UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

REALIZACION DE UN SISTEMA DE RECONOCIMIENTO DE COMANDOS HABLADOS

T E S I S

QUE PARA OBTENER EL TITULO DE

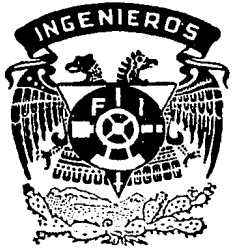
INGENIERO MECANICO ELECTRICISTA

P R E S E N T A N

MARIA GUADALUPE PEREZ ESTRADA

ALFONSO SANTOYO MORALES

Director de Tesis: DR. LUIS ANDRES BUZO DE LA PEÑA



MEXICO, D. F. 1989

UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

INDICE

INTRODUCCION.

El hombre siempre ha buscado la simplificación de actividades para ahorrar tiempo y esfuerzo, por lo que se han ido creando herramientas para lograr este objetivo. Gracias a los avances tecnológicos se han facilitado muchas funciones pero se han incrementado las actividades para operar las máquinas que hacen posible esta facilidad y ha surgido la necesidad de establecer una mejor comunicación hombre-máquina, es ésta una de las razones más importantes por las que el campo de las comunicaciones ha tenido un avance acelerado en los últimos años.

Por otra parte, el desarrollo de los sistemas digitales, ha hecho que la mayor parte de los problemas de ingeniería se enfoquen hacia este tipo de soluciones al tratar de resolverlos por medio del uso de microprocesadores o sistemas similares. De la combinación de estas dos herramientas, surge el Procesamiento Digital de Señales y dentro de éste, el procesamiento de voz, el cual puede aprovecharse para analizar la señal de voz y lograr así una comunicación hombre-máquina más estrecha por medio de la comunicación oral. Por ejemplo, el ser humano puede pronunciar un comando para que sea ejecutado por la máquina.

Históricamente hablando, el interés en las máquinas parlantes viene desde las civilizaciones antiguas, pero no fue hasta que comenzó el gran auge de la electrónica, cuando nuevamente se le vieron perspectivas a la comunicación oral hombre-máqina.

En la década de los 50's, se desarrollaron algunos sistemas para reconocimiento de vocales o dígitos que tenían un funcionamiento bastante aceptable, con un promedio de 90 a 100% de éxito, sin embargo, las técnicas empleadas para este objetivo, no pudieron extenderse a sistemas más complicados pues utilizaban mucho equipo y en general eran lentas.

Muchos de los sistemas desarrollados en los 60's, fueron sistemas de laboratorio costosos e inaceptables como para utilizarse en situaciones de la vida real debido a la elevada tasa de errores con que trabajaban. A partir de esa época y después en los 70's, se comenzaron a buscar métodos en cuanto a software (algoritmos) y hardware para lograr un procesamiento más rápido de los datos, fue entonces cuando el equipo para reconocimiento

se redujo a una cinta de voz pregrabada, un convertidor Analógico-Digital y una computadora de propósito general, sin embargo, aún así, los tiempos de procesamiento no se llevaban a cabo en tiempo real, en este momento surgió la necesidad de microprocesadores con intrucciones especiales para un tratamiento especial de los datos y con ciclos de instrucción más rápidos. Es así como aparecen en la década de los 80's, los microprocesadores de propósito particular y dentro de éstos, los Procesadores Digitales de Señales (PDS), muy útiles entre otras cosas, en el campo del procesamiento de la voz y por lo tanto en el reconocimiento de la misma.

Unido a lo anterior, es preciso tener un conocimiento acerca de la teoría de Reconocimiento de Patrones, para lograr un reconocedor de patrones de voz más eficiente.

El objetivo de esta Tesis, es presentar un sistema de reconocimiento de palabras aisladas dependiente del locutor, determinar si es posible aproximarlo a uno independiente del locutor (Cap. II) y elegir el tipo de orador, femenino o masculino, más adecuado para su funcionamiento.

El reconocimiento se lleva a cabo por medio de la comparación de ciertos parámetros de una palabra pronunciada, con los patrones creados a partir de la codificación, por medio de la Cuantización Vectorial, de los parámetros de la señal de voz propuestos por el modelo de Codificación Linear Predictiva (LPC). A su vez, el objetivo del reconocimiento de estas palabras, es la consideración de que éstas actúen como comandos para ejecutar cierta función.

Para el procesamiento en tiempo real de la señal de voz, se ha utilizado el PDS de Texas Instruments TMS32010, después, para la transmisión del comando reconocido, a una Computadora Personal (PC) que ejecutará la orden que indica el comando, se ha diseñado una interfase para la comunicación entre el μP y la PC.

Es así como en el primer capítulo se dá una descripción general de los PDS's, su software, hardware y aplicaciones, sobre todo del TMS32010 y se mencionan los PDS's que existen actualmente en el mercado.

En el Capítulo II se da un panorama general acerca de la teoría del Reconocimiento de Patrones, algunos aspectos importantes de la generación

y características de la voz y finalmente se hace una clasificación de los tipos de sistemas de reconocimiento del habla.

En el Capítulo III se presenta la descripción del hardware del sistema en cuanto al diseño y realización de la interfase para la transmisión de datos paralelo-serie entre el TMS32010 y la PC.

En el Capítulo IV encontramos la teoría necesaria para el funcionamiento del sistema, es decir, la relativa a la digitalización de la señal de voz y la necesaria para la implantación del software (que incluye Codificación Linear Predictiva, algoritmo de Cuantización Vectorial, etc.), la explicación de la generación de los patrones y la manera de llevar a cabo el reconocimiento.

El Capítulo V es una evaluación del sistema, donde se muestran las especificaciones del mismo, así como las pruebas realizadas y los resultados y conclusiones obtenidas.

Por último, en el Capítulo VI se habla de las aplicaciones y perspectivas en general de los sitemas de comunicación oral hombre-máquina y en especial del sistema presentado en el presente trabajo.

CAPITULO I

LOS PROCESADORES DIGITALES DE SEÑALES PDS

I.1 HARDWARE

Durante los últimos años, el Procesamiento Digital de Señales ha tenido un gran impulso, gracias a la aparición en el mercado de los Procesadores Digitales de Señales (PDS), que son por así decirlo, microprocesadores de uso específico, cuya arquitectura permite realizar operaciones a más de cinco millones de instrucciones por segundo (MPI's). Esto permite desarrollar algoritmos de procesamiento digital de señales en un sistema compacto (una o dos tarjetas, o inclusive un sólo circuito integrado) que antes solamente era posible realizar en computadoras grandes. Unido a lo anterior está la posibilidad de realizar análisis en tiempo real (se define tiempo real cuando un proceso es desarrollado sin crear un retardo notable para el usuario).

Una de las características más importantes que tienen los PDS es el hardware multiplicador, que es un sistema similar a la Unidad Aritmética Lógica (ALU) cuya única función es realizar la multiplicación. En los microprocesadores de propósito general, las multiplicaciones se realizan por una serie de sumas, por lo que consumen muchos ciclos de trabajo (típicamente 25 ciclos de su reloj); por otra parte, los PDS la ejecutan en un sólo ciclo de reloj gracias a este hardware multiplicador.

Por ejemplo, una multiplicación de 16 por 16 bits en el circuito WEDSP16, de la AT&T Bell Laboratories, requiere 60 nseg. o un sólo ciclo de reloj. En contraste, una multiplicación serialmente ejecutada en el microprocesador de propósito general MC68020 de Motorola Inc. toma 1500 nseg. o 25 ciclos de su reloj.

Otra característica muy importante es la rápidez para ejecutar una instrucción (ciclo de trabajo rápido). Esto se refleja directamente en la capacidad de procesamiento en tiempo real. Esto es, a medida que el ciclo de trabajo sea más rápido, podrán realizarse más operaciones entre dos

adquisiciones de muestras de la señal, permitiendo con ésto una mayor frecuencia de muestreo, o bien produciendo un retardo cada vez menor entre la señal de salida con respecto a la de entrada.

Los sistemas PDS tienen ciclos de trabajo menores de 200 nseg.

En la tabla 1.1 se muestran las frecuencias de muestreo para algunas aplicaciones típicas de los PDS, así como el número de instrucciones realizadas entre dos muestras utilizando un PDS con ciclo de trabajo de 200 nseg.

Aplicación	Frecuencia de muestreo	Número de instrucciones
Control	1 khz	5000
Telefonía	8 khz	625
Procesamiento de voz	8-10 khz	625-650
Procesamiento de audio (alta fidelidad)	40-48 khz	105-125
Procesamiento de video	14 Mhz	0.35

Tabla 1.1

Como puede verse en esta tabla, el número de instrucciones disponibles para aplicaciones que requieren una baja frecuencia de muestreo es muy grande, por lo tanto para este tipo de aplicaciones, por ejemplo control simple, los microprocesadores de propósito general o controladores microprogramados son los más apropiados. Sin embargo para otras aplicaciones de control, como por ejemplo robótica o control adaptivo, son mucho más adecuados los Procesadores Digitales de Señales.

Como también puede observarse, el número de instrucciones disponibles se reduce a medida que aumenta la frecuencia de muestreo. Para aplicaciones que requieren sólo unos cuantos cientos de instrucciones para poder ser realizadas en tiempo real como comunicaciones o procesamiento de

voz, los PDS son la solución ideal, debido a que existe tiempo suficiente entre las muestras para ejecutar un buen número de instrucciones.

Para aplicaciones con frecuencia de muestreo mayor, tales como procesamiento de video/imagen, los procesadores digitales de señales disponibles en la actualidad no son capaces de realizar esta función en tiempo real. Estas aplicaciones requieren además de una gran capacidad de memoria (más de 64000 palabras de espacio direccionable). En la actualidad existe al menos un PDS, que almacena y accesa 16 millones de palabras para poder realizar esta función de procesamiento de video e imágen (TMS320C30).

En la tabla 1.2 se dan las principales características (ciclo de trabajo, area de memoria, longitud de palabra, etc.) de varios procesadores digitales de señales.

Otra característica importante de los PDS es la arquitectura Harvard. En la cual la memoria de programa y la memoria de datos se encuentran en dos áreas separadas, permitiendo así, un traslape completo de las instrucciones de búsqueda y ejecución. La arquitectura Harvard modificada permite además la transferencia entre memoria de programa y memoria de datos, incrementando por ello la flexibilidad del sistema. Esta arquitectura maximiza la capacidad de procesamiento manteniendo dos estructuras de bus separadas para una mayor velocidad en la ejecución.

En conjunto con la arquitectura Harvard modificada, existe simultaneidad (pipelining) para reducir el ciclo de instrucción, incrementando así la velocidad del microprocesador.

En la operación de pipeline, las operaciones de búsqueda, decodificación y ejecución pueden llevarse a cabo independientemente, permitiendo de este modo el procesamiento de instrucciones simultáneamente.

El pipeline puede encontrarse en cualquier parte del sistema, desde dos hasta cuatro niveles, dependiendo del microprocesador de que se trate (fig.1.1). Esto permite que un sistema esté procesando de dos a cuatro instrucciones en paralelo, con cada instrucción en diferente etapa de su ejecución.

La familia TMS320 de Texas Instruments, utiliza dos niveles para su

COMPAÑIA	PRODUCTO	CAPACIDAD DE MEM. INTERNA		CAPACIDAD DE MEMORIA EXTERNA, BITS	CICLO DE INSTRUCC. NS	LONG. DE PALABRA, BITS	PARAMETROS DE MULTIPLICACION Y SUMA	
		RAM BITS	ROM BITS				BITS PARA ACUMULAC.	TIEMPO/OP, NS
ANALOG DEVICES INC. NORWOOD, MASS.	ADSP2100	16 X 24C	NINGUNO	16K X 24P 32K X 16D	125	16	40	125
AT&T BELL LABORATORIES, MURRAY HILL, N.J.	WE1SP16	512 X 16D,P	2K X 16D,P	64K X 16P	80	16	36	80
	WE1SP32	1K X 32D,P	512 X 32P	16K X 32D,P	244	8E 24M	8E 32M	244
FUJITSU LTD. TOKIO, JAPON	MB8764	256 X 16D	1K X 24P	1K X 16P,D	100	16	26	100
INMOS CORP. COLORADO SPRINGS, COLORADO	IMSA100	NA	NA	NA	NA	16	36	400
MOTOROLA INC., PHOENIX, ARIZONA	DSP56000	512 X 24D	512 X 24D 2K X 24P	64K X 24P 128K X 24D	98	24	56	98
	DSP56200	256 X 24D 256 X 16D	NA	NA	NA	16D, 24C	40	98
NEC ELECTRONICS INC., MOUNTAIN VIEW, CALIFORNIA	MPD77230X	128 X 16D	512 X 23P 512 X 23D (EPROM)	NA	250	16	16	500
	MPD77230	1K X 32D	1K X 32D 2K X 32P	4K X 32P,D	150	8E, 24M	8E, 16M	150
OKI ELECTRIC INDUSTRY CO. LTD., TOKYO, JAPAN	MSM6992	1K X 32P	1K X 32P	64K X 32D,P	100	8E, 16M	8E, 16M	100
SIGNETICS CORP., SUNNYVALE CALIFORNIA	PCB5010	256 X 16D 32 X 40P	512 X 16D 992 X 40P	64K X 16D,P	125	16	40	125
	PCB6011	256 X 16D	NINGUNA	512 X 16D 64K X 16D,P	125	16	40	125
THOMSON - CSF COMPONENTS CORP. CANOGA PARK CALIFORNIA	TS68930	256 X 16D	512 X 16D 1K X 32P	4K X 16D	160	16, 32	32	160N, 360C
	TS68931	256 X 16D	512 X 16D	4K X 16D 64K X 32P	160	16, 32	32	160N, 360C
TEXAS INSTRUMENTS INC., HOUSTON, TEXAS	TMS320C1X	256 X 16D	4K X 16D (EPROM)	4K X 16P	160, 200	16	32	32D, 400
	TMS32020	288 X 16D 256 X 16D,P	NINGUNO	64K X 16D,P	200	16	32	200
	TMS320C25	288 X 16D 256 X 16D,P	4K X 16P	64 X 16D,P	100	16	32	100
	TMS320C30	2K X 32D,P 64 X 32C	4K X 32D,P	16M X 32D,P	60	8E, 24M	8E, 32M	60
ZORAN CORP., SANTA CLARA, CALIF.	ZR32881	NA	NA	NA	NA	8	25	50
	ZR34161	256 X 17D	256 X 17D	64K X 16D,P	100	16	25	100

Tabla 1.2 PDS's en el mercado.

primera generación; tres niveles para su segunda generación y cuatro niveles para su tercera generación.

En la figura 1.1 se muestra un ejemplo de una operación de pipeline de tres niveles.

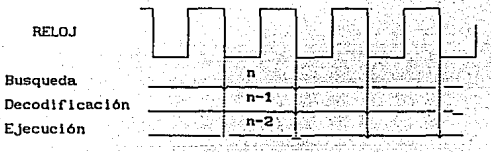


Figura 1.1

Durante cualquier ciclo de instrucción, se activan tres instrucciones diferentes, cada una en un estado de realización diferente. Por ejemplo, cuando la n-ésima instrución está siendo buscada (prefetch), la (n-1)-ésima instrucción está siendo decodificada (decode) y la (n-2)-ésima instrucción está siendo ejecutada (execute). En general esta simultaneidad es transparente para el usuario.

Los circuitos PDS pueden ser integrados en un sistema en tres configuraciones principales: stand-alone (sólo), esclavo (anfitrión/coprocesador) y multiprocesador.

En la configuración esclavo, el PDS actúa como un periférico de un microprocesador de propósito general. En este caso el mayor consumo de tiempo y proceso repetitivo lo realiza el PDS mientras las tareas de control y comunicación las realiza el microprocesador de propósito general. Esta arquitectura es particularmente adecuada para circuitos PDS de aplicación especifica, tal como reconocimiento de voz o procesamiento multicanal.

La mayor ventaja de un PDS se alcanza con la configuración de multiprocesamiento. Estos arreglos son apropiados para sistemas de radar,

en los cuales las tareas son distribuidas entre varios procesadores.

Para PDS de propósito general, el multiprocesamiento se maneja por al menos dos procesadores que envían y reciben información de control sobre ligas de comunicación. Estos procesadores comparten datos de una sola memoria.

I.1.1 Características Principales del TMS32010.

A manera de ilustración describiremos ahora las características del procesador digital de señales TMS32010 que es el que se utilizó en este trabajo:

a) Ciclo de trabajo: 200 ns
b) Memoria de datos RAM: 144 palabras
c) Memoria ROM de programa: 4 K de palabras
d) Expansión de memoria externa: 4 K de palabras
e) Multiplicador de 16 x 16 bits en paralelo, con 32 bits de resultado.
f) Registro de corrimiento Barrel para realizar corrimientos de palabras de memorias de datos dentro del ALU.
g) Registro de corrimiento paralelo.
h) Stack de 4 por 12 bits.
i) Dos registros auxiliares para direccionamiento indirecto.
j) Puerto serie de canal doble.

El TMS32010 fue introducido en 1982 y fue el primer microprocesador capaz de realizar 5 Mips. Utiliza la arquitectura Harvard modificada. Hay cuatro elementos aritméticos básicos:

a) El ALU es una unidad aritmético-lógica de propósito general que opera con una palabra de datos de 32 bits. La unidad puede sumar, restar y realizar operaciones lógicas.

b) El acumulador guarda la salida del ALU y también a menudo la

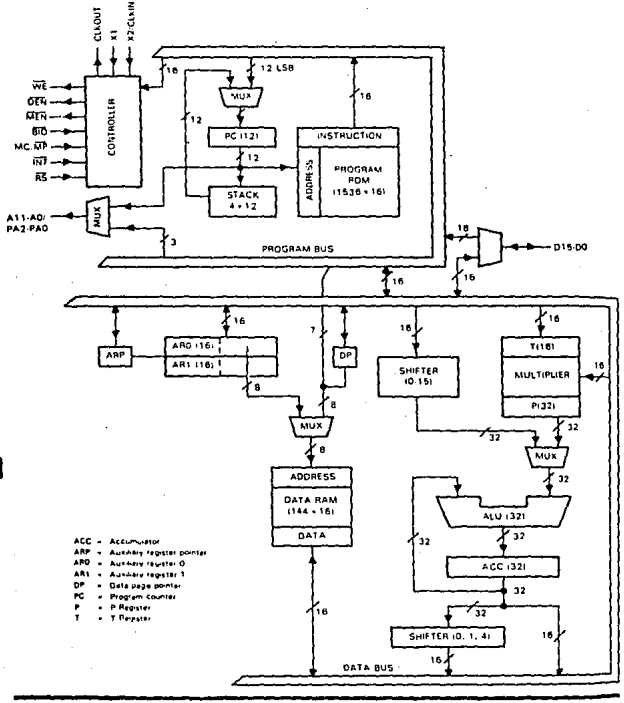
entrada. El acumulador esta dividido en dos palabras de 16 bits la parte alta (del bit 16 al 31) y la parte baja (del bit 0 al 15).

c) El multiplicador paralelo de 16 x 16 bits consiste de tres unidades: el registro T, el registro P, y el arreglo multiplicador. El registro T es un registro de 16 bits que guarda el multiplicando, mientras que el registro P es un registro de 32 bits que guarda el producto. Para utilizar el multiplicador, el multiplicando debe cargarse primero dentro del registro T desde la RAM de datos; Posteriormente se ejecuta la multiplicación con una de las siguientes instrucciones MPY ó MPYK, quedando el resultado en el registro P.

d) Corrimientos: Existen dos tipos de corrimientos en el TMS32010, uno de ellos se aplica al cargar el acumulador y puede ser de 0, 1, ó 4 bits. El otro se efectúa al transferir el contenido del acumulador a alguna localidad de memoria, y puede ser de 0 a 16 bits.
Todas las operaciones aritméticas son ejecutadas usando complemento aritmético de dos.

En la figura 1.2 se ilustra el diagrama de bloques del TMS32010.

functional block diagram



A

TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

Fig. 1.3 Arquitectura del TMS32010.

I.2 SOFTWARE DE LOS PDS.

En lo que se refiere al software, los PDS poseen una serie de instrucciones similares a las de cualquier microprocesador de propósito general, pero además como una característica muy importante, existe dentro de su conjunto (set) de instrucciones un grupo de instrucciones especiales muy útiles para el procesamiento digital de señales, como por ejemplo todas aquellas relacionadas con la multiplicación, o bien algunas que realizan dos o más operaciones en un sólo ciclo (carga, multiplicación y adición).

Dentro del procesamiento digital de señales, el operador de retraso (z^{-1}) es muy importante, de tal forma que todo PDS deberá tener entre sus instrucciones, alguna o algunas que realicen esta operación (desplazamiento de datos).

Normalmente una serie de instrucciones está organizado en los siguientes grupos (TMS320C30).

Instrucciones de carga y almacenamiento
Instrucciones aritméticas de dos comandos
Instrucciones lógicas de dos comandos
Instrucciones aritméticas de tres comandos
Instrucciones de operación paralela
Instrucciones aritmético/lógicas con instrucción de almacenamiento
Instrucciones de control de programa

Por otra parte, la mayoría de los fabricantes de PDS diseñan también útiles herramientas de software como por ejemplo ensambladores, ligadores y simuladores para trabajar en PC o en computadoras más grandes.

Los PDS de Texas Instruments, los TMS320, tienen un compilador para lenguaje C, el cual permite la inserción de código de lenguaje ensamblador en el programa fuente codificado en C. Pueden escribirse también funciones en lenguaje ensamblador y después llamarlas desde la fuente codificada en lenguaje C. De igual manera, las funciones en lenguaje C pueden llamarse

desde el lenguaje ensamblador. Las variables definidas en C pueden ser accesadas en los módulos del lenguaje ensamblador y viceversa. Como consecuencia de lo anterior es posible en este compilador, ajustar la cantidad de programación de alto nivel, contra la cantidad de lenguaje ensamblador de acuerdo a su aplicación. El compilador C está contenido en el TMS320C25 y en el TMS320C30.

I.3 APLICACIONES DE LOS PDS.

Como ya se mencionó, los procesadores digitales de señales son de gran importancia en la actualidad debido a su gran versatilidad. Las aplicaciones más comunes de estos dispositivos son: Transformadas Rápidas de Fourier (FFT), filtrado digital, etc. Existen ya otras aplicaciones tales como proceamiento de gráficas e imágenes, instrumentación y robótica entre otras.

Un elemento de gran importancia en el procesamiento digital de señales son los filtros digitales, dado que se encuentran en casi todas las aplicaciónes que describiremos más adelante, es por esta razón que dedicamos especial atención a este tema.

Un filtro FIR (Respuesta Finita al Impulso) es simplemente la suma de productos de un sistema de datos muestreados tal como la siguiente ecuación:

$$Y(n) = \sum_{i=1}^{N} a(i) * X(n-i) \qquad (1.1)$$

donde:

$Y(n)$ es la salida en el tiempo n
$a(i)$ es el i-ésimo coeficiente o factor de peso
$X(n-i)$ es la $(n-i)$-ésima entrada muestreada.

Como puede verse, para generar las $Y(n)$ salidas, se deberán realizar N multiplicaciones y sumas (suma de productos). Si el factor $a(i)$ permanece constante, cada paso del cálculo (multiplicación, suma y almacenamiento) podrá realizarse en una sola instrucción en muchos de los PDS fabricados últimamente.

Si los coeficientes son adaptados o actualizados con el tiempo, tal como en un filtro adaptivo o en cancelación de eco, el algoritmo de procesamiento digital de señales necesitará una mayor capacidad de cálculo. Las necesidades de adaptar cada uno de los coeficientes en cada muestra, pueden cumplirse en tres instrucciones en el TMS320C25.

a) Telecomunicaciones.

Muchos elementos de una red de telecomunicaciones pueden ser mejorados con un PDS, como por ejemplo un cancelador de eco o resonancia. En la cancelación de eco, un filtro adaptivo realiza la rutina de modelado y las modificaciones a la señal para cancelar el ruido causado por el mal acoplamiento de impedancias en las lineas de comunicación telefónica. Un filtro adaptivo de 256 pasos (32 mseg de cancelación de eco) puede ser ejecutado por un sólo circuito integrado sin memoria externa de datos o de programa.

Los PDS pueden ser utilizados para el diseño de modems tanto de alta como de baja velocidad, en un sistema de un sólo circuito integrado.

b) Procesamiento de video e imagen.

En aplicaciones de procesamiento gráfico y de imagen, es importante la capacidad de interconectar el PDS a un procesador anfitrión. Las configuraciones anfitrión/coprocesador son por lo tanto básicas para estas tareas. Las aplicaciones de procesamiento de gráficas e imágenes pueden accesar una gran area de datos y capacidad de memoria global, que permitan compartir las imágenes y gráficas en memoria con un procesador anfitrión, minimizando asi la transferencia innecesaria de datos.

Los modos de direccionamiento, indirecto e indexado, permiten el procesamiento fila por fila, cuando se están procesando multiplicaciones de matrices para rotación de imágenes tridimensionales, traslación y escalamiento.

c) Control de alta velocidad.

En aplicaciones de control de alta velocidad, se usan las caracteristicas de propósito general para operaciones de bit-test (prueba de bit) y operaciones lógicas, sincronización en tiempo y alta velocidad de transferencia de datos (10 millones de palabras de 16 bits por segundo). Estos equipos se utilizan en sistemas de malla cerrada para control de señales condicionadas, filtrado, cálculo de alta velocidad y capacidades de multicanalización y multiplexado.

