



TECNOLÓGICO
NACIONAL DE MÉXICO®



INSTITUTO TECNOLÓGICO SUPERIOR DE TEZIUTLÁN

Tesis



“Diseño de algoritmo de procesamiento de señales
electroacústicas resultantes del monitoreo del aparato
respiratorio”

PRESENTA:

JORGE LUIS URIBE PEÑA

CON NÚMERO DE CONTROL
19TE0381

PARA OBTENER EL TÍTULO DE:
INGENIERO MECATRÓNICO

CLAVE DEL PROGRAMA ACADÉMICO
IMCT-2010-229

DIRECTOR (A) DE TESIS:
M.S.C. Gabriel Angel Ramírez Vicente

“La Juventud de hoy, Tecnología del Mañana”

TEZIUTLÁN, PUEBLA, FEBRERO 2024



Preliminares

Agradecimientos

A mi madre

“La persona que principalmente le agradezco de toda mi familia la cual a estado siempre a mi lado y me ha apoyado incondicionalmente en todo lo que e hecho y no he hecho, mi madre siempre ha sido mi modelo a seguir para poder crecer como persona debido a todos sus valores que me a inculcado igualmente, al igual que es la persona que más admiro en toda mi vida”

A mi tía

“Al igual quiero agradecerle a mi tía la cual a estado conmigo igualmente desde pequeño y me a apoyado en todas las situaciones complicadas y que también le tengo mucho cariño por las motivaciones que me llega a dar para seguir adelante”

Resumen

En el presente trabajo se propone el diseño de un sistema de recepción de datos que sea capaz de ser almacenados para su visualización y/o su envío post procesamiento a un especialista y de esta manera agilizar el diagnóstico preciso disminuyendo tiempos y sin generar esfuerzos sobre los pacientes.

Se obtuvieron varios puntos para ser probados y asegurar que sean útiles para la finalidad de proyecto, esto para guardar datos adecuados y que sean de utilidad para el especialista que está atendiendo al paciente o simplemente para tener información que se pueda analizar posteriormente. Al igual se puede visualizar cada valor muestreados por sensores de sonido posicionados en el pecho del paciente mediante gráficas y guardado en archivos de audio

Se pudo obtener un resultado adecuado capaz de llegar a agilizar la obtención de datos por el método conocido como auscultación pulmonar y obtener datos útiles que pueden ser guardados y reproducidos posteriormente en cualquier momento para que puedan ser analizados adecuadamente por un especialista, pudiendo esto ayudar a personas con enfermedades respiratorias crónicas que lleguen a tener dificultades al momento de recibir atención medica

Introducción

Estos últimos años se pudieron apreciar grandes cambios en la sociedad, uno de los problemas que más se notaron en estos años son enfermedades de todo tipo, ya sea desde una simple gripa hasta el cáncer, los cuales han ido evolucionando con el paso de los años, una de las que puede llegar a afectar a un mayor número de individuos son las enfermedades respiratorias de las cuales hay una rama especializada de la medicina la cual es la neumología.

Muchas de estas enfermedades llegan a ser pasajeras, pero también pueden llegar a agravarse a tal punto que pueden provocar problemas muchos mayores como dolores de pecho recurrente, dificultades para respirar o hasta una la misma muerte de la persona.

Al igual que este tipo de enfermedades pueden llegar a afectar no solo a adultos, sino que también a niños llegando a provocar los mismos síntomas sobre todo por lo que paso en estos recientes años con la llegada del COVID 19 el cual provoco que muchas enfermedades respiratorias se agravaran aún más, como se ha visto en estos últimos tiempos con la pandemia, la cual provoco grandes problemas en la sociedad por si sola.

Para prevenir estos tipos de enfermedades se propuso realizar un sistema capaz de recolectar datos los cuales puedan ser utilizados posteriormente o puedan enviarse de manera sencilla al especialista que los utilizara para poder dar un diagnóstico posterior.

La finalidad de la lectura de los pulmones es poder facilitar y poder llegar a detectar enfermedades más graves de una manera más sencilla al igual que rápida, también debido a que muchas personas ya son afectadas por dichas enfermedades, tienen distintas dificultades y al seguir un tratamiento deben recibir un chequeo médico en el cual llegan a tener problemas, tal es el caso de las personas que llegan a tener complicaciones al respirar.

Para poder ayudar a la detección de este tipo de enfermedades se decidió ir por el camino de la auscultación pulmonar mediante un tipo de estetoscopio digital en diferentes zonas del cuerpo las cuales sean capaces de ser útiles a la hora de recabar información para dar un posterior diagnóstico por un especialista

Índice

Preliminares	2
Agradecimientos	3
Resumen.....	4
Introducción	5
Índice 7	
1.1 Descripción de la empresa u organización.....	11
1.1.1 Antecedentes	11
1.2 Problema	12
1.3 Preguntas de investigación.....	14
1.4 Objetivos	14
1.4.1 Objetivos generales	14
1.4.2 Objetivos específicos.....	14
1.5 Justificación de la investigación	15
2.1 Fundamentos teóricos	18
2.1.1 ¿Qué es el sonido?	18
2.1.2 Medición del Sonido.....	19
2.1.3 El Ruido.....	19
2.1.4 La calidad del sonido	19
2.1.5 ¿Cómo medir la Frecuencia del Sonido?	20
2.1.6 El procesamiento de señales ¿Qué es?	23
2.1.7 Comparación.....	24
2.1.8 Procesamiento de señales	24
2.1.9 ¿Cómo se realiza el procesamiento de la señal?	24
2.1.10 Tipos de Procesamiento de Señales.....	25
2.1.11 Filtración	25
2.1.12 Filtros.....	26
2.1.13 ¿Qué es mejor, el filtrado analógico o el digital?	27
2.1.14 Filtro digital.....	27
2.1.15 Filtrado IIR versus FIR	28
2.1.16 Análisis FFT.....	29
2.1.17 ¿Qué es el HIFI?.....	31

2.1.18	Conceptos a tener en cuenta con respecto al audio HIFI	31
2.1.18.1	Amplificador.....	31
2.1.18.2	Altavoces.....	31
2.1.18.3	Códec.....	32
2.1.19	Que es un convertidor analógico a digital.....	32
2.1.20	Proceso para la conversión	33
2.1.20.1	Muestreo:	33
2.1.20.2	Retención:	33
2.1.20.3	Cuantificación:	33
2.1.20.4	Codificación:	34
2.1.21	Tipos de convertidores Analógicos a Digitales.....	34
2.1.21.1	Flash ADC.....	34
2.1.21.2	Convertidor analógico a digital de extensión de tiempo (TS –.....	34
2.1.21.3	Convertidor analógico a digital Sigma-delta ($\Sigma\Delta$).....	34
2.1.21.4	Aplicaciones de los convertidores analógico digital	34
2.1.22	Procesamiento de señales digitales.....	35
2.1.23	Instrumentos científicos	35
2.1.24	Procesamiento de audio.....	35
3.1	Procedimiento y descripción de las actividades realizadas	37
3.1.1	Estado del arte.....	38
3.1.1.1	Procesamiento de señales	38
3.1.1.2	Filtros.....	38
3.1.1.3	Circuitos amplificadores.....	38
3.1.2	Selección del circuito amplificador	39
3.1.3	Verificación del circuito.....	40
3.1.4	Selección del receptor de señales.....	40
3.1.5	Pruebas de recepción de señales.....	42
3.1.6	Visión de las señales recibidas	43
3.1.7	Pruebas de procesado de señales.....	44
3.1.8	Realización completa del procesado de señales.....	45
3.1.9	Guardado de señales ya procesadas	46
3.1.10	Verificación de resultados	46
3.1.11	Revisiones preentrega final.....	47

3.1.12	Entrega del prototipo	47
3.2	Alcance y enfoque de la investigación	48
3.3	Hipótesis.....	48
3.4	Diseño y metodología de la investigación.....	48
4.1	Resultados	51
5.1	Conclusiones del proyecto.....	60
5.2	Conclusiones relativas a los objetivos específicos.....	60
5.3	Conclusiones relativas al objetivo general	61
5.4	Limitaciones del modelo planteado.....	61
5.5	Recomendaciones	61
6.1	Competencias desarrolladas y/o aplicadas.....	64
7.1	Fuentes de Información.....	67
	Bibliografía.....	67
8.1.1	Índice de figuras	80
8.1.2	Índice de Tabla	81

Capítulo I Generalidades del proyecto

1.1 Descripción de la empresa u organización

1.1.1 Antecedentes

El primer día del mes de septiembre de 1993 inició actividades el Instituto ofreciendo las carreras de Ingeniería Industrial y Licenciatura en Administración, siendo el primer Tecnológico Descentralizado del Estado de Puebla, junto con su similar de la Sierra Norte, designándose como primer director general a José Emilio Guillermo Ortega Balbuena. Las primeras actividades académicas se desarrollaron en el “Centro de Bachillerato Tecnológico, Industrial y de Servicios No. 44”, el cual resultó insuficiente ante la aceptación de los estudiantes; por lo que apenas un semestre después el Instituto se trasladó a una granja avícola y la casa anexa.

Es por ello que el Instituto asume el compromiso de certificarse a través de la Norma ISO 9001-2008 y es en el mes de abril del año 2006 cuando esta casa de estudios recibe orgullosamente y además con distinción, la certificación por parte de la empresa QMI-SAIGLOBAL, quien la certifica como una Institución de Calidad en su proceso de Enseñanza – Aprendizaje.

El Instituto Tecnológico Superior de Teziutlán, atento a las demandas de la sociedad, y a los principios de la Ley de Educación del Estado de Puebla, se consolida como una Institución cuyo objetivo es lograr una educación de calidad, moderna y eficaz, orientada al servicio, acercándola a las necesidades e intereses de la población, que promueva el uso transparente y eficiente de los recursos humanos, materiales y financieros de que disponga, y que cumpla puntualmente con sus programas de trabajo

1.2 Problema

Las infecciones respiratorias agudas, dentro de las cuales se encuentran la neumonía, influenza, según la secretaria de salud, representa la principal causa de morbilidad en el mundo y la causa más frecuente de utilización de los servicios de salud en todos los países. “En México constituyen un problema de salud prioritario por su alta morbilidad y mortalidad. Son la causa más frecuente de enfermedad tanto en niños como en adultos siendo estos últimos del 22 %” (GOB, 2021)

“Según el reporte preliminar Características de las defunciones registradas en México durante 2020, emitido el 29 de julio por el Instituto Nacional de Estadística y Geografía, del millón 86 mil 94 defunciones registradas durante ese año, 21 mil 972 personas fallecieron por lo que categoriza como enfermedades pulmonares obstructivas crónicas, las cuales se posicionan entre las 10 primeras causas en gente que fallece a partir de los 55 años. (Frías, 2021)

“Dentro de las IRAG (Infecciones Respiratorias Agudas Graves) los padecimientos con mayores tasas de incidencia y mortalidad se encuentran las neumonías adquiridas en la comunidad, la influenza” (GOB, 2021), la enfermedad pulmonar obstructiva crónica (EPOC), el asma, cáncer pulmonar y más recientemente el COVID-19.

También se debe tener en cuenta que el reciente COVID 19 hizo que varias enfermedades ya existentes se agravaran aún más teniendo que enfermedades que pueden no ser tan mortales llegan a ser mucho más peligrosas tales como el caso de la enfermedad pulmonar obstructiva crónica (EPOC) que se vio un agravante mayor debido a la combinación de estas enfermedades

Al ser las IRAG enfermedades altamente contagiosas y recientemente muy agravadas debido al reciente COVID 19 provoco que sus diagnósticos y monitoreos fuesen una actividad de alto riesgo para las personas pertenecientes al sector salud, así como, a los familiares de los pacientes que estuvieron en cuarentena en su domicilio. Esto género que las salas de diagnóstico y monitoreo se vieran saturadas. Los diagnósticos médicos de esta enfermedad por su naturaleza debían

ser realizados por médicos especialistas en la rama de Neumología lo que también supero al personal en los centros hospitalarios.

Dentro de la Neumología más específicamente en área de Auscultación pulmonar existen pacientes con problemas respiratorios muy agravados los cuales les dificulta la correcta obtención de diagnósticos por la dificultad que tienen al respirar las personas, por eso se decidió utilizar un medio para obtener sonidos los cuales guardar para su posterior reproducción

Así mismo, los tiempos de espera para las consultas y las condiciones de salud de los pacientes pueden provocar complicaciones de salud en los pacientes pudiendo provocar daños irreversibles e incluso la muerte.

Las afectaciones mencionadas anteriormente podrían ser disminuidas con la implementación de la tecnología para solventar las necesidades de diagnóstico médico, es decir, incorporar un sistema de adquisición de información que monitore el estado de los pacientes con enfermedades crónicas o altamente contagiosas y permita al médico especialistas realizar un diagnóstico no presencial, pudiendo realizar recomendaciones más rápidas y acertadas debido a que los tiempo de diagnóstico y consulta disminuirían.

1.3 Preguntas de investigación

¿Se puede captar y almacenar audios de sistema respiratorio para realizar un diagnóstico no presencial?

¿Es posible realizar la lectura de sonidos provenientes del sistema respiratorio mediante micrófonos de manera constante y en cualquier lugar a cualquier momento sin ningún tipo de percance y con la facilidad para cualquier tipo de persona?

1.4 Objetivos

1.4.1 Objetivos generales

Diseñar e implementar un algoritmo que permita realizar el procesamiento, análisis y reproducción de señales acústicas obtenidas del muestreo del aparato respiratorio para la obtención de un futuro diagnóstico

1.4.2 Objetivos específicos

- Realizar el estado del arte referente a algoritmos de procesamiento de señales de audio.
- Determinar el dispositivo de adquisición de audio y su sistema de filtraje.
- Determinar la plataforma tecnológica óptima para el desarrollo del algoritmo.
- Desarrollar el algoritmo para el procesamiento y reproducción de audio.
- Determinar el protocolo/formato de guardado de audio.
- Realizar pruebas de fidelidad de audio

1.5 Justificación de la investigación

Desde hace años se han visto una cantidad de enfermedades las cuales afectan a las vías respiratorias y no solo son en personas jóvenes o en adultos si no que atacan de manera indistinta. Al igual, estas infecciones se van agravando cada vez más si no se recibe una correcta atención, yendo desde una dificultad al respirar hasta a tal punto de la muerte, estas enfermedades tales como la neumonía, la enfermedad pulmonar obstructiva crónica (EPOC), entre otras, son las causantes de todos estos descensos. Al igual que con la reciente enfermedad del COVID 19 se ha visto afectado el sector salud con respecto a la cantidad de pacientes los cuales, enfermeros y médicos llegan a ser superados por un número considerable de enfermos con padecimientos respiratorios, no solo eso si no que también pueden llegar a contagiarse debido al contacto prolongado con los contagiados.

