

UNIVERSIDAD PRIVADA ANTONOR ORREGO
FACULTAD DE INGENIERÍA
PROGRAMA DE ESTUDIO DE INGENIERÍA ELECTRÓNICA



TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO

Análisis de la influencia de los parámetros de digitalización en la calidad del
audio producido digitalmente

Línea de investigación:
Sistemas Cognitivos

Autor/Autora/Autores:

León Cruz, Kelly Lidia
Romero Meléndez, Sandro Gastón

Jurado evaluador:

Presidente : De La Cruz Rodríguez, Oscar Miguel
Secretario : León Cerna, Alejandro Martín
Vocal : Alvarado Rodríguez, Luis Enrique

Asesor:

Azabache Fernández, Filiberto Melchor
Código Orcid: <https://orcid.org/0000-0003-2833-0493>

Trujillo–Perú
2023

Fecha de sustentación: 2023/07/21

Análisis de la influencia de los parámetros de digitalización en la calidad del audio producido digitalmente

INFORME DE ORIGINALIDAD



FUENTES PRIMARIAS

1	moam.info Fuente de Internet	4%
2	ocw.upc.edu Fuente de Internet	2%
3	repositorio.upao.edu.pe Fuente de Internet	2%
4	m-pulso.m-audio.com Fuente de Internet	1%
5	www.scribd.com Fuente de Internet	1%
6	la.mathworks.com Fuente de Internet	1%
7	patents.google.com Fuente de Internet	1%
8	www.megatron.ch Fuente de Internet	1%
9	documentop.com Fuente de Internet	1%



1%

Excluir citas Activo
Excluir bibliografía Activo

Excluir coincidencias < 1%

DECLARACION DE ORIGINALIDAD

Yo, Azabache Fernández Filiberto Melchor, docente del Programa de Estudio de pregrado del Programa de Estudio de Ingeniería Electrónica de la Universidad Privada Antenor Orrego, asesor de la tesis titulada “Análisis de la influencia de los parámetros de digitalización en la calidad del audio producido digitalmente”, de los autores Sandro Gastón Romero Meléndez y Kelly Lidia León Cruz, dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud del 12%. Así lo consigna el reporte de similitud emitido por el software Turnitin el día 25 de julio del 2023.
- He revisado con detalle dicho reporte de la tesis/ Trabajo de Suficiencia Profesional/proyecto de Investigación “Análisis de la influencia de los parámetros de digitalización en la calidad del audio producido digitalmente” y no se advierte indicios de plagio.
- Las citas a otros autores y sus respectivas referencias cumplen con las normas establecidas por la Universidad.

Trujillo, 07 de julio de 2023



APELLIDOS Y NOMBRES DEL ASESOR:
Azabache Fernández Filiberto Melchor
DNI: 17858660
ORCID: 0000-0003-2833-0493



APELLIDOS Y NOMBRES DEL AUTOR (ES):
León Cruz Kelly Lidia
DNI: 18100431

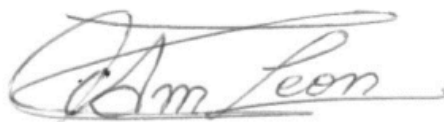


APELLIDOS Y NOMBRES DEL AUTOR (ES):
Sandro Gastón Romero Meléndez
DNI: 18073232

Jurado de sustentación Oral



Oscar Miguel De La Cruz Rodríguez
N° CIP 85598
Presidente



Alejandro Martín León Cerna
N° CIP 177967
Secretario



Alvarado Rodríguez, Luis Enrique
N° CIP 149200
Vocal

Entregado el:

Aprobado por:



Kelly Lidia León Cruz
DNI 18100431



Sandro Gastón Romero Meléndez
DNI 18073232



Filiberto M. Azabache Fernández
Asesor de Tesis

UNIVERSIDAD PRIVADA ANTONOR ORREGO
FACULTAD DE INGENIERÍA
PROGRAMA DE ESTUDIO DE INGENIERÍA ELECTRÓNICA



TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO

Análisis de la influencia de los parámetros de digitalización en la calidad del
audio producido digitalmente

Línea de investigación:
Sistemas Cognitivos

Autor/Autora/Autores:

León Cruz, Kelly Lidia
DNI: 18100431

Romero Meléndez, Sandro Gastón
DNI: 18073232

Jurado evaluador:

Presidente : De La Cruz Rodríguez, Oscar Miguel
Secretario : León Cerna, Alejandro Martín
Vocal : Alvarado Rodríguez, Luis Enrique

Asesor:

Azabache Fernández, Filiberto Melchor
Código Orcid: <https://orcid.org/0000-0003-2833-0493>

Fecha de sustentación: 21/07/2023

Trujillo-Perú
2023

Dedicatorias

Mi tesis la dedico con todo mi amor y cariño a mi padre Telmo León Villanueva quien actualmente se encuentra gozando de la presencia del señor. Él es quien me apoyo para ser lo que soy ahora y enseñó a seguir superándome, así como darme esa fuerza para seguir adelante.

A mi amada hija y quien es la inspiración para seguir superándome cada día y así seguir luchando para que la vida nos siga deparando algo mejor.

Kelly León

Para mis hijos Lucero y Fabio quienes con su ternura, inocencia y amor me impulsaban a perseverar hasta el logro de título así mismo, para demostrar que ellos también pueden llegar hasta donde ellos lo decidan, ellos fueron mi mayor inspiración. A mis queridos padres que invirtieron su tiempo y dedicación a mi carrera profesional y confiaron ciegamente en que sería un excelente profesional y que actualmente viven orgullosos de mi persona, a mis hermanos quienes que tuvieron que sacrificarse en algún momento por mí, dejando su tiempo y prioridades valiosas prefiriendo que culmine mi carrera. Finalmente, a mi tío Luis Franco porque sin él, no hubiera conocido la carrera hermosa de la electrónica quien, a pesar de su incapacidad visual, me pudo enseñar todo el tema práctico como si fuera una universidad andante.

Sandro Romero

Agradecimientos

A Dios por abrirme el camino para seguir adelante en la vida y a mi hija por estar siempre presente.

Quiero expresar mi más grande y sincero agradecimiento al Ms. Filiberto Azabache Fernández, principal colaborador en el desarrollo de la tesis quien me guío, con sus conocimientos y enseñanzas en su desarrollo.

Kelly León.

A mis padres, hijos y a mi asesor por el gran apoyo que me dio para no bajar la guardia y continuar en cerrar uno de mis ciclos profesionales que culmina con la titulación. También mi agradecimiento a mis segundos hijos Sandra y Rodrigo a quienes le debo mucho de mi tiempo y dedicación, son unos amores.

Sandro Romero

Resumen

La presente investigación documenta la evaluación de la influencia de los parámetros de digitalización en la calidad del audio producido digitalmente. Se produjo digitalmente el vals criollo Hilda a través de la captura del sonido de 14 instrumentos musicales digitalizados con una frecuencia de muestreo de 192 KHz y una profundidad de 24 bits y almacenados en una consola digital, con el archivo de la mezcla maestra se generaron otros archivos modificando los parámetros de digitalización mediante algoritmos en Matlab, se obtuvo información relevante de los archivos para comparar su calidad con la de la señal original. El estudio determinó que el factor influyente en la calidad del audio digital es la profundidad de bits y que la frecuencia de muestreo influye en el tamaño del archivo. Se concluye que la mejor respuesta es muestrear el audio a 48 KHz con una profundidad de 24 bits.

Palabras clave:

Digitalización del audio

Calidad del audio

Abstract

This research documents the evaluation of the influence of digitization parameters on the quality of digitally produced audio. The waltz Hilda was digitally produced by capturing the sound of 14 musical instruments digitized with a sampling frequency of 192 KHz and a depth of 24 bits and stored in a digital console, with the master mix file other files were generated by modifying the digitization parameters using algorithms in Matlab, relevant information was obtained from the files to compare their quality with that of the original signal. The study determined that the influential factor in the quality of digital audio is the bit depth and that the sampling frequency influences the file size. It is concluded that the best response is to sample the audio at 48 KHz with a bit depth of 24 bits.

Keywords:

Audio digitization

Audio quality

Presentación

Señores miembros del Jurado:

De conformidad con lo estipulado en el Reglamento de Grados y Títulos de la Universidad Privada Antenor Orrego, ponemos a su disposición el informe de tesis titulado “Análisis de la influencia de los parámetros de digitalización en la calidad del audio producido digitalmente” para que sea revisado y evaluado y de ser aprobado pueda ser defendido oralmente para optar el título profesional de Ingeniero Electrónico.

De antemano, nos excusamos de los errores involuntarios en que se hubiera incurrido en el desarrollo y redacción del misma, esperando del honorable jurado un justo dictamen.

Kelly Lidia León Cruz

Sandro Gastón Romero Meléndez

Tabla de contenidos

Jurado de sustentación Oral.....	iii
DECLARACION DE AUTENTICIDAD.....	¡Error! Marcador no definido.
Dedicatorias.....	vi
Agradecimientos.....	vii
Resumen.....	viii
Abstract.....	ix
Presentación.....	x
Tabla de contenidos.....	xi
Índice de tablas.....	xiii
Índice de figuras.....	xiv
I. INTRODUCCIÓN.....	1
1.1. Problema de Investigación.....	1
1.2. Objetivos.....	2
1.2.1. Objetivo general.....	2
1.2.2. Objetivos específicos.....	2
1.3. Justificación del estudio.....	2
II. MARCO DE REFERENCIA.....	4
2.1. Antecedentes del estudio.....	4
2.2. Marco teórico.....	5
2.2.1. El Sonido.....	5
2.2.1.1. Características del sonido.....	8
2.2.1.2. Espectro del sonido.....	10
2.2.2. Psico acústica.....	12
2.2.3. Digitalización del sonido.....	15
2.2.3.1. Muestreo de una señal de audio.....	16
2.2.3.2. Cuantificación.....	18
2.2.3.3. Codificación.....	23
2.2.3.4. Conversión de las palabras código a muestras analógicas.....	27
2.2.3.5. Ventajas e inconvenientes de la codificación digital de las señales.....	28
2.2.3.6. Tasa de bits (Bit rate: BR) del audio digital.....	29
2.2.3.7. Tamaño del archivo de audio.....	30
2.2.4. Producción musical.....	31
2.3. Marco Conceptual.....	32
2.4. Sistema de Hipótesis.....	33

2.5.	Variables e indicadores	33
III.	METODOLOGÍA EMPLEADA	34
3.1.	Tipo y nivel de investigación.....	34
3.2.	Población y muestra de estudio	34
3.3.	Diseño de investigación.....	34
3.4.	Técnicas e instrumentos de investigación.....	35
3.5.	Procesamiento y análisis de datos	35
IV.	PRESENTACIÓN DE RESULTADOS.....	58
4.1.	Análisis e interpretación de resultados	58
4.1.1.	Características sonoras de instrumentos musicales.....	58
4.1.2.	Producción digitalmente de una melodía multipista generada en estéreo.....	59
4.1.3.	Análisis de la calidad del audio producido al variar los parámetros de digitalización.....	59
4.2.	Docimasia de hipótesis.....	68
V.	DISCUSIÓN DE RESULTADOS.....	68
	CONCLUSIONES.....	70
	RECOMENDACIONES.....	71
	REFERENCIAS BIBLIOGRÁFICAS	72
	ANEXOS	73
	ANEXO 1: INSTRUMENTOS DE INVESTIGACIÓN	74
	Guía de análisis documental de instrumentos musicales	77
	ANEXO 2: Rangos de frecuencias de los instrumentos musicales.....	78
	ANEXO 3: Frecuencias empleadas por los instrumentos de acuerdo a su interpretación.....	79
	ANEXO 4: Algoritmos en Matlab para obtener información de los archivos de audio	80

Índice de tablas

Tabla 1: Niveles de presión sonora generados por diferentes sonidos.....	6
Tabla 2: Frecuencias de muestreo empleadas de acuerdo a sus usos.	18
Tabla 3: Niveles y códigos binarios de 3 bits en complemento a 2.....	25
Tabla 4: Variables e indicadores.....	33
Tabla 5: Técnicas e instrumentos de recolección de datos.....	35
Tabla 6: Instrumentos musicales empleados en la producción musical del vals Hilda	36
Tabla 7: Requerimiento de micrófonos de los instrumentos musicales.....	38
Tabla 8: Niveles de presión sonora y frecuencias de interpretación de los instrumentos musicales utilizados en la producción musical digital.....	40
Tabla 9: Información técnica de micrófonos usados para captar sonidos de los instrumentos de viento y percusión.	41
Tabla 10: Información obtenida de la señal de audio de la mezcla maestra Hilda_192_24.wav de	50
Tabla 11: Información obtenida de la señal de audio Hilda_96_24.wav	54
Tabla 12: Información obtenida de la señal de audio Hilda_48_24.wav	54
Tabla 13: Información obtenida de la señal de audio Hilda_44_24.wav	55
Tabla 14: Información obtenida de la señal de audio Hilda_192_16.wav	55
Tabla 15: Información obtenida de la señal de audio Hilda_96_16.wav	56
Tabla 16: Información obtenida de la señal de audio Hilda_48_16.wav	56
Tabla 17: Información obtenida de la señal de audio Hilda_44_16.wav	57
Tabla 18: Valores de las características de los archivos de audio al variar los parámetros de digitalización	59
Tabla 19: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 96 KHz con profundidad de 24 bits.	65
Tabla 20: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 48 KHz con profundidad de 24 bits.	65
Tabla 21: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 44,1 KHz con profundidad de 24 bits.....	65
Tabla 22: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 192 KHz con profundidad de 16 bits.....	66
Tabla 23: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 96 KHz con profundidad de 16 bits.	66
Tabla 24: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 48 KHz con profundidad de 16 bits.	67
Tabla 25: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 44,1 KHz con profundidad de 16 bits.....	67

Índice de figuras

Figura 1: Representación de un ciclo de una onda senoidal.	8
Figura 2: Señal multitono en el dominio del tiempo.	9
Figura 3: Representación de la fase de una señal.	10
Figura 4: Representación del espectro de audio en bandas de frecuencias.	11
Figura 5: Visualización espectral de una señal de audio.	12
Figura 6: Curvas de umbrales de audición del oído en función de las frecuencias.	13
Figura 7: Proceso de enmascaramiento.	15
Figura 8: Proceso completo de la digitalización del sonido	16
Figura 9: Muestreo de una señal de audio.	17
Figura 10: Efecto de la frecuencia de muestreo del ADC en la calidad de la señal obtenida.	17
Figura 11: Efecto del tamaño de palabra del ADC en la calidad de la señal obtenida.	20
Figura 12: Cuantificación y Ruido de cuantificación	20
Figura 13: Diferencia entre una señal que recorre todo su rango dinámico, $s(t)$, en rojo, y la señal cuantificada $s_q(t)$, en azul, es decir, el ruido o error de cuantificación, en verde.	22
Figura 14: Igualación del número de niveles de cuantificación en señales de distinto DR ajustando sus valores máximos y mínimos.	23
Figura 15: Codificación PCM en complemento a dos.	24
Figura 16: Ejemplo de codificación de una señal.	26
Figura 17: Transmisión de los datos PCM en forma de trama.	26
Figura 18: Diagrama de bloques de un sistema de conversión de digital a analógico	27
Figura 19: Cables con conectores para consola	39
Figura 20: Conexión de micrófonos e instrumentos a la consola digital Soundcraft Ui24R	39
Figura 21: Niveles de ganancia y salida a mezcla maestra de instrumentos musicales en la consola Soundcraft Ui24R.	41
Figura 22: Visualización de los 14 archivos de audio digital generados en formato *.wav para la producción musical del vals Hilda con una profundidad de 24 bits y 192 KHz de frecuencia de muestreo.	43
Figura 23: Algoritmos y capturas de las gráficas de la densidad espectral de potencia de cada uno de los instrumentos musicales empleados para la producción musical del vals Hilda.	43
Figura 24: Onda de la mezcla maestra del vals criollo Hilda y su espectrograma.	47
Figura 25: Número de muestras que presentan los archivos de audio.	60
Figura 26: Bitrate de los archivos de audio	61

Figura 27: Tamaño de los archivos de audio	62
Figura 28: Rango dinámico de los archivos de audio	63
Figura 29: Relación señal a ruido de los archivos de audio	64

I. INTRODUCCIÓN

1.1. Problema de Investigación

El audio digital es la representación digital de señales de audio, es decir, la conversión de ondas sonoras analógicas en una serie de valores numéricos que pueden ser almacenados, transmitidos y procesados por dispositivos digitales.

El audio digital es una forma de codificar la información de audio en una secuencia de números binarios (bits) que pueden ser almacenados en un formato digital, como MP3, WAV, FLAC, entre otros. Estos bits representan la amplitud y frecuencia de la señal de audio original y permiten una reproducción más precisa y confiable de la señal de audio en dispositivos digitales, como computadoras, teléfonos móviles y reproductores de audio portátiles.

La generación de audio digital se refiere a la creación de sonido utilizando tecnología digital. Esto se puede lograr utilizando software de producción musical, sintetizadores digitales, samplers, o incluso dispositivos de hardware dedicados.

En la producción musical, la generación de audio digital implica la grabación y edición de pistas de audio utilizando software de grabación y edición de audio. Esto permite a los productores y músicos grabar y editar audio con precisión y controlar una amplia variedad de parámetros, como la ecualización, la compresión, la reverberación, entre otros.

Por otro lado, los sintetizadores digitales y los samplers permiten a los músicos crear sonidos desde cero, manipulando ondas sonoras y muestras de audio para crear sonidos complejos y únicos. Estos dispositivos también permiten a los músicos y productores modificar los parámetros del sonido en tiempo real para crear efectos dinámicos. La generación de audio digital es un proceso clave en la producción musical moderna y es esencial para crear música y sonidos en una amplia variedad de géneros y estilos.

La producción de audio digital presenta una serie de hechos problemáticos como:

- El gran tamaño de los archivos sin compresión,
- El ruido generado por la cuantificación
- La pérdida de la calidad debido a la compresión
- Generación de formatos de archivo incompatibles.

Por lo expuesto, el problema se define en investigar la forma en que se relaciona la frecuencia de muestreo y los niveles de cuantificación para la generación de audio digital de alta calidad, formulándose el siguiente problema:

¿Cómo influyen los parámetros de digitalización en la calidad del audio producido digitalmente?

1.2. Objetivos

1.2.1. Objetivo general

Evaluar la influencia de los parámetros de digitalización en la calidad del audio producido digitalmente.

1.2.2. Objetivos específicos

- Determinar las características sonoras de instrumentos musicales.
- Producir digitalmente una melodía multipista y generarla en estéreo.
- Analizar la calidad del audio producido al variar los parámetros de digitalización.

1.3. Justificación del estudio

- La investigación se justifica por su conveniencia porque los resultados serán útiles para los profesionales que trabajan con audio digital.
- Se justifica desde el punto de vista práctico porque propone un proceso para determinar los valores adecuados de los parámetros de digitalización para producir una señal de audio digital de calidad.

- Se justifica por su valor teórico porque fortalecerán los conocimientos del procesamiento digital de señales de audio.
- Se justifica por su utilidad metodológica porque desarrolla un método de evaluación de la relación entre la frecuencia de muestreo de la señal de audio y los niveles de cuantificación necesarios para obtener audio digital de calidad.

II. MARCO DE REFERENCIA

2.1. Antecedentes del estudio

La investigación desarrollada por Tabares (2019) titulada “Desarrollo de un algoritmo de medición y normalización con base en la norma ITU-R BS.1770-4 para el control de los niveles de audio en la televisión colombiana”, aporta el análisis detallado de la norma ITU-R BS.1770-4, así como el desarrollo del sistema y el algoritmo para controlar los niveles de audio en televisión.

Agudo (2022) (Agudo, 2022) en su investigación “Estudio del reconocimiento de señales de audio mediante redes neuronales” aporta el desarrollo de un estudio de reconocimiento de palabras en forma de señales de audio, donde extrae las características de este tipo de señales para confeccionar una base de datos que se usa para predecir futuras muestras con alguna de las palabras incluidas en la base mediante distintos algoritmos de predicción y comparar todos estos para encontrar cuál es el mejor, enfatizando la red neuronal.

Pardo (2018) en la investigación “Análisis tecnológico de un estudio de grabación”, aporta un análisis de los diferentes tipos de micrófonos para su uso en la grabación de diversos instrumentos musicales, asimismo, aporta la evaluación de diferentes equipos e interfaces para la grabación digital.

Castán (2017) a través de la investigación titulada “Desarrollo de una aplicación en Matlab para medidas de sistemas de sonido con múltiples entradas y múltiples salidas mediante señales sweep”, aporta el desarrollo de una aplicación en MATLAB que permite la medida de sistemas acústicos. El software desarrollado dispone de interfaz gráfica que permite utilizar drivers multicanal de cualquier tarjeta de sonido que se conecte al ordenador. El desarrollo del software presenta pruebas acústicas en condiciones reales.

Chiuro, Frontini, & Otero (2021) en la investigación “Señales y música, una relación inseparable. Una aplicación de la serie de Fourier con Matlab”, aportan el desarrollo de algoritmos en matlab basados en las series de Fourier para el análisis de notas musicales en el dominio del tiempo y en el dominio de la frecuencia.

Pérez (2018) en la investigación “Desarrollo de herramienta de análisis para la codificación y síntesis paramétrica de señales de voz y aplicación de efectos”, aporta el desarrollo de una herramienta en Matlab que analiza la calidad de la señal de voz en función de las características de la codificación paramétrica empleada, así como realizar efectos o cambios en la sintetización de la misma en función de la manipulación de dichos parámetros.

2.2. Marco teórico

2.2.1. El Sonido

El sonido es una vibración mecánica que se propaga a través de un medio, como el aire o el agua. Estas vibraciones sonoras son captadas por nuestros oídos y transmitidas al cerebro, quien los interpreta como sonidos. La frecuencia de las vibraciones determina la altura del sonido, mientras que la amplitud determina su intensidad. (Jaramillo, 2007)

Además de ser una forma de comunicación, el sonido también tiene muchas aplicaciones prácticas, como en la música, la ingeniería de sonido, la medicina y la tecnología de la comunicación. En la naturaleza, el sonido también puede ser utilizado para la ecolocalización por parte de algunos animales, como los murciélagos y los delfines.