Una actividad muy importante, es la robótica, en donde un PDS puede

reemplazar tanto a los controladores digitales como al hardware de procesamiento de señales analógicas, para comunicación con un procesador central y para la realización de funciones de control que requieren gran cantidad operaciones aritméticas.

d) Instrumentación.

Las aplicaciones de instrumentación tales como analizadores de espectros y varios instrumentos de alta precisión y alta velocidad, a menudo requieren una gran area de memoria de datos y un alto rendimiento de un PDS.

Y en general cualquiera de las aplicaciones de procesamiento número, que requieren de características tales como ciclo de trabajo rápido, hardware multiplicador, capacidad de multiprocesamiento y expansión de memorias de datos. Todas estas, características de los PDS.

En la tabla 1.3 se resumen las principales aplicaciones del PDS de Texas instruments TMS32010.

PROCESAMIENTO DIGITAL

- FILTRADO DIGITAL
- CORRELACIONES
- TRANSFORMADAS DE HILBERT
- VENTANEO
- TRANSF. RAPIDAS DE FOURIER
- FILTRADO ADAPTIVO
- GENERACION DE SENALES
- PROCESAMIENTO DE VOZ
- PROCES. DE RADAR Y SONAR
- CONTEO ELECTRONICO
- PROCESAMIENTO SISMICO

TELECOMUNICACIONES

- EQUALIZADORES ADAPTIVOS
- CONVERSION LEY /A
- GENERADORES DE TIEMPO
- MODEMS DE ALTA VELOCIDAD
- MODEMS DE MULT. BIT-RATE
- MODULACION/DEMODULACION DE AMPLITUD, FREC. Y FASE
- ENCRIPTADO DE DATOS
- MEZCLA DE DATOS
- FILTRADO DIGITAL
- COMPRESION DE DATOS

PROCESAMIENTO DE IMAGENES

- RECONOCIMIENTO DE PATRONES
- MEJORAMIENTO DE LA IMAGEN
- COMPRESION DE IMAGENES
- PROCESAMIENTO HOMOMORFICO
- PROCESAMIENTO DE RADAR Y SONAR

CONTROL A ALTA VELOCIDAD

- SERVO LINKS
- CONTROL DE POSICION Y VELOCIDAD
- CONTROL DE MOTOR
- GUIA DE MISILES
- CONTROL DE RETROALIMENTACION REMOTA
- ROBOTICA

INSTRUMENTACION

- ANALIZADOR DE ESPECTRO
- FILTRADO DIGITAL
- PROMEDIACION
- GENERACION DE ONDAS ARBITRARIAS

PROCESAMIENTO NUMERICO

- MULTIPLICACIONES Y DIVISIONES RAPIDAS
- OPERACIONES DE DOBLE PRECISION
- RAPIDO ESCALAMIENTO
- CALCULO DE FUNCIONES NO LINEALES (P. EJ. SEN X, e)

PROCESAMIENTO DE VOZ

- ANALISIS DE VOZ
- SINTESIS DE VOZ
- RECONOCIMIENTO DE VOZ
- RECONOCIMIENTO Y OBTENCION DE VOZ
- VOCODERS
- VERIFICACION DEL ORADOR

Tabla 1.3 Aplicaciones principales de los µp's TMS.

CAPITULO II

RECONOCIMIENTO DE PATRONES DE VOZ.

II.1 RECONOCIMIENTO DE PATRONES.

Existen muchos proyectos de investigación en cuyo estudio, desarrollo y realización, interviene la teoria de Reconocimiento de Patrones, por ejemplo: Inteligencia Artificial, Computadoras auxiliares en todo tipo de diseños, Reconocimiento de Patrones psicológicos y biológicos, reconocimiento estructural, linguistico, etc. A nosotros nos interesa el Reconocimiento de Patrones enfocado al reconocimiento de la voz, sin embargo, antes de entrar de lleno a este tema en especifico, daremos un vistazo a la forma general de manejar un problema de Reconocimiento de Patrones [3].

Para plantear adecuadamente un problema de reconocimiento de patrones, se requiere de tres espacios: espacio de patrones, espacio de caracteristicas y espacio de clasificación.

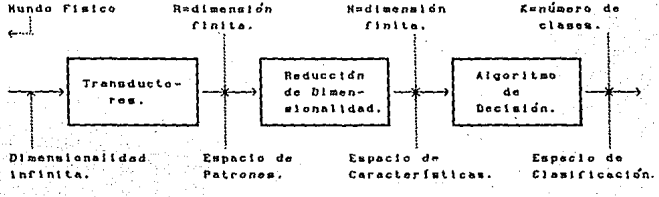


Figura 2.1 El problema de reconocimiento de patrones.

El mundo real tiene una dimensionalidad infinita, pero con ayuda de los

transductores, ese mundo se "transporta" a un espacio de dimensionalidad finita R, donde R es el número (generalmente grande) de valores escalares que describen al mundo real o parámetros que lo representan. A este espacio se le llama de patrones, sin embargo, para propósitos de clasificación, donde se llevan a cabo algoritmos largos que involucran muchos cálculos, es mejor tener un espacio de dimensionalidad reducida que mantenga las características escenciales para una clasificación adecuada, lo cual reducirá en gran medida el tiempo de procesamiento. Un espacio de este tipo se conoce como de características y su dimensionalidad N, es mucho menor que R.

Finalmente, el espacio de clasificación es el espacio de decisión en el cual se ha seleccionado una de las K clases.

El problema de reconocimiento de patrones se puede visualizar como una transformación desde el espacio de patrones P, al espacio de carácteristicas F, y finalmente al espacio de clasificación C.

$$P \longrightarrow F \longrightarrow C$$

II.1.1 Espacio de Patrones.

Es el dominio que está definido por los datos de entrada del mundo real. Sus ejes representan los diversos valores que pueden tomar los datos muestreados, en unidades del mundo real. Su dimensionalidad se denotará por R. Si todos los ejes representan diferentes unidades de datos del mundo real, un vector X en el espacio de patrones tendrá elementos escalares.

$$X = (X_1, X_2, X_3, \ldots, X_r, \ldots, X_R)^t \qquad (2.1)$$

donde: X_r valor particular asociado con la r-ésima dim.

r dimensión r-ésima del espacio de patrones.

R Número de dimensiones del espacio de patrones.

t transpuesta.

El vector X es un punto en el espacio de R dimensiones con valores de coordenadas Xr. A los vectores para los que tenemos un conocimiento previo acerca de su clasificación correcta, se les denomina prototipos, se denotan como $Y_m^{(k)}$, donde Y_m indica el emésimo prototipo de la clase S_k.

$$Y_m^{(k)} = (Y_{1m}^{(k)}, Y_{2m}^{(k)}, \ldots, Y_{rm}^{(k)}, \ldots, Y_{Rm}^{(k)})^t \qquad (2.2)$$

Puede haber Mk prototipos descriptivos de la k-ésima clase S_k.

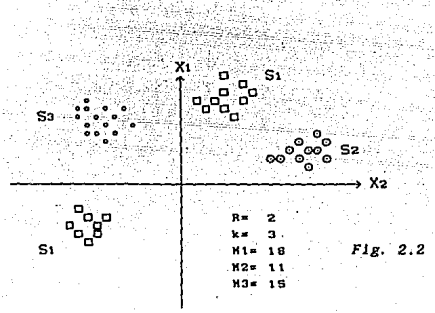


Fig. 2.2

El problema de la clasificación consiste en encontrar superficies separadas en el espacio R-dimensional, en donde, según algún criterio, se clasifiquen los prototipos y en donde se puedan clasificar los patrones desconocidos.

La similitud de un punto X en el espacio de patrones, con la k-ésima clase S_k, puede representarse como:

$$S(X, \{Y_m^{(k)}\}) = \frac{1}{Mk} \sum_{m=1}^{Mk} d^2 (X, Y_m^{(k)}) \qquad (2.3)$$

que es el promedio de la distancia Euclideana al cuadrado entre el punto X y el conjunto de prototipos $Y_m^{(k)}$ que definen la k-ésima clase.

Cada dimensión puede ser una medida de parámetros que deben ser normalizados.

II.1.2 Espacio de Caracteristicas.

Es un dominio intermedio entre el espacio de recolección de datos y el proceso de clasificación.

Debe estar definido por el poder discriminatorio de datos presentado en el espacio de patrones, pero optimizado. Existe la necesidad de tener un espacio en el cual se puedan llevar a cabo eficientemente los algoritmos de clasificación, este espacio, es el de caracteristicas.

Como ya hemos dicho antes, puesto que los algoritmos de clasificación son largos, complicados y con ello tardados, es conveniente que la dimensiónalidad N del espacio de caracteristicas, sea mucho menor que R (dimensionalidad del espacio de patrones), para que puedan simplificarse los algoritmos de clasificación, pues al tener menos datos para procesar, habrá un ahorro en tiempo, localidades de memoria, etc., sin embargo esta reducción de dimensionalidad deberá hacerse según algún criterio apropiado, para que las N dimensiones, contengan las caracteristicas más importantes y descriptivas del espacio de patrones y con ello se mantenga el poder discriminatorio para los procesos de clasificación.

De este modo, en el espacio de caracteristicas, se tiene:

$$X = (X_1, X_2, \ldots, X_n, \ldots, X_N)^T \qquad (2.4)$$

II.1.2.1 Extracción de Caracteristicas.

En el proceso de extracción de caracteristicas, hay una reducción en la dimensionalidad. Su objetivo es obtener las caracteristicas sobresalientes de los datos obtenidos en el espacio de patrones para el proceso de

reconocimiento.

Si un algoritmo de extracción de características es ineficiente, habrá muchos errores en los procesos de clasificación y reconocimiento, puesto que las características extraídas no representarán correctamente a los datos del espacio de patrones.

II.1.2.2 Extracción de Características en el Reconocimiento de voz.

Los modelos (subunidades) intermedios entre un procesador de señales y un interpretador semántico, son ejemplos de "características" y el proceso de comparación de un modelo desconocido de voz contra voz desconocida, es un ejemplo de "extracción de características". El campo del reconocimiento de patrones tiene muchos mecanismos para generar y representar características. Cualquiera que sea la estrategia para generar características intermedias, la función de éstas debe ser quitar la redundancia e información irrelevante de la voz.

La extracción de características comienza en el nervio auditivo y pasa a través de cuatro o cinco uniones sinápticas antes de llegar a la corteza cerebral auditiva. La fig 2.3 muestra ésto esquemáticamente. La detección de sonidos más complicados (como palabras enteras), se lleva a cabo con la ayuda de características simples y probablemente con retroalimentación de partes del cerebro más profundas que la corteza auditiva.

II.1.3 Espacio de Clasificación.

Es K dimensional y contiene las decisiones tomadas por los algoritmos de clasificación, los cuales realizan una partición del espacio de características (N dimensional), en varias regiones .

La partición más sencilla es lineal. En general, las particiones pueden definirse por algún criterio, por ejemplo: determinístico, estádístico, de información teórica, etc. y el resultado siempre es la segmentación del espacio de características N dimensional.

Es muy importante realizar una clasificación adecuada de la información que se tiene, ya que los algoritmos de decisión que van a reconocer, van a basar su funcionamiento en las diversas agrupaciones de información que se tengan.

No es nuestro objetivo hacer un análisis exhaustivo de las diversas técnicas de clasificación, por lo que en las siguientes secciones sólo se mencionarán y en ocasiones explicarán muy someramente algunas de ellas, debiéndose remitir el lector a la bibliografía correspondiente para más información al respecto [3].

II.1.3.1 Funciones Discriminantes.

Es una función que mide cada punto en el patrón o espacio de características y asigna a ese punto, un valor. Particiona el patrón o espacio de características en K regiones mutuamente exclusivas.

Considera las K clases de patrones. S1, ..., Sk; con prototipos definidos $Y_m^{(k)}$, donde m= 1, 2, ..., Mk, cuenta el número de prototipos de una clase dada.

A la construcción y ajuste de funciones discriminantes se le conoce como "entrenamiento" o "aprendizaje".

II.1.3.2 Clasificación de Libre Distribución.

Uno de los algoritmos de clasificación más simples, que utiliza una función discriminante lineal se conoce como clasificador de mínima distancia. Un clasificador que encuentra el punto promedio de los prototipos, definiendo una clase dada Sk, está dado por:

$$Y_k = \frac{1}{M_k} \sum_{m=1}^{M_k} Y_m(k) \qquad (2.5)$$

Existen k puntos en el espacio N, la regla de decisión es:

$$X \in S_j \qquad \text{si } d(X, \langle Y_j \rangle) = \min_k d(X, \langle Y_k \rangle) \tag{2.6}$$

Si deseamos definir la distancia de una X desconocida de una clase S_k:

$$X \in S_j \qquad \text{si } d(X, S_j) = \min_k d(X, S_k) \tag{2.7}$$

II.1.3.3 Clasificación Estadística.

Es posible modelar ciertos problemas en el reconocimiento de patrones utilizando parámetros estadísticos.

La función discriminante resultante se define por un conjunto de parámetros que se determinan a partir de prototipos e información ya existente.

II.1.3.4 Aprendizaje no Supervisado.

En el aprendizaje supervisado, se conocen los prototipos y su clasificación correcta. El aprendizaje no supervisado explica las técnicas de reconocimiento de datos sin clasificar. Clasifica los datos en subconjuntos tales que cada uno de ellos, contenga datos lo más parecidos posible. Usualmente se aplican técnicas iterativas para la formación de grupos, así tenemos como ejemplo el método de cadena, en el cual la primera muestra se toma como representativa del primer grupo y la distancia a la próxima muestra se calcula a partir del primer grupo. Si esa distancia es menor que un umbral α, se pone en la primera clase, de otra manera se forma un segundo grupo. El método se repite con todas las muestras, calculando la distancia entre cada muestra nueva y la muestra representativa de cada grupo. Si la distancia con respecto a todos y cada uno de los grupos anteriores, es mayor que el umbral, se forma una nueva

clase.

La convergencia implica que la distancia entre patrones y centros de grupos decrece continuamente.

II.1.3.5 Aprendizaje Secuencial.

La libre distribución y la clasificación estadística tienen un número fijo de prototipos. En el aprendizaje secuencial se aplican las técnicas de entrenamiento secuencialmente para definir un ordenamiento óptimo de características y para eliminar la información redundante.

II.2 LA VOZ.

Debido a que el objetivo de este trabajo es la realización de un sistema de reconocimiento de voz, es importante tratar algunos aspectos teóricos relacionados con ella así como ciertas definiciones. Todo esto nos va a proporcionar un conocimiento global de la voz para que al estar familiarizados con ella, se comprenda mejor el trabajo que se llevó a cabo.

II.2.1 El lenguaje.

El lenguaje es la capacidad que tiene el hombre para manifestar lo que piensa o siente o cualquier sistema que use para el ejercicio de dicha capacidad: lenguaje oral, escrito, visual, etc. [7].

Al problema de la naturaleza del lenguaje se le han dado diversas soluciones que pueden agruparse en dos corrientes:

A) Idealista o Espiritualista.

El lenguaje está producido por la actividad creadora de un espíritu.

B) Positivista o Naturalista.

El lenguaje oral es la forma natural de expresión por medio de la cual el hombre se comunica con sus semejantes. Está constituido por un conjunto de sonidos articulados que el hombre torna significativos.

El lenguaje se considera como un conjunto de hechos de diversa índole:

B1) Físico.

Comprende la generación y recepción de señales acústicas complejas. El individuo parlante (emisor) comunica sus pensamientos por medio de ondas sonoras al individuo escuchante (receptor) que capta dichas ondas.

B2) Fisiológico.

La voz está producida por la vibración de las cuerdas vocales formadas

por dos pliegues de la mucosa de las vías respiratorias. El paso del aire espirado hace vibrar las cuerdas vocales.

La intensidad del sonido emitido, depende de la fuerza de la corriente de aire.

La altura del sonido, depende de la longitud, tensión y delicadeza de las cuerdas vocales; así tenemos que en la mujer y en el niño, estas cuerdas son más cortas y finas que en el hombre, lo cual les permite emitir sonidos más altos.

El tono de la voz está dado por la forma de la laringe. La laringe se abre sobre la faringe por el orificio glótico, cubierto por la epiglotis. Sobre la glótis actúan cinco músculos motores como tensores o como constrictores variándola de forma y dimensión. La laringe está inervada por los nervios laríngeos salidos del nervio espinal.

Gracias al trabajo de la faringe, la boca y la nariz, el hombre disfruta la facultad de emitir sonidos articulados.

B3) Psíquico.

Los pensamientos del individuo emisor se llevan a cabo dentro de una jerarquía de niveles de procesamiento. En el más alto nivel se encuentran los conceptos fundamentales que dan lugar a los pensamientos, o ideas, éstos se codifican en forma de palabras, en un nivel inferior, al cual llamaremos nivel lingüístico. Las palabras se codifican en niveles sucesivos más bajos, involucrando procesos neuronales y movimientos articulatorios, hasta que se alcanza el más bajo nivel con la señal acústica.

De este modo, al conocer las principales características del lenguaje oral humano, será más fácil crear sistemas lingüísticos artificiales que lo imiten.

Se sabe además que en cada nivel de procesamiento lingüístico, se utiliza la redundancia para superar ambigüedades inherentes en ese nivel. Una forma de redundancia es la variedad de características intuitivas para distinguir diferentes elementos en la voz. Como otra forma de redundancia se puede considerar a las retroalimentaciones entre niveles. De este modo,

un error cometido en la identificación de un sonido del lenguaje a partir de su patrón acústico, puede corregirse referenciándolo a reglas físicas, linguísticas, etc., conocimiento de las características vocales del interlocutor, o del contexto del mensaje.

La redundancia aunque se agrega a cada nivel de procesamiento, trae como consecuencia un incremento en la capacidad necesaria de un canal para transmitir la información. Se ha estimado que, si el mensaje básico pudiera extraerse de la señal de voz y transmitirse en forma de elementos fonéticos, habría un ahorro de 1000:1 en la capacidad requerida del canal de transmisión, comparada con la que se necesitaría para transmitir la forma de onda acústica [5].

II.2.1.1 La Señal de voz.

La señal de voz es una señal no estacionaria, es decir, su comportamiento estadístico cambia con el tiempo, por eso, para poder aplicar el modelo L.P.C. (ver sec. IV.3.2), se divide a la señal de voz en segmentos cortos llamados estructuras, a los que se les considera cuasi-estacionarios.

La señal acústica cubre un rango de frecuencias de 15 KHz. aproximadamente, es el resultado de la interacción de dos funciones separadas del sistema vocal humano. Nos referiremos a estas funciones como la función de envolvente espectral y la función de excitación.

A) Función envolvente espectral.

Se debe a múltiples resonancias acústicas producidas en el tracto vocal según su forma y tamaño. Este se varía por medio de los articuladores: la lengua, mandíbula y paladar.

Debido a que las características físicas del tracto vocal varían de persona a persona, los parámetros que dependen de su forma y con ésto, de la forma de la función envolvente espectral, son muy utilizados para el reconocimiento de un orador específico.

La función envolvente del espectro de la señal acústica, se puede

describir según las frecuencias y amplitudes de sus primeros tres o cuatro picos. Estos picos se llaman formantes de los sonidos vocales.

Gran parte de la información hablada se transmite por medio de la función envolvente y, en teoría, debe ser posible extraer ésta información si se conocen las medidas de los formantes y su variación con respecto a un nivel de energía global.

B) Función de excitación.

En el espectro de la función de excitación, se imprime el patrón de las resonancias tractovocales. La función de excitación se produce por tres procesos básicos diferentes [9]:

1) Uno de éstos procesos se debe a la acción de la vibración de las cuerdas vocales, las cuales modulan un flujo de aire proveniente de los pulmones, esto equivale a una excitación de pulsos cuasi periódicos cuyo espectro es muy rico en componentes armónicas. El espectro se ve como una serie de líneas espectrales relacionadas armónicamente, de las cuales, las primeras 50 a 100 tienen energía significante. La sensación de tono depende de la frecuencia con la cual ocurran los pulsos individuales.

2) El segundo proceso de excitación se debe a la turbulencia de aire generada al forzar el aire proveniente de los pulmones a través de una constricción en el tracto vocal. El silbido producido es aleatorio y se conoce como ruido fricativo. El espectro es continuo y cubre un amplio rango de frecuencias, pero se modifica por las resonancias del tracto vocal, particularmente aquellas asociadas con la parte del tracto en frente del punto de generación.

3) El tercer tipo de excitación se produce cuando el tracto vocal se mantiene cerrado en algún punto mientras se crea una presión de aire por detrás de éste (sonidos oclusivos). La apertura brusca de la oclusión produce un efecto semejante a una excitación en escalón de presión que cae inversamente con la frecuencia (excitación transitoria).

II.2.1.2 Los Fonemas.

Etimológicamente, fonema quiere decir "sonido de la voz". El fonema se considera como la unidad de sonido más significativa que el escucha puede percibir, su representación gráfica es la letra, de este modo, las palabras aisladas se perciben como sucesiones de fonemas que forman un conjunto de sonidos articulados.

Los patrones asociados con un fonema particular varían considerablemente con el sujeto parlante, velocidad de articulación, fonemas cercanos, etc.

Puesto que la cantidad de fonemas en un lenguaje dado es relativamente pequeña comparada con la cantidad de todas las posibles palabras, el fonema se considera como la unidad de reconocimiento en el reconocimiento automático de voz.

Existen algunas variantes de los fonemas, éstas se conocen con el nombre de alófonos y éstos también forman parte del conjunto de sonidos de un lenguaje.

Los fonemas se clasifican en vocales y consonantes. Las primeras son más sonoras e intensas que las segundas. Dentro de cada grupo existen subclasificaciones como se podrá apreciar en la tabla 2.1 [8].

II.2.2 Reconocimiento de la voz.

Todos los reconocedores sintéticos o naturales de voz, biológicos, mecánicos o de cualquier otro tipo, tienen transductores que convierten las ondas sonoras en representaciones internas, es decir, existe una transformación del mundo físico (dimensionalidad infinita) al espacio de patrones (dimensionalidad finita). Estos reconocedores, poseen modelos internos guardados en memoria, de los patrones acústicos producidos por articulaciones, comparan articulaciones desconocidas con representaciones internas de palabras conocidas. Algunos modelos de patrones se visualizan como conjuntos de reglas (gramaticales, prosódicas, etc), otros están implícitos en las funciones de transferencia de los transductores

VOCALES			
	ANTERIOR	CENTRAL	POSTERIOR
CERRADA	I		U
MEDIA	E		O
ABIERTA		A	

Tabla 2.1 Clasificación de los fonemas.

CONSONANTES									
				BILABIAL	LABIO-DENTAL	LINGUO-DENTAL	LINGUO-INTERDENTAL	LINGUO-ALVEOLAR	LINGUO-VELAR	LINGUO-PALATAL
MOMENTANEAS	SONORAS	OCLUSIVAS		B		D			G	
		AFRICADAS								(CONYUJE
	SORDAS	OCLUSIVAS		P		T			K, C, Q	
		AFRICADAS								C H
CONTINUAS	SONORAS	NASALES		M				N		N
		LIQUIDAS	LATERALES					L		L L
			VIBRANTES: SIMPLE			R				
			VIBRANTES: MULTIPLE			R R				
		FRICATIVAS			V		(DEDO	Z	(MAGO	J
	SORDAS	FRICATIVAS			F		CAZA	S	X	SIN

(micrófonos, amplificadores, etc.) y compresores de datos.

Se busca que la operación de reconocimiento mecánico sea lo más parecida posible a la forma en que un humano reconoce el habla. El escucha humano además de que puede reconocer sonidos extremadamente breves, realiza el proceso de reconocimiento en medio de distorsiones debidas al medio acústico del orador, por lo que un reconocedor mecánico ideal, debería tener la capacidad para superar estos "obstáculos".

La figura 2.3 es un diagrama de flujo que ilustra la similitud entre los elementos en el reconocimiento humano del habla y en el reconocmiento mecánico, desde el oído (o micrófono), hasta el acto final de reconocimiento. La figura 2.4 muestra el flujo del habla de un sistema de reconocimiento experimental.

Todos los sonidos del habla que llegan al oído pueden modelarse como combinaciones lineales de ondas senoidales con fases diferentes. Toda la información acústica importante para el reconocimiento del habla se representa en el tiempo por el espectro de potencia.

El oído humano actúa como un analizador de espectro de potencia (sistema que mide las intensidades relativas de las ondas senoidales componentes de un sonido de la voz) y muchos sistemas de Reconocimiento Automático del Habla (ASR- Automatic Speech Recognition), realizan alguna clase de análisis espectral en el proceso de reconocimiento. No siempre es necesario realizar un análisis espectral; la codificación inicial de voz puede estar en términos de las autocorrelaciones de las variaciones de amplitud de las formas de onda del lenguaje, de códigos predictivos lineales o de estadísticas de cruces por cero.

La representación paramétrica inicial del habla contiene información redundante que puede quitarse; a ésto se le llama "compresión de datos".

Para el reconocimiento de una articulación desconocida, el primer paso es dividirla en segmentos cortos. El segundo paso es compar estos segmentos, con los segmentos correspondientes en todas las palabras prototipo para medir su similitud a intervalos iguales de tiempo. La articulación desconocida se clasifica como la palabra prototipo que tiene la mayor similitud con ella para todos sus segmentos cortos.

