han desarrollado circuitos para Modulación Delta Adaptiva (ADM) y PCM Diferencial Adaptiva (ADPCM).

IV.3.2 Codec TPS116.

Este CODEC es un dispositivo de tecnología CMOS que utiliza la ley μ (en formato magnitud signada) el cual contiene un convertidor A/D - D/A en forma separada, y otros circuitos como el de muestreo y retén, el de control interno, el de auto-zero, y el de referencia de voltaje, etc. (ver figura IV.8 y apéndice).

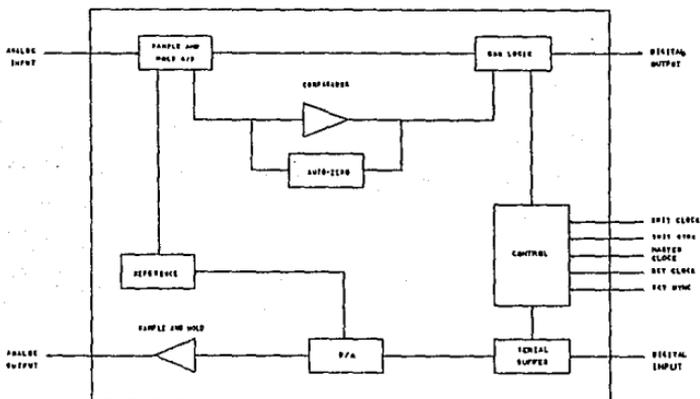


Fig. IV.8 Diagrama de bloque de un circuito CODEC TPS116.

Este circuito necesita de cinco señales de control, necesarias para su transmisión y recepción, las cuales son las siguientes:

-MASTER CLOCK.- Esta entrada es usada para controlar internamente la secuencia de codificación y decodificación, y puede ser 1.536 MHz, 1.544 MHz, ó 2.048 MHz.

-XMIT CLOCK.- Es el reloj de transmisión de los bits de la palabra dato, esta señal de entrada es usada para recorrer el dato PCM en la salida digital (DIGITAL OUTPUT), puede operar de 64 KHz hasta 2.048 MHz, y puede ser asincrónica con RCV CLOCK.

-XMIT SYNC.- Es el pulso de sincronía del " frame " (palabra dato = 8 bits), normalmente ocurre a 8 KHz, este pulso es nominalmente de ocho veces el ancho del pulso del XMIT CLOCK.

-RCV CLOCK.- Es el reloj de recepción de los bits de la palabra dato de entrada, es usado para recorrer el dato PCM en la entrada digital (DIGITAL INPUT).

-RCV SYNC.- Es el pulso de sincronía del decodificador , normalmente ocurre a 8 KHz, este pulso es ocho veces el ancho de pulso del RCV CLOCK.

Funcionamiento.- De las hojas de especificaciones (apéndice) del circuito, podemos ver que aproximadamente 4 μ seg después del flanco de subida del pulso de XMIT SYNC, el voltaje presentado a la entrada analógica(ANALOG INPUT) es muestreada y el proceso de codificación de las muestras es realizado en código PCM, apareciendo en la DIGITAL OUTPUT un código PCM de 8 bits en serie correspondiente a la muestra previa, teniéndose primero el bit de mayor peso (MSB); cada bit dato aparece en cada flanco de subida del XMIT CLOCK, con lo cual se tiene el dato completo en ocho ciclos del XMIT CLOCK, para después pasar a la etapa de conversión de serie-paralelo (S/P) (la cual se mencionará en la parte IV.5 de este capítulo).

Cuando el XMIT SYNC (normalmente ocho veces XMIT CLOCK) va a un nivel bajo, la DIGITAL INPUT se pone en alta impedancia. El MASTER CLOCK es

usado para operar la secuencia de codificación y decodificación.

Cuando se hace la conversión D/A los ocho bits del código PCM que vienen de un convertidor P/S son recorridos en la entrada digital (DIGITAL INPUT) con el flanco de subida del RCV CLOCK, una vez que el RCV SYNC está en un nivel alto. Esto quiere decir que aproximadamente 15 μ seg después de que pase a un nivel alto, la salida analógica (ANALOG OUTPUT) es cargada con el voltaje correspondiente a su código de entrada en PCM.

El RCV SYNC, al igual que el XMIT SYNC, es ocho veces el RCV CLOCK.

El uso de las señales de control para el CODEC nos la proporciona la base de tiempo, esto es una FMASTER CLOCK = 1.536 MHz en donde se cumple la condición de que RCV SYNC debe permanecer en nivel bajo cuando menos 17 ciclos de MASTER CLOCK (ver especificaciones en el apéndice A).

IV.4 BASE DE TIEMPO.

Como sabemos la gran mayoría de los sistemas de conversión A/D y D/A, necesitan una base de tiempo para poder operar de una manera eficiente. Este sistema no es la excepción, por lo que se da gran énfasis en la elaboración de ésta parte, ya que su buen funcionamiento, implicará que el sistema completo lo haga eficientemente.

Como se mencionó en el punto anterior, el CODEC que se utilizó requiere de cinco señales para su buena operación, y de las especificaciones se puede ver que la señales XMIT CLOCK y RCV CLOCK pueden ser de la misma frecuencia, al igual que XMIT SYNC con RCV SYNC. En el apéndice se muestran las frecuencias que se necesitan para cada una de las señales.

Para generar la frecuencia del MASTER CLOCK se utilizó un oscilador de 18.432 [MHz] debido a que éste valor es comercial. Luego se realizó una división entera entre 12 (U15 y U16), la cual nos proporciona la frecuencia requerida de 1.536 [MHz] en el MASTER CLOCK. Así mismo, debido a que se deseaba una frecuencia de muestreo que se pudiera programar, fué necesario poner un arreglo de divisores (U17 U18), de tal manera que al cargar un número determinado en los contadores se obtuviera la frecuencia deseada.

El número que se debía de cargar en los contadores estaba dada por la expresión siguiente :

$$F_{\text{MUESTREO}} = \frac{F_{\text{MASTER CLOCK}}}{N} \quad (4.5)$$

$$N = \frac{F_{\text{MASTER CLOCK}}}{F_{\text{MUESTREO}}} \quad (4.6)$$

de donde :

N : es el número que se carga a los contadores (U15 U16).

$F_{\text{MASTER CLOCK}}$: es la frecuencia MASTER CLOCK del CODEC.

F_{MUESTREO} : es la frecuencia de muestreo deseada.

El diseño de la base tiempo se muestra en la figura IV.9 de donde podemos observar que los circuitos U17 y U18 nos permiten dividir la frecuencia MASTER CLOCK hasta obtener un rango de frecuencia de muestreo de 1.536 MHz hasta 6.023 KHz en hardware, pero como el CODEC lo mas que nos permite es 72 KHz en las señales de XMIT SYNC y RCV SYNC, el rango de frecuencias real de F_{MUESTREO} estará comprendido entre 6.023 KHz a 72 KHz lo que nos da un valor de N comprendido entre 22 y 255.

IV.5 CONVERSION SERIE/PARALELO Y PARALELO/SERIE.

Esta parte es necesaria, ya que como mencionabamos anteriormente, en la conversión A/D el CODEC nos proporciona un dato en serie, la cual debe de convertirse en paralelo para su interfase con la PC, así como para direccionar las memorias EPROM donde se encuentran los valores expandidos en ley. μ . Del mismo modo, cuando se hace la conversión D/A el CODEC requiere en su DIGITAL INPUT los datos en serie por lo que es necesario hacer el proceso inverso, la conversión paralelo/serie (P/S).

IV.5.1 Conversión Serie/Paralelo.

Esta parte nos sirve para cuando se esta haciendo la conversión A/D. En ésta etapa de conversión se utilizó el circuito 74LS164 (que es un registro de corrimiento serie-paralelo), (U22) donde el dato a la entrada es recorrido de un flip-flop a otro en cada flanco de subida del reloj, siendo el primer flip-flop el A y el último el H. Este dispositivo cuenta con una señal de CLEAR que es activa baja, la cual da un borrado a ceros cuando se pone un nivel bajo en este terminal.

Según el diagrama de tiempos (figura IV.10), el ck del registro es la misma señal $XMIT SYNC$ solo que defasada y sincronizada con $XMIT CLOCK$, el CLEAR puede ser la señal $XMIT SYNC$ que es la que nos permite actualizar el dato.

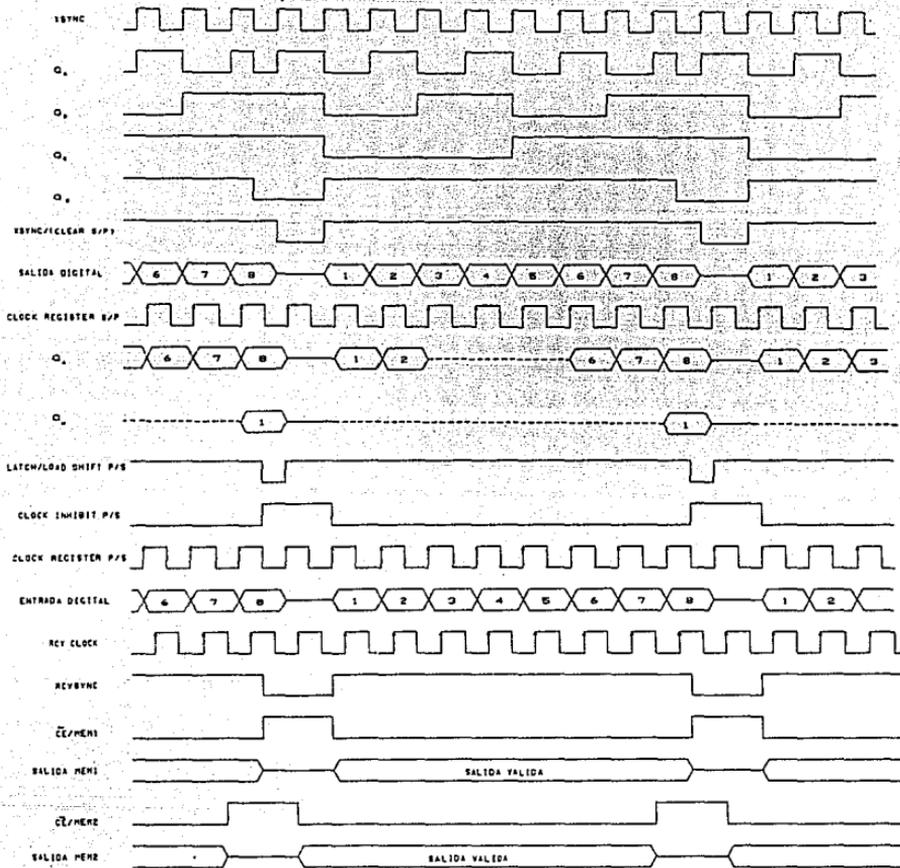


FIG. IV.10 DIAGRAMA DE TIEMPOS.

IV.5.2 Conversión Paralelo/Serie

Al igual que en el párrafo anterior esta parte nos sirve para cuando se hace la conversión D/A.

Debido a que se tiene un bus de datos (8 bits en paralelo), el cual nos proporciona la PC o la memoria 2764 (U31) (dato comprimido), es necesario hacer una conversión paralelo-serie debido a que el CODEC recibe los datos en serie, la cual se logra gracias al circuito 74LS165(que es un registro de corrimiento paralelo-serie de 8 bits) (U24).

El dato paralelo es accesado con un nivel bajo en la terminal SHIFT/LOAD y aparecerá en serie cuando la terminal CLOCK INHIBIT vaya a un nivel bajo y el SHIFT/LOAD pase a un nivel alto.

IV.6 EXPANSION Y COMPRESION.

La expansión consiste en pasar de 8 bits serie a 14 bits paralelo (incluyendo el bit de signo) como se puede observar en la tabla III, y la compresión de 14 bits paralelo a 8 bits serie, aplicando ley μ en ambos casos. Debe notarse que en la parte de compresión solo se toman 13 bits debido a que el bit menos significativo no es relevante como se puede observar en la tabla II.

La técnica de expansión y compresión consiste en direccionar unas memorias la cuales tienen grabadas los valores que nos permiten hacer el paso de 8 bits a 14 bits y viceversa segun las tablas II y III.

De las memorias disponibles, dos se programaron con los datos expandidos y 1 con los datos comprimidos.

El conjunto de memorias de expansión va acompañado de dos latch (U27 U28 los cuales forman parte de la interfase TMS32010 y que se explicará en el siguiente capítulo) que mantienen el dato a la salida del sistema CODEC, dato que es proporcional al valor analógico de entrada, el cual se actualizará cada vez que las memorias sean direccionados. El proceso de compresión funciona en forma similar solo que a la salida de las memorias no es necesario otro latch debido a que el registro paralelo-serie (U24) tiene el tiempo necesario para poder capturar el dato que le es proporcionado por la memoria (U31) para que así entren los datos en serie al CODEC

Las memorias utilizadas son EPROM 2716 de 2 kByte (U25 U26) con un tiempo de acceso de 350 ns. Estas se encuentran conectadas en paralelo, debido a la necesidad de obtener 14 bits a la salida , ya que cada memoria solo cuenta con 8 líneas de salida (bus de datos) , de tal manera que la primera memoria nos de los 8 bits menos significativos y los restantes 6 bits la segunda.

En el proceso inverso la memoria que se utilizó fué la EPROM 2764 de 8 kBytes (U31) el cual tiene 13 líneas de direccionamiento, debido a que en la compresión (ley $\mu 255$), como se explicó anteriormente solo se necesita 13 bits para su decodificación.

En la figura IV.11 se muestra el diagrama completo de todo el sistema CODEC, donde se puede observar la etapa de filtrado, la la de conversión A/D - D/A, la base de tiempo, la parte de conversión S/P - P/S, así como la de compresión y expansión.