Con la llegada del COVID 19, muchos enfermeros y médicos pudieron darse cuenta que la enfermedad llego a agravar enfermedades de las vías respiratorias ya existentes tales como la neumonía, el cáncer pulmonar, entre otras. Debido a esto los especialistas necesitan recopilar datos acústicos de sus distintos pacientes para poder precisar qué tipo de enfermedad pueden llegar a contraer las personas, pero esto causa una molestia para personas con enfermedades demasiado avanzadas, sobre todo porque llegan a tener dificultades al respirar. Con base en lo anterior se toma en cuenta la idea de una manera de guardar diferentes tipos de datos auditivos con la finalidad de mejorar la calidad de vida de los pacientes para que no se les dificulte aún más los ejercicios necesarios al momento de las consultas.

En algunos casos en los que el paciente no es capaz de realizar acciones como el levantarse de una cama o el simple hecho de respirar puede ver afectado al especialista debido a que es difícil realizar un diagnóstico para eso tenemos las tecnologías de comunicación, estas nos ayudarían a transmitir sonidos electroacústicos recibidos del sistema respiratorio, no solo se trataría de la transferencia, sino que también se guardarían estos datos para su posterior

análisis. Al igual que se podría llegar a tal punto en el cual dichos datos puedan ser analizados por una inteligencia artificial para dar un diagnóstico oportuno para el paciente o también alertar de alguna operación necesaria. También se puede llegar a tener este dispositivo de uso cotidiano para pacientes que lo necesiten y que estén en constante monitoreo.

Este dispositivo será no intrusivo, no estorboso y tendrá un uso fácil. Se pretende que será de uso doméstico si es necesario y no solo serán utilizados en casa si no que pacientes que se encuentren en camillas no tendrán que realizar ejercicios forzados debido a que el dispositivo estará activo en el lapso del día para obtener datos necesarios, al igual que facilitara la obtención de diagnósticos por especialistas, también se hacen la toma de mediciones como latidos del corazón estos también se podrían llegar a tomar mediante el dispositivo al igual que se implementarían más posteriormente para mejorar la cantidad de datos obtenidos al momento, igualmente estos serán guardados para su posterior visualización y o exportación ya que muchos de estos se revisan en laboratorios especializados.

Capitulo II Marco Teórico

2.1 Fundamentos teóricos

2.1.1 ¿Qué es el sonido?

El sonido es considerado un cambio de presión en el ambiente que el oído puede llegar a percibir en varias ocasiones. El sonido es capaz de transmitirse por el aire, el agua o también puede llegar a atravesar objetos sólidos. La música, el habla humana e incluso el canto de distintas voces o animales, son ejemplos los cuales podemos apreciar diariamente, por la forma que tiene a menudo se refieren a este como la presión del sonido. El ruido no es capaz de dispersarse sin un espacio, este se puede propagar a través de espacios tales como el aire o el agua en forma de ondas, al igual que en sólidos se llega a propagar en forma de ondas. Las ondas de sonidos pueden ser generadas gracias a una fuente de sonido como puede ser un altavoz, esta puede provocar vibraciones en su entorno. A medida que la fuente (el altavoz) va haciendo vibrar el medio, estas se van alejando del lugar a la velocidad del sonido formando lo que conocemos como ondas de sonido. Lo que conocemos como nivel de presión acústica (SPL) es la medida logarítmica de un sonido con la relación de un valor de referencia esta se mide en decibelios (dB) por arriba de un nivel de referencia estandarizado este es de 20 µPa en el aire u otros gases, el cual se considera el umbral de audición humana. Podemos ver en ecuación 1 nos muestra cómo calcular el nivel de presión sonora (L_p) en decibelios [dB] a partir de la presión sonora (P) en Pascal [Pa].

Ecuación 1 Cálculo de nivel de presión sonora

$$L_p = 10 \cdot \log_{10}\left(\frac{p_{rms}^2}{p_{ref}^2}\right) = 20 \cdot \log_{10}\left(\frac{p_{rms}}{p_{ref}}\right)$$

Fuente: Domingo C., Universidad Politécnica de Cartagena, s.f.

Donde:

P_{ref} es la presión de sonido de referencia y P_{rms} es la presión de sonido RMS que se está midiendo.

2.1.2 Medición del Sonido

La principal forma de medir el sonido es mediante micrófonos. Los micrófonos son capaces de medir ondas de presión en una cantidad distinta de frecuencias llegando a cubrir todo tipo de rangos, llegando desde el oído humano e incluso más haya. Muchos de estos micrófonos están diseñados para captar sonido aéreo, pero otros se pueden llegar a usar para medir sonidos que están en el agua, a estos micrófonos en específico se les conoce como hidrófonos, al igual que también en tierra los cuales se les conoce como micrófonos sísmicos o instrumentos sísmicos.

2.1.3 El Ruido

Al igual que el sonido se puede decir que el ruido es el sonido que no deseamos en nuestra señal debido a que es la conformación de varios tipos de sonidos ya que todo lo que nos rodea puede llegar a producir algún tipo de ruido

Cabe la posibilidad de que se conozca el acrónimo NVH el cual significa ruido, vibración y aspereza. Las pruebas NVH implican la medición de distintos sonidos los cuales no son requeridos los cuales provienen de distintas fuentes para poder reducirlos o eliminarlos

El diseñar maquinas para poder limitar o hacer una mejora al ruido que producen ayuda a impulsar las mediciones de sonido y ruido. Aparte de ser molestos o de distraer, algunos de estos sonidos pueden llegar a ser molestos o incluso llegar a ser dañinos, por lo que la reducción de estos o eliminación completa es la mejor opción. El primero de los pasos en cualquier medida correctiva es tomar medidas objetivas para saber con qué se está tratando.

2.1.4 La calidad del sonido

La evaluación de la calidad del sonido permite obtener una valoración objetiva de cómo el oído humano percibe los sonidos generados por las máquinas. Mediante estas mediciones, los ingenieros pueden optimizar y ajustar el sonido emitido por sus máquinas con el fin de crear productos más agradables tanto para los que lo usan, así como para quienes las rodean.

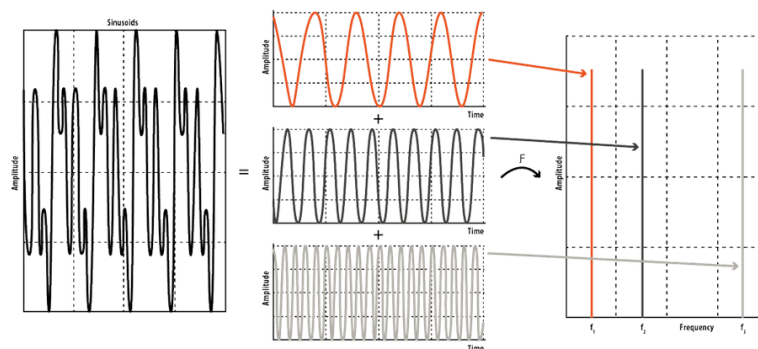
Aunque el sonido que podemos percibir como seres humanos es subjetivo, es factible realizar mediciones objetivas de los sonidos y posteriormente aplicarles diversos parámetros de calidad sonora.

2.1.5 ¿Cómo medir la Frecuencia del Sonido?

La mejor forma de poder medir frecuencias las cuales están situadas en el espectro sonoro es usar la transformada rápida de Fourier (FFT). La cual es capaz de convertir los conjuntos de algoritmos los cuales se encuentran en el dominio del tiempo al dominio de la frecuencia. Los sistemas encargados de la recolección de datos que regularmente son los que registran información de distintos tipos de fuentes entre ellos están los micrófonos. Posteriormente los datos recibidos y guardados son procesados y convertidos del dominio del tiempo al de la frecuencia en tiempo real

La cantidad de información obtenida de una señal en el dominio del tiempo es muy baja por lo que la transformada rápida de Fourier es la mejor opción al momento de pasarla al dominio de la frecuencia. "En este dominio las características de la señal se describen mediante componentes de frecuencias independientes." (Smith, 2022) Ya que en el dominio del tiempo esto es todo lo contrario mostrando solo una onda la cual contiene toda la información de la señal junta. Un ejemplo de esto se puede observar en la ilustración 1.

Ilustración 1 Una señal de tiempo que consta de componentes sinusoidales (izquierda) se convierte mediante algoritmos FFT al dominio de la frecuencia (derecha).



Fuente: Grant S., DEWEsoft, 2022

En un histograma en el cual se ve el dominio del tiempo, si vemos las ondas en

manera vertical podemos ver su amplitud la cual es la de la señal mientras que de manera horizontal podemos ver el tiempo que a transcurrido la señal, esto con la finalidad de lograr apreciar como se ve la amplitud de la señal en el tiempo

Ya que lo que se quiere conocer es sobre las frecuencias específicas contenidas en las señales complejas, el grafico de tiempo no puede llegar a tener utilidad en esta tarea, aunque son útiles en otros tipos de aplicaciones. Para poder conseguir la finalidad se usará una de las herramientas más útiles con respecto a conseguir datos relacionados a la frecuencia, gracias a la FFT podemos ver en la ilustración 2 una gráfica que nos muestra la frecuencia y la amplitud tomando los valores anteriores. Estos se muestran en el eje horizontal y vertical respectivamente, así podemos apreciar de mejor manera el espectro.

Se puede visualizar que el escalado de la frecuencia va desde los 0 Hz hasta los 10 kHz. Al igual que podemos ver un pico de amplitud que está en los 1200 Hz y un pico mucho más pequeño que esta por los 3750 HZ.

Ya que pudimos ver un ejemplo claro de los histogramas realizados por la FFT se puede saber que esta herramienta es una de las herramientas más esenciales con lo que respecta a la medición de frecuencia, al igual que esta se usa mucho en varios de los sistemas DAQ para la medición acústica, también una de las cualidades de estas graficas es que se pueden escalar linealmente o también logarítmicamente para poder obtener más información de estas

Ilustración 2 Ejemplo de un espectro en el dominio de la frecuencia de una señal de presión sonora medida en un entorno ruidoso

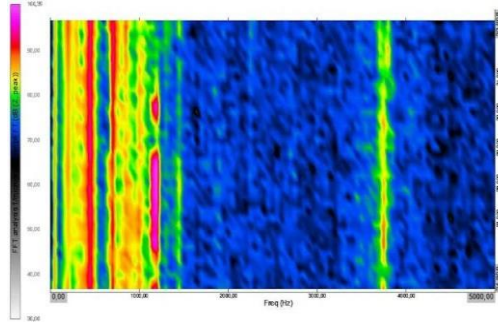


Fuente: Grant S., DEWEsoft, 2022

La denominación de tiempo de ventana se puede describir como la toma de una cierta cantidad de datos los cuales son procesados y recalculados mediante la transformada rápida de Fourier para posteriormente ser mostrados y que reciban un análisis mas adecuado.

Hay que tener en cuenta que es posible también representar el tiempo en un gráfico de la transformada rápida de Fourier (FFT), esto se puede lograr agregando otro eje por lo que tendríamos ahora 3. La imagen que se ve en la ilustración 3 es un ejemplo de esto la cual también es conocida como “pantalla de cascada”, en la cual se puede ver la magnitud, la frecuencia y el tiempo de esta, esta utiliza colores para poder facilitar la comprensión de la gráfica en cuestión.

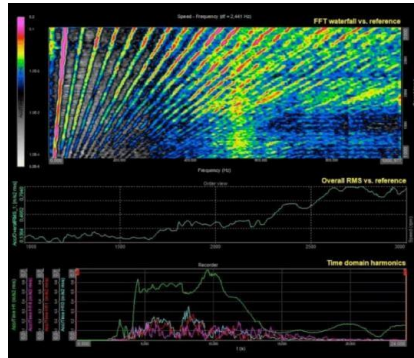
Ilustración 3 Ejemplo de análisis FFT en múltiples instancias de tiempo ilustrado en una pantalla 3D



Fuente: Grant S., DEWESoft, 2022

Hay muchos sistemas de adquisición de datos (DAQ) como los que fueron creados por Dewesoft, podemos ver una de sus distintas pantallas en la ilustración 4, estos incluyen diferentes herramientas para los análisis de la transformada rápida de Fourier (FFT) los cuales ya vienen incluidos por el mismo software.

Ilustración 4 Se muestran tres representaciones diferentes en una pantalla típica de DewesoftX: cascada FFT frente a señal de referencia, RMS general frente a señal de referencia y armónicos en el dominio del tiempo

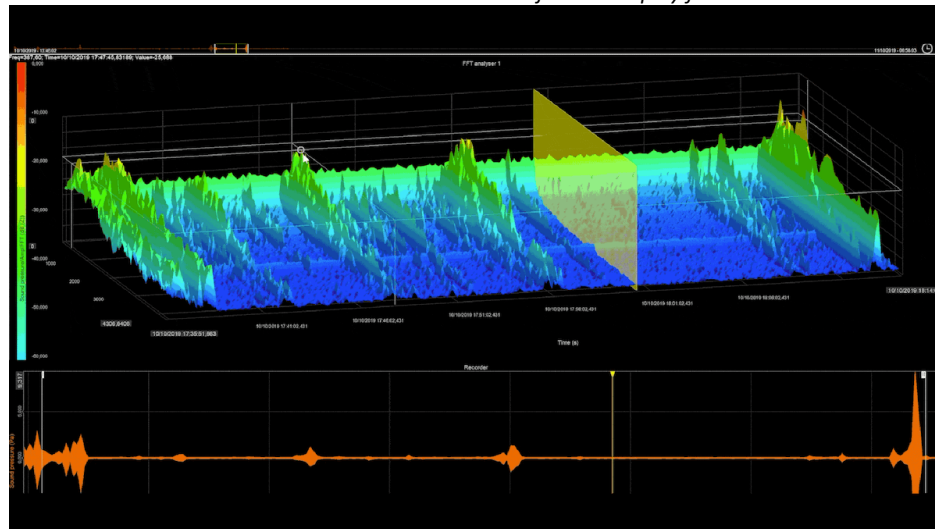


Fuente: Grant S., DEWEsoft, 2022

2.1.6 El procesamiento de señales ¿Qué es?

La manipulación de señales implica la transformación o modificación de información de manera que nos permita percibir características que no son visibles a través de la observación directa. Esta técnica permite a los ingenieros e investigadores analizar, optimizar y corregir señales, incluyendo transmisiones de datos científicos, audio, imágenes y video. Esto se puede observar en la figura 5.