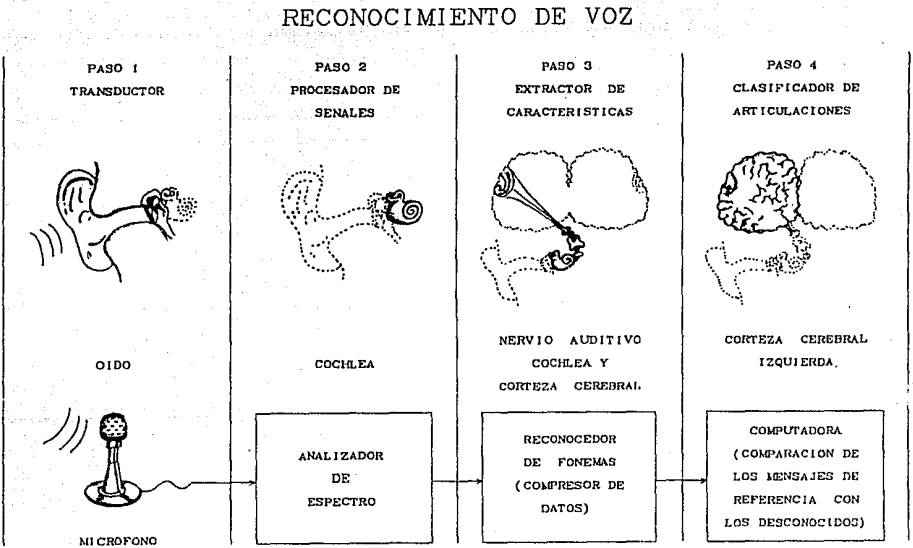


Fig. 2.3 Semejanza entre el reconocimiento humano de voz con el reconocimiento mecánico de voz.

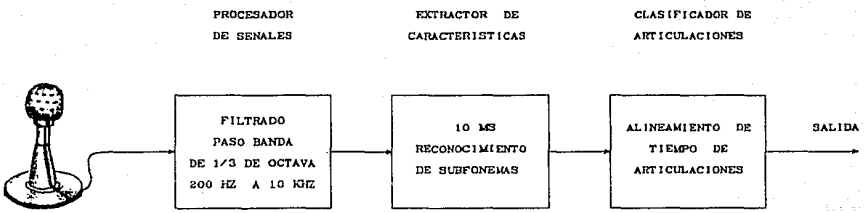


Fig. 2.4 Diagrama de bloques para un reconocedor de articulaciones aisladas.

II.3 CLASIFICACION DE LOS SISTEMAS DE RECONOCIMIENTO DE VOZ.

Dentro del area general de comunicación, podemos encontrar tres campos principales en la comunicación hombre-máquina por medio de voz [9,6].

A) Sistemas de Respuesta al Habla (SRH).
B) Sistemas de Reconocimiento del Orador (SRO).
C) Sistemas de Reconocimiento de Voz (SRV).

Los Sistemas de Respuesta al Habla tienen las siguientes caracteristicas:

1) Capacidad de almacenar un vocabulario que se utilice en el sistema de respuesta a la voz.
2) Reglas para formar mensajes con los elementos del vocabulario.
3) Un programa para componer mensajes de respuesta.

Los Sistemas de Reconocimiento del Orador se clasifican en dos grupos:

B1) Sistemas de verificación del orador.
El problema de verificación consiste en decidir si un mensaje reune las condiciones suficientes como para ser identificado con un mensaje previamente grabado en memoria y extraido del mismo individuo.
B2) Sistemas de identificación del orador.
El problema de identificación consiste en reconocer, entre una población de N, al individuo cuyas caracteristicas almacenadas coincidan en mayor grado con las extraidas del mensaje recibido, después de que la máquina hizo N comparaciones.

Los Sistemas de Reconocimiento de Voz pueden separarse en dos grupos según la naturaleza del proceso de reconocimiento empleado: Comparación de Patrones y Extracción de Caracteristicas, en las siguientes secciones se explicarán cada uno de estos dos grupos.

Estos Sistemas de Reconocimiento también se clasifican según su

complejidad, la cual se determina de acuerdo a diversas características, entre las cuales podemos encontrar las siguentes:

1) Tipo de voz de entrada: palabras aisladas, voz continua, etc.
2) Número de oradores: sistemas de un solo orador, varios oradores, población ilimitada, etc.
3) Tipo de oradores: hombre, mujer, niños, etc.
4) Ambiente en que se habla: laboratorio de computadoras, lugar público, etc.
5) Sistema de transmisión: micrófono de alta calidad, micrófono para hablar de cerca, teléfono, etc.
6) Sistema de entrenamiento: sin entrenamiento, entrenamiento continuo, etc.
7) Tamaño del vocabulario: pequeño (1-20 palabras), mediano (20-100 palabras), grande (más de 100 palabras).
8) Formato de entrada de la voz: texto restringido, texto libre, etc.

También en las siguientes secciones se explicarán algunas de estas características (las principales).

II.3.1 Comparación de Patrones.

La voz de entrada se compara contra cada uno de los patrones guardados en la máquina por medio de un algoritmo y se decide, según algún criterio, cual de los patrones guardados se acerca más a la voz producida; por lo tanto, el tiempo de respuesta en un sistema de este tipo, se incrementa linealmente con el tamaño del vocabulario.

Los patrones se producen por una transformación particular de la señal de voz ya sea por medio de un análisis espectral o de correlaciones.

Ventajas.

a) Facilidad con que los patrones guardados pueden cambiarse por otros

de acuerdo al idioma, orador o unidad de reconocimiento.

b) Simplicidad del sistema.

c) Facilidad del sistema para hacerse adaptivo por sí mismo.

Desventajas.

a) Dificultad para discriminar datos irrelevantes, puesto que el proceso de comparación debe cubrir todo el rango de frecuencias en el cual puede ocurrir un patrón.

A su vez, según la unidad de reconocimiento, podemos encontrar reconocedores de patrones de fonemas y reconocedores de patrones de palabras.

II.3.1.1 Reconocimiento de patrones de fonemas.

Al hablar de un reconocedor de fonemas, el término fonema, no se refiere propiamente al sonido de una letra (ver sec. II.2.1.2) que puede aislarse para ser reconocida; sino a un agrupamiento conceptual de sonidos de voz que permite que una articulación de voz se describa en términos de un alfabeto de símbolos fonémicos.

Las palabras aisladas se pueden representar como cadenas de fonemas, las cuales hacen posible una reducción, con respecto a los reconocedores de palabras, tanto en los requerimientos de memoria para guardar prototipos de palabras, como en el tiempo de procesamiento para el reconocimiento, ya que existen muchas más palabras que fonemas.

Si un fonema dado es un elemento común en un grupo de patrones, entonces, la compresión de datos se lleva a cabo guardando la información detallada del fonema sólo una vez y después reemplazando el fonema por su nombre, donde quiera que éste ocurra. Los nombres de los fonemas pueden entonces usarse para encontrar la información del mismo cuando sea necesario.

II.3.1.2 Reconocimiento de patrones de palabras.

Un reconocedor de patrones de palabras, comparará la palabra de entrada entera, con cada uno de los patrones que posee en su memoria. Debido a que existe una gran cantidad de palabras en un idioma determinado, es preferible que los vocabularios a reconocer giren alrededor de un tema central, es decir, estén conformados por palabras relacionadas entre sí.

Este tipo de reconocedores es muy utilizado para el reconocimiento de palabras aisladas, por esta razón, es necesario que cuenten con algoritmos apropiados para detectar el comienzo y el fin de las palabras ya sea para la generación de los patrones o para el reconocimiento.

II.3.2 Extracción de Características.

Estos sistemas se refieren a circuitos especializados o rutinas de procesamiento para detectar la presencia de características típicas de los elementos del sonido del habla. El reconocimiento de una articulación se basa en la ocurrencia secuencial de ciertas características, la salida puede ser sólo una indicación de que se ha detectado la ausencia o presencia de un elemento particular de un sonido.

Algunas de estas características pueden ser: la presencia simultánea de formantes en regiones definidas del dominio de la frecuencia, la razón de cambio de la frecuencia de formantes, transiciones espectrales, nivel global de energía, intensidad y duración del sonido, tasa de elevación de la intensidad, posición en relación a otros sonidos en una articulación, etc.

La identificación de sonidos en el nivel acústico, requiere del conocimiento de características prosódicas del individuo parlante, éstas se dan en el sonido de su voz como información fonémica: sexo, edad, estado emocional, rango y uso de la variación de tono, intensidad, ritmo, etc.

Ventajas:

a) Como cada sección está diseñada para la detección óptima de una característica particular, es fácil excluir datos irrelevantes y distinguir entre sonidos similares pero distintos.

Desventajas:

a) Es más complicado que la comparación de patrones.
b) Requiere de un periodo de estudio para adaptarse a cambios en el idioma, orador, etc.
c) Es difícil hacerlo autoadaptivo.

Para resultados óptimos, deben combinarse estos dos métodos, en los niveles de procesamiento más bajos, extracción de características; pero los patrones formados por secuencias de elementos de sonido, deben reconocerse por un proceso de comparación de patrones.

II.3.3 Tipo de entrada.

II.3.3.1 Palabras aisladas.

En un sistema de reconocimiento de palabras aisladas, se requiere, para separar una palabra de otra, una pausa de cierto tiempo lo suficientemente grande para que no se confunda con las pausas entre las sílabas de cada palabra y al mismo tiempo, para que procese la voz en este intervalo de tiempo. Este tipo de sistemas, debe contar con algoritmos para detección de comienzo y fin de las palabras. Es preferible que los vocabularios se refieran a un tema.

II.3.3.2 Voz conectada.

En un sistema de reconocimiento de voz conectada, no existe ningún tipo de pausa entre las palabras, es por ésto que la principal dificultad de estos sistemas, está en determinar donde termina una palabra y comienza otra, además, se tienen que estudiar otras características de la señal de voz. Todo esto hace necesaria la utilización de más información para el reconocimiento, aumentando con ello, la complejidad del sistema.

Un sistema así, presenta ventajas y desventajas con respecto a los sistemas de reconocimiento de palabras aisladas, entre las cuales destacan las siguientes:

Ventajas:

a) Vocabulario muy amplio.
b) Ahorro de tiempo al evitar esperar un lapso entre la pronunciación de cada palabra.
c) Mayor versatilidad.
d) Mayor similitud con la forma humana de hablar.

Desventajas:

a) Dificultad para distinguir las separaciones entre palabras.
b) Dificultad para distinguir todas las variedades sintácticas.
c) Dificultad para distinguir todas las variedades fonéticas.
d) Complejidad del sistema, etc.

II.3.4 Tamaño de la población.

a) Sistemas de reconocimiento para un solo orador.
b) Sistemas de reconocimiento para varios oradores.

Se recomienda que un sistema del tipo del primero, no se aproxime al segundo, tanto en el aprendizaje como en el reconocimiento.

II.3.5 Tamaño del vocabulario.

Para un vocabulario pequeño, se emplean métodos de comparación para el reconocimiento, sin embargo cuando el vocabulario ha aumentado, estos métodos se vuelven demasiado caros y cobran gran importancia las técnicas para la representación compacta de patrones acústicos de palabras y las técnicas para reducir la búsqueda restringiendo el número de posibles palabras que pueden ocurrir en cierto momento.

II.4 SISTEMAS.

En el cuadro siguiente se observa la clasificación de los diferentes tipos de sistemas de reconocimiento de voz que existen, ordenados de acuerdo a su dificultad. Los tamaños de los vocabularios son para sistemas típicos de cada grupo y varían de sistema a sistema [4].

Sistema	Modo del habla	Tamaño del vocabulario	Ambiente
Reconocimiento de palabras aisladas (WS).	Palabras Aisladas	10 - 300	——
Reconocimiento restringido de voz unida (CSR).	Voz Unida	30 - 500	Habitación en silencio
Entendimiento restringido de voz (SU).	Voz Unida	100- 2000	——
Dictado mecánico restringido (DM).	Voz Unida	1000- 10000	Habitación en silencio
Entendimiento de voz sin restringir (USU).	Voz Unida	∞	——
Reconocimiento de voz unida sin restric. (UCSR).	Voz Unida	∞	Habitación en silencio

Tabla 2.2

Las variaciones en las fuentes de conocimiento, afectan la posibilidad de funcionamiento y el rendimiento de los sistemas de reconocimiento de voz. Para entender lo que es una fuente de conocimiento, consideremos a un orador, el cual utiliza inconscientemente sus conocimientos acerca del

lenguaje, contexto del mensaje y ambiente para comprender una oración. Estas son las fuentes de conocimiento (F.C.), que incluyen características de los sonidos de la voz (características fonéticas), variabilidad en la pronunciación (fonología), patrones de tensión y entonación de la voz (c. prosódicas), patrones del sonido de las palabras (c. lexicas), estructura gramatical del lenguaje (sintaxis), significado de palabras y oraciones (c. semánticas) y contexto de la conversación (c. pragmáticas).

II.4.1 Sistemas de Reconocimiento de Palabras Aisladas. (Word Recognition-Isolated, WR).

Dado un vocabulario conocido (de 10 a 300 palabras) generado por un orador conocido, un sistema de esta clase puede reconocer correctamente una palabra pronunciada aisladamente en un 94 a 99%. El vocabulario y/o locutor pueden cambiarse, requiriéndose para este fin una sesión de entrenamiento.

En general se utiliza como estrategia de reconocimiento, la comparación de patrones, esto involucra una comparación de parámetros de la articulación de entrada con los patrones prototipo de referencia de cada una de las palabras del vocabulario. La figura 2.5 presenta el diagrama de flujo de un típico sistema de reconocimiento de palabras aisladas.

II.4.2 Sistemas de Reconocimiento Restringido de Voz conectada. (Connected Speech Recognition-Restricted, CSR).

Se considera incorrecta una oración, aún cuando sólo una palabra en toda la articulación es incorecta, por lo que los aciertos tienden a ser menores (30 a 81%).

Utilizan información limitada, por ejemplo vocabulario restringido y sintaxis sencilla, requieren que el orador hable muy claramente y en una habitación en silencio. Todo ésto hace que su estructura se mantenga relativamente simple como para ser un sistema de voz conectada, sin

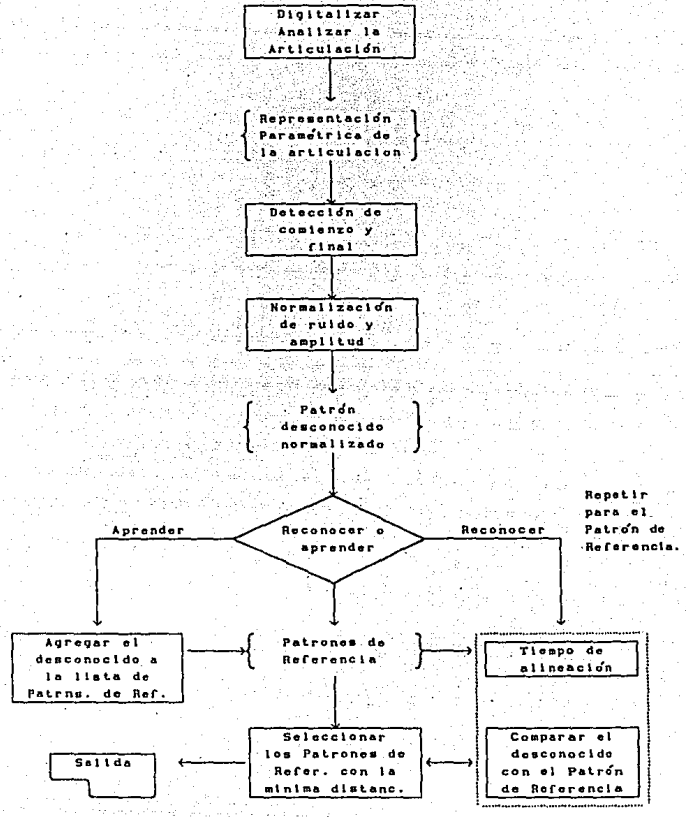


Fig. 2.5 Diagrama de flujo de un típico Sistema WSR.

embargo, es difícil determinar donde termina una palabra y donde comienza otra, además, las características acústicas de los sonidos y las palabras presentan mayor variabilidad en voz conectada, comparadas con las de palabras pronunciadas aisladamente.

En una secuencia de n palabras, se necesitarían n^n patrones para reconocer todas las posibles secuencias de esas palabras, esto requeriría de una gran cantidad tanto de memoria para el almacenamiento de patrones, como de tiempo para el procesamiento de los datos. La selección de la mejor secuencia de palabras requiere un algoritmo de búsqueda en árbol y una representación más compacta de los patrones de sonido de las palabras, ésto último se logra con la segmentación de la señal continua de voz en partes discretas acústicamente invariantes para representar a las palabras como secuencias de sonidos (fonémicos o silábicos), entonces se puede utilizar un diccionario fonémico para comparar y determinar qué palabra fue pronunciada. La fig. 2.6 muestra el diagrama de flujo de un típico sistema CSR.

II.4.3 Sistemas de Entendimiento Restringido de Voz. (Speech Understanding Restricted, SUS).

Estos sistemas alcanzan aproximadamente de un 25 a un 30% de resultados satisfactorios. Su objetivo es reconocer el mensaje, de modo que a medida que éste se entiende, no es importante reconocer todos y cada uno de los fonemas y/o palabras correctamente.

Se considera que la señal de voz no tiene toda la información necesaria para decodificar el mensaje, se deben utilizar todas las fuentes de conocimiento disponibles (fonología, prosodia, etc.), tema de la conversación, generación de la respuesta, etc.

De esta forma, además de los problemas que se tienen con los sistemas CSR, un sistema SU debe funcionar aún cuando la articulación de entrada no esté correctamente estructurada o presente murmullos.

Utiliza cadenas fonémicas y genera hipótesis acerca de las posibles siguientes palabras, los modelos del lenguaje se utilizan para verificar

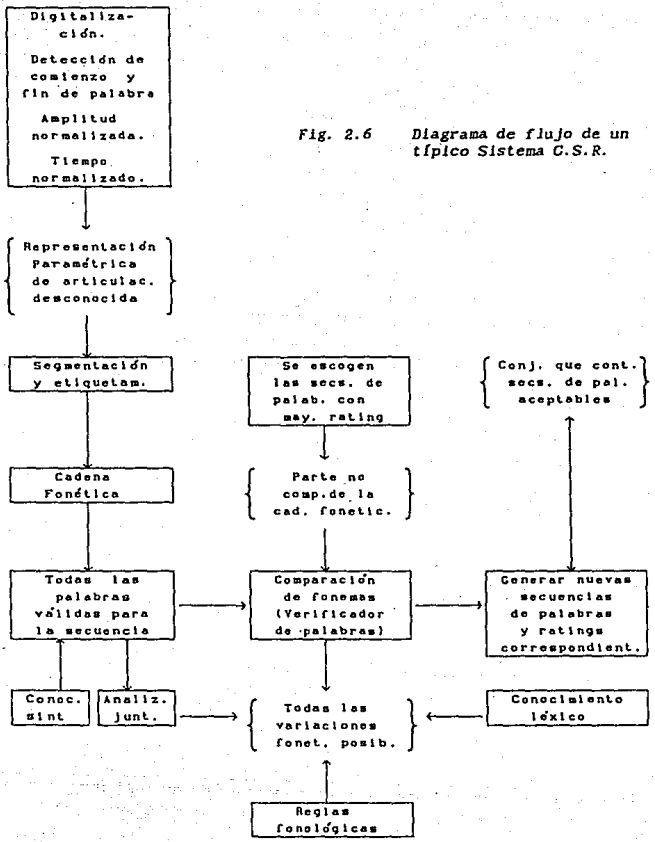


Fig. 2.6 Diagrama de flujo de un típico Sistema C.S.R.

tales hipótesis.

II.4.4 Sistemas de Dictado Mecánico Restringido. (Dictation Machine-Restricted, DM).

El usuario podría deletrear cualquier palabra desconocida al sistema.

II.4.5 Sistemas de Comprensión de Voz sin Restringir. (Unrestricted Connected Speech Understanding, USU).

Requiere un reconocimiento de voz conectada con vocabulario infinito, pero permite el uso de toda la información disponible.

II.4.6 Sistemas de Reconocimiento de Voz Conectada sin Restricción. (Unrestricted Connected Speech Recognition. UCSR).

Es el más complicado de todos los sistemas de reconocimiento, requiere de vocabulario ilimitado, pero no considera la disponibilidad de información.

CAPITULO III

ELEMENTOS DE UN SISTEMA PARA RECONOCIMIENTO DE COMANDOS.

III.1 ELEMENTOS DEL SISTEMA.

Un sistema de reconocimiento de comandos por voz está compuesto por dos unidades principales: la unidad que ejecutará el comando y un Sistema de Procesamiento Digital de Señales. La primera puede consistir en cualquier tipo de computadora o bien una interfase que adapte una señal (con niveles TTL) para poder realizar una función especifica. El sistema de procesamiento digital de señales puede consistir en un microprocesador de propósito general para aplicaciones sencillas, o bien un PDS de los mencionados en el capitulo uno, para aplicaciones más sofisticadas.

En la figura 3.1 se ilustra un diagrama de bloques de un sistema de procesamiento digital de señales.

Cualquier sistema de procesamiento de señales, digital o analógico, requiere que la señal que se desee analizar sea representada en forma de una variable eléctrica, función que es realizada por el transductor.

La conversión Analógica-Digital es una etapa muy importante de los sistemas PDS, debido a que de ella depende que la señal procesada digitalmente sea una buena representación de la señal continua de la cual se desea obtener información.

La parte medular de un sistema PDS es obviamente el CPU el cual, como ya se mencionó anteriormente puede consistir según la aplicación en uno o varios μP's o PDS's.

III.3.1 Descripcion del Sistema.

Por último tenemos la interfase de salida, la cual nos permite ver el resultado final del procesamiento digital de señales; en este punto cabe considerar la conversión Digital-Analógica como señal de salida.

Para este trabajo se utilizará una computadora personal compatible como

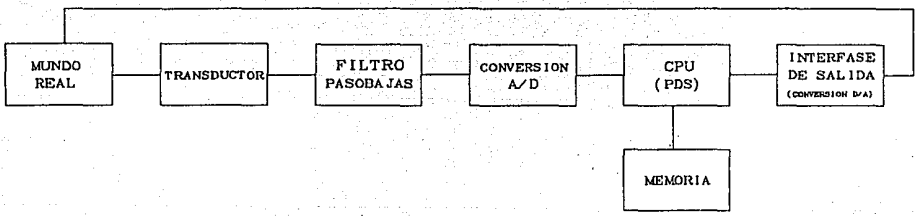


Fig. 3.1 Diagrama de bloques de un Sistema PDS.

unidad que ejecutará el comando, el EVM (Apendice A) modulo de evaluación del PDS TMS32010, como sistema de procesamiento digital de señales, así como la tarjeta de interface (AIB) (Apendice B) como convertidor Analógico Digital.

El Sistema de Procesamiento Digital de Señales mencionado anteriormente, el EVM, es una herramienta auxiliar de diseño, fabricada por Texas Instruments para el microprocesador TMS32010, el cual tiene funciones de edición, ejecución y depurado de programa. En dicho modulo se instaló un programa en lenguaje ensamblador para realizar el entrenamiento o el reconocimento.

En el diagrama de la fig. 3.2 se ilustra la conexión que se utilizó para este trabajo.

El programa de reconocimiento de comandos hablados es enviado de la PC al EVM por medio del puerto serie, este programa es almacenado en la memoria de programa del TMS32010.

La señal de voz es introducida al sistema por medio de un microfono y es filtrada por medio de un filtro pasobanda, esto con el fin de limitar en banda la señal de entrada; las frecuencias de corte utilizadas en este filtro son: 200 Hz. y 3400 Hz. que es el rango de frecuencias de una linea telefónica.

El convertidor A/D con el cual Trabaja el AIB es de 12 Bits, de tal forma que podemos garantizar una precisión aceptable en el sistema de adquisición.

La comunicación entre el EVM y la PC es realizada por medio del puerto serie, de tal forma que se puede almacenar información en la PC que posteriormente será utilizada para el reconocimiento de comandos, y a su vez el EVM envía el resultado del reconocimiento de comandos a la PC para que en ésta se efectuen las acciones correspondientes a la palabra reconocida.

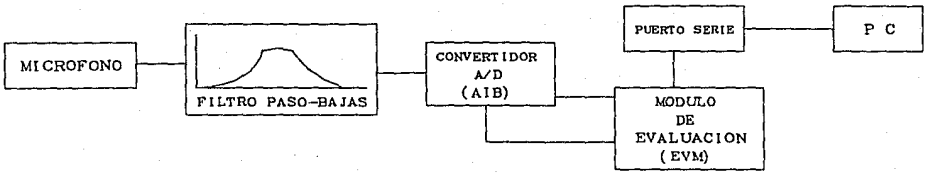


Fig. 3.2 Diagrama de bloques de la conexión usada.

III.2 DISEÑO DE UN PUERTO SERIE.

El objetivo de este puerto es lograr la comunicación de la tarjeta de conversión (AIB) con una PC para la transferencia de patrones almacenados en disco duro a la memoria del TMS32010, así como transmitir la clave de la palabra reconocida.

III.2.1 El 8251A (USART).

El circuito USART (Universal Synchronous Asynchronous Receiver/Transmitter) 8251A fue diseñado como periférico para la comunicación de datos de los procesadores de la familia INTEL, aunque pueden ser programados por practicamente cualquiera de los microprocesadores disponibles. El USART acepta datos de cualquier CPU en forma paralela y los convierte en una serie continua de datos para su transmisión; por otra parte puede convertir datos en forma serial y presentarlos al CPU en forma paralela.