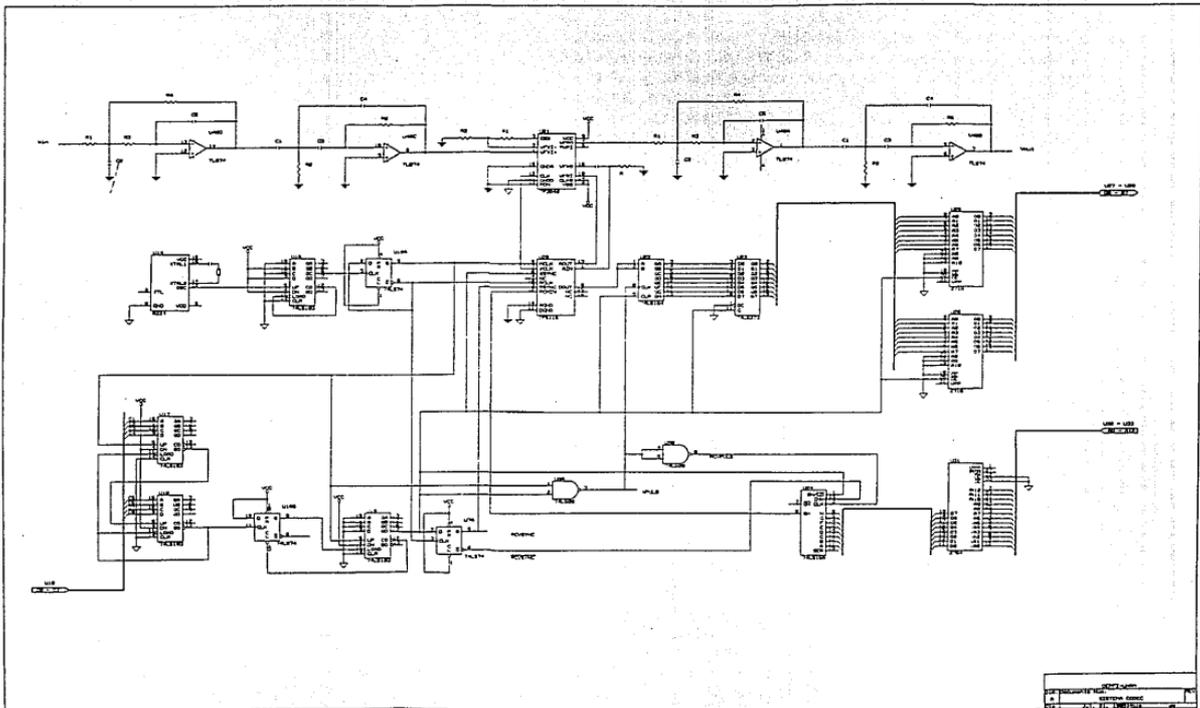


FIG. IV.11

V. DISEÑO DE LAS INTERFASES PARA LA PC Y EL TMS32010.

V.1 INTRODUCCION.

Debido a la necesidad de conectar el sistema CODEC a una computadora PC para almacenar la información digitalizada, o conectarla con un μP TMS32010 de tal manera que nos permitiera hacer procesamiento en tiempo real, se diseñaron las interfases para cada una de ellas, de tal manera que se tuviera la opción, dentro del sistema de conversión A/D - D/A para compresión y transmisión de señales de voz, de elegir cualquiera de éstas para una determinada aplicación.

En el presente capítulo se presenta el diseño de la Interfase Sistema CODEC - PC y el de la Interfase Sistema CODEC - TMS32010 , donde se presentará la teoría necesaria para su diseño y realización.

V.2 INTERFASE SISTEMA CODEC - PC

V.2.1 Hardware de Entrada/salida para una PC.

Hay cuatro partes principales para una interfase de entrada/salida entre un dispositivo periférico y una PC. Estas son:

- Habilitador para un dispositivo periférico.
- Habilitador para lectura y escritura (-IOR,-IOW) de la computadora al o del dispositivo periférico.
- Habilitador de lectura y escritura del dispositivo periférico.

- Latch de salida.

Habilitador para un dispositivo periférico.

Cuando la computadora esta direccionando un circuito de entrada/salida se tiene el habilitador de circuito en estado activo.

Esta habilitación se realiza internamente en la PC mediante las líneas de dirección y otras líneas de control que se explicaran en los siguientes párrafos. La figura V.1 muestra un conector de I/O de la PC desplegando el número de terminal y el nombre de las señales que la conforman.

La señal de habilitación en un sistema general se obtiene como combinación lógica de las líneas de dirección, que en nuestro caso son A0-A9; esta señal es conocida como PORTSEL y es activa baja "0" cuando la dirección del puerto se presenta, y "1" para cualquier otra dirección (Ver la figura V.2).

Así mismo, en la figura V.2 también se muestra la señal AEN. Esta línea es usada para informar al habilitador del dispositivo periférico que hay una dirección valida en el bus de direcciones . Cuando esta señal es activa baja "0", la dirección de salida A0-A9 es válida y puede ser usada para la decodificación de la dirección del puerto de salida.

Sin embargo cuando AEN es lógica "1", nos indica que la dirección no esta siendo generada por el microprocesador, y no puede ser usada para decodificar un puerto de salida.

Como se desean seleccionar un dispositivo ya sea de entrada o salida, el AEN debe ser usado y debe estar en un nivel lógico "0" como se ve en la figura V.2 . El BOARDSEL no puede ser "0" a menos que el PORTSEL y AEN

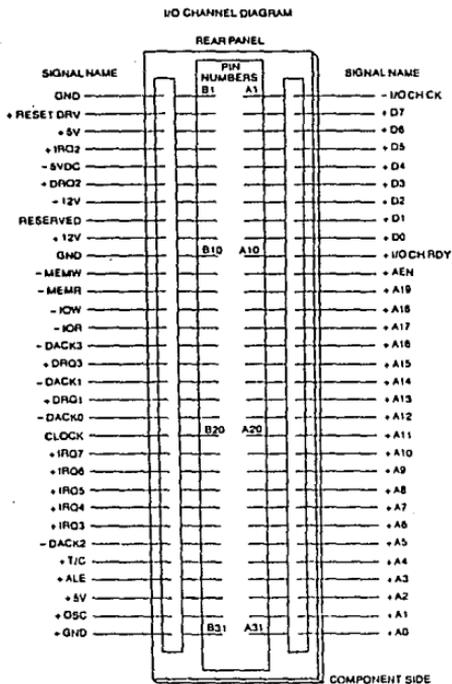


Fig. V.1 Conector de I/O de una IBM PC.

estén al mismo tiempo en un nivel lógico "0".

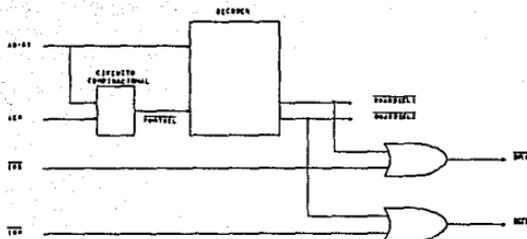


Fig. V.2 Hardware de entrada/salida de una PC.

Habilitador para lectura y escritura (\overline{IOR} , \overline{IOW}) de la computadora al o del dispositivo periférico.

Estas señales son etiquetadas como \overline{IOR} , \overline{IOW} , y ambas son activas bajas, en el caso de \overline{IOR} , cuando es "0" la PC lee el dato de puerto, y si \overline{IOW} es "0" la PC escribe un dato o saca un dato por el puerto de salida.

Habilitador de lectura y escritura del dispositivo periférico.

Estas señales de \overline{ORS} y \overline{OWS} nos permiten hacer una lectura o escritura del dato al puerto de entrada/salida del circuito. Las cuales son

generadas para el caso de la primera, con la señal de \overline{IOR} y la señal de $\overline{BOARDSEL1}$. Cuando ambas señales la IOR y la $\overline{BOARDSEL1}$ están en un nivel "0" se genera la señal de \overline{BRS} . Para la segunda señal es similar solo que en este caso en vez de ser \overline{IOR} es \overline{IOW} , la cual generará la señal \overline{BWS} . Las señales \overline{BRS} y \overline{BWS} se muestran como se generan en la figura V.2 .

Latch de Salida.

Es la parte final del hardware de entrada/salida. Estos latches tienen la función de almacenar temporalmente el dato, enviado por la PC hacia el circuito externo (ver figura V.3).

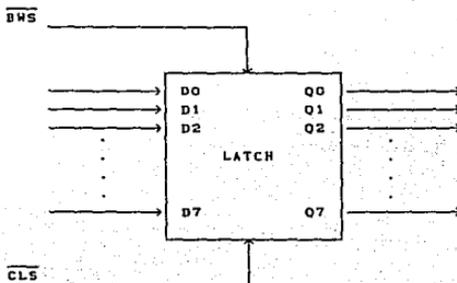


FIG V.3 Latch de Salida

V.2.2. Acceso Directo a Memoria.

El acceso directo a memoria, o DMA, es un método que consiste en la transferencia de bloques de datos de un puerto a memoria, o de memoria a puerto, sin tener que pasar por el microprocesador. Para controlar la transferencia de información de un periférico a la memoria de una PC se utiliza un circuito integrado llamado controlador de DMA.

Un controlador de DMA, toma temporalmente el bus de direcciones, el bus de datos y el bus de control del microprocesador, transfiriendo los datos directamente desde el puerto, hasta una serie de localidades de memoria.

En un controlador de DMA, la lógica incluye un registro de dirección, que almacena la posición a partir de la cual se almacenarán los datos en la memoria y un contador de bytes que almacena el número de bytes que deben transferirse. Tanto la dirección, el número de bytes a transferir y la indicación de si la transferencia es una entrada o una salida, deben preceder a la transferencia del DMA.

Los pasos en una transferencia son :

1. El controlador de DMA envía la solicitud a la línea DMA del μP .
2. El μP responde con una señal de reconocimiento de DMA y coloca los buses de direcciones, datos y control en alta impedancia.
3. a) Para leer de la memoria : el controlador de DMA transmite la dirección y un pulso de lectura a la memoria.
b) Para escribir en la memoria : el controlador DMA envía la dirección, y un pulso de escritura a la memoria.

4. Después que se ha transferido cada byte de memoria, se incrementa el registro de direcciones y el contador del número de bytes deseado se decrementa.
5. Después de completarse la transferencia, el controlador DMA retira la solicitud de DMA, regresando el control de los buses al μP .

Algunos controladores de DMA además de transferir datos de memoria a puerto, o de puerto a memoria también pueden transferir de memoria a memoria.

Un controlador de DMA comúnmente usado es el CI 8237, que se puede encontrar en una computadora IBM PC, PC/XT o compatibles. La figura V.4 muestra como el controlador de DMA ejecuta su función. Cuando la computadora PC se enciende, los interruptores están en la posición donde todos los buses (Datos, Direcciones y Control), están conectados del microprocesador a memoria y periféricos.

En esta misma figura se muestra un controlador de DMA, donde un dispositivo periférico envía una solicitud de DMA (DMAREQ) al controlador de DMA; si el canal de entrada del controlador no está enmascarado (no está deshabilitado), éste enviará una petición de HOLD (HREQ) a la entrada (HOLD) del microprocesador. El microprocesador responderá a ésta entrada enviando un reconocimiento (HLDA) al controlador de DMA.

Cuando el controlador recibe la señal HLDA pone los buses y los interruptores en la posición con el DMA, desconectando al procesador de sus buses. El controlador de DMA entonces pone en el bus de direcciones, la dirección de memoria donde se requiere enviar el dato del periférico. Después el controlador envía un reconocimiento de DMA (DACK) al

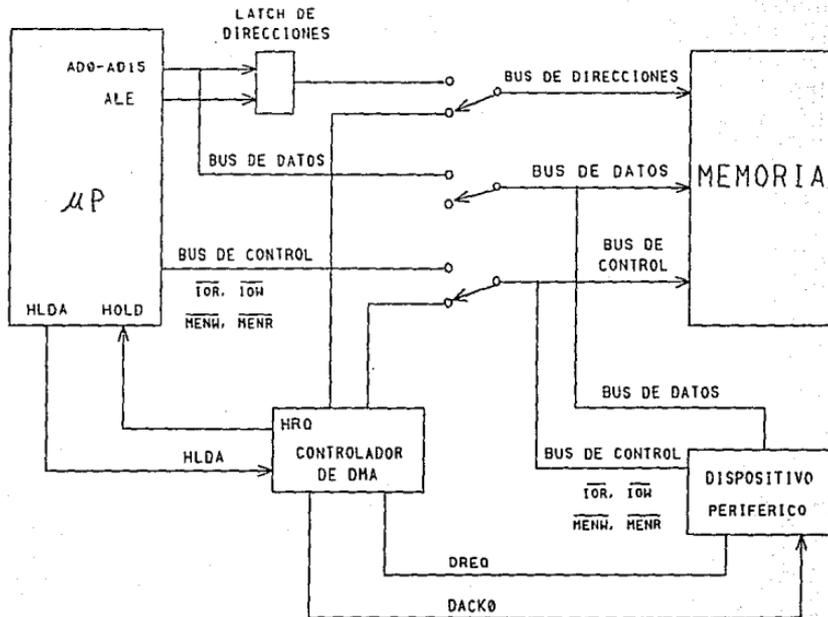


FIG.V.4 DIAGRAMA DE BLOQUES DE UN CONTROLADOR DE DMA OPERANDO EN UN SISTEMA

dispositivo periférico comunicándole que está listo para transferir el dato. Para nuestro caso el controlador de DMA tiene cuatro canales de DMAREQ, donde el DMAREQ0 es utilizado para el refresco de memoria, el DMAREQ2 es para lectura y escritura en disco duro, y en nuestro caso nosotros utilizamos el DMAREQ1 para controlar el sistema de conversión.

De la hoja de especificaciones del CI 8237 (apéndice) se puede decir que para inicializar al controlador de DMA se tienen los siguientes pasos:

- 1.- Se transfiere a la dirección (0CH = 00001100) el valor de 1 el cual da un clear al flip-flop de borrado a cero (para decirle al DMA que en la transferencia lea primero la parte baja y después la parte alta).
- 2.- Se envía a la dirección (0DH = 00001101) un reset maestro " 1 ".
- 3.- Se transfiere el modo a utilizar del canal de DMA (lectura o escritura). Esto es, se envía a la dirección (0BH = 00001011) el número 45H al registro de modo, para realizar una lectura (ver apéndice).
- 4.- Se transfiere por la dirección interna del DMA (02H = 00000011), el valor de la dirección de inicio del bloque de memoria que se va a transferir.
- 5.- Se transfiere por la dirección (03H = 00000011) el número de bytes que se va a transferir, al contador interno de bytes.
- 6.- Se envía el número de página de memoria que se va a transferir.
- 7.- Se da un clear al canal que se encuentra enmascarado. Esto es se envía por la dirección (0AH = 00001010) el valor de 1.