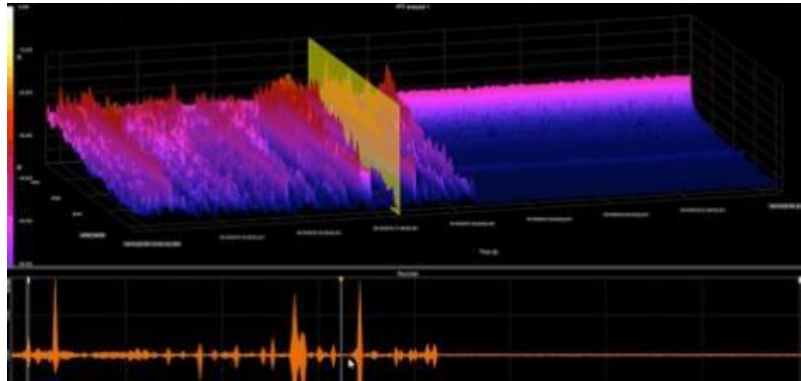
Ilustración 5 Visualización de datos de vibración en los ejes de tiempo y frecuencia simultáneamente



Fuente: Grant S., DEWEsoft, 2022

Todo esto tiene un propósito el cual es comprender cómo funcionará su producto usándolo en una actividad real. Esto solo puede obtenerse mediante la recopilación y el análisis de datos de alta calidad. Un ejemplo de esto se puede ver en la ilustración 6. (Smith, DEWEsoft, 2022).

Ilustración 6 Visualización de datos de vibración en los ejes de tiempo y frecuencia simultáneamente



Fuente: Grant S., DEWEsoft, 2022

Para poder comprender estos tipos de datos se necesitan realizar varios tipos de análisis como lo son:

2.1.7 Comparación

Una de las bases del análisis consiste en hacer una comparación entre C con D, como, por ejemplo, "¿cuánto aumenta el par del eje de transmisión al incrementar la velocidad de 100 a 200?" o "¿cómo varía la corriente en relación al voltaje?".

Estos métodos antes vistos se basan en la observación directa, donde se aplican diversos sensores y se registra la respuesta de la máquina o proceso en funcionamiento. Posteriormente, se examinan los datos obtenidos y se comparan los parámetros en diferentes etapas de operación. Sin embargo, existe otra herramienta de análisis de datos que se puede utilizar.

2.1.8 Procesamiento de señales

Transforma la información de tal forma que nos permite ver aspectos que no se pueden llegar a visualizar normalmente a través de simplemente verlo o también que no se puede delimitar por la comparación.

El procesamiento de señales digitales se presenta en diferentes formas dependiendo de la tarea que se desea realizar, muchos de estas tareas están relacionadas a la adquisición de datos (DAQ), utilizando el procesamiento de señales para analizar estos.

2.1.9 ¿Cómo se realiza el procesamiento de la señal?

El procesamiento de señales esta desde los inicios de la era tecnológica empezando porque este se hacia solamente en el dominio analógico, aunque aun se sigue realizando de esta forma, estos contruidos mediante componentes electrónicos tales como las resistencias y condensadores. Debido a que la tecnológica no deja de avanzar y cada vez la información se está volviendo más digital el procesamiento de señales tuvo que actualizarse y pasar del dominio analógico a la digital.

“Los chips DSP originales se desarrollaron en la década de 1960 y se utilizaron para mejorar RADAR y SONAR en aplicaciones militares.” (Smith, 2022) seguido de estos se siguieron ocupando para otras áreas como la geofísica con la finalidad de encontrar petróleo en otros lugares, al igual que en el área de la medicina tuvo un impacto fundamental con las imágenes medicas como lo son las tomografías computarizadas y las resonancias magnéticas en el escaneo del cuerpo humano

Muchas de las computadoras de hoy en día ya sean de escritorio o incluso portátiles llegan a tener mucha más potencia que muchas de las supercomputadoras que se ocupaban en el pasado para todos estos tipos de procesos complejos para la adquisición y manipulación de datos.

Muchos de estos sistemas se emplean para tareas fundamentales como los son los convertidores de analógico a digital (ADC) para los objetivos de filtrado, la combinación de flujo de bits y distintas cosas más. Ya que estos son capaces de manejar una gran cantidad de datos en muy poco tiempo. Muchos softwares ocupan el host de la computadora al igual que el DSP para realizar el procesamiento de la señal

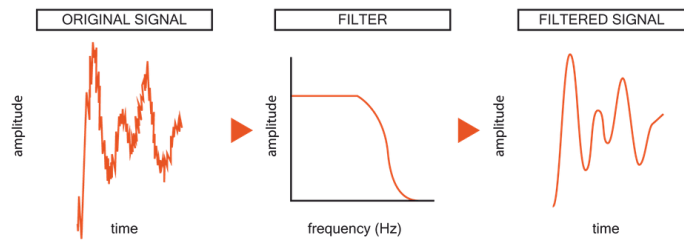
2.1.10 Tipos de Procesamiento de Señales

2.1.11 Filtración

Los filtros son una parte fundamental de todo tipo de procesamiento estos siendo la selección de lo que se quiere dejar pasar o que se pierda. O en un contexto más amplio “cualquier sistema que modifique ciertas frecuencias con respecto a otras

se denomina también filtro.” (Oppenheim, 2009)

Ilustración 7 Se utiliza un filtro de paso bajo para eliminar el ruido de alta frecuencia que no se desea de una señal

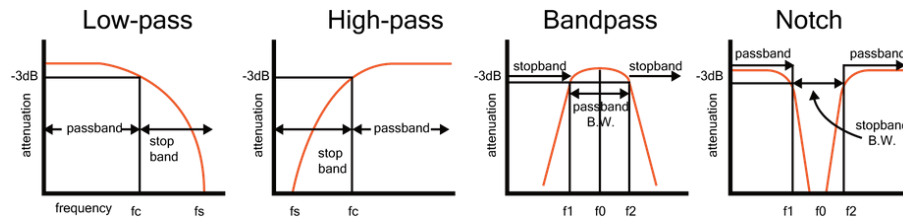


Fuente: Smith, DEWEsoft, 2022

En los sistemas de adquisición de datos (DAQ) los filtros de tipo analógicos se ocupan para poder tener una señal más tenue por arriba o por debajo de una frecuencia especificada, esto con la finalidad de poder tener una reducción amplia del ruido que produce la señal.

Los cuatro tipos básicos de filtros se pueden ver en la ilustración 8.

Ilustración 8 Tipos de filtro



Fuente: Grant S., DEWEsoft, 2022

Los filtros también están determinados por el número de polos que tienen. Cuantos más polos puedan generar, mayor será la señal en la que se puede bajar. Este tipo de pendiente significa los decibelios que pueden llegarse a reducir por cada octava de la señal. La especificación del filtro en cuestión normalmente reducirá dB/Q como máximo.

2.1.12 Filtros

La forma de hacer filtros se basa en elegir qué tipo de filtro que puede llegar a ser más adecuado para la señal con la que se estará trabajando, ya que ningún tipo de filtro está absuelto de distintos problemas como retrasos de fase, ondulaciones

u otro tipo de problemas en la banda de paso. Hay distintos tipos de filtros tales como los filtros Besse, Butterworth, los Elliptic y El conocido chebyshev, al igual que muchos más. Mientras pasa el tiempo se van desarrollando más tipos de filtros o prototipos de filtros también llamados estos para poder llegar a mejorar los resultados posibles de diferentes aplicaciones.

2.1.13 ¿Qué es mejor, el filtrado analógico o el digital?

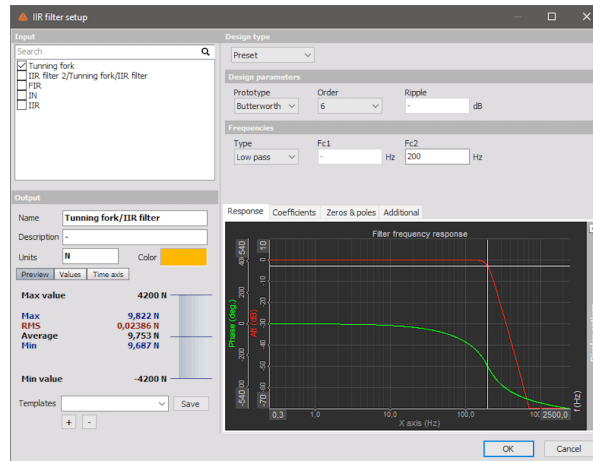
Dependiendo de que queremos llegar a conseguir se puede llegar a ocupar uno u otro regularmente para los filtros que solo se ocuparían en el dominio analógico son simples y no necesitan tener alguna alteración significativa. Mientras que los filtros digitales son mas utilizados para aplicaciones que están en un formato digital. También se tiene que tener en cuenta las complicaciones de componentes electrónicos utilizados en el filtrado analógico y la precisión de estos.

En el mundo de los sistemas de adquisición de datos los filtros se realizaban solamente dentro del front-end, todo esto siendo hasta que los sistemas de adquisición de datos pararon al formato digital después de 1988. Debido a esto no es muy común ver los acondicionadores de señal utilizando algún filtrado, salvo en algunos casos como lo son en las que hay que quitar ciertas frecuencias antes de la digitalización.

2.1.14 Filtro digital

Hay distintos softwares de computadora que pueden realizar filtros, a estos se les conoce como filtros digitales. Muchas de estas aplicaciones siendo de gama alta llegan a utilizar ASIC o FPGA para poder realizar las operaciones de filtrado. Al igual que también existen los DSP (Procesador de señal digital) dedicados.

Ilustración 9 Diseño del filtro en el software DewesoftX



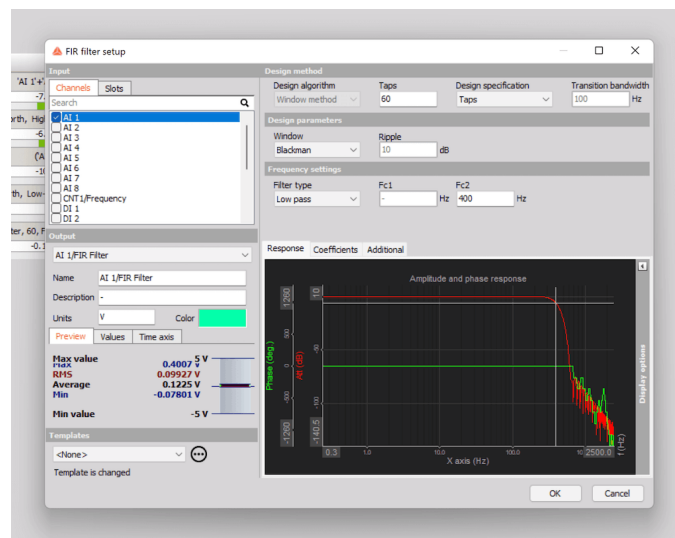
Fuente: Grant S., DEWEsoft, 2022

2.1.15 Filtrado IIR versus FIR

Hay dos diferentes enfoques los cuales son los básicos para el filtrado digital estos son los filtros IIR Y FIR para diferenciales se verán sus distintas desventajas y ventajas:

Los filtros de respuesta de impulso finito (FIR) estos se conocen por tener un cambio de fase esencialmente cero dentro de la banda de paso. Esto puede llegar a hacer un cambio muy considerable cuando de comparar señales en el eje del tiempo se trata, podemos ver en la ilustración 10 la pantalla de configuración del filtro FIR.

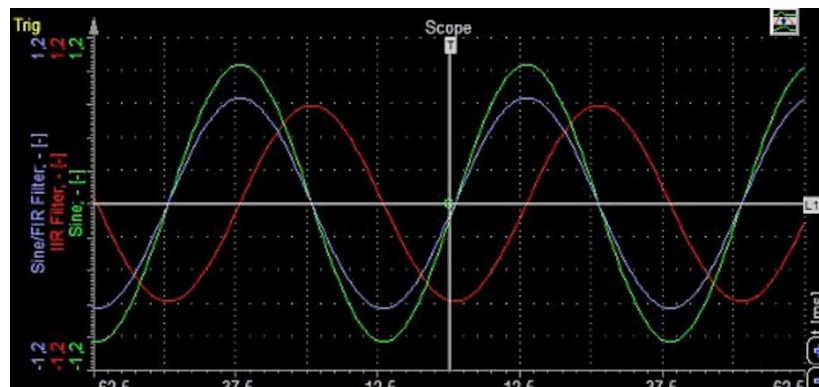
Ilustración 10 La pantalla de configuración del filtro FIR muestra el cambio de fase cero (línea verde) y la distorsión (línea roja) dentro de la banda de paso (línea gris vertical)



Fuente: Grant S., DEWEsoft, 2022

Los filtros de respuesta de impulso infinita (IIR) estos requieren de muchos menos datos que su contraparte llegando a consumir menos recursos durante el proceso. Estos son muy parecidos a los filtros analógicos. Un ejemplo claro de estos tipos de filtros es que se puede llegar a conseguir un filtro pasa bajas teórico exacto, al igual que otros tipos de filtros estandarizados que se pueden encontrar en los distintos circuitos analógicos. Podemos ver una comparación en software mediante la ilustración 11.

