



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

REDUCCIÓN DE RUIDO EN SEÑALES DE VOZ
MEDIANTE UN FORMADOR DE HAZ

TESIS

QUE PARA OBTENER EL TÍTULO DE
INGENIERO ELÉCTRICO ELECTRÓNICO

P R E S E N T A

SAMUEL SALVADOR VÁZQUEZ SÁNCHEZ

DIRECTOR DE TESIS

M. I. Larry Hipólito Escobar Salguero



Ciudad Universitaria

2012

Agradezco a mis papás Lupita y Salvador por apoyarme incondicionalmente en cualquier nueva empresa y a mi hermano Benjamín por ser siempre mi mejor amigo.

Agradezco a la universidad por mejorar la vida de muchos jóvenes, entre ellos yo. En especial a mi asesor de tesis y a los profesores que me mostraron que la ingeniería es algo interesante y útil, además a Daniel Martínez y a Ricardo Dorantes por ayudarme en la realización de las pruebas en la cámara anecoica.

Contenido

1	Introducción	1
2	Análisis de señales	7
2.1	Clasificación y propiedades de las señales	7
2.1.1	Tiempo continuo y tiempo discreto	8
2.1.2	Señales periódicas y aperiódicas	8
2.1.3	Promedio, energía y potencia de una señal	8
2.1.4	Señales determinísticas y aleatorias	9
2.1.5	Señales estacionarias	9
2.1.6	Relación señal a ruido	9
2.1.7	Ancho de banda de una señal	10
2.2	Análisis en frecuencia	10
2.2.1	Series de Fourier	11
2.2.2	Transformada de Fourier	12
2.2.3	Transformada de Fourier para tiempo discreto	14
2.3	Digitalización de una señal	15
2.3.1	Teorema de muestreo de Nyquist	15
2.3.2	Submuestreo	16
2.3.3	Interpolación	17
2.4	Filtros digitales	18
2.4.1	Filtros de respuesta infinita al impulso (IIR)	18
2.4.2	Filtros de respuesta finita al impulso (FIR)	20
2.4.3	Comparación de filtros IIR y FIR	21
2.5	Resumen	21
3	Acústica	23
3.1	Propagación del sonido	24
3.2	Acústica de recintos	28
3.2.1	Reflejos y dispersión	29
3.2.2	Reverberación y patrones de interferencia	31
3.3	Percepción humana	32
3.3.1	Oído humano	33
3.3.2	Voz humana	34

3.4	Micrófonos	35
3.5	Ruido y distorsión	36
3.6	Resumen	38
4	Tipos de formadores de haz	39
4.1	Procesamiento de arreglos de micrófonos	40
4.2	Formador de haz fijo para banda angosta	42
4.2.1	Patrón de radiación	44
4.2.2	Ancho del haz	45
4.2.3	Lóbulos gratinados	46
4.2.4	La técnica de ventanas	48
4.2.5	Direccionamiento del haz	49
4.2.6	Respuesta en frecuencia	50
4.3	Formador de haz de banda ancha	51
4.3.1	Formador de banda ancha usando filtros digitales	53
4.3.2	Formador de haz por Sub-bandas	54
4.4	Resumen	57
5	Diseño e implementación del sistema	59
5.1	Diseño del formador de haz	60
5.1.1	Restricciones y parámetros de diseño	61
5.1.2	Cálculo de los parámetros del formador de haz	63
5.2	Implementación del formador de haz en el DSP	67
5.2.1	Adquisición de datos	69
5.2.2	Cómputo de los filtros FIR	71
5.2.3	Cómputo de los formadores de haz	73
5.2.4	Manejo de procesos multitasa	74
5.2.5	Diagrama general del proceso de filtrado	75
5.2.6	Eficiencia computacional del programa	76
5.3	Composición física del sistema	77
5.3.1	Bloque de adquisición de señales	78
5.3.2	Bloque de transmisión de resultados	78
5.4	Resumen	81
6	Simulación, experimentación y resultados	83
6.1	Simulación	83
6.1.1	Metodología para la simulación	83
6.1.2	Resultados de la simulación	84
6.2	Pruebas en la cámara anecoica	87
6.2.1	Descripción de las condiciones de prueba en la cámara anecoica	88
6.2.2	Resultados en la cámara anecoica	89
6.3	Pruebas en el laboratorio de PDS	91
6.3.1	Descripción de las condiciones de prueba en el laboratorio de PDS	91

6.3.2	Influencia de los patrones de interferencia	93
6.3.3	Resultados en el laboratorio de PDS	95
6.4	Comparación de los resultados obtenidos	98
6.5	Resumen	98
7	Conclusiones	101
	Bibliografía	103
	Glosario	109
	Anexos	113

Índice de figuras

1.1	<i>Suma destructiva y constructiva de tres señales sinusoidales.</i>	2
1.2	<i>Esquema de un formador de haz fijo simple.</i>	3
1.3	<i>Esquema de un formador de haz fijo de banda ancha por separación de bandas.</i>	3
2.1	<i>SNR en señales de voz contra inteligibilidad [20].</i>	10
2.2	<i>Tres señales diferentes caracterizadas por la misma secuencia discreta.</i>	16
2.3	<i>Sub-muestreo de una señal discreta.</i>	17
2.4	<i>Sobre-muestreo de una señal discreta.</i>	18
2.5	<i>Estructura de un filtro IIR implementado en forma directa I.</i>	19
2.6	<i>Estructura de un filtro FIR.</i>	20
3.1	<i>Propagación de perturbaciones mecánicas en el aire.</i>	23
3.2	<i>Región de alta presión moviéndose a través de un tubo [20].</i>	25
3.3	<i>Atenuación del sonido en aire a 20 C. Humedad relativa: a) 0 %, b) 10 %, c) 40 %, d) 100 % [16].</i>	28
3.4	<i>Reflexión de una onda de sonido que incide perpendicularmente en una pared.</i>	29
3.5	<i>Reflexión de una onda de sonido que incide en una pared con irregularidades: (a) $d \ll \lambda$, (b) $d \sim \lambda$, (c) $d \gg \lambda$ [17].</i>	30
3.6	<i>Tiempos de reverberación de distintos tipos de recinto [20].</i>	31
3.7	<i>Patrón de interferencia que provoca una onda sinusoidal en una superficie reflejante.</i>	32
3.8	<i>Contornos de nivel para una intensidad igual en tonos puros en todo el espectro audible [20].</i>	34
3.9	<i>Comparación de intervalos de frecuencia contra intensidad de sonido para el espectro audible, música y voz [7].</i>	35
4.1	<i>Esquema general de un arreglo de cuatro micrófonos con tres señales incidentes.</i>	41
4.2	<i>Esquema de un arreglo lineal de N micrófonos.</i>	42
4.3	<i>Esquema de un formador de haz fijo de banda angosta.</i>	43
4.4	<i>Patrón de radiación de un formador de haz fijo con cuatro micrófonos.</i>	45

4.5	<i>Patrón de radiación de un formador de haz fijo con ocho micrófonos que presenta aliasing espacial.</i>	47
4.6	<i>Patrón de radiación de un formador de haz fijo con ocho micrófonos utilizando ventanas.</i>	49
4.7	<i>Patrón de radiación de un formador de haz fijo con ocho micrófonos apuntando a -40°.</i>	51
4.8	<i>Patrón de radiación de un formador de haz fijo con ocho micrófonos en el intervalo 150 a 4000 Hz.</i>	52
4.9	<i>Estructura general de un formador de haz de banda ancha.</i>	53
4.10	<i>Apertura continua unidimensional.</i>	54
4.11	<i>Estructura del formador de haz por sub-bandas.</i>	55
4.12	<i>Respuesta de un formador de haz por sub-bandas.</i>	56
4.13	<i>Vista superior de las respuestas de un formador de banda ancha usando sub-bandas (izquierda) y un formador de haz de banda angosta (derecha).</i>	56
5.1	<i>Esquema funcional del sistema.</i>	59
5.2	<i>Esquema general de un formador de haz de banda ancha por el método de sub-bandas.</i>	60
5.3	<i>Restricciones y parámetros de diseño del sistema.</i>	61
5.4	<i>Esquema de la distribución de micrófonos.</i>	66
5.5	<i>Ambiente de desarrollo Code Composer Studio v 4.1.</i>	68
5.6	<i>Recorrido de la ventana de datos.</i>	69
5.7	<i>Esquema conceptual de memoria con direccionamiento circular.</i>	70
5.8	<i>Manejo de listas circulares en memoria lineal.</i>	71
5.9	<i>Diferencia en el cálculo de filtros FIR con buffers lineales y buffers circulares.</i>	72
5.10	<i>Respuesta en frecuencia de los filtros usados para la separación de bandas.</i>	73
5.11	<i>Diagrama del cómputo de un formador de haz simple con ventana.</i>	73
5.12	<i>Diagrama del manejo de procesos multitasa.</i>	74
5.13	<i>Esquema conceptual del programa en el DSP.</i>	75
5.14	<i>Distribución del tiempo usado en cada proceso.</i>	77
5.15	<i>Bloques del sistema.</i>	77
5.16	<i>Diagrama del circuito acondicionador de señal.</i>	79
5.17	<i>Esquema del circuito conversor digital-analógico.</i>	79
5.18	<i>Esquema del circuito del microcontrolador PIC y la pantalla LCD gráfica.</i>	80
5.19	<i>Ejemplo de la información que despliega la pantalla gráfica.</i>	80
6.1	<i>Respuesta en frecuencia de las sub-bandas. De izquierda a derecha y de arriba a abajo: Banda 1, banda 2, banda 3 y banda 4.</i>	85
6.2	<i>Vista tridimensional de los resultados usando ventanas Chebyshev-Dolph con una disminución de 30 dB para las bandas 1, 2 y 3, y usando una ventana Chebyshev-Dolph con una disminución de 10 dB para la banda 4.</i>	86

6.3	<i>Vista superior de los resultados usando ventanas Chebyshev-Dolph con una disminución de 30 dB para las bandas 1, 2 y 3, y usando una ventana Chebyshev-Dolph con una disminución de 10 dB para la banda 4.</i> . . .	86
6.4	<i>Respuesta del algoritmo al ruido blanco por sub-banda.</i>	87
6.5	<i>Respuesta conjunta del algoritmo al ruido blanco.</i>	87
6.6	<i>Disposición de objetos durante las pruebas en la cámara anecoica.</i> . . .	88
6.7	<i>Rendimiento del sistema usando ruido rosa y ruido blanco en la cámara anecoica.</i>	89
6.8	<i>Rendimiento del sistema usando tonos de 250 Hz, 500 Hz, 1200 Hz, 2000 Hz y 4000 Hz en la cámara anecoica.</i>	90
6.9	<i>Disposición de objetos durante las pruebas en laboratorio de procesamiento digital de señales.</i>	91
6.10	<i>Interpolación para encontrar el tiempo de reverberación T_{60} en el laboratorio de PDS.</i>	92
6.11	<i>Posiciones vertical y horizontal del arreglo de micrófonos.</i>	93
6.12	<i>Efecto de los patrones de interferencia en la colocación del banco de micrófonos.</i>	93
6.13	<i>Diferentes posiciones del arreglo en las pruebas en el laboratorio.</i>	94
6.14	<i>Dimensiones del difusor.</i>	94
6.15	<i>Rendimiento del sistema con ruido blanco por bandas en el laboratorio de procesamiento digital de señales con los micrófonos en posición horizontal.</i> 96	96
6.16	<i>Rendimiento del sistema con ruido blanco en el Laboratorio de Procesamiento Digital de Señales con los micrófonos en posición horizontal.</i> .	96
6.17	<i>Rendimiento del sistema con ruido blanco por bandas en el Laboratorio de Procesamiento Digital de Señales con los micrófonos en posición vertical.</i> 97	97
6.18	<i>Rendimiento del sistema con ruido blanco en el Laboratorio de Procesamiento Digital de Señales con los micrófonos en posición vertical.</i> . .	97
6.19	<i>Comparación del rendimiento del sistema con ruido blanco en distintas circunstancias.</i>	98
7.1	<i>Diagrama de la cámara anecoica [6].</i>	113
7.2	<i>Esquema general de la arquitectura del DSP TMS320F28335 [27].</i> . . .	116
7.3	<i>Esquema de buses [28].</i>	118
7.4	<i>Esquema de multiplexión de interrupciones [32].</i>	119
7.5	<i>Esquema del módulo de conversión analógica-digital [30].</i>	120
7.6	<i>DSP TMS320F28335 montado en el docking station y sistema completo en funcionamiento.</i>	144
7.7	<i>Circuito de conversión digital a analógica (izquierda) y circuito acondicionador de las señales de los micrófonos.</i>	144
7.8	<i>Microcontrolador PIC18F4550 y la pantalla gráfica mostrando un ejemplo del patrón de radiación, la dirección apuntada y la potencia en esa dirección.</i>	145

7.9 *Sistema montado dentro de la cámara anecoica del CCADET y analizador de espectro marca B & K.* 145

Índice de cuadros

2.1	<i>Propiedades de la transformada de Fourier.</i>	14
3.1	<i>Velocidad del sonido en diferentes medios de propagación [16], [20].</i> . .	26
4.1	<i>Definición de ventanas comunes.</i>	48
5.1	<i>Propuesta de la distribución de las sub-bandas.</i>	63
5.2	<i>Distancia de campo lejano con cuatro sub-bandas y ocho micrófonos en cada sub-arreglo.</i>	64
5.3	<i>Distancias de campo lejano para cada banda con sub-muestreo.</i>	64
5.4	<i>Matriz de retardos por micrófono y dirección dados en T_s.</i>	65
5.5	<i>Número total de instrucciones en cada T_s.</i>	76
6.1	<i>Efecto del difusor en los patrones de interferencia.</i>	95
6.2	<i>Comparación de los resultados obtenidos.</i>	99

Resumen

Hoy más que nunca existe interés en diseñar sistemas que capturen, procesen y/o almacenen señales de una forma limpia, sin ruido ni distorsión. Debido a la actual demanda programas de dictado, sistemas controlados con comandos por voz y sistemas multimedia que funcionan en tiempo real como las videoconferencias, la voz tiene un rol protagonista en las interfaces actuales de usuario y en la experimentación en las nuevas formas de interacción humano-máquina.

En casi cualquier lugar, las señales de audio están afectadas por ruido ambiental ya sean otras fuentes de voz, ruido de motores, música, entre otras. Estas fuentes de ruido pueden contener señales con un ancho de banda similar a la señal de voz de interés, lo que complica diseñar un filtro adecuado para capturarla. Sin embargo, aunque las señales tengan características frecuenciales iguales, pueden estar generándose desde una posición distinta respecto al transductor que las recibe. Una parte del procesamiento digital se centra en el procesamiento de arreglos de transductores, donde la señal se puede analizar de forma espacio-temporal, esto ha dado lugar a la técnica llamada *formador de haz*, que es el tema central de este trabajo.

En esta tesis se detalla el desarrollo de un sistema que emplea un formador de haz de banda ancha para filtrar espacialmente señales audibles con un ancho de banda similar a la voz. Tal sistema se construyó usando trece micrófonos y un DSP TMS320F28335 de Texas Instruments, además de algunos otros circuitos que fueron necesarios para su correcto funcionamiento y para implementar una interfaz de usuario básica de fácil uso.

El sistema filtra una señal de voz que proviene de una dirección en el intervalo de $\pm 80^\circ$ respecto a una línea perpendicular al arreglo mientras que atenúa en tiempo real otros sonidos que inciden al arreglo de micrófonos en otros ángulos de arribo. La señal filtrada se convierte a un voltaje analógico para que pueda usarse en otros circuitos para su transmisión, almacenamiento o para otro tipo de post-procesamiento. La interfaz de usuario se implementó con un potenciómetro y una pantalla; para cambiar de dirección apuntada, el usuario mueve la perilla del potenciómetro indicando al DSP que ajuste mediante software la dirección del formador de haz; además, la potencia de la señal en la dirección actual y direcciones anteriores se muestran de forma gráfica y alfanumérica en una pantalla LCD de 64×128 pixeles.

El rendimiento del sistema fue evaluado en un ambiente anecoico (cámara anecoica del CCADET, UNAM) y en un ambiente reverberante cotidiano (Laboratorio de Procesamiento Digital de Señales en la Facultad de Ingeniería, UNAM).

Abstract

More than ever, today there is an interest in designing systems with the ability to receive, process and/or store clean signals, neither noisy nor distorted. Due to the current demand for speech-to-text programmes, voice-controlled systems and real-time multimedia applications such as videoconferencing, the human voice has a major role in current user interfaces and the research for new kinds of human-machine interaction.

Almost everywhere audio signals are corrupted by environmental noise such as voices, engines, music, among others. These sources of noise may produce signals with speech-like spectrum characteristics, that yield to complicated filter design. Despite those signals may have very similar spectrum characteristics they could be generated by different sources in different relative positions to the receiver. Some fields of study in digital signal processing are focused on array signals that allows to analyse information in a spatio-temporal manner, these studies has lead to the *beamforming* technique for spatial filtering, the main topic of this particular work.

In this thesis the development and construction of a system based on a wideband beamforming for speech spatial filtering is described. Such system was build with a Texas Instrument's DSP TMS320F28335 processing signals of thirteen microphones; in addition, some extra circuitry was used to capture those signals properly and implement a basic easy to use user interface.

The system amplifies speech signals impinging the microphones from one particular direction between $\pm 80^\circ$ while attenuating in real-time other signals impinging the array from other directions. The result is then converted into an analog voltage so it can be used in other systems for transmission, storage or any other purpose. The user interface is implemented with a potentiometer and a display, to change the target direction the user turns the potentiometer to ask the DSP to steer the beamforming beam's direction, in addition, the power and direction of the signals currently and previously filtered are displayed on a graphic LCD display of 64×128 pixels.

The performance was measured in anechoic conditions (anechoic chamber at CCADET, UNAM) and in reverberant conditions (Digital Signal Processing Laboratory at Engineering Faculty, UNAM)

Capítulo 1

Introducción

Hoy en día existe un gran interés en diseñar sistemas digitales que mejoren la calidad de señales recibidas para su posterior análisis o almacenamiento; entre ellas se encuentra la voz, la cual es una señal difícil de manipular pero de gran valor en muchas aplicaciones. Sin importar el lugar donde se esté, las señales acústicas casi siempre contienen una cierta cantidad de ruido provenientes de distintas fuentes. Esto provoca que las señales no puedan ser capturadas y procesadas en una forma pura, afectando profundamente la comunicación entre humanos o entre una máquina y un humano. Una parte del procesamiento digital de señales se centra en la manipulación de señales de arreglos de transductores para resolver problemas donde la información se debe analizar en una forma espacio-temporal. Esto ha dado lugar al *formador de haz*, el cual es el tema medular de esta tesis.

En este trabajo se presenta el diseño e implementación de un sistema que logre filtrar una señal de voz que proviene de una sola dirección inmersa en un ambiente con fuentes de ruido localizadas en direcciones distintas a la señal de interés, logrando suprimir señales indeseables que no se podrían separar con filtros clásicos de ponderación frecuencial.

Objetivos

General:

- *Diseñar y construir un sistema para la reducción de ruido en señales de voz que sirva como base para implementar algoritmos en el filtrado espacial.*

Específicos:

- *Estudiar aspectos básicos de la naturaleza de las señales de voz.*
- *Investigar tipos de formadores de haz y simular algunos en Matlab.*
- *Elegir un algoritmo de filtrado espacial adecuado para las capacidades computacionales de la arquitectura del DSP TMS320F28335 de Texas Instruments.*

- Optimizar el código del programa en el DSP para que el sistema funcione en tiempo real.
- Evaluar el rendimiento del sistema en ambientes anecoicos y reverberantes.

Descripción del método

Para lograr filtrar las señales a partir de su dirección de arribo al banco de micrófonos se puede usar el hecho de que las señales presentan un ángulo de desfase (también denominado desfasamiento) entre micrófono y micrófono, en función de su dirección de arribo y velocidad de propagación. Un concepto que usa este hecho para filtrado espacial es el llamado *formador de haz*.

Un formador de haz fijo simple se basa en la suma destructiva o constructiva de señales con desfase, cuando varias señales están en fase el resultado tiene una amplificación máxima, cuando las señales no están en fase la amplificación no es máxima, lo cual se ilustra en la Figura 1.1 utilizando señales sinusoidales.

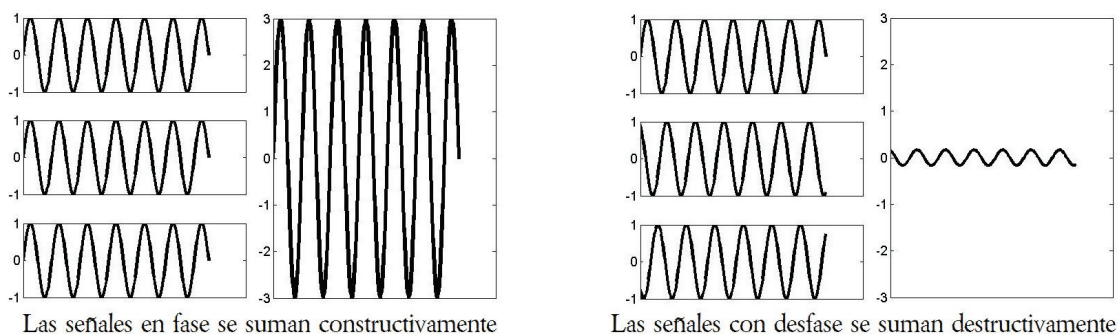


Figura 1.1 Suma destructiva y constructiva de tres señales sinusoidales.

El método formador de haz en su forma más simple tiene la misma estructura de un filtro de respuesta finita al impulso (*FIR*) en el sentido que corresponde a la suma de una serie de datos retrasados en el tiempo multiplicados por una constante. La Figura 1.2 muestra una representación gráfica de este proceso. Si el haz apunta a θ° respecto a la perpendicular del banco de micrófonos, los retrasos son nulos [3].

Los coeficientes w_n se pueden calcular por medio de métodos de ventanas como Hamming, Chebyshev, exponencial, cuadrada, etcétera, para modificar el rendimiento del formador, tal como se haría para un filtro FIR. Por otro lado, se debe considerar que un formador de haz como el descrito anteriormente se comporta como la versión espacial de un filtro pasa bajas por lo que no es conveniente usarlo con una señal de banda ancha como la voz. Para mejorar el rendimiento del formador de haz se divide la señal de banda ancha en varios intervalos que se puedan manejar por separado.

A lo largo del tiempo se han propuesto muchas variaciones y mejoras al formador de haz básico que se muestra en la Figura 1.2, entre los llamados formadores fijos están:

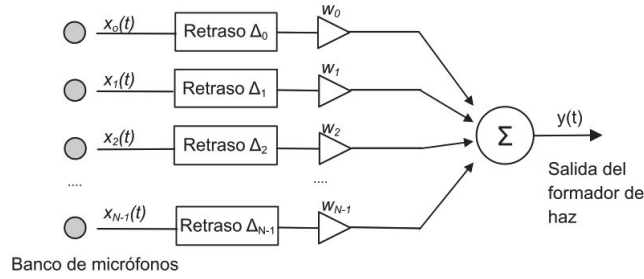


Figura 1.2 Esquema de un formador de haz fijo simple.

beamforming con ventanas, beamforming por suma y retardo, beamforming por filtrado y suma, beamforming por suma parcial, beamforming por interpolación, beamforming usando la Transformada Discreta de Fourier (DFT), entre otros [21].

En este trabajo se decidió usar un formador de haz de banda ancha por sub-bandas, cuya estructura se ilustra en la Figura 1.3. Este método tiene la facultad de mejorar la respuesta en señales de banda ancha porque cada micrófono es ponderado por coeficientes que dependen de la frecuencia de la señal. A pesar de ser un formador de haz de banda ancha simple en términos computacionales, se debe tener una frecuencia de muestreo f_s muy alta para poder direccionar el haz con precisión, donde f_s debe ser del orden de cuatro o cinco veces la frecuencia de la señal más alta e involucra muchas multiplicaciones y sumas debido a los múltiples filtros FIR y el proceso de interpolación asociado al mezclador de señales.

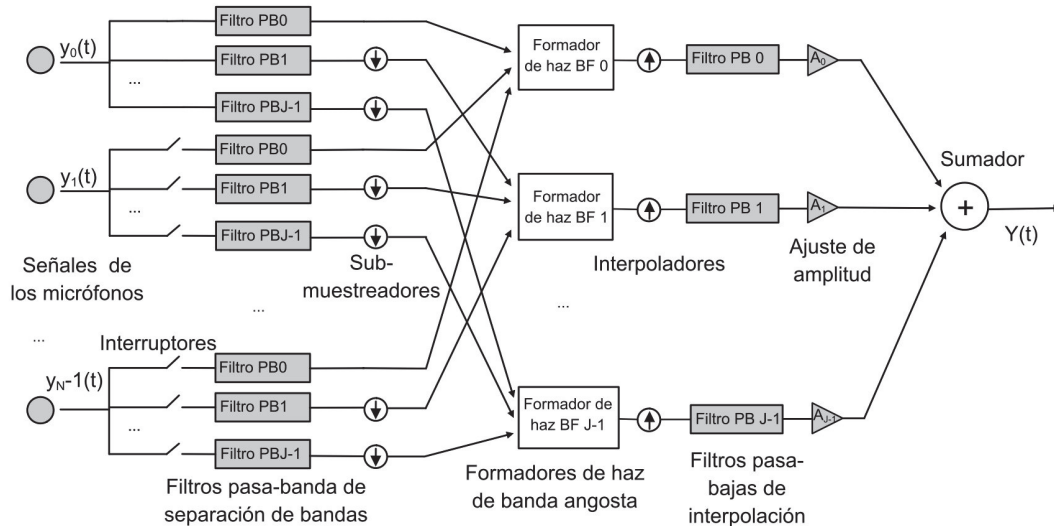


Figura 1.3 Esquema de un formador de haz fijo de banda ancha por separación de bandas.

Contribución del estudio presentado en esta tesis

Se ha demostrado desde hace tiempo que el ser humano tiene la habilidad de concentrarse voluntariamente en ciertos estímulos dentro de un ambiente con distractores, esto permite mantener una conversación en lugares colmados de ruido, música o conversaciones ajenas. A este interesante hecho se le ha denominado *el efecto de la fiesta de cocktail* y ha sido objeto de investigación por científicos a lo largo de décadas. Con base en experimentos psicoacústicos, se ha reportado que parte de esta habilidad para separar señales que comparten el mismo ancho de banda se debe al llamado *oído espacial*, es decir, que se puede identificar de dónde proviene la fuente de sonido de interés y después concentrarse únicamente en las señales que provienen de esa dirección particular; esto es posible debido la extraordinaria capacidad cognitiva como al hecho de que se poseen dos oídos [3]. En tanto que para cualquier humano es algo fácil y cotidiano resolver ese tipo de situaciones para una máquina es una tarea bastante más compleja.

El desarrollo de un sistema como éste es de gran importancia en cualquier proceso de adquisición de una señal de voz y puede ser una poderosa herramienta en la supresión de ruido en las siguientes circunstancias:

- Estudios de sonido
- Auditorios
- Cuartos con alta reverberación
- Dispositivos de ayuda auditiva
- Sistemas manos libres
- Sistemas de dictado *voz a texto*

Además, usando este tipo de tecnología prodría programarse un sistema similar que detecte, filtre o suprima una o varias señales simultaneamente en diferentes direcciones haciendo de este trabajo una plataforma propicia para el desarrollo de sistemas de reducción de ruido flexibles, completos y automatizados.

Esta tesis parte de los trabajos previos realizados sobre formadores de haz en el *Laboratorio de Procesamiento Digital de Señales* de la Facultad de Ingeniería, y extiende su aplicación para una señal de banda ancha, enfocándolo a la reducción de ruido en señales de voz.

Organización de los capítulos

En el Capítulo 2 *Análisis de señales*, se dan algunas definiciones importantes para la comprensión del problema a resolver. Se describe el concepto de *señal* en el ámbito ingenieril y sus clasificaciones más importantes para este trabajo. Posteriormente, se da

una introducción al análisis de frecuencia y se explican algunos conceptos importantes del manejo y filtrado de una señal digital.

En el Capítulo 3 *Acústica*, aborda la naturaleza física del sonido y se describe el oído y la voz humana. Además, se dan algunas definiciones útiles respecto al ruido.

En el Capítulo 4 *Tipos de formadores de haz*, se presenta la teoría básica que sirve para el entendimiento del formador de haz. Luego, sobre el formador de haz de banda angosta y se describe qué problemas existen al utilizarlo con señales de banda ancha. Al final, se presenta la teoría de un formador de haz de banda ancha.

En el Capítulo 5 *Diseño e implementación del sistema*, se explica el proceso de diseño para que, con base en la teoría y a las limitaciones físicas presentes, se tengan los mejores resultados posibles. También se detalla cómo se interpretó la teoría del capítulo 4 para trasladar el diseño a código en el DSP, teniendo en cuenta las capacidades computacionales del mismo. Además, se detallan todos los circuitos electrónicos y demás dispositivos involucrados en la tesis.

En el Capítulo 6 *Simulación, experimentación y resultados*, se exponen los resultados teóricos producto de simulaciones en *Matlab*. Posteriormente, se detalla las circunstancias en que se probó el rendimiento real del sistema y se incluyen las gráficas de rendimiento.

Finalmente, en el Capítulo 7 *Conclusiones*, se resumen los resultados obtenidos, se enuncian los objetivos logrados y se dan las observaciones finales.

Capítulo 2

Análisis de señales

Como la ingeniería es una actividad fundamentada en las ciencias exactas, se requiere una notación precisa y sencilla que pueda expresar ideas en forma condensada, para lograrlo se acude a las matemáticas. Este capítulo pretende proporcionar el contexto matemático y de herramientas en el análisis de señales en el que se desarrolla esta tesis.

En ingeniería se entiende por *señal* a una variable física que contiene información. Es la representación de las variaciones de algún fenómeno físico y es de utilidad en un gran número de áreas de la ciencia y la tecnología; como comunicaciones, aeronáutica, diseño de circuitos, acústica, sismología, control y procesamiento de voz [24]. Aún cuando la naturaleza física de las señales que surgen en estas disciplinas puede ser bastante diferente, todas ellas se pueden modelar y analizar de una forma matemáticamente similar. Las señales se pueden convertir de un tipo de energía a otra según convenga para su medición y control, por ejemplo, un micrófono cambia las vibraciones del aire a un voltaje analógico, luego un convertidor analógico a digital lo convierte a un código, el cual es fácil analizar y manipular en una computadora digital. De forma general, a los dispositivos que convierten una señal de un tipo de energía a otra se les llama *transductores*.

2.1 Clasificación y propiedades de las señales

Las señales se representan mediante *funciones* matemáticas de una o varias variables como el tiempo y el espacio, por lo que bajo este contexto se pueden usar estos términos *señal* y *función* sin hacer diferencia entre uno y otro. Las funciones tienen muchas características y formas de clasificación, a continuación se explicarán algunas definiciones que serán útiles para analizar y procesar una señal de voz a partir de su representación matemática.

Una función f es una regla que asigna a cada elemento de x de un conjunto D exactamente un elemento llamado $f(x)$ de un conjunto E . El conjunto D se llama *dominio* de la función. El *rango* de f es el conjunto de todos los valores posibles de $f(x)$ conforme x varía en todo el dominio.

2.1.1 Tiempo continuo y tiempo discreto

Las funciones continuas son las que varían respecto a una variable continua, por lo general el tiempo. Las funciones discretas son las que varían respecto a una variable discreta y sólo están definidas para ciertos valores de la variable independiente [5]. En este trabajo se usará la notación $x(t)$ para designar una función de tiempo continuo y $x[n]$ para designar una función de tiempo discreto.

Una función $x(t)$ es una función de tiempo continuo si t es una variable continua la cual puede tomar cualquier valor en un intervalo continuo (a, b) . Una función de tiempo continuo describe a una señal analógica. Una función $x[n]$ es una función de tiempo discreto por lo que n solo puede tomar valores enteros dentro de un intervalo $[c, d]$ [5]. Una función de tiempo discreto describe a una señal digital. Tanto $x(t)$ y $x[n]$ son funciones matemáticas que representan o modelan una señal física.

2.1.2 Señales periódicas y aperiódicas

Por su repetición en el tiempo, las señales se dividen en periódicas y aperiódicas. Si $x(t)$ es periódica entonces existe un número T tal que

$$x(t) = x(t + T), \quad \forall t \in \mathbb{R}.$$

En el tiempo discreto, $x[n]$ es periódica si existe un número entero N tal que

$$x[n] = x[n + N], \quad \forall n \in \mathbb{Z},$$

donde T y N son el periodo fundamental de las respectivas señales.

2.1.3 Promedio, energía y potencia de una señal

Se puede caracterizar una señal mediante su valor promedio, este valor representa una componente constante en la señal y se calcula como [5]

$$\mu = \lim_{T_1 \rightarrow \infty} \frac{1}{T_1} \int_{T_1} x(t) dt.$$

Si la señal es periódica con periodo T_0 en el tiempo continuo se puede reescribir como

$$\mu = \frac{1}{T_0} \int_{T_0} x(t) dt.$$

Comúnmente, la energía ε de la señal es de interés para el tiempo continuo en el intervalo (t_1, t_2) se define

$$\varepsilon_{(t_1, t_2)} = \int_{t_1}^{t_2} |x(t)|^2 dt$$

donde $|x(t)|$ denota la magnitud de x , posiblemente un número complejo. De manera similar para el caso discreto de una señal en el intervalo $[n_1, n_2]$ se tiene

$$\varepsilon_{[n_1, n_2]} = \sum_{n=n_1}^{n_2} |x[n]|^2.$$

Para el caso continuo la potencia promedio P se expresa como

$$P_{(t_1, t_2)} = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} |x(t)|^2 dt$$

y para el tiempo discreto

$$P_{[n_1, n_2]} = \frac{1}{n_2 - n_1 + 1} \sum_{n=n_1}^{n_2} |x[n]|^2.$$

2.1.4 Señales determinísticas y aleatorias

Otra clasificación de las señales se hace por la posibilidad de predecirlas, clasificándolas en determinísticas y aleatorias. Las señales determinísticas están bien definidas para cualquier tiempo dado y pueden ser modeladas con una función del tiempo conocida, es decir, no hay incertidumbre en su valor. Por ejemplo $x(t) = \text{sen}(t)$ es determinística. Una señal aleatoria $y(t) = \text{sen}(t) + n(t)$, donde $n(t)$ representa ruido, toma valores aleatorios en cualquier tiempo, porque existe incertidumbre sobre la señal $n(t)$ [37].

2.1.5 Señales estacionarias

Una señal es estacionaria si sus medidas de probabilidad no cambian con el tiempo, de otro modo se considera no-estacionaria. La propiedad de estacionariedad de las señales implica que todos los parámetros estadísticos tales como la media, varianza y potencia de las componentes espectrales, entre otros, se mantengan constantes. En la práctica, existen varios grados de estacionariedad, por ejemplo, las señales aleatorias como el ruido blanco tienen una media constante pero una potencia variable [34].

2.1.6 Relación señal a ruido

En el procesamiento de señales es común que se refiera al concepto de la *relación señal a ruido* (SNR). Esta cantidad se expresa en decibeles (dB) y corresponde al cociente de la potencia de la señal de interés entre la potencia del ruido [22], se expresa como

$$SNR = 10 \log_{10} \left[\frac{P_{\text{señal}}}{P_{\text{ruido}}} \right].$$

Cuando el ruido tiene mas energía que la señal de interés la SNR se vuelve negativa. En señales de voz la SNR está ligada a la inteligibilidad de las palabras, en la Figura

2.1 se muestra cómo la señal de voz se hace ininteligible cuando la SNR disminuye. Es de notar que para sílabas sueltas se necesita una SNR de ocho o más para considerar a la señal completamente inteligible, sin embargo, para palabras o frases completas esta restricción se vuelve mucho menos severa, hasta el punto de poder captar la mayor parte del mensaje con valores negativos de SNR, esto se debe a la gran capacidad de nuestro cerebro para analizar señales de voz que tienen un contenido semántico [20].

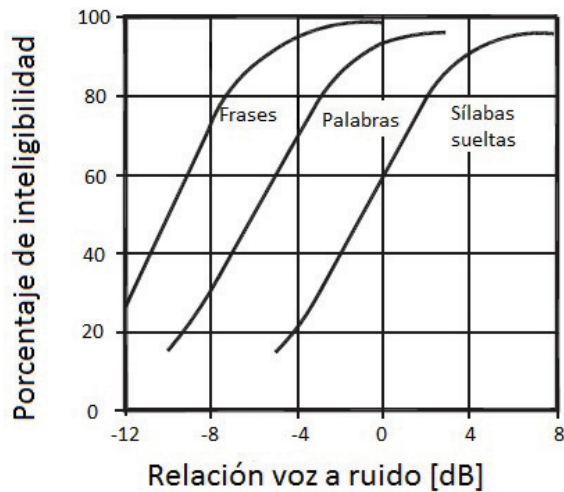


Figura 2.1 SNR en señales de voz contra inteligibilidad [20].

2.1.7 Ancho de banda de una señal

El ancho de banda de una señal (BW por sus iniciales en inglés) se puede definir de manera simplista como el intervalo de frecuencias donde se encuentra la mayor parte de la potencia de la señal. Es el intervalo comprendido entre la frecuencia $f_{inferior}$ hasta la frecuencia $f_{superior}$. Este mismo concepto también se puede aplicar para caracterizar filtros, transductores y cualquier sistema en general.

2.2 Análisis en frecuencia

La teoría de Fourier permite trasladar una señal o función del dominio del tiempo al dominio de la frecuencia, el resultado de esta transformación se le conoce como espectro de la señal. Este espectro es una función compleja de la variable frecuencia y usualmente se especifica en términos de su magnitud y ángulo de desfase, o simplemente fase, en forma separada. Como se detalla a continuación, si la señal es determinística se utiliza la transformada de Fourier, si la señal además de ser determinística es periódica se usa la Serie de Fourier. Esta teoría fue presentada por el físico francés Jean Baptiste Fourier (1768-1830) y es una herramienta poderosa para el modelado de sistemas físicos [15].

2.2.1 Series de Fourier

El desarrollo de la serie de Fourier para señales periódicas se logra mediante la descomposición de una señal dada en un conjunto de funciones ortogonales. La serie de Fourier es la mejor aproximación de un polinomio trigonométrico a una función dada.

Para entender cómo esto es posible, primero considérese que se tiene una función real y periódica f valuada en $[-\pi, \pi]$ que puede ser expresada como

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx) \quad (-\pi \leq x \leq \pi) \quad (2.1)$$

si se integra (2.1), se obtiene

$$\int_{-\pi}^{\pi} f(x) dx = \pi a_0 + \sum_{k=1}^{\infty} (a_k \int_{-\pi}^{\pi} \cos kx dx + b_k \int_{-\pi}^{\pi} \sin kx dx) = \pi a_0. \quad (2.2)$$

Entonces, asumiendo que todos los términos de la expresión son integrables, se tiene

$$a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) dx. \quad (2.3)$$

Ahora, para alguna $n \in \mathbb{N}$ se multiplica (2.1) por $\cos nx$ y se integra, obteniendo

$$\int_{-\pi}^{\pi} f(x) \cos nx dx = \frac{a_0}{2} \int_{-\pi}^{\pi} \cos nx dx + \sum_{k=1}^{\infty} (a_k \int_{-\pi}^{\pi} \cos kx \cos nx dx + b_k \int_{-\pi}^{\pi} \sin kx \cos nx dx).$$

Por ortogonalidad se sabe que $\int_{-\pi}^{\pi} \cos kx \cos nx dx = 0$ para $k, n = 0, 1, 2, \dots; k \neq n$ y que $\int_{-\pi}^{\pi} \cos kx \sin nx dx = 0$ para $k, n = 0, 1, 2, \dots$ se observa que únicamente un término $\int_{-\pi}^{\pi} \cos kx \cos nx dx$ es distinto a cero para $k = n$. Entonces,

$$\int_{-\pi}^{\pi} f(x) \cos kx dx = a_k \int_{-\pi}^{\pi} \cos^2 kx dx = \pi a_k,$$

entonces, para $k = 1, 2, \dots$

$$a_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos kx dx. \quad (2.4)$$

Nótese que si $k = 0$ en (2.4), se obtiene (2.3). Por eso se usó $\frac{a_0}{2}$ en (2.1) en lugar de a_0 . Siguiendo un procedimiento similar, se puede demostrar que

$$b_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin kx dx. \quad (2.5)$$

Si una función f es representable en la forma de (2.1), los coeficientes se encuentran usando (2.3), (2.4) y (2.5). Esto lleva a escribir que [13]

$$f = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos k\omega_0 t + b_k \sen k\omega_0 t) \quad (2.6)$$

donde $\omega_0 = \frac{2\pi}{T_0} = 2\pi f_0$. Para encontrar la forma de exponencial compleja de la serie de Fourier, usamos las identidades de Euler

$$\cos(\omega_0 t) = \frac{1}{2} e^{j\omega_0 t} + \frac{1}{2} e^{-j\omega_0 t} = \text{Re}\{e^{j\omega_0 t}\} \quad (2.7)$$

$$\sen(\omega_0 t) = \frac{1}{2j} e^{j\omega_0 t} - \frac{1}{2j} e^{-j\omega_0 t} = \text{Im}\{Ae^{j\omega_0 t}\}. \quad (2.8)$$

Sustituyendo (2.7) y (2.8) en (2.6) se obtiene

$$f = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2} + b_k \frac{e^{j\omega_0 t} - e^{-j\omega_0 t}}{2j} \right)$$

agrupando términos semejantes

$$f = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(\frac{a_k - jb_k}{2} e^{j\omega_0 t} + \frac{a_k + jb_k}{2} e^{-j\omega_0 t} \right)$$

se definen $c_0 = \frac{a_0}{2}$, $c_k = \frac{a_k - jb_k}{2}$ para $k \geq 1$, $c_k = \frac{a_{-k} + jb_{-k}}{2}$ para $k \leq -1$. De esta manera queda

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{j\omega_0 t}. \quad (2.9)$$

La ecuación (2.9) es la versión en notación de exponenciales complejas de la ecuación (2.6). Para calcular los coeficientes c_k se tienen

$$c_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx \quad (2.10)$$

$$c_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{j\omega_0 t} dx. \quad (2.11)$$

A la secuencia de coeficientes c_k de la serie se le denomina el espectro en frecuencia de la señal porque posee información sobre las frecuencias que contiene.

2.2.2 Transformada de Fourier

Como se puede intuir, la caracterización de una señal periódica en términos de su espectro en frecuencia es muy útil en el procesamiento de señales. Hasta el momento sólo se ha sustentado su uso para señales periódicas por lo que no se puede implementar

directamente esta idea a las señales que más conciernen a esta tesis: la voz humana y el ruido. Para aplicar una idea similar a señales no periódicas se formuló la *transformada de Fourier*. Fourier razonó que una señal aperiódica puede considerarse como una señal periódica de periodo infinito. En la representación en series de Fourier de una señal, conforme el periodo se incrementa la frecuencia fundamental disminuye y las componentes en frecuencia relacionadas armónicamente se hacen más cercanas una de otra. A medida que el periodo se hace infinito, las componentes en frecuencia se aproximan a un dominio continuo y la suma de la serie se convierte en una integral. Para estos casos el espectro está definido para todos los valores reales de la frecuencia angular ω , no solo para valores discretos múltiplos enteros de ω_0 . En otras palabras, para señales periódicas se tienen espectros discretos y para señales aperiódicas se tienen espectros continuos [24].

Para explicar esta idea considérese una señal $x(t)$ cuya duración es finita, es decir, $x(t) = 0$ si $|t| > T$, por lo que es evidente que $x(t)$ es aperiódica. A partir de esta señal se puede definir una señal periódica $\tilde{x}(t)$ para la cual $x(t)$ sea un periodo. Como se desea que el periodo T sea grande, $\tilde{x}(t)$ es idéntica a $x(t)$ durante un intervalo grande de tiempo, y conforme T tiende a infinito, \tilde{x} es igual a $x(t)$ en cualquier instante t .

De esta manera, recordando las ecuaciones (2.9) y (2.11) se puede escribir

$$\tilde{x}(t) = \sum_{k=-\infty}^{+\infty} c_k e^{jk\omega_0 t}$$

$$c_k = \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} \tilde{x}(t) e^{-jk\omega_0 t} dt \quad (2.12)$$

donde $\omega_0 = \frac{2\pi}{T_0}$. Ya que $\tilde{x}(t) = x(t)$ para $|t| < T_0$ y también $x(t) = 0$ fuera de este intervalo, la ecuación (2.12) se puede reescribir como

$$c_k = \frac{1}{T_0} \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} x(t) e^{-jk\omega_0 t} dt = \frac{1}{T_0} \int_{-\infty}^{+\infty} x(t) e^{-jk\omega_0 t} dt.$$

Por tanto, definiendo la envolvente $X(j\omega)$ como

$$X(j\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt \quad (2.13)$$

Se tiene para los coeficientes c_k

$$c_k = \frac{1}{T_0} X(jk\omega_0) \quad (2.14)$$

sabiendo que $\frac{2\pi}{T_0} = \omega_0$ y combinando las ecuaciones (2.13) y (2.14) se tiene

$$\tilde{x}(t) = \frac{1}{2\pi} \sum_{k=-\infty}^{+\infty} X(jk\omega_0) e^{jk\omega_0 t} \omega_0 \quad (2.15)$$

como ω_0 tiende a cero conforme T tiende a infinito, el miembro derecho de la ecuación (2.15) se vuelve una integral y ω_0 se convierte en un valor muy pequeño $d\omega$, finalmente se tiene

$$x(t) = \frac{1}{2\pi} \int_{k=-\infty}^{+\infty} X(jk\omega_0) e^{jk\omega_0 t} d\omega \quad (2.16)$$

$$X(j\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt. \quad (2.17)$$

A la ecuación (2.16) se le conoce como la transformada de Fourier, mientras que a la ecuación (2.17) como la transformada inversa de Fourier [24]. Este par de ecuaciones permiten cambiar la señal de un espacio temporal a uno frecuencial, donde es más fácil observar características de la composición espectral, lo cual es de gran ayuda para diseñar filtros digitales.

Cuadro 2.1 Propiedades de la transformada de Fourier.