III.2.2 Direccionamiento.

Como puede verse en el manual de usuario del AIB, (Apendice B) la tarjeta AIB tiene la capacidad de habilitar 8 puertos de entrada/salida por medio del decodificador 74LS138 (U51), estas salidas van de Y0 a Y7 y corresponden a las direcciones de 0000H a 0007H. Dos de dichos puertos permanecen libres por lo cual se pueden utilizar para habilitar al puerto serie; el puerto Y6 (dirección 0006H) es utilizado para la programación del USART y para la lectura del status del mismo, el puerto Y7 (dirección 0007H) es usado para la lectura y escritura de datos.

Los tiempos de direccionamiento del USART (250nseg.) presentan una limitante considerable; el TMS32010 direcciona sus periféricos únicamente por 200 nseg., tiempo en el cual escribe o lee un dato del bus de datos. Este tiempo es insuficiente para que el USART los escriba o lea del bus de

datos, por lo que es necesario mantener la habilitación, esto se hace con el circuito 74LS123 (U9 y U10) conectado como monoestable. Tiene la caraterística de poder ser disparado tanto por el flanco de subida como por el flanco de bajada y produce un pulso Q cuya duración se fija por un arreglo de resistencia y capacitor externos, la duración de este pulso Q deberá ser mayor que dos veces la del ciclo de trabajo de TMS32010, con el fin de evitar perturbaciones en las líneas de habilitación, provocadas por el segundo acceso consecutivo que debe realizar el TMS32010 para garantizar la lectura o escritura de un dato. En la figura 3.3 se ejemplifica lo anterior.

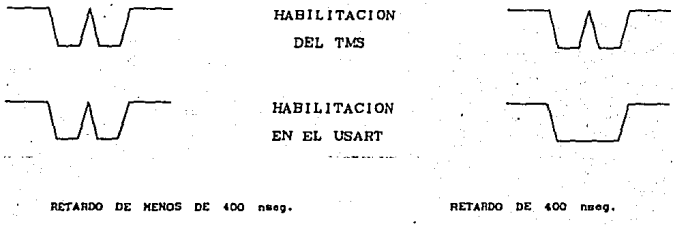


FIG. 3.3

La duración de este pulso Q esta dada por la ecuación:

$$T_w = kR_xC_x$$

donde: k=0.5
Tw en nseg. (duración del pulso)
Rx en KΩ
Cx en pF

III.2.3 Lectura y escritura de datos.

Como ya se mencionó anteriormente, el tiempo de habilitación del USART (250 nseg.) es mayor que el tiempo que el TMS32010 mantiene un dirección válida (200 nseg.). Esto implica que cuando el TMS32010 lee un dato del USART, este sería un dato no válido, debido a que el USART tarda más de 200 nseg en colocar un dato en su bus de datos; para asegurar que el dato leído por el TMS32010 es correcto, es necesario realizar dos lecturas repetidas al puerto serie, de esta manera al efectuar la segunda lectura, habrán pasado 400 nseg., tiempo en el cual el USART habra yá presentado un dato válido.

La misma situación se presenta cuando el TMS32010 realiza una escritura en el puerto serie, en éste caso se debe mantener un dato en el bus de datos del USART, hasta que éste lo pueda leer.

Esta función de almacenamiento temporal de datos, tanto de lectura como de escritura, es realizada por los latches de salida (U4,U5,U6,U7) 74LS373.

III.2.4 Base de tiempo.

El circuito 8251A requiere de una señal de reloj para su funcionamiento, la frecuencia de esta señal debe ser por lo menos 30 veces mayor que la velocidad máxima de transmisión y recepción, en la figura 3.4 puede verse que la terminal CLK del USART esta conectada a la terminal 6 del generador de reloj, la frecuencia de esta señal es igual a la frecuencia de oscilación del cristal (18.432 Mhz.) entre 9, lo que equivale a 2.048 Mhz.

La frecuencia de transmisión y recepción es generada a partir de la frecuencia del cristal (terminal 12 del generador de reloj) y es dividida por medio de un arreglo de contadores para generar así, diferentes frecuencias de transmisión y recepción.

III.3 COMUNICACION TMS32010-P.C.

La secuencia que se sigue para la transmisión de datos y el funcionamiento de este sistema se puede resumir de la siguiente manera:

1) Se carga el programa de reconocimiento con el menú principal en el EVM.

2) Se pronuncia una de las palabras del menú.

3) La señal de voz se procesa en el EVM para reconocer la palabra pronunciada.

4) El EVM envia a la P.C. vía la interfase paralelo-serie el código correspondiente a la palabra que fue reconocíada.

5) La P.C. recibe dicho código, identifica de entre los archivos almacenados en disco duro el correspondiente a la palabra pronunciada y lo envia al EVM. En este archivo se encuentran los patrones correspondientes a un nuevo menú.

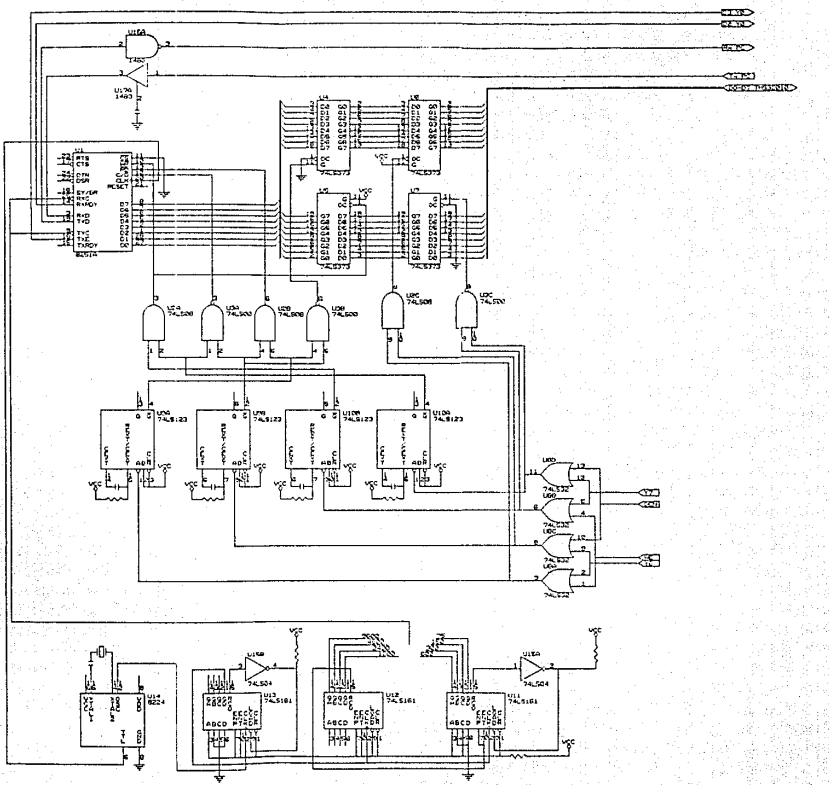
6) Si al EVM no llega como primer dato un cero, ir a 2, en caso contrario, continuar.

   Esto se debe a que ningún primer dato del archivo de patrones puede ser cero, puesto que la primera correlación es la mayor de todas. Dentro del algoritmo del sistema general, hay casos en los que la P.C. le envia al EVM un cero, indicando con ésto que el menú sigue siendo el mismo y que se debe pronunciar una de las mismas palabras del vocabulario de la vez anterior, motivo por el cual, la P.C. no le envia al EVM, los patrones para un nuevo menú.

7) Puesto que el EVM siempre debe recibir el mismo número de datos; con un contador verifica que le llegue el número exacto de ellos.

8) Ir al punto 2 donde la nueva palabra que se va a pronunciar tiene que ser una de las del nuevo menú, cuyos patrones ahora se encuentran almacenados en la memoria del TMS

Con esto se concluye la explicación de la forma como se lleva a cabo la comunicación P.C.- EVM y a la vez, la explicación del funcionamiento general del Sistema de Reconocimiento de Comandos Hablados.


PUERTO SERIE

CAPITULO IV.

UN EJEMPLO DE APLICACION. SOFTWARE.

En este Capitulo se explicarán los aspectos fundamentales de la teoria que respalda el funcionamiento de un Sistema de Reconocimiento de Comandos Hablados, asi como la forma en que dicha teoria se lleva a la práctica por medio de su implantación a nivel software, lo cual se hace posible con la programación del TMS32010 (Apéndice A), cuyas caracteristicas principales se vieron previamente, con la programación de la interfase serie-paralelo y con la programación de la Computadora Personal que se utilizó como objeto sensible a los comandos hablados que se procesarán con el TMS32010.

En capitulos anteriores ya se ha hablado del hardware del sistema.

Como hemos dicho previamente, el S.R.V. realizado en el presente trabajo, está basado en el reconocimiento de un número limitado de palabras. Todas y cada una de estas palabras tiene, dentro de la memoria de la máquina, un grupo de datos, conocidos como patrones, los cuales fueron previamente generados por el orador mediante un proceso de entrenamiento (que se explicará más adelante en la sección IV.4). Cuando el usuario articula una de esas palabras, la máquina, por medio de un algoritmo adecuado de comparación, entre patrones y articulación, decidirá cual de las palabras del vocabulario fue la que se pronunció (reconocimiento; sección IV.5).

Existen ciertos procesos comunes entre la generación de patrones (entrenamiento) y el reconocimiento de palabras. Dichos procesos son: Digitalización de la señal (muestreo, cuantización y codificación), detección de comienzo y fin de palabra y representación paramétrica de la señal (en este caso se hizo uso de los parámetros L.P.C., codificación por cuantización vectorial, etc.).

A continuación se da una explicación de cada uno de estos procesos y posteriormente su ubicación en el entrenamiento y reconocimiento, donde se explicarán cada uno de estos dos temas.

IV.1 DIGITALIZACION DE LA SENAL DE VOZ.

Como sabemos, la señal de voz es analógica, por lo tanto, para realizar un cálculo y procesamiento de parámetros, es necesario llevar a cabo una conversión A/D de la señal de voz.

El proceso de conversión Analógico-Digital comprende varias etapas: muestreo de la señal, cuantización y codificación de las muestras cuantizadas, expliquemos brevemente cada una de estas etapas [9,10]:

IV.1.1 Muestreo.

Una señal analógica puede tomar un número infinito de valores en el tiempo, para que pueda procesarse digitalmente, es necesario representarla como una secuencia de números, lo cual se logra al muestrearla periódicamente, quedándonos una señal de la forma:

$$X(n) = Xa(nT) \qquad -\infty < n < \infty \; ; \quad n = \text{entero} \qquad (4.1)$$

donde: $Xa(t)$= señal analógica.
$X(n)$= secuencia que representa a $Xa(t)$
T= periodo de muestreo.

Sin embargo, el muestreo de una señal debe hacerse bajo ciertas condiciones para que la secuencia X(n) represente correctamente a la señal analógica original y ésta pueda reconstruirse a partir de X(n), estas condiciones se indican en el teorema de muestreo.

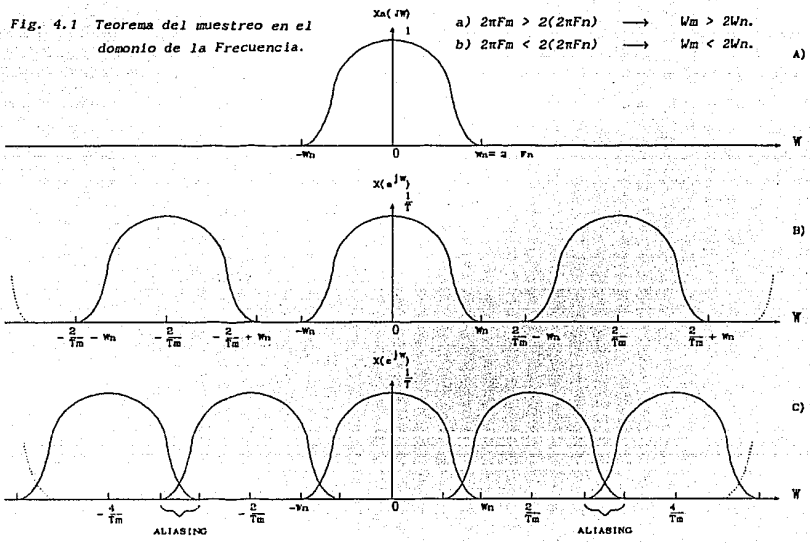
IV.1.1.1 Teorema de Muestreo.

Si una señal:
Xa(t), tiene una transformada de Fourier limitada en banda $Xa(j\omega)$, tal que:
$Xa(j\omega) = 0$ para $\omega \geq 2\pi Fn$,

Fig. 4.1 Teorema del muestreo en el domonio de la Frecuencia.

a) $2\pi Fm > 2(2\pi Fn) \longrightarrow Wm > 2Wn.$

b) $2\pi Fm < 2(2\pi Fn) \longrightarrow Wm < 2Wn.$



entonces la señal Xa(t) puede recontruirse únicamente a partir de muestras igualmente espaciadas Xa(nT), $-\infty < \infty$, si $\frac{1}{Tm} > 2Fn$.

donde: Fn= frecuencia de Nyquist

$\frac{1}{Tm}$ = Fm = frecuencia de muestreo.

Podemos considerar dos casos:

1) Si $2\pi Fm > 2(2\pi Fn)$, la imágen centrada en $2\pi Fm$ no se traslapa en la banda base $|\omega| < 2\pi Fn$.
2) Si $2\pi Fm < 2(2\pi Fn)$, la imágen centrada en $2\pi Fm$ se traslapa en la banda base $|\omega| < 2\pi Fn$.

Al caso 2) se le conoce con el nombre de "aliasing", pero si se cumple el caso 1) se evita el "aliasing" y entonces es posible reconstruir a la señal analógica original a partir de sus muestras (fig 4.1).

IV.1.2 Cuantización y codificación.

Una vez que se ha muestreado la señal de voz, se procede a realizar un proceso de cuantización. Las muestras de voz pueden tomar un número infinito de valores en amplitud, la cuantización consiste en aproximar estos valores a un predeterminado número de niveles de amplitud. En éste proceso hay una pérdida irreparable de información ya que no se puede reproducir exactamente a la señal original a partir de sus muestras cuantizadas.

Cada nivel de amplitud (de un total de M) en la señal cuantizada, puede representarse por un número finito de dígitos m, cada uno de los cuales tiene n posibles niveles de amplitud. Por lo tanto se tiene que:

$$M = n^m \qquad (4.2)$$

donde comunmente n= 2. A medida que m aumenta, se tiene un mayor número M de niveles de cuantización, aumentando así la calidad de la señal cuantizada de voz, pues se pierde menos información. A este proceso de asignar etiquetas a los niveles de cuantización, se le llama codificación.

Existen varios métodos para llevar a cabo el proceso de cuantización que se pueden dividir en dos grupos: Cuantización escalar y cuantización

vectorial [12].

En la cuantización escalar fig (4.2), cada parámetro de un conjunto de datos, se codifica por separado; en la cuantización vectorial se forman varios conjuntos de datos (vectores), mismos que se agrupan. Los vectores de cada grupo se cuantizan de acuerdo a un solo vector X que los represente (fig 4.3), para mayor información ver sec. IV.3.3.

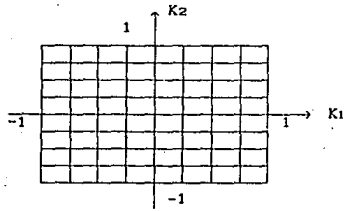
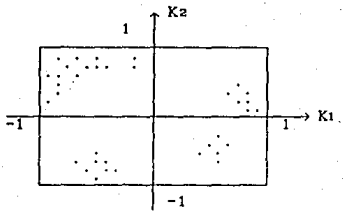
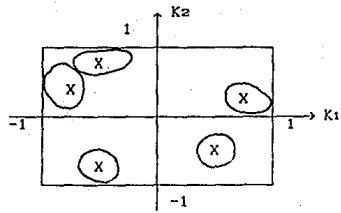


Fig 4.2
Espacio de dos dimensiones con cuantización escalar.



a) Distribución de los datos que se representan por un modelo de segundo orden.



b) Agrupamiento de las estructuras de datos para generar el alfabeto de vectores de cuantización para la Cuantiz. Vectorial.

Fig. 4,3 Cuantización Vectorial.

A continuación se da una breve descripción de los métodos de cuantización escalar más conocidos [10]:

IV.1.2.1 Cuantización Uniforme.

Se representa a la señal como un número finito de amplitudes, cada nivel de amplitud se separa a intervalos iguales de amplitud.

Consideremos la figura 4.4, todos los valores de S entre S_i y S_2 están representados por $\hat{S}_2$, tenemos también que:

$$\Delta = S_{i+1} - S_i \qquad (4.3)$$
$$\Delta = \hat{S}_{i+1} - \hat{S}_i$$

donde Δ es el intervalo de cuantización en un cuantizador uniforme.

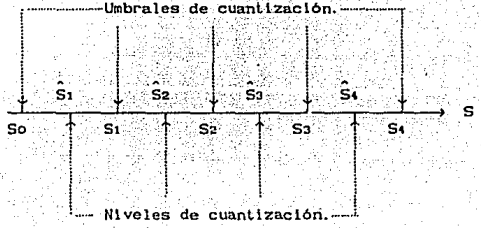


Fig. 4.4 Cuantizador de cuatro niveles.

La figura 4.5 muestra dos tipos de cuantización uniforme para un cuantizador de 8 niveles, cuando uno de sus niveles es cero, se le llama cuantizador de medio paso, cuando los niveles de cuantización no incluyen cero, se llama cuantizador de media subida.

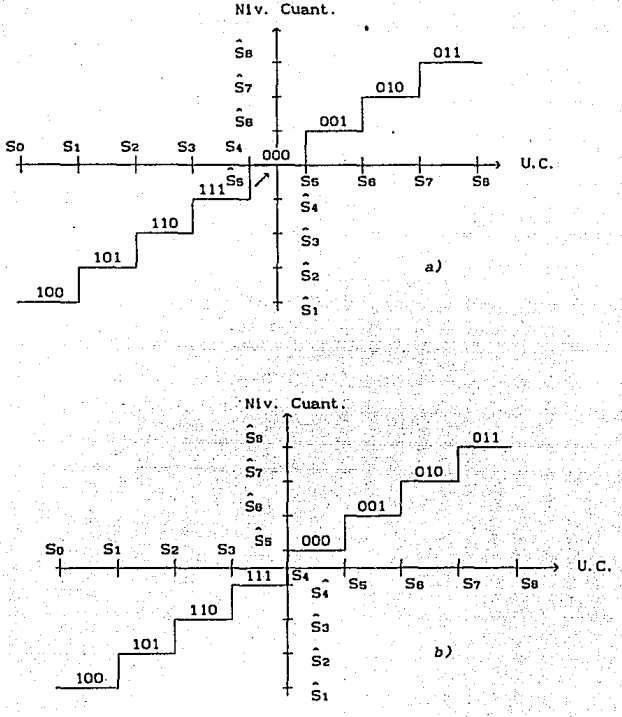


Fig. 4.5 Cuantización Uniforme para 8 niveles.
a) Cuantizador de medio paso. b) Cuantizador de media subida.

IV.1.2.2 Cuantización no Uniforme o Logarítmica.

Debido a que la amplitud de la señal de voz es muy variable, con un cuantizador uniforme de intervalos grandes, se perdería mucha información cuando la señal variara en un rango de amplitud muy reducido. Para resolver esto, se utiliza un cuantizador no uniforme cuyo intervalo de cuantización aumente proporcionalmente a la amplitud de la señal.

IV.1.2.3 Cuantización Adaptiva.

El comportamiento estadístico de una señal de voz cambia con el tiempo, por lo tanto, un cuantizador diseñado a partir de ciertas características en un segmento de voz, puede no ser adecuado para la señal completa. Esto se resuelve cambiando dinámicamente el tamaño del intervalo de cuantización, adaptándolo a las variaciones de las características estadísticas de la señal. A este tipo de cuantización se le conoce como adaptiva.

IV.1.2.4 Cuantización Diferencial.

Se considera que entre muestras adyacentes de voz, existe una correlación, puesto que no hay una variación considerable entre ellas. Debido a que esta variación es menor que la de toda la señal, se considera que la entrada del cuantizador es la diferencia:

$d(n) = S(n) - \hat{S}(n)$

donde: $S(n)$= muestra de entrada,
$\hat{S}(n)$= predictor de la muestra de entrada y
$d(n)$= señal de error de predicción.

Puede demostrarse que [10]:

$$e(n) = \hat{d}(n) - d(n) = \hat{S}(n) - S(n) \qquad (4.4)$$

donde: $\hat{d}(n)$= señal de diferencia cuantizada.

Esta señal tiene una variancia pequeña y por lo tanto el error de cuantización es menor que el que se tendría al cuantizar la señal original.

IV.1.3 Conversión A/D del S.R.V.

La primer etapa consiste en pasar la señal de voz por la Tarjeta de la Interfase Analógica (AIB), conectada al EVM del TMS32010 (ver Apéndices B y A respectivamente). En esta tarjeta se encuentra un filtro paso banda de 320 HZ. a 4.7 KHz. (esta frecuencia se puede variar por el usuario pero comunmente está fija en 4.7 Khz.), cuyo objetivo es limitar el ancho de banda de la señal para minimizar efectos de "aliasing". Posteriormente se introducirá la señal de voz a un convertidor A/D de aproximaciones sucesivas que se encuentra en esta misma tarjeta (ver Capítulo III).

Cada muestra de la señal de voz se somete a un filtrado para preénfasis. La función de transferencia de un filtro para preénfasis, es de la forma: $1-az^{-1}$. Este filtro enfatiza las frecuencias altas, para que el modelo L.P.C. trabaje bien en las frecuencias bajas y en las altas, pues como puede observarse en la figura 4.6 que muestra a una típica envolvente espectral de voz, el espectro decae en las frecuencias altas; ésto se debe a los efectos de radiación de los sonidos en los labios.

La ecuación del filtro es:

$$X(n)= U(n)- 0.95\ U(n-1) \qquad (4.5)$$

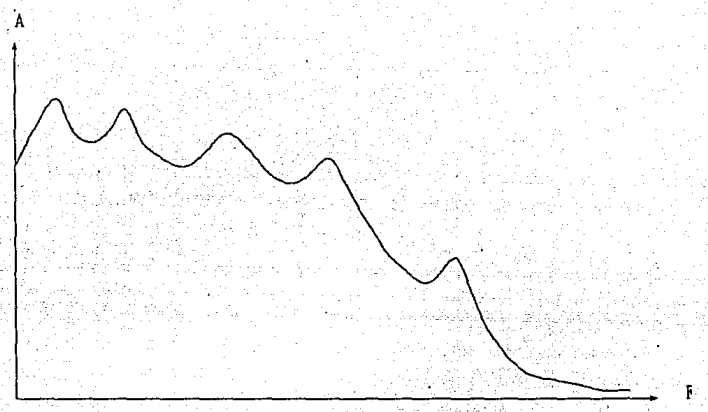


Fig. 4.6 Efecto de la radiación en la envolvente espectral de voz.

IV.2 DETECCION DE COMIENZO Y FIN DE PALABRA.

Un S.R.V. cuyo tipo de entrada son palabras aisladas, debe incluir un algoritmo que detecte el comienzo y el fin de una palabra; este algoritmo debe tener también la facultad de diferenciar entre un sonido cualquiera y una palabra.

Para detectar el comienzo de una articulación, se mide constantemente la energía de la señal de entrada, cuando ésta sobrepase un cierto umbral, significa que la señal ha sido suficientemente sonora como para que se le considere como posible inicio de una palabra. La autocorrelación cero R(0) de un vector de autocorrelaciones calculado a partir de una ventana de 128 muestras, se considera como la energía de la señal de entrada.

A partir de la detección de un posible principio de palabra, la energía de la señal en las siguientes ventanas, sobrepasará el umbral establecido; posteriormente, cuando se detecte un determinado número de ventanas donde la energía de la señal de entrada sea menor al umbral, se habrá encontrado el fin de la palabra, sin embargo, si el número total de vectores de autocorrelaciones con energía mayor al umbral es muy pequeño, significará que sólo se detectó un sonido cualquiera y no una palabra.

IV.3 REPRESENTACION PARAMETRICA DE LA SENAL.

Como se vió en el Capitulo II, el Reconocimiento de Patrones está dividido en tres estados o espacios. La digitalización de la señal pertenece al espacio de patrones, la representación paramétrica de la misma, al espacio de caracteristicas y la forma en como se van a agrupar los parámetros resultantes, al espacio de clasificación. Estos estados a su vez, están comprendidos dentro de los procesos de entrenamiento y reconocimiento, los cuales se explicarán más adelante (secciones IV.4 y IV.5).

Existen muchas razones por las que se desea reducir la dimensionalidad de los datos, como por ejemplo: la transmisión de una señal; ya que el canal de transmisión tiene un ancho de banda limitado y no es posible mandar todos los datos, motivo por el cual hay que mandar un número menor de ellos que representen a los originales sin que esta reducción presente una gran distorsión en la señal. Otra razón es que cuando la señal ha sido muestreada, es muy dificil manejar la gran cantidad de valores que se tiene, tanto para el proceso de entrenamiento, como para el de reconocimiento; además, la información está muy enmascarada y es dificil tratarla, por eso es necesario limitar esos valores.