Ilustración 11 Comparación el cambio de fase de los filtros IIR y FIR



Fuente: Grant S., DEWEsoft, 2022

La imagen que podemos ver en la ilustración 11 es un gráfico comparativo en el cual la línea verde curva es la onda original siendo esta sinusoidal.

La siguiente curva la cual es la roja es el resultado calculado con respecto a un filtro IIR. El retraso que se puede apreciar es con respecto a la salida de fase.

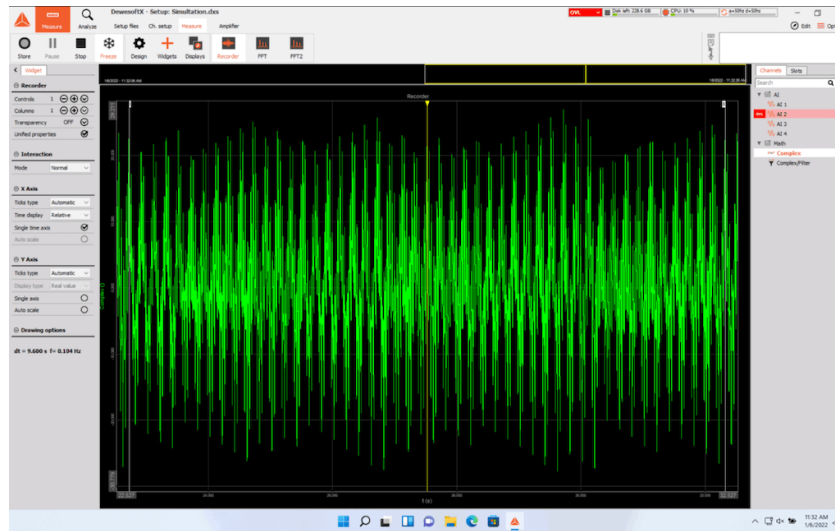
Mientras que por consiguiente la última curva configurada con color azul la cual es la calculada con un filtro FIR tiene un cambio de fase en cero lo que nos da por resultado que las señales no están siendo retrasadas por el filtro ya que lo que se quiere obtener es una comparación en la fase. Podemos ver que en este tipo de aplicaciones los filtros FIR tienen mayor peso

2.1.16 Análisis FFT

Al convertir una señal de su dominio principal el cual es el dominio del tiempo al dominio de la frecuencia podemos llegar a ver cosas que sin esta conversión serían incapaces de verse. Podemos ver un ejemplo de esta conversión en la ilustración 12 la cual es una forma de onda compleja CA que se estructura de distintas

frecuencias en grandes cantidades.

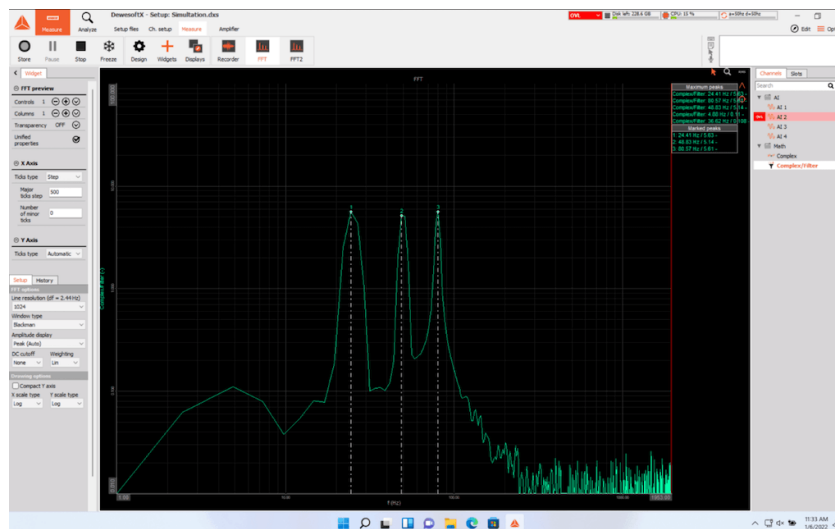
Ilustración 12 Una grabación de una forma de onda compleja



Fuente: Grant S., DEWEsoft, 2022

Una de las herramientas más útiles aparte de la conversión es la transformada rápida de Fourier (FFT) la cual en este caso delimita de mejor manera la cantidad de frecuencias de una magnitud más amplia usando por supuesto los datos anteriores, como se puede ver el resultado en la ilustración 13.

Ilustración 13 La misma señal se muestra en el dominio de la frecuencia (frecuencia frente a magnitud)



Fuente: Grant S., DEWEsoft, 2022

Gracias al procesamiento de señales de la transformada rápida de Fourier podemos visualizar nuestros datos de una manera más adecuada en el dominio de la frecuencia en lugar del dominio del tiempo

2.1.17 ¿Qué es el HIFI?

El término audio Hi-Fi solo significa audio de alta fidelidad. En español, realmente estamos hablando de un sonido con una alta fidelidad, con esto se refiere a un sonido el cual se acerca mucho mas al sonido original. Esto significa que no aparecen los típicos errores como la distorsión o el ruido.

2.1.18 Conceptos a tener en cuenta con respecto al audio HIFI

Hay 3 componentes que son indispensables para hacer la correcta reproducción de un sonido de alta fidelidad estos son:

2.1.18.1 Amplificador

Un amplificador es un dispositivo electrónico que amplifica el sonido y lo envía a un altavoz para que pueda ser escuchado por el oído humano. En general, este dispositivo no es responsable de la salida de sonido. En cambio, actúa como intermediario entre la persona y el hablante. Sin embargo, también es posible contar con reproductores con amplificadores incorporados que permiten la reproducción de CD y otros medios de almacenamiento.

2.1.18.2 Altavoces

Uno de los componentes mas importantes a la hora de hablar de sonido hifi son los altavoces los cuales están conectados a un amplificador para poder amplificar la señal de entrada en este caso el sonido, sin esta herramienta seria inaudible lo que se graba o se trata de escuchar.

La principal función de un altavoz es convertir las señales eléctricas las cuales provienen del amplificador para convertirlo en ondas que el oído pueda entender, estos pueden estar dentro o adaptado para estar conectado al dispositivo como por ejemplo los televisores, este tipo de conversión se basa en 2 etapas: la primera es en la cual la el sonido pasa por el diafragma del altavoz gracias a una señal eléctrica y posteriormente el movimiento que se produce gracias a esta señal eléctrica se convierte en ondas de presión(lo que conocemos como sonido) alrededor del altavoz. Una de las formas en las que se puede llegar a mejorar la calidad de sonido es colocando altavoces cerca de la caja de resonancia.

Hay 3 tipos de altavoces principales los cuales se pueden mostrar en el tipo de frecuencias que pueden emitir estos son:

Graves o woofer. Estos emiten frecuencias mucho mas bajas estas llegando a 1 kHz. Estas son generadas por instrumentos tales como el bajo

Medios. Estos son altavoces los cuales llegan a frecuencias de 2 kHz empezando desde los 256 Hz. Estos son los mejores ya que la calidad de sonido es bastante alta

“Agudos o tweeter. Este se encarga principalmente de las frecuencias mas altas o mas agudas siendo un altavoz pequeño.

2.1.18.3 Códec

En la gama de audio del mundo digital, el códec es un tipo de codificación el cual esta estandarizado para componentes de audio. Este proceso es de los mas importantes al momento de querer conseguir que el audio sea de alta fidelidad (HI-FI). Esto debido a que los códec principalmente pueden llegar a perder varios detalles importantes del sonido, por eso se fueron investigando otros formatos los cuales son sin compresión estos para mejorar la calidad del sonido, aunque ocupan más espacio algunos de estos formatos son el WAV o el AIFF.

Al igual que también existen formatos sin perdidas, estos se basan en aplicar una compresión al archivo de audio para que poder conservar todos los detalles de la pista original, algunos ejemplos de estos pueden ser el formato FLAC o el formato ALAC de Apple.

También existen formatos que tienen perdidas tales como el mas conocido del mundo que es el mp3, este formato tiene un peso casi nulo debido a que la calidad de sonido se pierde durante el proceso de codificación

2.1.19 Que es un convertidor analógico a digital

Un convertidor analógico a digital o ADC es un dispositivo electrónico capaz de convertir una señal analógica tales como sonidos grabados mediante un micrófono y pasarlos a un formato digital para un procesado mas sencillo de este, al igual

que las interferencias que se ven en una señal analógica llegan a ser eliminadas- Una señal analógica tiene una amplitud continua y una continuidad en el tiempo mientras que una señal digital tiene una amplitud discreta y un tiempo discreto igualmente. El convertidor analógico digital se encarga de realizar un proceso periódico para limitar el ancho de banda de la señal que entra, lo que provoca que la señal de entrada la cual es analógica se convierta en digital.

La finalidad de un ADC es convertir una señal eléctrica obtenida de distintos dispositivos como lo son micrófonos en un 0 o un 1. Al igual que un convertidor digital a analógico (DAC) hace lo contrario convierte una señal de 0 o 1 a una señal eléctrica.

2.1.20 Proceso para la conversión

El proceso el cual se lleva a cabo para la realización de una conversión de una señal digital a analógica se rige por una serie de procesos que son: el muestreo, la retención, la cuantificación y la codificación.

2.1.20.1 Muestreo:

El muestreo es la obtención de datos de forma constante, ya que este es un recurso técnico cuenta con distintas limitaciones siendo que durante el muestreo y aun en la retención la señal sigue estando en el dominio analógico y puede llegar a obtener cualquier tipo de valor, esto hasta llegar a la cuantificación en la cual la señal ya toma valores y se transforma en digital.

2.1.20.2 Retención:

En este momento, las muestras son procesadas por el operador de campo para poder comprobar su estado. Numéricamente, esta función se desprecia por las mismas razones que en el modelo.

2.1.20.3 Cuantificación:

Este procedimiento se encarga de medir el aire de los datos obtenidos, otorgándoles un límite del valor evaluado de la marca a un determinado nivel de producción. El problema es que el resultado siempre tiene un error llamado ruido de cuantización.

2.1.20.4 Codificación:

La codificación es un proceso en el que el valor obtenido por la cuantificación se convierte en un código binario, o códigos de otros tipos similares.

2.1.21 Tipos de convertidores Analógicos a Digitales

Existen muchos tipos de convertidores analógico a digital algunos de ellos son los siguientes:

2.1.21.1 Flash ADC

Este siendo uno de los ADC de mayor velocidad siendo que las 4 resistencias que tiene son del mismo valor. Al igual que el voltaje de conversión máxima es constante y la salida es de 2 bits, siendo los valores posibles 00 01 10 y 11. Mientras el codificador envíe a la salida un 1 no se convertirá a 0. Teniendo 8 bits para trabajar el ADC requiere 255 comparadores 256 resistencias y un codificador de 256 entradas

2.1.21.2 Convertidor analógico a digital de extensión de tiempo (TS – ADC)

Este es un tipo de ADC el cual al combinar varias tecnologías es capaz de digitalizar las señales las cuales tiene un gran ancho de banda el cual llega a ser mayor de lo que la mayoría de los ADC comunes pueden llegar a convertir. A este ADC se le llama igual “digitalizador de estiramiento de tiempo fotónico” (Menna, s.f.)

2.1.21.3 Convertidor analógico a digital Sigma-delta ($\Sigma\Delta$)

Este tipo de ADC ocupa técnicas de modulación delta las cuales convierten una señal analógica a una digital, pero en forma de bit-stream y enviando la señal por el canal específico de transmisión.

2.1.21.4 Aplicaciones de los convertidores analógico digital

Los convertidores analógico-digital son de los dispositivos electrónicos más útiles en la actualidad ya que muchos de los mecanismos y aparatos que usamos hoy en día lo necesitan para poder convertir señales análogas a señales digitales.

2.1.22 Procesamiento de señales digitales

La edición, modificación, procesamiento, almacenamiento y transporte de datos del dominio analógico al dominio digital es una de las principales aplicaciones de un convertidor de analógico a digital. Los dispositivos utilizados por estas aplicaciones incluyen microcontroladores, osciloscopios digitales y software crítico.

2.1.23 Instrumentos científicos

Muchos de los instrumentos que tenemos hoy en día necesitan ampliamente un ADC ya que los datos que reciben son de manera analógica los cuales posteriormente se convierten a una señal digital para su fácil comprensión, algunos de estos ejemplos son las tecnologías de radar los sistemas de teledetección, entre otros.

2.1.24 Procesamiento de audio

Al igual que los anteriores uno de los usos de los ADC es en el procesamiento de audio uno de los ejemplos más claros es al momento de las grabaciones de audio, estas se graban de manera analógica mediante micrófonos y convertidor transforma la señal a una digital.

Capitulo III

Desarrollo y metodología

3.1 Procedimiento y descripción de las actividades realizadas

Tabla 3 Cronograma de actividades

Cronograma de actividades																
Actividades	JUL	J-A	AGO	AGO	A-S	SEP	SEP	SEP	SEP	OCT	OCT	OCT	OCT	O-N	NOV	NOV
	24-29	31-05	07-12	14-19	28-02	04-09	11-16	18-23	25-30	02-07	09-14	16-21	23-28	30-04	06-11	13-18
Realización del estado del arte sobre procesamiento digital de señales																
Realización del estado del arte sobre Filtros																
Realización del estado del arte sobre circuitos amplificadores																
selección del circuito amplificador																
Verificación del circuito																
Selección del receptor de señales																
Pruebas de recepción de señales																
Visión de las señales recibidas																
Pruebas de procesado de señales																
Realización completa del procesado de señales																
Guardado de señales ya procesadas																
Verificación de resultados																
Revisiones preentrega final																
Entrega del prototipo																

Fuente: Autoría Propia 2023

3.1.1 Estado del arte

3.1.1.1 Procesamiento de señales

Para el inicio del proyecto se tiene el estado del arte, empezando por el procesamiento digital de señales que nos conduce desde el cambio de dominio de las señales pasando del dominio del tiempo al dominio de la frecuencia y viceversa hasta cualquier tipo de filtraje que se quiera realizar al igual que las amplificaciones y normalización todas estas en el ámbito de procesamiento de señales

3.1.1.2 Filtros

Ya que se tiene conocimientos sobre procesamiento de señales se puede continuar con familiarizarse con filtraje de todo tipo pasando desde el filtraje de manera analógica al igual que de manera digital, todos estos conocimientos para poder ser capaces de decidir qué tipo de filtro es el más adecuado para la señal con la que se está trabajando y también el tipo de filtro que se quiere utilizar habiendo de varios tipos

3.1.1.3 Circuitos amplificadores

Ahora que se tienen más conocimientos sobre filtraje y procesamiento de señales digitales se puede comenzar a investigar sobre distintos circuitos amplificadores conociendo así los componentes que ocupa y poder realizar un correcto análisis de distintos circuitos para saber cuál es el más adecuado para el proyecto en cuestión

Al igual que también se tomaron en cuenta sensores de sonido los cuales ya tienen micrófonos incorporados o hasta algún tipo de filtraje algunos de los que se tomaron en cuenta son el loudness sensor v0.9b y el sensor de sonido KY-038 estos se pueden ver en la ilustración 14.