Propiedad	Señal de tiempo continuo	Transformada de Fourier de tiempo continuo	Señal de tiempo discreto	Transformada de Fourier de tiempo discreto
<i>Linealidad</i>	$ax(t) + by(t)$	$aX(j\omega) + bY(j\omega)$	$ax[n] + bx[n]$	$aX(e^{j\omega}) + bY(e^{j\omega})$
<i>Desplazamiento de tiempo</i>	$x(t + t_0)$	$e^{j\omega t_0} X(j\omega)$	$x[n - n_0]$	$e^{-j\omega n_0} X(e^{j\omega})$
<i>Desplazamiento en frecuencia</i>	$e^{j\omega_0 t} x(t)$	$X(j(\omega - \omega_0))$	$e^{j\omega_0 n} x[n]$	$X(e^{j(\omega - \omega_0)})$
<i>Escalamiento de tiempo y frecuencia</i>	$x(at)$	$\frac{1}{ a } X\left(\frac{j\omega}{a}\right)$	$x_{(k)}[n] = \begin{cases} x\left[\frac{n}{k}\right] \\ 0 \end{cases}$ si n no es múltiplo de k , $x_{(k)}[n] = 0$	$X(e^{jk\omega})$
<i>Multiplicación</i>	$x(t)y(t)$	$\frac{1}{2\pi} X(j\omega) * Y(j\omega)$	$x[n]y[n]$	$\frac{1}{2\pi} \int_{2\pi} X(e^{j\omega}) Y(e^{j(\omega - \theta)}) d\theta$

2.2.3 Transformada de Fourier para tiempo discreto

Dada la señal $x[nT]$ en el tiempo discreto, la transformada de Fourier en tiempo discreto esta definida por

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[nT] e^{j\Omega n}.$$

Se observa que es la contraparte natural en el tiempo discreto de la ecuación (2.17).

En el Cuadro 2.1 se anuncian algunas propiedades utilizadas en este trabajo tanto en tiempo continuo como discreto [24].

2.3 Digitalización de una señal

La conversión de una señal analógica a una secuencia de números consiste en dos pasos básicos: muestreo y cuantización. El proceso de muestreo, si se realiza a una frecuencia lo suficientemente alta, no representa pérdida de información ya que la señal puede recuperarse mediante interpolación, como se verá más adelante. La cuantización convierte cada muestra a un valor de L bits e inevitablemente conlleva un error por redondeo. Sin embargo, en la práctica este error puede ser despreciado si se usa un número grande de bits para almacenar cada muestra [34]. Matemáticamente el muestreo se expresa como una señal de tiempo continuo $x(t)$ multiplicada por un tren de impulsos $p(t)$

$$\begin{aligned} x_{muestreada}(t) &= x(t)p(t) \\ &= x(t) \sum_{m=-\infty}^{\infty} \delta(t - mT_s). \end{aligned}$$

La mayor parte de las señales en el mundo real son analógicas. El procesamiento analógico de una señal usa circuitos también analógicos como resistores, condensadores, inductores y elementos activos como amplificadores operacionales, además de otros dispositivos no lineales.

El procesamiento digital de señales hace uso de computadoras de propósito específico que tienen tres elementos básicos; sumadores, multiplicadores y almacenamiento a memoria. El procesamiento digital consiste en una serie de cálculos donde no siempre es posible resolver el algoritmo en tiempo real. Si las entradas y salidas son señales analógicas es necesario usar convertidores analógico-digital *ADC* y digital a analógico *DAC*.

Bajo ciertas circunstancias un proceso digital tiene ventajas sobre uno analógico debido a que el digital puede ser usado para implementar diferentes versiones del sistema al cambiar el software en el procesador, esto le da flexibilidad y repetibilidad. En el caso analógico el sistema debe ser rediseñado cada vez que las especificaciones cambian, además, los dispositivos analógicos no siempre están disponibles en el mercado para el diseño y sufren de variaciones en su valor nominal debido a tolerancias en la fabricación, temperatura, humedad, fuente de polarización, entre otros [37].

2.3.1 Teorema de muestreo de Nyquist

En general, no deberíamos esperar que en ausencia de cualquier condición adicional, una señal pudiera ser especificada unívocamente por una secuencia de muestras igualmente espaciadas. En la Figura 2.2 se muestra un ejemplo de tres señales en tiempo continuo que tienen valores idénticos en múltiplos enteros de T , es decir

$$x_1(nT) = x_2(nT) = x_3(nT).$$

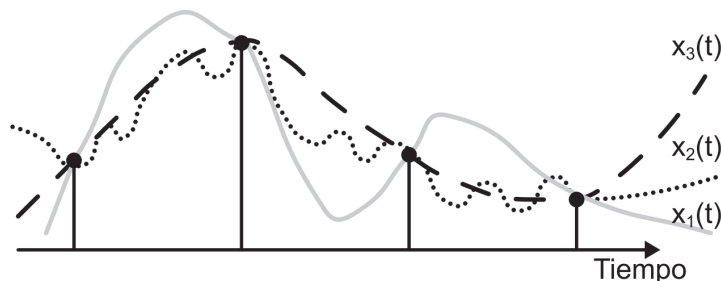


Figura 2.2 Tres señales diferentes caracterizadas por la misma secuencia discreta.

Es claro que existe un número infinito de señales que podrían ser descritas por la misma secuencia de muestras. Sin embargo, si la señal está limitada en banda y las muestras son tomadas lo suficientemente cercanas una de otra en relación con la frecuencia más alta presente en la señal, entonces la secuencia sí especifica unívocamente la señal y puede reconstruirse en el tiempo continuo perfectamente.

Matemáticamente el *teorema de muestreo de Nyquist* enuncia [24]

“Sea $x(t)$ una señal limitada en banda, donde su frecuencia máxima presente expresada en radianes es ω_M . Entonces $x(t)$ se determina unívocamente mediante sus muestras $x(nT)$, $n = 0, \pm 1, \pm 2, \dots$, si

$$\omega_s > 2\omega_M$$

donde $\omega_s = \frac{2\pi}{T_s}$ es la frecuencia de muestreo”. Cuando $\omega_s = 2\omega_M$, ω_s se conoce como *frecuencia de Nyquist*.

2.3.2 Submuestreo

Se llama submuestreo al proceso de reducir la frecuencia de muestreo de un sistema discreto. Para evitar pérdida de información la señal debe estar limitada en banda por lo que es usual que la señal de banda completa $\tilde{x}[n]$ se pase por un filtro anti-aliasing $h[n]$ que remueva las frecuencias superiores a $\frac{f_s}{2a}$ y luego se submuestra a una fracción $\frac{1}{a}$ de la frecuencia de muestreo original. En la Figura 2.3 se presenta un sub-muestreo con $a = 3$.

En el dominio del tiempo, la operación de submuestreo se expresa como

$$y[n] = x[Nn]$$

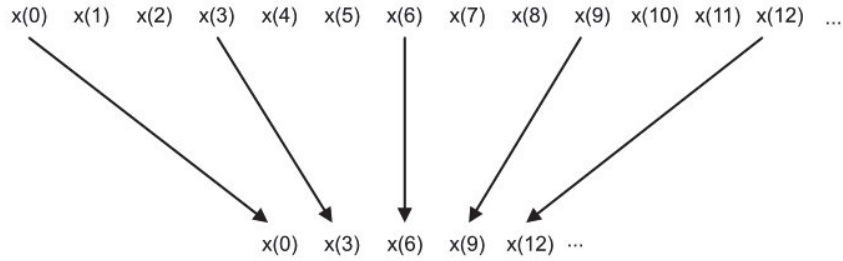


Figura 2.3 Sub-muestreo de una señal discreta.

y en el dominio de la frecuencia

$$Y(e^{j\Omega}) = \frac{1}{N} \sum_{n=0}^{N-1} X(e^{j(\frac{\Omega-2n\pi}{N})}) \quad (2.18)$$

donde $X(e^{j\Omega})$ y $Y(e^{j\Omega})$ son las transformadas de Fourier de la entrada y la salida respectivamente. De la ecuación (2.18) se puede ver que el submuestreo crea una repetición del espectro de la señal de entrada $N - 1$ veces, si estos espectros se traslapan uno sobre otro se vuelve imposible recuperar la señal original, esto se conoce como *aliasing* y conlleva pérdida de información [19]. Por ello es importante el filtrar la señal con un filtro antialiasing $h[n]$ antes del proceso de submuestreo.

2.3.3 Interpolación

Se llama interpolación al proceso de aumentar la frecuencia de muestreo de una señal discreta, para ello existen varias técnicas entre las más comunes están la interpolación líneal, interpolación polinómica, interpolación por promedio de vecinos cercanos, por *zero padding*, entre otras. La técnica por zero padding consiste en intercalar ceros entre las muestras de la señal a sobremuestrear y después filtrar la señal con un filtro de interpolación $g[n]$. Sobremuestrear una señal por una razón de N implica intercalar $N - 1$ ceros en las localidades entre muestras como lo muestra la Figura 2.4. De este modo aumenta la razón de muestreo.

En el dominio del tiempo el sobremuestreo se representa como

$$y[n] = \begin{cases} x[\frac{n}{N}] & : \quad n = 0, \pm N, \pm 2N, \dots \\ 0 & : \quad \text{otro } n \end{cases}$$

y en la frecuencia

$$Y(e^{j\Omega}) = X(e^{j\Omega N}).$$

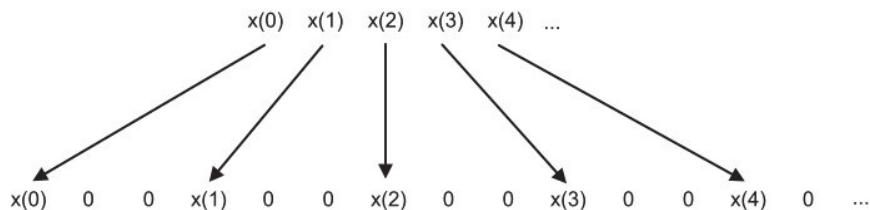


Figura 2.4 Sobre-muestreo de una señal discreta.

Lo que implica que el espectro de la señal de salida tiene un factor de compresión de N . Como la señal $x[n]$ es resultado de una digitalización, su espectro contiene N copias del espectro original; después del sobremuestreo todas las N copias del espectro original se empaquetan dentro del intervalo $[-\pi, \pi]$ de la señal de salida $Y(e^{j\Omega})$. La función del filtro $g[n]$ es filtrar sólo una de esas N copias del espectro, normalmente se usa un filtro pasa bajas con una frecuencia de corte igual al ancho de banda de la señal original para dejar la copia alrededor del origen $\omega = 0$.

2.4 Filtros digitales

En ingeniería electrónica, un filtro es un sistema que puede ser usado para modificar el espectro de una señal eléctrica para que cumpla con ciertas características deseadas. Los filtros tienen una gran variedad de aplicaciones como eliminar ruido en sistemas de comunicación, demodular señales, limitar en banda una señal antes de muestrearla, interpolar una señal muestreada, mejorar la calidad de un equipo de audio, entre otras [10]. En este trabajo no se profundizará en la teoría para el diseño de filtros digitales, en cambio se hará hincapié en sus características, forma de uso y relevancia.

La clasificación más elemental de los filtros es mediante su respuesta en frecuencia, pueden ser pasa bajas, pasa altas, pasa banda y rechaza banda.

2.4.1 Filtros de respuesta infinita al impulso (IIR)

Los filtros de respuesta infinita al impulso IIR son filtros recursivos caracterizados por una ecuación en diferencias como [10]

$$a_0y(n) + \dots + a_{M-1}y(n - M + 1) = b_0x(n) + \dots + b_{N-1}x(n - N + 1)$$

donde $a_0 = 1$. Como es una expresión válida para todo n , se tiene

$$y(n) = b_0x(n) + \dots + b_{N-1}x(n - N + 1) - a_1y(n - 1) \dots - a_{M-1}y(n - M + 1). \quad (2.19)$$

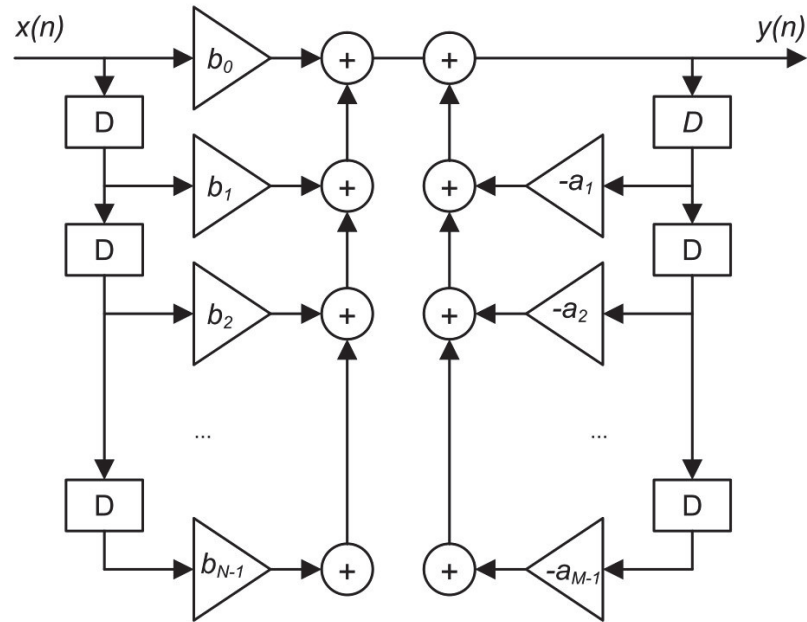


Figura 2.5 Estructura de un filtro IIR implementado en forma directa I.

En un filtro causal el valor de la salida no depende de valores futuros de la entrada. Un filtro que no sea causal no puede implementarse en tiempo real porque para computar el valor actual en la salida se requieren los valores futuros en la entrada. La ecuación en diferencias (2.19) puede representar tanto un filtro causal como uno no causal. Si la salida $y(n)$ se calcula para un valor creciente de n y los términos $x(n)$, $x(n-1)$, ..., $x(n-N+1)$, $y(n-1)$, ..., $y(n-M+1)$ son valores anteriores de las señales de salida y entrada, entonces la ecuación representa un filtro causal y por lo tanto es realizable. Sin embargo, si se quiere calcular $y(n-M+1)$, para un valor decreciente de n el filtro es no causal porque la salida depende de $x(n)$, ..., $x(n-N+1)$ que en este caso son entradas futuras. Los filtros no causales son útiles en aplicaciones donde la salida no necesita ser calculada en tiempo real o la variable n no representa al tiempo, como es el caso del procesamiento de imágenes.

Considerando el caso causal, es claro que la salida $y(n)$ depende de las salidas anteriores $y(n-2)$, ..., $y(n-M)$ y de todos los N valores de la señal de entrada $x(n)$, ..., $x(n-N+1)$ y puede ser implementado. Existen muchas estructuras para implementar los filtros IIR, la Figura 2.5 muestra la Forma directa I [22] donde los bloques D representan retardos unitarios. La implementación real de filtros IIR requiere que los coeficientes a_i y b_j se representen en un número finito de bits, la cuantización de estas cifras cambia el valor de los coeficientes y por lo tanto cambia la respuesta del filtro, incluso puede provocar inestabilidad [10].

2.4.2 Filtros de respuesta finita al impulso (FIR)

Los filtros de respuesta finita al impulso son muy populares en el procesamiento digital de señales debido a que sólo requieren una secuencia finita de la señal de entrada. Esto permite lograr especificaciones que no son posibles mediante filtros IIR, especialmente en la implementación de filtros causales de fase lineal. Además, los filtros FIR teóricamente siempre son estables. Dependiendo de la aplicación el orden de estos filtros varía entre 25 y 400 [22]. La Figura 2.6 muestra la estructura de estos filtros.

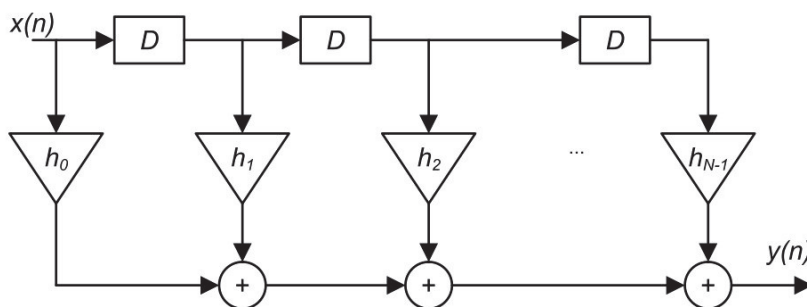


Figura 2.6 Estructura de un filtro FIR.

Para usar un filtro FIR se debe convolucionar los coeficientes del filtro $h[n]$ con la señal en tiempo discreto $x[n]$, teniendo la salida $y[n]$ definida como

$$y[n] = h[n] * x[n] = \sum_{j=0}^{N-1} h[j]x[n-j].$$

Al sintetizar un filtro FIR se desea que los coeficientes del filtro que están en el vector $h[n]$ aproximen a la salida $y[n]$ lo mejor posible a su respuesta ideal, una técnica común para hacer esto es el *método por ventanas*. Usualmente, no se puede procesar toda la secuencia que describe a la señal, ya sea por que no se tienen los recursos computacionales o porque se está implementando un sistema en tiempo real, por ello se toma una fracción de la secuencia total la cual se llamará *ventana*. La señal dentro de la ventana $x_w[n]$ puede expresarse como

$$x_w[n] = x[n]w[n]$$

donde $x[n]$ es la señal que se quiere analizar y $w[n]$ es un vector con coeficientes de ponderación, cualquier valor fuera de la ventana es cero. Aunque una ventana cuyos coeficientes sean unitarios parece ser la opción más simple, no necesariamente es la más popular. Si se cambian los valores de $w[n]$ por otros que no sean unitarios se modifica la respuesta del filtro, lo cual dependiendo de la aplicación puede llegar a ser deseable.

En el Capítulo 4 *Formadores de haz* se verá que un formador de haz se asemeja mucho a un filtro FIR, por lo que también se pueden utilizar ventanas para modificar

su rendimiento. En ese capítulo se profundizará la discusión respecto a las ventanas y como se comparan entre ellas.

2.4.3 Comparación de filtros IIR y FIR

Las principales diferencias entre estos filtros son [10]

- Si el orden de un filtro FIR está entre 25 y 400, su equivalente IIR tiene un orden de entre 5 y 20.
- Los filtros IIR pueden ser inestables.
- Los filtros IIR necesitan algunos valores anteriores de la salida $y[n]$ para su cómputo.
- Los filtros IIR necesitan sólo unos pocos valores de la entrada $x[n]$, los filtros FIR necesitan un número finito aunque más grande.
- Los filtros FIR pueden ser diseñados para tener una fase lineal.

2.5 Resumen

En este capítulo se explicó el concepto de señal, sus clasificaciones y propiedades más importantes para el entendimiento del problema a resolver, se repasa la representación de una señal en forma de exponencial compleja, se presenta una idea general de cómo funciona el análisis de Fourier y finalmente se explica qué son y cómo se usan los filtros digitales IIR y FIR, introduciendo al lector a los conceptos y terminología que en adelante se usarán.

Capítulo 3

Acústica

En este capítulo se mencionan algunas definiciones y características sobre el sonido, el oído, la voz y el ruido, las cuales en su conjunto ayudan a entender las propiedades de las señales implicadas en esta tesis.

Una onda de sonido es una fluctuación longitudinal de presión que se mueve en un medio elástico, se dice que es longitudinal porque el movimiento de la onda es en el mismo sentido en el que se propaga. El medio elástico en que esto ocurre puede ser gas, líquido o sólido, sin embargo, en la vida diaria el medio más común por el que se perciben sonidos es el aire [20].

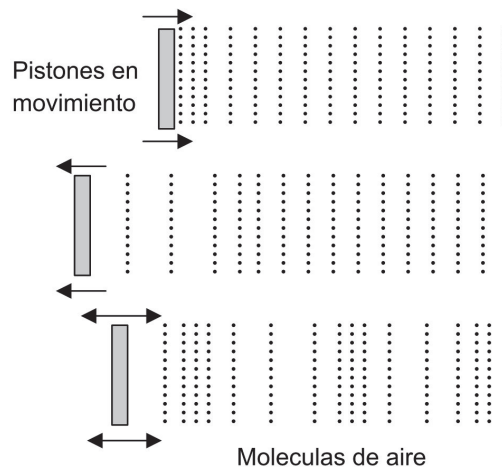


Figura 3.1 Propagación de perturbaciones mecánicas en el aire.

Todos los sonidos son producidos por vibraciones mecánicas; cuando un pistón, como en una bocina, mueve aire produce una zona en el aire adyacente donde la densidad y la presión son ligeramente mayores al resto del aire circundante [20]. Como se ve en la Figura 3.1, cuando dicho pistón se mueve de izquierda a derecha no puede desplazar todo el aire que tiene frente a él, en cambio presiona sólo una parte. Cuando se mueve

en dirección contraria el cuerpo succiona algo de aire, sin influir en toda la columna de aire sino expandiendo las moléculas más próximas [16].

En el párrafo anterior el término *vibración* se usa de forma general y no requiere que sea un movimiento cíclico, en cambio cuando se habla de una *oscilación* es un caso particular cuando la vibración es periódica. Éste se modela como un movimiento armónico simple como el descrito por un sistema masa-resorte. Las oscilaciones sinusoidales, también llamadas oscilaciones armónicas, son de gran importancia en el estudio de cualquier vibración periódica debido a que el análisis se simplifica sin perder validez, debido a que cualquier función periódica se puede representar como una suma de sinusoidales [14].

Cuando una onda se propaga en un medio como el aire, las moléculas se mueven hacia adelante y atrás conforme la onda pasa. Si se escribe una ecuación para describir el comportamiento de un volumen de aire pequeño perturbado por una onda de sonido que se propaga en la dirección x a una velocidad c se tiene

$$y = f(x - ct)$$

es decir, se define simplemente que la posición en el eje y de ese volumen diferencial es función de cierta posición inicial y del producto de la velocidad por el tiempo que dure el movimiento; si la onda es sinusoidal entonces

$$y = A \sin(k(x - ct))$$

donde k es el número de onda está dado por

$$k = \frac{2\pi}{\lambda} = \frac{\omega}{c}$$

donde λ es la longitud de onda.

3.1 Propagación del sonido

Para encontrar la forma en que se relaciona las propiedades físicas de un medio dado con la velocidad de propagación de la onda, primero considérese un volumen de aire de dimensiones lo suficientemente pequeñas para describir sólo los cambios locales de presión y densidad en una porción del medio, pero lo suficientemente grande para contener a un número muy grande de moléculas. También considérese que en un tubo se mueve un pistón el cual comprime el aire dentro de él y después se detiene. Para analizar el comportamiento del aire se pondrá el punto de referencia en el pistón mismo por lo que si el pistón se movió de izquierda a derecha a una velocidad c , el aire aparentará moverse de derecha a izquierda a una velocidad c . Cuando el aire encuentra la zona de mayor presión desacelera a $c - \Delta c$ [20]. Si se examina sólo un pequeño elemento del fluido como se ve en la Figura 3.2 la fuerza a la que se somete dicha rebanada de aire es

$$F = (P + \Delta P)S - PS = \Delta PS$$

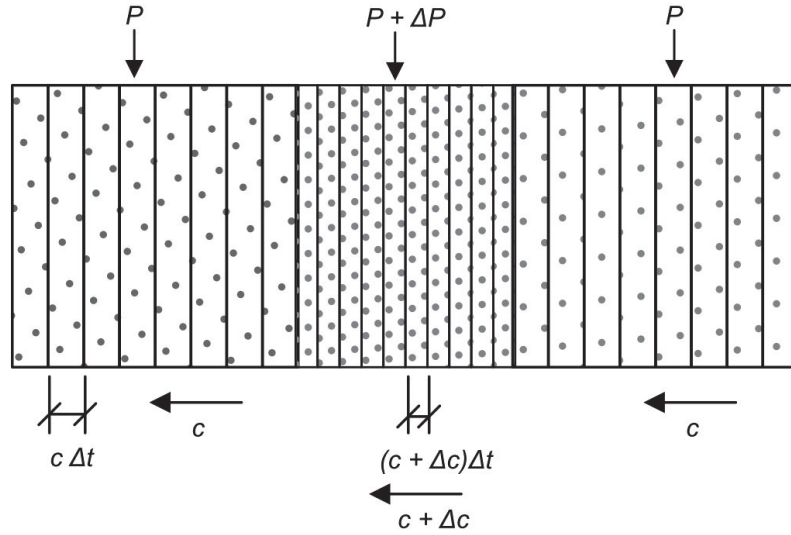


Figura 3.2 Región de alta presión moviéndose a través de un tubo [20].

donde S es el área transversal del tubo, P es la presión del medio que no está afectado por el pulso, ΔP es el aumento de presión debido al pulso, la longitud del elemento diferencial que se está analizando es $c\Delta t$, el volumen de dicho elemento resulta $cS\Delta t$ y su masa $\rho cS\Delta t$, donde ρ es la densidad del fluido fuera de la región afectada por el pulso. La aceleración que experimenta cuando se acerca a la región de alta presión es $-\frac{\Delta c}{\Delta t}$. Usando la segunda ley de Newton para relacionar la fuerza y la aceleración

$$F = ma$$

se puede escribir

$$\Delta P S = (\rho S c \Delta t) \left(-\frac{\Delta c}{\Delta t} \right)$$

acomodando términos

$$\rho c = -\frac{\Delta P}{\Delta c}$$

si se multiplican ambos lados por c

$$\rho c^2 = -\frac{\Delta P}{\frac{\Delta c}{c}}.$$

El volumen del elemento antes de ser comprimido era de $V = Sc\Delta t$ y pasó a ser $\Delta V = S\Delta c\Delta t$. El cambio de volumen dividido entre el volumen es

$$\frac{\Delta V}{V} = \frac{S\Delta c\Delta t}{Sc\Delta t} = \frac{\Delta c}{c}$$

entonces

$$\rho c^2 = -\frac{\Delta P}{\frac{\Delta V}{V}} \quad (3.1)$$

De esta forma se ha relacionado la velocidad de la onda con propiedades físicas del medio. La parte derecha de la ecuación (3.1) es una cantidad medible llamada *módulo de compresibilidad*. Entonces

$$c = \sqrt{\frac{B}{\rho}}$$

donde c es la velocidad del sonido dada en $\frac{m}{s}$, B es el módulo de compresibilidad del medio dado en Pa y ρ es la densidad del medio en $\frac{kg}{m^3}$. El módulo de compresibilidad puede ser medido o calculado con una ecuación de estado, la cual relaciona el comportamiento de la presión, densidad y temperatura de un gas [20]. Es evidente que la velocidad del sonido cambia con las características termodinámicas del medio.

En el Cuadro 3.1 se muestran diferentes velocidades de propagación de ondas de sonido en diferentes medios.

Cuadro 3.1 Velocidad del sonido en diferentes medios de propagación [16], [20].

Material	Temperatura °C	Densidad $\frac{kg}{m^3}$	Velocidad del sonido $\frac{m}{s}$
Argón	0	1.783	319
Helio	0	0.178	965
Oxígeno	0	1.429	316
Nitrógeno	0	1.251	334
Hidrógeno	0	0.090	1284
Aire	0	1.293	331
Aire	20	1.21	343
Agua	20	998	1483
Mercurio	20	13500	1451
Alcohol etílico	20	790	1159
Diesel	20	800	1250

En la práctica es común considerar la velocidad del sonido como $340 \frac{m}{s}$.

La *intensidad* de una onda de sonido se define como la potencia acústica que pasa a través de cierta área en la dirección normal a la superficie, es decir

$$I(\theta) = \frac{W \cos \theta}{S}$$

donde $I(\theta)$ es la intensidad que pasa a través del área en dirección a su normal dada en $\frac{W}{m^2}$, W es la potencia de sonido dada en W , S es el área de medición en m^2 y θ es

el ángulo entre la dirección de propagación y la normal a la superficie de medición [20]. La intensidad máxima se obtiene cuando $\theta = 0^\circ$, la ecuación se simplifica a

$$I = \frac{W}{S}.$$

Como el intervalo de intensidad es muy grande, comúnmente se expresa en *niveles*. Un nivel es una fracción expresada como 10 veces el logaritmo del cociente de dos números, es decir

$$Nivel = 10 \log \left(\frac{\text{Número de interés}}{\text{Número de referencia}} \right).$$

El nivel de la intensidad de sonido se calcula como

$$L_I = 10 \log \frac{I}{I_0}, \text{ en } dB$$

donde $I_0 = 10^{12} \frac{W}{m^2}$.

El indicador más popular para medir el sonido es la presión de sonido, este indicador se relaciona bien con la percepción humana de volumen y es fácil de medir con aparatos relativamente económicos. La referencia para medir el nivel de presión de sonido se situó en el umbral de audición de una persona joven a 1000 Hz . Cuando la presión de sonido es igual a esta referencia el nivel resultante es cero dB . El nivel de presión de sonido se define como

$$L_P = 10 \log \frac{P^2}{P_{ref}^2}, \text{ en } dB$$

donde P es la raíz media cuadrática de la presión del sonido dada en Pa y P_{ref} es la presión de referencia y tiene un valor fijo de $20 \mu Pa$. En adelante, cuando se hable de un nivel de sonido dado en dB se entenderá que se refiere al nivel de presión de sonido.

Al propagarse en un medio, la onda de sonido va perdiendo intensidad debido a la disipación de energía en un medio amortiguado, esta energía disipada se convierte principalmente en calor. Esto se debe en parte a que la temperatura del aire varía con la presión del sonido y como no existe un material sin conductividad térmica, se produce un intercambio de calor entre los elementos volumétricos adyacentes. Otro factor que provoca la atenuación del sonido es la deformación que sufre el medio; cualquier compresión de un elemento volumétrico en un fluido implica una pérdida por fricción debido a la viscosidad del medio. Además, para gases cuyas moléculas no son monoatómicas se presenta un fenómeno llamado *relajación térmica*, que se refiere al hecho de que las moléculas del gas pueden almacenar cierta energía al vibrar, trasladarse o torcerse de modo que aunque una fuerza externa actúe sobre el gas, toma cierto tiempo que las moléculas lleguen a un nuevo equilibrio térmico, lo que causa una especie de conductividad térmica que depende de la frecuencia, la cual se asocia con pérdida de energía. La magnitud en que ese proceso de relajación térmica contribuye a la atenuación de la

onda varía dependiendo del número y tipo de átomos en la molécula del gas. Otro factor que modifica fuertemente este valor es el agua disuelta, por esta razón la atenuación del sonido presenta una fuerte dependencia a la humedad [16]. En la Figura 3.3 se observa una gráfica de cómo se atenúa una onda de sonido en aire cuando la humedad relativa varía.

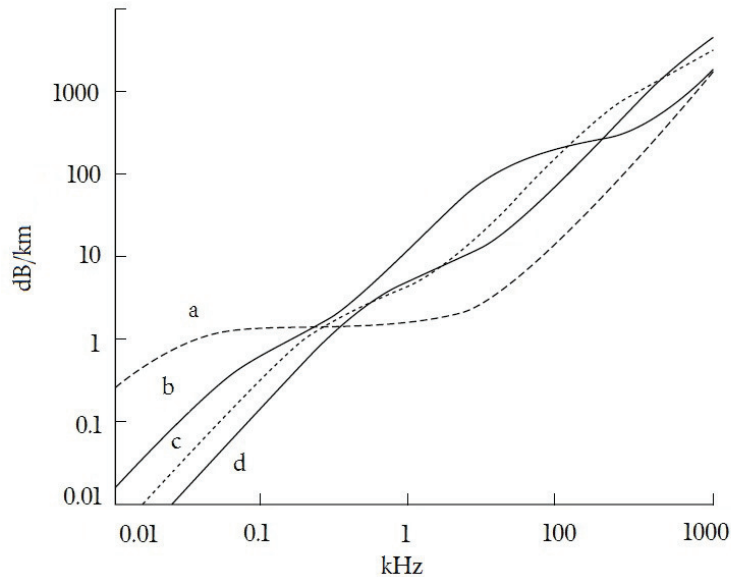


Figura 3.3 Atenuación del sonido en aire a 20 C. Humedad relativa: a) 0 %, b) 10 %, c) 40 %, d) 100 % [16].

Como se observa en la gráfica, las señales dispersan más energía a frecuencias altas, además, la pendiente de las curvas con una mayor humedad es mayor por lo que al aumentar la humedad relativa se enfatiza la dependencia a la frecuencia.

3.2 Acústica de recintos

La acústica de recintos analiza la propagación del sonido en un espacio cerrado donde el medio conductor del sonido está delimitado por paredes, techo y piso. Estas fronteras usualmente reflejan una porción de la energía sonora incidente y otra porción es absorbida, es decir, parte de la energía de la señal original se refleja dentro de la sala y otra parte se extrae del espacio confinado, ya sea en forma de calor o porque se transmite fuera del cuarto.

El sonido dentro de un cuarto cerrado se ve fuertemente afectado por la forma y las características de las superficies en las fronteras. Cuando un cuarto está cerrado mediante superficies reflejantes, ocurrirán inevitablemente múltiples reflejos de la onda incidente y se establecerá un campo reverberante. Entonces, en cualquier punto de una

sala la presión de sonido quedará en función de la energía del campo directo y del campo reverberante [17].

3.2.1 Reflejos y dispersión

Si una onda plana choca con una pared uniforme de longitud infinita, la onda reflejada tendrá una amplitud y una fase distinta a la onda incidente [17].

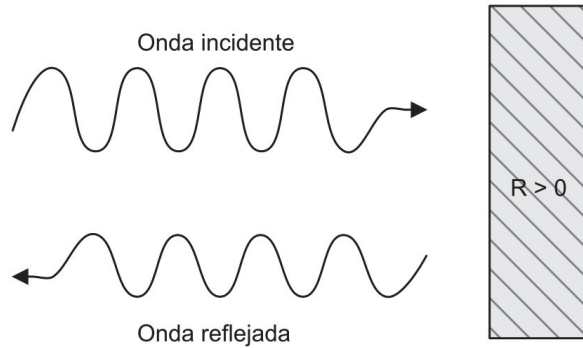


Figura 3.4 Reflexión de una onda de sonido que incide perpendicularmente en una pared.

Los cambios en la amplitud y la fase que tiene lugar durante la reflexión de la onda se expresan en la cantidad compleja

$$R = |R|e^{i\chi}$$

la cual es una propiedad de la pared. Su fase y valor absoluto dependen de la frecuencia y el ángulo de arribo de la señal. La intensidad de la onda reflejada estará atenuada a un valor de $|R|^2$ por lo que la porción $1 - |R|^2$ de la energía incidente se pierde en la reflexión. Esta cantidad se conoce como *coeficiente de absorción*.

$$\alpha = 1 - |R|^2.$$

Para una pared con cero reflectividad ($R = 0$) el coeficiente de absorción tiene su máximo valor 1 y la pared se dice que es totalmente absorbente. Cuando la pared es totalmente reflectiva y la fase no cambia ($R = 1$, $\chi = 0$) la pared es *rígida*; cuando la fase cambia 180° ($R = -1$, $\chi = \pi$) la pared se denomina *suave*.

En lo que concierne a la acústica de recintos, las características acústicas de una pared están definidas por el factor de reflectividad. Otra cantidad que relaciona incluso mejor el comportamiento físico de este fenómeno se basa en la velocidad de la partícula normal a la superficie que se genera para una presión de sonido en una pared dada. Esta cantidad se denomina *impedancia de la pared* y se define como

$$Z = \frac{p}{v_n}$$

donde v_n denota la velocidad de la partícula normal a la pared y p es la presión de sonido. Si la frecuencia de la señal que incide a la pared cambia, usualmente la impedancia también cambia [17].

Estas dos cantidades se relacionan de la siguiente forma

$$Z = \rho_0 c \frac{1 + R}{1 - R}$$

y

$$R = \frac{Z - \rho_0 c}{Z + \rho_0 c}$$

donde ρ_0 es la densidad del aire y c es la velocidad del sonido.

Como se ha dicho la impedancia de una pared cambia con la frecuencia de la señal que incida, además, no sólo el material de la pared sino su forma juega un papel decisivo en la reflexión de estas ondas.

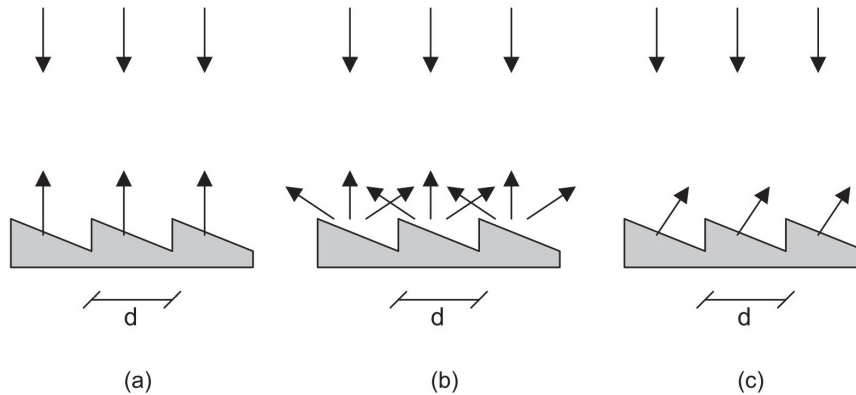


Figura 3.5 Reflexión de una onda de sonido que incide en una pared con irregularidades: (a) $d \ll \lambda$, (b) $d \sim \lambda$, (c) $d \gg \lambda$ [17].

Comúnmente las paredes no son completamente planas, si las irregularidades que presentan en su superficie son pequeñas comparadas con la longitud de onda de la señal, éstas no afectan la reflexión esperada de la pared como lo muestra la Figura 3.5 a, en el caso opuesto cuando las irregularidades son grandes comparadas con la longitud de onda, cada irregularidad debe tratarse como una sección de pared plana o curva, Figura 3.5 c. Existe un punto medio donde una fracción considerable de la onda incidente es dispersada en todas direcciones, en este caso se habla de una *pared difusamente reflejante*, como en la Figura 3.5 b [17].

3.2.2 Reverberación y patrones de interferencia

Cuando la señal llega al mismo punto después de varios reflejos desde diferentes puntos y superficies a través de múltiples trayectorias, se dice que causa reverberación o que forma un campo reverberante [34]. Para medir el grado de reverberación de un cuarto se usa el tiempo de reverberación T_{60} el cual está definido como el tiempo en que tarda la intensidad de sonido en caer a -60 dB de su valor original. Este tiempo puede estar entre $200 - 300$ ms para una oficina promedio y hasta varios segundos para una iglesia [11]. En general se desea que el tiempo de reverberación sea menor a un segundo para una mayor inteligibilidad, en contraste con eventos musicales donde es preferible tener un T_{60} entre 1 y 2 segundos [20].

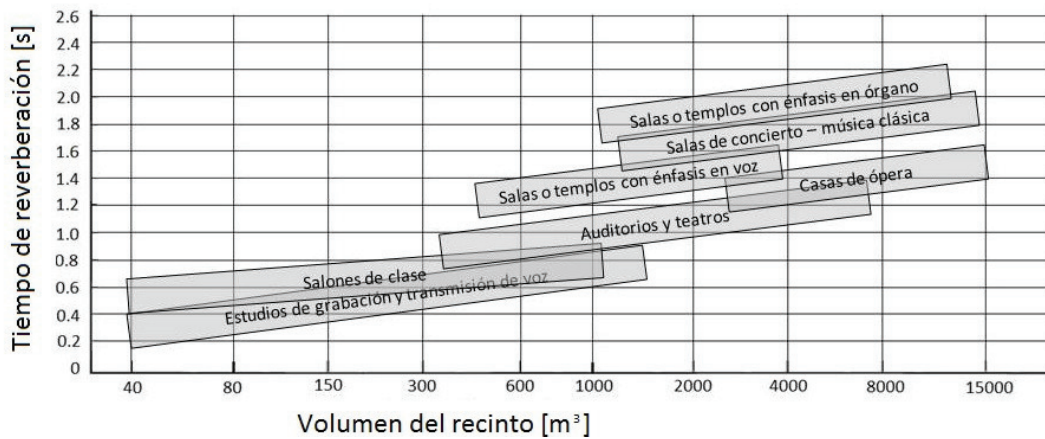


Figura 3.6 Tiempos de reverberación de distintos tipos de recinto [20].

La Figura 3.6 muestra los tiempos de reverberación recomendados para ciertas circunstancias [20]. Como es de suponerse, cuando se quiere capturar una señal de voz, como en el caso de estudios de grabación y auditorios, lo mejor es tener un T_{60} mínimo.

En un campo reverberante la energía del sonido no está distribuida uniformemente en todo el espacio, sino que se distribuye formando patrones de interferencia cerca de las paredes o cualquier cuerpo acústicamente reflejante. Esto se debe a que aunque el promedio de la energía de las señales incidentes en todas las direcciones es el mismo, las fases de estas señales no son del todo aleatorias; esta falta de aleatoriedad es lo que provoca que las señales formen un patrón de interferencia. Si las señales que provocan la reverberación son estacionarias, el patrón de radiación también será estacionario [36]. Un patrón de interferencia sinusoidal se forma cuando una señal incide en ciertos ángulos en un plano reflejante. El patrón es el resultado de la interacción de la señal original y la reflejada, una explicación gráfica de esto se presenta en la Figura 3.7. Si se considera una señal sinusoidal que incide en un plano reflejante de manera normal, el patrón de interferencia tendrá una forma sinusoidal con la misma longitud de onda de la señal original; si ésta incidiera de manera oblicua se formaría de igual forma un patrón de

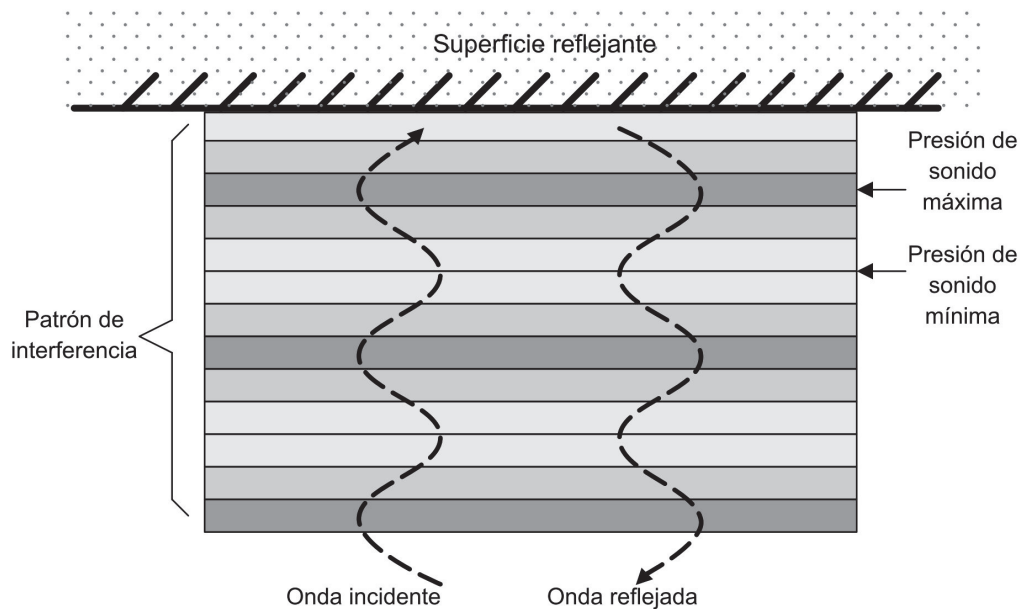


Figura 3.7 Patrón de interferencia que provoca una onda sinusoidal en una superficie reflejante.

interferencia sinusoidal pero con una longitud de onda mayor. En un ambiente reverberante las señales incidirían en las paredes en todas direcciones por lo que en realidad se forman muchos patrones de interferencia simultáneamente [36].

3.3 Percepción humana

El oído y la voz son parte de las capacidades fundamentales del ser humano y forman parte importante de nuestra vida diaria. El oído y la voz conforman el sistema primordial de comunicación entre individuos, por eso es importante mantener los mensajes de voz tan claros como sea posible, logrando tener inteligibilidad al mismo tiempo que la señal no pierda su tono natural. La audiología es la rama de las ciencias clínicas que estudia el oído humano; mientras la acústica se ha mantenido basada en la física, la audiología es una ciencia médica, como consecuencia el conocimiento que reúne la acústica no siempre es de primordial importancia en el campo de la audiología. No obstante, es necesario conocer los fundamentos de la acústica para entender nuevas aplicaciones y optimizar soluciones [20].

3.3.1 Oído humano

Generalmente cuando se habla de sonido, se hace en el contexto de las vibraciones que el oído humano puede escuchar, este intervalo en la realidad no está tan claramente delimitado porque varía de persona a persona, sin embargo, existe un consenso en considerar que va desde los 20 Hz hasta los 20,000 Hz. Los sonidos que están por arriba de la frecuencia máxima se consideran ultrasónicos y los que están por debajo de la frecuencia mínima se denominan infrasónicos [14]. Dentro de ese intervalo de frecuencias audibles el oído humano tiene una gran sensibilidad. El sonido mínimo que puede oír una persona promedio es alrededor de 0 dB de nivel de presión de sonido a 1000 Hz lo cual corresponde a una presión de sonido de 20 μPa , si se compara con la presión atmosférica de 101,000 Pa es claro que el oído puede responder a cambios diminutos de presión, incluso el umbral del dolor situado en 120 dB equivale solo a 20 Pa [20].

La respuesta del oído a la magnitud de la señal no es lineal, lo que permite tener un intervalo dinámico de alrededor de 120 dB. Además, la percepción de intensidad varía de forma compleja respecto a la presión del sonido y a la frecuencia [18].

En acústica y audiología, el *volumen* es la percepción humana de la magnitud de un sonido. Los primeros intentos por cuantificar el volumen fueron hechos por los músicos, así surgieron los términos “muy alto”, “alto”, “moderadamente alto”, “bajo” y “muy bajo” dándoles los símbolos *ff*, *f*, *mf*, *p* y *pp* respectivamente por su nombre en italiano, pero carecían de precisión para propósitos científicos. Medir el volumen no se puede hacer de forma fácil y la percepción del volumen puede variar de individuo a individuo e incluso del estado fisiológico o psicológico del individuo en un momento dado. Por ello, la medida del volumen es resultado de tendencias estadísticas, más que de una teoría analítica.