IV.3.1 Medidas de distorsión.

Se necesita de una medida cuantitativa para saber qué tanta distorsión sufrió una señal al ser reducida en dimensionalidad y/o codificada y también qué tan buena fue la clasificación que se hizo de ella. Algunas medidas de distorsión son [11,14]:

A) Error Cuadrático.
B) Distorsión de Ley V-ésima.
C) Medidas de distorsión espectral.
D) Distorsión para el espectro de voz.
E) Medidas de distorsión simetrizadas.

F) Distorsión de Itakura.

Se define como:

$$di(f,g) = d'is(f,g) = \min_{\lambda \geq 0} dis(f, \lambda g) \tag{4.6}$$

G) Distorsión de Itakura-Saito discreta.

$$dz(f,g) = T_k/S_k - \ln(T_k/S_k) - 1 \tag{4.7}$$

H) Distorsión Itakura-Saito (dis) y distorsión de ganancia Itaura-Saito (dGN) [12].

$$dis(f, \hat{f}) = \int_{-\pi}^{\pi} \frac{d\varphi}{2\pi} \left[\frac{f}{\hat{f}} - \ln \frac{f}{\hat{f}} - 1 \right] \tag{4.8}$$

Que es la medida de distorsión entre dos espectros de potencia $f(\omega)$ y $\hat{f}(\omega)$ cuyos estimados son f y $\hat{f}$ que tienen la forma:

$$f(\omega) = \frac{G^2}{|A(z)|^2} \tag{4.9}$$

donde: $$A(z) = \sum_{k=0}^{M} a_k z^{-k} \tag{4.10}$$

donde: $$z = e^{(i\omega)} \tag{4.11}$$

La distorsión dGN se da por:

$$dGN(f,\hat{f}) = dis\left(\frac{f}{G^2} \frac{\hat{f}}{G^2} \right) = \frac{\alpha}{G^2} - 1 \tag{4.12}$$

donde: $$\alpha = r(0)\hat{r}_a(0) + 2 \sum_{n=1}^{M} r(n)\hat{r}_a(n) \tag{4.13}$$

donde: $$\hat{r}_a(n) = \sum_{i=0}^{M-n} a_i \, a_{i+n} \tag{4.14}$$

donde: a_i son los coeficientes utilizados en L.P.C.

r(n) son las autocorrelaciones de f(υ).
(estos dos parametros se explicarán posteriormente).

$$G^2 = r(0) + \sum_{i=1}^{M} a_i\, r(i) \qquad (4.15)$$

dis depende de la forma del espectro y de la ganancia (G), ésta medida se utiliza en el diseño de grupos de datos.
dGN depende sólo de la forma del espectro.

I) Medida de distorsión de ganancia optimizada de Itakura-Saito (dGO) [12].

$$d_{GO}(f,\hat{f}) = \min_{\lambda>0} d_{IS}(f,\lambda\hat{f}) \qquad (4.16)$$

$$= \ln \int_{-\pi}^{\pi} \frac{d\omega}{2\pi} \left[\frac{f}{\hat{f}}\right] - \int_{-\pi}^{\pi} \frac{d\omega}{2\pi} \ln \left[\frac{f}{\hat{f}}\right] \qquad (4.17)$$

$$= \ln(\alpha) - \ln(G^2) \qquad (4.18)$$

depende sólo de la forma del espectro, se utiliza como medida de distorsión de clasificación y para espectro L.P.C., se expresa:

$$d_{GO}(f,\hat{f}) = \ln(\alpha) - \ln(G^2)$$

IV.3.2 Codificación Linear Predictiva (L.P.C).

El aparato vocal se puede modelar de la siguiente forma [9,13,14]:

$$S(z) = E(z)G(z)V(z)L(z) \qquad (4.19)$$

donde:

$$E(z) = \sigma \sum_{n=0}^{\infty} z^{-n} = \sigma\,/(1 - z^{-1}) \qquad , |z|>1 \qquad (4.20)$$

$$G(z) = 1\,/(1 - e^{-CT} z^{-1})^2 \qquad (4.21)$$

$$L(z) = 1 - e^{-1} \qquad (4.22)$$

$$V(z) = 1 / \left[\prod_{i=0}^{k} (1 - 2e^{-CIT}\cos(biT)z^{-1} + e^{-2CIT}z^{-2}) \right] \qquad (4.23)$$

Por otro lado, los modelos de síntesis y análisis son:

Síntesis:

$$S(z) = E(z) / A(z) \qquad (4.24)$$

donde:

$$A(z) = \sum_{i=0}^{M} a_i z^{-i} \quad , \; a_0 = 1 \; ; \; M \geq 2k+1 \qquad (4.25)$$

es el filtro inverso, solo tiene polos, mientras que $1/A(z)$ solo tiene ceros.

$E(z)$ = es la excitación.

Análisis:

$$E(z) = S(z)\, A(z) \qquad (4.26)$$

Para que los procesos de clasificación de prototipos sobre los cuales se van a aplicar los algoritmos de decisión se lleven a cabo satisfactoriamente, se debe tener un modelo paramétrico de la señal de voz que calcule los parámetros de $A(z)$ y $E(z)$, los cuales es necesario agrupar para la clasificación. En este trabajo se usa el modelo de Codificación Linear Predictiva, en donde se utiliza la ganancia cuadrada inversa de un filtro (G^2) y los Coeficientes Lineales Predictivos ($a(k)$, $k = 1, 2, \ldots, M$).

Existen varias opciones para la estimación de éstos parámetros. En el cuadro siguiente se resumen las posibilidades que pueden tomarse [14].

Estimación de Parámetros.

Método de mínimos cuadrados.

Coeficientes a(k).

Señal determinística.

Método de autocorrelación.

Método de covariancia.

Señal aleatoria.

Caso estacionario.

Caso no estacionario.

Ganancia G.

Entrada= impulso.

Entrada= ruido blanco.

El modelo general de Predicción Lineal [9,13,14] se representa por medio de la siguiente ecuación:

$$S(n) = -\sum_{k=1}^{M} a(k)S(n-k) + G\sum_{l=0}^{q} b(l)U(n-l) \quad ; \quad b(0)=1 \tag{4.27}$$

donde:

S(n)= salida del sistema.

U(n)= entrada al sistema.

$a(k)= 1 \leq k \leq M$

$b(l)= 1 \leq l \leq q$

G = ganancia.

La función de transferencia de este modelo es:

$$H(z) = S(z) \,/\, U(z) \tag{4.28}$$

$$= G(z)(1 + \sum_{l=1}^{q} b(l)z^{-l})/(1 + \sum_{k=1}^{M} a(k)z^{-k}) \tag{4.29}$$

Se dice que se está prediciendo una señal S(n) con una combinación lineal de entradas anteriores más salidas anteriores. En el programa usado en este sistema se utilizó el modelo de polos (donde $b(l)= 0,\ 1 \leq l \leq q$), que permite un tratamiento lineal de la señal, pues los sonidos sonoros no nasales carecen de ceros para el modelo del aparato fonador y en los sonidos nasales y no sonoros, el modelo tiene polos y ceros y éstos últimos pueden aproximarse por polos múltiples. Tenemos entonces:

$$S(n) = -\sum_{k=1}^{M} a(k)S(n-k) + GU(n) \qquad (4.30)$$

Su función de transferencia es:

$$H(z) = G / (1 + \sum_{k=1}^{M} a(k)z^{-k}) \qquad (4.31)$$

En este modelo se calculan los coeficientes a(k) y la ganancia G por el método de mínimos cuadrados, donde se obtiene una señal $\hat{S}(n)$ a partir de las M muestras anteriores. La señal $\hat{S}(n)$ es el estimado de la señal S(n).

$$\hat{S}(n) = -\sum_{k=1}^{M} a(k)S(n-k) \quad ; \quad 1 \le k \le M \text{ , k-ésima muestra ant.} \qquad (4.32)$$

$$= a(1)S(n-1)+a(2)S(n-2)+\ldots+a(M)S(n-M) \qquad (4.33)$$

puesto que U(n) es desconocida.

La diferencia entre el valor estimado $\hat{S}(n)$ y el valor real S(n) es:

$$d = S(n)-\hat{S}(n) = S(n)+\sum_{k=1}^{M} a(k)S(n-k) \qquad (4.34)$$

$$= \sum_{k=0}^{M} a(k)S(n-k) \quad ; \quad 0 \le k \le M \text{ , } a(0) = 1 \qquad (4.35)$$

suponiedo a la señal como determinística, el valor cuadrático medio es:

$$E = \sum_{n} e^2(n) = \sum_{n} [\sum_{k=1}^{M} a(k)S(n-k)]^2 \qquad (4.36)$$

minimizando:

$$\partial \varepsilon / \partial a(i) = 0 = \sum_{k=0}^{M} a(k) \sum_{n} S(n-k)S(n-i) \quad ; \qquad (4.37)$$

$1 \le i \le M$, i-ésima muestra anterior.

si: $$\sum_{n} S(n-k)S(n-i) = C_{ki} \qquad (4.38)$$

donde: C_{ki} son los coeficientes de correlación.

entonces: $$\sum_{k=0}^{M} a(k)C_{ki} = 0 \qquad (4.39) \quad \Leftrightarrow \quad \sum_{k=1}^{M} a(k)C_{ki} = -C_{oi} \qquad (4.40)$$

pues: $$\sum_{k=1}^{M} a(k) \sum_{n} S(n-k)S(n-i) = -\sum_{n} S(n)S(n-i) \quad , \quad 1 \le i \le M \qquad (4.41)$$

Por el método de autocorrelación que minimiza el error de $-\infty < n < \infty$, la ec. 4.40 queda como:

$$\sum_{k=1}^{M} a(k)R(i-k) = -R(i) \qquad (4.42)$$

donde: $$R(i) = \sum_{n=-\infty}^{\infty} S(n)S(n+i) \qquad (4.43)$$

es la autocorrelación de S(n)

Multiplicando la señal S(n) por una ventana de orden N, tenemos una función de autocorrelación de la forma:

$$R(i) = \sum_{n=0}^{N-1} \hat{S}(n)\hat{S}(n+i) \quad ; \quad \text{n-ésima muestra en la ventana} \qquad (4.44)$$

De este modo, se tienen las ecuaciones 4.38, 4.40, 4.42 y 4.44 y se quieren calcular los parámetros deseados. Para calcular cada a(k), se tienen que calcular primero los coeficientes C_{ki} con la ec. 4.38:

$[S(n-k)]\ S(n-i)]=C_{ki}]$ ⟵ para cada i desde 1 a M ⟶
0<k<M; ∴ M sistemas diferentes.

⟵ cambia para cada sistema por la i.

⟵ en todas las i's es la misma matríz.

luego los coeficientes a(k) con la ec. 4.40:

$$[C_{ki}]\ a(k)] = -C_{oi}] \quad \Rightarrow \quad a(k)] = [C_{ki}]^{-1} - C_{oi}]$$

pero en el algoritmo utilizado en este trabajo, se utiliza la ec. 4.44 para encontrar las autocorrelaciones:

$$R(i)] = [\hat{S}(n)][\hat{S}(n+i)]$$

y la ec. 4.42 para encontrar los coeficientes a(k), de acuerdo al método

$$-R(i)] = [R(i-k)]\ a(k)] \quad \Rightarrow \quad a(k)] = [R(i-k)]^{-1} - R(i)]$$

Levinson-Durbin, puesto que este método trae consigo un ahorro bastante significativo en localidades de memoria, operaciones y con ello, en tiempo:

$$E(0) = R(0) \tag{4.45}$$

$$R(i) = -\sum R(i) + \sum_{j=1}^{i-1} a(j,i-1)R(i-j) \ / E(i-1) \tag{4.46}$$

$$a(i,j) = k(i) \tag{4.47}$$

$$a(i,j) = a(j,i-1) + k(i)a(i-j,i-1) \tag{4.48}$$

$$E(i) = (1-k(i)^2)\ E(i-1) \tag{4.49}$$

Una vez que se han obtenido los parámetros de L.P.C., se va a buscar un método adecuado para la codificación de dichos parámetros. Los métodos comúnmente empleados, están basados en la clasificación de patrones por medio de funciones de distancia. En este trabajo se utilizó la cuantización vectorial, la cual se explicará más detalladamente a continuación.

IV.3.3 Cuantización Vectorial.

Es un principio de compresión de datos entre cuyas aplicaciones se encuentran la codificación y reconocimiento de voz [12]. En el sistema que estamos tratando, el tipo de entrada son palabras aisladas, que con éste método se clasifican por medio de la distorsión promedio (de la que ya hemos hablado) que resulta de codificarlas con códigos o alfabetos (vectores de reproducción) con la técnica de Cuantización Vectorial (V.Q.), que es una técnica de codificación de voz, de ancho de banda estrecho, basada en la Codificación Linear Predictiva, y que como sabemos, va a comprimir un número N de datos, en un vector índice con menos elementos (M), los cuales representan a los primeros. Esto trae como consecuencia, ventajas en los cálculos, tiempo de procesamiento y requerimientos de memoria, pues los reduce considerablemente.

Un código o alfabeto V.Q. se genera aplicando una técnica de agrupamiento iterativa. En éste algoritmo de agrupamiento, se representa cada palabra del vocabulario, como un conjunto de espectros independientes, ya que el presente, es un sistema de vocabulario limitado.

IV.3.3.1 Conceptos.

Para tener una idea más clara de éste algoritmo, es conveniente precisar la manera como se van a formar los conjuntos de datos, el nombre que se da a cada grupo y la relación que guarda cada uno de ellos con ciertas divisiones hechas a cada palabra del vocabulario; así como algunos otros conceptos relacionados con el proceso [12].

Consideremos la fig. 4.7 En donde se está representando el conjunto de palabras del vocabulario de reconocimiento, Tenemos entonces que:

$k = 1,2,\ldots,V$

donde: k es la k-ésima palabra,

V es el número de ellas.

La k-ésima palabra está dividida en T_k articulaciones.

$q = 1,2,\ldots,T_k$

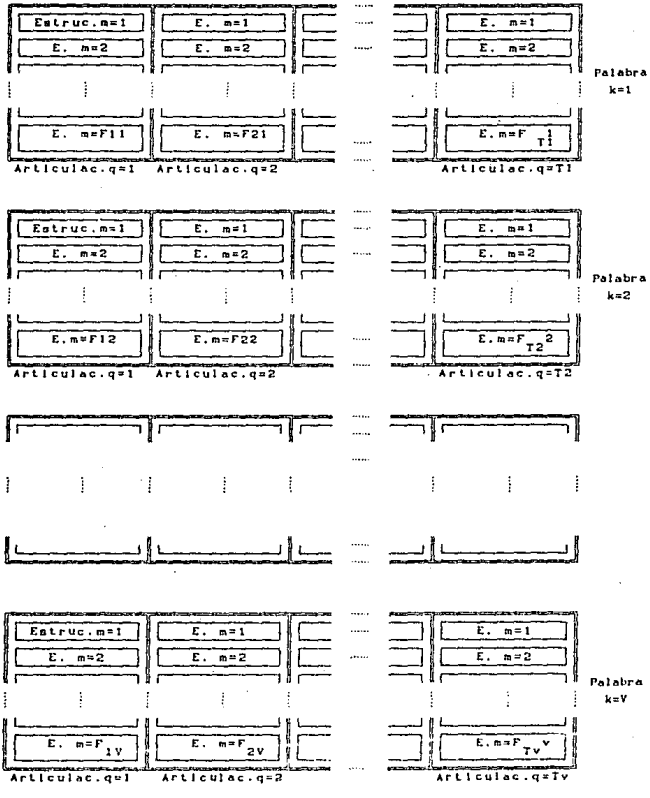


Figura 4.7 Representación del Conjunto de Palabras del vocabulario de reconocimiento.

donde: q es la q-ésima articulación en la secuencia de entren. y
T_k es el número de ellas en la k-ésima palabra.

Cada articulación q de la palabra k comprende F_{qk} estructuras
$m = 1,2,\ldots,F_{qk}$
donde: m es la m-ésima estructura= U_{mqk} y
F_{qk} es el número de ellas en la q-ésima articulación de la k-ésima palabra.

Por otro lado:
A) Conjunto de códigos o afabetos C- Comprende una serie de V códigos o alfabetos multisección C_k.

B) Código o alfabeto C_k- Representa a la k-ésima palabra del voc. de rec. ∴ hay V C_k's. Está diseñado a partir de T_k articulaciones. Comprende una secuencia de códigos o alfabetos de sección C_{kj}.

C) Código o alfabeto de sección C_{kj}- Representa a la q-ésima articulación de la k-ésima palabra del vocabulario de reconocimiento.
$j = 1,2,\ldots,N_k$
donde: j es j-ésimo código o alfabeto y
N_k es el número de ellos en el código o alfabeto multisección C_k.
Comprende una serie de palabras de código C_{kji}. Está diseñado a partir de n estructuras de cada secuencia de entrenamiento ó U_{mqk}.

Factor de Compresión n- Es el número de estructuras en cada sección de la palabra. Si $n = (j-1)n+1,\ldots,jn$. C_{ki} se diseña a partir de las primeras n estructuras U_{mqk}.

D) Palabra de código C_{kji}- Representa a la m-ésima estructura, de la q-ésima articulación en la k-ésima palabra del vocabulario de reconocimiento.
$i = 1,2,\ldots,N_{kj}$

donde: i es la i-ésima palabra de código y
N_{kj} es el número de ellos en el código o alfabeto C_{kj} del código o alfabeto multi sección C_k.

Comprende un número de parámetros L.P.C.

E) Parámetros L.P.C.- Ganancia cuadrada inversa G^2, coeficientes linealmente predictivos a(k), k= 1,2,...,M.

F)Longitud normalizada- Las secciones de palabra de entrada y códigos de sección, tienen igual longitud.

Tenemos además que existen tres tipos de códigos o alfabetos:

A) Código de tamaño fijo- Todos los Códigos de sección en un C_k, están formados por el mismo número N_{kj} de palabras de código. Donde $N_{kj}= 2^{r_k}$ y r_k es la razón de C_{kj}.

B) Código de distorsión fija- Al generar cada código se va incrementando su tamaño hasta lograr que codifique la secuencia de entrenamiento con una distorsión promedio ≥ T, por lo cual, los códigos de sección en un C_k, varían en tamaño.

C) Código sin agrupar- Se genera sin el algoritmo de agrupamiento, simplemente haciendo palabras de código fuera de cada estructura en la secuencia de entrenamiento.

Los parámetros para la generación de códigos o alfabetos son:

A) Número de articulaciones en la secuencia de entrenamiento.
B) Umbral de energía:

$$E= \sum_{i=1}^{M} X_i^2 \qquad (4.50)$$

donde:

W ancho de ventana.

Xi muestras después de preenfatizar.

C) Tipo de alineamiento ("left" o de longitud normalizada).

D) Factor de compresión n.

E) El tipo de código y medida: Código de tamaño fijo; $T= 2^2$.

Los parámetros para la clasificación de articulaciones son:

A) Factor de compresión n.

B) Alineamiento de articulaciones.

C) Umbral de energía.

IV.3.3.2 Algoritmo.

Existen dos pasos a seguir en el Algoritmo de Cuantización Vectorial [11], el primero de ellos consiste en encontrar un alfabeto inicial de reproducción Ao partiendo de un solo centroide el cual se perturba hasta lograr que Ao tenga un determinado número de elementos. El segundo paso consiste en calcular un alfabeto de reproducción óptimo a partir de Ao.

Para lograr un mejor cálculo de patrones, en el presente trabajo el alfabeto de reproducción óptimo se va a calcular en cada paso de perturbación de Ao hasta alcanzar el número deseado de elementos. De este modo, el Algoritmo de Cuantización Vectorial queda:

Sea f= 0,1, ... ,N

donde: f= f-ésimo nivel de partición.

N= niveles totales de partición.

$_a$= 0,1, ... ,N(f)

donde: a = a-ésimo nivel de cuantización para el nivel de partición f.

N(f)= niveles totales de cuantización para el nivel de partición f. $N(f)= 2^f$.

M = longitud de bloque.

n = longitud de la secuencia de entrenamiento.

ε = umbral de distorsión.

ε_p = elemento fijo de perturbación.

0) $N(0)= 2^0= 1$; $\varepsilon_p = 0.001$; $\varepsilon_p = 0.001$;
{Xj; j= 0,1, ... ,n-1}= secuencia de entrenamiento.
Ao(1)= $\hat{X}(A)$= centroide de la secuencia de entrenamiento.

1) Dado el alfabeto de reproducción Ao(N(f)) que contiene N(f) vectores {Ya, a= 1, ... ,N(f)}, perturbar cada vector Ya en dos vectores cercanos Ya + ε_p y Ya - ε_p. La colección $\hat{A}$ de:
{Ya + ε_p, Ya - ε_p; a= 1,2, ... ,N(f)} tiene 2N(f) vectores. Hacer f= f+1, es decir, reemplazar N(f) por N(f+1)= 2N(f).

2) Ao= $\hat{A}$(N(f)). Entonces Ao es el alfabeto inicial de reproducción para el algoritmo de cuantización del nivel de perturbación f. Fijar m=0, $D_{-1}= \infty$.

a) Dado Am= {Ya; a= 1, ... ,N(f)}. Encontrar la partición de mínima distorsión P(Am)= {Sa; a= 1, ... ,N(f)}.
Si d(Xj,Ya) ≤ d(Xj,Yi) para toda i, entonces Xj ∈ Sa.

$$Dm= D(\{\hat{Am}, P(\hat{Am})\})= \frac{1}{n}\sum_{j=0:Y\in Am}^{n-1} \min d(Xj,Y) \qquad (4.51)$$

b) Si $(Dm-1 - Dm)/Dm \leq \varepsilon$, entonces el alfabeto de reproducción óptimo es Am. Ir a 3).

c) Encontrar el alfabeto de reproducción:
$\hat{X}(P(\hat{Am}))= \{\hat{X}(Sa);\ a= 1, \ldots ,N(f)\}$ para $P(\hat{Am})$. $\hat{X}(Sa)$ es el

centroide Euclideano dado por:

$$X(S_a) = \frac{1}{||S_a||} \sum_{j : X_j \in S_a} X_j. \qquad (4.52)$$

donde $||S_a||$ es el número de vectores de entrenamiento en la celda S_a. Si $||S_a|| = 0$, hacer $\hat{X}(S_a) = Y_a$. Definir $A_{m+1} = \hat{X}(P(A_m))$; m=m+1. Ir a a).

3) ¿ Es N(f)= N(N) ?. Si la respuesta es afirmativa, el algoritmo termina, en caso contrario, ir a 1).

IV.4 ENTRENAMIETO O APRENDIZAJE.

En apartados previos, hemos hablado de algunas técnicas de clasificación de datos y como sabemos, el entrenamiento o aprendizaje es la construcción y ajuste de funciones discriminantes, las cuales miden cada punto en el patrón o espacio de características, que particionan en regiones mutuamente exclusivas y asignan a ese punto un valor.

Este proceso se lleva a cabo para generar los patrones con respecto a los cuales se va a hacer una comparación (con ayuda de medidas de distancia) de las palabras que se pronuncien con el objeto de que sean reconocidas.

El primer paso en este trabajo, consiste en hacer la digitalización de la señal de voz de entrada, proceso que se realiza tanto en el entrenamiento o aprendizaje, como en el reconocimiento. Esto se logra de la manera en que se trató el asunto en la sección IV.1

Cuando la señal de voz ha sido digitalizada, es preciso representarla paramétricamente. En los programas que se utilizaron para este trabajo, se hace uso de la técnica L.P.C., de la que ya hemos hablado, para el cumplimiento de este objetivo.

Para el caso del entrenamiento, se va a repetir cada palabra del vocabulario un cierto número de veces. Para cada pronunciación de la palabra a entrenar, se calculará un vector de autocorrelaciones R(i) de orden M, cada 128 muestras de la señal de voz. El número total de vectores de autocorrelaciones, es decir, la palabra, se divide en cuatro partes (que más tarde se considerarán articulaciones dentro del algoritmo de Cuantización Vectorial).

Hasta este momento, los procesos de entrenamiento y reconocimiento son muy similares, la diferencia está en que en el segundo caso, la palabra a reconocer se va a pronunciar una vez. Para el caso del aprendizaje, el siguiente paso es agrupar datos con el Método de Cuantización Vectorial.

Una Cuantización Vectorial se diseña a partir de una secuencia de entrenamiento. Se consideran, para nuestro caso en particular, cuatro secuencias de entrenamiento, cada una de ellas coresponde a una articulación y está formada por todos los vectores de autocorrelaciones

para esa articulación de todas las repeticiones de la palabra cuyos patrones se quieren almacenar.

Los datos de la secuencia de entrenamiento se cuantizan vectorialmente y de aquí se obtienen los patrones (correlaciones Ra(k) de los coeficientes a(k)). Para exponer esto de manera más clara, relacionaremos los parámetros LPC aquí mencionados, con el algoritmo de Cuantización Vectorial (sec. IV.3.3.2) [11].

Inicio.

Se tiene un solo nivel de cuantización ($N(0)=2^0=1$),

Una secuencia de entrenamiento Xj : Rj(i)= vectores de autocorrelaciones donde:

j= 0,1, ... ,n-1 e

i= 0,1, ... ,M-1

Con estos vectores se calculará un alfabeto de reproducción inicial Am para un nivel A0 {Y_a ; a=1} donde:

$$Y_1 = R(1) = \frac{1}{n} \sum_{j=0}^{n-1} Rj(1)$$: centroide de la secuencia de entrenamieto.