Ilustración 14 loudness sensor y sensor ky-038.



Fuente: PRACTINET, 2023.

3.1.2 Selección del circuito amplificador

Ya que se tiene conocimientos adquiridos de las distintas investigaciones anteriores podemos comenzar a hacer la selección del circuito amplificador y preamplificador decidiendo que tipo de circuito es más conveniente para el micrófono, el cual es un micrófono electret, este se puede ver en la ilustración 15, el cual cuenta con un pequeño preamplificador dentro.

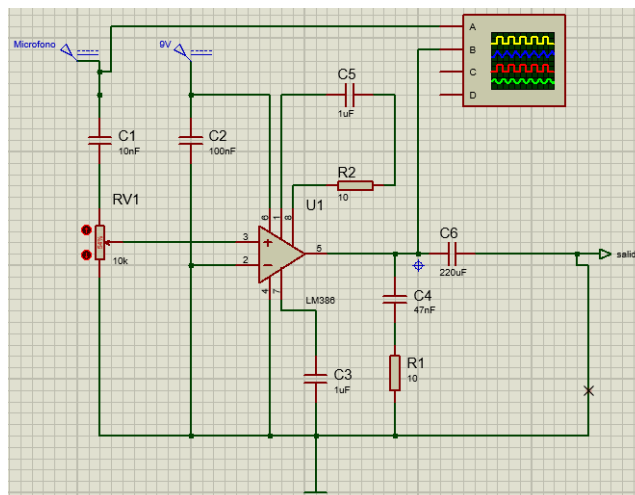
Ilustración 15 Micrófono electret.



Fuente: INTESC, 2023.

Primer boceto del circuito amplificador para el micrófono.

Ilustración 16 Circuito amplificador en proteus.

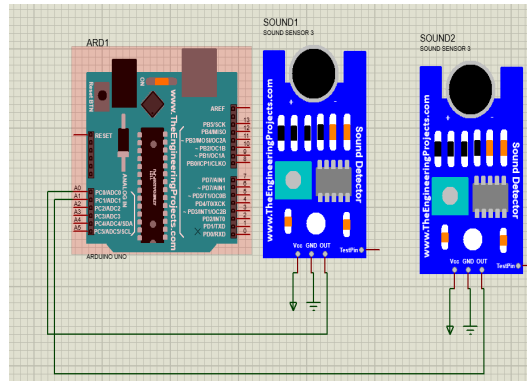


Fuente: Autoría propia, 2023.

Pasamos por distintos circuitos y sensores para poder conseguir el resultado final entre ellos están como previamente mencionados los sensores loudness sensor, los sensores ky-038, al igual que circuitos para poder polarizar los micrófonos como pudimos ver anteriormente al igual que un módulo max4466 que es un amplificador de micrófono electret este contando con 3 pines los cuales son gnd, vcc y out usando fundamentalmente out para la salida de datos conectándolo a nuestro Arduino para poder obtener la señal en cuestión que se procesara posteriormente probando desde códigos simples para poder cerciorarse de que funcionaba correctamente. Al igual que se hicieron pruebas con los sensores de sonido para ver

que podía ser el resultado. Se puede ver en la ilustración 17

Ilustración 17 conexión de sensores con arduino



Fuente: Autoría propia, 2023.

3.1.3 Verificación del circuito

Una vez obtenidos los conocimientos de las previas investigaciones y al igual que hecho la selección del circuito o sensor el cual se va a utilizar se procede a la verificación del circuito empezando por probarlo en una aplicación como es proteus en este caso.

Posterior a esto se puede empezar a realizar el circuito en físico para comenzar a hacer pruebas de rendimiento del circuito iniciando por recibir datos con este y viendo si hay alguna falla o alguna mejora que se pueda llegar a hacer en el circuito en el caso de que se decida usar el que se ve en la ilustración 16. y al igual la realización de la placa si es que se llega a hacer. Probando varias opciones como las que ya se han estado viendo anteriormente.

Ya que regularmente los sensores que se ocupaban solo tenían 3 pines que son los que ya se mencionaron anteriormente podemos usar la misma conexión para poder optimizar las pruebas al igual que se puede utilizar el mismo código para la visión de los sensores.

3.1.4 Selección del receptor de señales

Ya que se tienen los conocimientos previos y el circuito o sensor que se tomara para el proyecto en este caso gracias a posteriores investigaciones se logró encontrar una opción más adecuada la cual es un módulo apto para el micrófono electret el cual se llama MAX4466 el cual cuenta con un amplificador de audio

para el dicho micrófono.

Gracias a los conocimientos previos se puede tomar una decisión adecuada con respecto a los 3 tipos de placas que se planteaban usar, para eso se realizó una tabla comparativa con las características ventajas y desventajas de cada una de estas, esto se puede apreciar en la tabla 3.

Tabla 4 Comparación entre distintos microcontroladores.

Comparación entre microcontroladores			
	Atmega328p	Spartan-3E Starter Board de Digilent INC	TMS320VC5416
Características	"Fabricante: Atmel (Microchip). Voltaje de operación: 1.8 a 5.5 VDC. Arquitectura de CPU: 8 bit AVR Memoria flash: 32 KB. Memoria RAM: 2 KB. EEPROM: 2 KB. Frecuencia de operación: 20 MHz. Pines de IO: 23 Canales ADC: 10. Interfaces: UART, TWI, SPI. Temperatura de Operación: -40° a 85° C" (Inc., s.f.)	"Conector Hirose FX2 de 100 pines Tres puertos Pmod de 6 pines Teclado PS / 2 Dos conectores DB9 RS-232 Módulo Ethernet RJ-45 Cabezal de 16 pines para módulos LCD opcionales Conector SMA para entrada de reloj de alta velocidad" (Digilent, s.f.)	"DSP type: 1 C54x DSP (max)(MHz): 120, 160 CPU: 16-bit Rating: Catalog Operating temperature range (°C): -40 to 85 ROM en chip de 16K × 16 bits configurada para memoria de programa" (instruments, s.f.)
Ventajas	Fácil adquisición Bajo costo Tipo de lenguaje de programación de alto nivel C Conocimientos previos de uso del microcontrolador	Fácil adquisición Lenguaje de programación VHDL Costo medio	Fácil adquisición Codec de 16 bits Lenguaje de programación de alto nivel C componente especializado en procesamiento de señales DSP
Desventajas	ADC resolución reducida No cuenta con DAC	Conocimiento nulo de uso de la tarjeta Reducida resolución de DAC contra ADC del mismo FPGA.	Costo alto Conocimiento nulo de uso de la tarjeta

Fuente: Autoría propia, 2023.

Finalmente se decidió optar por la placa Arduino UNO debido a las ventajas con las que se señalaron principalmente en la tabla anterior.

3.1.5 Pruebas de recepción de señales

Ya que se tienen todos estos parámetros y los conocimientos previos podemos proceder a iniciar con la recepción de señales iniciando por el código en Arduino el cual consiste en definir como entrada analógica el pin A0 y A1 para los 2 sensores.

Esto debido a que en estos pines tiene el convertidor ADC, la definición de los pines como entrada analógica y la lectura se puede ver en la ilustración las siguientes líneas de código respectivamente.

```
const int sensorPin1 = A0;  
const int sensorPin2 = A1;  
int sensorValue1 = analogRead(sensorPin1);  
int sensorValue2 = analogRead(sensorPin2);
```

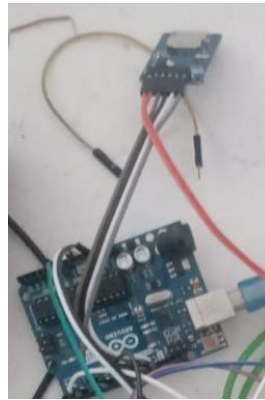
Ya que se definieron los pines y se definió la lectura de datos podemos empezar a mandarlos por el puerto serial el cual se definió con un Baudrate de 115200 para un envío mayor de datos por segundo al igual que el delay se definió a 1 microsegundo.

Para el mandado de datos se imprimieron las 2 variables en el puerto serial separado por comas como se puede ver en la ilustración 20.

```
Serial.print(sensorValue1);  
Serial.print(" , ");  
Serial.print(sensorValue2);
```

Se tomaron en cuenta varias formas de hacerse el código entre ellas también está la de la escritura en la sd que principalmente se probó guardando datos en un bloc de notas del sensor en la misma sd para poder ver si servía de manera adecuada podemos ver la conexión de Arduino y el módulo sd en la ilustración 21.

Ilustración 18 Conexión de un módulo sd a Arduino uno.



Fuente: Autoría propia, 2023.

Todo esto con la finalidad de asegurarnos que todo lo que íbamos a utilizar estuviera en correctas condiciones y se sepa usar de una manera correcta.

3.1.6 Visión de las señales recibidas

Ya que se tiene el sistema de adquisición de datos, los sensores y finalmente los conocimientos previos podemos empezar a recibir datos de los sensores para procesarlos para esto se ocupó Python como lenguaje de programación en pycharm su IDE más popular para esto empezamos a importar librerías empezando por Py serial para leer el puerto serial mientras que para mostrar las gráficas se ocupa la función Matplotlib.

Para eso podemos ver en las siguientes líneas de código como se realiza la adquisición de datos. Los datos se separan por comas y guardando los valores no sin antes verificar que son 2 valores los que se están detectando al igual que de que no esté guardando lugares vacíos, estos datos son cruciales ya que son los datos crudos que después pasaran a ser procesados, al igual que también se actualizan las gráficas de los 2 sensores para que se vean en tiempo real agregando constantemente los valores en los distintos ejes de la gráfica.

Empezando por las conexiones que se ocuparan entre Arduino y Python delimitando el puerto serial de Arduino al que tenemos conectada la placa.

```
ser = serial.Serial('COM3', 9600)
```

Procedemos a crear las listas en las cuales nuestros datos van a ser guardados. Al igual que la actualización de los datos para el guardado

```

data1 = []
data2 = []
data1.append(int(values[0]))
data2.append(int(values[1]))

```

Aquí se puede mostrar las líneas de código encargadas de las gráficas empezando por delimitar la ventana, definir títulos y la actualización de las gráficas.

```

fig, (ax1, ax2) = plt.subplots(2, 1)
ax1.set_ylabel('Sensor 1')
ax2.set_ylabel('Sensor 2')
ax2.set_xlabel('Muestras')
line1, = ax1.plot(data1, label='Sensor 1')
line2, = ax2.plot(data2, label='Sensor 2')
plt.ion()
plt.show()

```

y finalmente se usó un while true para la obtención de datos y la actualización de las listas. Esto con la finalidad de que fuera constante. Se puede ver la lectura de datos, la división de estos, la actualización de las listas, la actualización de las graficas con respecto a lo guardado en las listas y el reajuste de las graficas para que se vieran todos los datos al igual que una pequeña pause entre la visión de las gráficas.

```

try:
while True:
line = ser.readline().decode('utf-8').rstrip()
values = line.split(',')
data1.append(int(values[0]))
data2.append(int(values[1]))
line1.set_ydata(data1)
line2.set_ydata(data2)
ax1.relim()
ax1.autoscale_view()
ax2.relim()
ax2.autoscale_view()
plt.pause(0.01)

```

Todo esto para poder saber si se está recibiendo la señal de los 2 sensores y ver si esta es correcta o podemos notar algún tipo de falla.

3.1.7 Pruebas de procesamiento de señales

Ya que se sabe que los datos recibidos son los correctos y no tiene errores se

puede proceder a empezar a procesar las señales, iniciando por definir qué es lo que se hará primero, esto se sabe gracias a las investigaciones previas por lo que lo primero que se realizara será convertir la señal al dominio de la frecuencia para poder ver cómo es la gráfica de la señal y poder definir en donde se le aplicaran los demás procedimientos en este caso un filtraje para eliminar gran parte del ruido que se está recibiendo.

Empezando por las líneas mostradas en la siguiente parte, estas se encargan de hacer la transformada rápida de Fourier la cual ya es una función que trae implementada la librería Numpy de Python.

```
freq1 = np.fft.fftfreq(len(fft_y1), 1/sr1)
freq2 = np.fft.fftfreq(len(fft_y2), 1/sr2)
```

3.1.8 Realización completa del procesamiento de señales

Ya que se tiene el espectro de frecuencias gracias a la transformada rápida de Fourier que previamente se mostró en el código del 3.1.7 se continua con hacer el filtraje, el cual se realiza en el dominio de la frecuencia, el filtro es un filtro Butterworth de orden 8 pasa bajas con una frecuencia de corte de 0.1 esto lo hace la siguiente línea de código.

```
b, a = signal.butter(filter_order, normal_cutoff, btype='low', analog=False)
```

Al igual que se hace una amplificación de la señal antes y después del filtraje esto para que la primera amplificación mejore la relación de señal ruido mientras que la segunda amplificación ayuda a ajustar la amplitud general de la señal.

```
amplificacion_antes = 1.5
data_amplificada_antes = np.array(data) * amplificacion_antes
frecuencia_corte = 0.5 # Cutoff frequency
fs = 100 # Sampling frequency
b, a = signal.butter(8, frecuencia_corte, 'lowpass', fs=fs)
data_filtrada_antes = signal.filtfilt(b, a, data_amplificada_antes)
amplificacion_despues = 1.2
```

```
data_filtrada_amplificada_despues = data_filtrada_antes * amplificacion_despues
```

Después procedemos a realizar la normalización esto para poder normalizar la señal y que este dentro del rango adecuado para el posterior guardado en un archivo de audio todo esto mediante la línea se puede ver abajo.

```
data_filtrada_normalizada = np.int16(data_filtrada_amplificada_despues /
```

```
np.max(np.abs(data_filtrada_amplificada_despues)))
```

3.1.9 Guardado de señales ya procesadas

Después que se tienen las gráficas listas y la señal procesada ya podemos proceder a guardarla en su distintivo formato, los formatos que se pensaron para su uso fue el formato de audio .WAV debido a sus características las cuales son:

El formato es sin compresión al igual que no conlleva ninguna pérdida.