Las vibraciones de frecuencias comprendidas entre 20 Hz y 20KHz, se denominan señales de audio porque son percibidas por el sistema auditivo humano. Las vibraciones cuyas frecuencias son menores de 20 Hz se

denominan infrasonidos y las que oscilan por encima de los 20 KHz ultrasonidos.

El sonido se mide por nivel de presión sonora (SPL) que este presenta, se expresa en decibelios resultado de la siguiente operación matemática:

$$d_{BSPL} = 20 \log \left[\frac{\text{Presión medida}}{\text{Presión de referencia}} \right] \dots (1)$$

Siendo la presión de referencia 20-μPa ó 0 dB_{SPL}, que es la mínima presión sonora que puede percibir el oído humano medio y se presenta en torno a los 2 KHz.

En la tabla 1, se muestran niveles de presión promedios generados por diferentes sonidos:

Tabla 1: Niveles de presión sonora generados por diferentes sonidos

Tipo de sonido	Nivel de presión sonora (dB_{SPL})
Umbral de audición	0
Susurro de las hojas	10
Murmullo de las hojas	20
Radio de volumen bajo	40
Conversación normal	57
Conversación de tono elevado	64
Una calle transitada	80
Transporte subterráneo	100
En una discoteca	115
Umbral de dolor	120
Motor a propulsión	140 ~ 160

El sonido tiene la capacidad de moverse a través de distintos medios físicos, ya sean sólidos, líquidos o gaseosos (como el aire). La velocidad de propagación del sonido en cada medio está determinada por la densidad del mismo, es decir, cuanto más denso sea el medio, mayor será la velocidad

de propagación. Sin embargo, en el vacío, donde no hay partículas que puedan vibrar, no hay propagación de sonido.

Si el medio donde se propaga el sonido tiene mayor compresibilidad ($1/K$), menor será la velocidad. También un factor importante en la velocidad de propagación es la densidad (ρ) del medio, a mayor densidad menor es la velocidad de propagación, considerando igual a todo lo demás. La velocidad del sonido se relaciona con esas magnitudes mediante la siguiente ecuación:

$$v \propto \sqrt{\frac{K}{\rho}} \dots (2)$$

En el aire, el sonido se propaga a una velocidad promedio de 343 m/s. Esta velocidad varía con la densidad del aire y es afectada por factores como la temperatura o la humedad relativa. En cualquier caso, para distancias de decenas de metros las variaciones son mínimas. El nivel de presión sonora disminuye conforme el sonido se propaga en el aire. Cuando el frente de onda es esférico, en la mayoría de los casos, el nivel de presión disminuye 6 dB cada vez que se duplica la distancia. Estas pérdidas se denominan *pérdidas por divergencia esférica*. Cuando el frente de onda es plano, no hay pérdidas por divergencia. Un ejemplo de este tipo de propagación se da en la propagación del sonido por el interior de una tubería. (Parrondo y Col., 2005)

En el agua, un valor típico de velocidad del sonido son 1500 m/s (el agua es más densa que el aire). En el agua, la densidad varía mucho en función de factores como la profundidad, la temperatura o la salinidad.

En materiales metálicos, el sonido se propaga a velocidades mayores, por ejemplo, en el acero el sonido se propaga a una velocidad en torno a 5000 m/s. En materiales sólidos se utiliza el sonido y las propiedades de reflexión para detectar fallas estructurales y grietas, sin necesidad de tener acceso a toda la estructura. Por ejemplo, en una viga, bastará con acceder a una de sus terminaciones para poder conocer su estado, empleando ultrasonidos y ecogramas. (Jaramillo, 2007 p.143)

2.2.1.1. Características del sonido

Según Mandal (2003, p.13) el sonido tiene 3 características de percepción: el tono, la intensidad y el timbre. Estas coinciden con 3 características físicas: la frecuencia, la amplitud y la composición armónica o forma de onda. La fase de un sonido es otra característica importante.

Frecuencia

La frecuencia se refiere al número de ciclos completos u oscilaciones que se producen en un fenómeno periódico en un segundo. En el contexto del sonido, la frecuencia se relaciona con la cantidad de veces que una onda sonora se repite en un segundo. Se mide en hercios (Hz), que representa el número de ciclos por segundo. La figura 2 representa un ciclo de una onda senoidal.

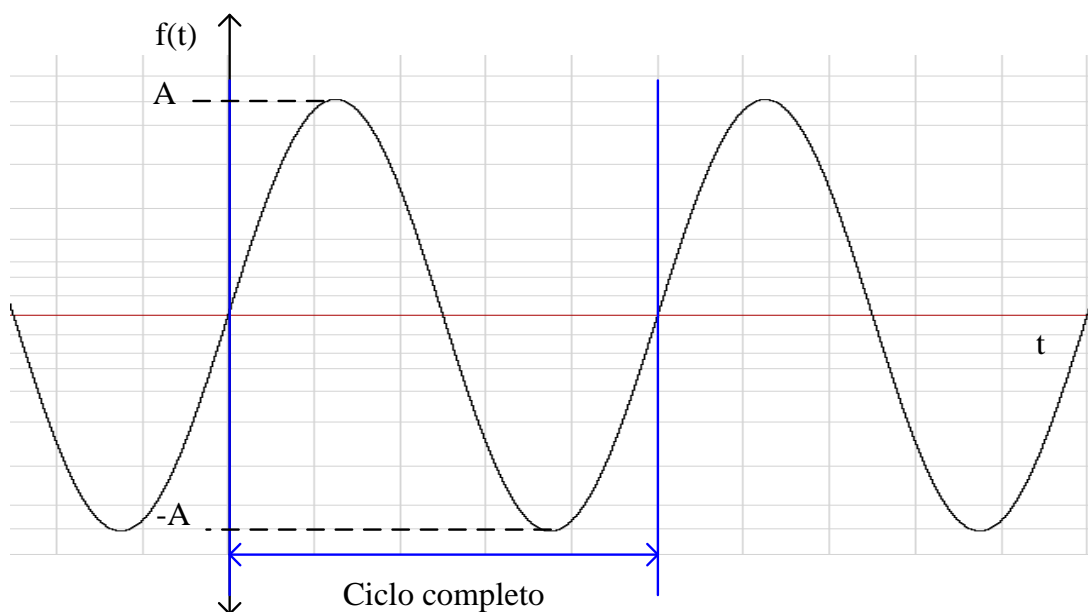


Figura 1: Representación de un ciclo de una onda senoidal.

Los sonidos que provienen de una sola frecuencia se llaman tonos puros, mientras que los sonidos que contienen varias frecuencias se llaman multitonos. Cualquier tipo de sonido, ya sea voz, música o ruido, está compuesto por múltiples frecuencias. Un sonido se puede desglosar en diversos tonos puros. Si un sonido cambia a lo largo del

tiempo, la amplitud y fase de cada tono puro o frecuencia que lo compone también variarán a medida como se muestra en la figura 3.

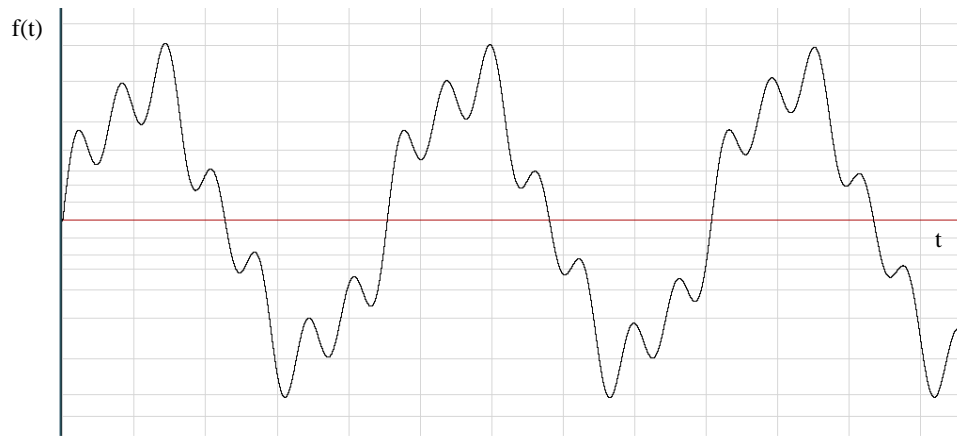


Figura 2: Señal multitono en el dominio del tiempo.

Amplitud

La amplitud de la oscilación, en una señal eléctrica como se observa en la figura 2, está dada por el módulo definido en voltios representado en la amplitud (A), mientras más alta sea la amplitud (A) de la señal, el sonido tendrá más intensidad.

Timbre

El timbre de un sonido está en función de las frecuencias que lo componen y generan su forma de onda. Las señales pueden tener el mismo tono (frecuencia), pero diferente timbre. El oído percibe de manera diferente los sonidos generados por ondas del mismo tono, pero de diferente timbre. Por ejemplo, el sonido generado por una onda triangular suena diferente al generado por una onda cuadrada, esto se debe por el contenido armónico que tiene cada señal.

Fase

La fase indica la posición de la onda que oscila en el momento de empezar a contar el tiempo. La fase se mide en radianes (rad) o en grados ($^{\circ}$), donde: $360^{\circ} = 2\pi$ rad. En la figura 4 se muestra que para $T=0$ rad la señal $f_1(t)$ es una onda seno y para $T=\pi/2$ la onda $f_2(t)$ es un coseno.

La fase relativa se refiere a la diferencia entre dos movimientos con la misma frecuencia. Si dos ondas tienen la misma frecuencia, pero una alcanza el punto cero mientras que la otra no lo hace o lo hace en dirección opuesta, tendrán una fase relativa distinta de cero. Si una onda alcanza el punto cero mientras que la otra lo hace en dirección opuesta, tendrán una fase relativa de 180° . En este caso, si las dos ondas tienen la misma amplitud, se cancelarán mutuamente y el resultado total será cero.

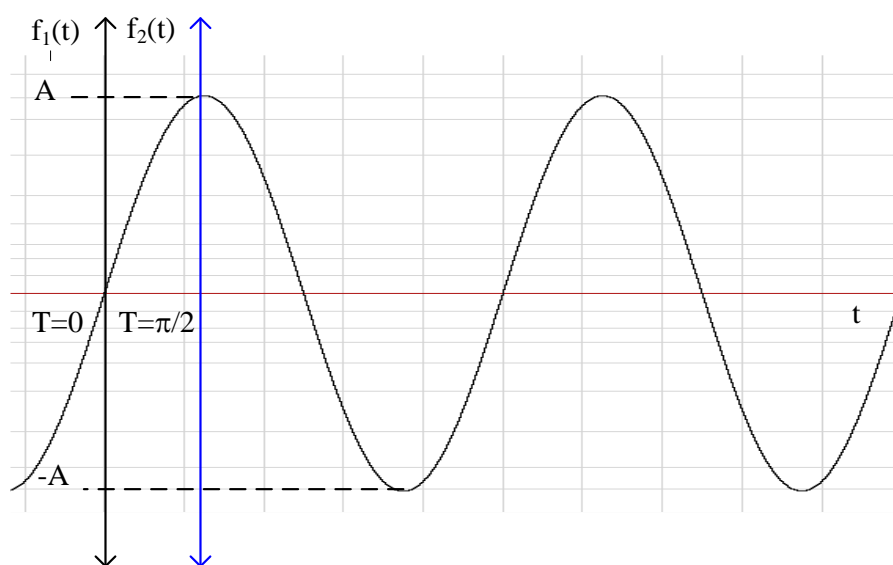


Figura 3: Representación de la fase de una señal.

2.2.1.2. Espectro del sonido

El espectro, es la representación de las frecuencias que componen un sonido. El espectro se obtiene calculando la energía que aporta cada frecuencia al sonido total. Normalmente la representación no se hace en términos de energía directamente, sino que se calcula el nivel ($10 \log$) respecto a la energía de referencia. Con esto se obtiene el "Nivel espectral" expresado en dB. Aunque el proceso de cálculo es bastante tedioso, la electrónica e informática actuales, con hardware o software que trabajan con TFT y FFT que simplifica enormemente el proceso de obtención.

Para emplear menos recursos y menos tiempo en el cálculo, no se calcula en nivel espectral para cada frecuencia (cerca de 20000) sino que éstas se agrupan en bandas, dando lugar a la representación en "bandas de

frecuencia" como se muestra en la figura 4, en donde, los números ubicados representan las frecuencias centrales expresadas en Hz.

Las representaciones en octavas suelen contar con 10 bandas y las de media octava con 20 bandas. También se utilizan las de tercio de octava (30 o 31 bandas). Las representaciones se hacen sobre ejes de frecuencia logarítmicos, esto hace que en la representación se vea la misma distancia entre las frecuencias 100 Hz y 200 Hz que entre 1 KHz y 2 KHz. Para los tres casos el espectro está representado de 20 Hz a 20 KHz.

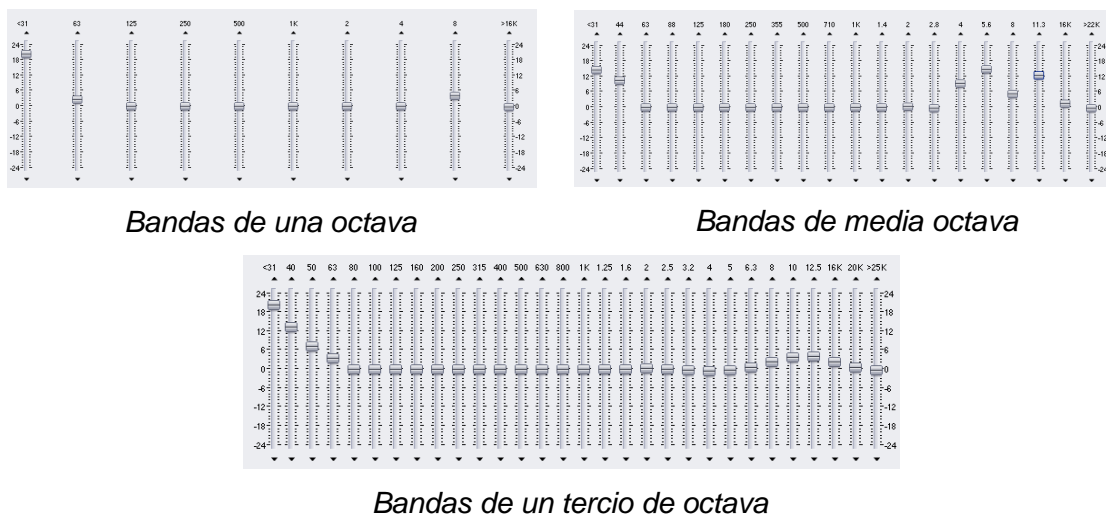


Figura 4: Representación del espectro de audio en bandas de frecuencias.

La relación entre las frecuencias centrales sigue el siguiente patrón: en bandas de octava: $f_2 = 2 \cdot f_1$. Para bandas de media octava: $f_2 = \sqrt{2} \cdot f_1$. En bandas de tercio de octava: $f_2 = \sqrt[3]{2} \cdot f_1$. En estas expresiones, f_1 representa la frecuencia central de una banda y f_2 representa la frecuencia central de la banda superior adyacente.

La representación espectral puede ser muy útil si se sabe interpretar; básicamente, proporciona información sobre la contribución de potencia de cada frecuencia al sonido total. En otras palabras, el espectro permite "visualizar" el sonido que llega al oído. Sin embargo, lo que el oído percibe e interpreta puede ser diferente. La figura 5 muestra el espectro de una señal de audio.

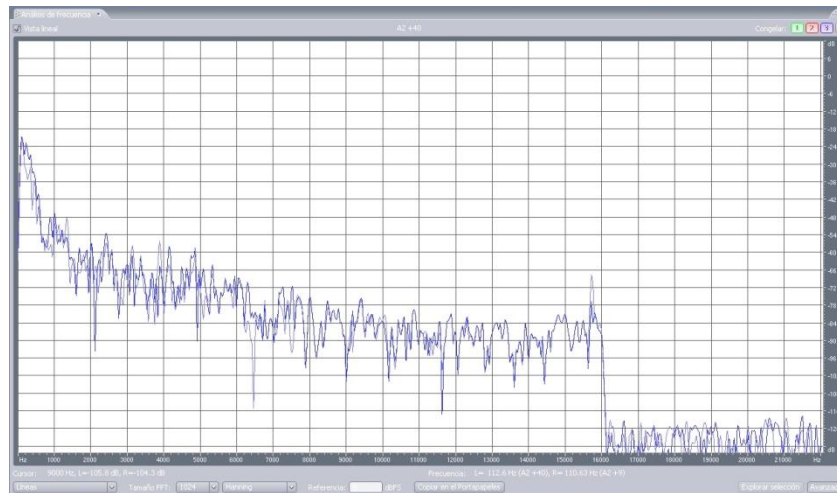


Figura 5: Espectro de una señal de audio.

2.2.2. Psico acústica

Audición y frecuencia

El umbral de audición, para el promedio de los humanos, se fija en $20 \mu\text{Pa}$, para frecuencias entre 2KHz y 4KHz. Por encima y por debajo de estas frecuencias, la presión requerida para excitar el oído es mayor. Esto significa que nuestro oído no responde igual a todas las frecuencias. Un tono puro, a la frecuencia de 125 Hz y con 15 dB de nivel, sería prácticamente inaudible, mientras que, si aumentamos la frecuencia, hasta 500 Hz, sin variar el nivel de presión, se obtendría un tono claramente audible. En la figura 6, las líneas marcan los niveles de presión necesarios a cada frecuencia, para que el oído detecte (subjetivamente) la misma sonoridad en todas. Esto quiere decir que si reproducimos un tono de 31.5 Hz a 100 dB (SPL), luego otro de 63 Hz a 90 dB y otro de 125 Hz a 80 dB, el oyente dirá que todos sonaban al mismo volumen.

En 2 KHz el umbral de audición se fija en 0 dB y a 4 KHz es incluso menor de 0 dB, ya que a 3600 Hz se encuentra la frecuencia de resonancia del oído humano.

Por debajo de 2 KHz y según se va bajando en frecuencia, el oído se vuelve menos sensible. Los umbrales de audición para frecuencias menores de 2 KHz son: 5 dB a 1 KHz, 7 dB a 500 Hz, 11 dB a 250 Hz, 21 dB a 125 Hz, 35 dB a 63 Hz, 55 dB a 31 Hz. Por encima de los 4 KHz, el oído es menos sensible, pero no tanto como en bajas frecuencias. Sin embargo, se producen

fluctuaciones a frecuencias cercanas, debido a las perturbaciones que produce la cabeza del oyente en el campo sonoro. Los umbrales de audición son: 15 dB a 8 KHz y 20 dB a 16 KHz.

Todos los receptores de sonido, tienen un comportamiento que varía con la frecuencia. En el caso del oído humano, sucede lo mismo, ya que se trata del receptor más complicado y más eficiente que existe.

El umbral de audición define la mínima presión requerida para excitar el oído. El límite del nivel de presión sonora se sitúa generalmente alrededor de 120 dB, coincidiendo con el umbral del dolor (molestias en el oído). La pérdida de audición de manera súbita, por daños mecánicos (en el oído medio) se produce a niveles muchos mayores. La exposición suficientemente prolongada a niveles superiores a 130 dB produce pérdida de audición permanente y otros daños graves.

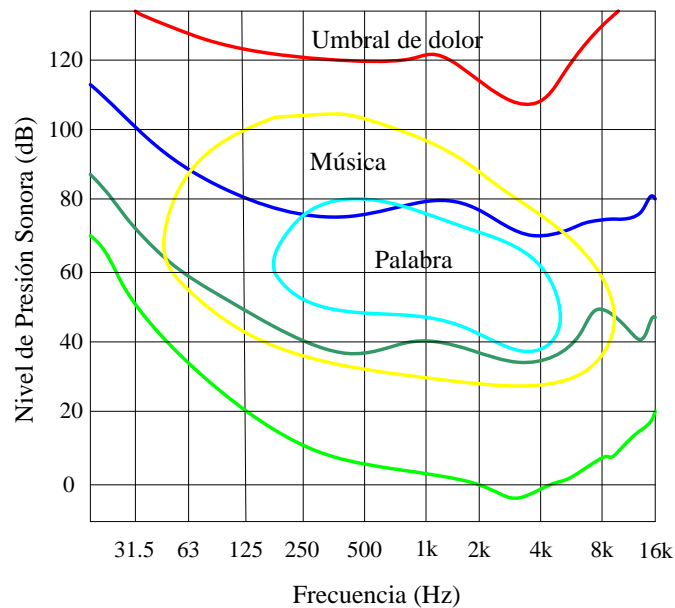


Figura 6: Curvas de umbrales de audición del oído en función de las frecuencias.

En acústica, las frecuencias siempre se tratan de manera logarítmica. El motivo principal es que el oído humano interpreta las frecuencias de manera casi logarítmica.

Agudeza auditiva

El oído humano tiene una agudeza auditiva bastante precisa para distinguir entre diferentes frecuencias, pero su habilidad para localizar la fuente de un sonido es limitada. La agudeza frecuencial del oído se sitúa en torno al 1%, lo que significa que puede diferenciar tonos cuyas frecuencias varían en tan solo un 1%. Por ejemplo, puede distinguir entre un tono de 100 Hz y otro de 101 Hz, pero no entre uno de 1000 Hz y otro de 1001 Hz. Sin embargo, puede percibir la diferencia entre uno de 1000 Hz y otro de 1010 Hz debido a la naturaleza logarítmica de la audición.

En términos de sonoridad, el oído humano se comporta como un conjunto de 24 filtros de 1/3 de octava. Esto se evidencia al realizar mediciones de sonoridad y comparar los niveles de sonoridad de ruidos con bandas estrechas, manteniendo constante la presión total. Se ha descubierto que, para anchos de banda menores a 1/3 de octava, todas las señales se perciben con igual sonoridad. Sin embargo, cuando el ancho de banda supera 1/3 de octava, la sonoridad percibida aumenta. Esto se debe a que comienzan a activarse los nervios en las áreas adyacentes de filtrado en el oído interno. Por esta razón, las representaciones en tercios de octava son ampliamente utilizadas y resultan muy útiles en el campo del sonido.