En las décadas de 1920 y 1930 científicos en los Laboratorios Bell hicieron pruebas para medir el volumen. El procedimiento era que el sujeto comparara el volumen de un tono puro, a una frecuencia y amplitud dada, con el de una referencia fija a 1000 Hz. Estos tonos eran presentados en audífonos por un segundo cada uno intercalados por un segundo y medio de silencio, luego el sujeto debía decidir cuál tono tenía mayor volumen o si ambos presentaban el mismo volumen. Las gráficas de estos experimentos se conocen como las *curvas de Fletcher-Munson*. Diferentes variaciones del mismo experimento se han implementado desde entonces con resultados similares [20].

En la Figura 3.8 se muestra el resultado de estos experimentos la cual consiste en curvas de tonos puros que se perciben como si tuvieran la misma intensidad, donde la línea segmentada es el umbral mínimo de audición [18] y un fonio es una unidad que compensa la dependencia de la frecuencia en el oído humano la cual en 1000 Hz es igual a 1 dB de nivel de presión de sonido. La gráfica muestra que el oído es significativamente menos sensible a sonidos de baja frecuencia.

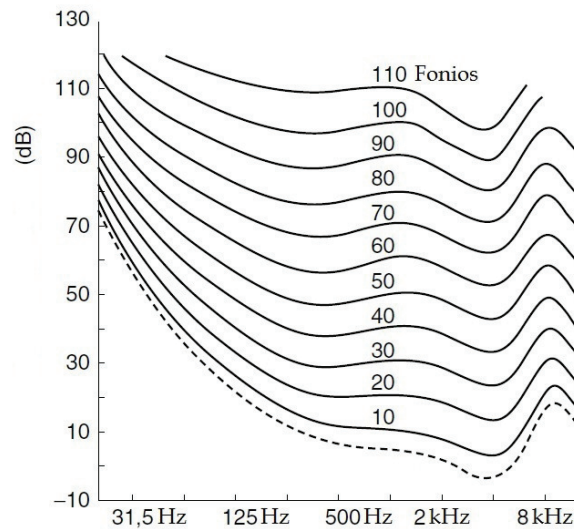


Figura 3.8 Contornos de nivel para una intensidad igual en tonos puros en todo el espectro audible [20].

3.3.2 Voz humana

La capacidad de comunicación oral es una de las características primordiales del ser humano, en ella no sólo existe información lingüística sino también el tono y emociones del locutor. La voz es una señal de banda ancha con frecuencias que varían desde los 100 Hz a los 8000 Hz. Para sonidos vibrantes como “r” o “v”, muy poca potencia está presente arriba de los 4 kHz, después de esta frecuencia la potencia espectral decae 6 dB por octava. Sin embargo para sonidos no-vibrantes como “t” o “s” el espectro en frecuencia resulta ser más plano y aún tiene una energía apreciable por encima de los 8 kHz. Para que la voz sea inteligible es suficiente con las componentes entre 300 Hz y 3400 Hz por lo que se tomó como estándar en la telefonía [11]. Para estas frecuencias una tasa de muestreo de 8 kHz es suficiente para tener una calidad aceptable; sin embargo, como se explicará en el siguiente capítulo, para implementar un formador de haz es conveniente sobrepasar varias veces esta frecuencia.

Por otro lado, la voz es una señal no-estacionaria con componentes en tiempo y en frecuencia cambiando constantemente. Bajo algunas circunstancias se puede considerar una señal estacionaria (sobre todo en vocales) pero únicamente en un periodo de tiempo muy corto del orden de 20-30 ms, otras veces se asemeja al ruido (consonantes fricativas como “s”) o a impulsos (consonantes oclusivas como “p”). Además, la voz es una señal intermitente debido a los espacios de silencio entre las palabras, en una conversación cotidiana más del 50 % de la señal es silencio [11].

La Figura 3.9 muestra en dónde se encuentra el espectro de voz en relación a la música y el umbral de percepción humana. Un parámetro importante de las señales de voz es la frecuencia fundamental o *altura* (*pitch* en inglés). La distribución de la altura

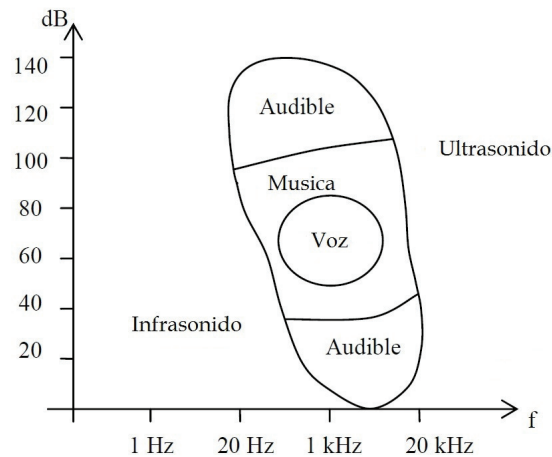


Figura 3.9 Comparación de intervalos de frecuencia contra intensidad de sonido para el espectro audible, música y voz [7].

graficada contra la frecuencia en una escala logarítmica se asemeja a una distribución normal. El promedio y la desviación estándar para una voz masculina son 125 y 20.5 Hz respectivamente; para una voz femenina es el doble de estos valores [18].

3.4 Micrófonos

Los micrófonos son dispositivos electroacústicos que convierten energía acústica a energía eléctrica. Todos los micrófonos tienen un diafragma o algún tipo de superficie móvil que se mueve al paso del sonido. Para medir el rendimiento de un micrófono se toman en consideración principalmente tres aspectos, uno de ellos es la sensibilidad. Este parámetro se define como el cociente en circuito abierto del voltaje y la presión de sonido aplicada, está dado en $\frac{V}{Pa}$ o en $\frac{mV}{Pa}$. Otro aspecto importante es la dependencia a la frecuencia de la sensibilidad la cual se asocia con distorsiones no lineales. Finalmente, otra característica útil para medir el rendimiento de un micrófono es su respuesta espacial o directividad [16]. Por la respuesta espacial se clasifican en omnidireccionales, que su respuesta es igual a señales que inciden en cualquier dirección, bidireccionales los cuales responden de manera máxima a señales que incidan con 180° de separación y responden de manera mínima a señales separadas a 90° de las direcciones máximas y en unidireccionales, que capturan señales de manera máxima sólo de una dirección [2].

Por el método de transducción se dividen en [2]

- Micrófono de carbón: Fue uno de los primeros tipo de transductor usado para micrófono, tiene una respuesta en frecuencia muy limitada, añade mucho ruido, alta distorsión y requiere una fuente de polarización de alto voltaje.

- **Micrófono cerámico y de cristal:** Estos dos tipos de micrófonos tienen un funcionamiento similar y alguna vez fueron populares por ser de bajo costo y por tener una alta impedancia de salida, lo que permitía conectarlos directamente a un tubo de vacío. Este tipo de micrófonos emplea materiales piezoeléctricos para producir un voltaje cuando el micrófono se somete a cierta presión. Algunos cristales que presentan características piezoeléctricas son el tartrato mixto de potasio y sodio (también conocido como sal de la Rochelle), titanato de bario y el cuarzo. Los materiales cerámicos no tienen características piezoeléctricas en su estado original pero se les pueden añadir mediante un proceso de polarización.
- **Micrófono dinámico:** Los micrófonos dinámicos usan un pequeño diafragma y una bobina que se mueven en un campo magnético permanente. El sonido que incide en el diafragma mueve la bobina dentro del campo magnético, generando un voltaje proporcional a la presión de sonido en la superficie del diafragma.
- **Micrófono condensador:** En un micrófono condensador la presión de sonido mueve un diafragma que forma parte de un condensador de placas paralelas, este cambio causa variaciones de una señal eléctrica en función de las señales acústicas. La cabeza del micrófono consiste en un condensador de unos 40-50 pF cuya separación entre placas es del orden de 25 μm . Los micrófonos condensador en general son pequeños y de poca masa, poseen una buena respuesta en frecuencia, son capaces de medir muy altos niveles de presión de sonido y necesitan polarizarse con voltajes relativamente altos.
- **Micrófono electret:** Un micrófono electret es un micrófono condensador cuya cabeza está permanentemente cargada, de tal forma que no necesita una fuente de alto voltaje para ser polarizado. Este tipo de micrófonos simplifica la polarización necesaria para usar los micrófonos condensador pero sufre de poca estabilidad y pérdida de carga sobre todo cuando no se almacenan apropiadamente.

3.5 Ruido y distorsión

El ruido puede definirse como una señal indeseada que interfiere con la medición de alguna otra señal de interés. Debe entenderse que casi cualquier señal en sí misma lleva información, el que tenga calidad de ruido simplemente es debido a que no es información de interés en cierta aplicación. Las fuentes de ruido son muy variadas, desde el ruido inherente a la conducción de electrones, ruido acústico por vibraciones, vehículos, viento, voces, ventiladores, ruido electromagnético, entre otros. Por otro lado, la distorsión es un término que se usa para referirse a los cambios sistemáticos que sufre una señal debido a las no idealidades del sistema de comunicación, reverberación, eco e información perdida en la transmisión. El ruido y la distorsión son los factores más importantes que limitan la capacidad de los sistemas de telecomunicación de datos y la

exactitud de los sistemas de medición, es por eso que modelar y remover estos efectos ha sido tema de gran interés en el procesamiento de señales [34].

Dependiendo de su fuente se puede clasificar al ruido de la siguiente forma [34]

- Ruido acústico: es el que emana de fuentes que se mueven, vibran o chocan y es el ruido más común en ambientes cotidianos. Se genera por carros, sistemas de aire acondicionado, ventiladores, gente hablando, viento, lluvia, etcétera.
- Ruido térmico y de disparo: es el ruido que es generado por el movimiento aleatorio de partículas energizadas por un proceso termodinámico en un conductor eléctrico, es intrínseco a todos los conductores y se presenta sin una diferencia de potencial eléctrico. El ruido de disparo consiste en fluctuaciones aleatorias de la corriente eléctrica en un conductor, es intrínseco al flujo de corriente y es causado porque la corriente se forma por portadores con cargas discretas con fluctuaciones y tiempos de llegada aleatorios.
- Ruido electromagnético: es el que está presente en todas las frecuencias y en particular en el intervalo de radiofrecuencia donde están contenidas señales de telecomunicación. Todos los dispositivos eléctricos como televisores y radios generan ruido electromagnético.
- Ruido electrostático: es el ruido que se genera por la presencia de un voltaje con o sin corriente eléctrica, su fuente más común son las lámparas fluorescentes.
- Ruido de procesamiento: es el resultado de la conversión analógico-digital que sufre una señal, principalmente se debe al ruido de cuantización (redondeo de un número real a un entero) y a la pérdida de información en los paquetes de datos.

En esta clasificación el tipo de ruido que más compete a esta tesis es ruido acústico. Debido al interés de implementar sistemas de reducción de ruido en espacios cerrados como oficinas y automóviles, se han hecho exhaustivos estudios sobre las características del ruido en este tipo de locaciones.

Por otro lado, dependiendo de su espectro en frecuencia o características temporales se puede clasificar como [34]:

- Ruido blanco: es ruido aleatorio puro no correlacionado que teóricamente contiene la misma potencia en todas las frecuencias contenidas en el intervalo $(-\infty, \infty)$. Frecuentemente se modela el ruido dentro de un sistema a partir de ruido blanco porque se considera ruido puro y sirve como un modelo simple. Tener una potencia igual en todas las frecuencias posibles implicaría que posee una potencia infinita, por lo que el ruido blanco es solo un concepto teórico. La propiedad de ser no correlacionado implica que la autocorrelación del ruido blanco con media cero en el tiempo continuo con una varianza de σ^2 es una función delta, lo que representa su no periodicidad

$$r_{NN}(\tau) = \mathcal{E}[N(t)N(t + \tau)] = \sigma^2\delta(\tau)$$

Por lo tanto, en una secuencia de datos que representen ruido blanco, cada porción de la secuencia es completamente distinta a cualquier otra porción.

- Ruido blanco de banda limitada: es ruido no correlacionado que contiene la misma potencia en todas las frecuencias contenidas en cierto intervalo finito denominado ancho de banda, usualmente el ancho de banda de este ruido incluye el ancho de banda del dispositivo o señal de interés. Por ejemplo, para un sistema de audio con ancho de banda de 10 kHz, cualquier ruido acústico con respuesta plana con un ancho de banda igual o mayor a 10 kHz se considera ruido blanco [34]. Además, una señal de ruido en el tiempo discreto es forzosamente limitada en banda debido que solo se pueden reconstruir frecuencias hasta un medio de la frecuencia de muestreo.
- Ruido de banda angosta: es ruido cuya energía se concentra en un intervalo pequeño de frecuencias, como el ruido de 60 Hz que proviene del suministro eléctrico.
- Ruido de color: es cualquier ruido de banda ancha que no tenga una potencia uniforme en las frecuencias de su ancho de banda, como el ruido rosa, cuya potencia es inversamente proporcional a la frecuencia.
- Ruido de impulsos: es ruido que consiste en pulsos de corta duración con amplitud aleatoria.

3.6 Resumen

En este capítulo se ha mostrado de manera simple cómo se produce una onda de sonido y cómo es afectada por las condiciones del medio en que se propaga. También se comentaron algunas de las características más importantes del oído y la voz humana. Además, se desarrollaron algunos puntos importantes sobre el ruido y la distorsión haciendo énfasis en el ruido y la distorsión acústicas en recintos cerrados.

Capítulo 4

Tipos de formadores de haz

En este capítulo se presenta la teoría del formador de haz el cual es el proceso que permite filtrar una señal de voz en función de su localización espacial.

La forma más adecuada para capturar la información espacial de una señal audible es con un conjunto de micrófonos localizados en posiciones conocidas, una vez capturada la información espacial es posible separar diferentes señales que inciden en el arreglo desde diferentes direcciones como una combinación lineal de las señales capturadas por cada sensor. En el procesamiento de arreglos de transductores se han desarrollado técnicas para procesar información de manera espacio-temporal, esto ha dado lugar al *formador de haz*.

El formador de haz tiene una historia larga y ha sido usado en el desarrollo de muchas tecnologías como radar, sonar, sismología, comunicaciones, entre otras. Tiene una amplia variedad de aplicaciones como para detectar la presencia de una señal, estimar la dirección de arribo de una señal, mejorar una señal corrompida por ruido y dereverberación [3].

Los problemas que se tienen en un arreglo de micrófonos parecen ser fácilmente resueltos con ayuda de tecnología originalmente desarrollada para telecomunicaciones, sin embargo esto puede ser engañoso. Algunos aspectos que deben tomarse en cuenta en el procesamiento de señales de arreglos de micrófonos son [3]

- La voz es una señal de banda ancha
- El ruido puede tener las mismas características espectrales que la señal deseada
- La reverberación suele ser alta
- La voz es altamente no-estacionaria
- El número de sensores normalmente no es grande
- El oído humano es muy sensible y puede notar pequeñas filtraciones de ruido con facilidad.

4.1 Procesamiento de arreglos de micrófonos

Un arreglo de micrófonos consiste en un conjunto de micrófonos posicionados de tal forma que la información temporal y espacial es capturada adecuadamente. Entonces, el objetivo principal del procesamiento de las señales de un arreglo de micrófonos es la estimación de algunos parámetros característicos o la extracción de alguna señal de interés usando la información espacio-temporal a la salida de los micrófonos [3].

Normalmente se asume que todos los micrófonos tienen las mismas características y son omnidireccionales (o isotrópicos), es decir que la respuesta de cada micrófono a una señal incidente no depende de su dirección de arribo. De acuerdo con la localización relativa entre micrófonos, los arreglos se pueden dividir en tres clases [19]

- Arreglos lineales o unidimensionales
- Arreglos planares o bidimensionales
- Arreglos volumétricos o tridimensionales.

Cada uno de éstas se divide a su vez en dos categorías [19]

- Espaciados de forma regular, incluyendo uniforme o no uniforme.
- Espaciados de manera irregular o aleatoria.

Dependiendo de la aplicación, la geometría del arreglo juega un papel importante en la formulación de los algoritmos de procesamiento.

Para observar cómo se presenta la información espacial en las señales de un arreglo de micrófonos considérese una onda de sonido con frente de onda plano con una frecuencia f propagándose en la dirección del eje z .

En el plano definido por $z = cte$, la fase de la señal se puede expresar como

$$\Phi(t, z) = 2\pi ft - kz$$

donde t es el tiempo y k es el número de onda definido como

$$k = \frac{\omega}{c} = \frac{2\pi}{\lambda}$$

donde ω es la frecuencia angular, c es la velocidad de propagación de la señal y λ es la longitud de onda. La interpretación del número de onda k es que mientras la onda cubre la distancia z en la dirección de propagación, la fase de la señal cambia en un valor de kz radianes, entonces, también puede referirse a k como la “frecuencia espacial”, ya que el número de onda de una señal que se propaga en un espacio tridimensional posee información tanto temporal como espacial. A diferencia de ω , la cual es unidimensional, la frecuencia espacial k es tridimensional y su dirección es igual a la dirección de propagación de la señal pero en sentido opuesto. Al vector de tres elementos

que contiene la fase de la señal se le denominará K . Para el caso general donde las señales inciden en el arreglo de un ángulo de elevación θ y un ángulo azimutal ϕ , como se ve en la Figura 4.1, el vector K está dado por [19]

$$K = \begin{bmatrix} k_x \\ k_y \\ k_z \end{bmatrix} = k \begin{bmatrix} \text{sen}\theta\text{cos}\phi \\ \text{sen}\theta\text{sen}\phi \\ \text{cos}\theta \end{bmatrix}.$$

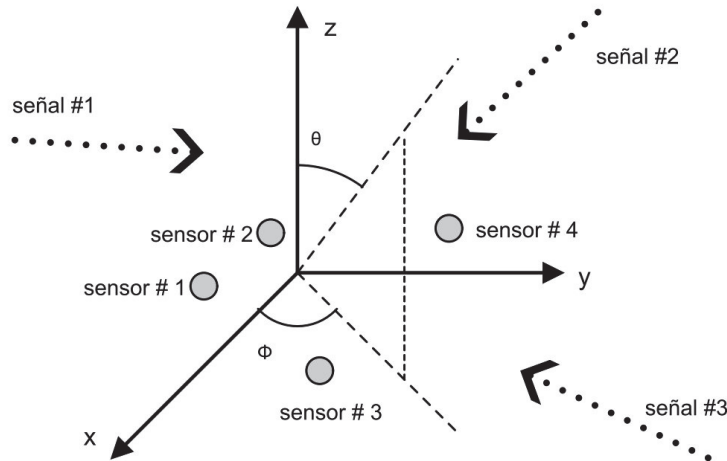


Figura 4.1 Esquema general de un arreglo de cuatro micrófonos con tres señales incidentes.

El número de onda tridimensional representado por el vector K es un parámetro espacio-temporal porque cambia según la dirección de la fuente de la señal y, como todos los micrófonos están en una posición diferente uno del otro, el voltaje a la salida de cada micrófono en un instante dado es distinto en cada elemento del arreglo.

Ahora considérese una señal de sonido $s(t)$ con frecuencia f y longitud de onda λ que incide en un arreglo lineal con N elementos numerados $0, 1, \dots, N - 1$, donde los micrófonos están separados $\frac{\lambda}{2}$ y reciben las N señales denominadas $x_n(t)$ como se muestra en la Figura 4.2. Como la distancia entre micrófonos no es grande, se puede considerar que las señales $x_n(t)$ están correlacionadas, es decir, que esencialmente es la misma señal $s(t)$ con un retraso en el tiempo. Se considera que el micrófono número cero es el sensor de referencia para medir la fase de la señal que incide en cada uno de los otros micrófonos. Para medir la diferencia de fase de las señales es necesario medir el retraso en tiempo de las señales $x_n(t)$ con respecto a la señal $x_0(t)$ del micrófono cero [12].

Examinando la geometría del arreglo de la Figura 4.2 y usando trigonometría y la velocidad de propagación del sonido se llega a que el retraso de cada señal $x_n(t)$ se puede calcular como

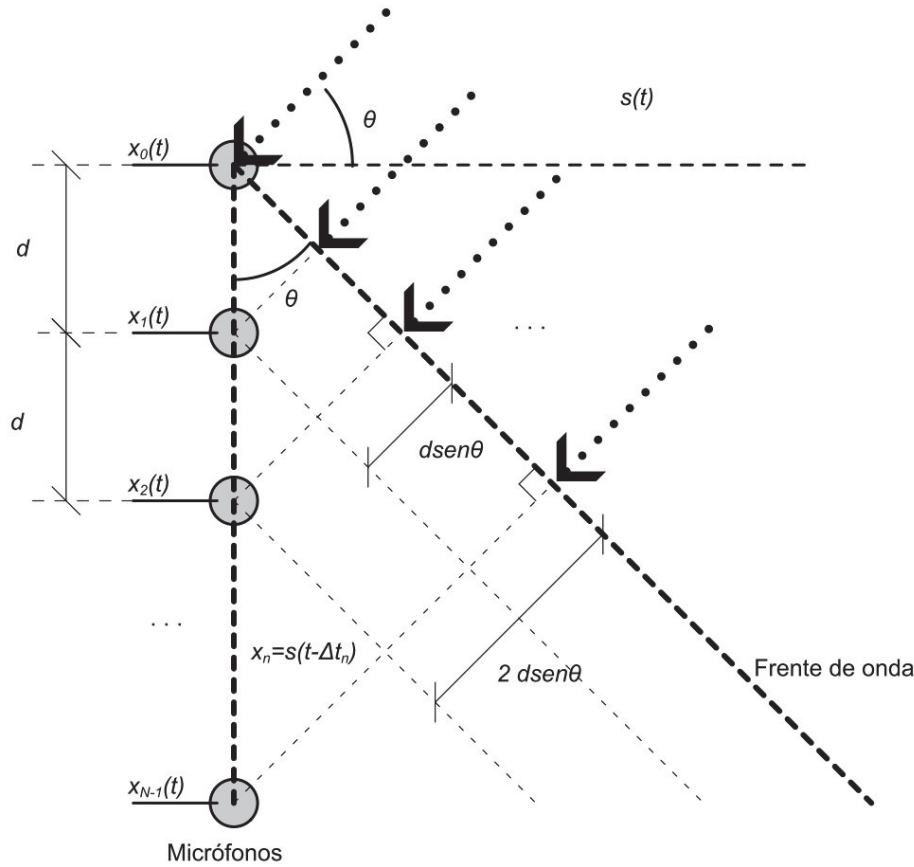


Figura 4.2 Esquema de un arreglo lineal de N micrófonos.

$$\Delta t_n = \frac{n d \operatorname{sen} \theta}{c}. \quad (4.1)$$

Entonces, si la señal que recibe el micrófono cero es $e^{j2\pi f t}$

$$x_0(t) = e^{j2\pi f t}$$

y la señal recibida por el n -ésimo micrófono es [19]

$$x_n(t) = e^{j2\pi f (t - \Delta t_n)}.$$

4.2 Formador de haz fijo para banda angosta

En un formador de haz, con base en las señales de un arreglo de micrófonos, se desea amplificar una señal de interés que incide de una dirección en particular en presencia de ruido y señales que interfieren su adquisición en forma pura. La secuencia de valores capturados por los sensores son procesadas para atenuar o eliminar las señales no

deseadas, extrayendo la señal de interés. Así, la respuesta espacial del sistema es como un haz que apunta en la dirección de la señal deseada y atenúa el resto de las señales incidentes [19].

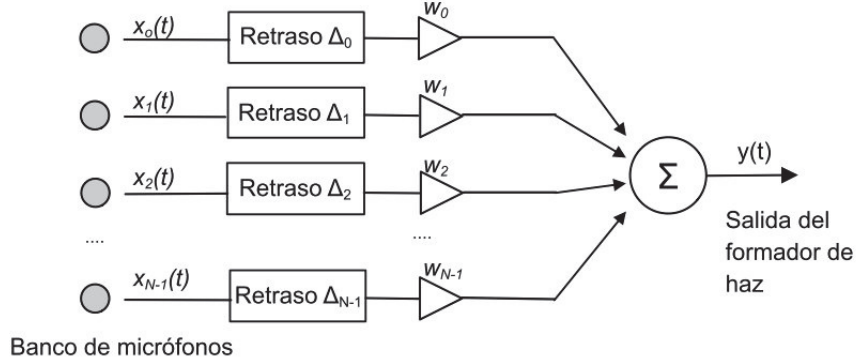


Figura 4.3 Esquema de un formador de haz fijo de banda angosta.

La Figura 4.3 muestra la estructura de un formador de haz fijo para un arreglo lineal, donde N micrófonos muestrean la señal y la salida $y(t)$ es la combinación lineal instantánea de estas muestras, matemáticamente cuando $\theta = 0^\circ$ se expresa como

$$y(t) = \sum_{n=0}^{N-1} x_n(t) w_n^*.$$

El formador de haz asociado con esta estructura sólo es útil para señales de banda angosta. Si se analiza la respuesta de este formador de haz para la señal incidente $e^{j\omega t}$ con una frecuencia angular ω y una dirección de arribo θ , donde $\theta \in [-\pi/2, \pi/2]$ y se mide con respecto a la perpendicular del arreglo de micrófonos como lo muestra la Figura 4.2 [19], entonces la salida del arreglo es

$$y(t) = e^{j\omega t} \sum_{n=0}^{N-1} e^{-\Delta t_n} w_n^* = W^H D(\omega, \theta)$$

donde el vector W contiene los N coeficientes conjugados complejos de ponderación de cada micrófono

$$W = [w_0 \quad w_1 \quad \dots \quad w_{N-1}]^T$$

y el vector $D(\omega, \theta)$ se denomina vector de direccionamiento el cuál representa los retrasos en el tiempo de las señales en cada micrófono y está dado por

$$D = [1 \quad e^{-j\omega(t-\Delta t_1)} \quad \dots \quad e^{-j\omega(t-\Delta t_{N-1})}]^T. \quad (4.2)$$

4.2.1 Patrón de radiación

La forma más directa de observar el rendimiento del formador de haz es mediante su patrón de radiación, el cuál provee una caracterización completa del comportamiento entrada-salida del sistema. De lo mostrado hasta el momento se puede inferir que el formador de haz es un *filtro espacial* y su patrón de radiación se define como la magnitud de la respuesta del sistema cuando una señal incide en cada dirección.

Para encontrar una expresión analítica a la respuesta del formador de haz respecto al ángulo de incidencia de las señales considérese una señal $s(t)$ que incide en un arreglo lineal como el mostrado en la Figura 4.2. En cierto momento el frente de onda llega al micrófono cero, para llegar al micrófono uno debe viajar una distancia adicional de $dsen\theta$, para llegar al micrófono dos debe viajar $2dsen\theta$ y así sucesivamente. Si consideramos la fase de la señal del micrófono cero como referencia, la fase de la señal en el resto de micrófonos es

$$\phi_n = k(N - n)dsen\theta \quad \text{para } n = 0, 1, \dots, N - 1$$

donde k es el número de onda de la señal.

En notación compleja se puede escribir como

$$s_n(\theta) = s_e(\theta)a_n e^{jk(N-n)dsen\theta} \quad \text{para } n = 0, 1, \dots, N - 1$$

donde $s_e(\theta)$ representa el patrón de radiación de un sólo micrófono por separado y a_n es la amplitud de la señal recibida en cada sensor. Si se consideran micrófonos omnidireccionales ideales se puede suponer que $s_e = 1$ y $a_n = 1$. Si ahora se suman las señales como lo indica el esquema en la Figura 4.3 con $w_n = 1$ se obtiene que la respuesta del formador de haz está dada por [35]

$$PR(\theta) = \sum_{n=0}^{N-1} e^{jk(N-n)dsen\theta} = \sum_{n=0}^{N-1} e^{j(N-n)T} \quad (4.3)$$

donde $T = kdsen\theta$.

La parte derecha de la ecuación (4.3) corresponde a una serie geométrica finita. Para encontrar una forma mas cómoda de calcular la respuesta se multiplican ambos lados de la ecuación por e^{jT}

$$PR(\theta)e^{jT} = e^{jNT} + e^{j(N-1)T} + \dots + e^{j2T} + e^{jT}. \quad (4.4)$$

Ahora se resta la ecuación (4.3) de (4.4) y se obtiene

$$PR(\theta)(e^{jT} - 1) = e^{jNT} - 1$$

lo cual, después de separar y reagrupar términos, puede reescribirse como

$$PR(\theta) = \frac{e^{j\frac{NT}{2}}(e^{j\frac{NT}{2}} - e^{-j\frac{NT}{2}})}{e^{j\frac{T}{2}}(e^{j\frac{T}{2}} - e^{-j\frac{T}{2}})} = e^{j\frac{N-1}{2}T} \frac{\text{sen}(\frac{N}{2}T)}{\text{sen}(\frac{1}{2}T)}$$

finalmente

$$|PR(\theta)| = \left| \frac{\text{sen}\left(\frac{N}{2}kdsen\theta\right)}{\text{sen}\left(\frac{1}{2}kdsen\theta\right)} \right| = \left| \frac{\text{sen}\left(\pi\frac{Nd}{\lambda}sen\theta\right)}{\text{sen}\left(\pi\frac{d}{\lambda}sen\theta\right)} \right|. \quad (4.5)$$

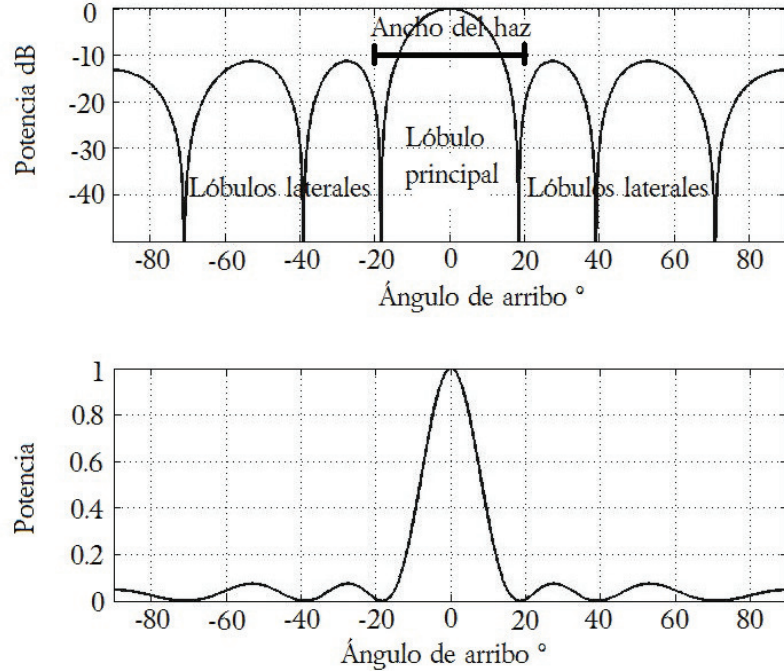


Figura 4.4 Patrón de radiación de un formador de haz fijo con cuatro micrófonos.

En la Figura 4.4 se muestra el patrón de radiación de un formador de haz fijo de cuatro micrófonos con una separación $d = 9 \text{ cm}$, $w_n = 1$, con una frecuencia de muestreo $f_s = 8000 \text{ Hz}$, un tono de prueba con frecuencia $f = 3000 \text{ Hz}$, apuntando a 0° . En la parte superior está graficada la potencia de la señal de salida del formador en dB , mientras que la gráfica inferior está en escala lineal. Generalmente sólo se muestra la respuesta en dB porque permite visualizar mejor las magnitudes pequeñas, por lo que en adelante únicamente se incluirán las gráficas con potencia en escala logarítmica.

Al lóbulo de mayor amplitud se le denomina *lóbulo principal*, al resto de los lóbulos se les denominan *lóbulos laterales*.

4.2.2 Ancho del haz

El ancho del haz en un formador de haz se define como la diferencia de los ángulos del par de *ceros* más cercanos a la dirección deseada, es decir, el ancho del lóbulo principal [3]. En la Figura 4.4 se puede observar que estos *ceros* están aproximadamente en los

ángulos -18° y 18° . Analíticamente se pueden encontrar estos ángulos partiendo de la ecuación (4.5) e igualándola a 0 se tiene que

$$\text{sen}\left(\pi \frac{Nd}{\lambda} \text{sen}\theta\right) = 0$$

como $\lambda = \frac{c}{f}$

$$\pi f N \frac{d}{c} (\text{sen}\theta) = m\pi$$

donde $m = \pm 1, \pm 2, \dots$. Si agregamos una componente θ_0 que representa una dirección donde apunta el formador de haz

$$\pi f N \frac{d}{c} (\text{sen}\theta_0 - \text{sen}\theta) = m\pi.$$

Los primeros dos ángulos θ_1 y θ_2 que hacen cero a la ecuación se pueden calcular sustituyendo $m = 1$ y $m = -1$. Despejando se obtiene

$$\theta_1 = \arcsen\left(\text{sen}\theta_0 - \frac{c}{dfN}\right)$$

$$\theta_2 = \arcsen\left(\text{sen}\theta_0 + \frac{c}{dfN}\right)$$

el ancho del haz AH se calcula con la resta de θ_1 y θ_2

$$AH = \arcsen\left(\text{sen}\theta_0 + \frac{c}{dfN}\right) - \arcsen\left(\text{sen}\theta_0 - \frac{c}{dfN}\right)$$

Para el caso particular donde $\theta = 0^\circ$ se tiene

$$AH_0 = 2\arcsen\left(\frac{c}{dfN}\right). \quad (4.6)$$

De las ecuaciones anteriores se observa que mientras se incrementa la distancia d o el número de sensores N , disminuye el ancho del lóbulo principal porque el argumento del arco seno en la ecuación (4.6) se hace más pequeño [1]. Sin embargo, como se verá en la siguiente sección, no se debe aumentar la distancia d indefinidamente.

4.2.3 Lóbulos gratinados

En la conversión analógico a digital, se muestrea una señal de tiempo continuo a una secuencia de números en tiempo discreto; en este contexto el *fenómeno de aliasing* se refiere a que muchas señales en tiempo continuo se pueden representar por la misma secuencia discreta, lo cual ocurre cuando la frecuencia de muestreo es menor a la frecuencia de Nyquist. En el procesamiento de arreglos de micrófonos, los sensores muestrean las señales incidentes espacialmente y si las señales de distintas direcciones no se muestrean por un arreglo lo suficientemente denso, es decir, que el espacio entre cada elemento del arreglo es demasiado grande, distintas direcciones de incidencia

pueden estar representadas por el mismo vector de direccionamiento por lo que no se puede determinar unívocamente su dirección. De forma similar al caso temporal, se tiene un *fenómeno de aliasing espacial* debido a la ambigüedad en las direcciones de incidencia [19].

Así, para señales de la misma frecuencia angular ω y su correspondiente longitud de onda λ pero diferentes ángulos de incidencia θ_1 y θ_2 cuando $(\theta_1, \theta_2) \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, el aliasing espacial implica que $D(\theta_1, \omega) = D(\theta_2, \omega)$.

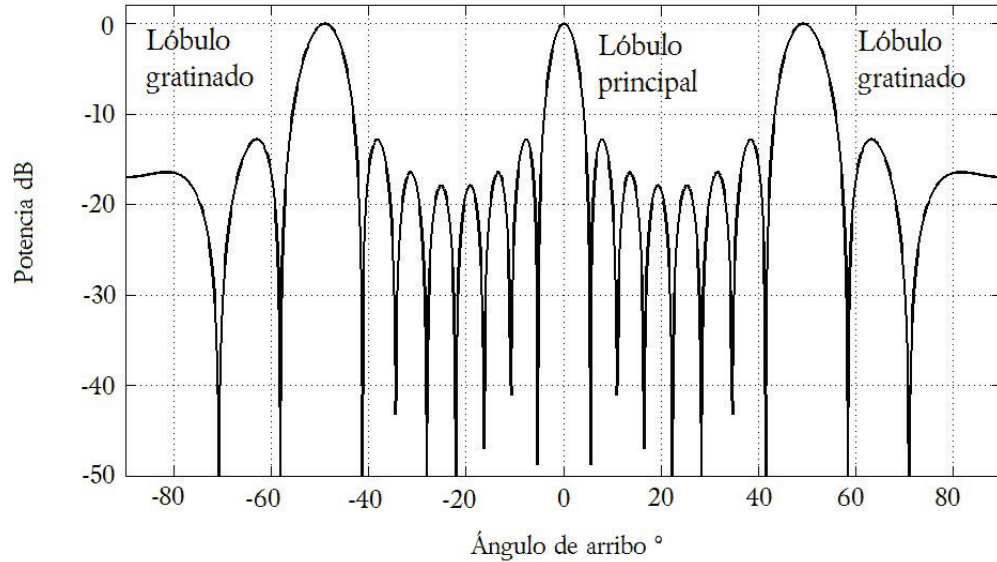


Figura 4.5 Patrón de radiación de un formador de haz fijo con ocho micrófonos que presenta aliasing espacial.

En la Figura 4.5 se presenta el patrón de radiación de un formador de haz fijo de ocho micrófonos con una separación $d = 15 \text{ cm}$, $w_n = 1$, con una frecuencia de muestreo $f_s = 10000 \text{ Hz}$, un tono de prueba con frecuencia $f = 3000 \text{ Hz}$, apuntando a 0° que presenta lóbulos gratinados en $\pm 50^\circ$ debido al aliasing espacial.

Para encontrar una expresión analítica que evite provocar este fenómeno considérese un arreglo lineal espaciado uniformemente con distancia entre elementos d , con el retraso de cada micrófono expresado en la ecuación (4.1) multiplicada por ω se tiene

$$\omega \Delta t_n = \frac{2\pi n d s e n \theta}{\lambda}$$

si existe aliasing espacial entonces los elementos del vector de direccionamiento son iguales, observando cada elemento del vector en la ecuación (4.2) se puede escribir que existe aliasing espacial cuando

$$e^{-j\omega \frac{2\pi n d s e n \theta_1}{\lambda}} = e^{-j\omega \frac{2\pi n d s e n \theta_2}{\lambda}}.$$

Para evitar caer en esta situación la condición $|\frac{2\pi d \text{sen}\theta}{\lambda}|_{\theta=\theta_1, \theta_2} < \pi$ se debe cumplir. Entonces se tiene

$$|\frac{d}{\lambda \text{sen}\theta}| < \frac{1}{2}$$

como $|\text{sen}\theta| \leq 1$, se llega a que $d < \frac{\lambda}{2}$ para evitar aliasing espacial. Cuando esta condición no se cumple aparecen múltiples direcciones donde la potencia de la señal es máxima, a estas direcciones se les denomina *lóbulos gratinados* y se puede calcular su ubicación mediante [1]

$$\theta_g = \text{sen}^{-1}(\text{sen}\theta_0 - m \frac{c}{df})$$

donde θ_g es la dirección del lóbulo gratinado, θ_0 es la dirección del lóbulo principal y es un entero $m = \pm 1, \pm 2, \dots$

4.2.4 La técnica de ventanas

La *técnica de ventanas* es una técnica de síntesis de filtros FIR; debido a la semejanza en la estructura de un filtro FIR y un formador de haz se puede usar este concepto para modificar el rendimiento del formador de haz al sustituir los coeficientes de la ventana por los w_n de la Figura 4.3. Esta técnica es una de las más viejas para el diseño de filtros FIR, en parte debido a su simplicidad de cálculo ya que no se requieren resolver problemas de optimización complejos. En el caso de un formador de haz, el efecto de las ventanas se refleja en la disminución o el aumento del ancho del lóbulo principal y en la disminución o el aumento de la magnitud de los lóbulos laterales.

Algunas de las ventanas más comunes son la rectangular, triangular, Hamming, Hanning, Blackman, Kaiser y Dolph-Chebyshev [1], [37], aunque existen otras. En el Cuadro 4.1 se muestra cómo calcular los pesos para algunas de éstas ventanas.

La mayoría de las ventanas se calculan de forma fija, sin embargo, la ventana Dolph-Chebyshev, también llamada ventana *minmax*, es más bien un método de diseño donde se minimiza el ancho del haz en función de una disminución de los lóbulos laterales

Cuadro 4.1 Definición de ventanas comunes.

Ventana	Definición
<i>Rectangular</i>	$w_n = 1$ para $n = 0, 1, \dots, N - 1$
<i>Triangular</i>	$w_n = \begin{cases} \frac{2n}{N-1} & , 0 < n < \frac{N-1}{2} \\ 2 - \frac{2n}{N-1} & , \frac{N-1}{2} < n < N - 1 \end{cases}$ para $n=0,1,\dots,N-1$
<i>Hanning</i>	$w_n = 0.5 - 0.5\cos(2\pi \frac{n}{N})$ para $n = 0, 1, \dots, N - 1$
<i>Hamming</i>	$w_n = 0.54 - 0.46\cos(2\pi \frac{n}{N})$ para $n = 0, 1, \dots, N - 1$
<i>Blackman</i>	$w_n = 0.42 - 0.5\cos(2\pi \frac{n-1}{N}) + 0.08\cos(4\pi \frac{n-1}{N})$ para $n = 0, 1, \dots, N - 1$

deseada. El método para calcular la ventana Chebyshev-Dolph se puede encontrar en [1], además Matlab tiene funciones específicas para calcular ventanas.

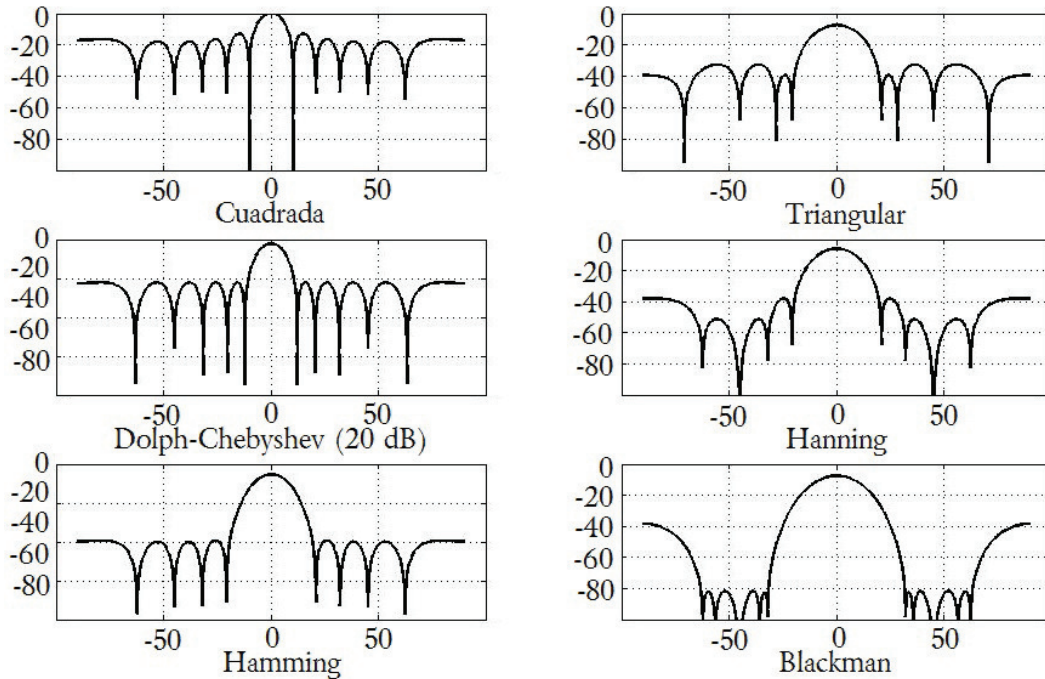


Figura 4.6 Patrón de radiación de un formador de haz fijo con ocho micrófonos utilizando ventanas.

En la Figura 4.6 se muestra el patrón de radiación de un formador de haz fijo de ocho micrófonos con una separación $d = 8 \text{ cm}$, con una frecuencia de muestreo $f_s = 10000 \text{ Hz}$, un tono de prueba con frecuencia $f = 3000 \text{ Hz}$, apuntando a 0° y utilizando las ventanas antes mencionadas. La ventana Dolph-Chebyshev se diseñó para tener una disminución de 20 dB en los lóbulos laterales pero se puede diseñar con cualquier otro valor dependiendo de las necesidades del diseño. En general se observa que mientras mayor sea la atenuación de los lóbulos laterales mayor es el ancho del lóbulo principal. La atenuación de los lóbulos laterales en la ventana cuadrada es muy malo pero el ancho del lóbulo principal es bueno, en contraste con la ventana Blackman cuya atenuación de lóbulos laterales es excelente pero el ancho del principal es tres veces el ancho de la ventana cuadrada.

4.2.5 Direccionamiento del haz

Para un arreglo lineal uniformemente espaciado, su respuesta también se puede expresar como

$$P(\sin\theta) = \sum_{n=0}^{N-1} w_n^* e^{jn\omega \frac{d}{c} \sin\theta}. \quad (4.7)$$

Considérese que el conjunto de coeficientes w_n , $n = 0, 1, \dots, N - 1$ forma un haz que apunta a la dirección perpendicular al arreglo, es decir $\theta = 0^\circ$. Para cambiar la dirección del haz a otro valor θ_0 , se debe añadir un retraso de $(N - 1) \frac{d \text{sen} \theta_0}{c}$ a la señal recibida por el primer elemento del arreglo, un retraso de $(N - 2) \frac{d \text{sen} \theta_0}{c}$ para el segundo elemento y así sucesivamente. La nueva respuesta del arreglo se puede expresar como

$$P(\text{sen} \theta - \text{sen} \theta_0) = e^{-j(N-1) \frac{d\omega}{c} \text{sen} \theta_0} \sum_{n=0}^{N-1} w_n^* e^{-jn \frac{d\omega}{c} (\text{sen} \theta - \text{sen} \theta_0)} \quad (4.8)$$

donde el término $e^{-j(N-1) \frac{d\omega}{c} \text{sen} \theta_0}$ representa un retraso constante y en adelante será ignorado.

Para evitar aliasing espacial se considera $d = \frac{\lambda}{2}$. Entonces el término $\frac{d\omega}{c} = \pi$ y la ecuación (4.7) cambia a

$$P(\text{sen} \theta - \text{sen} \theta_0) = \sum_{n=0}^{N-1} w_n^* e^{-jn\pi(\text{sen} \theta - \text{sen} \theta_0)}.$$

Como la función $e^{-jn\pi x}$ es periódica con periodo dos, si se compara con la ecuación (4.7), la respuesta de la ecuación (4.8) es simplemente una versión recorrida circularmente de (4.7) por un periodo $\text{sen} \theta \in [-1, 1]$.