A partir de este centroide de autocorrelaciones, se calculan los coeficientes de L.P.C. a(k) con el algoritmo de Levinson-Durbin (sec. IV.3.2, ecs. 4.45 a 4.49) y los coeficientes de reflexión o COrrelaciones PARciales Parcor (vector K): K(i); i= 0,1, ... ,M-1.

Paso 1:

El vector K es precisamente el que se perturbará, entonces nuestro nivel de perturbación cambiará de 0 a 1 y $N(1)= 2^1= 2$, obteniéndose:

$$K(i) \pm c_p = K_a(i);\ a= 1, \ldots, N(1).$$

Con éstos, se obtienen los vectores de coeficientes $a_a(k)$, posteriormente se va a seguir con un cálculo de correlaciones entre los

coeficientes de cada vector:

$$Ra_a(i)= \sum_{k=0}^{N-i} a_a(k)a_a(k+i) \; ; \qquad a= 1, \ldots, N(f)$$

Paso 2:

Se calcula la partición de mínima distorsión:

$\hat{P}(Am)= \{S_a \; ; \; a= 1, \ldots, N(f) \}$.

La distorsión se calcula entre cada vector de autocorrelaciones Rj(i) de la secuencia de entrenamiento con respecto a cada vector de correlaciones $Ra_a(i)$ con la medida de distancia de Itakura-Saito.

Paso 3:

Se vuelven a calcular N(f) centroides con las autocorrelaciones Rj(i) que quedaron asignadas a las celdas S_a :

$$X(S_a)= \frac{1}{|Sa|} \sum_{j:Xj\in Sa} Xj \qquad : \qquad R_a(i)= \frac{1}{|Sa|} \sum_{j:Rj(i)\in Sa}^{|ga|} Rj(i)= Y_a$$

que es nuestro alfabeto de reproducción Am= Ai{ Ya; a=1,2 }.

Se continúa con el proceso de acuerdo al algoritmo presentado en IV.3.3.2, hasta encontrar el alfabeto de reproducción óptimo para ese nivel de partición. A partir de este alfabeto se vuelven a calcular los coeficientes a(k) y el vector K que se perturbará hasta que N(f)= 4 (estructura de la palabra). El algoritmo termina cuando para N(f)= 4 se haya encontrado el alfabeto de reproducción óptimo, con sus respectivos vectores aa(k) y Raa(k). Estos últimos son los códigos o alfabetos patrón de la palabra para una articulación específica.

En resumen, de cada palabra, se van a obtener cuatro articulaciones, cada una de ellas formada por cuatro estructuras, en total, dieciseis por palabra.

La codificación de una palabra, trae como consecuencia una cierta distorsión promedio, la cual se minimiza con la Cuantización Vectorial. De

este modo, un código o alfabeto C, se diseña de tal forma que [ ]:

$$C = \frac{1}{L} \sum_{j=1}^{L} \min_{i} d(T_j, C_i) \qquad (4.53)$$

donde: T_j es una secuencia de entrenamiento y
C_i es una palabra de código.

Además, la forma del espectro de la j-ésima estructura se codifica:

$$d(S_j, C_b) = \min_{i} d(S_j, C_i) \qquad (4.54)$$

donde: S_j es el estimado de autocorrelación de la j-ésima estructura.
C_b palabra de código a la que mejor representa S_j.

Si la secuencia de entrenamiento se representa por el código o alfabeto C con una distorsión pequeña, ese mismo código o alfabeto codificará voz con una distorsión similarmente pequeña. Cada palabra nueva, se clasifica encontrando el código o alfabeto V.Q. con el cual sufra la menor distorsión promedio al codificarse.

Para el sistema con que se cuenta, un programa en la Computadora Personal (P.C.), indica al orador cuando pronunciar una palabra, el locutor introduce su voz al TMS32010, el microprocesador entonces, calcula las correlaciones en tiempo real (ciclo de instrucción del TMS32010: 200 nseg), éstas se envian posteriormente a la P.C. por medio del puerto paralelo-serie y se almacenan en la memoria de la computadora. Después, en la P.C, se dividen los datos de cada palabra en cuatro partes y se procesan para calcular los coeficientes a(k) y las correlaciones de las a(k)'s. Cada parte se almacena en archivos en la P.C.

Por último, se mezclan dichos archivos con los de las demás palabras del vocabulario, de tal manera que queden juntos y en el mismo orden, los archivos que corresponden a cada una de las cuatro partes de cada palabra hasta formar un solo archivo con todas las correlaciones $R_a(1)$ de todas las palabras, mismos que serán los patrones de un cierto vocabulario, concluyendo con ésto el proceso de entrenamiento o aprendizaje.

IV.5 RECONOCIMIENTO.

El objetivo del proceso de entrenamiento, es la generación de los patrones de reconocimiento, mientras que, el objetivo del proceso de reconocimiento es reconocer de entre un vocabulario, la palabra que se haya pronunciado.

La primera parte del reconocimiento, es la digitalización de la señal de voz de la palabra; ésta entra al TMS32010 donde es filtrada y luego muestreada. A cada bloque de 128 muestras, se le calcularán sus autocorrelaciones y una vez que se hubo detectado que esa señal de entrada es en efecto de una palabra y no de un ruido cualquiera, las correlaciones calculadas se almacenan en la memoria del microprocesador.

Después de ésto, nuevamente se tiene un conjunto de datos que se deben codificar; para ello, la palabra desconocida, que se quiere reconocer (número total de vectores de autocorrelaciones), se divide en varias secciones (cuatro en nuestro caso) y para cada una de éstas, se diseña un código o alfabeto de sección; después, cada uno de ellos se compara con los códigos o alfabetos de la sección correspondiente del patrón. La palabra desconocida se clasifica de acuerdo al código o alfabeto con respecto al cual se haya obtenido la mínima distorsión promedio como resultado de la comparación.

Es decir:

Se tiene el código o alfabeto patrón C_{kji}.

$C_{kji} = R_a(l)_{kqm}$

donde: k No. de palabra, k=1, 2, ..., 5

j=q No. de articulación, j=q= 1, 2, ..., 4

i=m No. de estructura, i=m= 1, 2, ..., 4

$R_a(l)$ Correlación de a.

donde: l orden de la correlación de a's

l= 0, 1, 2, ..., 15

y el código o alfabeto de la palabra de entrada C_{kji}'

$C_{kji}' = R(l)$

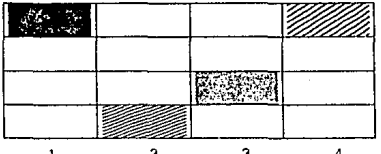
donde: R(l) es la correlación de las muestras de entrada.

Entonces se encuentra la distancia mínima d_j^n para cada código o alfabeto de sección C_{kj}.

$$d_j^n = \min_i \left[R(0)^n R_a(0) + 2 \sum_{l=0}^{15} R(l)^n R_a(l)_{kjl} \right] \qquad (4.55)$$

y la distancia mínima D_j para el código multisección C_k.

$$D_j = \sum_{n=1}^{N} d_j^n \qquad (4.56)$$



$$d_j^1 + d_j^2 + d_j^3 + d_j^4 = D_j$$

de este modo, la palabra reconocida r es aquella que tenga la D_j menor, es decir:

$$D_r = \min_k D_k \qquad (4.57)$$

si se desea, se puede fijar un valor de umbral D_{min} para que:

$$D_r < D_{min} \qquad (4.58)$$

Cada palabra tiene una clave especial, por lo tanto, la clave de la palabra reconocida, se almacenará en alguna localidad de memoria de datos del TMS, para transmitirse posteriormente a la P.C.

Para saber qué tan bueno fue el reconocimiento de una palabra, se puede utilizar un criterio que dá por hecho que Dr es la menor distorsión promedio de todas las distorsiones y D' es la segunda menor distorsión promedio. Por tanto:

$$R = \frac{D' - D_r}{D_r} \qquad (4.59)$$

Si R>0, la clasificación es correcta, si R<0, es incorrecta.

Otro criterio es simplemente contar los aciertos y errores en el reconocimiento.

Con ésto, se han terminado de explicar los principios teóricos para la parte de reconocimiento y las principales características para la definición del tipo de sistema empleado en esta etapa. Más adelante, en el diagrama de bloques y en el de flujo de las figuras 4.8 y 4.9 respectivamente, se apreciará todo este proceso dentro de la parte de software del TMS32010 junto con la parte que se le agregó correspondiente a la programación de la interfase para transmisión de datos TMS - P.C. y la integración de ambas etapas en lenguaje ensamblador en el sistema general de Reconocimiento de Comandos Hablados.

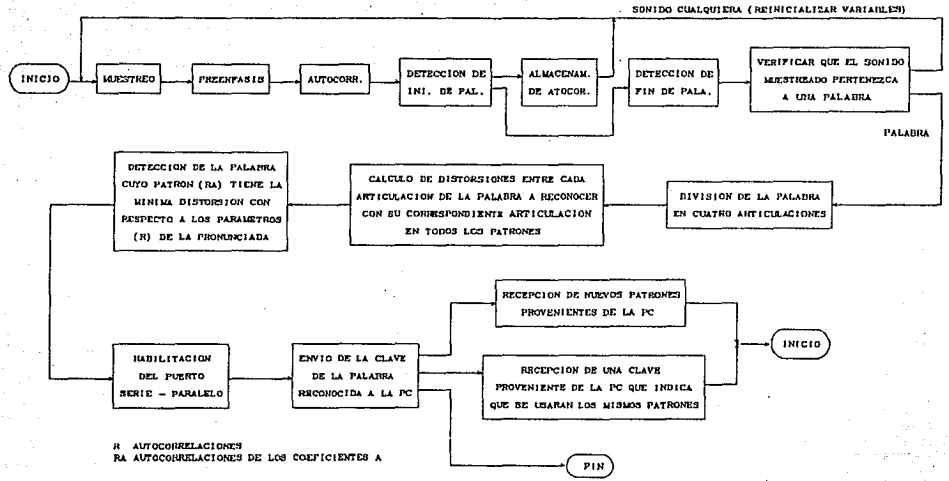


Fig. 4.8 Diagrama de bloques del Software en el TMS32010 del Sistema de Reconocimiento de Comandos Hablados.

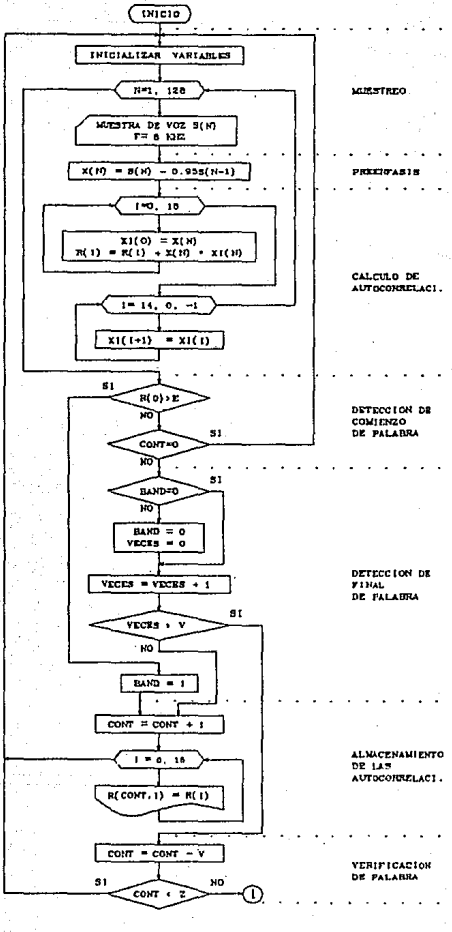

INICIO
INICIALIZAR VARIABLES
N=1, 128
MUESTREO
MUESTRA DE VOZ S(N)
F= 8 KHZ
X(N) = S(N) - 0.955(N-1)
PREENFASIS
I=0, 10
XI(0) = X(N)
R(I) = R(I) + X(N) * XI(N)
CALCULO DE AUTOCORRELACI.
I= 14, 0, -1
XI(I+1) = XI(I)
SI
R(0)>E
NO
DETECCION DE COMIENZO DE PALABRA
CONT=0
SI
NO
SI
BAND=0
NO
BAND = 0
VECES = 0
VECES = VECES + 1
DETECCION DE FINAL DE PALABRA
SI
VECES > V
NO
BAND = 1
CONT = CONT + 1
I = 0, 10
ALMACENAMIENTO DE LAS AUTOCORRELACI.
R(CONT,I) = R(I)
CONT = CONT - V
VERIFICACION DE PALABRA
SI
CONT < Z
NO
1

Fig. 4.9 a).

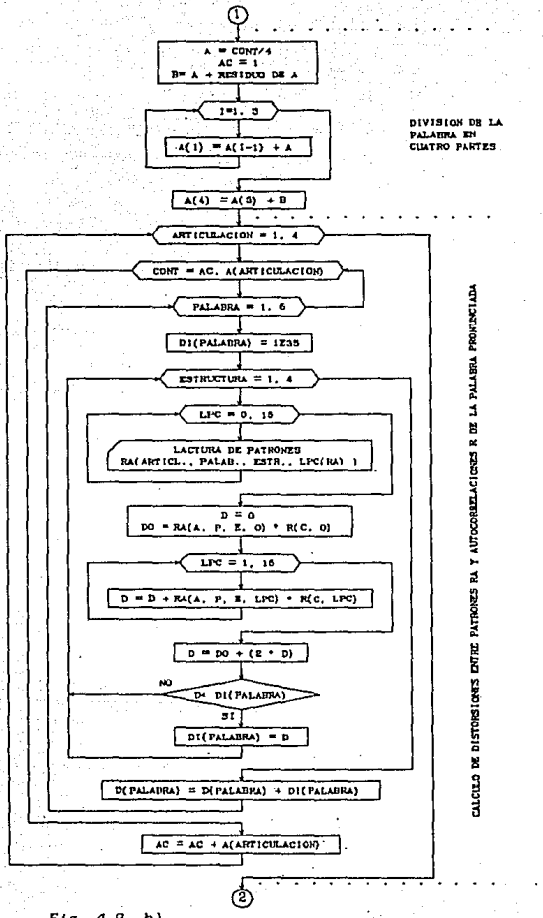

1
A = CONT/4
AC = 1
B= A + RESIDUO DE A
I=1, 3
A(I) = A(I-1) + A
DIVISION DE LA PALABRA EN CUATRO PARTES
A(4) = A(3) + B
ARTICULACION = 1, 4
CONT = AC, A(ARTICULACION)
PALABRA = 1, 6
DI(PALABRA) = 1E35
ESTRUCTURA = 1, 4
LPC = 0, 16
LACTURA DE PATRONES
RA(ARTICL., PALAB., ESTR., LPC(RA))
D = 0
DO = RA(A, P, E, 0) * R(C, 0)
LPC = 1, 16
D = D + RA(A, P, E, LPC) * R(C, LPC)
D = DO + (2 * D)
NO
D< DI(PALABRA)
SI
DI(PALABRA) = D
D(PALABRA) = D(PALABRA) + DI(PALABRA)
AC = AC + A(ARTICULACION)
2
CALCULO DE DISTORSIONES ENTRE PATRONES RA Y AUTOCORRELACIONES R DE LA PALABRA PRONUNCIADA

Fig. 4.9 b)

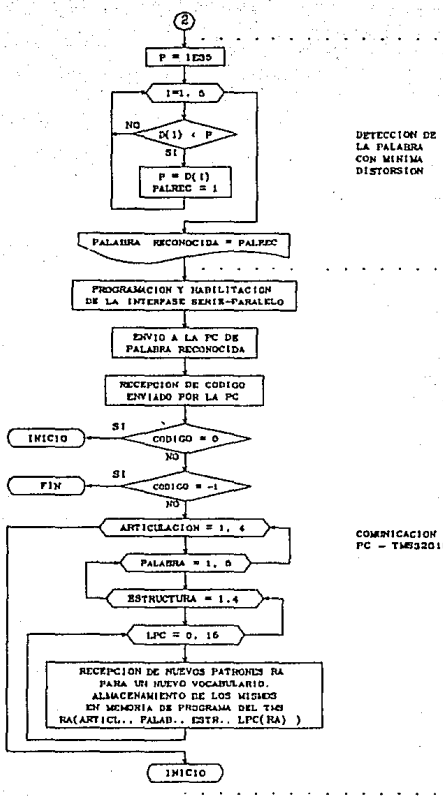


Fig. 4.9 c)

DONDE :

F = 8 KHZ
V = 2 = 9
H = 250 h
P = 1 E35

Diagrama de flujo del Software en el TMS32010 del Sistema de Reconocimiento de Comandos Hablados.

CAPITULO V

EVALUACION DEL S.R.V.

Los S.R.V.'s serán más confiables entre menos varien las condiciones bajo las que operan. Uno de los factores que más interviene en el funcionamiento de un S.R.V. es el ruido, por lo cual es preferible que antes de llevar a cabo el entrenemiento y/o reconocimiento se califiquen sea cuantitativa y/o cualitativamente estas condiciones, a fin de que se minimice el ruido y se traten de reproducir las mismas caracteristicas ambientales cada vez que se haga uso del sistema.

Las principales fuentes de ruido que pueden afectar un S.R.V. son: ruido de fondo, ruido de respuesta en frecuencia del teléfono (o micrófono) empleado, ruido de reverberación, etc.

Dicho lo anterior, a continuación mencionaremos las especificaciones bajo las cuales se llevaron a cabo las pruebas para determinar el tipo de orador que generó los vocabularios para el sistema ejemplo.

V.1 ESPECIFICACIONES.

Un sistema global de Reconocimiento de Comandos Hablados puede tener varias posibilidades de funcionamiento, de manera que el sistema final es el resultado de una serie de decisiones tomadas a lo largo de varios niveles, cada uno de ellos con varias opciones, lo cual forma una especie de árbol.

Los factores que definen un S.R.V. ya han sido mencionados en el Capítulo II. Para nuestro sistema particular, se tomaron las siguientes decisiones para cada uno de ellos:

A) Tipo de voz de entrada: palabras aisladas.

El sistema que se presenta es de palabras aisladas, siendo éste uno de los más sencillos, puesto que para voz conectada, la complejidad

aumenta entre otras cosas, por la determinación del final de una palabra y el comienzo de otra.

B) Tamaño de la población: un solo orador, dos oradores.

En la generación del vocabulario para reconocimiento, pueden intervenir varios individuos (sistema independiente del locutor) o uno solo (sistema dependiente del locutor o adaptivo). El primero es más difícil que el segundo, ya que es necesario establecer un conjunto de parámetros característicos de cada sonido, independientemente del locutor que los articule. Los parámetros que caracterizan un S.R.V. de este tipo son:

- Distribución temporal de energía.
- Distribución temporal de cruces por cero.
- Distribución espectral de energía.
- Distribución de zonas de sonidos sonoro-sordos.
- Error residual de L.P.C.

Como éste es un sistema dependiente del locutor, en los experimentos también se pretende comprobar esta afirmación.

C) Tipo de oradores: hombre, mujer.

Nosotros generamos vocabularios de varios tipos para los experimentos: sólo un orador masculino, sólo un orador femenino, y dos oradores; uno masculino y uno femenino.

D) Tamaño del vocabulario: vocabularios de cinco palabras (dígitos del uno al cinco para experimentos).
cinco vocabularios de cinco palabras y un vocabulario de dos palabras (para el S.R.V. definitivo).

A medida que un vocabulario aumenta, dejan de tener validez métodos de reconocimiento más simplificados, que se consideran válidos para vocabularios más restringidos, pues la complejidad de un S.R.V. aumenta

en razón directa al tamaño del vocabulario.

Los vocabularios del SRV son cinco de cinco palabras más uno de dos palabras, acomodados en forma de árbol y pudiendose intercambiar de manera que parezca un sólo vocabulario de 23 palabras, puesto que una de ellas se repite en cuatro vocabularios y otra en dos.

E) Ambiente en que se habla: laboratorio de computadoras.

El ambiente en que se habla se ve afectado por algunas fuentes de ruido, por ejemplo el ruido de fondo y el de reverberación.

El ruido de fondo se produce por lámparas fluorescentes, computadoras, conversaciones de fondo, pasos, tráfico externo, puertas, aire acondicionado, ventiladores, máquinas de escribir, etc. Este ruido es aditivo por naturaleza y dependiendo del ambiente varía de 60 a 90 dB.

El ruido de reverberación se presenta al hablar en una habitación con superficies reflejantes y a diferencia del ruido de fondo, es multiplicativo.

El laboratorio de computadoras donde se hizo el entrenamiento y las pruebas, se ve afectado por las primeras seis fuentes de ruido de fondo mencionadas anteriormente y para minimizar sus efectos, el aprendizaje se realizó en días no laborables y con las luz apagada, únicamente con el ruido producido por la computadora y las pruebas se hicieron tanto en días no laborables, como en días hábiles con actividad normal.

F) Medio de transmisión: micrófono.

Aún los movimientos más leves del orador, relativos al micrófono, causarán fluctuaciones en el nivel de voz y para evitar ésto, es recomendable que el micrófono esté fijo.

El nivel promedio de voz aumenta 3 dB. cada vez que hay una disminución de aproximadamente 2.54 cm en la distancia entre el micrófono y el orador. Cuando esta distancia es menor que 2.54 cm se tiene un nivel de voz de 90 a 100 dB. [4].

La separación empleada en este sistema, entre orador y micrófono es en promedio de 12 cm. (nivel de voz aproximado de (90 dB. a 100 dB.) -

11.17 dB. = 78.83 a 88.83).

Por otro lado, los aspectos que deben tomarse en cuenta en un vocabulario limitado, son:

A) Digitalización de la señal.

El TMS32010 se programa para muestrear la señal de voz a una tasa de 8 KHz. (0.125 mseg/muest.) y calcular los parámetros L.P.C. cada 128 muestras, es decir cada 16 mseg. Dado que el TMS32010 tiene un ciclo de instrucción de 200 nseg, el programa de reconocimiento que se utilizó, optimiza el tiempo entre muestra y muestra (625 ciclos de reloj), aprovechándolo para calcular los parámetros L.P.C. después de someter cada muestra a un filtrado para preénfasis.

B) Detección de comienzo y final de una palabra.
(ver sección IV.2).

C) Representación paramétrica de la unidad a reconocer.

C1) Cálculo de los parámetros L.P.C. (ver secc. IV.3.2)

a) Autocorrelaciones R(i) ($0 \leq i \leq 15$) (Orden M= 16).

b) Coeficientes de reflexión o Correlaciones parciales (parcor) K(i) ($1 \leq i \leq 15$).

c) Coeficientes del filtro a(i) ($0 \leq i \leq 15$).

d) Correlaciones de los coeficientes a(i), Ra(i) ($0 \leq i \leq 15$).

La elección del orden M del predictor, depende de la frecuencia de muestreo.

El espectro de voz bajo análisis y por lo tanto del tracto vocal, se representan por un par de polos complejos conjugados por cada KiloHertz, ya que el valor de los formantes en este espectro, depende, entre otras cosas, de la forma y tamaño del tracto vocal. De acuerdo al Teorema de Nyquist, para una frecuencia de muestreo de 8 KHz., se requieren de 8 polos más tres o cuatro para representar el espectro de excitación y los efectos de radiación. Esto hace un

total de M= 12, sin embargo, entre mayor es M, mejor es el predictor [9]; por lo tanto, se escoge M=16.

C2) Codificación de parámetros.

El agrupamiento de datos se hizo mediante la generación de códigos o alfabetos con el algoritmo de Cuantización Vectorial. Los valores que se dieron a los parámetros para la generación de estos códigos son:

a) Número de articulaciones en la secuencia de entrenamiento: 4.

b) Umbral de energía: R(0)= 250h(Q15)= 592d(Q15)= 0.00762963

$$E= R(i)= \sum_{n=1}^{W} S(n)S(n+i) \qquad ; \qquad i=0,\ W=128$$

El tamaño W de la ventana, debe ser lo suficientemente largo para que los efectos de ahusamiento de ésta, no afecten seriamente los resultados. Generalmente, la duración utilizada para la ventana va de W= 100 a W= 400 muestras (a una frecuencia de muestreo de 10 KHz.) [9]. En este sistema se usó W= 128.

c) Factor de compresión n: 4.

d) Tipo y tamaño de código: Código de tamaño fijo T= 2^2= 4.

D) Normalización de la señal de entrada.

De acuerdo a las características del convertidor A/D de la tarjeta AIB (Apéndice B), la señal de entrada debe abarcar un rango no mayor de ± 10 Volts.

E) Medida de distancia en el algoritmo de comparación.

Distorsión de ganancia de Itakura-Saito (IV.3.1 - h)).

V.2 EXPERIMENTOS Y RESULTADOS.

Se generaron tres vocabularios de números del uno al cinco, cada uno por un tipo de orador diferente. Cada tipo de orador repitió quince veces cada digito durante el entrenamiento. Después, durante las pruebas, cada orador repitió treinta veces cada palabra de cada vocabulario, en total 150 repeticiones por vocabulario.