Enmascaramiento

Se presenta cuando existe dificultad o imposibilidad para escuchar un sonido deseado, como música o habla, debido a la presencia simultánea de otro sonido considerado ruido. En los procesos de enmascaramiento se observan los siguientes efectos:

- a) Una banda estrecha de ruido causa más enmascaramiento que un tono puro de igual frecuencia central e intensidad.
- b) Cuando el ruido tiene un nivel bajo, el enmascaramiento ocurre en una banda de frecuencia estrecha alrededor de la frecuencia central del ruido.
- c) El efecto de enmascaramiento no es simétrico alrededor de la frecuencia central del ruido enmascarante. Las frecuencias superiores son más afectadas por el enmascaramiento.

La figura 7 muestra los efectos b) y c). La línea curva inferior representa el umbral promedio de audición. El gráfico muestra las áreas que están sujetas

a los efectos del enmascaramiento, con un ruido de banda estrecha centrado en 1200 Hz y diferentes niveles de presión sonora del ruido. En el caso más extremo, con un ruido de 110 dB (curva más alta), se obtiene la mayor área enmascarada. Por ejemplo, en este caso, el oyente no sería capaz de detectar un tono de 8 KHz con un nivel de presión de 50 dB, ni un sonido de 4 KHz con un nivel de presión de 70 dB.

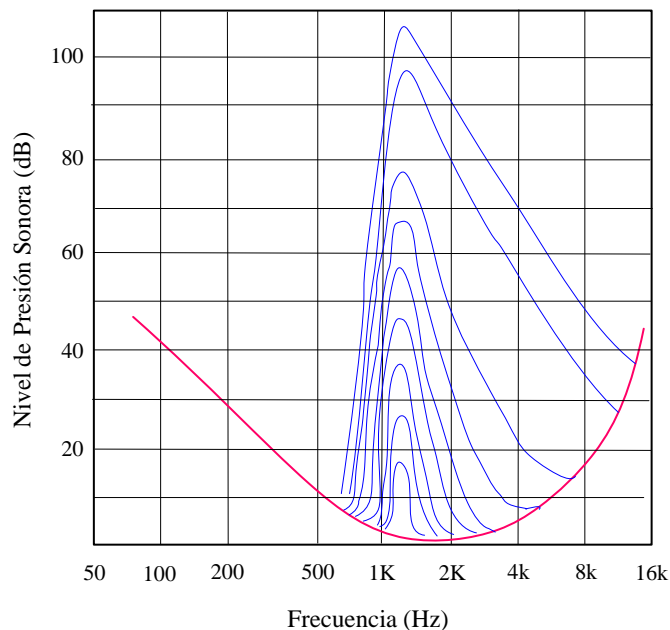


Figura 7: Proceso de enmascaramiento.

2.2.3. Digitalización del sonido

Consiste en convertir una señal de audio analógica en una secuencia digital binaria y para ello se llevan a cabo tres procesos: muestreo, cuantificación y codificación.

El sonido, para ser procesado digitalmente, ha de captarse por medios electrónicos. El primer elemento de este proceso es el micrófono, que convierte las variaciones de presión del medio en señales eléctricas, las cuales son después pre-amplificadas para que alcancen los niveles adecuados para atacar las siguientes etapas del proceso.

La digitalización de señales, consiste en convertir los valores de intensidad de la señal en valores numéricos que la representen. Para ello se utilizan circuitos convertidores de analógico a digital (ADC) que llevan a cabo una conversión o lectura cada cierto tiempo. A cada lectura se la llama *muestra* y

el número de muestras que se toman por segundo es la *frecuencia de muestreo*. Lógicamente, en algún momento esa misma señal o una versión mezclada, procesada o alterada de ella, ha de enviarse nuevamente al exterior en forma de sonido para lo cual se usa un convertidor de digital a analógico (DAC) conectado a un sistema de audio de salida y acoplado a un altavoz, tal como se muestra en la figura 8.

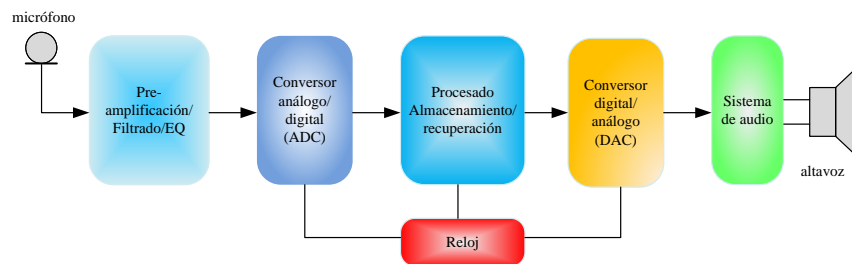


Figura 8: Proceso completo de la digitalización del sonido

2.2.3.1. Muestreo de una señal de audio

Una señal de audio tiene diferentes valores en el eje de tiempo y en el eje de amplitud, por lo que se toman muestras en intervalos concretos del tiempo (Pohlmann, 2005; p.26).

La frecuencia de muestreo es un factor crucial para garantizar la precisión con la que una señal se representa en su forma digital y la fidelidad de la reproducción posterior. Según establece el Teorema de Nyquist, la frecuencia de muestreo (f_s) debe ser al menos el doble de la máxima frecuencia presente en la señal de audio que se desea digitalizar.

En otras palabras, para capturar de manera adecuada todas las frecuencias contenidas en una señal de audio, es necesario muestrear a una velocidad que permita registrar al menos el doble de la frecuencia más alta presente en esa señal. Esto se debe a que el proceso de muestreo toma muestras periódicas de la señal en intervalos de tiempo específicos, y para evitar la pérdida de información, es necesario que haya suficientes muestras para representar con precisión las variaciones de la señal.

Cumplir con el Teorema de Nyquist asegura que la señal de audio se capture de manera adecuada y que la información necesaria para su reproducción fiel se conserve en su forma digital.

$$f_s \geq 2f_{\text{máx audio}} \dots(6)$$

En la figura 9, se ilustra el proceso de muestreo de una señal de audio.

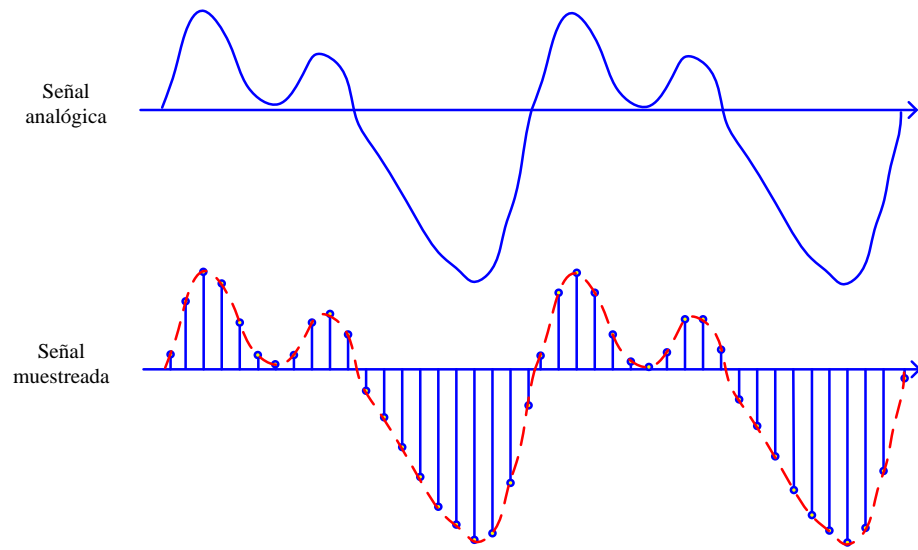


Figura 9: Muestreo de una señal de audio.

Cuando se realiza el muestreo de una señal, se está descartando información, y si no se cumple la ecuación (6), se produce un efecto de distorsión en la señal conocido como "aliasing".

En la figura 10 se muestra el efecto que ocurre al muestrear una señal a una frecuencia inferior al doble de su frecuencia original.

Para lograr una reproducción de sonido precisa y fiel a la señal original, dado que la frecuencia máxima del audio es de 20 KHz, la frecuencia de muestreo debe ser mayor a 40 KHz (40,000 muestras por segundo). A medida que se aumenta la frecuencia de muestreo, la señal digitalizada será de mejor calidad, pero esto también resultará en un mayor tamaño del archivo de audio digital almacenado.

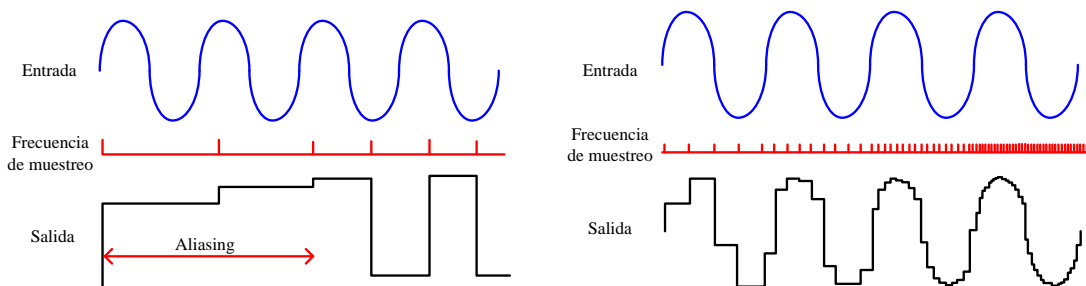


Figura 10: Efecto de la frecuencia de muestreo del ADC en la calidad de la señal obtenida.

En la tabla N° 2, se muestran los valores de muestreo mayormente empleados.

Tabla 2: Frecuencias de muestreo empleadas de acuerdo a sus usos.

Frecuencia de muestreo	Usos
8000 muestras/s	Teléfonos, adecuado para la voz humana pero no para la reproducción musical. En la práctica permite reproducir señales con componentes de hasta 3,5 kHz.
22050 muestras/s	Radio, en la práctica permite reproducir señales con componentes de hasta 10 kHz.
32000 muestras/s	Vídeo digital en formato miniDV .
44100 muestras/s	CD, En la práctica permite reproducir señales con componentes de hasta 20 kHz. También común en audio en formatos MPEG-1(VCD,SVCD, MP3).
47250 muestras/s	Formato PCM de Nippon Columbia (Denon) . En la práctica permite reproducir señales con componentes de hasta 22 kHz.
48000 muestras/s	Sonido digital utilizado en la televisión digital, DVD, formato de películas, audio profesional y sistemas DAT.
50000 muestras/s	Primeros sistemas de grabación de audio digital de finales de los 70de las empresas 3M y Soundstream .
96000 ó 192400 muestras/s	HD DVD, audio de alta definición para DVD y BD-ROM (Blu-ray Disc).
2 822 400 muestras/s	SACD, Direct Stream Digital, desarrollado por Sony y Philips.

2.2.3.2. Cuantificación

La cuantificación define la precisión o resolución con que se codifican las muestras de la señal; está directamente relacionado con el número de bits asignado a cada muestra (Pohlmann, 2005)

El número de niveles de cuantificación en que se divide a la amplitud de la señal depende del número de bits y se determina por la siguiente relación:

$$M = 2^n \dots (7)$$

Donde: " M " representa al número de niveles cuánticos que tendrá la onda y " n " el número de bits empleados para la digitalización de la señal.

Si se utilizan pocos bits para la cuantificación, el sonido perderá nitidez y aparecerá ruido. Además, existen ruidos causados por el redondeo en el proceso de cuantificación, aunque se han desarrollado técnicas para mitigarlo, como el Dithering. Aunque agregar ruido a una señal puede parecer extraño en el mundo del audio analógico, en el ámbito del audio digital tiene sentido y aporta beneficios significativos a una señal. Esto se debe a que el error de cuantificación ocurre cuando el número de bits se reduce en comparación con la señal original, siendo más evidente cuando una señal analógica (que teóricamente tiene un número infinito de bits) se convierte en digital.

Por ejemplo, una onda senoidal, si fuera perfecta, al ser cuantificada se convierte en una señal escalonada en lugar de una onda con curvas suaves. Esto se debe al error de cuantificación y generalmente resulta en una pérdida de resolución que afecta negativamente el sonido. Sin embargo, agregar un poco de ruido blanco a esta señal cuantificada puede suavizar la forma escalonada, creando un sonido más natural. Sorprendentemente, este proceso también amplía el rango dinámico de manera que se pueden representar señales mucho más pequeñas que el nivel digital más bajo, lo que produce una degradación más natural en el sonido después de aplicar el "Dithering".

La resolución tiene un impacto directo en la calidad de la señal. En la figura 11 se puede observar claramente este efecto y cómo la cantidad de bits de cuantificación afecta la representación y calidad del sonido.

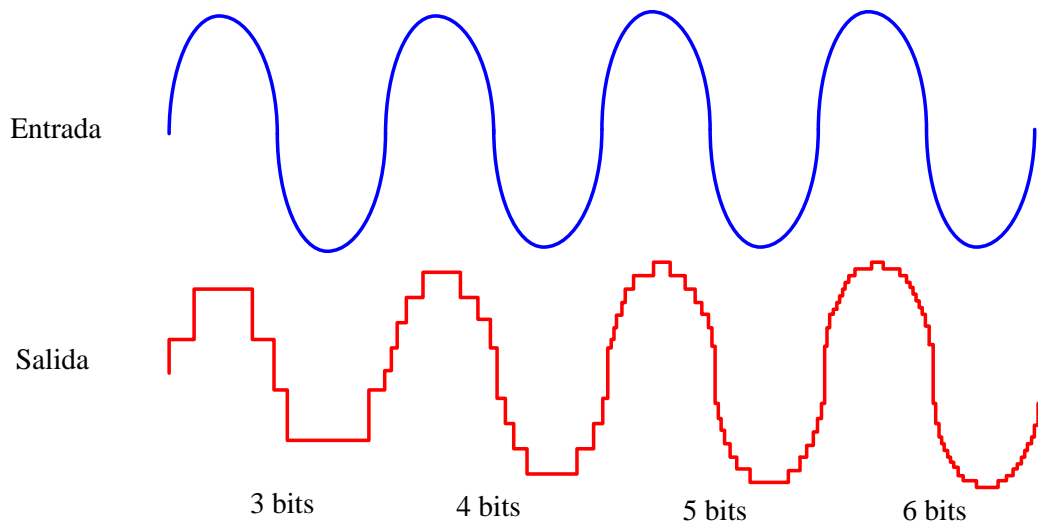


Figura 11: Efecto del tamaño de palabra del ADC en la calidad de la señal obtenida.

En la figura 12 se ilustra el proceso de discretización de la amplitud de una señal, mostrando cómo se introduce un error entre el valor real de la señal analógica y el valor que se codificará al digitalizarla. Este error aleatorio se asemeja a la adición de una componente de ruido. El valor cuantificado representa la señal original más este "ruido virtual" que se superpone a la señal, resultando en el valor que finalmente se adquiere. Este ruido puede ser tanto positivo como negativo, y su valor máximo es igual a la mitad del nivel de cuantización.

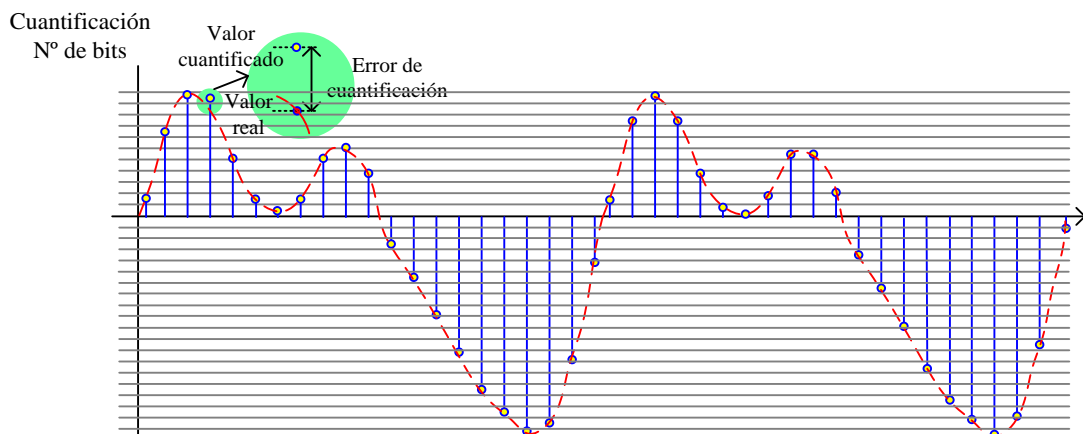


Figura 12: Cuantificación y Ruido de cuantificación

Rango Dinámico (DR)

El Rango Dinámico (DR) se refiere al nivel de señal que un sistema puede manejar sin generar distorsión, ya sea en un dispositivo de grabación o reproducción de audio. Formalmente, el Rango Dinámico describe la relación entre el sonido más suave y el más alto de una señal o instrumento eléctrico, medido en decibelios (dB). Por cada bit adicional de resolución, el Rango Dinámico se incrementa en 6 dB. Por ejemplo, una resolución de 16 bits proporcionaría un Rango Dinámico de 96 dB.

Aumentar la resolución en bits también aumenta el Rango Dinámico, lo que significa que la señal deseada se aleja del nivel de ruido de fondo. Esto resulta en una grabación más limpia y libre de distorsiones (Davis & Jones, 1990).

Es aquel nivel de señal que un sistema puede soportar sin generar distorsión cuando se habla de un aparato que graba o reproduce audio. Formalmente, el Rango Dinámico (DR) describe la relación entre el sonido más suave comparado contra el más alto de un instrumento o señal eléctrica medido en decibeles (dB). Por cada bit que se aumente de resolución, se incrementa en 6 dB, el DR que se consigue con una resolución de 16 bits sería de 96 dB. Entonces, el incrementar la resolución en bits incrementa también el rango dinámico, es decir, aleja la señal deseada del piso del ruido, lo que resulta en una grabación más limpia (Davis & Jones, 1990).

El Rango Dinámico (DR) se utiliza en equipos de audio para indicar la salida máxima del equipo y medir su nivel de ruido de fondo (noise floor). Como referencia, el rango dinámico del oído humano es de aproximadamente 120 dB.

En la figura 13 se muestra la diferencia entre una señal que abarca todo su rango dinámico y la misma señal cuantificada digitalmente, así como el error o ruido de cuantificación resultante en el dominio temporal.

Al reducir el nivel o el paso de cuantificación, se disminuye el nivel de ruido presente en la señal digitalizada. Por otro lado, para que la versión digital de la señal reproduzca con fidelidad el contenido de la información analógica, es necesario ajustar el nivel de cuantificación de acuerdo a las necesidades del sistema. Para señales de audio, el nivel de cuantificación puede definirse

como aquel en el que las capacidades del sistema auditivo no pueden distinguir entre la señal original y la señal digitalizada.

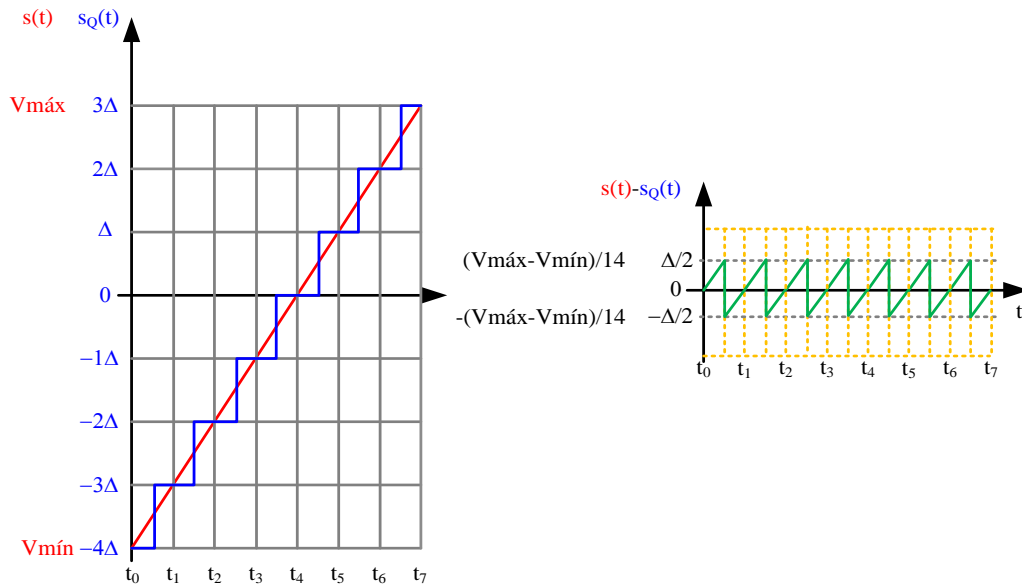


Figura 13: Diferencia entre una señal que recorre todo su rango dinámico, $s(t)$, en rojo, y la señal cuantificada $s_Q(t)$, en azul, es decir, el ruido o error de cuantificación, en verde.

El Rango Dinámico (DR) de una señal está estrechamente relacionado con el nivel de cuantificación. Si uno de ellos aumenta, el otro también debe aumentar en la misma proporción. Por ejemplo, consideremos dos señales de audio con el mismo contenido, pero una de ellas tiene un rango de voltaje de ± 1 mV, mientras que la otra tiene un rango de ± 5 mV. Si queremos escuchar ambas señales en las mismas condiciones, será necesario amplificar la primera señal en un factor de 5 veces más que la segunda (como se muestra en la figura 14). Para que los niveles de cuantificación tengan el mismo efecto auditivo, el nivel de cuantificación de la primera señal debe ser 5 veces menor que el de la segunda.

El factor más importante en cuanto a la fidelidad de la señal digitalizada es el número de niveles o valores que se pueden cuantificar.

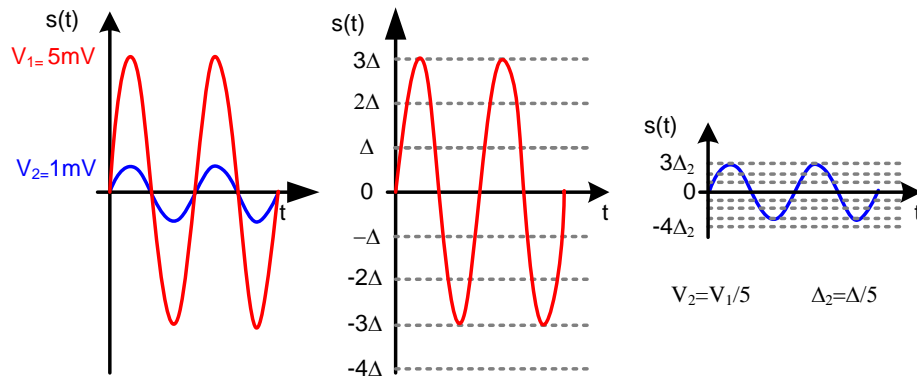


Figura 14: Igualación del número de niveles de cuantificación en señales de distinto DR ajustando sus valores máximos y mínimos.