En la Figura 4.7 se observa el patrón de radiación de un formador de haz fijo de ocho micrófonos con una separación $d = 8.5 \text{ cm}$, con una frecuencia de muestreo $f_s = 10000 \text{ Hz}$, un tono de prueba con frecuencia $f = 2000 \text{ Hz}$, utilizando una ventana cuadrada. La línea segmentada es la respuesta del sistema apuntando a 0° y la línea sólida es la respuesta apuntando a 40° .

4.2.6 Respuesta en frecuencia

Hasta ahora se ha utilizado un solo tono puro para presentar el patrón de radiación de los formadores de haz, sin embargo, las características de un formador de haz, en especial el ancho del haz, dependen de la frecuencia de la señal de entrada como lo mostró la ecuación (4.6). Este hecho se ilustra en la Figura 4.8 donde se muestra el patrón de radiación de un formador de haz fijo de ocho micrófonos con una separación $d = 5 \text{ cm}$, con una frecuencia de muestreo $f_s = 10000 \text{ Hz}$ que se probó con tonos de prueba con frecuencias entre 150 y 4000 Hz , apuntando a 0° utilizando una ventana cuadrada.

Conforme la frecuencia disminuye, el ancho del haz aumenta hasta el punto en que para frecuencias muy bajas el haz es tan ancho que prácticamente no existe filtrado espacial. Esto se debe a que al variar la frecuencia cambia el valor de λ y como se ha mostrado antes, mientras d sea menor a $\frac{\lambda}{2}$ el ancho del haz tiende a aumentar. Por lo tanto es claro que el patrón de radiación no es el mismo a través del intervalo de frecuencias de forma que las señales en ángulos distintos al que apunta el haz no serán canceladas, en cambio serán afectadas no uniformemente por un filtro pasa bajas. Los formadores

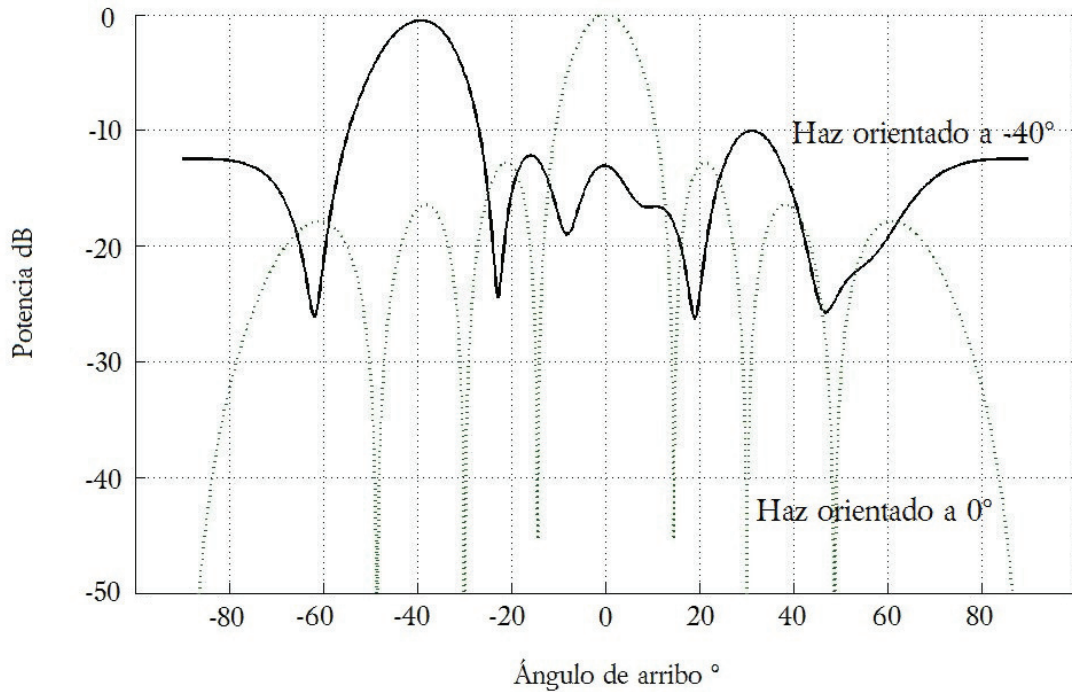


Figura 4.7 Patrón de radiación de un formador de haz fijo con ocho micrófonos apuntando a -40° .

de haz diseñados para manipular señales de banda ancha se denominan *formadores de haz de respuesta invariante* o *formadores de haz de banda ancha* y son el tema de la siguiente sección.

4.3 Formador de haz de banda ancha

La estructura del formador anterior funciona de manera eficiente solo para un intervalo pequeño de tonos puros; cuando el ancho de banda de la señal se incrementa, su rendimiento se deteriora significativamente. Esto se puede explicar de la siguiente manera, supóngase que tenemos N señales nombradas $s_n(t)$ con $n = 0, 1, \dots, N - 1$ que inciden en el arreglo con un ángulo θ_n con $n = 0, 1, \dots, N - 1$ respectivamente. La primera señal $s_0(t)$ es la señal de interés y las demás son consideradas ruido. El vector de direccionamiento d_n para estas señales está dado por

$$d_n(\omega, \theta) = [1 \quad e^{-j\omega\Delta t_1(\theta_n)} \quad \dots \quad e^{-j\omega\Delta t_{N-1}\theta_n}].$$

Idealmente, se desea que el formador de haz tenga una respuesta fija hacia la señal de interés y cancele totalmente el resto de señales. Esta idea se puede expresar en la siguiente matriz

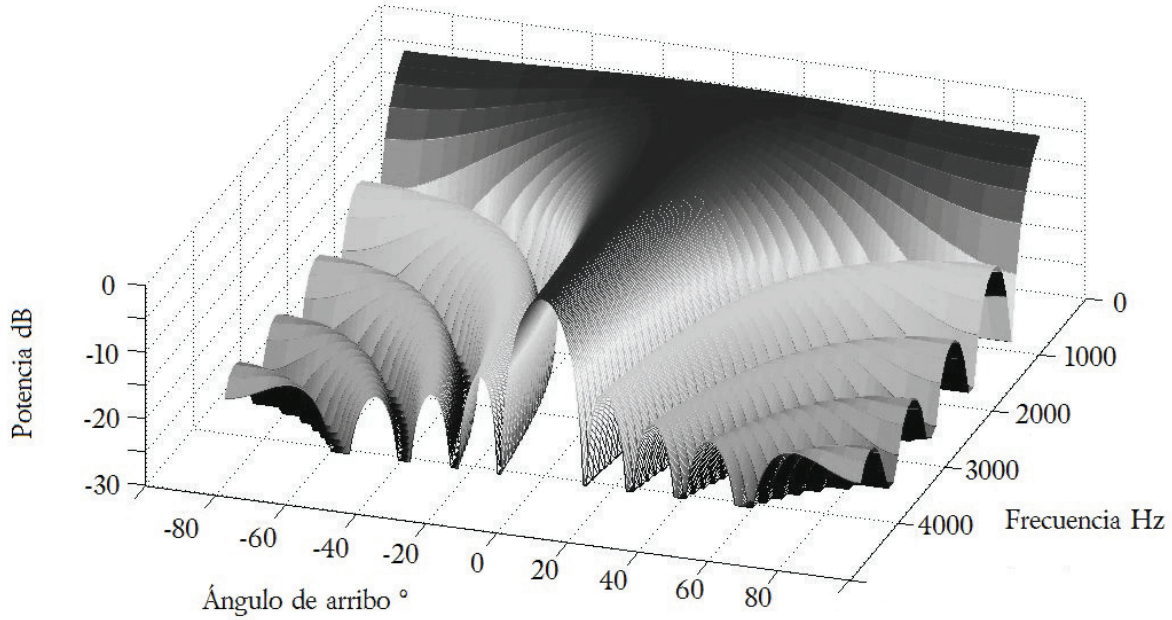


Figura 4.8 Patrón de radiación de un formador de haz fijo con ocho micrófonos en el intervalo 150 a 4000 Hz.

$$\begin{bmatrix} 1 & e^{-j\omega\Delta t_1(\theta_0)} & \dots & e^{-j\omega\Delta t_{N-1}(\theta_0)} \\ 1 & e^{-j\omega\Delta t_1(\theta_1)} & \dots & e^{-j\omega\Delta t_{N-1}(\theta_1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\omega\Delta t_1(\theta_{N-1})} & \dots & e^{-j\omega\Delta t_{N-1}(\theta_{N-1})} \end{bmatrix} \begin{bmatrix} w_0^* \\ w_1^* \\ \vdots \\ w_{N-1}^* \end{bmatrix} = \begin{bmatrix} cte \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Si la matriz es de rango completo se puede encontrar un conjunto de coeficientes w_n para cancelar las $N - 1$ señales, y el valor exacto de estos coeficientes para anular las señales de ruido estarán dados en función del ángulo de arribo θ_n [19].

Para señales de banda ancha, con un número infinito de tonos a distintas frecuencias, los coeficientes deben ser distintos para cada frecuencia que esté en la señal. Entonces los coeficientes se deben presentar como

$$w(\omega) = [w_0(\omega) \quad w_1(\omega) \quad \dots \quad w_{N-1}(\omega)].$$

Tradicionalmente, una forma de calcular esos coeficientes es mediante una línea de retraso en el tiempo o filtros digitales (FIR o IIR) aunque también es posible implementarlos con arreglos bidimensionales o con algoritmos que trabajan directamente con la transformada de Fourier de la señal [1]. Con estos métodos se logra tener un proceso temporal para filtrar frecuencialmente cada sensor del arreglo y compensar la dependencia frecuencial del formador de haz. La salida de un formador de haz como el descrito se puede expresar como [19]

$$y(t) = \sum_{n=0}^{N-1} \sum_{i=0}^{J-1} x_n(t - iT_s) w_{n,i}^*$$

donde $J-1$ es el número de unidades de retardos asociados a cada uno de los N sensores y T_s es el valor en segundos de unidad de retardo. La Figura 4.9 presenta la estructura de un formador de haz de banda ancha [19].

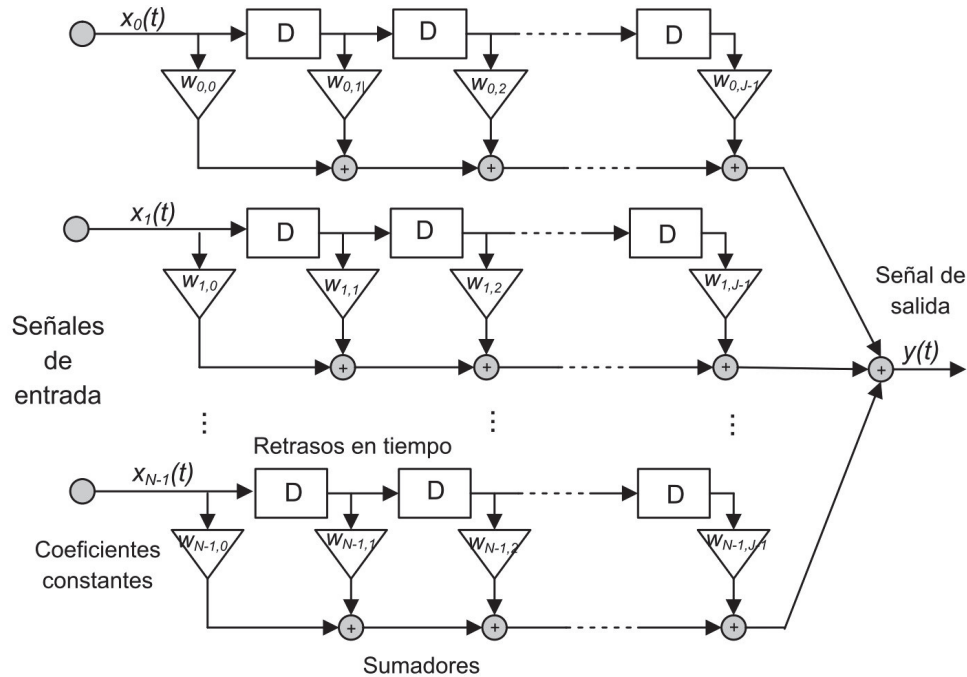


Figura 4.9 Estructura general de un formador de haz de banda ancha.

Dentro de las opciones disponibles para implementar un formador de haz de respuesta invariante se ha decidido usar la técnica por filtros digitales debido a su sencillez para el diseño y su relativamente baja complejidad computacional.

4.3.1 Formador de banda ancha usando filtros digitales

En el procesamiento de arreglos de transductores se le llama *apertura* a la longitud total del arreglo [11]; cuando en la apertura existe un número finito de sensores se dice que es una apertura discreta, si se tiene un número infinito de sensores en una apertura de longitud finita se dice que es una apertura continua. El concepto de apertura continua es un concepto solamente teórico pero como se verá adelante es de utilidad para formular una solución a la dependencia frecuencial del formador de haz.

Considérese una apertura continua unidimensional que va de $x = 0$ a $x = l_{max}$ como se observa en la Figura 4.10.

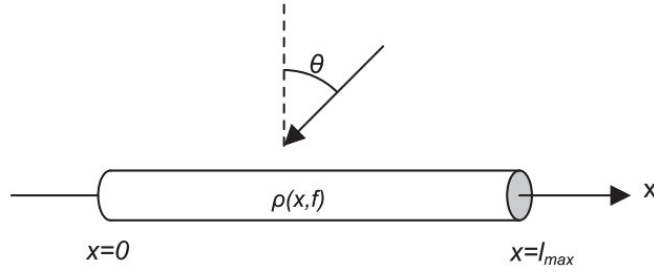


Figura 4.10 Apertura continua unidimensional.

La respuesta de esta apertura a ondas planas se puede escribir como

$$h(\theta, f) = \int_0^{\infty} e^{j2\pi \frac{f \text{sen} \theta}{c} x} \rho(x, f) dx \quad (4.9)$$

donde $\rho(x, f)$ es una función continua de la posición x y la frecuencia f y se le llama la *iluminación de la apertura*, se asume que $\rho(x, f) = 0$ para $x > x_{max}$ y está dada por

$$\rho(x, f) = fG(xf)$$

donde $G(xf)$ es una función compleja arbitraria de una variable. Entonces, la respuesta del sistema expresada en la ecuación (4.9) queda como

$$h(\theta, f) = \int_0^{\infty} e^{j2\pi \frac{f \text{sen} \theta}{c} x} fG(xf) dx$$

con un cambio de variable $\psi = xf$

$$h(\theta, f) = \int_0^{\infty} e^{j2\pi \frac{\psi \text{sen} \theta}{c}} G(\psi) d\psi = h(\theta) \quad (4.10)$$

de esta forma la respuesta sólo depende del ángulo de arribo. La ecuación (4.10) nos permite observar que el filtrado para lograr un formador de haz de banda ancha se puede dividir en dos partes; una en la que únicamente depende de la frecuencia y la posición y otra que únicamente depende de la posición [1]. La apertura continua es una forma de considerar que podemos colocar tantos micrófonos como sea necesario en cualquier posición de la apertura, sin embargo, físicamente no es posible implementar un sistema así. Una forma de implementar una aproximación es colocando varios sub-arreglos dentro de la apertura cuyas distancia d son distintas, de tal forma que cada sub-arreglo procesa solo una fracción del ancho de banda total de la señal [19].

4.3.2 Formador de haz por Sub-bandas

El formador de haz por sub-bandas tiene tres etapas principales: separación de sub-bandas, filtrado espacial de cada sub-banda e integración de las sub-bandas para obtener

una salida con el mismo ancho de banda que la entrada. Normalmente esta técnica involucra dos tipos de filtros, uno para descomponer la señal en sub-bandas y otro para combinar las sub-bandas y formar la señal de salida. A ese sistema de descomposición y combinación se le llama *banco de filtrado*. Cuando el ancho de banda de una señal se separa en intervalos o sub-bandas, cada sub-banda puede muestrearse a menor frecuencia debido a su menor ancho de banda. Esto permite que cada sub-banda se procese por separado a una velocidad menor, lo cual implica una disminución en la demanda de recursos computacionales, y como se verá en el siguiente capítulo, es una forma de disminuir la distancia de campo lejano. Cuando se ocupan distintas frecuencias de muestreo durante distintas partes del sistema, se le conoce como banco de filtrado multi-tasa. Para lograr cambiar la frecuencia de muestreo se usan las operaciones de *decimación e interpolación* vistas en el Capítulo 2 *Análisis de señales*.

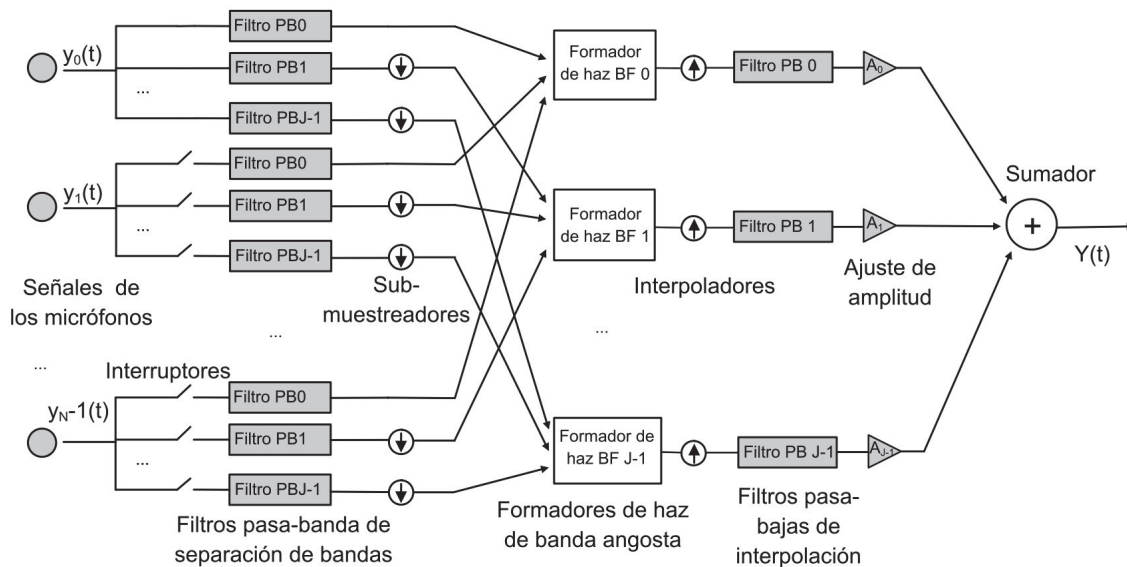


Figura 4.11 Estructura del formador de haz por sub-bandas.

En la Figura 4.11 se muestra finalmente la estructura de un formador de haz realizable que compensa la dependencia frecuencial el cual será el proceso utilizado en esta tesis. Como se observa en la Figura 4.11, el sensor cero debe forzosamente pertenecer a todas las M sub-bandas, las sub-bandas pueden compartir sensores pero ningún par de sub-bandas debe tener los mismos elementos. Cada sensor se filtra frecuencialmente para separar el ancho de banda en intervalos menores, luego de reducir la frecuencia de muestreo, las señales de cada sub-banda se filtran espacialmente en un formador de haz de banda angosta diseñado para ser eficiente en el ancho de banda de cada sub-banda, el resultado de cada formador de haz de banda angosta se interpola para finalmente conformar una señal filtrada espacialmente con un ancho de banda y una frecuencia de muestreo igual a la señal de entrada.

En la Figura 4.12 se muestra un formador de haz de banda ancha diseñado por el método

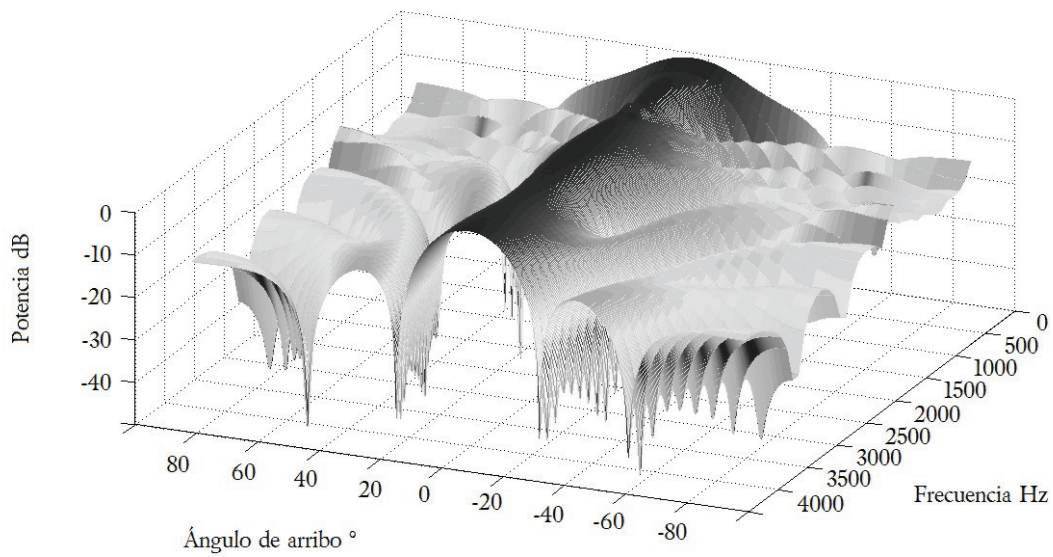


Figura 4.12 Respuesta de un formador de haz por sub-bandas.

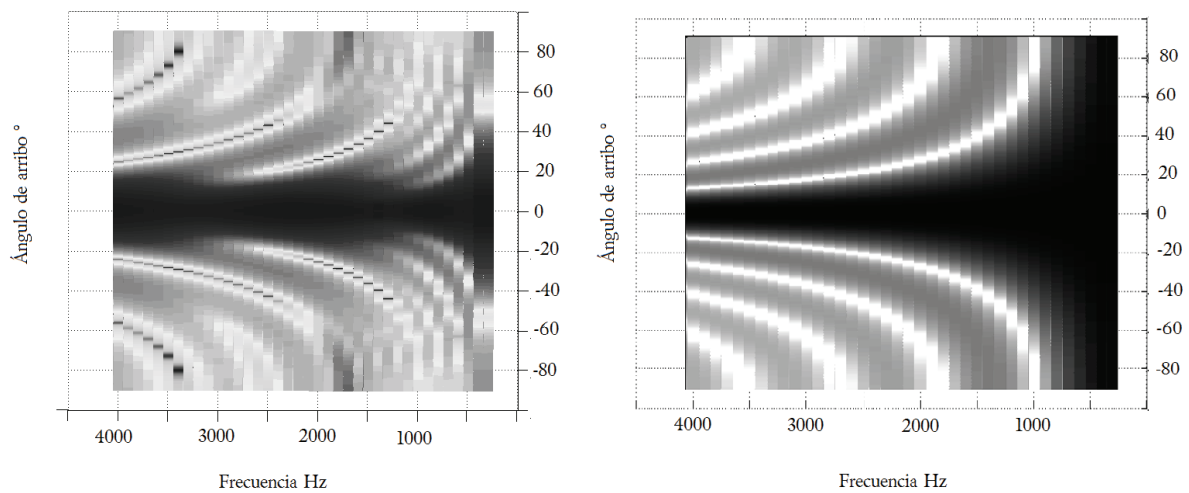


Figura 4.13 Vista superior de las respuestas de un formador de banda ancha usando sub-bandas (izquierda) y un formador de haz de banda angosta (derecha).

antes descrito, el cual se compone de cinco sub-bandas de ocho micrófonos cada una. Cada sub-arreglo se relaciona armónicamente con el anterior, es decir, las distancias entre cada elemento se relacionan de la siguiente manera

$$d_5 = 2d_4 = 4d_3 = 8d_2 = 16d_1.$$

Comparando las gráficas de vista superior que muestra la Figura 4.13 es más evidente que el formador de haz de banda angosta funciona como un filtro paso bajas, se observa que el ancho del haz crece conforme baja la frecuencia de tal modo que las frecuencias menores a $1000Hz$ casi no se filtran espacialmente, en contraste con el formador de haz de banda ancha implementado con filtros digitales donde la respuesta del sistema se mantiene casi constante en todo el intervalo, teóricamente si se implementan un número infinito de sub-bandas la respuesta puede ser totalmente invariante y se tendría una apertura continua.

4.4 Resumen

En este capítulo se describe cómo se captura la información de una señal usando un arreglo de sensores. Posteriormente se explica el formador de haz de banda angosta y cómo se puede usar la técnica de ventanas para modificar su rendimiento. Asimismo, se expone la dependencia frecuencial del formador de haz y se describe la manera en que su rendimiento se degrada al considerar un intervalo grande de frecuencias, lo que justifica por qué no se debe usar en señales de voz. Finalmente, se propone una estructura realizable que compensa las deficiencias del formador de haz de banda angosta posibilitando su uso para señales de banda ancha.

Capítulo 5

Diseño e implementación del sistema

Partiendo de la teoría presentada anteriormente, se diseñó y construyó un sistema que adquiere señales de un banco de micrófonos y las filtra espacialmente usando un formador de haz implementado con la técnica de sub-bandas. En este capítulo se detalla el proceso de diseño del formador de haz partiendo de las especificaciones deseadas y limitaciones prácticas que conlleva este proyecto. Además, se explica cómo se implementaron los conceptos matemáticos vistos en capítulos anteriores en una máquina digital como un *DSP*. Otro punto importante dentro del diseño del sistema es que su función pueda ser adaptada y expandida a otras aplicaciones de filtrado espacial por lo que también se contempló la forma en que la señal filtrada se comunica al exterior.

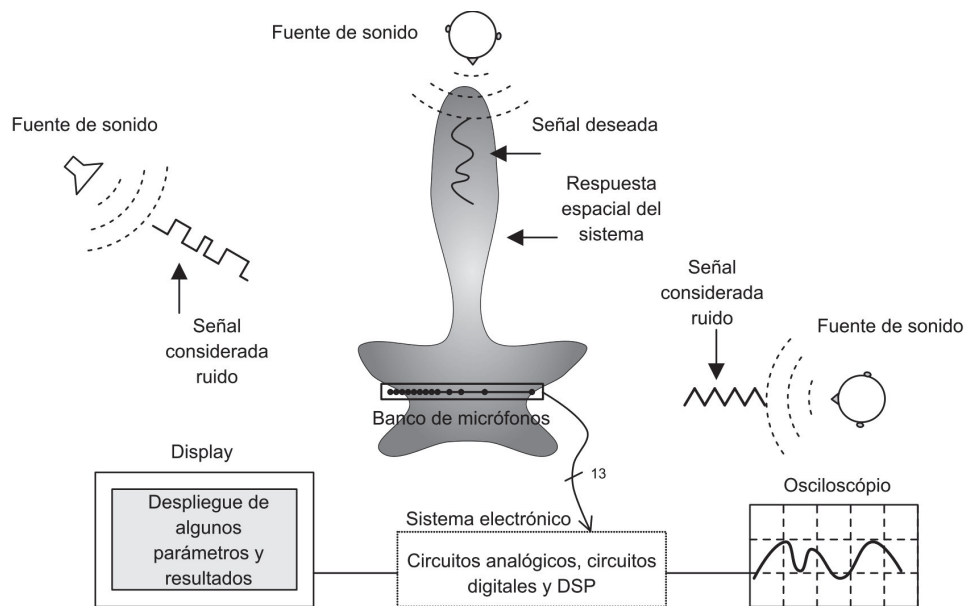


Figura 5.1 Esquema funcional del sistema.

Se quiere que el sistema filtre la señal o señales de voz que provienen de una dirección y atenúe en tiempo real otras fuentes de ruido acústico sin importar sus características espectrales. La Figura 5.1 muestra el esquema funcional de tal sistema.

La utilidad global del sistema no sólo depende del número de decibeles que disminuya las señales indeseadas, sino también otros aspectos como del ancho del haz, la distancia mínima a la que deben estar las fuentes de sonido del arreglo de micrófonos, la consistencia de la respuesta en todo el intervalo de frecuencias y que funcione en tiempo real. En las siguientes secciones se mostrará cómo se construyó tal sistema.

5.1 Diseño del formador de haz

Se decidió usar la técnica del formador de haz por sub-bandas usando filtros digitales de respuesta finita al impulso (FIR) debido a su fácil diseño e implementación, además de su gran estabilidad y fase lineal. Usar la técnica antes mencionada significa que se tendrán un número J de sub-arreglos lineales de micrófonos contenidos en la misma apertura; para optimizar el uso de los sensores, cada micrófono puede estar incluido en uno o más sub-arreglos.

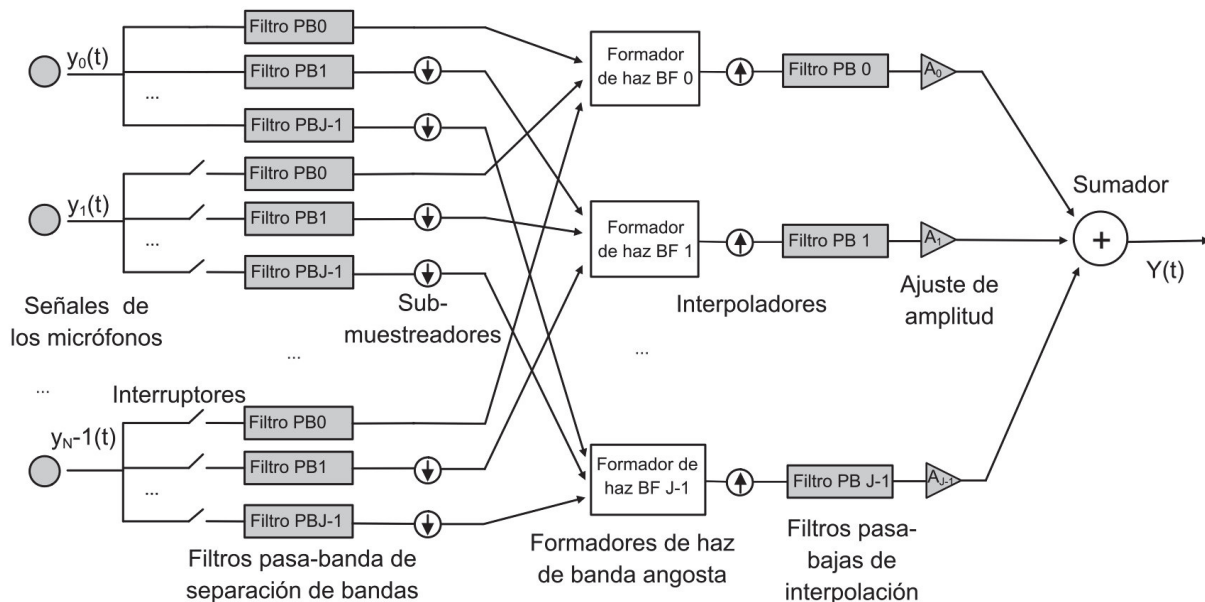


Figura 5.2 Esquema general de un formador de haz de banda ancha por el método de sub-bandas.

Las señales recibidas por los micrófonos se limitarán en banda por filtros FIR dependiendo a qué sub-arreglo o sub-arreglos pertenezcan, una vez separadas las componentes frecuenciales de la señal, se submuestrearán las señales de forma que se tenga una frecuencia de muestreo menor sin perder información y se calcularán los J formadores de

haz de banda angosta. Una vez calculadas las salidas de los formadores de haz, se interpolan estos resultados de tal forma que las J señales filtradas espacialmente regresen a la misma frecuencia de muestreo y se puedan sumar para formar una señal de salida total. La Figura 5.2 muestra un esquema de los procesos que componen un formador de banda ancha por el método de sub-bandas. Los interruptores antes de cada filtro pasa-banda significa que ese micrófono puede o no pertenecer a cierta sub-banda. Cada formador de haz de banda angosta procesa un intervalo de frecuencias diferente y se pueden diseñar con ventanas distintas.

5.1.1 Restricciones y parámetros de diseño

Para empezar a diseñar el formador de haz se contemplaron limitaciones ligadas a tres aspectos generales del sistema:

- Características del DSP
- Características físicas del entorno de pruebas
- Características de las señales de voz.

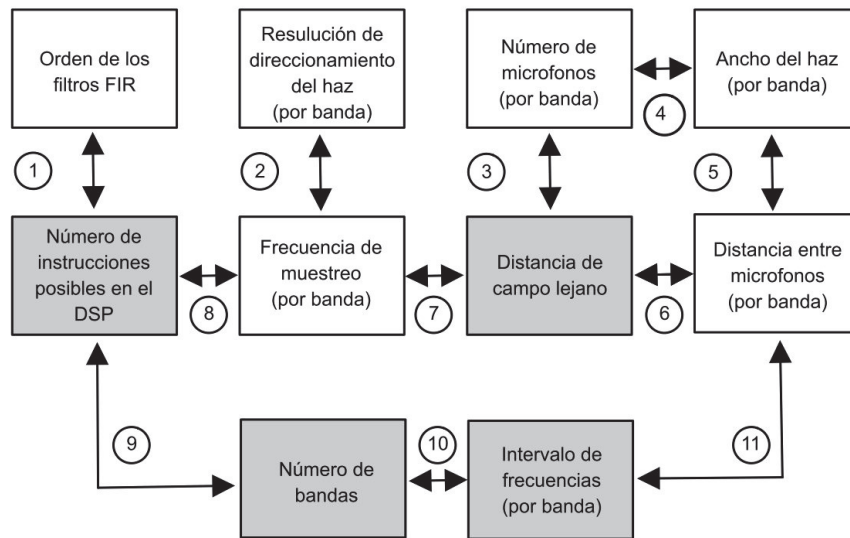


Figura 5.3 Restricciones y parámetros de diseño del sistema.

Cada uno de estos aspectos se relacionan entre sí como lo muestra la Figura 5.3, donde cada número indica una ecuación o definición que liga a dos bloques que contienen aspectos distintos del formador de haz, los cuales se detallan a continuación. Los cuadros en color gris corresponden a los parámetros que se les propuso un valor como requisito en el diseño del sistema.

Relaciones entre las restricciones y los parámetros de diseño:

1 Los algoritmos para calcular los filtros digitales son la actividad más demandante computacionalmente, debido a la gran cantidad de operaciones multiplicación-acumulación que involucra su implementación, además los filtros FIR necesitan una ventana de datos de longitud $L + 1$ donde L es el orden del filtro. Como la capacidad cómputo y la cantidad de memoria disponible para almacenar las ventanas de datos es finita, el orden de los filtros no debe ser arbitrariamente alto.

2 El periodo de muestreo T_s de la señal es la unidad indivisible con la que se expresan los vectores de direccionamiento; si $f_s = \frac{1}{T_s}$ tiene un valor escasamente superior a la frecuencia de Nyquist los vectores de direccionamiento en cierto intervalo continuo de direcciones tendrán el mismo vector de direccionamiento, por lo que se puede pensar que el haz pierde resolución de direccionamiento, en cambio, si el f_s es muchas veces (diez o más) el valor de la frecuencia de Nyquist intervalos de ángulos de arriba más pequeños tendrán un vector de direccionamiento distinto y el haz tendrá buena resolución de direccionamiento.

3 Durante todos los desarrollos anteriores se ha considerado que el frente de onda de la señal es plano. Las ondas de sonido en realidad se propagan de forma esférica por lo que sólo se puede considerar que una onda de sonido tiene un frente de onda plano si [11]

$$r > \frac{d_{tot}^2 f_s}{c} \quad (5.1)$$

donde r es la distancia de la fuente de sonido al arreglo de micrófonos, d_{tot} es la longitud total del arreglo, f_s es la frecuencia de muestreo y c la velocidad de propagación de la señal; para este trabajo se consideró la velocidad del sonido como $340 \frac{m}{s}$. Por lo tanto, entre más micrófonos existan en el arreglo, la distancia d_{tot} aumenta y la distancia r debe ser más grande para que la teoría antes vista sea válida. Esto implica que, de acuerdo a la distancia r a la cual se deba diseñar el sistema, el número de micrófonos no debe ser arbitrariamente alto.

4, 5 Como se expresó en el capítulo anterior mediante la ecuación (4.6) para obtener un lóbulo principal angosto sin presentarse lóbulos gratinados se debe cumplir que la distancia entre micrófonos $d \leq \frac{\lambda}{2}$ y el número de micrófonos debe ser máximo.

6 Como

$$d_{tot} = (N - 1)d_j \quad (5.2)$$

donde N es el número de micrófonos en un sub-arreglo dado y d_j la distancia entre micrófonos en ese sub-arreglo, d_j afecta directamente a la distancia de campo lejano r según las ecuaciones (5.1) y (5.2).

7 La frecuencia de muestreo f_s afecta directamente a la distancia de campo lejano r según la ecuación (5.1).

8 Debido a que el sistema funcionará en tiempo real, todos los procesos relacionados con el filtrado espacial se deben computar dentro del tiempo de muestreo T_s , por lo que

se dispone de un número finito de instrucciones en cada ciclo para computar todos los algoritmos.

9 Al aumentar el número de bandas aumenta el número de filtros digitales, formadores de haz de banda angosta, decimadores e interpoladores, esto provoca un aumento en el número de operaciones a computar dentro del periodo T_s .

10 A cada sub-banda se le asocia un intervalo de frecuencias donde un formador de haz de banda angosta es razonablemente eficiente, si el intervalo de frecuencias asociado a cada sub-banda es pequeño se necesitarán muchas sub-bandas para abarcar el ancho de banda de la señal de voz.

11 Cada sub-arreglo se diseña para implementar un formador de banda angosta eficaz en filtrar espacialmente cada sub-banda. Como la distancia entre micrófonos $d_j = \frac{\lambda_j}{2}$, las frecuencias asociadas al intervalo de la sub-banda definen la distancia d_j .

Como todos estos aspectos están relacionados se inició fijando como parámetros los cuadros de color gris de la Figura 5.3; debido a que se pretende que funcione para lugares como auditorios, salas de grabación y salones de clase se consideró que la distancia del arreglo de micrófonos a la fuente de sonido no excediera las cuatro metros. Además, al tratarse de señales de voz se definió el intervalo de interés de 250 a 4300 Hz , el cual es incluso mayor que el considerado estándar en telefonía. Para maximizar el uso de micrófonos conviene que cada sensor pertenezca a varias sub-bandas simultáneamente, lo cual se puede lograr con facilidad si los sub-arreglos están relacionados armónicamente. Por tanto se decidió separar el intervalo contemplado como lo muestra el Cuadro 5.1.

Cuadro 5.1 Propuesta de la distribución de las sub-bandas.

Número de banda	Intervalo de frecuencias Hz	$\frac{\lambda_i}{2}$ de la frecuencia más alta m
1	250-500	0.34
2	500-1000	0.17
3	1000-2000	0.085
4	2000-4000	0.0425

Además, se consideró que una frecuencia de muestreo de cuatro veces la frecuencia más alta en el intervalo daría una resolución aceptable para la aplicación contemplada y se fijó en $T_s = 62 \mu s$, $f_s = 16129 Hz$ lo que implica que, usando el DSP *TMS320F28335* de *Texas Instruments* el cual hace un ciclo de instrucción en $6.67 ns$, se cuenta con 9295 ciclos de instrucción para terminar todos los procesos en tiempo real.

5.1.2 Cálculo de los parámetros del formador de haz

Con los parámetros de diseño propuestos se puede continuar con el diseño del formador de haz. Para obtener el número de micrófonos por banda se debe considerar la distancia r que ya se ha fijado a cuatro metros y el ancho del haz que debe ser mínimo, por lo tanto, se debe tener tantos micrófonos como lo permita la distancia $r = 4 m$. Si se

considera colocar ocho micrófonos en cada banda con una frecuencia de muestreo igual en todas las sub-bandas $f_s = 16129$ Hz, queda el Cuadro 5.2.

Cuadro 5.2 *Distancia de campo lejano con cuatro sub-bandas y ocho micrófonos en cada sub-arreglo.*

Número de banda	Intervalo de frecuencias Hz	$\frac{\lambda}{2}$ de la frecuencia más alta m	Número de micrófonos	Espacio entre micrófonos d_j m	Distancia de campo lejano r m
1	250-500	0.34	8	0.34	268.7
2	500-1000	0.17	8	0.17	67.1
3	1000-2000	0.085	8	0.085	16.8
4	2000-4000	0.0425	8	0.0425	4.2

Como se observa en el Cuadro 5.2 para frecuencias bajas la distancia de campo lejano se eleva a un valor muy por encima del requerido y sólo la sub-banda cuatro tiene un valor aceptable. Para disminuir estos valores se pueden tomar las siguientes acciones: disminuir el número de micrófonos en las sub-bandas que lo requieran, disminuir el espacio entre micrófonos o disminuir la frecuencia de muestreo a la que se procesa cada sub-banda. Como se ha mencionado anteriormente, cada sub-banda procesa únicamente una porción del ancho de banda de la señal, por lo que cada sub-banda se puede procesar en distintas frecuencias de muestreo con la única restricción que se siga cumpliendo con el teorema de Nyquist. Considerando el efecto del submuestreo en la frecuencia, ajustando ligeramente la distancia entre micrófonos a $d_j = 0.82 \frac{\lambda_j}{2}$ y la restricción de campo lejano a $r < 4.2$ m, se llega al Cuadro 5.3.

Cuadro 5.3 *Distancias de campo lejano para cada banda con sub-muestreo.*

Número de banda	Intervalo de frecuencias Hz	Frecuencia de muestreo Hz	Número de micrófonos	Espacio entre micrófonos d_j m	Distancia de campo lejano r m
1	250-500	2016	4	0.28	4.18
2	500-1000	4032	5	0.14	3.72
3	1000-2000	8064	7	0.07	4.18
4	2000-4000	16129	9	0.035	3.72

Para un formador de haz con estas características se pueden calcular los vectores de direccionamiento simplemente calculando el retardo en tiempo que necesita la señal para arribar a un micrófono i respecto al micrófono cero de referencia y poniendo este valor en términos del periodo de muestreo de cada banda T_s . Dividiendo la ecuación (4.1) del capítulo anterior entre T_s se llega a

$$\delta_n = \frac{d_j n \text{sen} \theta}{T_s c}$$

donde δ_n es el retraso del n-ésimo micrófono necesario para apuntar el haz en la dirección θ , d_j es la distancia entre micrófonos de cada sub-banda, T_s es el periodo de muestreo de cada sub-banda y c es la velocidad del sonido. Como los sub-arreglos están relacionados armónicamente al igual que su frecuencia de muestreo, el cociente $\frac{d_j}{T_s} = 564.51$ que es constante. En consecuencia, bajo estas circunstancias todas las sub-bandas pueden usar los mismos vectores de direccionamiento y sólo cambia el número de micrófonos que poseen. Si los vectores de direccionamiento se acomodan dentro de una matriz, queda como se muestra en el Cuadro 5.4.

Cuadro 5.4 Matriz de retardos por micrófono y dirección dados en T_s .

Retardos T_s									Ángulo de incidencia
0	0	0	0	0	0	0	0	0	0°
0	0	1	1	1	1	2	2	2	10°
0	1	1	2	2	3	3	4	5	20°
0	1	2	2	3	4	5	6	7	30°
0	1	2	3	4	5	6	7	9	40°
0	1	3	4	5	6	8	9	10	50°
0	1	3	4	6	7	9	10	12	60°
0	2	3	5	6	8	9	11	12	70°
0	2	3	5	7	8	10	11	13	80°
0	2	3	5	7	8	10	12	13	90°

De donde se observa que para apuntar a la dirección 0° las señales de los micrófonos no deben retrasarse, si se quiere apuntar a 90° la señal del micrófono cero se mantiene sin retardo, la del uno se retrasa $2T_s$, la del dos se retrasa $3T_s$, la del tres se retrasa $5T_s$ y así sucesivamente. Para ángulos negativos estos vectores se invierten, de tal forma que para apuntar a -90° la señal del micrófono cero se retrasa $13T_s$, la del dos se retrasa $12T_s$, la del tres se retrasa $10T_s$, etcétera. También se observa que todos los vectores de dirección son diferentes pero para 80° y 90° apenas son distintos en dos números por lo que en los extremos la resolución de direccionamiento del haz ronda los 10° .

De acuerdo al diseño antes presentado, se construyó un arreglo lineal de 13 micrófonos electret con una distribución como lo muestra el diagrama de la Figura 5.4, donde los números debajo de cada sensor representa la sub-banda o sub-bandas a la cual pertenece tal micrófono. El micrófono cero pertenece a todas las bandas y es el micrófono que sirve de referencia para orientar el haz.

En este formador de haz de banda ancha por sub-bandas los filtros FIR cumplen dos funciones principales, separar la señal de voz en intervalos y realizar la interpolación de las señales sobre-muestreadas de forma que regresen a una señal digital con frecuencia de muestreo f_s . Con el diseño antes mencionado y tomando en cuenta la distribución de sensores en las sub-bandas como lo muestra el Cuadro 5.3 se usan 28 filtros FIR distribuidos de la siguiente forma:

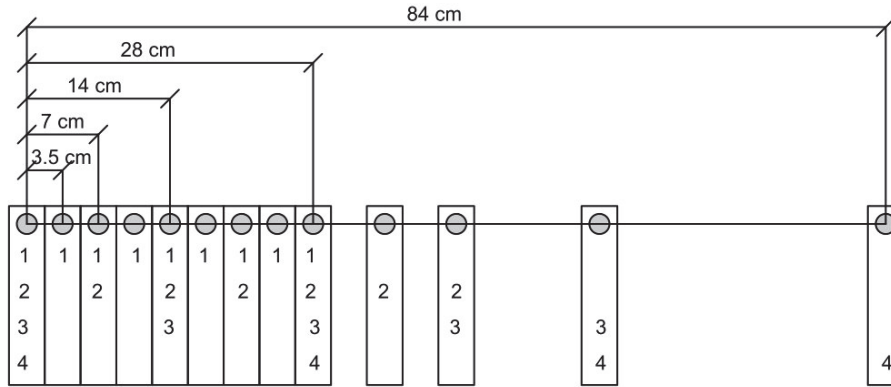


Figura 5.4 Esquema de la distribución de micrófonos.

- Nueve filtros pasa-banda para separar la banda 1
- Siete filtros pasa-banda para separar la banda 2
- Cinco filtros pasa-banda para separar la banda 3
- Cuatro filtros pasa-banda para separar la banda 4
- Tres filtros pasa-bajas para interpolar el resultado del formador de haz de las bandas 2, 3 y 4 después del sobre-muestreo.