La máxima capacidad de espacio que puede ocupar es de 4 Gb.

Su utilización es principalmente en un formato no comprimido o PCM.

Su compatibilidad con la mayoría de codificadores de audio.

El consumo de espacio por minuto es menor a los 10 Mb.

Al igual que el formato MP3 es una de las opciones para su uso por sus características:

La compresión de datos de audio es muy eficiente.

La capacidad de eliminación de varios datos que se escuchan fuera del rango auditivo.

Reduce la calidad a los sonidos que no se escuchan.

Ya que se decide el formato de audio en este caso .WAV se empieza a realizar la escritura de los datos en el archivo de audio para eso tenemos las líneas de código que se muestran a continuación.

```
with wave.open('sonidos_respiratorios.wav', 'wb') as wf:  
    wf.setnchannels(1)  
    wf.setsampwidth(1)  
    wf.setframerate(frecuencia_muestreo)  
    wf.writeframes(data_filtrada_normalizada.tobytes())
```

Primero se definen el número de canales, el ancho de muestra y la frecuencia de muestreo, en este caso se establece el 1 para el numero de canales que se refiere a que la señal guardada en el archivo .WAV será mono y el ancho de muestra se define igualmente como 1 que se refiere a que estará para archivos de audio de 8 bits o 1 byte.

3.1.10 Verificación de resultados

Ya terminada la lógica de programación y recibidos los datos en cuestión se

procede a verificar si los resultados obtenidos por cada parte del programa son adecuados, esto verificando paso por paso mediante impresiones en la consola gráficas, entre otros métodos.

Esto para no tener problemas en cualquier parte del código sobre todo al momento de guardar los datos en el archivo de audio ya que si llegaran a corromperse los datos o modificar de manera incorrecta en alguna de las partes anteriores al guardado no se obtendrá un resultado óptimo.

3.1.11 Revisiones preentrega final

Después de que se verificaran los resultados y se revisara de posibles fallos o errores en los resultados procedemos a obtener estos datos de manera continua para seguir visualizando si el código esta correctamente optimizado y ver si es viable o poder cambiar algo al igual que si hay algún problema después de estar recibiendo datos después de un tiempo de ejecución.

3.1.12 Entrega del prototipo

En caso de que hubiera algún tipo de percance en la recolección de la señal se pueden llegar a realizar cambios al circuito por si alguna parte de este llegara a dañarse o simplemente no nos sirva para el circuito que queremos

Una vez que se tomó la consideración del formato y obtenidas los resultados anteriores de las señales se puede seguir con la lógica de programación para el guardado del audio.

Una vez teniendo una señal estable y ya procesada procedemos a realizar la exportación de estas por los medios que se requieran para poder visualizarlas o algún otro tipo de método de visualización.

Ya que se tienen los resultados finales procedemos a verificarlos de acuerdo a nuestro asesor por si hubiera algún tipo de desacuerdo o error en los resultados y poder solucionarlo de manera adecuada.

Después de todo lo realizado se revisará exhaustivamente el prototipo en busca de algún faltante o falla posterior a alguna parte del prototipo para finalmente se

pueda entregar de manera completa como se tenía previsto.

3.2 Alcance y enfoque de la investigación

El impacto del proyecto está orientado al sector salud al igual que esta parte de este está enfocada en la programación y electrónica, principalmente está enfocado en la adquisición y valoración de parámetros médicos en el área de Neumología, así como, en el registro del historial respiratorio de pacientes con problemas y/o enfermedades respiratorias crónicas o graves.

Debido a que no se cuentan con los conocimientos necesarios para poder dar un diagnóstico claro principalmente se tomaron datos de personas de la institución al igual que del estado de Puebla.

Se planea que esto llegue a ser utilizado en hospitales al momento de algún diagnóstico por alguna enfermedad, por más simple que sea como una tos podemos obtener varios datos para poder ocuparlos de manera más adecuada.

3.3 Hipótesis

Un sistema de adquisición de señales acústicas para el muestreo de audio pulmonar y respiratorio puede ser una alternativa al proceso de auscultación pulmonar para personas con enfermedades crónicas imposibilitadas para hacer los ejercicios respiratorios en consultorio médico.

3.4 Diseño y metodología de la investigación

Para la realización correcta del proyecto se definieron varias rutas capaces de dar una explicación consistente de lo que se realiza en cada fase crucial del trabajo en cuestión dándonos información oportuna y constante capaz de definir los pasos a seguir para la conclusión del proyecto entre algunos de los pasos a seguir están:

Ilustración 19 Diagrama



Fuente: Autoría propia, 2023.

Empezando por la investigación para tener una idea de en lo que se va a trabajar, esto conlleva todo lo que se aplicará en el ámbito del estado del arte que será la parte inicial del proyecto.

Posteriormente se desarrolla el algoritmo de procesado de señales y lo que conlleva esto el cual es la obtención por algún tipo de placa conectada a un circuito o sensor, al igual que el traspaso de datos de la placa para poder utilizarlos de manera adecuada en un posterior programa.

Pasando por el uso de los datos mediante graficas para poder visualizarlos de manera correcta y realizar un análisis estratégico para el filtraje de la señal.

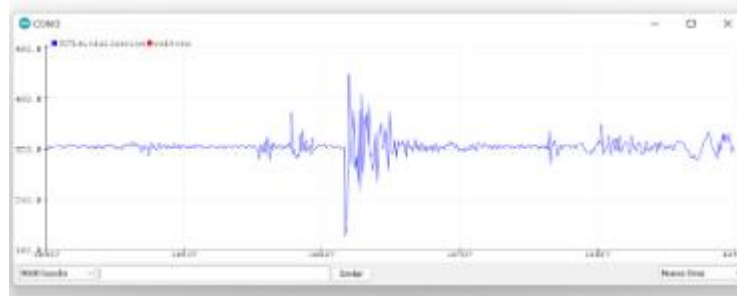
Hasta que finalmente se obtenga el resultado esperado que es el guardado y reproducción de datos en un formato de audio audible como lo es el .WAV.

Capítulo IV Resultados

4.1 Resultados

Empezando por la obtención de los valores de los sensores podemos verificar que se están recibiendo datos en Arduino de manera correcta mediante su monitor serie o el serial plotter igualmente de Arduino estos se muestran en la ilustración 20.

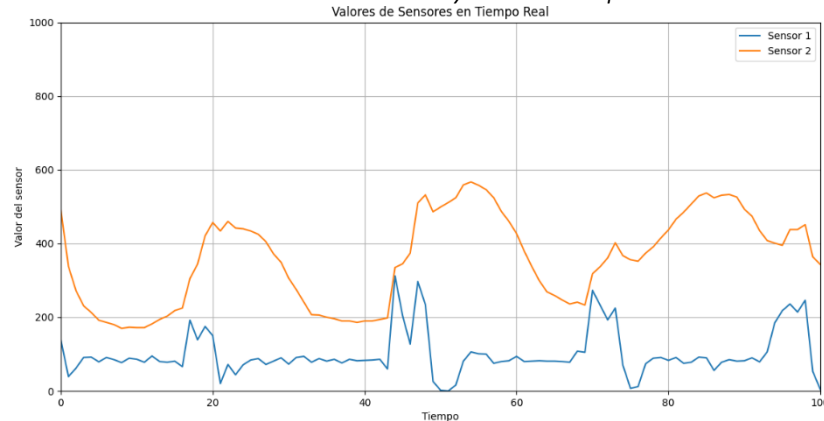
Ilustración 20 serial plotter y monitor serie de Arduino.



Fuente: Autoría propia, 2023.

Para que se pueda verificar cada micrófono de manera correcta se sobrepusieron en una gráfica con la finalidad de poder visualizar si es viable tener los micrófonos en esas partes del cuerpo o cambiarlos de lugar para una mejor verificación de datos como lo podemos ver en la ilustración 21.

Ilustración 21 Datos del sensor 1 y sensor 2 sobrepuestos.



Fuente: Autoría propia, 2023.

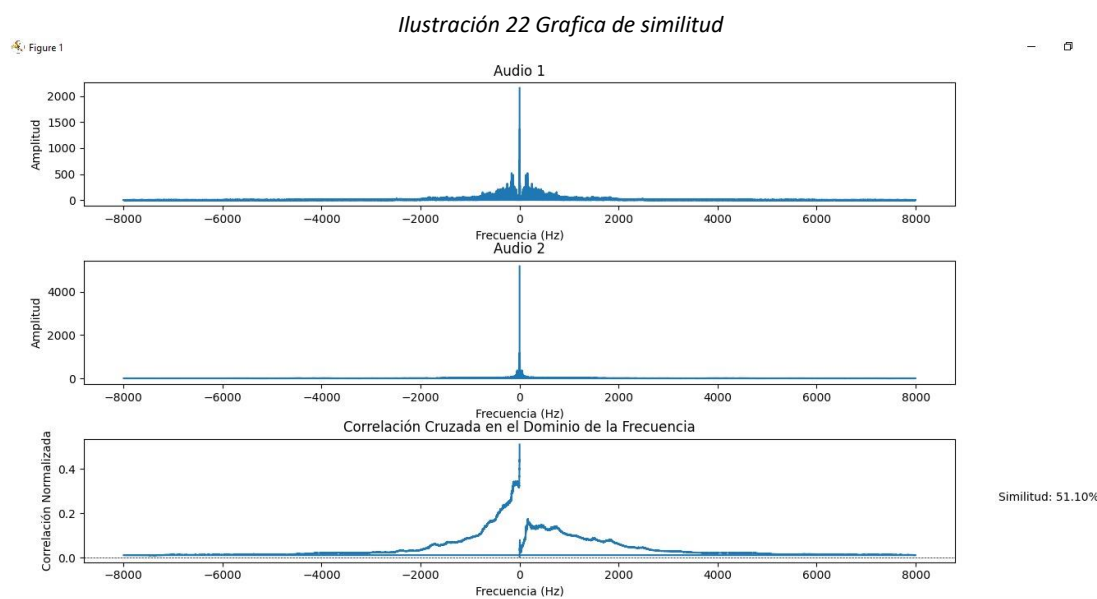
Una vez sabiendo que los datos se muestran de manera correcta y podemos visualizarlos en Arduino y Python con los métodos previamente mencionados estos datos se obtuvieron de un módulo sd conectado a Arduino al igual que el sensor.

El sensor actuaba como receptor de audio y el módulo sd era el que guardaba estos datos en un formato audible como lo es .WAV.

Ya que existe la posibilidad de cambiar un archivo de audio .WAV a un formato .TXT o block de notas pudimos trabajar de una manera adecuada la señal para poder aplicarle todo el procesamiento que requiere.

Esto debido a que el traslado de datos entre Arduino y Python por la conexión serial es lento al igual que llega a tener problemas con los caracteres y pueden llegar a generarse caracteres basura o saltos de línea no deseados.

Para poder realizar las conversiones adecuadas a cualquier tipo de señal se probaron distintos lugares específicos los cuales son los que se usaban mas recurrentemente a la hora de realizar la revisión pulmonar de pacientes y se probaba si tenían algún porcentaje de similitud. Podemos ver las graficas comparadas en la ilustración 22.

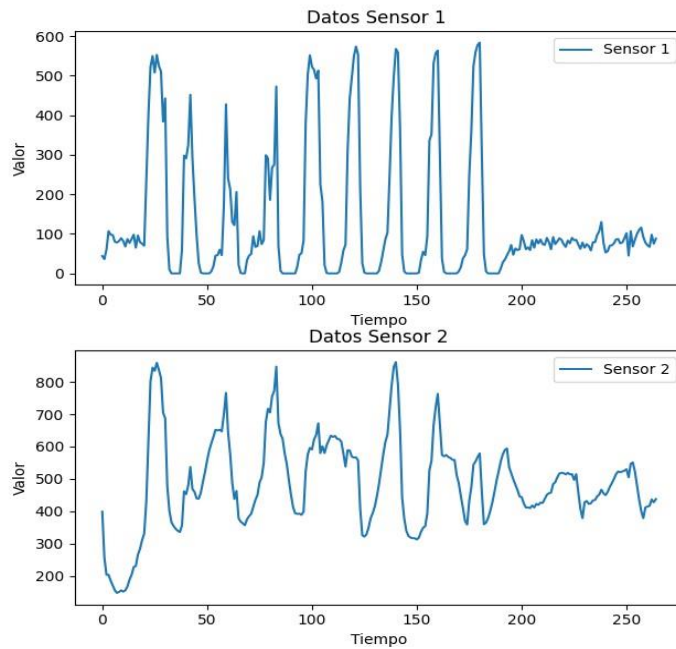


Fuente: Autoría propia, 2023.

Antes de que se empiece a aplicar algún cambio a la señal, se realizaron pruebas con gráficos para poder ver si se estaban recibiendo algún tipo de señal al igual de saber si esta era coherente ya que esto es con lo que se va a trabajar no tiene que tener ningún tipo de problema ni cambio brusco la señal. Las señales de 2 sensores pueden ver en la ilustración 23.

Ilustración 23 Datos recibidos por Python y mostrados en gráficas.