La resolución de una señal de audio juega un papel crucial en su precisión de representación. En el ámbito del audio digital para el consumo masivo, como en los CDs o las tarjetas de sonido de las computadoras, se utiliza una resolución de 16 bits. Esto proporciona un rango de números binarios de: $2^{16} = 65,534$ niveles de cuantificación.

Sin embargo, en sistemas de audio profesional se emplean resoluciones más altas, como 20 o 24 bits. Estas mayores resoluciones permiten un mayor número de niveles de cuantificación y, por lo tanto, una representación más precisa de la señal de audio. Esto resulta en una calidad de sonido mejorada y una mayor fidelidad en la reproducción.

2.2.3.3. Codificación

La codificación PCM (Pulse Code Modulation) consiste en asignar un código binario a cada nivel de cuantificación de la señal. Con ello, cada una de las muestras queda codificada con una palabra de un número de bits fijo. Este tipo de codificación se utiliza en el sistema Compact Disc Digital Audio y en diversos sistemas de vídeo digital no comprimidos (Bosi&Goldberg, 2003) En formatos de audio comprimidos (ADPCM, Minidisc, MP3, MPEG) la codificación de la señal en PCM suele constituir la primera fase del proceso de compresión. Una vez la señal se ha convertido a PCM se analiza y procesa con el objeto de reducir el número total de bits con el que se representa, codificándola en formatos alternativos que pueden resultar más o menos complejos y/o eficientes. En cualquier caso, para reproducir una

señal comprimida también es habitual pasarla previamente al formato PCM y posteriormente convertirla a una señal analógica que pueda aplicarse a los altavoces o al sistema de representación gráfica.

En la figura 15, se muestran los códigos binarios que serían asignados a un cuantificador del tipo uniforme, es decir, que divide el rango dinámico en divisiones del mismo valor.

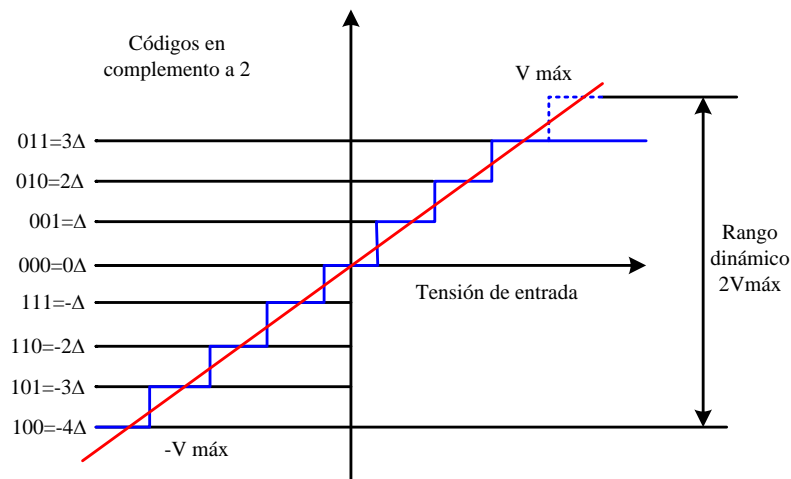


Figura 15: Codificación PCM en complemento a dos.

La codificación utilizada en la cuantificación de audio y video se conoce como codificación en complemento a 2 y es ampliamente utilizada en estos campos. En este tipo de codificación, los códigos binarios de los niveles de cuantificación positivos siempre comienzan con cero, seguido de la indicación del número de niveles de cuantificación asignados. Por ejemplo, el nivel de cuantificación 3Δ se codifica con el código 011, donde el primer 0 indica que el nivel es positivo y los siguientes dos dígitos codifican el número 3 ($1 \times 2^1 + 1 \times 2^0$).

La codificación de los niveles de cuantificación negativos es un poco más compleja. Para codificar un nivel de $-k\Delta$, primero se expresa el número k en binario natural, luego se cambian todos los 1 por 0 y viceversa (complemento a 1), y finalmente se suma 1 al resultado obtenido (complemento a 2). De esta manera, todos los códigos asociados a números negativos comienzan con 1. Por ejemplo, si consideramos el nivel de -2Δ , el código binario correspondiente al nivel 2 es 010. Al aplicar el complemento a 1, se obtiene 101, y luego al sumar 1 se obtiene el código final de 110.

En la Tabla 3 se presentan los niveles asignados para cada código de 3 bits en la codificación en complemento a 2.

Tabla 3: Niveles y códigos binarios de 3 bits en complemento a 2.

Nivel	Código binario
3	11
2	10
1	1
0	0
-1	111
-2	110
-3	101
-4	100

La codificación en complemento a dos tiene la ventaja de permitir realizar operaciones de suma entre niveles positivos y negativos sin tener en cuenta el signo. Por ejemplo, si se suman directamente los códigos correspondientes a 2 (010) y -3 (101), se obtiene el código (111 = 010 + 101) que representa el nivel -1 (2-3).

En la figura 16 se muestra un esquema del proceso de codificación de una señal. A cada muestra de la señal se le asigna el código binario correspondiente, que representa el nivel de amplitud de la señal. Estos datos pueden ser almacenados en una memoria para su posterior procesamiento, transmitidos a través de un medio adecuado o registrados en un soporte de almacenamiento masivo.

Cuando se almacenan en una memoria RAM para su tratamiento inmediato, como en efectos o ecualización digital, los datos se registran en un búfer y esperan ser procesados. Un procesador puede tomar los datos de una memoria de entrada, transformarlos y depositarlos en una memoria de salida. Los datos en el búfer de salida pueden ser reproducidos (convertidos de PCM a señal analógica), almacenados en soportes masivos como CD o DVD, o transmitidos a otros dispositivos.

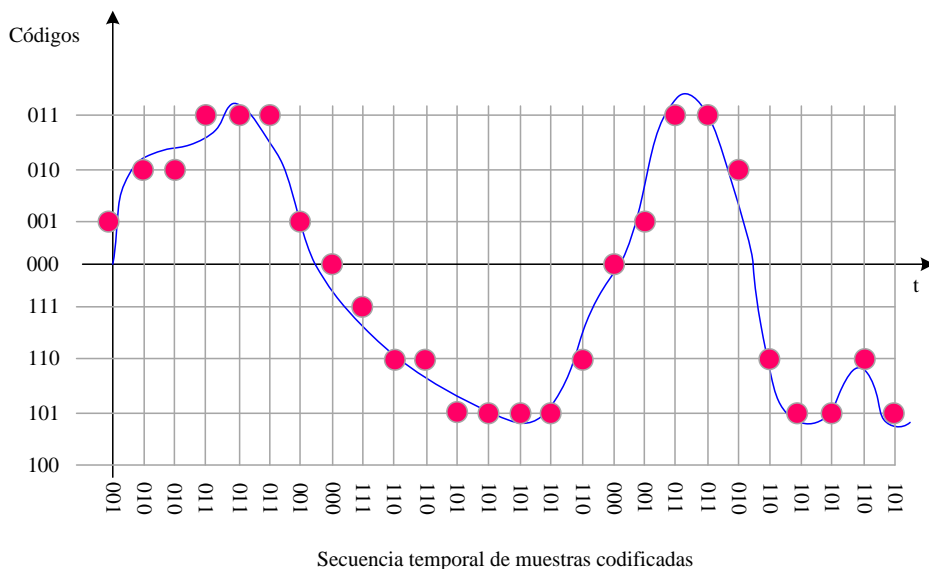


Figura 16: Ejemplo de codificación de una señal.

Para la transmisión o registro de datos, es común agrupar varias palabras de código en una misma trama. En esta trama, se incluyen bits adicionales de sincronismo que facilitan la recuperación de la secuencia. Las palabras se transmiten en formato serie, es decir, un bit después del otro, e se insertan posibles palabras de sincronismo adicionales.

En algunos casos, las palabras de código se protegen mediante palabras adicionales que permiten detectar la presencia de errores e incluso corregirlos. Esto se logra mediante técnicas de detección y corrección de errores.

La figura 17 ilustra un esquema simplificado de una posible configuración de una trama de datos, mostrando cómo se pueden estructurar los bits de sincronismo y las palabras de código en la transmisión o el registro de datos.

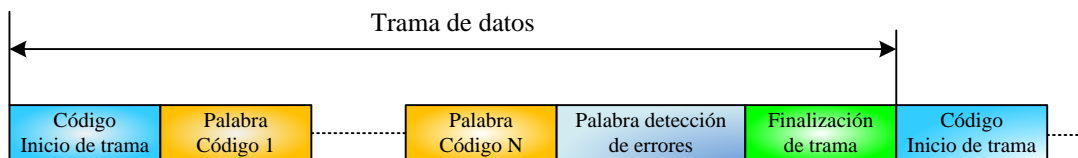


Figura 17: Transmisión de los datos PCM en forma de trama.

2.2.3.4. Conversión de las palabras código a muestras analógicas

Pohlmann (2005) manifiesta que, una vez que la señal es procesada (filtrada, transmitida, almacenada, etc.), interesa convertirla nuevamente en una señal analógica para escucharla en los altavoces. Este proceso de conversión de las palabras digitales a señales analógicas se realiza mediante conversores DAC (convertidores de digital a analógico) y filtros paso bajo. Desde un punto de vista puramente funcional (no electrónico) los conversores de digital analógico se encargan de convertir a valores de tensión los códigos binarios correspondientes a cada una de las muestras y mantienen este valor en la salida durante todo el periodo de muestreo. Los códigos binarios de las muestras deben presentarse al convertor con la misma cadencia con la que ha sido muestreada la señal. En el supuesto de que las frecuencias de muestreo de grabación y reproducción no coincidan se produce un efecto parecido al cambio de velocidad de reproducción en un disco de vinilo o en una cinta magnética. El filtro paso bajo situado en la salida del convertor se encarga de suavizar esta señal, es decir, de eliminar las componentes de frecuencias altas, que se corresponden con los saltos producidos por el empleo de señales cuantificadas. En algunos circuitos integrados, el convertor y el filtro reconstructor de salida están integrados en el mismo sistema.

En la figura 18 se representa un diagrama de bloques del sistema de conversión DAC y las formas de onda representativas.

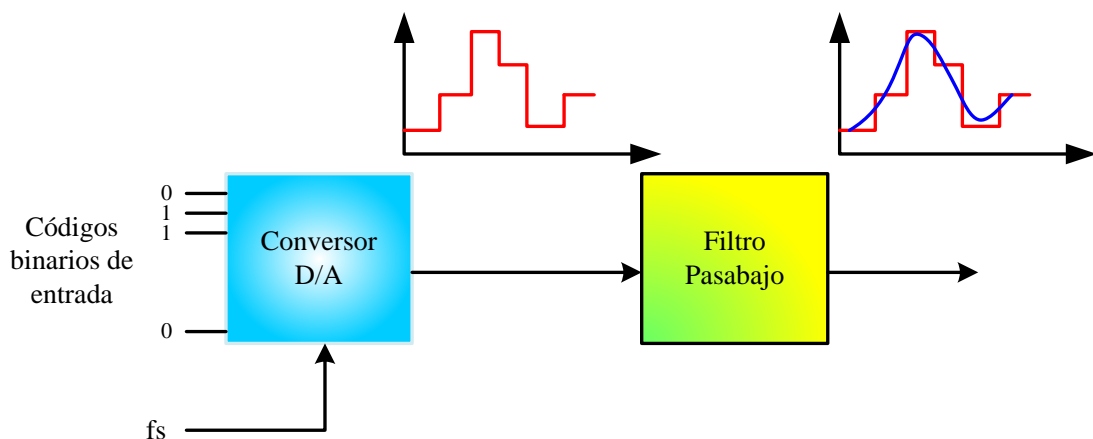


Figura 18: Diagrama de bloques de un sistema de conversión de digital a analógico

2.2.3.5. Ventajas e inconvenientes de la codificación digital de las señales

La codificación digital de señales presenta varias ventajas y desventajas, que se describen a continuación:

Ventajas de la codificación digital:

Precisión: La codificación digital permite una representación precisa de las señales, ya que se utilizan valores discretos y se pueden utilizar diferentes niveles de cuantificación para capturar detalles finos.

Reproducibilidad: Las señales digitales se pueden reproducir con alta fidelidad, siempre y cuando se mantenga la resolución y el rango dinámico adecuados.

Manipulación y procesamiento: Las señales digitales se pueden manipular y procesar de manera más flexible y eficiente que las señales analógicas. Esto permite aplicar una amplia gama de técnicas de procesamiento de señales, como ecualización, filtrado, modificación del tono, efectos especiales, etc.

Almacenamiento y transmisión eficientes: Las señales digitales se pueden comprimir y almacenar en formatos más compactos, lo que permite un uso eficiente del espacio de almacenamiento. Además, se pueden transmitir de manera más confiable y con mayor capacidad a través de canales de comunicación digital.

Inmunidad al ruido y degradación: Las señales digitales tienen una mayor resistencia al ruido y la degradación en comparación con las señales analógicas. Esto se debe a que los sistemas digitales pueden emplear técnicas de detección y corrección de errores, lo que permite una mayor robustez en la transmisión y reproducción de señales.

Desventajas de la codificación digital:

Pérdida de información: En algunos casos, la cuantificación y discretización de las señales analógicas pueden resultar en una pérdida de información, lo que puede afectar la calidad y la fidelidad de la señal original.

Limitaciones de la resolución: La resolución limitada de los sistemas digitales puede introducir errores y artefactos en la señal, especialmente en señales de baja amplitud o en cambios rápidos de amplitud.

Mayor complejidad y costo: Los sistemas digitales suelen requerir una mayor complejidad y costos asociados, como la necesidad de convertidores analógico-digitales (ADC) y digitales-analógicos (DAC), así como de procesadores y almacenamiento de mayor capacidad.

Latencia: En algunas aplicaciones en tiempo real, la conversión y procesamiento digital de señales puede introducir una cierta cantidad de retraso o latencia, lo que puede ser problemático en ciertos contextos, como la grabación de audio en vivo o la comunicación en tiempo real.

En general, a pesar de las posibles limitaciones, la codificación digital de señales ha revolucionado numerosos campos y ha permitido avances significativos en el procesamiento, almacenamiento y transmisión de información de señales.

2.2.3.6. Tasa de bits (Bit rate: BR) del audio digital

El bitrate de un archivo de audio se refiere a la cantidad de bits que se utilizan por unidad de tiempo para codificar o representar la información de audio en el archivo. Es una medida de la tasa de transferencia de datos de audio y determina la calidad y el tamaño del archivo.

El bitrate se expresa típicamente en kilobits por segundo (kbps) o megabits por segundo (Mbps). Un bitrate más alto indica una mayor cantidad de datos utilizados por segundo y generalmente resulta en una mayor calidad de audio, ya que se pueden capturar más detalles y matices del sonido.

Sin embargo, un bitrate más alto también implica un tamaño de archivo más grande, lo que puede requerir más almacenamiento y ancho de banda para transmitir o descargar el archivo. Por lo tanto, el bitrate de un archivo de audio es un equilibrio entre la calidad del sonido deseada y las consideraciones de almacenamiento y transmisión.

Para el cálculo completo, se emplea la siguiente ecuación:

$$\text{bitrate} = (\text{Frec. de muestreo}) \times (\text{N}^\circ \text{ de bits}) \times \left(\frac{\text{N}^\circ \text{ de } C}{\text{hs}} \right) \dots (8)$$

El flujo de información de un archivo de audio debe incluir bits adicionales relacionados con códigos correctores de errores, palabras de sincronización, códigos de tiempo, códigos de acondicionamiento de datos, entre otros. Esto implica que la tasa mínima real a la que deben leerse los bits del archivo es aproximadamente cuatro veces mayor que la tasa calculada inicialmente.

Incluso si no se introducen códigos de protección, sincronización y acondicionamiento, la tasa de bits de 1,41 Mbps sigue siendo muy alta en comparación con la información en formato analógico original, que solo ocupaba 40 KHz (20 KHz por canal).

El ancho de banda necesario para transmitir este flujo de bits (sin errores) depende del tipo de modulación digital utilizada. Aproximadamente, se requiere alrededor de 0,5 Hz de ancho de banda por cada bit que se desea transmitir. Esta aproximación se basa en la suposición de que los bits se transmiten en banda base, uno después del otro, asignando un valor de tensión positivo para los unos y negativo para los ceros. Según esta aproximación, la frecuencia máxima de la señal resultante corresponde a una secuencia de unos y ceros alternados (101010101), que, a una tasa de 1,4 Mbps, resulta en una frecuencia fundamental de aproximadamente 700 KHz. En la práctica, existen formas de codificar la información (modulaciones) que pueden reducir ligeramente esta frecuencia máxima.

En resumen, si se desea transmitir los bits resultantes después del proceso de digitalización, se requeriría un ancho de banda mucho mayor que los 40 KHz originales, aproximadamente del orden de los 700 KHz. Esto implica un aumento en el ancho de banda de la señal digital de alrededor de 20 veces. Por lo tanto, los cables utilizados para la conexión de equipos de audio digital, como CD, Minidisc y DAT, deben ser capaces de proporcionar este ancho de banda para evitar la degradación de las señales durante el intercambio de información.

2.2.3.7. Tamaño del archivo de audio

Según Mandal (2003), el tamaño de un archivo de audio puede variar significativamente dependiendo de varios factores, como la duración de la grabación, el formato de archivo utilizado y el bitrate.

En general, los archivos de audio sin comprimir, como los archivos WAV o PCM, tienden a ser más grandes debido a que almacenan la información de audio de forma más detallada y precisa. Estos archivos pueden ocupar varios megabytes o incluso gigabytes, dependiendo de la duración y la calidad del audio.

Por otro lado, los archivos de audio comprimidos, como los archivos MP3 o AAC, utilizan algoritmos de compresión para reducir el tamaño del archivo. La compresión de audio se basa en eliminar cierta información redundante o menos perceptible para el oído humano. El tamaño de los archivos comprimidos puede variar según la configuración de compresión utilizada, como el bitrate. Un archivo MP3 con un bitrate más bajo tendrá un tamaño más pequeño, pero también puede resultar en una calidad de audio inferior. En resumen, el tamaño de un archivo de audio depende de la duración, el formato de archivo y el bitrate utilizado. Los archivos sin comprimir tienden a ser más grandes, mientras que los archivos comprimidos pueden variar en tamaño dependiendo de la configuración de compresión.

En la siguiente ecuación, se muestra la fórmula para calcular el tamaño del archivo de audio:

$$\left(\begin{array}{l} \text{Tamaño del} \\ \text{archivo en} \\ \text{Bytes} \end{array} \right) = \left(\begin{array}{l} \text{frec. de} \\ \text{muestreo} \end{array} \right) \times (\text{N}^\circ \text{ de bits}) \times \left(\begin{array}{l} \text{N}^\circ \text{ de} \\ \text{Chs} \end{array} \right) \times \left(\begin{array}{l} \text{Tiempo} \\ \text{en} \\ \text{segundos} \end{array} \right) \times (1/8) \dots (9)$$

2.2.4. Producción musical

La producción musical es el proceso de crear y supervisar la grabación y el desarrollo de una canción o un álbum. Comprende una serie de etapas que incluyen la preproducción, la grabación, la mezcla y la masterización.

La preproducción es la etapa inicial, donde se planifica y se establecen los objetivos para la grabación. Aquí se definen los arreglos musicales, se seleccionan los músicos y se preparan las partituras de ser necesario.

La grabación es el proceso de capturar las interpretaciones musicales. El productor musical supervisa las sesiones y trabaja con los músicos para obtener las mejores tomas. Puede sugerir cambios en los arreglos, probar

diferentes micrófonos y equipos, y asegurarse de que la calidad de sonido sea óptima.

La mezcla, en esta etapa, se ajusta el balance de los diferentes elementos de la canción, como las pistas vocales, los instrumentos y los efectos. El productor trabaja junto a un ingeniero de mezcla para lograr un equilibrio sonoro coherente y una buena calidad de audio.

La masterización, es la etapa, donde se realiza el proceso final de ajuste y optimización del sonido. Se aplican técnicas de ecualización, compresión y limitación para mejorar la claridad, el volumen y la cohesión del álbum o de la canción individual. El objetivo es que la música suene bien en diferentes sistemas de reproducción y mantenga un nivel de calidad profesional.

Además de estas etapas, el productor musical también puede estar involucrado en la selección de canciones, la dirección artística, la gestión del presupuesto y la promoción del proyecto.

2.3. Marco Conceptual

Parámetros de digitalización de audio:

Es una variable que considera como parámetros clave para digitalizar el audio a la frecuencia de muestreo, el número de bits y el formato del archivo de audio.

Calidad de audio digital

Es una variable que se refiere a la precisión y fidelidad con la que se reproduce o se almacena el sonido en formato digital con la mejor relación señal a ruido.

2.4. Sistema de Hipótesis

Los valores más altos de los parámetros de digitalización influyen en la calidad de un archivo de audio producido digitalmente.

2.5. Variables e indicadores

Tabla 4: Variables e indicadores

Variable	Tipo	Indicador	Unidad de medida	Técnica	Instrumento
parámetros de digitalización	Independiente	Frecuencia de muestreo	Hz	Observación	Guía de observación
		Profundidad de bits	Nº de bits	Observación	Guía de observación
Calidad de archivo de audio	Dependiente	Relación S/N	dB	Observación	Guía de observación

III. METODOLOGÍA EMPLEADA

3.1. Tipo y nivel de investigación

Tipo de investigación: Documental y de campo

Nivel de investigación: Explicativa

3.2. Población y muestra de estudio

La población del estudio estará formada por archivos de audio digital generados con los parámetros de digitalización más altos.

La muestra estará formada por archivos de audio digital configurados con formato WAV con frecuencias de muestreo superiores al índice de Nyquist y con profundidad de bits de 16 y 24.