Es importante mencionar que por cada página, se programa el DMA y se manda un clear al flip flop de borrado a cero, verificando que el contador de bytes haya completado toda la transferencia de los datos.

V.2.3 Diseño y Realización de la Interfase Sistema CODEC - PC.

El diseño de la interfase entre la PC y el sistema CODEC se basó en la técnica de DMA, la cual nos permite una mayor rapidez en la transferencia de datos sin tener que pasar por el μ P. En ésta interfase se utilizó la decodificación de puerto siguiente, la dirección "EO" para leer un dato del sistema CODEC y la dirección "CO" para escribir un dato en la misma.

La interfase que se diseñó es muy flexible, ya que nos permite variar la frecuencia de muestreo. Esto se logra al programar los contadores de la base de tiempo (cargar un número "N" en las entradas de carga en paralelo de los contadores), el número "N" que se carga se obtiene a partir de la Ecuación 4.6. Esta programación se logra al decodificar la dirección "CO" que es la señal que habilita al decoder 74LS138 (U1). Este decoder genera la señal de COUNT STRB logrando así realizar la carga del número "N" en el latch 74LS373 (U10) el cual a su vez, es el valor de la carga en paralelo de los contadores (U17 U18).

El modo de lectura se realiza una vez que ha sido programado la frecuencia de muestreo y el DMA, posteriormente se decodifica la dirección (TIMER1 = EO) y a la vez se envía el dato (READ = 2), generando el DREQ1 que se sincroniza con la señal XSYNC, a través del 74LS74 (U7), los cuales a su vez generarán las señales de control para la comunicación entre la PC y la sistema CODEC. De manera similar al modo de lectura, en

el de escritura se requiere que primero la frecuencia de muestreo sea programada así como el DMA, después se decodifica la dirección (TIMER = CO) y paralelamente se envía el dato (WRITE = 3) para que se genere el DREQ1, y sincronizar con XSYNC, para así realizar la comunicación y la transferencia de datos (ver figura V.5).

V.2.4 Programación de la Interfase Sistema CODEC - PC.

La programación de la interfase consta de 2 partes que son :

- Programación para la conversión A/D.
- Programación para la conversión D/A.

V.2.4.1 Programación para la conversión A/D.

En esta parte se programa el DMA y el sistema CODEC para la adquisición y envío de datos hacia la PC. Este programa permite a la PC, llenar un número de páginas que le es proporcionado por el usuario, a partir de la página 2. Donde una página es un bloque de memoria igual 64536 bytes. La razón por la cual se empieza de la página 2 es que en la primera página se encuentra almacenado el sistema operativo y el programa ejecutable para cada una de las etapas de conversión. En el diagrama de flujo (figura V.6) se muestra la conversión A/D.

La capacidad de almacenamiento de la señal digitalizada en páginas, se calculó tomando en cuenta la memoria RAM del sistema que se tiene en una PC con 512 Kbytes de RAM. Este cálculo se muestra a continuación.

524,288.00	Bytes de RAM	(8 páginas)
- 38,992.00	Bytes del SISTEMA	(almacenados en la 1 página).
- <u>22,844.00</u>	Bytes del PROGRAMA (anadig)	
462,452.00	Bytes de MEMORIA DISPONIBLE	
+ <u>65,536.00</u>	Bytes de TAMAÑO DE PAGINA	
7.05	# de páginas disponibles	

Para un ejemplo de 6 páginas, esto es 393,216 Bytes (6 x 64k Bytes) de memoria, se tiene que el tiempo necesario para almacenar estas páginas se calcula de la siguiente manera :

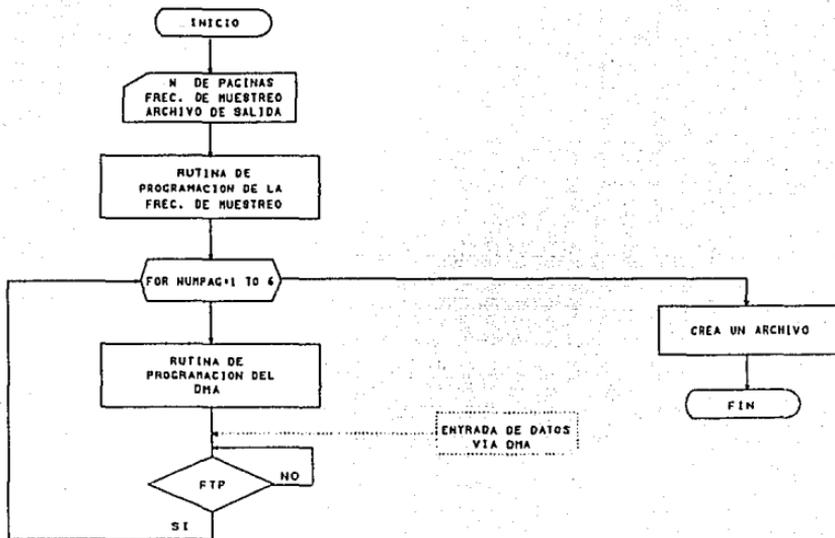
6 paginas = 393,216 bytes 1 muestra = 1 byte
 si se supone $F_{\text{muestreo}} = 8000 \text{ Hz} = 8000 \text{ muestras/seg}$
 entonces tenemos :

$$\text{TALMACENAMIENTO} = \frac{393,216 \text{ muestras}}{8000 \text{ muestras/seg}} = 49.152 \text{ seg.}$$

V.2.4.2 Programación para la conversión D/A.

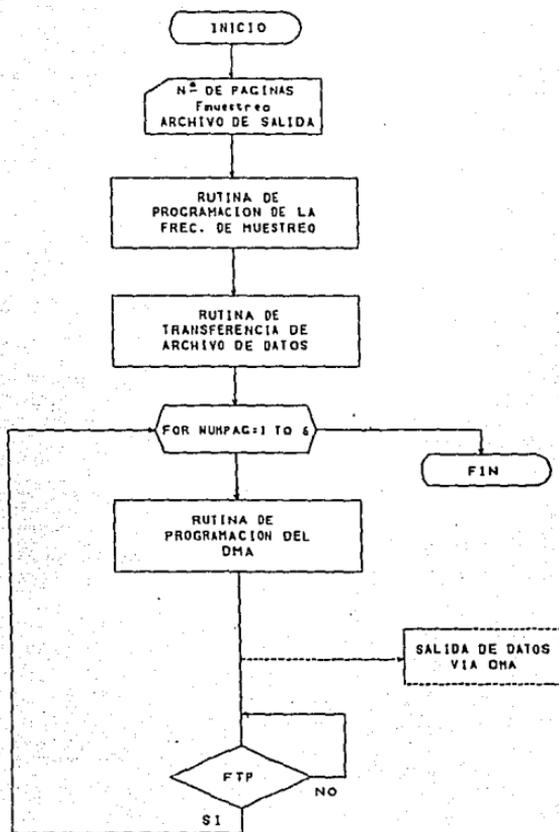
Este programa nos permite leer datos del disco y llenar un número de páginas a partir de la página 2, y luego las envía a la tarjeta CODEC para realizar la conversión D/A.

El diagrama de flujo de esta parte se presenta en la figura V.7 .



• FTP : Fin de transferencia de página.

Fig. V.6 Diagrama de flujo para la conversión A/D.



FTP : Fin de transferencia de página.

Fig. V.7 Diagrama de flujo para la conversión D/A.

V.3 INTERFASE SISTEMA CODEC - TMS32010.

V.3.1 El Microprocesador TMS32010.

El TMS32010 es el primer miembro de la familia TMS320 de procesadores de señales digitales, diseñados para soportar altas velocidades y aplicaciones numéricas intensivas. Este microprocesador de 16 bits combina la flexibilidad de un controlador de alta velocidad con la capacidad numérica de un microprocesador de arreglos.

La familia TMS320 contiene la primera microcomputadora MOS capaz de ejecutar más de 6 millones de instrucciones por segundo.

V.3.2 Características del Microprocesador TMS32010.

Estas son las siguientes :

- Arquitectura Harvard
- Pipelining extensivo
- Hardware dedicado a multiplicaciones
- Instrucciones especiales para DSP
- Ciclo de instrucciones rápidas

Arquitectura Harvard.

El TMS32010 utiliza una arquitectura Harvard modificada la cual le da una flexibilidad y una gran velocidad. En una arquitectura Harvard, el programa y la memoria de datos se almacenan en 2 memorias separadas,

permitiendo una coincidencia total en instrucciones *FETCH* y *EXECUTE*. Las modificaciones de la arquitectura Harvard de la familia TMS320 permite mayor transferencia entre la memoria de datos y el programa, logrando así una mayor flexibilidad en el dispositivo. Esta modificación de la arquitectura elimina la necesidad de separar los coeficientes de la ROM y además maximiza el poder de procesamiento para mantener los dos buses de estructura separada (programa y datos) para una ejecución rápida.

Pipelining Extensivo.

En conjunto con una arquitectura Harvard, el "pipelining" es usado extensamente para reducir el tiempo de ciclo de instrucción a un mínimo absoluto, y así incrementar el "throughput" del procesador. Entendiéndose por "throughput" al trabajo útil por unidad de tiempo, esto es, las instrucciones que verdaderamente sirven para la solución del programa por unidad de tiempo. El pipeline puede ser cualquiera de 2 ó 4 niveles, dependiendo de que procesador de la familia es usado. La arquitectura de la familia TMS320 usa 2 niveles de pipeline en su primera generación, 3 niveles de pipeline en su segunda generación y 4 niveles de pipeline para su tercera generación de procesadores. Esto significa que el dispositivo procesa de 2 a 4 instrucciones en paralelo, y cada instrucción esta en un diferente estado en la ejecución, la figura V.8 muestra una operación de pipeline de 3 niveles.

En la operación pipelining el prefetch, la codificación y la ejecución pueden ser manejadas independientemente, permitiendo la ejecución de instrucciones de traslape. Durante cualquier ciclo de

instrucción, tres instrucciones son activadas en cada uno de los estados de terminación. Por ejemplo como la n -ésima instrucción está en prefetch, la instrucción $(n-1)$ está siendo decodificada y la instrucción $(n-2)$ está siendo ejecutada, en general el pipelining es transparente al usuario.

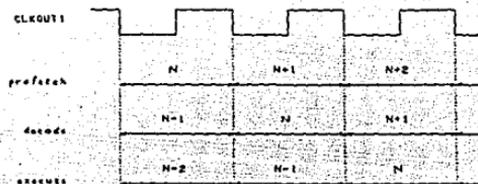


Fig. V.B Operación de un pipelining de tres niveles.

Hardware Dedicado a Multiplicaciones.

Las multiplicaciones juegan un papel importante en el procesamiento digital de señales. La instrucción de multiplicación es construida por una serie de adiciones, por lo que toman muchos ciclos de reloj. En la familia TMS320, las multiplicaciones están en un solo ciclo de instrucción como resultado de un hardware dedicado a multiplicaciones.

Instrucciones Especiales para PDS.

Una de las características de la familia TMS320 es el uso de instrucciones especiales. Dentro del conjunto de instrucciones cuenta con algunas que realizan sumatorias de productos de una manera muy fácil. Tal es el caso de LT, LTD, MPY que son instrucciones importantes para la realización de filtros digitales, transformadas de Fourier, etc. Por otro lado cuenta con instrucciones de un solo ciclo de reloj, que permiten minimizar o reducir el número total de ciclos de instrucción.

Instrucciones de Ciclo Rápido.

La capacidad de procesamiento en tiempo real, puede ser definida como un proceso en donde el retraso creado por el PDS no es notable para el usuario. Esta característica combinado con la capacidad de optimización del circuito integrado diseñado para velocidad, da al dispositivo de PDS, un tiempo de ciclo de instrucción de menos de 200 nseg.

Como se puede ver en la figura V.9, muchos ciclos de instrucción son capaces de procesar señales o generar comandos para aplicación en tiempo real. Por lo tanto, para una simple aplicación de control, un microprocesador de propósito general o un controlador puede ser adecuado para su uso, pero en aplicaciones donde se requiere un uso intensivo de operaciones matemáticas, tales como aplicaciones de robótica o control adaptativo, los procesadores de PDS tienen un mejor uso. El número de ciclos de instrucciones disponibles, es reducido al incrementar la fre-

cuencia de muestreo de 8 KHz, para aplicaciones de telecomunicaciones, a 40-48 KHz, para procesamiento de audio, etc. Para aplicaciones donde se requiere altas velocidades de muestreo, tales como, procesamiento de imagen o video, no hay procesadores de PDS capaces de manejar este procesamiento en tiempo real, por lo tanto, para éstas aplicaciones se requieren múltiples procesadores de PDS.

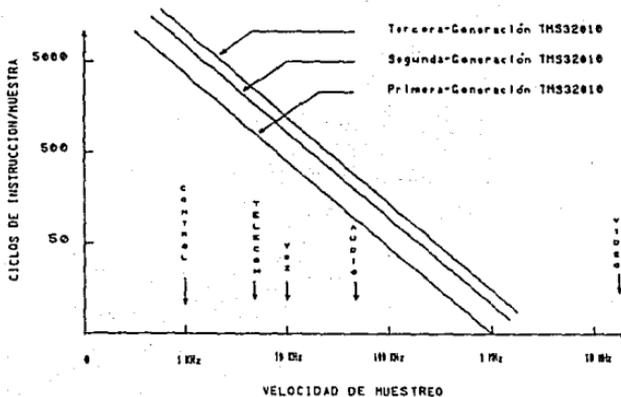


Fig. V.9 Número de ciclos de instrucción/muestra Vs velocidad de muestreo para la familia TMS32010.

V.3.3 Arquitectura del TMS32010.

