

# **ESCUELA POLITECNICA NACIONAL**

**ESCUELA DE INGENIERIA**

**DISEÑO E IMPLEMENTACION DE PROTOTIPO DE AUDIMETRO  
DIGITAL DE TONOS PUROS DE CONDUCCION AEREA**

**PROYECTO PREVIO A LA OBTENCION DEL TITULO DE INGENIERO EN  
ELECTRONICA Y TELECOMUNICACIONES**

**VICTOR RAUL BARRAGAN BORJA**

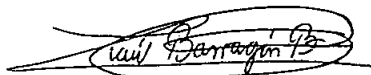
**DIRECTORA: INGENIERA TANIA PEREZ R.**

**Quito, octubre de 2005**

## DECLARACION

Yo, Víctor Raúl Barragán Borja, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

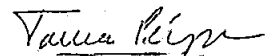
A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

A handwritten signature in black ink, appearing to read 'Víctor Raúl Barragán Borja', enclosed within a horizontal oval shape.

Víctor Raúl Barragán Borja

## CERTIFICACION

Certifico que el presente trabajo fue desarrollado por Víctor Raúl Barragán Borja, bajo mi supervisión.



Ing. Tania Pérez R.

DIRECTORA DEL PROYECTO

# *Agradecimiento*

*Primeramente agradezcô a Dios. Además y de manera muy sentida a todas las personas que a lo largo de mi carrera me han ayudado, especialmente a las que me han ofrecido su amor, comprensión, consejo y apoyo incondicional:*

*Mi Padre: Raúl Barragán C.  
Mi Hermanita: Anita Carolina  
Mi Abuelita: Carmelita Bustos  
Mi Novia : Cristina Miranda*

*Quiero agradecer también a aquellas personas sin la ayuda de las cuales hubiera sido imposible alcanzar mi objetivo. Por toda su ayuda y asesoramiento, agradezco a:*

*Ing. Tania Pérez  
Ing. Fernando Vásquez  
Dr. Fabián Alvarez  
Sr. Hernán Barba  
Doña Vilma Guerrero*

*No puedo dejar de agradecer a quienes me han brindado su amistad en todo momento:*

*Sr. Diego Chamorro  
Srta. Yadira Acurio  
Srta. Karen Medina  
Sr. David Maldonado*

*Finalmente agradezcô a :*

*Lic. Sofía Borja  
Lic. Miriam Ramos  
Ing. Francisco Torres*

*Gracias*

*Víctor Raúl*

# *Dedicatoria*

*Mamitá:*

*Quizás mi sueño más grande en la vida era el de compartir este momento con Usted. Me duele en el alma que no pueda acompañarme justo ahora que tanto la necesito, y daría mi vida entera con tal de que estuviera junto a mi.*

*Se que Usted también anhelaba con todas las fuerzas de su corazón acompañarme, para seguir ayudándome y dándome todo su amor; pero ya ve, nos separaron antes de tiempo.*

*Usted está presente en los recuerdos más hermosos de mi vida, cada día que pasa la recuerdo y la llevo en mi pensamiento mucho más.*

*Hace ya mucho tiempo me dio la vida, y desde entonces, empezamos un hermoso camino juntos, Usted me dio su apoyo, su guía, su comprensión, su amor. Siempre fue para mi el sostén en el que me apoyé en mis penas y alegrías.*

*Por todo lo que hizo por mi, en homenaje a todo lo que vivimos juntos, y porque la quiero mucho, le dedico este trabajo con todo mi amor.*

*La quiero y siempre la querré.*

*Raulito*

## RESUMEN

El presente trabajo tuvo como objetivo principal construir un audímetro digital de tonos puros de conducción aérea, tratando de reducir el costo de todos los elementos involucrados en la implementación del mismo.

Para esto fue necesario, en primera instancia, adquirir ciertos conocimientos fundamentales sobre la parte de la medicina correspondiente al estudio del oído. Además, dado que el dispositivo debió utilizar tecnología digital, era indispensable tratar ciertos tópicos básicos sobre dicha rama .

Por esta razón todo el capítulo I, se refirió al estudio de la audiometría y del audímetro con todos los conceptos que para entenderlos se requiere y además a los principales aspectos del procesamiento digital de señales, así como de circuitos integrados que permitieran aplicar dichas técnicas.

Una vez realizada la primera fase, se procedió a definir todos los requerimientos de diseño y a estudiar todos los elementos involucrados en la construcción, tanto para software como para hardware. Fue también necesario especificar el alcance y las limitaciones del audímetro que se construyó. Todos estos puntos fueron abordados en el capítulo II.

Ya con un conocimiento claro de las características del dispositivo que se quería construir y de los recursos con que para ello se contaba , se procedió al diseño del equipo en cuestión. Esta fue la parte medular del trabajo ya que, si bien es cierto que las dos primeras etapas eran fundamentales, es en la tercera en la que se pone en práctica toda la teoría estudiada y es justamente en el capítulo III en el cual constan de manera explícita y detallada todos los pasos necesarios para la construcción del audímetro digital, al cual se le bautizó con el nombre AMCAEC I. Se debe mencionar que todo el procesamiento digital de señales se realizó con el Ez – Kit Lite.

Luego de construido el audímetro se realizaron pruebas para verificar la validez y aplicabilidad del mismo. Este fue el motivo fundamental para que en el capítulo IV se realice un análisis de resultados tanto desde el punto de vista electrónico como médico y para complementar y dar mayor fuerza a los resultados obtenidos, se incluyó un análisis estadístico de los mismos.

El presente trabajo fue culminado exitosamente, debido a que los resultados finales obtenidos con el AMCAEC I. que son las audiometrías, dieron resultados similares a los obtenidos con un equipo comercial. Como punto importante, cabe recalcar que el equipo construido tuvo un costo relativamente bajo en comparación al de un equipo comercial.

# PRESENTACIÓN

Considerando todas las limitaciones que existen para desarrollar hardware en el país, se creyó que una alternativa para realizar investigación, era adentrarse en la elaboración de un software, el que acompañado de elementos básicos de hardware diera como resultado un equipo capaz de competir con otro de fabricación extranjera.

Es por esto que el presente trabajo demostró que aplicando los conocimientos adquiridos durante la carrera, es posible obtener un producto de alta tecnología a partir de elementos que no son de usuario final y que por ende son accesibles económicamente hablando.

Así por ejemplo, se consiguió mejorar la respuesta de frecuencia de audífonos de aplicación no específica mediante el uso de filtros digitales, obteniendo un comportamiento comparable a audífonos de alta precisión los cuales presentan un costo excesivamente alto.

Otro aspecto importante es el hecho que se desarrolló un software sumamente amigable y que proporciona varias funciones de valor agregado las mismas que facilitan su manejo al usuario final. Esto es de suma importancia ya que permite darse cuenta de todas las ventajas que pueden obtenerse con la explotación adecuada de tecnología común, tal como un computador personal.

Además se puso en claro las ventajas del uso de las técnicas digitales sobre las analógicas, las que si bien es cierto presentan un mayor grado de complejidad desde el punto de vista matemático, este inconveniente se ve recompensado con la exactitud y precisión en los resultados obtenidos.

Conjugando todos los aspectos anteriores, se pone en disposición del lector del presente trabajo el Audímetro de Conducción Aérea Económico AMCAEC I.



**CAPITULO I: CONCEPTOS GENERALES**

**1.1 PRINCIPIOS DE OTOLOGÍA**

1.1.1 Fisiología de la Audición	1
1.1.2 Tipos de Audición	2
1.1.2.1 Audición por Conducción Aérea	3
1.1.2.2 Audición por Conducción Osea	3
1.1.3 Hipoacusia	3
1.1.3.1 Hipoacusia de conducción	3
1.1.3.2 Hipoacusia Neurosensiva	4
<b>1.1.1 LA AUDIOMETRÍA</b>	4
1.2.1 Audiometría Tonal Umbral	4
1.2.2 El Audiograma	5
1.2.2.1 Nivel de referencia de la intensidad	8
1.2.2.2 Claves y Señalización	9
<b>1.3 EL AUDIMETRO</b>	9
<b>1.4 PROCESAMIENTO DIGITAL DE SEÑALES</b>	11
1.4.1 Muestreo	12
1.4.1.1 Teorema del Muestreo	13
1.4.1.2 Aliasing	13
1.4.2 Cuantificación y Codificación	15
1.4.3 Procesamiento Teórico	15
1.4.4 Procesamiento Práctico	17
1.4.4.1 Procesadores Digitales de Señales	17
1.4.4.1.1 Arquitecturas estándar en DSP	17
1.4.4.1.2 Diferencias entre Microcontroladores y DSP's	19
1.4.4.1.3 Ventajas de los DSP's	19
1.4.4.1.4 Desventajas de los DSP's	20
1.4.4.1.5 Aplicaciones de los DSP's	20
1.4.4.1.6 Clases de DSP's	21

## **CAPITULO II: REQUERIMIENTOS, ALCANCE Y LIMITACIONES DEL DISPOSITIVO AMCAEC I.**

<b>2.1</b>	<b>REQUERIMIENTOS DE DISEÑO</b>	23
2.1.1	Generación de Tonos	23
2.1.2	Intensidad de los Tonos Generados	23
2.1.3	Mejora de Respuesta de Frecuencia de Audífonos de aplicación no especificada	24
2.1.4	Sistemas de Medición de Intensidad	24
2.1.4.1	Decibelio dB	24
2.1.4.2	Decibelio Ponderado dBA	25
2.1.5	Trazo de Audiograma	27
<b>2.2</b>	<b>REQUERIMIENTO DE HARDWARE</b>	28
2.2.1	ADSP 2181	28
2.2.1.1	Descripción General	28
2.2.1.2	Sistema de Desarrollo	29
2.2.1.2.1	Assembler	29
2.2.1.2.2	Linker	30
2.2.1.2.3	Simulator	31
2.2.1.3	Arquitectura	31
2.2.1.3.1	Unidades Computacionales	31
2.2.1.3.2	Registro de Datos	33
2.2.1.3.3	Buses de Transferencia	34
2.2.1.3.4	Pórticos Seriales	34
2.2.1.3.5	Interrupciones	34
2.2.1.3.6	Operación en Baja Potencia	35
2.2.1.3.7	Asignación de Pines	36
2.2.2	CODEC 1847	37
2.2.2.1	Descripción General	37
2.2.2.2	Descripción Funcional de Audio	39
2.2.2.3	Reloj y Frecuencias de Muestreo	40
2.2.2.4	Interface Serial de Datos	40
2.2.2.5	Reset y Power Down	40
2.2.2.6	Definiciones de los Formatos de Datos	41

2.2.2.6.1	Formato de 16 bits con signo	41
2.2.2.6.2	Formato de 8 bits compandidos	42
2.2.2.6.3	Formato de 8 bits sin signo	43
2.2.2.7	Asignación de Pines	43
2.2.3	EZ-KIT LITE	44
<b>2.3</b>	<b>FILTROS DIGITALES</b>	48
2.3.1	Aspectos Generales	48
2.3.2	Ecuaciones de Diferencias	49
2.3.3	La Transformada Z	50
2.3.4	Función de Transferencia	50
2.3.4.1	Tipos de sistemas según la función de transferencia	52
2.3.5	Tipos de Filtros	53
2.3.6	Orden de un Filtro	54
2.3.7	Filtros Pasa Bajos	55
2.3.8	Filtros Pasa Banda	56
2.3.9	Filtros Pasa Altos	56
2.3.10	Filtros FIR	57
2.3.11	Filtros IIR	58
2.3.12	Técnicas de Diseño de Filtros	59
<b>2.4</b>	<b>ALCANCE DEL DISPOSITIVO AMCAEC I</b>	60
<b>2.5</b>	<b>LIMITACIONES DEL DISPOSITIVO AMCAEC I</b>	61

## **CAPITULO III: IMPLEMENTACION DE HARDWARE Y DE SOFTWARE**

<b>3.1</b>	<b>GENERACION DE TONOS</b>	63
3.1.1	Cálculo del Avance	63
3.1.2	Algoritmo de Generación de Tonos	67
3.1.3	Cálculo del Seno	69
<b>3.2</b>	<b>FILTRO DIGITAL</b>	72
3.2.1	Pasos a seguir para diseño	72
3.2.2.	Frecuencias a utilizar	73
3.2.2.1	Frecuencia de paso	73
3.2.2.2.	Frecuencia de corte	73
3.2.2.3	Frecuencia de muestreo	73

3.2.3	Atenuaciones a utilizar	73
3.2.3.1	Atenuación en la frecuencia de paso	73
3.2.3.2	Atenuación en la frecuencia de corte	74
3.2.4	Normalización de frecuencias	74
3.2.5	Diseño del Filtro Analógico	75
3.2.6	Digitalización del Filtro Analógico	78
3.2.6.1	La Transformada Bi Lineal	78
3.2.7	Diseño de Filtro IIR, CHEBYSHEV tipo I, Pasabajo y digitalización con la técnica de la transformada Bi Lineal	80
3.2.7.1	Determinación de las frecuencias a utilizar	80
3.2.7.2	Normalización de Frecuencias	81
3.2.7.3	Predistorsión o prealabeo de las frecuencias digitales	81
3.2.7.4	Cálculo de $\epsilon$ , $\lambda$ y N	82
3.2.7.5	Cálculo de $\alpha$ y $\beta_i$	83
3.2.7.6	Cálculo de Polos	83
3.2.7.7	Cálculo de la Función de Transferencia del Filtro Analógico H(s)	84
3.2.7.8	Digitalización del Filtro Analógico	85
<b>3.3</b>	<b>IMPLEMENTACION DEL FILTRO DIGITAL EN EL ADSP-2181</b>	<b>85</b>
<b>3.4</b>	<b>CONTROL DE VOLUMEN</b>	<b>88</b>
<b>3.5</b>	<b>CALIBRACIÓN DE VOLUMEN</b>	<b>90</b>
<b>3.6</b>	<b>SELECCIÓN DE CANAL PARA LA SEÑAL DE SALIDA</b>	<b>92</b>
<b>3.7</b>	<b>ANÁLISIS DEL SOFTWARE IMPLEMENTADO</b>	<b>93</b>
<b>3.8</b>	<b>ANÁLISIS DEL PROGRAMA IMPLEMENTADO EN EL ADSP-281</b>	<b>97</b>
<b>3.9</b>	<b>DESCRIPCION DEL AMCAEC I</b>	<b>101</b>
3.9.1	Hardware	101
3.9.1.1.	Ez – Kit Lite	101
3.9.2	Software	102
3.9.2.1	Elementos del Software	102
3.9.3	Procedimiento para realizar una evaluación audiométrica	107
3.9.4	Errores	110
3.9.5	Avisos	111

## **CAPITULO IV: PRUEBAS Y ANÁLISIS DE RESULTADOS**

<b>4.1</b>	<b>TIPOS DE ANÁLISIS</b>	113
4.1.1	Análisis Electrónico	114
4.1.1.1	Análisis de la Frecuencia de los Tonos Generados	114
4.1.1.1.1	Ondas con las Frecuencias Normalizadas	114
4.1.1.1.2	Frecuencias nominales y frecuencias generadas	118
4.1.1.2	Análisis de la Intensidad de los Tonos Generados	119
4.1.1.2.1	Ondas para evaluación del Oído Izquierdo	120
4.1.1.2.2	Ondas para evaluación del Oído Derecho	124
4.1.2.	Análisis Médico	126
4.1.2.1	Nivel de Hipoacusia	127
4.1.2.2	Tipo de Hipoacusia	128
4.1.2.3	Tabulación de Audiogramas	128
4.1.3	Análisis Estadístico	138
4.1.3.1	Distribución Normal	138
4.1.3.2	Distribución de t	140
4.1.3.3	Prueba de t	141
4.1.3.3.1	t sorteada	142
4.1.3.3.2	t pareada	143
4.1.3.4	Análisis de Datos	143
4.1.3.5	Análisis de Resultados	146
4.1.4	Prototipo Construido	147
4.1.5	Análisis de Costos	148

## **CAPITULO V: CONCLUSIONES**

<b>5.1</b>	<b>CONCLUSIONES</b>	149
<b>5.2</b>	<b>RECOMENDACIONES</b>	152

## **CAPITULO VI: BIBLIOGRAFIA**

<b>6.1</b>	<b>BIBLIOGRAFIA</b>	153
------------	---------------------	-----

## INDICE DE ANEXOS

- **Anexo 1**

Proceso de transducción de energía acústica a energía eléctrica.

Tomado de : **HARRISON. Principios de Medicina Interna. Volumen 1. 15<sup>ava</sup> Edición. Editorial Panamericana. España. 2001.**

- **Anexo 2**

Signos audiométricos y ejemplos de hipoacusias.

Tomado de : [www.interacustic.com](http://www.interacustic.com)

- **Anexo 3**

Perspectiva histórica y promesa a futuro del procesamiento digital de señales.

Tomado de: **OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000**

- **Anexo 4**

Demostración del Teorema de Muestreo.

Tomado de : <http://www.tecnun.com/asignaturas/tratamiento%20digital/TEMA8/>

- **Anexo 5**

Función de los pines del CODEC AD 1847.

Tomado de: **Manual de usuario del Ez- Kit Lite**

- **Anexo 6**

Subrutina de inicialización del CODEC AD 1847.

Tomado de : **Curso DSP, Ing. Fernando Vásquez**

- **Anexo 7**

Propiedades de linealidad, causalidad e invariabilidad en el tiempo de un sistema.

Tomado de: **OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000**

- **Anexo 8**

Transformada Z.

Tomado de : [www.elo.utfsm.cl/elo385/documentos/intro-dsp.pdf](http://www.elo.utfsm.cl/elo385/documentos/intro-dsp.pdf)

- **Anexo 9**

Respuesta de un sistema al impulso.

Tomado de: **OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000**

- **Anexo 10**

Datasheet del circuito integrado TDA2822M

- **Anexo 11**

Subrutina de Inicialización de comunicación serial entre CPU y DSP

Tomado de ; **Curso DSP, Ing. Fernando Vásquez**

- **Anexo 12**

Programa implementado en el ADSP-2181.

- **Anexo 13**

Evaluaciones audiométricas realizadas con equipo médico comercial.

- **Anexo 14**

Tabla de "t" de Student.

Tomado de ; **DANIEL Wayne, Biostatística, Tercera Edición, Editorial Limusa, México. 1974**

# CAPITULO I

## CONCEPTOS GENERALES

### 1.1 PRINCIPIOS DE OTOLOGÍA

La rama de la medicina encargada del estudio de las enfermedades es la patología, de la cual se derivan muchas otras ramas menores, dependiendo de la parte específica del cuerpo sobre la cual concentran su estudio. Así pues, según el diccionario enciclopédico VOX<sup>1</sup>, " otología es la parte de la patología que estudia las enfermedades del oído"

Para poder entender las patologías de un órgano, es necesario primero revisar aspectos de su fisiología, entendiéndose por fisiología la ciencia que estudia el funcionamiento de un órgano del cuerpo.

#### 1.1.1 FISIOLOGÍA DE LA AUDICIÓN

Los sonidos que se producen en la naturaleza viajan a través del aire en forma de ondas sonoras hasta llegar al oído humano. Estas ondas son receptadas por el pabellón de la oreja y viajan a través del conducto auditivo externo al final del cual está la membrana timpánica (la que está fuertemente templada y actúa como la lonja de un tambor). Esta membrana, es decir el tímpano, marca el límite entre el oído externo y el oído medio y además actúa como un mecanismo de control de impedancia que mejora la eficiencia de la transferencia de energía desde el aire hasta el oído interno, el cual está relleno de líquido. El oído medio se inicia con la trompa de Eustaquio la cual se comunica con la garganta y está a continuación del tímpano; la función de la trompa de Eustaquio es permitir la entrada de aire suficiente desde la boca para equilibrar la presión causada por las vibraciones del lado externo del tímpano. En la parte posterior del tímpano están ubicados tres pequeños huesos llamados estribo, yunque y martillo; estos están en contacto tanto con el oído medio como con el oído interno. Cuando el tímpano vibra, transmite esta vibración a estos tres huesos, los cuales a su vez transmiten esta vibración al líquido contenido en la cóclea , la cual forma parte del oído interno.

---

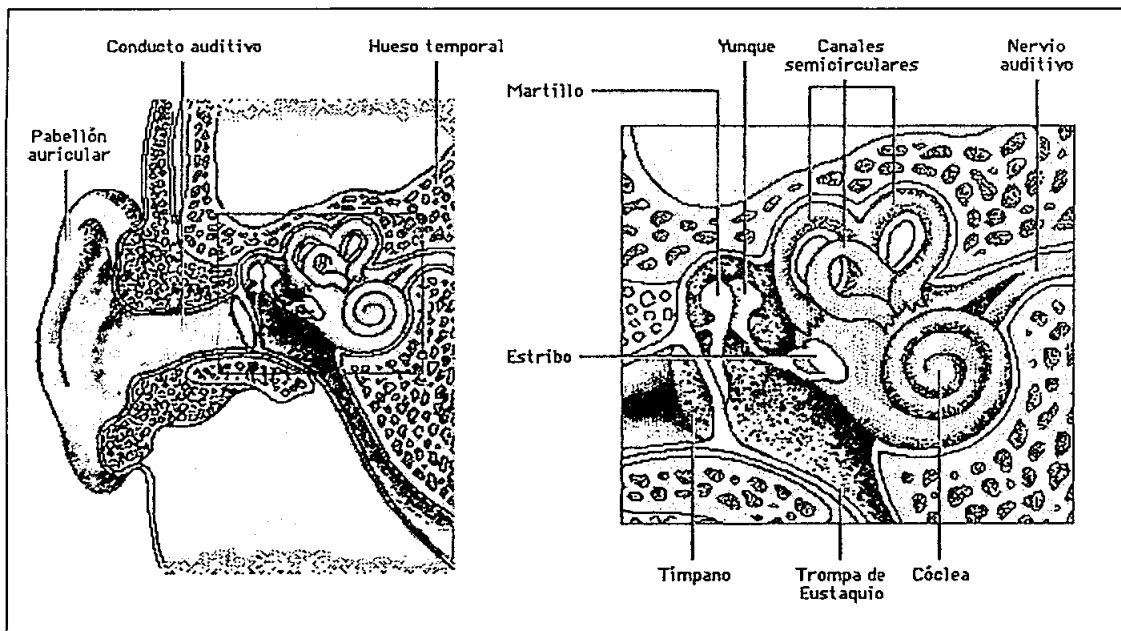
<sup>1</sup> DICCIONARIO ENCICLOPÉDICO VOX. Tomo IV. Editorial Bibliograf. España. Año 1970. p. 2574



La cóclea contiene una serie de células las cuales forman el órgano de Corti, cuya función es transmitir las vibraciones al nervio auditivo el cual envía la información al cerebro y este finalmente interpreta la información recibida (sonido).

Los conductos semicirculares también forman parte del oído interno y están llenos de líquido pero su función no es la de receptor sonidos sino de brindar la sensación de equilibrio conjuntamente con la vista.

La figura 1.1 muestra las partes de las cuales se habla en la descripción del proceso fisiológico de la audición:



*Figura 1.1: Partes del oído*

Una explicación detallada de cómo se realiza la transducción de la energía acústica a energía eléctrica consta en el anexo 1.

### 1.1.2 TIPOS DE AUDICIÓN

La conducción es la forma en la cual llegan las ondas sonoras al líquido del oído interno. Existen dos tipos de conducción: conducción ósea y conducción aérea.

**1.1.2.1 Audición por Conducción Aérea:** Según Harrison<sup>2</sup>, en este tipo de audición, "las ondas de sonido alcanzan el oído por su propagación en el aire", luego de lo cual se produce el proceso explicado en la parte referente a la fisiología de la audición.

Este tipo de audición es el más común

**1.1.2.2 Audición por Conducción Ósea:** Harrison<sup>2</sup> también explica este tipo de audición y menciona que "cuando el origen del sonido, que está en contacto con la cabeza, origina una vibración de las estructuras óseas del cráneo, incluido el hueso temporal, lo que genera una onda de movimiento en el líquido de la cóclea".

Esto quiere decir que los huesos del cráneo vibran y estas vibraciones se transmiten hasta llegar al líquido de la cóclea, luego de la cual se continúa con el proceso indicado en la fisiología de la audición.

### 1.1.3 HIPOACUSIA

Tal como lo indica su etimología, hipo igual bajo y acusia igual audición, hipoacusia significa pérdida de audición y es un trastorno del sentido del oído que impide al ser humano escuchar adecuadamente y tal como menciona el mismo Harrison<sup>2</sup>, "constituye uno de los trastornos sensitivos más frecuentes del ser humano y casi el 10 % de la población presenta algún tipo de hipoacusia".

Existen 2 tipos de hipoacusia: de conducción y neurosensitiva.

**1.1.3.1 Hipoacusia de conducción :** Se produce por la obstrucción del conducto auditivo externo debido a cerumen (cera), hinchazón del revestimiento del propio conducto, estrechamiento del conducto o por tumores del conducto.

Es decir, esta enfermedad se produce cuando aparece un obstáculo en el trayecto que tiene que atravesar el sonido para llegar al tímpano, pero todos los órganos del oído funcionan correctamente.

---

<sup>2</sup> HARRISON. "Principios de Medicina Interna". Volumen 1. 15ava Edición. Editorial Panamericana. España. 2001p. 217

**1.1.3.2 Hipoacusia Neurosensitiva** : Se produce cuando existe una lesión de las células del órgano de Corti debido a ruidos muy intensos, infecciones, medicamentos ototóxicos, fracturas del hueso temporal, meningitis o envejecimiento, entre otras.

Otra causa común de este tipo de hipoacusia, es la malformación congénita del oído interno y otra suele ser la predisposición genética.

Es importante también mencionar la presbiacusia, que es la pérdida de audición asociada a la edad y que representa la causa más frecuente de hipoacusia neurosensitiva en el adulto (los órganos se atrofian por el paso del tiempo).

Se puede decir que este tipo de enfermedad se produce por el malfuncionamiento de alguna parte del oído, no importa cuál sea la causa.

## **1.2 LA AUDIOMETRÍA**

La audiometría es un examen cuyo objetivo es determinar si el oído de la persona examinada padece una alteración auditiva, afectación uni o bilateral y tipo de hipoacusia.

### **1.2.1 AUDIOMETRÍA TONAL UMBRAL**

Este examen, también conocido como audiometría de tonos puros, evalúa la respuesta del oído a estímulos sonoros específicos, considerando las características de altura y sonoridad. La altura es la frecuencia del sonido y se mide en Hz, mientras que la sonoridad, en la cual intervienen elementos sensoriales, se refiere al volumen del sonido y se mide en dB.

La manera mediante la cual se evalúa al paciente es con la medición de los umbrales de los tonos puros; esto quiere decir que se debe determinar para qué nivel de sonoridad el paciente escucha cada uno de los tonos a los cuales se somete al oído. Esto se consigue con la colaboración del mismo, pues éste debe indicar el momento en cual escuchó cada tono.

Se puede decir que la audiometría de tonos puros permite evaluar la agudeza auditiva con respecto a los tonos puros.

Existen 2 maneras para hacer llegar el sonido al oído del paciente: la primera es encerrándolo en una cámara con aislamiento acústico y generando el sonido con parlantes y la segunda es poniéndole al paciente audífonos de precisión, con lo cual se evita la construcción de la cámara mencionada.

El examen se debe realizar para cada oído, ya que el funcionamiento de cada uno es independiente y los valores de frecuencia de los tonos puros con los que se realiza la evaluación son: 250, 500, 1000, 2000, 4000 y 8000 Hz<sup>3</sup> ya que estudios previos demuestran que éstas frecuencias son las más importantes para la comprensión del lenguaje.

Una vez determinado esto, se procede a establecer una gráfica, y según las características de la misma se puede determinar el tipo de patología que sufre el paciente examinado. Esta gráfica se conoce como audiograma.

Todo el proceso mencionado se denomina audiometría tonal umbral.

Si los resultados de este examen indican algún tipo de hipoacusia, se debe proceder a examinar la vía ósea, para lo cual se utilizan otro tipo de mecanismos más complejos, tales como el generar vibraciones mecánicas con un oscilador a las frecuencias indicadas anteriormente y ponerlas en cada parietal del paciente y realizar la misma evaluación anterior. Esto con el objetivo de brindar al médico tratante un panorama más amplio sobre el estado auditivo del paciente. Sin embargo, cabe mencionar que la evaluación del umbral óseo es mucho más delicada y además su interpretación más compleja ya que alcanza directamente la cóclea y por lo tanto permite determinar la pérdida del oído interno, mientras que la conducción aérea indica problemas en el oído externo y medio.

### 1.2.2 EL AUDIOGRAMA

El audiograma es un gráfico en el cual consta la información que se obtiene como resultado de la evaluación realizada al paciente.

El audiograma es una gráfica de intensidad en decibelios en la que aparece reflejado el umbral frente a la frecuencia.

---

<sup>3</sup> Valores que constan en la plantilla estandarizada por la ISO o por la ASA en la cual se trazan los audiogramas

Este gráfico es un sistema de ejes cartesianos, en el cual las distintas frecuencias se ponen en el eje de las abscisas y la intensidad en el eje de las ordenadas.

En este gráfico consta la intensidad a la cual el paciente escuchó cada uno de los tonos puros. Una vez determinado esto (punto de corte entre intensidad y valor de frecuencia para cada tono) , se debe trazar una curva que una cada uno de estos puntos. Según la forma de esta curva se determina el tipo de patología.

Así pues, la sordera se expresa como el número de exceso de dB por arriba del punto 0, al cual debe aumentarse la intensidad del sonido para que este sea apenas detectado por el oído alterado.

La siguiente es una guía general para determinar la gravedad de la sordera:

0 – 20 dB normal

20 – 40 dB sordera leve

40 – 60 dB sordera moderada

60 – 75 dB sordera grave

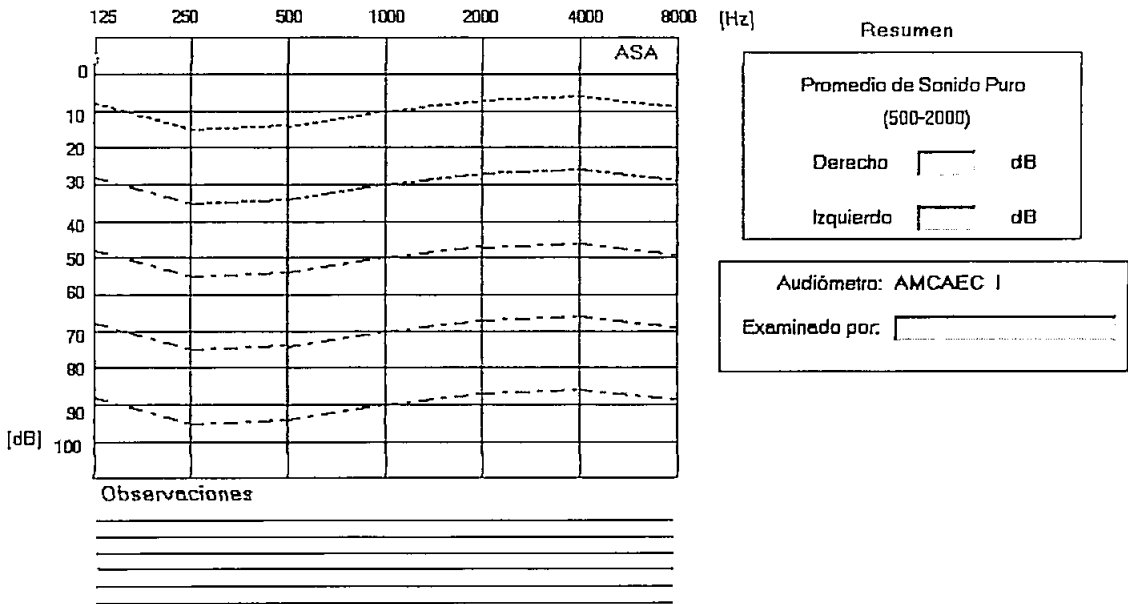
75 – 100 dB sordera profunda (completa)

Las características de esta gráfica están normalizadas por la ISO o por la ASA y deben ser siempre las mismas. La diferencia entre éstas dos se analiza detalladamente en el capítulo II.

La figura 1.2 muestra el formato típico de una gráfica para audiometría:

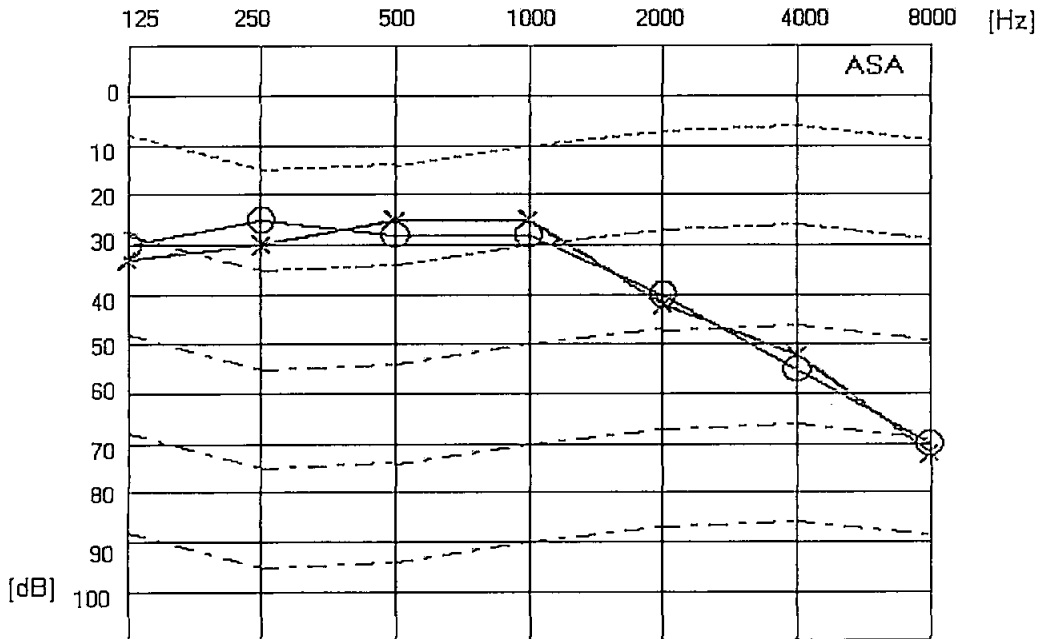
## EVALUACION AUDIOLOGICA

Nombre :  Edad :  Fecha:

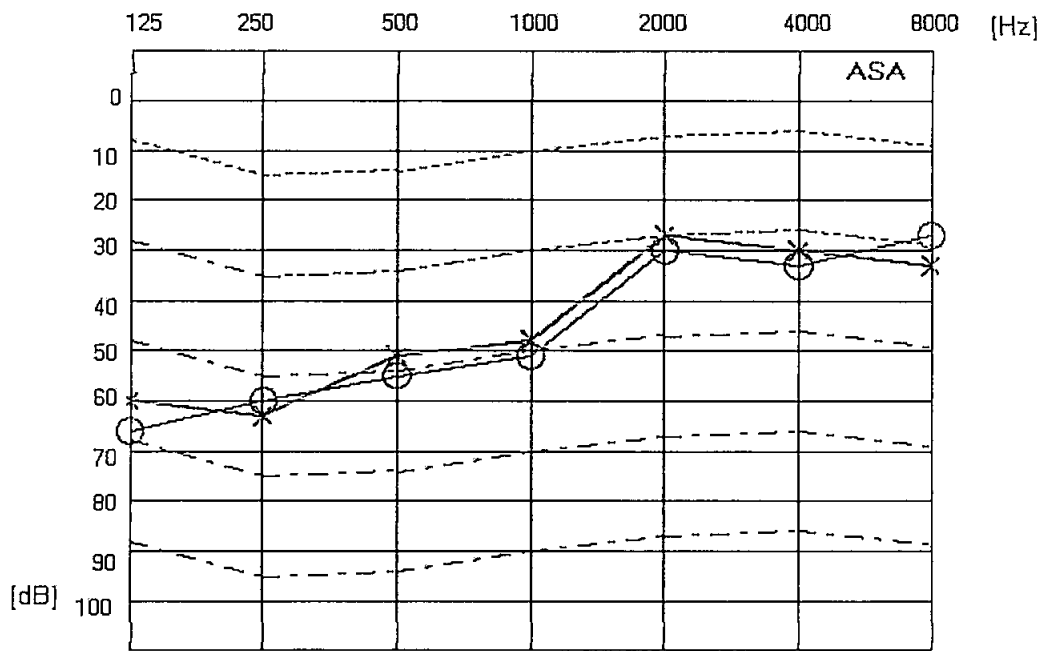


**Figura 1.2: Plantilla para realizar audiometría**

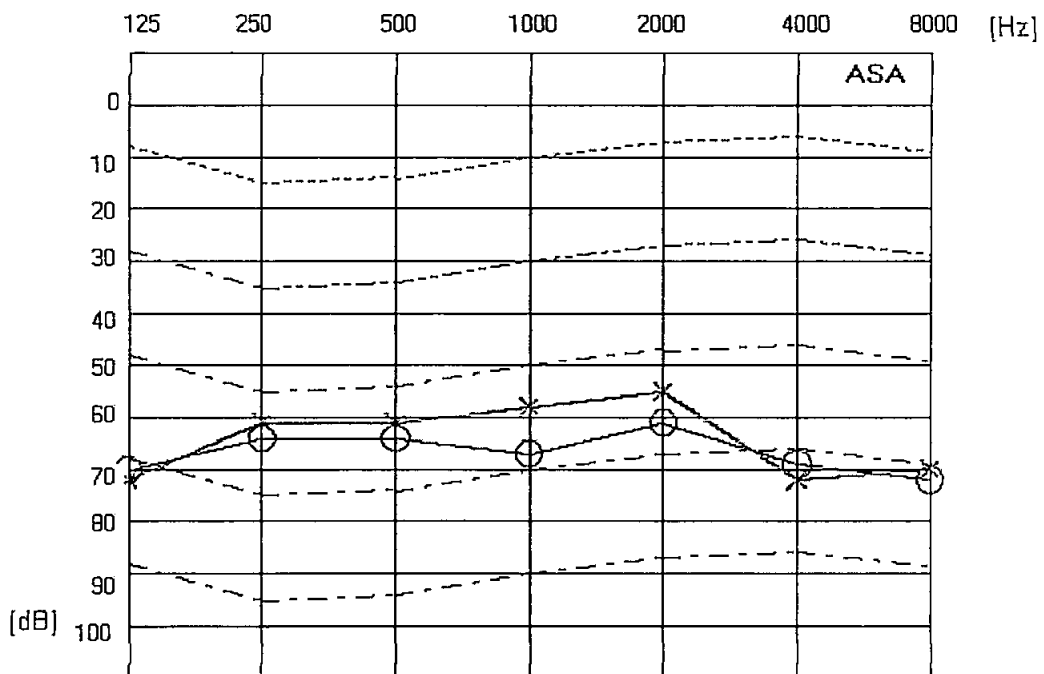
Como ya se dijo anteriormente, con la forma de la curva obtenida se puede determinar el tipo de patología; las figuras 1.3, 1.4 y 1.5 muestran ejemplos de esto:



**Figura 1.3: Sordera Neurosensorial**



**Figura 1.4: Sordera Conductiva**



**Figura 1.5: Sordera Mixta**

1.2.2.1 Nivel de referencia de la intensidad (0 dB) :especifica el umbral para esta frecuencia según un grupo extenso de personas con audición normal y además

requiere una energía acústica distinta para cada frecuencia; es decir que la potencia de nivel 0 es diferente para cada tono.

**1.2.2.2 Claves y Señalización:** En el audiograma, cada punto de corte debe ser representado por un pequeño círculo para el oído derecho y una pequeña cruz para el izquierdo. Así pueden inscribirse ambos oídos en un mismo gráfico, el derecho en rojo y el izquierdo en azul. Además, la curva de conducción aérea se hace con trazo continuo mientras que la de la ósea con un trazo entrecortado. Estas son las principales claves y señales.

En el anexo 2 constan el resto de claves y señales y además algunos ejemplos de interpretaciones de formas de curvas con claves y señalización que diagnostican varios tipos de hipoacusia.

### 1.3 EL AUDÍMETRO

Para realizar la audiometría, es necesario disponer de algún instrumento que sea capaz de generar tonos que sean lo más puros posibles.

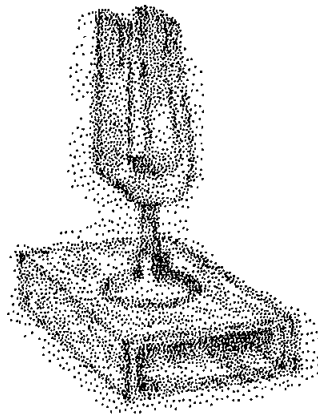
En un principio, para conseguir esto se usaba el diapasón. El diapasón es un instrumento musical constituido por una varilla metálica en forma de U generalmente apoyada sobre una caja de resonancia y cuyas vibraciones al ser golpeado por una varilla metálica dan como resultado una nota musical fija.

El diapasón, sin embargo, presenta muchos errores en la generación de tonos ya que los resultados obtenidos dependen totalmente de la habilidad de la persona que lo usa para generar los tonos y además, por sus mismas condiciones físicas, no presenta uniformidad al momento de la generación misma y la frecuencia más alta que éste puede generar es de 512 Hz.

Otro inconveniente que este presenta, es que a pesar de poder tener un dato bastante aproximado con respecto a la frecuencia del tono generado, aunque no exacto, no se puede determinar de manera exacta la intensidad del mismo.

La figura 1.6 muestra la forma de un diapasón:





*Figura 1. 6 : Diapasón*

Para solucionar todos los inconvenientes mencionados anteriormente, se puede usar el audímetro, el cual básicamente es un aparato electrónico de alta tecnología que genera sonidos de distintas frecuencias. La misión principal de este instrumento es generar tonos puros cuya intensidad y frecuencia pueden ser controlados con exactitud según los requerimientos del usuario.

El sonido generado por el audímetro se transmite al paciente con audífonos, los cuales deben ser de muy alta calidad para que el sonido no llegue distorsionado a los oídos de la persona examinada .

La diferencia básica de este con el diapasón, es que se puede controlar la intensidad de los tonos generados.

Las frecuencias de los tonos que debe generar todo audímetro son aquellas especificadas en los requerimientos para realizar una audiometría (250, 500, 1000, 2000, 4000 y 8000 Hz) . Se nota que cada frecuencia es el doble de la anterior; esto implica que la diferencia de los valores de frecuencia de los tonos generados corresponden a una octava de incremento.

En cuanto a la intensidad de los tonos, ésta debe poder ser controlada de la manera más exacta, lo cual depende de la calidad del dispositivo. La intensidad se controla con un atenuador y generalmente va de 0 a 110 dB.

Dependiendo del costo, un audímetro puede consistir de lo siguiente:

- Generador de frecuencias. (indispensable)
- Atenuador de intensidad en dB (indispensable)
- Generador de ruidos enmascarantes (para analizar umbral óseo)

- Vibrador óseo para estudio de audición ósea (para analizar umbral óseo)
- Micrófono para comunicarse con el paciente (para evaluar la comprensión de lenguaje).

Cada uno de estos componentes brindan mayores parámetros para determinar el tipo de anomalía. Sin embargo, se debe mencionar que los más importantes son los dos primeros por las razones expuestas en el estudio de la audiometría. Es necesario explicar que el ruido enmascarante es aquel que se usa para anular el sonido que pueda llegar al oído que no está siendo evaluado cuando se está realizando una examinación ósea.

En cuanto a la comprensión de lenguaje, es un tipo de prueba muy subjetiva que no permite determinar la agudeza del sentido de la audición sino la capacidad del paciente para interpretar y discriminar las palabras escuchadas; esta prueba no es muy usada actualmente.

La figura 1.7 muestra un modelo comercial de audímetro:



**Figura 1.7: Audímetro marca Earscan**

## 1.4 PROCESAMIENTO DIGITAL DE SEÑALES

El procesamiento digital de señales consiste básicamente en convertir una señal analógica en una digital con el objeto de poder procesarla más fácil y eficientemente.

Es importante mencionar que una señal se define como una cantidad física que varía en el tiempo y que lleva información, generalmente acerca del estado o comportamiento de un sistema.

Este procesamiento se realiza basándose en el principio evidente de que manejar o procesar a una señal digital es mucho mejor que a una analógica y además en que la calidad y confiabilidad de una señal digital es mucho mayor que la de una analógica.

Sobre los usos del procesamiento digital de señales, Oppenheim<sup>4</sup> dice: "...una buena parte del tratamiento de señales original involucra el proceso de una señal para obtener otra. Otra clase importante de problemas del tratamiento de señales es el de interpretación de señales. En estos problemas el objetivo del proceso no es obtener una señal de salida sino obtener una caracterización de la señal de entrada.....y otra categoría relativamente nueva de técnicas de tratamiento de señales es en la que se realiza la manipulación simbólica de expresiones..."

El anexo 3 brinda mayor información sobre las aplicaciones, la perspectiva histórica y la promesa a futuro del procesamiento digital de señales.

Se puede decir que procesar una señal es transformarla para operarla.

Los principales procesos que involucran el procesamiento digital de señales son los siguientes:

- Muestreo
- Cuantificación
- Codificación

---

<sup>4</sup> OPPENHEIM Alan, SCHAFER Ronald. "Tratamiento de Señales en Tiempo Discreto" Segunda Edición. Editorial Prentice Hall, Iberia. 2000 p. 2

### 1.4.1 MUESTREO

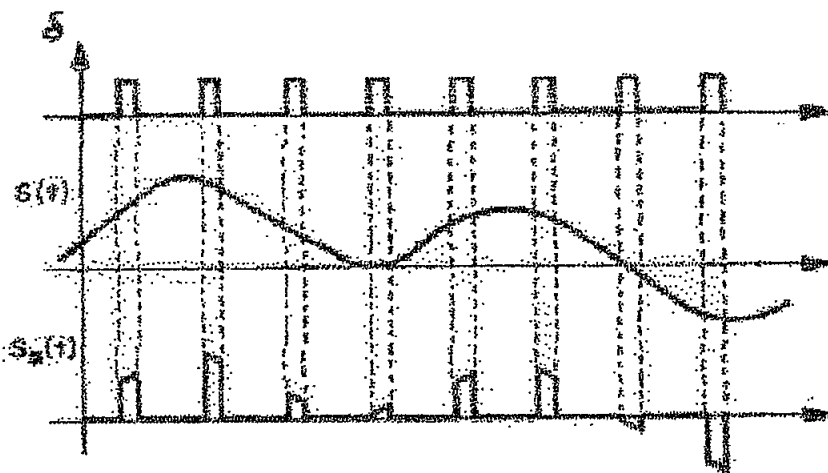
El muestreo es el primer procesamiento que se efectúa sobre las señales con el propósito de digitalizarlas, y consiste en tomar valores de una señal continua (señal analógica) en intervalos sucesivos de tiempo con una señal periódica (señal muestreadora), para de esta manera construir una secuencia discreta (señal muestreada)

La figura 1.8, muestra las tres señales principales involucradas en el muestreo, donde:

$\delta$  = señal muestreadora

$s(t)$  = señal analógica

$s_g(t)$  = señal muestreada



*Figura 1.8: Señales involucradas en el proceso de muestreo*

El principal problema que se presenta en el muestreo es la pérdida de información. Esto quiere decir que la señal muestreadora debe ser lo más lenta posible para que no ocupe mucho ancho de banda, pero a su vez lo suficientemente rápida para que no se pierda información de la señal muestreada y de esta manera poder reconstruir la señal continua a partir de las muestras. El tiempo que transcurre entre la toma de una muestra y otra se denomina período de muestreo y su inverso es la frecuencia de muestreo.

Es claro que la señal muestreadora debe cumplir ciertos requisitos para que no se pierda información. Estas condiciones vienen determinadas por el teorema del muestreo o de Nyquist

#### 1.4.1.1 Teorema del Muestreo

El enunciado del teorema del muestreo es el siguiente:

“Una señal de banda limitada de valor real sin componentes espectrales por encima de  $B$  Hz, se determina en forma unívoca por sus valores equidistantes en intervalos no mayores que  $1/(2B)$  segundos”

Esto quiere decir que la frecuencia de la señal muestreadora debe ser al menos el doble de la máxima componente de frecuencia de la señal muestreada.

Si la señal muestreadora cumple esta condición, la señal analógica puede ser reconstruida totalmente a partir de un conjunto de muestras discretas equidistantes en el tiempo.

La validez de este teorema se puede demostrar por medio de la propiedad de convolución de frecuencia de la transformada de Fourier.

Esta demostración consta de manera explícita en el anexo 4.

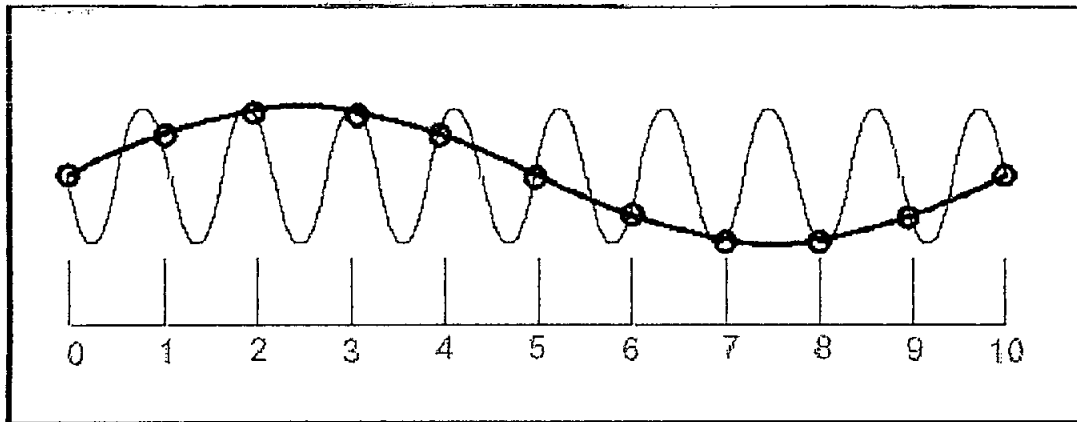
#### 1.4.1.2 Aliasing

Es un fenómeno de confusión de frecuencias que se produce cuando se muestrea una señal rompiendo las condiciones del teorema de Nyquist. Al intentar reconstruir la señal original podemos obtener frecuencias que no contenía la señal original.

Esta situación se puede presentar aún cuando supuestamente la señal a muestrear parece cumplir con la condición de Nyquist debido al ruido de alta frecuencia presente en el ambiente. Por ello, si se desea evitar la aparición de aliasing de frecuencia es importante el uso de filtros antialiasing que no dejen pasar frecuencias más allá de la permitida por la condición de Nyquist.

En la figura 1.9 se ilustra el muestreo de una señal de frecuencia  $f_1 = 0.1\text{hz}$  y otra de frecuencia  $f_2 = 0.9\text{hz}$ . La frecuencia de muestreo en ambos casos es  $f_s = 1\text{hz}$ . Como puede verse de la figura, ambas señales se confunden a esta

frecuencia de muestreo, es decir,  $f_1$  es un alias de  $f_2$  a esta frecuencia de muestreo.



*Figura 1.9: Ejemplo de Aliasing*

En el momento de la reconstrucción de la señal muestreada, aparecerán frecuencias que no estaban presentes en la señal original.

#### 1.4.2 CUANTIFICACION Y CODIFICACION

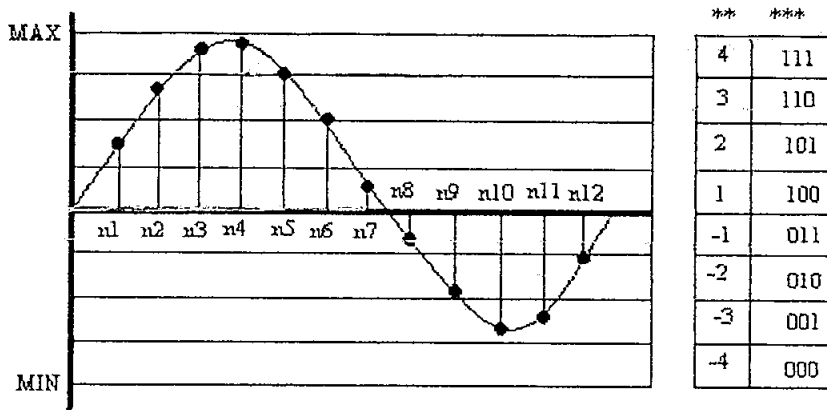
Es necesario que cada muestra sea representada por un código binario o palabra código, para lo cual es necesario la existencia de niveles de cuantificación. El número de niveles de cuantificación depende del número de bits con los que se represente una palabra código.

De esta manera, a cada muestra se le asigna un nivel de cuantificación (proceso denominado cuantificación) y cada nivel de cuantificación es representado por un código binario (proceso llamado codificación), correspondiendo finalmente un código binario para cada muestra.

El hecho de que los niveles de cuantificación sean discretos, implica que necesariamente existirá un error de cuantificación, el cual es la diferencia entre el valor real de la muestra y el valor que se le asigna a esa muestra según el nivel correspondiente. A este error se lo denomina ruido de cuantificación.

Entre mayor sea el número de bits que se use para representar cada muestra, menor será el error de cuantificación, pero en cambio esto implicaría manejar un mayor ancho de banda, lo cual en la mayoría de casos no es deseable.

La figura 1.10 muestra la cuantificación y codificación de una señal usando para ello 3 bits. Cada muestra (de n1 a n12) cae dentro de un nivel de cuantificación, para cada uno de los cuales se asigna un código binario.



\*\* Nivel de Cuantificación

\*\*\* Código

**Figura 1.10: Cuantificación y Codificación**

La Tabla 1 muestra el nivel y el código correspondiente a cada muestra:

<b>Muestra</b>	<b>Nivel de Cuantificación</b>	<b>Código</b>
n1	2	101
n2	3	110
n3	4	111
n4	4	111
n5	4	111
n6	3	110
n7	1	100
n8	-1	011
n9	-2	010
n10	-3	001
n11	-3	001
n12	-2	010

**Tabla 1.1: Asignación de Niveles de Cuantificación y de Códigos**

Otra forma de reducir el error de cuantificación consiste en trabajar con niveles no uniformes de cuantificación, sin embargo este método resulta bastante complejo, especialmente en la recuperación de la información.

El procesamiento digital de señales puede dividirse en dos grandes campos:

- El teórico
- El práctico

#### 1.4.3 PROCESAMIENTO TEÓRICO

Este campo maneja una matemática muy compleja y proporciona las herramientas para poder entender los procesos a los que se someten las señales analógicas al momento de digitalizarlas y además indica los algoritmos a seguir para dicha transformación.

Entre algunas de las herramientas matemáticas más comúnmente usadas podemos citar las siguientes:

- Transformada de Fourier
- Transformada Z
- Convolución
- Transformada discreta de Fourier
- Transformada Bilineal
- Transformada discreta de Hilbert

#### 1.4.4 PROCESAMIENTO PRÁCTICO

Este campo abarca todo el hardware y software necesario para realizar el procesamiento de las señales indicado por la parte teórica.

Dicho procesamiento se realiza con los *procesadores digitales de señales* (mejor conocidos como *DSP's*)



**1.4.4.1 Procesadores Digitales de Señales:** Son circuitos integrados que corresponden a la tecnología de integración a gran escala (VLSI, Very Large Scale Integration).

Un Procesador Digital de Señales (DSP, sigla en inglés de *Digital Signal Processor*) es un tipo de microprocesador, muy rápido y poderoso. Un DSP es único porque procesa señales en tiempo real. Esta capacidad de procesamiento en tiempo real hace a los DSP ideales para aplicaciones que no toleran ningún retardo.

Como casi la totalidad de las señales que existen en la naturaleza son de carácter analógico, los DSP's deben primeramente digitalizar las señales para luego procesarlas.

Para esto se usa conversores A/D o mejor conocidos como ADC's (dispositivos encargados de convertir una señal analógica en una digital).

Luego de esta conversión, viene el tratamiento de la señal, el cual implica repetidas operaciones numéricas, razón por la cual un DSP es altamente numérico y repetitivo. Los sistemas basados en DSP's deben trabajar en tiempo real, capturando y procesando información en el mismo momento que ocurre.

Los conversores análogo – digitales deben adquirir la información lo suficientemente seguido como para captar todas las fluctuaciones relevantes de las señales.

Si el ADC es muy lento se perderá información. El DSP también debe trabajar rápido para no perder información que le llega desde el ADC y además cumplir con el adecuado procesamiento de las señales.

Una de las más importantes características de un DSP es su capacidad de realizar operaciones de multiplicación y acumulación (MACs) en sólo un ciclo de reloj. No obstante ello, es necesario que el dispositivo posea la característica de manejar aplicaciones críticas en tiempo real.

Esto requiere de una arquitectura que soporte un flujo de datos a alta velocidad hacia y desde la unidad de cálculo y memoria. Esta ejecución a menudo requiere el uso de unidades DMA (Direct Memory Access) y generadores de direcciones duales (DAG's) que operan en paralelo con otras partes del chip. Los DAG's realizan los cálculos de direcciones, permitiendo al DSP buscar dos

datos distintos para operar con ellos en un solo ciclo de reloj, de tal forma que sea posible ejecutar algoritmos complejos en tiempo real.

Es importante para el DSP tener un mecanismo efectivo de salto para la ejecución de loops ya que el código generalmente programado es altamente repetitivo. La arquitectura permite realizar estos loops sin instrucciones adicionales ni demoras, las que al ejecutarse millones de veces empiezan a generar retardos significativos.

*1.4.4.1.1 Arquitecturas estándar en DSP:* Las arquitecturas de los computadores actuales están comúnmente clasificadas como *RISC's (Reduced Instruction Set Computers)* y *CISC's (Complex Instruction Set Computers)*. Estos últimos tienen un gran número de instrucciones sumamente poderosas, mientras que la arquitectura RISC posee pocas instrucciones y realiza movimientos de datos entre registros en un ciclo de máquina. Hoy en día los computadores RISC comienzan a reemplazar a los CISC's, porque se puede alcanzar un más alto rendimiento por medio del uso de un eficiente compilador como a través de la ejecución de instrucciones simples en forma ordenada.

Los estándares para DSP's tienen muchos rasgos de una arquitectura tipo RISC, pero los DSP's son procesadores de propósitos específicos cuya arquitectura es especialmente diseñada para operar en ambientes de alta necesidad de cálculo. Un DSP estándar ejecuta varias operaciones en paralelo mientras que un RISC usa unidades funcionales altamente eficientes que pueden iniciar y completar una instrucción simple en uno o dos ciclo de reloj.

*1.4.4.1.2 Diferencias entre Microcontroladores y DSP's :* Una de las diferencias más importante encontrada entre un DSP y un Microcontrolador es la estructura de memoria que poseen. En un microcontrolador es posible encontrar una memoria lineal, en la que se almacenan tanto datos como instrucciones de programa. Esto obliga a generar programas que no sobrepasen límites de tamaño ya que podrían sobrescribirse datos por instrucciones o viceversa. Un DSP posee dos bloques separados e independientes de memoria, cada uno con su propio bus de acceso, permitiendo así al procesador ir a buscar la siguiente instrucción y dato en el mismo ciclo de reloj (Fetch).

Otra diferencia importante entre un Microcontrolador y un DSP (y aún entre DSP's) es la cantidad de unidades de ejecución que poseen, y que son capaces de realizar operaciones en paralelo.

*1.4.4.1.3 Ventajas de los DSP's:* Entre las principales ventajas de usar DSP's, se puede mencionar las siguientes:

- La Tecnología VLSI (*Very Large Scale Integration*) da la posibilidad de diseñar sistemas con la capacidad para ejecutar procesamiento en tiempo real de muchas de las señales de interés para aplicaciones en comunicaciones, control, procesamiento de imagen, multimedia, etc.
- Los sistemas digitales son más confiables que los correspondientes sistemas analógicos.
- Los sistemas digitales ofrecen una mayor flexibilidad que los correspondientes sistemas analógicos.
- Mayor precisión y mayor exactitud pueden ser obtenidas con sistemas digitales, en comparación con los correspondientes sistemas analógicos.
- Un sistema programable permite flexibilidad en la reconfiguración de aplicaciones *DSP*.
- La tolerancia de los componentes en un sistema analógicos hacen que sea difícil para el diseñador controlar la exactitud de la señal de salida analógica. Por otro lado, la exactitud de la señal de salida para un sistema digital es predecible y controlable por el tipo de aritmética usada y el número de bits usado en los cálculos.
- Las señales digitales pueden ser almacenadas en un disco flexible, Disco Duro o CD-ROM, sin la pérdida de fidelidad más allá del ruido introducido por el conversor Análogo Digital (ADC). Éste no es el caso para las señales analógicas.

*1.4.4.1.4 Desventajas de los DSP's:* Entre las principales desventajas de usar DSP's, se puede mencionar las siguientes:

- La conversión de una señal analógica en digital, obtenida muestreando la señal y cuantificando las muestras, produce una distorsión que impide la reconstrucción exacta de la señal analógica original a partir de muestras cuantificadas.
- Existen efectos debidos a la precisión finita que deben ser considerados en el procesado digital de las muestras cuantificadas.

• Para muchas señales de gran ancho de banda, se requiere procesado en tiempo real. Para tales señales, el procesado analógico, o incluso óptico, son las únicas soluciones válidas. Sin embargo, cuando los circuitos digitales existen y son de suficiente velocidad se hacen preferibles.

*1.4.4.1.5 Aplicaciones de los DSP's:* Entre las principales aplicaciones de los DSP's se puede mencionar las siguientes:

- Wireless LAN
- Filtros Digitales
- Reconocimiento de voz
- Manejo de imágenes digitales
- Reproductores digitales de audio
- Teléfonos celulares
- Módems inalámbricos
- Cámaras digitales
- Automatización de fábricas

*1.4.4.1.6 Clases de DSP's:* Existen varias marcas y modelos de DSP's y el tipo de DSP's a usarse estará en íntima relación con la aplicación en la que se esté trabajando.

Entre los fabricantes y tipos más conocidos se destacan:

- **Texas Instruments** (<http://dspvillage.ti.com/docs/dspproducthome.jhtml>)

#### Familias

TMS320C6000

TMS320C5000

TMS320C2000

- **Motorola** (<http://www.motorola.com/semiconductors>)

#### Familias

56300

56800

56800E

MSC8100

- **Analog Devices** (<http://www.analog.com/technology/dsp/index.html>)

#### Familias

Blackfin

Familia Sharc

TigerSharc

ADSP-21xx

Es importante mencionar que la mayoría de los DSP's, debido a todos los procesos que deben realizar, no vienen como circuitos integrados solos sino como kits de trabajo en los cuales constan todos los dispositivos necesarios para realizar los procesamientos explicados.

## CAPITULO II

# REQUERIMIENTOS, ALCANCE Y LIMITACIONES DEL DISPOSITIVO AMCAEC I

## 2.1 REQUERIMIENTOS DE DISEÑO

### 2.1.1 GENERACIÓN DE TONOS

Para diseñar un audímetro digital de tonos puros de conducción aérea, primeramente se requiere generar tonos puros de la manera más exacta posible, ya que alteraciones significativas en la frecuencia de los tonos generados, pueden dar como resultado un audiograma erróneo, significando esto que no se podrá diagnosticar acertadamente patologías otológicas en base al mismo. Es por esta razón que los tonos serán generados de manera digital.

Estudios médicos, y posteriormente estandarizados por la ISO y por la ASA, demuestran que el oído es más sensible a unas frecuencias que a otras, y han concluido que, las frecuencias que sirven como referencia para analizar la sensibilidad del oído humano son las correspondientes a: 125, 250, 500, 1000, 2000, 4000 y 8000 Hz. Hay también frecuencias referenciales importantes que son las de 26 Hz y sus armónicos.

Es por esto necesario generar tonos puros correspondientes a las frecuencias ya mencionadas.

Los tonos puros se generarán de manera digital y en tiempo real, utilizando el ADSP-2181 del fabricante Analogic Devices.

Este DSP se encuentra incluido en el kit de trabajo EZ – KIT LITE del mismo fabricante.

Estos dos dispositivos y sus principales características se encuentran detalladas en la parte pertinente a requerimientos de hardware.

### **2.1.2 INTENSIDAD DE LOS TONOS GENERADOS**

Para poder realizar la audiometría, es necesario poder controlar de manera individual la intensidad de cada uno de los tonos generados.

Para esto, se debe poder variar la intensidad en saltos de dB lo más pequeños posibles para obtener mayor precisión en la audiometría.

Para realizar el control de volumen, se usará el CODEC AD-1847.

Este dispositivo y sus principales características se encuentran detalladas en la parte pertinente a requerimientos de hardware.

### **2.1.3 MEJORA DE RESPUESTA DE FRECUENCIA DE AUDIFONOS DE APLICACIÓN NO ESPECIFICA**

Es necesario transmitir el sonido a los oídos del paciente, lo cual se realizará con audífonos comerciales de uso común o aplicación no específica.

Para mejorar la respuesta de frecuencia de estos audífonos, se implementarán filtros digitales, aprovechando que una de las principales aplicaciones del ADSP-2181 es su uso para la implementación de dichos filtros.

De esta manera, se mejorará la calidad de audífonos de uso común y de bajo costo así como la calidad de los tonos generados, obteniéndose así una alta calidad en lo referente a la generación de tonos.

Un estudio más amplio sobre filtros se encuentra en la parte pertinente a filtros.<sup>5</sup>

### **2.1.4 SISTEMAS DE MEDICION DE INTENSIDAD AUDITIVA**

Existen 2 sistemas de referencia para medir la intensidad de un tono generado.

La primera es en dB y la segunda es en dBA

---

<sup>5</sup> Ver Capítulo II, numeral 2.3

### 2.1.4.1 Decibelio dB

El decibelio es una unidad logarítmica de medida que involucra niveles y que utilizada en diferentes disciplinas de la ciencia. En todos los casos se usa para comparar una cantidad con otra llamada de referencia. Normalmente el valor tomado como referencia es siempre el menor valor de la cantidad. En algunos casos puede ser un valor promediado aproximado.

Este nivel de referencia tomado en acústica, es una aproximación al nivel mínimo que hace que nuestro oído sea capaz de percibir un sonido.

La razón por la que se utiliza el decibelio, es que si no, se manejaría números o muy pequeños o excesivamente grandes, llenos de ceros, con lo que la posibilidad de error sería muy grande al hacer cálculos.

Además, se debe también considerar que el comportamiento del oído humano está más cerca de una función logarítmica que de una lineal, ya que no percibe la misma variación de nivel en las diferentes bandas de frecuencias.

Así, el nivel de referencia para la potencia sonora (en el aire), a una frecuencia de 1000 Hz, corresponde a  $10^{-12}$  w. Este valor corresponde a 0 dB.

El número de dB existentes en una relación se obtiene mediante la ecuación 2.1:

$$\# dB = 10 \log \left( \frac{P_m}{P_{ref}} \right)$$

**Ecuación 2.1**

Donde

$P_m$  = potencia medida

$P_{ref}$  = potencia de referencia

Este sistema de medición de intensidad fue determinado por la ASA (Sociedad Acústica de América – Acoustical Society of America)

### 2.1.4.2 Decibelio Ponderado dBA

El oído no se comporta igual para el mismo nivel de intensidad en diferentes frecuencias. Por ejemplo, si se toma un sonido lineal en toda la banda de 20 Hz a 20 kHz con un nivel de 30 dB, y si la respuesta del oído humano fuese lineal,



se escucharía la misma intensidad en las frecuencias bajas, las medias y las agudas.