La banda 1 es donde se procesan las frecuencias más altas y se procesa a la frecuencia de muestreo $f_s = 16129$ Hz por lo tanto no se debe sub-muestrear, sobre-muestrear ni interpolar. Para las ventanas se decidió usar las ventanas Chebyshev-Dolph diseñada a 30 dB para las sub-bandas 1, 2 y 3, mientras que para la sub-banda 4 se diseñó a 10 dB. Esta combinación de ventanas produce una fuerte atenuación de lóbulos laterales al mismo tiempo que mantiene en frecuencias bajas el ancho del haz tan pequeño como sea posible. Los valores de los coeficientes de las ventanas son los siguientes

$$w_{sub-banda1}(n_1) = [0.2527 \quad 0.4589 \quad 0.7194 \quad 0.9229 \quad 1 \quad 0.9229 \quad 0.7194 \quad 0.4589 \quad 0.2527]$$

$$w_{sub-banda2}(n_2) = [0.2642 \quad 0.5683 \quad 0.8738 \quad 1 \quad 0.8738 \quad 0.5683 \quad 0.2642]$$

$$w_{sub-banda3}(n_3) = [0.3185 \quad 0.7683 \quad 1 \quad 0.7683 \quad 0.3185]$$

$$w_{sub-banda4}(n_4) = [1 \quad 0.8794 \quad 0.8794 \quad 1].$$

5.2 Implementación del formador de haz en el DSP

Para el desarrollo de esta tesis se compró una plataforma de desarrollo de Texas Instruments llamada *TMS320F28335 Experimenter Kit*, el cual consta físicamente de dos partes principales [33], la *Control card*, que es una tarjeta de tamaño aproximado a 9×2.5 cm de 100 pines donde está el DSP y la *Docking station* que es una base para dicha tarjeta y es donde se encuentra el emulador XDS100 para la comunicación JTAG, además posee conectores *headers* para tener un fácil acceso a los pines del DSP. El DSP TMS320F28335 tiene las siguientes características principales [27]:

- Comunicación *Joint Test Action Group* JTAG.
- Entrada de voltaje con puerto USB o fuente externa, regulador de voltaje interno a 5 V y 3.3 V.
- Reloj principal a 150 MHz.
- 150 millones de instrucciones por segundo (MIPS).
- Multiplicador por hardware de 32×32 en punto fijo y multiplicador por hardware de punto flotante IEEE-754 de precisión simple.
- 256 KB \times 16 de memoria FLASH y 34 KB \times 16 de memoria SARAM.
- Hasta 88 entradas/salidas de propósito general.
- Hasta 18 salidas PWM de los cuales seis pueden configurarse como PWM de alta resolución (HRPWM).
- ADC con 16 canales y 12 bits de resolución.

Texas Instruments provee, junto con el hardware de sus microcontroladores y DSPs, un ambiente de desarrollo específicamente diseñado para programar sus dispositivos llamado *Code Composer Studio (CCS)*; esta tesis se programó usando la versión 4.1 de dicho software. La comunicación entre el DSP y el CCS se mantiene mediante el módulo emulador XDS100, el cual provee acceso JTAG al DSP desde un puerto USB en la PC [33]. Dicha comunicación puede mantener acceso en tiempo real a memoria y registros de control en el DSP, además de poder contar ciclos de reloj entre dos segmentos de código, graficar segmentos de memoria, ver código desensamblado, entre otras características.

En la Figura 5.5 se muestra la interfaz gráfica del CCS la cual está basada en el ambiente de desarrollo con licencia libre *Eclipse* con las ventanas esenciales para monitorear el comportamiento del sistema, las cuales se componen como sigue:

1. Navegador de proyectos. Organiza todos los archivos que pertenecen a un proyecto con extensiones .c, .asm, .cmd, .obj o cualquier otro.

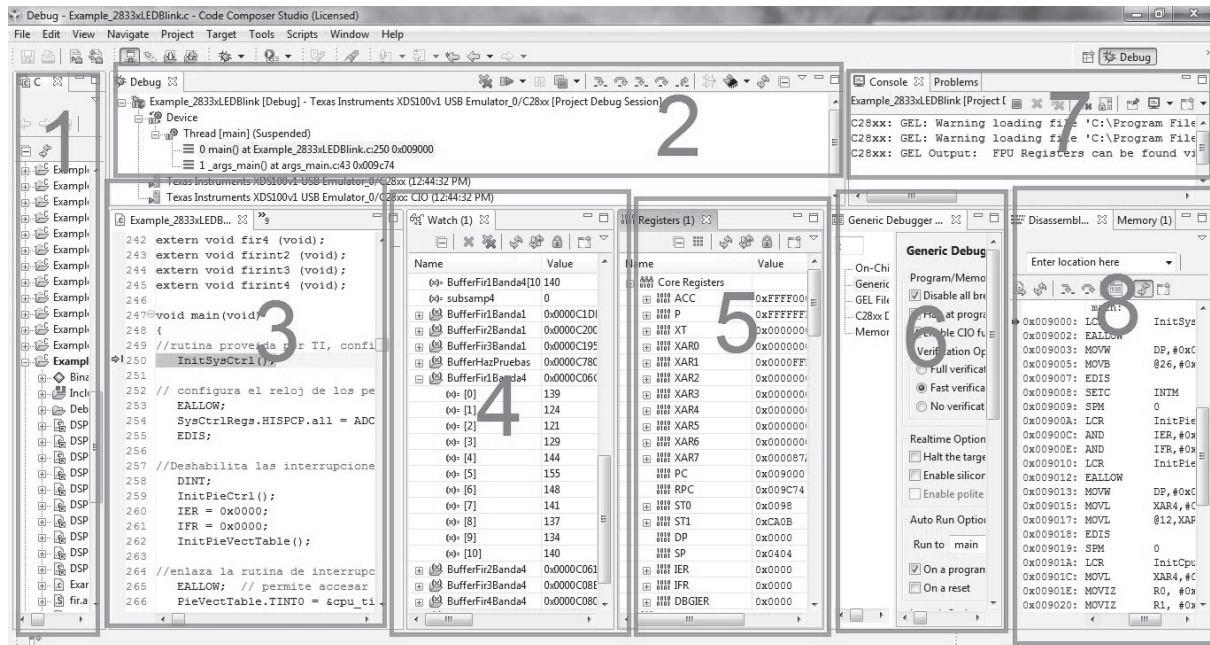


Figura 5.5 Ambiente de desarrollo Code Composer Studio v 4.1.

2. Panel de control de una sesión de depuración. Cuando se está en una sesión de depuración o *debug*, permite manipular la forma de ejecución del programa en el DSP pausando, reiniciando, ejecutando paso a paso, etcétera.
3. Ventana de edición. Es la ventana donde se tiene acceso a los archivos del proyecto, en la Figura se muestra el programa en lenguaje C.
4. Ventana de monitoreo. Sirve para observar y modificar las variables en memoria RAM.
5. Ventana de registros. Permite monitorear, y en algunos casos modificar los registros del DSP ya sean del CPU, interrupciones, módulo ADC, módulo PWM, entre otros.
6. Panel de configuración de la sesión de depuración. Permite elegir el grado de interacción de la PC y el DSP en una sesión de depuración.
7. Ventana de notificaciones. Muestra la actividad del CCS y notificaciones respecto a la compilación de código o la comunicación con el DSP.
8. Ventana de código desensamblado. Muestra el código en ensamblador tal como se cargó en el DSP.

5.2.1 Adquisición de datos

El DSP TMS320F28335 cuenta con un módulo convertidor digital a analógico (ADC) de 12 bits que puede convertir un voltaje analógico entre $0 V$ y $3 V$. El módulo ADC puede convertir hasta 16 canales usando dos circuitos muestreador-retenedor y un núcleo de conversión. Este módulo puede configurarse para usarse como dos módulos independientes de ocho canales o como uno solo de 16 canales. El comienzo de la conversión o *start of conversion (SOC)* de cada canal puede iniciarse mediante interrupciones externas, el módulo de modulación por ancho de pulso (PWM), por software o incluso mediante la bandera de término de conversión *End of conversion (EOC)* de algún otro canal [30]. En esta tesis se usó esta última forma de iniciar la conversión, de modo que cuando acabe la conversión del primer micrófono se inicia la conversión del segundo y así sucesivamente. Cuando termina la conversión del micrófono 13 se inicia la del micrófono 1, por lo que el módulo ADC está digitalizando las señales de los 13 micrófonos todo el tiempo, almacenando los resultados en los registros del CPU llamados *ADCRESULT3 - ADCRESULT15*. Además se colocó un potenciómetro para interactuar con el programa y mover la dirección del haz sin pausar el programa; esta señal también se digitaliza con el módulo ADC y su valor se almacena en *ADCRESULT0*.

Por otro lado, la tasa de muestreo fija a $f_s = 16129 Hz$ se logra usando la interrupción del *Timer0*. Cuando el *Timer0* lanza una interrupción al CPU se inicia una subrutina que contiene el algoritmo de filtrado espacial, cuando esto ocurre se capturan los valores que en ese momento tengan los registros *ADCRESULT3 - ADCRESULT15* y el resto del proceso comienza.

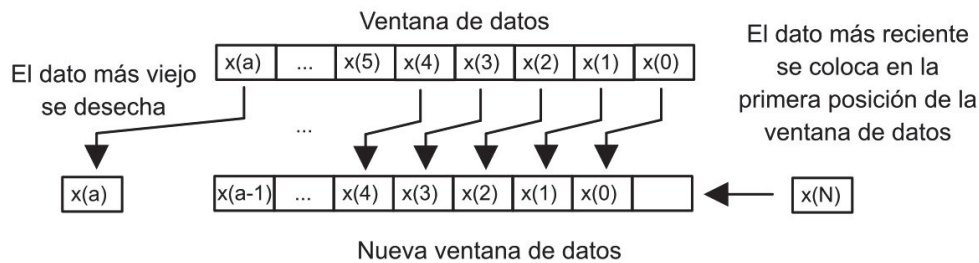


Figura 5.6 Recorrido de la ventana de datos.

En una aplicación en tiempo real se tiene un flujo de datos entrando y saliendo constantemente del sistema. Debido a que la memoria de una máquina digital no es infinita, no es posible almacenar los datos de toda la señal, en cambio, sólo se conserva un conjunto finito de datos que provienen de un intervalo en el tiempo que generalmente se le denomina *ventana de datos*. Entonces, los datos que se procesan en cada periodo de muestreo T_s son una colección finita de datos inmediatos anteriores, con cada T_s se agregan los datos nuevos y se desechan los datos viejos. Cuando se ingresa un dato nuevo a memoria lo más común es recorrer todas las muestras anteriores para dar espacio al nuevo dato como se muestra en la Figura 5.6.

Cuando las ventanas de datos son muy grandes o se tiene un gran número de ellas, este proceso primordial puede llegar a ser muy demandante e ineficiente. Una opción más adecuada es usar direccionamiento circular en el cuál se considera que la ventana de datos no tiene una dirección de inicio o fin, en cambio funciona como una lista circular.

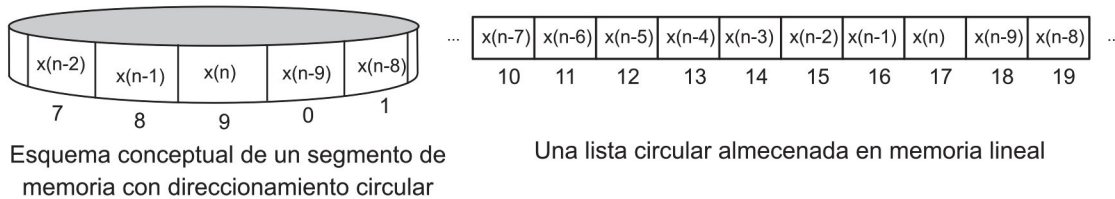


Figura 5.7 Esquema conceptual de memoria con direccionamiento circular.

En la Figura 5.7 se muestra conceptualmente una lista circular y una lista circular en memoria lineal, donde los números debajo de cada dato representan la dirección en memoria de cada celda. Como se observa, en la memoria circular el dato más reciente siempre tendrá la última dirección y para recorrerlo existe una trayectoria ininterumpida, en cambio cuando una lista circular se encuentra en memoria lineal, la lista está segmentada en dos partes y para recorrer la lista es necesario saber dónde está el último dato, la longitud de la lista y la dirección de inicio del vector.

En la arquitectura del DSP se considera este tipo de direccionamiento y es posible usar los apuntadores por hardware llamados *XAR1* y *XAR6* para utilizar segmentos de memoria como si tuvieran direccionamiento circular con un tamaño máximo de 256 palabras [28]; sin embargo, se consideró que implementar el manejo de las listas circulares directamente sobre C podría ser más sencillo y que las operaciones adicionales no implicarían una carga considerable al DSP, por lo que no se usó el modo de direccionamiento indirecto con buffer circular, en cambio, se programó directamente en C haciendo uso de contadores y condicionales.

Manejar las variables de forma que emulen un segmento de memoria circular reduce el número de instrucciones requeridas en cada periodo T_s y no agrega muchas operaciones extra, para el manejo de dichas listas es suficiente saber la dirección de inicio del vector, el número de elementos del vector y la dirección donde se puso el último dato agregado, como muestra la Figura 5.8, el dato nuevo se agrega a la localidad siguiente inmediata del último dato que se agregó, si esa dirección está fuera del vector entonces se agrega a la primera localidad de la variable. Aunque se tuvieron que agregar algunas operaciones extra para manejar las direcciones de inicio y fin del vector de datos, resultó muy conveniente implementar este concepto en el programa.

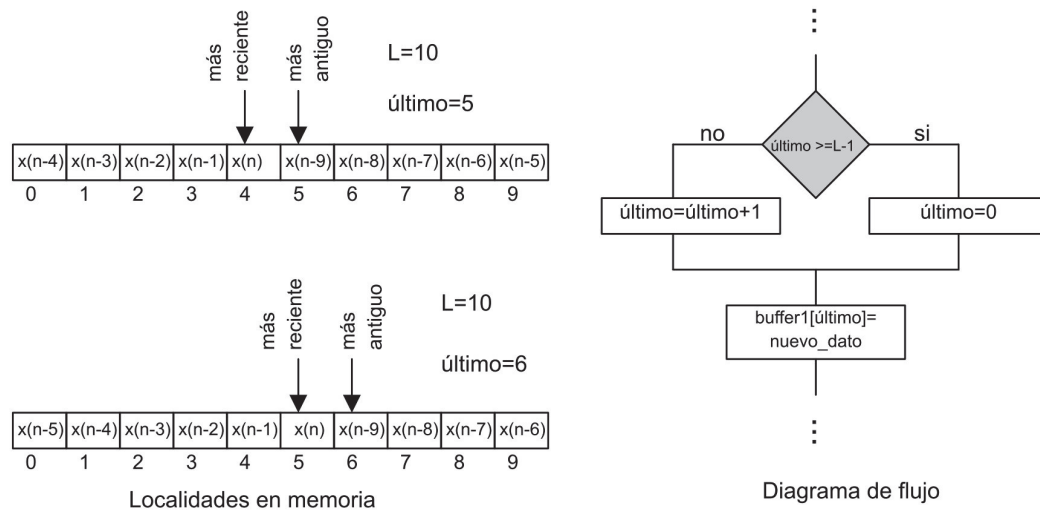


Figura 5.8 Manejo de listas circulares en memoria lineal.

5.2.2 Cómputo de los filtros FIR

Como muchas otras operaciones en el procesamiento digital de señales, el filtrado con filtros FIR son una serie de movimientos en memoria, multiplicaciones y sumas consecutivas. A pesar de la simplicidad de cada una de estas operaciones, la gran cantidad de datos a procesar hacen que estos algoritmos ocupen la mayor parte de los 9295 ciclos de instrucción disponibles. Aunque el ambiente de desarrollo Code Composer Studio tiene métodos para traducir eficientemente las instrucciones en lenguaje C a ensamblador, no es tan eficiente como para poder terminar todos los procesos en tiempo real, por ello, el cálculo de los filtros FIR para la separación de las bandas y los filtros FIR para el proceso de interpolación se programaron directamente en lenguaje ensamblador, disminuyendo más de seis veces el tiempo necesario para calcularlos.

Cada rutina en ensamblador se programa en un archivo independiente *.asm* y se llama desde el programa principal o *main* en C como una subrutina externa, haciendo posible comunicar las direcciones en memoria de las variables de C a las rutinas de ensamblador y a su vez las rutinas de ensamblador modifican el contenido de tales direcciones en la memoria del DSP.

Operativamente el cálculo de un filtro FIR es el producto punto de dos vectores de magnitudes iguales, uno es la ventana de datos y el otro el vector de coeficientes del filtro en el tiempo de muestreo n . Tal como lo muestra la Figura 5.9, normalmente sería suficiente con colocar apuntadores en la dirección de inicio de cada vector e incrementar en uno cada vez que se hace una multiplicación-acumulación; sin embargo, cuando las ventanas de datos están acomodadas para emular el direccionamiento circular se debe tener un manejo de los apuntadores distinto, debido a que el vector de datos de la señal no está guardado en forma consecutiva, por ello es importante tener identificado en todo momento en qué localidad se colocó el último dato.

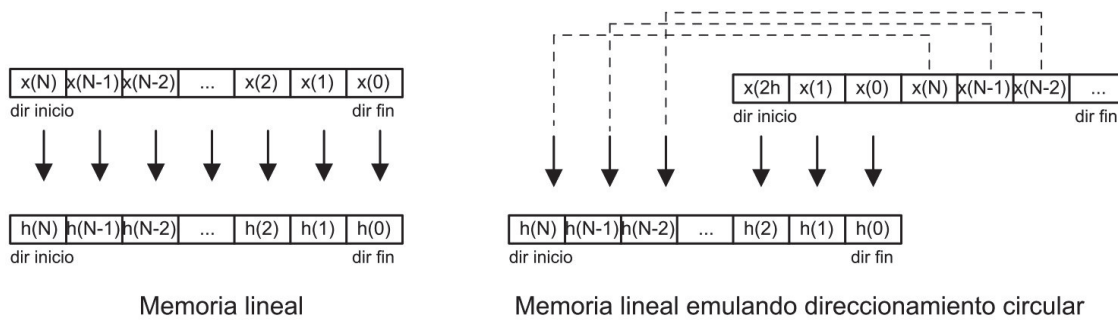


Figura 5.9 Diferencia en el cálculo de filtros FIR con buffers lineales y buffers circulares.

Para obtener la mayor aproximación numérica, los coeficientes de los filtros FIR y los coeficientes de las ventanas se representaron en formato de punto fijo Q_{15} . A pesar de que el DSP tiene implementado en hardware multiplicaciones de punto flotante se decidió usar sólo la unidad de punto fijo empleando variables de 16 bits, lo que se justifica con base en lo siguiente: las variables de tipo flotante usan 32 bits para ser almacenadas lo que implica un aumento en la memoria requerida, además, algunas de las instrucciones en ensamblador de la unidad de punto flotante usan dos *ciclos de pipeline* lo que significa que se puede empezar una instrucción cada ciclo de reloj pero el resultado estará disponible hasta un ciclo de reloj después; esto implica que en ocasiones las instrucciones deben ser seguidas por una instrucción *NOP* para evitar conflictos en la sincronía del pipeline [31]. En general se puede decir que usar punto flotante consume algunos ciclos de máquina extra, la programación en ensamblador es más complicada y las variables necesitan más memoria. Sin duda, la ventaja de usar punto flotante es la mayor precisión numérica a la que se están operando las variables pero este sistema está limitado por la precisión de los datos de entrada y salida, debe recordarse que el convertidor ADC que digitaliza las señales de audio es de 12 bits y que el convertidor DAC que transmite la señal filtrada es de 10 bits, por lo que carece de sentido manejar una precisión numérica muy por encima de los datos que se reciben y se transmiten, aunque se decida usar una precisión numérica muy alta se terminaría redondeando el número para normalizarlo a 10 bits y enviarlo al DAC.

Para esta tesis los filtros responsables de la separación de bandas son de orden 29, el de interpolación de la banda 2 es de orden 31, el de la banda 3 es de orden 63 y el de la banda 4 de orden 127. El orden de éstos filtros de separación de bandas en general puede calificarse como bajo pero debe tomarse en cuenta que incrementar el orden de estos filtros implica un incremento de la memoria ocupada por estos vectores de datos así como en las operaciones necesarias para calcularlos; además, la calidad de estos filtros no afecta directamente en el rendimiento del sistema porque su función sólo es separar el ancho de banda de la señal en intervalos donde cada formador de haz de banda angosta sea razonablemente eficiente.

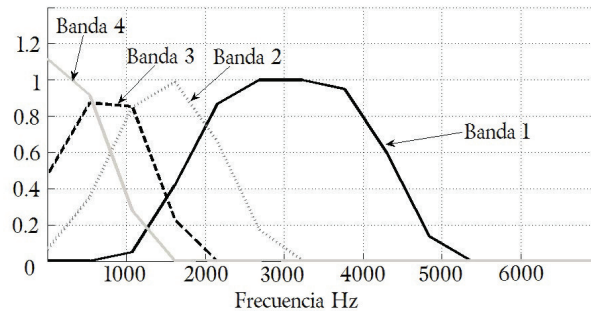


Figura 5.10 Respuesta en frecuencia de los filtros usados para la separación de bandas.

La banda de pasos de los filtros implementados no corresponde exactamente con el intervalo de las sub-bandas porque se observó mediante simulaciones que era mejor que las bandas de transición de los filtros estuvieran encimadas para que al conjuntar las cuatro señales de los formadores de haz de banda angosta, la señal de salida tuviera una respuesta más uniforme en todo el intervalo contemplado. La respuesta teórica de los filtros de separación de bandas se muestra en la Figura 5.10.

5.2.3 Cómputo de los formadores de haz

El cálculo de los formadores de haz consta de dos partes, la primera es seleccionar una muestra retrasada de cada micrófono según indique la matriz de retardos, la segunda parte es hacer el producto punto de las señales retrasadas con el vector de pesos de cada micrófono (ventana).

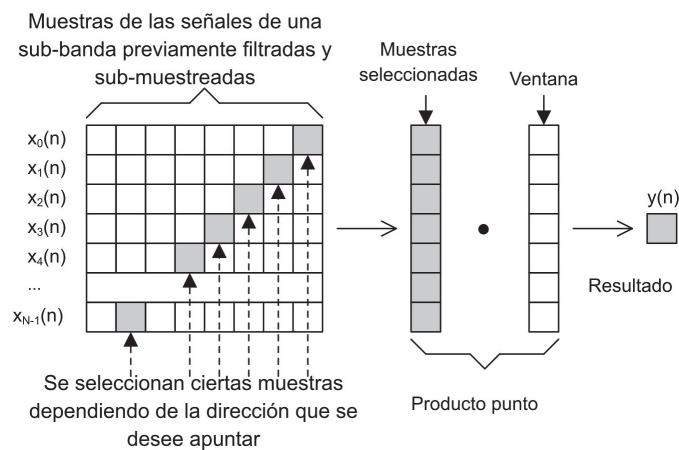


Figura 5.11 Diagrama del cómputo de un formador de haz simple con ventana.

Los formadores de haz incluyendo la ponderación por ventanas, se programaron en lenguaje *C* porque no representan una carga significativa en términos computacionales. La Figura 5.11 muestra conceptualmente el cálculo de los formadores de haz.

5.2.4 Manejo de procesos multitasa

Debido a que no todas las sub-bandas se procesan a la misma tasa de muestreo, se obviaron algunos procesos para eficientar el uso del CPU de modo que el número de instrucciones realizadas en cada periodo T_s no siempre es el mismo. Esta variación en el número de instrucciones se debe básicamente a que no se aplican todos los procesos de filtrado y formación de haz en las señales que debido al sub-muestreo se desecharían.

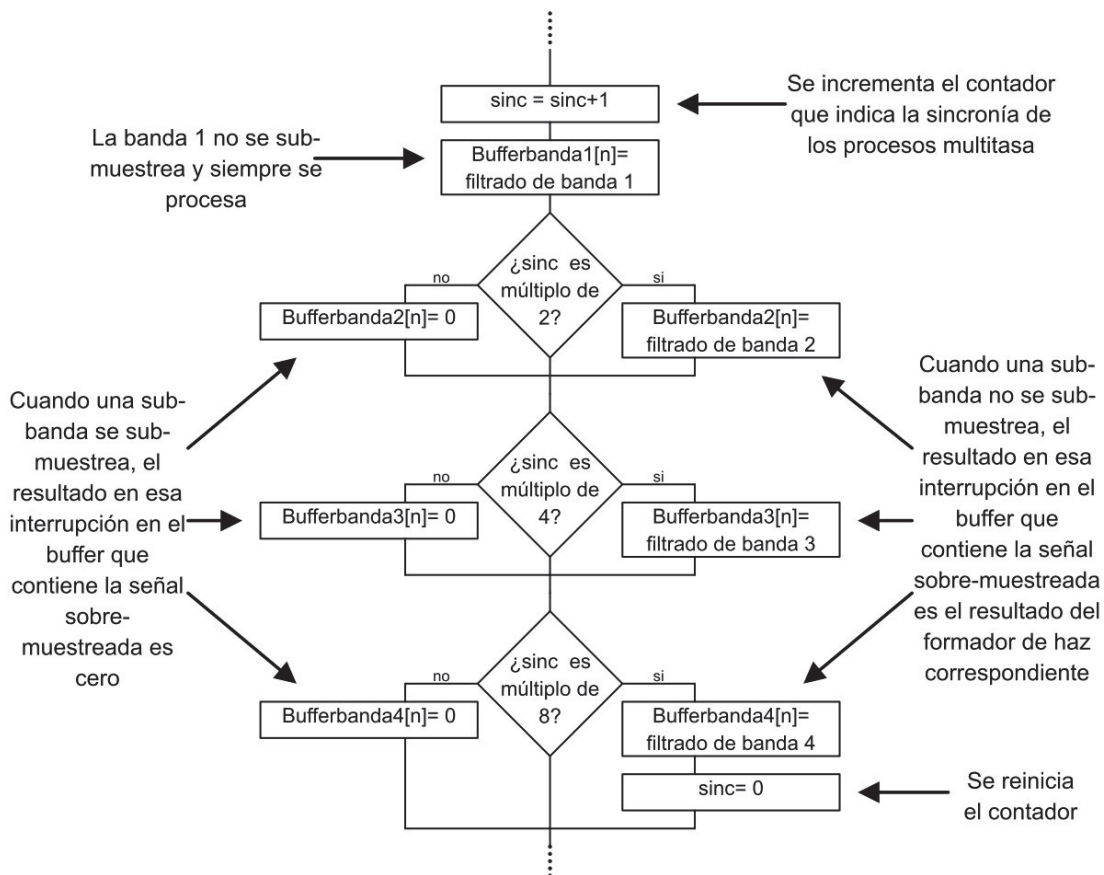


Figura 5.12 Diagrama del manejo de procesos multitasa.

Para mantener la sincronía en los procesos que se llevan a cabo a diferentes tasas simplemente se hace uso de un contador el cual se incrementa en cada interrupción, sabiendo este número se puede inferir que procesos se deben hacer y cuáles no. La Figura 5.12 muestra un diagrama de flujo de esta idea.

5.2.5 Diagrama general del proceso de filtrado

El programa se divide en dos grandes bloques, el programa principal y la subrutina de interrupción del timer0. En el principal se declaran todas las variables y constantes involucradas y se configuran tanto aspectos generales del DSP como de los periféricos involucrados (relojes, PLL, vector de interrupciones, timers, ADC, GPIO, etc), el programa principal termina con un ciclo infinito donde espera que ocurran las interrupciones. En la subrutina de interrupción se aloja el código propiamente del filtrado espacial y es donde se invocan a las subrutinas externas escritas en ensamblador.

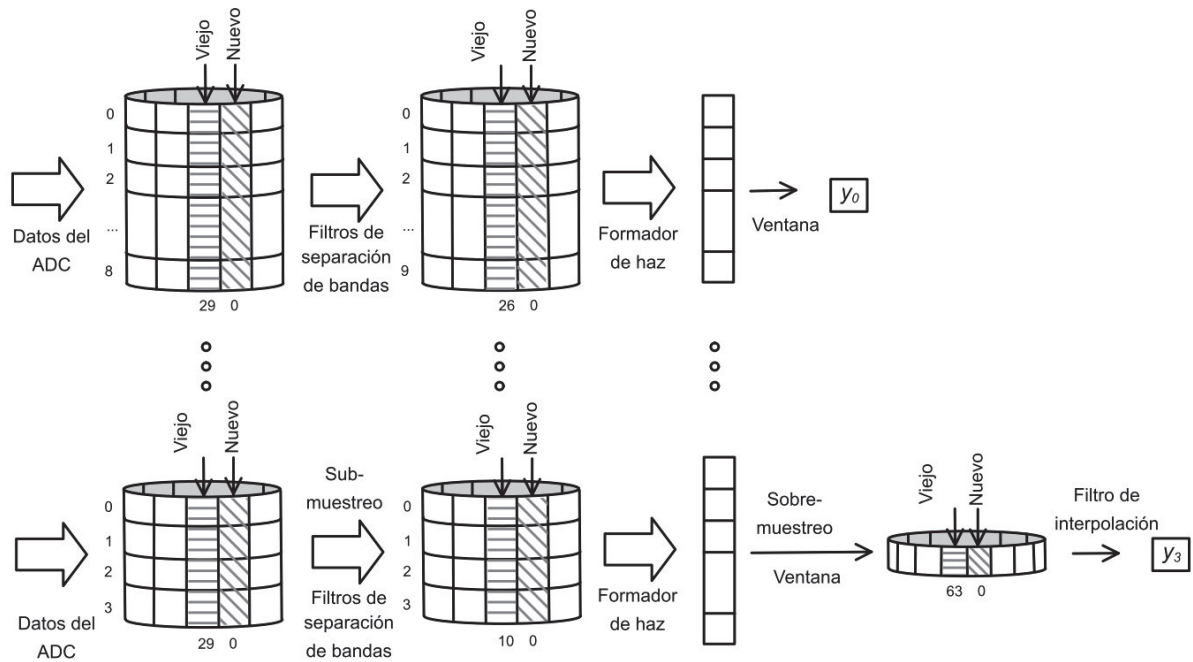


Figura 5.13 Esquema conceptual del programa en el DSP.

La Figura 5.13 muestra un resumen del proceso de filtrado espacial que se lleva a cabo en el DSP en cada periodo T_s para la banda 1 (parte superior) y la banda 4 (parte inferior). Como se observa, las señales provenientes del ADC se almacenan en una ventana de datos de 30 elementos en buffer circular, posteriormente se filtran y simultáneamente se sub-muestran para separar las componentes en frecuencia asociadas con cada formador de haz guardando el resultado en otras ventanas de datos de señales filtradas, estas ventanas difieren en longitud en cada banda. Luego, se seleccionan las muestras a ponderar dependiendo de la dirección que se quiera apuntar y se realiza la ventana de ponderación de las señales de cada micrófono. Como la banda uno no se sub-muestra el resultado del formador de haz de banda angosta es el valor final del proceso, mientras que para la banda cuatro, el resultado del formador de haz se almacena en otro vector al cual se intercalan siete ceros por cada valor del formador de

haz. Finalmente, al vector con la señal sobre-muestreada se le aplica un filtro FIR paso-bajas para interpolar la señal. Una vez que se tienen los cuatro resultados se multiplican por una constante que homogeneiza la amplitud de las cuatro sub-bandas y se suman para formar la señal de salida conjunta.

Adicionalmente, se calcula la potencia de cada señal para tener un valor de referencia y calificar el rendimiento del sistema. Por otro lado, usando el puerto A del DSP se comunica la señal de salida al circuito del convertidor digital-analógico y el valor de la potencia a un microcontrolador PIC18F4550 que controla la pantalla GLCD.

5.2.6 Eficiencia computacional del programa

Como se ha dicho, no en todas las interrupciones se realiza el mismo número de instrucciones. Utilizando la herramienta de conteo de ciclos de reloj en el Code Composer Studio se llenó el Cuadro 5.5 donde muestra el número exacto de ciclos de instrucción usados.

Cuadro 5.5 Número total de instrucciones en cada T_s .

Número de interrupción de timer0	Filtrado de banda 1	Filtrado de banda 2	Filtrado de banda 3	Filtrado de banda 4	Número total de instrucciones	Porcentaje de utilización del DSP
1	Sí	Sí	Sí	Sí	4685	50.4 %
2	Sí	No	No	No	2416	26 %
3	Sí	Sí	No	No	3406	36.6 %
4	Sí	No	No	No	2416	26 %
5	Sí	Sí	Sí	No	4111	44.2 %
6	Sí	No	No	No	2416	26 %
7	Sí	Sí	No	No	3406	36.6 %
8	Sí	No	No	No	2416	26 %

Este esquema se repite periódicamente de tal forma que en la novena interrupción se harían 4685 instrucciones, en la décima 2416 y así sucesivamente.

Considerando la interrupción donde se llevan a cabo todos los procesos de filtrado, la distribución del tiempo empleado en cada proceso se muestra en la Figura 5.14. El proceso de separación de bandas es la parte más demandante del algoritmo y a pesar de estar programada completamente en ensamblador ocupa cerca del 26 % del número de instrucciones disponibles, si se hubiera programado en lenguaje C se requerirían alrededor de 15,000 instrucciones y volvería inviable realizar el proyecto con este DSP. Los formadores de haz con ventana no requieren muchas operaciones para ser calculados, pero como se programaron en lenguaje C ocupan más del 11 % de la capacidad del DSP, si se programara en ensamblador este porcentaje podría bajar a un 2 a 3 %. La adquisición de datos se mantiene al mínimo debido a la emulación de direccionamiento circular en las ventanas de datos y representa un poco más del 3.5 % de T_s . El tiempo

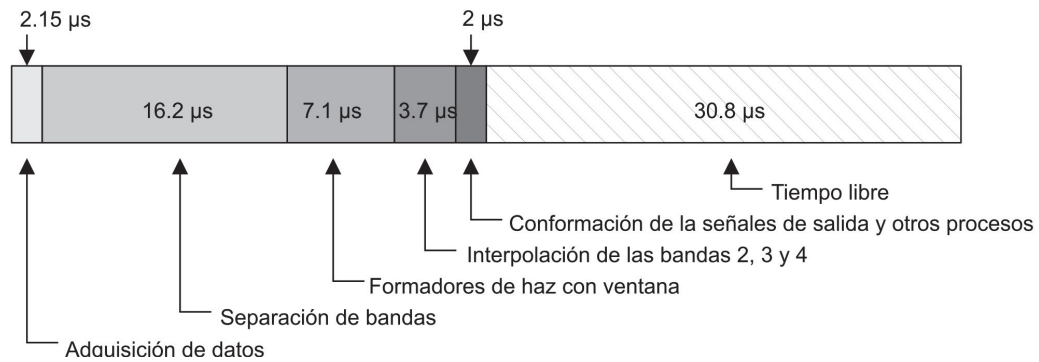


Figura 5.14 Distribución del tiempo usado en cada proceso.

libre, en el cual el CPU no realiza ninguna acción programada y solamente espera la siguiente interrupción del timer, representa el 49% de T_s , por lo que el sistema logra fácilmente trabajar en tiempo real.

5.3 Composición física del sistema

Como lo describe la Figura 5.15, el sistema se compone físicamente de tres bloques principales: el bloque de adquisición de señales, el bloque de la máquina digital (DSP) y el bloque de transmisión de resultados.

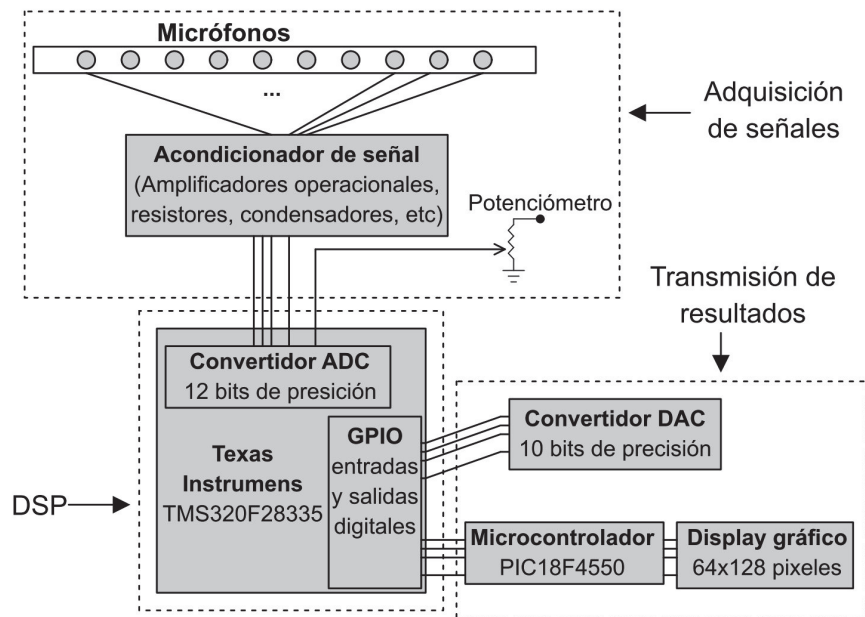


Figura 5.15 Bloques del sistema.

El bloque de adquisición de señales es el responsable de adquirir las señales audibles, capturar la información espacial de las señales de audio y convertirlas a un voltaje adecuado para la conversión en el ADC dentro del DSP; también se introduce la señal que representa la posición del potenciómetro la cual comunica la dirección que el usuario solicita apuntar el haz.

El bloque de la máquina digital se compone solamente del DSP y es donde se realiza todo el procesamiento de las señales de audio.

Finalmente, se añadió el bloque de transmisión de resultados para hacer más rápido y sencillo el proceso de medir el rendimiento del algoritmo y al mismo tiempo agregar facultades que faciliten la ampliación y perfeccionamiento del sistema en trabajos futuros; se implementaron dos formas de comunicar los resultados del algoritmo al exterior del DSP: un convertidor digital a analógico y una pantalla LCD gráfica.

5.3.1 Bloque de adquisición de señales

Para la implementación de la tesis se usaron micrófonos electret por su amplia disponibilidad, fácil uso y bajo costo. Utilizando un calibrador de micrófonos marca *TES* modelo *1356* se determinó que con un tono de 1 KHz a 94 dB (1 Pa) un micrófono como los empleados para construir el arreglo presentaba un voltaje de 35 mV_p lo que equivale a una sensibilidad de -29.1 dB con la referencia de 0 dB en $1\frac{\text{V}}{\text{Pa}}$.

Por otro lado, el DSP incluye un convertidor analógico a digital de 12 bits para señales analógicas entre 0 y 3 V , por lo que no se puede capturar correctamente la señal en los micrófonos si no se cuenta con un circuito externo para polarizar los micrófonos y amplificar su señal.

Lo usual en ese tipo de situaciones donde se debe acondicionar la amplitud y offset de la señal es implementar un circuito analógico usando amplificadores operacionales los cuales son económicos y fáciles de usar. Para el desarrollo de esta tesis se usó un circuito acondicionador de señales para 16 micrófonos electret que se construyó como parte de una tesis previa en el Laboratorio de Procesamiento Digital de Señales [25]. Dicho circuito se construyó para amplificar 174.4 veces la señal de cada micrófono y añadir un offset de 1.5 V cuando se polariza con $\pm 4\text{ V}$, además, cada canal tiene un filtro pasa banda para evitar aliasing cuyas frecuencias de corte son 12 Hz y 23000 Hz [25]. El diagrama eléctrico de cada canal se muestra en el Figura 5.16.

5.3.2 Bloque de transmisión de resultados

Se usó el convertidor digital a analógico (DAC) de 10 bits AD7533 usando la configuración unipolar la cual requiere algunos resistores, capacitores y un amplificador operacional para su funcionamiento. Con dicha configuración el circuito convierte un número digital en un voltaje analógico de la siguiente forma

$$V_{out} = -V_{REF} \left(\frac{num_{DSP}}{1024} \right) \quad (5.3)$$

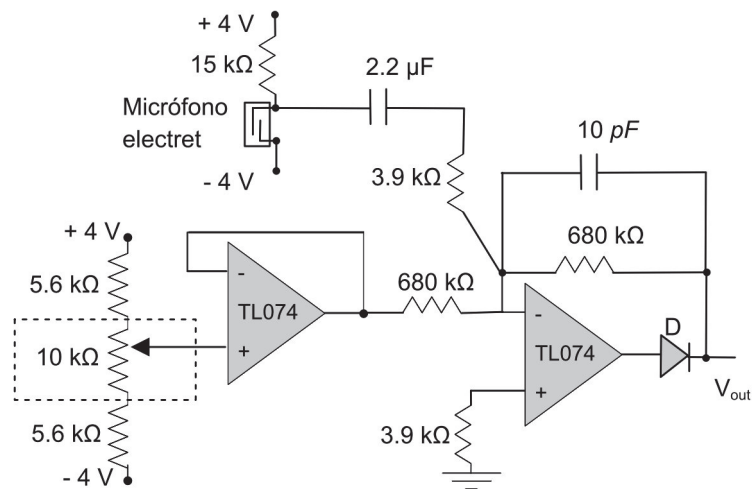


Figura 5.16 Diagrama del circuito acondicionador de señal.

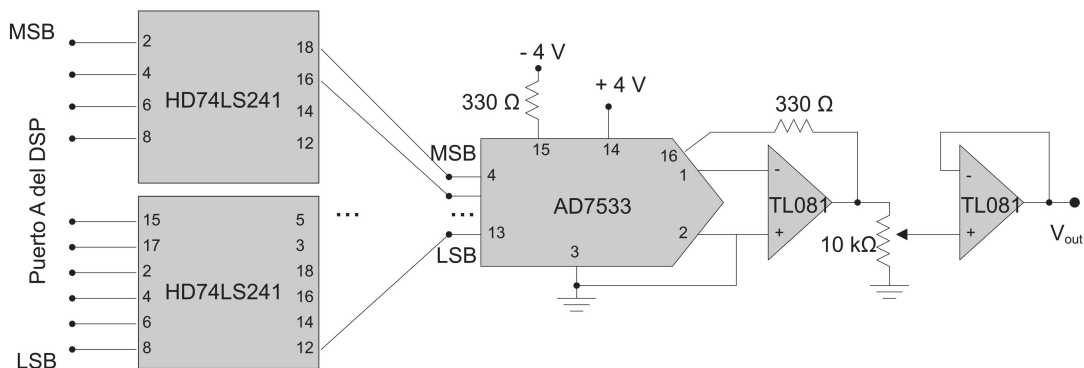


Figura 5.17 Esquema del circuito convertidor digital-analógico.

donde num_{DSP} es un número de 10 bits que proviene del puerto A del DSP.

Para proteger al DSP de algún imprevisto en el DAC, se usaron buffers HD74LS241 como interfaz. Además, para poder variar la amplitud de la señal de salida se usó un potenciómetro y un amplificador operacional en configuración seguidor de voltaje. El esquema general del circuito se presenta en la Figura 5.17.

También se usó una pantalla gráfica retroiluminada de 64×128 pixeles JHD-6221864 la cual es manejada por un PIC18F4550. La decisión de usar un microcontrolador extra sólo para el control de la pantalla se debió a que de esta forma el despliegue de información prácticamente no representa una carga para el DSP, además, existen bibliotecas para los microcontroladores PIC que permiten usar este tipo de pantallas con facilidad. En la Figura 5.18 se muestra el diagrama de las conexiones de este

circuito.

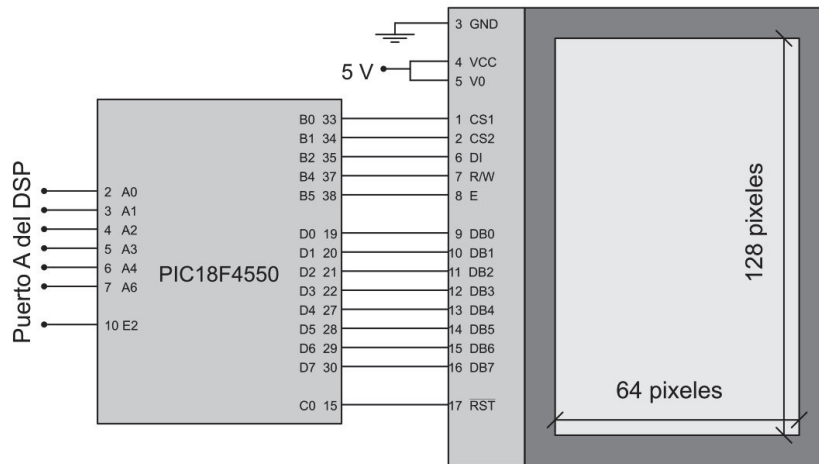


Figura 5.18 Esquema del circuito del microcontrolador PIC y la pantalla LCD gráfica.

El programa dentro del PIC convierte los datos que recibe del DSP a una gráfica en la pantalla de forma que la potencia de la señal proveniente de distintas direcciones pueda ser visualizada sin necesidad de pausar el programa. El PIC recibe siete bits de información en forma paralela, uno de estos bits indica si el valor de los otros seis bits representa una dirección o una potencia, el programa en el PIC decodifica estos datos juntando parejas de dirección-potencia para luego desplegarlas en forma de una gráfica de barras en escala logarítmica; además, despliega en símbolos alfanuméricos la dirección en grados que actualmente se está filtrando en el DSP así como su correspondiente potencia. La Figura 5.19 muestra un dibujo de la información que despliega la pantalla.

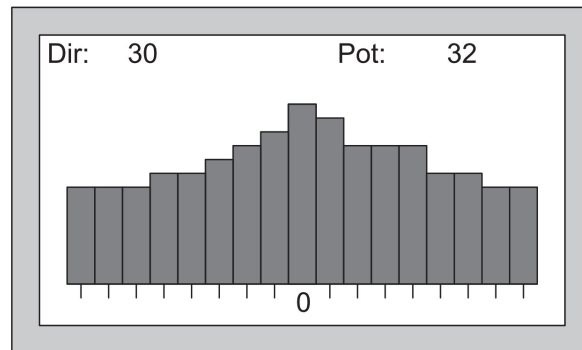


Figura 5.19 Ejemplo de la información que despliega la pantalla gráfica.