El TMS32010 utiliza funciones de hardware alambradas que otros procesadores típicos desarrollan en software. Por ejemplo este dispositivo contiene un multiplicador alambrado para realizar multiplicaciones en un solo ciclo de 200 nseg. Además hay una terminal para corrimiento de datos en su camino hacia el ALU. Finalmente, tiene un hardware extra que incluye un registro auxiliar, lo cual provee un direccionamiento indirecto a RAM. Un diagrama de bloques se muestra en la figura V.10.

ALU de 32 bits/ACUMULADOR.

El TMS32010 contiene un ALU de 32 bits y un acumulador que soporta aritmética de doble precisión. El ALU opera en palabras de 16 bits tomadas de la RAM o derivadas de una instrucción anterior, además de ejecutar las operaciones booleanas habituales.

Multiplicador Paralelo de 16 x 16.

El TMS32010 tiene un multiplicador de 16 bits paralelo, alambrado en hardware, que multiplica el registro T con el dato presente en el bus de datos y el resultado se almacena en el registro P de 32 bits. Este multiplicador desarrolla la multiplicación en complemento a 2^{'C} en un ciclo de instrucción de 200 nseg. La rapidez del multiplicador en dicho circuito, permite al TMS32010 desarrollar operaciones fundamentales como convolución, correlación y filtrado a una velocidad mayor de 3 millones de

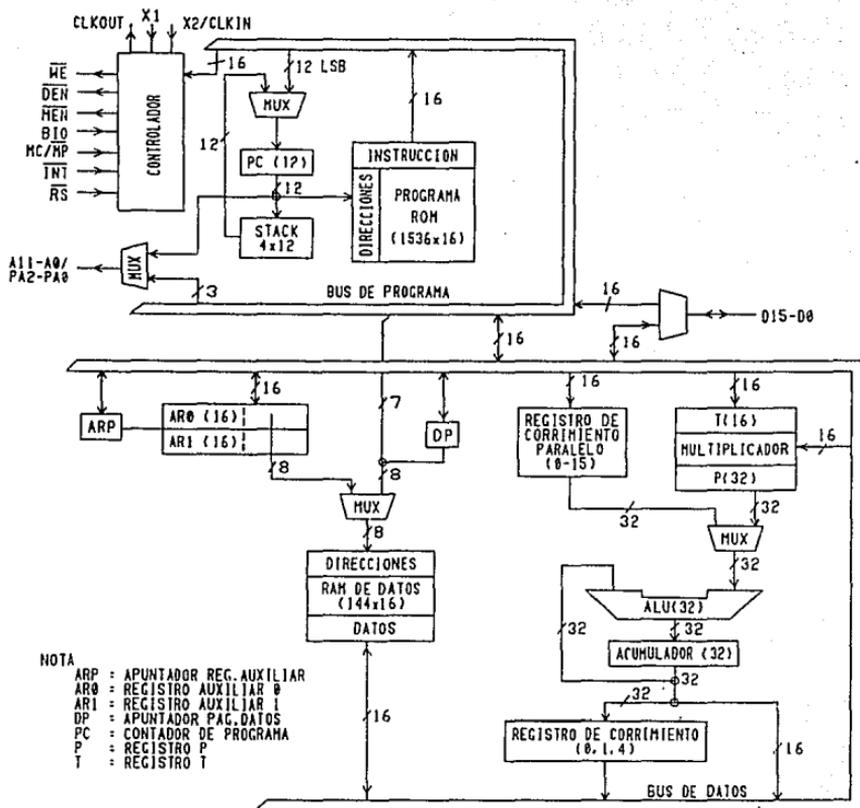


Fig. V.10 Arquitectura de un TMS32010.

Registro de Corrimiento.

Un registro de corrimiento (barril) es disponible para realizar corrimientos de datos a la izquierda 0 a 15 lugares antes de cargarse, sustraídos o sumados del acumulador. Un segundo registro recorre a la izquierda la parte alta del acumulador de 0, 1 ó 4 lugares, mientras está siendo almacenado en RAM.

Entrada/Salida.

El TMS32010 tiene un bus de datos paralelo de 16 bits que puede ser utilizado para desarrollar funciones de I/O a velocidades altas de 50 millones de bits/seg, también tiene la capacidad de disponer de 128 bits de entrada y 128 bits de salida, consistiendo de 8 puertos de entrada y 8 puertos de salida multiplexados con 16 bits cada una, y además tiene una entrada de "polling" para un bit de prueba y operaciones de saltos (BIO) y una terminal de interrupción (INT). También contiene un stack (pila) de 4 niveles para almacenar al contador de programa, durante un salto a subrutina o una atención a una interrupción.

V.3.4 Diseño y Realización de la Interfase Sistema CODEC -TMS32010

El diseño de éste circuito se basó en el hecho de conectar el sistema de conversión A/D - D/A para compresión de y transmisión de señales voz a μ P TMS32010 viendo como puerto al sistema CODEC.

Las señales de control que requiere un TMS32010 para realizar una interfase con un dispositivo periférico se muestran en el apéndice.

Como se puede ver en la figura.V.11, la decodificación de puerto de Entrada/salida es habilitada solo si A3-A11 son ceros lógicos. Esta decodificación la realiza el decoder 74LS138 (U35) . Las líneas de dirección PA0-PA2 son decodificadas dando de 1 a 8 salidas, las cuales van de Y0-Y7 (la dirección de puerto es 0002H).

Así mismo, como se puede ver en la figura V.11, los circuitos U8 U38 son compuertas que dan un retraso a las señales WE y DEN, permitiendo capturar el dato a través de los latches (U27, U28), el cual es proporcionado por el sistema CODEC (memorias). Este dato digitalizado que se encuentra en complemento a 2, es proporcional al valor analógico de entrada del sistema de conversión. Posteriormente este dato pasa por los buffers bidireccionales en camino hacia el sistema en base al μP TMS32010.

Esta interfase inicia su funcionamiento cuando aparece la señal de XSYNC, esta señal habilita el circuito 74ALS74 (U39) que es un circuito de tecnología ALS, es decir, que es un circuito que tiene características buenas en cuanto a velocidad, consumo de potencia, etc., y se emplea por que, debido a las especificaciones del TMS32010 para sincronizar una señal cualesquiera con la señal CLKOUT, se necesita un circuito que tenga cuando menos un retraso de 15 nseg. Como se mencionaba anteriormente este circuito deja pasar la señal de habilitación que se sincroniza con CLKOUT, generando la señal de \overline{BIO} o \overline{INT} que será detectada por el μP , y que a su vez generará la dirección, el dato, y las señales de control de la misma, para que posteriormente en el μP se pueda leer bien un dato en el sistema CODEC . El diseño de la interfase se muestra en la figura V.11

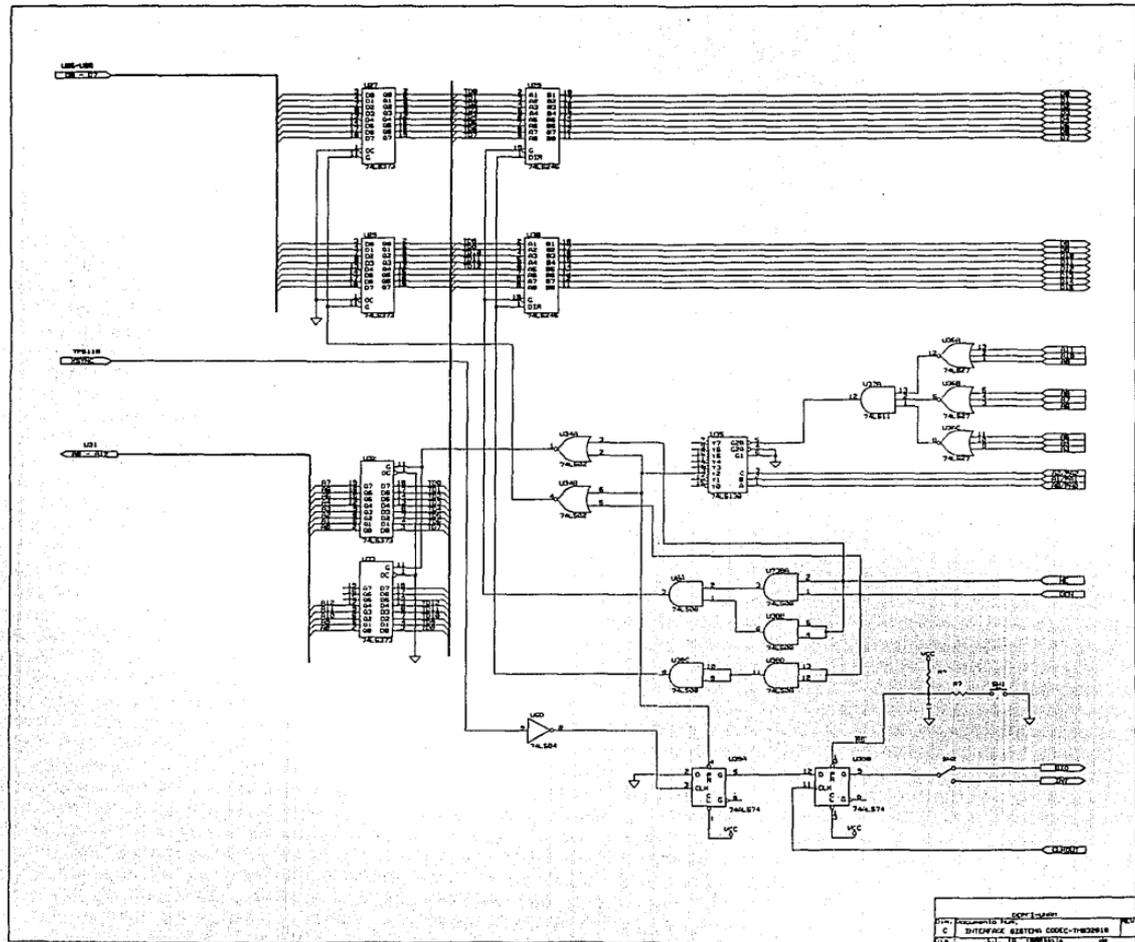


FIG. V.11

174-1
 C
 G

V.4 APLICACIONES

Una vez que se ha digitalizado la señal de voz, independientemente de que los datos esten almacenados en una PC o que estén presentes en un sistema TMS32010 para su procesamiento en tiempo real, a está se le pueden aplicar varias técnicas de procesamiento digital, que dependerá de la aplicación que se le de a ésta. La figura V.12 muestra varias áreas de aplicación de Procesamiento de Señales de Voz.



Fig. V.12 Aplicaciones de procesamiento de señales de voz.

Transmisión Digital y Almacenamiento de Voz.

Una de las aplicaciones más importantes del procesamiento de voz son los Vocoder o codificadores de voz. El propósito del vocoder es reducir el ancho de banda requerido para la transmisión de señales de voz, la

necesidad de reducir el ancho de banda existe a pesar de que hay sistemas como: los satélites, microondas, sistemas de comunicación óptica, etc, que nos proporcionan un gran ancho de banda.

Sistemas de Síntesis de Voz.

Estos sistemas de síntesis de voz han sido estimulados por la necesidad económica de almacenamiento digital de voz para los sistemas de cómputo de respuesta de voz. Un sistema de cómputo de respuesta de voz es un sistema básicamente digital. Por ejemplo el servicio de información automática, responde con la información deseada con voz, cuando es interrogado por una persona a través de un teclado o una terminal.

Sistema de Verificación e Identificación de Locutor.

Las técnicas de identificación y verificación de locutor consisten, como su nombre lo dice, en identificar la voz de un locutor. Un sistema de verificación de locutor debe decidir si un locutor es la persona que ha sido interrogada. Estos sistemas son de gran aplicación en sistemas de seguridad y en áreas donde se requiere información confidencial.

Sistema de Reconocimiento de Voz.

Estos sistemas de reconocimiento de voz son en forma general, una conversión de una señal acústica a un código equivalente de el mensaje de información, los problemas que se presentan dependen de las restricciones

del locutor (timbre de voz e intensidad de la misma). Estos sistemas son aplicables, cuando se requiere una comunicación directa con algun dispositivo. Un ejemplo de este sistema consiste en controlar con la voz una silla de ruedas (velocidad y dirección) o una computadora, de tal manera que ejecute ciertas tareas a partir de ciertas ordenes que le de el usuario.

Mejoramiento de calidad de señal.

En muchas situaciones, las señales de voz son degradadas y deterioradas en sus transmisión. En tales casos las técnicas de procesamiento digital de señales pueden ser aplicadas para mejorar la calidad de la señal. Ejemplos donde se aplican técnicas de procesamiento digital son la eliminación de reverberación o ecos de voz, eliminación de ruido, etc.

VI. CONCLUSIONES Y PERSPECTIVAS.

De la experiencia que se obtuvo durante el desarrollo de este trabajo, se observó que las señales de voz tienen la mayor cantidad de información en las amplitudes pequeñas, razón por la cual se decidió utilizar la técnica de cuantización no uniforme en el desarrollo del sistema de conversión A/D - D/A para compresión y transmisión de señales de voz, valiéndose para ello del circuito CODEC (TP5116), que tiene implantada la ley $\mu 255$ de compansión . Por otro lado, de los resultados obtenidos en el capítulo III, observamos que para una relación señal a ruido de 30 dB, la técnica de cuantización no uniforme nos representa en 7 bits la señal de voz de una manera eficiente y sin mayor pérdida de fidelidad, mientras que para obtener estos mismos resultados utilizando la técnica de cuantización uniforme necesitaríamos 12 bits para su representación. Se concluye entonces que, la técnica empleada proporciona un ahorro en la cantidad de memoria al tratar de almacenar la información.

Es importante mencionar que este sistema de conversión nos da una gran flexibilidad al brindarnos la posibilidad de poder conectarnos a una PC o a un sistema en base a un μP TMS32010, en donde se aprovecha la capacidad de almacenamiento de la primera y la velocidad de procesamiento en tiempo real de la segunda.