3.3. Diseño de investigación

El diseño de contrastación será de tipo pre - experimental:

$$O_1 \quad X \quad O_2$$

Donde:

O₁: Medición de calidad de archivo de audio digital sin variación de parámetros

X: Variación de frecuencia de muestreo y profundidad de bits

O₂: Medición de calidad de archivo de audio digital con variación de parámetros.

3.4. Técnicas e instrumentos de investigación

Para la presente investigación, se emplearon las técnicas e instrumentos de investigación que se muestran en la tabla siguiente:

Tabla 5: Técnicas e instrumentos de recolección de datos

Técnica de Recolección de datos	Instrumento de recolección de datos
Análisis documental	Guía de análisis documental de información relevante de niveles de presión sonora y frecuencias generadas por los instrumentos musicales
	Guía de análisis documental de información relevante de micrófonos que se emplean para captar el sonido de los instrumentos musicales usados en la producción musical.
Observación	Guía de observación para obtener información de los archivos de audio generados a partir de la mezcla maestra modificando la profundidad de bits y la frecuencia de muestreo.

Los instrumentos de Investigación empleados en el presente trabajo, se muestran en el anexo 1.




3.5. Procesamiento y análisis de datos

Considerando la disponibilidad de instrumentos musicales, equipamiento y músicos para realizar la producción musical, se eligió realizar la producción musical del vals criollo Hilda cuyo autor fue el maestro Alberto Haro y que fuera interpretada por los Embajadores Criollos entre los años 1950 y 1960.

1. Se emplearon 14 instrumentos musicales, de los cuales 6 son instrumentos de cuerda (3 guitarras electroacústicas, 1 guitarra silent, 1 bajo acústico y 1 bajo electrónico) 2 instrumentos sintetizadores electrónicos (piano y órgano electrónico), 1 instrumento de viento

(clarinete) y 5 instrumentos de percusión (2 cajones, 1 hi hat, 1 bombo y castañuelas), los cuales se detallan en la tabla 6.

Tabla 6: Instrumentos musicales empleados en la producción musical del vals Hilda

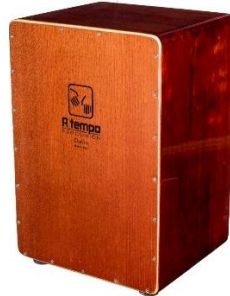
Nº	Instrumento	Imagen	Marca	Modelo
1	Guitarra Silent		Yamaha	SLG200N F/S
2	Guitarra electroacústica		Fender	FA-125CE
3	Guitarra electroacústica		Yamaha	NCX5 NT
4	Guitarra electroacústica		Fender	CD-60SCE
5	Bajo acústico		Fender	Kingman V2
6	Bajo electrónico		Ibanez	GSR205 BK
7	Piano		Kurzweil	KAG100BP
8	Órgano		Yamaha	PSR-E473
9	Clarinete		Yamaha	YCL-255

10 Cajón base



Percusión
Real

11 Cajón de
relleno



Atempo

12 Hi hat



Millenium MPS-750X

13 Bombo



Evans BD24GB4

14 Castañuelas



2. Para captar los sonidos de los 14 instrumentos musicales, se identificaron los que requería de micrófonos de acuerdo al tipo de instrumento musical, no todos los instrumentos requieren de micrófonos para captar su sonido

porque pueden captarse directamente a la consola de audio; en la tabla 7, se muestran los instrumentos que requieren el uso de micrófonos para captar los sonidos que generan.

Tabla 7: Requerimiento de micrófonos de los instrumentos musicales.

Nº	Instrumento	Micrófono	Marca	Modelo
1	Guitarra Silent	No requiere	-	-
2	Guitarra electroacústica	No requiere	-	-
3	Guitarra electroacústica	No requiere	-	-
4	Guitarra electroacústica	No requiere	-	-
5	Bajo acústico	No requiere	-	-
6	Bajo electrónico	No requiere	-	-
7	Piano	No requiere	-	-
8	Órgano	No requiere	-	-
9	Clarinete		Shure	SM 137
10	Cajón base		Shure	SM 57
11	Cajón de relleno		Shure	SM 57
12	Hi hat		Shure	SM137
13	Bombo		Shure	Beta 52A
14	Castañuelas		Shure	SM 58

- Los sonidos que generen los diferentes instrumentos musicales, son mezclados en una consola digital de marca Soundcraft modelo Ui24R, la conexión se hizo por cables con conectores tipo plug para los instrumentos musicales que no requieren micrófono y por cables con conectores XLR para los micrófonos, como se muestra en la figura 19.



Cable con conectores plug



Cables con conectores XLR

Figura 19: Cables con conectores para consola

La conexión completa de micrófonos e instrumentos musicales, se muestra en la figura 20.

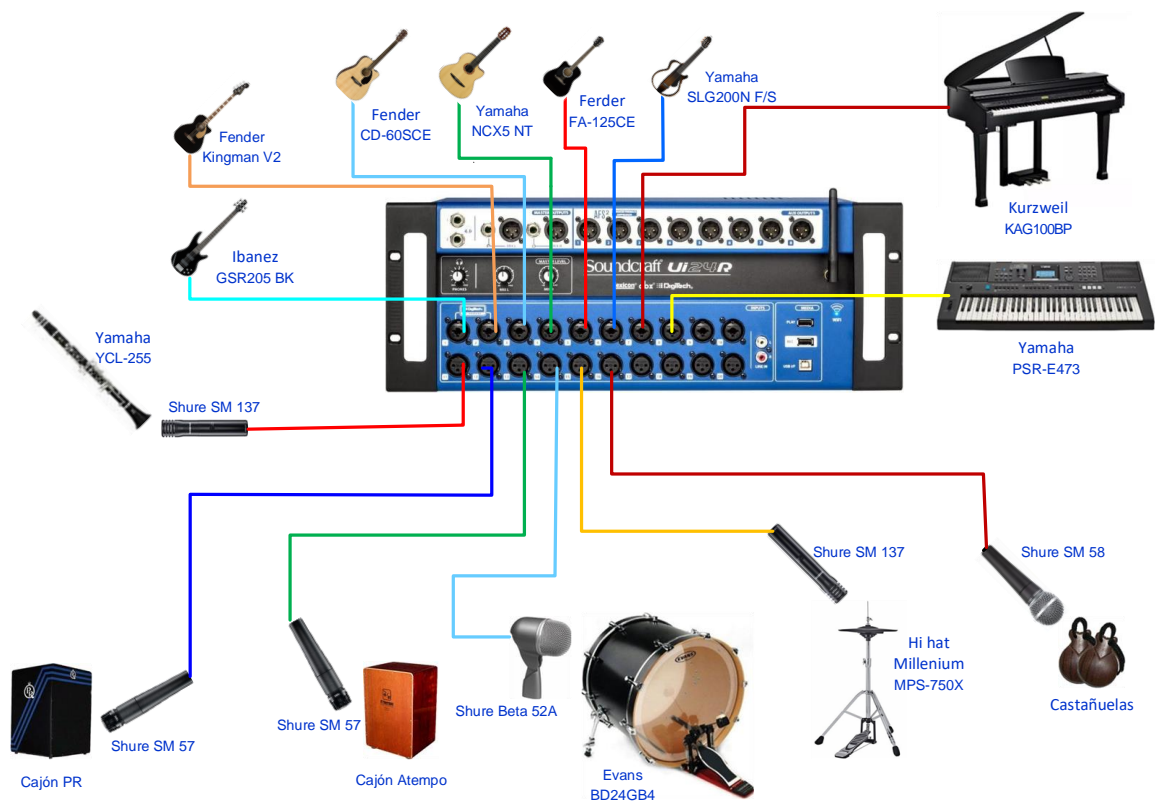


Figura 20: Conexión de micrófonos e instrumentos a la consola digital Soundcraft Ui24R

4. Para configurar los niveles de ganancia de cada una de las entradas y los niveles de salida a la mezcla maestra en la consola Soundcraft Ui24R, es necesario conocer los niveles de presión sonora que produce cada uno de los instrumentos musicales así como las frecuencias que generan cuando interpretan música, estos niveles se obtuvieron mediante la

técnica de análisis documental y recolectados de acuerdo a la guía de análisis documental de instrumentos musicales considerando la información de los anexos 2 y 3; asimismo, es necesario conocer las características técnicas de los diferentes tipos de micrófonos que se emplearán para captar los sonidos de los instrumentos musicales.





En la tabla 8, se muestran los datos procesados de los niveles de presión sonora y las frecuencias de interpretación de instrumentos musicales.

Tabla 8: Niveles de presión sonora y frecuencias de interpretación de los instrumentos musicales utilizados en la producción musical digital.

Nº	Tipo de instrumento musical	Descripción	Nivel de Presión sonora (dB _{SPL})	Frecuencias de interpretación	
				Frecuencia Fundamental (Hz)	Frecuencia de Armónicos máx (Hz)
1	Cuerda	Guitarra 1	70 - 90	300 – 800	15 KHz
2	Cuerda	Guitarra 2	70 - 90	125 – 400	14 KHz
3	Cuerda	Guitarra 3	70 - 90	200 – 1200	15,6 KHz
4	Cuerda	Guitarra 4	70 - 90	200 – 700	14,5 KHz
5	Cuerda	Bajo acústico	70 - 100	90 – 250	5 KHz
6	Cuerda	Bajo electrónico	80 - 110	85 – 300	6,2 KHz
7	Electrónico	Piano	60 - 100	125 -3000	8 KHz
8	Electrónico	Órgano	70 - 110	130 – 800	15,5 KHz
9	Viento	Clarinete	80 - 95	180 – 1200	11 KHz
10	Percusión	Cajón base	70 - 90	80 – 300	5 KHz
11	Percusión	Cajón de relleno	70 - 90	250 – 340	7 KHz
12	Percusión	Hi hat	80 – 95	350 -500	16 KHz
13	Percusión	Bombo	90 - 105	50 – 150	6, 5 KHz
14	Percusión	Castañuelas	70 - 90	300 – 3000	10 KHz

En la tabla 9, se muestran las características técnicas obtenidas mediante la guía de análisis documental de información relevante de micrófonos que se usaron para captar los sonidos de los instrumentos de viento y percusión para la producción musical.

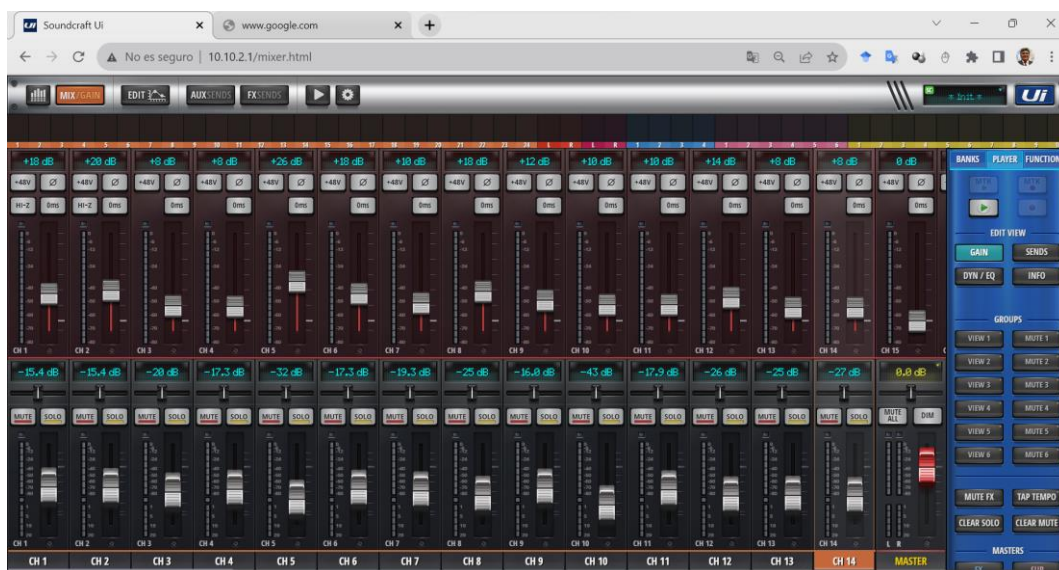
Tabla 9: Información técnica de micrófonos usados para captar sonidos de los instrumentos de viento y percusión.

Nº	Micrófono	Marca	Modelo	Tipo	Respuesta en frecuencia	Sensibilidad dBV/Pa	Patrón polar
1		Shure	SM 137	Electreto	20 Hz a 20 KHz	-41	Cardioide
2		Shure	SM 57	Dinámico	40 Hz a 15 KHz	-54.5	Cardioide
3		Shure	Beta 52A	Dinámico	20 Hz a 10 KHz	-64	Super cardioide
4		Shure	SM 58	Dinámico	50 Hz a 15 KHz	-54.5	Cardioide

- Se probaron todos los instrumentos musicales y se configuraron los niveles de ganancia y de salida a la mezcla maestra, así como la ecualización en base a la información de la tabla 8.

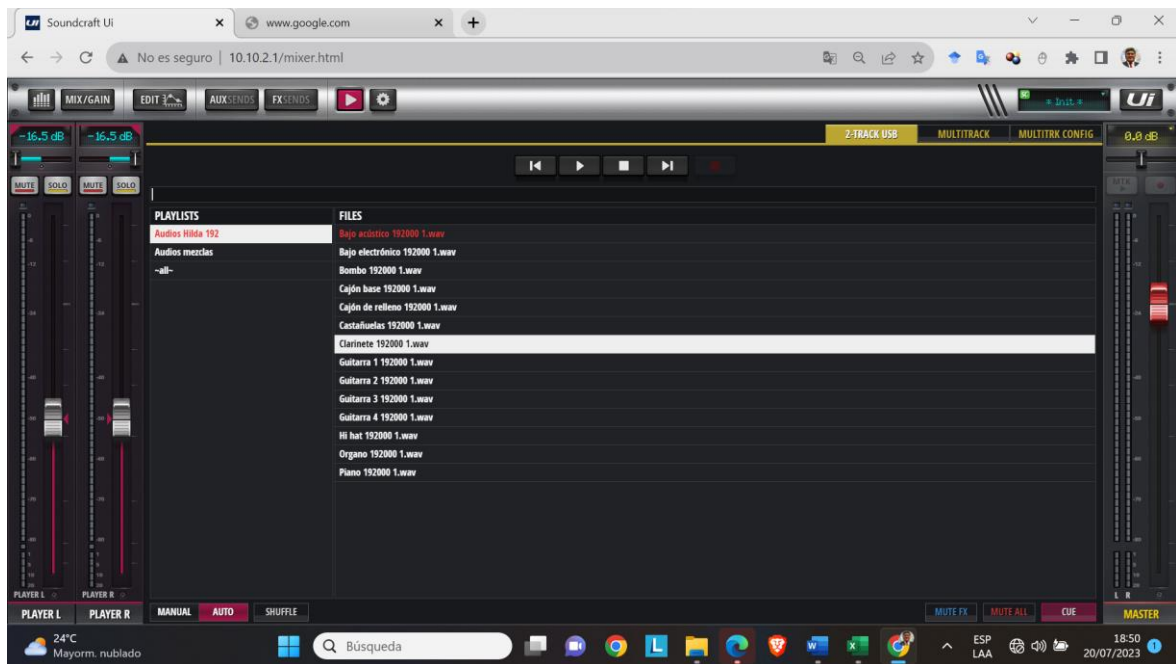
En la figura 21, se ilustra la configuración de los niveles de ganancia y salida de cada uno de los instrumentos musicales en la consola Soundcraft Ui24R.

Figura 21: Niveles de ganancia y salida a mezcla maestra de instrumentos musicales en la consola Soundcraft Ui24R.



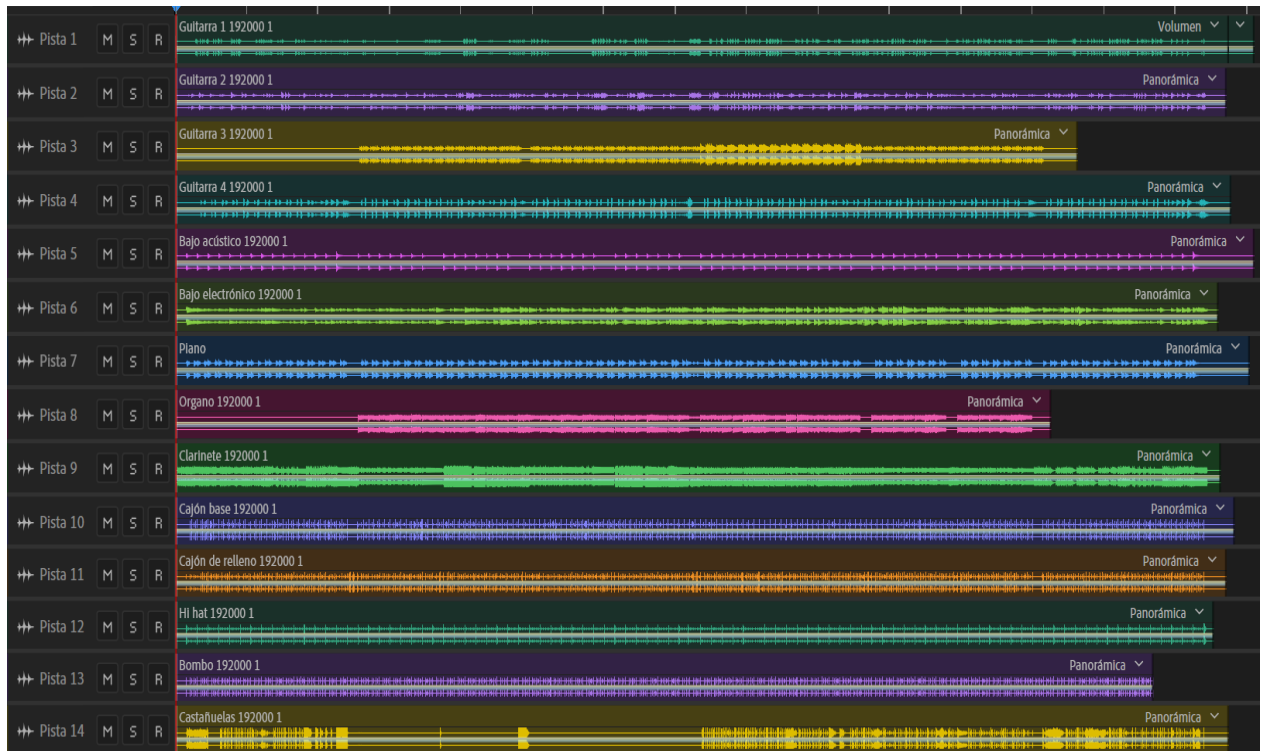
- Los sonidos de cada instrumento musical, y la mezcla maestra, se grabaron como multipista, es decir cada instrumento en forma independiente, asimismo el archivo de la mezcla maestra en la consola Soundcraft Ui24R como archivos de audio en formato *.wav con una profundidad de 24 bits y una frecuencia de muestreo de 192 KHz.

Figura 22: Visualización de los 14 archivos de audio digital generados en formato *.wav en la consola Soundcraft Ui24R para la producción del vals criollo Hilda.



- Los archivos de audio generados por los 14 instrumentos musicales, se cargaron en el software Adobe Audition 2022, a fin de evidenciar que la reproducción de todos los archivos de audio reproduce digitalmente el vals criollo Hilda. En la figura 22, se muestran todos los archivos de audio cargados.

Figura 23: Visualización de los 14 archivos de audio digital generados en formato *.wav para la producción musical del vals Hilda con una profundidad de 24 bits y 192 KHz de frecuencia de muestreo.



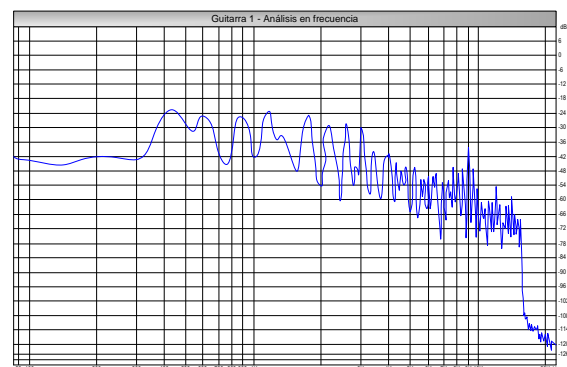
8. Cada uno de los archivos de audio de los instrumentos musicales, presenta diferente densidad espectral de potencia, para obtener las gráficas de la densidad espectral de potencia, se han empleado algoritmos en el programa Matlab, los cuales se muestran la figura 23.