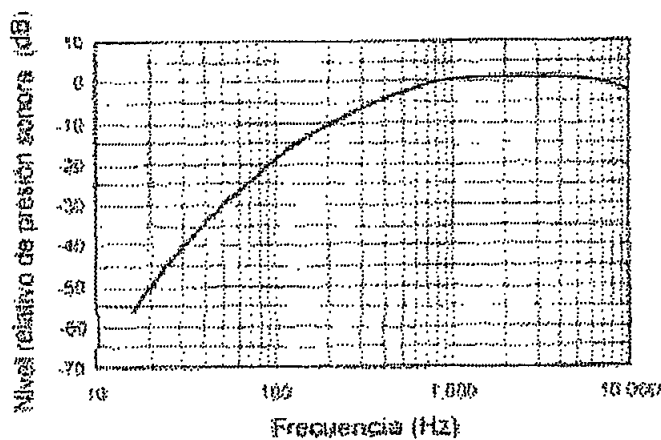
Sin embargo, esto es falso, ya que el oído humano tiene una menor sensibilidad en las frecuencias graves y en las más agudas frente a las medias.

Lo que más se percibe, por lo tanto, son las frecuencias medias, luego las más graves y finalmente las más agudas.

Como se ve es necesario encontrar una forma de ajustar los niveles de dB que se han medido con la percepción que el oído tiene de los mismos según cada frecuencia. Esta corrección se realiza ponderando los dB medidos mediante una tabla de ponderación (o red de ponderación) ya especificada y que se llama tabla "A". Los decibelios ya ponderados en "A" se representan como dBA y los no ponderados como dB.

Una red de ponderación es un sistema de corrección de los niveles de intensidad por frecuencias mediante factores de compensación fijos en decibelios que dependen de la red usada, a semejanza de lo que hace el oído humano.

La red de ponderación "A" se obtiene a partir de la figura 2.1, la cual es ya establecida y obtenida empíricamente:



**Figura 2.1: Tabla de ponderación A**

Basados en la figura 2.1, se puede realizar una tabla de compensación para las frecuencias de los tonos generados.

Esta compensación se indica explícitamente en la tabla 2.1:

Frecuencia (Hz)	125	250	500	1000	2000	4000	8000
Corrección (dB)	-16	-9	-3	0	+1	+1	-1

*Tabla 2.1*

Este sistema de medición de intensidad fue determinado por la ISO (Organismo Internacional de Estándares – International Standards Organization) y su único objetivo fue el de adoptar una unidad estándar para la medición de audio.

Para pasar de un sistema a otro, únicamente se debe aplicar el factor de corrección respectivo según la tabla 2.1.

La tabla 2.2 muestra un ejemplo de lo expresado:

Frecuencia	63	125	250	500	1000	2000	4000	8000
DB	70	79	80	82	85	96	98	100
Corrección	-26	-16	-9	-3	0	+1	+1	-1
DBA	44	63	71	79	85	97	99	99

**Tabla 2.2**

#### 2.1.4 TRAZO DE AUDIOGRAMA

Para trazar y visualizar el audiograma se requiere un medio en el cual se puedan almacenar los datos de la intensidad a la cual el paciente ha detectado el sonido de cada tono y en el cual también se pueda visualizar el audiograma.

Para esto se utilizará un software, el cual funcionará en cualquier computador personal que tenga un puerto serial con el fin de conectarse e interactuar con el EZ-Kit Lite.

El computador será usado para control, almacenamiento de datos y para el trazo del audiograma.

EL software debe tener las siguientes opciones básicas:

- Permitir seleccionar el oído que se desea evaluar (izquierdo o derecho)
- Permitir seleccionar el tono
- Permitir variar la intensidad

- Almacenar datos de la intensidad a la cual el paciente escuchó cada tono
- Trazar el audiograma

Es importante también mencionar que, en la plantilla en la cual se va a trazar el audiograma, se debe especificar si se está usando el sistema ISO o el sistema ASA, para saber si las intensidades están expresadas en dBA o en dB.

Para el caso particular del trabajo en cuestión, se trabajará con el sistema ASA, ya que como se verá posteriormente en el análisis de resultados en el capítulo IV, se usará como referencia un equipo comercial que trabaja con dB, es decir, en el sistema ASA.

## 2.2 REQUERIMIENTOS DE HARDWARE

### 2.2.1 ADSP 2181

#### 2.2.1.1 Descripción General

El ADSP 2181 es un circuito integrado diseñado específicamente para procesamiento digital de señales. Básicamente se compone de 3 partes:

- Tres unidades computacionales
- Generadores de direcciones de datos
- Un secuenciador de programa.

Estas tres unidades son la base de la arquitectura de la familia de procesadores ADSP 2100, pero el ADSP 2181 además contiene 2 pórticos seriales, un pórtico interno DMA (acceso directo a memoria) de 16 bits, un pórtico byte DMA, un temporizador programable, una bandera I/O, amplias capacidades de interrupción y memoria de datos y de programa incorporadas al circuito integrado.

Este chip une 80K bytes de memoria integrada al circuito configurados como 16K palabras de 24 bits de memoria RAM de programa y 16K palabras de 16 bits cada una de memoria RAM de datos.

Adicionalmente, se tiene la posibilidad de trabajar con instrucciones de multiplicación, se puede realizar redondeo, se puede realizar operaciones ALU

sin necesidad de almacenar el resultado, transferencia de memoria I/O y se puede enmascarar interrupciones globales para aumentar la flexibilidad.

Este circuito es fabricado con tecnología CMOS de bajo consumo de potencia, doble metal y alta velocidad y opera con un ciclo de máquina de 30ns por instrucción.

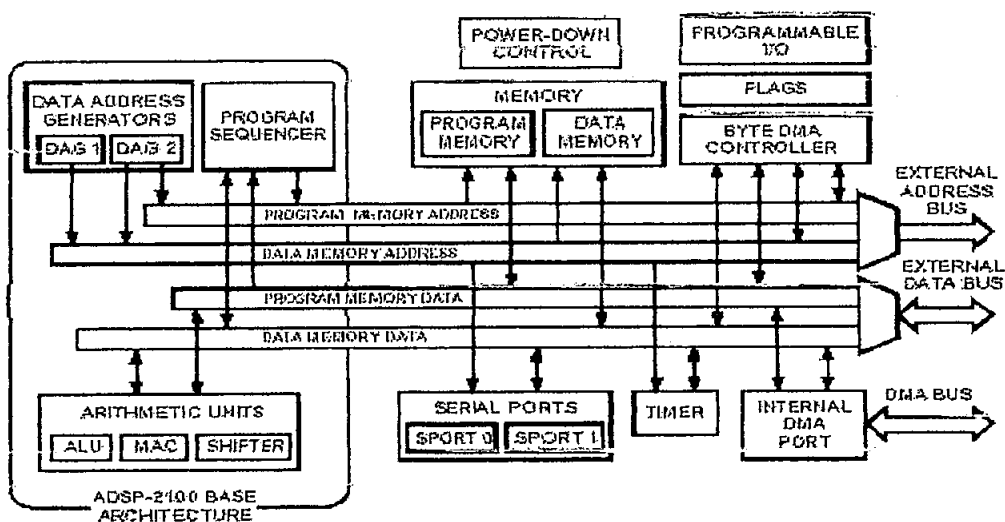
Otra de las ventajas que presenta el ADSP 2181 es que, debido a su arquitectura flexible y su sencillo set de instrucciones, se pueden realizar varias operaciones en paralelo. Así, en un mismo ciclo de máquina se puede :

- Generar la siguiente dirección de programa
- Ejecutar la siguiente instrucción
- Mover uno o dos datos
- Actualizar uno o dos punteros de direcciones
- Realizar una operación computacional.

Esto ocurre mientras el procesador continua realizando las siguientes tareas:

- Recibiendo y transmitiendo datos a través de los dos pórtricos seriales
- Recibiendo y transmitiendo datos a través del pórtrico interno DMA
- Recibiendo y transmitiendo datos a través del pórtrico byte DMA
- Decrementando el timer

La figura 2.2 muestra un diagrama de bloques del ADSP-2181, allí se puede observar cada una de los componentes del chip.



**Figura 2.2: Diagrama de Bloques Funcional del ADSP-2181**

### 2.2.1.2 Sistemas de Desarrollo

El Software de desarrollo de la Familia ADSP-2100 ofrece una amplia gama de herramientas de desarrollo tanto a nivel de software como a nivel de hardware. Uno de los más importantes conjuntos de herramientas a nivel de software, y que además será el usado en el presente trabajo es el compuesto por Assembler, Linker y Simulator.

#### 2.2.1.2.1 "Assembler"

Este software permite escribir el código fuente del programa que se desea ejecutar, y luego ensamblarlo. Una vez hecho esto, se crean dos archivos de tipo objeto.

El Assembler tiene una sintaxis algebraica muy sencilla de programar y de depurar. Los registros mneumónicos y sus significados constan en la siguiente lista:

- AX0,AX1,AY0,AY1 Entradas del ALU
- AR Resultado del ALU
- AF Realimentación del ALU
- MX0,MX1,MY0,MY1 Entradas del multiplicador
- MR0,MR1,MR2 Resultados del multiplicador
- MF Realimentación del multiplicador
- SI Entrada del shifter
- SE Exponente del shifter
- SR0,SR1 Resultado del shifter
- SB Bloque del shifter
- PX Bus de intercambio PMD-DMD
- I0-I7 Registros index del DAG
- M0-M7 Registros modify del DAG
- L0-L7 Registros length del DAG
- PC Contador de programa
- CNTR Contador para lazos
- ASTAT Estado aritmético
- MSTAT Estado de modo

- SSTAT                                    Estado del stack
- IMASK                                    Máscara de interrupciones
- ICNTL                                    Modos de control de interrupciones
- RX0,RX1                                Registros de datos recibidos
- TX0,TX1                                Registro de datos transmitidos

#### 2.2.1.2.2        “Linker”

Este software, al ser ejecutado, toma los dos archivos con extensión .obj creados por el Assembler y los combina, creando un archivo ejecutable.

El archivo ejecutable creado por el Linker será el programa que el Simulator use para realizar la simulación y además es el programa que será transferido a la memoria de programa del ADSP 2181 para ser ejecutado. La transferencia se realiza con un PROM splitter.

El splitter modifica al archivo ejecutable creado por el Linker, dándole un formato que puede ser reconocido por un programador PROM. Con esta herramienta se puede pasar a una memoria PROM el programa creado por el usuario.

#### 2.2.1.2.3        “Simulator”

Este software permite simular el programa realizado sin necesidad de trabajar con el hardware. Esto es posible gracias a que se provee al usuario de una simulación de instrucciones interactivas con un interface de usuario reconfigurable que muestra diferentes partes del ambiente de hardware.

Otra opción que se tiene como sistema de desarrollo es el compilador C.

Para usar este sistema, se debe escribir el programa en alguna de las versiones del lenguaje de programación C compatibles con el ADSP 2181 y luego ejecutar el compilador C, el cual genera un código fuente.

Junto con esta alternativa se tiene una librería, la cual contiene alrededor de 100 funciones específicas que permiten relacionar el lenguaje C con el ADSP 2181.

### 2.2.1.3 Arquitectura

La figura 2.3 muestra un diagrama de bloques global del ADSP-2181.

### 2.2.1.3.1 Unidades Computacionales:

El procesador contiene 3 unidades computacionales independientes:

- **La ALU:** Es la unidad lógica aritmética y permite realizar operaciones aritméticas y lógicas
- **El MAC :** Es el acumulador / multiplicador y permite realizar operaciones de multiplicación en un solo ciclo, suma y multiplicación y, multiplicación y resta con 40 bits disponibles para almacenar los resultado.
- **El SHIFTER :** Es el cambiador y permite realizar cambios lógicos y aritméticos, normalización y denormalización así como operaciones de exponenciales derivados. Además permite implementar eficientemente control de formatos numéricos incluyendo multipalabra y representaciones de punto flotante en bloque.

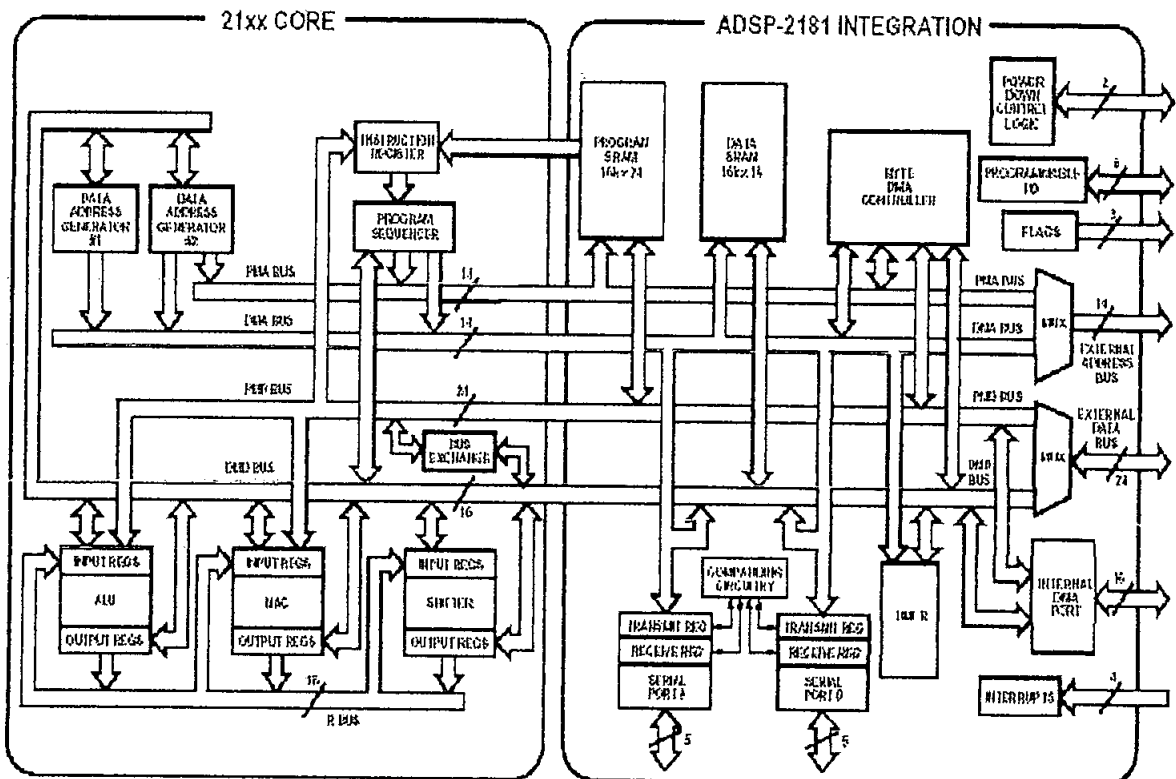


Figura 2.3: Diagrama de Bloques Global del ADSP-2181

La unidad computacional procesa datos de 16 bits y acepta operaciones multipresición, lo cual quiere decir que tiene diferentes formatos para ingresar los números con los que se realizarán los cálculos , dependiendo de la

exactitud que requieran los resultados. Cada formato corresponde a una diferente variación por bit, a un máximo valor positivo, a un máximo valor negativo, a un número de bits que representan decimales y a un número de bits que representan enteros. Los diferentes formatos se pueden apreciar en la Tabla 2.3.

El bus de resultados internos R conecta las unidades computacionales de tal manera que la salida de cualquier unidad puede ser la entrada de cualquier unidad para el siguiente ciclo. Se cuenta además con un poderoso secuenciador y dos generadores de direcciones de datos, los cuales aseguran la correcta entrega de los operandos a las unidades de cálculo. Este secuenciador soporta instrucciones de saltos y, llamadas y retornos de subrutinas en un solo ciclo de máquina.

El ADSP-2181 tiene además lazos contadores internos, con lo cual los cálculos iterativos se ejecutan sin necesidad de instrucciones explícitas de saltos, aumentando de esta manera la rapidez de los cálculos y, por otro lado, reduciendo el número de instrucciones necesarias.

Format	Number of Integer Bits	Number of Fractional Bits	Largest Positive Value (0x7FFF) In Decimal	Largest Negative Value (0x8000) In Decimal	Value of 1 LSB (0x0001) In Decimal
1.15	1	15	0.999969482421875	-1.0	0.000030517578125
2.14	2	14	1.999938964843750	-2.0	0.000061035156250
3.13	3	13	3.999877929687500	-4.0	0.000122070312500
4.12	4	12	7.999755859375000	-8.0	0.000244140625000
5.11	5	11	15.999511718750000	-16.0	0.000488281250000
6.10	6	10	31.999023437500000	-32.0	0.000976562500000
7.9	7	9	63.998046875000000	-64.0	0.001953125000000
8.8	8	8	127.996093750000000	-128.0	0.003906250000000
9.7	9	7	255.992187500000000	-256.0	0.007812500000000
10.6	10	6	511.984375000000000	-512.0	0.015625000000000
11.5	11	5	1023.968750000000000	-1024.0	0.031250000000000
12.4	12	4	2047.937500000000000	-2048.0	0.062500000000000
13.3	13	3	4095.875000000000000	-4096.0	0.125000000000000
14.2	14	2	8191.750000000000000	-8192.0	0.250000000000000
15.1	15	1	16383.500000000000000	-16384.0	0.500000000000000
16.0	16	0	32767.000000000000000	-32768.0	1.000000000000000

**Tabla 2.3: Formatos Numéricos del ADSP-2181**

2.2.1.3.2 *Registros de Datos*

EL chip tiene registros generadores de datos (DAGs), los cuales proveen las direcciones necesarias para realizar operaciones simultáneas.



Existen 2 DAGs: el DAG 1, que provee direcciones de memoria de datos; y, el DAG 2, que provee direcciones de memoria de programa y de memoria de datos.

Cada DAG tiene 12 registros de 14 bits cada uno, los cuales son de tres tipos:

- **Registros Index I:** Son 4 y sirven para almacenar punteros
- **Registros Modify M:** Son 4 y sirven para actualizar punteros
- **Registros Length L:** Son 4 y sirven para implementar lazos.

Estos registros trabajan como 4 grupos independientes de 3 registros cada uno (I1, M1, L1; I2, M2, L2; I3, M3, L3; I4, M4, L4 ), y pueden ser leídos o escritos mediante el bus DMA. Si se altera un registro I, automáticamente se alteran los registros M y L asociados.

#### 2.2.1.3.3 *Buses de Transferencia*

Para conseguir una adecuada transferencia de datos, se tienen 5 buses internos, los cuales son:

- **PMA:** Bus de direcciones de memoria de programa
- **PMD:** Bus de direcciones de memoria de datos
- **DMA:** Bus de acceso directo a memoria
- **DMD:** Bus de datos obtenidos desde la memoria de datos
- **R:** Bus de resultados

#### 2.2.1.3.4 *Pórticos Seriales*

El ADSP-2181 tiene 2 pórticos seriales sincrónicos para comunicación serial con dispositivos periféricos y para comunicaciones multiprocesador.

Estos pórticos son el Pórtico Serial 1 y el Pórtico Serial 0, y sus principales características son:

- Trabajan en modo bidireccional
- Para la sincronización, pueden usar un reloj externo o generar su propio reloj interno.
- Pueden trabajar con palabras de 3 a 16 bits y de manera alternativa, pueden trabajar con la ley A o ley  $\mu$
- Tienen un interface multicanal y ofrece la posibilidad de hacer multiplexación por división de tiempo.

- Pueden ser configurados para tener dos interrupciones externas (IRQ0 y IRQ1) con señales de bandera de entrada y de salida. Se puede trabajar con estas interrupciones aún cuando se esté generando el reloj de manera interna.

#### 2.2.1.3.5 Interrupciones

El ADSP-2181 tiene 4 pines específicos para interrupciones externas: IRQ2, IRQ1, IRQ0 e IRQE. Adicionalmente, se puede configurar al pínico serial para interrupciones con bandera de entrada o de salida, teniendo en total 6 opciones de interrupciones externas.

Además, se tienen las interrupciones internas del timer, del pínico DMA, de los dos pínicos seriales por software y el circuito de control del Power Down.

Todas las interrupciones pueden ser enmascarables de manera individual (excepto el Down y el Reset). Además se puede priorizarlas y definir si se activan por flanco o por estado.

El cuadro 2.1 muestra las direcciones del vector de interrupciones y sus prioridades:

Source of Interrupt	Interrupt Vector Address (Hex)
Reset (or Power-Up with PUCR = 1)	0000 ( <i>Highest Priority</i> )
Power-Down (Nonmaskable)	002C
IRQ2	0004
IRQ1	0008
IRQ0	000C
SPORT0 Transmit	0010
SPORT0 Receive	0014
IRQE	0018
DMA Interrupt	001C
SPORT1 Transmit or IRQ1	0020
SPORT1 Receive or IRQ0	0024
Timer	0028 ( <i>Lowest Priority</i> )

**Cuadro 2.1 : Prioridad de las Interrupciones y Direcciones del Vector de Interrupciones**

#### 2.2.1.3.6 Operación en Baja Potencia

Hay 3 modos de trabajo que reducen significativamente la disipación de potencia cuando el dispositivo está en modo de espera:

- **Power Down:** En este modo, el consumo de potencia es sumamente bajo cuando el dispositivo está inactivo. Esta opción es controlada por software o por hardware.
- **Idle:** En este modo, el procesador espera indefinidamente que se produzca una interrupción consumiendo muy baja potencia. Una vez que se detecta la interrupción, se da el servicio de la misma y luego se retorna el estado Idle.
- **Slow Idle:** Con esta opción, el reloj interno del procesador se vuelve más lento, por lo cual se reduce el consumo de potencia. La frecuencia del reloj se reduce en una fracción de la velocidad nominal del mismo, y esta fracción es especificada por un divisor seleccionable el cual es especificado con la instrucción Idle.

EL pin de clockout puede ser también deshabilitado para reducir la disipación de potencia.

#### *2.2.1.3.7 Asignación de Pines*

El ADSP-2181 tiene 128 pines, y está disponible en 2 tipos de encapsulados: el TQFP y el PQFP.

El dispositivo a implementar utilizará el ADSP-2181 con encapsulado PQFP. La figura 2.4 muestra una vista del chip y el cuadro 2.2 muestra su asignación de pines.

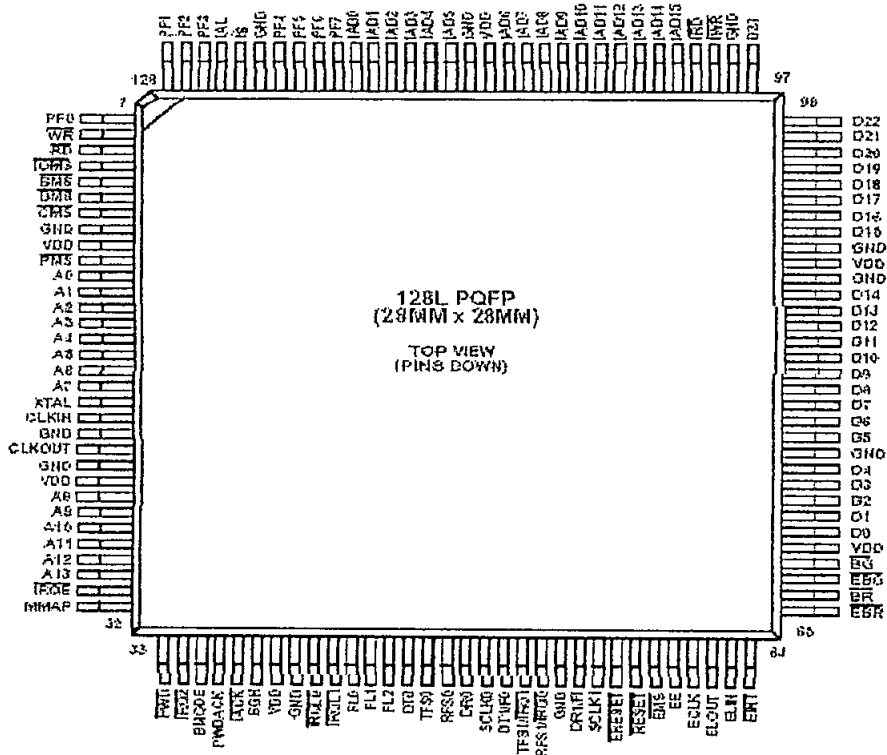


Figura 2.4: Vista superior del encapsulado PQFP del ADSP-2181

PQFP Number	Pin Name	PQFP Number	Pin Name	PQFP Number	Pin Name	PQFP Number	Pin Name
1	PFI	33	PWT0	65	IRR	97	I23
2	WR	34	IRQ2	66	BR	98	GND
3	RD	35	BNCODE	67	BRG	99	IWR
4	IOM0	36	PWDACK	68	BR	100	IRR
5	BMS	37	JACK	69	VDD	101	IAD15
6	DM0	38	BCODE	70	I30	102	IAD14
7	DM1	39	VDD	71	I31	103	IAD13
8	GND	40	GND	72	I32	104	IAD12
9	VDD	41	IRQL0	73	I33	105	IAD11
10	PMS	42	IRQL1	74	I34	106	IAD10
11	A0	43	FL0	75	GND	107	IAD9
12	A1	44	FL1	76	I35	108	IAD8
13	A2	45	FL2	77	I36	109	IAD7
14	A3	46	D10	78	I37	110	IAD6
15	A4	47	TFS0	79	I38	111	VDD
16	A5	48	RFS0	80	I39	112	GND
17	A6	49	I30	81	I40	113	IAD5
18	A7	50	SCLK0	82	I41	114	IAD4
19	XTAL	51	DTLPO	83	I42	115	IAD3
20	CLKIN	52	RES1/IRQ1	84	I43	116	IAD2
21	GND	53	RES2/IRQ2	85	I44	117	IAD1
22	CLKOUT	54	GND	86	GND	118	IAD0
23	GND	55	DR1/P1	87	VDD	119	PF7
24	VDD	56	SCLK1	88	GND	120	PF6
25	A8	57	IRESET	89	I45	121	PF5
26	A9	58	RESET	90	I46	122	PF4
27	A10	59	IAMS	91	I47	123	GND
28	A11	60	IE	92	I48	124	IS
29	A12	61	ECLK	93	I49	125	IAL
30	A13	62	ELCOUT	94	I20	126	PF3
31	IRQ0	63	BLIN	95	I21	127	PF2
32	MMAP	64	IRNT	96	I22	128	PF1

Cuadro 2.2: Asignación de pines del ADSP-2181

## 2.2.2 CODEC 1847

### 2.2.2.1 Descripción General

El Codec AD1847 es un pórtico de sonido estéreo que integra en un solo chip la conversión de datos de audio y el control de funciones. Tiene dos secciones claramente definidas: una digital y una analógica. Trabaja con una fuente de alimentación de 5 V, la cual debe ser implementada externamente y además posee un interface serial el cual le permite trabajar en conjunto con un DSPs o también con cualquier tipo de tarjeta. Es especialmente diseñado para trabajar con audio digital y multimedia. La Figura 2.5 muestra un diagrama de ejemplo:

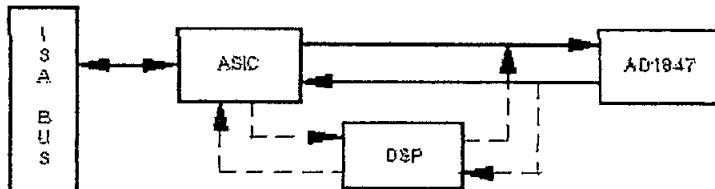


Figura 2.5: AD1847 conectado con DSP

Se puede ver que los requerimientos para trabajar con circuitos externos son mínimos, ya que el mismo integrado posee filtros de salida, lo cual permite que los datos transmitidos por el DAC (digital - to - analog converter) no requieran ningún tratamiento adicional.

EL rango dinámico del Codec excede los 70 dB para la banda de audio de 20 kHz y presenta frecuencias de muestreo desde 5.5 kHz hasta 48 kHz, las cuales son obtenidas con cristales externos.

El chip incluye un par de conversores AD y un par de conversores DA. Esto permite obtener muestras digitales a partir de una entrada analógica y además generar una señal de salida analógica a partir de muestras discretas.

Las entradas al conversor AD pueden ser seleccionadas de 4 pares de señales estéreo analógicas :

- Línea uno
- Línea dos
- Auxiliar 1
- Auxiliar 1

El Codec trabaja con dos salidas de 16 bits cada una (dos canales), las cuales son transmitidas a través del p rtico serial y cuenta tambi n con dos entradas de 16 bits las cuales provienen del conversor AD (resultado de muestrear una se al anal gica).

Los tipos de datos que el Codec acepta y genera son los siguientes:

- Datos digitales lineales PCM de 16 bits en complemento a 2
- Datos lineales PCM de 8 bits sin asignaci n de magnitud
- Datos de 8 bits con ley  $\mu$  o con ley A

La Figura 2.6 muestra un diagrama de bloques funcional del Codec AD1847:

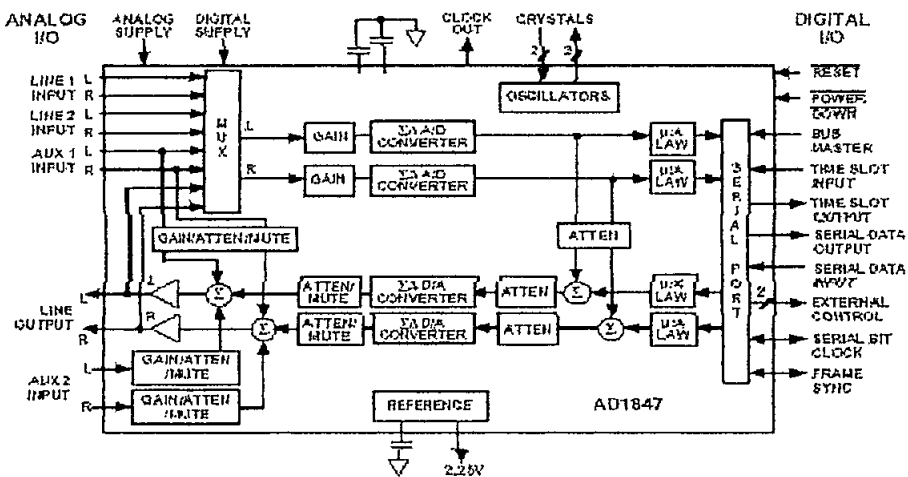


Figura 2.6: Diagrama de bloques funcional del Codec AD1847

Se puede observar que las entradas (convertidores AD) son precedidas de amplificadores y que tambi n antes y despu s de las salidas (convertidores DA) se tiene atenuadores, lo cual permite tener un control total e independiente de volumen. Las ganancias tanto de los amplificadores como de los atenuadores son controladas por software.

#### 2.2.2.2 Descripci n Funcional de Audio

Cada una de las entradas del Codec, las cuales ya fueron mencionadas anteriormente, son multiplexadas para ser transmitidas a la etapa del amplificador programable interno, el cual permite trabajar con ganancias independientes para cada canal e incluso se tiene la opci n de mute para cada canal.

La vía de datos analógica-digital tiene un filtro pasivo de un polo, el cual permite eliminar todos los efectos no deseados del aliasing. La vía de datos digital-analógica posee un atenuador programable y estas salidas analógicas están disponibles en pines externos. Si se quiere obtener la salida por otros medios, tales como audífonos o parlantes, es necesario implementar circuitería adicional específica.

#### 2.2.2.3 Reloj y Frecuencias de Muestreo

El Codec opera con 2 cristales externos, los cuales deben ser conectados a los pines XTAL1 y XTAL2. Con estos cristales es posible obtener una amplia gama de frecuencias de muestreo. Adicionalmente se puede anexar cristales externos para usos más específicos. Para el funcionamiento del Codec, es indispensable que se conecte al menos el cristal en el pin XTAL1( pin 19), ya que es la entrada que se selecciona por defecto luego de cada reset. En este caso, se deshabilita XTAL2 y el pin respectivo debe estar conectado a tierra. Las especificaciones del dispositivo indican que las frecuencias recomendadas de los cristales son de 16.9344 MHz y 24.576 MHz. Si se usa estos cristales, las frecuencias de muestreo que se pueden seleccionar son: 5.5125, 6.615, 8, 9.6, 11.025, 16, 18.9, 22.05, 27.42857, 32, 33.075, 37.8, 44.1 y 48 kHz.

#### 2.2.2.4 Interface Serial de Datos

El interface serial del AD1847 usa multiplexación por división de tiempo en un esquema totalmente compatible con el pórtilo serial de los DSPs. Para que exista dicha compatibilidad, los pórtilos de los DSPs deben estar configurados en modo multi-canal con slots de tiempo de 16 bits. Esta configuración cambia según el DSP que se use.

El Codec AD1847 es siempre el bus serial maestro y transmite de manera permanente el reloj para sincronización serial (SCLK) y la trama para sincronización serial de datos (SDFS).

### 2.2.2.5 Reset y Power Down

Un reset total se define como cualquier evento que requiera que tanto las secciones analógicas como digitales retornen a un estado estable conocido.

Power Down es un estado en el cual el consumo de potencia del Codec es reducido de manera significativa y es usado cuando el chip se encuentra en estado de espera.

Si se desea obtener tanto el reseteo como el paso a estado de bajo consumo de potencia del chip, se requiere enviar un cero lógico al pin de power down del Codec.

Si se desea únicamente resetear el Codec, se debe enviar un cero lógico al pin de reset del integrado. Este reseteo actúa únicamente sobre la sección digital del dispositivo.

Estas dos opciones se pueden conseguir por software o por hardware.

### 2.2.2.6 Definiciones de los Formatos de Datos

Existen 4 tipos de formatos de datos que pueden ser usados al trabajar con el Codec AD1847:

- 16 bits con signo
- 8 bits sin signo
- 8 bits compandidos<sup>6</sup> con ley  $\mu$
- 8 bits compandidos con ley A

Este tipo de datos son los más usados para una amplia gama de aplicaciones.

#### 2.2.2.6.1 Formato de 16 bits con signo

Los datos con signo de 16 bits, o también llamados datos de 16 bits en complemento a dos, constituyen el método estándar para representar datos de audio. Este formato permite un rango dinámico de 94.5 dB y es el más usado por usuarios de reproductores de discos compactos.

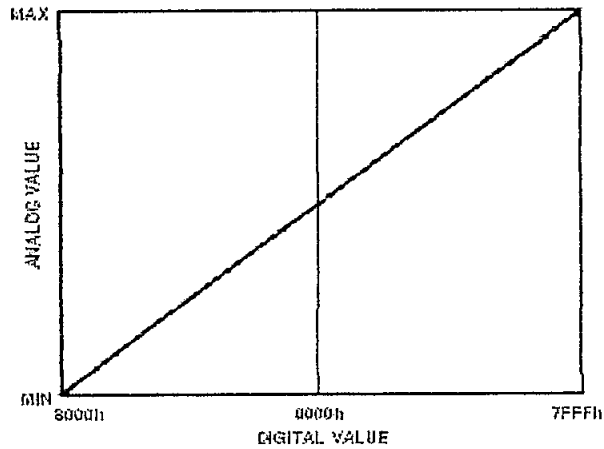
---

<sup>6</sup> Compandir: Contracción de comprimir y expandir. Este término se usa para indicar que se realiza un proceso de comprensión y luego de expansión



El mínimo valor de amplitud de la señal analógica es  $-32768(8000H)$  mientras que el máximo es  $32767(7FFFH)$ . Los valores intermedios se pueden obtener usando una interpolación lineal simple entre los valores máximo y mínimo.

La Figura 2.7 muestra un esquema para este tipo de datos:



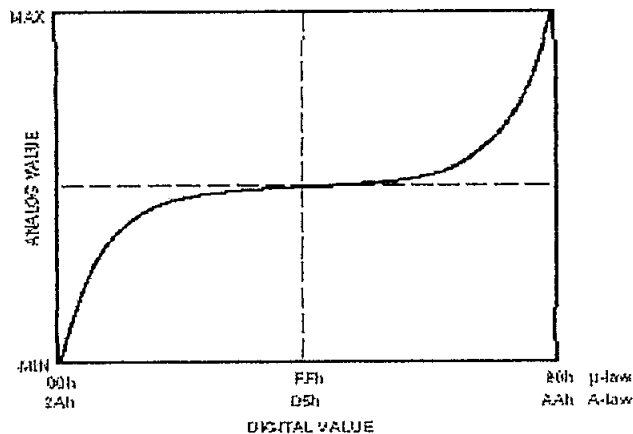
**Figura 2.7: Formato de datos de 16 bits con signo**

#### 2.2.2.6.2 Formato de 8 bits compandidos

Los datos de 8 bits compandidos (Ley A y Ley  $\mu$ ) son usados en la industria de las telecomunicaciones, específicamente en ISDN (integrated service digital network). La ley  $\mu$  es el estándar usado por los Estados Unidos y el Japón, mientras que el estándar A es el usado por Europa.

El audio compandido permite obtener un rango dinámico de 64 o 72 dB, dependiendo de la ley usada, únicamente con 8 bits por muestra. Esto se consigue usando una ecuación no lineal la cual asigna más códigos digitales a las señales de menor amplitud; es decir, se usa una cuantificación no uniforme.

La Figura 2.8 muestra un esquema para este tipo de datos:

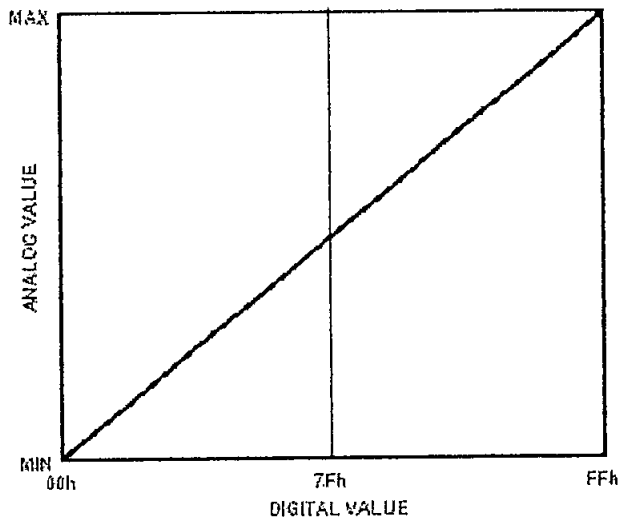


**Figura 2.8: Formato de datos de 8 bits compandidos**

### 2.2.2.6.3 Formato de 8 bits sin signo

Este tipo de datos es comúnmente usado en la industria de computadores personales. Este formato ofrece 48 dB de rango dinámico. EL valor 0(00H) se usa para representar el menor valor de amplitud de la señal analógica mientras que el 255(FFH) representa el mayor valor de amplitud de la señal. Los valores intermedios se calculan usando interpolación lineal entre el valor máximo y el mínimo.

La Figura 2.9 muestra un esquema para este tipo de datos:



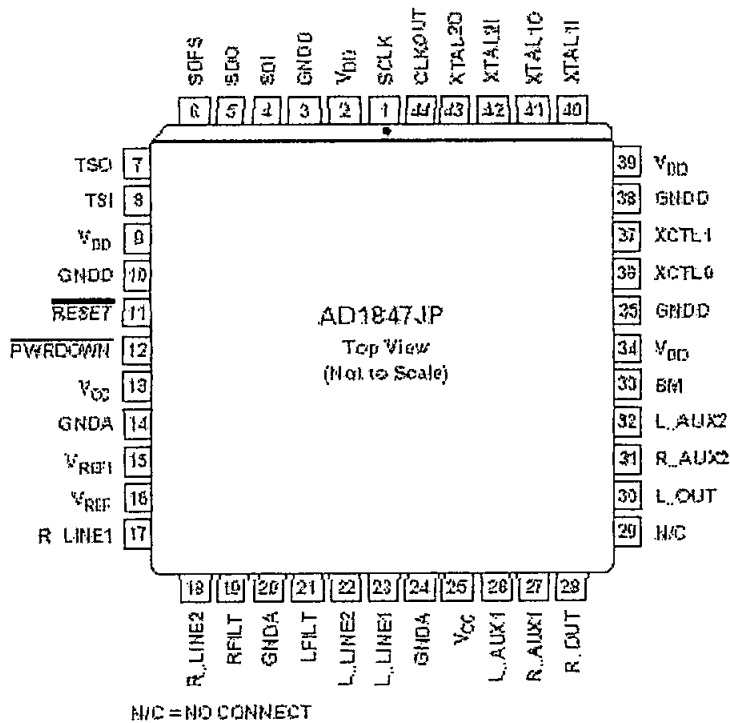
**Figura 2.9: Formato de datos de 8 bits sin signo**

### 2.2.2.7 Asignación de Pines

El Codec AD1847 tiene 44 pines, y está disponible en 2 tipos de encapsulados: el TQFP y el PLCC.

El dispositivo a implementar utilizará el Codec AD1847 con encapsulado PQFP.

La figura 2.10 muestra una vista del chip y su asignación de pines:



**Figura 2.10: Vista superior del encapsulado PQFP del CODEC AD1847**

El anexo 5 detalla la función específica de cada uno de los pines.

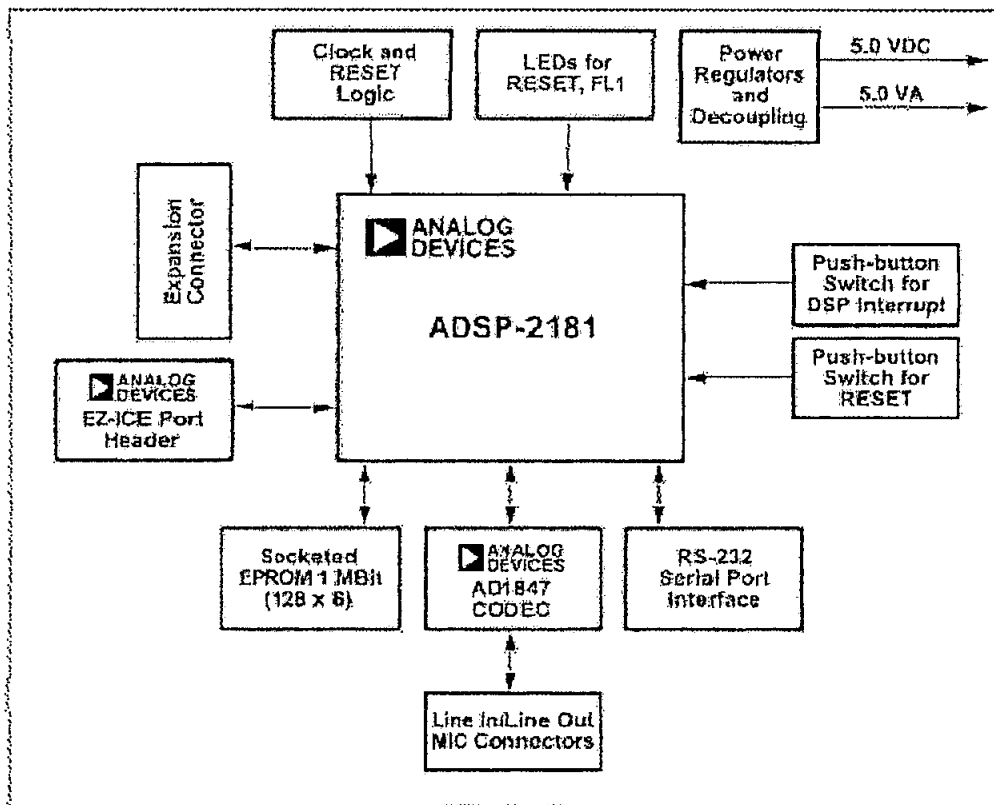
Es importante mencionar que toda la programación para la inicialización del Codec AD1847 viene, en general, dada por el fabricante para cada uno de los DSP compatibles. Así, la inicialización del Codec AD1847 para e ADSP-2181 consta de manera explícita en el anexo 6.

### 2.2.3 EZ-KIT LITE

El EZ-Kit Lite es una tarjeta en la cual están integrados los elementos necesarios para desarrollar aplicaciones usando el ADSP-2181.

En esta tarjeta están conectados entre sí el ADSP-2181 y el CODEC AD1847 y toda la circuitería necesaria para que estos operen correctamente.

La Figura 2.11 muestra un esquema de la arquitectura de esta tarjeta:



**Figura 2.11: Diagrama de Bloques Sistemático del EZ-Kit Lite**

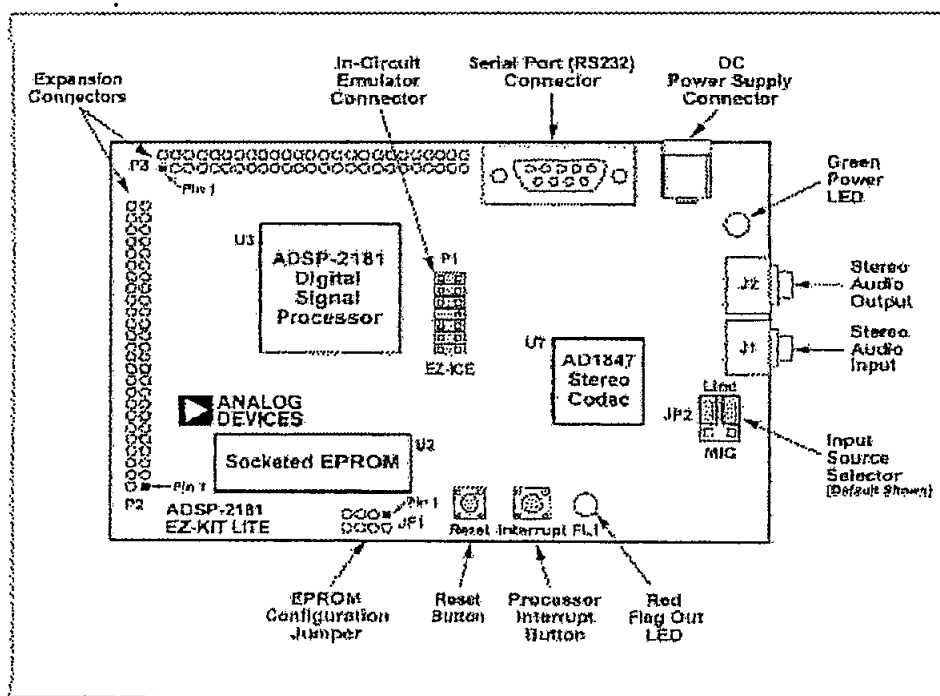
Adicionalmente, se tiene todos los elementos complementarios para que lo único que se requiera al trabajar con este kit sea una entrada de audio (micrófono o dispositivo similar), una salida de audio (parlantes, amplificadores o dispositivo similar) y alimentación de poder. La entrada y la salida de audio corresponden a J1 y J2 respectivamente en la Figura 2.12.

Los elementos complementarios con los que cuenta este kit son los siguientes:

- **Pórtico serial 232:** Permite establecer comunicación con un computador personal. Trabaja a una velocidad de 9600 kbps. Usa un conector hembra DB9.
- **Memoria EPROM:** Sirve para almacenar los programas que serán ejecutados por el DSP cuando la tarjeta trabaje en modo stand alone.
- **Entrada de alimentación:** Esta tarjeta usa una alimentación de 9 voltios DC, la cual está conectada directamente al DSP y a un regulador voltaje 7805 , el cual brinda una fuente de 5 voltios DC para alimentar al CODEC.

- **Extensión de pórticos seriales:** Son una extensión de los pórticos seriales del ADSP-2181. Corresponde a P2 y P3 en la Figura 2.12.
- **Selector de señal de entrada:** Es un jumper de 6 pines y permite seleccionar el tipo de entrada de audio con la que se va a trabajar, ya sea esta un micrófono o una señal de línea . Corresponde a JP2 en la Figura 2.12.
- **2 Leds:** Trabajan como indicadores. El uno es rojo y el otro es verde. El verde indica que existe una fuente de alimentación correcta mientras que el rojo indica que se está ejecutando un programa determinado.
- **Dos pulsadores:** Uno para el reseteo de la tarjeta y otro para ejecutar las interrupciones del DSP.
- **Conector de expansión:** A pesar de que el ADSP-2181 no trabaja con IDMA (acceso indirecto para memoria de datos), cuenta con un conector de 14 pines de expansión el cual hace posible usar dicha herramienta usando dispositivos periféricos. Corresponde a P1 en la Figura 2.12.
- **Pórtico de la EPROM:** Es un jumper de 8 pines y permite configurar el funcionamiento de la memoria EPROM. En caso de no usar estos pines, la memoria trabaja en el modo configurado por defecto por el fabricante. Corresponde a JF1 en la Figura 2.12.

La Figura 2.12 muestra la forma física de la tarjeta vista desde su parte superior:



**Figura 2.12: Vista superior de la tarjeta EZ-Kit Lite**

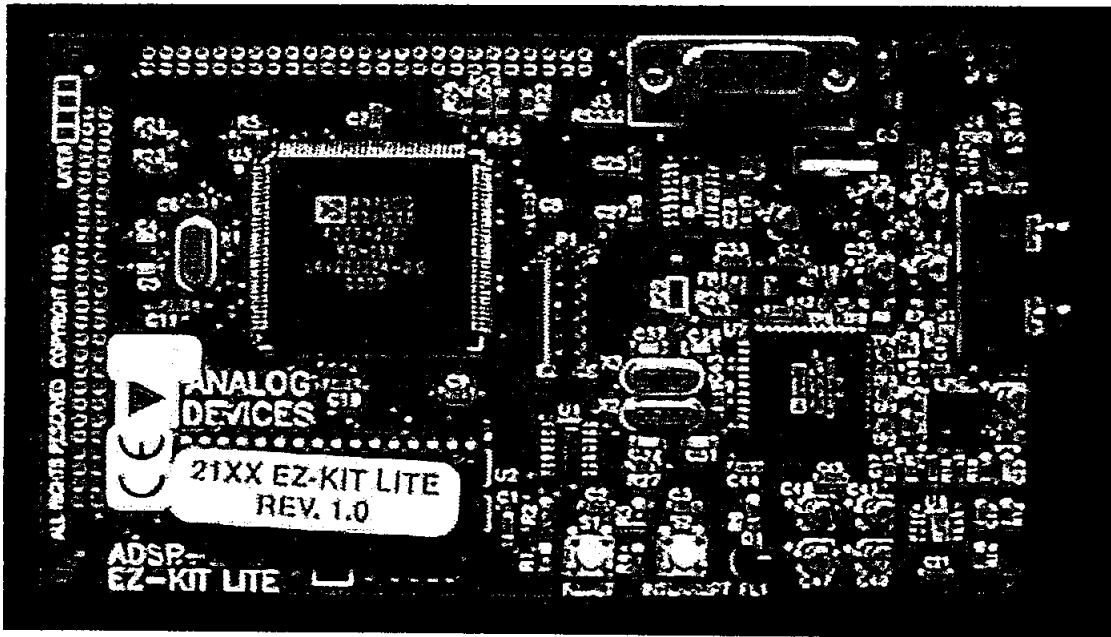
Para procesar una señal usando este kit, se siguen los siguientes pasos:

1. Se ingresa la señal analógica de audio
2. Se transmite la señal analógica al Codec
3. Se digitaliza la señal analógica en el Codec
4. Se transmite la señal digital al DSP
5. Se procesa la señal digital en el DSP
6. Se transmite la señal digital al Codec
7. Se transforma la señal digital en analógica en el Codec
8. Se transmite señal analógica procesada a la salida de audio.

Estos pasos son los regularmente usados al procesar una señal. Sin embargo, en ciertas aplicaciones, no se ingresa señal de audio analógica, sino que se puede generar una señal digital de manera directa con el DSP, para luego procesarla con el mismo dispositivo (ADSP-281) y posteriormente continuar con el proceso mencionado desde el paso el 6.

Es muy recomendable usar esta tarjeta, ya que, debido al gran número de pines tanto del ADSP-2181 como del Codec AD1847, resulta poco práctico que el usuario realice las conexiones de manera manual.

La Figura 2.13 muestra una foto del EZ-Kit Lite:



*Figura 2.13: Foto del EZ-Kit Lite*

## 2.3 FILTROS DIGITALES

### 2.3.1 ASPECTOS GENERALES

Los filtros digitales son una clase de sistemas lineales, causales e invariantes con el tiempo que, teóricamente hablando, dejan pasar ciertas componentes de frecuencia y rechazan completamente otras.

El anexo 7 explica de manera más detallada las propiedades de linealidad, causalidad e invariabilidad en el tiempo de un sistema.

Una señal, luego de haber sido sometida al filtrado digital, puede haber cambiado en amplitud y fase o sólo en uno de éstos parámetros dependiendo del tipo de filtro.

El filtrado digital es parte del procesado de una señal digital. Se le da la denominación de digital más por su funcionamiento interno que por su dependencia del tipo de señal a filtrar. (ya que esta puede ser analógica o digital)

El proceso de filtrado digital realiza internamente un procesamiento de los datos de entrada de forma matemática-numérica, operando cada muestra de la señal según ciertos parámetros, y obteniendo a la salida el resultado de dicha operación.

Para que este proceso pueda llevarse a cabo, tanto la salida como la entrada del filtro serán digitales, por lo que, para uso de filtros digitales en señales analógicas, será necesario una conversión analógico-digital y digital-analógico. Esto quiere decir que se debe realizar la digitalización de la señal analógica que se quiere procesar.

Los filtros digitales se usan frecuentemente para tratamiento digital de imágenes o para tratamiento de sonido digital.

Para poder entender el trabajo con filtros digitales, es necesario revisar los conceptos referentes a ecuaciones de diferencias, transformada Z y funciones de transferencia, ya que son los puntos más importantes.

### 2.3.2 ECUACIONES DE DIFERENCIAS

Como ya se mencionó anteriormente, el producto de digitalizar una señal es obtener un conjunto de muestras discretas, de tal manera que tanto la señal de entrada como la de salida de un sistema serán cada una, conjuntos de muestras discretas.

Una ecuación de diferencias es una sumatoria en la cual la entrada y la salida se relacionan por un conjunto de coeficientes constantes. Esto significa que la entrada y la salida satisfacen una sumatoria específica. Por lo tanto, una ecuación de diferencias define el comportamiento de un sistema.

Una ecuación de diferencias en general está representada por la siguiente expresión:

$$\sum_{k=0}^N a_k y[n-k] = \sum_{m=0}^M b_m x[n-m]$$

Donde:

$y[n]$  = muestras de salida

$x[n]$  = muestras de entrada

$b_m$  = coeficientes de entrada

$a_k$  = coeficientes de salida



N = número de muestras de entrada

M = número de muestras de salida

Las ecuaciones de diferencias son sistemas lineales e invariantes en el tiempo (sistemas LTI).

### 2.3.3 LA TRANSFORMADA Z

La transformada Z constituye la herramienta más utilizada en el análisis y diseño de sistemas en tiempo discreto, y es la análoga a la transformada de Laplace para el estudio de sistemas en tiempo continuo.

El empleo de la transformada Z permite hacer un análisis de las características de un determinado filtro definido en tiempo discreto, y a su vez permite la transformación de señales, originalmente en tiempo continuo a señales digitales, lo que se traduce en que este método facilita el análisis y diseño de sistemas a través de funciones de transferencia en el dominio Z.

Para determinar la respuesta de un sistema a una entrada previamente aplicada, es necesario resolver la ecuación de diferencias. Con el método de la transformada Z, la solución de la ecuación de diferencias, se convierte en un problema de naturaleza algebraica.

Si se considera a  $x[n]$  como una secuencia infinita que representa a una señal analógica muestreada, se tiene que la transformada Z de  $x[n]$ , es decir,  $X(Z)$ , viene dada por la ecuación 2.2:

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n}$$

*Ecuación 2.2: Transformada Z de  $x[n]$*

El anexo 8 contiene mayor información sobre la transformada Z.

### 2.3.4 FUNCIÓN DE TRANSFERENCIA

Si se considera que  $x[n]$  es una secuencia de pulsos (señal analógica muestreada), y que a esta secuencia la sometemos a un proceso H, de tal manera que la salida sea la secuencia discreta  $y[n]$ , se dice que H es la función de transferencia del sistema.

Así pues, se tiene lo siguiente:

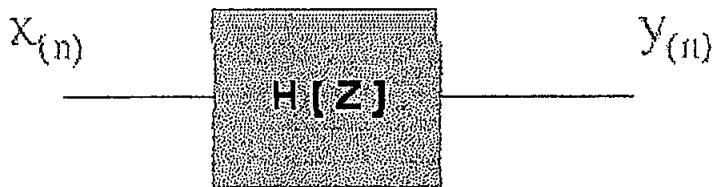
$$Y(z) = X(z) * H(z)$$

De donde podemos concluir que:

$$H(z) = \frac{Y(z)}{X(z)}$$

Siendo H(z) la función de transferencia del sistema.

La Figura 2.14 muestra la representación típica de un sistema con entrada x, salida y; y, función de transferencia H.



**Figura 2.14: Sistema discreto con función de transferencia**

Como se puede observar, la función de transferencia se expresa en el dominio Z y es la transformada Z de la ecuación de diferencias la que determina el comportamiento de un sistema específico.

Si se opera sobre la expresión general de una ecuación de diferencias, de tal manera que y [n] aparezca de manera explícita, se tiene que:

$$y[n] - \sum_{k=1}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k]$$

Y si a esta ecuación de diferencias se le aplica la transformada Z, se obtiene:

$$1 - \sum_{k=1}^N a_k z^{-k} = \sum_{k=0}^M b_k z^{-k}$$

Como en general, el número de muestras de entrada es el mismo que el número de muestras de salida, se tiene que N = M. Además, considerando que todo el lado izquierdo de esta ecuación representa a y[n] mientras que el lado derecho a x[n], aplicando la fórmula de H(z), obtenemos que la función de

transferencia de una ecuación de diferencias viene dada en general por la ecuación 2.3:

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}}$$

**Ecuación 2.3: Función de transferencia de una ecuación de diferencias**

Esta función de transferencia corresponde a la de un filtro digital en general. Es evidente la relación y correspondencia que existe entre una ecuación de diferencias y su respectiva función de transferencia.

Independientemente de la realización concreta del filtro (analógico, digital o mecánico) la forma de comportarse del mismo se describe por su función de transferencia. Ésta determina la forma en que la señal que aplicamos cambia en amplitud y en fase al atravesar el filtro. La función de transferencia elegida tipifica el filtro.

#### 2.3.4.1 Tipos de sistemas según la función de transferencia

De acuerdo a la forma que tome la función de transferencia, el sistema al que ésta describe adquiere diferentes características.

Si la función de transferencia es:

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}}$$

El sistema presenta las siguientes características:

- Sistema IIR (infinite impulse response)
- Sistema de polos y ceros
- Sistema recursivo
- Sistema regresivo
- Sistema ARMA (auto regressive moving average)

Si la función de transferencia es:

$$H(z) = \frac{A}{1 - \sum_{k=1}^N a_k z^{-k}}$$

El sistema presenta las siguientes características:

- Sistema IIR
- Sistema de solo polos
- Sistema recursivo
- Sistema regresivo
- Sistema AR

Si la función de transferencia es:

$$H(z) = \sum_{k=0}^N b_k z^{-k}$$

El sistema presenta las siguientes características:

- Sistema FIR (finite impulse response)
- Sistema de solo ceros
- Sistema no recursivo
- Sistema no regresivo
- Sistema MA (moving average)

### 2.3.5 TIPOS DE FILTROS

Hay varios tipos de filtros, así como distintos parámetros de clasificación.

De acuerdo con la parte del espectro que dejan pasar y que atenúan hay:

- Filtros de paso alto.
- Filtros de paso bajo.
- Filtros de banda
  - elimina banda
  - multibanda
  - pasa todo

- Resonador
- Oscilador
- Filtro peine (comb filter)
- Filtro ranura (notch filter)

De acuerdo con su orden:

- primer orden
- segundo orden
- n-ésimo orden

De acuerdo con la estructura con que se implementan:

- Lattice
- Varios en cascada
- Varios en paralelo

De acuerdo con el tipo de respuesta ante entrada unitaria:

- FIR (*Finite Impulse Response*)
- IIR (*Infinite Impulse Response*)
- TIIR (Truncated Infinite Impulse Response)

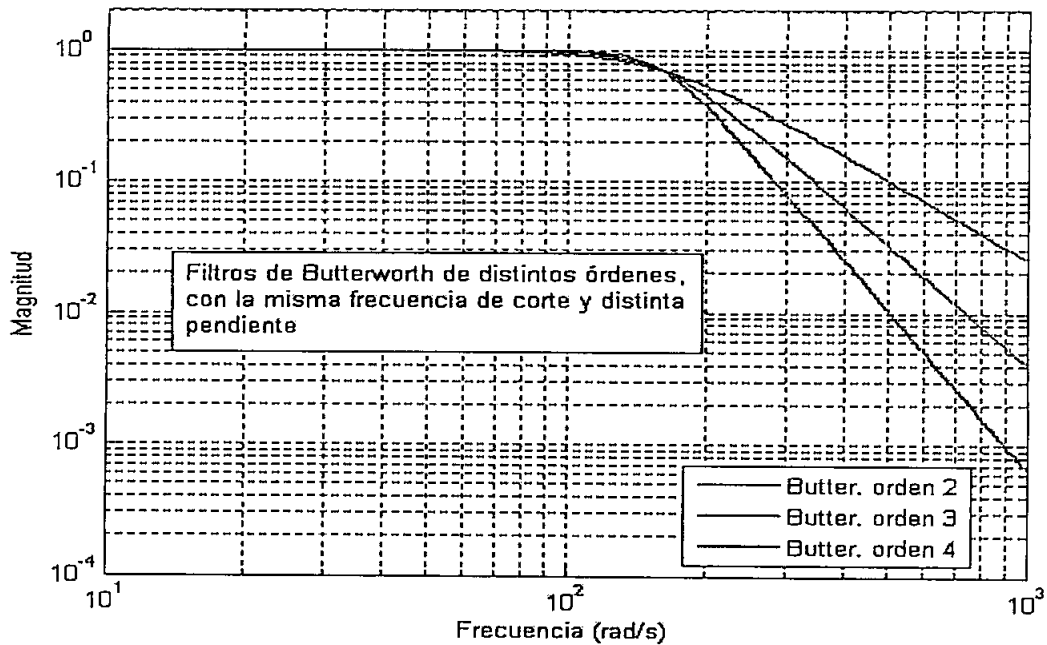
El anexo 9 contiene información complementaria sobre la respuesta de un sistema al impulso.

De todos los tipos de filtros mencionados anteriormente, los más importantes y comúnmente usados son los filtros pasa bajos, pasa altos, pasa banda, FIR e IIR, así como es también importante analizar el orden de un filtro.

### 2.3.6 ORDEN DE UN FILTRO

El orden de un filtro describe el grado de aceptación o rechazo de frecuencias por arriba o por debajo, de la respectiva frecuencia de corte. Es decir, determina su nivel de selectividad. Por ejemplo, un filtro de primer orden, cuya frecuencia de corte sea igual a  $(F)$  presentará una atenuación de 6 dB a la primera octava ( $2F$ ), 12dB a la segunda octava ( $4F$ ), 18dB a la tercer octava ( $8F$ ) y así sucesivamente. Uno de segundo orden tendría el doble de pendiente (representado en escala logarítmica). Esto se relaciona con los polos y ceros: los polos hacen que la pendiente suba con 20dB y los ceros que baje; de esta forma los polos y ceros pueden compensar su efecto.

La Figura 2.15 muestra la variación del comportamiento de un filtro en función de su orden:

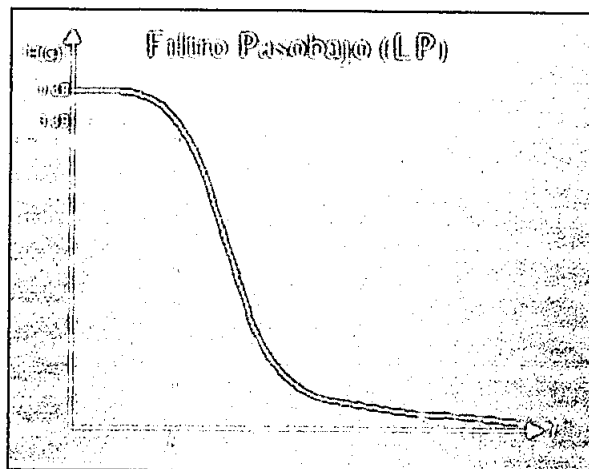


**Figura 2.15: Filtros de Butterworth de distintos órdenes**

En la Figura 2.15, se nota claramente que a medida que aumenta el orden del filtro, aumenta su nivel de selectividad, y por lo tanto, aumenta su pendiente, aproximándose más a la frecuencia de corte.

### 2.3.7 FILTROS PASA BAJOS

Son aquellos que dejan pasar frecuencias desde cero hasta un límite superior. La Figura 2.16 muestra la respuesta típica de un filtro pasa bajos:



**Figura 2.16: Filtro pasa bajos**

### 2.3.8 FILTROS PASA BANDA

Son aquellos que dejan pasar frecuencias desde un límite inferior diferente de cero hasta un límite superior distinto del infinito. La Figura 2.17 muestra la respuesta típica de un filtro pasa banda:

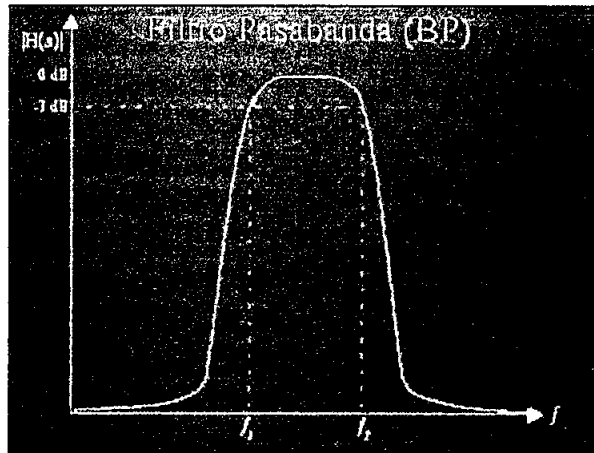


Figura 2.17: Filtro pasa banda

### 2.3.9 FILTROS PASA ALTOS

Son aquellos que dejan pasar frecuencias desde un límite inferior diferente de cero hasta el infinito. La Figura 2.18 muestra la respuesta típica de un filtro pasa altos:

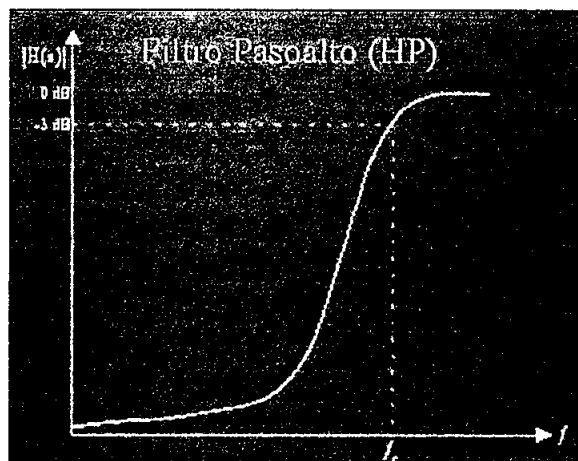


Figura 2.18: Filtro pasa altos

### 2.3.10 FILTROS FIR

Este tipo de filtros presenta una respuesta finita ante el impulso, y de allí proviene su nombre. ( FIR = finite impulse response).