Cabe mencionar que la transferencia de datos entre el sistema de conversión y la PC fue realizado vía DMA, permitiendo hacer una transferencia de información más rápida en comparación a las convencionales (interrupciones o polling), ya que esta transferencia de datos de memoria a puerto o de puerto a memoria se realiza sin tener que pasar por el μP de

la PC. Así mismo, en el diseño de la interfase del sistema con el TMS32010 se dio la opción de poder escoger la técnica de transferencia deseada, esto es, por interrupciones (INT) o por polling (BIO).

Ya que los circuitos CODECS fueron diseñados especialmente para señales de voz en sistemas PCM, estos tienen una gran ventaja en la transmisión de estas señales en comparación con los sistemas que tienen una cuantización uniforme (12 bits), debido que para la primera solo se transmiten 8 bits mientras que para la segunda se necesitarían transmitir 12 bits, dando como resultado un incremento en el ancho de banda del sistema.

Es importante mencionar que a pesar de que este sistema fue diseñado para procesar señales de voz, mediante algunas modificaciones en la etapa de filtrado (la cual está restringida al rango de una línea telefónica 200 - 3400 Hz) , así como en la base de tiempo en donde se limitó la frecuencia de muestreo a un rango determinado, se puede obtener un sistema con gran flexibilidad que podría ser utilizado en otras áreas, en donde se presentan señales que no tienen una variación uniforme o lineal.

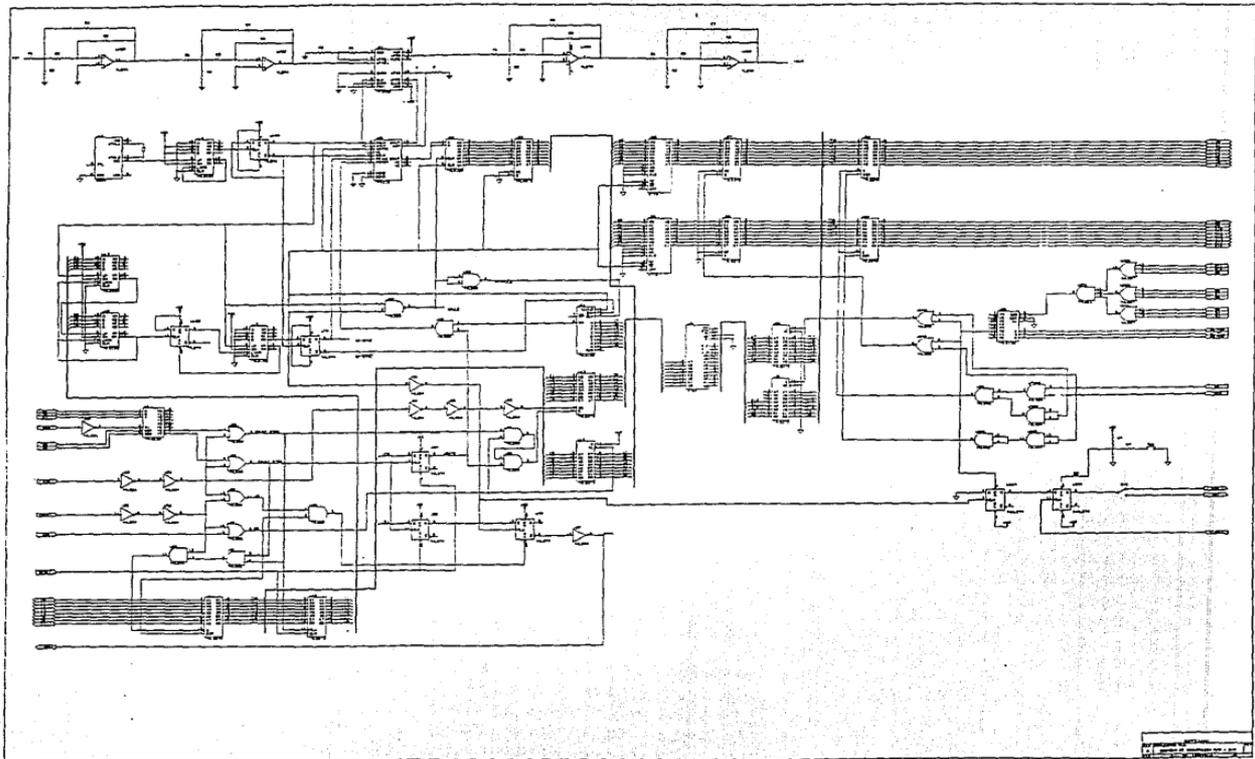
BIBLIOGRAFIA

	Capitulos
[1] Alcántara R. " PROCESAMIENTO DIGITAL DE SEÑALES (Apuntes de la materia)" DEPFI-UNAM, 1988.	III
[2] AMICEE " INSTRUMENTACION DIGITAL " Limusa, 1986.	II
[3] Belamy J.C. " DIGITAL TELEPHONY " Wiley Interscience, 1982.	III, IV
[4] Coffron James W. " THE IBM PC CONNECTION " Sibex, Berkeley, 1984.	IV
[5] CORONA PC TECHNICAL MANUAL Corona Data System Inc., EUA, 1983.	IV
[6] Hall Douglas V. " MICROPROCESOR AND INTERFACING (Programing and Hardware) " McGraw-Hill 1986.	IV
[7] Hnatek Eugene R. " USER'S HANDBOOK OF D/A AND A/D CONVERTERS " John Wiley & Sons, New York 1976.	II
[8] Hoeschele David F. " A/D , D/A CONVERSION TECHNIQUES " John Wiley & sons, Inc. 1976.	II

- [9] MEMORY COMPONENTS HANDBOOK IV, V
Intel, Santa Clara, C.A. 1986.
- [10] MICROSYSTEM COMPONENTS HANDBOOK, VOL. I IV, V
Intel, Santa Clara, C.A. 1986.
- [11] Kun-Shan, I. I, III, V
"DIGITAL SIGNAL PROCESSING APPLICATIONS WITH THE TMS320
FAMILY "
Prentice Hall, Englewood Cliffs, New Jersey, 1987.
- [12] Kun-Shan, L., Frantz, G.E. and Sinar, Ray Jr. III, V
" THE TMS320 FAMILY OF DIGITAL SIGNAL PROCESSORS "
Proceedings of the IEEE, Sep. 1987, Vol 75 pp 1143-1159.
- [13] LINEAR DATABOOK IV, V
National semiconductor Corporation, 1984.
- [14] Papamichalis, Panos E. I, III, IV
" PRACTICAL APPROACHES TO SPEECH CODING "
Prentice Hall, Englewood Cliffs, New Jersey, 1987.
- [15] Rabiner L.R. and Shafer R.W. I, III, V
" DIGITAL PROCESSING OF SPEECH SIGNAL "
Prentice Hall, Englewood Cliffs, New Jersey, 1978.
- [16] Shanmugam Sam K. III
" DIGITAL ANALOG COMMUNICATION SYSTEM "
John Wiley & Sons 1985.
- [17] Sheingold Daniel H. II
" ANALOG-DIGITAL CONVERSION HANDBOOK "
Prentice Hall, Englewood Cliffs, New Jersey 1986.

- [18] " TMS32010 ANALOG INTERFACE BOARD USER'S GUIDE " V
Texas Instruments, 1984, Huston, Tx .
- [19] " TMS32010 USER'S GUIDE " V
Texas Instruments, 1984, Houston, Tx .
- [19] THE TTL DATA BOOK (for Design Engineers) IV, V
Texas Instruments.
- [20] LINEAR INTERFACING (Handbook) IV, V
Texas instruments.

APENDICE



Copyright © 1975 by General Electric Company



**National
Semiconductor**

TP5116A, TP5117A, TP5156A Monolithic CODECs

General Description

The TP5116A, TP5117A and TP5156A are monolithic PCM CODECs implemented with double poly CMOS technology. The TP5116A and TP5117A are intended for μ law applications and the TP5156A is for A law applications. The TP5117A has a DJ compatible format for line card compatibility with the TP5156A.

Each device contains separate OIA and A/D circuitry, all necessarily sample and hold capacitors, and internal auto zero circuits. Each device also contains a precision internal voltage reference, eliminating the need for an external reference. There are no internal connections to pins 15 or 16, making them directly interchangeable with CODECs using external reference components.

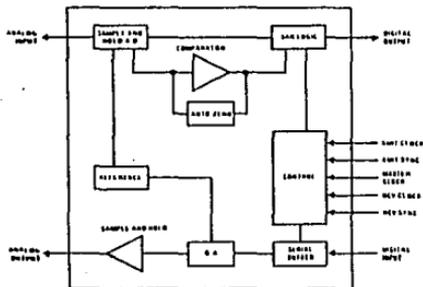
All devices are intended to be used with the TP3040 monolithic PCM filter which provides the input anti-aliasing function for the encoder and smooths the output

of the decoder and corrects for the sin/x distortion introduced by the decoder sample and hold output.

Features

- Low operation power — 50 mW typical
- \pm 5V operation
- TTL compatible digital interface
- Internal precision reference on TP5116A, TP5117A and TP5156A
- Internal sample and hold capacitors
- Internal auto zero circuit
- TP5116A — μ law coding (sign plus magnitude format)
- TP5117A — μ law, DJ compatible format
- TP5156A — A law coding
- Synchronous or asynchronous operation

Simplified Block Diagram



Connection Diagram



Absolute Maximum Ratings

Operating Temperature	-25°C to +125°C
Storage Temperature	-65°C to +150°C
V ⁺ with Respect to DIGITAL GROUND	7V
V ⁺ with Respect to V ⁻	14V
V ⁻ with Respect to DIGITAL GROUND	-7V
Voltage at Any Input or Output	V ⁻ - 0.3V to V ⁺ + 0.3V

DC Electrical Characteristics

Unless otherwise noted T_A = 0°C to 70°C, V⁺ = 5.0V ± 5%, V⁻ = -5.0V ± 5%. Typical characteristics are specified at V⁺ = 5.0V, V⁻ = -5.0V and T_A = 25°C. All digital signals are referenced to DIGITAL GROUND. All analog signals are referenced to ANALOG GROUND.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
DIGITAL INTERFACE						
I _I	Input Current	0V < V _{IN} < V ⁺	-10		10	μA
V _{IL}	Input Low Voltage				0.6	V
V _{IH}	Input High Voltage		2.2			V
V _{OL}	Output Low Voltage	I _{OL} = 3.2 mA			0.4	V
V _{OH}	Output High Voltage	I _{OH} = 6 mA	2.4			V
ANALOG INTERFACE						
Z _I	Analog Input Impedance when Sampling	Resistance in Series with Approximately 70 pF	2			Ω
Z _O	Output Impedance at Analog Output			10	20	Ω
I _{IN}	Analog Input Bias Current	V _{IN} = 0V	-0.1		0.1	μA
R1 × C1	DC Blocking Time Constant		4.0			ms
C1	DC Blocking Capacitor		0.1			μF
R1	Input Bias Resistor				50	Ω
POWER DISSIPATION						
I _{CC1}	Operating Current, V _{CC}			3.5	7.0	mA
I _{BB1}	Operating Current, V _{BB}			3.5	7.0	mA
AC Electrical Characteristics						
Unless otherwise noted, the analog input is a 0 dBm _r , 1.02 kHz sine wave. The DIGITAL INPUT is a PCM bit stream generated by passing a 0 dBm _r , 1.02 kHz sine wave through an ideal encoder. All output levels are sin x/c corrected.						
Symbol	Parameter	Conditions	Min	Typ	Max	Units
	Absolute Level	The nominal 0 dBm _r levels for the TP5116A and TP5117A are 1.227 Vrms and 1.231 Vrms for the TP5156A. The resulting nominal overload level is 2.5V peak for all devices. All gain measurements for the encode and decode portions of the devices are based on these nominal levels after the necessary sin x/c corrections are made.				
G _{RA}	Receive Gain, Absolute	T _A = 25°C, V ⁺ = 5V, V ⁻ = -5V	-0.1		0.1	dB
G _{RAT}	Absolute Receive Gain Variation with Temperature	T _A = 0°C to 70°C	-0.05		0.05	dB
G _{RAV}	Absolute Receive Gain Variation with Supply Voltage	V ⁺ = 5V ± 5%, V ⁻ = -5V ± 5%	-0.07		0.07	dB

AC Electrical Characteristics (Continued)

 Unless otherwise noted, the analog input is a 0 dBm0, 1.02 kHz sine wave. The DIGITAL INPUT is a PCM bit stream generated by passing a 0 dBm0, 1.02 kHz sine wave through an ideal encoder. All output levels are sin π/x corrected.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
G_{JA}	Transmit Gain, Absolute	$T_A = 25^\circ\text{C}$, $V^+ = 5\text{V}$, $V^- = -5\text{V}$	-0.1		0.1	dB
G_{JAT}	Absolute Transmit Gain Variation with Temperature	$T_A = 0^\circ\text{C}$ to 70°C	-0.05		0.05	dB
G_{JAV}	Absolute Transmit Gain Variation with Supply Voltage	$V^+ = 5\text{V} \pm 5\%$, $V^- = -5\text{V} \pm 5\%$	-0.07		0.07	dB
G_{RL}	Absolute Receive Gain Variation with Level	CCITT Method 2 Relative to -10 dBm0 0 dBm0 to 3 dBm0 -40 dBm0 to 0 dBm0 -50 dBm0 to -40 dBm0 -55 dBm0 to -50 dBm0	-0.3 -0.2 -0.4 -1.0		0.3 0.2 0.4 1.0	dB
G_{AL}	Absolute Transmit Gain Variation with Level	CCITT Method 2 Relative to -10 dBm0 0 dBm0 to 3 dBm0 -40 dBm0 to 0 dBm0 -50 dBm0 to -40 dBm0 -55 dBm0 to -50 dBm0	-0.3 -0.2 -0.4 -1.0		0.3 0.2 0.4 1.0	dB
SD_A	Receive Signal to Distortion Ratio	Sinusoidal Test Method Input Level -30 dBm0 to 0 dBm0 -40 dBm0 -45 dBm0	35 29 25			dBc dBc dBc
SD_A	Transmit Signal to Distortion Ratio	Sinusoidal Test Method Input Level -30 dBm0 to 0 dBm0 -40 dBm0 -45 dBm0	35 29 25			dBc dBc dBc
N_A	Receive Idle Channel Noise	$D_A = \text{Steady State PCM Code}$			0	dBm0
N_A	Transmit Idle Channel Noise	TPS116A, TPS117A, $V_F = 0\text{V}$ (No Signaling) TPS156A, $V_F = 0\text{V}$			13	dBm0p
HD_A	Receive Harmonic Distortion	2nd or 3rd Harmonic			-47	dB
HD_A	Transmit Harmonic Distortion	2nd or 3rd Harmonic			-47	dB
$PPSR_A$	Positive Power Supply Rejection, Transmit	Input Level = 0V, $V_{CC} = 5.0\text{V}_{DC}$ +20 mVrms, $f = 1.02\text{kHz}$	50			dB
$PPSR_A$	Positive Power Supply Rejection, Receive	$D_A = \text{Steady PCM Code}$, $V_{CC} = 5.0\text{V}_{DC}$ + 20 mVrms, $f = 1.02\text{kHz}$	40			dB
$NPSR_A$	Negative Power Supply Rejection, Transmit	Input Level = 0V, $V_{SS} = -5.0\text{V}_{DC}$ +20 mVrms, $f = 1.02\text{kHz}$	50			dB
$NPSR_A$	Negative Power Supply Rejection, Receive	$D_A = \text{Steady PCM Code}$, $V_{SS} = -5.0\text{V}_{DC}$ + 20 mVrms, $f = 1.02\text{kHz}$	45			dB
CT_{AR}	Transmit to Receive Crosstalk	$D_A = \text{Steady PCM Code}$			-75	dB
CT_{AR}	Receive to Transmit Crosstalk	Transmit Input Level = 0V			-70	dB