Figura 24: Algoritmos y capturas de las gráficas de la densidad espectral de potencia de cada uno de los instrumentos musicales empleados para la producción musical del vals Hilda.

```
% Cargar la señal
[x, fs] = audioread('Guitarra 1.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Guitarra 1 - Análisis en frecuencia');
```



Espectro de Frecuencia de guitarra 1

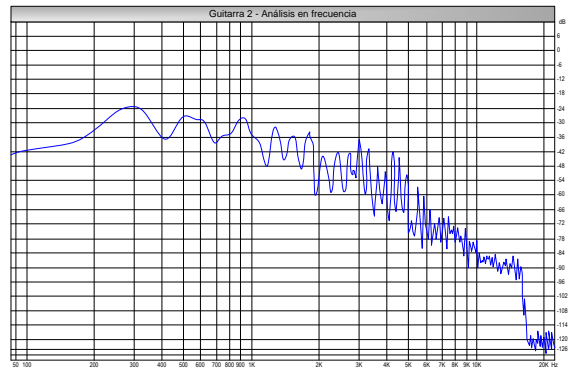
```

% Cargar la señal
[x, fs] = audioread('Guitarra 2.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Guitarra 2 - Análisis en frecuencia');

```



Spectro de Frecuencia de bajo guitarra 2

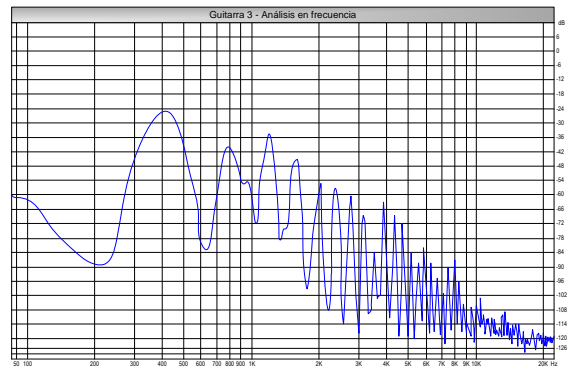
```

% Cargar la señal
[x, fs] = audioread('Guitarra 3.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Guitarra 3 - Análisis en frecuencia');

```



Spectro de Frecuencia de guitarra 3

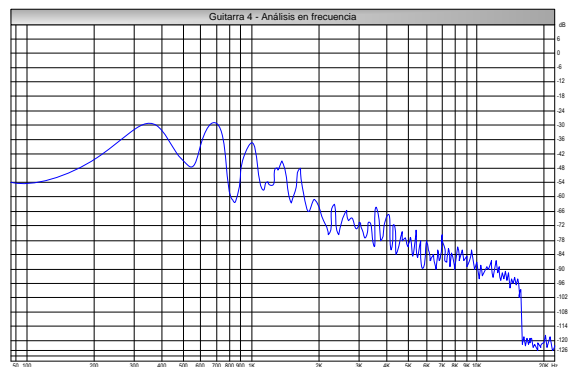
```

% Cargar la señal
[x, fs] = audioread('Guitarra 4.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Guitarra 4 - Análisis en frecuencia');

```



Spectro de Frecuencia de guitarra 4

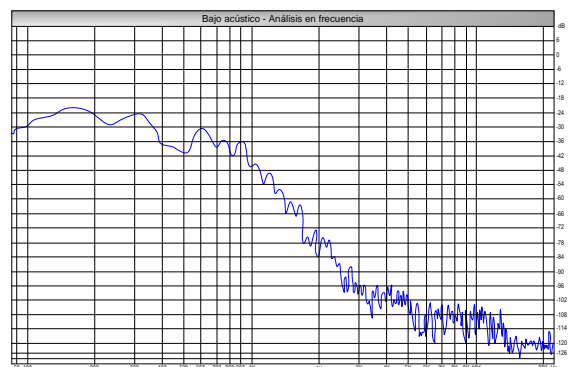
```

% Cargar la señal
[x, fs] = audioread('Bajo acústico.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Bajo acústico - Análisis en frecuencia');

```



Spectro de Frecuencia de bajo acústico

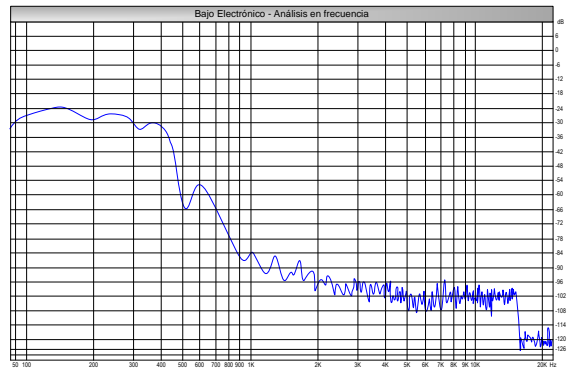
```

% Cargar la señal
[x, fs] = audioread('Bajo electrónico.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Bajo electrónico - Análisis en frecuencia');

```



Espectro de Frecuencia de bajo electrónico

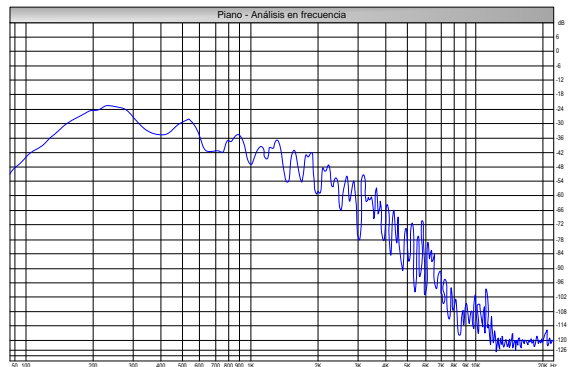
```

% Cargar la señal
[x, fs] = audioread('Piano.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Piano - Análisis en frecuencia');

```



Espectro de Frecuencia de piano

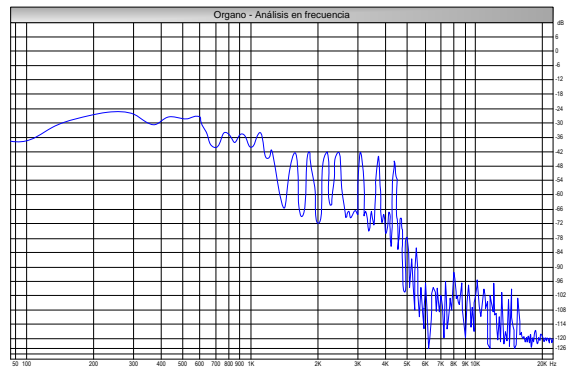
```

% Cargar la señal
[x, fs] = audioread('Organo.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Organo - Análisis en frecuencia');

```



Espectro de Frecuencia de órgano

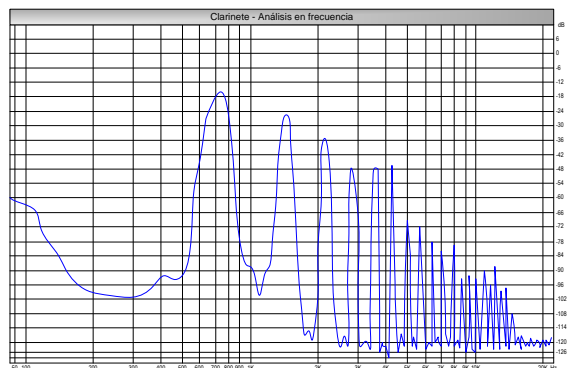
```

% Cargar la señal
[x, fs] = audioread('Clarinete.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Clarinete - Análisis en frecuencia');

```



Espectro de Frecuencia de clarinete

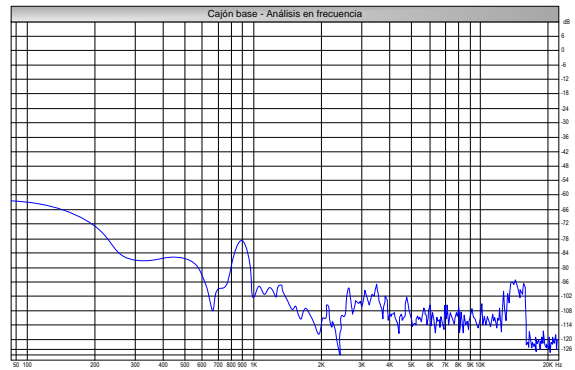
```

% Cargar la señal
[x, fs] = audioread('Cajón base.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Cajón base - Análisis en frecuencia');

```



Espectro de Frecuencia de cajón base

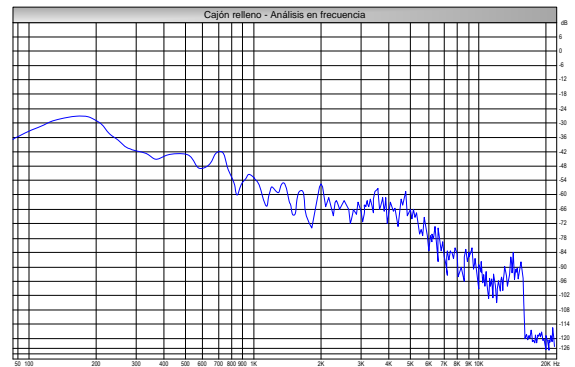
```

% Cargar la señal
[x, fs] = audioread('Cajón relleno.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Cajón relleno - Análisis en frecuencia');

```



Espectro de Frecuencia de cajón de relleno

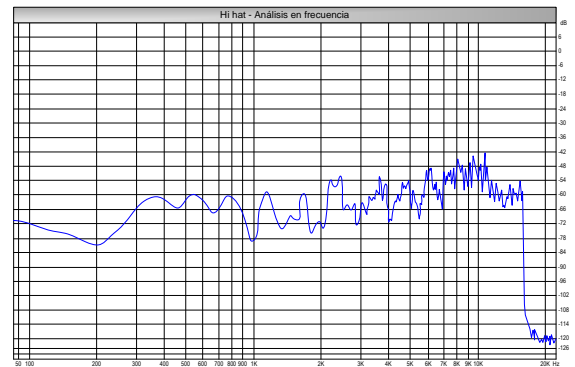
```

% Cargar la señal
[x, fs] = audioread('Hi hat.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Hi hat - Análisis en frecuencia');

```



Espectro de Frecuencia de hi hat

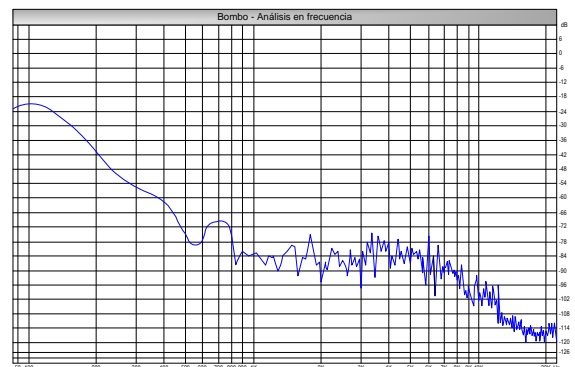
```

% Cargar la señal
[x, fs] = audioread('Bombo.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Bombo - Análisis en frecuencia');

```



Espectro de Frecuencia de bombo

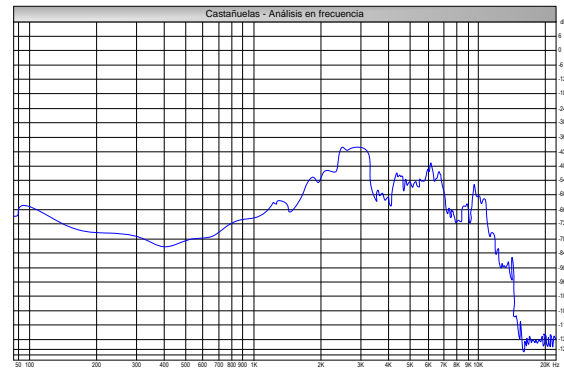
```

% Cargar la señal
[x, fs] = audioread('Castañuelas.wav');

% Calcular la densidad espectral de potencia
[Pxx, f] = pwelch(x, [], [], [], fs);

% Graficar la densidad espectral de potencia
semilogx(f, 10*log10(Pxx));
xlabel('Hz');
ylabel('dB');
title('Castañuelas - Análisis en frecuencia');

```



Espectro de Frecuencia de castañuelas

9. A partir de todos los archivos de audio de los 14 instrumentos musicales se genera la mezcla maestra que sería el resultado de la producción musical, en la figura 24, se muestra la onda de la mezcla maestra y su espectrograma generada en formato *.wav con una profundidad de 24 bits y 192000 Hz de frecuencia de muestreo.

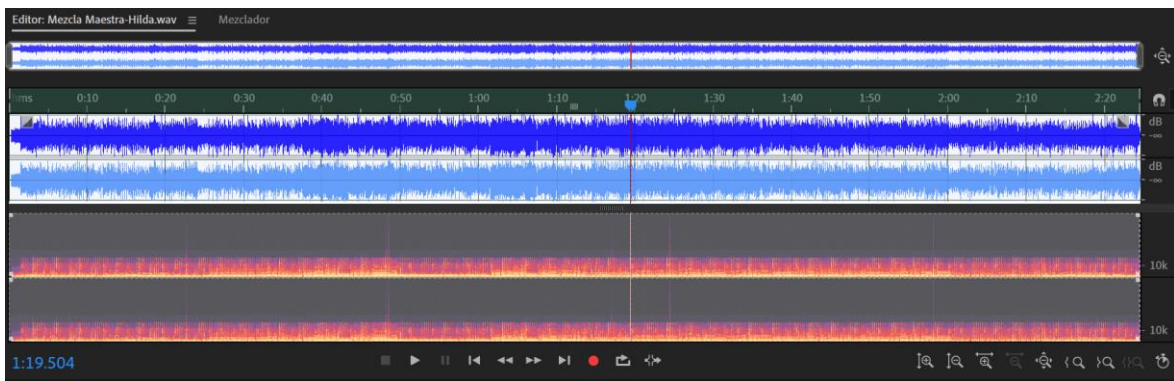


Figura 25: Onda de la mezcla maestra del vals criollo Hilda y su espectrograma.

10. La reproducción del sonido de la producción musical del vals Hilda generado en la mezcla maestra tiene una duración de 2 min con 31 segundos lo que equivale a 151 segundos, el archivo de audio fue muestreado a 192000 Hz, cuantificada con 24 bits, generada en 2 canales de audio, esta configuración, generó 28992000, el bitrate y el tamaño del archivo se obtienen por los siguientes cálculos:

$$\text{Bitrate} = (fs) \times (N^\circ \text{ de bits}) \times (N^\circ \text{ Chs}) = 92000 \times 24 \times 2 = 9216 \text{ Kbps}$$

$$\left(\begin{array}{l} \text{Tamaño de} \\ \text{archivo en} \\ \text{Bytes} \end{array} \right) = (fs) \times (N^\circ \text{ de bits}) \times (N^\circ \text{ Chs}) \times (t \text{ en seg}) \times \left(\frac{1}{8} \right)$$

$$\left(\begin{array}{l} \text{Tamaño de} \\ \text{archivo en} \\ \text{Bytes} \end{array} \right) = 192000 \times 24 \times 2 \times 151 \times 1/8 = 173,95 \text{ Mbytes}$$

11. Para poder analizar la calidad del audio que contiene el archivo de la mezcla maestra, es necesario obtener la información que contiene el archivo que no se puede evaluar directamente de la onda de la mezcla maestra ni en el espectrograma de la figura 24, por ello mediante algoritmos en Matlab se obtuvo la información necesaria para analizar la calidad del audio producido y almacenado en el archivo Hilda_192_24.wav, los algoritmos se muestran a continuación:

Algoritmo para determinar amplitud pico de la señal de audio

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_24.wav');

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);

% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));

% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_24.wav');

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);

% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);

% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
```

```

% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);

```

Algoritmo para determinar el desplazamiento DC de la señal de audio

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_24.wav');

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);

% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);

% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);

```

Algoritmo para determinar el rango dinámico de la señal de audio

```

% Leer el archivo de audio
[audio, ~] = audioread('Hilda_192_24.wav');

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);

% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);

% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));

% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_24.wav');

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);

% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);

% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);

```



```

ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);

% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);

% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```

Al ejecutar los algoritmos sobre el archivo de audio de la mezcla maestra Hilda_192_24.wav, se obtuvo la información que se resume en la tabla 10.

Tabla 10: Información obtenida de la señal de audio de la mezcla maestra Hilda_192_24.wav de

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	192 KHz	
Profundidad de bits	24	
Número de canales	2	
Número de muestras	2899200	
Bitrate	9,22 Mbps	
Tamaño de archivo	173,95 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,47 dB	38,26 dB
Relación señal a ruido	146.25 dB	146.22 dB

12. A fin de analizar comparativamente la calidad del audio de la mezcla maestra producida con las variaciones de los parámetros de digitalización es necesario hacer las modificaciones de la profundidad de bits y las frecuencias de muestreo del archivo de la mezcla maestra, para ello se realizaron algoritmos en Matlab que se muestran a continuación:

Algoritmo para generar el archivo de 96 KHz y 24 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```
input_file = 'Hilda_192_24.wav';
output_file = 'Hilda_96_24.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);
% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 96000;
target_bits_per_sample = 24;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);
```

Algoritmo para generar el archivo de 48 KHz y 24 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```
input_file = 'Hilda_192_24.wav';
output_file = 'Hilda_48_24.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);
% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 48000;
target_bits_per_sample = 24;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);
```

Algoritmo para generar el archivo de 44,1 KHz y 24 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```
input_file = 'Hilda_192_24.wav';
output_file = 'Hilda_44_24.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);
% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 44100;
target_bits_per_sample = 24;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
```

```

resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);

```

Algoritmo para generar el archivo de 192 KHz y 16 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```

input_file = 'Hilda_192_24.wav';
output_file = 'Hilda_192_16.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);
% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 192000;
target_bits_per_sample = 16;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);

```

Algoritmo para generar el archivo de 96 KHz y 16 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```

input_file = 'Hilda_192_24.wav';

output_file = 'Hilda_96_16.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);
% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 96000;
target_bits_per_sample = 16;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);

```

Algoritmo para generar el archivo de 48 KHz y 16 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```

input_file = 'Hilda_192_24.wav';

output_file = 'Hilda_48_16.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);

```

```

% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 48000;
target_bits_per_sample = 16;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);

```

Algoritmo para generar el archivo de 44,1 KHz y 16 bits modificando los parámetros de configuración del archivo de la mezcla maestra Hilda_192_24.wav:

```

input_file = 'Hilda_192_24.wav';