La respuesta es de duración finita, ya que si la entrada se mantiene en cero durante  $M$  periodos consecutivos, la salida será también cero.

La ecuación de diferencias que describe el comportamiento de un filtro FIR de orden  $M$  es:

$$y[n] = \sum_{k=0}^M b_k x[n - k]$$

Y la función de transferencia respectiva es:

$$H(z) = \sum_{k=0}^M b_k z^{-k}$$

La secuencia  $b_k$  corresponde a los coeficientes del filtro.

Según la función de transferencia de un filtro FIR, sus principales características son:

- Sistemas MA (moving average)
- Sistema de sólo ceros
- Sistemas no recursivos
- Sistemas no regresivos

**Sistemas MA** :Resulta fácil entender que los filtros FIR son sistemas MA, ya que por simple inspección nos damos cuenta que su función de transferencia únicamente tiene un numerador polinómico.

**Sistemas de sólo ceros**: Notamos que la función de transferencia tiene un denominador constante y por lo tanto sólo puede tener ceros.

**Sistemas no recursivos** : En este tipo de filtros, no existe recursión, Esto quiere decir que la salida depende sólo de la entrada y no de valores pasados de la salida, por lo que la respuesta es una suma ponderada de valores



pasados y presentes de la entrada. De ahí que a estos filtros se les llame también Media en Movimiento (Moving Average)

**Sistemas no regresivos** : Son sistemas no regresivos ya que en un sentido estadístico, la salida  $y[n]$  no tiene regresión a sus valores pasados.

### 2.3.11 FILTROS IIR

Este tipo de filtros presenta una respuesta infinita ante el impulso, y de allí proviene su nombre. (IIR = infinite impulse response).

La respuesta es de duración infinita ya que si la entrada se mantiene en cero durante  $M$  periodos consecutivos, la salida no necesariamente es cero.

La ecuación de diferencias que describe el comportamiento de un filtro IIR de orden  $M$  es:

$$y[n] - \sum_{k=1}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k]$$

Y la función de transferencia respectiva es:

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}}$$

La secuencias  $a_k$  y  $b_k$  corresponden a los coeficientes del filtro.

Según la función de transferencia de un filtro IIR, sus principales características son:

- Sistemas ARMA (auto regressive moving average)
- Sistema de polos y ceros
- Sistemas recursivos
- Sistemas regresivos

**Sistemas ARMA** : Resulta fácil entender que los filtros IIR son sistemas ARMA, ya que por simple inspección nos damos cuenta que su función de transferencia tiene un numerador polinómico y un denominador polinómico y además son sistemas autoregresivos, como se explicará más adelante.

**Sistemas de polos y ceros:** Notamos que la función de transferencia tiene un numerador polinómico y un denominador polinómico, por lo tanto debe tener polos y ceros.

**Sistemas recursivos :** En este tipo de filtros, existe recursión, o sea que la salida depende de la entrada y de valores pasados de la salida, por lo que la respuesta es una suma ponderada de valores pasados y presentes de la entrada y de la salida.

**Sistemas regresivos :** Son sistemas regresivos o auto regresivos ,ya que en un sentido estadístico, la salida  $y[n]$  si tiene regresión a sus valores pasados.

### **Diferencias entre filtros FIR e IIR**

Para elegir si se va a trabajar con un filtro FIR o IIR, es importante considerar los siguientes aspectos:

- Los filtros IIR producen en general distorsión de fase, es decir la fase no es lineal con la frecuencia.
- Los filtros FIR son de fase lineal.
- El orden de un filtro IIR es mucho menor que el de un filtro FIR para una misma aplicación.
- Los filtros FIR son siempre estables.
- Los filtros IIR son altamente selectivos en frecuencia
- Los filtros FIR son los más adecuados para compensar distorsión de fase
- Los filtros IIR son los más adecuados para compensar distorsión de amplitud

### **2.3.1.2 TÉCNICAS DE DISEÑO DE FILTROS**

El proceso de diseño de un filtro digital, de cualquier tipo, requiere tres pasos generales:

- Establecer las especificaciones del filtro para unas determinadas prestaciones. Estas especificaciones son las mismas que las requeridas por un filtro analógico : frecuencias de para-banda y pasa-banda, atenuaciones, ganancia dc, etc.

- Determinar la función de transferencia que cumpla las especificaciones requeridas.
- Implementar la función de transferencia en hardware o software.

Es recomendable realizar un gráfico en el que consten los requerimientos antes mencionados. La Figura 2.19 es un ejemplo de esto:

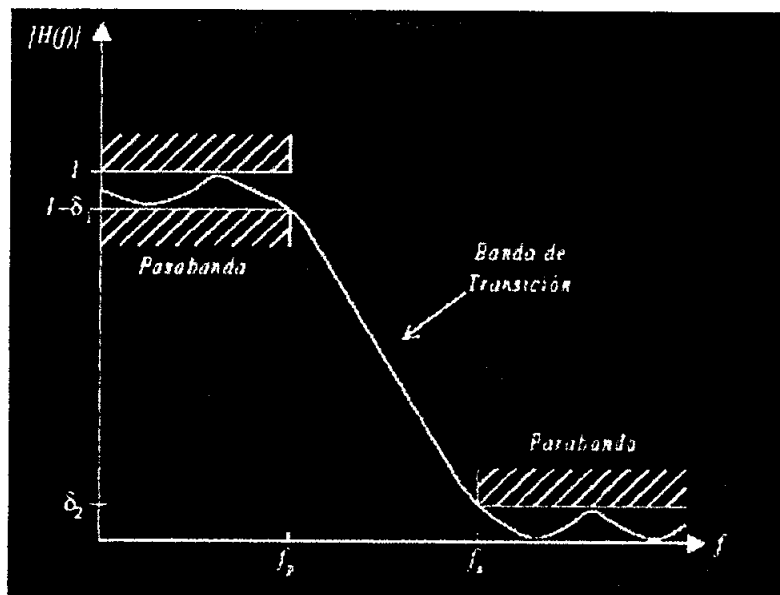


Figura 2.19: Especificaciones de un filtro

## 2.4 ALCANCE DEL DISPOSITIVO AMCAEC I

El dispositivo será capaz de generar tonos de frecuencia e intensidad variable, y almacenar los datos referentes a la intensidad (medidos en dB) a la cual el paciente escuchó cada tono, y finalmente, valiéndose de dicha información, trazar el audiograma según el sistema ASA.

Con esto, se podrá diagnosticar el tipo y severidad de hipoacusia de conducción aérea que padece la persona examinada. La hipoacusia de conducción aérea es el tipo de patología que se determina con la evaluación de tonos puros.

Las frecuencias de los tonos generados serán: 125, 250, 500, 1000, 2000, 4000 y 8000 Hz mientras que la intensidad de los mismos podrá variarse en saltos de dB lo más pequeños posibles.

Los tonos, al ser generados digitalmente, prácticamente no presentarán error en cuanto a su frecuencia.

Respecto a la intensidad de los mismos, el valor máximo al que puede llegar un tono es el correspondiente al rango dinámico del CODEC-AD1847, es decir, 94.5 dB. Este valor se alcanzará con saltos lo más pequeños posibles.

Este examen, a nivel profesional, se realiza con audífonos de alta precisión, los cuales presentan un costo aproximado de USD 800. Para suplir esto, se implementarán filtros digitales con el ADSP-2181, los cuales compensarán la respuesta de frecuencia no ideal de audífonos de aplicación no específica de bajo costo que se utilizarán. Con estos filtros, se optimizará la respuesta de frecuencia de amplitud de los audífonos en cuestión.

El sistema audiométrico a diseñar e implementar estará constituido por un computador personal, el cual estará conectado al EZ-Kit Lite, al cual a su vez se conectarán a los audífonos. Todas las funciones de selección, cambio de tonos e intensidad y control serán manejadas por software.

Por esta razón, se deberá conectar el dispositivo al computador de manera permanente mientras sea usado con el fin de obtener un correcto desempeño, ya que de no ser así, se verán totalmente limitadas las capacidades del mismo.

El nombre del dispositivo será AMCAEC I (Audímetro de Conducción Aérea Económico).

## 2.5 LIMITACIONES DEL DISPOSITIVO AMCAEC I

- El dispositivo no alcanzará a dar a los tonos generados los 100 dB de intensidad que se indican en la plantilla para trazar el audiograma. Si bien es cierto que los requerimientos planteados por la ASA indican que se debe alcanzar una intensidad de 100 dB, no existe mayor inconveniente al no cumplirlo, ya que, como se mencionó en el capítulo I, es fácil darse cuenta de que aquella persona que no escucha un tono de 94.5 dB, padece una sordera realmente aguda y más que una evaluación, requiere tratamiento inmediato.

- EL dispositivo, para poder funcionar, debe estar conectado de manera permanente a un computador personal, es decir, no funciona en STAND ALONE.

Esto no se constituye en un inconveniente serio, ya que hoy en día, la gran mayoría de médicos tienen en su consultorio un computador que cumple con los requerimientos para trabajar con el dispositivo. Además, el computador facilita mucho el uso del dispositivo, ya que el interface con el usuario es muy amigable y además, es mucho más sencillo para un nuevo usuario, aprender a manejar sólo un software en lugar de aprender a manejar un hardware y software.

- El filtro a implementar únicamente compensará la amplitud de la respuesta de frecuencia.

El no compensar la fase de la respuesta de frecuencia no afecta al sonido ya que al estar constituido el sonido por una sola frecuencia, si esta se atrasa o se adelanta, no afectará la percepción del sonido del paciente. Debemos recordar que la distorsión de fase se produce cuando algunas componentes de frecuencia se desfasan de manera diferente con respecto de otras. Al no haber varias frecuencias, no es necesario compensar la fase de la respuesta de frecuencia.

Además, el resultado final al que se llega es al audiograma, en el cual lo más importante, aparte de la exactitud de los tonos generados, es la intensidad de los mismos, así que lo más importante es realizar una compensación en la amplitud de la respuesta de frecuencia.

- El audímetro podrá ser usado por cualquier persona que tenga conocimientos básicos de computación y otología, es decir, no necesita ser utilizado por un experto en la materia. El punto en el cual se requiere la presencia de un experto, es en el análisis e interpretación de los resultados arrojados por el audiograma.

Para aprender a usar este dispositivo, el usuario deberá referirse al manual de uso, del cual se hablará de manera amplia y explícita en el capítulo III.

## CAPITULO III

### IMPLEMENTACIÓN DE HARDWARE Y DE SOFTWARE

#### 3.1 GENERACION DE TONOS

##### 3.1.1 CALCULO DEL AVANCE

Se requiere tener tonos de las siguientes frecuencias.

- 125 Hz
- 250 Hz
- 500 Hz
- 1000 Hz
- 2000 Hz
- 4000 Hz
- 8000 Hz

Para esto, se generan muestras digitales de ondas sinusoidales correspondientes a cada una de las frecuencias mencionadas. Estas muestras digitales son obtenidas con el ADSP-2181

Esto da como resultado una señal discreta en el dominio del tiempo; pero este conjunto de valores discretos, son enviados al Codec-1847 del EZ-Kit Lite, donde se convierten al dominio analógico con la ayuda de los conversores con los que dicho Kit cuenta.

El Codec trabaja con una frecuencia de muestreo de 48 KHz (fs). De esta manera, ya que la máxima frecuencia generada es de 8 KHz, se cumple holgadamente con el teorema de Nyquist, dando así mayor exactitud a la onda generada.

El hecho de que el Codec trabaje con la frecuencia indicada quiere decir que cada  $20.833 \mu\text{s}$  ( $1/f_s$ ) se produce una interrupción y se ejecuta el cálculo del valor de la amplitud de la onda sinusoidal en dicho instante, dando como resultado una muestra digital.

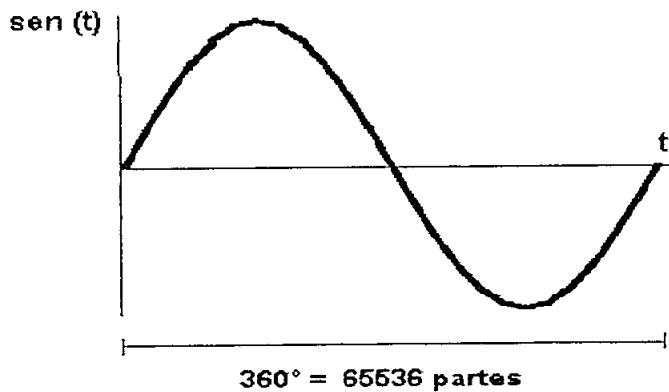
El valor del ángulo del cual se calcula el seno en cada interrupción, depende de la frecuencia de la onda generada y dicho ángulo en grados se almacena en el registro AR (el cual es de 16 bits).

Por lo tanto, a un período completo de la señal (independientemente de su frecuencia), se los divide en 65536 partes<sup>7</sup> ( $2^{16}$ ), lo cual corresponde a  $360^\circ$ .

Esto significa que el DSP ve a los grados como partes, y que cada una de estas partes corresponde a  $5,493 \times 10^{-3}^\circ$ .

Es también importante mencionar que, dado que el valor del seno se almacena de igual manera en el registro AR<sup>1</sup>, cada muestra digital es representada con 16 bits.

En la figura 3.1 se puede apreciar el ciclo completo de una señal de frecuencia  $f$ , lo cual corresponde a  $360^\circ$  y a 65536 partes.

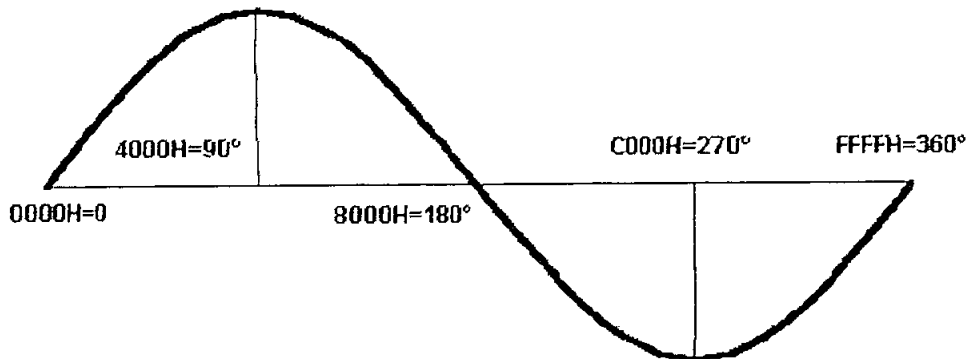


**Figura 3.1**

---

<sup>7</sup> Esto se explica más detalladamente en el análisis de la rutina utilizada para el cálculo del seno

En la figura 3.2 se puede apreciar la correspondencia entre los grados para la señal y las partes para el DSP:



**Figura 3.2**

Acorde con lo ya mencionado y aplicando la ecuación 3.1, se puede determinar el número de muestras que constituirán un período completo de cada uno de los tonos generados.

$$Nm = \frac{T}{Ts} = \frac{fs}{f}$$

**Ecuación 3.1**

Donde:

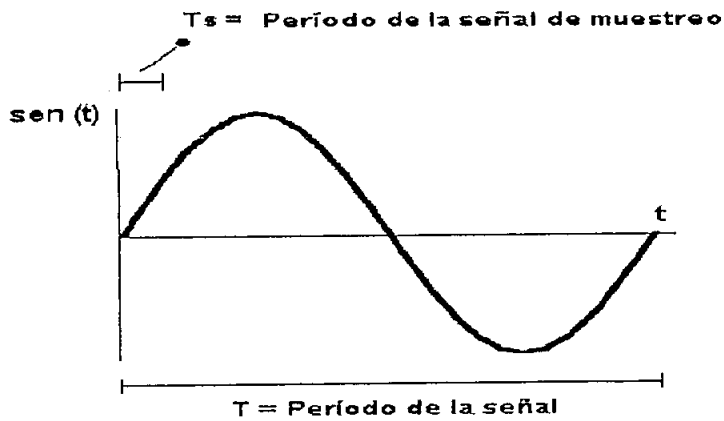
**Nm** = número de muestras contenidas en un ciclo completo de una sinusoide de frecuencia **f**

**fs** = frecuencia de muestreo (**Ts** = período de la señal de muestreo)

**f** = frecuencia del tono a generar (**T** = período de la onda a generar)

La figura 3.3 permite apreciar claramente que el período de la señal de muestreo, es decir, el utilizado por el Codec, es mucho menor que el de la señal a generar.





**Figura 3.3**

Dado que el número de muestras es diferente para cada tono según su frecuencia, se debe determinar el valor del ángulo del cual se debe calcular el seno cada vez que se produzca una interrupción del Codec según la frecuencia del tono generado. A este ángulo se lo denomina avance

El valor del avance se obtiene dividiendo los 360° para el número de muestras que se alcancen a generar según la frecuencia del tono, pero como para el DSP los 360° grados corresponde a 65536 partes, el valor del avance se obtendrá aplicando la ecuación 3.2:

$$Avance = \frac{65536}{N_m}$$

**Ecuación 3.2**

Reemplazando la ecuación 3.1 en la ecuación 3.2, se obtiene :

$$Avance = 65536 \frac{f}{f_s}$$

**Ecuación 3.3**

La ecuación 3.3 permite calcular el valor del ángulo de avance para cualquier tono de frecuencia  $f$  que se desee generar en función de la frecuencia de muestreo  $f_s$  y de la frecuencia que se desee generar( $f$ ).

Dado que el Codec siempre trabaja con la frecuencia de muestreo de 48 KHz, la ecuación 3.3 queda reducida a :

$$\text{Avance} = 1.3653 f$$

#### **Ecuación 3.4**

La ecuación 3.4 permite calcular de manera directa el valor del avance únicamente en función de la frecuencia del tono a generar.

Frecuencia [Hz]	Número de Muestras	Avance [partes]
125	384	171
250	192	341
500	96	683
1000	48	1365
2000	24	2731
4000	12	5461
8000	6	10923

**Tabla 3.1**

La tabla 3.1 indica el número de muestras que constituyen cada uno de los tonos generados y además el valor del avance para cada caso.

### **3.1.2 ALGORITMO DE GENERACIÓN DE TONOS**

De manera general, para producir un tono de frecuencia  $f$ , se deben considerar 2 valores:

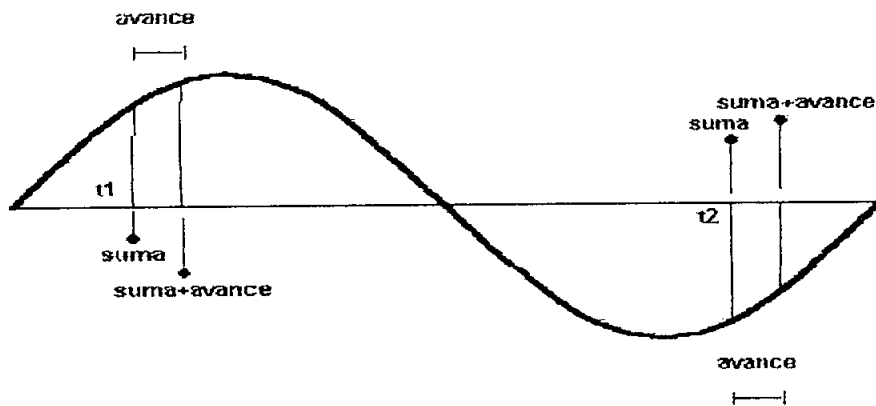
El primero corresponde al valor de la muestra actual, al cual se le llama suma.

El segundo corresponde al avance, el cual incrementa el valor para el cálculo de la siguiente muestra.

De esta manera, cada vez que se produce una interrupción del Codec, al valor de la suma se le incrementa el valor del avance.

Esto quiere decir que el avance corresponde a un incremento, el cual permite determinar la siguiente muestra mientras que el valor de la muestra actual o suma permite ubicar donde se encuentra la muestra en el eje del tiempo.

La figura 3.4 muestra de manera gráfica este proceso para 2 tiempos ( $t_1$  y  $t_2$ ) dentro de un ciclo de la señal generada



*Figura 3.4*

De manera más específica y detallada, para generar un tono de frecuencia  $f$ , se realiza el siguiente procedimiento:

1. En la primera interrupción serial del Codec, se hace lo siguiente:
  - Se carga el valor del avance correspondiente a la frecuencia  $f$  en el registro AY0 del DSP.
  - Se carga en el registro AX0 del DSP el valor 0.
  - Se guarda la suma de AX0 y AY0 en el registro AR del DSP
  - Se copia el valor de AR en la variable llamada  $c\_cntr$  (se almacena en la variable  $c\_cntr$  el valor suma)
  - Se calcula seno de AR, valor que corresponde a la primera muestra digital de la amplitud de la onda sinusoidal generada.
  - Se transmite al Codec el resultado del seno del ángulo ingresado.
2. En la siguiente interrupción del Codec, se hace lo siguiente:
  - Se carga el valor del avance correspondiente a la frecuencia  $f$  en el registro AY0 del DSP.
  - Se copia en el registro AX0 el valor guardado en la variable  $c\_cntr$  (que corresponde al valor de la muestra actual)
  - Se guarda la suma de AX0 y AY0 en el registro AR del DSP (que en este caso corresponde a dos avances)
  - Se copia el valor de AR en la variable llamada  $c\_cntr$  (se almacena en la variable  $c\_cntr$  el valor suma, es decir, dos avances)

- Se calcula seno de AR, valor que corresponde a la segunda muestra digital de la amplitud de la onda sinusoidal generada.
  - Se transmite al Codec el resultado del seno del ángulo ingresado.
3. En la siguiente interrupción del Codec, se hace lo siguiente:
- Se carga el valor del avance correspondiente a la frecuencia f en el registro AY0 del DSP.
  - Se copia en el registro AX0 el valor guardado en la variable c\_cntr (que corresponde al valor de la muestra actual)
  - Se guarda la suma de AX0 y AY0 en el registro AR del DSP (que en este caso corresponde a tres avances)
  - Se copia el valor de AR en la variable llamada c\_cntr (se almacena en la variable c\_cntr el valor suma, es decir, tres avances)
  - Se calcula seno de AR, valor que corresponde a la tercera muestra digital de la amplitud de la onda sinusoidal generada.
  - Se transmite al Codec el resultado del seno del ángulo ingresado.

Este proceso se repite hasta que se complete un ciclo de la senoide y luego se empieza nuevamente desde cero, ya que una vez que el valor guardado en la variable c\_cntr sobrepase los 65536, este toma el valor de cero.

En el algoritmo descrito, es fácil notar que cada vez que se produce una interrupción del Codec, el valor del avance se carga en el registro AY0, mientras que la variable c\_cntr es la encargada de dar el valor de la muestra actual, es decir, cumple el rol del valor suma.

### 3.1.3 CALCULO DEL SENO

Para obtener el valor del seno, se utilizará una rutina de cálculo del mismo, la cual es muy eficiente, ya que se ejecuta en 25 ciclos de máquina, lo que resulta más conveniente que utilizar tablas de búsqueda.

La rutina del cálculo del seno usa la siguiente expansión polinómica de Taylor de quinto orden <sup>8</sup>:

$$\text{sen}(x) = 3.140625x + 0.02026367x^2 - 5.325196x^3 + 0.5446778x^4 + 1.800293x^5$$

---

<sup>8</sup> ANALOG DEVICES." ADSP-2100 Family User's Manual". 3ra Edición. Editorial Computer Products División. Canadá. 1995. Capítulo 14

Es importante mencionar que el valor del ángulo del cual se quiere calcular el seno debe ser ingresado en formato 1.15. En dicho formato , el incremento de un LSB , corresponde a un incremento de 0.000030517578125 en decimal.

Este es un requerimiento obligatorio y es resultado del tratamiento que se ha dado al polinomio de Taylor.

Así, para calcular el seno de un ángulo en grados, se debe primeramente hallar a qué valor corresponde dicho ángulo para el DSP. Esto se realiza con la ecuación 3.5:

$$\text{ángulo}(DSP) = \text{ángulo}(grados) * \frac{65536}{360} = \text{ángulo}(grados) * 182.0444$$

### ***Ecuación 3.5***

Luego, se multiplica el valor obtenido de la ecuación por el valor de 1 bit en el formato 1.15 y se obtiene el valor de X que será usado en la expansión de Taylor. El resultado es entregado en número decimal normal.

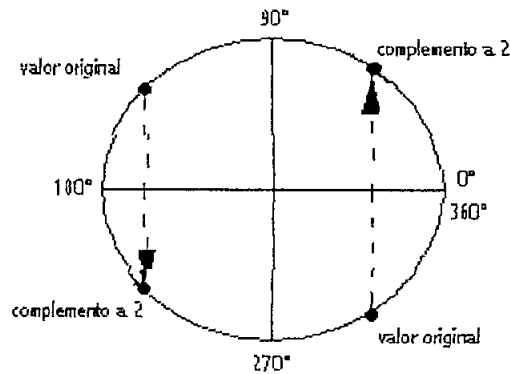
Todo lo mencionado corresponde únicamente a un análisis matemático para explicar el comportamiento de la expansión de Taylor usada, ya que en la programación misma del DSP, basta con cargar en el registro AX0 el valor de la suma más el avance y el dispositivo se encarga de hacer toda la traducción explicada y da como resultado el valor del seno del ángulo ingresado, y este resultado se carga en el registro AR.

Es importante mencionar que tanto el registro AX0 como el AR son de 16 bits, así que el seno es calculado con 16 bits significativos.

La expansión de Taylor que se usa trabaja de manera adecuada únicamente para el primer cuadrante, pero como el ángulo ingresado está en el rango de 0° a 360 °, lo primero que debe hacer el programa del cálculo del seno es modificar el ángulo ingresado para que este se ubique en el primer cuadrante. En vista de que  $\text{sen}(-x) = -\text{sen}(x)$  y  $\text{sen}(x) = \text{sen}(180^\circ - x)$ , calculando el seno del ángulo en el primer cuadrante, se puede deducir el seno de cualquier ángulo.

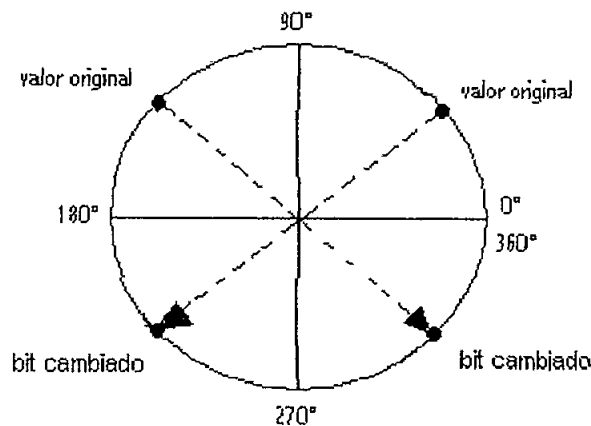
Para hacer la reubicación del ángulo ingresado y ponerlo en el primer cuadrante, se usan dos herramientas básicas:

- La primera consiste en hallar el complemento a dos del valor ingresado, lo cual equivale a un movimiento vertical en el círculo unitario. La figura 3.5 indica de manera explícita este procedimiento:



**Figura 3.5**

- La segunda es cambiar el primer bit del valor ingresado, lo cual equivale a un movimiento diagonal en el círculo unitario. La figura 3.6 indica de manera explícita este procedimiento:



**Figura 3.6**

Así, el algoritmo que sigue la subrutina del cálculo del seno es la siguiente:

- Si el valor del ángulo ingresado está en el primer cuadrante, no se modifica su valor y se calcula el seno de manera directa.
- Si el valor del ángulo está en el segundo cuadrante, se obtiene el complemento a 2 de dicho valor (lo cual lo cambia al tercer cuadrante),

luego se cambia el primer bit del valor obtenido (lo cual lo cambia al primer cuadrante) y luego se calcula el seno.

- Si el valor del ángulo ingresado está en el tercer cuadrante, se cambia el primer bit del valor ingresado (lo cual lo cambia al primer cuadrante), se calcula el seno y finalmente se pone el signo negativo al valor obtenido.
- Si el valor del ángulo ingresado está en el cuarto cuadrante, se obtiene el complemento a dos de dicho valor (lo cual lo cambia al primer cuadrante) , se calcula el seno y finalmente se pone el signo negativo al valor obtenido.

La implementación de esta subrutina, se ve explícitamente en el análisis del programa implementado en el ADSP-281<sup>9</sup>

## 3.2 FILTRO DIGITAL

### 3.2.1 PASOS A SEGUIR PARA EL DISEÑO

Dado que los sonidos a generar se constituyen únicamente por una componente de frecuencia, no se puede hablar de una distorsión de fase, así que lo que interesa es realizar una compensación para la distorsión de amplitud. Esto se consigue de manera óptima mediante el uso de un filtro IIR.

El rol principal que cumple el filtro es el de no permitir el ingreso de frecuencias no deseadas dentro de la banda de evaluación (0 a 8000 HZ) y de acentuar las componentes de frecuencia generadas en dicha banda.

Para implementar filtros IIR, se diseña un filtro analógico que cumpla con los requerimientos necesarios para obtener el fin que se persigue y luego se lo digitaliza usando un procedimiento adecuado.

Para diseñar un filtro IIR, se siguen los siguientes pasos:

- Se determinan los parámetros de :
  - Frecuencia de paso
  - Frecuencia de corte
  - Frecuencia de muestreo

---

<sup>9</sup> Ver Capítulo III, numeral 3.10

- Atenuación en la frecuencia de paso
- Atenuación en la frecuencia de corte
- Se normaliza las frecuencias (convertir frecuencias analógicas a formato digital)
- Se diseña el filtro analógico
- Se digitaliza el filtro usando un procedimiento adecuado

### 3.2.2 FRECUENCIAS A UTILIZAR

#### 3.2.2.1 Frecuencia de paso

Es el límite superior de la banda de paso, banda en la cual se acepta todas las componentes de frecuencias que ingresen.

Es además el límite inferior de la banda de transición y de aquí en adelante se la denominará  $\Omega_p$

#### 3.2.2.2 Frecuencia de corte

Es el límite inferior de la banda de corte, banda en la cual se rechaza o se atenúa fuertemente todas las componentes de frecuencias que ingresen.

Es además el límite superior de la banda de transición y de aquí en adelante se la denominará  $\Omega_s$

#### 3.2.2.3 Frecuencia de Muestreo

Es la frecuencia de muestreo del filtro

### 3.2.3 ATENUACIONES A UTILIZAR

#### 3.2.3.1 Atenuación en la frecuencia de paso

Es la atenuación que tendrán todas las componentes de frecuencia que se hallen dentro de la banda de paso. De aquí en adelante se la denominará  $att_{\omega_p}$ .



### 3.2.3.2 Atenuación en la frecuencia de corte

Es la atenuación que tendrán todas las componentes de frecuencia que se hallen dentro de la banda de corte. De aquí en adelante se la denominará  $att_{\omega_s}$ .

### 3.2.4 NORMALIZACION DE FRECUENCIAS

Este procedimiento consiste en normalizar la frecuencias antes explicadas con respecto a la frecuencia de muestreo del filtro.

Es importante mencionar que a las frecuencias analógicas se las denomina utilizando la letra  $\Omega$  mientras que a las digitales se las denomina con la letra  $\omega$ .

Al normalizar las frecuencias, se dice que éstas han sido digitalizadas, y de aquí en adelante, se trabajará solo con frecuencias digitales.

Hablando de frecuencias normalizadas, el máximo valor que puede alcanzar una frecuencia es de  $2\pi$ , y este valor corresponde justamente a la frecuencia de muestreo.

Así, pues para determinar el valor normalizado que le corresponde a cada frecuencia, se aplica la ecuación 3.6 :

$$\omega_n = \frac{\Omega_n}{\Omega} 2\pi$$

**Ecuación 3.6**

Donde:

$\omega_n$  = frecuencia digital

$\Omega_n$  = frecuencia analógica

$\Omega$  = frecuencia de muestreo

Las frecuencias digitales se expresan como funciones de  $\pi$ .

### 3.2.5 DISEÑO DEL FILTRO ANALÓGICO

Se ha escogido diseñar un filtro pasa bajos de Chebyshev I, ya que este es altamente selectivo para un orden relativamente bajo.

La función de transferencia característica de este filtro, la cual se grafica en el plano S ( por estar trabajando en el dominio analógico), es la siguiente:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \varepsilon^2 V_N^2(\Omega / \Omega_p)}$$

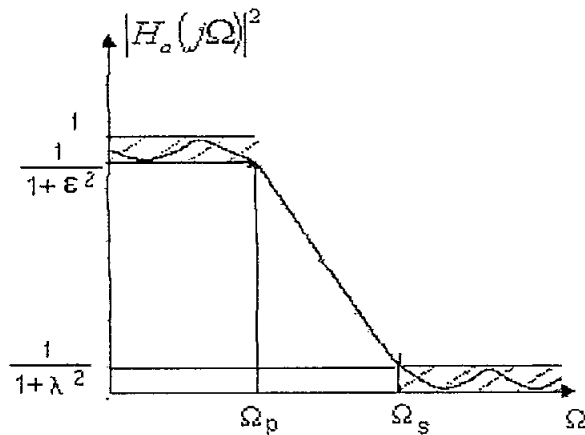
Donde:

$V_N(x)$  es el polinomio de Chebyshev de orden N

$$V_N(x) = \cos(N \arccos x)$$

Otros dos parámetros importantes que se deben definir son  $\varepsilon$  y  $\lambda$ .

Estos se refieren al rizado que existe en la banda de paso y en la banda de corte respectivamente. La siguiente figura muestra como intervienen estos parámetros en el filtro que se diseñe.



Una vez determinadas las frecuencias de paso, de corte, muestreo y atenuaciones, se calcula  $\varepsilon$  y  $\lambda$  utilizando las ecuaciones 3.7 y 3.8 respectivamente:

$$\varepsilon = \sqrt{10^{\frac{att \ \omega_p}{10}} - 1}$$

**Ecuación 3.7**

$$\lambda = \sqrt{10^{\frac{att \omega s}{10}} - 1}$$

**Ecuación 3.8**

Una vez obtenidos estos dos valores, se calcula el orden del filtro con la ecuación 3.9:

$$N \geq \frac{\cosh^{-1}\left(\frac{\lambda}{\varepsilon}\right)}{\cosh^{-1}\left(\frac{\Omega_s}{\Omega_p}\right)}$$

**Ecuación 3.9**

N debe ser necesariamente un entero, así que en caso de que la división no sea exacta, se lo aproxima al inmediato superior.

Una vez calculados estos valores, se procede a hallar los polos de la función de transferencia, cuyo número es el mismo que el valor de N.

Antes de calcular los polos, se deben calcular dos valores, los cuales intervienen en la ecuación que se usa para hallar los polos.

Estos valores son  $\alpha$  y  $\beta_i$ , y vienen dados por las ecuaciones 3.10 y 3.11 respectivamente.

$$\alpha = \frac{1}{N} \sinh^{-1}\left(\frac{1}{\varepsilon}\right)$$

**Ecuación 3.10**

$$\beta_i = \frac{2i + N - 1}{2N} \pi \quad \text{con } i = 1, 2, 3, \dots, 2N$$

**Ecuación 3.11**

Una vez obtenidos estos valores, cada polo se obtiene mediante la ecuación 3.12:

$$S_i = \omega_p' (\sinh(\alpha) \cos(\beta_i) + j \cosh(\alpha) \sin(\beta_i))$$

**Ecuación 3.12**

Donde:

$S_i$  = polo a calcular

$\omega_p'$  = frecuencia de paso digitalizada

El número de polos obtenidos, según  $\beta_i$  corresponde a  $2N$ . Sin embargo, para garantizar la estabilidad del filtro, es necesario trabajar únicamente con los polos del semiplano  $S$  izquierdo, es decir, aquellos polos cuya parte real es negativa.

De esta manera, el número de polos queda reducido a  $N$ .

Ya hallados los polos, es posible calcular la función de transferencia del filtro analógico, la cual toma la siguiente forma:

$$H_a(s) = \frac{k}{(s - s_1)(s - s_2)(s - s_3) \dots (s - s_N)}$$

Para determinar el valor de  $k$ , se deben considerar si  $N$  es par o impar.

En caso de que  $N$  sea par, el valor de  $k$  se calcula con la ecuación 3.13:

$$H_a(0) = 10^{-\frac{att \omega_p}{20}}$$

**Ecuación 3.13**

En caso de que  $N$  sea impar, el valor de  $k$  se calcula con la ecuación 3.14:

$$H_a(0) = 1$$

**Ecuación 3.14**

Con todos estos valores, es posible calcular la función de transferencia del filtro analógico.

### 3.2.6 DIGITALIZACIÓN DEL FILTRO ANALÓGICO

Una vez obtenida la función de transferencia del filtro analógico, se lo debe digitalizar. Para esto se utiliza el método de la transformada bi lineal.

Cabe mencionar que esta no es la única técnica de digitalización, pero si es la más adecuada para los fines que se persiguen, ya que es ésta la que permite la más simple implementación en el DSP.

#### 3.2.6.1 La Transformada Bi Lineal

Es la Transformación dada por el algoritmo trapezoidal

$$z \rightarrow \frac{2 + st_s}{2 - st_s} \Rightarrow s \rightarrow \frac{2}{ts} \frac{z - 1}{z + 1}$$

**Ecuación 3.15**

Se toma  $t_s = 2$ , para simplificar las expresiones. Se sustituye  $s$  por  $j\omega$  en la expresión anterior :

$$z = \frac{2 + j\omega}{2 - j\omega}$$

**Ecuación 3.16**

Esta expresión dice que un punto del eje imaginario  $s$ , se transforma en un punto en el círculo unitario en el plano  $z$ .

Generalizando, para  $s = \sigma + j\omega$ , se obtiene que:

$$z = \frac{1 + (\sigma + j\omega)}{1 - (\sigma + j\omega)} = \frac{(1 + \sigma) + j\omega}{(1 - \sigma) - j\omega} \Rightarrow |z|^2 = \frac{(1 + \sigma)^2 + \omega^2}{(1 - \sigma)^2 + \omega^2}$$

**Ecuación 3.17**

Lo que implica que un punto de la mitad del plano  $s$  donde  $s < 0$ , se transforma en un punto dentro del círculo unidad en el plano  $z$ . Por lo tanto, un diseño estable en  $s$ , será también estable en  $z$  bajo la transformación bilineal. Además, no hay aliasing, ya que a cada frecuencia analógica le corresponde una única frecuencia digital.

Interesa la relación entre los puntos del eje imaginario  $s$  con los puntos del círculo unitario en el plano  $z$ . Para esto, se tiene la ecuación 3.18 :

$$j\Omega = \frac{2}{ts} \frac{e^{j\omega ts} - 1}{e^{j\omega ts} + 1} = j \frac{2}{ts} \tan \left( \frac{\omega ts}{2} \right)$$

**Ecuación 3.18**

De esta relación se puede también obtener otra entre la frecuencia digital ( $\omega$ ) y la frecuencia analógica ( $\Omega$ ). Esta relación vienen dada por la ecuación 3.19:

$$\Omega = \frac{2}{ts} \tan \left( \frac{\omega ts}{2} \right) \Rightarrow \omega = \frac{2}{ts} \tan^{-1} \left( \frac{\Omega ts}{2} \right)$$

**Ecuación 3.19**

Estas expresiones indican que existe una distorsión en las frecuencias cuando se hace la transformación bilineal. El rango entero de frecuencias analógicas se muestrea con frecuencias digitales entre  $-fs/2$  y  $fs/2$ . Hay una compresión de frecuencias o una distorsión frecuencial (la relación es no-lineal).

Para compensarlo, lo que se hace es predistorsionar (prewarping) las especificaciones del filtro analógico original mediante la ecuación 3.20:

$$\Omega' = \frac{2}{ts} \tan \left( \frac{\omega ts}{2} \right) = \frac{2}{ts} \tan \left( \frac{\omega}{2 fs} \right) = \frac{2}{ts} \tan \left( \frac{\Omega}{2} \right)$$

**Ecuación 3.20**

Donde  $ts$  es el período de la señal de muestreo y  $\Omega'$  es la frecuencia analógica predistorsionada.

Al realizar los cálculos de diseño del filtro, se podrá observar que las operaciones son transparentes al valor tomado para  $f_s$ , de forma que se puede utilizar la ecuación 3.21 para predistorsionar :

$$\omega' = 2 \tan \left( \frac{\omega}{2} \right)$$

**Ecuación 3.21**

Donde  $\omega'$  es la frecuencia digital predistorsionada.

Esta predistorsión se debe realizar al iniciar el diseño del filtro analógico, una vez que ya se han normalizado las frecuencias a usar.

Cuando ha sido hallada la función de transferencia del filtro analógico, se evalúa dicha función en

$$s = 2 \frac{1 - z^{-1}}{1 + z^{-1}}$$

Y se obtiene como resultado la función de transferencia del filtro digital diseñado.

### 3.2.7 DISEÑO DE FILTRO IIR, CHEBYSHEV TIPO I, PASABAJO Y DIGITALIZADO CON LA TÉCNICA DE LA TRANSFORMADA BI LINEAL.

#### 3.2.7.1 Determinación de las frecuencias a utilizar

$\Omega_p = 8000\text{Hz}$  (frecuencia de paso)

$\Omega_s = 12000\text{Hz}$  (frecuencia de corte)

$\Omega = 48000\text{Hz}$  (frecuencia de muestreo)

$\text{att}_{\omega_p} = 1 \text{ dB}$  (atenuación en la frecuencia de paso)

$\text{att}_{\omega_s} = 20 \text{ dB}$  (atenuación en la frecuencia de corte)

Se han escogido estos valores ya que el filtro no debe permitir el ingreso de frecuencias no deseadas dentro de la banda de evaluación (0 a 8000 Hz) y de

acentuar las componentes de frecuencia generadas en dicha banda, especialmente las bajas frecuencias, por razones de ponderación.<sup>10</sup>

La exigencia del filtro no es muy alta, especialmente en el aspecto de la banda de transición ( $\Omega_s - \Omega_p$ ), ya que las señales, al ser generadas digitalmente, son bastante exactas.

### 3.2.7.2 Normalización de Frecuencias

Se debe normalizar tanto la frecuencia de corte como la de paso con respecto a la frecuencia de muestreo. Para conseguir esto, se aplica la ecuación 3.6 :

$$\omega_n = \frac{\Omega_n}{\Omega} 2\pi$$

Para la frecuencia de paso se tiene:

$$\omega_p = \frac{\Omega_p}{\Omega} 2\pi = \frac{8000}{48000} 2\pi = 0.333 \pi$$

Para la frecuencia de corte se tiene:

$$\omega_s = \frac{\Omega_s}{\Omega} 2\pi = \frac{12000}{48000} 2\pi = 0.5 \pi$$

Por lo tanto, las frecuencias digitalizadas son:

$$\omega_p = 0.33 \pi \text{ [rad/s] ( frecuencia digital de paso )}$$

$$\omega_s = 0.5 \pi \text{ [rad/s] (frecuencia digital de corte)}$$

### 3.2.7.3 Predistorsión o prealabeo de las frecuencias digitales

Se debe predistorsionar tanto la frecuencia de corte como la de paso. Para conseguir esto, se aplica la ecuación 3.21 :

$$\omega^1 = 2 \tan \left( \frac{\omega}{2} \right)$$

---

<sup>10</sup> Ver Capítulo II, numeral 2.1.4.1



Para la frecuencia de paso se tiene:

$$\omega_p^{-1} = 2 \tan \left( \frac{\omega_p}{2} \right) = 2 \tan \left( \frac{0.33 \pi}{2} \right) = 1.155$$

Para la frecuencia de corte se tiene:

$$\omega_s^{-1} = 2 \tan \left( \frac{\omega_s}{2} \right) = 2 \tan \left( \frac{0.5 \pi}{2} \right) = 2$$

Por lo tanto, las frecuencias digitalizadas predistorsionadas son:

$$\omega_p^{-1} = 1.155 \text{ [rad/s] (frecuencia digital de paso predistorsionada)}$$

$$\omega_s^{-1} = 2 \text{ [rad/s] (frecuencia digital de corte predistorsionada)}$$

#### 3.2.7.4 Cálculo de $\varepsilon$ , $\lambda$ y $N$

$\varepsilon$  y  $\lambda$  se calculan utilizando las ecuaciones 3.7 y 3.8 respectivamente:

$$\varepsilon = \sqrt{10^{\frac{att \omega_p}{10}} - 1} = \sqrt{10^{0.1} - 1} = 0.509$$

$$\lambda = \sqrt{10^{\frac{att \omega_s}{10}} - 1} = \sqrt{10^2 - 1} = 9.949$$

Por lo tanto:

$$\varepsilon = 0.509$$

$$\lambda = 9.949$$

Una vez obtenidos estos dos valores, se calcula el orden del filtro con la ecuación 3.9:

$$N \geq \frac{\cosh^{-1} \left( \frac{\lambda}{\varepsilon} \right)}{\cosh^{-1} \left( \frac{\Omega_s}{\Omega_p} \right)} = \frac{\cosh^{-1} \left( \frac{9.949}{0.509} \right)}{\cosh^{-1} \left( \frac{2}{1.155} \right)} = 3.199$$

N debe ser necesariamente un entero por lo que se lo aproxima al inmediato superior.

Por lo tanto:

$$\varepsilon = 0.509$$

$$\lambda = 9.949$$

$$N = 4$$

### 3.2.7.5 Cálculo de $\alpha$ y $\beta_i$

Par calcular  $\alpha$  y  $\beta_i$  se utilizan las ecuaciones 3.10 y 3.11 respectivamente:

$$\alpha = \frac{1}{N} \sinh^{-1} \left( \frac{1}{\varepsilon} \right) = \frac{1}{4} \sinh^{-1} \left( \frac{1}{0.509} \right) = 0.357$$

$$\beta_i = \frac{2i + N - 1}{2N} \pi = \frac{2i + 3}{8} \pi$$

Dado que el  $N = 4$ ,  $\beta$  toma 8 valores, que son los siguientes:

$$\beta_1 = 5/8 \pi, \beta_2 = 7/8 \pi, \beta_3 = 9/8 \pi, \beta_4 = 11/8 \pi, \beta_5 = 13/8 \pi, \beta_6 = 15/8 \pi,$$

$$\beta_7 = 17/8 \pi, \beta_8 = 19/8 \pi$$

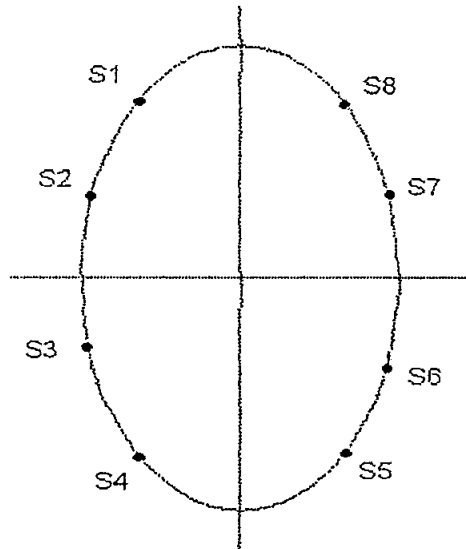
### 3.2.7.6 Cálculo de Polos

Calculados  $\alpha$  y  $\beta$ , se obtiene cada polo con la ecuación 3.12:

$$S_i = \omega_p' (\sinh(\alpha) \cos(\beta_i) + j \cosh(\alpha) \sin(\beta_i))$$

Donde  $\omega_p' = 1.155$  [rad/s]

La siguiente figura indica la ubicación de los polos que deben ser calculados:



Los polos hallados son:

$$S_1 = -0.161 + j 1.136$$

$$S_2 = -0.388 + j 0.47$$

$$S_3 = -0.388 - j 0.47$$

$$S_4 = -0.161 - j 1.136$$

$$S_5 = 0.161 - j 1.136$$

$$S_6 = 0.388 - j 0.47$$

$$S_7 = 0.388 + j 0.47$$

$$S_8 = 0.161 + j 1.136$$

Para garantizar estabilidad, se toma los polos del semiplano izquierdo únicamente (es decir, los que tienen signo negativo en su parte real). Por lo tanto, los polos del filtro analógico son:

$$S_1 = -0.161 + j 1.136$$

$$S_2 = -0.161 - j 1.136$$

$$S_3 = -0.388 + j 0.47$$

$$S_4 = -0.388 - j 0.47$$

### 3.2.7.7 Cálculo de la Función de Transferencia del Filtro Analógico H(s)

H(s) toma la siguiente forma.

$$H_a(s) = \frac{k}{(s - s_1)(s - s_2)(s - s_3)(s - s_4)}$$

Es necesario determinar el denominador de la función de transferencia  $H_N(s)$ , lo cual resulta muy simple dado que ya se conocen los polos de la misma.

Realizando las operaciones algebraicas necesarias, se obtiene el denominador de H(s), el cual es el siguiente:

$$H_N(s) = (s^2 + 0.323s + 1.317)(s^2 + 0.776s + 0.371)$$

Ahora se requiere determinar el valor de k.

Dado que N es par, para el cálculo de k se debe aplicar la ecuación 3.13:

$$H_a(0) = 10^{-\frac{1}{20}} = \frac{k}{0.489} \rightarrow k = 10^{-\frac{1}{20}} (0.489)$$

Por lo tanto:

$$k = 0.436$$

Finalmente, la función de transferencia del filtro analógico es:

$$H_a(s) = \frac{0.436}{(s^2 + 0.323s + 1.317)(s^2 + 0.776s + 0.371)}$$

### 3.2.7.8 Digitalización del Filtro Analógico

Para digitalizar el usa el método de la transformada bilineal.

Para que la función de transferencia del filtro analógico (H(s)) se transforme en una función de transferencia en el dominio discreto (H(z)), se la debe evaluar en:

$$s = 2 \frac{1 - z^{-1}}{1 + z^{-1}}$$

Por lo tanto:

$$H(z) = H(s) \Big|_{s=2\frac{1-z^{-1}}{1+z^{-1}}}$$

Realizando todas las operaciones algebraicas necesarias, se obtiene la función de transferencia del filtro digital que será implementado:

$$H(z) = \frac{(0.012 + 0.0247 z^{-1} + 0.012 z^{-2})(1 + 2z^{-1} + z^{-2})}{(1 - 0.9z^{-1} + 0.784 z^{-2})(1 - 1.225 z^{-1} + 0.476 z^{-2})}$$

Cuando se abarque el tópico referente a la implementación de este filtro, se explicará la razón por la cual no se han reducido todos los términos semejantes de la ecuación de transferencia.<sup>11</sup>

### 3.3 IMPLEMENTACIÓN DEL FILTRO DIGITAL EN EL ADSP-2181

La función de transferencia del filtro es:

$$H(z) = \frac{(0.012 + 0.0247 z^{-1} + 0.012 z^{-2})(1 + 2z^{-1} + z^{-2})}{(1 - 0.9z^{-1} + 0.784 z^{-2})(1 - 1.225 z^{-1} + 0.476 z^{-2})}$$

Para implementar un filtro digital IIR en el ADSP-2181, se requiere que la función de transferencia del mismo esté expresada como el producto de varias secciones de segundo orden.

De esta manera, se puede expresar un filtro de un orden alto poniendo en cascada varias secciones de segundo orden, únicamente especificando los coeficientes adecuados. Con esto se consigue minimizar la cuantificación de los coeficientes y el error de acumulación recursiva.

Así pues, de acuerdo a la función de transferencia obtenida, se nota que las dos secciones de segundo orden del filtro vienen dadas por las ecuaciones 3.22 y 3.23 respectivamente:

$$H^1(z) = \frac{(0.012 + 0.0247 z^{-1} + 0.012 z^{-2})}{(1 - 0.9z^{-1} + 0.784 z^{-2})}$$

**Ecuación 3.22**

---

<sup>11</sup> Ver Capítulo III, numeral 3.3

$$H^2(z) = \frac{(1 + 2z^{-1} + z^{-2})}{(1 - 1.225z^{-1} + 0.476z^{-2})}$$

**Ecuación 3.23**

Y por consecuencia, se tiene que la función de transferencia completa del filtro viene dada por la ecuación 3.24:

$$H(z) = H^1(z)H^2(z)$$

**Ecuación 3.24**

Cada una de estas secciones, se ajusta a la forma indicada por la ecuación 3.25:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{(B_0 + B_1z^{-1} + B_2z^{-2})}{(1 - A_1z^{-1} + A_2z^{-2})}$$

**Ecuación 3.25**

A dicha función de transferencia, le corresponde una ecuación de diferencias de la forma indicada por la ecuación 3.26:

$$Y(n) = B_0X(n) + B_1X(n-1) + B_2X(n-2) - A_1Y(n-1) + A_2Y(n-2)$$

**Ecuación 3.26**

Para implementar el filtro, los coeficientes deben ser ingresados en formato 2.14<sup>12</sup> y además deben estar ordenados de la siguiente forma:

$B_2, B_1, B_0, A_2$  y  $A_1$  para cada sección.

Así, los coeficiente deben ser ingresados en el siguiente orden:

- $B_2^i = 0.012$
- $B_1^i = 0.0247$
- $B_0^i = 0.012$

---

<sup>12</sup> Ver Capítulo II, Tabla 2

- $A_2^i = 0.784$
- $A_1^i = 0.9$
- $B_2^{ii} = 1$
- $B_1^{ii} = 2$
- $B_0^{ii} = 1$
- $A_2^{ii} = 0.476$
- $A_1^{ii} = -1.225$

Para hacer la transformación a formato 2.14, se debe recordar que el incremento de un LSB corresponde a un incremento de 0.00006103515625 en decimal, así que para hallar el número de bits requerido para cada coeficiente, aplicamos la ecuación 3.27:

$$\#bits = \text{coeficiente decimal} / 0.00006103515625$$

***Ecuación 3.27***

Ya hallado el número de bits para cada coeficiente, se debe transformar este número de bits a hexadecimal, y con esto se halla el valor correspondiente a cada coeficiente. Este es el valor que se debe ingresar al ADSP-2181 para que se ejecute la subrutina del filtro digital IIR requerido.

Los coeficientes en hexadecimal son:

- $B_2^i = 00CB \text{ H}$
- $B_1^i = 0195 \text{ H}$
- $B_0^i = 00CB \text{ H}$
- $A_2^i = 322B \text{ H}$
- $A_1^i = C660 \text{ H}$
- $B_2^{ii} = 4000 \text{ H}$
- $B_1^{ii} = 7FFF \text{ H,}$
- $B_0^{ii} = 4000 \text{ H}$
- $A_2^{ii} = 1E75 \text{ H}$
- $A_1^{ii} = B19C \text{ H}$

De esta manera quedan definidos los valores requeridos por la subrutina del filtro.

La implementación de esta subrutina, se ve explícitamente en el análisis del programa implementado en el ADSP-2181 <sup>13</sup>

### 3.4 CONTROL DE VOLUMEN

El Codec AD-1847, permite trabajar de una manera sencilla y relativamente fácil con atenuadores programables, asignando el valor de la atenuación necesaria en un registro específico del ADSP -2181.

Este registro es el (tx\_buf).

El rango de variación de la atenuación es de 0 a 94.5 dB y si se trabaja en modo típico (que es lo más fácil y conveniente), se puede variar la intensidad en saltos de 1.5 dB.

Para realizar un cambio de intensidad, se guarda el valor de la intensidad que se requiere asignar al tono generado en un registro cualquiera (en general se usa una etiqueta variable), y se realiza un OR lógico entre dicho valor en hexadecimal y 8600H o 8700H, según se requiera variar el volumen del canal derecho o izquierdo respectivamente.

Para explicar mejor este procedimiento, se cita a continuación un ejemplo:

Si suponemos que atenuacionR es variable asignada para controlar la ganancia del canal derecho, y que se requiere tener una intensidad de 15 dB (cantidad que se encuentra almacenada previamente en el registro ar), se hace lo siguiente:

- Se graba en ar el valor en sistema decimal de la atenuación que se requiere
- Se graba en atenuacionR el valor 10 en sistema decimal
- Se graba en ax0 el valor 10 en sistema decimal, lo cual implica 15 dB
- Se graba en ay0 el valor en sistema hexadecimal que requiero para variar la intensidad del oído derecho
- Se realiza la operación lógica entre los valores requeridos
- Se asigna al registro que controla el volumen, el valor de atenuación deseada

---

<sup>13</sup> Ver Capítulo III, numeral 3.10



Si se programa en lenguaje assembler, es procedimiento para asignar una intensidad a un tono generado, es el siguiente:

```
ar=10;
dm(atenuacionR)= ar;
ax0 = dm(atenuacionR);
ay0 = 0x8600;
ar = ax0 or ay0;
dm(tx_buf) = ar;
```

Al utilizar este procedimiento, el valor de la atenuación se graba en la variable `atenuacionR`, dejando de esta manera libres los registros `ar` y `ax0` para ser utilizados en otras operaciones.

El Codec únicamente tiene atenuadores programables, mas no amplificadores. Así que no se debe olvidar que si se asigna al registro `tx_buf` el valor de 0, se tendrá el valor de intensidad máximo, ya que la atenuación es mínima ;y , que si se asigna al registro `tx_buf` el valor de 64, se tendrá el valor de intensidad mínimo, ya que la atenuación es máxima.

Cada variación de una unidad en sistema decimal en la variable de control de volumen, implica un salto de 1.5 dB. Es por esta razón que la máxima atenuación se obtiene cuando en dicha variable se carga el valor 64 (ya que  $64 * 1,5 = 94,5$ ).

La implementación de este procedimiento, se ve explícitamente en el análisis del programa implementado en el ADSP-281<sup>14</sup>

### 3.5 CALIBRACIÓN DE VOLUMEN

La calibración del dispositivo se debe realizar en base al tono de 1000 Hz , ya que en audiología, se lo toma como frecuencia referencial<sup>15</sup>.

La calibración se realizó de manera empírica.

Para esto, el investigador se sometió a la evaluación audiométrica<sup>16</sup>, y se obtuvo que el tono de 1000 Hz se escuchó a 17 dB. Cabe mencionar que el

---

<sup>14</sup> Ver Capítulo III, numeral 3.10

<sup>15</sup> Ver Capítulo II, numeral 2.1.4.1

<sup>16</sup> Examen Realizado en la Fundación Vista Para Todos

examen fue realizado con un audiómetro que se encuentra en el mercado y cuyas características específicas serán detalladas en el análisis de resultados en el capítulo IV.

Ya se mencionó que El Codec AD-1847, permite trabajar con atenuadores programables, consiguiendo un rango de variación de la atenuación es de 0 a 94.5 dB en saltos de 1.5 dB.

Esta variación produce un voltaje máximo de  $1.2 V_{pp}$ .

Este nivel de voltaje no permitió que el evaluado consiga escuchar el tono a la intensidad requerida (17 dB) utilizando sólo audífonos, así que fue necesario recurrir a la ayuda de un amplificador.

La ganancia de dicho amplificador es constante ya que la variación de la intensidad ya está controlada por el Codec .

Para implementar dicho amplificador, se recurrió a la ayuda del circuito integrado TDA2822M, el mismo que es un amplificador de potencia dual de baja señal.

La figura 3.7 muestra el circuito del amplificador de ganancia constante implementado:

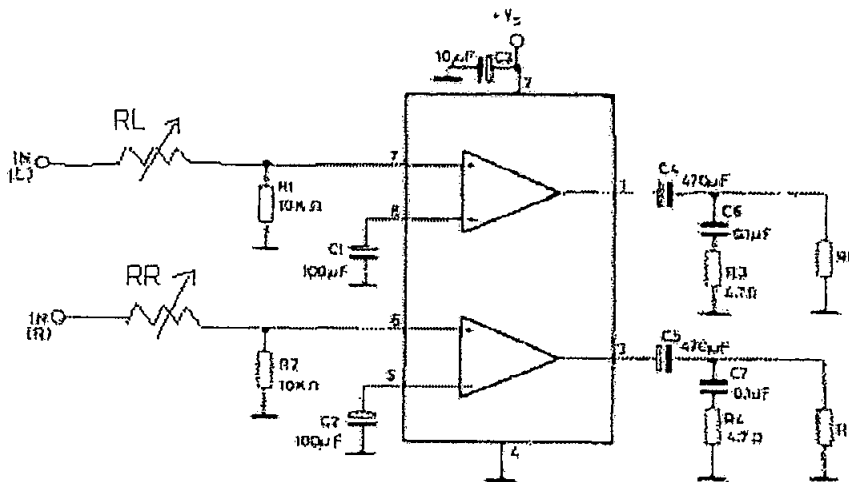
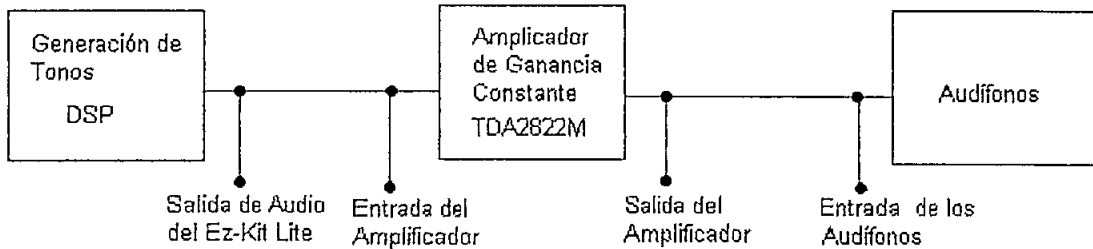


Figura 3.7

Este circuito permite controlar de manera independiente la ganancia de cada canal (izquierdo o derecho) variando los valores de RL o RR respectivamente. Sin embargo, en el caso particular del AMCAEC I, se requiere que la ganancia sea la misma para ambos canales, así que se ha usado un potenciómetro de doble lado.

Este amplificador fue tomado del Datasheet del integrado en cuestión, y esta información consta de manera explícita en el anexo 10.

La figura 3.8 muestra, en forma de diagrama de bloques, la presencia de este amplificador:



**Figura 3.8**

Así, se conectó la salida de audio de la tarjeta Ez-Kit Lite a la entrada del amplificador de ganancia constante y a la salida de éste se conectaron los audífonos.

Una vez hecho esto, se realizó la evaluación audiométrica a tres pacientes de características similares (edad y estado de salud auditiva) tanto con el AMCAEC I como con un equipo comercial y se fue variando el potenciómetro hasta que los resultados de las evaluaciones con ambos dispositivos fueran lo más similares posibles.

De esta manera, quedó determinada empíricamente la ganancia del amplificador.

Tal como se explica en el capítulo II, el AMCAEC I trabaja con dB y no con dBA, así que con el procedimiento anterior, quedó también calibrado el audiómetro implementado.

### 3.6 SELECCIÓN DE CANAL PARA LA SEÑAL DE SALIDA

El Codec AD-1847, dispone de 2 canales de salida, cada uno de los cuales corresponde a un registro específico del ADSP -2181.

Estos registros son el (tx\_buf+1) y (tx\_buf+2), y en el caso del audímetro diseñado, corresponden al audífono derecho e izquierdo respectivamente.

Así pues, en el audífono derecho se escucha el valor que se asigne al registro (tx\_buf+1) mientras que en el oído izquierdo se escucha el valor que se le asigne al registro (tx\_buf+2).

Esto es de suma importancia, ya que, tal como se explicó en el capítulo anterior, cada oído debe ser evaluado de manera independiente, y con esta opción que brinda el Codec AD-1847, se consigue esto sin necesidad de poner interruptores en los conductores de cada audífono.

De esta manera, se puede elegir el oído a evaluar mediante software, brindando así mayor facilidad y comodidad al usuario final.

Para conseguir lo mencionado, se debe indicar al ADSP-2181 que en caso de que se evalúe el oído derecho, se coloque el valor que toma la onda sinusoidal en cada interrupción en el registro (tx\_buf+1) y cero en el registro (tx\_buf+2), y se realiza el procedimiento análogo en caso de querer evaluar el oído izquierdo.

El procedimiento mediante el cual se selecciona el oído a evaluar, se explica de manera explícita en el análisis del programa implementado en el ADSP-281 <sup>17</sup>

### 3.7 ANÁLISIS DEL SOFTWARE IMPLEMENTADO

Para conseguir el correcto funcionamiento del AMCEAC-I, se debe enviar ciertos caracteres desde la computadora hacia el Ez – Kit Lite.

Según el carácter recibido, el DSP realiza una función específica.

El protocolo implementado se especifica en la tabla 3.2:

Carácter	Proceso a Realizar
D	Evaluar Oído Derecho
I	Evaluar Oído Izquierdo
C	Cambiar frecuencia del tono generado
2	Bajar Intensidad del Oído Derecho
3	Bajar Intensidad del Oído Izquierdo
4	Subir Intensidad del Oído Derecho

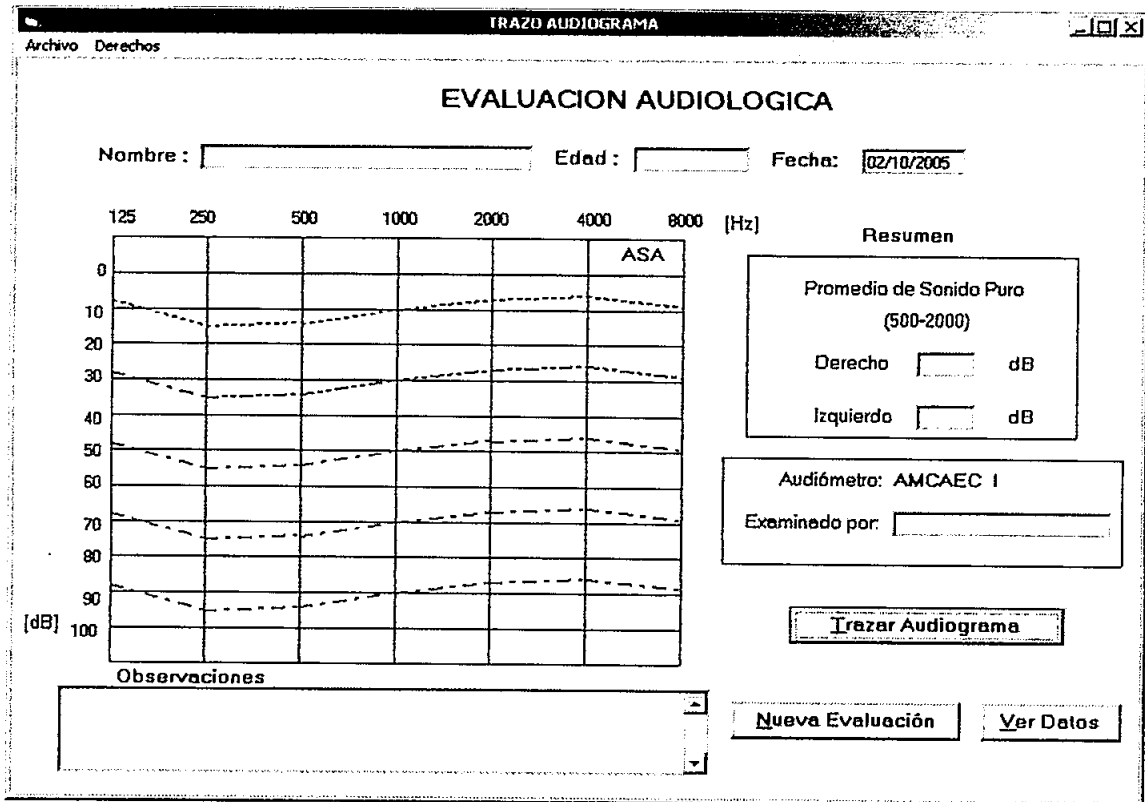
<sup>17</sup> Ver Capítulo III, numeral 3.10

5	Subir Intensidad del Oído Izquierdo
F	Silenciar al ADSP-2181

**Tabla 3.2**

Así, varios comandos del software implementado, tienen por función transmitir dichos caracteres, mientras que otros cumplen funciones internas en el programa.