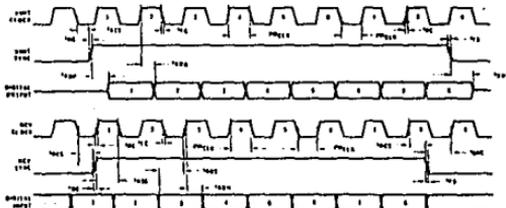
Timing Specifications Unless otherwise noted, $T_A = 0^\circ\text{C to } 70^\circ\text{C}$, $V^+ = +5V \pm 5\%$, $V^- = -5V \pm 5\%$. All digital signals are referenced to DIGITAL GROUND and are measured at V_{IH} and V_{IL} as indicated in the Timing Waveforms.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
F_M	MASTER CLOCK Frequency		15	2 048	2.1	MHz
F_S, F_R	XMIT, RCV CLOCK Frequency		0.064	2 048	2.1	MHz
PW_{CLK}	Clock Pulse Width	MASTER, XMIT, RCV CLOCKS	150			ns
t_{RC}, t_{FC}	Clock Rise and Fall Time	MASTER, XMIT, RCV CLOCKS			50	ns
t_{RS}, t_{FS}	Sync Pulse Rise and Fall Time	RCV, XMIT SYNC			50	ns
t_{ACS}, t_{ACA}	Clock to Sync Delay	RCV, XMIT	0			ns
t_{AS}	XMIT SYNC Set-Up Time				150	ns
t_{DQ}	XMIT Data Delay	Load = 100 pF + 2 LS TTL Loads			200	ns
t_{DP}	XMIT Data Present	Load = 100 pF + 2 LS TTL Loads			200	ns
t_{DT}	XMIT Data TRI-STATE*				150	ns
t_{RNC}	RCV CLOCK to RCV SYNC Delay		0			ns
t_{RDS}	RCV Data Set-Up Time		0			ns
t_{RS}	RCV SYNC Set-Up Time				150	ns
t_{RHS}	RCV Data Hold Time		100			ns
t_{XSL}	XMIT SYNC Low Time	64 kHz Operation	300			ns
t_{RSL}	RCV SYNC Low Time	64 kHz Operation	17			(Note 1)

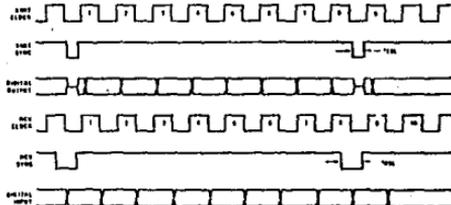
Note 1: RCV SYNC must remain low for 17 cycles of MASTER CLOCK.

Timing Waveforms

72 kHz or Greater Operation



64 kHz Operation



TM 61111 is a registered trademark of National Semiconductor Corp.



TP3040/TP3040A PCM Monolithic Filter

General Description

The TP3040/TP3040A filter is a monolithic circuit containing both transmit and receive filters specifically designed for PCM CODEC filtering applications in 8 kHz sampled systems.

The filter is manufactured using double-poly silicon gate CMOS technology. Switched capacitor integrators are used to simulate classical LC ladder filters which exhibit low component sensitivity.

TRANSMIT FILTER STAGE

The transmit filter is a fifth order elliptic low pass filter in series with a fourth order Chebyshev high pass filter. It provides a flat response in the passband and rejection of signals below 200 Hz and above 3.4 kHz.

RECEIVE FILTER STAGE

The receive filter is a fifth order elliptic low pass filter designed to reconstruct the voice signal from the decoded/demultiplexed signal which, as a result of the sampling process, is a staircase signal having the inherent sin x/sz frequency response. The receive filter approximates the function required to compensate for the degraded frequency response and restore the flat passband response.

Features

- Exceeds all D3/D4 and CCITT specifications
- +5V, -5V power supplies
- Low power consumption:
 - 45 mW (500) 0 dBm load
 - 30 mW (power amps disabled)
- 20 dB gain adjust range
- No external anti aliasing components
- Sin x/sz correction in receive filter
- 5060 Hz rejection in transmit filter
- TTL and CMOS compatible logic
- All inputs protected against static discharge due to handling

Block and Connection Diagrams

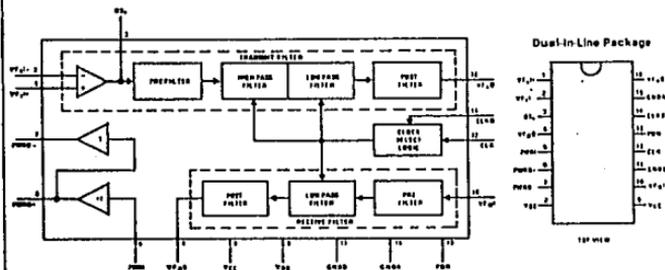


FIGURE 1

Absolute Maximum Ratings

Supply Voltage	± 7V
Power Dissipation	1 W/Package
Input Voltage	± 7V
Output Short Circuit Duration	Continuous
Operating Temperature Range	- 25°C to + 125°C
Storage Temperature	- 65°C to + 150°C
Lead Temperature (Soldering, 10 seconds)	300°C

DC Electrical Characteristics

Unless otherwise noted, $T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5.0\text{V} \pm 5\%$, $V_{BB} = -5.0\text{V} \pm 5\%$, clock frequency is 2.048 MHz. Typical parameters are specified at $T_A = 25^\circ\text{C}$, $V_{CC} = 5.0\text{V}$, $V_{BB} = -5.0\text{V}$. Digital interface voltages measured with respect to digital ground, GNDD. Analog voltages measured with respect to analog ground, GNDA.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
POWER DISSIPATION						
I_{CC}	V_{CC} Standby Current	$PDN = V_{DD}$, Power Down Mode		50	100	μA
I_{BB}	V_{BB} Standby Current	$PDN = V_{DD}$, Power Down Mode		50	100	μA
I_{CC1}	V_{CC} Operating Current	$PWR1 = V_{BB}$, Power Amp Inactive		3.0	4.0	mA
I_{BB1}	V_{BB} Operating Current	$PWR1 = V_{BB}$, Power Amp Inactive		3.0	4.0	mA
I_{CC2}	V_{CC} Operating Current	Note 1		4.6	6.4	mA
I_{BB2}	V_{BB} Operating Current	Note 1		4.6	6.4	mA
DIGITAL INTERFACE						
I_{IC}	Input Current, CLK	$V_{BB} > V_{IH} > V_{CC}$	-10		10	μA
I_{IP}	Input Current, PUN	$V_{BB} > V_{IH} > V_{CC}$	-100			μA
I_{IO}	Input Current, CLK0	$V_{BB} > V_{IH} > V_{CC} - 2\text{V}$	-10		-0.1	μA
V_L	Input Low Voltage, CLK, PDN		0		0.8	V
V_{IH}	Input High Voltage, CLK, PDN		2.2		V_{CC}	V
V_{LO}	Input Low Voltage, CLK0		V_{BB}		$V_{BB} + 0.5$	V
V_{IO}	Input Intermediate Voltage, CLK0		-0.8		0.8	V
V_{HO}	Input High Voltage, CLK0		$V_{CC} - 0.5$		V_{CC}	V
TRANSMIT INPUT OP AMP						
I_{B1}	Input Leakage Current, V_{F1}	$V_{BB} > V_{F1} > V_{CC}$	-100		100	nA
R_{i1}	Input Resistance, V_{F1}	$V_{BB} > V_{F1} > V_{CC}$	10			M Ω
V_{DS1}	Input Offset Voltage, V_{F1}	$-2.5\text{V} \leq V_{IH} \leq +2.5\text{V}$	-20		20	mV
V_{CM}	Common-Mode Range, V_{F1}		-2.5		2.5	V
CMRR	Common-Mode Rejection Ratio	$-2.5\text{V} \leq V_{IH} \leq 2.5\text{V}$	60			dB
PSRR	Power Supply Rejection of V_{CC} or V_{BB}		60			dB
R_{OL}	Open Loop Output Resistance, G_{S1}			1		$\text{k}\Omega$
R_L	Minimum Load Resistance, G_{S1}		10			$\text{k}\Omega$
C_L	Maximum Load Capacitance, G_{S1}			25		pF
V_{OJ}	Output Voltage Swing, G_{S1}	$R_L \leq 10\text{k}$	± 2.5			V
A_{VOL}	Open Loop Voltage Gain, G_{S1}	$R_L \leq 10\text{k}$	5,000			V/V
f_s	Open Loop Unity Gain Bandwidth, G_{S1}			2		MHz

AC Electrical Characteristics

Unless otherwise specified, $T_A = 25^\circ\text{C}$. All parameters are specified for a signal level of 0 dBm0 at 1 kHz. The 0 dBm0 level assumed to be 1.54 Vrms measured at the output of the transmit or receive filter.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
TRANSMIT FILTER (Transmit Filter Input on amp set to the non-inverting unity gain mode, with $V_{F1} = 1.1$ Vrms unless otherwise noted.)						
RL	Minimum Load Resistance, V_{F0}		10			Ω
CL	Load Capacitance, V_{F0}				25	pF
RO	Output Resistance, V_{F0}			1	3	Ω
PSRR1	V_{CC} Power Supply Rejection, V_{F0}	$f = 1$ kHz, $V_{F1} = 0$ Vrms	30			dB
PSRR2	V_{BB} Power Supply Rejection, V_{F0}	Same as Above	35			dB
GA	Absolute Gain	$f = 1$ kHz (TP3040A) $f = 1$ kHz (TP3040)	2.9	3.0	3.125	dB
GR	Gain Relative to GA_{10}	Below 50 Hz 50 Hz 60 Hz 200 Hz (TP3040A) 200 Hz (TP3040) 300 Hz to 3 kHz (TP3040A) 300 Hz to 3 kHz (TP3040) 3.3 kHz 3.4 kHz 4.0 kHz 4.8 kHz and Above	-15 -41 -35 -15 -15 -0.125 -0.15 -0.35 -0.70		-35 -35 -30 0 0.05 0.125 0.15 0.03 -0.1 -14 -32	dB
DA	Absolute Delay at 1 kHz				230	ns
DD	Differential Envelope Delay from 1 kHz to 2.6 kHz				60	ns
DR1	Single Frequency Distortion Products				-48	dB
DR2	Distortion at Maximum Signal Level	0.18 Vrms, 1 kHz Signal Applied to V_{F1} , $\text{Gain} = 20$ dB, $R_L = 10\Omega$			-45	dB
NC1	Total C Message Noise at V_{F0}			2	5	dBm0
NC2	Total C Message Noise at V_{F0}	Gain Setting Op Amp at 20 dB, Non-Inverting, Note 3 $T_A = 0^\circ\text{C}$ to 70°C		3	6	dBm0
GA _T	Temperature Coefficient of 1 kHz Gain			0.0004		dB/°C
GA _B	Supply Voltage Coefficient of 1 kHz Gain	$V_{CC} = 5.0\text{V} \pm 5\%$ $V_{BB} = -5.0\text{V} \pm 5\%$		0.01		dB/V
CTrx	Cross talk, Receive to Transmit $20 \log \frac{V_{F0}}{V_{F1}}$	Receive Filter Output = 2 Vrms $V_{F1} = 0$ Vrms, $f = 0.2$ kHz to 3.4 kHz Measure V_{F0}			-70	dB
GR _L	Gaintracking Relative to GA_{10}	Output Level = +3 dBm0 +2 dBm0 to -40 dBm0 -40 dBm0 to -55 dBm0	-0.1 -0.05 -0.1		0.1 0.05 0.1	dB

AC Electrical Characteristics (Continued)