El eje horizontal representa las direcciones de -80° a $+80^\circ$ y el eje vertical corresponde a un valor normalizado en escala logarítmica de la potencia de la señal de salida cuando

se apunta a esa dirección, donde cada dos pixeles en la altura de la barra representa un dB.

El PIC se programó en lenguaje *C* con el compilador *CCS C* usando la biblioteca *GLCD.c*; para grabarlo se usó una interfaz marca *VOLNIA* y el software *USP 3.0*.

5.4 Resumen

En este capítulo se detalla la forma en que se transporta la teoría de los capítulos anteriores a especificaciones para construir un sistema de filtrado espacial. Se discute bajo qué parámetros se diseñó el formador de haz y cómo las restricciones físicas y computacionales llevaron a un diseño final. Además, se menciona cómo se usó el DSP para llevar a cabo este proceso de filtrado espacial y sobre los circuitos periféricos que mantienen funcionando el sistema adecuadamente.

Capítulo 6

Simulación, experimentación y resultados

En este capítulo se muestra el rendimiento teórico y práctico del sistema. Para evaluar el rendimiento del sistema se comparó cómo responde a la misma señal cuando ésta cambia su ángulo de arribo o cuando la señal se mantiene fija al tiempo que el haz se direcciona a otro ángulo. Lo que se esperaría medir es que la señal resultante sea la misma señal de entrada con una amplitud máxima cuando el ángulo de arribo de la señal coincide con la dirección apuntada por el formador de haz; si la fuente de sonido o el banco de micrófonos se mueve angularmente o el haz se direcciona a otro ángulo, la señal de salida deja de ser máxima y la señal resultante es de la misma forma que la entrada pero con una amplitud menor.

Para poder evaluar el rendimiento del sistema, se consideraron los resultados de la simulación y pruebas reales tanto en un ambiente anecoico como en uno reverberante. Se explican estas condiciones ambientales y se muestran los resultados de estas mediciones mencionando algunos pormenores encontrados.

6.1 Simulación

Durante el proceso de diseño se hicieron un gran número de pruebas simuladas para verificar la factibilidad del proyecto, dichas pruebas fueron hechas en Matlab con tonos puros generados dentro del propio programa. Las simulaciones permitieron visualizar el rendimiento del sistema en forma ideal. A continuación se muestra la simulación del diseño final probado tanto con tonos puros y con ruido blanco, ambos generados desde Matlab.

6.1.1 Metodología para la simulación

En el programa de Matlab se simuló todo el proceso a realizar en el DSP pero no se contemplaron las imperfecciones del medio ni de los circuitos analógicos involucrados

en el sistema, es decir, se hicieron las siguientes consideraciones

- La señal incidente tiene un frente de onda completamente plano
- No existen ondas reflejadas
- La señal recibida en todos los sensores es exactamente de la misma amplitud y forma de onda
- Todos sensores capturan la señal exactamente al mismo tiempo
- No existe ningún tipo de ruido eléctrico o acústico ajeno a la señal de prueba.

Salvo por la precisión numérica inherente al formato de las variables, el proceso digital se trató de mantener intacto y se usaron los valores reales de la distancia entre micrófonos d_j , el número de micrófonos en cada sub-banda N_j , la frecuencia de muestreo f_s , los coeficientes de los filtros FIR, los coeficientes de las ventanas y los procesos de decimación e interpolación.

Como el programa de simulación no funciona en tiempo real, no se tuvo un flujo de datos entrando y saliendo, el programa trabaja con una ventana suficientemente grande de datos para poder obtener mediciones representativos de potencia. El programa generó 13 vectores de 1000 muestras emulando las señales recibidas en cada micrófono, luego por cada sub-banda hace un barrido en la frecuencia de 150 Hz a 4500 Hz cada 100 Hz , en cada frecuencia se hace otro barrido que emula el recorrido angular de la fuente de sonido de -90° a 90° cada 0.02° . En cada caso se calculó la potencia de la señal de salida del haz y finalmente se dibujó la gráfica del efecto conjunto de las cuatro bandas.

Por otro lado, también se probó el rendimiento del algoritmo usando ruido blanco usando la función *rand()* de Matlab. En dicha prueba se simuló que todos los micrófonos recibían la misma señal de ruido proveniente de 0° , luego se calculó la potencia de la señal de salida al direccionar el haz de -90° a 90° cada 5° .

6.1.2 Resultados de la simulación

Se probó la ventana cuadrada como referencia y dos ventanas tipo Chebyshev-Dolph diseñadas con una disminución de 10 y 30 dB respectivamente. Finalmente, con los resultados de estas simulaciones se decidió que se implementaría una combinación de ventanas Chebyshev-Dolph, la banda 4 con disminución de 10 dB y el resto de las bandas con 30 dB , esto con la intención de tener un patrón de radiación cuyos lóbulos laterales fueran atenuados fuertemente. En la Figura 6.1 se muestra la vista superior de la respuesta de las sub-bandas por separado, el color oscuro implica zonas de baja potencia de la señal de salida y la zona clara una potencia alta de la señal de salida; se hace evidente la intervención de los filtros de separación de bandas que acotan el intervalo de frecuencias que procesa cada formador de haz de banda angosta. Como

se mencionó al final del Capítulo 4 *Tipos de formadores de haz*, si los filtros de separación de bandas no acotaran el intervalo de frecuencias que procesan se presentarían lóbulos gratinados en las bandas 2, 3 y 4, además las bandas 1, 2 y 3 tendrían un comportamiento más acentuado de una versión espacial de filtro pasa bajas por lo que no podrían filtrar espacialmente frecuencias bajas.

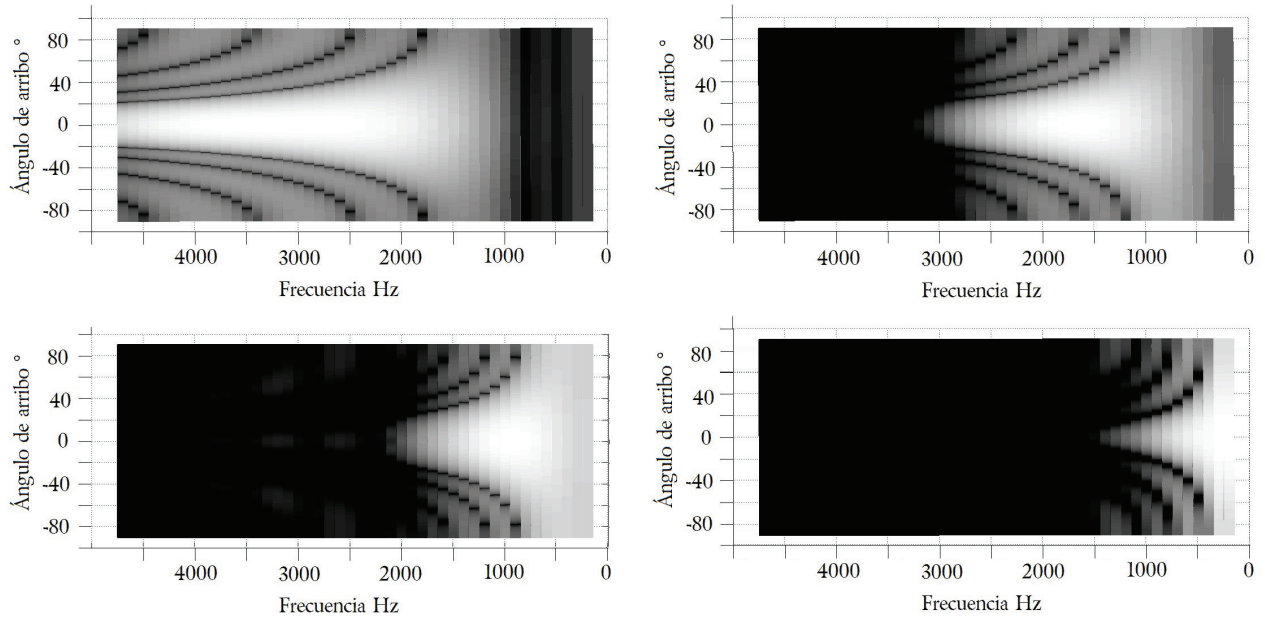


Figura 6.1 Respuesta en frecuencia de las sub-bandas. De izquierda a derecha y de arriba a abajo: Banda 1, banda 2, banda 3 y banda 4.

Unir las respuestas de todas las sub-bandas es como empalmar o sumar las gráficas de la Figura 6.1, las zonas blancas se acomodan de tal forma que forman un haz a lo largo de todo el intervalo de frecuencias. La Figura 6.2 y la Figura 6.3 muestran los resultados de la simulación con la señal que combina el resultado de todas las sub-bandas.

La versión final del programa se pensó para obtener la mayor disminución de lóbulos laterales sacrificando el ancho del haz, por lo tanto, las bandas 1, 2 y 3 ocupan la ventana Chebyshev-Dolph a 30 dB y la banda 4 ocupa la ventana Chebyshev-Dolph a 10 dB. En la Figura 6.3 se aprecia que este diseño combina una fuerte disminución en las sub-bandas agudas y mantiene el ancho del lóbulo de la sub-banda 4 tan pequeño como es posible. Es evidente que para frecuencias bajas el ancho del lóbulo principal crece, esto es de esperarse pues las sub-bandas asociadas con las frecuencias bajas tienen menos micrófonos que las sub-bandas de frecuencias altas. Debe recordarse que esta limitación es producto de que no se puede tener una apertura arbitrariamente grande debido a la consideración de campo lejano. No obstante, en todas las frecuencias se logra cierto nivel de filtrado espacial.

En la Figura 6.4 se muestran los resultados de la simulación con ruido blanco de cada sub-banda por separado y la respuesta conjunta se dibuja en la Figura 6.5. Otra

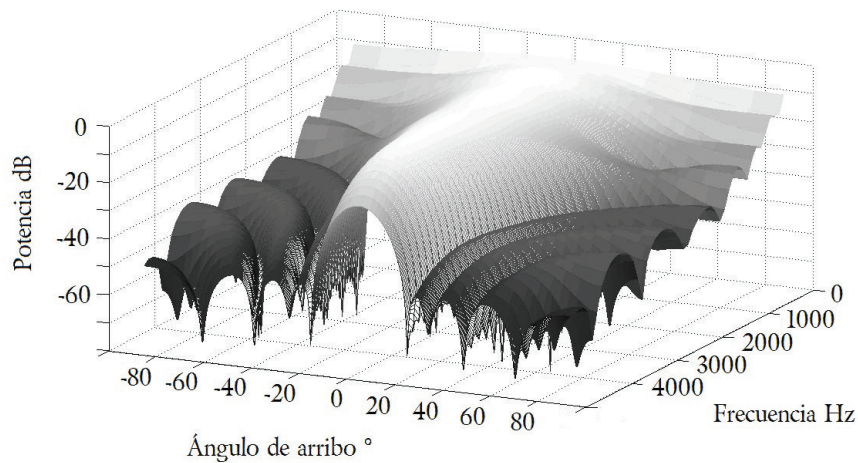


Figura 6.2 Vista tridimensional de los resultados usando ventanas Chebyshev-Dolph con una disminución de 30 dB para las bandas 1, 2 y 3, y usando una ventana Chebyshev-Dolph con una disminución de 10 dB para la banda 4.

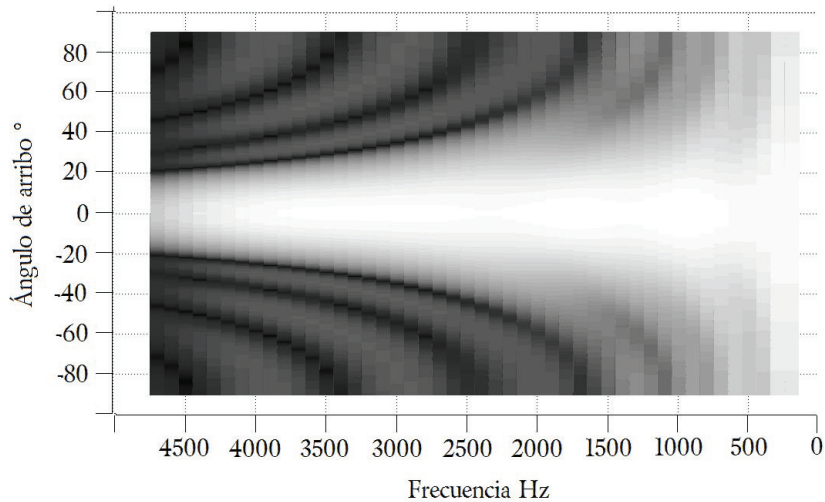


Figura 6.3 Vista superior de los resultados usando ventanas Chebyshev-Dolph con una disminución de 30 dB para las bandas 1, 2 y 3, y usando una ventana Chebyshev-Dolph con una disminución de 10 dB para la banda 4.

vez es evidente que la respuesta del algoritmo mejora en las bandas que involucran más micrófonos, la banda cuatro la cuál es responsable de las frecuencias más bajas tiene el peor rendimiento por tener sólo cuatro micrófonos; además, como el filtro de separación de bandas es de un orden bajo no presenta un comportamiento de filtro pasa banda

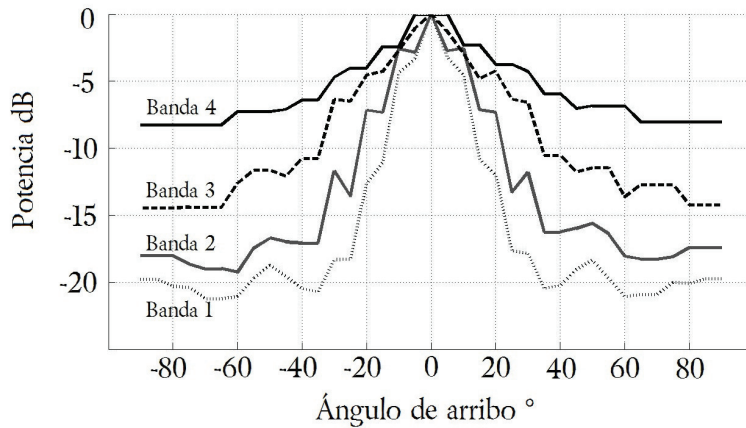


Figura 6.4 Respuesta del algoritmo al ruido blanco por sub-banda.

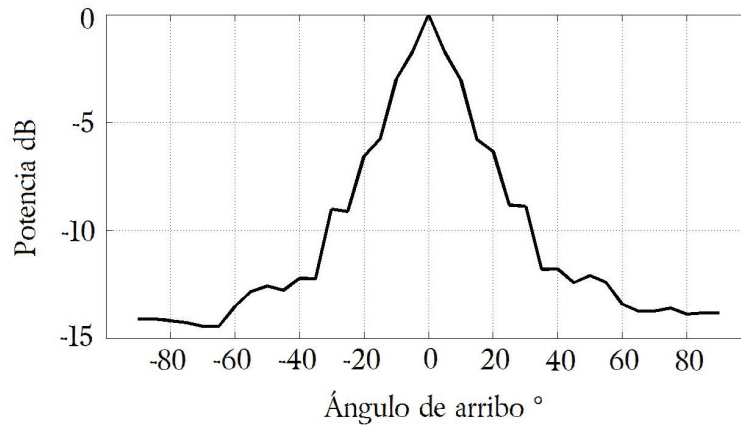


Figura 6.5 Respuesta conjunta del algoritmo al ruido blanco.

sino de pasa bajas permitiendo el paso de frecuencias cuyas longitudes de onda superan en muchas veces la distancia d_j entre micrófonos; este efecto no es tan pronunciado en el resto de las sub-bandas como se presentó en el capítulo anterior en la Figura 5.10. En la Figura 6.5 se presenta la respuesta total del sistema, la atenuación máxima para ángulos mayores a 50° supera los 12 dB.

6.2 Pruebas en la cámara anecoica

Como se mencionó anteriormente, las simulaciones consideran que las ondas que inciden al banco de micrófonos siguen su camino sin causar reverberación, en la realidad existen pocos lugares donde esto sucede. Para poder medir el rendimiento del sistema en condiciones acústicamente controladas se realizaron pruebas en la cámara anecoica

del Centro de Ciencias Aplicadas y Desarrollo Tecnológico (CCADET).

Una cámara anecoica es un ambiente artificial creado en un espacio cerrado cuyas paredes, piso y techo contienen grandes cantidades de material absorbente de sonido en forma de cuñas, creando artificialmente un campo donde las reflexiones se mantienen al mínimo a través de la mayor parte del espectro audible. Este tipo de ambiente controlado permitió tener las condiciones acústicamente ideales para medir el rendimiento real del sistema.

6.2.1 Descripción de las condiciones de prueba en la cámara anecoica

Para realizar las pruebas dentro de la cámara se usó una fuente omnidireccional hecha con 12 bocinas, manejada con un amplificador marca *Crown* modelo XTi-1000 y un generador de funciones marca HP; también se usó como testigo un micrófono condensador de marca *Brüel & Kjør* modelo 4165 y un analizador de espectros también de la marca *Brüel & Kjør* modelo 2133 que sirvieron para verificar el nivel de presión sonora durante las pruebas, el espectro de las señales de ruido y la distorsión armónica de los tonos. Los tonos se reprodujeron por la fuente omnidireccional a un promedio de 75 dB de presión sonora, la distorsión armónica de los tonos fue baja pues la potencia de los armónicos más cercanos fueron menores a la potencia de la fundamental por al menos 45 dB.

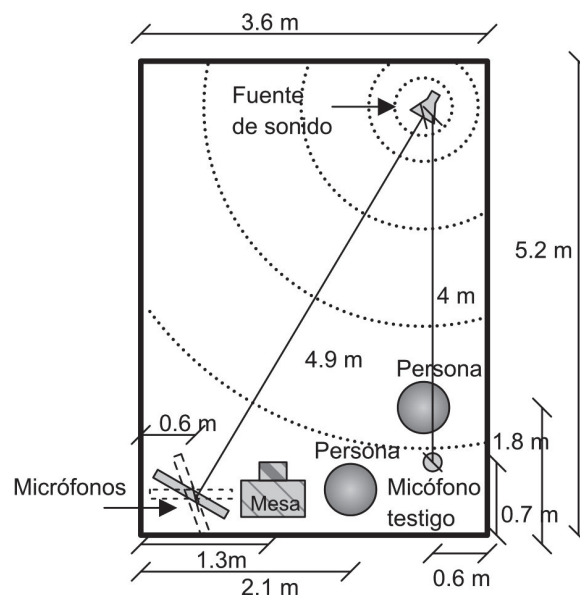


Figura 6.6 Disposición de objetos durante las pruebas en la cámara anecoica.

La fuente omnidireccional y el arreglo de micrófonos se colocaron a 4.9 m de distancia y 1.5 m de altura de las cuñas de material absorbente en el piso, la cámara no permaneció del todo vacía durante las pruebas porque se requería manipular el arreglo de

micrófonos para cambiar su posición durante las pruebas y pausar el programa en el DSP para ver el resultado de la potencia de la señal de salida, por lo que en la cámara había una mesa con el sistema, una fuente de alimentación simétrica, un osciloscopio y dos personas para manipular el sistema. Para mantener condiciones similares durante las pruebas las personas dentro de la cámara se mantuvieron en las mismas posiciones durante el proceso de medición. En la Figura 6.6 se muestra la disposición de la cámara durante las pruebas.

6.2.2 Resultados en la cámara anecoica

Las pruebas del sistema consistieron en producir una señal de sonido con la fuente omnidireccional con ruido (generado con el analizador de espectro) o tonos puros (generado con el generador de funciones) potenciados con el amplificador. Una vez hecho esto se corría el programa y se medía la potencia de la señal digital de salida con el haz apuntando a 0° . Este procedimiento se repitió girando el arreglo de micrófonos de -90° a 90° cada 10° .

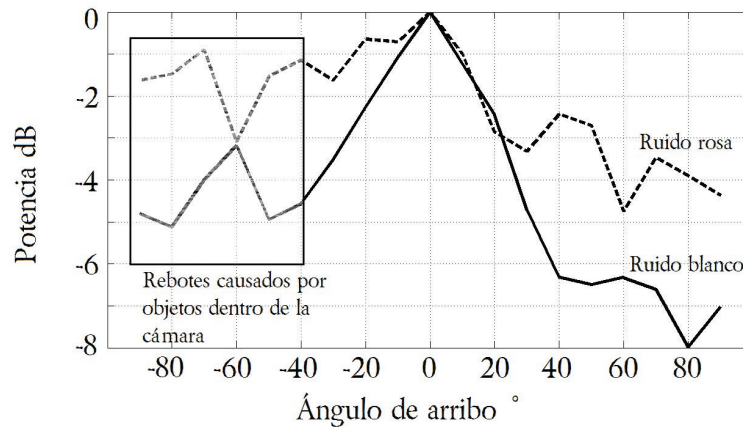


Figura 6.7 Rendimiento del sistema usando ruido rosa y ruido blanco en la cámara anecoica.

La Figura 6.7 muestra los resultados obtenidos con ruido blanco y ruido rosa. Como la sala no estaba vacía, véase Figura 6.6, es de esperarse que los objetos reflejaran una parte de la potencia de la señal en ciertos ángulos, lo cual se muestra claramente en la medición de potencia. Por lo tanto, las muestras encerradas en el rectángulo punteado son mediciones corruptas y no deben tomarse en cuenta, en general la respuesta del haz debe ser simétrica y se puede considerar que con la sala vacía el rendimiento del haz en ángulos negativos sería muy similar al presentado en ángulos positivos.

Como se ha dicho ya en el Capítulo 3, sección *Ruido y distorsión*, el ruido rosa contiene todas las frecuencias audibles pero la potencia de cada frecuencia es directamente proporcional a su longitud de onda, en otras palabras, las frecuencias bajas

tienen más potencia que las altas y es por esta razón que el sistema no reduce en la misma magnitud estos dos tipos de ruido.

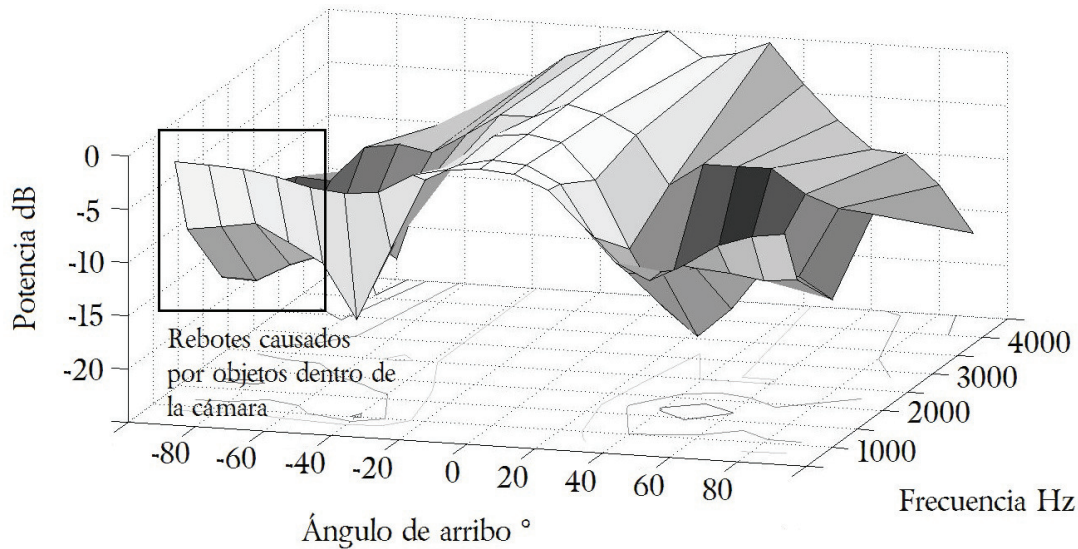


Figura 6.8 Rendimiento del sistema usando tonos de 250 Hz, 500 Hz, 1200 Hz, 2000 Hz y 4000 Hz en la cámara anecoica.

En la Figura 6.8 se presentan los resultados de las pruebas con algunos tonos puros, como se observa la forma general de la gráfica sí corresponde a lo esperado pero el rendimiento es menor a lo predicho por la simulación. Además, nuevamente algunas mediciones de potencia en ciertos ángulos se vieron afectadas por los reflejos causados por objetos dentro de la sala.

Comparando los resultados experimentales de la Figura 6.7 y la Figura 6.8 con la simulación, se observa que siguen una forma similar pero la atenuación es menor a lo esperado. Esto se puede deber a múltiples fenómenos asociados con el comportamiento real de los dispositivos electrónicos que no se consideran en la simulación, como es la distorsión armónica propia en cada micrófono y en el circuito acondicionador de señal; la teoría del formador de haz se basa en que los micrófonos reciben exactamente la misma señal retrasada un cierto tiempo, sin embargo, el proceso de transducción en cada una de las 13 señales las altera de forma distinta, y por tanto las señales no podrán ser idénticas y habrá cierta potencia de la señal que no se pueda eliminar con el método que aquí se propone. Por otro lado, la versión de software usado en la cámara anecoica no resultó ser la versión final ya que después de asistir al CCADET se notó que el bajo rendimiento de los filtros de interpolación agregaban ruido a la señal de salida por lo que se aumentó el orden de los filtros para resolverlo; además, se modificó la forma en que el sistema calcula el el valor constante que se debe agregar a las señales para quitar la componente de directa, incluso se modificaron las constantes de amplitud de las cuatro bandas para mejorar su ecualización.

6.3 Pruebas en el laboratorio de PDS

También se realizaron pruebas en el Laboratorio de Procesamiento Digital de Señales en el edificio de Posgrado de Ingeniería para obtener resultados sobre el rendimiento del sistema en ambientes cerrados sin ningún tipo de tratamiento acústico especializado.

6.3.1 Descripción de las condiciones de prueba en el laboratorio de PDS

El laboratorio tiene una disposición típica de un salón de clases cuyas medidas se muestran en la Figura 6.9, donde los rectángulos grises representan mesas con objetos arriba de ellas (monitores, teclados, osciloscopios, fuentes de voltaje, entre otros equipos electrónicos).

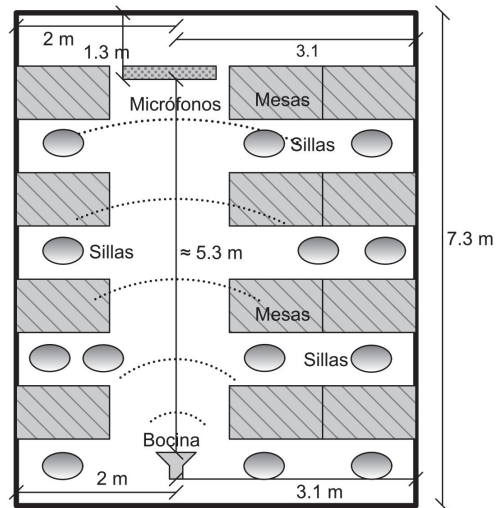


Figura 6.9 Disposición de objetos durante las pruebas en laboratorio de procesamiento digital de señales.

Para saber cuán reverberante es el laboratorio se midió el tiempo en el que al interrumpir una señal de ruido blanco la potencia cae de 82 a 76 dB, de 76 a 66 dB y de 66 a 54 dB y luego interpoló el tiempo a una disminución de -60 dB, para estas mediciones se usó un sonómetro marca *TES* modelo *1352A*. Para medir lo mejor posible con el equipo disponible se conectó la salida analógica del sonómetro, cuyo valor corresponde a $10 \frac{mV}{dB}$, a un osciloscopio digital, luego se generaba ruido blanco con un nivel de presión de sonido de 82 dB en ponderación A y luego se apagaba; cómo el osciloscopio estaba en una frecuencia de muestreo muy baja era capaz de registrar el valor instantáneo de la señal del sonómetro por un par de segundos, usando los cursores en la gráfica del osciloscopio se determinaban valores de tiempo y voltaje con bastante exactitud.

El experimento se repitió seis veces en distintos lugares cercanos a donde se colocó el arreglo de micrófonos.

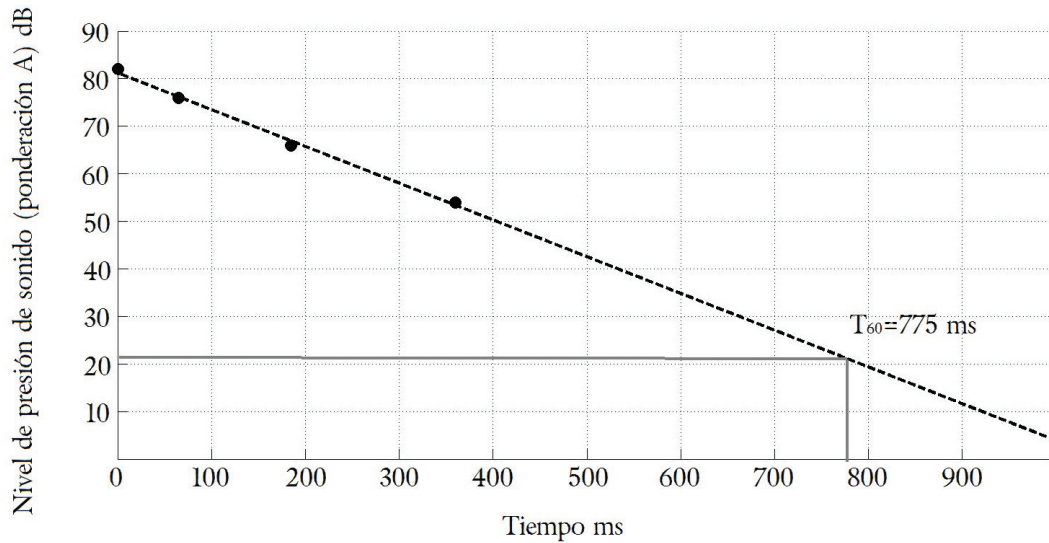


Figura 6.10 Interpolación para encontrar el tiempo de reverberación T_{60} en el laboratorio de PDS.

Usando interpolación lineal se determinó que el tiempo de reverberación del laboratorio es $T_{60} = 775ms$. La Figura 6.10 muestra con círculos el promedio de los valores experimentales y la línea de interpolación.

El arreglo de micrófonos se montó a $1.2 m$ del piso que corresponde a la altura que se encuentran los monitores, osciloscopios y otro equipo electrónico en el laboratorio, durante las pruebas dichos objetos estaban a menos de $40 cm$ de distancia de los micrófonos, por lo que se puede decir que a pesar de que una parte del frente de onda de la señal incidió directamente en el arreglo de micrófonos, el sistema estuvo inmerso en un ambiente con objetos acústicamente reflejantes.

El arreglo se puede posicionar de forma horizontal o vertical. Como se muestra en la Figura 6.11, horizontalmente los micrófonos están alienados en cierto ángulo deseado a la bocina pero se dirigen al techo; verticalmente los micrófonos conservan el mismo ángulo respecto a la bocina pero se dirigen directamente a la fuente de sonido. La diferencia entre una u otra posición debería ser imperceptible si los micrófonos fueran completamente omnidireccionales; como los micrófonos electret que se usaron suelen tener características distintas a las ideales se hicieron pruebas con ambas posiciones para poder concluir la influencia de la posición del arreglo sobre la respuesta total del sistema.

Las señales de prueba se emitieron con un *iPad* usando el software *iFunctionGenerator* y una bocina de computadora marca *Harman Kardon*. Para facilitar la dinámica de las pruebas se posicionaron la bocina y el arreglo de micrófonos de forma que la

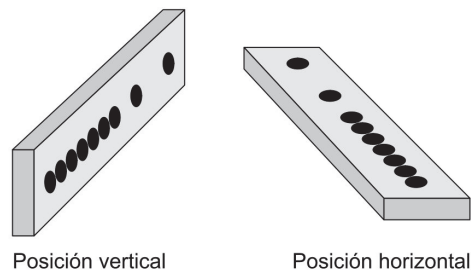


Figura 6.11 Posiciones vertical y horizontal del arreglo de micrófonos.

señal incidiera a 0° y se varió la dirección del haz mediante software, es decir, no se movió físicamente el arreglo. En el momento de realizar las pruebas el laboratorio permaneció cerrado, pero cierta cantidad de ruido proveniente de los ventiladores de la fuente de alimentación estuvo siempre presente, además de que ocasionalmente se filtraba ruido del pasillo y de los jardines adyacentes.

6.3.2 Influencia de los patrones de interferencia

Una de las desventajas de colocar el arreglo en un espacio como el laboratorio es que la reverberación del cuarto provoca patrones de interferencia (como se mostró en el capítulo 3) los cuales afectan fuertemente la amplitud de la señal recibida en los micrófonos, es decir, se corre el riesgo que la interferencia de las señales disminuya la presión de sonido en la zona donde se localizan algunos micrófonos al mismo tiempo que aumenta en la zona donde se encuentran otros micrófonos, provocando que los sensores reciban señales distintas. Esta idea la muestra la Figura 6.12.

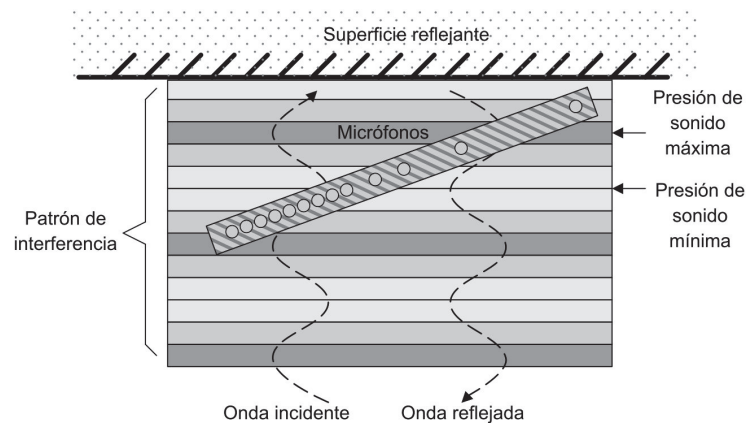


Figura 6.12 Efecto de los patrones de interferencia en la colocación del banco de micrófonos.

Con cualquier cambio en la frecuencia del tono de prueba, o con el cambio de posición

de la bocina el patrón de interferencia se modifica, de modo que no es posible medir el rendimiento del sistema usando tonos puros sin ajustar la posición de la bocina de tal forma que los micrófonos no se encuentren en una zona de baja presión de sonido. Desde luego que si se usa una señal de ruido, la falta de periodicidad de la señal no permite que se formen estos patrones de manera tan marcada y este fenómeno no afecta las pruebas tan severamente.

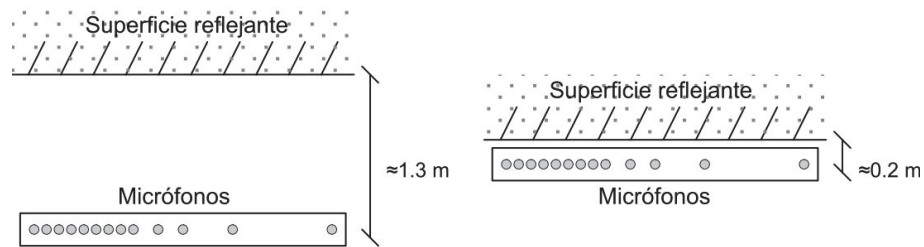


Figura 6.13 Diferentes posiciones del arreglo en las pruebas en el laboratorio.

Se realizaron pruebas moviendo el arreglo de micrófonos perpendicularmente respecto a la pared y se observó cómo se formaban los patrones de interferencia con el banco de micrófonos separado alrededor de 1.3 m de la pared opuesta a la fuente de sonido y otra con el banco de micrófonos a 0.2 m de la pared.

También se usó un difusor diseñado para dividir el frente de onda de una señal de voz en varios frentes de onda en distintas direcciones, lamentablemente las dimensiones del arreglo sobrepasaban las del difusor por lo que no se pudo probar apropiadamente su efecto en las pruebas finales; sin embargo, se usó en algunas pruebas para medir la magnitud de los patrones de interferencia. La Figura 6.14 muestra las dimensiones de la superficie que cubre así como las medidas del perfil con el que forma el campo difuso.

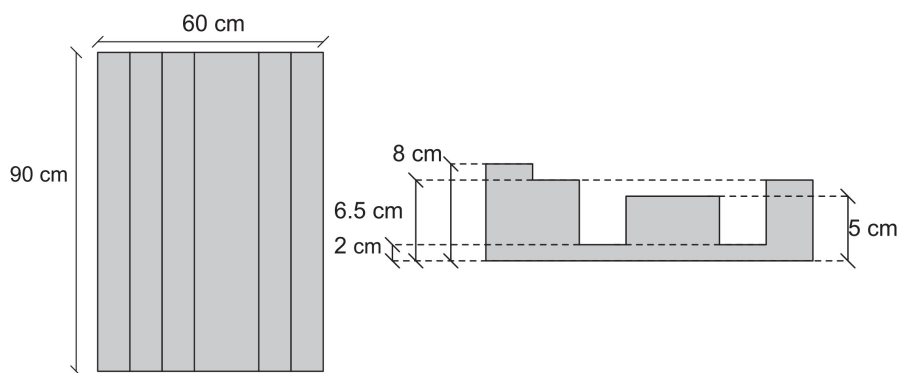


Figura 6.14 Dimensiones del difusor.

Para tener una idea de la magnitud de los patrones de interferencia se hicieron medidas sobre la potencia de la señal registrada por un solo micrófono en zonas de baja y

alta presión con y sin difusor. Los resultados se muestran en el Cuadro 6.1, donde la diferencia de potencia se refiere a la diferencia en dB de la potencia registrada cuando un micrófono está en una zona de máxima presión de sonido en comparación con el micrófono si se encuentra en una zona de mínima presión de sonido, entre menor sea esta diferencia significa que el patrón de interferencia es menos pronunciado. Usando señales de ruido no fue posible identificar zonas de mínima o máxima presión de sonido porque la falta de periodicidad de la señal impide que se formen patrones de interferencia estacionarios por lo que se experimentó sólo con tonos puros.

Cuadro 6.1 Efecto del difusor en los patrones de interferencia.

Tipo de señal	Diferencia en potencia sin difusor	Diferencia en potencia con difusor
Sinusoidal 250 Hz	7.6 dB	6.7 dB
Sinusoidal 500 Hz	3.8 dB	5 dB
Sinusoidal 1000 Hz	4 dB	5.5 dB

Para la medición de estos valores se colocó un micrófono del arreglo a 15 *cm* del difusor el cual estaba apoyado en una pared del laboratorio, luego se colocó la bocina a una distancia del micrófono cercana a 4 *m* de tal forma que el micrófono se situara en una región de mínima presión de sonido y se midió la potencia de la señal, luego se alejó la bocina hasta colocar el micrófono en una zona de máxima presión de sonido y se midió nuevamente la potencia, esto se repitió para tres frecuencias distintas. Posteriormente, se repitió el experimento quitando el difusor por lo que el micrófono se situó a 30 *cm* de la pared.

Como se observa en el Cuadro 6.1, no se logró medir una diferencia importante y únicamente en una frecuencia se reportó mejoría usando el difusor. Esto indica que el difusor no tiene las medidas necesarias para hacer una diferencia significativa en la formación de los patrones y que probablemente otras superficies reflejantes que no tienen difusores como el piso, techo y mesas también contribuyen fuertemente a este fenómeno por lo que no es suficiente con el difusor probado.

6.3.3 Resultados en el laboratorio de PDS

Se encontró que al variar la distancia del arreglo a la pared no se afectó de manera significativa la influencia de la señal recibida con los patrones de interferencia causados cuando se prueban tonos puros. De manera similar, usando el difusor tampoco se encontró que exista una ganancia sustancial y no logró quitar los patrones de interferencia con tonos puros. Por tanto, las pruebas que se consideraron finales se hicieron usando ruido blanco, sin difusor, con el arreglo separado aproximadamente 1.3 *m* de la pared y se varió la posición del arreglo en horizontal y vertical.

Las pruebas que se presentan a continuación corresponden al arreglo en posición horizontal, los resultados en escala logarítmica de la potencia de la señales digitales de salida al hacer incidir ruido blanco se muestran en la Figura 6.15 y la Figura 6.16.

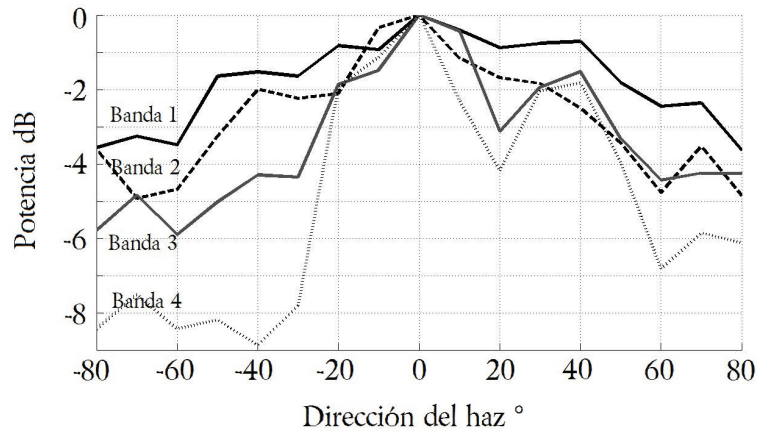


Figura 6.15 Rendimiento del sistema con ruido blanco por bandas en el laboratorio de procesamiento digital de señales con los micrófonos en posición horizontal.

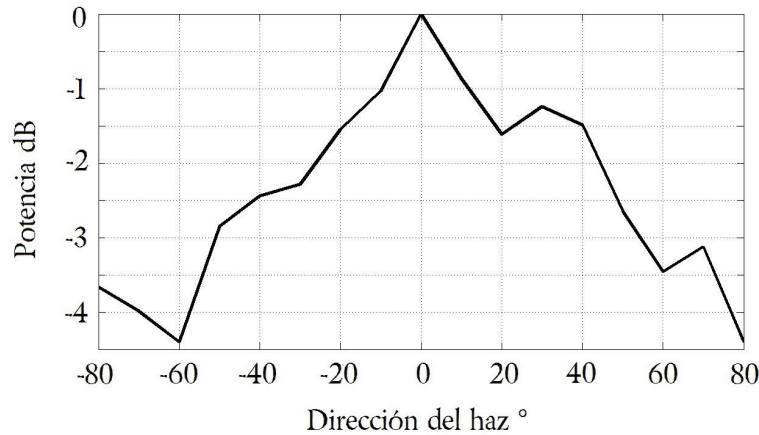


Figura 6.16 Rendimiento del sistema con ruido blanco en el Laboratorio de Procesamiento Digital de Señales con los micrófonos en posición horizontal.

Como se observa en la Figura 6.15, todas las bandas tienen rendimiento distinto pero siguen formas similares, la falta de simetría se puede explicar por los objetos dentro del laboratorio. En la Figura 6.16 se muestra la potencia de la señal combinada, se observa que la disminución máxima que se logra es de más de 4 dB.

Posteriormente, se hicieron las mismas pruebas con el arreglo en posición vertical. Los resultados en escala logarítmica se muestran en la Figura 6.17 y la Figura 6.18.

Se observó que hubo una mejoría de hasta 1 dB en algunas de las direcciones mayores a 50°, elevando la disminución máxima de 4 dB a más de 5 dB. Esto pudo deberse a que la respuesta de los micrófonos no es del todo omnidireccional y es mejor ori-

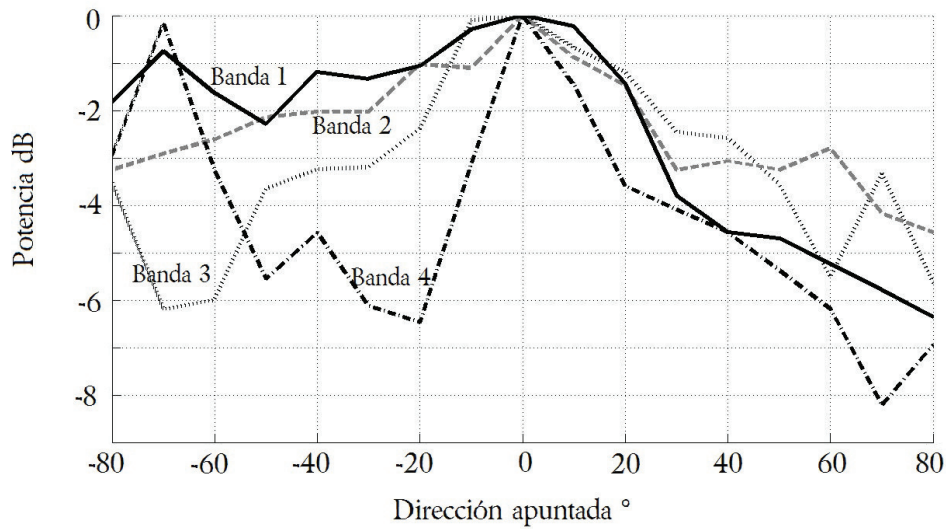


Figura 6.17 Rendimiento del sistema con ruido blanco por bandas en el Laboratorio de Procesamiento Digital de Señales con los micrófonos en posición vertical.

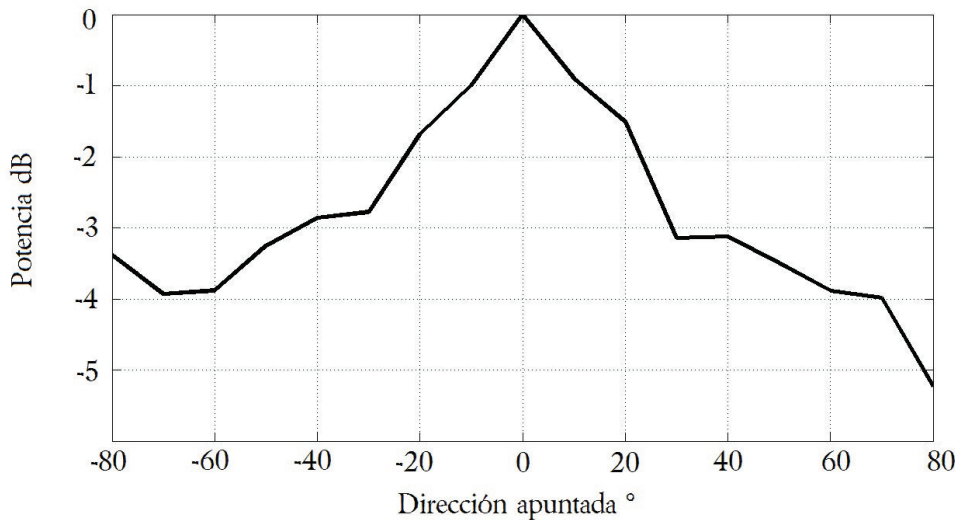


Figura 6.18 Rendimiento del sistema con ruido blanco en el Laboratorio de Procesamiento Digital de Señales con los micrófonos en posición vertical.

entarnos directamente a la fuente de sonido para que las señales se capturen con más fidelidad y funcione mejor el algoritmo. En la sección anterior donde se mostraron las pruebas en la cámara anecoica se hizo evidente que incluso la presencia de un par de personas y una mesa puede modificar las gráficas de rendimiento del sistema, en un

espacio cómo el Laboratorio de Procesamiento Digital donde existen paredes y objetos acústicamente reflejantes, la señal incide innumerables veces al arreglo con distinta potencia y fase debido a que se forma un ambiente reverberante; esto hace predecible la aparente disminución de rendimiento, no obstante, se evidencia cierto nivel de filtrado incluso en ambientes reverberantes.