Cabe decir que los caracteres que envía el computador, son en formato ASCII. Las figura 3.9 y 3.10 indican las pantallas donde se ubican los comandos encargados de realizar estos procesos:



**Figura 3.9**

## EVALUACION

**FRECUENCIA**

Cambiar

**INTENSIDAD**

Subir

Bajar

**SONIDO**

Generado

No Generado

Iniciar Evaluación

Ver Plantilla

**EVALUAR**

Oído Derecho

Oído Izquierdo

**DATOS**

Grabar Info

Enviar Info

[Hz]

**OIDO EVALUADO**

Derecho

Izquierdo

**DATOS OIDO DERECHO**

Frecuencia [Hz]:	125	250	500	1000	2000	4000	8000
Intensidad [dB]:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

**DATOS OIDO IZQUIERDO**

Frecuencia [Hz]:	125	250	500	1000	2000	4000	8000
Intensidad [dB]:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

*Figura 3.10*

A continuación, se describe la función que realiza cada comando:

- **Oído Derecho:** En caso de que el puerto de comunicaciones serial esté cerrado, lo abre.

Luego transmite el carácter D.

En caso de que el puerto de comunicaciones serial esté abierto, transmite el carácter D directamente.

Al transmitir este carácter, se le indica al DSP que se va a evaluar el oído derecho.

Es importante mencionar que el puerto de comunicaciones está configurado para trabajar a una velocidad de 9600 kbps y los datos son enviados en formato n,8,1. Esto se debe a que estos son los parámetros con los que trabaja el Ez – Kit Lite.

- **Oído Izquierdo:** En caso de que el puerto de comunicaciones serial esté cerrado, lo abre.

Luego transmite el carácter I.

En caso de que el puerto de comunicaciones serial esté abierto, transmite el carácter I directamente.

Al transmitir este caracter, se le indica al DSP que se va a evaluar el oído izquierdo.

- **Iniciar Evaluación** : En caso de que se haya escogido el oído derecho para iniciar la evaluación, al presionar este comando se envían de manera seguida los caracteres C y 4. Se envía el caracter 4 ya que, en un principio, el sonido se genera a la intensidad máxima, pero como dicha intensidad está programada para trabajar de manera circular (luego del máximo valor, se regresa al mínimo), al subirla, se le está asignando al sonido la mínima intensidad.

Con esto se consigue indicar al DSP que inicie la generación del tono de 125 Hz y que disminuya a cero la intensidad del mismo.

En caso de que se haya escogido el oído izquierdo para iniciar la evaluación, al presionar este comando se envían de manera seguida los caracteres C y 5. El caracter 5 cumple la misma función que el 4 en el caso anterior.

Con esto se consigue indicar al DSP que inicie la generación del tono de 125 Hz y que disminuya a cero la intensidad del mismo.

- **Generado**: Esta opción indica que el tono está siendo generado, por lo cual no se puede variar la frecuencia del mismo, sino únicamente su intensidad. Por esta razón, se deshabilita la sección frecuencia y queda habilitada la sección intensidad.
- **No Generado**: Esta opción se usa para dejar de generar un tono, una vez que se ha obtenido la respuesta del paciente.

Al presionar este comando, se envía el caracter F, con lo cual se silencia al DSP, por lo cual, al no haber sonido, no se puede variar la intensidad del mismo, sino únicamente la frecuencia. Por esta razón, se habilita la sección frecuencia y queda deshabilitada la sección intensidad.

- **Subir** :Este comando sirve para aumentar la intensidad del tono generado.

Al presionarlo, se envía al DSP el caracter 4 o 5, dependiendo de si se está evaluando el oído derecho o izquierdo respectivamente.

- **Bajar**: Este comando sirve para disminuir la intensidad del tono generado.

Al presionarlo, se envía al DSP el caracter 2 o 3, dependiendo de si se está evaluando el oído derecho o izquierdo respectivamente.

- **Cambiar:** Este comando sirve para cambiar la frecuencia del tono generado.

Al presionarlo, se envía al DSP el caracter C y 4 o C y 5, dependiendo de si se está evaluando el oído derecho o izquierdo respectivamente. Este proceso es similar al realizado al presionar el comando Iniciar Evaluación

De esta manera, se produce el cambio de frecuencia y se pone en cero la intensidad del tono generado.

- **Grabar Dato:** Al presionar esta opción, se almacena la información obtenida en la casilla de la sección de datos correspondiente. Al grabar los datos obtenidos, se elimina la parte decimal de la intensidad, ya que, luego de consultar con un especialista, se ha considerado que 0.5 dB no son significativos para los fines que persigue la evaluación, sino que más bien pueden crear confusión. No se transmite ningún carácter.
- **Borrar Dato:** Al presionar esta opción, se borra la información obtenida en la casilla de la sección de datos correspondiente. No se transmite ningún carácter.

Su función se explica de manera explícita en la parte concerniente a la Descripción del AMCAEC I en este mismo capítulo.

- **Ver Plantilla:** Esta opción da paso a la pantalla de la figura 9 y además envía al DSP el carácter F para silenciarlo.
- **Trazar Audiograma:** Esta opción permite trazar el audiograma según los datos obtenidos. No se transmite ningún carácter.

Su función se explica de manera explícita en la parte concerniente a la Descripción del AMCAEC I en este mismo capítulo.

- **Nueva Evaluación:** Permite realizar una nueva evaluación, No se transmite ningún carácter.

Su función se explica de manera explícita en la parte concerniente a la Descripción del AMCAEC I en este mismo capítulo.

- **Ver Datos:** Esta opción da paso a la pantalla indicada en la figura 3.9. No se transmite ningún carácter.



Su función se explica de manera explícita en la parte concerniente a la Descripción del AMCAEC I en este mismo capítulo.

Es importante mencionar que para una mejor comprensión de todo lo expuesto, se debe analizar en conjunto el Software Implementado, la Descripción del Amacaec I<sup>18</sup> y el Programa Implementado en el ADSP-2181<sup>19</sup>

### 3.8 ANÁLISIS DEL PROGRAMA IMPLEMENTADO EN EL ADSP-281

Primero se debe dar un nombre e inicializar al programa implementado.

Luego, se deben declarar y definir las constantes a ser usadas.

Estas constantes son:

- **Fs** :Frecuencia de muestreo del Codec
- **Ns**: Número de secciones de segundo orden del filtro
- **N\_coefs** :Número de coeficientes del filtro
- **N\_delay** : Longitud de la línea de retardo del filtro

Luego, se deben declarar y definir las variables a ser usadas.

Estas variables son:

- **tbl\_frec[7]** : Vector de 7 valores conteniendo a los avances correspondientes a cada tono
- **sin\_coef[5]** : Vector de 5 valores conteniendo a los coeficientes de la ecuación de cálculo del seno.
- **Coefs[N\_coefs]** : Vector de 10 valores conteniendo a los coeficientes de las secciones del filtro.
- **N\_delay\_data[N\_delay]** : Vector de 6 valores para almacenar las líneas de retardo del filtro (buffer).

Luego se deben definir las banderas a ser usadas en el programa.

Estas banderas son variables que el programador usa para controlar ciertos eventos. Son las siguientes:

- **frec\_actual**: aquí se almacena el valor del avance del tono generado.
- **c\_cntr** : aquí se almacena el valor de la suma del tono generado.

---

<sup>18</sup> Ver Capítulo III, Numeral 3.11

<sup>19</sup> Ver Capítulo III, Numeral 3.10

- **Derecho** : indica si se va a evaluar el oído derecho
- **Izquierdo**: indica si se va a evaluar el oído izquierdo
- **Seno** : aquí se almacena el valor del seno calculado
- **var\_R**: indica que se va a variar la intensidad del oído derecho
- **var\_L**: indica que se va a variar la intensidad del oído izquierdo
- **atenuacionR**: aquí se almacena el valor de atenuación del oído derecho.
- **AtenuacionL**: aquí se almacena el valor de atenuación del oído izquierdo.

Luego, se definen las interrupciones a usar.

Las interrupciones a usar son las siguientes:

- Interrupción de transmisión del pòrtico serial 0
- Interrupción de recepción del pòrtico serial 0
- Interrupción del timer
- Interrupción externa IRQE

Luego, se llama a la subrutina de inicialización del Codec. Esta subrutina consta de manera explícita en el anexo 6.

Luego, se llama a las subrutinas de inicialización de la transmisión y recepción serial. Esta subrutina consta de manera explícita en el anexo 11.

Con esta , se llama a la subrutina `call_get_char_ax1`, contenida en la anterior, y el valor recibido por el DSP, se almacena en `AX1`. De esta manera se puede procesar la información recibida desde el computador.

Luego, se inicializan todas las variables y banderas definidas anteriormente.

Luego, se realiza el chequeo de los caracteres recibidos por DSP, y según el mismo protocolo definido anteriormente, se realiza la función correspondiente.

Para cada carácter recibido, existe una función específica.

Estas funciones, según el carácter recibido, son las siguientes:

- **Der**: el programa viene aquí cuando se ha recibido una D. En este proceso, se pone en uno a la bandera derecho y en cero a la bandera izquierdo. Sirve para seleccionar el canal por el cual saldrá el tono generado.
- **Izq**: el programa viene aquí cuando se ha recibido una I. En este proceso, se pone en uno a la bandera izquierdo y en cero a la bandera

derecho. Sirve para seleccionar el canal por el cual saldrá el tono generado.

- **Cam\_ton:** el programa viene aquí cuando se ha recibido una C. En este proceso, se cambia el valor del avance. Sirve para cambiar la frecuencia del tono generado.
- **baje\_der:** el programa viene aquí cuando se ha recibido un 2. En este proceso, se aumenta la atenuación del tono generado y se guarda su valor en atenuaciónR. Además, se pone en uno la bandera var\_R para indicar que se va a alterar la intensidad del oído derecho. Se usa para bajar la intensidad del tono generado en el oído derecho.
- **baje\_izq :** el programa viene aquí cuando se ha recibido un 3. En este proceso, se aumenta la atenuación del tono generado y se guarda su valor en atenuaciónL. Además, se pone en uno la bandera var\_L para indicar que se va a alterar la intensidad del oído izquierdo. Se usa para bajar la intensidad del tono generado en el oído izquierdo.
- **suba\_der:** el programa viene aquí cuando se ha recibido un 4. En este proceso, se disminuye la atenuación del tono generado y se guarda su valor en atenuaciónR. Además, se pone en uno la bandera var\_R para indicar que se va a alterar la intensidad del oído derecho. Se usa para subir la intensidad del tono generado en el oído derecho.
- **suba\_izq:** el programa viene aquí cuando se ha recibido un 5. En este proceso, se disminuye la atenuación del tono generado y se guarda su valor en atenuaciónL. Además, se pone en uno la bandera var\_L para indicar que se va a alterar la intensidad del oído izquierdo. Se usa para subir la intensidad del tono generado en el oído izquierdo.
- **Fin\_ton:** el programa viene aquí cuando se ha recibido una F. En este proceso, se encera el valor del avance, lo cual implica que se deja de producir sonido. Además se realizan los cambios necesarios para que el siguiente tono a generarse empiece con la intensidad mínima. Estos cambios incluyen la verificación del oído que se está evaluando.

Luego, se indica el proceso que se debe realizar cada vez que se produce una interrupción del Codec. Dicho proceso es el siguiente:

- Se habilitan los registros secundarios

- Se llama a la subrutina volumen, en la cual se produce el cambio de intensidad del tono generado según los valores de atenuación previamente almacenados.
- Se suman los valores almacenados en suma y avance.
- Se llama a la subrutina seno, el cual se calcula con la expansión polinómica de Taylor
- Se llama a la subrutina filtro, en la cual se filtra la muestra obtenida.
- Se llama a la subrutina oído, en la cual se define por qué canal se transmite la muestra obtenida, la cual ya está filtrada y determinada su intensidad.
- Se transmite la información digital al Codec para que este genere una señal analógica.

El programa implementado, consta de manera explícita en el anexo 12.

### 3.9 DESCRIPCIÓN DEL AMCAEC I

Esta sección tiene por objetivo ser una guía para que el usuario del audiómetro diseñado pueda operarlo.

El dispositivo AMCAEC I se constituye de dos partes básicas:

- Hardware
- Software

#### 3.9.1 HARDWARE

##### 3.9.1.1 Ez – Kit Lite

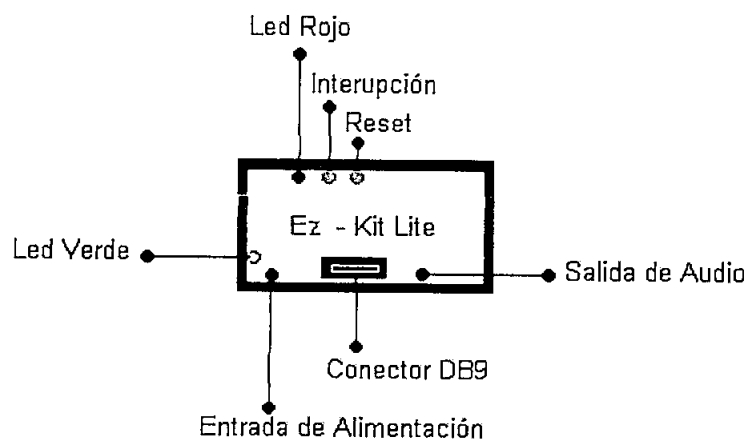
El usuario final tiene acceso a siete partes de esta tarjeta , las cuales son las siguientes:

- **Conector DB-9:** Aquí se conecta en cable que comunica al computador con la tarjeta. El cable que se conecta es un DB9. El otro extremo del cable se conecta al puerto serial del computador.

- **Salida de Audio:** Aquí se conectan los audífonos.
- **Pulsador de reset:** Permite reiniciar el programa que hace que la tarjeta en cuestión realice las funciones requeridas por el AMCAEC I.
- **Pulsador de Interrupción:** No es usado.
- **Entrada de alimentación:** Aquí se conecta el adaptador que alimenta a la tarjeta. El otro extremo del adaptador se conecta a un tomacorriente cualquiera.
- **Led Rojo:** Permanece encendido mientras se realiza una evaluación. El momento que esta termina, dicho Led se apaga.
- **Led Verde:** Permanece encendido mientras el dispositivo se encuentra conectado a la alimentación de un tomacorriente.

Todos estos elementos vienen incluidos en el dispositivo.

La figura 3.11 muestra, de manera esquemática, cada una de las partes descritas:



**Figura 3.11**

La figura 3.12 muestra, de manera esquemática, cada una de las conexiones que deben ser realizadas para que el AMCAEC I entre en operación:

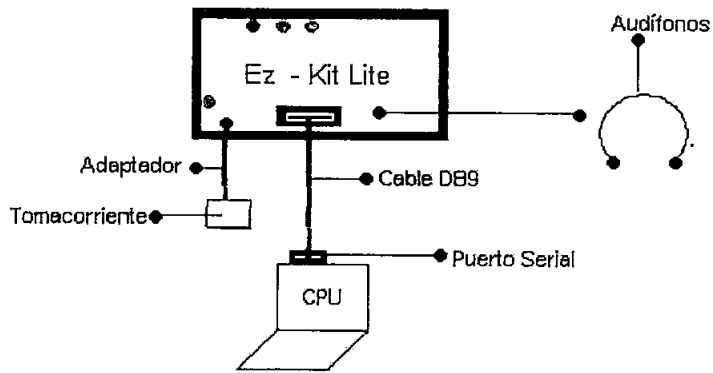


Figura 3.12

### 3.9.2 SOFTWARE

Es un programa que permite obtener el audiograma y registrar los datos obtenidos del paciente evaluado.

Una vez instalado el programa (el cual es un solo archivo ejecutable que se debe copiar en cualquier ubicación del disco duro), se lo corre directamente.

#### 3.9.2.1 Elementos del Software:

Al iniciar el programa, aparece la pantalla que se muestra en la figura 3.13:

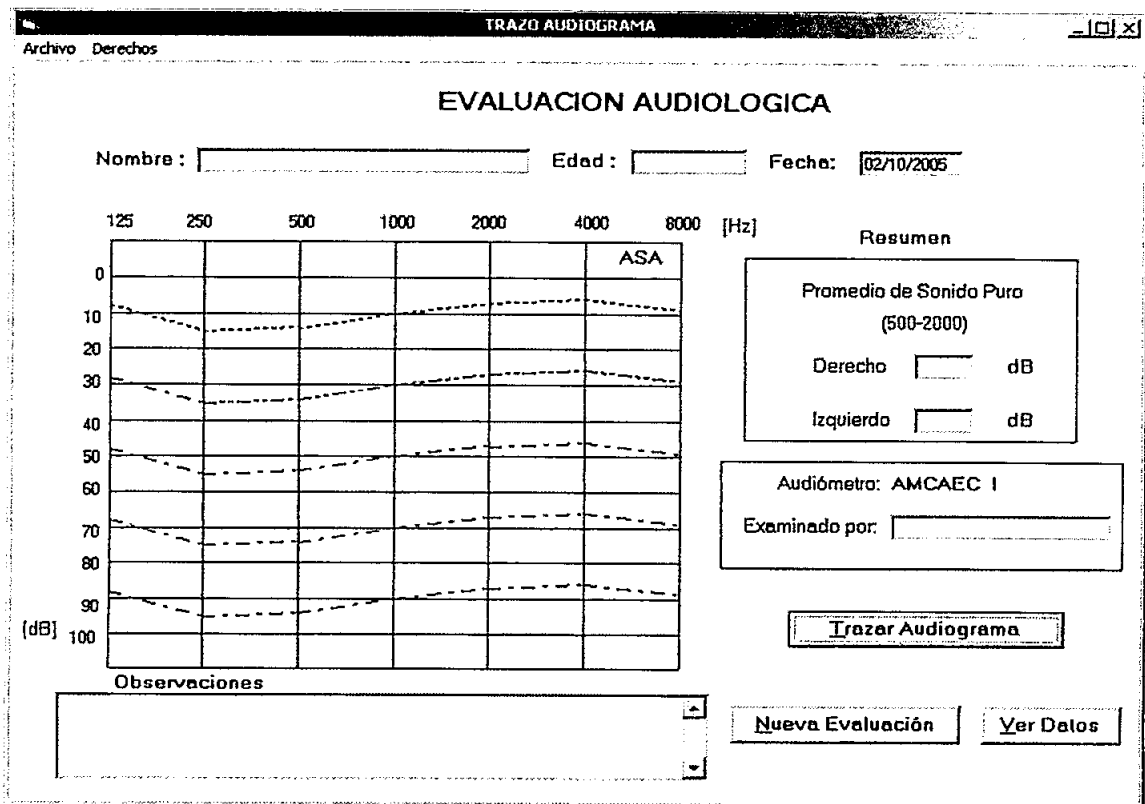


Figura 3.13

- **Casilla “Nombre”:** En esta casilla, el usuario debe escribir el nombre del paciente evaluado.
- **Casilla “Edad”:** En esta casilla , el usuario debe escribir la edad del paciente evaluado.
- **Casilla “Fecha” :** Esta casilla se carga automáticamente con la fecha del día en el que se realiza la evaluación.
- **Cuadro “Resumen” y casilla “Observaciones”:** En la casilla Promedio de Sonido Puro se graba el promedio de la intensidad a la cual el paciente escuchó los tonos de 500, 1000 y 2000 Hz. Este proceso se realiza automáticamente (este valor es también calculado sin decimales). Las casilla “Observaciones” debe ser llenada por el médico que interpreta el audiograma obtenido
- **Opción “Trazar Audiograma”:** Esta opción permite trazar el audiograma en base a los datos obtenidos del paciente evaluado.  
Es importante mencionar que los datos con los cuales se traza el audiograma, provienen de los valores almacenados que constan en la pantalla indicada por la figura 3.9.
- **Opción “Nueva Evaluación”:** Esta opción permite borrar todos los datos obtenidos en una evaluación anterior (tanto gráfica como datos almacenados).
- **Opción “Ver Datos”:** Esta opción da paso a la pantalla indicada en la figura 3.14:

## EVALUACION

**FRECUENCIA**

Cambiar

**INTENSIDAD**

Subir

Bajar

**SONIDO**

Generado

No Generado

Iniciar Evaluacion

Ver Plantilla

**Evaluar**

Oído Derecho

Oído Izquierdo

**DATOS**

Grabar Datos

Borrar Datos

[H7]

**OÍDO EVALUADO**

Derecho

Izquierdo

**DATOS OÍDO DERECHO**

Frecuencia [Hz]:	125	250	500	1000	2000	4000	8000
Intensidad [dB]:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

**DATOS OÍDO IZQUIERDO**

Frecuencia [Hz]:	125	250	500	1000	2000	4000	8000
Intensidad [dB]:	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

*Figura 3.14*

En esta pantalla, el usuario puede realizar la evaluación para obtener los datos que permitirán realizar el audiograma correspondiente a la misma.

En esta pantalla aparecen las siguientes secciones perfectamente diferenciadas:

- **Sección de Datos Almacenados:** Esta sección tiene dos sub secciones:
  - **“Datos Oído Izquierdo”:** Aquí se almacenan los datos de la intensidad a la cual el paciente escuchó cada tono en la evaluación del oído izquierdo.
  - **“Datos Oído Derecho”:** Aquí se almacenan los datos de la intensidad a la cual el paciente escuchó cada tono en la evaluación del oído derecho.
- **Sección “Evaluar”:** En esta sección se tienen dos opciones:
  - **Opción “Oído Derecho”:** Al presionar esta opción, se indica que se iniciará la evaluación del oído derecho.
  - **Opción “Oído Izquierdo”:** Al presionar esta opción, se indica que se iniciará la evaluación del oído izquierdo.



Es importante mencionar que cuando se escoge una de las dos opciones, se da inicio a una nueva evaluación. Si no se escoge una de estas, no se está realizando ninguna evaluación.

- **Sección “Oído Evaluado”**: Esta sección tiene 2 etiquetas, las cuales indican qué oído se está evaluando :
  - **Etiqueta “Oído Derecho”**: Esta etiqueta se resalta con negrilla cuando se está evaluando el oído derecho.
  - **Etiqueta “Oído Izquierdo”**: Esta etiqueta se resalta con negrilla cuando se está evaluando el oído izquierdo.

\* **Sección “Frecuencia”** : Esta sección contiene una opción y una casilla de texto.

- **Opción “Cambiar”** : Esta opción permite cambiar la frecuencia del tono generado. Las frecuencias de los tonos que se generan son de 125, 250, 500, 1000, 2000, 4000 y 8000 Hz. Luego, esto se repite de manera circular.
- **Casilla de texto**: El usuario no tiene acceso a escribir en esta casilla. Aquí se indica la frecuencia del tono generado.

Con respecto a esta sección, se debe decir que esta sección está activa únicamente cuando el sonido no está siendo generado, ya que cuando hay sonido, la sección queda deshabilitada.

\* **Sección “Intensidad”** : Esta sección contiene dos opciones y una casilla de texto.

- **Opción “Subir”** : Esta opción permite subir la intensidad del tono generado.
- **Opción “Bajar”** : Esta opción permite bajar la intensidad del tono generado.
- **Casilla de texto**: El usuario no tiene acceso a escribir en esta casilla. Aquí se indica la intensidad del tono generado. Esta intensidad está medida en dB.

Con respecto a esta sección, es importante indicar que, cada vez que se inicia un tono nuevo, éste tiene una intensidad de cero. Cuando la intensidad es cero, se desactiva la opción “Bajar”, con lo cual se impide disminuir la intensidad del tono generado.

La intensidad máxima que se puede alcanzar es 94.5. Cuando la intensidad es máxima, se desactiva la opción "Subir", con lo cual se impide aumentar la intensidad del tono generado.

Esta sección está activa únicamente cuando el sonido está siendo generado, ya que cuando no hay sonido, la sección queda deshabilitada.

- **Sección "Sonido"**: Esta sección tiene dos opciones y permite generar el sonido o dejar de generarlo.
  - **Opción "Generado"** : Cuando se escoge esta opción, se empieza a generar el sonido según la frecuencia indicada en la sección correspondiente. Además, si esta opción es escogida, se habilita la sección "Intensidad" y se deshabilita la sección "Frecuencia"
  - **Opción "No Generado"**: Cuando se escoge esta opción, se deja de generar sonido. Además, si esta opción es escogida, se habilita la sección "Frecuencia" y se deshabilita la sección "Intensidad" .

\* **Opción "Iniciar Evaluación"**: Esta opción, permite iniciar una evaluación, para lo cual se habilitan las opciones correspondientes a las secciones de "Sonido" e "Intensidad" y se marca la opción "Generado".

Además, cuando se selecciona esta opción, luego de escogerla, la misma se desactiva y no se vuelve a activar sino cuando se va a evaluar el otro oído.

\* **Opción "Ver Plantilla"** : Esta opción da paso a la pantalla indicada en la figura 3.13.

- **Sección "Datos"**: En esta sección se tienen dos opciones y una casilla de texto.
  - **Opción "Grabar Dato"**: Al presionar esta opción, se graba el dato obtenido en la sección de Datos correspondiente al oído evaluado. No se debe olvidar que cada dato se constituye de frecuencia e intensidad.
  - **Opción "Borrar Dato"**: Esta opción sirve para borrar datos erróneos. El dato que se borra es el correspondiente a la frecuencia indicada en la casilla de texto.

- **Casilla de texto:** El usuario tiene la posibilidad de escribir en esta casilla. Aquí se debe escribir la frecuencia del tono del cual se desea borrar la intensidad almacenada.

### 3.9.3 PROCEDIMIENTO PARA REALIZAR UNA EVALUACIÓN AUDIOMÉTRICA

Como ya se mencionó, al iniciar el programa, aparece la pantalla indicada en la figura 3.13. En esta pantalla, se debe elegir la opción "Ver Datos", con lo cual aparece la pantalla indicada en la figura 3.14.

En la pantalla 3.14, está activa la sección "Evaluar", en la cual se debe elegir el oído con el cual se va a iniciar la evaluación del paciente. Al escoger una de las dos opciones posibles, se excluye automáticamente la otra.

Además está activa la opción "Ver Plantilla", con la cual se puede acceder a la primera pantalla. Es importante mencionar que el paso de pantalla es permitido únicamente cuando los datos de la evaluación están completos o cuando no se ha dado inicio a una evaluación.

Cuando se elige el oído que se quiere evaluar (el cual puede ser cualquiera), se da inicio a la evaluación, y en la sección "Oído Evaluado", se indica el oído al cual se está examinando. Además, al escoger el oído, se activa la opción "Iniciar Evaluación".

Al escoger esta opción, se empieza a generar el primer tono con una intensidad de cero y se activan las secciones "Sonido", "Intensidad" y "Datos".

Luego se varía la intensidad del tono generado hasta que el paciente lo escuche. Una vez conseguido esto, se presiona la opción "Grabar Dato", con lo cual se almacena el dato obtenido.

Una vez que se graba el dato, se debe dejar de generar el sonido, y esto se consigue con la opción "No Generado" de la sección "Sonido". Cuando se hace esto, se desactiva la sección "Intensidad" y se activa la sección "Frecuencia", con lo cual se puede cambiar la frecuencia del tono generado.

Al cambiar dicha frecuencia, se desactiva la sección "Cambiar" y se activa la sección "Intensidad" y se activa la opción "Generado" de la sección "Sonido".

Así, se repite el proceso hasta completar los datos de intensidad para cada frecuencia en cada oído.

En caso de existir algún error al grabar un dato, se puede usar la opción "Borrar Dato", la misma que ya fue escrita anteriormente.

Una vez obtenidos todos los datos, se debe escoger la opción "Ver plantilla", con lo cual se pasa a la pantalla de la figura 3.14.

En esta pantalla, se debe elegir la opción "Trazar Audiograma", con lo cual se traza el audiograma.


Ya obtenido el audiograma, el evaluador debe llenar la casilla correspondiente a diagnóstico (casilla "Observaciones").

Para realizar una nueva evaluación, se debe escoger la opción "Nueva Evaluación", con lo cual se borran todos los datos hasta aquí obtenidos.

Hecho eso, se procede a repetir todo el procedimiento indicado para realizar una nueva evaluación.

La figura 3.15 indica un ejemplo con pantalla de datos completa:

### EVALUACION



**FRECUENCIA**

Cambiar

**INTENSIDAD**

Subir

Bajar

**SONIDO**

Generado

No Generado

Iniciar Evaluación

**Ver Plantilla**

**Evaluar**

Oído Derecho

Oído Izquierdo

**DATOS**

Grabar Dato

Borrar Dato

[Hz]

**OIDO EVALUADO**

Derecho

Izquierdo

**DATOS OIDO DERECHO**

Frecuencia [Hz]:	125	250	500	1000	2000	4000	8000
Intensidad [dB]:	<input type="text" value="25"/>	<input type="text" value="21"/>	<input type="text" value="21"/>	<input type="text" value="15"/>	<input type="text" value="10"/>	<input type="text" value="15"/>	<input type="text" value="12"/>

**DATOS OIDO IZQUIERDO**

Frecuencia [Hz]:	125	250	500	1000	2000	4000	8000
Intensidad [dB]:	<input type="text" value="30"/>	<input type="text" value="27"/>	<input type="text" value="15"/>	<input type="text" value="10"/>	<input type="text" value="6"/>	<input type="text" value="21"/>	<input type="text" value="10"/>

Figura 3.15

La figura 3.16 indica un ejemplo de un audiograma trazado con los datos de la figura 3.15:

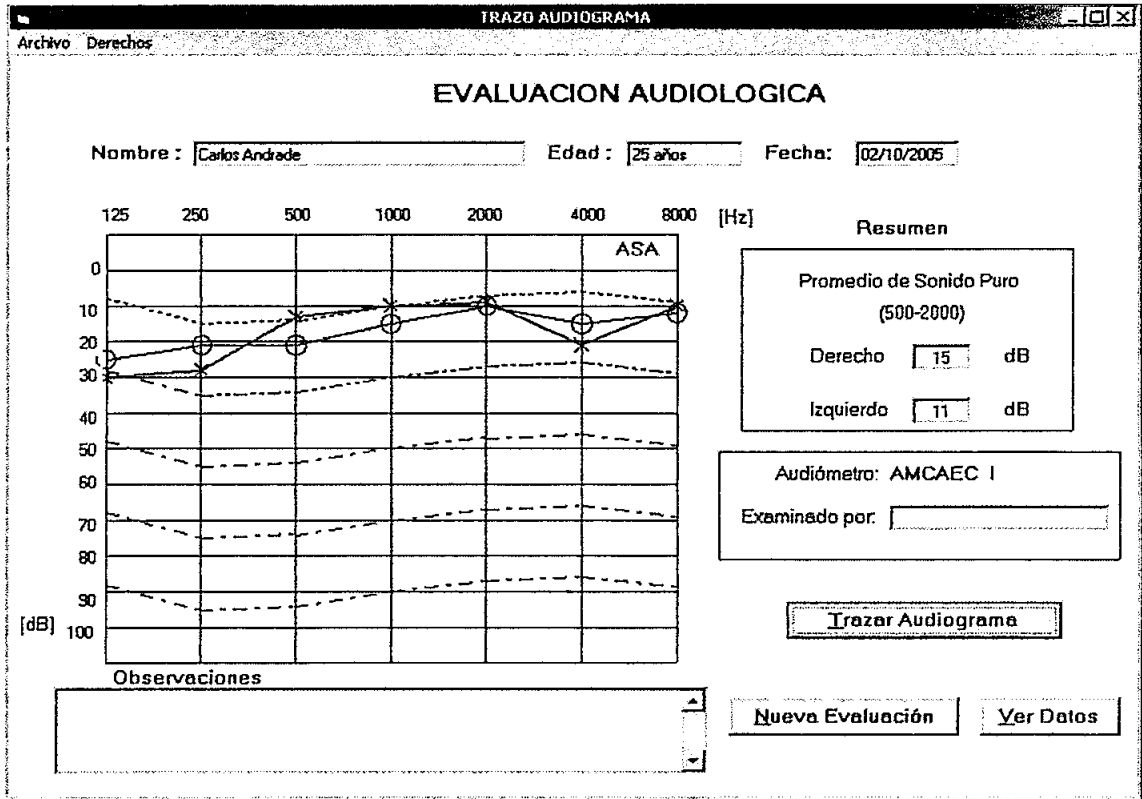


Figura 3.16

### 3.9.3 ERRORES

Al manejar el software, puede ocurrir que el usuario cometa errores. Esto son los siguientes:

- En caso de escoger la opción "Trazar Audiograma", en la pantalla indicada por la figura 3.13, sin tener los datos completos, aparece el cuadro de diálogo indicado en la figura 3.17:

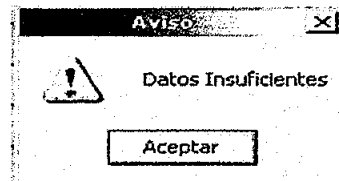
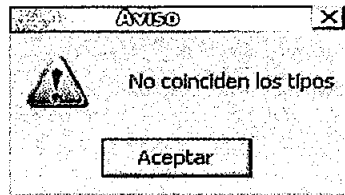


Figura 3.17

Luego de presionar "Aceptar", se vuelve a la pantalla inicial.

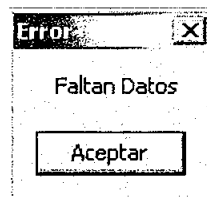
- En caso de escoger la opción "Nueva Evaluación", en la pantalla indicada por la figura 3.13, sin haber realizado una evaluación previa, aparece el cuadro de diálogo indicado en la figura 3.18:



**Figura 3.18**

Luego de presionar "Aceptar", se vuelve a la pantalla inicial.

- En caso de escoger la opción "Ver Plantilla", en la pantalla indicada por la figura 3.14, el cambio de pantalla ocurre sólo en caso de no haber iniciado una evaluación. En caso de ya haber iniciado una evaluación (haber seleccionado el oído a evaluar) y de escoger la opción de la cual se está hablando, aparece el cuadro de diálogo indicado en la figura 3.19:



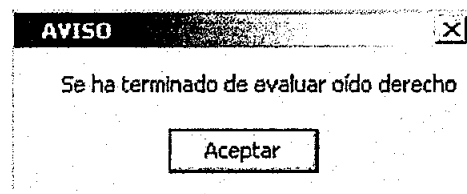
**Figura 3.19**

Luego de presionar "Aceptar", se vuelve a la pantalla inicial.

### 3.9.5 AVISOS

Al manejar el software, luego de ciertos eventos, aparecerán los siguientes avisos . Esto son los siguientes:

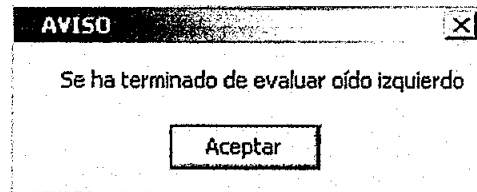
- En caso de haber completado los datos del oído derecho , aparecerá el cuadro de diálogo indicado en la figura 3.20:



**Figura 3.20**

Luego de presionar "Aceptar", se vuelve a la pantalla inicial.

- En caso de haber completado los datos del oído izquierdo , aparecerá el cuadro de diálogo indicado en la figura 3.21:

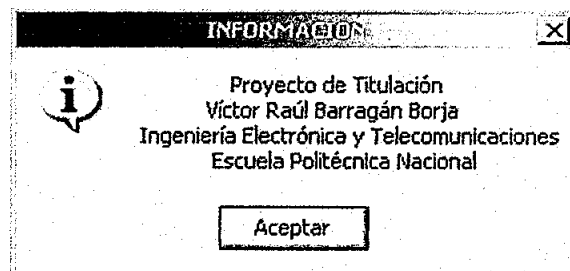


**Figura 3.21**

Luego de presionar "Aceptar", se vuelve a la pantalla inicial.

Finalmente, se debe mencionar que el programa cuenta con un menú, en el cual se tienen las opciones de Abrir, Guardar, Imprimir y Salir, las cuales cumplen la misma función que en cualquier programa de Windows.

En ese mismo menú se tiene la opción "Derechos", en la cual constan los datos del realizador del presente proyecto de investigación. Al seleccionar dicha opción, se ve el mensaje de texto indicado en la figura 3.22:



**Figura 3.22**

## CAPITULO IV

### PRUEBAS Y ANÁLISIS DE RESULTADOS

#### 4.1 TIPOS DE ANÁLISIS

El análisis de los resultados se enfocó en los dos siguientes aspectos:

1. Determinar que el AMCAEC I trabaje de acuerdo a lo previsto en el diseño e implementación.
2. Determinar que los resultados obtenidos con el mismo, es decir los audiogramas, son válidos.

Con respecto al primer punto, se consideraron dos parámetros fundamentales de los tonos generados:

- Frecuencia de los tonos generados
- Intensidad de los tonos generados

Para analizar dichos parámetros, se realizó un estudio electrónico de dichos tonos, para de esta manera verificar que las características de los sonidos se ajusten a los requerimientos planteados en el capítulo II.

Con respecto al segundo punto, se evaluaron a 9 personas tanto con el AMCAEC I como con un audímetro comercial utilizado por un médico otólogo.

De esta manera, se compararon las diferencias de los resultados obtenidos con cada uno de los dispositivos.

Este es el análisis más importante, ya que en último término, la obtención de un audiograma válido es el fin que se persiguió desde el inicio de la realización del presente proyecto.

Es importante mencionar que, dada la dificultad tanto en el aspecto económico como en el factor de tiempo, no se pudo realizar pruebas con más pacientes.

Sin embargo, para compensar este inconveniente, se escogió pacientes de características sumamente diferentes y además se realizó un análisis estadístico de los datos producto de las evaluaciones con el afán de demostrar que los resultados obtenidos con el AMCAEC I, es decir los audiogramas, pueden ser considerados válidos y de esta manera demostrar que el dispositivo trabaja correctamente para cualquier tipo de paciente, sin importar el número de los mismos o sus características.



De todo lo dicho anteriormente, se desprende que se realizaron 3 tipos de análisis, los cuales son los siguientes:

- **Análisis electrónico** : En éste, se analizaron las características electrónicas de los tonos generados.
- **Análisis médico**: En éste , se compararon los audiogramas obtenidos con el dispositivo comercial y con el prototipo construido.
- **Análisis estadístico**: Se analizó estadísticamente los resultados obtenidos.

#### 4.1.1 ANALISIS ELECTRÓNICO

##### 4.1.1.1 Análisis de la Frecuencia de los Tonos Generados

La frecuencia de cada uno de los tonos para ambos oídos (izquierdo y derecho) se analizó con la ayuda del software de audio Cool Edit Pro 2.0, el cual permite, entre muchas otras opciones, visualizar una señal de audio que ingresa por el puerto de sonido del computador.

Para realizar el análisis, se visualizó la señal ingresada por el puerto, y luego se le ordenó al programa indicar la ubicación de dos cruces por cero consecutivos en un salto de positivo a negativo de la señal. Al hacer esto, se obtuvo el valor de tiempo en el cual se produjeron dichos cruces para un período completo de la señal y por diferencia de estos valores de tiempo, se obtuvo el período de la señal y su frecuencia.

Cabe mencionar que por efectos de visualización, todos los tonos han sido medidos a una misma escala de tiempo y a una intensidad conveniente.

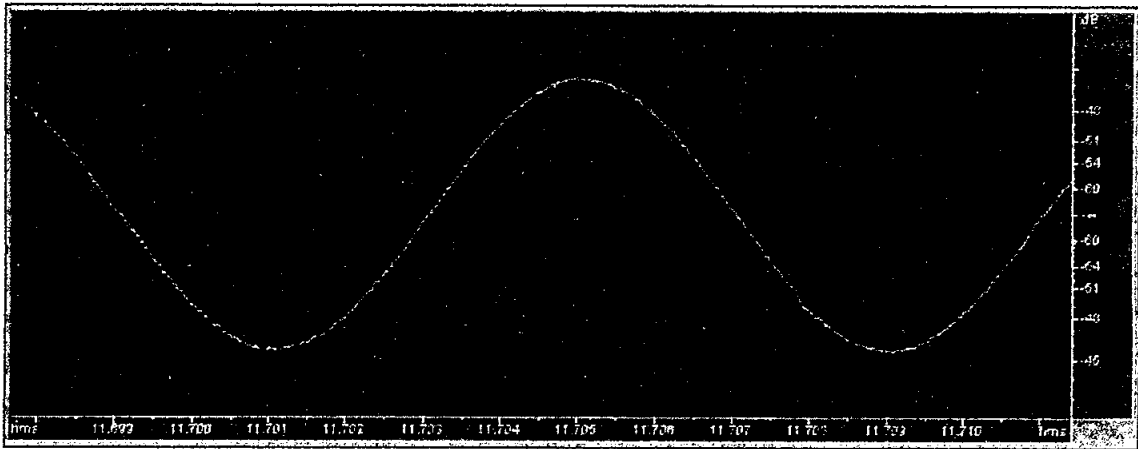
Por el momento, la intensidad no es importante ya que se está analizando sólo la frecuencia. Posteriormente se presentan los resultados obtenidos para la intensidad.

##### 4.1.1.1.1 Ondas con las Frecuencias Normalizadas

Las ondas para los oídos izquierdo y derecho son las mismas y a continuación se muestran las ondas generadas para las distintas frecuencias:

- Frecuencia de 125 Hz

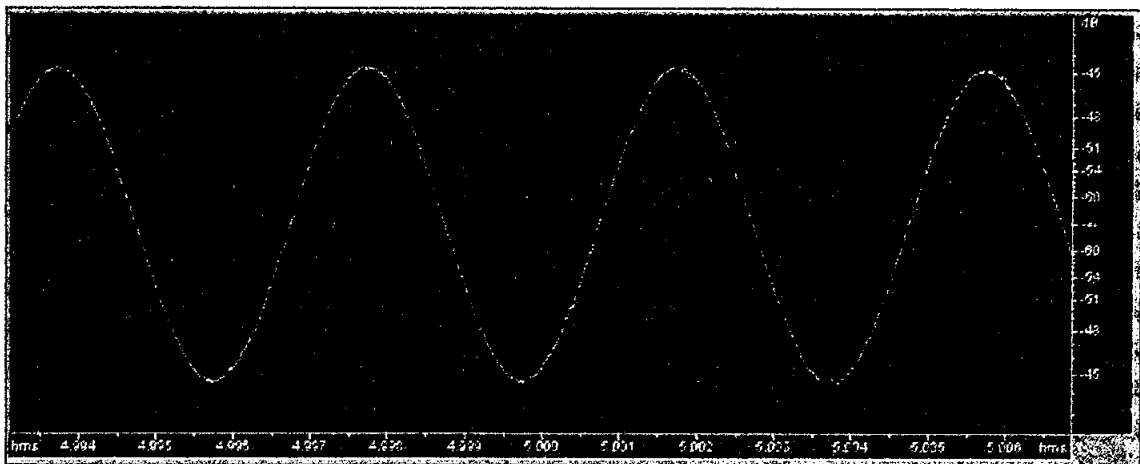
La visualización de la señal generada con frecuencia de 125 Hz consta en la figura 4.1:



*Figura 4.1*

- Frecuencia de 250 Hz

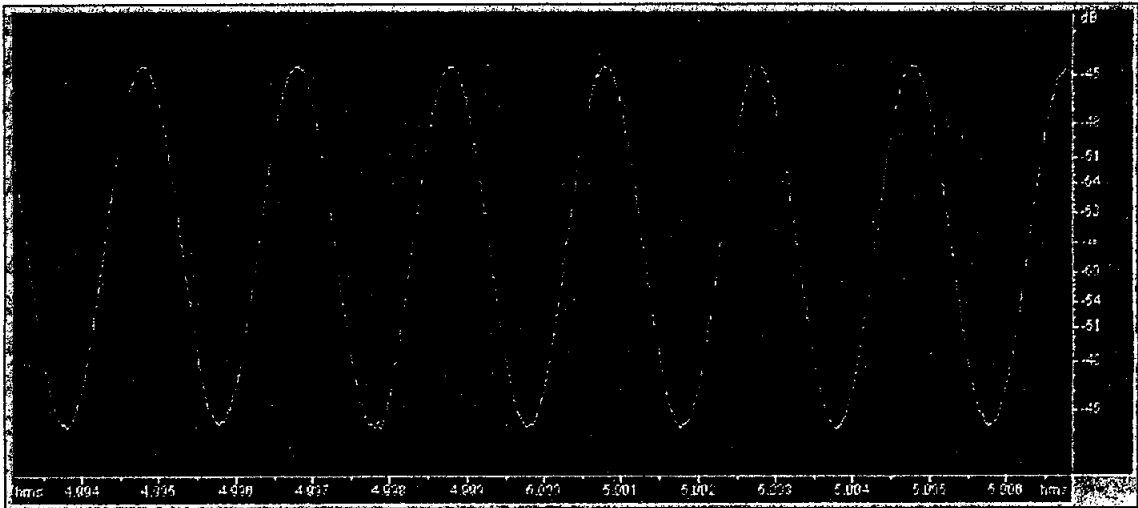
La visualización de la señal generada con frecuencia de 250 Hz consta en la figura 4.2:



*Figura 4.2*

- Frecuencia de 500 Hz

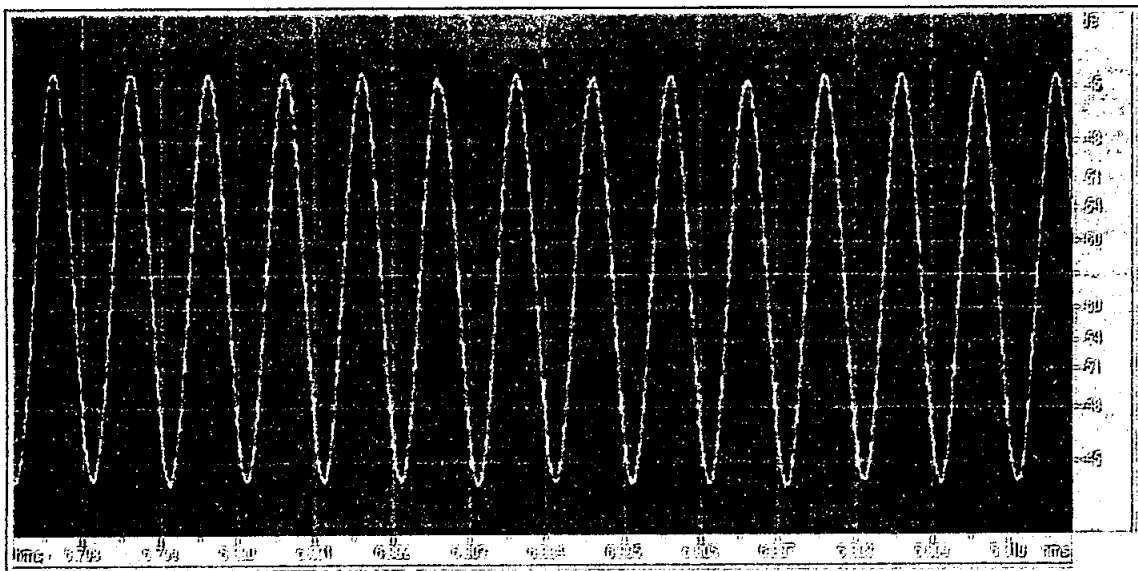
La visualización de la señal generada con frecuencia de 500 Hz consta en la figura 4.3:



*Figura 4.3*

- Frecuencia de 1000 Hz

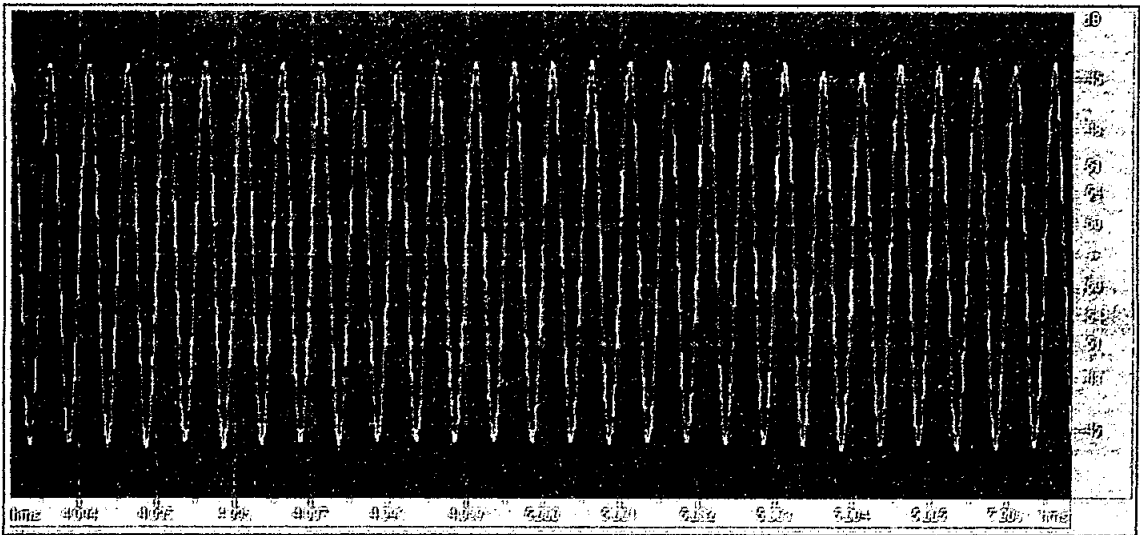
La visualización de la señal generada con frecuencia de 1000 Hz consta en la figura 4.4:



*Figura 4.4*

- Frecuencia de 2000 Hz

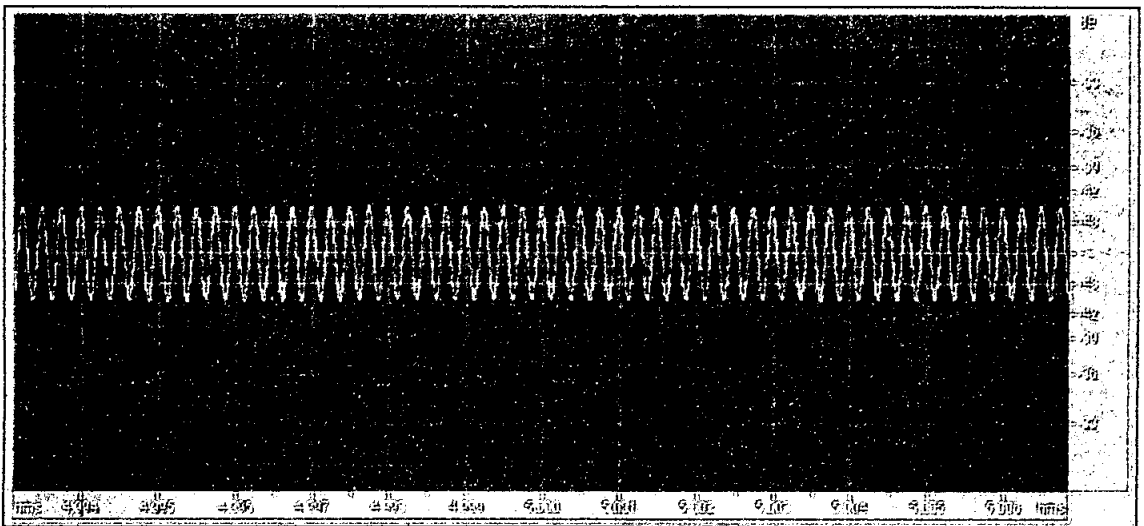
La visualización de la señal generada con frecuencia de 2000 Hz consta en la figura 4.5:



*Figura 4.5*

- Frecuencia de 4000 Hz

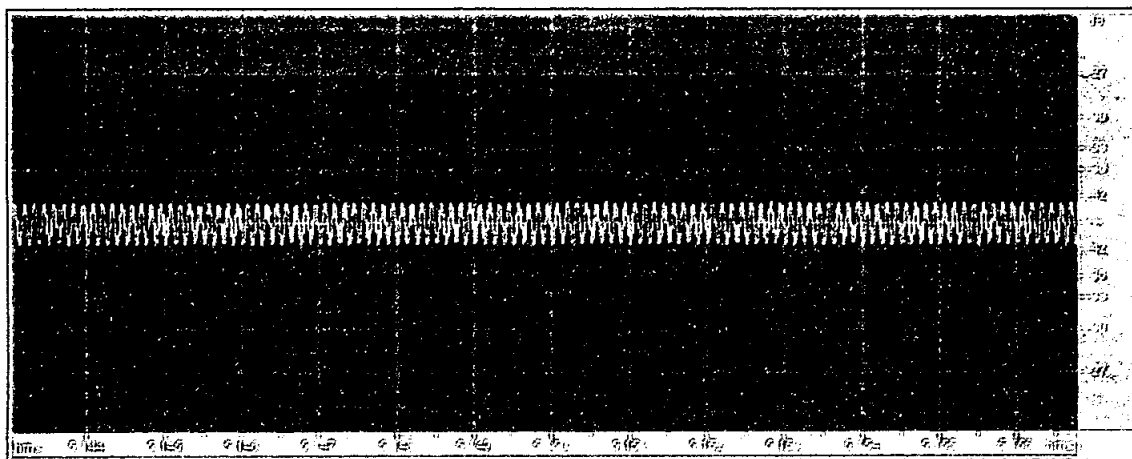
La visualización de la señal generada con frecuencia de 4000 Hz consta en la figura 4.6:



*Figura 4.6*

- Frecuencia de 8000 Hz

La visualización de la señal generada con frecuencia de 8000 Hz consta en la figura 4.7:



**Figura 4.7**

#### 4.1.1.1.2 Frecuencias nominales y frecuencias generadas

En la tabla 4.1, constan la frecuencia ideal, el primer corte por cero, el segundo corte por cero, el período de la señal generada y la frecuencia real.

Frecuencia Ideal [Hz]	Primer cruce por cero ( $t_1$ ) [seg]	Segundo cruce por cero ( $t_2$ ) [seg]	Período de la Señal ( $T = t_2 - t_1$ ) [seg]	Frecuencia Real ( $f = 1/T$ ) [Hz]
125	11.7003	11.7011	0.008	125
250	5.000	5.005	0.005	250
500	4.998	5.000	0.002	500
1000	6.800	6.801	0.001	1000
2000	5.002	5.003 *	0.0005	2000
4000	5.001	5.002 **	0.00025	4000
8000	5.006	5.007 ***	0.000125	8000

**Tabla 4.1**

\* Valor correspondiente a 2 períodos completos de la señal

\*\* Valor correspondiente a 4 períodos completos de la señal

\*\*\* Valor correspondiente a 8 períodos completos de la señal

Es importante mencionar que el software que se usó para este análisis, puede dar los saltos a los cruces por cero mínimo en pasos de 1 ms, razón por la cual, para frecuencias mayores a 1 KHz, se tuvo que hacer el ajuste que se indica. Finalmente, y para una verificación más completa, se midió la frecuencia de los tonos generados con un osciloscopio digital<sup>20</sup>.

La tabla 4.2 muestra la tabulación los resultados de dicha medición:

Frecuencia Ideal [Hz]	Frecuencia Real (Oído Derecho) [Hz]	Frecuencia Real (Oído Izquierdo) [Hz]
125	125.26	124.86
250	249.782	250.278
500	500.29	500.45
1000	999.85	1000.23
2000	2000.41	2000.19
4000	4000.08	4000.22
8000	8000.95	8000.84

**Tabla 4.2**

Se puede observar que, en el caso de las mediciones realizadas con el Cool Edit Pro, las frecuencias de los tonos generados son exactas, mientras que en las mediciones realizadas con el osciloscopio, en ninguno de los casos existe un error mayor a 1 Hz, así que se puede decir que se cumplen exitosamente los requerimientos de frecuencia de los tonos generados.

#### 4.1.1.2 Análisis de la Intensidad de los Tonos Generados

La intensidad de cada uno de los tonos para ambos oídos (izquierdo y derecho) se analizó, al igual que en el caso de las frecuencias, con la ayuda del software de audio Cool Edit Pro 2.0.

<sup>20</sup> Instrumento disponible en el Laboratorio de Sistemas Digitales de la Escuela Politécnica Nacional

Para realizar dicho análisis, se generó una señal con una intensidad predeterminada, se la ingresó por el puerto de audio, se la visualizó y luego se tomó nota del valor de voltaje correspondiente al pico positivo de la misma. Con estos datos, se obtuvo matemáticamente, basándose en la ecuación 4.1, el voltaje de referencia para cada caso.

$$\#dB = 20 \log \frac{V_m}{V_{ref}}$$

**Ecuación 4.1**

Donde :

$V_m$  = voltaje medido

$V_{ref}$  = voltaje de referencia

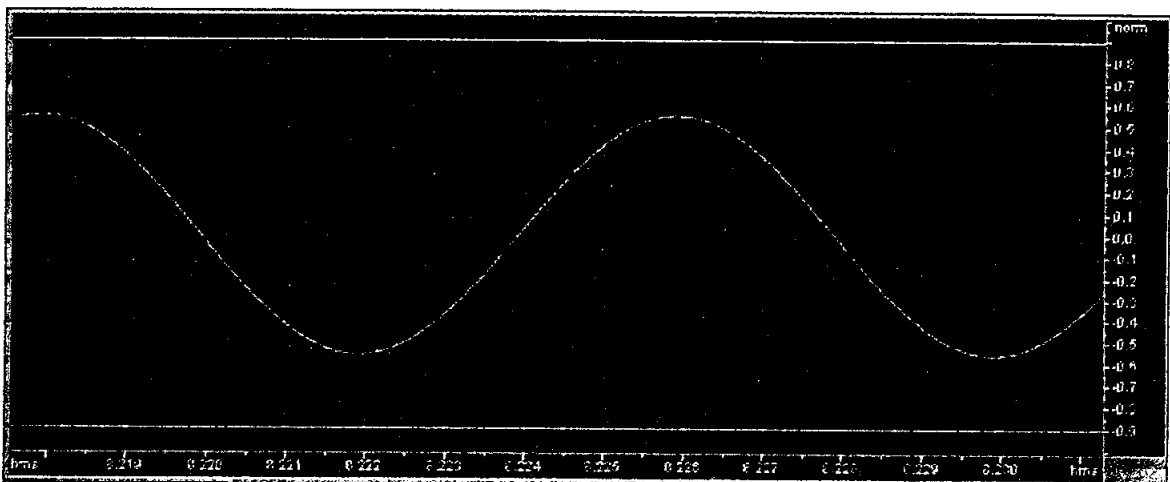
Así, se procedió primeramente a la obtención de los datos necesarios.

#### 4.1.1.2.1 Ondas para evaluación del Oído Izquierdo

Para el oído izquierdo, se muestran la visualizaciones de las formas de onda generadas y se indican los valores de voltaje obtenidos y la correspondiente intensidad en dB.

- **Frecuencia de 125 Hz**

La visualización de la señal generada con frecuencia de 125 Hz y con 94.5 dB de intensidad, consta en la figura 4.8:

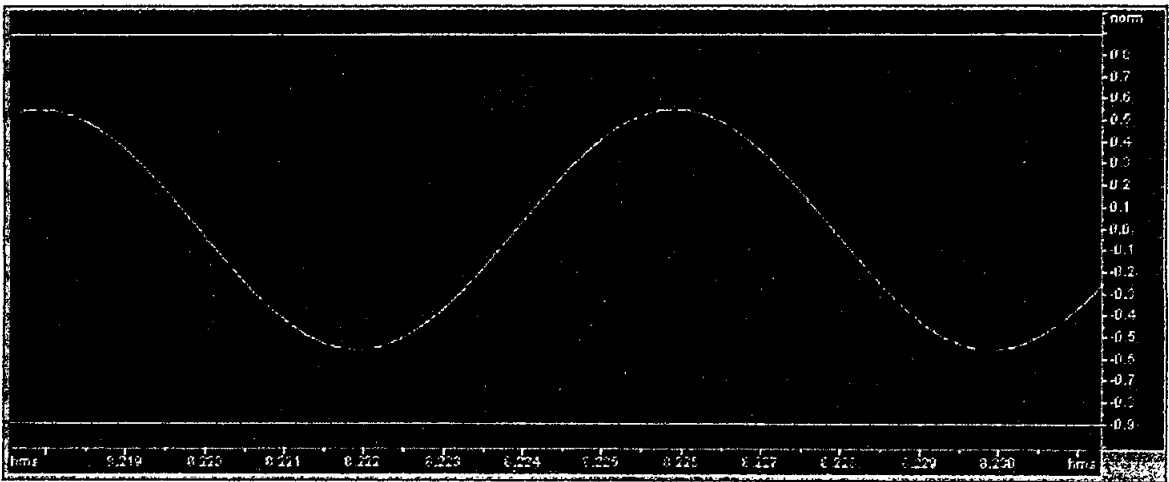


**Figura 4.8**

El valor de voltaje anotado del pico positivo fue de 0.59 V.

- **Frecuencia de 250 Hz**

La visualización de la señal generada con frecuencia de 250 Hz y con 90 dB de intensidad, consta en la figura 4.9:

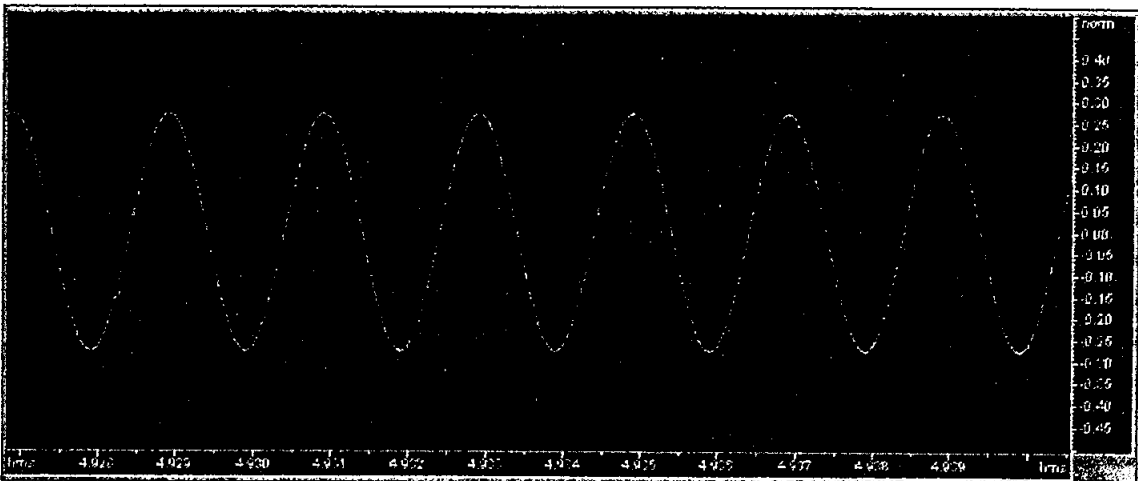


**Figura 4.9**

El valor de voltaje anotado del pico positivo fue de 0.36 V.

- **Frecuencia de 500 Hz**

La visualización de la señal generada con frecuencia de 500 Hz y con 87 dB de intensidad, consta en la figura 4.10:



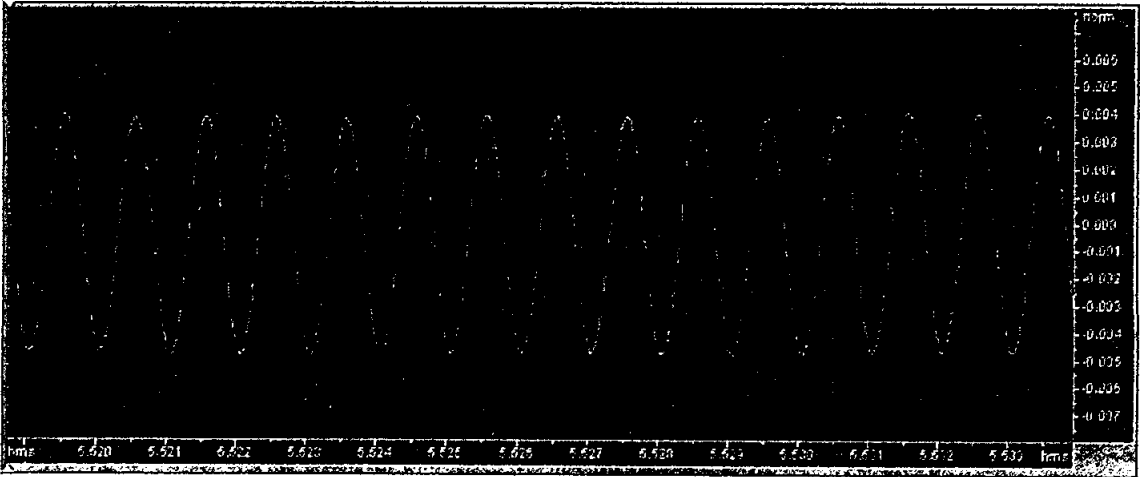
**Figura 4.10**

El valor de voltaje anotado del pico positivo fue de 0.25 V.



- Frecuencia de 1000 Hz

La visualización de la señal generada con frecuencia de 1000 Hz y con 51 dB de intensidad, consta en la figura 4.11:

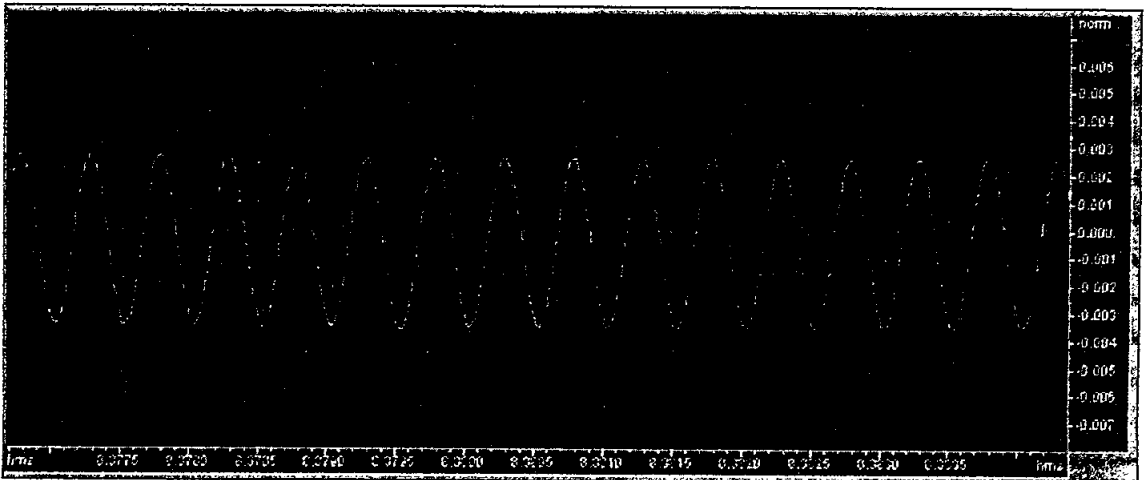


**Figura 4.11**

El valor de voltaje anotado del pico positivo fue de 0.004 V.