Unless otherwise specified, $T_c = 25^\circ\text{C}$. All parameters are specified for a signal level of 0 dBm0 at 1 kHz. The 0 dBm0 level is assumed to be 1.54 Vrms measured at the output of the transmit or receive filter.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
RECEIVE FILTER (Unless otherwise noted, the receive filter is preceded by a 6nA/z filter with an input signal level of 1.6 Vrms.)						
I_{B_A}	Input Leakage Current, $\text{VF}_{\mu\text{I}}$	-3.2V \pm V_{IN} \pm 3.2V	-100		100	nA
R_{i_A}	Input Resistance, $\text{VF}_{\mu\text{I}}$		10			M Ω
R_{O_A}	Output Resistance, $\text{VF}_{\mu\text{O}}$			1	3	Ω
C_{L_A}	Load Capacitance, $\text{VF}_{\mu\text{O}}$				25	pF
R_{L_A}	Load Resistance, $\text{VF}_{\mu\text{O}}$		10			k Ω
PSRR_3	Power Supply Rejection of V_{CC} or V_{SS} , $\text{VF}_{\mu\text{O}}$	$\text{VF}_{\mu\text{I}}$ Connected to GND $f = 1 \text{ kHz}$	35			dB
VOS_{DO}	Output DC Offset, $\text{VF}_{\mu\text{O}}$	$\text{VF}_{\mu\text{I}}$ Connected to GND	-200		200	mV
GA_A	Absolute Gain	$f = 1 \text{ kHz}$ (TP3040A) $f = 1 \text{ kHz}$ (TP3040)	-0.1	0	0.1	dB
			-0.125	0	0.125	dB
GR_A	Gain Relative to Gain at 1 kHz	Below 300 Hz 300 Hz to 3.0 kHz (TP3040A) 300 Hz to 3.0 kHz (TP3040) 3.3 kHz 3.4 kHz 4.0 kHz 4.6 kHz and Above	-0.125		0.125	dB
			-0.15		0.15	dB
			-0.35		0.03	dB
			-0.7		-0.1	dB
					-14	dB
					-32	dB
DA_A	Absolute Delay at 1 kHz				100	μs
DD_A	Differential Envelope Delay 1 kHz to 2.6 kHz				100	μs
DP_{N}	Single Frequency Distortion Products	$f = 1 \text{ kHz}$			-48	dB
DP_{P}	Distortion at Maximum Signal Level	2.2 Vrms Input to Sin \times z Filter, $f = 1 \text{ kHz}$, $R_L = 10\Omega$			-45	dB
NC_A	Total C Message Noise at $\text{VF}_{\mu\text{O}}$			3	5	dBm0
GA_T	Temperature Coefficient of 1 kHz Gain			0.0004		dB/°C
GA_{S}	Supply Voltage Coefficient of 1 kHz Gain			0.01		dB/V
CT_{R}	Crosstalk, Transmit to Receive $\text{VF}_{\mu\text{O}}$ 20 log $\text{VF}_{\mu\text{O}}$	Transmit Filter Output = 2.2 Vrms $\text{VF}_{\mu\text{I}} = 0$ Vrms, $f = 0.3 \text{ kHz}$ to 3.4 kHz Measure $\text{VF}_{\mu\text{O}}$			-70	dB
GR_{A}	Gaintracking Relative to GA_A	Output Level = +3 dBm0 +2 dBm0 to -40 dBm0 -40 dBm0 to -55 dBm0 Note 5	-0.1		0.1	dB
			-0.05		0.05	dB
			-0.1		0.1	dB

AC Electrical Characteristics (Continued)

Unless otherwise specified, $T_A = 25^\circ\text{C}$. All parameters are specified for a signal level of 0 dBm at 1 kHz. The 0 dBm level is assumed to be 1.54 Vrms measured at the output of the transmitter or receiver filter.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
RECEIVE OUTPUT POWER AMPLIFIER						
IBP	Input Leakage Current, PWRI	$-3\text{ V} < V_{PW} < 3\text{ V}$	0.1		3	μA
RIP	Input Resistance, PWRI		10			M Ω
ROP1	Output Resistance, PWRO +, PWRO -	Amplifiers Active		1		Ω
CLP	Load Capacitance, PWRO +, PWRO -				500	pF
GA _{P+}	Gain, PWRI to PWRO +	$R_L = 600\Omega$ Connected Between PWRO + and PWRO - , Input Level = 0 dBm (Note 4)		1		V/V
GA _{P-}	Gain, PWRI to PWRO -			-1		V/V
ORL	Gaintracking Relative to 0 dBm Output Level	$V = 2.05\text{ Vrms}, R_L = 600\Omega$ $V = 1.75\text{ Vrms}, R_L = 300\Omega$ (Notes 4, 5)	-0.1		0.1	$^\circ/\text{B}$
SID _P	Signal/Distortion	$V = 2.05\text{ Vrms}, R_L = 600\Omega$ $V = 1.75\text{ Vrms}, R_L = 300\Omega$ (Notes 4, 5)			-45	dB
VOSP	Output DC Offset, PWRO +, PWRO -	PWRI Connected to GND	-50		50	mV
PSHRS	Power Supply Rejection of V_{CC} or V_{BB}	PWRI Connected to GND	45			dB

Note 1: Maximum power dissipation will depend on the load impedance connected to the power amplifier. The typical values listed assume a 100 Ω load connected to the output of the power amplifier.

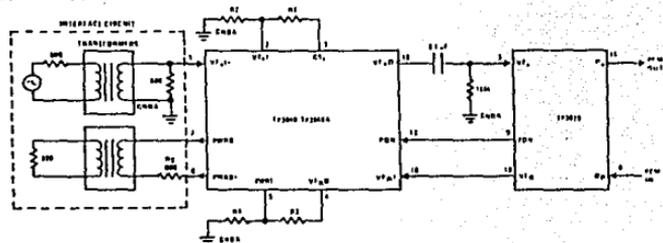
Note 2: Voltage input in receiver mode at 0V, V_{IF} is connected to PWRI. Output is measured from PWRO + to PWRO -.

Note 3: The 0 dBm level for the filter is a signal level of 1.54 Vrms measured at the output of the 100 Ω or 300 Ω filter.

Note 4: The 0 dBm level for the power amplifiers is load dependent. For $R_L = 600\Omega$ in GND, the 0 dBm level is 1.54 Vrms measured at the amplifier output. For $R_L = 300\Omega$ the 0 dBm level is 1.27 Vrms.

Note 5: V_{IF} is connected to PWRI, input signal applied to V_{IF}2.

Typical Application



Note 1: Transmit voltage gain = $\frac{R_1 + R_2}{R_2} \times 2$ (The filter still includes a 10dB gain ($R_1 + R_2 = 10\Omega$))

Note 2: Receive gain = $\frac{R_4}{R_3 + R_4}$
($R_3 + R_4 = 10\Omega$)

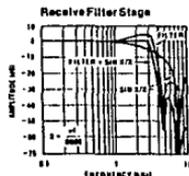
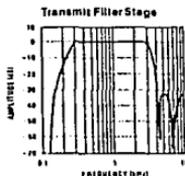
Note 3: In this configuration shown, the receiver filter power amplifiers will drive a 600 Ω load (maximum signal level of 1.54 Vrms). An alternative arrangement, using a transformer coupling network instead of the 100 Ω and 300 Ω resistors, will provide a maximum signal level of 1.27 Vrms to a 100 Ω termination impedance.

FIGURE 2

Description of Pin Functions

Pin No.	Name	Function	Pin No.	Name	Function
1	VF _{I+}	The non-inverting input to the transmit filter stage.	11	GNDD	Digital ground input pin. All digital signals are referenced to this pin.
2	VF _{I-}	The inverting input to the transmit filter stage.	12	CLK	Master input clock. Input frequency can be selected as 2.048 MHz, 1.544 MHz or 1.536 MHz.
3	GS ₁	The output used for gain adjustments of the transmit filter.	13	PDN	The input pin used to power down the TP3040/TP3040A during idle periods. Logic 1 (V _{CC}) input voltage causes a power down condition. An internal pull-up is provided.
4	VF _{F0}	The low power receive filter output. This pin can directly drive the receive port of an electronic hybrid.	14	CLK0	This input pin selects internal counters in accordance with the CLK input clock frequency: CLK Connect CLK0 to: 2048 kHz V _{CC} 1544 kHz GNDD 1536 kHz V _{DD} An internal pull-up is provided.
5	PWRI	The input to the receive filter differential power amplifier.	15	GNDA	Analog ground input pin. All analog signals are referenced to this pin. Not internally connected to GNDD.
6	PWRO+	The non-inverting output of the receive filter power amplifier. This output can directly interface conventional transformer hybrids.	16	VF _O	The output of the transmit filter stage.
7	PWRO-	The inverting output of the receive filter power amplifier. This output can be used with PWRO+ to differentially drive a transformer hybrid.			
8	V _{DD}	The negative power supply pin. Recommended input is -5V.			
9	V _{CC}	The positive power supply pin. The recommended input is 5V.			
10	VF _I	The input pin for the receive filter stage.			

Typical Performance Characteristics





8237A/8237A-4/8237A-5 HIGH PERFORMANCE PROGRAMMABLE DMA CONTROLLER

- Enable/Disable Control of Individual DMA Requests
- Four Independent DMA Channels
- Independent Autoinitialization of all Channels
- Memory-to-Memory Transfers
- Memory Block Initialization
- Address Increment or Decrement
- High performance: Transfers up to 1.8M Bytes/Second with 5 MHz 8237A-5
- Directly Expandable to any Number of Channels
- End of Process Input for Terminating Transfers
- Software DMA Requests
- Independent Polarity Control for DREQ and DACK Signals
- Available in EXPRESS - Standard Temperature Range
- Available in 40-Lead Cerdip and Plastic Packages
(See Packaging Spec. Order #231305)

The 8237A Multimode Direct Memory Access (DMA) Controller is a peripheral interface circuit for microprocessor systems. It is designed to improve system performance by allowing external devices to directly transfer information from the system memory. Memory-to-memory transfer capability is also provided. The 8237A offers a wide variety of programmable control features to enhance data throughput and system optimization and to allow dynamic reconfiguration under program control.

The 8237A is designed to be used in conjunction with an external 8-bit address register such as the 8262. It contains four independent channels and may be expanded to any number of channels by cascading additional controller chips.

The three basic transfer modes allow programmability of the type of DMA service by the user. Each channel can be individually programmed to Autoinitialize to its original condition following an End of Process (EOP).

Each channel has a full 84K address and word count capability.

The 8237A-4 and 8237A-5 are 4 MHz and 5 MHz selected variations of the standard 3 MHz 8237A respectively.

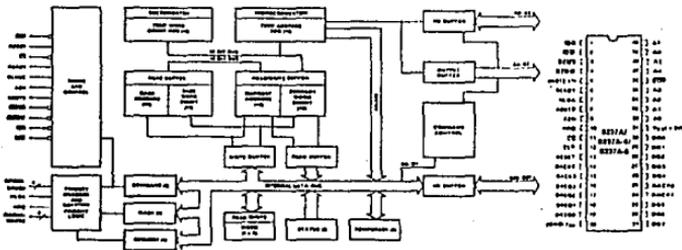
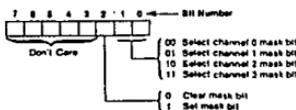


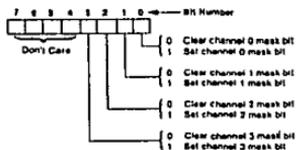
Figure 1. Block Diagram

Figure 2.
Pin Configuration

Mask Register — Each channel has associated with it a mask bit which can be set to disable the incoming DREQ. Each mask bit is set when its associated channel produces an EOP if the channel is not programmed for AutoInitialize. Each bit of the 4-bit Mask register may also be set or cleared separately under software control. The entire register is also set by a Reset. This disables all DMA requests until a clear Mask register instruction allows them to occur. The instruction to separately set or clear the mask bits is similar in form to that used with the Request register. See Figure 5 for instruction addressing.



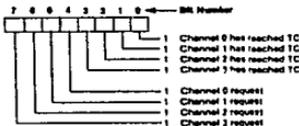
All four bits of the Mask register may also be written with a single command.



Register	Operation	Signals							
		CS	TCR	IOW	A3	A2	A1	A0	
Command	Write	0	1	0	1	0	0	0	0
Mode	Write	0	1	0	0	1	0	1	1
Request	Write	0	1	0	1	0	0	0	1
Mask	Set/Reset	0	1	0	1	0	1	0	0
Mask	Write	0	1	0	1	1	1	1	1
Temporary	Read	0	0	1	1	1	0	1	1
Status	Read	0	0	1	1	0	0	0	0

Figure 5. Definition of Register Codes

Status Register — The Status register is available to be read out of the 8237A by the microprocessor. It contains information about the status of the devices at this point. This information includes which channels have reached a terminal count and which channels have pending DMA requests. Bits 0-3 are set every time a TC is reached by that channel or an external EOP is applied. These bits are cleared upon Reset and on each Status Read. Bits 4-7 are set whenever their corresponding channel is requesting service.



Temporary Register — The Temporary register is used to hold data during memory-to-memory transfers. Following the completion of the transfers, the last word moved can be read by the microprocessor in the Program Condition. The Temporary register always contains the last byte transferred in the previous memory-to-memory operation, unless cleared by a Reset.

Software Commands — These are additional special software commands which can be executed in the Program Condition. They do not depend on any specific bit pattern on the data bus. The three software commands are:

Clear First/Last Flip-Flop: This command is executed prior to writing or reading new address or word count information to the 8237A. This initializes the flip-flop to a known state so that subsequent accesses to register contents by the microprocessor will address upper and lower bytes in the correct sequence.

Master Clear: This software instruction has the same effect as the hardware Reset. The Command, Status, Request, Temporary, and Internal First/Last Flip-Flop registers are cleared and the Mask register is set. The 8237A will enter the idle cycle.

Clear Mask Register: This command clears the mask bits of all four channels, enabling them to accept DMA requests.