6.4 Comparación de los resultados obtenidos

Dependiendo de la locación el sistema tiene un rendimiento distinto, como es de esperarse el mejor rendimiento lo mostró la simulación porque todas las señales son creadas y procesadas artificialmente en Matlab y el modelo presentado funcionara de manera ideal, también no fue sorprendente que el sistema construido funcione mejor en ambientes anecoicos ya que la teoría del formador de haz supone condiciones de campo libre. En la Figura 6.19 se muestra en una sola gráfica los resultados mostrados en las secciones anteriores.

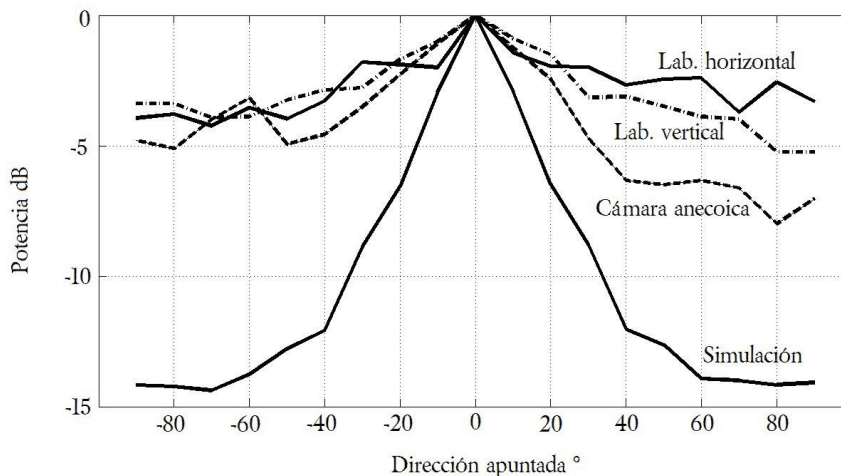


Figura 6.19 Comparación del rendimiento del sistema con ruido blanco en distintas circunstancias.

En el Cuadro 6.2 se presenta un resumen de los resultados obtenidos usando ruido blanco en la simulación y en situaciones reales.

6.5 Resumen

En este capítulo se detalló cómo se evaluó el rendimiento del sistema en tres distintos ámbitos: simulación en Matlab, en un ambiente anecoico y en un ambiente reverberante. Se muestra que los niveles de disminución de fuentes acústicas provenientes de direcciones indeseadas con el sistema real no llegaron al nivel de disminución mostrado en

Cuadro 6.2 Comparación de los resultados obtenidos.

Tipo de prueba	Disminución máxima	Dirección a -3 dB	Porcentaje de disminución máxima respecto a la simulación
Simulación en Matlab	14 dB	$\sim \pm 10^\circ$	–
Cámara anecoica	8 dB	$\sim \pm 20^\circ$	57 %
Laboratorio PDS (posición horizontal)	4.5 dB	$\sim \pm 40^\circ$	32 %
Laboratorio PDS (posición vertical)	5.3 dB	$\sim \pm 30^\circ$	37 %

la simulación y se mencionan algunas razones por las cuales puede estar presentándose esta disminución de rendimiento.

Capítulo 7

Conclusiones

- Se logró desarrollar un sistema que alimentado con las señales provenientes de un banco de micrófonos y un algoritmo programado en un DSP TMS320F28335 de Texas Instruments filtre una señal con ancho de banda comparable con la voz proveniente de una dirección, atenuando en tiempo real otras fuentes de sonido localizadas alrededor.
- Debido a que sólo se está usando cerca del 50% de la capacidad de cómputo del DSP, este sistema puede servir como base en la implementación de otros algoritmos de filtrado espacial más efectivos y robustos.
- Los resultados que se obtuvieron en la cámara anecoica están por debajo de lo que arrojaron las simulaciones, se probó que el sistema sólo lograba atenuar hasta 8 dB de potencia y no los 14 dB que indican las pruebas simuladas.
- En el laboratorio se probó la versión final del sistema pero aún así las mediciones disminuyeron a un 65% de lo reportado en la cámara anecoica; ésta aparente disminución del rendimiento es atribuible a las condiciones reverberantes del recinto. Cuando se mueve el haz a una dirección diferente a donde proviene el sonido, se esperaría que la potencia disminuyera porque se apunta a una dirección donde *únicamente proviene silencio*, sin embargo, en un ambiente reverberante esto no es verdad, existe una cierta cantidad de sonido proveniente de todas las direcciones por lo que es comprensible que no se logre medir el mismo grado de efectividad que se observó en un ambiente anecoico.
- Aumentar la SNR carece de sentido si no se incrementa la inteligibilidad de la señal de voz. Para estar seguros de que el procesamiento de las señales no disminuye la claridad de la señal de voz, sería necesario implementar mediciones que indiquen el rendimiento de los algoritmos en términos de la inteligibilidad.
- El sistema fue capaz de interactuar con el usuario sin intervención de la PC y se logró implementar una interfaz para manipular el sistema ya que el programa monitorea la posición de un potenciómetro y transforma ese valor en la dirección

en la que apunta el formador de haz; de esta forma el usuario puede cambiar la dirección de filtrado en un intervalo de -80° a $+80^\circ$ cada 10° moviendo una perilla. Por otro lado, la información sobre la dirección de los formadores de haz y la potencia de la señal filtrada fue presentada en la pantalla en símbolos alfanuméricos y en una gráfica de barras en escala logarítmica.

- El proceso de separación de bandas fue la parte más demandante del programa y a pesar de estar escrita completamente en lenguaje ensamblador, ocupó cerca del 26 % del número de instrucciones disponibles, si se hubiera programado en lenguaje *C* requeriría más de cinco veces el número de ciclos de reloj y volvería inviable realizar el proyecto con este DSP.
- Los formadores de haz con ventana no requirieron muchas operaciones para ser calculados pero como se programaron en lenguaje *C* ocuparon más del 11 % de la capacidad del DSP; si se programara en ensamblador este porcentaje podría bajar menos de un quinto y usar únicamente un 2 %-3 % del tiempo total disponible.
- La adquisición de datos se mantuvo al mínimo debido a la emulación de direccionamiento circular en las ventanas de datos y representó un poco más del 3.5 % de T_s .
- El tiempo libre, en el cual el CPU no realiza ninguna acción programada y solamente espera la siguiente interrupción del timer, representó el 49 % de T_s por lo que el sistema logró trabajar en tiempo real.

Trabajos futuros

La reducción de ruido en señales de voz mediante un formador de haz no es un trabajo simple y en realidad conlleva toda una línea de investigación; resta mucho trabajo por delante y esta tesis consiste únicamente en una plataforma para probar soluciones más robustas en este ámbito. A continuación se mencionan algunas propuestas que pueden mejorar el rendimiento del sistema o aumentar su utilidad.

- Cambiar los formadores de haz de banda angosta por formadores de haz con algoritmos adaptables y evaluar su rendimiento.
- Implementar un formador de haz que explore todas las direcciones sistemáticamente para que se detecte en qué dirección existen fuentes de sonido y después decida hacer un seguimiento automático de una de ellas.
- Programar múltiples formadores de haz de banda ancha para filtrar varias direcciones simultáneamente.
- Diseñar formadores de haz que puedan funcionar eficientemente en la región de campo cercano.

- Implementar pruebas para evaluar el rendimiento de los algoritmos en función de la inteligibilidad de la señal de voz, ya sea con base en pruebas psicoacústicas o alimentando a un sistema de reconocimiento de voz con la salida del DAC.

Glosario

A

Aliasing. Fenómeno que ocurre cuando la frecuencia de muestreo es menor a la frecuencia de Nyquist (dos veces la frecuencia más alta presente en la señal), lo que implica que muchas señales en tiempo continuo se pueden representar por la misma secuencia discreta.

ADC. Analog-to-Digital Converter, es un dispositivo electrónico que convierte un voltaje analógico en un número binario.

Ambiente anecoico. Es un ambiente que recrea las condiciones ideales de campo libre, es decir, no existe ningún reflejo. Regularmente esto se logra al montar grandes cantidades de material absorbente en paredes, piso y techo.

Ambiente reverberante. En contraste con un ambiente anecoico, en un ambiente reverberante el reflejo de una señal no debe atenuarse y requiere paredes, piso y techo altamente reflejantes. Regularmente esto se logra con materiales muy pesados, lisos y libres de poros.

B

Banda ancha. Se dice que una señal es de banda ancha si el intervalo de frecuencias que contiene la señal es mayor al valor de la frecuencia central de dicho intervalo.

Banda angosta. Se dice que una señal es de banda angosta si el intervalo de frecuencias que contiene la señal es menor al valor de la frecuencia central de dicho intervalo.

C

Campo cercano. Es la región donde el frente de onda de una señal debe considerarse curvo.

Campo lejano. Es la región donde el frente de onda de una señal puede considerarse plano.

CCADET. Centro de Ciencias Aplicadas y Desarrollo Tecnológico.

D

DAC. Digital-to-Analog Converter, es un dispositivo electrónico que convierte un número binario en un voltaje analógico.

Decibel dB. Es una unidad relativa empleada para comparar dos magnitudes una de referencia y otra de prueba.

Difusor acústico. Es una superficie que por su geometría y las propiedades acústicas del material del que está hecho perturba el frente de onda de una onda acústica.

DSP. Digital Signal Processor, es una máquina digital diseñada para ser eficiente en la implementación de algoritmos de procesamiento digital de señales.

E

Efecto de la fiesta de cocktail. Describe la capacidad cognitiva de concentrarse en una señal de voz inmersa en un ambiente ruidoso.

F

FIR, filtro. Un filtro de respuesta finita al impulso en un filtro digital el cual si se alimenta con una señal impulso, su respuesta será una secuencia finita de términos distintos a cero. Para su cómputo se necesita un conjunto finito de valores anteriores de la señal de entrada.

Fonio. Es una unidad de potencia de sonido que compensa la dependencia a la frecuencia de la respuesta del oído humano.

Formador de haz. Es una técnica de filtrado espacial usando arreglos de transductores.

Frente de onda. Es una región dentro de un medio de propagación donde los elementos que propagan la señal son alcanzados en un mismo instante por una determinada onda.

I

IIR, filtro. Un filtro de respuesta infinita al impulso es un filtro digital el cual si se alimenta con una señal impulso, su respuesta será una secuencia infinita de términos distintos a cero. Para su cálculo se necesita un conjunto finito de valores anteriores de las señales de entrada y salida.

Inteligibilidad. Es una medida de qué tan comprensible es una señal de voz.

Interpolación. Es el proceso de manipular una secuencia discreta que representa una señal al intercalar un número determinado de ceros entre cada muestra seguido de un proceso de filtrado, de tal forma que se aumente artificialmente la frecuencia de muestreo.

J

JTAG. Joint Test Action Group, es el nombre utilizado para la norma IEEE 1149.1 titulada Standard Test Access Port and Boundary-Scan Architecture usada como mecanismo de depuración en sistemas embebidos.

M

MIPS. Millones de Instrucciones Por Segundo, es una medida de rendimiento en procesadores.

N

Nivel de presión de sonido, ponderación A. Es un método para medir la potencia de una señal acústica que pondera la potencia de las componentes en frecuencia de manera similar a la respuesta del oído humano (véase sección 3.3.1 *Oído humano*).

P

PDS. Procesamiento Digital de Señales.

PIC. Es una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc.

Pipeline. Es una técnica de procesamiento en paralelo ampliamente usado en DSPs.

Patrón de radiación. Es la forma en que se distribuye la energía al variar el ángulo de arribo de la señal que recibe un arreglo de micrófonos cuyas señales se procesan por un formador de haz.

S

SNR. Es una medida expresada en dB que relaciona la potencia de una señal deseada con la potencia del ruido presente.

T

Transductor. Es un dispositivo que cambia una señal de un tipo de energía a otra.

Z

Zero padding. Es una técnica de sobremuestreo que consiste en intercalar ceros entre las muestras de la señal a sobremuestrear.

Anexos

Cámara anecoica

Esta cámara forma parte de las instalaciones del departamento de acústica y vibraciones del CCADET en Ciudad Universitaria [6]. La cámara consta de dos cubos de concreto uno dentro del otro. El cubo interior tiene paredes de 0.3 m de espesor y se apoya en el cubo exterior con seis cojines de neopreno de 0.2 m de altura, como se muestra en la Figura 7.1.

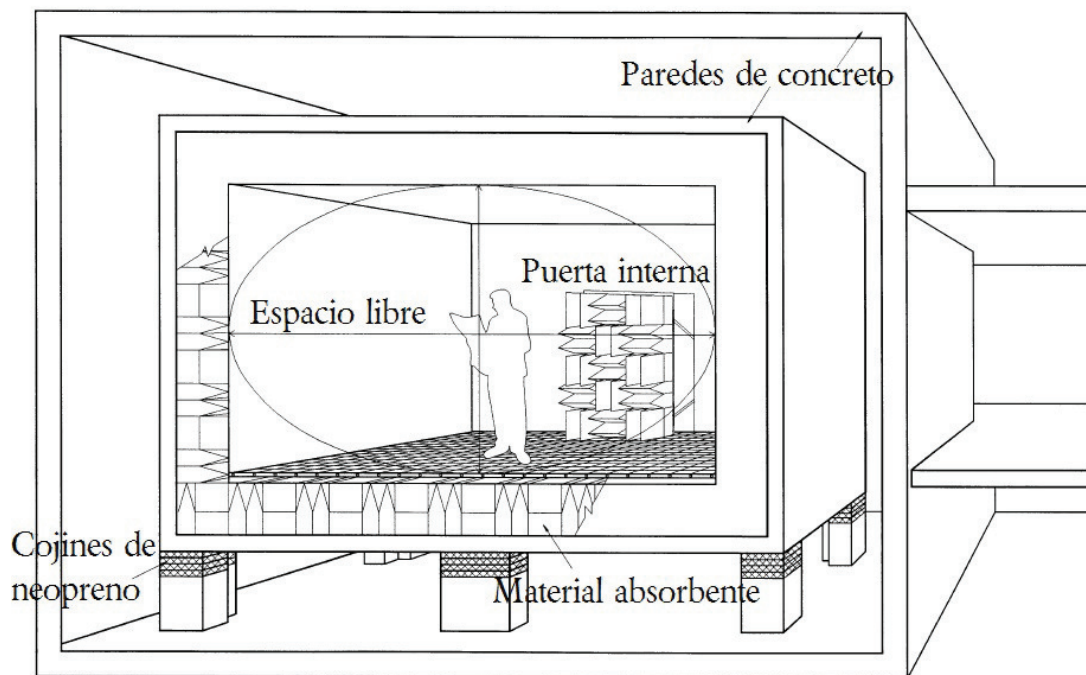


Figura 7.1 Diagrama de la cámara anecoica [6].

El material absorbente consta de cuñas de espuma de poliuretano de 1.1 m de largo. El piso es una malla de alambre de 3 mm colocada a 0.2 m de altura respecto a la punta de las cuñas del piso. La puerta de acceso a la cámara está hecha de láminas de

acero separadas por un marco de neopreno, está rellena de arena fina de playa y pesa alrededor de 1000 *kg*.

Además de cumplir con las características de campo libre para señales de 80 *Hz* a 20,000 *Hz* también provee gran aislamiento al ruido exterior, atenuando más de 60 dB sonidos externos.

Difusores

Los difusores surgen como respuesta a disminuir los reflejos en grandes salas. A partir del siglo XX la ingeniería de la época hizo posible construir edificios con grandes superficies planas que cambiaron la acústica de sala de los edificios construidos. Hacia finales del siglo pasado se manifestó el interés por usar superficies rugosas para controlar los reflejos en salas de concierto. Uno de los pioneros en el diseño de estas superficies fue M. R. Schroeder con propuestas como *Quadratic Residue Diffuser (QRD)*. Otras personas que contribuyeron al desarrollo de las técnicas de diseño de difusores fueron D'Antonio y los neozelandeses Marshall y Hyde [8].

El sonido que impacta una superficie es transmitido a otro cuerpo adyacente, absorbido o reflejado; la cantidad de energía que se transmite, se absorbe o se refleja depende de las propiedades acústicas de la superficie en la que impacta. El sonido reflejado puede ser redireccionado, si es una superficie grande y plana, o se dispersa por una superficie difusora. Cuando una porción significativa del sonido es espacial y temporalmente dispersada, se refleja de forma difusa, la superficie involucrada se le denomina *difusor*. Dependiendo del efecto que se desee obtener en la reducción de la reverberación, el nivel de los primeros reflejos, la densidad de los reflejos tardíos, entre otros parámetros, se debe combinar difusores con superficies absorbentes para prevenir la distorsión acústica. Un difusor es colocado en una pared para dispersar los reflejos y reducir la distorsión sin quitar energía sonora del lugar, para reducir la energía se necesita una superficie acústicamente absorbente [9].

Uno de los mayores adelantos en el diseño de difusores fue la invención del difusor de fase gratinado de Schroeder. Además de una construcción simple, los difusores propuestos anteriormente no dispersaban el sonido de una forma predecible. El difusor de Schroeder permite diseñar una solución con base en parámetros de diseño usando un conjunto pequeño de ecuaciones sencillas. Existen difusores de una dimensión y de dos dimensiones, las subdivisiones de los difusores se les denomina *pozos*. Lo ancho y profundo de estos pozos se diseñan usando tres ecuaciones básicas [9]

$$width = \frac{\lambda_{min}}{2}$$

donde *width* es el ancho del pozo y λ_{min} es la longitud de onda mínima del intervalo de diseño ,

$$s_n = n^2 \text{mod}(N)$$

donde s_n es la secuencia numérica para diseñar la profundidad de los pozos, n es el identificador de cada pozo, mod es la función residuo, N es un número primo que genera la secuencia y es el número de pozos en cada secuencia

$$d_n = \frac{s_n \lambda_0}{2N}$$

donde d_n es la profundidad del n -ésimo pozo y λ_0 es la longitud de onda de la frecuencia más baja del intervalo de diseño.

El difusor que se usó en algunas pruebas en el laboratorio se diseñó usando este método usando $width = 8.5$ cm, $\lambda_0 = 21$ cm y $N = 7$. Se tiene

$$d_n = [0 \quad 1.5 \quad 6 \quad 3 \quad 3 \quad 6 \quad 1.5]$$

El difusor es unidimensional y sólo corresponde a un periodo de la secuencia.

DSP TMS320F28335

El DSP se conforma esquemáticamente como lo muestra la Figura 7.2 y tiene las siguientes características principales:

- Hasta 150 *MHz* (ciclo de máquina de 6.67 *ns*)
- CPU de alto rendimiento (TMS320C28x) con unidad de punto flotante de precisión simple IEEE-754, instrucciones de multiplicación-acumulación (MAC) de 16×16 y 32×32, MAC doble de 16×16, arquitectura tipo Harvard, rápida respuesta a interrupciones
- Unidad de acceso directo a memoria (DMA) de seis canales
- 256 KB×16 de memoria Flash, 34 KB×16 de memoria SARAM y 1KB ×16 de memoria permanente de un solo uso OTP ROM
- Oscilador dentro del chip con PLL reconfigurable dinámicamente y módulo Watch-dog timer
- Hasta 18 salidas de modulación por ancho de pulso PWM y hasta seis salidas de modulación de ancho de pulso de alta resolución HRPWM
- Tres timers de 32 bits
- Periféricos seriales CAN, SCI, McBSP, SPI e I2C
- 16 canales de conversión analógico-digital de 12 bits de resolución con dos circuitos muestreador-retenedor y una tasa de conversión de 80 *ns*
- Hasta 88 entradas/salidas multiplexables individualmente programables
- Soporte para depuración por JTAG.

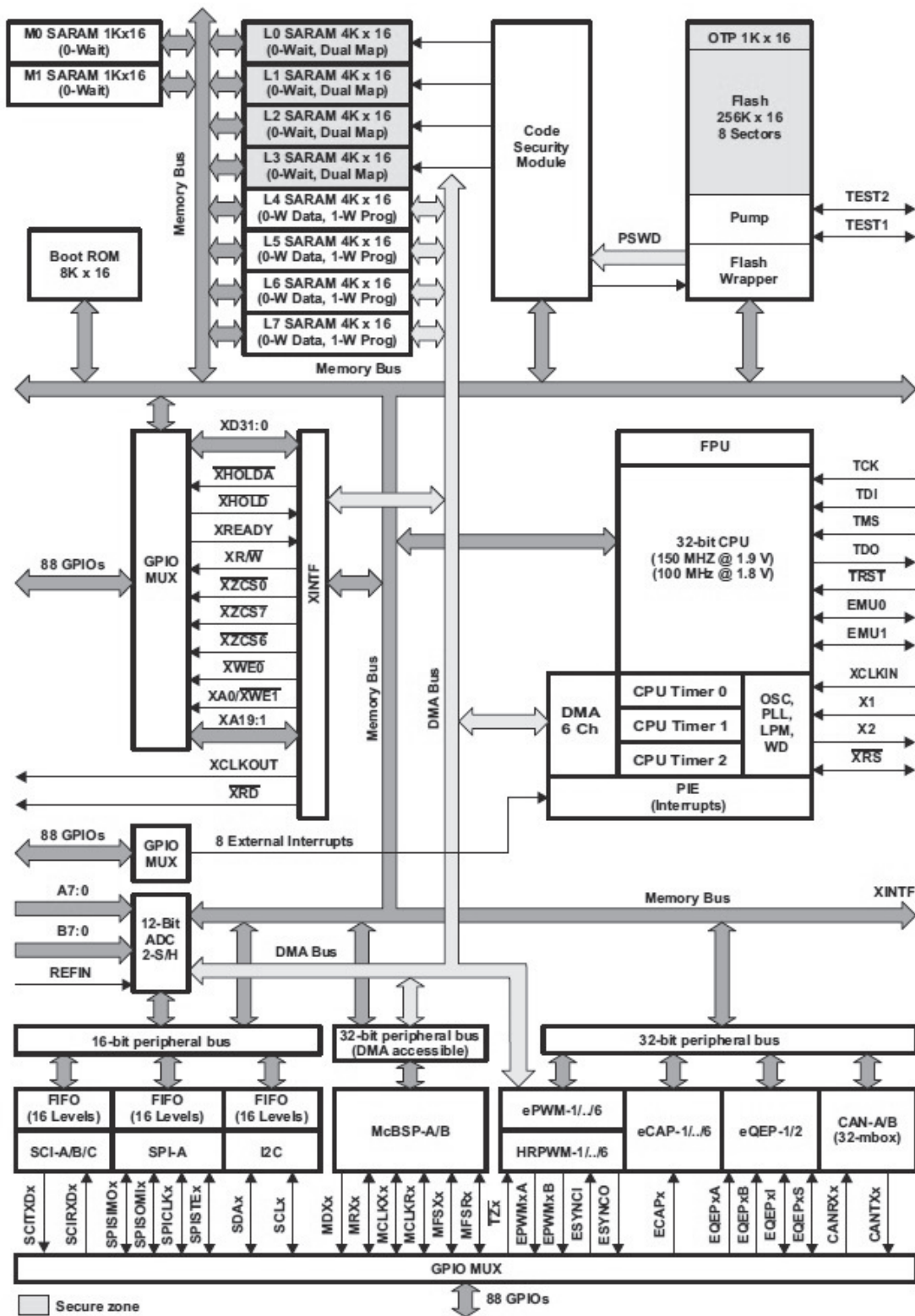


Figura 7.2 Esquema general de la arquitectura del DSP TMS320F28335 [27].

CPU C28x

El F28335 es miembro de la familia TMS320C2000. El núcleo C28x+FPU tiene la misma arquitectura de punto fijo a 32 bits que el resto de la familia C28x pero también incluye una unidad de punto flotante de precisión simple IEEE 754. La MAC de 32×32 permite al DSP manejar problemas de con alta precisión numérica eficientemente. La rápida respuesta a interrupciones y la forma automatizada en que guarda el contexto de los registros críticos hacen posible atender procesos asíncronos con un mínimo de retardo.

Buses

El F28335 usa múltiples buses para mover datos entre la memoria, los periféricos y el CPU. Como muestra la Figura 7.3, la arquitectura de la familia C28x contiene un bus de lectura de direcciones, un bus de lectura de datos y un bus de escritura de datos. El bus de lectura de programa consta de 22 líneas para direcciones y de 32 líneas para datos, los buses de lectura y escritura de datos son de 32 líneas de direcciones y 32 líneas de datos cada uno. Esta arquitectura permite hacer *fetch* a una instrucción, leer un dato y escribir un dato en un mismo ciclo.

Bloque de expansión de las interrupciones de periféricos (PIE)

El bloque PIE sirve para multiplexar las numerosas fuentes de interrupción en un conjunto más pequeño de entradas de interrupción. El bloque PIE multiplexa 96 interrupciones agrupadas en ocho grupos, cada grupo se alimenta de una de las 12 líneas de interrupción (INT1 a INT12), como se muestra en la Figura 7.4. Cada una de las 96 interrupciones posee su propio vector de interrupción que se localiza en un bloque de la RAM que puede ser modificado por el usuario. Al CPU le toma ocho ciclos de reloj hacer el *fetch* del vector de interrupción y guardar los registros críticos del CPU.

Reloj de periféricos

Los relojes que alimentan a cada módulo periféricos pueden ser desconectados individualmente para disminuir el consumo de energía. Además el reloj de cada periférico (excepto I2C y eCAN) pueden ser escalados a un múltiplo del reloj de CPU.

Módulo ADC

Como se muestra en la Figura 7.5, este módulo contiene un núcleo de conversión de 12 bits de resolución que puede convertir voltajes entre 0 y 3 V. Cada conversión se lleva a cabo en 80 ns cuando el reloj del ADC se configura a 25 MHz, lo que equivale a 12.5 millones de muestras por segundo (MSPS). El ADC posee dos circuitos muestreador-retenedor los cuales multiplexan ocho canales cada uno.

Este módulo se configura mediante sesiones de conversión, donde cada sesión consiste de una secuencia de conversiones consecutivas de un conjunto de los 16 canales

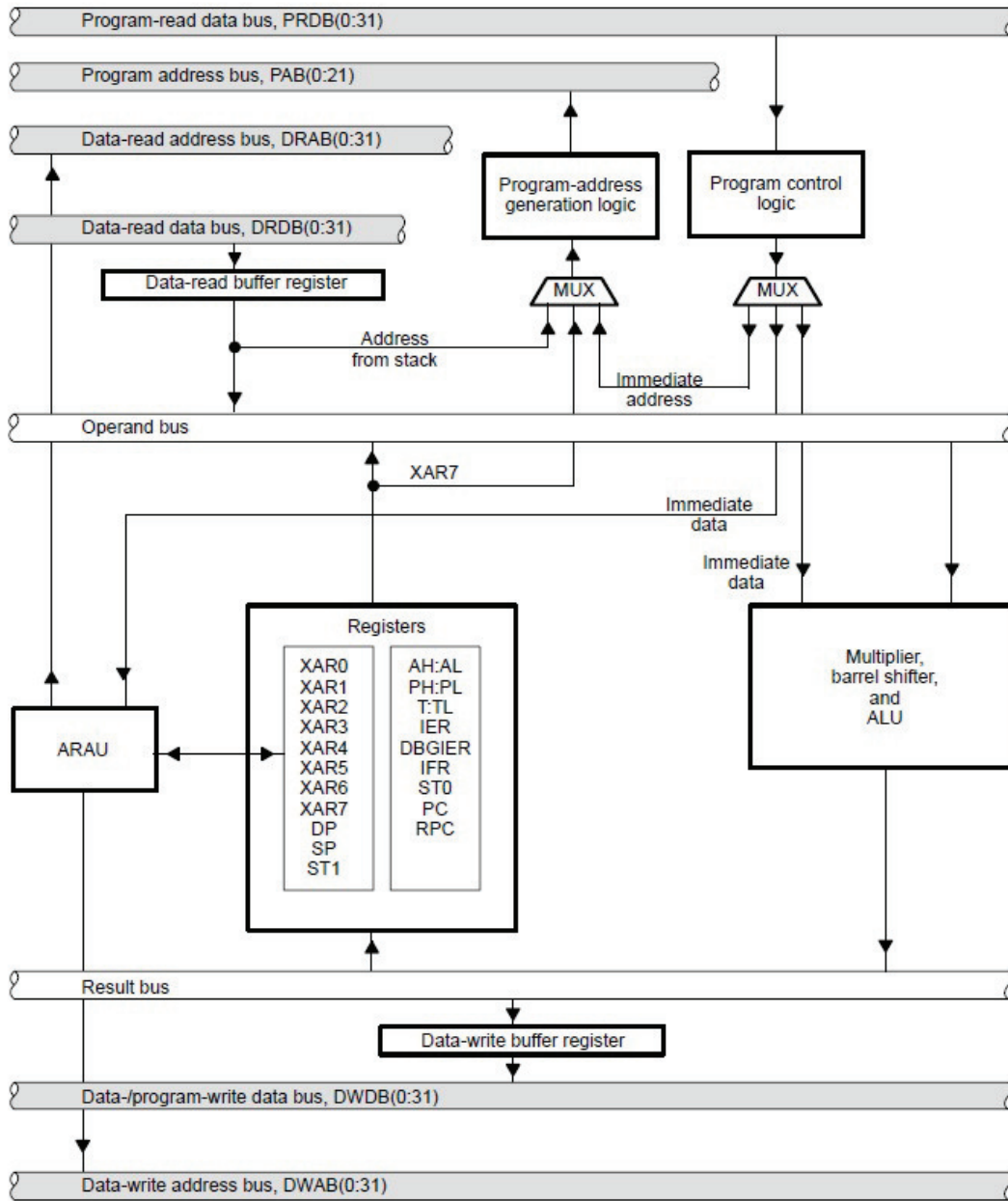


Figura 7.3 Esquema de buses [28].

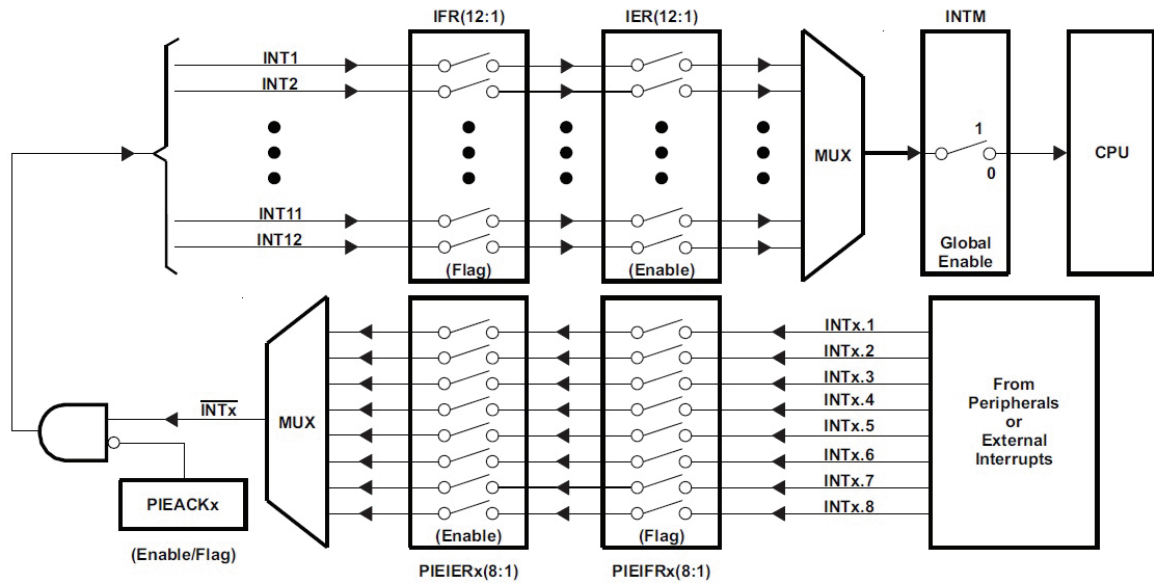


Figura 7.4 Esquema de multiplexación de interrupciones [32].

analógicos en el orden que indique el programador. Los resultados de la sesión de conversión se guardan en 16 registros especiales para esta función y son accesibles para el usuario.

El inicio de conversión (SOC) puede ser solicitado por múltiples fuentes como la interrupción XINT2, el módulo ePWM o por software al activar una bandera, además existen auto-secuencias que se inician automáticamente una vez que la sesión pasada ha terminado.

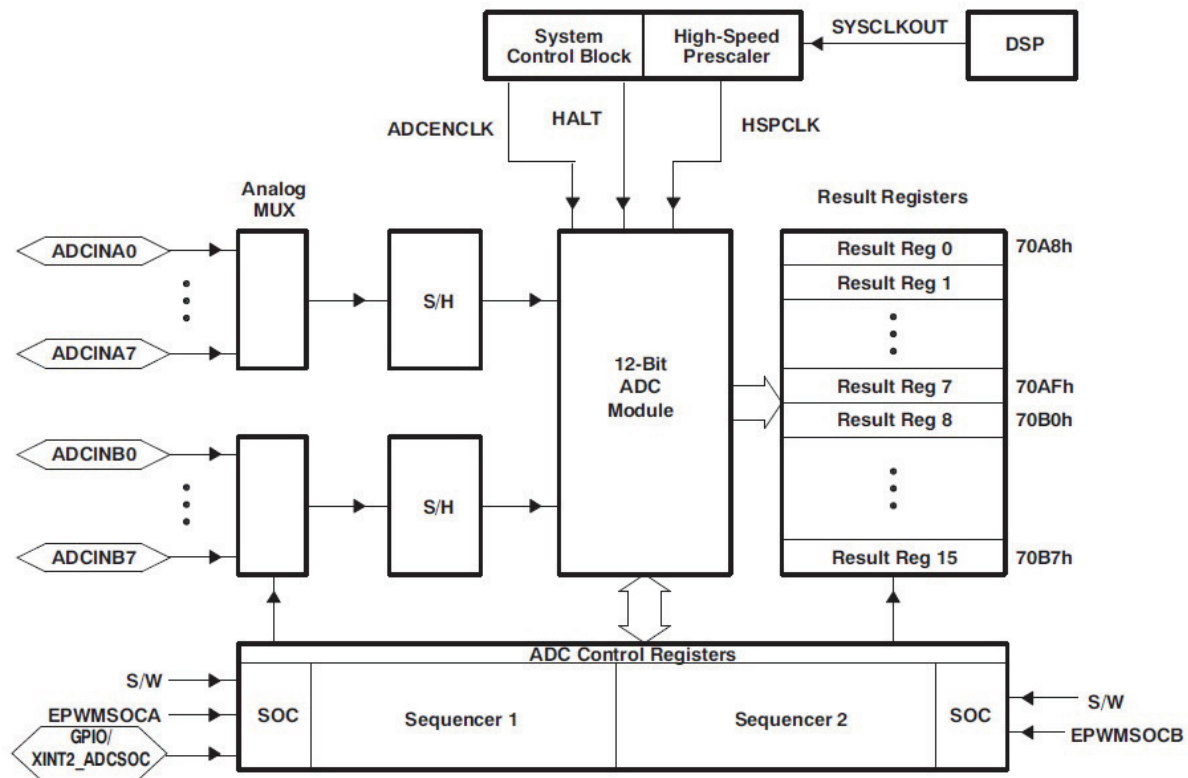


Figura 7.5 Esquema del módulo de conversión analógica-digital [30].

Códigos

Se presenta el código de los programas involucrados.

Programa principal del DSP

```
#include "DSP28x_Project.h"
//algunas constantes para configuracion de relojes, buffer de datos
#define ADC_MODCLK 0x3 // reloj de perifericos de alta velocidad
HSPCLK = SYSCLKOUT/2*ADC_MODCLK2 = 150/(2*3) = 25.0 MHz
#define ADC_CKPS 0x0 // reloj del ADC clock =
HSPCLK/2*ADC_CKPS = 25.0MHz/(1*1) = 25MHz
#define ADC_SHCLK 0x1 // periodos de sample and hold periods = 1 ADC clocks
#define AVG 100 // limite de muestras
#define BUF_SIZE 30 // tamaño total del buffer BUF_SIZE X 16

// Declaracion de vectores de adquisicion de datos
int BufferMic1Banda1[BUF_SIZE], BufferMic2Banda1[BUF_SIZE],
BufferMic3Banda1[BUF_SIZE], BufferMic4Banda1[BUF_SIZE],
BufferMic5Banda1[BUF_SIZE], BufferMic6Banda1[BUF_SIZE],
BufferMic7Banda1[BUF_SIZE], BufferMic8Banda1[BUF_SIZE],
```

```

BufferMic9Banda1[BUF_SIZE];
int BufferMic1Banda2[BUF_SIZE], BufferMic2Banda2[BUF_SIZE],
BufferMic3Banda2[BUF_SIZE], BufferMic4Banda2[BUF_SIZE],
BufferMic5Banda2[BUF_SIZE], BufferMic6Banda2[BUF_SIZE],
    BufferMic7Banda2[BUF_SIZE];
int BufferMic1Banda3[BUF_SIZE], BufferMic2Banda3[BUF_SIZE],
BufferMic3Banda3[BUF_SIZE], BufferMic4Banda3[BUF_SIZE],
BufferMic5Banda3[BUF_SIZE];
int BufferMic1Banda4[BUF_SIZE], BufferMic2Banda4[BUF_SIZE],
BufferMic3Banda4[BUF_SIZE], BufferMic4Banda4[BUF_SIZE];

// coeficientes de los filtros fir en Q15 c2
const int firbanda1[31]={0x0009, 0x0009, 0xFFA1, 0xFFCC, 0xFFED, 0xFFC6,
0xFF04, 0x02D7, 0x03BC, 0xFFD9, 0x023D, 0x0635, 0xF650, 0xE2D5, 0xF78B,
0x231C, 0x231C, 0xF78B, 0xE2D5, 0xF650, 0x0635, 0x023D, 0xFFD9, 0x03BC,
0x02D7, 0xFF04, 0xFFC6, 0xFFED, 0xFFCC, 0xFFA1, 0x0000};
const int firbanda2[31]={0x004B, 0x004B, 0x000D, 0xFFBC, 0xFF81, 0xFFB5,
0x0058, 0x0089, 0xFED3, 0xFA90, 0xF56B, 0xF316, 0xF73C, 0x0229, 0x0FA7,
0x18F2, 0x18F2, 0x0FA7, 0x0229, 0xF73C, 0xF316, 0xF56B, 0xFA90, 0xFED3,
0x0089, 0x0058, 0xFFB5, 0xFF81, 0xFFBC, 0x000D, 0x0000};
const int firbanda3[31]={0xFFED, 0xFFED, 0xFFB2, 0xFF3C, 0xFF77, 0xFD4A,
0xFBFC, 0FAEF, 0FAA7, 0FBA3, 0FE2C, 0x0226, 0x0710, 0x0C12, 0x1027,
0x1271, 0x1271, 0x1027, 0x0C12,0x0710, 0x0226, 0xFE2C, 0FBA3, 0FAA7,
0FAEF, 0xFBFC, 0FD4A, 0FE77, 0xFF3C, 0FFB2, 0x0000};
const int firbanda4[31]={0xFF3F, 0xFF3F, 0xFF42, 0xFF35, 0xFF33, 0xFF66,
0FFF0, 0x00E5,0x0251, 0x0428, 0x0652, 0x089D, 0x0AD4, 0x0CBF, 0x0E28,
0x0EE6, 0x0EE6, 0x0E28, 0x0CBF, 0x0AD4, 0x089D, 0x0652, 0x0428, 0x0251,
0x00E5, 0FFF0, 0xFF66, 0xFF33, 0xFF35, 0xFF42, 0x0000};

const int firinterpol2[33]={914, 914, 250, 20, -337, -509, -239, 387, 892,
718, -268, -1470, -1793, -353, 2792, 6460, 8921, 8921, 6460, 2792, -353,
-1793, -1470, -268, 718, 892, 387, -239, -509, -337, 20, 250, 0};

const int firinterpol3[65] = { -33, -33, -30, -44, -44, -24, 18, 69, 112,
123, 86, -1, -116, -223, -273, -228, -78, 149, 384, 536, 522, 295, -118,
-615, -1033, -1181, -904, -129, 1095, 2597, 4110, 5338, 6026, 6026, 5338,
4110, 2597, 1095, -129, -904, -1181, -1033, -615, -118, 295, 522, 536, 384,
149, -78, -228, -273, -223, -116, -1, 86, 123,112, 69, 18, -24, -44, -44,
-30, 0};

const int firinterpol4[129]= { -69, -69, -15, -15, -14, -11, -6, 1, 9, 18,
27, 35, 42, 47, 48, 45, 37, 25, 8,-12, -34, -57, -79, -97, -110, -115, -111,
-97, -73, -39, 3, 50, 101, 149, 192, 225, 244, 245, 225, 185, 123, 42, -53,
-158, -265, -366, -452, -514, -542, -531, -473, -364, -205, 5, 259, 552, 872,
1207, 1544, 1869, 2167, 2424, 2628, 2770, 2842, 2842, 2770, 2628, 2424, 2167,
1869, 1544, 1207, 872, 552, 259, 5, -205, -364, -473, -531, -542, -514, -452,
-366, -265, -158, -53, 42, 123, 185, 225, 245, 244, 225, 192, 149, 101, 50, 3,
-39, -73, -97, -111, -115, -110, -97, -79, -57, -34, -12, 8, 25, 37, 45, 48, 47,
42, 35, 27, 18, 9, 1, -6, -11, -14, -15, -15, 0};

//vectores de señales limitadas en banda

```

```
int BufferFir1Banda1[27], BufferFir2Banda1[27], BufferFir3Banda1[27],  
BufferFir4Banda1[27], BufferFir5Banda1[27], BufferFir6Banda1[27],  
BufferFir7Banda1[27], BufferFir8Banda1[27], BufferFir9Banda1[27];
```

```
int BufferFir1Banda2[21], BufferFir2Banda2[21], BufferFir3Banda2[21],  
BufferFir4Banda2[21], BufferFir5Banda2[21], BufferFir6Banda2[21],  
BufferFir7Banda2[21];
```

```
int BufferFir1Banda3[15], BufferFir2Banda3[15], BufferFir3Banda3[15],  
BufferFir4Banda3[15], BufferFir5Banda3[15];
```

```
int BufferFir1Banda4[11], BufferFir2Banda4[11], BufferFir3Banda4[11],  
BufferFir4Banda4[11];
```

```
Uint16 i;
```

```
//offsets para autocalibracion a cero del ADC
```

```
long int loffsetacon1 = 0;  
long int loffsetacon2 = 0;  
long int loffsetacon3 = 0;  
long int loffsetacon4 = 0;  
long int loffsetacon5 = 0;  
long int loffsetacon6 = 0;  
long int loffsetacon7 = 0;  
long int loffsetacon8 = 0;  
long int loffsetacon9 = 0;  
long int loffsetacon10 = 0;  
long int loffsetacon11 = 0;  
long int loffsetacon12 = 0;  
long int loffsetacon13 = 0;
```

```
int offsetacon1 = 0;  
int offsetacon2 = 0;  
int offsetacon3 = 0;  
int offsetacon4 = 0;  
int offsetacon5 = 0;  
int offsetacon6 = 0;  
int offsetacon7 = 0;  
int offsetacon8 = 0;  
int offsetacon9 = 0;  
int offsetacon10 = 0;  
int offsetacon11 = 0;  
int offsetacon12 = 0;  
int offsetacon13 = 0;
```

```
int perilladir = 0;
```

```
//potencias de distintas bandas
```

```
long int potcrudo = 0;  
long int auxcrudo = 0;  
long int auxbanda1 = 0;  
long int auxbanda2 = 0;
```



```

long int auxbanda3 = 0;
long int auxbandatot = 0;
long int auxbanda4 = 0;
long int potbanda1 = 0;
long int potbanda2 = 0;
long int potbanda3 = 0;
long int potbanda4 = 0;
long int potbandatot = 0;

//variables de apoyo en el calculo del formador de haz
long int apoyoventana = 0;
long int apoyoventana2 = 0;
long int apoyoventana3 =0;

//apoyo para sacar datos por el puerto A
long int apoyogpio = 0;
long int potnormalizada = 0;

//contadores para potencias
int contpot1 = 0;
int contpot2 = 0;
int contpot3 = 0;
int contpot4 = 0;
int contpot5 = 0;

int lol = 0;
int apun=18;
int dirdisplay = 8;
int sincdisplay = 0;
int potdisplay = 0;
int promedio=0;

//pesos de las ventanas
//Cheby 10 en Q15
const int w4che10[4]={0x7FFF, 0x6890, 0x6890, 0x7FFF};
//Cheby 30 en Q15
const int w1che[9]={0x2058, 0x3ABD, 0x5C15, 0x7621, 0x7FFF, 0x7621, 0x5C15,
0x3ABD, 0x2058};
const int w2che[7]={0x21D1, 0x48BD, 0x6FD8, 0x7FFF, 0x6FD8, 0x48BD, 0x21D1};
const int w3che[5]={0x28C4, 0x6257, 0x7FFF, 0x6257, 0x28C4};
const int w4che[4]={0x36E9, 0x7FFF, 0x7FFF, 0x36E9};

//matriz de retrasos

int MatRet[37][9]={{13,12,10,8,7,5,3,2,0},{13,12,10,8,7,5,3,2,0},
{13,11,10,8,7,5,3,2,0} , {13,11,10,8,6,5,3,2,0},{12,11,9,8,6,5,3,2,0},
{12,11,9,8,6,5,3,2,0},{12,10,9,7,5,4,3,1,0},{11,10,8,7,5,4,3,1,0},
{10,9,8,6,5,4,3,1,0},{9,8,7,6,5,4,2,1,0},{9,7,6,5,4,3,2,1,0},
{8,7,6,5,4,3,2,1,0},{7,6,5,4,3,2,2,1,0},{6,5,4,4,3,2,1,1,0},
{5,4,3,3,2,2,1,1,0},{3,3,3,2,2,1,1,0,0},{2,2,2,1,1,1,1,0,0},
{1,1,1,1,1,0,0,0,0},{0,0,0,0,0,0,0,0,0},{0,0,0,0,1,1,1,1,1},