output_file = 'Hilda_44_16.wav';
% Leer el archivo de audio original
[audio, sample_rate] = audioread(input_file);
% Especificar la frecuencia de muestreo objetivo y los bits por muestra
objetivo
target_sample_rate = 44100;
target_bits_per_sample = 16;
% Convertir el audio a los bits por muestra objetivo
audio = audio * (2^(target_bits_per_sample-1));
% Resamplear el audio a la frecuencia de muestreo objetivo
resampled_audio = resample(audio, target_sample_rate, sample_rate);
% Convertir el audio a los bits por muestra original
resampled_audio = resampled_audio / (2^(target_bits_per_sample-1));
% Guardar el audio resampleado en el archivo de salida
audiowrite(output_file, resampled_audio, target_sample_rate, 'BitsPerSample',
target_bits_per_sample);

```

13. Generados y grabados los archivos de audio con los cambios en la profundidad de bits y frecuencias de muestreo, se procede a identificar las propiedades de cada uno de los audios generados aplicando los algoritmos que se presentan en el anexo 4.

Tabla 11: Información obtenida de la señal de audio Hilda_96_24.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	96 KHz	
Profundidad de bits	24	
Número de canales	2	
Número de muestras	14496000	
Bitrate	4,61 Mbps	
Tamaño de archivo	86,98 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,40 dB	38,35 dB
Relación señal a ruido	143.21 dB	143.20 dB

Tabla 12: Información obtenida de la señal de audio Hilda_48_24.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	48 KHz	
Profundidad de bits	24	
Número de canales	2	
Número de muestras	7248000	
Bitrate	2,3 Mbps	
Tamaño de archivo	43,49 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,46 dB	38,25 dB
Relación señal a ruido	140.12 dB	140.10 dB

Tabla 13: Información obtenida de la señal de audio Hilda_44_24.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	44,1 KHz	
Profundidad de bits	24	
Número de canales	2	
Número de muestras	6659100	
Bitrate	2,12 Mbps	
Tamaño de archivo	39,95 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,43 dB	38,24 dB
Relación señal a ruido	138.3 dB	138.1 dB

Tabla 14: Información obtenida de la señal de audio Hilda_192_16.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	192 KHz	
Profundidad de bits	16	
Número de canales	2	
Número de muestras	28992000	
Bitrate	6,14 Mbps	
Tamaño de archivo	115,97 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,46 dB	38,25 dB
Relación señal a ruido	98.09 dB	98.08 dB

Tabla 15: Información obtenida de la señal de audio Hilda_96_16.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	96 KHz	
Profundidad de bits	16	
Número de canales	2	
Número de muestras	14496000	
Bitrate	3,07 Mbps	
Tamaño de archivo	57,98 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,47 dB	38,26 dB
Relación señal a ruido	95.07 dB	95.06 dB

Tabla 16: Información obtenida de la señal de audio Hilda_48_16.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	48 KHz	
Profundidad de bits	16	
Número de canales	2	
Número de muestras	7248000	
Bitrate	1,53 Mbps	
Tamaño de archivo	28,99 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,45 dB	38,25 dB
Relación señal a ruido	92.03 dB	92.01 dB

Tabla 17: Información obtenida de la señal de audio Hilda_44_16.wav

Propiedad	Canal Izquierdo	Canal Derecho
Formato del archivo	*.wav	
Duración del archivo	151 segundos	
Frecuencia de muestreo	44,1 KHz	
Profundidad de bits	16	
Número de canales	2	
Número de muestras	6659100	
Bitrate	1,41 Mbps	
Tamaño de archivo	26,64 MB	
Amplitud de pico:	0,00 dB	0,00 dB
Amplitud RMS total:	-8,20 dB	-7,85 dB
Desplazamiento de DC:	0,90 %	0,90 %
Rango dinámico:	38,43 dB	38,24 dB
Relación señal a ruido	90.02 dB	90.01 dB

IV. PRESENTACIÓN DE RESULTADOS

4.1. Análisis e interpretación de resultados

4.1.1. Características sonoras de instrumentos musicales

La investigación documental, ha permitido identificar las características de los instrumentos empleados en la producción digital.

- Los instrumentos de cuerda como las guitarras acústicas y tipo silent, generan niveles de presión sonora que alcanzan los 100 dB, sus frecuencias fundamentales se generan de 300 Hz a 1,2 KHz y sus frecuencias armónicas llegan hasta de 15, 6 KHz. El bajo acústico produce niveles de presión sonora hasta 100 dB y generan frecuencias desde 200 Hz a 700 Hz y sus frecuencias armónicas llegan hasta 5 KHz. El bajo electrónico produce niveles de presión sonora hasta 110 dB, sus frecuencias van desde los 90 Hz a 250 Hz y sus frecuencias armónicas llegan hasta 6,2 KHz.
- Los instrumentos electrónicos empleados en la producción musical fueron un piano electrónico que produce niveles de presión sonora hasta 100 dB y genera frecuencias de 125 Hz a 8 KHz y sus frecuencias armónicas llegan hasta 5 KHz; el órgano electrónico produce niveles de presión sonora hasta 110 dB y genera frecuencias desde 130 Hz a 800 Hz y sus frecuencias armónicas llegan hasta 15,5 KHz.
- Como instrumento de viento se usó al clarinete que produce niveles de presión sonora hasta 95 dB y genera frecuencias desde los 180 Hz a 1,2 KHz y sus frecuencias armónicas llegan hasta 11 KHz.
- Como instrumentos de percusión se emplearon cajones peruanos que producen niveles de presión hasta los 90 dB, generando frecuencias hasta 340 Hz y sus frecuencias armónicas llegan hasta 7 KHz. El hi hat producen niveles de presión hasta los 95 dB, generando frecuencias hasta 340 Hz y sus frecuencias armónicas llegan hasta 7 KHz. El bombo produce niveles de presión hasta los 150 dB, generando frecuencias hasta 150 Hz y sus frecuencias armónicas llegan hasta 6,5 KHz. Las castañuelas producen niveles de presión hasta los 90 dB, generando

frecuencias hasta 3 KHz y sus frecuencias armónicas llegan hasta 10 KHz.

Todos los instrumentos empleados permitieron generar una producción digital de calidad a 192 KHz de frecuencia de muestreo y una profundidad de 24 bits.

4.1.2. Producción digitalmente de una melodía multipista generada en estéreo

La producción musical se realizó con 14 instrumentos grabados en la consola digital Soundcraft modelo Ui24R en sesión multipista y generada en estéreo a 192 KHz de frecuencia de muestreo y una profundidad de 24 bits. Su reproducción permite apreciar una producción de alta calidad por la excelente interpretación musical de músicos de la Peña del Colegio de Ingenieros y equipamiento y micrófonos de alta gama.

4.1.3. Análisis de la calidad del audio producido al variar los parámetros de digitalización.

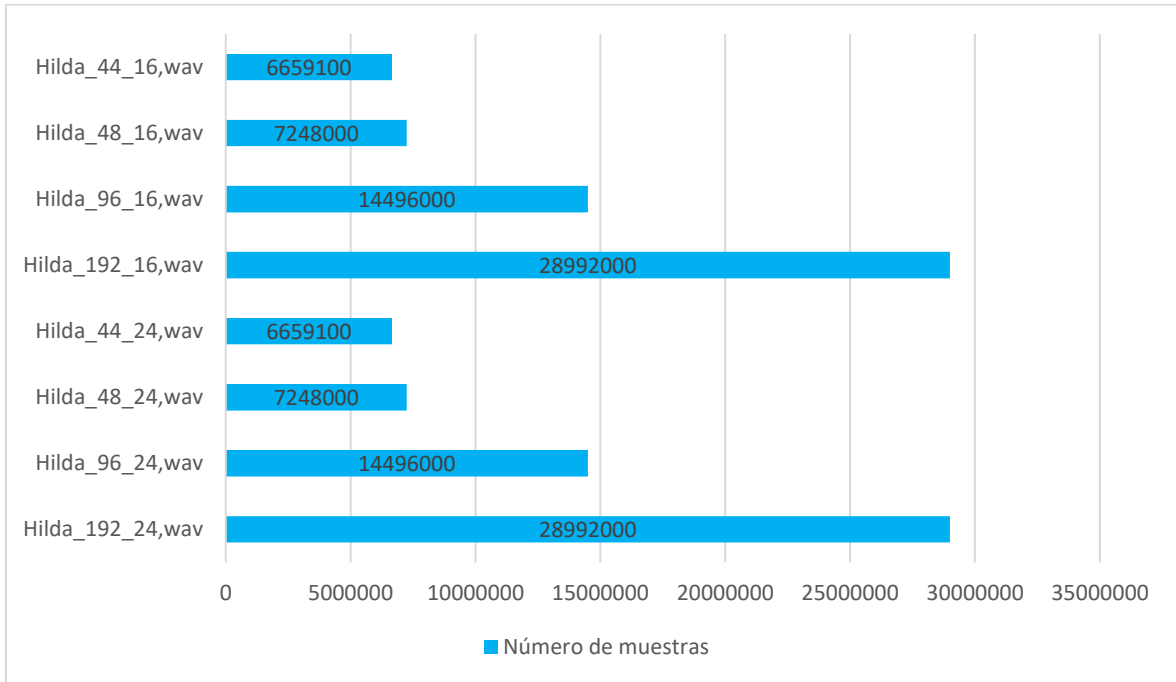
La tabla 18 y 19 muestran los valores que presentan los archivos de audio de los canales derecho e izquierdo generados.

Tabla 18: Valores de las características de los archivos de audio al variar los parámetros de digitalización

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24,wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_96_24,wav	14496000	4,61	86,98	38,40	38,35	143,21	143,2
Hilda_48_24,wav	7248000	2,3	43,49	38,46	38,25	140,12	140,1
Hilda_44_24,wav	6659100	2,12	39,95	38,43	38,24	138,3	138,1
Hilda_192_16,wav	28992000	6,14	115,97	38,46	38,25	98,09	98,08
Hilda_96_16,wav	14496000	3,07	57,98	38,47	38,26	95,07	95,06
Hilda_48_16,wav	7248000	1,53	28,99	38,45	38,25	92,03	92,01
Hilda_44_16,wav	6659100	1,41	26,64	38,43	38,24	90,02	90,01

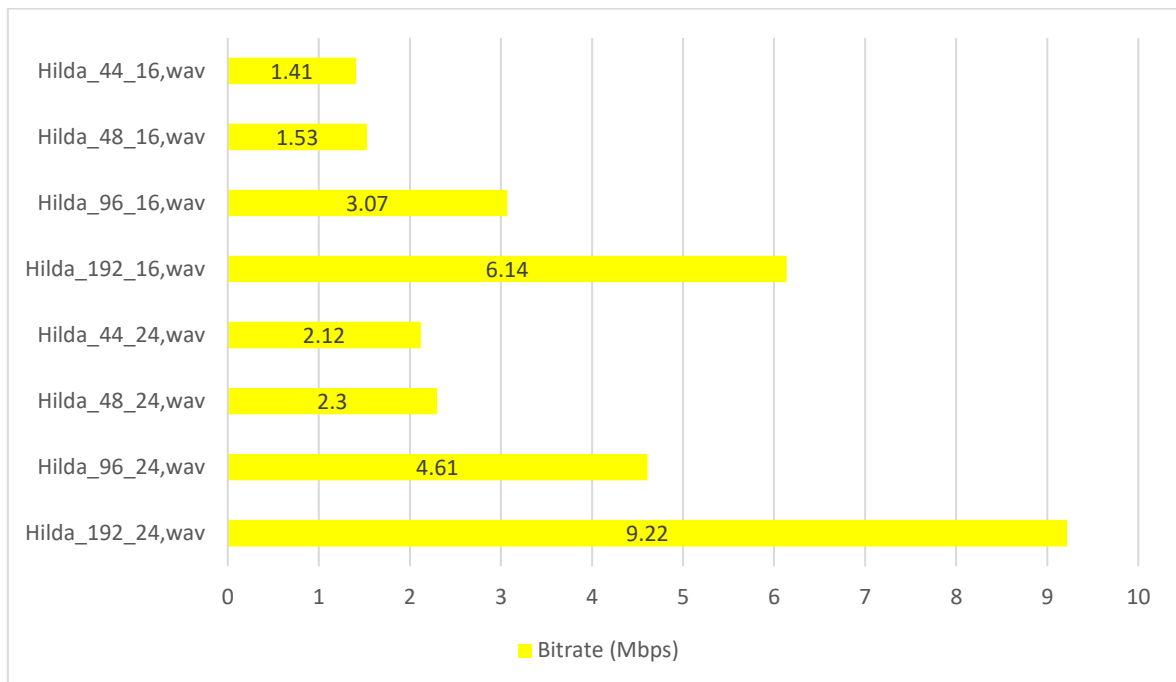
Para poder analizar mejor estos valores se muestran las siguientes figuras:

Figura 26: Número de muestras que presentan los archivos de audio.



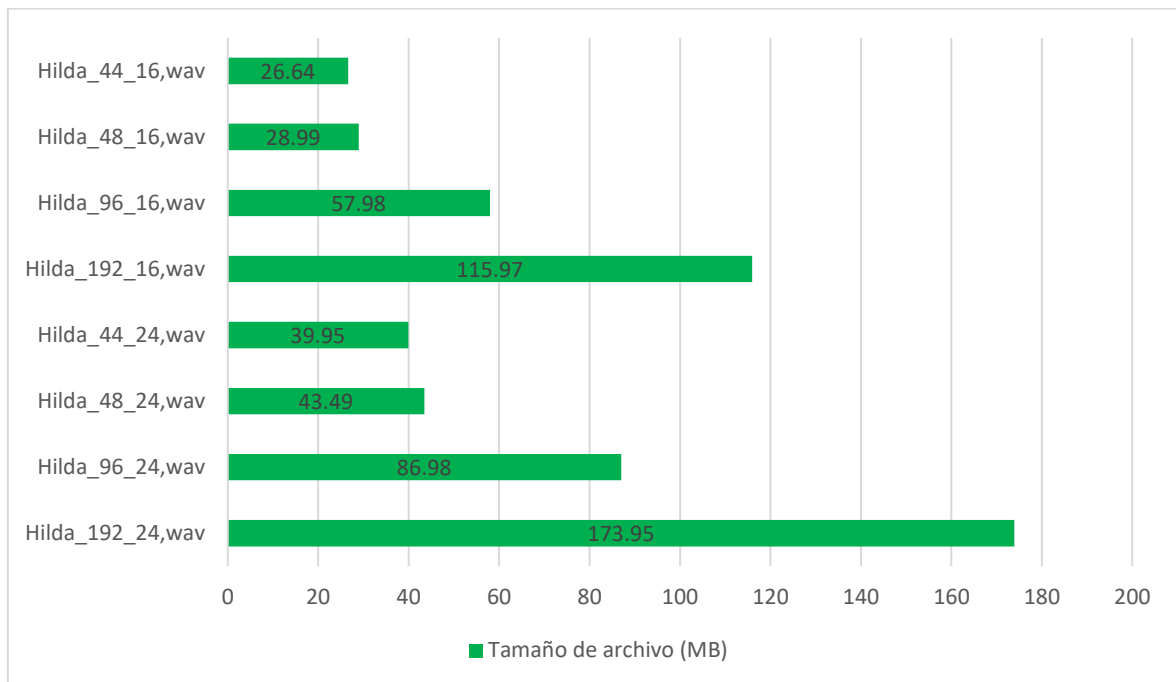
Ambos canales de audio de los archivos analizados presentan el mismo número de muestras, la figura muestra que al disminuir la frecuencia de muestreo el número de muestras disminuye, esta disminución no depende del número de bits. La calidad del audio se afecta al disminuir la frecuencia de muestreo, pero no es un factor preponderante porque las frecuencias de muestreo de los archivos de audio son mayores a 40 KHz que supera al índice de Nyquist mínimo de una señal de audio de 20 KHz.

Figura 27: Bitrate de los archivos de audio



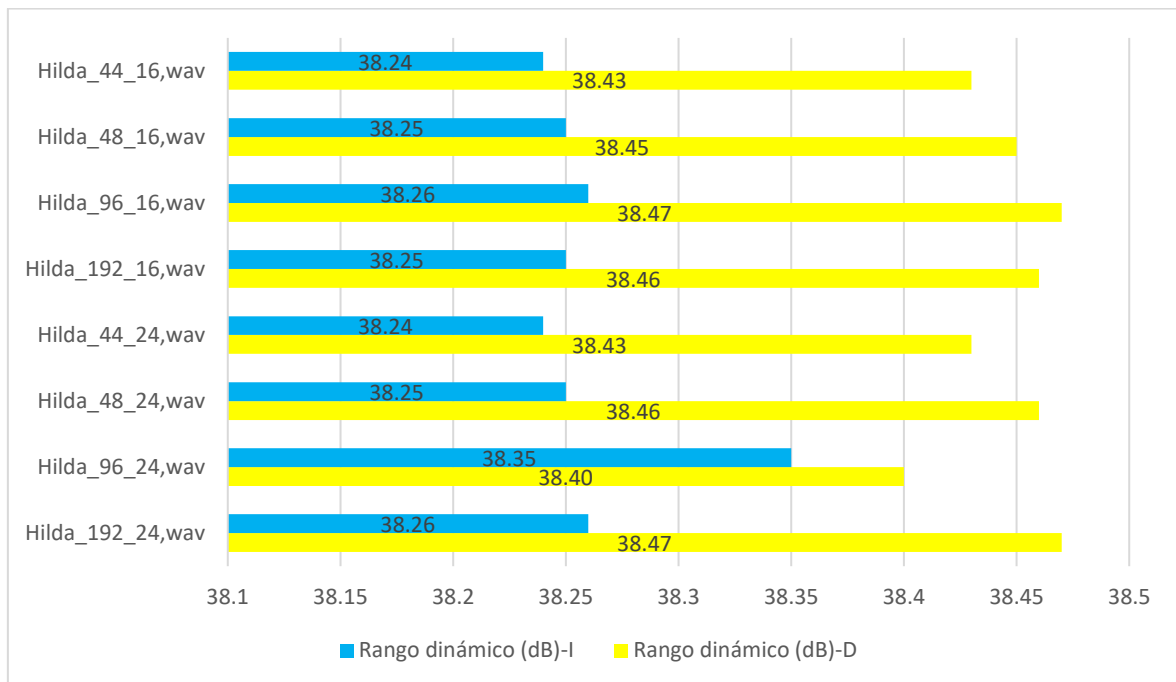
El bitrate es un parámetro que permite determinar el flujo de información que se transmitirá por un medio. Ambos canales de audio de los archivos analizados presentan el mismo bitrate, la figura muestra que al disminuir la frecuencia de muestreo y la profundidad de bits se afecta el bitrate. El valor del bitrate limita el medio donde se reproduce el archivo, un bitrate más alto permite una reproducción más nítida, con más detalles y menos pérdida de calidad. Un bitrate más alto permite una compresión menos agresiva y, por lo tanto, una menor pérdida de calidad durante la compresión.

Figura 28: Tamaño de los archivos de audio



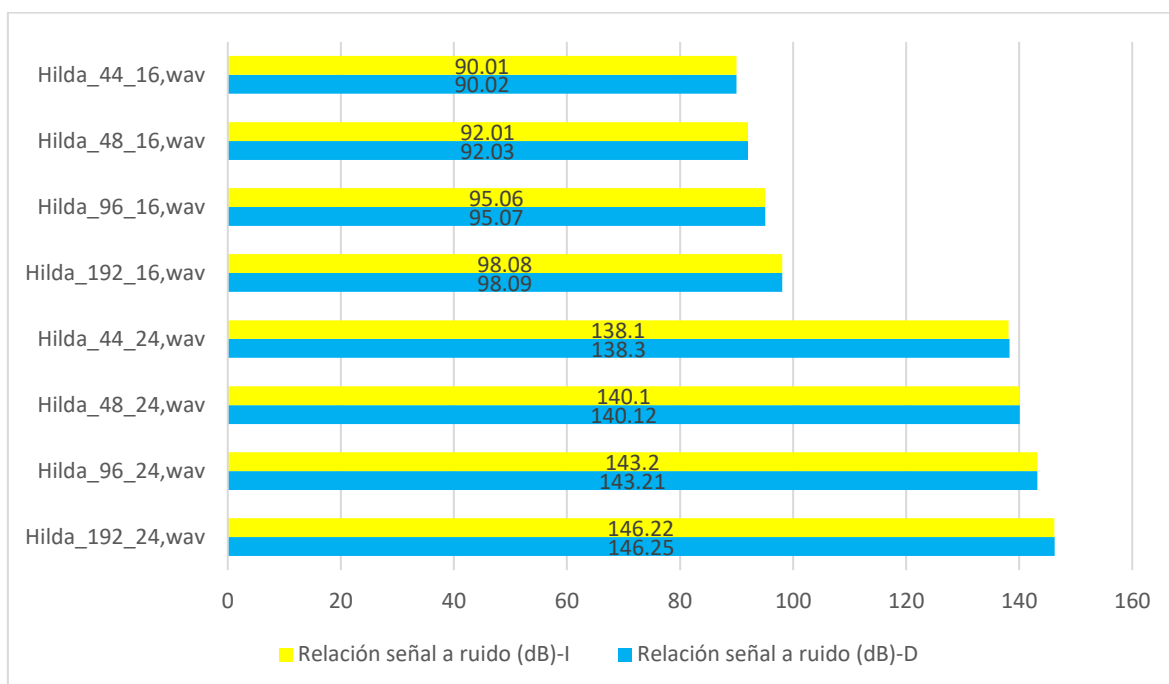
El tamaño de un archivo de audio no es independiente de si el audio es generado en estéreo, El tamaño del archivo depende directamente de la frecuencia de muestreo, la profundidad de bits, el número de canales de audio y la duración del archivo de audio dividido entre 8 expresado en Bytes. Como se aprecia en la figura, una menor frecuencia de muestreo y un menor número de bits generan un archivo más pequeño, pero afectan mucho la calidad del audio.

Figura 29: Rango dinámico de los archivos de audio



El rango dinámico se refiere a la diferencia entre el valor más alto y el más bajo que un sistema o dispositivo puede capturar o reproducir. En el contexto de los archivos de audio, el rango dinámico describe la diferencia entre los sonidos más suaves y los más fuertes en la grabación o interpretación musical. En la grabación de audio, el rango dinámico se mide en decibelios (dB) y representa la diferencia entre el nivel de señal más bajo (generalmente el ruido de fondo) y el nivel de señal más alto (generalmente el punto de distorsión). Un rango dinámico más amplio significa que el sistema o dispositivo puede capturar o reproducir una gama más amplia de niveles de volumen, lo que se traduce en una experiencia auditiva más rica y realista. Como se observa en la figura 28, la profundidad de bits y la frecuencia de muestreo no influyen significativamente en el rango dinámico ya que su variación está en torno a 1 dB al variar estos parámetros.

Figura 30: Relación señal a ruido de los archivos de audio



La relación señal a ruido se refiere a la relación entre la amplitud de la señal de audio deseada y el nivel de ruido presente en la grabación, se expresa típicamente en decibelios (dB) y se calcula dividiendo la potencia de la señal de interés por la potencia del ruido. Una relación señal a ruido más alta indica una mayor relación entre la señal y el ruido, lo que implica que la señal deseada es más fuerte en comparación con el ruido de fondo. En una grabación de audio, una relación señal a ruido más alta es deseable, ya que significa que la señal de audio deseada es más prominente en relación con el ruido de fondo no deseado (como el zumbido eléctrico o el ruido ambiental). Una relación señal a ruido más bajo provoca una calidad de audio deficiente, ya que el ruido puede interferir con la percepción y la comprensión de la señal de interés. Como se observa en la gráfica la frecuencia de muestreo no afectan mucho a la calidad del audio, pero la profundidad de bits si afecta significativamente porque el cambiar la profundidad de bits de 24 a 16 baja la relación señal a ruido en más de 48 dB afectando la calidad del audio.

Tabla 19: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 96 KHz con profundidad de 24 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_96_24.wav	14496000	4,61	86,98	38,4	38,35	143,21	143,2
Diferencia	14496000	4,61	86,97	0,07	-0,09	3,04	3,02
% de cambio	50,00%	50,00%	50,00%	0,18%	-0,24%	2,08%	2,07%

Las diferencias más significativas (50 %) son el número de muestras, el bitrate y el tamaño de archivo, los otros parámetros no tienen diferencia marcada.

Tabla 20: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 48 KHz con profundidad de 24 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_48_24.wav	7248000	2,3	43,49	38,46	38,25	140,12	140,1
Diferencia	21744000	6,92	130,46	0,01	0,01	6,13	6,12
% de cambio	75,00%	75,05%	75,00%	0,03%	0,03%	4,19%	4,19%

Las diferencias más significativas (75 %) son el número de muestras, el bitrate y el tamaño de archivo, los otros parámetros no tienen diferencia marcada.

Tabla 21: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 44,1 KHz con profundidad de 24 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_44_24.wav	6659100	2,12	39,95	38,43	38,24	138,3	138,1
Diferencia	22332900	7,1	134	0,04	0,02	7,95	8,12
% de cambio	77,03%	77,01%	77,03%	0,10%	0,05%	5,44%	5,55%

Las diferencias más significativas (alrededor del 77 %) son el número de muestras, el bitrate y el tamaño de archivo, los otros parámetros no tienen diferencia marcada.

Tabla 22: Comparación de parámetros entre los archivos de la mezcla maestra y el audio muestreado a 192 KHz con profundidad de 16 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_192_16.wav	28992000	6,14	115,97	38,46	38,25	98,09	98,08
Diferencia	0	3,08	57,98	0,01	0,01	48,16	48,14
% de cambio	0,00%	33,41%	33,33%	0,03%	0,03%	32,93%	32,92%

Las diferencias más significativas son el bitrate (33,41 %), el tamaño de archivo (33,33 %) y la relación señal a ruido (32,93 %), los otros parámetros no tienen diferencia marcada.

Tabla 23: Comparación de parámetros entre los archivos de la mezcla maestra y el audio muestreado a 96 KHz con profundidad de 16 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_96_16.wav	14496000	3,07	57,98	38,47	38,26	95,07	95,06
Diferencia	14496000	6,15	115,97	0	0	51,18	51,16
% de cambio	50,00%	66,70%	66,67%	0,00%	0,00%	34,99%	34,99%

Las diferencias más significativas son el bitrate y el tamaño de archivo (66,7 %), el número de muestras (50 %) y la relación señal a ruido (39,99 %), el rango dinámico no tiene diferencia.

Tabla 24: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 48 KHz con profundidad de 16 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_48_16.wav	7248000	1,53	28,99	38,45	38,25	92,03	92,01
Diferencia	21744000	7,69	144,96	0,02	0,01	54,22	54,21
% de cambio	75,00%	83,41%	83,33%	0,05%	0,03%	37,07%	37,07%

Las diferencias más significativas son el bitrate y el tamaño de archivo (más de 83 %), el número de muestras (75 %) y la relación señal a ruido (37 %), el rango dinámico no tiene diferencia marcada.

Tabla 25: Comparación de parámetros entre el los archivos de la mezcla maestra y el audio muestreado a 44,1 KHz con profundidad de 16 bits.

Nombre del Archivo	Número de muestras	Bitrate (Mbps)	Tamaño de archivo (MB)	Rango dinámico (dB)-D	Rango dinámico (dB)-I	Relación señal a ruido (dB)	Relación señal a ruido (dB)-I
Hilda_192_24.wav	28992000	9,22	173,95	38,47	38,26	146,25	146,22
Hilda_44_16.wav	6659100	1,41	26,64	38,43	38,24	90,02	90,01
Diferencia	22332900	7,81	147,31	0,04	0,02	56,23	56,21
% de cambio	77,03%	84,71%	84,69%	0,10%	0,05%	38,45%	38,44%

Las diferencias más significativas son el bitrate y el tamaño de archivo (cerca de 85 %), el número de muestras (77 %) y la relación señal a ruido (más de 38 %), el rango dinámico no tiene diferencia marcada.

4.2. Docimasia de hipótesis

O₁ X O₂

O₁: Medición de parámetros de la mezcla maestra

S/N de mezcla maestra: 146,22 dB

X: Modificación de frecuencia de muestreo y profundidad de bits

O₂: Medición de parámetros de la mezcla maestra después de las modificaciones

S/N (96 KHz y 24 bits) = 143,2 dB

S/N (48 KHz y 24 bits) = 140,1 dB

S/N (44,1 KHz y 24 bits) = 138,1 dB

S/N (192 KHz y 16 bits) = 98,08

S/N (96 KHz y 16 bits) = 95,06 dB

S/N (48 KHz y 16 bits) = 92,01 dB

S/N (44,1 KHz y 16 bits) = 90,01 dB

Los resultados obtenidos demuestran la hipótesis que la modificación de los valores de la frecuencia de muestreo y la profundidad de 24 bits influyen en la calidad de la producción musical de un audio digital en estéreo, siendo el factor más influyente la profundidad de bits.

V. DISCUSIÓN DE RESULTADOS

- Con respecto al objetivo 1, los diferentes instrumentos musicales usados en el presente trabajo de investigación, ya sean estos, de cuerda como las guitarras y bajos; de viento como el clarinete; de percusión, como el cajón, hi hat, bombo y castañuelas y los equipos electrónicos, como el órgano y piano, son fundamentales para la producción musical cuya interpretación permitió generar un hermoso vals criollo, concordando con el trabajo de Chiuro, Frontini, & Otero (2021) que las señales y la música son una relación inseparable.

- Con respecto al objetivo 2, se presenta un cambio revolucionario en la producción musical digital en la forma en que se crea, graba y produce música, concordamos con Pardo (2018) y evidenciamos que es factible producir digitalmente música de alta calidad sin necesidad de sofisticados estudios de grabación y equipos especializados, pero si se debe tener una adecuada selección de instrumentos musicales y equipamiento para registrar los sonidos como la consola digital Soundcraft Ui24R usada en el trabajo.
- Con respecto al objetivo 3, concordamos con Pérez (2018), con respecto al análisis de la calidad de audio mediante el empleo de algoritmos en Matlab. La investigación demuestra que, al disminuir los valores de la frecuencia de muestreo y la profundidad de bits, se afecta la calidad de la señal de audio. El factor más influyente es la profundidad de bits, pues al disminuir de 24 a 16 bits, la relación señal a ruido de la señal disminuye más de 48 dB lo que es una afectación grande en su calidad de un audio, aunque no sea tan fácilmente notorio. El estudio también determinó que un muestreo a 48 KHz y una profundidad de 24 bits es suficiente para producir una señal de muy alta calidad para el vals criollo Hilda alcanzando una relación señal a ruido de 140 dB y un tamaño de archivo de 43,49 Mbytes.

CONCLUSIONES

- Los instrumentos musicales generan niveles de presión sonora y frecuencias que cubren todo el espectro audible, dando lugar a melodías armoniosas. Son elementos esenciales en la producción musical digital, aportan creatividad, amplia variedad de sonidos, permiten la interacción en vivo, crean texturas y arreglos, y contribuyen a la identidad y estilo musical. La selección adecuada de los instrumentos musicales y el equipamiento, marca la diferencia en la calidad y el impacto de una producción musical, permitiendo a los músicos expresar su arte y conectarse de manera significativa con el público.
- La producción musical digital ha revolucionado la creación, grabación y producción al proporcionar acceso a herramientas avanzadas, flexibilidad creativa y colaboración en línea. La presente investigación a través de la producción musical digital del vals criollo Hilda, demostró que es posible producir en forma digital música de alta calidad sin requerir estudios de grabación costosos o equipos especializados.
- La producción de audios digitales de calidad codificado en el formato sin pérdidas *.wav, demostró que, la profundidad de bits es el parámetro más influyente en la calidad de los audios digitales, ya que, a mayor número de bits, menor será el error de cuantificación y mayor será la relación señal a ruido. El rango dinámico no se vió afectado por la variación de los parámetros de digitalización. Una frecuencia de muestreo más alta produce más muestras en la onda de audio y genera una señal digital más similar al sonido original, mientras que una frecuencia de muestreo por debajo del índice de Nyquist produce distorsiones. Un bitrate más alto proporciona una reproducción más clara y detallada, minimizando la pérdida de calidad si se comprime el audio. El tamaño del archivo de audio depende directamente de los parámetros de digitalización, cuantos más altos sean, más alto será el tamaño del archivo; el tamaño puede reducirse eligiendo una frecuencia de muestreo adecuada. La investigación evidencia que, un muestreo a 48 kHz y una profundidad de 24 bits son suficientes para lograr una producción musical digital de alta calidad. También es importante tener en cuenta que la calidad final de una producción musical digital depende del talento de los músicos, la técnica y el enfoque artístico del productor musical.

RECOMENDACIONES

- Se recomienda ampliar la investigación empleando diversos formatos con compresión a fin de evaluar el impacto de la compresión de ellos en la calidad.
- Se recomienda trabajar los archivos generados en un DAW (digital audio Workstation) empleando tasas de bits constantes (CBR) y tasas de bits variables (VBR) para determinar su influencia de la variabilidad de la tasa de bits y su efecto en el tamaño de los archivos.

REFERENCIAS BIBLIOGRÁFICAS

Agudo, J. (2022). *Estudio del reconocimiento de señales de audio mediante redes neuronales*. Barcelona: Universitat Politècnica de Catalunya.

Castán, P. (2017). *Desarrollo de una aplicación en Matlab para medidas de sistemas de sonido con múltiples entradas y múltiples salidas mediante señales sweep*. Valencia: Universitat Politècnica de Valencia.

Chiuro, C., Frontini, G., & Otero, F. (2021). *Señales y música, una relación inseparable. Una aplicación de la serie de Fourier con Matlab*. Libro de actas del XXII Encuentro Nacional y XIV Internacional de Educación Matemática en Carreras de Ingeniería, EMCI 2021, 317-323.

Jaramillo, A. (2007). *Acústica: La ciencia del sonido*. Medellín: Editorial Instituto Tecnológico Metropolitano.

Pardo, J. (2018). *Análisis tecnológico de un estudio de grabación*. Sevilla: Universidad de Sevilla.

Pérez, C. (2018). *Desarrollo de herramienta de análisis para la codificación y síntesis paramétrica de señales de voz y aplicación de efectos*. Gandia: Universitat Politècnica de Valencia.

Tabares, E. (2019). *Desarrollo de un algoritmo de medición y normalización con base en la norma ITU-R BS.1770-4 para el control de los niveles de audio en la televisión colombiana*. Medellín: Universidad de San Buenaventura Colombia.

Mandal, Mrinal Kr (2003). *Multimedia Signals and Systems*. 1ª Ed. United States of America. Editorial Kluwer Academic Publishers.

Pohlmann, Ken C. *Principles of Digital Audio*. 5ª Ed. New York. Editorial McGraw-Hill.

ANEXOS

ANEXO 1: INSTRUMENTOS DE INVESTIGACIÓN

Guía de análisis documental de instrumentos musicales

Objetivo del instrumento:

Obtener información relevante para identificar los niveles de presión sonora y las frecuencias que generan los instrumentos musicales que se usarán para la realización de la producción musical digital.

Procedimiento:

1. Se seleccionarán 14 instrumentos musicales que se emplearán para realizar la producción musical digital del vals peruano Hilda.
2. Se determinará la presión sonora que generan los instrumentos musicales seleccionados para la producción musical digital, a través del siguientes enlace: <https://invassat.gva.es/documents/161660384/167600130/AT-190302+Determinaci%C3%B3n+de+los+niveles+de+presi%C3%B3n+sonora+continua+equivalente+en+los+Conservatorios+Profesionales+de+M%C3%BAsica+de+la+Generalitat+Valenciana.pdf/5983a424-fb83-4de9-8cba-87350c081ba3?t=1580998100858>
3. Se identificarán las frecuencias que generan los instrumentos musicales seleccionados para la producción musical digital presentados en el anexo 2, a través del siguiente enlace: <http://www.hispasonic.com/revista/22>
4. Se identificarán los rangos de frecuencias de interpretación de los instrumentos musicales presentados en el anexo 3, a través del siguiente enlace: http://www.astormastering.com.ar/Tabla_de_frecuencias_de_instrumentos.pdf
5. Finalmente, con la información obtenida en los puntos anteriores, se completará la siguiente tabla:

Nº	Tipo de instrumento musical	Instrumento	Nivel de Presión sonora (dB _{SPL})	Frecuencias de interpretación	
				Frecuencia Fundamental (Hz)	Frecuencia de Armónicos máx (Hz)
1	Cuerda	Guitarra 1			
2	Cuerda	Guitarra 2			
3	Cuerda	Guitarra 3			
4	Cuerda	Guitarra 4			
5	Cuerda	Bajo acústico			
6	Cuerda	Bajo electrónico			
7	Cuerda	Piano			
8	Sintetizador	Órgano			
9	Viento	Clarinete			
10	Percusión	Cajón base			
11	Percusión	Cajón de relleno			
12	Percusión	Hi hat			
13	Percusión	Bombo			
14	Percusión	Castañuelas			

Guía de análisis documental de información relevante de micrófonos a ser usados en la producción musical

Objetivo del instrumento:

Obtener información relevante de micrófonos que se usarán para captar los sonidos de los instrumentos para la producción musical digital.

Procedimiento:

1. Se seleccionarán 4 tipos de micrófonos para captar el sonido de instrumentos de viento y percusión para realizar la producción musical digital del vals peruano Hilda.
2. Los datos técnicos relevantes de los micrófonos seleccionados serán obtenidos de la página web de la marca Shure a través del siguiente enlace: <https://www.shure.com/en-US/products/microphones/>
3. Con la información obtenida de las hojas técnicas, se completará la siguiente tabla:

Nº	Instrumento	Marca	Modelo	Tipo	Respuesta en frecuencia	Sensitividad	Patrón polar
1	Clarinete						
2	Cajón base						
3	Cajón de relleno						
4	Hi hat						

Guía de observación documental para obtener información de los archivos de audio generados a partir de la mezcla maestra modificando la profundidad de bits y la frecuencia de muestreo

Objetivo del instrumento:

Obtener información del archivo de audio de la mezcla maestra y de los archivos de audio generados a partir de la mezcla maestra modificando la profundidad de bits y la frecuencia de muestreo.

Procedimiento:

1. Se genera una rutina en Matlab para leer la información del archivo de audio de la mezcla maestra de la producción musical
2. Se generan rutinas en Matlab para modificar la frecuencia de muestreo y la profundidad de bits del archivo de la mezcla maestra.
3. Finalmente, con la información obtenida de los archivos de audio, se elaborará la siguiente tabla:

Característica	Canal Izquierdo	Canal Derecho
Amplitud de pico:		
Amplitud de pico real:		
Valor máximo de muestra:		
Valor mínimo de muestra:		
Amplitud RMS total:		
Amplitud RMS máxima:		
Amplitud RMS mínima:		
Amplitud RMS media:		
Desplazamiento de DC:		
Profundidad de bits		
Rango dinámico:		
Rango dinámico utilizado:		

Guía de análisis documental de instrumentos musicales

Objetivo del instrumento:

Obtener información relevante para identificar los niveles de presión sonora y las frecuencias que generan los instrumentos musicales que se usarán para la realización de la producción musical digital.

Procedimiento:

4. Se seleccionarán 14 instrumentos musicales que se emplearán para realizar la producción musical digital del vals peruano Hilda.
5. Se determinará la presión sonora que generan los instrumentos musicales seleccionados para la producción musical digital, a través del siguientes enlace:
<https://invassat.gva.es/documents/161660384/167600130/AT-190302+Determinaci%C3%B3n+de+los+niveles+de+presi%C3%B3n+sonora+continua+equivalente+en+los+Conservatorios+Profesionales+de+M%C3%BAsica+de+la+Generalitat+Valenciana.pdf/5983a424-fb83-4de9-8cba-87350c081ba3?t=1580998100858>
6. Se identificarán las frecuencias que generan los instrumentos musicales seleccionados para la producción musical digital presentados en el anexo 2, a través del siguiente enlace:
<http://www.hispasonic.com/revista/22>
7. Se identificarán los rangos de frecuencias de interpretación de los instrumentos musicales presentados en el anexo 3, a través del siguiente enlace:
http://www.astormastering.com.ar/Tabla_de_frecuencias_de_instrumentos.pdf
8. Finalmente, con la información obtenida en los puntos anteriores, se completará la siguiente tabla:

Nº	Tipo de instrumento musical	Instrumento	Nivel de Presión sonora (dB _{SPL})	Frecuencias de interpretación	
				Frecuencia Fundamental (Hz)	Frecuencia de Armónicos máx (Hz)
1	Cuerda	Guitarra 1			
2	Cuerda	Guitarra 2			
3	Cuerda	Guitarra 3			
4	Cuerda	Guitarra 4			
5	Cuerda	Bajo acústico			
6	Cuerda	Bajo electrónico			
7	Electrónico	Piano			
8	Electrónico	Órgano			
9	Viento	Clarinete			
10	Percusión	Cajón base			
11	Percusión	Cajón de relleno			
12	Percusión	Hi hat			
13	Percusión	Bombo			
14	Percusión	Castañuelas			

ANEXO 2: Rangos de frecuencias de los instrumentos musicales

Instrumento	Frecuencia Fundamental (Hz)	Frecuencia de Armónicos
Flauta	261-2349	3-8 KHz
Oboe	261-1568	2-12 KHz
Clarinete	165-1568	2-10 KHz
Fagot	62-587	1-7 KHz
Trompeta	165-988	1-7.5 KHz
Trombón	73-587	1-4 KHz
Tuba	49-587	1-4 KHz
Tambor	100-200	1-20 KHz
Bombo	30-147	1-6 KHz
Platillos	300-587	1-15 KHz
Violín	196-3136	4-15 KHz
Viola	131-1175	2-8.5 KHz
Cello	65-698	1-6.5 KHz
Bajo acústico	41-294	1-5KHz
Bajo eléctrico	41-300	1-7 KHz
Guitarra acústica	82-988	1-15 KHz
Guitarra eléctrica (amplif.)	82-1319	1-3.5 KHz
Guitarra eléctrica (directa)	82-1319	1-15 KHz
Piano	28-4196	5-8 KHz
Saxo Soprano	247-1175	2-12 KHz
Saxo alto	175-698	2-12 KHz
Saxo tenor	131-494	1-12 KHz
Cantante	87-392	1-12 KHz

La interpretación musical de los instrumentos presenta algunas características de acuerdo como se interprete, estas son: golpe, ataque, palma, cuerpo, brillo, incisivo, parche, aire, caja, madera, despegue, peso, claridad, macillos, rugosidad y para las voces: cuerpo, resonancia, nasalidad, presencia y sibilancia.

<http://www.hispasonic.com/revista/22>

ANEXO 3: Frecuencias empleadas por los instrumentos de acuerdo a su interpretación.

Instrumento	Frecuencias
Bombo rock	Golpe entre 60-80 Hz / Ataque 3 KHz
Bombo pop	Golpe sobre los 100 Hz / Ataque 2,5 KHz
Cajón Peruano	Golpe 80 Hz / Resonancia 300 Hz / Ataque y palma 3-5 KHz
Tambor	Cuerpo 190-230 Hz/ Despegue 2,5 KHz / Parche 5 KHz
Toms	Cuerpo 160-240 Hz / Ataque 5 KHz
Tom de piso	Cuerpo 80-120 Hz / Ataque 5 KHz
Hi-hat	Cuerpo 200 Hz / Brillo por sobre 8 KHz / Incisivo 10 KHz
Platillos	Cuerpo 200 Hz / Brillo sobre 6,5 KHz
Congas	Cuerpo 180-250 Hz / Parche 5 KHz
Bongó	Cuerpo 200-250 Hz / Parche 4-6 kHz
Udu	Aire 60 Hz 7 Cuerpo 125 Hz / Brillo 4 KHz
Bajo acústico	Cuerpo 80 Hz / Caja 200 Hz / Madera 750 Hz / Brillo 3 KHz
Bajo electrónico	Cuerpo 60-80 Hz / Dedos y madera 650-900 Hz / Brillo 3 KHz
Guitarra acústica	Peso 100 Hz / Cuerpo 250 Hz / Brillo 3-6 KHz
Guitarra Eléctrica	Cuerpo 220 Hz / Madera 650 Hz / Despegue 2,5 KHz
Cuerdas	Peso 100 Hz / Cuerpo 200 Hz / Claridad 2 KHz / Filo 10 KHz
Piano acústico	Peso 80-125 Hz / Claridad 2-4 KHz / Macillos 7,5 KHz
Piano eléctrico	Peso 80-125 Hz / Cuerpo 250 Hz / Claridad 2-4 KHz
Bronces	Cuerpo 125-250 Hz / Brillo 5-8 KHz
Bandoneón	Cuerpo 150-225 Hz / Presencia 1,5 KHz / Rugosidad 6 KHz
Voces	Cuerpo 125 Hz / Resonancia 200-300 Hz / Nasalidad 800-1200 Hz / Presencia 3-5 KHz / Sibilancia 6,8-10 KHz

En base a la información proporcionada en las tablas N° 16 y N° 17, y evaluando espectralmente cada uno de los instrumentos, se elabora la tabla N° 18, que indica las frecuencias máximas de los instrumentos usados en el desarrollo del presente trabajo.

En la figura N° 44, se muestran los espectros de frecuencia generados por cada instrumento musical empleado en el trabajo, el espectro se ha obtenido mediante un programa desarrollado en el programa matlab que lee los archivos almacenados en la computadora en formato wav.

Fuente: http://www.astormastering.com.ar/Tabla_de_frecuencias_de_instrumentos.pdf

ANEXO 4: Algoritmos en Matlab para obtener información de los archivos de audio

ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_96_24.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_96_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_96_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_96_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
```

```
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_96_24.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_96_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);
```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_96_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);
```


ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_48_24.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_48_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_48_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_48_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_48_24.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_48_24.wav');
```

```

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_48_24.wav

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```

ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_44_24.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_44_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_44_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_44_24.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_44_24.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_44_24.wav');
```

```

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_44_24.wav

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_24.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```

ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_192_16.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_192_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_192_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_192_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_192_16.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_192_16.wav');
```

```

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_192_16.wav

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_192_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```

ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_96_16.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_96_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_96_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_96_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_96_16.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_96_16.wav');
```

```

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_96_16.wav

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_96_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```


ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_48_16.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_48_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_48_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_48_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_48_16.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_48_16.wav');
```

```

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_48_16.wav

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_48_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```

ALGORITMOS PARA OBTENER INFORMACIÓN DEL ARCHIVO DE AUDIO Hilda_44_16.wav

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_44_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud pico para cada canal
amplitudPicoIzquierdo = max(abs(canalIzquierdo));
amplitudPicoDerecho = max(abs(canalDerecho));
% Mostrar los resultados
fprintf('Amplitud pico del canal izquierdo: %f\n', amplitudPicoIzquierdo);
fprintf('Amplitud pico del canal derecho: %f\n', amplitudPicoDerecho);
```