Figure 6 lists the address codes for the software commands:

Signals							Operation
A3	A2	A1	A0	TCR	IOW		
1	0	0	0	0	1	Set/Reset Status Register	
1	0	0	0	1	0	Write Command Register	
1	0	0	1	0	1	Request	
1	0	0	1	0	0	Write Mask Register	
1	0	1	0	1	0	Request	
1	0	1	0	1	0	Write Request Mask Register Bit	
1	0	1	1	0	1	Request	
1	0	1	1	1	0	Write Mask Register	
1	1	0	0	0	1	Request	
1	1	0	0	1	0	Clear First/Last Flip-Flop	
1	1	0	0	1	1	Read Temporary Register	
1	1	0	1	0	0	Master Clear	
1	1	0	1	0	1	Request	
1	1	0	1	1	0	Clear Mask Register	
1	1	1	0	1	1	Request	
1	1	1	1	1	0	Write All Mask Register Bits	

Figure 6. Software Command Codes



Channel	Register	Operation	Signals							Internal / B-Flag	Data Bus DB0-DB7	
			ZS	DS	DSW	A3	A2	A1	A0			
0	Base and Current Address	Write	0	1	0	0	0	0	0	0	0	AD-A7 AB-A15
	Current Address	Read	0	0	1	0	0	0	0	0	0	AD-A7 AB-A15
	Base and Current Word Count	Write	0	1	0	0	0	0	0	1	0	WD-W7 WB-W15
	Current Word Count	Read	0	0	1	0	0	0	1	0	0	WD-W7 WB-W15
				0	0	1	0	0	0	1	1	
1	Base and Current Address	Write	0	1	0	0	0	1	0	0	0	AD-A7 AB-A15
	Current Address	Read	0	0	1	0	0	1	0	0	0	AD-A7 AB-A15
	Base and Current Word Count	Write	0	1	0	0	0	1	1	0	0	WD-W7 WB-W15
	Current Word Count	Read	0	0	1	0	0	1	1	0	0	WD-W7 WB-W15
				0	0	1	0	0	1	1	1	
2	Base and Current Address	Write	0	1	0	0	1	0	0	0	0	AD-A7 AB-A15
	Current Address	Read	0	0	1	0	1	0	0	0	0	AD-A7 AB-A15
	Base and Current Word Count	Write	0	1	0	0	1	0	1	0	0	WD-W7 WB-W15
	Current Word Count	Read	0	0	1	0	1	0	1	0	0	WD-W7 WB-W15
				0	0	1	0	1	0	1	1	
3	Base and Current Address	Write	0	1	0	0	1	1	0	0	0	AD-A7 AB-A15
	Current Address	Read	0	0	1	0	1	1	0	0	0	AD-A7 AB-A15
	Base and Current Word Count	Write	0	1	0	0	1	1	1	0	0	WD-W7 WB-W15
	Current Word Count	Read	0	0	1	0	1	1	1	0	0	WD-W7 WB-W15
				0	0	1	0	1	1	1	1	

Figure 7. Word Count and Address Register Command Codes



REGISTER DESCRIPTION

Current Address Register — Each channel has a 16-bit Current Address register. This register holds the value of the address used during DMA transfers. The address is automatically incremented or decremented after each transfer and the intermediate values of the address are stored in the Current Address register during the transfer. This register is written or read by the microprocessor in successive 8-bit bytes. It may also be reinitialized by an Autoinitialize back to its original value. Autoinitialize takes place only after an EOP.

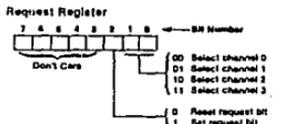
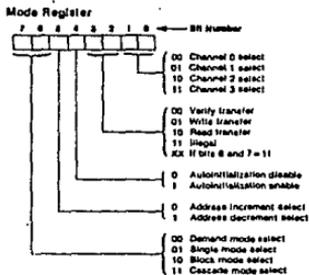
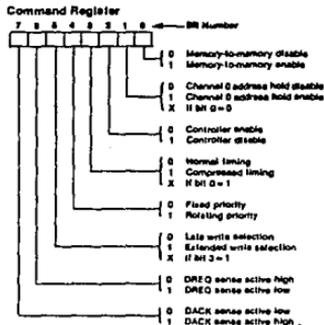
Current Word Register — Each channel has a 16-bit Current Word Count register. This register determines the number of transfers to be performed. The actual number of transfers will be one more than the number programmed in the Current Word Count register (i.e., programming a count of 100 will result in 101 transfers). The word count is decremented after each transfer. The intermediate value of the word count is stored in the register during the transfer. When the value in the register goes from zero to FFFFH, a TC will be generated. This register is loaded or read in successive 8-bit bytes by the microprocessor in the Program Condition. Following the end of a DMA service it may also be reinitialized by an Autoinitialize back to its original value. Autoinitialize can occur only when an EOP occurs. If it is not Autoinitialized, this register will have a count of FFFFH after TC.

Base Address and Base Word Count Registers — Each channel has a pair of Base Address and Base Word Count registers. These 16-bit registers store the original value of their associated current registers. During Autoinitialize these values are used to restore the current registers to their original values. The base registers are written simultaneously with their corresponding current register in 8-bit bytes in the Program Condition by the microprocessor. These registers cannot be read by the microprocessor.

Command Register — This 8-bit register controls the operation of the 8237A. It is programmed by the microprocessor in the Program Condition and is cleared by Reset or a Master Clear instruction. The following table lists the function of the command bits. See Figure 6 for address coding.

Mode Register — Each channel has a 6-bit Mode register associated with it. When the register is being written to by the microprocessor in the Program Condition, bits 0 and 1 determine which channel Mode register is to be written.

Request Register — The 8237A can respond to requests for DMA service which are initiated by software as well as by a DREQ. Each channel has a request bit associated with it in the 4-bit Request register. These are non-maskable and subject to prioritization by the Priority Encoder network. Each register bit is set or reset separately under software control or is cleared upon generation of a TC or external EOP. The entire register is cleared by a Reset. To set or reset a bit, the software loads the proper form of the data word. See Figure 5 for register address coding. In order to make a software request, the channel must be in Block Mode.



ately under software control or is cleared upon generation of a TC or external EOP. The entire register is cleared by a Reset. To set or reset a bit, the software loads the proper form of the data word. See Figure 5 for register address coding. In order to make a software request, the channel must be in Block Mode.

**PROGRAMMABLE
PRODUCTS**

**TMS32010
DIGITAL SIGNAL PROCESSOR**

MAY 1983

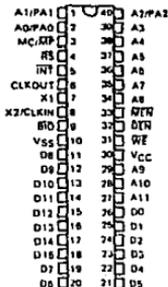
- 200-ns Instruction Cycle
- 288-Byte On-Chip Data RAM
- Microprocessor Version — TMS32010
- Microcomputer Version — TMS320M10 (3K-Byte On-Chip Program ROM)
- External Memory Expansion to A Total of 8K Bytes at Full Speed
- 16-Bit Instruction/Data Word
- 32-Bit ALU/Accumulator
- 16 × 16-Bit Multiply in 200 ns
- 0 to 15 Bit Barrel Shifter
- Eight Input and Eight Output Channels
- 16-bit Bidirectional Data Bus with 40 Megabits-per-Second Transfer Rate
- Interrupt with Full Context Save
- Signed Two's Complement Fixed-Point Arithmetic
- NMOS Technology
- Single 5-V Supply

description

The TMS32010 is the first member of the new TMS320 digital signal processing family, designed to support a wide range of high-speed or numeric-intensive applications. This 16/32 bit single chip microcomputer combines the flexibility of a high speed controller with the numerical capability of an array processor, thereby offering an inexpensive alternative to multichip bit slice processors. The TMS320 family contains the first MOS microcomputers capable of executing five million instructions per second. This high throughput is the result of the comprehensive, efficient, and easily programmed instruction set and of the highly populated architecture. Special instructions have been incorporated to speed the execution of digital signal processing (DSP) algorithms.

The TMS320 family's unique versatility and power give the design engineer a new approach to a variety of complicated applications. In addition, these microcomputers are capable of providing the multiple functions often required for a single application. For example, the TMS320 family can enable an industrial robot to synthesize and recognize speech, serve objects with faster or optical intelligence, and perform mechanical operations through digital servo loop computations.

**TMS32010 . . . JO4 PACKAGE
(TOP VIEW)**



PN NOMENCLATURE

SIGNATURE	IO	DEFINITION
A11-AD	OUT	External address bits. I/O port address multiplexed over PA2-PA0.
PA2-PA0	IN	External polling input for bit test and parity operations.
CLKOUT	OUT	System clock output, \bar{K} crystal/CLKIN frequency.
D15-D0	IO	16 bit data bus.
DTN	OUT	Data enable indicates the processor accepting input data on D15-D0.
INT	IN	Interrupt.
MC/MP	IN	Memory mode select pin. High selects microcomputer mode. Low selects microprocessor mode.
MEM	OUT	Memory enable indicates that D15-D0 will accept external memory instruction.
RS	IN	Reset used to initialize the device.
VCC	IN	Power.
VSS	IN	Ground.
WE	OUT	Write enable indicates valid data on D15-D0.
X1	IN	Crystal input.
X2/CLKIN	IN	Crystal input or external clock input.

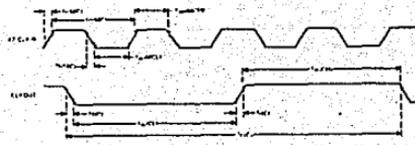
ADVANCE INFORMATION

This document contains information on a new product. Specifications are subject to change without notice.

Copyright © 1983 by Texas Instruments Incorporated

**TEXAS
INSTRUMENTS**
POST OFFICE BOX 220117 • DALLAS, TEXAS 75266

clock timing



NOTE: Clock frequency is not recommended to be less than 10 MHz and a rise/fall time of 2.0 ns and a hold time of 2.0 ns are advised.

MEMORY AND PERIPHERAL INTERFACE TIMING

switching characteristics over recommended operating conditions

PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Setup time before CS#		10	10	ns
Hold time after CS#		10	10	ns
Setup time before WE#		10	10	ns
Hold time after WE#		10	10	ns
Setup time before OE#		10	10	ns
Hold time after OE#		10	10	ns
Setup time before CE#		10	10	ns
Hold time after CE#		10	10	ns
Setup time before RAS#		10	10	ns
Hold time after RAS#		10	10	ns
Setup time before CAS#		10	10	ns
Hold time after CAS#		10	10	ns
Setup time before BA#		10	10	ns
Hold time after BA#		10	10	ns
Setup time before WE#		10	10	ns
Hold time after WE#		10	10	ns
Setup time before OE#		10	10	ns
Hold time after OE#		10	10	ns
Setup time before CE#		10	10	ns
Hold time after CE#		10	10	ns
Setup time before RAS#		10	10	ns
Hold time after RAS#		10	10	ns
Setup time before CAS#		10	10	ns
Hold time after CAS#		10	10	ns
Setup time before BA#		10	10	ns
Hold time after BA#		10	10	ns

loading characteristics over recommended operating conditions

PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Setup time before CS#		10	10	ns
Hold time after CS#		10	10	ns
Setup time before WE#		10	10	ns
Hold time after WE#		10	10	ns
Setup time before OE#		10	10	ns
Hold time after OE#		10	10	ns
Setup time before CE#		10	10	ns
Hold time after CE#		10	10	ns
Setup time before RAS#		10	10	ns
Hold time after RAS#		10	10	ns
Setup time before CAS#		10	10	ns
Hold time after CAS#		10	10	ns
Setup time before BA#		10	10	ns
Hold time after BA#		10	10	ns

NOTE: Data bus for read/write is not shown. CS# is active low.

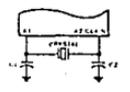


FIGURE 1 MULTIPLE EDGE OPTION

INTERNAL CLOCK SIGNAL

Additional clock signals can be used by applying an frequency divider to the CLK pin with a 50% duty cycle. The internal frequency divider may contain up to the number listed in the table below.

loading requirements over recommended operating conditions

PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Setup time before CS#		10	10	ns
Hold time after CS#		10	10	ns
Setup time before WE#		10	10	ns
Hold time after WE#		10	10	ns
Setup time before OE#		10	10	ns
Hold time after OE#		10	10	ns
Setup time before CE#		10	10	ns
Hold time after CE#		10	10	ns
Setup time before RAS#		10	10	ns
Hold time after RAS#		10	10	ns
Setup time before CAS#		10	10	ns
Hold time after CAS#		10	10	ns
Setup time before BA#		10	10	ns
Hold time after BA#		10	10	ns

switching characteristics over recommended operating conditions

PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Setup time before CS#		10	10	ns
Hold time after CS#		10	10	ns
Setup time before WE#		10	10	ns
Hold time after WE#		10	10	ns
Setup time before OE#		10	10	ns
Hold time after OE#		10	10	ns
Setup time before CE#		10	10	ns
Hold time after CE#		10	10	ns
Setup time before RAS#		10	10	ns
Hold time after RAS#		10	10	ns
Setup time before CAS#		10	10	ns
Hold time after CAS#		10	10	ns
Setup time before BA#		10	10	ns
Hold time after BA#		10	10	ns

NOTE: CS# is active low. CS# is active low. CS# is active low.

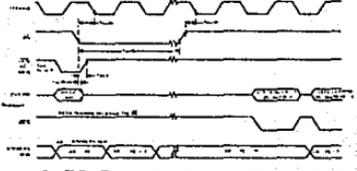
INTERNAL CLOCK



MEMORY TIMING

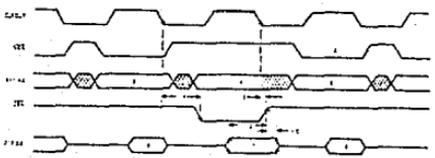
PARAMETER	TEST CONDITIONS	MIN	MAX	UNIT
Setup time before CS#		10	10	ns
Hold time after CS#		10	10	ns
Setup time before WE#		10	10	ns
Hold time after WE#		10	10	ns
Setup time before OE#		10	10	ns
Hold time after OE#		10	10	ns
Setup time before CE#		10	10	ns
Hold time after CE#		10	10	ns
Setup time before RAS#		10	10	ns
Hold time after RAS#		10	10	ns
Setup time before CAS#		10	10	ns
Hold time after CAS#		10	10	ns
Setup time before BA#		10	10	ns
Hold time after BA#		10	10	ns

clock timing



1. CS# is active low.
2. WE# is active low.
3. OE# is active low.
4. CE# is active low.
5. RAS# is active low.
6. CAS# is active low.
7. BA# is active low.

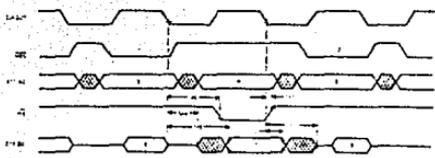
data bus timing



setup

1. CS# is active low.
2. WE# is active low.
3. OE# is active low.
4. CE# is active low.
5. RAS# is active low.
6. CAS# is active low.
7. BA# is active low.

DATA BUS TIMING



setup

1. CS# is active low.
2. WE# is active low.
3. OE# is active low.
4. CE# is active low.
5. RAS# is active low.
6. CAS# is active low.
7. BA# is active low.