- Frecuencia de 2000 Hz

La visualización de la señal generada con frecuencia de 2000 Hz y con 48 dB de intensidad, consta en la figura 4.12:

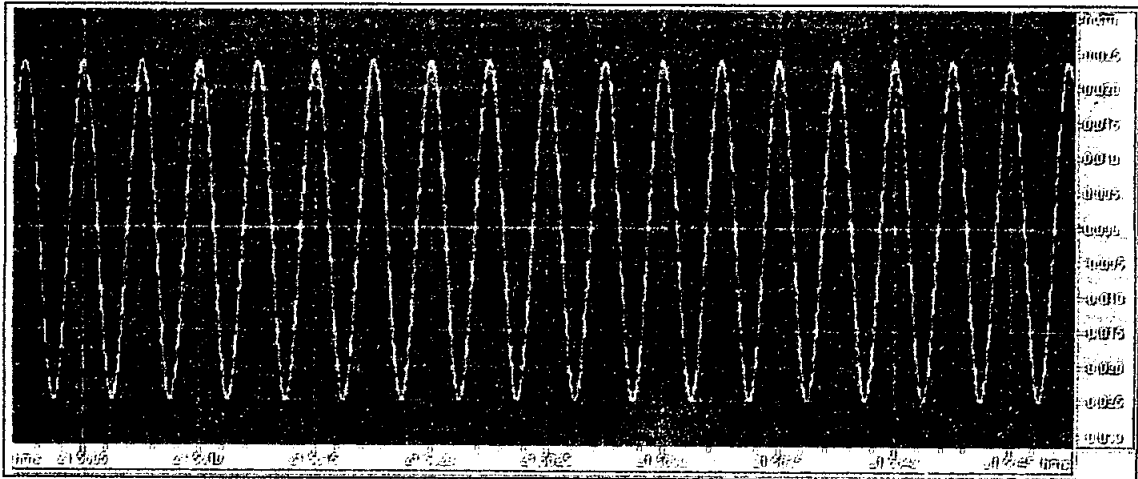


**Figura 4.12**

El valor de voltaje anotado del pico positivo fue de 0.0028 V.

- Frecuencia de 4000 Hz

La visualización de la señal generada con frecuencia de 4000 Hz y con 66 dB de intensidad, consta en la figura 4.13:

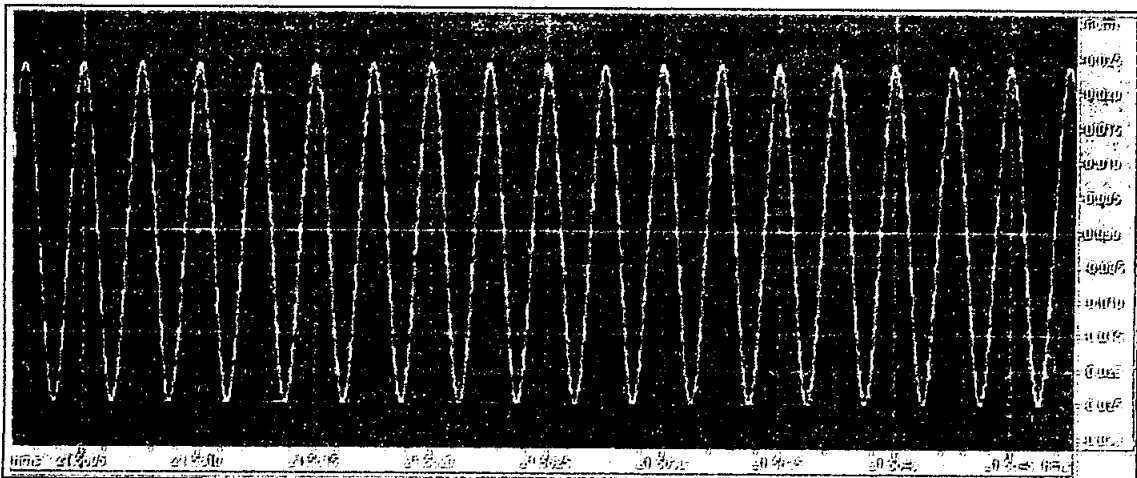


*Figura 4.13*

El valor de voltaje anotado del pico positivo fue de 0.022 V.

- Frecuencia de 8000 Hz

La visualización de la señal generada con frecuencia de 8000 Hz y con 55.5 dB de intensidad, consta en la figura 4.14:



*Figura 4.14*

El valor de voltaje anotado del pico positivo fue de 0.007 V.

#### 4.1.1.2.2 Ondas para evaluación del Oído Derecho

Para el oído derecho, se repitió el mismo procedimiento que para el izquierdo, y se obtuvieron los mismos valores de voltaje, así que, para continuar con el análisis, se asigna un valor de voltaje pico para cada intensidad predeterminada, sin importar si se trata del oído izquierdo o derecho.

Para hallar el voltaje de referencia, se lo despejó de la ecuación 4.1, obteniendo la ecuación 4.2:

$$V_{ref} = \frac{V_m}{\log^{-1} \frac{\# dB}{20}}$$

**Ecuación 4.2**

Aplicando esta ecuación, se halla el voltaje de referencia para cada uno de los valores anotados anteriormente, y los resultados de dichas operaciones se indican de manera explícita en la tabla 4.3:

Frecuencia Ideal [Hz]	Intensidad [dB]	Voltaje Pico [V]	Voltaje de Referencia [V]
125	94.5	0.59	$1.1113 * 10^{-5}$
250	90	0.36	$1.1384 * 10^{-5}$
500	87	0.25	$1.1167 * 10^{-5}$
1000	51	0.004	$1.1273 * 10^{-5}$
2000	48	0.0028	$1.1147 * 10^{-5}$
4000	66	0.022	$1.1026 * 10^{-5}$
8000	55.5	0.007	$1.1751 * 10^{-5}$

**Tabla 4.3**

Finalmente, y para una verificación más completa, se realizó el mismo proceso de asignación de intensidad y medición de voltaje pico correspondiente con un osciloscopio digital<sup>21</sup>.

<sup>21</sup> Instrumento disponible en el Laboratorio de Sistemas Digitales de la Escuela Politécnica Nacional

La tabla 4.4 muestra la tabulación los resultados de dicha medición:

Frecuencia Ideal [Hz]	Intensidad [dB]	Voltaje Pico [V]	Voltaje de Referencia [V]
125	94.5	0.586	$1.1038 * 10^{-5}$
250	90	0.355	$1.1226 * 10^{-5}$
500	87	0.243	$1.0854 * 10^{-5}$
1000	51	0.004	$1.1273 * 10^{-5}$
2000	48	0.003	$1.1943 * 10^{-5}$
4000	66	0.024	$1.2028 * 10^{-5}$
8000	55.5	0.007	$1.1751 * 10^{-5}$

**Tabla 4.4**

Es importante mencionar que los valores de intensidad asignados a cada tono, han sido escogidos al azar, para de esta forma hacer un análisis general de los valores de intensidad.

Además, se debe decir que las intensidades asignadas han sido relativamente altas ( todas mayores a 40 dB) ya que, como se puede observar, los valores de voltaje son muy pequeños, así que si se hubieran asignado intensidades menores, se habría dificultado la medición de los picos.

Se puede observar que, en el caso de las mediciones realizadas con el Cool Edit Pro, el valor que toma el voltaje de referencia es casi el mismo, y que la diferencia se nota en la séptima cifra decimal, lo cual corresponde al orden de décimas de micro voltios, lo cual indica que el error en la variación de la intensidad en dB es muy pequeño y puede ser considerado despreciable.

En el caso de las mediciones realizadas con el osciloscopio, se nota que la diferencia del valor del voltaje de referencia aparece en la sexta cifra decimal, lo cual corresponde al orden de micro voltios (pero en ningún caso se supera el valor de  $1 \mu\text{V}$ ), lo cual indica que el error en la variación de la intensidad en dB es también muy pequeño.

Las diferencia que toma el voltaje de referencia se debe a que se trabajó con valores de voltaje sumamente pequeños, razón por la cual se dificulta su exacta medición con los equipos de los cuales se dispone.

Si se toma la variación del voltaje de referencia como tolerable, se puede notar que existe una correspondencia entre la intensidad asignada, el voltaje pico medido y el voltaje de referencia calculado, ya que al ser prácticamente el mismo voltaje de referencia para todos los casos , se puede afirmar que la variación de la intensidad de los tonos generados, se ajusta a los requerimientos planteados al inicio del presente proyecto.

#### 4.1.2 ANALISIS MEDICO

En este análisis se realizó una comparación entre los audiogramas obtenidos con el AMCAEC I y con un audiómetro comercial.

El audiómetro comercial que fue tomado como referencia, es el equipo usado de manera diaria en el consultorio de un médico otorrinolaringólogo .

Dicho equipo es un Audiómetro Manual, marca Welch Allyn, modelo 232 , fabricado en Estados Unidos.

En cuanto a las características de los pacientes evaluados, éstas son especificadas en la tabulación de cada audiograma.

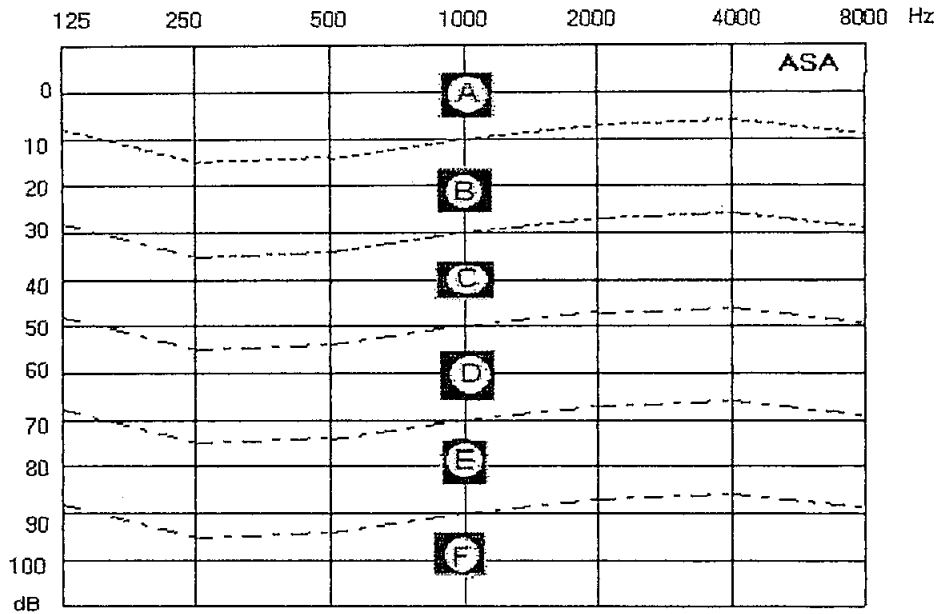
Las características que se deben tomar en cuenta son la edad y la ocupación, ya que según éstos dos parámetros, se produce la mayor diferencia en cuanto a los resultados obtenidos.

Se debe además tomar en cuenta dos aspectos muy importantes en cuanto a los resultados:

- Nivel de hipoacusia
- Tipo de hipoacusia

#### 4.1.2.1 Nivel de Hipoacusia

La figura 4.15 muestra la plantilla en la cual se traza el audiograma:



*Figura 4.15*

Se nota claramente la presencia de 6 franjas, las cuales han sido denominadas con letras. Cada una de estas franjas representa un nivel de hipoacusia, de la manera siguiente:

- A:** Paciente con excelente condición auditiva
- B:** Paciente con buena condición auditiva
- C:** Paciente con leve afectación auditiva
- D:** Paciente con grave afectación auditiva
- E:** Paciente con severa afectación auditiva
- F:** Paciente sordo

#### **4.1.2.2 Tipo de Hipoacusia**

Tal como se puede apreciar en la figura 4.15, se nota que la frecuencia de 1000 Hz es la intermedia, ya que existen 3 frecuencias mayores a ésta y 3 menores.

En caso de que la deficiencia auditiva (el paciente escucha a mayor intensidad) aparezca en las 3 frecuencias inferiores, se tiene un caso de hipoacusia de conducción, y en caso de que la deficiencia se note en las 3 frecuencias superiores, se tiene un caso de hipoacusia neuro sensorial. La explicación más detallada de éstas dos patologías aparece de manera explícita en el capítulo I

Por lo mencionado anteriormente, es necesario que para que los dos audiogramas comparados sean considerados iguales desde el punto de vista médico, las curvas de ambos deben estar dentro de la misma franja y además tener la misma forma o tendencia.

#### **4.1.2.3 Tabulación de Audiogramas**

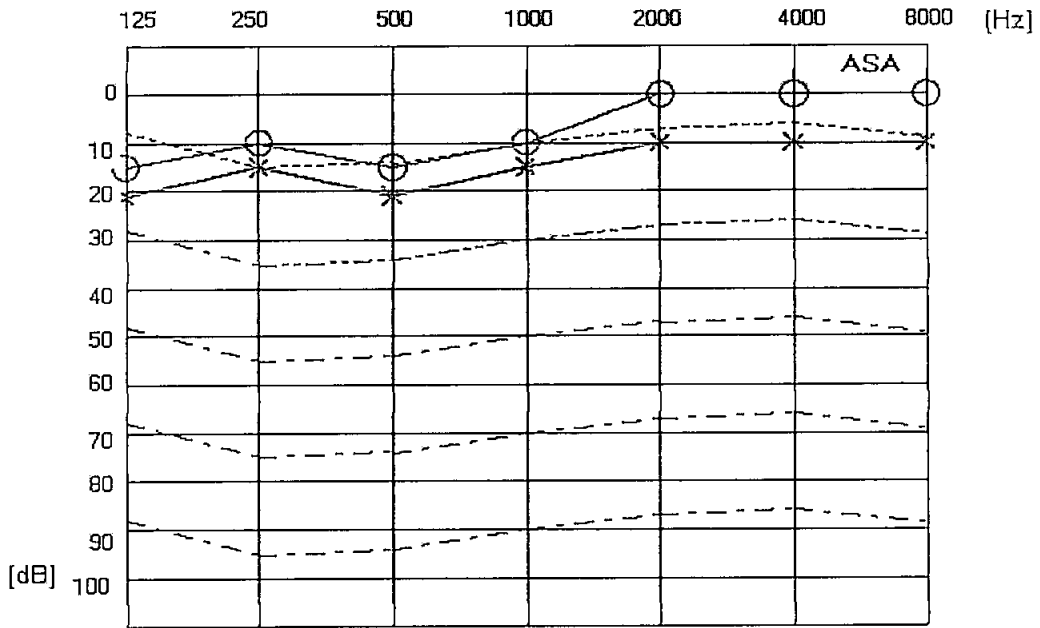
Es importante mencionar que, para facilitar la visualización y comparación entre los dos audiogramas, se dibujó a ambos con el software del AMCAEC I. Sin embargo, se debe anotar que los exámenes originales realizados con el equipo comercial constan en el anexo 13.

• **Paciente 1**

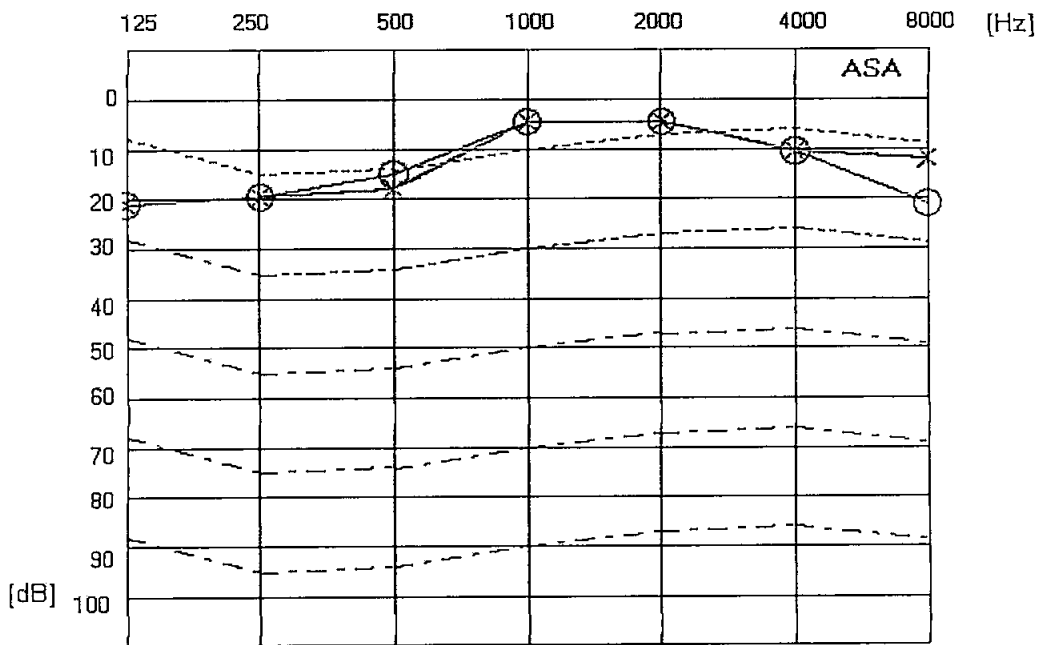
Nombre: Grace Vallejo

Edad: 40 años

Ocupación: Oficinista en empresa de seguros



**Audímetro Comercial**



**AMCAEC I**

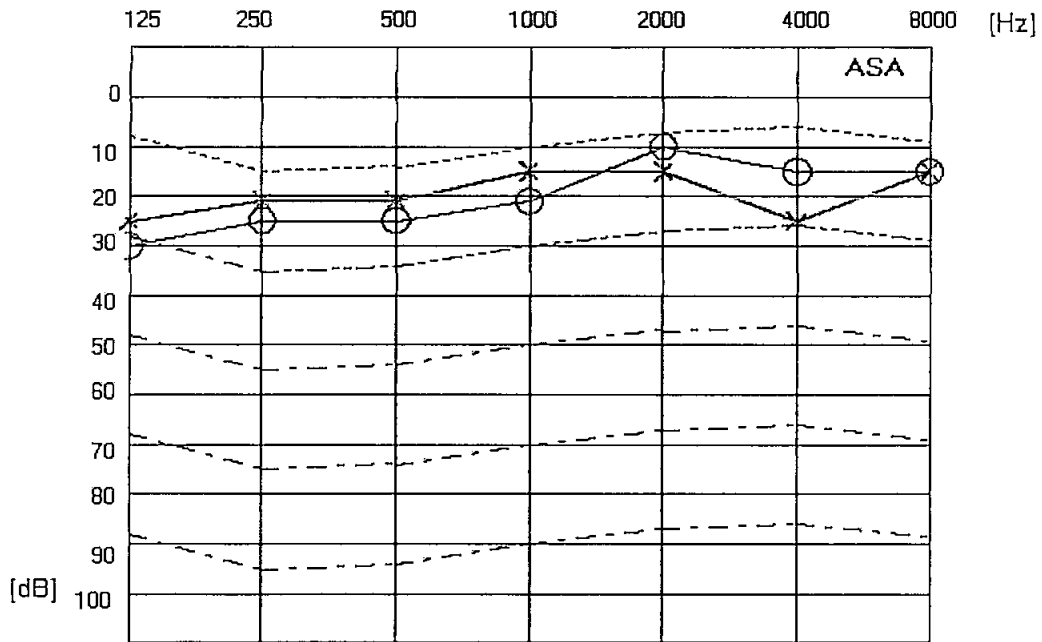


• **Paciente 2**

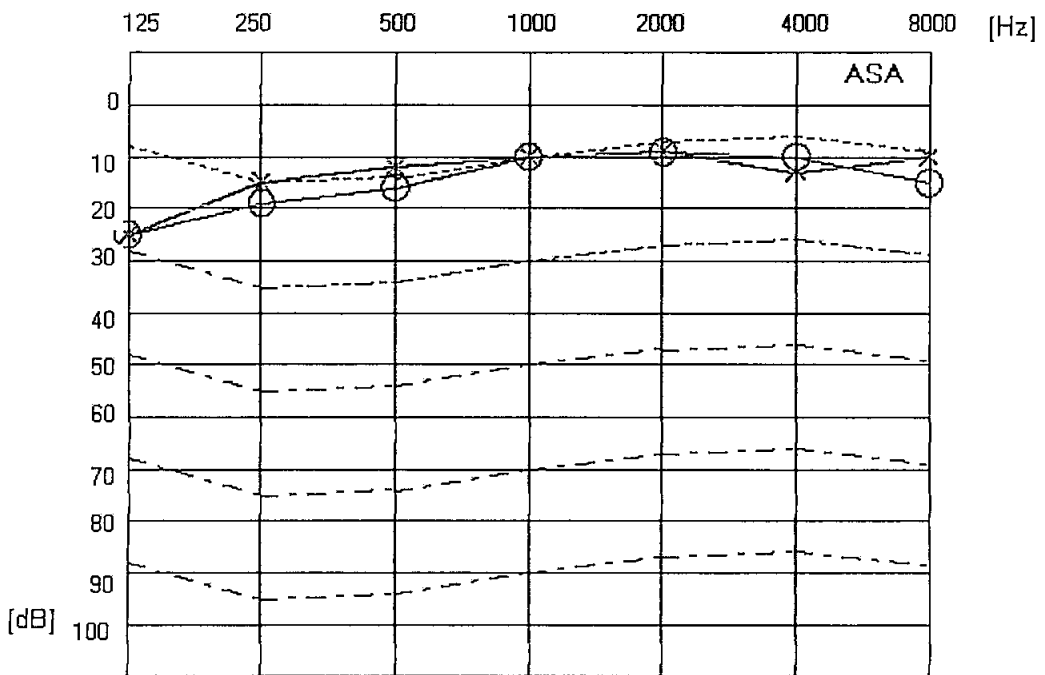
Nombre: Carmen Miranda

Edad: 33 años

Ocupación: Secretaria de un colegio secundario



**Audímetro Comercial**



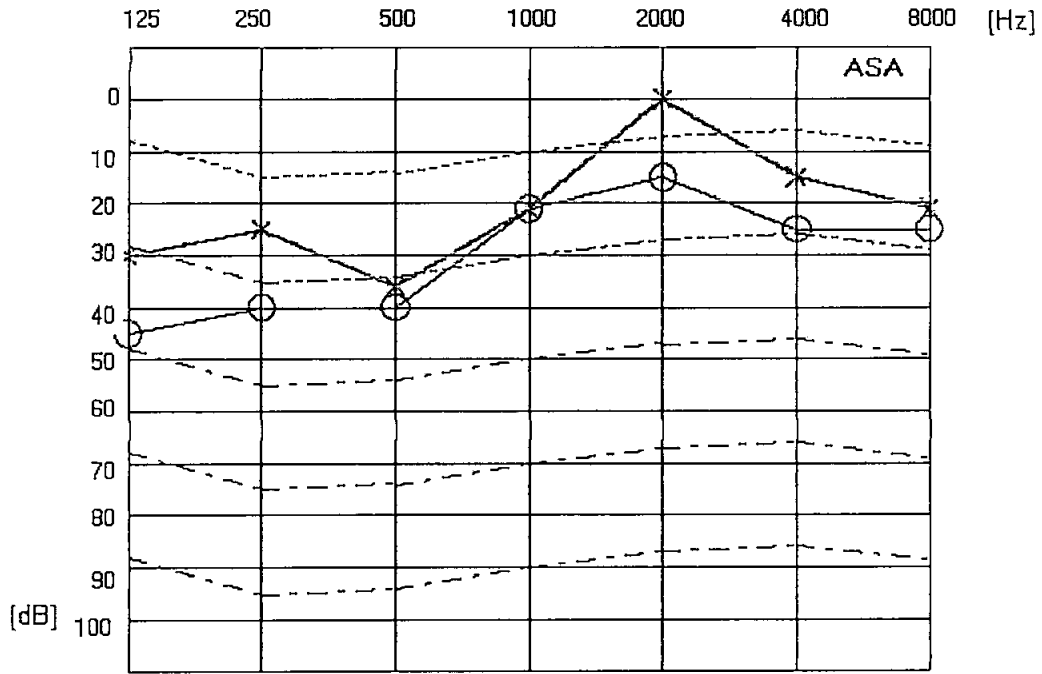
**AMCAEC I**

• **Paciente 3**

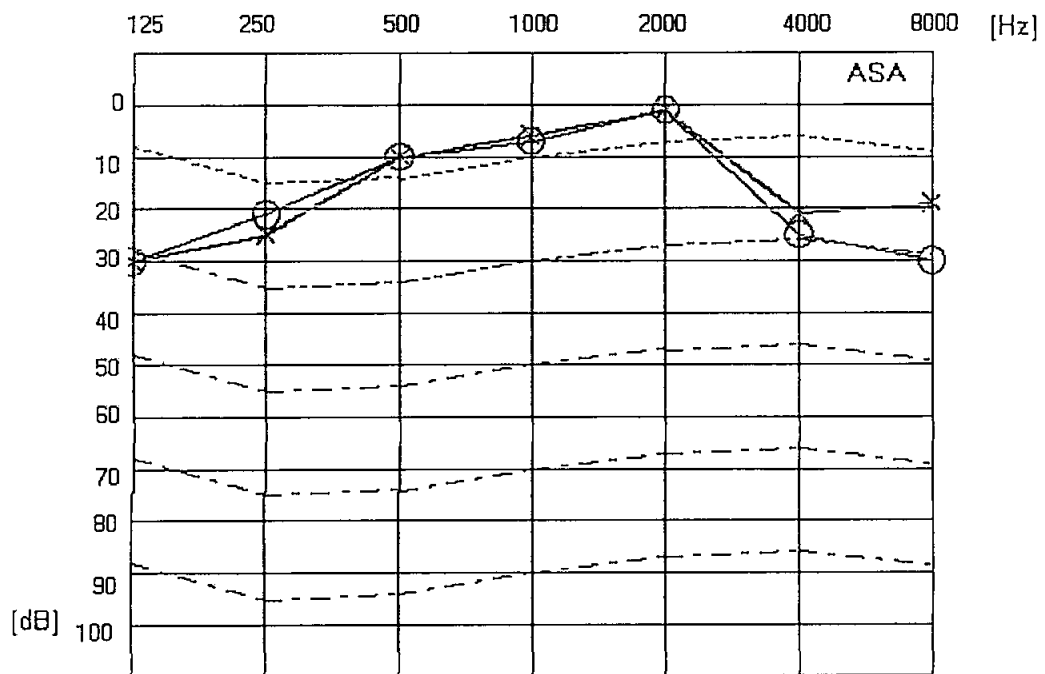
Nombre: Raúl Barragán C.

Edad: 52 años

Ocupación: Docente universitario



**Audímetro Comercial**



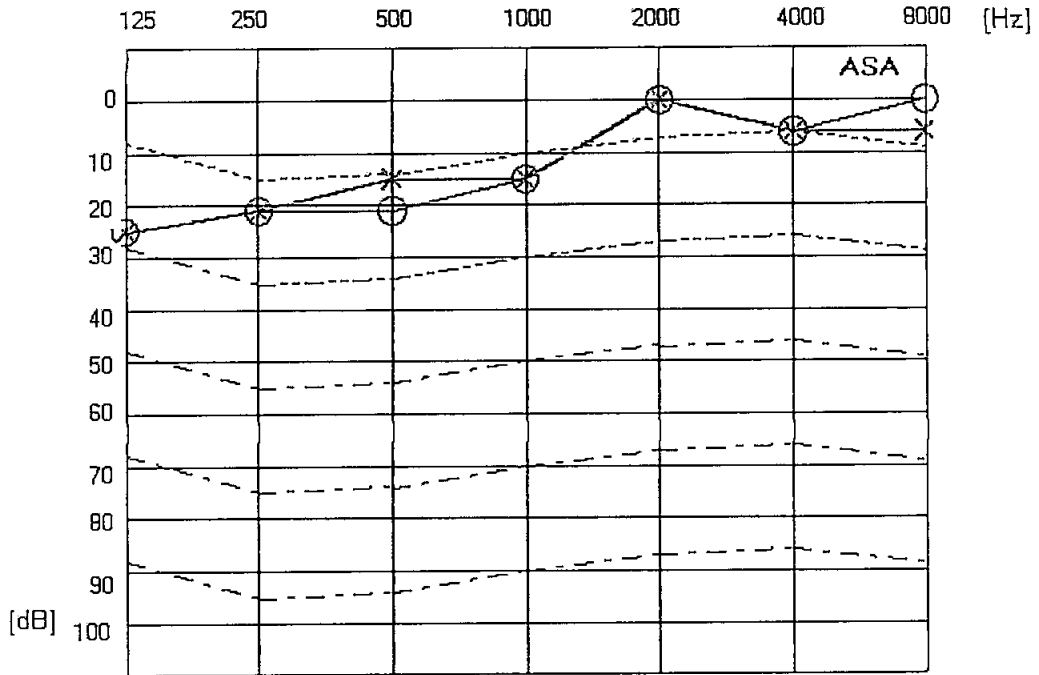
**AMCAEC I**

• **Paciente 4**

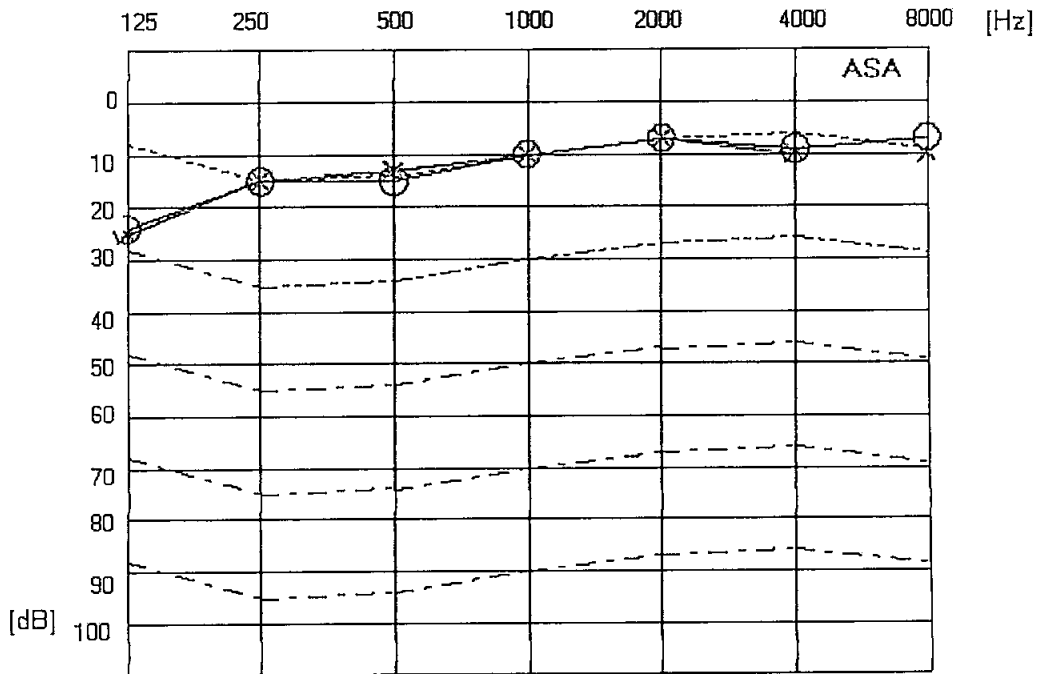
Nombre: Antonio Vallejo

Edad: 27 años

Ocupación: Docente de secundaria



**Audímetro Comercial**



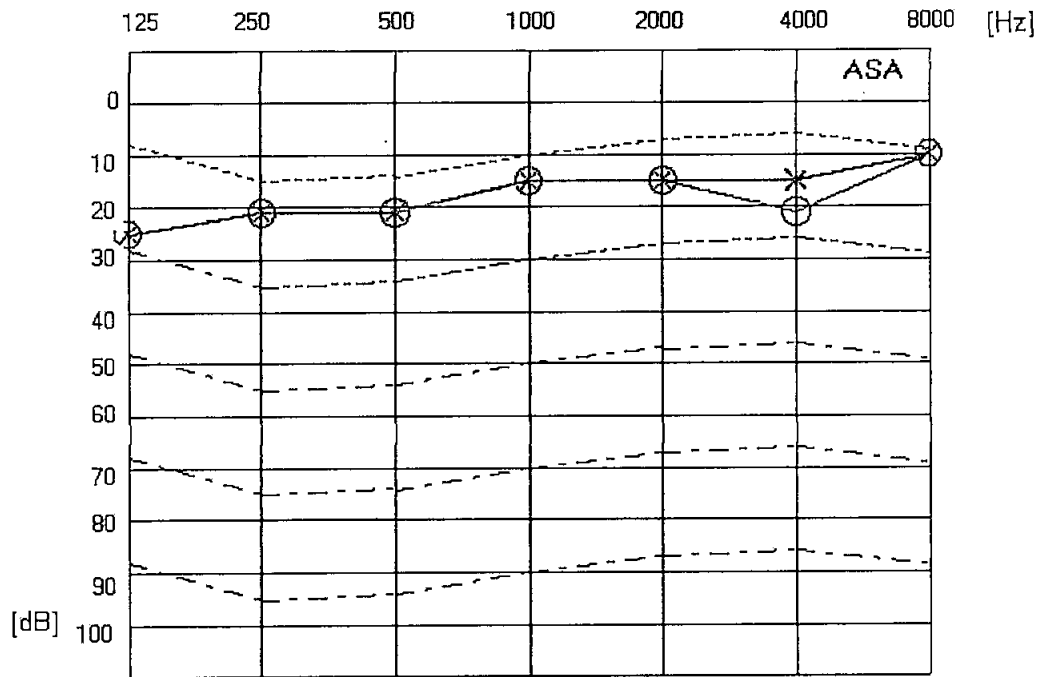
**AMCAEC I**

• **Paciente 5**

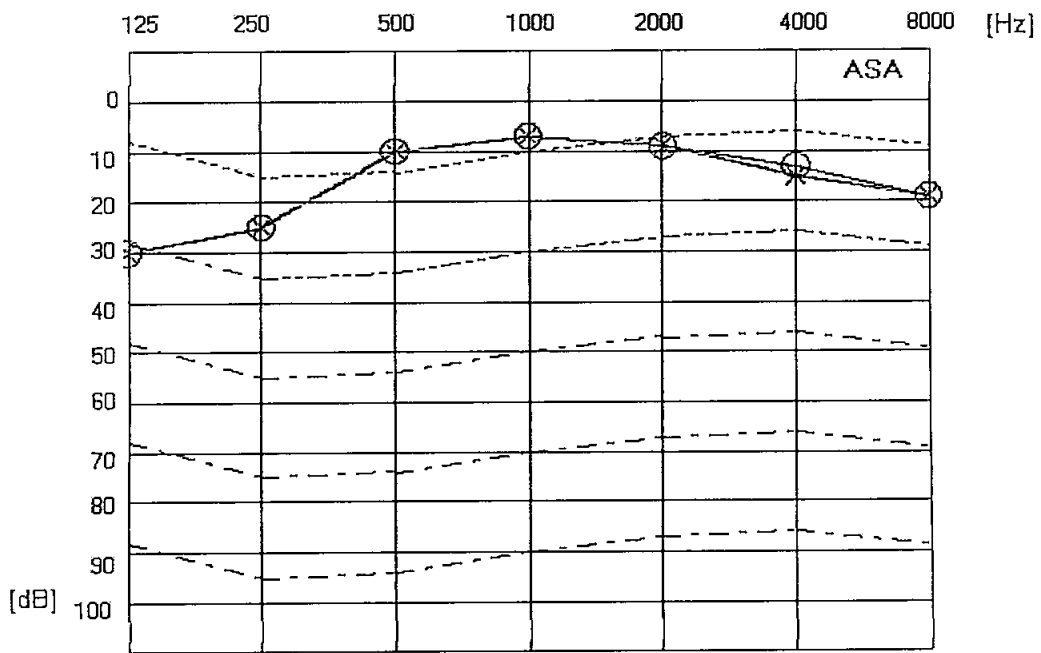
Nombre: Belén Miranda

Edad: 31 años

Ocupación: Ama de casa



**Audímetro Comercial**



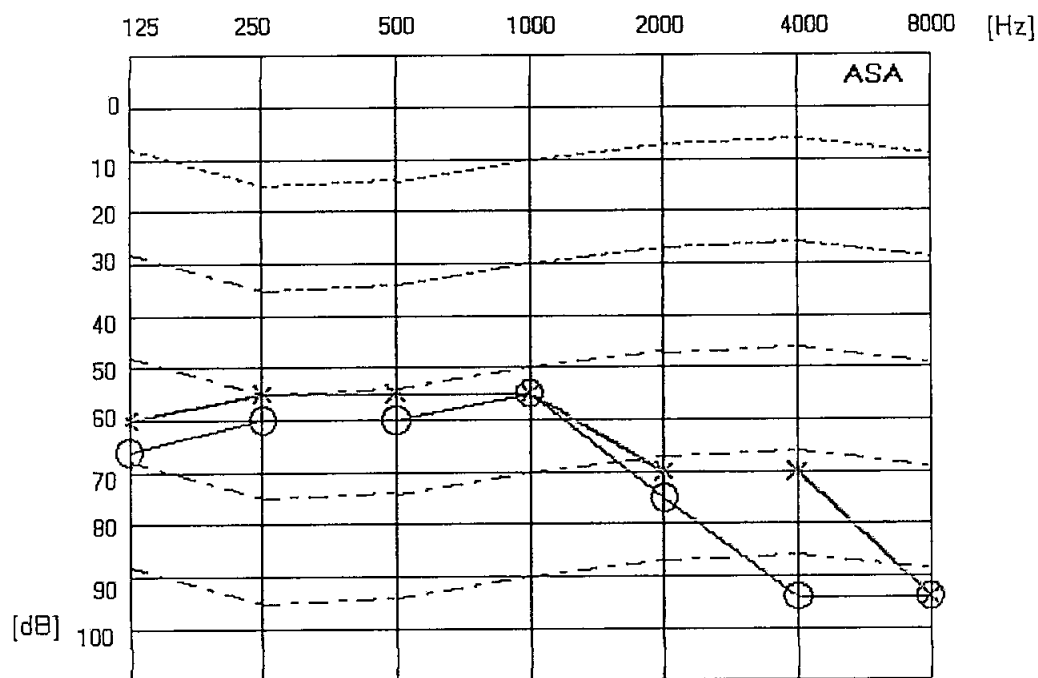
**AMCAEC I**

• **Paciente 6**

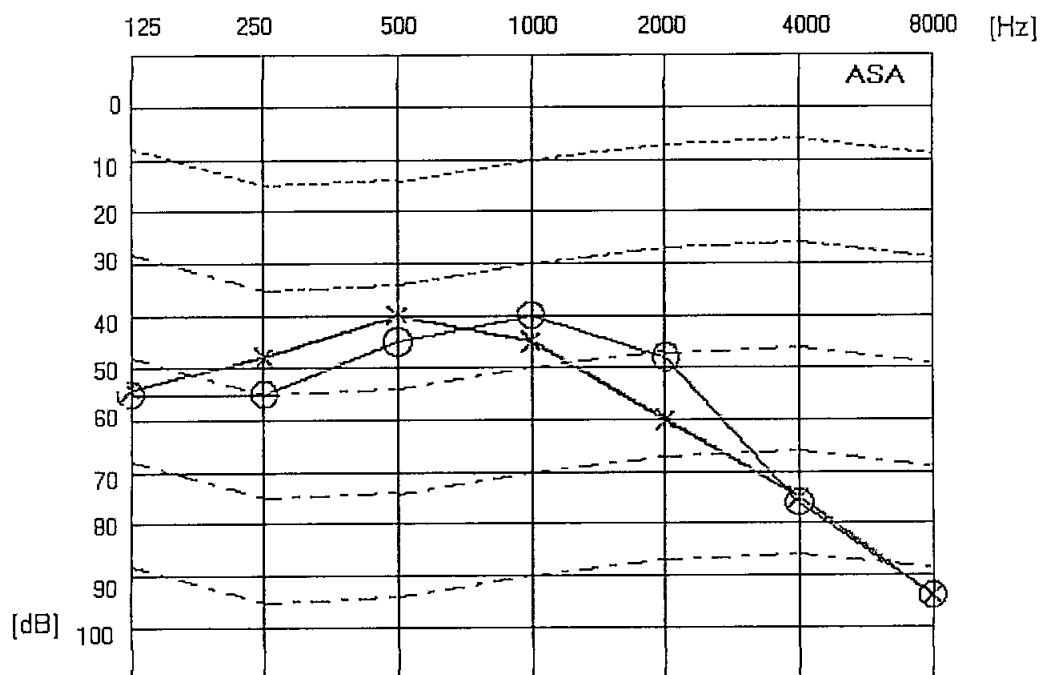
Nombre: Carmela Bustos

Edad: 90 años

Ocupación: Ama de casa



**Audímetro Comercial**



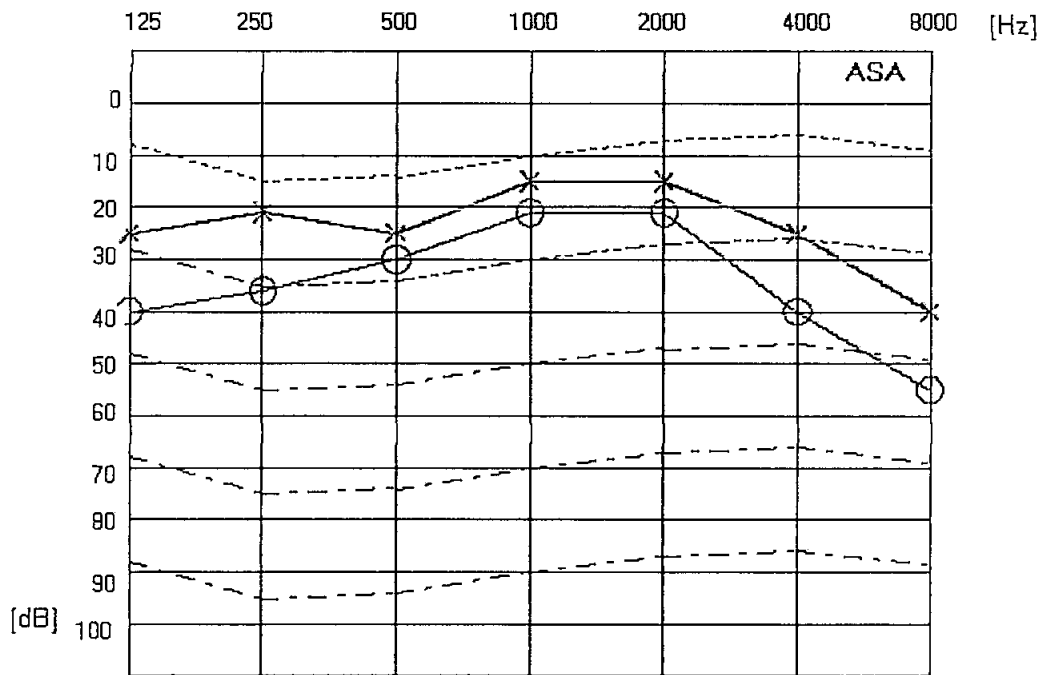
**AMCAEC I**

• **Paciente 7**

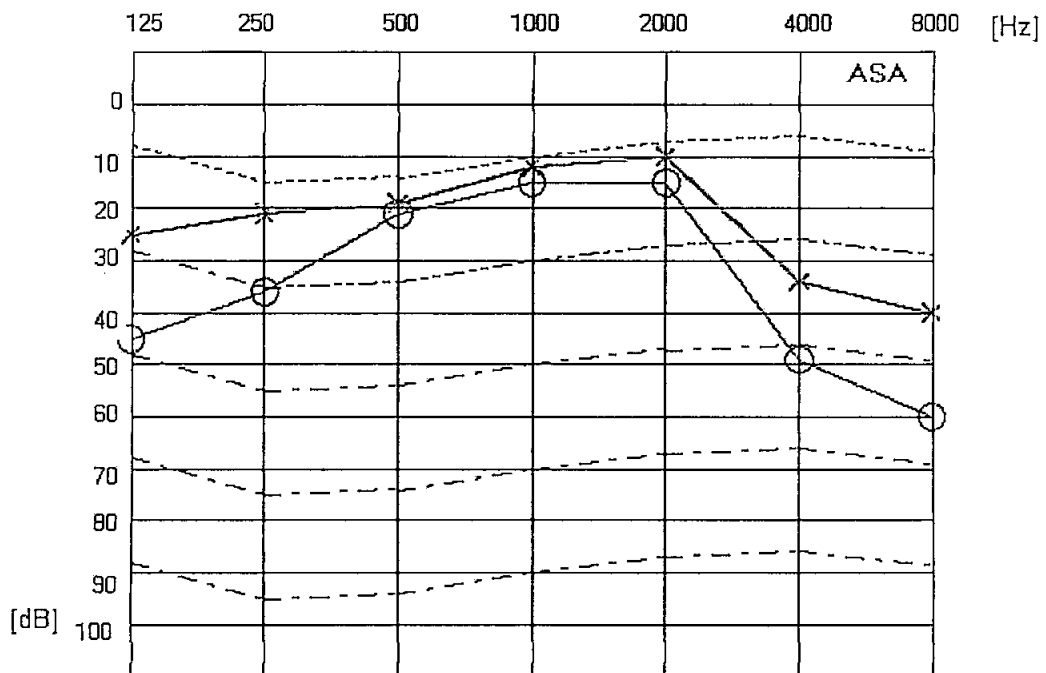
Nombre: Ana Barragán

Edad: 21 años

Ocupación: Estudiante



**Audímetro Comercial**



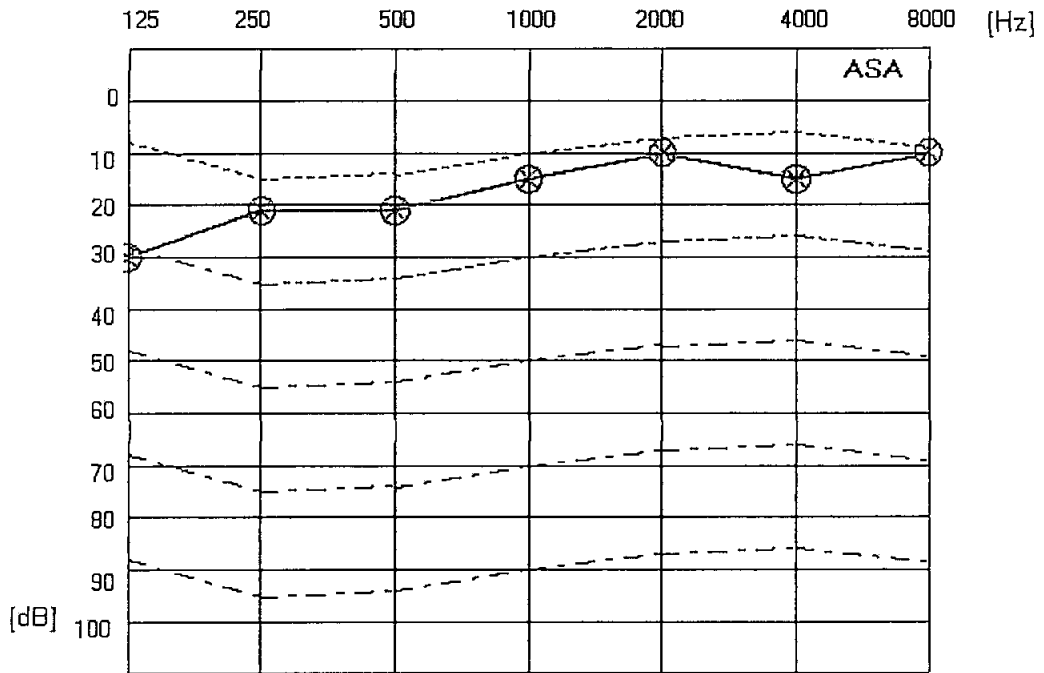
**AMCAEC I**

• **Paciente 8**

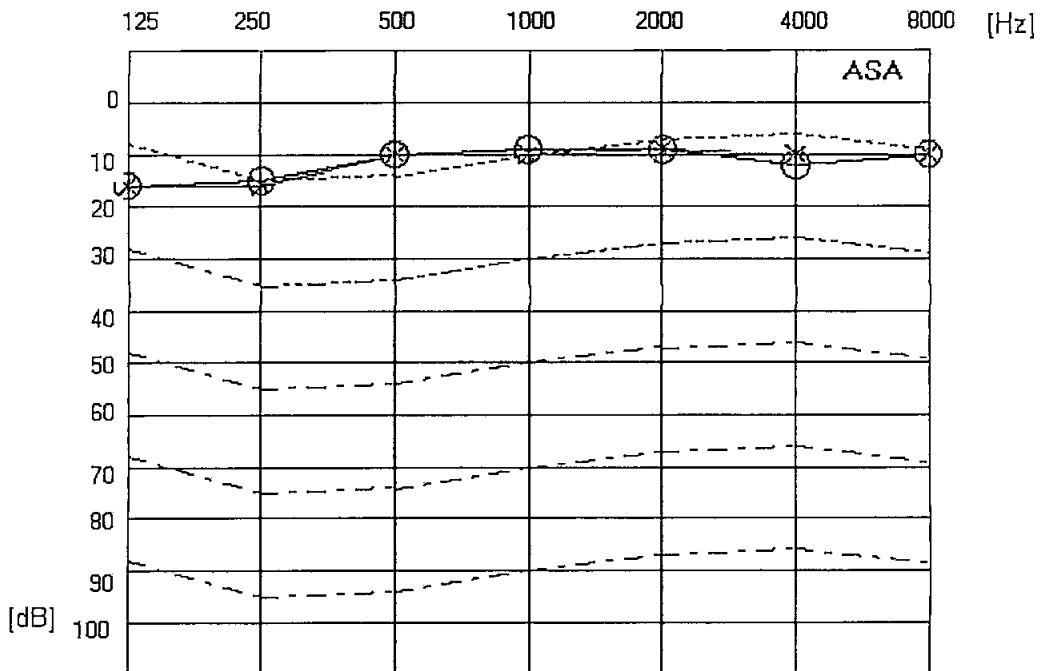
Nombre: Raúl Barragán

Edad: 24 años

Ocupación: Estudiante



**Audímetro Comercial**



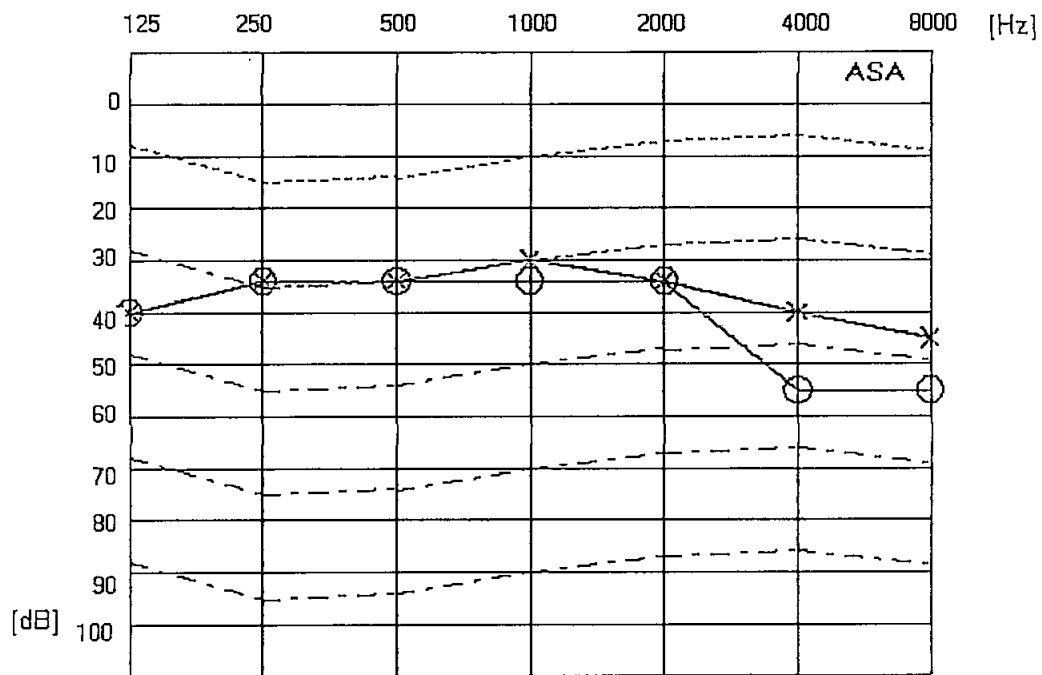
**AMCAEC I**

• **Paciente 9**

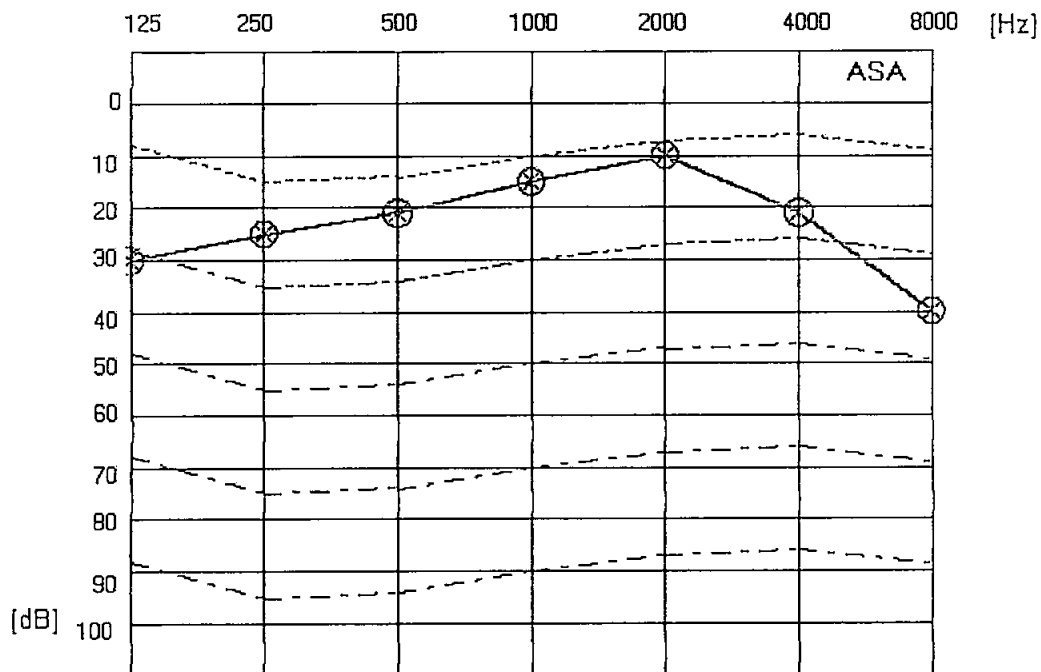
Nombre: Carmen Gaibor

Edad: 54 años

Ocupación: Ama de casa



**Audímetro Comercial**



**AMCAEC I**



Tal como se puede apreciar en cada una de las gráficas anteriores, no existe una diferencia notoria, en cuanto a nivel y tipo de hipoacusia, entre los audiogramas obtenidos con el equipo comercial y los obtenidos con el AMCAEC I

Es importante mencionar que todo fenómeno externo produce alteraciones en cuanto a los resultados obtenidos. Así pues, afecta el ruido exterior, la hora del día, el estado de ánimo del paciente, su nivel de concentración, etc.

Tomando en cuenta ésta última consideración, se debe ser tolerante en cuanto a las diferencias obtenidas.

### 4.1.3 ANALISIS ESTADÍSTICO

#### 4.1.3.1 Distribución Normal

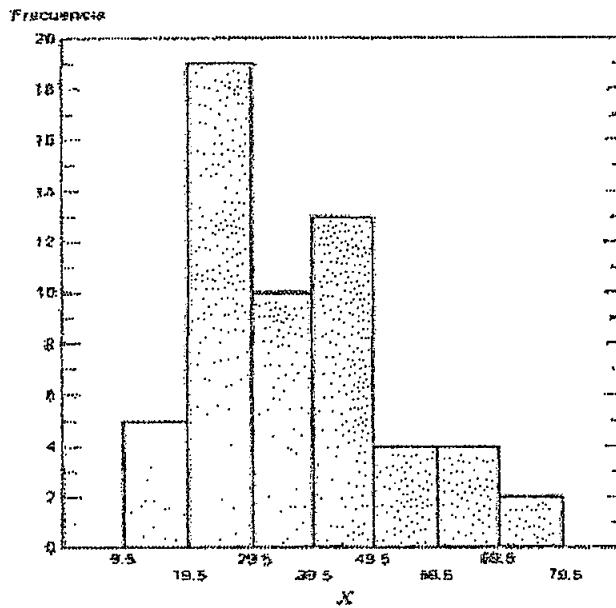
Un histograma es un gráfico que resulta de la representación de la distribución de frecuencias de una variable determinada. Al construir un histograma, los valores de la variable en consideración constituyen el eje horizontal, mientras que el eje vertical tiene como escala la frecuencia de ocurrencia.

Cabe aclarar que la variable en consideración es aquella que se está midiendo y su valor depende de del individuo en el cual se la evalúa.

Para agrupar a un conjunto de observaciones se selecciona un conjunto de intervalos contiguos que no se traslapen, de modo que cada valor en el conjunto de observaciones pueda colocarse en uno y sólo uno de los intervalos. Estos intervalos se conocen como intervalos de clase. Además se debe decidir la amplitud de los intervalos.

Por encima de cada intervalo de clase sobre el eje horizontal se levanta una barra rectangular o celda, como a veces se conoce, de modo que su altura corresponda con la frecuencia correspondiente. Las celdas de un histograma deben quedar unidas para evitar que queden espacios entre ellas.

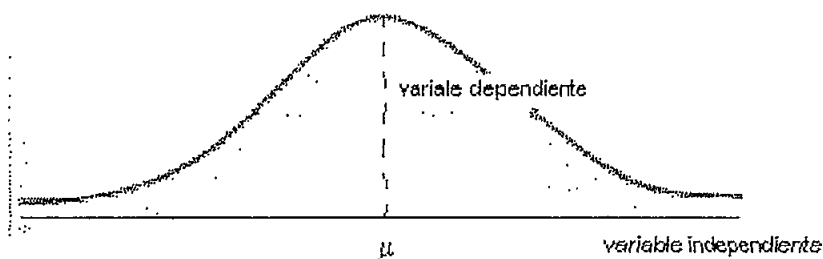
La figura 4.16 muestra un ejemplo de histograma:



**Figura 4.16: Histograma de los pesos en onzas de los tumores maligno extirpados del abdomen de 57 personas**

Cuando se tienen un alto número de intervalos de clase, el tamaño de la celda se hace más pequeño y al unir los puntos medios de cada celda, se obtiene una curva con forma de campana, a la que comúnmente se le denomina campana de Gauss.

Esto se puede apreciar en la figura 4.17:



**Figura 4.17: Campana de Gauss**

La distribución normal es una distribución de probabilidad que es utilizada para encontrar el área entre dos puntos cualquiera que proviene de la construcción de un histograma cuyos puntos están tan cercanos que toman la forma de una

curva acampanada normalmente conocida como la campana de Gauss, cuyo autor es Carl Friedrich Gauss (1777-1855).

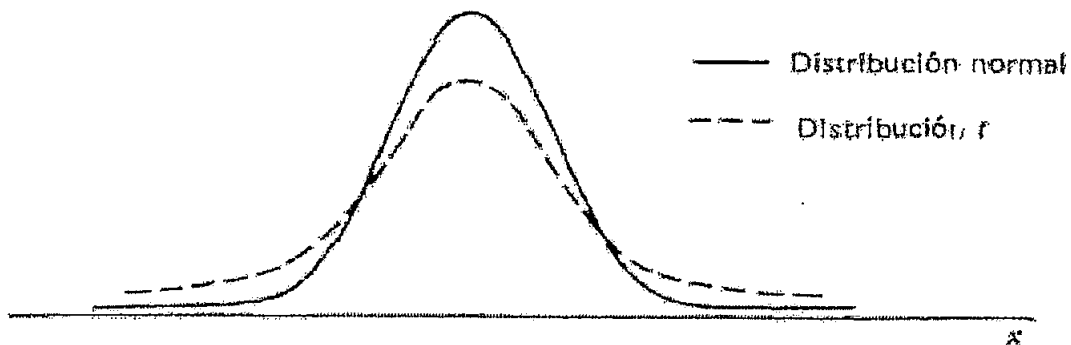
Dicha campana incluye el 100% de los valores observados, lo que equivale a un valor de 1 ; esta campana es simétrica y la mitad tiene un área positiva y la otra mitad un área negativa.

#### 4.1.3.2 Distribución de t

Cuando el número de datos es pequeño, se sigue todo el mismo proceso anterior y se obtiene la distribución de t, la misma que es considerada como una hermana menor de la distribución normal.

Esta curva tiene las mismas características de la campana de Gauss, pero es menos puntiaguda en el centro y tiene las colas más altas.

La figura 4.18 muestra la diferencia de las curvas normal y de t.



**Figura 4.18: Comparación de las distribuciones normal y t**

La distribución de t es utilizada para muestras pequeñas en las cuales cada observación tiene un grado de libertad y a medida que se cambian dichos grados de libertad , y por ende las observaciones, el valor de t calculada ( de la cual se habla explícitamente más adelante) también se modifica, lo que no sucede cuando se trabaja con poblaciones grandes en las cuales la distribución normal no tiene ninguna variación si el número de observaciones se altera.

En forma práctica y resumida, un grado de libertad es dado a cada observación, ya sean estos individuos, cosas, etc. con las que se trabaja y un grado de libertad corresponde a una observación. Así pues, si se tiene 10 observaciones, se tendría 10 grados de libertad. Sin embargo, dado que se

está trabajando con una muestra muy pequeña y que se está tratando de estimar a la población se debe siempre considerar, para el cálculo matemático, un grado de libertad menos que el total de las observaciones ( $n-1$ ).

Cuando el investigador tiene como propósito tomar una decisión en torno a una población, examinando una muestra de ella, recurre a las pruebas de hipótesis, que no son más que una afirmación acerca de una o más poblaciones, lo cual permite llegar a una conclusión con respecto a un hecho o característica de algo o alguien

Un investigador trata siempre con dos tipos de hipótesis: las hipótesis de investigación y la hipótesis estadística. La hipótesis de investigación es la conjetura o suposición que motiva la investigación y ésta conduce directamente a la hipótesis estadística, la misma que se establece en tal forma que pueda ser evaluada a través de técnicas estadísticas apropiadas. Un caso de esto es la prueba de  $t$ .

Cabe mencionar que la hipótesis de investigación es cualitativa y la hipótesis estadística es cuantitativa.

#### 4.1.3.3 Prueba de $t$

Esta prueba es utilizada para comparar dos tratamientos y en ella se plantean 2 tipos de hipótesis: la hipótesis nula  $H_0$  y la hipótesis alternativa  $H_a$ .

La primera es la de la no diferencia entre dos tratamientos, es decir, que los dos tratamientos son iguales estadísticamente hablando.

La segunda es la de la diferencia entre dos tratamientos, es decir, que los tratamientos son diferentes estadísticamente hablando.

En toda prueba de hipótesis es necesario considerar un valor de  $\alpha$  que es el nivel de seguridad que se tiene sobre la afirmación o negación de una hipótesis y se recomienda trabajar con dos niveles: al 5 % y al 1%.

Cuando se habla del 5% se quiere decir que se tiene la probabilidad de afirmar una hipótesis con el 95 % de probabilidades de estar en lo cierto y 5 % de estar en el error; en el otro caso, se tiene la probabilidad de afirmar con el 99% de probabilidad de estar en lo cierto y 1% de error.

Se entiende por lo tanto, que la prueba de t sirve para comparar dos tratamientos, los mismo que pueden ser de dos tipos:

- t sorteada: para grupos independientes
- t pareada: para grupos dependientes

#### 4.1.3.3.1 t sorteada

Cuando los datos provienen de poblaciones distintas, se procede a utilizar la prueba de t sorteada o para grupos independientes. Para este caso, a cada población se le asigna n -1 grados de libertad, y el total de grados de libertad corresponde a la suma de ambos (n1-1)+(n2-1).

La fórmula para el cálculo de t, viene dada por la ecuación 4.3

$$t_c = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{S_c^2 \frac{n_1 + n_2}{n_1 * n_2}}}$$

**Ecuación 4.3**

Donde:

$\bar{x}_1$  : media de la población 1

$\bar{x}_2$  : media de la población 2

n1: número de datos de la muestra 1

n2: número de datos de la muestra 2

Sc<sup>2</sup> : suma de cuadrados de las poblaciones, y:

$$S_c^2 = \frac{\left[ \frac{\sum x_1^2 - (\sum x_1)^2}{(n_1 - 1)} \right] + \left[ \frac{\sum x_2^2 - (\sum x_2)^2}{(n_2 - 1)} \right]}{(n_1 - 1) + (n_2 - 1)}$$

Donde:

$\sum x_1^2$  :Suma de cuadrados de los datos de la población 1

$(\sum x_1)^2$  : Cuadrado de la suma de los datos de la población 1

$\sum x_2^2$  : Suma de cuadrados de los datos de la población 2

$(\sum x_2)^2$  : Cuadrado de la suma de los datos de la población 2

#### 4.1.3.3.2 *t* pareada

Cuando los datos provienen de dos condiciones o tratamientos que son comparados en individuos, objetos o cosas en los que se trata de medir si el tratamiento uno tiene el mismo efecto que el tratamiento dos pero en el mismo individuo, objeto o cosa, se recurre a la prueba de *t* pareada o de grupos dependientes. En este caso, el un dato depende del otro en su comparación y por lo tanto a cada observación se le da un grado de libertad y, en esta situación, los grados de libertad totales corresponden a  $n-1$ .

La fórmula para el cálculo de *t*, viene dada por la ecuación 4.4

$$t_c = \frac{\overline{xd_i}}{\sqrt{\frac{\sum d_i^2 - \frac{(\sum d_i)^2}{n}}{n(n-1)}}$$

**Ecuación 4.4**

Donde:

$\overline{xd_i}$  : media de las diferencias

$\sum d_i^2$  : Suma de los cuadrados de las diferencias

$(\sum d_i)^2$ : Cuadrado de la suma de las diferencias

$n$  = Número de pares

En el caso de la presente investigación, se utilizó la prueba de *t* pareada, dado que lo que se quería comprar es si el equipo de referencia arroja los mismos resultados que el AMCAEC I desde el punto de vista estadístico.

#### 4.1.3.4 Análisis de Datos

Para analizar los datos, se los debe agrupar según niveles establecidos, y en el caso de este estudio, se los ha agrupado según cada tono y oído, con el fin de determinar si existe diferencia significativa entre la medición de la intensidad a

la que cada paciente escuchó cada tono en cada equipo. Por ende, se obtuvieron 14 niveles.

La tabla 4.5 muestra un ejemplo de como se agruparon los datos:

Nivel 1: Oído Derecho, 125 Hz

PERSONA	AUDIMETRO COMERCIAL	AMCAEC I
Persona 1	40	45
Persona 2	30	16
Persona 3	40	30
Persona 4	65	55
Persona 5	25	30
Persona 6	25	24
Persona 7	45	30
Persona 8	30	25
Persona 9	15	21

**Tabla 4.5**

En la segunda columna, consta la intensidad a la cual cada paciente escuchó el tono indicado, con el oído indicado en el audímetro comercial.