VOCABULARIO	ORADOR QUE GENERO EL VOC.
A	ORADOR FEMENINO F
B	ORADOR MASCULINO M
C	ORADORES FEM. F Y MASC. M

tabla 5.1

Según los aciertos contra los errores, se obtuvo un porcentaje promedio de eficiencia. En la tabla 5.2 se resumen los resultados obtenidos

ORADOR / VOCABULARIO	F			M		
	ACIERTOS	ERRORES	% EXITO	AC.	ER.	% E
A	146	4	97.33	106	44	70.67
B	101	49	67.33	146	4	97.33
C	119	31	79.33	105	45	70

Tabla 5.2

Para un conocimiento más detallado de la información pasada, las tablas 5.3 a 5.8, muestran la matriz de confusión de cada caso, es decir, el número de veces que un digito articulado se identificó con cada digito del vocabulario.

VOCABULARIO: A		ORADOR: F				
		DIGITO ARTICULADO				
		1	2	3	4	5
DIGITO RECONOCIDO	1	29	2			
	2	1	28		1	
	3			30		
	4				29	
	5					30
ERRORES		1	2	0	1	0
% EXITO		96.67	93.33	100	96.67	100
% PROM.		97.33				

Tabla 5.3

VOCABULARIO: A		ORADOR: M				
		DIGITO ARTICULADO				
		1	2	3	4	5
DIGITO RECONOCIDO	1	24	1		2	
	2	3	27		17	
	3			30		15
	4	2	2		10	
	5	1			1	15
ERRORES		6	3	0	20	15
% EXITO		80	90	100	33.33	50
% PROM.		70.67				

Tabla 5.4

VOCABULARIO: B		ORADOR: F				
		DIGITO ARTICULADO				
		1	2	3	4	5
DIGITO RECONOCIDO	1	25	12		3	24
	2	5	16	1		
	3			27		
	4		2	2	27	
	5					6
ERRORES		5	14	3	3	24
% EXITO		83.33	53.33	90	90	20
% PROM.		67.33				

Tabla 5.5

VOCABULARIO: B		ORADOR: M				
		DIGITO ARTICULADO				
		1	2	3	4	5
DIGITO RECONOCIDO	1	30	2		1	
	2		27			
	3			30		
	4		1		29	
	5					30
ERRORES		0	3	0	1	0
% EXITO		100	90	100	96.67	100
% PROM.		97.33				

Tabla 5.6

VOCABULARIO: C		ORADOR: F				
		DIGITO ARTICULADO				
		1	2	3	4	5
DIGITO RECONOCIDO	1	30	4			20
	2		20			
	3		1	29		
	4		5	1	30	
	5					10
ERRORES		0	10	1	0	20
% EXITO		100	66.67	96.67	100	33.33
% PROM.		79.33				

Tabla 5.7

VOCABULARIO: C		ORADOR: M				
		DIGITO ARTICULADO				
		1	2	3	4	5
DIGITO RECONOCIDO	1	18	3		3	24
	2	3	24			
	3			30		
	4	9	3		27	
	5					6
ERRORES		12	6	0	3	24
% EXITO		60	60	100	90	20
% PROM.		70				

Tabla 5.8

De lo anterior puede observarse lo siguiente:

1) Los porcentajes de éxito más elevados (97.33%), los tienen los vocabularios cuyos oradores generadores, son los mismos que los evaluaron.

2) El porcentaje de éxito más bajo, le corresponde al vocabulario generado por el orador masculino y evaluado por el orador femenino (67.33%), seguido por el vocabulario generado por ambos oradores y probado por el orador masculino (70%). A continuación, con una diferencia muy reducida, tenemos al vocabulario generado por el orador femenino y probado por el orador masculino (70.67%).

3) El vocabulario intermedio es entonces el generado por ambos oradores y probado por el orador femenino con un porcentaje de éxito de 79.33%.

4) El mayor número de errores cometidos al pronunciar un dígito y reconocer otro y por lo tanto menor número de aciertos, corresponde al orador masculino con 93 errores, teniendo el orador femenino 84 errores.

5) La voz masculina responde mejor a los vocabularios generados por voz femenina que el caso inverso.

6) La voz femenina reconoce mejor que la voz masculina en los vocabularios generados por ambos oradores.

7) El mayor número de veces que un dígito pronunciado fue reconocido erróneamente, le corresponde al número 5 con 83 veces, ésto es por la cantidad de errores que hubo en los vocabularios cuyo orador generador no fue el orador evaluador.

8) El menor número de veces que se presentó el error pasado correspondió al número 3 con sólo cuatro veces.

9) El mayor número de veces que un dígito no pronunciado fue reconocido erróneamente, le corresponde al número 1 con 10 veces.

10) El menor número de veces que se cometió el error pasado, corresponde al número 5 con 2 veces.

11) Específicamente, en cuanto a los vocabularios generados en su totalidad por la misma persona que los evaluó, los resultados para los errores 7, 8, 9 y 10 respectivamente son: mayor= 2 con 5 veces, menor= 3 y 5 con cero veces; mayor= 1 con 5 veces y menor = 3 y 5 con 0 veces.

12) En cuanto al resto de los vocabularios, se observa lo siguiente para los errores 7, 8, 9 y 10 respectivamente: mayor= 5 con 83 veces, menor= 3 con 4 veces; mayor= 1 con 96 veces y menor= 5 con 2 veces.

13) El vocabulario que tuvo el mayor número de dígitos reconocidos al 100% fue el generado y probado por el orador masculino con tres dígitos.

14) El vocabulario que tuvo el menor número de dígitos reconocidos al 100% fue el generado por el orador masculino y probado por el femenino con cero dígitos.

V.3 CONCLUSIONES.

De lo anterior, es fácil visualizar que según los resultados indicados en los incisos 2 y 3, el porcentaje de éxito es muy reducido comparado con el del inciso 1 que se acerca más a los porcentajes obtenidos en experimentos encontrados en la literatura para este tipo de sistemas, por lo tanto, podemos comprobar que un sistema dependiente del locutor no puede aproximarse a uno independiente del locutor esperando resultados muy satisfactorios.

Para determinar el tipo de orador que será más adecuado para la generación de los patrones del S.R.V. (femenino o masculino), tomemos el argumento del inciso 4 en cuyo caso se escogería la voz femenina, tomando en cuenta los incisos 5 y 6 no se tendría una elección, sin embargo, considerando que estos argumentos corresponderían a un sistema independiente del locutor y dado que éste no lo es, tomaremos el inciso 13 que nos indica que con un orador masculino, existen más palabras en un vocabulario reconocidas con un porcentaje de 100% de éxito. Por lo tanto, el orador elegido para la generación de los patrones del S.R.V. es el masculino.

Se han escogido vocabularios de cinco palabras para ahorrar localidades de memoria en el TMS32010, ya que el programa de reconocimiento y comunicación TMS-PC para cinco palabras, ocupa casi los 4 K de palabras de memoria de programa con que cuenta el TMS32010.

En este programa se almacenan, tanto la palabra a reconocer (0.992 segundos correspondientes a 1984 localidades de memoria de programa para almacenar parámetros de la palabra), como los patrones del vocabulario de cinco articulaciones. Estos patrones utilizan 1280 localidades de memoria, por lo que una palabra más, necesita de 256 localidades más para almacenar sus patrones, por lo tanto, si se desea un vocabulario de diez palabras, se necesita de otras 1280 localidades de memoria de programa en el TMS32010 que ya no se tienen disponibles, debido a la extensión del programa que se menciona en el párrafo anterior.

En cuanto a la memoria de datos, se cuenta con dos páginas, una con 128 localidades y otra con 16, utilizandose 104 localidades de la primera

página para el programa de cinco palabras, con un vocabulario de 10 articulaciones sólo se necesitan 10 localidades más, lo cual no representa ningún problema.

Para un vocabulario de cinco palabras, el programa empleado utiliza una cantidad de instrucciones para el reconocimiento equivalente a 862 845 ciclos de reloj ó 0.17257 seg. (un c.i.= 200 nseg.), tiempo considerado como real. Para un vocabulario de diez palabras y dadas las características del programa, se necesitaría de aproximadamente 1 709 550 ciclos de reloj, es decir, 0.34191 seg.

Previo al reconocimiento, se realiza el cálculo de autocorrelaciones de la palabra a reconocer, el cual se lleva a cabo realizando operaciones entre muestras adyacentes, mismas que se toman cada 0.125 mseg. ó 625 ciclos de reloj; tiempo suficiente para ejecutar las instrucciones de estos cálculos que ocupan 566 ciclos de reloj ó 0.1132 mseg., es decir, sobran 11.8 μseg. ó 59 ciclos de reloj, por lo tanto, el cálculo de las autocorrelaciones de orden 16 para 128 muestras se realiza cada 0.125 mseg. x 128 = 16 mseg.

Después de cada bloque de 128 muestras, se ejecuta otro conjunto de instrucciones para nuevamente regresar a tomar muestras y se continúa así hasta terminar la palabra pronunciada.

Además de lo anterior, se tienen otras cuantas instrucciones para ejecutar diversas funciones de apoyo que ocupan un tiempo aproximado de 132 μseg.

CAPITULO VI

APLICACIONES Y PERSPECTIVAS.

El campo del Reconocimiento de Patrones de voz está adquiriendo una gran importancia en todos los terrenos de la vida moderna. Sus aplicaciones pueden ser tan variadas como la imaginación lo permita. A continuación daremos un breve panorama de lo que se puede hacer y lo que se está haciendo dentro de este campo.

VI.1 APLICACIONES.

Debido a que existe una amplia gama de sistemas para reconocimiento de voz, cada tipo de sistema es adecuado para ciertas aplicaciones específicas. Así tenemos que hay sistemas de palabras conectadas y sistemas de palabras aisladas; sistemas de vocabulario infinito y de vocabulario finito; de vocabulario amplio y de vocabulario reducido, etc. además existen sistemas ya sea de verificación o identificación del orador, cada uno con características específicas y por lo mismo, aplicaciones muy particulares.

Cualquiera que sea el tipo de sistema, éste será una herramienta de mucha utilidad para el objetivo que se tenga en mente, como por ejemplo:

1) Ahorro de tiempo en el desempeño de alguna actividad.

Podemos encontrar gran cantidad de trabajos en los que es necesario desempeñar varias actividades. Todas estas actividades van encaminadas a un solo objetivo, pero no se pueden realizar todas al mismo tiempo sino en forma secuencial. Sin embargo, en un sistema en donde se den órdenes o se controle alguna máquina mientras las manos y/o la vista realizan otra operación, habrá un considerable ahorro de tiempo. Como ejemplo tenemos:

A) Inspección y control de calidad.

En este trabajo, el operador tiene que inspeccionar los artículos y luego realizar un informe del estado en que se encuentra cada uno de ellos. Es decir, son dos actividades, sin embargo, existen sistemas que permiten al usuario revisar el artículo mientras le reporta verbalmente a alguna máquina, las condiciones en que se halló el artículo.

B) Programación automática de máquinas de control numérico.

Mientras el usuario prepara una máquina de este tipo con ayuda de manos y ojos, tiene acceso a la programación de la misma por medio de palabras en su idioma.

2) Verificación del orador.

Se puede decir que estos sistemas responden a la pregunta: ¿Soy yo?, esto es, dada una muestra de voz X, y la identidad que se pretende i, la máquina aceptará que X corresponde a i si la primera es lo suficientemente parecida a las muestras de voz pertenecientes a i que se encuentran guardadas en la memoria de la máquina. Dichos sistemas pueden aplicarse en:

A) Verificación de claves de seguridad y/o autorización para la realización de alguna actividad.

A1) Autorización para transacciones bancarias y/o de negocios que dará la máquina después de que el usuario se halla identificado.

A2) Autorización para el acceso a bancos de datos, etc.

A3) Acceso a areas restringidas a personal autorizado.

Este acceso se le permitirá al orador sólo en el caso de que el sistema lo identifique como una de las personas registradas para que se les permita el paso.

3) Identificación del orador.

Estos sistemas responden a la pregunta ¿Quién soy yo?. Es decir, dada una muestra de voz X, la máquina deberá encontrar al orador i de una población de n, cuyas muestras de voz almacenadas en la memoria de la máquina son lo suficientemente parecidas a X, para que X haya sido originada por i. Estos sistemas se pueden aplicar en:

A) Identificación de criminales.

Al igual que se comparan las huellas digitales contra un cierto banco de ellas para identificar a su propietario, podría hacerse lo mismo con patrones de voz.

4) Aplicaciones Militares.

Lamentablemente el desarrollo de la tecnología no únicamente se aprovecha para fines pacíficos. Siempre surgen ideas para usarla con fines bélicos y el reconocimiento de voz no podía ser la excepción.

A) Seguridad.

A1) Verificación e identificación del orador.

A2) Reconocimiento de códigos verbales, etc.

B) Comandos y control.

B1) Control de registros administrativos, etc.

Sin embargo, como podemos darnos cuenta, el proceso podría invertirse, es decir, tomar las ideas militares para aplicarlas en otros terrenos.

5) Transmisión de datos y Communicaciones.

A) Telefonía.

A1) Marcaje de números telefónicos por medio de voz con marcadores automáticos que funcionen con claves verbales.

A2) Transacciones bancarias o de negocios vía telefónica.

6) Sistemas de Información Automática.

A) Transportes.

Información de vuelos, salidas de trenes, autobuses, etc.

B) Reservaciones.

Reservaciones sea en hoteles o medios de transporte por parte de algún cliente con clave verbal especial.

7) Control de Tráfico aereo, naval, etc.

8) Administración.

En este campo se pueden utilizar muchas de las otras aplicaciones que hemos estado tratando, sea para negocios, seguridad, etc., interrelacionándolas de manera conveniente para adecuarlas a las necesidades administrativas que se tengan. Además podemos encontrar otras aplicaciones como:

A) Acceso por voz a programación.

Sea de bases de datos, paquetes de cómputo, etc.

B) Secretarias automáticas.

Es decir, máquinas que escriban automáticamente lo que se les vaya dictando, mientras el "jefe" realiza cualquier otra actividad.

9) Medicina.

A) Diagnósticos médicos por medio de análisis de voz.

B) Ayuda a minusválidos.

Posibilidad de manejo de algún tipo de aparato o máquina a personas impedidas. Esto es factible gracias a que no se necesita ni la visión ni las manos ni las piernas para accionar algún equipo, únicamente la voz, a su vez esto:

B1) Contribuye en gran medida a la integración de estas personas a la sociedad, ya que podrían desempeñar trabajos en los que con la voz controlaran algún mecanismo.

B2) Facilita la autonomía de estas personas por medio del uso de aparatos motrices auxiliares que puedan manejarse con la voz.

10) Educación.

A) Sistemas de aprendizaje de idiomas.

B) Sistemas verbales de aprendizaje general, etc.

El sistema que tenemos, es de palabras aisladas y de vocabulario limitado, pero éste último tiene la posibilidad de ampliarse tanto como nos lo permita la memoria de la computadora, pues en ella se almacenan los archivos de patrones que se van a ir intercambiando. Esto nos da la posibilidad de crear sistemas más sofisticados, ampliándose

considerablemente las posibilidades de aplicación de un sistema de este tipo.

Algunas de las ideas que se nos vienen a la mente son:

A) Sistema diagnosticador.

Algún tipo de sistema diagnosticador, con preguntas que tengan varias opciones de respuesta. Cada una de estas respuestas traerá consigo otra pregunta con sus respectivas opciones y así sucesivamente hasta que al final se llegue a un diagnóstico particular según el camino que se halla elegido para llegar a ese final. Esto a su vez, es aplicable a :

A1) Sistemas de enseñanza.

A2) Diagnósticos médicos urgentes.

A3) Juegos, etc.

B) Sistemas de información.

C) Claves secuenciales de seguridad.

D) Idiomas.

Un sistema de este tipo, pero con mejoras y más elementos, podría usarse también en el aprendizaje de idiomas.

E) Diccionario.

Diccionario de palabras en el lenguaje natural del usuario o de palabras extrangeras. Esto significa un ahorro de tiempo en la búsqueda, ya que mientras el usuario repite la palabra deseada, puede realizar otras actividades sin detenerse. El problema de cantidad se resuelve al haber escogido una serie de opciones, como en un árbol, con distintos rangos de letras, hasta haber llegado al rango en el que se encuentre la palabra deseada, de este modo, la P.C. enviará el archivo elegido que consta de pocas palabras y el usuario podrá nombrar la que requiera para que aparezca en pantalla el significado.

Nosotros hacemos una ejemplificación burda de ésto al agregar las definiciones de las palabras que se tienen en los diferentes menús.

Así podríamos seguir mencionando aplicaciones para los sistemas de reconocimiento de voz y como podemos darnos cuenta, son muchas y muy variadas, para todo tipo de sistemas, y todas ellas resuelven, de una u otra forma, alguna necesidad.

VI.2 PERSPECTIVAS.

El reconocimiento de voz presenta grandes perspectivas para el futuro, la prueba es que se han hallado muchos medios para desarrollarlo y muchas posibles aplicaciones que traerían grandes ventajas al mundo moderno.

Hay muchos tipos de sistemas y muchos algoritmos para lograr este fin y para todos ellos, hay grandes posibilidades de uso según sus características propias.

En cuanto a nuestro sistema particular de Reconocimiento de Comandos Hablados, aunque presenta muchas limitaciones, presenta también muchas posibilidades de aplicación, por este motivo en el futuro se pretende mejorarlo y adecuarlo a alguna necesidad específica.

En general, se desea obtener sistemas de reconocimiento de patrones, muy elaborados. Se piensa que lo ideal sería tener un sistema de reconocimiento de voz con características similares al del sistema humano, es decir, que fuera tan sofisticado, que prácticamente se pudiera "conversar" con él. En la actualidad esto se hace cada vez más factible y de hecho muchas universidades y grandes empresas, quienes cuentan con la tecnología necesaria, se ocupan del estudio y mejoramiento de los sistemas de reconocimiento de voz, obteniéndose grandes logros al respecto.

Para obtener un sistema de este tipo, se requeriría de un proceso de reconocimiento de voz unida. En este sistema, se daría una mayor importancia al análisis espectral de la señal de voz para encontrar los formantes de cada sonido y de esta forma, realizar un reconocimiento sobre fonemas. También se tendría que considerar la forma como éstos se modifican de acuerdo a los fonemas que se tengan alrededor, para que según un algoritmo adecuado, se puedan obtener palabras claras. Además se debe considerar la entonación y tono que se le da a la voz al ir hablando. Esto lo debe aplicar el sistema tanto a la hora de reconocimiento (cuando un individuo le hable), como a la hora en que la máquina pueda articular palabras para generar una contestación.

El sistema que presentamos aquí no es muy sofisticado, pero esperamos que de alguna forma sirva como base para el desarrollo, estudio y creación de otros mejores, que puedan usarse para objetivos más específicos.

BIBLIOGRAFIA

Capitulos.

[1] Kun-Shan Lin, Gene A. Frantz, Ray Simar, Jr. — I
"The TMS320 Family of Digital Signal Processors".
IEEE, Vol 75, No. 9, Septiembre de 1987.

[2] Amnon Aliphas, Joel A. Feldman. — I
"The Versatility of Digital Signal Processing Chips".
IEEE Spectrum, Junio de 1987.

[3] Harry C. Andrews. — II
"Introduction to Mathematical Techniques in Pattern Recognition".
Wiley - Interscience, 1972.

[4] D. Raj Reddy. — II
"Speech Recognition by Machine: A Review".
IEEE, Abril de 1976.

[5] George M. White. — II
"Speech Recognition: A Tutorial Overview".
Computer, Mayo de 1976.

[6] S. R. Hyde. — II
"Automatic Speech Recognition: A Critical Survey and Discussion of the Literature".
Human Communication: A Unified View.
E. E. David, Jr and P. B. Dones, eds. 1972.

[7] "Diccionario Enciclopédico Quillet". — II
Tomo V, página 387.
Ed. Argentina Aristides Quillet, S. A., Marzo de 1973.

[8] Agustin Mateos Muñoz. II
"Compendio de Etimologias Grecolatinas del Español".
Ed. Esfinge, S. A.; 16ª edición, Abril de 1980, pág 44.

[9] Rabinner L.R. and Shafer R.W. II, IV
"Digital Processing of Speech Signal".
Prentice Hall, Engewood Cliffs, New Jersey, 1978.

[10] Papamichelis, Panos E. IV
"Practical Approaches to Speech Coding".
Prentice Hall, Engewood Cliffs, New Jersey, 1987.

[11] Yoseph Linde, Andrés Buzo y Robert M. Gray. IV
"An Algorithm for Vector Quantizer Design".
IEEE Transactions on Communications, Vol.Com. 28, No.1,
Enero de 1980.

[12] A. Buzo, H. G. Martinez, C. Rivera. IV
"Discrete Utterance Recognition Based Upon Sorce Coding
Techniques".
Proceedings ICAS SP82, págs. 539-542.
IEEE 82 CH1746-7, Mayo de 1982.

[13] Victor Garcia Garduño, Miguel Moctezuma Flores, Sergio IV
Popocatl N.
Tesis: "Codificación de Voz en Tiempo Real a Baja Tasa
de Transmisión (4800 bits/seg.) Utilizando L. P. C."
Director: Dr. L. Andrés Buzo de la P., Octubre de 1986.

[14] Carlos Rivera Rivera. IV
Tesis: "Reconocimiento de Voz por Computadora".
Director: Dr. L. Andrés Buzo de la P., Abril de 1985.

[15] John E. Shore. IV
"Isolated-Word Speech Recognition Using Multi-Section Vector Quantization Code Books".
Naval Research Laboratory. Washington D. C., 1983.

[16] B. Beek, E. P. Neuberg y D. C. Hodge. VI
"An Assessment of the Technology of Automatic Speech Recognition for Military Applications".
IEEE Transactions on Acoustics, Speech and Signal Processing, Agosto de 1977.

[17] "TMS32010. User's Guide". Ap. A
Digital Signal Processors Products.
Texas Instruments.

[18] "TMS32010. Analog Interface Board. User's Guide". Ap. B
Digital Signal Processors Products.
Texas Instruments.

[19] "Microsystem Components Handbook, Vol. II". Ap. C
Intel, Septiembre de 1984.

A P E N D I C E A

TMS32010.

Evaluation Module EVM.

TMS32010 EVALUATION MODULE

- Target Connector for Full In-Circuit Emulation
- Debug Monitor Including Over 60 Commands with Full Prompting
- Reverse Assembler
- Transparency Mode for Host CPU Upload/Download
- Up to Eight Instruction Breakpoints
- Flexible Single Step with Software Trace
- Execution from EVM Program Memory or Target Memory
- Event Counter for One Breakpoint

The Evaluation Module (EVM) is a single board which enables a user to determine inexpensively if the TMS32010 meets the speed and timing requirements of the application. The EVM is a stand-alone module which contains all the tools necessary to evaluate the TMS32010 as well as to provide full in-circuit emulation via a target connector. A powerful firmware package contains a debug monitor, editor, assembler, reverse assembler, EPROM programmer, communication software to talk to two EIA ports, and an audio cassette interface. The resident assembler will convert incoming source text into executable code in just one pass by automatically resolving labels after the first assembly pass is completed. The EVM can be configured with a dumb terminal, power supplies, and either a host computer, or an audio cassette. Either source or object code can be downloaded into the EVM via the EIA ports provided on the board.