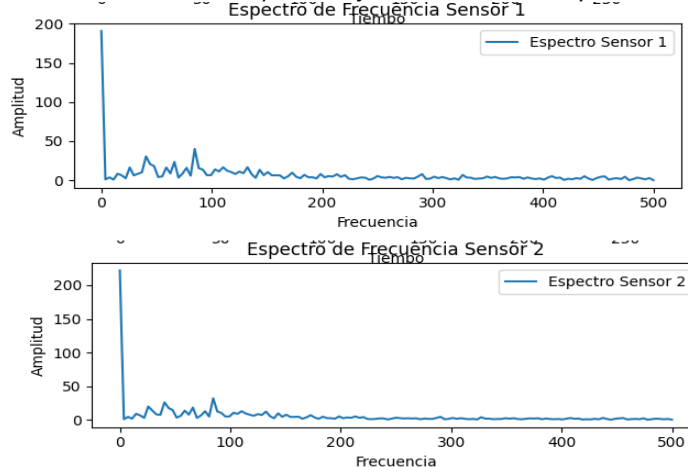
Figure 1



Fuente: Autoría propia, 2023.

Después de realizar la verificación de datos y visualizar que todo este correcto con respecto a el recibimiento de señales por medio de Arduino ya se puede comenzar a tratar la señal empezando por la conversión de la señal, del dominio del tiempo al dominio de la frecuencia mediante la transformada rápida de Fourier que nos da la biblioteca Numpy en Python como previamente se vio en el capítulo III pruebas de procesamiento de señales, una vez realizada la transformada podemos ver los resultados de esta mediante una gráfica y analizar las frecuencias que tiene la señal de cada sensor como se puede ver en la ilustración 24.

Ilustración 24 Espectro de frecuencia de sensor 1 y 2.

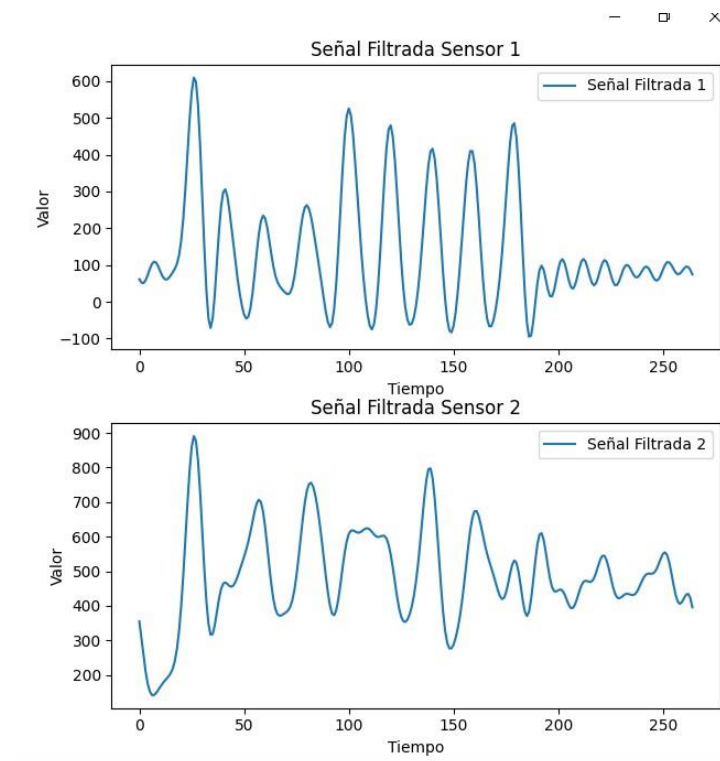


Fuente: Autoría propia, 2023.

Tenemos la amplitud de las frecuencias en el eje y mientras que los niveles de frecuencias a los que llegamos en el eje x, podemos ver picos de amplitud en ciertas frecuencias, ya que se analizaron las gráficas y se puede saber que frecuencias queremos eliminar y cuales queremos conservar se procede a generar un filtraje este se aplica a los datos en el dominio de las frecuencias para enfocarnos en eliminar el tipo de frecuencia que no nos interesa en la señal

El filtro que se decidió fue un filtro Butterworth pasa bajas de grado 8 con una frecuencia de corte de 0.1hz al igual que previa a la filtración se optó por aplicar de igual forma una amplificación a la señal, al igual que pos filtraje podemos ver la señal con los cambios realizados previamente mencionados en la ilustración 25.

Ilustración 25 señal filtrada.



Fuente: Autoría propia, 2023.

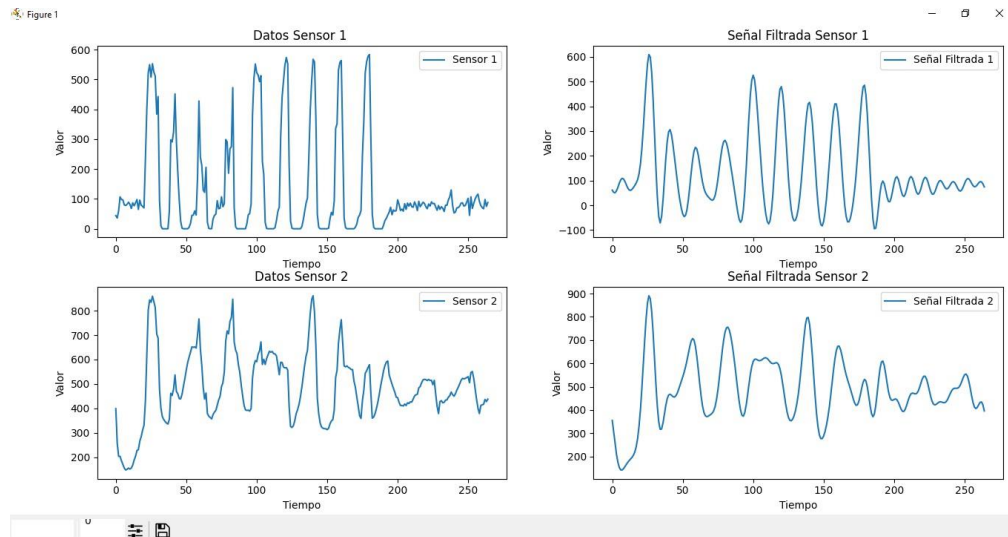
Una vez obtenidos los cambios a la señal y se puede ver que cada cambio se aplicó de manera correcta y se encontró ningún percance podemos pasar a normalizar la señal lo que hará que nuestra señal esté en condiciones de escribirse de manera correcta en el archivo de audio ya normalizada paso se regresó al dominio del tiempo para la escritura en el archivo de audio en este caso con formato .WAV, las

gráficas se mostraban en tiempo real para una mejor visualización de estos

Para que los procesos pudieran producirse sin necesidad de esperar a que el sensor 1 terminara para pasar al sensor 2 se optó por usar hilos y poner el proceso que cada sensor lleva independientemente y que se estén produciendo los métodos antes vistos de manera simultánea y que no llegue a hacer interrupciones

Ya que las principales pruebas se realizaron con 2 micrófonos podemos obtener 2 señales al mismo tiempo de la señal en estado crudo y la señal filtrada con todos los cambios que se mencionaron previamente esto se puede ver en la ilustración 26.

Ilustración 26 Graficas de lo sensores (señal cruda a la izquierda, señal filtrada a la derecha).



Fuente: Autoría propia, 2023.

Podemos ver como la gráfica muestra pico en los diferentes momentos en los que la persona está respirando al igual que también se muestra una señal filtrada.

La señal filtrada se debe a que después de la realización de la toma de datos al momento de la respiración de la persona estos se procesaran para poderlos guardar posteriormente en un archivo de audio para su posterior reproducción podemos ver el resultado que nos da en la ilustración 27.

Ilustración 27 Archivos de audio guardados.

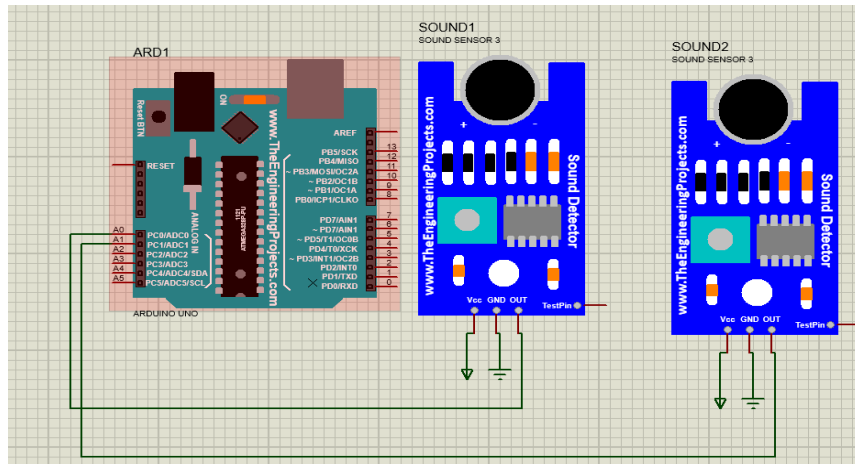
Nombre	Fecha de modificación	Tipo	Tamaño
1.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1 KB
2.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1,336 KB
3.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1 KB
4.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1 KB
5.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1 KB
6.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1 KB
7.WAV	01/01/2000 01:00 a. m.	Archivo WAV	1 KB

Fuente: Autoría propia, 2023.

Podemos ver el circuito en la ilustración 28, que se creó con el Arduino y el sensor conectando a la alimentación y la tierra del mismo Arduino la conexión de cada sensor se dio en los puertos analógicos del Arduino tales como A0, A1, etc.

Con los códigos anteriores podemos obtener el código para los demás sensores fácilmente, sería la misma lógica de programación para cada sensor al igual que solo sería colocarlos en otros hilos para que se pueda ejecutar secuencialmente.

Ilustración 28 Conexión de sensores con Arduino.



Fuente: Autoría propia, 2023.

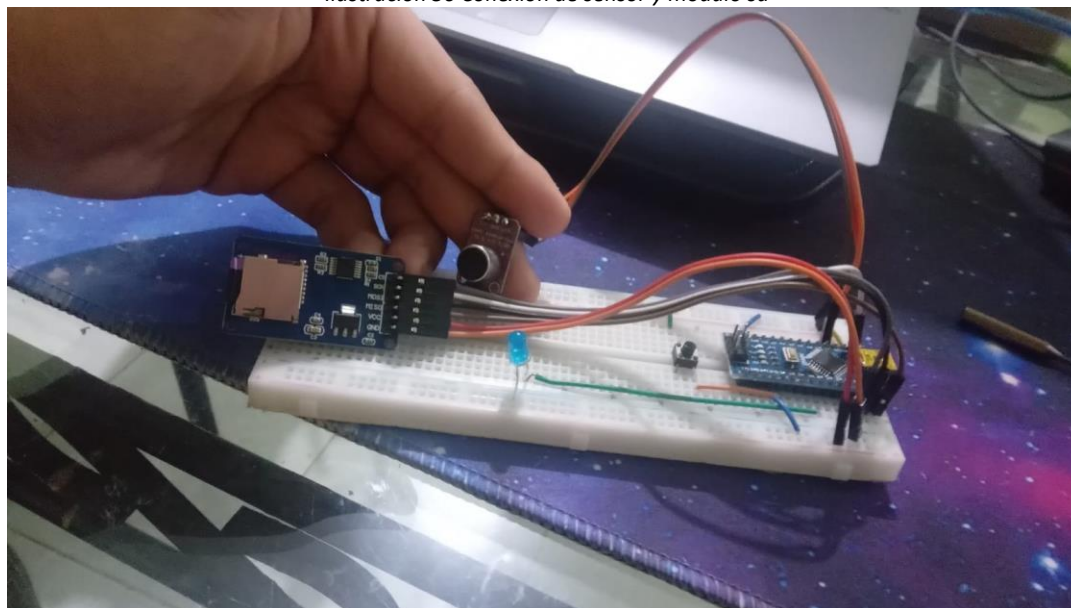
Para poder obtener el resultado de manera óptima se probó con una carcasa para limitar el ruido que genera cualquier tipo de percance externo que se pueda generar por el chaleco o por algún agente externo también podemos visualizar el circuito hecho en físico antes de ponerlo en el chaleco en la ilustración 29 al igual que en la 30.

Ilustración 29 Conexión de sensor y modulo sd



Fuente: Autoría propia, 2023.

Ilustración 30 Conexión de sensor y modulo sd



Fuente: Autoría propia, 2023.

Las selecciones de los puntos que se pueden ver en la ilustración 34 se verifico con los datos anteriores los cuales pudimos apreciar en los capítulos anteriores y se probó que cada sensor tuviera valores parecidos para verificar que si sirvan estos lugares seleccionados.

Ilustración 31 puntos en los que se pusieron los micrófonos para la recepción de datos.



Fuente: Autoría propia, 2023.

Capítulo V Conclusiones

5.1 Conclusiones del proyecto

Después de las constantes investigaciones y pruebas que se realizaron se puede llegar a concluir que la relación que hay entre las enfermedades respiratorias y la forma en que estas se pueden llegar a detectar no llega a estar ligada 100% a saber qué tipo de respiración se tiene, aunque esto igualmente da datos extras para poder saber con más exactitud el tipo de enfermedad.

Esto aparte llega a dar la posibilidad de reproducir dicho audio para poder analizarlo con más detenimiento, también se pueden analizar las gráficas que obtenemos de los micrófonos por si se quisieran analizar a más detalle, igualmente mientras más información tengamos de la persona en cuestión mejor se podrá detectar el problema.

También se llegó a resolver la problemática de las personas que no son capaces de respirar durante mucho tiempo o que les cuesta respirar debido a su enfermedad, esto se logró gracias a que se graban desde los micrófonos que captan el sonido de los pulmones.