En la tercera columna, consta la intensidad a la cual cada paciente escuchó el tono indicado, con el oído indicado en el AMCAEC I.

Una vez hecho esto, se procede a calcular el valor de  $t$  para cada nivel aplicando la ecuación 4.4:

$$t_c = - \frac{\overline{xd_i}}{\sqrt{\frac{\sum d_i^2 - \frac{(\sum d_i)^2}{n}}{n(n-1)}}$$

La tabla 4.6 muestra la obtención de los parámetros que intervienen en la ecuación 4.4:

Nivel 1: Oído Derecho, 125 Hz

PERSONA	AUDIMETRO COMERCIAL	AMCAEC I	d <sub>i</sub>	d <sub>i</sub> <sup>2</sup>
Persona 1	40	45	-5	25
Persona 2	30	16	14	196
Persona 3	40	30	10	100
Persona 4	65	55	10	100
Persona 5	25	30	-5	25
Persona 6	25	24	1	1
Persona 7	45	30	15	225
Persona 8	30	25	5	25
Persona 9	15	21	-6	36
Sumatoria			39	733
(Sumatoria) <sup>2</sup>			1521	
xdi			4,33	

**Tabla 4.6**

Reemplazando valores, se obtiene lo siguiente:

$$t_c = - \frac{\overline{xdi}}{\sqrt{\frac{\sum di^2 - \frac{(\sum di)^2}{n}}{n(n-1)}}} = - \frac{4.33}{\sqrt{\frac{733 - \frac{1521}{9}}{9(8)}}} = -1.547$$

Luego se debe recurrir a una tabla, la misma que consta en el anexo 14; y, según el número de grados de libertad, se debe buscar el valor con el cual se debe comparar el valor de la t calculada.

Como la hipótesis planteada consistía en conocer si había diferencias entre el equipo 1 y el equipo 2, se utilizó la prueba de dos colas, en la que no importa el valor del signo de la t calculada (positivo o negativo) ya que el área del valor de t se divide en 2 partes iguales.

Dado que el número de personas evaluadas fue de 9, se tienen 8 grados de libertad.



El valor de tabla es 2.3060 para el 5% y 3.3554 para el 1%, para todos los niveles (ya que los grados de libertad son siempre los mismos)

Luego se debe comparar la t calculada con el valor de tabla:

Para hacer esta comparación se debe considerar los dos siguientes aspectos:

1. Si el valor calculado es mayor que el valor de t de tabla, la diferencia es significativa, y por lo tanto se rechaza la hipótesis nula, pues, los dos equipos son diferentes.
2. Si el valor calculado es menor que el valor de t de tabla, la diferencia es no significativa, y por lo tanto se acepta la hipótesis nula, pues, los dos equipos pueden considerarse iguales.

Para cada caso se repite el mismo proceso, considerando siempre 8 grados de libertad, y por lo tanto obteniendo un valor de t calculada para cada nivel.

#### 4.1.3.5 Análisis de Resultados

Los resultados obtenidos en la presente investigación para cada uno de los niveles, se presentan a continuación.

##### • Oído izquierdo

NIVEL	t calculada	t al 5%	t al 1%	
Oído izquierdo, 125 Hz	1,32	2,306	3,3554	ns
Oído izquierdo, 250 Hz	1,45	2,306	3,3554	ns
Oído izquierdo, 500 Hz	4,26	2,306	3,3554	**
Oído izquierdo, 1000 Hz	5,89	2,306	3,3554	**
Oído izquierdo, 2000 Hz	1,89	2,306	3,3554	ns
Oído izquierdo, 4000 Hz	0,36	2,306	3,3554	ns
Oído izquierdo, 8000 Hz	1,03	2,306	3,3554	ns

• **Oído Derecho**

NIVEL	t calculada	t al 5%	t al 1%	
Oído Derecho, 125 Hz	1,54	2,306	3,3554	ns
Oído Derecho, 250 Hz	1,49	2,306	3,3554	ns
Oído Derecho, 500 Hz	4,09	2,306	3,3554	**
Oído Derecho, 1000 Hz	5,43	2,306	3,3554	**
Oído Derecho, 2000 Hz	1,99	2,306	3,3554	ns
Oído Derecho, 4000 Hz	0,29	2,306	3,3554	ns
Oído Derecho, 8000 Hz	1,94	2,306	3,3554	ns

Luego de haber realizado los cálculos estadísticos, se llega a determinar que existe diferencia significativa al 1%, tanto en el oído derecho como en el izquierdo en la frecuencias de 500 y 1000 Hz.

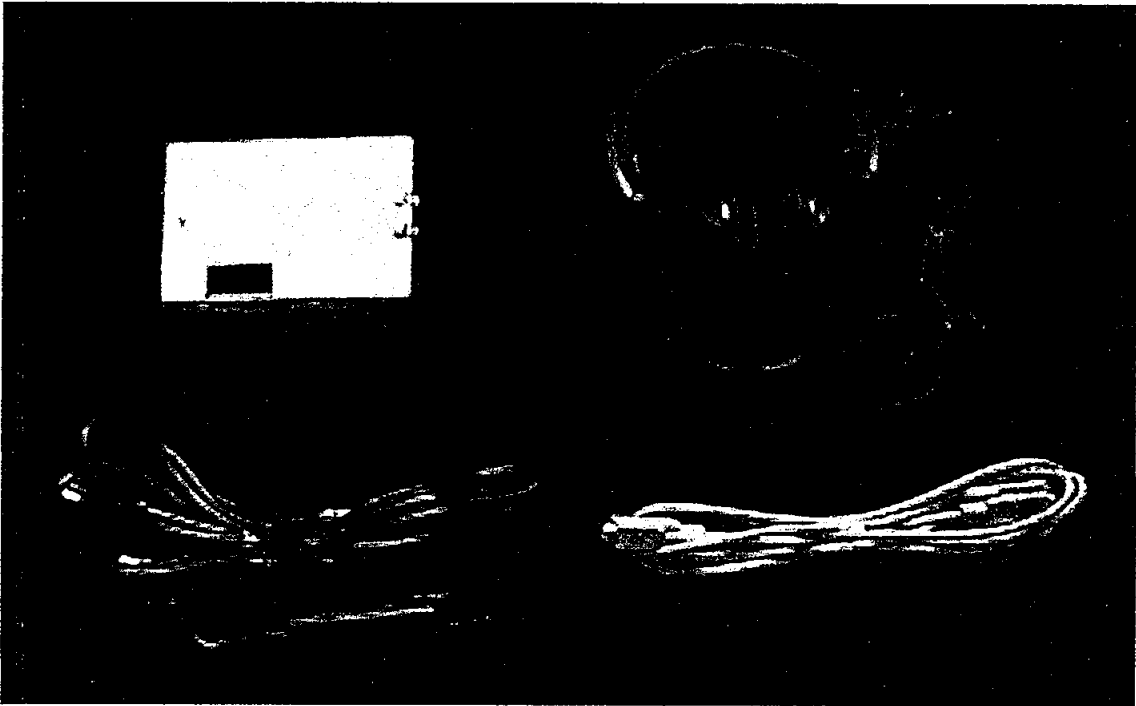
En cambio, la diferencia es no significativa para los dos oídos para el resto de niveles.

Esto quiere decir que, en general, ambos equipos tienen el mismo comportamiento para dos niveles no significativos. Sin embargo, son diferentes para los niveles significativos, lo cual podría deberse a que cuando se realizaron las evaluaciones en el consultorio del doctor, existió mucho ruido en el ambiente.

**4.1.4 PROTOTIPO CONSTRUIDO**

La foto 4.1 muestra el dispositivo AMCAEC I terminado y con todos sus complementos de hardware (excepto el software) :

- Dispositivo AMCAEC I (color gris)
- Audífonos (color negro)
- Cable DB – 9 (color gris)
- Cable de alimentación (color negro)



*Foto 1*

#### 4.1.5 ANALISIS DE COSTOS

Los elementos usados para construir el AMCAEC I fueron los siguientes:

- **Ez – Kit Lite**

Incluye la tarjeta que fue descrita ampliamente en el capítulo II, un cable de alimentación y el conector DB9.

Su precio de adquisición fue de USD 318.38

- **Amplificador de Audio**

Incluye la entrada de audio y un amplificador de ganancia constante y fue descrito detalladamente en el capítulo III.

Su costo de fabricación fue de USD 15

- **Audífonos**

Se usaron audífonos de marca RadioShack, con control de volumen incorporado, respuesta de frecuencia ideal de 20 a 20000 Hz y conector 1/8 “.

Su costo de adquisición fue de USD 40.80

- **Caja**

Fue construida específicamente para encapsular y proteger los elementos físicos del dispositivo. Está hecha de acrílico y es de color plateado.

Su costo de fabricación fue de USD 20

En total, sumando todos los valores anteriores, y sin incluir el costo de ingeniería, el precio final de fabricación del AMCAEC I ascendió a USD 394.18

Se debe además incluir un costo indirecto, el mismo que si bien no influye en el costo final de fabricación del equipo, si influyó en la realización del presente trabajo de investigación.

Este costo es el de la realización de los 9 exámenes con un equipo médico comercial con el fin de realizar las comparaciones en el análisis médico y el estadístico.

Cada examen tuvo un valor de USD 6, dando como resultado un total de USD 54.

Si se considera este último valor, el costo total del proyecto asciende a USD 448.18

# CAPITULO V

## CONCLUSIONES Y RECOMENDACIONES

### 5.1 CONCLUSIONES

1. El sentido de la audición está directamente ligado con el estado de ánimo de cada persona, ya que la capacidad auditiva de la misma varía según su estado de ánimo, nivel de concentración e incluso se ve afectado por la hora del día, por lo que al realizarse una evaluación audiométrica, nunca se obtendrán dos resultados iguales, aún para una misma persona
  
2. El oído humano es más sensible a las frecuencias medias que a las altas y a las bajas, entendiéndose, para el caso de la audiometría, por frecuencias altas aquellas que superan los 2000 Hz pero que están por debajo de los 8000 Hz, por frecuencias bajas aquellas que están por debajo de los 1000 Hz y frecuencias medias a aquellas que están entre 1000 y 2000 Hz.
  
3. Si un paciente no muestra deficiencia auditiva para las frecuencias medias, puede omitir el someterse a un tratamiento otológico, ya que el rango de conversación normal entre dos personas oscila entre dichas frecuencias medias.
  
4. Los pacientes padecen hipoacusia neurosensorial más frecuentemente que una hipoacusia de conducción, y además dicha patología se manifiesta de manera más profunda.
  
5. Los sentidos humanos en general y el oído en particular, responden a estímulos externos extremadamente bajos, lo cual complica de manera muy significativa su tratamiento, evaluación y estudio con recursos electrónicos limitados desde el punto de vista económico.

6. Al analizar de manera individual las componentes de frecuencia de un sonido que ha sido sometido a un filtrado digital, es posible visualizar que la variación de la amplitud de la señal en la banda de paso se ajusta al valor del rizado calculado.
7. El manejo y diseño de filtros digitales, así como su implementación requieren tanto de un sólido conocimiento de matemáticas como de un procesador muy potente que permita implementarlo. En el primer caso por la variedad y complejidad de tipos de filtros disponibles (todos ellos con una matemática muy avanzada). En el segundo caso porque se requiere realizar gran cantidad de operaciones matemáticas y a muy alta velocidad con el fin de que los retardos no afecten a la señal procesada. El segundo punto es fundamental en caso de que se trabaje con aplicaciones en tiempo real.
8. El uso de filtros digitales permite optimizar el ancho de banda disponible en cualquier aplicación, dado su alto grado de selectividad.
9. Los DSP's posee muchas ventajas en cuanto se refiere al procesamiento de señales, pero sin embargo, presentan mucha complejidad para realizar funciones de control.
10. Para trabajar con audio en tiempo real, se debe escoger un DSP que tenga una velocidad lo suficientemente alta para permitir cumplir con el teorema de Nyquist en toda la banda de audición humana, ya que caso contrario, se presentarían inconvenientes en la digitalización de las señales analógicas.
11. Para amplificar audio, dada la sensibilidad del oído humano, se debe escoger cuidadosamente el tipo de amplificador a usar, ya que la intervención del ruido es muy significativa.

12. La programación de un DSP es bastante compleja, pero si se la usa de manera óptima, brinda una infinita gama de opciones para la realización de complejas operaciones matemáticas en tiempos sumamente pequeños.
13. En caso de que se estén comparando dos tratamientos sobre individuos, animales o cosas, el uso de la estadística permite determinar de manera clara y objetiva las diferencias o similitudes entre los resultados obtenidos.
14. El uso de métodos estadísticos es indispensable para generalizar un procedimiento o tratamiento sobre una población
15. El uso de un software adecuado implementado en un computador personal, permite optimizar ampliamente las funciones de cualquier tipo de dispositivo periférico, ya que brinda muchas funciones de valor agregado sumamente amigables para el usuario final.
16. En caso de trabajar con dos procesadores, se debe dejar que el procesador más potente realice la mayor cantidad de funciones y que el menos potente realice de ser posible, únicamente su función específica.
17. Luego de analizar los resultados del AMCAEC I, se ha observado que su comportamiento es igual al de uno comercial.
18. Al hacer un análisis del costo final del AMCAEC I, se puede notar que se ha conseguido obtener un audímetro digital a un precio menor que el de un equivalente que se encuentre en el mercado.

## 5.2 RECOMENDACIONES

19. Se recomienda ampliar el alcance del dispositivo diseñado, para que el mismo realice otras pruebas complementarias a la evaluación de la conducción aérea, con el fin de aumentar el valor agregado del AMCAEC I.
20. Se recomienda realizar un estudio estadístico más amplio y utilizando el mismo dispositivo construido, en el que se aumente tanto el número de personas evaluadas como el de equipos de referencia, para poder garantizar al 100% los resultados obtenidos con el AMCAEC I.
21. Se recomienda modificar al dispositivo implementado para que éste pueda funcionar en modo "stand alone", para brindar mayor flexibilidad y movilidad al uso del mismo.
22. Se recomienda, para todos los puntos anteriores, trabajar con un médico especialista, para de esta manera cumplir no solamente los requerimientos electrónicos del equipo sino también con los netamente médicos.
23. En vista de la gran cantidad de sistemas que usan procesadores digitales de señales, y dada la importancia actual de las técnicas digitales, se recomienda implementar una materia como parte del pensum académico en la cual se imparta, a más de la teoría correspondiente, prácticas de procesamiento digital. Sería también recomendable que se enseñe la programación de al menos un procesador digital de señales.
24. En vista de que el proyecto de titulación es un trabajo netamente investigativo, se recomienda que se incluya como requisito obligatorio un análisis estadístico de resultados en todos los proyectos en los que sea pertinente hacerlo.



## CAPITULO VI

### BIBLIOGRAFÍA

#### Libros:

1. STANLEY, Farb. Otorrinolaringología. Segunda edición. Editorial El Manual Moderno. México. 1981
2. TRASERRA J.; AVELLANEDA R.; CUCHI M. A.; ABELLO P. Atlas Práctico Para el Médico General, Otorrinolaringología. Tomo 10. Editorial Salvat. Barcelona. 1984
3. HARRISON. Principios de Medicina Interna. Volumen 1. 15 ava Edición. Editorial Panamericana. España. 2001.
4. OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000
5. SOLER Brian; SPOTTS Jeff. Edición Especial Visual Basic 6. Editorial Prentice Hall. Iberia. 1999.
6. BELLAMY John, Digital Telephony. 3ra Edición. Editorial John Wiley & Sons. Estados Unidos de Norteamérica. 2000
7. HIDALGO Pablo, Apuntes de Comunicación Digital. Escuela Politécnica Nacional. Octubre 2003
8. ANALOG DEVICES, ADSP-2100 Family User's Manual, 3ra Edición. Editorial Computer Products División. Canadá. 1995
9. DANIEL Wayne, Bioestadística. Tercera Edición. Editorial Limusa. México. 1974 .

## Direcciones electrónicas:

10. [www.interacustic.com](http://www.interacustic.com)
11. [www.elo.utfsm.cl/elo385/documentos/intro-dsp.pdf](http://www.elo.utfsm.cl/elo385/documentos/intro-dsp.pdf)
12. [www.analog.com/analog\\_root/prductpage/producthome/0,,21xx%252dezlite\\_00.html](http://www.analog.com/analog_root/prductpage/producthome/0,,21xx%252dezlite_00.html)
13. [www.ee.vt.edu/tikuma/adsp2181/](http://www.ee.vt.edu/tikuma/adsp2181/)
14. [www.analog.com/productselection/pdf/adsp2181\\_d.pdf](http://www.analog.com/productselection/pdf/adsp2181_d.pdf)
15. [www.analog.com/analog\\_root/prductpage/producthome/0%2c%2cadsp%25252d2181%2c00.html](http://www.analog.com/analog_root/prductpage/producthome/0%2c%2cadsp%25252d2181%2c00.html)
16. [www.analog.com/productselection/pdf/ad1847\\_d.pdf](http://www.analog.com/productselection/pdf/ad1847_d.pdf)
17. <http://bips.bi.ehu.es/users/imanol/akustika/ElektroAkustikaApunteak.pdf>
18. <http://roble.pntic.mec.es/~ddoncel/>
19. [www.abcdatos.com](http://www.abcdatos.com)
20. [www.sgs-thomsom.com](http://www.sgs-thomsom.com)
21. <http://www.tecnun.com/asignaturas/tratamiento%20digital/TEMA8/>
22. <http://www.alldatasheet.com/datasheet-pdf/pdf/STMICROELECTRONICS/TDA2822.html>

## Otros:

23. Manual de usuario del Ez- Kit Lite
24. Curso DSP, Ing. Fernando Vásquez

# **Anexo 1**

Proceso de transducción de energía acústica a energía  
eléctrica

Tomado de : **HARRISON. Principios de Medicina Interna. Volumen 1. 15<sup>ava</sup>  
Edición. Editorial Panamericana. España. 2001.**

**TRATAMIENTO** del déficit gustativo es limitado. No obstante, los trastornos del gusto debidos a la administración de fármacos suelen desaparecer cuando se cambia la medicación prescrita. La xerostomía puede tratarse mediante saliva artificial, lo que aporta algún beneficio a los pacientes con alteración del medio salival. La pilonidoma por vía bucal puede ser útil en algunas formas de xerostomía. El tratamiento apropiado de las infecciones bacterianas y fúngicas de la cavidad oral puede ser de gran ayuda para mejorar la función gustativa. La disfunción del gusto tras un traumatismo puede desaparecer de manera espontánea sin ningún tipo de intervención, y lo hace con mayor frecuencia que la disfunción olfatoria traumática. La alteración del sentido del gusto por lesión química de la rama timpánica del nervio facial suele mejorar al cabo de 4 meses, mientras que cuando se produce una sección completa del nervio la disfunción suele ser permanente. En la mayoría de los pacientes con alteraciones del sentido del gusto de causa metabólica, el problema permanece estable o empeora. Algunos autores han recomendado la administración de zinc y vitaminas en los trastornos gustativos, aunque no se ha demostrado la eficacia de este tratamiento. En los trastornos neurosensitivos del gusto no existe ninguna forma eficaz de tratamiento.

**HIPOACUSIA** (pérdida de audición) constituye uno de los trastornos neurosensitivos más frecuentes del ser humano. Casi el 10% de la población adulta presenta algún tipo de hipoacusia. En muchos de estos pacientes, la hipoacusia se inicia durante los primeros años de vida. No obstante, la hipoacusia puede aparecer a cualquier edad. Entre el 30 y el 35% de las personas mayores de 65 años de edad presenta una hipoacusia de intensidad suficiente como para necesitar un audífono.

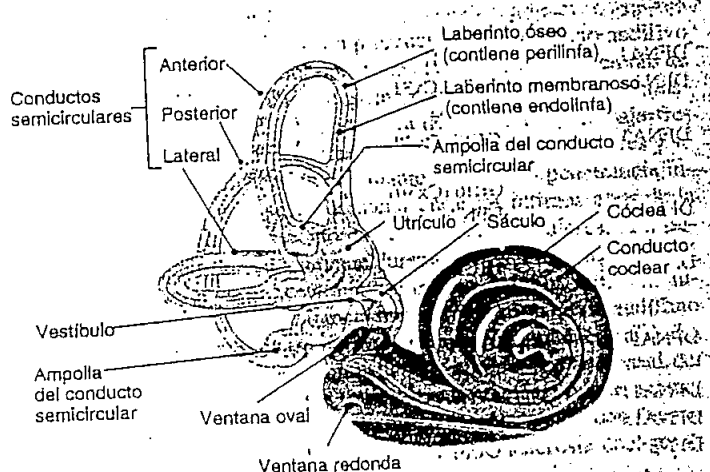
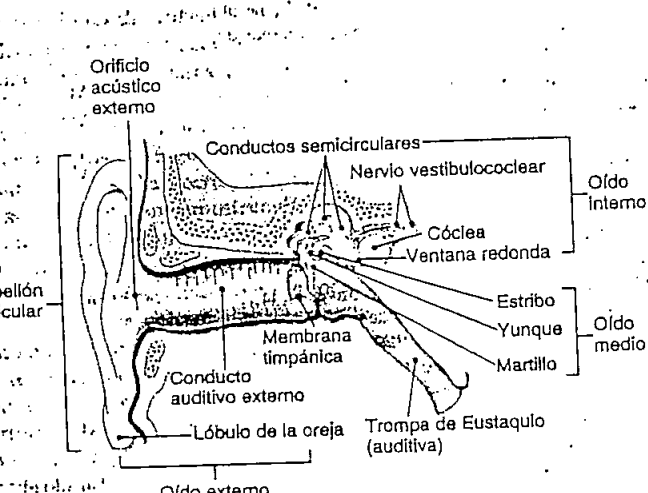
**FISIOLÓGIA DE LA AUDICIÓN** (Fig. 29-1). La audición se realiza a través de la conducción aérea y la conducción ósea del sonido. En la conducción aérea, las ondas del sonido alcanzan el oído por reflexión en el aire, se introducen en el conducto auditivo externo y mueven la membrana timpánica que, a su vez, mueve el malleus y el yunque. El movimiento de la base del estribo promueve modificaciones en la presión del líquido contenido en el oído interno, lo que genera una onda de movimiento en la membrana basilar de la cóclea. La membrana timpánica y la cadena osicular del oído medio actúan como un mecanismo de control de impedancia

que mejora la eficiencia de la transferencia de energía desde el aire hasta el oído interno relleno de líquido. La audición por conducción ósea se produce cuando el origen del sonido, que está en contacto con la cabeza, origina una vibración de las estructuras óseas del cráneo, incluido el hueso temporal, lo que genera una onda de movimiento en la membrana basilar.

Los estereocilios de las células ciliadas del órgano de Corti que descansan sobre la membrana basilar, están en contacto con la membrana tectorial y sufren deformación por esta onda de movimiento. El desplazamiento máximo de la membrana basilar está determinado por la frecuencia del tono de la estimulación. Los tonos de alta frecuencia producen un desplazamiento máximo de la membrana basilar en las proximidades de la base de la cóclea. A medida que disminuye la frecuencia del tono de estimulación, el punto de desplazamiento máximo se dirige hacia la punta de la cóclea.

Las células ciliadas internas y externas del órgano de Corti muestran un patrón de inervación diferente, pero ambas son mecanorreceptores. La inervación aferente depende principalmente de las células ciliadas internas y la eferente, de las externas. La motilidad de las células ciliadas externas modifica la micromecánica de las células ciliadas internas creando un amplificador coclear, lo que explica la gran sensibilidad y la selectividad de frecuencias de la cóclea.

El concepto actual sobre la transmisión coclear es que el desplazamiento de las puntas de los cilios establece vínculos entre ellas que facilitan la entrada de potasio dentro de la célula, provocando su despolarización. El potasio abre los canales del calcio próximos a la base de la membrana, con lo que entra calcio dentro de la célula. Los iones de calcio estimulan la liberación de transmisores. Se supone que el neurotransmisor en la interfaz de las dendritas de las células ciliadas y del nervio coclear es el glutamato. El potencial de acción en el VIII par craneal tiene lugar 0.5 ms después de que se inicie el potencial microfónico coclear. Cada una de las neuronas del nervio coclear puede ser activada con una frecuencia de intensidad específica para esa célula. Esta especificidad se mantiene en cada punto de la vía auditiva central: los núcleos cocleares dorsal y ventral, el cuerpo trapezoide, el complejo de la oliva superior, el lemnisco externo, el tubérculo cuadrigémino inferior, el cuerpo geniculado interno y la corteza auditiva. Para frecuencias bajas, las fibras del VIII par pueden responder de una forma más o menos sincrónica con el tono de estimulación. Para frecuencias más elevadas, se produce un bloqueo de fase, de forma que las neuronas vuelven a responder frente a fases particulares del ciclo de la onda de sonido. La intensidad está codificada por el grado de actividad nerviosa de las distintas neuronas, el número de neuronas que son activas y las neuronas específicas que son activadas.



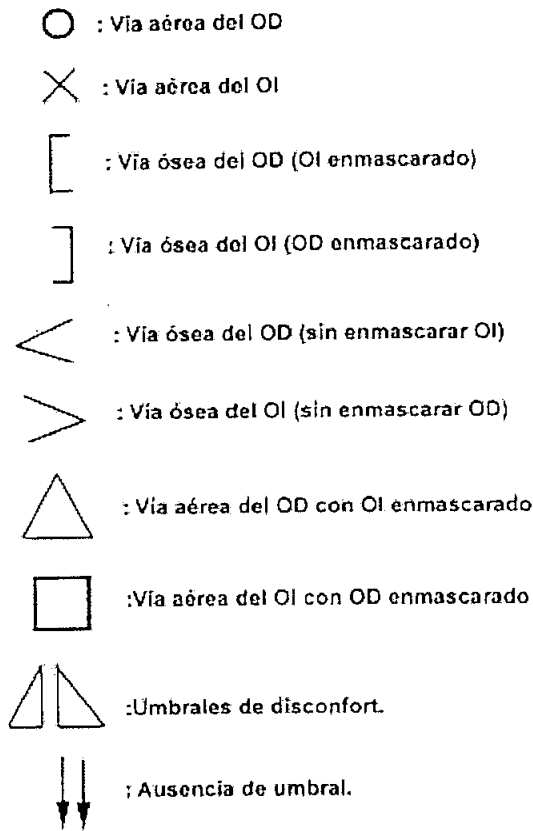
**FIGURA 29-1.** A. Dibujo de un corte coronal modificado efectuado a través del oído externo y del hueso temporal en el que se observan las estructuras de los oídos medio e interno. B. Dibujo realizado sobre una imagen a mayor aumento del oído interno.

## **Anexo 2**

Signos audiométricos y ejemplos de hipoacusias

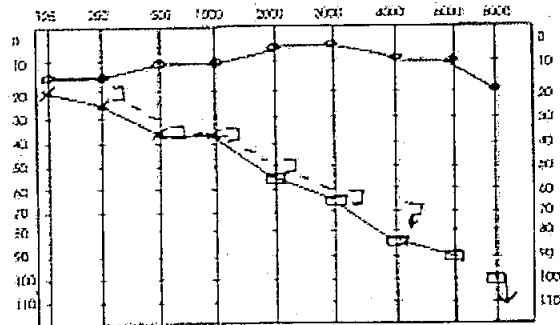
Tomado de : [www.interacustic.com](http://www.interacustic.com)

## SIGNOS AUDIOMÉTRICOS



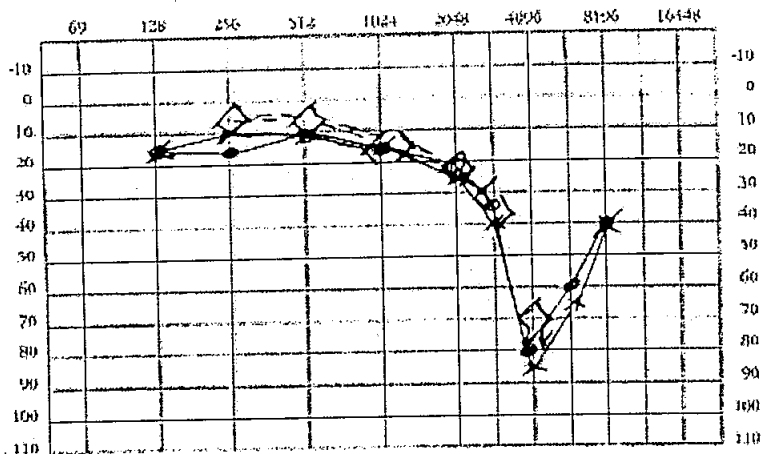
## EJEMPLOS

### HIPOACUSIA SENSORIONEURAL DEL OÍDO IZQUIERDO



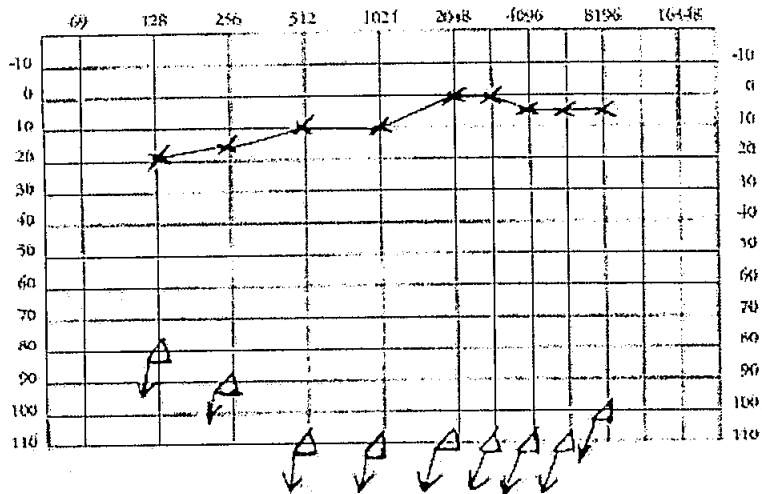
En este audiograma observamos audición normal del OD. En el OI existe una pérdida de la misma magnitud tanto de la vía ósea como de la aérea; en este caso importan las pruebas supraliminales para objetivar la presencia de reclutamiento y/o fatiga auditiva patológica en caso de tratarse de una cortipatía o una lesión retrococlear. Nótese en el OD la presencia de LDL + (cortipatía).

## TRAUMA ACUSTICO BILATERAL



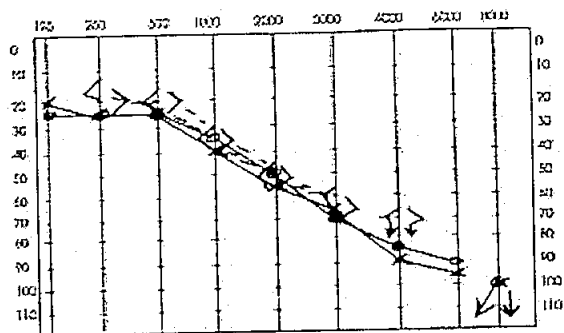
En el trauma acústico es característico el escotoma que se produce en las frecuencias agudas del 3000, 4000 y 6000 Hz, con recuperación en la frecuencia de 8000Hz. En estos casos encontraremos positivos los Test de reclutamiento como el LDL, SISI y Fowler.

## ANACUSIA DEL OD



Podemos observar audición normal del OI y en el OD una ausencia total de umbrales, lo que se transcribe como flechas hacia abajo. En estos casos es de suma importancia utilizar las mejores técnicas para enmascarar, además la logaudiometría de la palabra que en el oído anacúsico debe ser nula.

# HIPOACUSIA SENSORINEURAL DESCENDENTE SIMÉTRICA



Audiograma más característico de las presbiacusias .



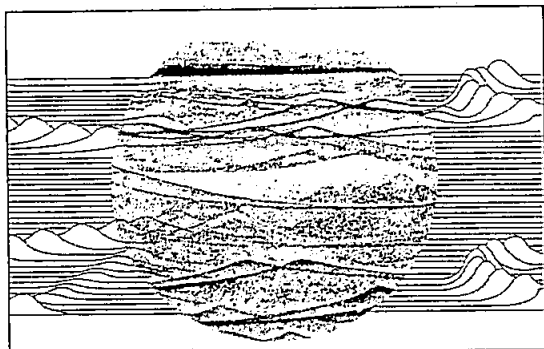
## **Anexo 3**

Perspectiva histórica y promesa a futuro del  
procesamiento digital de señales

Tomado de: **OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en  
Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000**

# 1

## INTRODUCCIÓN



El tratamiento de señales posee una larga y rica historia. Es una tecnología que se relaciona con un inmenso conjunto de disciplinas entre las que se encuentran el ocio, las comunicaciones, la exploración del espacio, la medicina y la arqueología, por nombrar sólo unas pocas. Existe un amplio conjunto de sistemas donde son de especial importancia algoritmos sofisticados y hardware para tratamiento de señales, sistemas que van desde sistemas militares altamente especializados, pasando por aplicaciones industriales, hasta llegar a la electrónica de consumo, de bajo coste y altos volúmenes de ventas. Aunque de forma rutinaria vemos como cotidianas las prestaciones de sistemas de entretenimiento doméstico, como la televisión o el equipo de audio de alta fidelidad, estos sistemas siempre han estado fuertemente basados en el estado del arte del tratamiento de señales. Hoy en día, esta afirmación es incluso más cierta con el nacimiento de la televisión avanzada y de los sistemas de información y entretenimiento multimedia. Es más, a medida que los sistemas de comunicación se van convirtiendo cada vez más en sistemas sin hilos, móviles y multifunción, la importancia de un tratamiento de señales sofisticado en dichos equipos continúa creciendo. En conjunto, si observamos el futuro, es claro que el papel que juega el tratamiento de señales en nuestra sociedad se está acelerando, conducido en parte por la convergencia de las comunicaciones, los computadores y el tratamiento de señales, tanto en el área de consumo como en las aplicaciones industriales avanzadas y en aplicaciones gubernamentales.

El campo del tratamiento de señales se ha beneficiado siempre de una relación muy estrecha entre la teoría, las aplicaciones y las tecnologías de realización de sistemas. El creciente número de aplicaciones y la demanda de algoritmos de sofisticación creciente va de la mano con la rápida marcha de la tecnología de dispositivos para la realización de sistemas de tratamiento de señales. Algunas estimaciones indican que la capacidad de proceso de los microprocesadores para tratamiento de señales se incrementará probablemente por un factor de 200 o más en los próximos diez años. Parece claro que en muchos aspectos la importancia y el papel del tratamiento de señales está expandiéndose y acelerándose.

El tratamiento de señales trata de la representación, transformación y manipulación de señales y de la información que contienen. Por ejemplo, podríamos desear separar dos o más señales que se han combinado de alguna forma, o podríamos querer realzar alguna componente de la señal o algún parámetro de un modelo de señal. En sistemas de comunicaciones, generalmente es necesario realizar procesos previos como modulación, acondicionamiento de señal y compresión antes de realizar la transmisión por un canal y realizar después un procesado posterior en el receptor. Antes de los años 60, la tecnología para el tratamiento de señales era casi exclusivamente tecnología analógica en tiempo continuo<sup>1</sup>. La rápida evolución de los computadores y los microprocesadores digitales junto con algunos importantes desarrollos teóricos como el algoritmo de la transformada rápida de Fourier (FFT) fueron la causa de un importante desplazamiento hacia las tecnologías digitales, naciendo así el campo de tratamiento digital de señales. Un aspecto fundamental del tratamiento digital de señales es que se basa en el procesamiento de secuencias de muestras. La naturaleza de tiempo discreto de la tecnología del tratamiento digital de señales es también característica de otras tecnologías de tratamiento de señal como los dispositivos de onda acústica superficial (SAW, *Surface Acoustic Wave*), los dispositivos de acoplo de carga (CCD, *Charge-coupled Devices*), los dispositivos de transporte de carga (CTD, *Charge Transport Devices*) y las tecnologías de condensadores conmutados. En tratamiento digital de señales, las señales se representan como secuencias de números de precisión finita y el procesado se realiza utilizando cómputo digital. El término más general *tratamiento de señales en tiempo discreto* incluye el tratamiento digital de señales como un caso especial, pero incluye también la posibilidad de que las secuencias de muestras (datos muestreados) sean procesadas con otras tecnologías en tiempo discreto. A menudo la distinción entre los términos *tratamiento de señales en tiempo discreto* y *tratamiento digital de señales* es de poca importancia, ya que ambos tratan con señales en tiempo discreto. Aunque hay muchos ejemplos en los que las señales que se van a procesar son ya de por sí secuencias, la mayoría de las aplicaciones requieren el uso de la tecnología de tiempo discreto para procesar señales en tiempo continuo. En este caso, las señales en tiempo continuo se convierten en una secuencia de muestras, es decir, en señales en tiempo discreto. Tras el proceso en tiempo discreto, las secuencias de salida se vuelven a convertir en señales en tiempo continuo. A menudo es deseable que esos sistemas funcionen en tiempo real, lo que significa que el sistema en tiempo discreto se implementa de forma que las muestras de la salida se calculan a la misma velocidad a la que se muestrea la señal en tiempo continuo. El tratamiento en tiempo discreto y en tiempo real de señales en tiempo continuo es práctica común en sistemas de comunicaciones, radar y sonar, codificación y realce de voz y vídeo e ingeniería biomédica, por nombrar sólo unos pocos. Un reproductor de discos compactos es un ejemplo algo diferente en el sentido de que se almacena una forma procesada de la entrada (en el disco compacto) y el tratamiento final se realiza en tiempo real cuando se desea obtener la salida. El sistema de grabación y reproducción de discos compactos se basa en muchos de los conceptos de tratamiento de señales que presentamos en este libro.

Una buena parte del tratamiento de señales tradicional involucra el proceso de una señal para obtener otra señal. Otra clase importante de problemas de tratamiento de señales es el de la *interpretación de señales*. En estos problemas el objetivo del proceso no es obtener una señal de salida sino obtener una caracterización de la señal de entrada. Por ejemplo, en un sistema de reconocimiento o comprensión de voz el objetivo es interpretar la señal de entrada o extraer información de la misma. Típicamente, un sistema como ése aplicará un procesado di-

<sup>1</sup> En un contexto general, típicamente nos referiremos a la variable independiente como «tiempo», incluso aunque en contextos específicos la variable independiente pueda tomar una cualquiera de un amplio conjunto de posibles dimensiones. En consecuencia, tiempo continuo y tiempo discreto se deben ver sólo como términos genéricos que se refieren a una variable independiente continua o discreta, respectivamente.

gital previo (filtrado, estimación de parámetros, etc.) seguido por un sistema de reconocimiento de patrones que produzca una representación simbólica, como por ejemplo una transcripción fonémica de la voz. Esta salida simbólica puede ser a su vez la entrada de un sistema de tratamiento simbólico, como un sistema experto basado en reglas, que proporcione la interpretación final de la señal. Hay todavía otra categoría relativamente nueva de técnicas de tratamiento de señales en las que se realiza la manipulación simbólica de expresiones. Este tipo de tratamiento es particularmente útil en estaciones de trabajo dedicadas al tratamiento de señales y en el diseño asistido por computador de sistemas de tratamiento de señales. En este caso las señales y los sistemas se representan y se manejan como objetos abstractos de datos. Los lenguajes de programación orientada a objetos proporcionan un entorno conveniente para manejar señales, sistemas y expresiones de tratamiento de señales sin evaluar explícitamente las secuencias de datos y proporcionan las bases para esta clase de tratamiento. La sofisticación de los sistemas diseñados para realizar tratamiento de expresiones de señales está influenciada directamente por la incorporación de conceptos, teoremas y propiedades fundamentales del tratamiento de señales como los que forman las bases de este libro. Por ejemplo, un entorno de tratamiento de señales que incorpora la propiedad de que la convolución en el dominio del tiempo corresponde a multiplicación en el dominio de la frecuencia, puede explorar una variedad de disposiciones de estructuras de filtros, incluyendo aquellas que utilizan directamente la transformada discreta de Fourier y el algoritmo de transformada rápida de Fourier. Análogamente, los entornos que incorporan la relación entre la frecuencia de muestreo y el solapamiento pueden hacer uso efectivo de las estrategias de diezmado e interpolación para la realización de filtros. Actualmente se están explorando ideas similares para realizar tratamiento de señales en entornos de red. En este tipo de entornos, los datos se pueden etiquetar potencialmente con una descripción de alto nivel del procesamiento a realizar y los detalles de la realización se pueden basar dinámicamente en los recursos disponibles en la red.

El desarrollo de los entornos orientados a objetos para el diseño de sistemas asistido por computador y para el tratamiento de señales en redes que cambian dinámicamente está todavía en etapas muy tempranas y una discusión detallada estaría fuera de los objetivos de este texto. Sin embargo, es importante reconocer que los conceptos básicos que se tratan en este libro no se deben ver como sólo de naturaleza teórica, ya que probablemente formen parte explícita de los entornos de tratamiento de señales asistidos por computador en estaciones de trabajo y en redes.

La mayoría de los conceptos y técnicas de diseño presentados en este libro se han incorporado ya en la estructura de sistemas software sofisticados, como por ejemplo Matlab. En muchos casos en los que las señales en tiempo discreto se adquieren y almacenan en computadores, estas herramientas permiten realizar operaciones de tratamiento de señales extremadamente sofisticadas, construidas utilizando funciones básicas. En esos casos, generalmente no es necesario conocer los detalles del algoritmo que realiza el cómputo de una operación como la FFT, pero es esencial entender lo que se está calculando y cómo se debe interpretar. En otras palabras, es esencial una correcta comprensión de los conceptos considerados en este texto para realizar un uso inteligente de las herramientas software para tratamiento de señales que hoy en día están ampliamente disponibles.

Los problemas de tratamiento de señales no están confinados, por supuesto, a señales unidimensionales. Aunque hay algunas diferencias fundamentales entre las teorías del tratamiento de señales unidimensional y multidimensional, una buena parte del material que presentamos en este libro tiene su contrapartida directa en sistemas multidimensionales. La teoría del tratamiento digital de señales multidimensionales se presenta con detalle en los libros de Dudgeon y Mersereau (1984), Lim (1989) y Bracewell (1986)<sup>2</sup>. En muchas aplica-

---

<sup>2</sup> Los nombres de los autores y las fechas que se utilizan en el libro se refieren a libros y artículos que se recogen en la bibliografía al final del texto.

ciones de tratamiento de imagen se requiere el uso de técnicas de tratamiento de señales bidimensionales. Es el caso de áreas como la codificación de vídeo, imágenes médicas, realce y análisis de fotografías aéreas, análisis de fotografías de satélites meteorológicos, y realce de transmisiones de vídeo de sondas lunares y de espacio profundo. Las aplicaciones del tratamiento de señales multidimensionales al tratamiento de imagen se presentan en Andrews y Hunt (1977), Macowski (1983), Pratt (1991), Castleman (1996), Jain (1989) y Chelappa *et al.* (1998). El análisis de datos sísmicos que se requiere en aplicaciones de exploración petrolífera, medición de terremotos, y monitorización de pruebas nucleares utiliza también técnicas de tratamiento de señales multidimensionales. Las aplicaciones sísmicas se discuten en Robinson y Treitel (1980) y en Robinson y Durrani (1985).

El tratamiento de señales multidimensionales es sólo uno de los muchos temas avanzados y especializados que se basan en los fundamentos que se presentan en este libro. El análisis espectral que se basa en el uso de la transformada discreta de Fourier y el uso del modelado de señales es otro aspecto particularmente rico e importante del tratamiento de señales. Presentamos muchas facetas de este tema, enfocándonos en los conceptos y técnicas básicas relacionados con el uso de la transformada discreta de Fourier. Además de estas técnicas, existe una variedad de métodos de análisis espectral que se basan de una forma u otra en modelos de señal específicos. Por ejemplo, una clase de métodos de análisis espectral de alta resolución denominados métodos de máxima entropía (análisis espectral MEM, *Maximum Entropy Methods*) se basa en representar la señal que se va a analizar como la respuesta de un filtro en tiempo discreto lineal e invariante con el tiempo a un impulso o a ruido blanco. El análisis espectral se realiza estimando los parámetros (es decir, los coeficientes de la ecuación en diferencias) del sistema y calculando el módulo al cuadrado de la respuesta en frecuencia del filtro del modelo. Un tratamiento completo y detallado de los aspectos y técnicas de este procedimiento de modelado de señales y análisis espectral se puede construir directamente a partir de los fundamentos de este texto. En los libros de Kay (1988), Marple (1987) y Hayes (1996) se pueden encontrar presentaciones detalladas. El modelado de señales juega también un papel importante en la compresión y codificación de datos, y de nuevo los fundamentos sobre ecuaciones en diferencias proporcionan la base para entender muchas de estas técnicas. Por ejemplo, una clase de técnicas de codificación de señales denominada codificación predictiva lineal (LPC, *Linear Predictive Coding*) explota la noción de que si una señal es la respuesta de cierta clase de filtros en tiempo discreto, el valor de la señal en cualquier índice temporal es una función lineal de los valores anteriores y por tanto se puede predecir a partir de ellos mediante una función lineal. En consecuencia, se pueden obtener representaciones eficientes de señales estimando esos parámetros de predicción y utilizándolos junto con el error de predicción para representar las señales. Dichas señales se pueden volver a generar cuando sea necesario a partir de los parámetros del modelo. Esta clase de técnicas de codificación de señales ha sido particularmente efectiva en codificación de voz y se describe con considerable detalle en Jayant y Noll (1984), Markel y Gray (1976), Rabiner y Schafer (1978) y Deller *et al.* (1993).

Otro tema avanzado de importancia considerable es el tratamiento adaptativo de señales. Los sistemas adaptativos representan una clase particular de sistemas variantes con el tiempo y, en cierto sentido, no lineales, con amplia aplicación y con técnicas efectivas y bien establecidas para su diseño y análisis. De nuevo muchas de estas técnicas se basan en los fundamentos del tratamiento de señales en tiempo discreto que se presentan en este libro. Los detalles sobre el tratamiento adaptativo de señales se pueden encontrar en Haykin (1996) y en Widrow y Stearns (1985).

Éstos representan sólo unos pocos de entre los muchos temas avanzados que van más allá de la materia que se presenta en este libro. Entre otros podemos citar los procedimientos de diseño de filtros avanzados y especializados, una variedad de algoritmos especializados para la evaluación de la transformada de Fourier, estructuras especializadas de filtros y

técnicas avanzadas de tratamiento de señales multitasas, incluyendo bancos de filtros y transformadas *wavelet*.

A menudo se dice que el propósito de un libro de texto fundamental debería ser descubrir una materia más bien que cubrirla, y al escoger los temas y la profundidad de su tratamiento en este libro nos ha guiado esta filosofía. La breve discusión anterior sobre temas avanzados y la bibliografía que aparece al final del libro es muy sugerente sobre la rica variedad de direcciones que estos fundamentos comienzan a descubrir.

## PERSPECTIVA HISTÓRICA

El tratamiento de señales en tiempo discreto ha avanzado con pasos desiguales durante un largo periodo de tiempo. Mirando hacia atrás en el desarrollo de este campo podemos obtener una valiosa perspectiva de los fundamentos que permanecerán en el núcleo en el futuro. Desde la invención del cálculo en el siglo XVII, los científicos e ingenieros han desarrollado modelos para representar fenómenos físicos en términos de funciones de variable continua y ecuaciones diferenciales. Se han utilizado técnicas numéricas para resolver estas ecuaciones cuando no es posible obtener una solución analítica. De hecho, Newton utilizó métodos de diferencias finitas que son casos especiales de algunos sistemas en tiempo discreto que presentamos en este libro. Los matemáticos del siglo XVIII, como Euler, Bernoulli y Lagrange desarrollaron métodos de integración e interpolación numérica de funciones de variable continua. Una interesante investigación histórica realizada por Heideman, Johnson y Burrus (1984) demostró que Gauss descubrió el principio fundamental de la transformada rápida de Fourier (que se presenta en el Capítulo 9) ya en 1805, incluso antes de la publicación del tratado de Fourier sobre la representación de funciones mediante series armónicas.

Hasta principios de los años cincuenta el tratamiento de señales como ha sido definido se hacía generalmente con sistemas analógicos que se realizaban con circuitos electrónicos o incluso con dispositivos mecánicos. Aunque los computadores digitales estaban ya disponibles en entornos de negocios y en laboratorios científicos, eran caros y de capacidad relativamente limitada. Por esa época, la necesidad de un tratamiento de señales más sofisticado en algunas áreas de aplicación creó un considerable interés en el tratamiento de señales en tiempo discreto. Uno de los primeros usos de los computadores digitales en el tratamiento digital de señales fue en la prospección petrolífera. Se grababan datos sísmicos en cinta magnética para su procesamiento posterior. Este tipo de tratamiento de señales no se podía realizar generalmente en tiempo real: a menudo eran necesarios minutos o incluso horas de tiempo de computador para procesar sólo algunos segundos de datos. Incluso así, la flexibilidad del computador digital y los potenciales beneficios económicos hicieron esta alternativa extremadamente interesante.

También durante los años cincuenta, el uso de los computadores digitales en el tratamiento de señales surgió de una forma diferente. Debido a la flexibilidad de los computadores digitales, a menudo resultaba útil simular un sistema de tratamiento de señales en un computador digital antes de realizarlo utilizando un hardware analógico. De este modo, un nuevo algoritmo o sistema para tratamiento de señales se podía estudiar en un entorno experimental flexible antes de comprometer recursos económicos y de ingeniería para realizarlo. Ejemplos típicos de esas simulaciones fueron las simulaciones del *vocoder* realizadas en el Laboratorio Lincoln y en los Laboratorios Bell. En la realización de un *vocoder* analógico de canales, por ejemplo, las características de los filtros afectaban a la calidad percibida de las señales de voz en formas que eran difíciles de cuantificar objetivamente. Mediante las simulaciones en computador, se pudieron ajustar las características de los filtros y la calidad percibida del sistema de codificación de voz previamente a la construcción del equipo analógico.

En todos los ejemplos de tratamiento de señales mediante computadores digitales, el computador ofrecía tremendas ventajas de flexibilidad. Sin embargo, el procesado no se podía realizar en tiempo real. En consecuencia, una actitud que prevaleció era que el computador digital se utilizaba para *aproximar* o *simular* un sistema de tratamiento de señales analógicas. De acuerdo con ese estilo, los primeros trabajos en filtrado digital se dedicaron en buena parte a las formas en las que se podía programar un filtro en un computador digital de manera que con una conversión analógico-digital de la señal, seguida por un filtrado digital y seguida por una conversión digital-analógico, el sistema completo se aproximara a un buen filtro analógico. La noción de que los sistemas digitales podrían de hecho ser prácticos para la realización en tiempo real del tratamiento de señales necesario en comunicaciones de voz, procesado radar, y cualquier otra aplicación dentro de un amplio conjunto parecía, incluso en los momentos más optimistas, ser altamente especulativa. La velocidad, el coste y el tamaño eran, por supuesto, tres factores importantes a favor del uso de componentes analógicos.

Cuando las señales se procesaron en computadores digitales, los investigadores tuvieron una tendencia natural a experimentar con algoritmos de tratamiento de señales de complejidad creciente. Algunos de esos algoritmos surgieron de la flexibilidad de los computadores digitales y no tenían aparentemente realizaciones prácticas con componentes analógicos. Por tanto, muchos de esos algoritmos se consideraron como ideas interesantes pero poco prácticas. El desarrollo de esos algoritmos de tratamiento de señales hizo incluso más tentadora la noción de una realización completamente digital de los sistemas de tratamiento de señales. Comenzaron los trabajos activos de investigación en vocoders digitales, analizadores digitales de espectro y otros sistemas completamente digitales, con la esperanza de que esos sistemas pudieran llegar a ser prácticos.

La evolución de un nuevo punto de vista hacia el tratamiento de señales en tiempo discreto se aceleró más con el descubrimiento de Cooley y Tukey (1965) de un algoritmo eficiente para el cálculo de transformadas de Fourier. Esta clase de algoritmos se conocen con el nombre de transformada rápida de Fourier, o FFT (FFT, *Fast Fourier Transform*). La FFT fue significativa por varias razones. Muchos algoritmos de tratamiento de señales desarrollados en computadores digitales requerían tiempos de proceso varios órdenes de magnitud superiores al tiempo real. A menudo eso era debido a que el análisis espectral era un componente importante del proceso de tratamiento de las señales y no existía una forma eficiente de implementarlo. El algoritmo de la transformada rápida de Fourier redujo en órdenes de magnitud el tiempo de cómputo de la transformada de Fourier, permitiendo la realización de algoritmos de tratamiento de señales de complejidad creciente, con tiempos de proceso que permitían la experimentación interactiva con el sistema. Además, al darse cuenta de que los algoritmos de transformada rápida de Fourier podrían, de hecho, ser realizables utilizando hardware digital de propósito específico, muchos algoritmos de tratamiento digital de señales que previamente eran impracticables comenzaron a verse como posibles diseños prácticos.

Otra implicación importante del algoritmo de la transformada rápida de Fourier era que se trataba de un concepto inherentemente de tiempo discreto. Estaba dirigido hacia el cálculo de la transformada de Fourier de una señal en tiempo discreto o secuencia, y utilizaba propiedades y conceptos matemáticos que eran exactos en el dominio discreto, es decir, no eran una aproximación a la transformada de Fourier en tiempo continuo. Esto tuvo el efecto de estimular una reformulación de muchos conceptos y algoritmos de tratamiento de señales en términos de matemáticas de tiempo discreto, y estas técnicas formaron después una serie de relaciones exactas en el dominio del tiempo discreto. Siguiendo con este desplazamiento desde la idea de que el tratamiento de señales en un computador digital era meramente una aproximación a las técnicas analógicas de tratamiento de señales, surgió un fuer-

te interés en tratamiento de señales en tiempo discreto como un campo importante de investigación con derecho propio.

Otro desarrollo importante en la historia del tratamiento de señales en tiempo discreto ocurrió en el terreno de la microelectrónica. El descubrimiento y la posterior proliferación de los microprocesadores preparó el terreno para las realizaciones de bajo coste de los sistemas de tratamiento de señales en tiempo discreto. Aunque los primeros microprocesadores eran demasiado lentos para implementar en tiempo real la mayoría de los sistemas en tiempo discreto, a mediados de los ochenta la tecnología de circuitos integrados había avanzado hasta el nivel de permitir la realización de microcomputadores en punto fijo y en punto flotante con arquitecturas especialmente diseñadas para realizar algoritmos de tratamiento de señales en tiempo discreto. Con esta tecnología llegó, por primera vez, la posibilidad de una amplia aplicación de las técnicas de tratamiento de señales en tiempo discreto.

## **PROMESA DE FUTURO**

Los ingenieros en microelectrónica continúan esforzándose por incrementar la densidad de los circuitos y los rendimientos de producción y como resultado, la complejidad y la sofisticación de los sistemas microelectrónicos está creciendo continuamente. De hecho, la complejidad y la capacidad de los chips de tratamiento digital de señales ha crecido exponencialmente desde principios de los ochenta y no muestra signos de ralentizarse. A medida que las técnicas de integración de obleas se vayan desarrollando progresivamente, se implementarán sistemas de tratamiento de señales con bajo coste, tamaño miniaturizado, y bajo consumo de potencia. En consecuencia, la importancia del tratamiento de señales en tiempo discreto continuará creciendo casi con toda certeza y es probable que los futuros desarrollos en este campo sean incluso más importantes que los que hemos descrito. Las técnicas de tratamiento de señales en tiempo discreto ya están promoviendo avances revolucionarios en algunas áreas de aplicación. Un ejemplo notable es el área de las telecomunicaciones, en el que las técnicas de tratamiento de señales en tiempo discreto, la tecnología de microelectrónica, y la transmisión por fibra óptica se combinan para cambiar la naturaleza de los sistemas de comunicaciones de forma verdaderamente revolucionaria. Un impacto similar se puede esperar en muchas otras áreas de la tecnología.

Aunque el tratamiento de señales en tiempo discreto es un campo dinámico y rápidamente creciente, sus fundamentos están bien establecidos. Nuestra meta con este libro es proporcionar un tratamiento coherente de la teoría de los sistemas lineales en tiempo discreto, filtrado, muestreo y análisis de Fourier en tiempo discreto. Los temas que se presentarán proporcionarán al lector los conocimientos necesarios para apreciar el amplio rango de aplicaciones del tratamiento de señales en tiempo discreto y las bases para contribuir a los futuros desarrollos en este excitante campo de la tecnología.



# **Anexo 4**

## Demostración del Teorema de Muestreo

Tomado de : <http://www.tecnun.com/asignaturas/tratamiento%20digital/TEMA8/>

### 12.1.4.2 Transformada inversa

Se plantea el problema inverso, obtener la secuencia  $\{x_k\}$  a partir de su transformada en  $z$ ,  $X(z)$ . Para el caso de tener  $X(z)$  en forma de función racional, se puede calcular de forma sencilla a partir del teorema de los residuos, estableciendo que:

$$x_n = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz = \sum [\text{residuos de } X(z)z^{n-1} \text{ para los polos interiores de } C] \quad (12.30)$$

siendo  $C$  una curva cerrada que envuelve el origen.

En el caso de secuencias con términos únicamente de índice positivo, o únicamente negativo, se puede utilizar el cálculo de los coeficientes mediante el método de la división larga.

Una posibilidad más de cálculo de la transformada inversa es por descomposición en fracciones simples. Si  $X(z)$  es racional, con todos sus polos simples y el grado del numerador inferior al del denominador, entonces:

$$X(z) = \sum_{i=1}^N \frac{A_i}{1 - a_i z^{-i}} \quad (12.31)$$

y la transformada inversa será la suma de las transformaciones inversas, que, en el caso de secuencias de índice positivo, será transformada en una suma de secuencias exponenciales y por tanto en este caso:

$$\begin{aligned} x_k &= \sum_{i=1}^n A_i a_i^k & \text{para } k \geq 0 \\ x_k &= 0 & \text{para } k < 0 \end{aligned} \quad (12.32)$$

### 12.1.5 Muestreo de señales

La idea de muestras referidas a un sistema basado en un computador digital, se concreta en la toma de valores de una señal continua en sucesivos instantes de tiempo. El muestreo de señales consiste en la construcción de secuencias a partir de señales continuas. Un problema que salta a la vista es la pérdida de información.

En este punto se analizará fundamentalmente este problema, dándose las condiciones en las que el proceso de muestreo/reconstrucción, (CAD/CDA), no signifique pérdida de información.

El caso más interesante de muestreo es el periódico, que se caracteriza por que los instantes de toma de muestras están regularmente espaciados en el tiempo y a éste se le llama periodo de muestreo,  $T$ , y a su inversa, frecuencia de muestreo,  $f_s$ .

El elemento que realiza este proceso se denomina “muestreador”, siendo la forma usual de representarlo en los diagrama de bloques la indicada en la figura 12.6

En este bloque la salida está relacionada con la entrada por la expresión:

$$x_k = x(kT) \tag{12. 33}$$

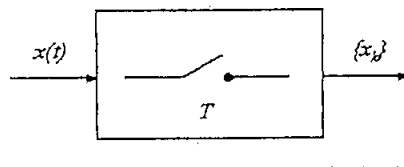


Figura 12. 6 Símbolo del muestreador

Intuitivamente se puede apreciar que el periodo de muestreo va a jugar un papel fundamental en este proceso. La elección de la frecuencia de muestro es un problema fundamental en el sistema de datos muestrados. Conviene muestrear los datos con la menor frecuencia posible, pero que sea lo suficientemente grande como para poder evitar la pérdida de información y sea posible reconstruir la función continua a través de las muestras. Justamente, el teorema del muestreo es el que puede ayudar a determinar la frecuencia óptima del convertidor analógico/digital.

Para poder establecer las conclusiones del teorema del muestreo, se va a recurrir al estudio frecuencial. Si una señal continua,  $x(t)$ , con transformada de Fourier continua,  $X_c(\omega)$ . Entonces las relaciones señal/transformada serán del tipo:

$$X_c(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X_c(\omega)e^{j\omega t} d\omega \tag{12. 34}$$

La relación existente entre la secuencia procedente del muestreo de esta señal y ella misma será (12. 33) con lo que:

$$x_k = x(kT) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X_c(\omega) e^{j\omega kT} d\omega \quad (12.35)$$

Si la secuencia  $\{x_k\}$  tiene asimismo transformada discreta de Fourier  $X_d(\omega)$ , entonces estará relacionada con su transformada por medio de:

$$x_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} X_d(\omega) e^{j\omega k} d\omega \quad (12.36)$$

El problema es estudiar si existe alguna relación entre estas dos transformadas, la de la señal continua  $X_c(\omega)$  y la de la secuencia  $X_d(\omega)$ . Se puede demostrar que la relación existente es:

$$X_d(\omega) = \sum_{r=-\infty}^{\infty} X_c\left(\omega + \frac{2\pi r}{T}\right) \quad r = 0, \pm 1, \pm 2, \pm 3, \dots \quad (12.37)$$

Por tanto, si la señal continua tiene una transformada de Fourier, cuya curva de módulos, por ejemplo, se representa en la figura 12.7.

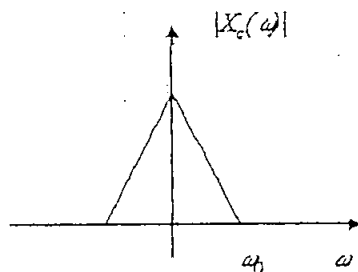


Figura 12.7 Transformada de Fourier de una señal continua determinada

Entonces la curva de módulos de la transformada de la secuencia obtenida por muestreo, será periódica  $2\pi/T$  tal como queda reflejado en la figura 12.8

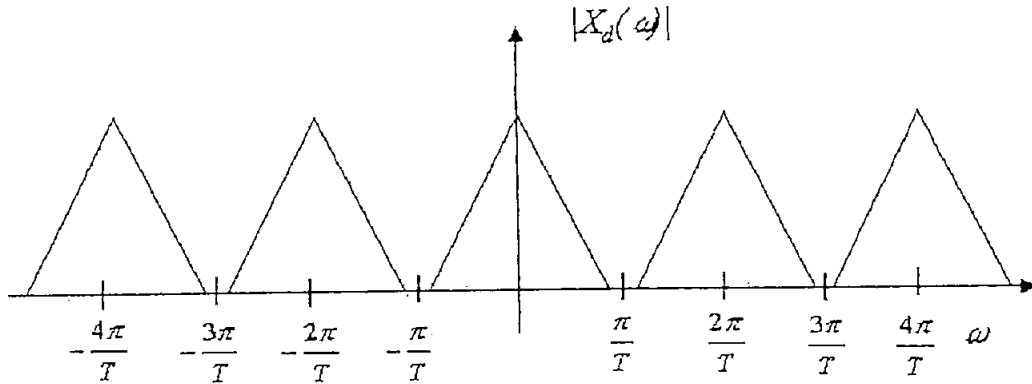


Figura 12. 8 Transformada discreta de Fourier de la señal continua con un muestreador de periodo T

Llegando así a las condiciones que se deben cumplir en el proceso de muestreo para que se pueda realizar sin pérdida de información. Suponiendo que la señal continua  $x(t)$  con transformada de Fourier  $X_c(\omega)$  sea nula a partir de una determinada frecuencia  $\omega_0$ .

$$X_c(\omega) = 0 \quad |\omega| > \omega_0 \tag{12. 38}$$

Las señales que cumplen esta condición se denominan de banda limitada. Si una señal de este tipo se muestrea con un periodo que verifique:

$$\frac{2\pi}{T} > 2\omega_0 \tag{12. 39}$$

Es decir:

$$T < \frac{\pi}{\omega_0} \tag{12. 40}$$

La transformada de Fourier será, siguiendo el ejemplo, de la forma indicada en la figura 12.8. A  $\pi/T$  se la llama frecuencia angular de Nyquist.

Si por el contrario, el periodo de muestreo es superior a este valor,  $T > \pi/\omega_0$ , la transformada de la secuencia del ejemplo será de la forma indicada en la figura 12.9, donde se puede apreciar que  $X_c(\omega)$  y  $X_d(\omega)$  no coinciden, y por lo tanto el conocimiento de la

## **Anexo 5**

Función de los pines del CODEC AD 1847

Tomado de: **Manual de usuario del Ez- Kit Lite**

# AD1847

## PIN DESCRIPTIONS

### Parallel Interface

Pin Name	PLCC	TQFP	I/O	Description
SCLK	1	39	I/O	Serial Clock. SCLK is a bidirectional signal that supplies the clock as an output to the serial bus when the Bus Master (BM) pin is driven HI and accepts the clock as an input when the BM pin is driven LO. The serial clock output is fixed at 12.288 MHz when XTAL1 is selected, and 11.2896 MHz when XTAL2 is selected. SCLK runs continuously. An AD1847 should always be configured as the serial bus master unless it is a slave in a daisy-chained multiple codec system.
SDFS	6	44	I/O	Serial Data Frame Sync. SDFS is a bidirectional signal that supplies the frame synchronization signal as an output to the serial bus when the Bus Master (BM) pin is driven HI and accepts the frame synchronization signal as an input when the BM pin is driven LO. The SDFS frequency powers up at one half of the AD1847 sample rate (i.e., FRS bit = 0) with two samples per frame and can be programmed to match the sample rate (i.e., FRS bit = 1) with one sample per frame. An AD1847 should always be configured as the serial bus master unless it is a slave in a daisy-chained multiple codec system.
SDI	4	42	I	Serial Data Input. SDI is used by peripheral devices such as the host CPU or a DSP to supply control and playback data information to the AD1847. All control and playback transfers are 16 bits long, MSB first.
SDO	5	43	O	Serial Data Output. SDO is used to supply status/index readback and capture data information to peripheral devices such as the host CPU or a DSP. All status/index readback and capture data transfers are 16 bits long, MSB first. Three-state output driver.
$\overline{\text{RESET}}$	11	5	I	Reset. The $\overline{\text{RESET}}$ signal is active LO. The assertion of this signal will initialize the on-chip registers to their default values. See the "Control Register Definitions" section for a description of the contents of the control registers after $\overline{\text{RESET}}$ is deasserted.
$\overline{\text{PWRDOWN}}$	12	6	I	Powerdown. The $\overline{\text{PWRDOWN}}$ signal is active LO. The assertion of this signal will reset the on-chip control registers (identically to the $\overline{\text{RESET}}$ signal) and will also place the AD1847 in a low power consumption mode. $V_{\text{REF}}$ and all analog circuitry are disabled.
BM	33	27	I	Bus Master. The assertion (HI) of this signal indicates that the AD1847 is the serial bus master. The AD1847 will then supply the serial clock (SCLK) and the frame sync (SDFS) signals for the serial bus. One and only one AD1847 should always be configured as the serial bus master. If BM is connected to logic LO, the AD1847 is configured as a bus slave, and will accept the SCLK and SDFS signals as inputs. An AD1847 should only be configured as a serial bus slave when an AD1847 serial bus master already exists, in daisy-chained multiple codec systems.
TSO	7	1	O	Time Slot Output. This signal is asserted HI by the AD1847 coincidentally with the LSB of the last time slot used by the AD1847. Used in daisy-chained multiple codec systems.
TSI	8	2	I	Time Slot Input. The assertion of this signal indicates that the AD1847 should immediately use the next three time slots (TSSEL = 1) or the next six time slots (TSSEL = 0) and then activate the TSO pin to enable the next device down the TDM chain. TSI should be driven LO when the AD1847 is the bus master or in single codec systems. Used in daisy-chained multiple codec systems.
CLKOUT	44	38	O	Clock Output. This signal is the buffered version of the crystal clock output and the frequency is dependent on which crystal is selected. This pin can be three-stated by driving the BM pin LO or by programming the CLKTS bit in the Pin Control Register. See the "Control Registers" section for more details. The CLKOUT frequency is 12.288 MHz when XTAL1 is selected and 16.9344 MHz when XTAL2 is selected.