```

```

{0,0,1,1,1,1,2,2,2},{0,0,1,1,2,2,3,3,3},{0,1,1,2,2,3,3,4,5},
{0,1,1,2,3,4,4,5,6},{0,1,2,2,3,4,5,6,7},{0,1,2,3,4,5,6,7,8},
{0,1,2,3,4,5,6,7,9},{0,1,2,4,5,6,7,8,9},{0,1,3,4,5,6,8,9,10},
{0,1,3,4,5,7,8,10,11},{0,1,3,4,6,7,9,10,12},{0,2,3,5,6,8,9,11,12},
{0,2,3,5,6,8,9,11,12},{0,2,3,5,6,8,10,11,13},{0,2,3,5,7,8,10,11,13},
{0,2,3,5,7,8,10,12,13},{0,2,3,5,7,8,10,12,13}};

//variable para elegir el renglo de la MatRet
int direccion = 0;
int offsetaux = 0;

//resiltados de los formadores de haz por banda
int haz1 = 0;
int haz2 = 0;
int haz3 = 0;
int haz4 = 0;
int hazprueba1 = 0;
int hazprueba2 = 0;
int hazprueba3 = 0;
int hazprueba4 = 0;
int hazpruebatot = 0;

//variable puntero para el nuevo valor de los buffers circulares
int nuevovalor = 0;
int nuevovalorneg = 0;
int nuevovalorant = 0;
//variables para acomodar los buffers circulares de las señales limitadas en
banda para los formadores de haz
int nuevofirbanda1 =13;
int offsetdirecbanda1=0;
int nuevofirbanda2 =10;
int offsetdirecbanda2=0;
int nuevofirbanda3 =7;
int offsetdirecbanda3=0;
int nuevofirbanda4 =5;
int offsetdirecbanda4=0;
//variables para decidir submuestreo, decide si filtrar o saltar
int subsamp2=0;
int subsamp3=0;
int subsamp4=0;
int reacciona=32;
//vectores de muestras del resultado del formador de haz sobre-muestreados
int BufferHaz1[32];
int BufferHaz2[32];
int BufferHaz3[64];
int BufferHaz4[128];
int BufferHazFinal[64];
int BufferHazPruebas[64];
int BufferHazPruebas2[64];

//indice para buffers de los haces sobremuestreados
int nuevohaz = 0;

```

```
int nuevohazneg = 0;
int nuevohaz4neg =0;
int nuevohaz4 =0;
int nuevohaz5neg =0;
int nuevohaz5 =0;

//valores finales del formador de haz
int final1 = 0;
int final2 = 0;
int final3 = 0;
int final4 = 0;
int finaltotal = 0;
int potsalida = 0;
int jk = 0;
int Bufferfinal[64];
int nuevobufferfinal=0;

// declaracion de la rutina de interrupcion del timer 0
interrupt void cpu_timer0_isr(void);

//declaracion de las subrutinas externas en ensamblador
extern void fir (void);
extern void fir2 (void);
extern void fir3 (void);
extern void fir4 (void);
extern void firint2 (void);
extern void firint3 (void);
extern void firint4 (void);

void main(void){
//rutina proveida por TI, configura PLL a 150mhz, desactiva WD y habilita el reloj
de los perifericos
    InitSysCtrl();

// configura el reloj de los perifericos de acuerdo con las constantes definidas
    EALLOW;
    SysCtrlRegs.HISPCP.all = ADC_MODCLK; // HSPCLK =
    SYSCLKOUT/ADC_MODCLK
    EDIS;

//Deshabilita las interrupciones e inicializa el vector PIE
    DINT;
    InitPieCtrl();
    IER = 0x0000;
    IFR = 0x0000;
    InitPieVectTable();

//enlaza la rutina de interrupcion
    EALLOW; // permite acceder a registros protegidos
    PieVectTable.TINT0 = &cpu_timer0_isr;
    EDIS; // protege registros
```

```

// inicializa los timers del CPU con una rutina de TI
    InitCpuTimers();

#if (CPU_FRQ_150MHZ)
// Configura el CPU-Timer 0 para interrumpir cada 62 microsegundos
    ConfigCpuTimer(&CpuTimer0, 150, 62);
#endif

//activa al timer 0 y habilita su interrupcion
    CpuTimer0Regs.TCR.all = 0x4001; // set TSS bit = 0 para activar
    timer y TIE=1 para activar interrupcion

//*****CONFIGURACION ADC*****
//inicia el convertidor analogico digital
    InitAdc(); // prende el convertidor
// configura ancho de la ventana sample and hold
    AdcRegs.ADCCTRL1.bit.ACQ_PS = ADC_SHCLK;
//prescalador del reloj del ADC
    AdcRegs.ADCCTRL3.bit.ADCCLKPS = ADC_CKPS;
//configura a modo cascada, los 16 canales funcionan con una sola secuencia
    AdcRegs.ADCCTRL1.bit.SEQ_CASC = 1;
//enlaza cada start of conversion de la secuencia con un canal del ADC
    AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0x0;
    AdcRegs.ADCCHSELSEQ1.bit.CONV01 = 0x1;
    AdcRegs.ADCCHSELSEQ1.bit.CONV02 = 0x2;
    AdcRegs.ADCCHSELSEQ1.bit.CONV03 = 0x3;
    AdcRegs.ADCCHSELSEQ2.bit.CONV04 = 0x4;
    AdcRegs.ADCCHSELSEQ2.bit.CONV05 = 0x5;
    AdcRegs.ADCCHSELSEQ2.bit.CONV06 = 0x6;
    AdcRegs.ADCCHSELSEQ2.bit.CONV07 = 0x7;
    AdcRegs.ADCCHSELSEQ3.bit.CONV08 = 0x8;
    AdcRegs.ADCCHSELSEQ3.bit.CONV09 = 0x9;
    AdcRegs.ADCCHSELSEQ3.bit.CONV10 = 0xA;
    AdcRegs.ADCCHSELSEQ3.bit.CONV11 = 0xB;
    AdcRegs.ADCCHSELSEQ4.bit.CONV12 = 0xC;
    AdcRegs.ADCCHSELSEQ4.bit.CONV13 = 0xD;
    AdcRegs.ADCCHSELSEQ4.bit.CONV14 = 0xE;
    AdcRegs.ADCCHSELSEQ4.bit.CONV15 = 0xF;
//configura para que convierta indefinidamente
    AdcRegs.ADCCTRL1.bit.CONT_RUN = 1;
//configura que cada secuencia tiene 16 SOCs
    AdcRegs.ADCMAXCONV.all = 0xf;

//*****CONFIGURACION GPIO*****
    EALLOW;
//Gpio 0 al 19 + 31(led de la controlcard) como entradas/salidas de uso general
    GpioCtrlRegs.GPAMUX1.bit.GPIO0 = 0;
    GpioCtrlRegs.GPAMUX1.bit.GPIO1 = 0;
    GpioCtrlRegs.GPAMUX1.bit.GPIO2 = 0;
    GpioCtrlRegs.GPAMUX1.bit.GPIO3 = 0;
    GpioCtrlRegs.GPAMUX1.bit.GPIO4 = 0;
    GpioCtrlRegs.GPAMUX1.bit.GPIO5 = 0;

```

```
GpioCtrlRegs.GPAMUX1.bit.GPIO6 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO7 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO8 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO9 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO10 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO11 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO12 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO13 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO14 = 0;
GpioCtrlRegs.GPAMUX1.bit.GPIO15 = 0;
GpioCtrlRegs.GPAMUX2.bit.GPIO16 = 0;
GpioCtrlRegs.GPAMUX2.bit.GPIO17 = 0;
GpioCtrlRegs.GPAMUX2.bit.GPIO18 = 0;
GpioCtrlRegs.GPAMUX2.bit.GPIO19 = 0;
GpioCtrlRegs.GPAMUX2.bit.GPIO31 = 0;
// las GPIO configuradas como salidas
GpioCtrlRegs.GPADIR.bit.GPIO0 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO1 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO2 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO3 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO4 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO5 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO6 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO7 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO8 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO9 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO10 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO11 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO12 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO13 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO14 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO15 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO16 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO17 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO18 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO19 = 1;
GpioCtrlRegs.GPADIR.bit.GPIO31 = 1;
EDIS;

//activa SOC SEQ1, como esta en cont run no hace falta activar SOC otra vez
AdcRegs.ADCTRL2.all = 0x2000;

// limpia los buffers de datos de las 4 bandas
for (promedio=0;promedio<512;promedio++){

    for (i=0; i<BUF_SIZE; i++){
        BufferMic1Banda1[i] = 1;
        BufferMic2Banda1[i] = 1;
        BufferMic3Banda1[i] = 1;
        BufferMic4Banda1[i] = 1;
        BufferMic5Banda1[i] = 1;
        BufferMic6Banda1[i] = 1;
    }
}
```

```

    BufferMic7Banda1[i] = 1;
    BufferMic8Banda1[i] = 1;
    BufferMic9Banda1[i] = 1;
}
for (i=0; i<BUF_SIZE; i++){
    BufferMic1Banda2[i] = 1;
    BufferMic2Banda2[i] = 1;
    BufferMic3Banda2[i] = 1;
    BufferMic4Banda2[i] = 1;
    BufferMic5Banda2[i] = 1;
    BufferMic6Banda2[i] = 1;
    BufferMic7Banda2[i] = 1;
}
for (i=0; i<BUF_SIZE; i++){
    BufferMic1Banda3[i] = 1;
    BufferMic2Banda3[i] = 1;
    BufferMic3Banda3[i] = 1;
    BufferMic4Banda3[i] = 1;
    BufferMic5Banda3[i] = 1;
}
for (i=0; i<BUF_SIZE; i++){
    BufferMic1Banda4[i] = 1;
    BufferMic2Banda4[i] = 1;
    BufferMic3Banda4[i] = 1;
    BufferMic4Banda4[i] = 1;
}

//Captura los valores iniciales en los canales. Al comienzo debe estar en
//silencio para que este valor sirva para quitar el offset de la señal digital
loffsetacon1 = loffsetacon1 + (AdcRegs.ADCRESULT3>>4);
loffsetacon2 = loffsetacon2 + (AdcRegs.ADCRESULT4>>4);
loffsetacon3 = loffsetacon3 + (AdcRegs.ADCRESULT5>>4);
loffsetacon4 = loffsetacon4 + (AdcRegs.ADCRESULT6>>4);
loffsetacon5 = loffsetacon5 + (AdcRegs.ADCRESULT7>>4);
loffsetacon6 = loffsetacon6 + (AdcRegs.ADCRESULT8>>4);
loffsetacon7 = loffsetacon7 + (AdcRegs.ADCRESULT9>>4);
loffsetacon8 = loffsetacon8 + (AdcRegs.ADCRESULT10>>4);
loffsetacon9 = loffsetacon9 + (AdcRegs.ADCRESULT11>>4);
loffsetacon10 = loffsetacon11 + (AdcRegs.ADCRESULT12>>4);
loffsetacon11 = loffsetacon11 + (AdcRegs.ADCRESULT13>>4);
loffsetacon12 = loffsetacon12 + (AdcRegs.ADCRESULT14>>4);
loffsetacon13 = loffsetacon13 + (AdcRegs.ADCRESULT15>>4);
}

offsetacon1 = loffsetacon1>>9;
offsetacon2 = loffsetacon2>>9;
offsetacon3 = loffsetacon3>>9;
offsetacon4 = loffsetacon4>>9;
offsetacon5 = loffsetacon5>>9;
offsetacon6 = loffsetacon6>>9;
offsetacon7 = loffsetacon7>>9;
offsetacon8 = loffsetacon8>>9;

```

```

offsetacon9 = loffsetacon9>>9;
offsetacon10 = loffsetacon10>>9;
offsetacon11 = loffsetacon11>>9;
offsetacon12 = loffsetacon12>>9;
offsetacon13 = loffsetacon13>>9;

//*****ACTIVA INTERRUPCIONES*****
// habilita la interrupcion INT1 la cual esta conectada con el CPU-Timer 0:
    IER |= M_INT1;
// Habilita la TINTO en el PIE: Grupo 1 interrupcion 7
    PieCtrlRegs.PIEIER1.bit.INTx7 = 1;
// Habilita interrupciones globales
    EINT;    // Enable INTM
    ERTM;    // Enable DBGM

//empieza ciclo infinito
    for(;;) {
        }
//fin del main
}

////////////////////////////////////

//rutina de interrupcion del timer 0 cada interrupcion comienza el proceso de filtrado
//para un termino nuevo en la secuencia digital de la señal de voz
interrupt void cpu_timer0_isr(void){
    CpuTimer0.InterruptCount++;

////////////////////////////////////
    //limpia la bandera de la interrupcion cuando acaba una secuencia
    AdcRegs.ADCST.bit.INT_SEQ1_CLR = 1;

    //se decide el offset para el nueva dato en todos los buffers de adquisicion
    if (nuevovalor==29){
        nuevovalor=0;
        nuevovalorneg=29 - nuevovalor;
        nuevovalorant=nuevovalor-1;
    }else{
        nuevovalor++;
        nuevovalorneg=29 - nuevovalor;
        nuevovalorant=nuevovalor-1;
    }
    if (nuevohaz==31){
        nuevohaz=0;
        nuevohazneg=31;
    }else{
        nuevohaz++;
        nuevohazneg = 31-nuevohaz;
    }
}
if (nuevohaz4==63){
    nuevohaz4=0;
}

```

```

    nuevohaz4neg=63;
}else{
    nuevohaz4++;
    nuevohaz4neg=63-nuevohaz4;
}
if (nuevohaz5==127){
    nuevohaz5=0;
    nuevohaz5neg=127;
}else{
    nuevohaz5++;
    nuevohaz5neg=127-nuevohaz5;
}

//se trasladan los datos nuevos al buffer circular, como los ADCRESULT
//están justificados a la derecha se recorren 4 lugares, se le resta un
//valor para quitar el offset proveniente del acondicionador de señal
//Banda 1
BufferMic1Banda1[nuevovalor] =((AdcRegs.ADCRESULT3>>4) - offsetacon1);
BufferMic2Banda1[nuevovalor] =((AdcRegs.ADCRESULT4>>4) - offsetacon2);
BufferMic3Banda1[nuevovalor] =((AdcRegs.ADCRESULT5>>4) - offsetacon3);
BufferMic4Banda1[nuevovalor] =((AdcRegs.ADCRESULT6>>4) - offsetacon4);
BufferMic5Banda1[nuevovalor] =((AdcRegs.ADCRESULT7>>4) - offsetacon5);
BufferMic6Banda1[nuevovalor] =((AdcRegs.ADCRESULT8>>4) - offsetacon6);
BufferMic7Banda1[nuevovalor] =((AdcRegs.ADCRESULT9>>4) - offsetacon7);
BufferMic8Banda1[nuevovalor] =((AdcRegs.ADCRESULT10>>4) - offsetacon8);
BufferMic9Banda1[nuevovalor] =((AdcRegs.ADCRESULT11>>4) - offsetacon9);
//banda 2
BufferMic1Banda2[nuevovalor] =((AdcRegs.ADCRESULT3>>4) - offsetacon1);
BufferMic2Banda2[nuevovalor] =((AdcRegs.ADCRESULT5>>4) - offsetacon3);
BufferMic3Banda2[nuevovalor] =((AdcRegs.ADCRESULT7>>4) - offsetacon5);
BufferMic4Banda2[nuevovalor] =((AdcRegs.ADCRESULT9>>4) - offsetacon7);
BufferMic5Banda2[nuevovalor] =((AdcRegs.ADCRESULT11>>4) - offsetacon9);
BufferMic6Banda2[nuevovalor] =((AdcRegs.ADCRESULT12>>4) - offsetacon10);
BufferMic7Banda2[nuevovalor] =((AdcRegs.ADCRESULT13>>4) - offsetacon11);
//banda3
BufferMic1Banda3[nuevovalor] =((AdcRegs.ADCRESULT3>>4) - offsetacon1);
BufferMic2Banda3[nuevovalor] =((AdcRegs.ADCRESULT7>>4) - offsetacon5);
BufferMic3Banda3[nuevovalor] =((AdcRegs.ADCRESULT11>>4) - offsetacon9);
BufferMic4Banda3[nuevovalor] =((AdcRegs.ADCRESULT13>>4) - offsetacon11);
BufferMic5Banda3[nuevovalor] =((AdcRegs.ADCRESULT14>>4) - offsetacon12);
//banda 4
BufferMic1Banda4[nuevovalor] =((AdcRegs.ADCRESULT3>>4) - offsetacon1);
BufferMic2Banda4[nuevovalor] =((AdcRegs.ADCRESULT11>>4) - offsetacon9);
BufferMic3Banda4[nuevovalor] =((AdcRegs.ADCRESULT14>>4) - offsetacon12);
BufferMic4Banda4[nuevovalor] =((AdcRegs.ADCRESULT15>>4) - offsetacon13);

//potencia del valor crudo del mic 1 contpot1=contpot1 + 1;
if (contpot1 == 600){
    potcrudo = auxcrudo;
    contpot1 = 0;
    auxcrudo = 0;
}

```



```
auxcrudo = auxcrudo + (BufferMic3Banda4[nuevovalor]>>3) *
(BufferMic3Banda4[nuevovalor]>>3);

//se submuestra. para no filtrar secuencias que por el submuestreo se van a desechar,
//se filtra solo los valores que la tasa de submuestreo pide
//submuestreos banda 1
//*La banda 1 se procesa a 16,129Hz y no necesita submuestreo. Se decide el offset para
//los nuevos datos de los buffers circulares de señales limitadas en banda
    if (nuevofirbanda1==26){
        nuevofirbanda1=0;
    }else{
        nuevofirbanda1=nuevofirbanda1 +1;
    }
    if (offsetdirecbanda1==26){
        offsetdirecbanda1=0;
    }else{
        offsetdirecbanda1=offsetdirecbanda1 +1;
    }
    //subrutina externa en ensamblador para filtrar la banda1
fir();

//submuestreos banda 2 *La banda 2 se procesa a la mitad 16,129Hz
subsamp2=subsamp2+1;
if(subsamp2==2){

    if (nuevofirbanda2==20){
        nuevofirbanda2=0;
    }else{
        nuevofirbanda2=nuevofirbanda2 +1;
    }
    if (offsetdirecbanda2==20){
        offsetdirecbanda2=0;
    }else{
        offsetdirecbanda2=offsetdirecbanda2 +1;
    }
    //subrutina externa en ensamblador para filtrar la banda2
fir2();
subsamp2=0;
}

//submuestreos banda 3
//*La banda 3 se procesa a una cuarta parte de 16,129Hz
subsamp3=subsamp3+1;
if(subsamp3==4){
    if (nuevofirbanda3==14){
        nuevofirbanda3=0;
    }else{
        nuevofirbanda3=nuevofirbanda3 +1;
    }
    if (offsetdirecbanda3==14){
        offsetdirecbanda3=0;
    }else{
```



```
//////////3
offsetaux=offsetdirecbanda1 + MatRet[direccion] [2];
if(offsetaux>26){
    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir3Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[2];
apoyoventana2 =apoyoventana2 + apoyoventana;
//////////4
offsetaux=offsetdirecbanda1 + MatRet[direccion] [3];
if(offsetaux>26){
    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir4Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[3];

apoyoventana2 =apoyoventana2 + apoyoventana;
//////////5
offsetaux=offsetdirecbanda1 + MatRet[direccion] [4];
if(offsetaux>26){
    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir5Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[4];
apoyoventana2 =apoyoventana2 + apoyoventana;
//////////6
offsetaux=offsetdirecbanda1 + MatRet[direccion] [5];
if(offsetaux>26){
    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir6Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[5];
apoyoventana2 =apoyoventana2 + apoyoventana;
//////////7
offsetaux=offsetdirecbanda1 + MatRet[direccion] [6];
if(offsetaux>26){
    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir7Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[6];
apoyoventana2 =apoyoventana2 + apoyoventana;
//////////8
offsetaux=offsetdirecbanda1 + MatRet[direccion] [7];
if(offsetaux>26){
    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir8Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[7];
apoyoventana2 =apoyoventana2 + apoyoventana;
//////////9
offsetaux=offsetdirecbanda1 + MatRet[direccion] [8];
if(offsetaux>26){
```

```

    offsetaux=offsetaux-27;
}
apoyoventana3 = BufferFir9Banda1[offsetaux];
apoyoventana = apoyoventana3*w1che[8];
apoyoventana2 =apoyoventana2 + apoyoventana;

apoyoventana2=apoyoventana2>>15;
haz1 = apoyoventana2;
BufferHaz1[nuevohaz]=haz1;

//formador de haz banda 2
if(subsamp2==0){
haz2 = 0;
apoyoventana = 0;
    apoyoventana2 = 0;
    apoyoventana3 = 0;
    apoyoventana3=BufferFir1Banda2[offsetdirecbanda2];
apoyoventana = apoyoventana3*w2che[0];
apoyoventana2 = apoyoventana2+apoyoventana;

//////////2
offsetaux=offsetdirecbanda2 + MatRet[direccion][1];
if(offsetaux>20){
    offsetaux=offsetaux-21;
}
apoyoventana3=BufferFir2Banda2[offsetaux];
apoyoventana = apoyoventana3*w2che[1];
apoyoventana2 = apoyoventana2+apoyoventana;

//////////3
offsetaux=offsetdirecbanda2 + MatRet[direccion][2];
if(offsetaux>20){
    offsetaux=offsetaux-21;
}
apoyoventana3=BufferFir3Banda2[offsetaux];
apoyoventana = apoyoventana3*w2che[2];
apoyoventana2 = apoyoventana2+apoyoventana;

haz2=haz2+BufferFir3Banda2[offsetaux]*w2[2];
//////////4
offsetaux=offsetdirecbanda2 + MatRet[direccion][3];
if(offsetaux>20){
    offsetaux=offsetaux-21;
}
apoyoventana3=BufferFir4Banda2[offsetaux];
apoyoventana = apoyoventana3*w2che[3];
apoyoventana2 = apoyoventana2+apoyoventana;

//////////5
offsetaux=offsetdirecbanda2 + MatRet[direccion][4];
if(offsetaux>20){
    offsetaux=offsetaux-21;
}

```

```

}
apoyoventana3=BufferFir5Banda2[offsetaux];
apoyoventana = apoyoventana3*w2che[4];
apoyoventana2 = apoyoventana2+apoyoventana;

//////////6
offsetaux=offsetdirecbanda2 + MatRet[direccion] [5];
if(offsetaux>20){
    offsetaux=offsetaux-21;
}
apoyoventana3=BufferFir6Banda2[offsetaux];
apoyoventana = apoyoventana3*w2che[5];
apoyoventana2 = apoyoventana2+apoyoventana;

//////////7
offsetaux=offsetdirecbanda2 + MatRet[direccion] [6];
if(offsetaux>20){
    offsetaux=offsetaux-21;
}
apoyoventana3=BufferFir7Banda2[offsetaux];
apoyoventana = apoyoventana3*w2che[6];
apoyoventana2 = apoyoventana2+apoyoventana;
apoyoventana2 = apoyoventana2>>15;
haz2=apoyoventana2;

BufferHaz2[nuevohaz]=haz2;
}else{
    BufferHaz2[nuevohaz]=0;
}

//formador de haz banda 3
if(subsamp3==0){
apoyoventana = 0;
    apoyoventana2 = 0;
    apoyoventana3 = 0;
haz3 = 0;
apoyoventana3=BufferFir1Banda3[offsetdirecbanda3];
    apoyoventana = apoyoventana3*w3che[0];
    apoyoventana2=apoyoventana2+apoyoventana;
//////////2
offsetaux=offsetdirecbanda3 + MatRet[direccion] [1];
if(offsetaux>14){
    offsetaux=offsetaux-15;
}
apoyoventana3=BufferFir2Banda3[offsetaux];
apoyoventana=apoyoventana3*w3che[1];
apoyoventana2=apoyoventana2 + apoyoventana;
//////////3
offsetaux=offsetdirecbanda3 + MatRet[direccion] [2];
if(offsetaux>14){
    offsetaux=offsetaux-15;
}

```

```

apoyoventana3=BufferFir3Banda3[offsetaux];
apoyoventana=apoyoventana3*w3che[2];
apoyoventana2=apoyoventana2 + apoyoventana;

//////////4
offsetaux=offsetdirecbanda3 + MatRet[direccion][3];
if(offsetaux>14){
    offsetaux=offsetaux-15;
}
apoyoventana3=BufferFir4Banda3[offsetaux];
apoyoventana=apoyoventana3*w3che[3];
apoyoventana2=apoyoventana2 + apoyoventana;

//////////5
offsetaux=offsetdirecbanda3 + MatRet[direccion][4];
if(offsetaux>14){
    offsetaux=offsetaux-15;
}
apoyoventana3=BufferFir5Banda3[offsetaux];
apoyoventana=apoyoventana3*w3che[4];
apoyoventana2=apoyoventana2 + apoyoventana;
apoyoventana2=apoyoventana2>>15;
haz3=apoyoventana2;
BufferHaz3[nuevohaz4]=haz3;
}else{
BufferHaz3[nuevohaz4]=0;
}

//formador de haz banda 4
if(subsamp4==0){
haz4 = 0;
apoyoventana = 0;
    apoyoventana2 = 0;
    apoyoventana3 = 0;
    apoyoventana3=BufferFir1Banda4[offsetdirecbanda4];
apoyoventana = apoyoventana3*w4che10[0];
apoyoventana2=apoyoventana2+apoyoventana;
//////////2

offsetaux=offsetdirecbanda4 + MatRet[direccion][1];
if(offsetaux>10){
    offsetaux=offsetaux-11;
}
apoyoventana3=BufferFir2Banda4[offsetaux];
apoyoventana=apoyoventana3*w4che10[1];
apoyoventana2=apoyoventana2+apoyoventana;
//////////3
offsetaux=offsetdirecbanda4 + MatRet[direccion][2];
if(offsetaux>10){
    offsetaux=offsetaux-11;
}
apoyoventana3=BufferFir3Banda4[offsetaux];

```

```

apoyoventana=apoyoventana3*w4che10[2];
apoyoventana2=apoyoventana2+apoyoventana;

//////////4
offsetaux=offsetdirecbanda4 + MatRet[direccion][3];
if(offsetaux>10){
    offsetaux=offsetaux-11;
}
apoyoventana3=BufferFir4Banda4[offsetaux];
apoyoventana=apoyoventana3*w4che10[3];
apoyoventana2=apoyoventana2+apoyoventana;
apoyoventana2=apoyoventana2>>15;
haz4=apoyoventana2;

    BufferHaz4[nuevohaz5]=haz4;
}else{
BufferHaz4[nuevohaz5]=0;
}

//////////aqui interpolacion de bandas por filtros FIR//////////
//interpolacion banda 2
firint2();

//interpolacion banda 3
firint3();

//interpolacion banda 4
firint4();

//////////aqui conformacion del haz final//////////

//compensa la amplitud
hazprueba1 = (haz1>>2);
hazprueba2 = final2;
hazprueba3 = (final3>>3);
hazprueba4 = (final4>>1);

hazpruebatot = 500 + (haz1>>2)+ final2 + (final3>>3) + (final4>>1);

//limitador de amplitud
if (hazpruebatot<1){
hazpruebatot = 0;
}
if (hazpruebatot>1000){
hazpruebatot = 1000;
}

BufferHazPruebas[nuevohaz4]=hazpruebatot;
BufferHazPruebas2[nuevohaz4]=hazprueba4;

//////////aqui se calcula la potencia algunas señalesde interes//////////
potsalida=hazpruebatot-500;

```

```

contpot2=contpot2 + 1;
  if (contpot2 == 4800){
    potbanda1 = auxbanda1>>11;
    auxbanda1 = 0;
    potbanda2 = auxbanda2>>10;
    auxbanda2 = 0;
    potbanda3 = auxbanda3>>10;
    auxbanda3 = 0;
    potbanda4 = auxbanda4>>10;
    auxbanda4 = 0;
    potbandatot = auxbandatot>>10;
    auxbandatot = 0;
    contpot2 = 0;
  }
  auxbandatot = auxbandatot + (potsalida>>4)*(potsalida>>4);
auxbanda2 = auxbanda2 + (final2>>3)*(final2>>3);
auxbanda3 = auxbanda3 + (final3>>2)*(final3>>2);
auxbanda4 = auxbanda4 + (final4>>1)*(final4>>1);
auxbanda1 = auxbanda1 + (haz1>>4)*(haz1>>4);

//////////aquí se mandan los datos en paralelo para mostrarlos en la GLCD
//captura de la direccion solicitada por el usuario
perilladir =(AdcRegs.ADCRESULT0>>4);
//aumenta el contador para la sincronia en la comunicacion al PIC
sincdisplay=sincdisplay +1;

if(sincdisplay==16000){
//conversion del voltaje en la perilla a la direccion para el PIC y el
//algoritmo de filtrado
if(perilladir<2080){
dirdisplay=15;
apun=32;
if(perilladir<1950){
dirdisplay=14;
apun=30;
if(perilladir<1820){
dirdisplay=13;
apun=28;
if(perilladir<1690){
dirdisplay=12;
apun=26;
if(perilladir<1560){
dirdisplay=11;
apun=24;
if(perilladir<1430){
dirdisplay=10;
apun=22;
if(perilladir<1300){
dirdisplay=9;
apun=20;
if(perilladir<1170){
dirdisplay=8;

```



```

_firint4:
;este bloque carga en ARs las direcciones de los vectores de coefs
; y el de datos con su offset
ZAPA ;cero al acumulador
MOVL XAR7,#_firinterp04 ;AR7 apunta a coefs
MOVL XAR3,#_BufferHaz4 ;AR3 a datos con offset=0
MOVL XAR5,#_nuevohaz5 ;AR5 a loc donde esta el offset
MOVL ACC,XAR3
ADD ACC,*XAR5 ;suma offset y loc datos
MOVL XAR2,ACC ;loc datos con ofsset en AR2
;para implementar un buffer circular debo saber donde esta el dato
;mas nuevo, ese dato esta en _nuevohaz5, nuevohaz5neg=129-nuevohaz5 la
;longitud del buffer es 129
MOVL XAR5,#_nuevohaz5neg ;se multiplica y acumula
ZAPA ;0 a nuevoneg del fir
RPT *XAR5 ;y de nuevohaz5 a 128 de datos
  ||MAC P,*XAR2+,*XAR7++
    MOVL XAR5,#_nuevohaz5 ;ahora se multiplica y acumula
    MOVL XAR2,#_BufferHaz4 ;de nuevohaz5neg a 128 del fir
    RPT *XAR5 ;y de 0 a nuevohaz5 en datos
  ||MAC P,*XAR2+,*XAR7++
    NOP
    MOV T,#0xF
    LSRL ACC,T
;guarda el resultado en la loc correspondiente
MOVL XAR7,ACC ;el resultado se guarda en
MOVL XAR3,#_final4 ;la loc de final
MOV *XAR3,AR7

LRETR

```

Programa en el microcontrolador PIC

```

#include <18F4550.h>
#fuses HS,NOWDT,NOPROTECT,NOLVP
#use delay(clock=48000000)
#include <GLCD.c>
#use standard_io(a)

#BYTE TRISA =0xF92
#BYTE PORTA =0xF80
#BYTE TRISE =0xF96
#BYTE PORTE =0xF84
#BYTE TRISC =0xF94
#BYTE PORTC =0xF82

int poten;
int dir;
int xinf;

```

```

int xxinf;
int ysup;
int cadena;
int direccion;

void main(){

    TRISA=0xFF; //puerto A como entrada
    TRISE=0xFF; //puerto E como entrada
    CHAR SA[ ]="Samuel S.";
    CHAR cero[ ]="0";
    CHAR VAZ[ ]="Vazquez S.";
    CHAR BF[ ]="Tesis 2011";
    CHAR POT[ ]="12345678";
    CHAR A[ ]="Dir.";
    CHAR B[ ]="Pot.";
    dir=8;
    poten=0;
    direccion=0;
    glcd_init(ON);
    glcd_text57(10, 30, VAZ, 2, 1);
    glcd_text57(10, 5, SA, 2, 1);
    glcd_text57(30, 50, BF, 1, 1);
    delay_ms(200);
    glcd_init(ON);
    ///Dibuja los ejes y marcas
    glcd_line(0,54,127,54,1);
    glcd_line(7,54,7,56,1);
    glcd_line(14,54,14,56,1);
    glcd_line(21,54,21,56,1);
    glcd_line(28,54,28,56,1);
    glcd_line(35,54,35,56,1);
    glcd_line(42,54,42,56,1);
    glcd_line(49,54,49,56,1);
    glcd_line(56,54,56,56,1);
    glcd_text57(61, 56, cero, 1, 1);
    glcd_line(70,54,70,56,1);
    glcd_line(77,54,77,56,1);
    glcd_line(84,54,84,56,1);
    glcd_line(91,54,91,56,1);
    glcd_line(98,54,98,56,1);
    glcd_line(105,54,105,56,1);
    glcd_line(112,54,112,56,1);
    glcd_line(119,54,119,56,1);
    glcd_text57(0, 0, A, 1, 1);
    glcd_text57(80, 0, B, 1, 1);

    while(true){
        cadena=input_a();
        if(input(PIN_E2)){ ///Caso que se manda el dato de direccion
            dir=cadena;
            direccion = (dir-8)*10;
        }
    }
}

```

```
        sprintf(POT, "%i", direccion);
        glcd_rect(30,0,45,7,1,0);
        glcd_text57(30, 0, POT, 1, 1);
    }else{ ///Caso que manda el dato de potencia
        sprintf(POT, "%i", cadena);
        poten=cadena;
        glcd_rect(110,0,125,7,1,0);
        glcd_text57(110, 0, POT, 1, 1);
        xinf = (7*dir)+4;
        xxinf=xinf+6;
        ysup=53-poten;
        glcd_rect(xinf,53,xxinf,8,1,0);
        glcd_rect(xinf,53,xxinf,ysup,1,1);
    }
    delay_ms(40);
}
}
```

Fotografías

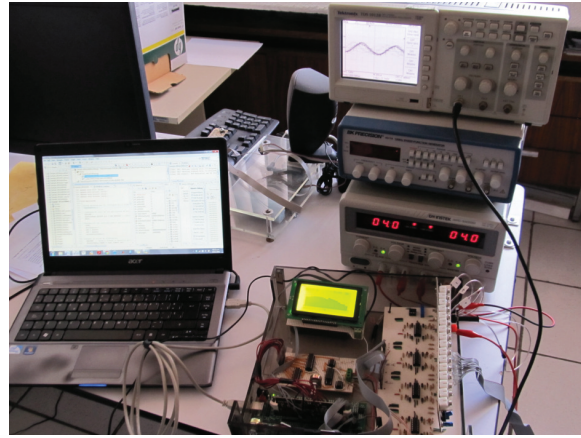
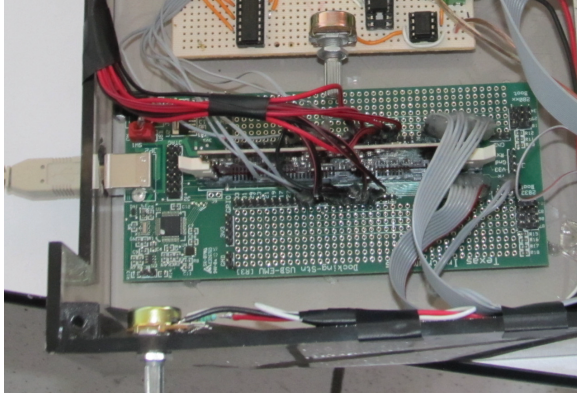


Figura 7.6 DSP TMS320F28335 montado en el docking station y sistema completo en funcionamiento.

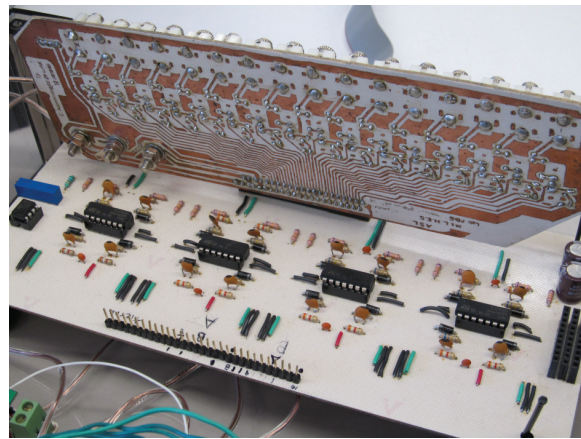
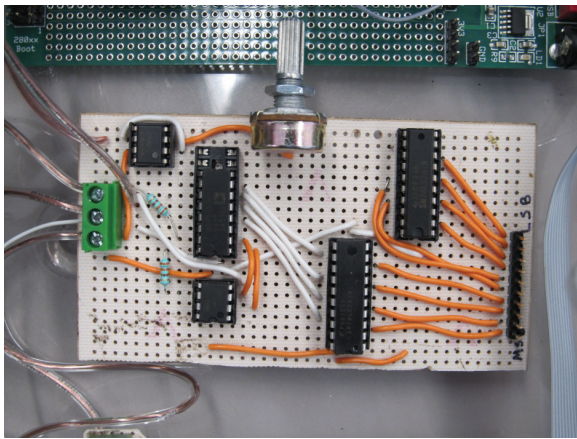


Figura 7.7 Circuito de conversión digital a analógica (izquierda) y circuito acondicionador de las señales de los micrófonos.

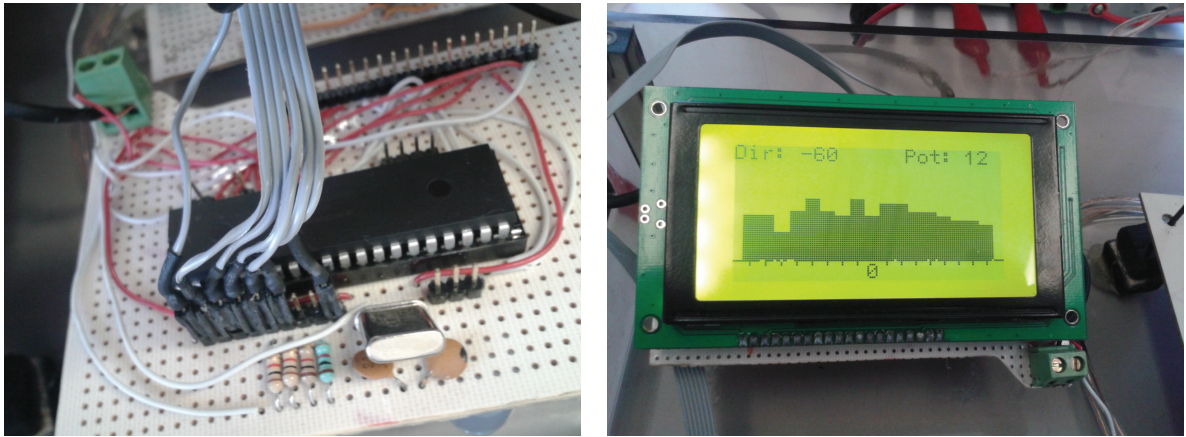


Figura 7.8 Microcontrolador PIC18F4550 y la pantalla gráfica mostrando un ejemplo del patrón de radiación, la dirección apuntada y la potencia en esa dirección.

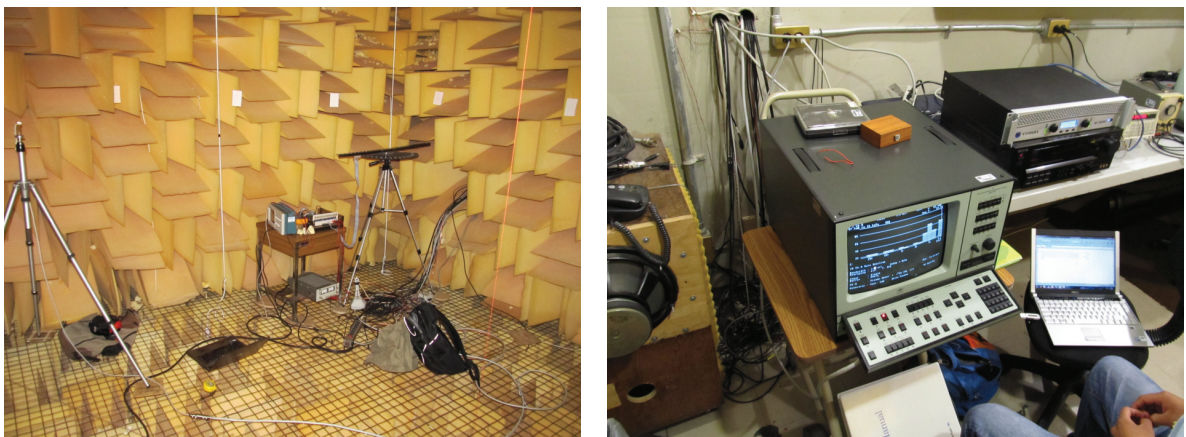


Figura 7.9 Sistema montado dentro de la cámara anecoica del CCADET y analizador de espectro marca B & K.

Bibliografía

- [1] B. Allen & M. Ghavami. *Adaptive Array Systems: Fundamentals and applications*. John Wiley and Sons. England, 2005.
- [2] G. Ballou. *Electroacoustic Devices*. Focal Press. UK, 2009.
- [3] J. Benesty, J. Chen, Y. Huang. *Microphone Array Signal Processing*. Springer. Germany, 2008.
- [4] F. Borman. *Texas Instruments C2000 teaching materials*. Germany, 2004.
- [5] B. Boulet. *Fundamentals of signals and systems*. Thomson. USA, 2006.
- [6] R. Boullosa & A. Pérez. *Some acoustical properties of the anechoic chamber at the Centro de Instrumentos, Universidad Nacional Autónoma de México*. Applied Acoustics, Volumen 56, Número 3. 1999.
- [7] M. Bruneau. *Fundamentals of acoustics*. ISTE. England, 2006.
- [8] T. Cox. *Acoustic diffusers: The good, the bad and the ugly*. Proceedings of the Institute of Acoustics.
- [9] T. Cox. & P D'Antonio *Acoustic absorbers and diffusers: Theory, design and applications* . Spon Press. England, 2004.
- [10] W. Chen. *Passive, active and digital filters*. CRC. USA, 2006.
- [11] S. Doclo. *Multi-microphone noise reduction and dereverberation techniques for speech applications*. Tesis de doctorado. Belgium, 2003.
- [12] J. Foutz, A. Spanias & M. Banavar *Narrowband Direction of Arrival Estimation for Antenna Arrays*. Morgan & Claypool publishers. USA, 2008.
- [13] R. Goldberg *Methods of real analysis*. John Wiley & Sons. USA, 1976.
- [14] P. Haughton. *Acoustics for audiologists*. Academic press. USA, 2002.
- [15] E. Kamen. *Introducción a señales y sistemas*. CECSA. México, 1999.
- [16] H. Kuttruff. *Acoustics an introduction*. Taylor & Francis. USA, 2007.

- [17] H. Kuttruff. *Room acoustics*. Spon Press. USA, 2009.
- [18] E. Larsen. *Audio Bandwidth Extension: Application of psychoacoustics, signal processing and loudspeaker design*. John Wiley & Sons. England, 2004.
- [19] W. Liu & S. Weiss. *Wideband beamforming: Concepts and techniques*. Wiley. England, 2010.
- [20] M. Long & M. Banavar *Architectural Acoustics*. Elsevier Academic Press. England, 2006.
- [21] R. Mucci. *Comparison of efficient beamforming algorithms*. IEEE transaction on acoustics, speech and signal processing. Volumen asp-32 Número 3. 1984
- [22] M. Najim. *Digital filters design for signal and image processing*. ISTE. England, 2006.
- [23] T. Oetiker. *The not so short introduccion to LATEX*. 2002.
- [24] A. Oppenheim, A. Willsky & S. Nawab *Señales y sistemas*. Pearson. México, 1998.
- [25] A. Silva *Detector de ángulo de arribo de una señal en un arreglo de micrófonos*. Tesis de licenciatura. Facultad de Ingeniería, UNAM. México, 2010.
- [26] Texas Instruments. *SPRA590. Writing a C Callable Assembly Function Using the TMS320C2000*. USA, 1999.
- [27] Texas Instruments. *SPRS439H. TMS320F28335, TMS320F28334, TMS320F28332, TMS320F28235, TMS320F28234, TMS320F28232. Digital Signal Controllers (DSCs)*. USA, 2010.
- [28] Texas Instruments. *SPRU430E. TMS320C28x CPU and Instruction Set Reference Guide*. USA, 2009.
- [29] Texas Instruments. *SPRU513C. TMS320C28x Assembly Language Tools v5.0.0*. USA, 2007.
- [30] Texas Instruments. *SPRU812A. TMS320x2833x Analog-to-Digital Converter (ADC) Module*. USA, 2007.
- [31] Texas Instruments. *SPRUEO24. TMS320C28x Floating Point Unit and Instruction Set*. USA, 2008.
- [32] Texas Instruments. *SPRUFB0D. TMS320x2833x, 2823x System Control and Interrupts*. USA, 2010.
- [33] Texas Instruments. *SPRUF5E. TMS320C2000 Experimenter Kit Overview*. USA, 2010.

- [34] S. Vaseghi. *Advanced digital signal processing and noise reduction*. Wiley. England, 2006.
- [35] H. Visser. *Phaced Array Antenna Basics*. John Wiley & Sons. England, 2005.
- [36] R. Waterhouse. *Interference patterns in reverberant sounds fields*. The journal of the acoustical society of America. Volumen 27 Número 2. 1954.
- [37] R. Yarlagadda. *Analog and digital signals and systems*. Springer. USA, 2010.