Algoritmo para determinar amplitud RMS total de la señal de audio Hilda_44_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la amplitud RMS para cada canal
amplitudRMSIzquierdo = rms(canalIzquierdo);
amplitudRMSDerecho = rms(canalDerecho);
% Calcular la amplitud RMS total
amplitudRMSTotal = rms(audio(:));
% Mostrar los resultados
fprintf('Amplitud RMS del canal izquierdo: %f\n', amplitudRMSIzquierdo);
fprintf('Amplitud RMS del canal derecho: %f\n', amplitudRMSDerecho);
```

Algoritmo para determinar el desplazamiento DC de la señal de audio Hilda_44_16.wav

```
% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el desplazamiento DC para cada canal
desplazamientoDCIzquierdo = mean(canalIzquierdo);
desplazamientoDCDerecho = mean(canalDerecho);
% Mostrar los resultados
fprintf('Desplazamiento DC del canal izquierdo: %f\n',
desplazamientoDCIzquierdo);
fprintf('Desplazamiento DC del canal derecho: %f\n', desplazamientoDCDerecho);
```

Algoritmo para determinar el rango dinámico de la señal de audio Hilda_44_16.wav

```
% Leer el archivo de audio
[audio, ~] = audioread('Hilda_44_16.wav');
```

```

% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular el rango dinámico para cada canal
rangoDinamicoIzquierdo = max(canalIzquierdo) - min(canalIzquierdo);
rangoDinamicoDerecho = max(canalDerecho) - min(canalDerecho);
% Calcular el rango dinámico total
rangoDinamicoTotal = max(audio(:)) - min(audio(:));
% Mostrar los resultados
fprintf('Rango dinámico del canal izquierdo: %.2f\n', rangoDinamicoIzquierdo);
fprintf('Rango dinámico del canal derecho: %.2f\n', rangoDinamicoDerecho);

```

Algoritmo para determinar la relación señal a ruido de la señal de audio Hilda_44_16.wav

```

% Leer el archivo de audio
[audio, frecuenciaMuestreo] = audioread('Hilda_44_16.wav');
% Obtener los canales izquierdo y derecho
canalIzquierdo = audio(:, 1);
canalDerecho = audio(:, 2);
% Calcular la potencia de la señal
potenciaSenalIzquierdo = sum(canalIzquierdo.^2) / length(canalIzquierdo);
potenciaSenalDerecho = sum(canalDerecho.^2) / length(canalDerecho);
% Calcular la potencia del ruido
ruidoIzquierdo = canalIzquierdo - mean(canalIzquierdo);
ruidoDerecho = canalDerecho - mean(canalDerecho);
potenciaRuidoIzquierdo = sum(ruidoIzquierdo.^2) / length(ruidoIzquierdo);
potenciaRuidoDerecho = sum(ruidoDerecho.^2) / length(ruidoDerecho);
% Calcular la relación señal-ruido (SNR)
snrIzquierdo = 10 * log10(potenciaSenalIzquierdo / potenciaRuidoIzquierdo);
snrDerecho = 10 * log10(potenciaSenalDerecho / potenciaRuidoDerecho);
% Mostrar los resultados
fprintf('Relación señal-ruido (SNR) del canal izquierdo: %.2f dB\n',
snrIzquierdo);
fprintf('Relación señal-ruido (SNR) del canal derecho: %.2f dB\n', snrDerecho);

```