### Analog Signals

Pin Name	PLCC	TQFP	I/O	Description
L_LINE1	23	17	I	Left Line Input #1. Line level input for the #1 left channel.
R_LINE1	17	11	I	Right Line Input #1. Line level input for the #1 right channel.
L_LINE2	22	16	I	Left Line Input #2. Line level input for the #2 left channel.
R_LINE2	18	12	I	Right Line Input #2. Line level input for the #2 right channel.
L_AUX1	26	20	I	Left Auxiliary Input #1. Line level input for the AUX1 left channel.
R_AUX1	27	21	I	Right Auxiliary Input #1. Line level input for the AUX1 right channel.
L_AUX2	32	26	I	Left Auxiliary Input #2. Line level input for the AUX2 left channel.
R_AUX2	31	25	I	Right Auxiliary Input #2. Line level input for the AUX2 right channel.
L_OUT	30	24	O	Left Line Output. Line level output for the left channel.
R_OUT	28	22	O	Right Line Output. Line level output for the right channel.

## Miscellaneous

Pin Name	PLCC	TQFP	I/O	Description
XTAL1I	40	34	I	24.576 MHz Crystal #1 Input.
XTAL1O	41	35	O	24.576 MHz Crystal #1 Output.
XTAL2I	42	36	I	16.9344 MHz Crystal #2 Input.
XTAL2O	43	37	O	16.9344 MHz Crystal #2 Output.
XCTL1:O	37 & 36	31 & 30	O	External Control. These TTL signals reflect the current status of register bits inside the AD1847. They can be used for signaling or to control external logic.
V <sub>REF</sub>	16	10	O	Voltage Reference. Nominal 2.25 volt reference available externally as a voltage datum for dc-coupling and level-shifting. V <sub>REF</sub> should not have any signal dependent load.
V <sub>REFI</sub>	15	9	I	Voltage Reference Internal. Voltage reference filter point for external bypassing only.
L_FILT	21	15	I	Left Channel Filter Capacitor. This pin requires a 1.0 $\mu$ F capacitor to analog ground for proper operation.
R_FILT	19	13	I	Right Channel Filter Capacitor. This pin requires a 1.0 $\mu$ F capacitor to analog ground for proper operation.
NC	29	23		No Connect. Do not connect.

## Power Supplies

Pin Name	PLCC	TQFP	I/O	Description
V <sub>CC</sub>	13 & 25	7 & 19	I	Analog Supply Voltage (+5 V).
GNDA	14, 20, 24	8, 14, 18	I	Analog Ground.
V <sub>DD</sub>	2, 9, 34, 39	40, 3, 28, 33	I	Digital Supply Voltage (+5 V).
GNDD	3, 10, 35, 38	41, 4, 29, 32	I	Digital Ground.

(Continued from page 1)

The  $\Sigma\Delta$  DACs are preceded by a digital interpolation filter. An attenuator provides independent user volume control over each DAC channel. Nyquist images are removed from the DACs' analog stereo output by on-chip switched-capacitor and continuous-time filters. Two stereo pairs of auxiliary line-level inputs can also be mixed in the analog domain with the DAC output.

The AD1847 serial data interface uses a Time Division Multiplex (TDM) scheme that is compatible with DSP serial ports configured in Multi-Channel Mode with 32 16-bit time slots (i.e., SPORT0 on the ADSP-2101, ADSP-2115, etc.).

## AUDIO FUNCTIONAL DESCRIPTION

This section overviews the functionality of the AD1847 and is intended as a general introduction to the capabilities of the device. As much as possible, detailed reference information has been placed in "Control Registers" and other sections. The user is not expected to refer repeatedly to this section.

## Analog Inputs

The AD1847 SoundPort Stereo Codec accepts stereo line-level inputs. All inputs should be capacitively coupled (ac-coupled) to the AD1847. LINE1, LINE2, and AUX1, and post-mixed DAC output analog stereo signals are multiplexed to the internal programmable gain amplifier (PGA) stage.

The PGA following the input multiplexer allows independent selectable gains for each channel from 0 to 22.5 dB in +1.5 dB steps. The Codec can operate either in a global stereo mode or in a global mono mode with left-channel inputs appearing at both channel outputs.

## Analog Mixing

AUX1 and AUX2 analog stereo signals can be mixed in the analog domain with the DAC output. Each channel of each auxiliary analog input can be independently gained/attenuated from +12 dB to -34.5 dB in -1.5 dB steps or completely muted. The post-mixed DAC output is available on L\_OUT and R\_OUT externally and as an input to the ADCs.

Even if the AD1847 is not playing back data from its DACs, the analog mix function can still be active.

## Analog-to-Digital Datapath

The  $\Sigma\Delta$  ADCs incorporate a proprietary fourth-order modulator. A single pole of passive filtering is all that is required for antialiasing the analog input because of the ADC's high 64 times oversampling ratio. The ADCs include digital decimation filters that low-pass filter the input to  $0.4 \times F_s$ . ("F<sub>s</sub>" is the word rate or "sampling frequency.") ADC input overrange conditions will cause status bits to be set that can be read.

## Digital-to-Analog Datapath

The  $\Sigma\Delta$  DACs contain a programmable attenuator and a low-pass digital interpolation filter. The anti-imaging interpolation filter oversamples and digitally filters the higher frequency images. The attenuator allows independent control of each DAC channel from 0 dB to -94.5 dB in 1.5 dB steps plus full mute. The DACs'  $\Sigma\Delta$  noise shapers also oversample and convert the signal to a single-bit stream. The DAC outputs are then filtered in the analog domain by a combination of switched-capacitor and continuous-time filters. These filters remove the very high frequency components of the DAC bitstream output. No external components are required.



# **Anexo 6**

Subrutina de inicialización del CODEC AD 1847

Tomado de : **Curso DSP, Ing. Fernando Vásquez**

```
*****
7.dsp - Modulo para inicializacion del CODEC AD1847 conectado con el
        ADSP-2181 en el EZ-KIT Lite
```

Basado en el programa CODEC.DSP incluido en el programa monitor del EZ-KIT Lite

Fernando Vásquez - DSP Soluciones - Oct. 99

Entrada -----> ax0 = frecuencia de muestreo

frecuencias de muestreo posibles

0xc850 = 8		0xc851 = 5.5125		0xc852 = 16
0xc853 = 11.025		0xc854 = 27.42857		0xc855 = 18.9
0xc856 = 32		0xc857 = 22.05		0xc859 = 37.8
0xc85b = 44.1		0xc85c = 48		0xc85d = 33.075
0xc85e = 9.6		0xc85f = 6.615		

5, i7 y m7 son utilizados para rx y tx de datos con el CODEC no pueden ser utilizados en el programa principal

6 -----> utilizado para rx. de datos en el CODEC

7 -----> utilizado para tx. de datos en el CODEC

7=1 -----> puede ser utilizado pero no cambiado

registros modificados:

```
SPORT0_Autobuf
SPORT0_Control_Reg
System_Control_Reg
```

```
dm (SPORT0_Autobuf)
dm (SPORT0_RFSDIV)
dm (SPORT0_SCLKDIV)
dm (SPORT0_Control_Reg)
dm (SPORT0_TX_Channels0)
dm (SPORT0_TX_Channels1)
dm (SPORT0_RX_Channels0)
dm (SPORT0_RX_Channels1)
dm (System_Control_Reg)
```

```
LE/RAM CODEC;
UDE <c:\adi_dsp\21xx\include\reg_ctrl.dsp>;
```

```
T L_CTRL_REG_VALUE= 0xc002; /* EZ-KIT values */
T R_CTRL_REG_VALUE= 0xc102;
```

```
Y next_cmd; { CODEC initialization commands }
Y codecInit; { CODEC initialization }
Y codecStop; { CODEC stop: irq and auto-buffering }
```

```
AL stat_flag; { CODEC init status }
AL rx_buf, tx_buf;
```

```
DM/CIRC rx_buf[3]; { Status + L data + R data }
DM/CIRC tx_buf[3]; { Cmd + L data + R data }
DM/CIRC init_cmds[13]; { Comandos de inicializacion del AD1847 }
DM stat_flag;
```

```
tx_buf: 0xc000, 0x0000, 0x0000; /* Initially set MCE */
```

```
init_cmds:
```

```
L_CTRL_REG_VALUE,
```

```
{ EZ-KIT: 0xc00f
  Left input control reg
  b7-6: 0=left line 1
        1=left aux 1
        2=left line 2
        3=left line 1 post-mixed loopback
  b5-4: res
  b3-0: left input gain x 1.5 dB
```

```
R_CTRL_REG_VALUE,
```

```
{ EZ-KIT: 0xc10f
  Right input control reg
  b7-6: 0=right line 1
        1=right aux 1
```

```

    2=right line 2
    3=right line 1 post-mixed loopback
b5-4: res
b3-0: right input gain x 1.5 dB
}
0xc288, {
    left aux 1 control reg
    b7 : 1=left aux 1 mute
    b6-5: res
    b4-0: gain/atten x 1.5, 08= 0dB, 00= 12dB
}
0xc388, {
    right aux 1 control reg
    b7 : 1=right aux 1 mute
    b6-5: res
    b4-0: gain/atten x 1.5, 08= 0dB, 00= 12dB
}
0xc488, {
    left aux 2 control reg
    b7 : 1=left aux 2 mute
    b6-5: res
    b4-0: gain/atten x 1.5, 08= 0dB, 00= 12dB
}
0xc588, {
    right aux 2 control reg
    b7 : 1=right aux 2 mute
    b6-5: res
    b4-0: gain/atten x 1.5, 08= 0dB, 00= 12dB
}
0xc680, {
    left DAC control reg
    b7 : 1=left DAC mute
    b6 : res
    b5-0: attenuation x 1.5 dB
}
0xc780, {
    right DAC control reg
    b7 : 1=right DAC mute
    b6 : res
    b5-0: attenuation x 1.5 dB
}
0xc850, {
    data format register
    b7 : res
    b5-6: 0=8-bit unsigned linear PCM
           1=8-bit u-law companded
           2=16-bit signed linear PCM
           3=8-bit A-law companded
    b4 : 0=mono, 1=stereo
    b0-3: 0= 8.
           1= 5.5125
           2= 16.
           3= 11.025
           4= 27.42857
           5= 18.9
           6= 32.
           7= 22.05
           8= .
           9= 37.8
           a= .
           b= 44.1
           c= 48.
           d= 33.075
           e= 9.6
           f= 6.615
    (b0) : 0=XTAL1 24.576 MHz; 1=XTAL2 16.9344 MHz
}
0xc909, {
    interface configuration reg
    b7-4: res
    b3 : 1=autocalibrate
    b2-1: res
    b0 : 1=playback enabled
}
0xca00, {
    pin control reg
    b7 : logic state of pin XCTL1
    b6 : logic state of pin XCTL0
}

```



```
dm(init_cmds)
```

```
memory mapped control registers:
```

```
dm (Sport0_Autobuf_Ctrl)
dm (SPORT0_RFSDIV)
dm (SPORT0_SCLKDIV)
dm (SPORT0_Control_Reg)
dm (Sport0_Tx_Words0)
dm (Sport0_Tx_Words1)
dm (Sport0_Rx_Words0)
dm (Sport0_Rx_Words1)
dm (Sys_Ctrl_Reg)
```

```
alls : none
```

```
*****}
```

```
Init:
```

```
dm(init_cmds+8)=ax0;
{ shut down sport 0 during re-initialization }
call codecStop;

ifc = b#000000111111111;      { clear pending interrupt }
nop;
m7=1;
m3=1;
i6 = ^rx_buf;
l6 = %rx_buf;
i7 = ^tx_buf;
l7 = %tx_buf;
i3 = ^init_cmds;
l3 = %init_cmds;
```

```
----- SERIAL PORT #0 STUFF -----}
```

```
ax0 = 0x0fef; dm (Sport0_Autobuf_Ctrl) = ax0;
{
  ||| ||| -/| /| -/| /| +- receive autobuffering 0=off, 1=on
  ||| ||| | | | | +- transmit autobuffering 0=off, 1=on
  ||| ||| | | | | +---- receive m?
  ||| ||| | | | | m5
  ||| ||| | | | | +----- receive i?
  ||| ||| | | | | i5
  ||| ||| | | | | |
  ||| ||| | | | | +----- transmit m?
  ||| ||| | | | | m5
  ||| ||| | | | | +----- transmit i?
  ||| ||| | | | | i6
  ||| ||| | | | | |
  ||| ||| | | | | +----- BIASRND MAC biased rounding control bit
  ||| ||| | | | | +----- 0
  ||| ||| | | | | +----- CLKODIS CLKOUT disable control bit
  ||| ||| | | | | +----- 0
}
```

```
ax0 = 0; dm (SPORT0_RFSDIV) = ax0;
{ RFSDIV = SCLK Hz/RFS Hz - 1 }
ax0 = 0; dm (SPORT0_SCLKDIV) = ax0;
{ SCLK = CLKOUT / (2 (SCLKDIV + 1) ) }
ax0 = b#1000011000001111; dm (SPORT0_Ctrl_Reg) = ax0;
{ multichannel
  ||| +- -/| ||| | +/ + - -/| | number of bit per word - 1
  ||| | | | | | | | | | | | | = 15
  ||| | | | | | | | | | | | | |
  ||| | | | | | | | | | | | | | +----- 0=right just, 0=fill; 1=right just, signed
  ||| | | | | | | | | | | | | | | 2=compand u-law; 3=compand A-law
  ||| | | | | | | | | | | | | | | +----- receive framing logic 0=pos, 1=neg
  ||| | | | | | | | | | | | | | | +----- transmit data valid logic 0=pos, 1=neg
  ||| | | | | | | | | | | | | | | +----- RFS 0=ext, 1=int
  ||| | | | | | | | | | | | | | | +----- multichannel length 0=24, 1=32 words
  ||| | | | | | | | | | | | | | | +----- frame sync to occur this number of clock
  ||| | | | | | | | | | | | | | | | cycle before first bit
  ||| | | | | | | | | | | | | | | +----- ISCLK 0=ext, 1=int
  ||| | | | | | | | | | | | | | | +----- multichannel 0=disable, 1=enable
}
{ non-multichannel
  ||| | | | | | | | | | | | | | | +- - -/| | number of bit per word - 1
  ||| | | | | | | | | | | | | | | | = 15
  ||| | | | | | | | | | | | | | | |
}
```



```

| | | +----- | SPORT0 trx
| | | +----- | IRQL0
| | | +----- | IRQL1
| | | +----- | IRQL2

```

```

}

```

```

{ set flag, disable tx ints }
ax0 = 0;
dm(stat_flag) = ax0;

```

The following codes make sure the address register wrap around is sync with the frame and verifies the codec is responding correctly. }

```

ax0=0; { tx 0 to left and right }
dm(tx_buf+1) = ax0;
dm(tx_buf+2) = ax0;
ax0=dm(init_cmds+11); { codec to switch to 32 slot per frame }
dm(tx_buf) = ax0;

ax0=dm(i7.m7); { start interrupt }
tx0=ax0;

idle;

ax0 = 0x8dac; { write ac to Digital mix control }
dm (tx_buf) = ax0;

idle;
ax0 = 0x2d00; { readback digital mix control }
dm (tx_buf) = ax0;
idle;
idle; { for 32 slot, readback is in next frame }
ax0 = dm (rx_buf);
ay0 = 0x2dac;
none = ax0 - ay0;
toggle fl1;}
if ne jump again; { if not reading back, not in sync }

ax0 = 0x8d50; { write 50 to Digital mix control }
dm (tx_buf) = ax0;

idle;
ax0 = 0x2d00; { readback digital mix control }
dm (tx_buf) = ax0;
idle;
idle; { for 32 slot, readback is in next frame }
ax0 = dm (rx_buf);
ay0 = 0x2d50;
none = ax0 - ay0;
if ne jump again; { if not reading back, not in sync }

{ clear flag }
ax0 = 1;
dm(stat_flag) = ax0;

ax0 = dm (init_cmds + 12); { codec to switch to 32 slot per frame }
dm (tx_buf) = ax0;

```

```

_init:
idle;
ax0 = dm (stat_flag); { wait for entire init }
none = pass ax0; { buffer to be sent to }
if ne jump check_init; { the codec }

```

```

_aci1:
idle;
ay0 = 2;
ax0 = dm (rx_buf); { once initialized, wait for codec }
ar = ax0 and ay0; { to come out of autocalibration }
if eq jump check_aci1; { wait for bit set }

```

```

_aci2:

```

```

idle;
ax0 = dm (rx_buf);           { wait for bit clear }
ar = ax0 and ay0;
if ne jump check_aci2;

ay0 = 0xbf3f;                { unmute left DAC }
ax0 = dm (init_cmds + 6);
ar = ax0 AND ay0;
dm (tx_buf) = ar;
idle;

ax0 = dm (init_cmds + 7);    { unmute right DAC }
ar = ax0 AND ay0;
dm (tx_buf) = ar;
idle;

ax0 = 0xa000;                { control word to clear over-range flags }
dm (tx_buf) = ax0;
idle;

ifc = b#00000011111111;     { clear any pending interrupt }
nop;

imask = 0x20;

```

-----  
wait for interrupt and loop forever  
-----}

rts;

-----  
transmit interrupt used for Codec initialization

#### REGISTER USAGE SUMMARY:

```

input   : m3 = 1
update  : i3
output  : tx0
destroy : none (second register bank)
keep    : none
memory  : dm (tx_buf), dm (stat_flg), init_cmds
calls   : none

```

-----}

```

cmd:
ena sec_reg;
ax0=dm (i3, m3);           { fetch next control word and }
dm(tx_buf)=ax0;           { place in transmit slot 0   }
ax0=i3;
ay0=~init_cmds;
ar=ax0-ay0;
if gt rti;                { rti if more control words still waiting }
ax0=0xaf00;               { else set done flag and }
dm(tx_buf)=ax0;           { remove MCE if done initialization }
ax0=0;
dm(stat_flg)=ax0;         { reset status flag }
rti;

```

OD;



# **Anexo 7**

Propiedades de linealidad, causalidad e invariabilidad en  
el tiempo de un sistema

Tomado de: **OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en  
Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000**

Retrasando  $n_0$  muestras la salida  $y[n]$  obtenemos

$$y[n - n_0] = x[M(n - n_0)]. \quad (2.43)$$

Comparando estas dos salidas, vemos que  $y[n - n_0]$  no es igual a  $y_1[n]$  para todo  $M$  y  $n_0$ , y por tanto el sistema no es invariante.

Es también posible demostrar que un sistema no es invariante encontrando un único contraejemplo que viole la propiedad de invarianza temporal. Un contraejemplo para el caso del compresor puede ser el caso de  $M = 2$ ,  $x[n] = \delta[n]$  y  $x_1[n] = \delta[n - 1]$ . Para este conjunto de entradas y este valor de  $M$ ,  $y[n] = \delta[n]$ , pero  $y_1[n] = 0$ , y resulta claro que para este caso,  $y_1[n] \neq y[n - 1]$ .

#### 2.2.4. Causalidad

Se dice que un sistema es causal si para cualquier valor de  $n_0$ , el valor de la secuencia de salida en el índice  $n = n_0$  depende sólo de los valores de la secuencia de entrada para  $n \leq n_0$ . Esto implica que si  $x_1[n] = x_2[n]$  para  $n \leq n_0$ , entonces  $y_1[n] = y_2[n]$  para  $n \leq n_0$ . Es decir, el sistema es *no anticipativo*. El sistema del Ejemplo 2.3 es causal para  $n_d \geq 0$  y es no causal para  $n_d < 0$ . El sistema del Ejemplo 2.4 es causal si  $-M_1 \geq 0$  y  $M_2 \geq 0$ ; en los otros casos es no causal. El sistema del Ejemplo 2.5 es causal, como el acumulador del Ejemplo 2.6 y el sistema no lineal del Ejemplo 2.7. Sin embargo, el sistema del Ejemplo 2.9 es no causal si  $M > 1$  ya que  $y[1] = x[M]$ . En el siguiente ejemplo se presenta otro sistema no causal.

#### Ejemplo 2.10. Los sistemas de diferencias progresivas y regresivas

Consideremos el *sistema de diferencias progresivas* definido por la relación

$$y[n] = x[n + 1] - x[n] \quad (2.44)$$

Este sistema es no causal, ya que el valor actual de la salida depende de un valor futuro de la entrada. La violación de la causalidad se puede demostrar considerando las dos entradas  $x_1[n] = \delta[n - 1]$  y  $x_2[n] = 0$ , y sus correspondientes salidas  $y_1[n] = \delta[n] - \delta[n - 1]$  e  $y_2[n] = 0$ . Nótese que  $x_1[n] = x_2[n]$  para  $n \leq 0$ , por lo que la definición de causalidad exige que  $y_1[n] = y_2[n]$  para  $n \leq 0$ , lo que claramente no es cierto para  $n = 0$ . Así, mediante este contraejemplo, hemos demostrado que el sistema no es causal.

El *sistema de diferencias regresivas*, definido así:

$$y[n] = x[n] - x[n - 1], \quad (2.45)$$

tiene una salida que depende sólo de los valores presente e inmediatamente anterior de la entrada. Como no hay forma de que la salida en un instante concreto  $y[n_0]$  incorpore valores de la entrada para  $n > n_0$ , el sistema es causal.

#### 2.2.5. Estabilidad

Un sistema es estable en el sentido de entrada acotada, salida acotada (BIBO, *Bounded-Input, Bounded-Output*), si y sólo si cualquier secuencia acotada a su entrada produce una secuencia de salida acotada. Se dice que la entrada  $x[n]$  está acotada si existe un valor finito positivo fijo,  $B_x$ , tal que

$$|x[n]| \leq B_x < \infty, \quad \text{para todo } n. \quad (2.46)$$

La estabilidad requiere que, para cualquier entrada acotada, exista un valor finito positivo fijo,  $B_y$ , tal que

$$|y[n]| \leq B_y < \infty, \quad \text{para todo } n. \quad (2.47)$$

Es importante tener en cuenta que las propiedades que hemos definido en esta sección son propiedades de los *sistemas*, y no de las entradas a dichos sistemas. Es decir, podemos ser capaces de encontrar entradas para las que las propiedades se mantengan, pero la existencia de una propiedad para algunas entradas no significa que el sistema tenga esa propiedad. Para que el sistema posea esa propiedad, debe cumplirse para *todas* las entradas. Por ejemplo, un sistema inestable puede tener algunas entradas acotadas para las que la salida esté acotada, pero para que el sistema tenga la propiedad de estabilidad, debe cumplirse que para *todas* las entradas acotadas, las salidas correspondientes están acotadas. Si pudiéramos encontrar tan sólo una entrada para la que no se mantuviera esa propiedad, habríamos demostrado que el sistema *no* posee esa propiedad. El siguiente ejemplo ilustra la comprobación de la estabilidad para algunos de los sistemas que hemos definido.

### Ejemplo 2.11. Comprobación de la estabilidad e inestabilidad

El sistema del Ejemplo 2.5 es estable. Para demostrarlo supongamos que la entrada  $x[n]$  está acotada de forma que  $|x[n]| \leq B_x$  para todo  $n$ . Entonces  $|y[n]| = |x[n]|^2 \leq B_x^2$ . Podemos escoger por tanto  $B_y = B_x^2$  para demostrar que  $y[n]$  está acotada.

Análogamente, podemos ver que el sistema definido en el Ejemplo 2.7 es inestable, ya que  $y[n] = \log_{10}(|x[n]|) = -\infty$  para todos los valores de índice temporal en los que  $x[n] = 0$ , incluso aunque la salida no esté limitada para todas las muestras de la entrada distintas de cero.

El acumulador, definido en el Ejemplo 2.6 mediante la Ecuación (2.29) tampoco es estable. Por ejemplo, consideremos el caso de  $x[n] = u[n]$ , entrada claramente limitada por  $B_x = 1$ . Para esta entrada, la salida del acumulador es

$$y[n] = \sum_{k=-\infty}^n u[k] \quad (2.48)$$

$$= \begin{cases} 0, & n < 0, \\ (n + 1), & n \geq 0. \end{cases} \quad (2.49)$$

No existe ningún valor finito  $B_y$  tal que  $(n + 1) \leq B_y < \infty$  para todo  $n$ , y por tanto, el sistema es inestable.

Utilizando argumentos similares, se puede demostrar que los sistemas de los Ejemplos 2.3, 2.4, 2.9 y 2.10 son todos estables.

## 2.3. SISTEMAS LINEALES E INVARIANTES CON EL TIEMPO

Una clase de sistemas particularmente importante es la de los sistemas lineales e invariantes con el tiempo. La combinación de estas dos propiedades conduce a representaciones especialmente convenientes de los sistemas que las cumplen. Y lo que es más importante, esta clase de sistemas tiene importantes aplicaciones en tratamiento de señales. La clase de los sistemas lineales está definida por el principio de superposición que se indica en la Ecuación (2.27). Si la propiedad de linealidad se combina con la representación general de una secuencia como una combinación lineal de impulsos retrasados, como se indica en la Ecuación (2.6), se puede concluir que un sistema lineal queda completamente caracterizado por su respuesta al impulso. Concretamente, sea  $h_k[n]$  la respuesta del sistema a la entrada  $\delta[n - k]$ , un impulso en  $n = k$ . Entonces, aplicando la Ecuación (2.6),

$$y[n] = T \left\{ \sum_{k=-\infty}^{\infty} x[k] \delta[n - k] \right\}. \quad (2.50)$$

Aplicando el principio de superposición de la Ecuación (2.27), podemos escribir

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]T\{\delta[n-k]\} = \sum_{k=-\infty}^{\infty} x[k]h_k[n]. \quad (2.51)$$

De acuerdo con la Ecuación (2.51), la respuesta del sistema a cualquier entrada se puede expresar en términos de las respuestas del sistema a las secuencias  $\delta[n-k]$ . Si sólo se impone la propiedad de linealidad,  $h_k[n]$  dependerá tanto de  $n$  como de  $k$ , en cuyo caso la utilidad computacional de la Ecuación (2.51) es limitada. Obtenemos un resultado más útil si imponemos la restricción adicional de invarianza temporal.

La propiedad de invarianza temporal implica que si  $h[n]$  es la respuesta a  $\delta[n]$ , entonces la respuesta a  $\delta[n-k]$  es  $h[n-k]$ . Con esta restricción adicional, la Ecuación (2.51) se convierte en

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]. \quad (2.52)$$

Como consecuencia de la Ecuación (2.52), los sistemas lineales e invariantes con el tiempo (que abreviaremos algunas veces como LTI, *Linear Time-Invariant*), están completamente caracterizados por su respuesta al impulso  $h[n]$  en el sentido de que dada  $h[n]$  es posible utilizar la Ecuación (2.52) para calcular la salida  $y[n]$  debida a cualquier entrada  $x[n]$ .

La Ecuación (2.52) se denomina comúnmente *suma de convolución*. Si  $y[n]$  es una secuencia cuyos valores se relacionan con los valores de las secuencias  $h[n]$  y  $x[n]$  como indica la Ecuación (2.52), diremos que  $y[n]$  es la convolución de  $x[n]$  con  $h[n]$ , y lo representaremos mediante la notación

$$y[n] = x[n] * h[n]. \quad (2.53)$$

La operación de convolución en tiempo discreto toma dos secuencias  $x[n]$  y  $h[n]$ , y produce una tercera secuencia  $y[n]$ . La Ecuación (2.52) expresa cada muestra de la salida en función de todas las muestras de las secuencias de entrada y respuesta al impulso.

La obtención de la Ecuación (2.52) sugiere la interpretación de que la muestra de entrada en  $n = k$ , representada como  $x[k]\delta[n-k]$ , es transformada por el sistema en una secuencia de salida  $x[k]h[n-k]$  para  $-\infty < n < \infty$  y que, para cada valor de  $k$ , estas secuencias se superponen para formar la secuencia total de salida. La Figura 2.8 ilustra esta interpretación. En ella se muestra una respuesta al impulso, una secuencia simple con tres valores distintos de cero, las salidas individuales debidas a cada muestra y la salida compuesta debida a todas las muestras de la secuencia de entrada. Concretamente,  $x[n]$  se puede descomponer como la suma de tres secuencias:  $x[-2]\delta[n+2]$ ,  $x[0]\delta[n]$  y  $x[3]\delta[n-3]$ , que representan los tres valores distintos de cero de la secuencia  $x[n]$ . Las secuencias  $x[-2]h[n+2]$ ,  $x[0]h[n]$  y  $x[3]h[n-3]$  son las respuestas del sistema a  $x[-2]\delta[n+2]$ ,  $x[0]\delta[n]$  y  $x[3]\delta[n-3]$ , respectivamente. La respuesta a la señal  $x[n]$  es la suma de estas tres respuestas individuales.

Aunque la expresión de la suma de convolución es análoga a la integral de convolución de la teoría de sistemas lineales en tiempo continuo, no debe verse la suma de convolución como una aproximación a la integral de convolución. La integral de convolución tiene un papel más bien teórico en la teoría de sistemas lineales en tiempo continuo. Pero veremos que la suma de convolución, además de su importancia teórica, sirve a menudo como una realización explícita de un sistema en tiempo discreto. Por tanto, es importante detenerse a considerar las propiedades de la suma de convolución en cálculos reales.

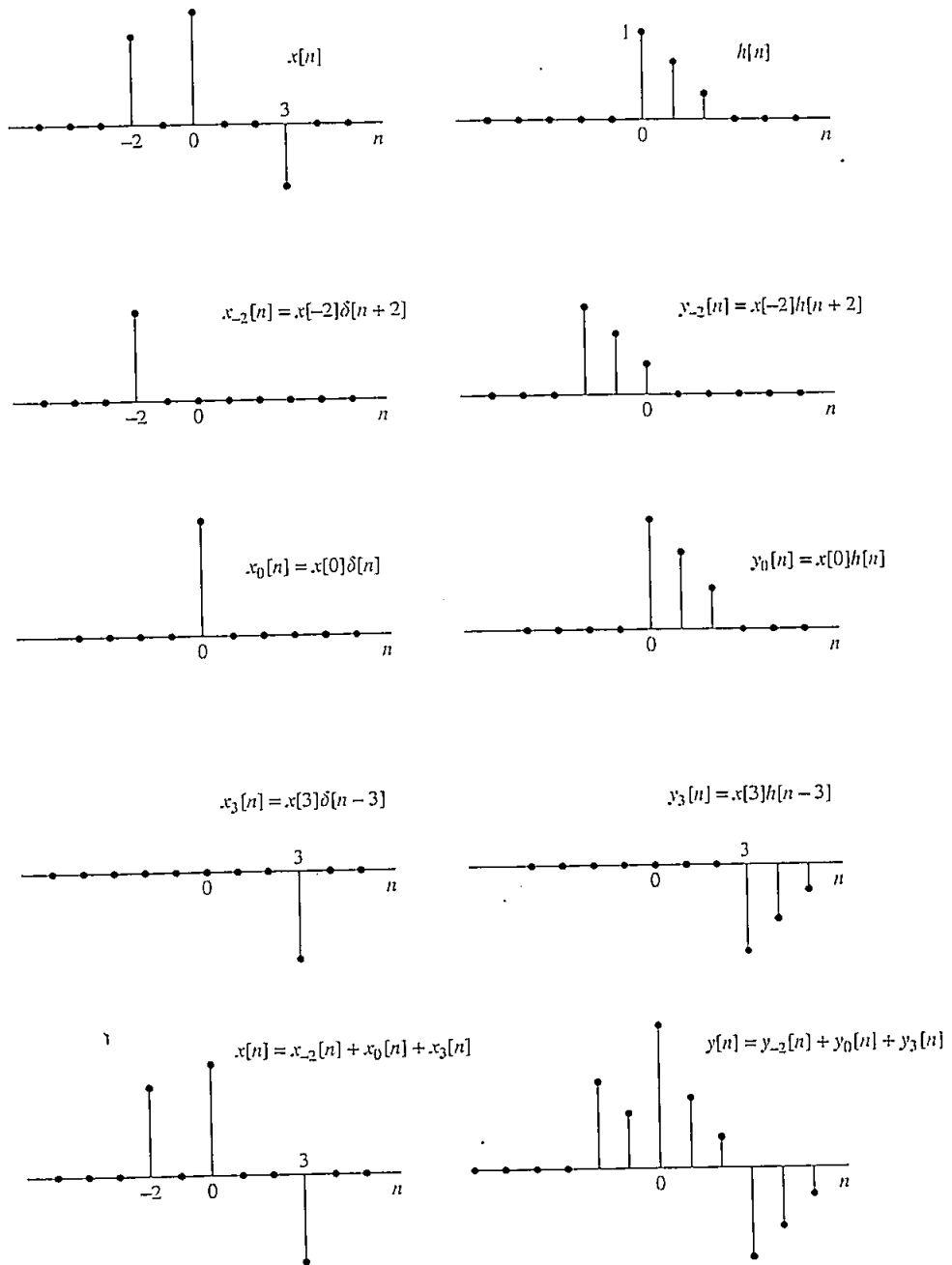


Figura 2.8. Representación de la salida de un sistema lineal invariante con el tiempo como la superposición de las respuestas a cada muestra de la entrada.

La anterior interpretación de la Ecuación (2.52) resalta el hecho de que la suma de convolución es un resultado directo de las propiedades de linealidad e invarianza temporal. Sin embargo, si observamos la Ecuación (2.52) desde una perspectiva ligeramente diferente llegamos a una interpretación de particular utilidad computacional. Vista como una fórmula para calcular un solo valor de la secuencia de salida, la Ecuación (2.52) dice que  $y[n]$  (el

$n$ -ésimo valor de la salida) se obtiene multiplicando la secuencia de entrada (expresada en función de  $k$ ) por la secuencia cuyos valores son  $h[n - k]$ ,  $-\infty < k < \infty$ , y después, para cada valor fijo de  $n$ , se suman todos los valores de los productos  $x[k]h[n - k]$ , siendo  $k$  el índice de la suma. Por tanto, la operación de convolución de dos secuencias requiere realizar los cálculos para todos los valores de  $n$ , generándose así la secuencia completa de salida  $y[n]$ ,  $-\infty < n < \infty$ . La clave para realizar los cálculos de la Ecuación (2.52) y obtener  $y[n]$  es saber cómo se forma la secuencia  $h[n - k]$ ,  $-\infty < k < \infty$ , para todos los valores de  $n$  de interés. Con este fin, es útil tener en cuenta que

$$h[n - k] = h[-(k - n)]. \quad (2.54)$$

La interpretación de la Ecuación (2.54) se realiza mejor con ayuda de un ejemplo.

### Ejemplo 2.12. Cálculo de la suma de convolución

Supongamos que  $h[k]$  es la secuencia que se muestra en la Figura 2.9(a), y que deseamos calcular  $h[n - k] = h[-(k - n)]$ . Definimos  $h_1[k] = h[-k]$ , como se muestra en la Figura 2.9(b). Definimos seguidamente  $h_2[k]$  como  $h_1[k]$  retrasada  $n$  muestras en el eje  $k$ , es decir,  $h_2[k] = h_1[k - n]$ . La Figura 2.9(c) muestra la secuencia que resulta al retrasar  $n$  muestras la secuencia de la Figura 2.9(b). Utilizando la relación entre  $h_1[k]$  y  $h[k]$ , se puede demostrar que  $h_2[k] = h_1[k - n] = h[-(k - n)] = h[n - k]$  y por tanto, la figura de abajo es la señal deseada. Resumiendo, para calcular  $h[n - k]$  a partir de  $h[k]$  reflejamos primero  $h[k]$  en el tiempo alrededor del origen  $k = 0$  y después retrasamos  $n$  muestras la señal reflejada.

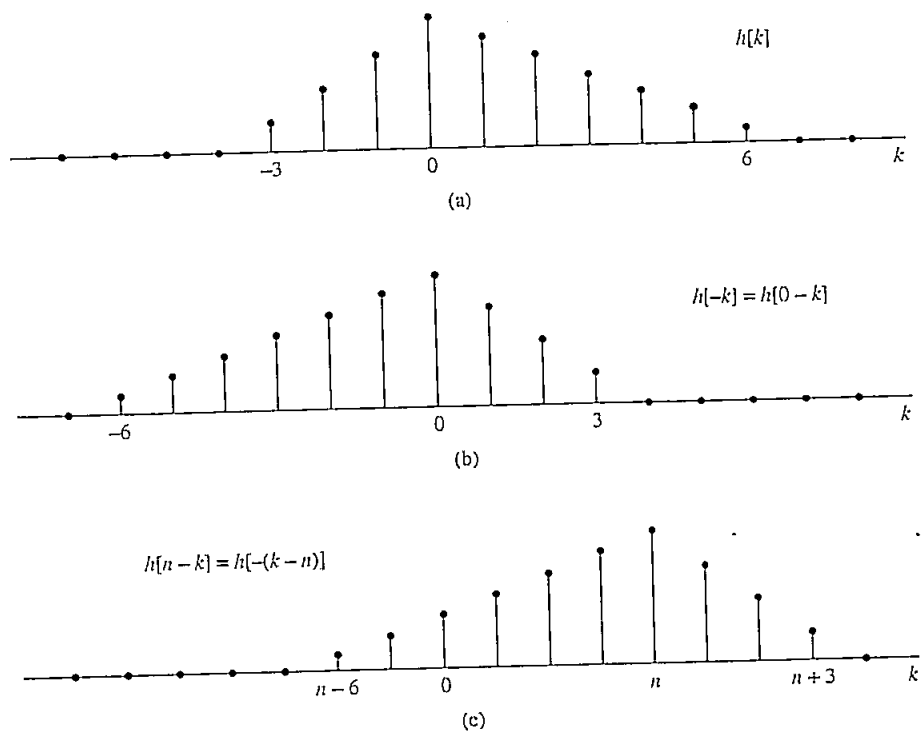


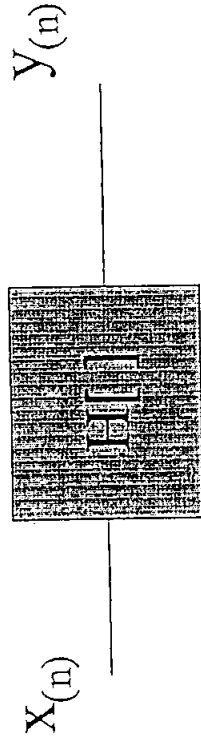
Figura 2.9. Formación de la secuencia  $h[n - k]$ . (a) La secuencia  $h[k]$  en función de  $k$ . (b) La secuencia  $h[-k]$  en función de  $k$ . (c) La secuencia  $h[n - k] = h[-(k - n)]$  en función de  $k$  para  $n = 4$ .

# Anexo 8

## Transformada Z

Tomado de : [www.elo.utfsm.cl/elo385/documentos/intro-dsp.pdf](http://www.elo.utfsm.cl/elo385/documentos/intro-dsp.pdf)

Sea el siguiente sistema.



Si se define la secuencia aplicada a la entrada por  $X_{(n)} \rightarrow X_{(z)}$  y la obtenida en la salida por  $Y_{(n)} \rightarrow Y_{(z)}$  se tiene que:

$$Y_{(z)}$$

$$H_{(z)} = \text{-----}$$

$$X_{(z)}$$

Donde:  $X_{(z)}$ ,  $Y_{(z)}$  y  $H_{(z)}$  representan a la transformada  $Z$  de  $X_{(n)}$ ,  $Y_{(n)}$  y de la función de transferencia  $H_{[ ]}$  respectivamente.

$H_{(z)}$  es un polinomio en potencias no positivas de  $z$ , y según su naturaleza define filtros del tipo: no Recurrentes y filtros Recurrentes.



Para definir la transformada  $Z$  de una función en tiempo continuo  $x(t)$  es necesario muestrear la señal y considerar los respectivos valores muestreados de  $x(t)$ , es decir  $x(0)$ ,  $x(T)$ ,  $x(2T)$  ...,  $x(nT)$ , donde  $T$  representa el período de muestreo de la señal.

Luego  $x(t) \rightarrow x(kT) = x(n)$ , con  $k = 0, 1, 2, 3, \dots$

La transformada  $Z$  de  $x(n)$  se define como :

$$Z[x(n)] = X(z) = \sum_{k=0}^{\infty} x(kT) \cdot z^{-k}, \text{ con } k = 0, 1, 2, \dots \dots 8.$$

$$kT = n \rightarrow Z[x(n)] = X(z) = \sum_{n=0}^{\infty} x(n) \cdot z^{-n}, \text{ con } n = 0, 1, 2, \dots \dots 8.$$

Definida de esta forma, se conoce como transformada  $Z$  unilateral. La transformada  $Z$  unilateral presenta, en la mayoría de aplicaciones en ingeniería, una solución apropiada en forma cerrada en su región de convergencia. Puede observarse que  $X(z)$  es una serie infinita en  $z^{-1}$  que converge fuera del círculo  $|z| = R$ , donde  $R$  se conoce como radio de convergencia. Al utilizar el método de la transformada  $Z$  para resolver problemas en tiempo discreto no es necesario especificar los valores de  $z$  para los cuales  $X(z)$  converge.

# Anexo 9

## Respuesta de un sistema al impulso

Tomado de: **OPPENHEIM Alan; SCHAFER Ronald. Tratamiento de Señales en Tiempo Discreto. Segunda Edición. Editorial Prentice Hall. Iberia. 2000**

### Ejemplo 2.28. Determinación de la respuesta al impulso a partir de la respuesta en frecuencia

La respuesta en frecuencia de un filtro paso alto con retardo es

$$H(e^{j\omega}) = \begin{cases} e^{-j\omega n_d}, & \omega_c < |\omega| < \pi, \\ 0, & |\omega| < \omega_c, \end{cases} \quad (2.179)$$

que se sobrentiende periódica de periodo  $2\pi$ . Esta respuesta en frecuencia se puede expresar así

$$H(e^{j\omega}) = e^{-j\omega n_d}(1 - H_{ip}(e^{j\omega})) = e^{-j\omega n_d} - e^{-j\omega n_d} H_{ip}(e^{j\omega}),$$

siendo  $H_{ip}(e^{j\omega})$  periódica de periodo  $2\pi$  y

$$H_{ip}(e^{j\omega}) = \begin{cases} 1, & |\omega| < \omega_c, \\ 0, & \omega_c < |\omega| < \pi. \end{cases}$$

Utilizando el resultado del Ejemplo 2.22 para obtener la transformada inversa de  $H_{ip}(e^{j\omega})$ , junto con las Propiedades 1 y 2 de la Tabla 2.2 tenemos que

$$\begin{aligned} h[n] &= \delta[n - n_d] - r[n - n_d] \\ &= \delta[n - n_d] - \frac{\text{sen } \omega_c(n - n_d)}{\pi(n - n_d)}. \end{aligned}$$

### Ejemplo 2.29. Determinación de la respuesta al impulso a partir de una ecuación en diferencias

En este ejemplo determinaremos la respuesta al impulso de un sistema lineal e invariante con el tiempo y estable, tal que su entrada  $x[n]$  y su salida  $y[n]$  satisfacen la siguiente ecuación en diferencias lineal de coeficientes constantes

$$y[n] - \frac{1}{2}y[n-1] = x[n] - \frac{1}{4}x[n-1]. \quad (2.180)$$

En el Capítulo 3 veremos que la transformada  $Z$  es más útil que la transformada de Fourier para manejar ecuaciones en diferencias. Sin embargo, este ejemplo nos permite tener una idea de la utilidad de los métodos basados en transformadas en el análisis de sistemas lineales. Para calcular la respuesta al impulso, haremos  $x[n] = \delta[n]$ . Si  $h[n]$  indica la respuesta al impulso, la Ecuación (2.180) se transforma en

$$h[n] - \frac{1}{2}h[n-1] = \delta[n] - \frac{1}{4}\delta[n-1]. \quad (2.181)$$

Aplicando la transformada de Fourier a ambos miembros de la Ecuación (2.181) y utilizando las Propiedades 1 y 2 de la Tabla 2.2, obtenemos

$$H(e^{j\omega}) - \frac{1}{2}e^{-j\omega}H(e^{j\omega}) = 1 - \frac{1}{4}e^{-j\omega}, \quad (2.182)$$

o

$$H(e^{j\omega}) = \frac{1 - \frac{1}{4}e^{-j\omega}}{1 - \frac{1}{2}e^{-j\omega}}. \quad (2.183)$$

Para calcular  $h[n]$ , tenemos que determinar la transformada inversa de  $H(e^{j\omega})$ . Para ello, volveremos a escribir la Ecuación (2.183) de este modo

$$H(e^{j\omega}) = \frac{1}{1 - \frac{1}{2}e^{-j\omega}} - \frac{\frac{1}{4}e^{-j\omega}}{1 - \frac{1}{2}e^{-j\omega}}. \quad (2.184)$$

Aplicando la transformada número 4 de la Tabla 2.3,

$$\left(\frac{1}{2}\right)^n u[n] \xleftrightarrow{\mathcal{F}} \frac{1}{1 - \frac{1}{2}e^{-j\omega}}.$$

Combinando esta transformada con la Propiedad 3 de la Tabla 2.2, obtenemos

$$-\left(\frac{1}{4}\right)\left(\frac{1}{2}\right)^{n-1} u[n-1] \xleftrightarrow{\mathcal{F}} -\frac{\frac{1}{4}e^{-j\omega}}{1 - \frac{1}{2}e^{-j\omega}}. \quad (2.185)$$

Entonces, utilizando la Propiedad 1 de la Tabla 2.2,

$$h[n] = \left(\frac{1}{2}\right)^n u[n] - \left(\frac{1}{4}\right)\left(\frac{1}{2}\right)^{n-1} u[n-1]. \quad (2.186)$$

## 2.10. SEÑALES ALEATORIAS EN TIEMPO DISCRETO

En las secciones anteriores nos hemos centrado en las representaciones matemáticas de las señales en tiempo discreto y en las interpretaciones que se pueden obtener de dichas representaciones matemáticas. Hemos visto que las señales y los sistemas en tiempo discreto se pueden representar tanto en el dominio del tiempo como en el dominio de la frecuencia, y cada una de esas representaciones tiene un papel importante en la teoría y en el diseño de los sistemas de tratamiento de señales en tiempo discreto. Hasta este momento hemos supuesto que las señales son deterministas, es decir, que cada valor de la secuencia está exactamente determinado por una expresión matemática, una tabla de datos o una regla de algún tipo.

En muchas situaciones, los procesos que generan las señales son tan complejos que realizar una descripción precisa de una señal se hace extremadamente difícil o no es deseable, e incluso puede ser imposible. En estos casos, resulta útil modelar analíticamente la señal como un proceso estocástico. Por ejemplo, en el Capítulo 6 veremos que muchos de los efectos que aparecen al realizar sistemas de tratamiento de señales utilizando registros de longitud finita se pueden representar como ruido aditivo, es decir, como una secuencia estocástica. Muchos sistemas mecánicos generan señales acústicas o vibratorias que se pueden procesar para diagnosticar fallos potenciales. Estas señales, a menudo, se procesan más adecuadamente considerándolas señales estocásticas. Las señales de voz que son procesadas para realizar reconocimiento automático o compresión de ancho de banda y las señales musicales que se procesan para mejora de calidad son dos ejemplos más de entre otros muchos.

Una señal estocástica se considera como miembro de un conjunto de señales en tiempo discreto caracterizada por un conjunto de funciones de densidad de probabilidad. Más concretamente, dada una señal particular en un instante específico, su amplitud en ese instante se supone determinada por un esquema de probabilidades subyacente. Es decir, cada muestra individual  $x[n]$  de una señal particular se supone que es una realización de una variable aleatoria subyacente  $x_n$ . La señal completa se representa mediante un conjunto de esas variables aleatorias, una para cada muestra temporal,  $-\infty < n < \infty$ . Este conjunto de variables aleatorias se denomina *proceso aleatorio*, y supondremos que cada secuencia particular de muestras,  $x[n]$  para  $-\infty < n < \infty$ , es generada por el proceso aleatorio subyacente. Para describir completamente el proceso aleatorio necesitamos especificar las distribuciones de probabilidad individuales y conjuntas de todas las variables aleatorias.

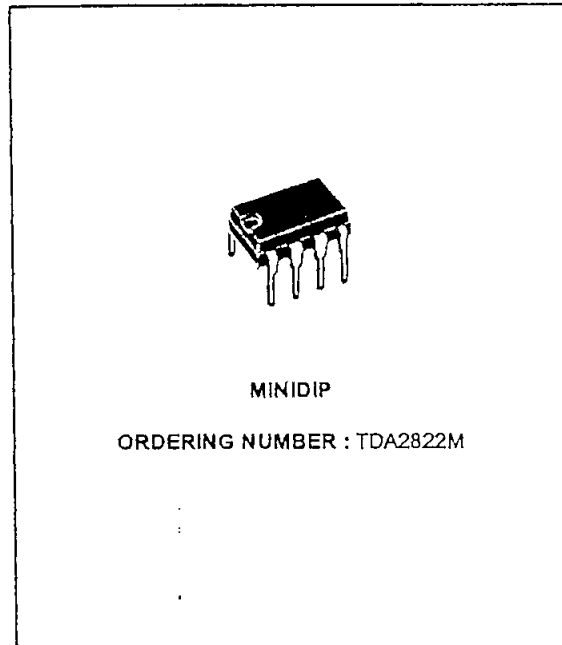
La clave para obtener resultados útiles mediante estos modelos de señales está en la descripción en términos de promedios que se puede calcular utilizando leyes de probabilidad asumidas o estimadas a partir de las señales concretas. Aunque las señales estocásticas no

## **Anexo 10**

Datasheet del circuito integrado TDA2822M

**DUAL LOW-VOLTAGE POWER AMPLIFIER**

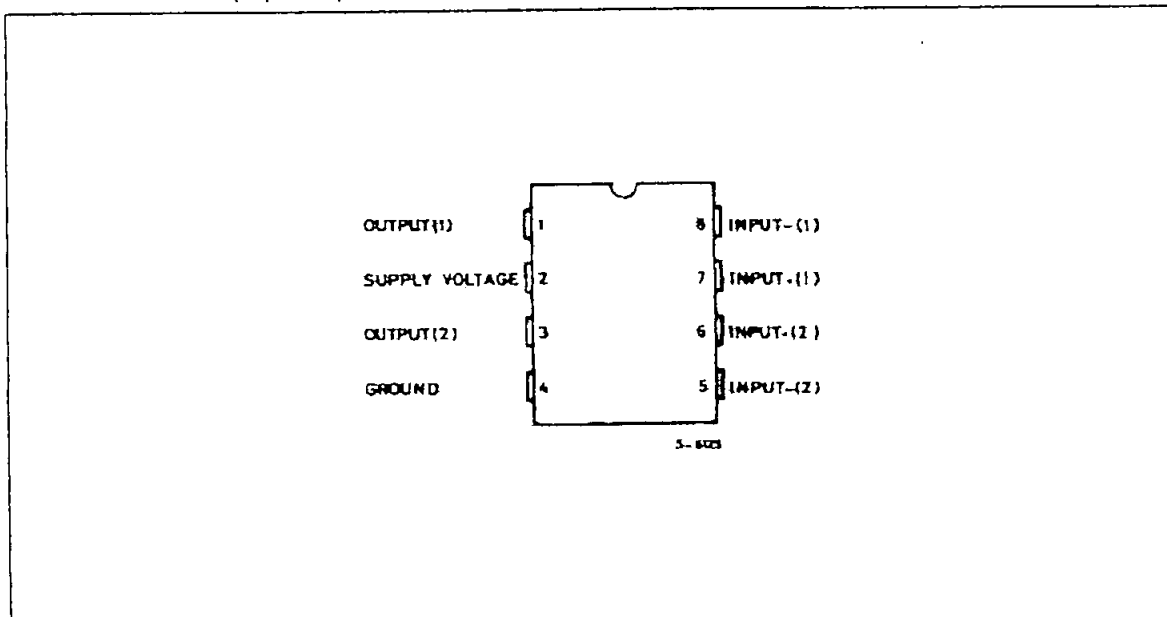
- SUPPLY VOLTAGE DOWN TO 1.8V
- LOW CROSSOVER DISTORSION
- LOW QUIESCENT CURRENT
- BRIDGE OR STEREO CONFIGURATION



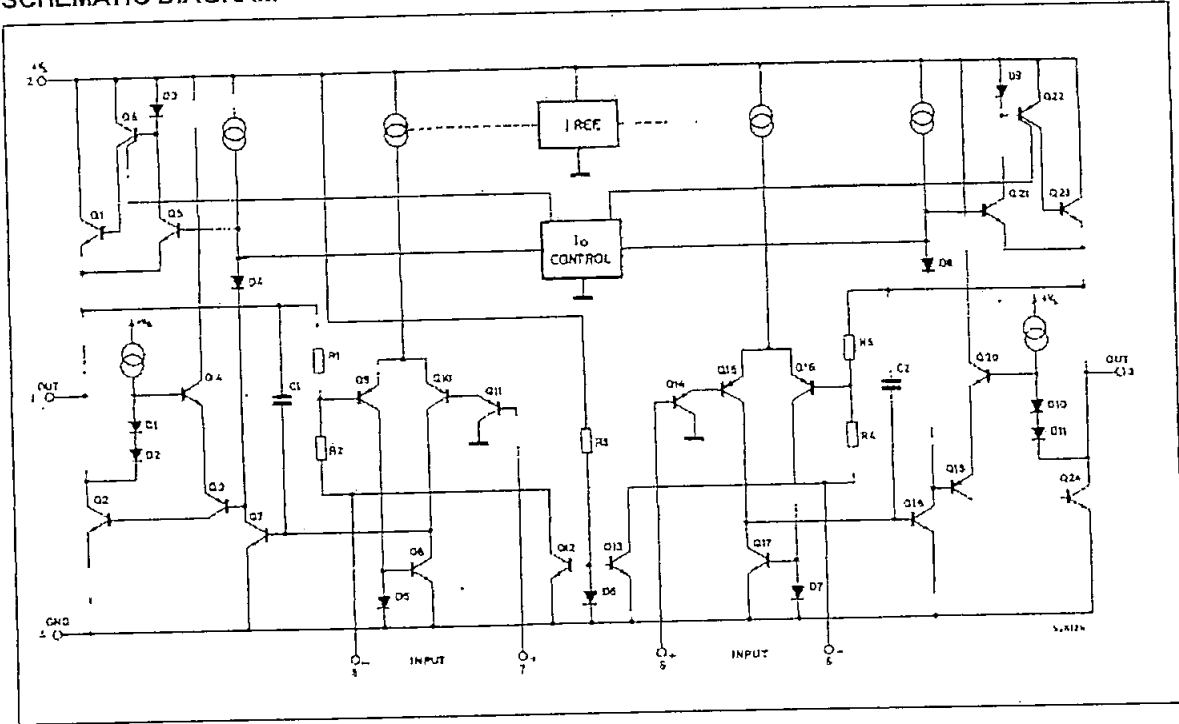
**DESCRIPTION**

The TDA2822M is a monolithic integrated circuit in 8 lead Minidip package. It is intended for use as dual audio power amplifier in portable cassette players and radios.

**PIN CONNECTION (Top view)**



SCHEMATIC DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
$V_s$	Supply Voltage	15	V
$I_o$	Peak Output Current	1	A
$P_{tot}$	Total Power Dissipation at $T_{amb} = 50\text{ }^\circ\text{C}$ at $T_{case} = 50\text{ }^\circ\text{C}$	1 1.4	W W
$T_{stg}, T_j$	Storage and Junction Temperature	-40, +150	$^\circ\text{C}$

THERMAL DATA

Symbol	Parameter	Value	Unit
$R_{th\ j-amb}$	Thermal Resistance Junction-ambient	Max. 100	$^\circ\text{C/W}$
$R_{th\ j-case}$	Thermal Resistance Junction-pin (4)	Max. 70	$^\circ\text{C/W}$

ELECTRICAL CHARACTERISTICS ( $V_S = 6V$ ,  $T_{amb} = 25^\circ C$ , unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
<b>STEREO</b> (test circuit of Figure 1)						
$V_S$	Supply Voltage		1.8		15	V
$V_o$	Quiescent Output Voltage	$V_S = 3V$		2.7 1.2		V V
$I_d$	Quiescent Drain Current			6	9	mA
$I_b$	Input Bias Current			100		nA
$P_o$	Output Power (each channel) ( $f = 1kHz$ , $d = 10\%$ )	$R_L = 32\Omega$ $V_S = 9V$ $V_S = 6V$ $V_S = 4.5V$ $V_S = 3V$ $V_S = 2V$ $R_L = 16\Omega$ $V_S = 6V$ $R_L = 8\Omega$ $V_S = 9V$ $V_S = 6V$ $R_L = 4\Omega$ $V_S = 6V$ $V_S = 4.5V$ $V_S = 3V$	90 15 170 300 450	300 120 60 20 5 220 1000 380 650 320 110		mW
$d$	Distortion ( $f = 1kHz$ )	$R_L = 32\Omega$ $P_o = 40mW$ $R_L = 16\Omega$ $P_o = 75mW$ $R_L = 8\Omega$ $P_o = 150mW$		0.2 0.2 0.2		% % %
$G_v$	Closed Loop Voltage Gain	$f = 1kHz$	36	39	41	dB
$\Delta G_v$	Channel Balance				$\pm 1$	dB
$R_i$	Input Resistance	$f = 1kHz$	100			k $\Omega$
$e_N$	Total Input Noise	$R_s = 10k\Omega$ B = Curve A B = 22Hz to 22kHz		2 2.5		$\mu V$ $\mu V$
SVR	Supply Voltage Rejection	$f = 100Hz$ , $C_1 = C_2 = 100\mu F$	24	30		dB
$C_s$	Channel Separation	$f = 1kHz$		50		dB

**BRIDGE** (test circuit of Figure 2)

$V_S$	Supply Voltage		1.8		15	V
$I_d$	Quiescent Drain Current	$R_L = \infty$		6	9	mA
$V_{os}$	Output Offset Voltage (between the outputs)	$R_L = 8\Omega$			$\pm 50$	mV
$I_b$	Input Bias Current			100		nA
$P_o$	Output Power ( $f = 1kHz$ , $d = 10\%$ )	$R_L = 32\Omega$ $V_S = 9V$ $V_S = 6V$ $V_S = 4.5V$ $V_S = 3V$ $V_S = 2V$ $R_L = 16\Omega$ $V_S = 9V$ $V_S = 6V$ $V_S = 3V$ $R_L = 8\Omega$ $V_S = 6V$ $V_S = 4.5V$ $V_S = 3V$ $R_L = 4\Omega$ $V_S = 4.5V$ $V_S = 3V$ $V_S = 2V$	320 50 900 200	1000 400 200 65 8 2000 800 120 1350 700 220 1000 350 80		mW
$d$	Distortion	$P_o = 0.5W$ , $R_L = 8\Omega$ , $f = 1kHz$		0.2		%
$G_v$	Closed Loop Voltage Gain	$f = 1kHz$		39		dB
$R_i$	Input Resistance	$f = 1kHz$	100			k $\Omega$
$e_N$	Total Input Noise	$R_s = 10k\Omega$ B = Curve A B = 22Hz to 22kHz		2.5 3		$\mu V$ $\mu V$
SVR	Supply Voltage Rejection	$f = 100Hz$		40		dB
B	Power Bandwidth (-3dB)	$R_L = 8\Omega$ , $P_o = 1W$		120		kHz



Figure 1 : Test Circuit (Stereo)

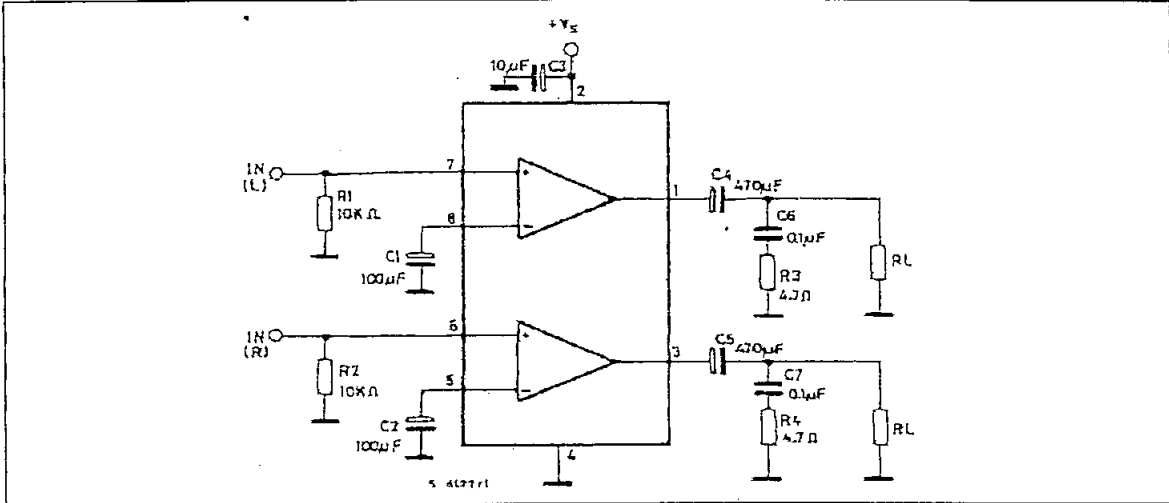


Figure 2 : Test Circuit (Bridge)

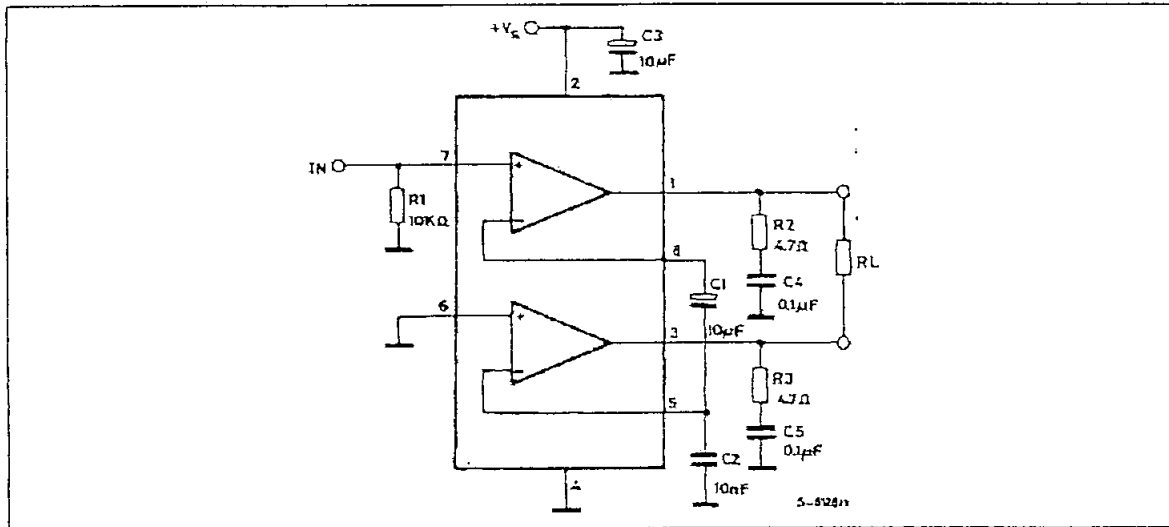


Figure 3 : P.C. Board and Components Layout of the Circuit of Figure 1

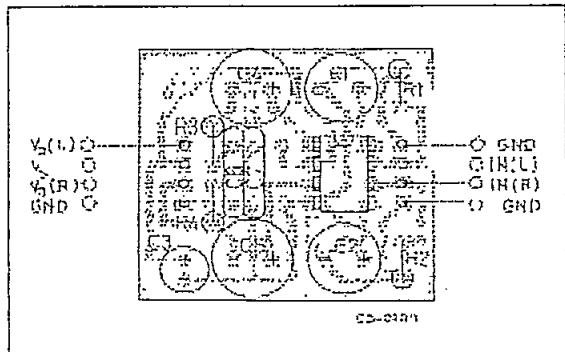


Figure 4 : P.C. Board and Components Layout of the Circuit of Figure 2

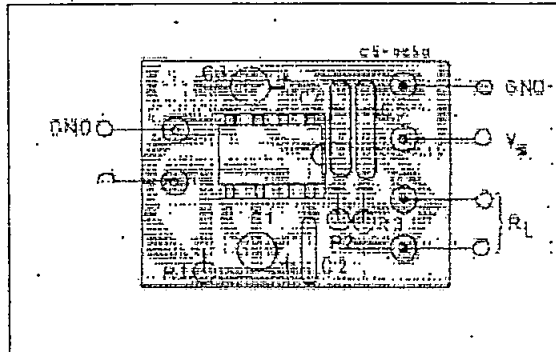


Figure 5 : Quiescent Current versus Supply Voltage

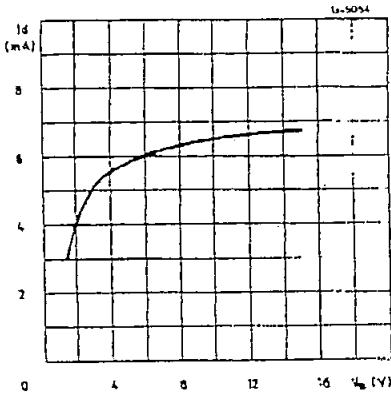


Figure 6 : Supply Voltage Rejection versus Frequency

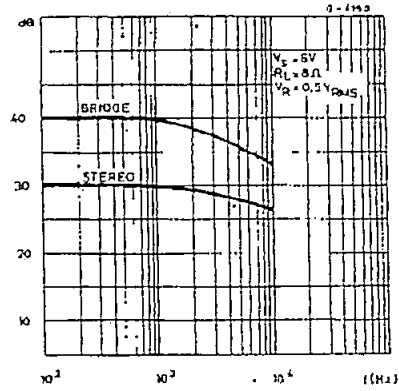


Figure 7 : Output Power versus Supply Voltage (THD = 10%, f = 1kHz Stereo)

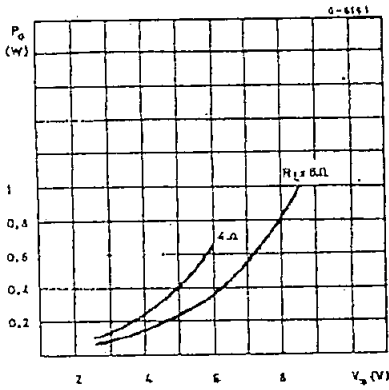


Figure 8 : Distorsion versus Output Power (Stereo)

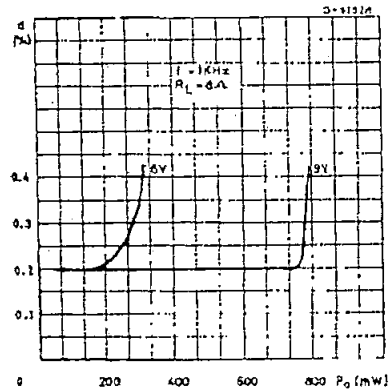


Figure 9 : Distorsion versus Output Power (Stereo)