PART NUMBER	POWER SUPPLIES (TM990/518A)	UNITS
RTC/EVM 320A-03	OUTPUT A: +5 VDC (+ – 3%) B: +12 VDC (+ – 3%) C: –12 VDC (+ – 3%)	4.0 A 0.6 A 0.4 A

C

TABLE 2 - TMS32010 INSTRUCTION SET SUMMARY (CONTINUED)

BRANCH INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER
B	Branch unconditionally	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BANZ	Branch on auxiliary register not zero	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BGEZ	Branch if accumulator ≥ 0	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BGZ	Branch if accumulator > 0	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BIOZ	Branch on I/O = 0	2	2	1 1 1 1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BLEZ	Branch if accumulator ≤ 0	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BLZ	Branch if accumulator < 0	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BNZ	Branch if accumulator ≠ 0	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BV	Branch on overflow	2	2	1 1 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
BZ	Branch if accumulator = 0	2	2	1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 ◄— BRANCH ADDRESS —►
CALA	Call subroutine from accumulator	2	1	0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 0
CALL	Call subroutine immediate	2	2	0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 ◄— BRANCH ADDRESS —►
RET	Return from subroutine or interrupt routine	2	1	0 1 1 1 1 1 1 1 1 0 0 0 1 1 0 1

T REGISTER, P REGISTER AND MULTIPLY INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER
APAC	Add P register to accumulator	1	1	0 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1
LT	Load T register	1	1	0 1 1 0 1 0 0 1 ◄— D —►
LTA	LTA combines LT and APAC into one instruction	1	1	0 1 1 0 1 0 0 1 ◄— D —►
LTD	LTD combines LT, APAC and DMOV into one instruction	1	1	0 1 1 0 1 0 1 1 ◄— D —►
MPY	Multiply T register with stored product in P register	1	1	0 1 1 0 0 1 ◄— D —►
MPYK	Multiply T register with immediate operand, store product in P register	1	1	1 0 0 ◄— K —►
PAC	Load accumulator from P register	1	1	0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 0
SPAC	Subtract P register from accumulator	1	1	0 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0

I/O AND DATA MEMORY OPERATIONS

MNEMONIC	DESCRIPTION	NO CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER
DMOV	Copy contents of data memory location into next location	1	1	0 1 1 0 1 0 0 1 ◄— D —►
IN	Input data from port	2	1	0 1 0 0 0 ◄PA► 1 ◄— D —►
OUT	Output data to port	2	1	0 1 0 1 1 ◄PA► 1 ◄— D —►
TBLR	Table read from program memory to data RAM	3	1	0 1 1 0 1 1 1 1 ◄— D —►
TBLW	Table write from data RAM to program	3	1	0 1 1 1 1 0 1 1 ◄— D —►

TABLE 2 - TMS32010 INSTRUCTION SET SUMMARY

ACCUMULATOR INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO. CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER
ABS	Absolute value of accumulator	1	1	0 1 1 1 1 1 1 1 1 0 0 1 0 0 0
ADD	Add to accumulator with shift	1	1	0 0 0 0 ◄— S —► 1 ◄— D —►
ADDH	Add to high order accumulator bits	1	1	0 1 1 0 0 0 0 1 ◄— D —►
ADDS	Add to accumulator with no sign extension	1	1	0 1 1 0 0 0 0 1 1 ◄— D —►
AND	AND with accumulator	1	1	0 1 1 1 1 0 0 1 1 ◄— D —►
LAC	Load accumulator with shift	1	1	0 0 1 0 ◄— S —► 1 ◄— D —►
LACK	Load accumulator immediate	1	1	0 1 1 1 1 1 1 0 ◄— K —►
OR	OR with accumulator	1	1	0 1 1 1 1 0 1 0 1 ◄— D —►
SACH	Store high order accumulator bits with shift	1	1	0 1 0 1 1 ◄S► 1 ◄— D —►
SACL	Store low order accumulator bits	1	1	0 1 0 1 0 0 0 1 ◄— D —►
SUB	Subtract from accumulator with shift	1	1	0 0 0 1 ◄— S —► 1 ◄— D —►
SUBC	Conditional subtract (for divide)	1	1	0 1 1 0 0 1 0 0 1 ◄— D —►
SUBH	Subtract from high order accumulator bits	1	1	0 1 1 0 0 1 0 1 1 ◄— D —►
SUBS	Subtract from accumulator with no sign extension	1	1	0 1 1 0 0 1 0 1 1 ◄— D —►
XOR	Exclusive OR with accumulator	1	1	0 1 1 1 1 0 0 0 1 ◄— D —►
ZAC	Zero accumulator	1	1	0 1 1 1 1 1 1 1 0 0 1 1 0 0 1 0
ZALH	Zero accumulator and load high order bits	1	1	0 1 1 0 1 0 1 1 ◄— D —►
ZALS	Zero accumulator and load low order bits with no sign extension	1	1	0 1 1 0 1 1 0 1 ◄— D —►

CONTROL INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER
DINT	Disable interrupt	1	1	0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1
EINT	Enable interrupt	1	1	0 1 1 1 1 1 1 1 1 0 0 0 0 0 1 0
LST	Load status register	1	1	0 1 1 1 1 0 1 0 1 ◄— D —►
NOP	No operation	1	1	0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
POP	POP stack to accumulator	1	1	0 1 1 1 1 1 1 1 1 0 0 1 1 1 0 1
PUSH	PUSH stack from accumulator	1	1	0 1 1 1 1 1 1 1 1 0 0 1 1 1 0 0
RPVM	Reset overflow mode	1	1	0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 0
SOVM	Set overflow mode	1	1	0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1
SST	Store status register	1	1	0 1 1 1 1 1 0 0 1 ◄— D —►

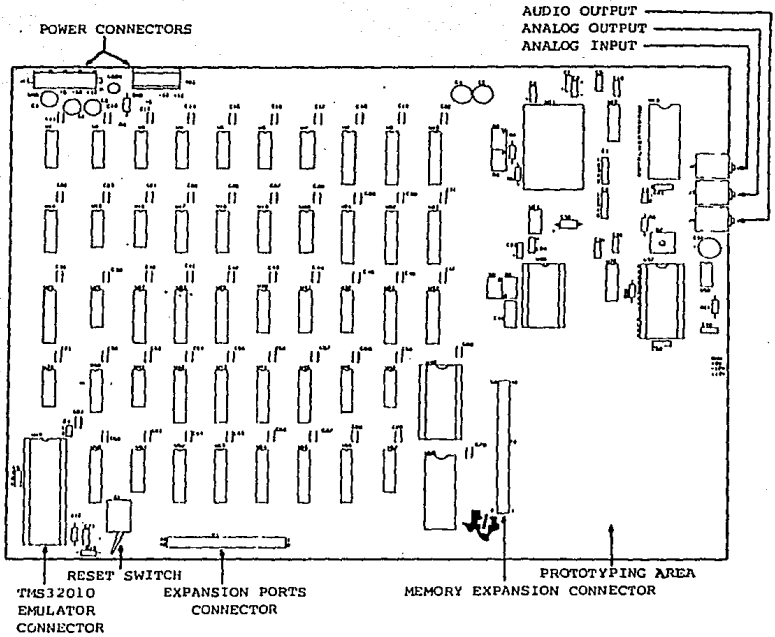
AUXILIARY REGISTER AND DATA PAGE POINTER INSTRUCTIONS

MNEMONIC	DESCRIPTION	NO. CYCLES	NO WORDS	OPCODE INSTRUCTION REGISTER
LAR	Load auxiliary register	1	1	0 0 1 1 1 0 0 1 ◄— D —►
LARK	Load auxiliary register immediate	1	1	0 1 1 1 0 0 0 0 ◄— R —► ◄— K —►
LARP	Load auxiliary register pointer immediate	1	1	0 1 1 0 1 0 0 0 1 1 0 0 0 0 0 0 ◄— K —►
LDP	Load data memory page pointer	1	1	0 1 1 0 1 1 1 1 ◄— D —►
LDPK	Load data memory page pointer immediate	1	1	0 1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 ◄— K —►
MAR	Modify auxiliary register and pointer	1	1	0 1 1 0 1 0 0 0 ◄— D —►
SAR	Store auxiliary register	1	1	0 0 1 1 0 0 0 1 ◄— D —►

A P E N D I C E B

Tarjeta de Interfase Analógica.

(Analog Interface Board. AIB)



TARJETA DE INTERFASE ANALOGICA (ANALOG INTERFACE BOARD, AIB)

Caracteristicas Generales.

a) Convertidor A/D de 12 bits con muestra y retén.
b) Convertidor D/A de 12 bits.
c) Puerto de salida de 16 bits para un convertidor D/A o una aplicación definida por el usuario.
d) Un puerto de entrada de 16 bits para un convertidor A/D o una aplicación definida por el usuario.
e) Dos filtros paso bajas.
f) Decodificador de TBLW (Table Write).
g) Memoria de expansión de datos de Entrada/Salida.
h) Area de prototipos para aplicaciones del usuario.

Especificaciones Generales.

Convertidor Analógico a Digital.

Resolución	: 12 bits.
Entrada Analógica	: -10 V a 10 V.
Salida Digital	: 16 bits en complemento a dos.
Tiempo de Conversión	: 25 microsegundos (máximo).

Muestreador Retenedor.

Tiempo de Adquisición a 0.1%	: 4 microsegundos (Vout = 10 V).
Velocidad de salida	: 0.3 V/seg. (25C).
Escalón de retención	: 10 mV (25C).

Reloj de Muestreo.

Rango	: 76.29 Hz. a 5 MHz.

Memoria Extendida.

Capacidad de la tarjeta	: 8192 x 16 bits.

Especificaciones del equipo empleado.

Fuente de Voltaje.

+5	@	1.2 A.
-12	@	0.25 A.
+12	@	0.25 A.

Emulador TMS32010 EVM, Xds, u otro emulador.

El AIB puede ser configurado para trabajar como una tarjeta auxiliar para sistemas con el TMS32010 XDS, EVM, u otros Emuladores, la configuración es la siguiente:

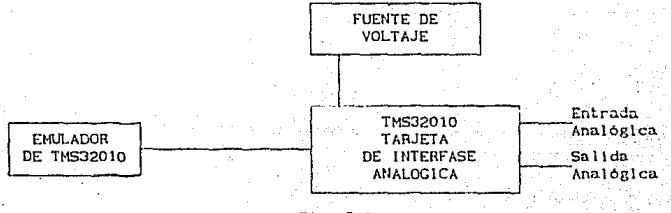


Fig. B.1

La tarjeta AIB, contiene interfases A/D y D/A al EVM del microprocesador TMS32010. La transferencia de datos entre ambas tarjetas se realiza con las instrucciones IN y OUT que direccionan uno de los 8 puertos de Entrada/Salida por medio de los tres bits menos significativos del bus de direcciones, en la tabla B1 se aprecia cada dirección y su función correspondiente.

Puerto	Función de Entrada	Función de Salida
0	Lectura del registro de status del convertidor A/D.	Carga el registro de control del AIB.
1	No usado	Carga el reloj de Muestreo.
2	Lectura de datos del convertidor A/D.	Escritura de datos en convertidor D/A.
3	Lectura del puerto de expansión.	Escritura en el puerto expansión.
4	Lectura de la memoria expandida (Direcciones).	Escritura de la memoria expandida (direcciones).
5	Lectura de la memoria expandida (datos).	Escritura de la memoria expandida (datos).
6	No usado	No usado
7	No usado	No usado

Tabla B.1

El registro de control se carga por el TMS32010 para definir el modo de operación del AIB (ver tabla B2). El patrón de bits de control deseado, se guarda en memoria de datos y se envía al puerto cero con una instrucción de OUT.

15 8	7	6	5	4	3	2	1	0
NO USADOS	DCW	DCR	U/D	CAD2	CAD1	CDA1	CDA2	CCLR

Bit 0 (CCLR) - 1 = Deshabilitación del reloj de muestreo.
0 = Habilitación del reloj de muestreo.

Bit 1 (CDA1) - 1 = Modo transparente para el D/A.
0 = Modo de retraso de muestreo para el D/A.

Bit 2 (CDA2) - 1 = Modo Transparente para expansión de D/A.
0 = Modo muestreo para expansión de D/A.

Bit 3 (CAD1) - 1 = Modo de recepción automática para A/D.
0 = Modo de recepción asincrona para A/D.

Bit 4 (CAD1) - 1 = Modo de recepción automática para expansión A/D.
0 = Modo de recepción asincrona para expansión A/D.

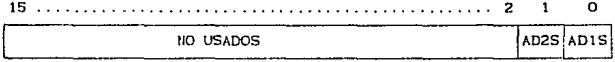
Bit 5 (U/D) - 1 = Contador ascedente para direcciones de memoria extendida.
0 = Contador descendente para direcciones de memoria extendida.

Bit 6 (DCR) - 1 = El contador de direcciones de la memoria extendida es deshabilitado para contar lectura de datos.
0 = El contador de direcciones de la memoria extendida contará las lecturas a la memoria de datos.

Bit 7 (DCW) - 1 = El contador de direcciones de la memoria extendida es deshabilitado para el conteo de la escritura de datos.
0 = El contador de direcciones de la memoria extendida contará las escrituras en la memoria de datos.

Tabla B.2

Se tienen convertidores A/D y D/A con puertos de expansión para convertidores A/D y D/A para convertidores adicionales. El registro de status se utiliza en aplicaciones donde el convertidor A/D de expansión se ocupa y se está trabajando con dos convertidores A/D al mismo tiempo.



Bit 0 (AD1S) - 1 = Convertidor A/D en conversión.
0 = El convertidor A/D tiene un dato listo.

Bit 1 (AD2S) - 1 = Convertidor A/D de expansión en conversión.
0 = El convertidor A/D de expansión tiene un dato listo.

Tabla B.3

La AIB cuenta también con sockets para 8 K de memoria de expansión. Esta memoria se direcciona a través del puerto de entrada/salida y soporta direccionamiento automático o manual. Pueden direccionarse hasta 64 K de memoria usando un conector para expansión de memoria.

El reloj para la tasa de muestreo en la AIB se deriva del CLKOUT del TMS32010 y puede programarse para proporcionar una salida y/o entrada analógica periódica. Consiste en un contador programable dividido entre una constante N (de 16 bits de longitud) que se envia, de una localidad de memoria, al puerto uno con una instrucción "OUT" para cargarse en el registro de muestreo automático del reloj.

La relación de salida para la frecuencia de muestreo es la siguiente:

$$F_{sr} = \frac{Fclkout}{N + 1} \qquad \text{ó} \qquad N = \frac{Fclkout}{F_{sr}} - 1$$

donde Fclkout = 5 Mhz.
N = Constante cargada en el registro de control
(de 144 a 65536 ó 76.29 Hz. 34.48 KHz.).

Existen dos filtros analógicos pasobajas en la AIB. Un filtro en la entrada A/D limita en banda la entrada para minimizar los efectos de "aliasing". El otro filtro suaviza la salida del convertidor D/A. La respuesta en frecuencia de los fitros se controla variando los componentes externos de los mismos. Esta frecuencia generalmente está fija en 4.7 Kiiz.

Para aplicaciones con la salida de audio, se proporciona un amplificador de audio que controla una bocina de 8 Ω.

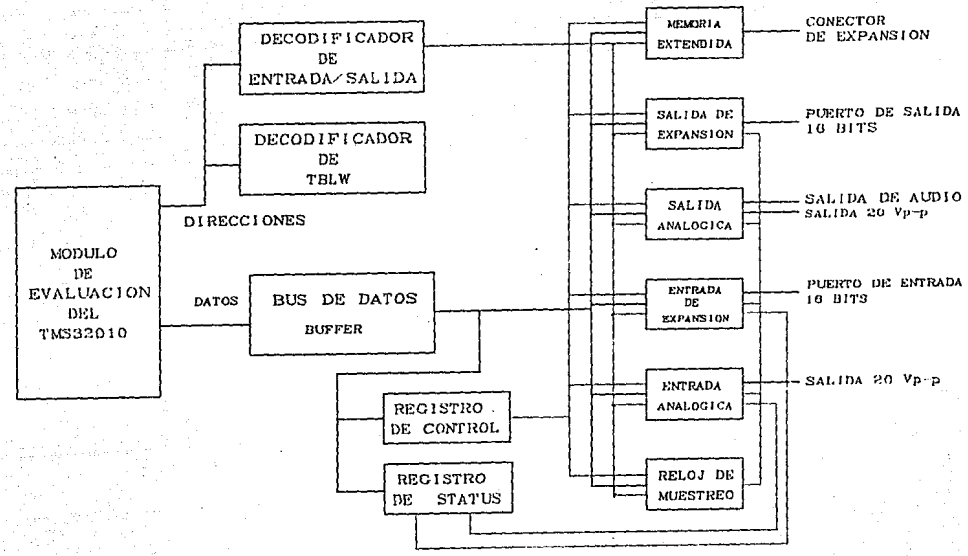


Diagrama de bloques de la AIB del TMS32010.

B-7

A P E N D I C E C

U. S. A. R. T.

Universal Synchronous/Asynchronous Receiver/Transmitter.

intel

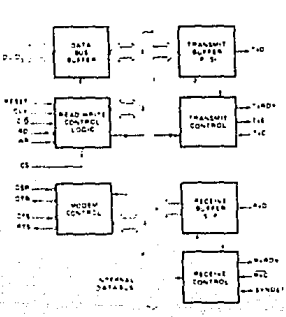
8251A/S2657
PROGRAMMABLE COMMUNICATION INTERFACE

- Synchronous and Asynchronous Operation
- Synchronous 5-8 Bit Characters; Internal or External Character Synchronization; Automatic Sync Insertion
- Asynchronous 5-8 Bit Characters; Clock Rate—1, 16 or 64 Times Baud Rate; Break Character Generation; 1, 1½, or 2 Stop Bits; False Start Bit Detection; Automatic Break Detect and Handling.
- Synchronous Baud Rate — DC to 64K Baud
- Asynchronous Baud Rate — DC to 19.2K Baud
- Full Duplex, Double Buffered, Transmitter and Receiver
- Error Detection — Parity, Overrun and Framing
- Fully Compatible with 8080/8085 CPU
- 28-Pin DIP Package
- All Inputs and Outputs are TTL Compatible
- Single +5V Supply
- Single TTL Clock

The Intel® 8251A is the enhanced version of the industry standard, Intel® 8251 Universal Synchronous/Asynchronous Receiver/Transmitter (USART), designed for data communications with Intel's new high performance family of microprocessors such as the 8085. The 8251A is used as a peripheral device and is programmed by the CPU to operate using virtually any serial data transmission technique presently in use including IBM "Bisync". The USART accepts data characters from the CPU in parallel format and then converts them into a continuous serial data stream for transmission. Simultaneously, it can receive serial data streams and convert them into parallel data characters for the CPU. The USART will signal the CPU whenever it can accept a new character for transmission or whenever it has received a character for the CPU. The CPU can read the complete status of the USART at any time. These include data transmission errors and control signals such as SYNDET, TxEMPTY. The chip is constructed using N-channel silicon gate technology.

PIN CONFIGURATION

BLOCK DIAGRAM



PIN NAMES

D0-D7	Data Bus (Bi-Directional)	TxE	Transmitter Empty
C/D	Control/Data Select	TxRDY	Transmitter Ready
RD	Read Control Input	SYNDET/BD	Sync Detect/Break Detect
WR	Write Data/Control Select		
CS	Chip Select		
CLK	Clock Pulse	RxD	Receive Data
RESET	Reset	TxC	Transmitter Clock
TxD	Transmit Data	RxC	Receiver Clock
DTR	Data Terminal Ready	DSR	Data Set Ready
RxD	Receive Data	CTS	Clear To Send
RxRDY	Receiver Ready	RTS	Request To Send
TxRDY	Transmitter Ready		

FEATURES AND ENHANCEMENTS

8251A is an advanced design of the industry standard USART, the Intel 8251. The 8251A operates with an extended range of Intel microprocessors that includes the new 8085 CPU and maintains compatibility with the 8251. Familiarization time is minimal because of compatibility and involves only knowing the additional features and enhancements, and reviewing the AC and DC specifications of the 8251A.

The 8251A incorporates all the key features of the 8251 and has the following additional features and enhancements:

- 8251A has double-buffered data paths with separate I/O registers for control, status, Data In, and Data Out, which considerably simplifies control programming and minimizes CPU overhead.
- In asynchronous operations, the Receiver detects and handles "break" automatically, relieving the CPU of this task.
- A refined Rx initialization prevents the Receiver from starting when in "break" state, preventing unwanted interrupts from a disconnected USART.
- At the conclusion of a transmission, TxD line will always return to the marking state unless SBRK is programmed.
- Tx Enable logic enhancement prevents a Tx Disable command from halting transmission until all data previously written has been transmitted. The logic also prevents the transmitter from turning off in the middle of a word.
- When External Sync Detect is programmed, Internal Sync Detect is disabled, and an External Sync Detect status is provided via a flip-flop which clears itself upon a status read.
- Possibility of false sync detect is minimized by ensuring that if double character sync is programmed, the characters be contiguously detected and also by clearing the Rx register to all ones whenever Enter Hunt command is issued in Sync mode.
- As long as the 8251A is not selected, the $\overline{RD}$ and $\overline{WR}$ do not affect the internal operation of the device.
- The 8251A Status can be read at any time but the status update will be inhibited during status read.
- The 8251A is free from extraneous glitches and has enhanced AC and DC characteristics, providing higher speed and better operating margins.
- Synchronous Baud rate from DC to 64K.
- Fully compatible with Intel's new industry standard, the MCS-85.

Other Timings:

SYMBOL	PARAMETER	MIN.	MAX.	UNIT	TEST CONDITIONS
t_{CY}	Clock Period	320	1350	ns	Notes 5, 6
t_D	Clock High Pulse Width	140	$t_{CY}-90$	ns	
t_S	Clock Low Pulse Width	90		ns	
t_R, t_F	Clock Rise and Fall Time		20	ns	
t_{DTx}	TxD Delay from Falling Edge of TxC		1	μs	
f_{Tx}	Transmitter Input Clock Frequency				
	1x Baud Rate	DC	64	kHz	
	16x Baud Rate	DC	310	kHz	
	64x Baud Rate	DC	615	kHz	
t_{TPW}	Transmitter Input Clock Pulse Width				
	1x Baud Rate	12		t_{CY}	
	16x and 64x Baud Rate	1		t_{CY}	
t_{TPD}	Transmitter Input Clock Pulse Delay				
	1x Baud Rate	15		t_{CY}	
	16x and 64x Baud Rate	3		t_{CY}	
f_{Rx}	Receiver Input Clock Frequency				
	1x Baud Rate	DC	64	kHz	
	16x Baud Rate	DC	310	kHz	
	64x Baud Rate	DC	615	kHz	
t_{RPW}	Receiver Input Clock Pulse Width				
	1x Baud Rate	12		t_{CY}	
	16x and 64x Baud Rate	1		t_{CY}	
t_{RPD}	Receiver Input Clock Pulse Delay				
	1x Baud Rate	15		t_{CY}	
	16x and 64x Baud Rate	3		t_{CY}	
t_{TxRDY}	TxRDY Pin Delay from Center of Last Bit		8	t_{CY}	Note 7
$t_{TxRDY\ CLEAR}$	TxRDY ↓ from Leading Edge of WR		6	t_{CY}	Note 7
t_{RxRDY}	RxRDY Pin Delay from Center of Last Bit		24	t_{CY}	Note 7
$t_{RxRDY\ CLEAR}$	RxRDY ↓ from Leading Edge of RD		6	t_{CY}	Note 7
t_{IS}	Internal SYNDET Delay from Rising Edge of RxC		24	t_{CY}	Note 7
t_{ES}	External SYNDET Set-Up Time Before Falling Edge of RxC	16		t_{CY}	Note 7
$t_{TxEMPTY}$	TxEMPTY Delay from Center of Last Bit	20		t_{CY}	Note 7
t_{WC}	Control Delay from Rising Edge of WRITE (T_xE_n, $\overline{DTR}$, $\overline{RTS}$)	8		t_{CY}	Note 7
t_{CR}	Control to READ Set-Up Time ($\overline{DSR}$, $\overline{CTS}$)	20		t_{CY}	Note 7

5. The TxC and RxC frequencies have the following limitations with respect to CLK
   For 1x Baud Rate, f_{Tx} or $f_{Rx} \leq \frac{1}{30} t_{CY}$
   For 16x and 64x Baud Rate, f_{Tx} or $f_{Rx} \leq \frac{1}{4.5} t_{CY}$

6. Reset Pulse Width = 6 t_{CY} minimum. System Clock must be running during Reset.

7. Status update can have a maximum delay of 28 clock periods from the event affecting the status.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias	0°C to 70°C
Storage Temperature	−65°C to +150°C
Voltage On Any Pin With Respect to Ground	−0.5V to +7V
Power Dissipation	1 Watt

*COMMENT: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. CHARACTERISTICS

$T_A = 0°C$ to $70°C$, $V_{CC} = 5.0V \pm 5\%$, GND = 0V

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.2	V_{CC}	V	
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2.2$ mA
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -400$ µA
I_{OFL}	Output Float Leakage		±10	µA	$V_{OUT} = V_{CC}$ TO 0.45V
I_{IL}	Input Leakage		±10	µA	$V_{IN} = V_{CC}$ TO 0.45V
I_{CC}	Power Supply Current		100	mA	All Outputs = High

CAPACITANCE

$T_A = 25°C$, $V_{CC} = GND = 0V$

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
C_{IN}	Input Capacitance		10	pF	fc = 1MHz
$C_{I/O}$	I/O Capacitance		20	pF	Unmeasured pins returned to GND

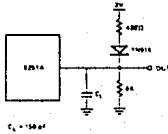


Figure 16. Test Load Circuit

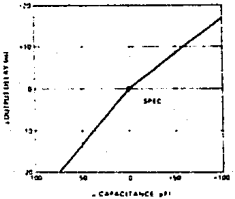


Figure 17. Typical Δ Output Delay vs. Δ Capacitance (pF)

A.C. CHARACTERISTICS

T_A = 0°C to 70°C, V_{CC} = 5.0V ±5%, GND = 0V

Bus Parameters (Note 1)

Read Cycle:

SYMBOL	PARAMETER	MIN.	MAX.	UNIT	TEST CONDITIONS
t_{AR}	Address Stable Before $\overline{READ}$ (CS, C/D)	50		ns	Note 2
t_{RA}	Address Hold Time for $\overline{READ}$ (CS, C/D)	50		ns	Note 2
t_{RR}	$\overline{READ}$ Pulse Width	250		ns	
t_{RD}	Data Delay from $\overline{READ}$		250	ns	3 C_L = 150 pF
t_{DF}	$\overline{READ}$ to Data Floating	10	100	ns	

Write Cycle:

SYMBOL	PARAMETER	MIN.	MAX.	UNIT	TEST CONDITIONS
t_{AW}	Address Stable Before $\overline{WRITE}$	50		ns	
t_{WA}	Address Hold Time for $\overline{WRITE}$	50		ns	
t_{WW}	$\overline{WRITE}$ Pulse Width	250		ns	
t_{DW}	Data Set Up Time for $\overline{WRITE}$	150		ns	
t_{WD}	Data Hold Time for $\overline{WRITE}$	50		ns	
t_{RV}	Recovery Time Between WRITES	6		t_{CY}	Note 4

NOTES 1. AC timings measured V_{OH} = 2.0, V_{OL} = 0.8, and with load circuit of Figure 1.
2. Chip Select (CS) and Command Data (C/D) are considered as Addresses.
3. Assumes that Address is valid before $\overline{RD}$.
4. This recovery time is for Mode Initialization only. Write Data is allowed only when TxRDY = 1.
Recovery Time between Writes for Asynchronous Mode is 8 t_{CY} and for Synchronous Mode is 16 t_{CY}.

Input Waveforms for AC Tests