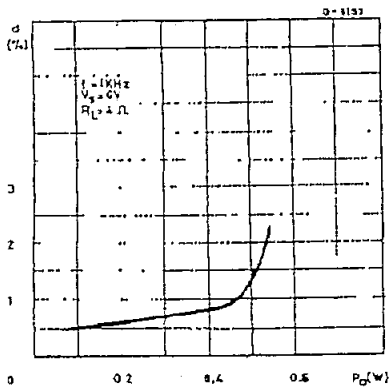


Figure 10 : Output Power versus Supply Voltage (Bridge)

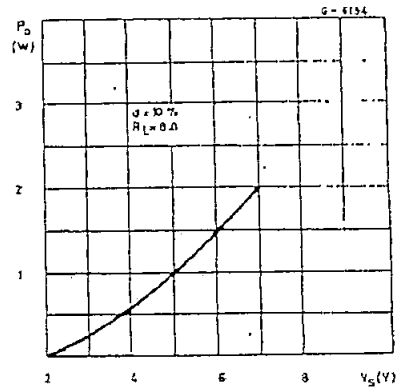


Figure 16 : Typical Application in Portable Players

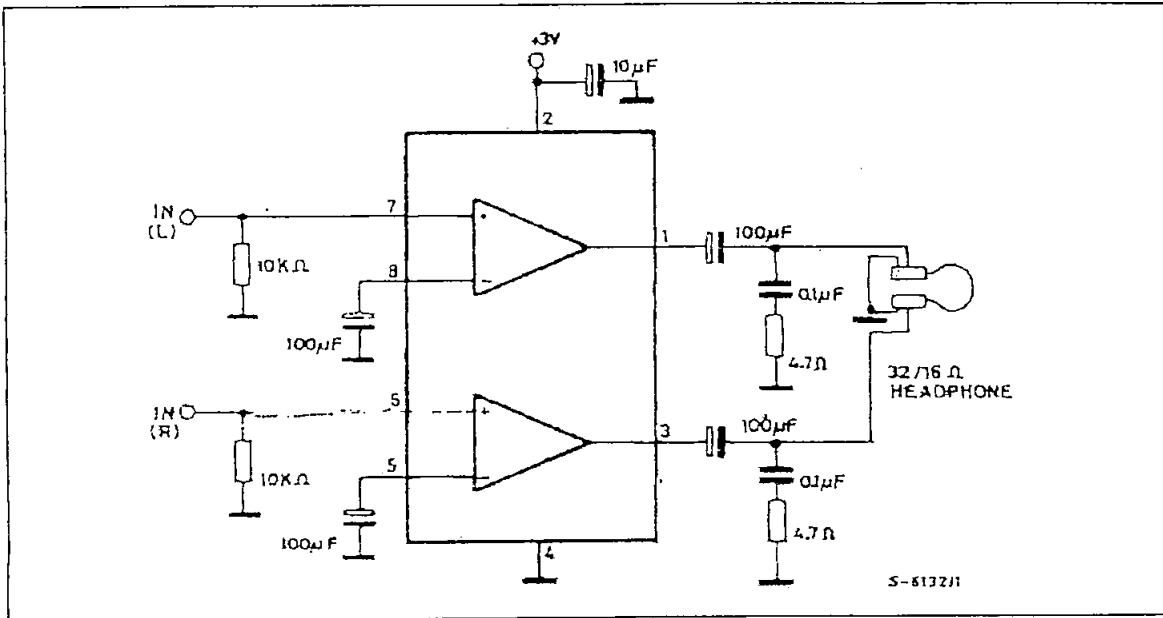
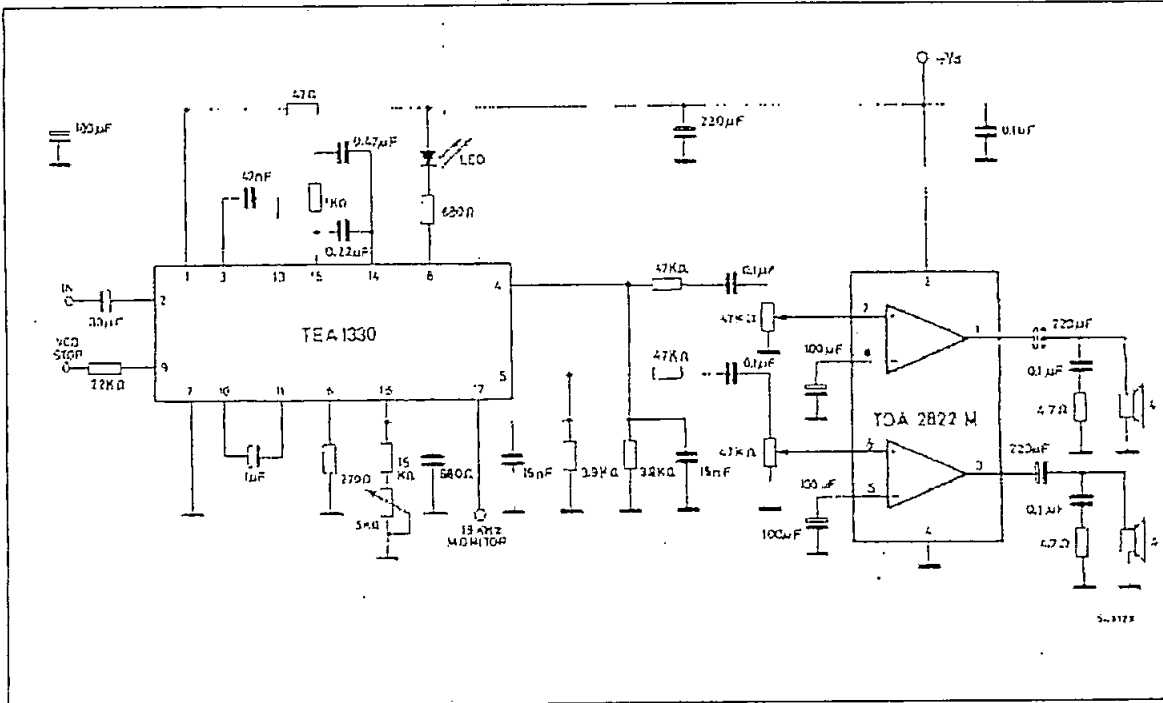


Figure 17 : Application in Portable Radio Receivers



# TDA2822M

Figure 18 : Portable Radio Cassette Players

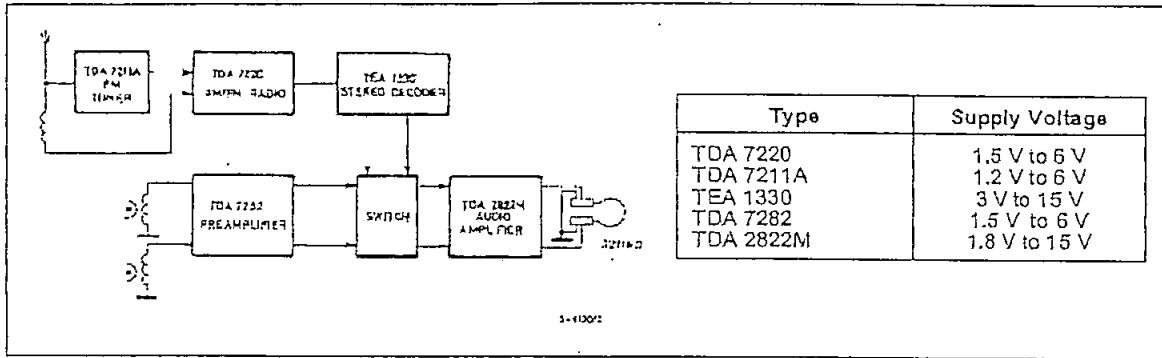


Figure 19 : Portable Stereo Radios

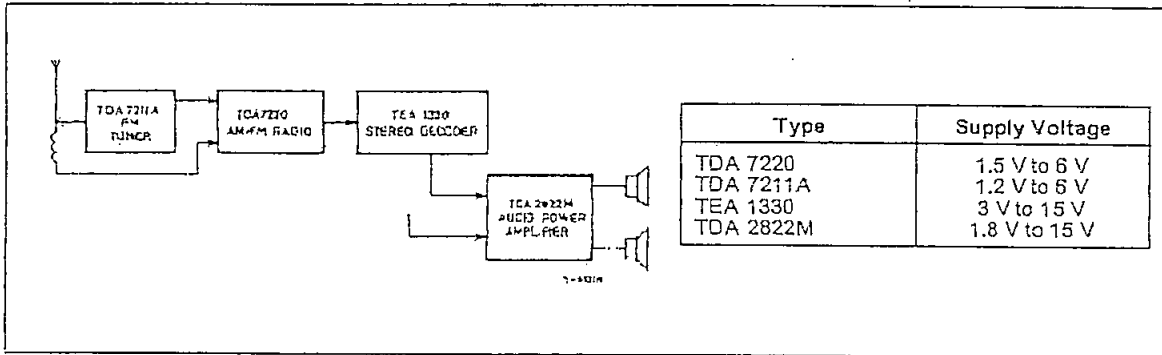
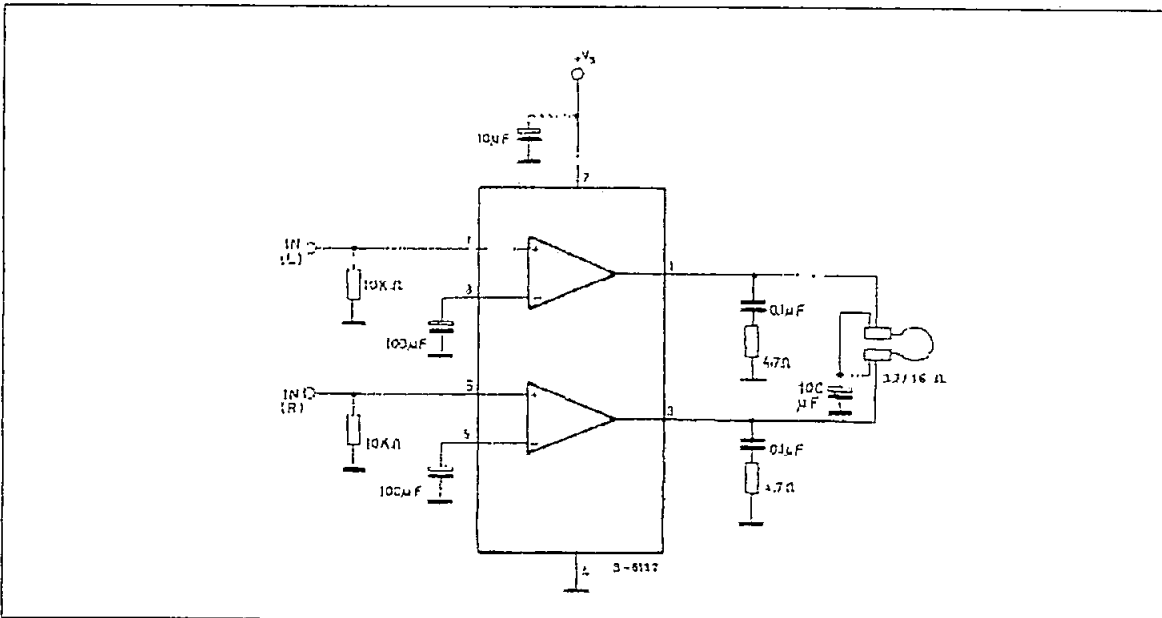


Figure 20 : Low Cost Application in Portable Players (using only one 100µF output capacitor)



# **Anexo 11**

## **Subrutina de Inicialización de comunicación serial entre CPU y DSP**

Tomado de : Curso DSP, Ing. Fernando Vásquez

## SUBROUTINA PARA INICIALIZACION DE COMUNICACIÓN SERIAL ENTRE CPU Y DSP

uart.dsp      Programa para implementar emulacion de terminal UART  
 Fernando Vasquez - DSP Soluciones - Enero 2001

Modificado para iniciar velocidad de tx. en programa que  
 utiliza subrutina UART escribiendo en dm(baud\_period)

Ej: ax0=1145;  
 dm(baud\_period)=ax0;    ----> Vtx = 9600 baudios

```

36666 -----> 300 baudios
18333 -----> 600 baudios
9166  -----> 1200 baudios
4583  -----> 2400 baudios
2291  -----> 4800 baudios
1145  -----> 9600 baudios
572   -----> 19200 baudios
286   -----> 38400 baudios
190   -----> 57600 baudios
95    -----> 115200 baudios
}
  
```

```

{*****
*****
  
```

ADSP-2101 Family Software UART

UART.DSP

This uses FLAG\_IN, FLAG\_OUT and the TIMER of ADSP-2101 to interface to an RS-232 asynchronous serial device such as a VT100 terminal. ex:

ADSP-2101 FLAG\_OUT -----> AD233 -----> RS-232 RX

ADSP-2101 FLAG\_IN <----- AD233 <----- RS-232 TX  
 (TIMER maintains baudrate)

Parameters bits/word, baudrate, stopbits & parity are user-programmable.

An RS-232 line driver chip (such as the AD233) can be used to electrically interface +5 VDC to the RS-232 line voltage levels.

The operation of the transmitter setup routine is completely independent on the the receiver setup routine operation. Although both tx and rx use the same timer as a master clock source, the xmitted bits need not be in sync with the received bits. The default state of the reciever is OFF, so the "turn\_rx\_on" subroutine must be used to enable RX.

Calling Argument:

for autobaud load the baud constant  
dm(baud\_period)=(Proc\_frequency/(3\*Baudrate))-1

Useful Subroutines:

- init\_uart Must be called after system reset.
- get\_char\_ax1 Waits for RX input and returns with it in ax1.
- out\_char\_ax1 Waits for last TX output and transmits data from ax1.
- turn\_rx\_on Must be called to enable the receipt of RX data.
- turn\_rx\_off Can be used to ignore input RX data.

Useful Flag:

DM(flag\_rx\_ready) If this DM location is all ones it indicates that the UART is ready to rx new word. If it is zero then data is being received. Can be used for xon xoff flow control.

Author: Fares Eidi, 21-May-90, Analog Devices Inc.  
 modified: Christoph D. Cavigioli, 17-Dec-90  
 modified: Steven Cox, 20-Dec-91, extensive rewrite  
 modified: Steven Cox, 31-Dec-91, Reset flag\_rx\_stop\_yet to 1, Comments.  
 modified: Steven Cox, 11-Feb-92, Added support for autobaud.  
 modified: Philip Holdgate, 02-Apr-92, modified for Apps Note, cleaned up.

\*\*\*\*\*  
\*\*\*\*\*}

{\_\_\_\_\_The Constants below must be changed to modify uart parameters \_\_\_\_\_}

```
.const tx_num_of_bits = 10; { start bits + tx data bits + stop bits }
.const rx_num_of_bits = 8; { rx data bits (start&stop bits not counted)}
.const RX_BIT_ADD = 0x0100; { = 1<<rx_num_of_bits }
.const TX_BIT_ADD = 0xfe00; { = 0xffff<<(tx data bits+1) }
```

{...These constants can be used if autobaud is not needed....}

```
.const PERIOD = 74; {13&57600}{ PERIOD = (Proc_frequency/(3*Baudrate))-1}
{.const PERIOD = 112;} {13&38400}{ PERIOD = (Proc_frequency/(3*Baudrate))-1}
{.const PERIOD = 225;} {13&19200}{ PERIOD = (Proc_frequency/(3*Baudrate))-1}
{.const PERIOD = 450;} {13& 9600}{ PERIOD = (Proc_frequency/(3*Baudrate))-1}
```

{\_\_\_\_\_Definitions of memory mapped control registers \_\_\_\_\_}

```
.const TSCALE= 0x3ffb;
.const TCOUNT= 0x3ffc;
.const TPERIOD= 0x3ffd;
.const System_Control_Reg= 0x3fff;
```

```
{  
_____ }  
_____ }
```

```
.entry init_uart;      { UART initialize baudrate etc. }  
.entry out_char_ax1;   { UART output a character }  
.entry get_char_ax1;   { UART wait & get input character }  
.entry turn_rx_on;     { UART enable the rx section }  
.entry turn_rx_off;    { UART disable the rx section }  
.entry process_a_bit;  { UART timer interrupt routine for RX and TX }
```

```
.global flag_rx_ready;  
.global baud_period;
```

```
.var flag_tx_ready;    { flag indicating UART is ready for new tx word }  
.var flag_rx_ready;    { flag indicating UART is ready to rx new word }  
.var flag_rx_stop_yet; { flag tells that a rx stop bit is not pending }  
.var flag_rx_no_word;  { indicates a word is not in the user_rx_buffer }  
.var flag_rx_off;      { indicates a that the receiver is turned off }  
.var timer_tx_ctr;     { divide by 3 ctr, timer is running @ 3x baudrate }  
.var timer_rx_ctr;     { divide by 3 ctr, timer is running @ 3x baudrate }  
.var user_tx_buffer;   { UART tx reg loaded by user before UART xmit }  
.var user_rx_buffer;   { UART rx reg read by user after word is rcvd }  
.var internal_tx_buffer; { formatted for serial word, adds start&stop bits }  
                        { 'user_tx_buffer' is copied here before xmission }  
.var internal_rx_buffer;  
.var bits_left_in_tx;  { number of bits left in tx buffer (not yet clkd out) }  
.var bits_left_in_rx;  { number of bits left to be rcvd (not yet clkd in) }  
.var baud_period;     { loaded by autobaud routine }
```

```
{ _____ Initializing subroutine _____ }
```

```
init_uart:  
  ax0=0;  
  dm(TSCALE)=ax0;      { decrement TCOUNT every instruction cycle }  
  
  ax0=dm(baud_period); { from autobaud or use constant: ax0=PERIOD; }  
                      { ...and comment in the appropriate constant }  
  
  dm(TCOUNT)=ax0;  
  dm(TPERIOD)=ax0;     { interrupts generated at 3x baudrate }  
  
  ax0=dm(System_Control_Reg);  
  ay0=0xf3ff;  
  ar=ax0 and ay0;  
  dm(System_Control_Reg)=ar; { no bmwait, pmwait states, SPORT1 = FI/FO }  
  
  ax0=1;  
  dm(flag_tx_ready)=ax0; { set the flags showing that UART is not busy }  
  dm(flag_rx_ready)=ax0;  
  dm(flag_rx_stop_yet)=ax0;
```



```
dm(flag_rx_no_word)=ax0;
dm(flag_rx_off)=ax0;    { rx section off }
```

```
set flag_out;           { UART tx output is initialized to high }
ifc=0x00ff;            { clear all pending interrupts }
nop;                    { wait for ifc latency }
```

```
ax0=imask;
ay0=b#0000000001;
ar=ax0 or ay0;
imask=ar;               { enable TIMER interrupt handling }
```

```
ena timer;             { start timer now }
rts;
```

---

```
{ _____ Process_a_bit (TIMER interrupt routine) _____
```

This routine is the heart of the UART. It is called every timer interrupt (i.e. 3x baudrate). This routine will xmit one bit at a time by setting/clearing the FLAG\_OUT pin of the ADSP-2101. This routine will then test if the uart is already receiving. If not it will test flagin (rx) for a start bit and place the uart in receive mode if true. If already in receive mode it will shift in one bit at a time by reading the FLAG\_IN pin. Since the internal timer is running at 3x baudrate, bits need only be transmitted/received once every 3 timer interrupts.

---

```
_____}
process_a_bit:
ena sec_reg;           { Switch to the background dreg set }
ax0=dm(flag_tx_ready); { if not in "transmit", go right to "receive" }
ar=pass ax0;
if ne jump receiver;
    { _____ Transmitter Section _____ }
ay0=dm(timer_tx_ctr); { test timer ctr to see if a bit }
ar=ay0-1;              { is to be sent this time around }
dm(timer_tx_ctr)=ar;   { if no bit is to be sent }
if ne jump receiver;   { then decrement ctr and return }

sr1=dm(internal_tx_buffer); { shift out LSB of internal_tx_buffer }
sr=lshift sr1 by -1 (hi);   { into SR1. Test the sign of this bit }
dm(internal_tx_buffer)=sr1; { set or reset FLAG_OUT accordingly }
ar=pass sr0;              { this effectively clocks out the }
if ge reset flag_out;     { word being xmitted one bit at a time }
if lt set flag_out;      { LSB out first at FLAG_OUT. }

ay0=3;                  { reset timer ctr to 3, i.e. next bit }
dm(timer_tx_ctr)=ay0;    { will be sent after 3 timer interrupts }

ay0=dm(bits_left_in_tx); { number of bits left to be xmitted }
ar=ay0-1;               { is now decremented by one, }
```

```
dm(bits_left_in_tx)=ar;      { indicating that one is now xmitted }
if gt jump receiver;        { if no more bits left, then ready }
```

```
ax0=1;                      { flag is set to true indicating }
dm(flag_tx_ready)=ax0;      { a new word can now be xmitted }
```

```
{ _____Receiver Section_____ }
```

```
receiver:
```

```
ax0=dm(flag_rx_off);        { Test if receiver is turned on }
ar=pass ax0;
if ne rti;
```

```
ax0=dm(flag_rx_stop_yet);   { Test if finished with stop bit of }
ar=pass ax0;                { last word or not. if finished then }
if ne jump rx_test_busy;    { continue with check for receive. }
```

```
ay0=dm(timer_rx_ctr);       { decrement timer ctr and test to see }
ar=ay0-1;                   { if stop bit period has been reached }
dm(timer_rx_ctr)=ar;        { if not return and wait }
if ne rti;
```

```
ax0=1;                      { if stop bit is reached then reset }
dm(flag_rx_stop_yet)=ax0;   { to wait for next word }
dm(flag_rx_ready)=ax0;
```

```
ax0=dm(internal_rx_buffer); { copy internal rx buffer }
dm(user_rx_buffer)=ax0;     { to the user_rx_buffer }
```

```
ax0=0;                      { indicated that a word is ready in }
dm(flag_rx_no_word)=ax0;    { the user_rx_buffer }
rti;
```

```
rx_test_busy:
```

```
ax0=dm(flag_rx_ready);      { test rx flag, if rcvr is not busy }
ar=pass ax0;                 { receiving bits then test for start.If it }
if eq jump rx_busy;          { is busy, then clk in one bit at a time }
```

```
if flag_in jump rx_exit;     { Test for start bit and return if none }
```

```
ax0=0;
dm(flag_rx_ready)=ax0;       { otherwise, indicate rcvr is now busy }
dm(internal_rx_buffer)=ax0;  { clear out rcv register }
```

```
ax0=4;                      { timer runs @ 3x baud rate, so rcvr }
dm(timer_rx_ctr)=ax0;        { will only rcv on every 3rd interrupt }
                              { initially this ctr is set to 4. This }
                              { will skip the start bit and will }
                              { allow us to check FLAG_IN at the center }
                              { of the received data bit }
```

```
ax0=rx_num_of_bits;
```

```

dm(bits_left_in_rx)=ax0;
rx_exit:
    rti;

rx_busy:
    ay0=dm(timer_rx_ctr);        { decrement timer ctr and test to see }
    ar=ay0-1;                    { if bit is to be rcvd this time around }
    dm(timer_rx_ctr)=ar;         { if not return, else receive a bit }
    if ne rti;

rcv:    { Shift in rx bit }
    ax0=3;                          { reset the timer ctr to 3 indicating }
    dm(timer_rx_ctr)=ax0;           { next bit is 3 timer interrupts later }

    ay0=RX_BIT_ADD;
    ar=dm(internal_rx_buffer);
    if not flag_in_jump pad_zero; { Test RX input bit and }
    ar=ar+ay0;                      { add in a 1 if hi }

pad_zero:
    sr=lshift ar by -1 (lo);        { Shift down to ready for next bit }
    dm(internal_rx_buffer)=sr0;

    ay0=dm(bits_left_in_rx);        { if there are more bits left to be rcvd }
    ar=ay0-1;                       { then keep UART in rcv mode }
    dm(bits_left_in_rx)=ar;         { and return }
    if gt rti;                      { if there are no more bits then.. }

    { That was the last bit }
    ax0=3;                          { set timer to wait for middle of the }
    dm(timer_rx_ctr)=ax0;           { stop bit }
    ax0=0;                          { flag indicated that uart is waiting }
    dm(flag_rx_stop_yet)=ax0;      { for the stop bit to arrive }
    rti;

```

---

{ \_\_\_\_\_ invoke\_UART\_transmit subroutine \_\_\_\_\_ }

This is the first step in the transmit process. The user has now loaded 'user\_tx\_buffer' with the ascii code and has also invoked this routine.

---

```

_____}
invoke_UART_transmit:
    ax0=3;                          { initialize the timer decimator ctr }
    dm(timer_tx_ctr)=ax0;           { this divide by three ctr is needed }
    { since timer runs @ 3x baud rate }

    ax0=tx_num_of_bits;            { this constant is defined by the }
    dm(bits_left_in_tx)=ax0;        { user and represents total number of }
    { bits including stop and parity }

```

```
        { ctr is initialized here indicating }
        { none of the bits have been xmitted }
```

```
sr1=0;
sr0=TX_BIT_ADD;      { upper bits are hi to end txmit with hi }
ar=dm(user_tx_buffer); { transmit register is copied into }
sr=sr or lshift ar by 1 (lo); { the internal tx reg & left justified }
dm(internal_tx_buffer)=sr0; { before it gets xmitted }
```

```
ax0=0;      { indicate that the UART is busy }
dm(flag_tx_ready)=ax0;
rts;
```

```
{ _____ get an input character _____
```

```
output:   ax1
modifies: ax0
```

---

```
_____}
get_char_ax1:
```

```
ax0=dm(flag_rx_no_word);
ar=pass ax0;
if ne jump get_char_ax1; { if no rx word input, then wait }

ax1=dm(user_rx_buffer); { get received ascii character }
ax0=1;
dm(flag_rx_no_word)=ax0; { word was read }
rts;
```

```
{ _____ output a
character _____
```

```
input:   ax1
modifies: ax0, sr1, sr0, ar
```

---

```
_____}
out_char_ax1:
```

```
ax0=dm(flag_tx_ready);
ar=pass ax0;
if eq jump out_char_ax1; { if tx word out still pending, then wait }
dm(user_tx_buffer)=ax1;
call invoke_UART_transmit; { send it out }
rts;
```

```
{ _____ enable the RX
section _____
```

```
modifies: ax0
```

---

```
_____}
turn_rx_on:
```

```
ax0=0;
dm(flag_rx_off)=ax0;
```

rts;

{ \_\_\_\_\_ disable the RX

section \_\_\_\_\_

modifies: ax0

---

\_\_\_\_\_ }

turn\_rx\_off:

ax0=1;

dm(flag\_rx\_off)=ax0;

rts;

{ ----- Obtiene caracter con time out ----- }

get\_char\_ax1\_to:

ax1 = ar; { preserve ar }

ar = 30000; { 3 ticks per bit; 9600 bps, time out }

{ at 1 sec ==> 9600 \* 3 == 30000 }

get\_char\_ax1\_to1:

idle; { wait for timer interrupt }

ar = ar - 1;

if eq jump get\_char\_ax1\_to2; { time out now }

ax0=dm(flag\_rx\_no\_word);

none=pass ax0;

if ne jump get\_char\_ax1\_to1; { if no rx word input, then wait }

ar = ax1; { restore ar }

ax1=dm(user\_rx\_buffer); { get received ascii character }

ax0=1;

dm(flag\_rx\_no\_word)=ax0; { word was read }

rts;

get\_char\_ax1\_to2:

ar = ax1; { restore ar }

ax1 = 0xffff;

rts;

## **Anexo 12**

Programa implementado en el ADSP-2181

## PROGRAMA IMPLEMENTADO EN EL ADSP-281

```
.MODULE/RAM/ABS=0 AMCAEC I;

.CONST fs=0xc85c;      { Fs=16kHz }
.CONST Ns=4;          { Numero de secciones de segundo orden }
.CONST N_coefs=5*Ns;  { Numero de coeficientes del filtro }
.CONST N_delay=2*(Ns+1); { Longitud de la linea de retardo }

.VAR/DM/CIRC tbl_freq[7];
.INIT tbl_freq: 171, 341, 683, 1365, 2731, 5461, 10923;

.VAR/DM fin_flag, freq_actual, c_cntr, derecho, izquierdo, seno, var_R, var_L,
atenuacionR, atenuacionL;

.VAR/DM sin_coeff[5];
.INIT sin_coeff: H#3240, H#0053, H#AACC, H#08B7, H#1CCE;

.VAR/DM/CIRC coefs[N_coefs]; { B2,B1,B0,A2,A1 coeficientes del filtro }
.VAR/DM/CIRC delay_data[N_delay]; { Buffer para almacenar la linea de retardo }

.INIT coefs:
H#16a,H#02d4,H#016a,H#df94,H#5ae3,H#0173,H#02e7,H#0173,H#dbd1,H#5e6f,H#0
189,H#0312,H#0189,H#d408,H#65c6,H#01b0,H#0360,H#01b0,H#c7d4,H#7146;

{ ----- Vector de interrupciones ----- }
jump inicio; rti; rti; rti;      {00: reset }
rti; rti; rti; rti;             {04: IRQ2 }
rti; rti; rti; rti;             {08: IRQL1 }
rti; rti; rti; rti;             {0c: IRQL0 }
ar = dm(stat_flag);             {10: SPORT0 tx }
ar = pass ar; if eq rti; jump next_cmd;
jump rx_muestras; rti; rti; rti; {14: SPORT0 rx }
jump irqe; rti; rti; rti;        {18: IRQE }
rti; rti; rti; rti;             {1c: BDMA }
rti; rti; rti; rti;             {20: SPORT1 tx or IRQ1 }
rti; rti; rti; rti;             {24: SPORT1 rx or IRQ0 }
jump process_a_bit; rti; rti; rti; {28: timer }
rti; rti; rti; rti;             {2c: power down }

inicio:

ax0=fs;
call codecInit;                 { Inicializacion del CODEC }
imask=0;

ar=1145;                         { Vel. de tx=9600 }
dm(baud_period)=AR;
```

```

baje_der:
    ar=dm(atenuacionR);
    ar=ar+1;
    dm(atenuacionR)=ar;
    ax0=0xff;
    dm(var_R)=ax0;
    jump espera_char;

baje_izq:
    ar=dm(atenuacionL);
    ar=ar+1;    {Atenuacion en pasos de 4*1.5 = 6dB ; Cambio minimo de 1.5dB si se
incrementa en uno}
    dm(atenuacionL)=ar;
    ax0=0xff;
    dm(var_L)=ax0;
    jump espera_char;

suba_der:
    ar=dm(atenuacionR);
    ar=ar-1;    {Atenuacion en pasos de 4*1.5 = 6dB}
    dm(atenuacionR)=ar;
    ax0=0xff;
    dm(var_R)=ax0;
    jump espera_char;

suba_izq:
    ar=dm(atenuacionL);
    ar=ar-1;    {Atenuacion en pasos de 4*1.5 = 6dB ; Cambio minimo de 1.5dB si se
incrementa en uno}
    dm(atenuacionL)=ar;
    ax0=0xff;
    dm(var_L)=ax0;
    jump espera_char;

fin_ton:                                {Pongo en cero a frecuencia actual}
    ax0=0;
    dm(frec_actual)=ax0;
    dm(c_cntr)=ax0;
    reset fl1;

    ax0=dm(derecho);
    ay0=0xff;
    none=ay0-ax0;
    if eq jump res_der;

    ax0=dm(izquierdo);
    ay0=0xff;
    none=ay0-ax0;
    if eq jump res_izq;

```



```
call init_uart;
call turn_rx_on;
```

```
{ Inicializacion del UART }
{ Habilita recepcion del UART }
```

```
{ ----- Insertar aqui inicializacion de variables ----- }
```

```
ax0=0;
dm(fin_flag)=ax0;
dm(c_cntr)=ax0;
dm(frec_actual)=ax0;
dm(var_R)=ax0;
dm(var_L)=ax0;
dm(derecho)=ax0;
dm(izquierdo)=ax0;
dm(seno)=ax0;
ax0=62;
dm(atenuacionR)=ax0;
dm(atenuacionL)=ax0;
```

```
m2=1;
l2=%tbl_frec;
i2=^tbl_frec;
```

```
ax0=0;
l0=%delay_data;
m0=1;
m4=1;
m1=-3;
m3=1-N_delay;
i0=^delay_data;
l4=%coefs;
i4=^coefs;
cntr=N_delay;
do cero until ce;
    cero: dm(i2,m0)=ax0;
```

```
ifc=0xff;
nop;
imask=0x31;
```

```
espera_char:
```

```
call get_char_ax1;    { Espera recibir caracter del UART }
```

```
ay1=0x044;           {Espero letra D, iniciar evaluacion oido derecho}
ar=ay1-ax1;
if eq jump der;
```

```
ay1=0x049;           {Espero letra I, iniciar evaluacion oido izquierdo}
ar=ay1-ax1;
if eq jump izq;
```

```

ay1=0x043;           {Espero letra C, cambio de tono}
ar=ay1-ax1;
if eq jump cam_ton;

ay0=0x32;           {Espero caracter 2, bajo volumen derecho}
none=ax1-ay0;
if eq jump baje_der;

ay0=0x33;           {Espero caracter 3, bajo volumen izquierdo}
none=ax1-ay0;
if eq jump baje_izq;

ay0=0x34;           {Espero caracter 4, subo volumen derecho}
none=ax1-ay0;
if eq jump suba_der;

ay0=0x35;           {Espero caracter 5, subo volumen izquierdo}
none=ax1-ay0;
if eq jump suba_izq;

ay1=0x046;           {Espero letra F, finalizo tono}
ar=ay1-ax1;
if eq jump fin_ton;

jump espera_char;

der:                 {Pongo en uno la bandera derecho y cero la izquierdo}
ax0=0xff;
ay0=0x00;
dm(derecho)=ax0;
dm(izquierdo)=ay0;
jump espera_char;

izq:                 {Pongo en uno la bandera izquierdo y cero la derecho}
ax0=0xff;
ay0=0x00;
dm(derecho)=ay0;
dm(izquierdo)=ax0;
jump espera_char;

cam_ton:             {Actualizo frecuencia actual}
ax0=dm(i2,m2);
dm(frec_actual)=ax0;
ax0=0;
dm(c_cntr)=ax0;
jump espera_char;

```

```
res_der:
    ax0=62;
    dm(atenuacionR)=ax0;
    ax0=0xff;
    dm(var_R)=ax0;
    jump espera_char;
```

```
res_izq:
    ax0=62;
    dm(atenuacionL)=ax0;
    ax0=0xff;
    dm(var_L)=ax0;
    jump espera_char;
```

```
irqe:
    toggle fl1;
    rti;
```

```
{ ----- Subrutina de atención de rx. de muestras ----- }
```

```
rx_muestras:
```

```
    ena sec_reg;
    call volumen;
    ay0=dm(frec_actual);
    ax0=dm(c_cntr);
    ar=ax0+ay0;
    dm(c_cntr)=ar;
    ax0=ar;
    call sin;
    srl=ar;
    call filtro;
    ar=srl;
    dm(seno)=ar;
    call oido;
    dm(tx_buf+1)=srl;
    dm(tx_buf+2)=srl;
    rti;
```

```
{ *****subrutinas implementadas para AMCAEC I ***** }
```

```
{ ***** Subrutina de control de volumen ***** }
```

```
volumen:
```

```
    ax0=dm(var_R);
    ay0=0xff;
    none=ay0-ax0;
    if eq jump att_der;
```

```
ax0=dm(var_L);
ay0=0xff;
none=ay0-ax0;
if eq jump att_izq;
```

```
jump salga;
```

```
att_der:
```

```
ax0=dm(atenuacionR);
ay0=0x8600;
ar=ax0 or ay0;
dm(tx_buf)=ar;
ax1=0;
dm(var_R)=ax1;
jump salga;
```

```
att_izq:
```

```
ax0=dm(atenuacionL);
ay0=0x8700;
ar=ax0 or ay0;
dm(tx_buf)=ar;
ax1=0;
dm(var_L)=ax1;
jump salga;
```

```
salga:
```

```
rts;
```

```
{***** Fin subrutina para control de volumen *****}
```

```
{***** Subrutina para calculo del seno *****}
```

```
sin: I3=^sin_coeff;           {Pointer to coeff. buffer}
AY0=H#4000;
AR=AX0, AF=AX0 AND AY0;      {Check 2nd or 4th quad.}
IF NE AR=-AX0;              {If yes, negate input}
AY0=H#7FFF;
AR=AR AND AY0;              {Remove sign bit}
MY1=AR;
MF=AR*MY1 (RND), MX1=DM(I3,M3); {MF = x**2}
MR=MX1*MY1 (SS), MX1=DM(I3,M3); {MR = C1*x}
CNTR=3;
DO approx UNTIL CE;
    MR=MR+MX1*MF (SS);
approx: MF=AR*MF (RND), MX1=DM(I3,M3);
    MR=MR+MX1*MF (SS);
```

```

SR=ASHIFT MR1 BY 3 (HI);
SR=SR OR LSHIFT MR0 BY 3 (LO);      {Convert to 1.15 format}
AR=PASS SR1;
IF LT AR=PASS AY0;                  {Saturate if needed}
AF=PASS AX0;
IF LT AR=-AR;                        {Negate output if needed}
RTS;

```

```
{***** Fin subrutina para calculo del seno *****}
```

```
{***** Subrutina para filtrado de la señal *****}
```

filtro:

```

se=1;      { Para este filtro todos los coeficientes de las secciones en 2.14 }
cntr=Ns;   { se carga cntr con el numero de secciones del filtro }

```

do secciones until ce;

```

mx0=dm(i0,m0), my0=pm(i4,m4);
mr=mx0*my0(ss), mx1=dm(i0,m0), my0=pm(i4,m4);
mr=mr+mx1*my0(ss), my0=pm(i4,m4);
mr=mr+sr1*my0(ss), mx0=dm(i0,m0), my0=pm(i4,m4);
mr=mr+mx0*my0(ss), mx0=dm(i0,m1), my0=pm(i4,m4);
dm(i0,m0)=mx1, mr=mr+mx0*my0(rnd);

```

secciones:

```

dm(i0,m0)=sr1, sr=ashift mr1(hi);
dm(i0,m0)=mx0;
dm(i0,m3)=sr1;
rts;

```

```
{***** Fin Subrutina para filtrado de la señal *****}
```

```
{***** Subrutina para selección de canal *****}
```

oido:

```

ax0=dm(derecho);
ay0=0xff;
ar=ay0-ax0;
if eq jump dere;

```

```

ax0=dm(izquierdo);
ay0=0xff;
ar=ay0-ax0;
if eq jump izqu;

```

```
jump fin;
```

dere:

sr0=dm(seno);

sr1=0;

jump fin;

izqu:

sr0=0;

sr1=dm(seno);

jump fin;

fin:

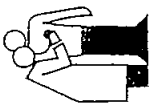
rts;

{\*\*\*\*\* Fin de subrutina para selección de canal \*\*\*\*\*}

.ENDMOD;

## **Anexo 13**

Evaluaciones audiométricas realizadas con equipo médico  
comercial



LA FUNDACION

Vista Para Todos

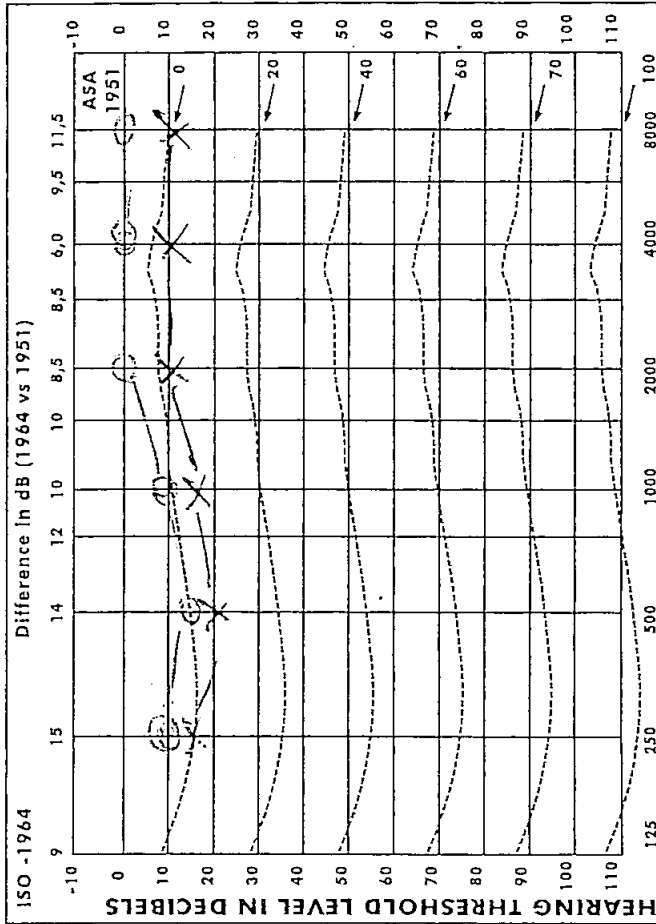
NAME:

*Socor Valdez*

AUDIOGRAM

DATE:

BY



AIR	LEFT EAR		RIGHT EAR
	BLUE	RED	RED
BOHE COND. W. MASKING DB	X		O
EFFECTIVE MASKING	I		I

A. M. A. % HEARING LOSS		
L.	R.	COMB.

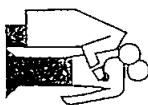
LEFT EAR		RIGHT EAR
S. R. T.		
M. C. L.		
T. D.		

This audiogram is plotted on the basis of:

- 1964 ISO reference thresholds
  - 1951 ASA reference thresholds
- (Check one of these squares)

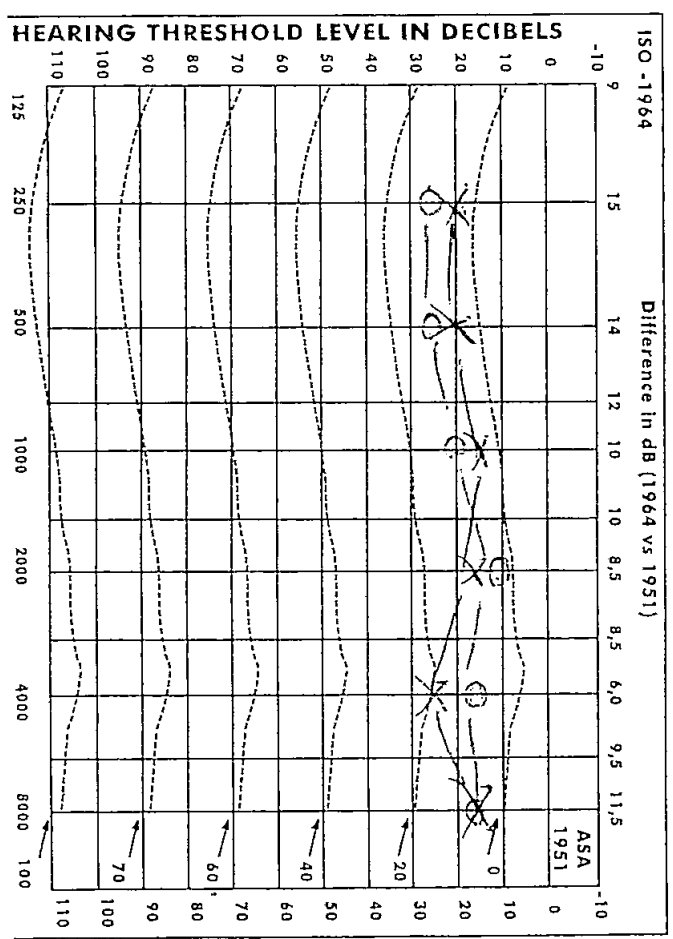
Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Threshold Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".





LA FUNDACION  
*Vista Para Todos*  
**AUDIOGRAM**

NAME: Rosa Maria Prieto  
 DATE: \_\_\_\_\_  
 BY: \_\_\_\_\_



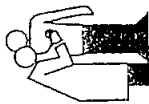
	LEFT EAR BLUE	RIGHT EAR RED
AIR	X	O
BONE COND. W. MASKING	J	I
D B EFFECTIVE MASKING		

A. M. A. % HEARING LOSS	
L.	R.
	COMB.

S. R. T.	LEFT EAR	RIGHT EAR
M. C. L.		
T. D.		

This audiogram is plotted on the basis of:  
 1964 ISO reference thresholds  
 1951 ASA reference thresholds  
 (Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Thresholds Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".

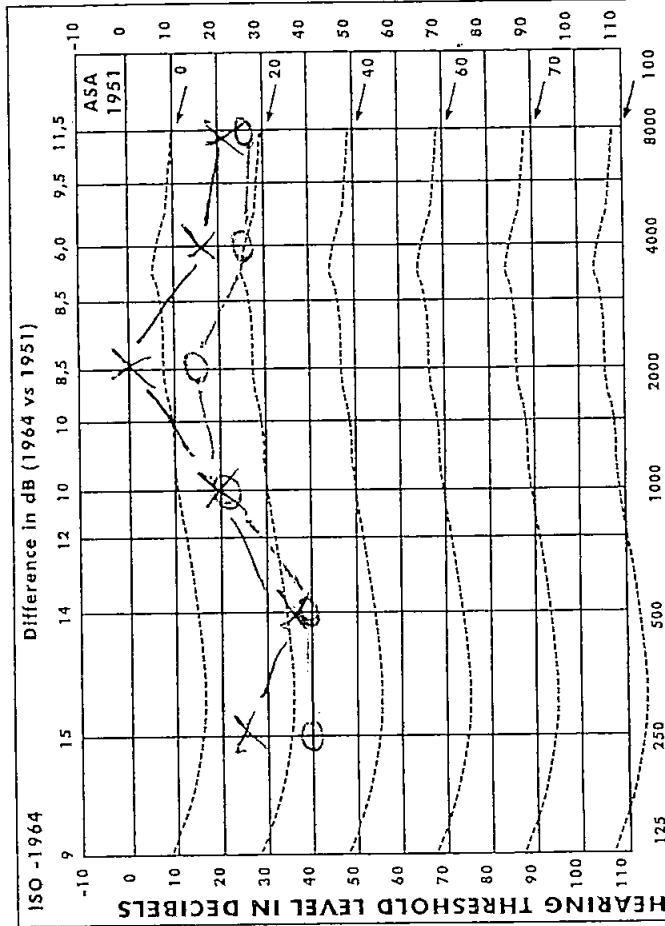


LA FUNDACION

*Usta Para Todos*

# AUDIOGRAM

NAME: Raul Davago  
 DATE: \_\_\_\_\_ BY \_\_\_\_\_



LEFT EAR		RIGHT EAR	
BLUE	RED	BLUE	RED
X		O	
AIR			
BONE COND. W. MASKING			
DB EFFECTIVE MASKING			
A. M. A. % HEARING LOSS			
L.	R.	COMB.	
LEFT EAR		RIGHT EAR	
S. R. T.			
M. C. L.			
T. D.			

This audiogram is plotted on the basis of:  
 1964 ISO reference thresholds  
 1951 ASA reference thresholds  
 (Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Threshold Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



LA FUNDACION

Vista Para Todos

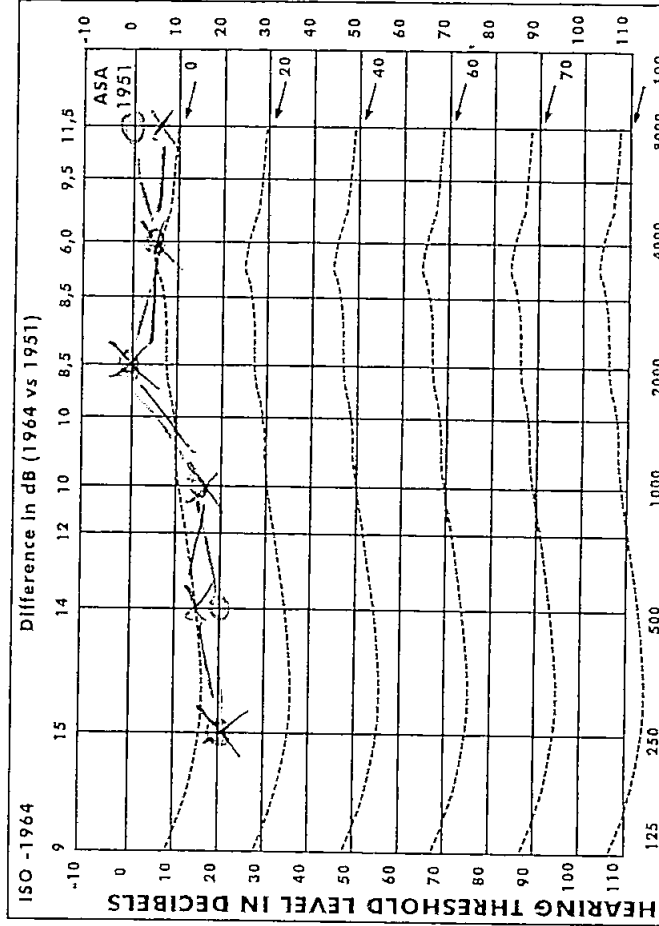
AUDIOGRAM

NAME:

Vallejo Antonio

DATE:

BY:



AIR	LEFT EAR	RIGHT EAR
	BLUE	RED
BONE COND. W. MASKING	X	O
	I	I
D B EFFECTIVE MASKING		

A. M. A. % HEARING LOSS	
L.	R.
	COMB.

LEFT EAR		RIGHT EAR	
S. R. T.			
M. C. L.			
T. D.			

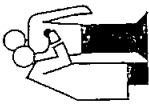
This audiogram is plotted on the basis of:

1964 ISO reference thresholds

1951 ASA reference thresholds

(Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Thresholds Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



LA FUNDACION

Vista Para Todos

NAME:

*Delia H. V...*

AUDIOGRAM

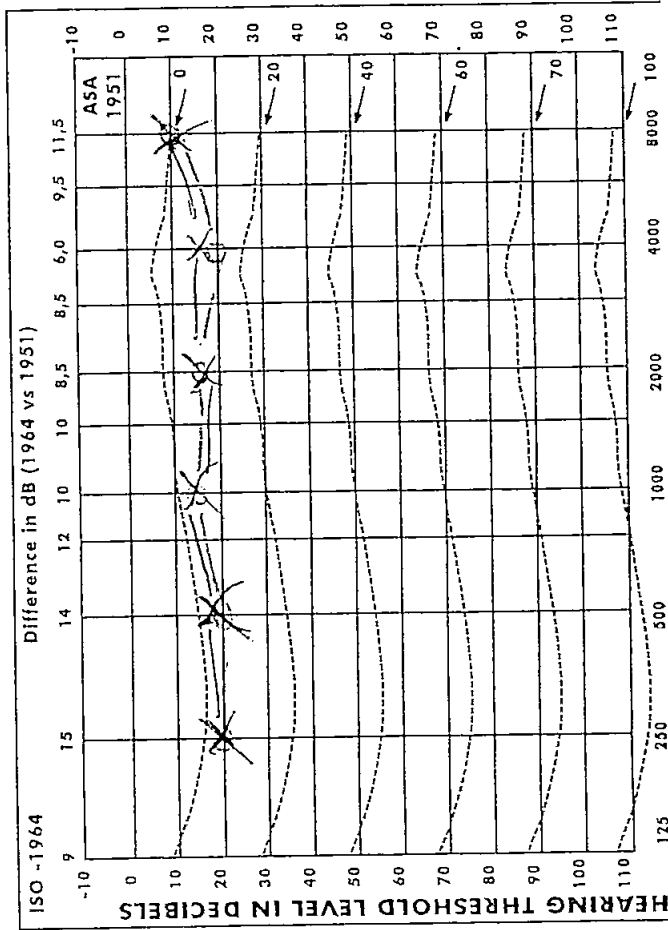
DATE:

BY:

AIR	LEFT EAR	RIGHT EAR
	BLUE	RED
BONE COND. W. MASKING	X	O
	J	I
EFFECTIVE MASKING		

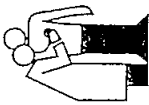
A. M. A. % HEARING LOSS	
L.	R.
	COMB.

LEFT EAR	RIGHT EAR
S. R. T.	
M. C. L.	
T. D.	



This audiogram is plotted on the basis of:  
 1964 ISO reference thresholds  
 1951 ASA reference thresholds  
 (Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Thresholds Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



LA FUNDACION

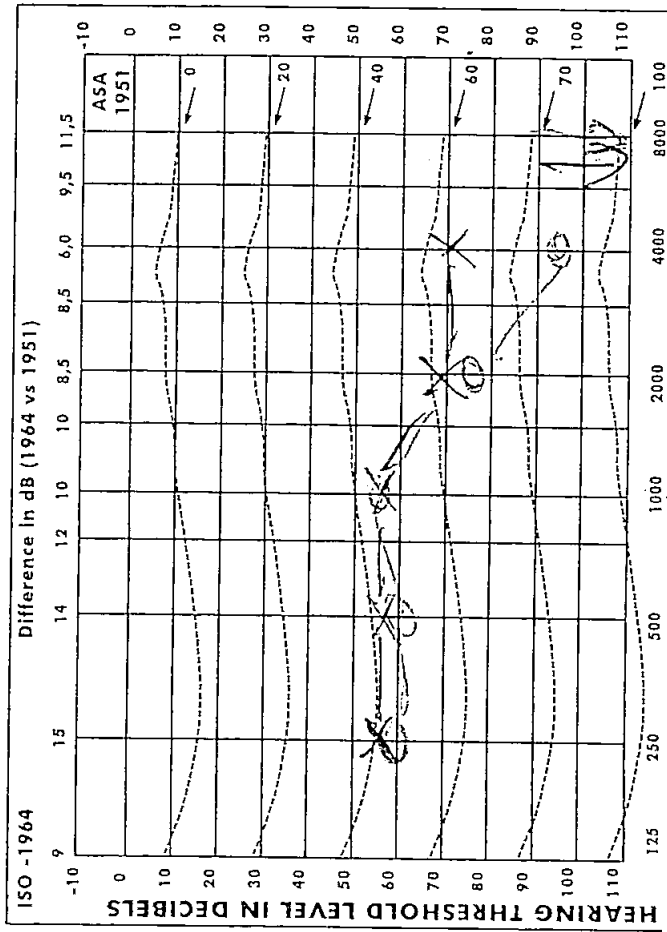
Vista Para Todos

AUDIOGRAM

NAME: C. Canales

BY: Bush

DATE:



AIR	LEFT EAR		RIGHT EAR
	BLUE	RED	RED
BONE COND. W. MASKING DB	X		O
EFFECTIVE MASKING	J		I

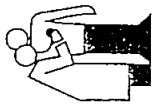
A. M. A. % HEARING LOSS	
L.	R.
	COMB.

LEFT EAR		RIGHT EAR
S. R. T.		
M. C. L.		
T. D.		

This audiogram is plotted on the basis of:

- 1964 ISO reference thresholds
  - 1951 ASA reference thresholds
- (Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Thresholds Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



LA FUNDACION

Vista Para Todos

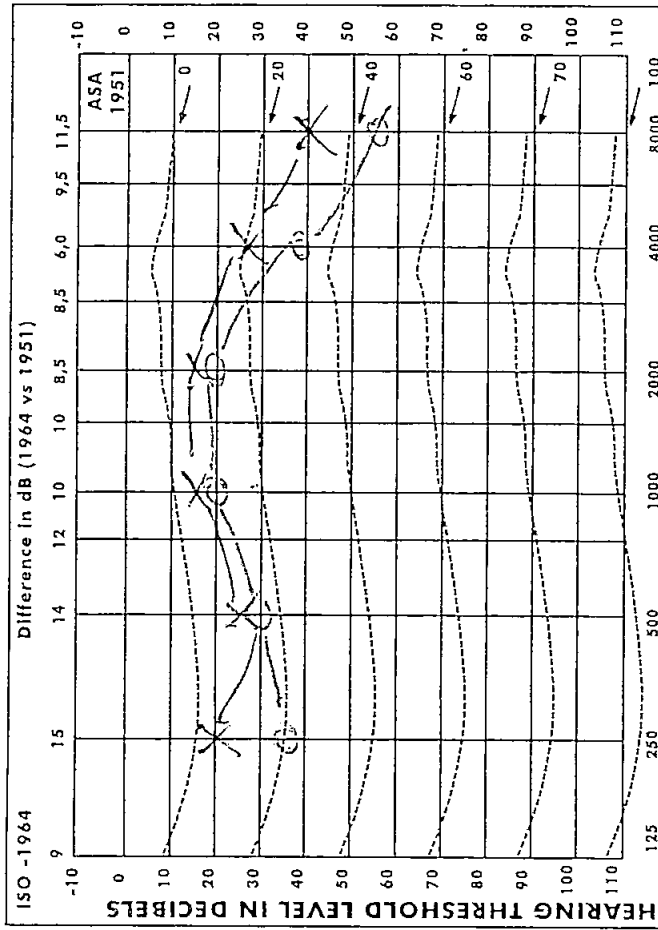
NAME:

*Antonio Roberto*

# AUDIOGRAM

DATE:

BY



AIR	LEFT EAR	RIGHT EAR
	BLUE	RED
BONE COND. W. MASKING	X	O
	I	I
D.B. EFFECTIVE MASKING		

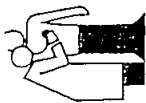
A. M. A. % HEARING LOSS	
L.	R.
	COMB.

LEFT EAR	RIGHT EAR
S. R. T.	
M. C. L.	
T. D.	

This audiogram is plotted on the basis of:

- 1964 ISO reference thresholds
  - 1951 ASA reference thresholds
- (Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Threshold Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



LA FUNDACION

*Vista Para Todos*

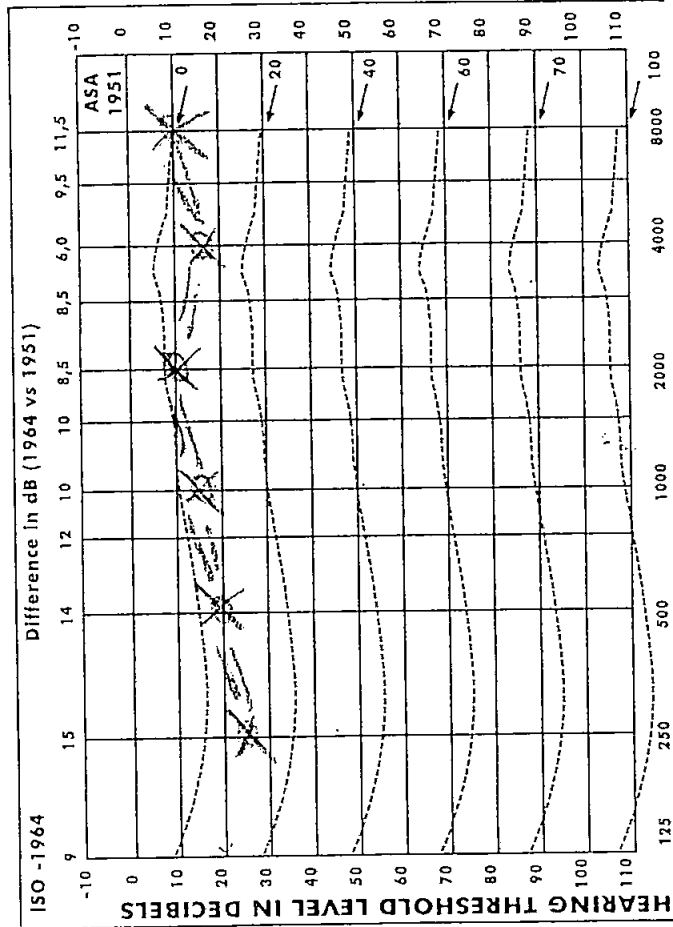
NAME:

*Raúl Barragán*

# AUDIOGRAM

DATE:

BY



AIR	LEFT EAR	RIGHT EAR
	BLUE	RED
BONE COND. W. MASKING DB	X	O
	]	[
EFFECTIVE MASKING		

A. M. A. % HEARING LOSS	
L.	R.
	COMB.

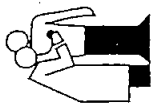
S. R. T.	LEFT EAR	RIGHT EAR
M. C. L.		
T. D.		

This audiogram is plotted on the basis of:

- 1964 ISO reference thresholds
- 1951 ASA reference thresholds

(Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Thresholds Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



LA FUNDACION

Vista Para Todas

NAME:

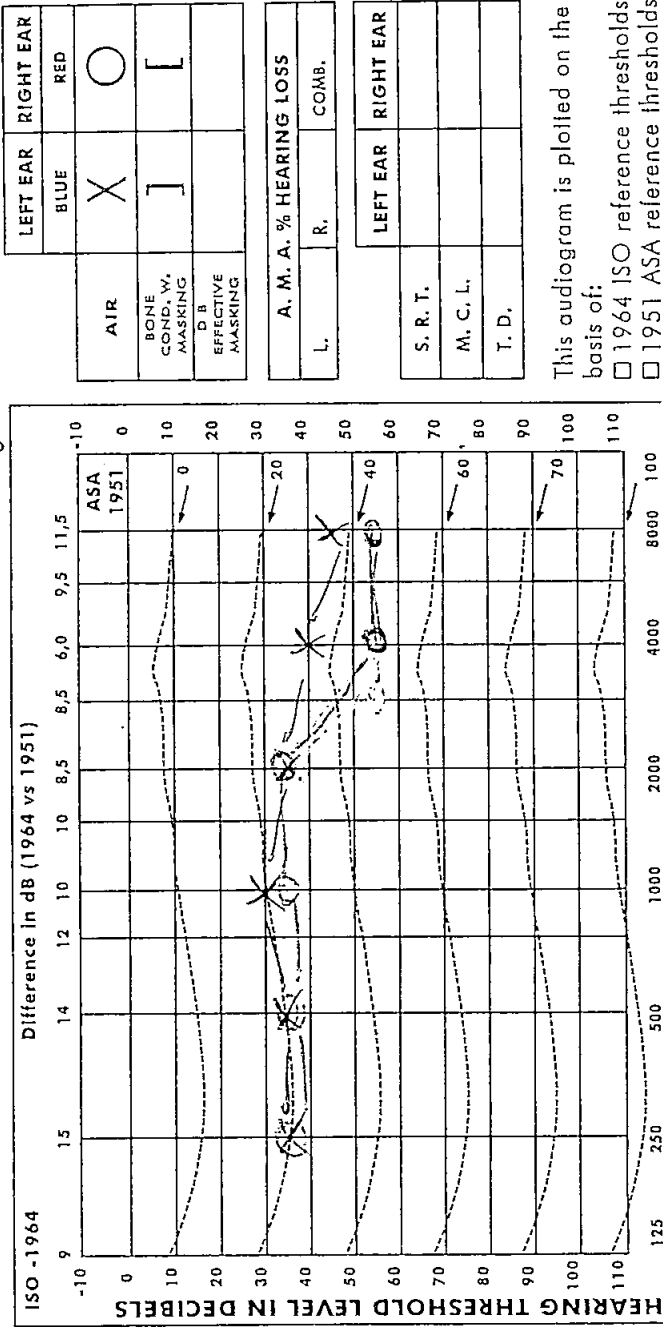
Columen faith

AUDIOGRAM

DATE:

1 VIII 1987

BY:



This audiogram is plotted on the basis of:

- 1964 ISO reference thresholds
- 1951 ASA reference thresholds

(Check one of these squares)

Readings obtained on an audiometer calibrated to the 1951, ASA thresholds may be converted to, and plotted as, "Hearing Thresholds Levels" based on the 1964 ISO reference thresholds by adding the appropriate "Difference in dB" at each frequency. To convert readings based on the 1964 ISO reference thresholds to readings based on the 1951 ASA reference thresholds, subtract the "difference in dB".



# **Anexo 14**

## Tabla de “t” de Student

Tomado de : **DANIEL Wayne, Bioestadística. Tercera Edición. Editorial Limusa. México. 1974**

Tabla H Percentiles de la distribución  $t$ .



G.L.	$P(t_{10} < 2.2281) = .975$		$P(t_{10} < 2.2281) = .975$		$P(t_{10} < 2.2281) = .975$	
	$t_{.95}$	$t_{.975}$	$t_{.95}$	$t_{.975}$	$t_{.95}$	$t_{.975}$
1	3.078	6.3138	12.706	31.821	63.657	
2	1.886	2.9200	4.3027	6.965	9.9248	
3	1.638	2.3534	3.1825	4.541	5.8409	
4	1.533	2.1318	2.7764	3.747	4.6041	
5	1.476	2.0150	2.5706	3.365	4.0321	
6	1.440	1.9432	2.4469	3.143	3.7074	
7	1.415	1.8946	2.3646	2.998	3.4995	
8	1.397	1.8595	2.3060	2.896	3.3554	
9	1.383	1.8331	2.2622	2.821	3.2498	
10	1.372	1.8125	2.2281	2.764	3.1693	
11	1.363	1.7959	2.2010	2.718	3.1058	
12	1.356	1.7823	2.1788	2.681	3.0545	
13	1.350	1.7709	2.1604	2.650	3.0123	
14	1.345	1.7613	2.1448	2.624	2.9768	
15	1.341	1.7530	2.1315	2.602	2.9467	
16	1.337	1.7459	2.1199	2.583	2.9208	
17	1.333	1.7396	2.1098	2.567	2.8982	
18	1.330	1.7341	2.1009	2.552	2.8784	
19	1.328	1.7291	2.0930	2.539	2.8609	
20	1.325	1.7247	2.0860	2.528	2.8453	
21	1.323	1.7207	2.0796	2.518	2.8314	
22	1.321	1.7171	2.0739	2.508	2.8188	
23	1.319	1.7139	2.0687	2.500	2.8073	
24	1.318	1.7109	2.0639	2.492	2.7969	
25	1.316	1.7081	2.0595	2.485	2.7874	
26	1.315	1.7056	2.0555	2.479	2.7787	
27	1.314	1.7033	2.0518	2.473	2.7707	
28	1.313	1.7011	2.0484	2.467	2.7633	
29	1.311	1.6991	2.0452	2.462	2.7564	
30	1.310	1.6973	2.0423	2.457	2.7500	
35	1.3062	1.6896	2.0301	2.438	2.7239	
40	1.3031	1.6839	2.0211	2.423	2.7045	
45	1.3007	1.6794	2.0141	2.412	2.6896	
50	1.2987	1.6759	2.0086	2.403	2.6778	
60	1.2959	1.6707	2.0003	2.390	2.6603	
70	1.2938	1.6669	1.9945	2.381	2.6480	
80	1.2922	1.6641	1.9901	2.374	2.6388	
90	1.2910	1.6620	1.9867	2.368	2.6316	
100	1.2901	1.6602	1.9840	2.364	2.6260	
120	1.2887	1.6577	1.9799	2.358	2.6175	
140	1.2876	1.6558	1.9771	2.353	2.6114	
160	1.2869	1.6545	1.9749	2.350	2.6070	
180	1.2863	1.6534	1.9733	2.347	2.6035	
200	1.2858	1.6525	1.9719	2.345	2.6006	
$\infty$	1.282	1.645	1.96	2.326	2.576	

Tabla tomada de:  
 Bioestadística  
 Wayne W. Daniel  
 Editorial Noriega  
 3ra edición  
 Tabla H