5.2 Conclusiones relativas a los objetivos específicos

se empezó por la investigación debido a que se necesita saber en qué ámbito se va a empezar el proyecto en este caso se procedió por empezar a investigar referente al procesamiento de señales al igual que circuitos eléctricos enfocadas a los micrófonos

gracias al micrófono pudimos obtener datos adecuados enfocados a las bajas frecuencias que obtenemos de la respiración humana

la plataforma usada para el proyecto fue Python por la capacidad que tiene para realizar proyectos de este ámbito al igual que su tasa de productividad y facilidad de obtención al igual que por previos conocimientos asociados a este lenguaje fue más fácil el desarrollo del algoritmo

el formato de audio decidido fue gracias a sus características las cuales eran las mejores en comparación con otras y también dio buenos resultados como se esperaba

la fidelidad de audio nos ayudó para poder escuchar de mejor manera los audios producidos por el algoritmo del código

5.3 Conclusiones relativas al objetivo general

El algoritmo realizado para el proyecto se basó en el procesado de la señal obtenida por los micrófonos esto paso por una obtención de conocimientos previos a la creación del algoritmo estos son en el procesado de la señales, algo de electrónica y también en programación todo esto para poder realizar de manera adecuada el algoritmo que pasa desde la obtención de datos por el paso hacia convertir los datos en la trasformada de Fourier para poder aplicar un filtro, mostrando graficas de estos y después convirtiendo el resultado en un archivo de audio

5.4 Limitaciones del modelo planteado

Debido a la inaccesibilidad a pruebas con niños, se opta por hacer pruebas con hombres adultos.

Debido a que este proyecto solo va a recolectar y mostrar datos, no está destinado a hacer diagnósticos, ya que no contamos con los conocimientos necesarios de neumología, eso le corresponde a un especialista.

La complexión será para una persona de entre 40 a 60 años.

El prototipo está destinado para una talla mediana masculina.

5.5 Recomendaciones

Se podría mejorar la obtención de datos al igual que los micrófonos, también la información podría servir para que en un futuro esta pueda recopilarse en una

base de datos con el fin de obtener un diagnóstico automáticamente para poder agilizar la resolución de enfermedades de este tipo, de igual forma el sistema podría evolucionar en un sistema de monitoreo continuo y de uso diario por los pacientes.

Capítulo VI Competencias

Desarrollo

6.1 Competencias desarrolladas y/o aplicadas

Aunque se ya se tenían conocimientos previos de varias tecnologías empleadas en el trabajo fueron usadas de mejor forma al igual que se podría decir que evoluciono la forma en las que se usaron para conseguir algo de mayor impacto mientras que las tecnologías que nunca se habían llegado a usar fueron gratificadamente recibidas, al igual que también hubo una mejora al momento de utilizarlas.

Al momento de realizar cualquier trabajo con un equipo se debe de utilizar un aspecto muy importante el cual es la comunicación, esta herramienta no solo es obligatoria si no que facilita el trabajo de todo el involucrado en el equipo al igual que la toma de notas de todos los puntos clave en las conversaciones de equipo son de amplia importancia para llevar a cabo mejoras al producto final.

Ya que el trabajo de investigación es una de las partes fundamentales de cualquier tipo de documento ya sea trabajo de investigación, tesis, etc. Se mejoraron las formas de búsqueda la toma de información al igual que la interpretación que sobre todo es lo mas importante ya que es lo que entendemos de cada cosa que se extrae de una amplia variedad de documentación

Para el desarrollo de un trabajo se empiezan por distintas problemáticas dando hincapié a resolverlas, siendo estas el principio de más problemáticas que se tienen que resolver, ya que los problemas son de todos los días se mejoro la forma en la que se visualizan para poder dar un resultado para la mejora de dicho problema hasta la eliminación en casos necesarios.

Dicho ya problemas, deducciones, análisis, comunicación entre otras cosas se puede decir que la toma de decisiones es una de las partes importantes para cada aspecto descrito y una de las características que se mejoraron mediante el trabajo en cuestión para la obtención de mejores resultados.

Ya que se habló de la comunicación relacionado a los trabajos en equipo se mejoró ampliamente este parte relacionado a que principalmente el trabajo fue captado por varios participantes de este llevándonos desde abajo hasta el resultado final

compartiendo puntos de vista y delimitando mejoras para el dicho trabajo con la finalidad de la obtención de algo utilizable o mejorable

Ya que los trabajos de este tipo requieren de tecnologías o conocimientos de la edad moderna, se deben de desarrollar conocimientos o técnicas relacionados a estos al igual que aprender distintos tipos de formas de utilizar, gestionar e implementar los recursos como se debería todo esto continuamente ya que se va actualizando con el tiempo.

Para todo esto se debe de investigar, una palabra que esta diariamente plasmada en la vida ya que uno nunca deja de aprender ya no solo de nuestro alrededor si no de los errores que podemos llegar a tener enfrente. Esta es una de las mejoras que se van obteniendo continuamente ya que es lo que más se ocupa cotidianamente. Las ganas de investigar las cosas son parte de cualquier persona.

Capítulo VII Fuentes de información

7.1 Fuentes de Información

Bibliografía

- Alencastro, R. M. (octubre de 1998). *Escuela Politecnica Nacional*. Obtenido de <https://bibdigital.epn.edu.ec/bitstream/15000/10692/1/T1391.pdf>
- Blázquez, R. V. (Junio de 2017). *UNIVERSIDAD AUTONOMA DE MADRID*. Obtenido de https://repositorio.uam.es/bitstream/handle/10486/679384/Varela_Blazquez_Raul_tfg.pdf?sequence=1
- Candela, D. A. (s.f.). *Academia*. Obtenido de Academia: https://www.academia.edu/38378565/Tema_2_Series_y_transformadas_de_Fourier_pdf
- Candela, D. A. (s.f.). *Universidad Politécnica de Cartagena*. Obtenido de https://www.dmae.upct.es/~paredes/am_ti/apuntes/Tema%202.%20Series%20y%20transformadas%20de%20Fourier.pdf
- Carrizo, M. (s.f.). *Universidad Nacional del Sur*. Obtenido de <http://lcr.uns.edu.ar/fvc/NotasDeAplicacion/FVC-Martin%20Carrizo.pdf>
- DavidReveloLuna. (10 de Junio de 2022). *GitHub*. Obtenido de <https://github.com/DavidReveloLuna/ProcesamientoDatos>
- Digilent. (s.f.). *Digilent*. Obtenido de <https://digilent.com/reference/programmable-logic/spartan-3e/start>
- Fiore, J. M. (21 de Noviembre de 2022). *Operational Amplifiers & Linear Integrated Circuits*. Obtenido de Libretexts: http://www.dissidents.com/resources/OperationalAmplifiersAndLinearICs_3E.pdf
- gizmoj. (s.f.). *gizmoj*. Obtenido de <https://www.gizmojo.com.ar/products/grove-sound-sensor>
- Gjelstrup, L. (4 de Octubre de 2021). *DEWEsoft*. Obtenido de <https://dewesoft.com/es/blog/guia-para-el-analisis-fft>
- González, G. (1997). Series de Fourier, Transformadas de fourier y aplicaciones. *Divulgaciones Matemáticas*, 43-60.
- Huircán, J. I. (2007). *Academia.edu*. Obtenido de Academia.edu: <https://d1wqtxts1xzle7.cloudfront.net/40389335/ad03-libre.pdf?1448521821=&response-content-disposition=inline%3B+filename%3DAd03.pdf&Expires=1694736505&Signature=fNIB9sel5a23AiJvfHh8G8TTFGi6by1bCTMH7w3RJWeaebo5Kp-dRctgqTbgXELvGQWrJVINGavG4pKXF~fa0QxDIzEx>
- Instruments, T. (s.f.). *Texas Instruments*. Obtenido de https://www.ti.com/product/TMS320VC5416?utm_source=google&utm_medium=cpc&ut

m_campaign=epd-null-null-GPN_EN-cpc-pf-google-
 wwe&utm_content=TMS320VC5416&ds_k=%7B_dssearchterm%7D&DCM=yes&gad_sourc
 e=1&gclid=Cj0KCQiAgqGrBhDtARIsAM5s0_nJhe7TGs6unhvSDC1UnSeWepMV_
 INTESC. (s.f.). *INTESC*. Obtenido de <https://intesc.mx/productos/microfono-electret/>

Lukacs, Z. W. (1972). *Fourier series and Integrals*. Orlando: Academic Press, INC.

Marcelino Martinez Sober, A. J. (2010). *UV*. Obtenido de Ingeniería y arquitectura: <http://ocw.uv.es/ingenieria-y-arquitectura/1-1/tema1.pdf>

MARTÍNEZ, J. F. (2011). *UV*. Obtenido de Ingeniería y arquitectura: http://ocw.uv.es/ingenieria-y-arquitectura/1-5/ib_material/IB_T4_OCW.pdf

MARTÍNEZ, J. F. (2011). *UV*. Obtenido de Ingeniería y arquitectura: http://ocw.uv.es/ingenieria-y-arquitectura/1-5/ib_material/IB_T2_OCW.pdf

Martinez, M. (12 de Mayo de 2021). *NOBBOT*. Obtenido de NOBBOT: <https://www.nobbot.com/que-es-la-transformada-de-fourier-y-para-que-sirve/>

Mendoza, L. (2021). *studocu*. Obtenido de studocu: <https://www.studocu.com/es-mx/document/instituto-tecnologico-de-colima/fisicoquimica-termodinamica/adquisicion-de-senales-auditivas-sonido-senal-de-audio-historia-del-registro-del-sonido/14240053>

Microchip. (s.f.). *Microchip*. Obtenido de <https://www.microchip.com/en-us/product/atmega328>

Molina, F. J. (s.f.). *Universidad de Alicante*. Obtenido de <https://vumet3r.github.io/jbledam/Memoria%20Final%20PDS.pdf>

Mondragón, N. A. (Mayo de 2008). *Tecnologico de monterrey*. Obtenido de <https://repositorio.tec.mx/bitstream/handle/11285/629497/33068001111794.pdf?sequence=1&isAllowed=y>

Peces, G. G. (Junio de 2017). *UAM*. Obtenido de UAM: https://repositorio.uam.es/bitstream/handle/10486/679384/Varela_Blazquez_Raul_tfg.pdf?sequence=1

PRACTINET. (s.f.). *PRACTINET*. Obtenido de <https://practinet.mx/producto/ky-037-4pin-sensor-de-voz-arduino-deteccion-de-sonido-modulo-3/>

Rico, Q. (2015). *Universidad la Salle*. Obtenido de <https://repositorio.lasalle.mx/bitstream/handle/lasalle/2046/DISEÑO%20Y%20CONSTRUCCIÓN%20DE%20UN%20ESTETOSCOPIO%20ELECTRÓNICO%20DE%20BAJO%20COS.pdf?sequence=1&isAllowed=y>

Sanz, Á. M. (Septiembre de 2017). *uc3m*. Obtenido de uc3m: https://e-archivo.uc3m.es/bitstream/handle/10016/28105/TFG_Alvaro_Moreno_Sanz_2017.pdf?sequence=1&isAllowed=y

Smith, G. M. (21 de Febrero de 2022). *DEWEsoft*. Obtenido de DEWEsoft: <https://dewesoft.com/es/blog/que-es-procesamiento-de-senal>

Smith, G. M. (6 de Enero de 2022). *DEWEsoft*. Obtenido de DEWEsoft:
<https://dewesoft.com/es/blog/medicion-de-sonido-con-microfonos>

Smith, G. M. (28 de Junio de 2022). *DEWEsoft*. Obtenido de <https://dewesoft.com/es/blog/que-es-adquisicion-de-datos>

UNAM. (s.f.). *UNAM*. Obtenido de
<http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/276/A4.pdf?sequence=4&isAllowed=y>

Alegsa, L. (31 de Julio de 2023). *Alegsa*. Obtenido de
<https://www.alegsa.com.ar/Dic/wav.php#gsc.tab=0>

Atico34, G. (s.f.). *Grupo Atico34*. Obtenido de <https://protecciondatos-lopd.com/empresas/tipos-formato-audio/>

Digilent. (s.f.). *Digilent*. Obtenido de <https://digilent.com/reference/programmable-logic/spartan-3e/start>

Frías, L. (15 de Noviembre de 2021). *Gaceta UNAM*. Recuperado el 202, de
<https://www.gaceta.unam.mx/epoc-agravante-de-la-covid-19/>

GOB. (17 de Diciembre de 2021). *GOB*. Obtenido de
https://www.gob.mx/cms/uploads/attachment/file/706929/PAE_IRA_cF.pdf

Inc., M. T. (s.f.). *Microchip*. Obtenido de <https://www.microchip.com/en-us/product/atmega328p>

instruments, T. (s.f.). *Texas instruments*. Obtenido de
https://www.ti.com/product/TMS320VC5416?utm_source=google&utm_medium=cpc&utm_campaign=epd-null-null-GPN_EN-cpc-pf-google-ww&utm_content=TMS320VC5416&ds_k=%7B_dssearchterm%7D&DCM=yes&gad_source=1&gclid=CjwKCAiAzc2tBhA6EiwArv-i6Z555LTdBjxMOMtP0MUTeK8mH1yU0

López, K. V. (s.f.). *Cloudfront*. Obtenido de
https://d1wqtxts1xzle7.cloudfront.net/53336969/Reporte_ADC_y_DAC_Equipo___7-libre.pdf?1496194505=&response-content-disposition=inline%3B+filename%3DADC_y_DAC.pdf&Expires=1706389678&Signature=X85c4O-8FMQfg0VtLeqg1hkT2mq8T5mU1kwtNwMe1dw6Nub3JvFOU~jC1YqUPCwZ

Menna. (s.f.). *Como funciona*. Obtenido de <https://como-funciona.co/como-funciona-un-conversor-analogico-a--digital/>

Oppenheim, A. V. (2009). *Tratamiento de señales en tiempo discreto*. Madrid : PEARSON EDUCACIÓN S. A. Obtenido de
https://d1wqtxts1xzle7.cloudfront.net/64145456/Tratamiento_de_senales_en_tiempo_discreto_3ed_-_Oppenheim-libre.pdf?1597118568=&response-content-disposition=inline%3B+filename%3DTratamiento_de_senales_en_tiempo_discret.pdf&Expires=1706294768&Signature=I5wT

review, P. (s.f.). *Profesional review*. Obtenido de
<https://www.profesionalreview.com/2022/04/16/hi-fi-audio/amp/>

Smith, G. M. (21 de Febrero de 2022). *DEWEsoft*. Obtenido de DEWEsoft:
<https://dewesoft.com/es/blog/que-es-procesamiento-de-senal>

Vigo, U. d. (s.f.). *Universidad de Vigo*. Obtenido de
outlook.office.com/mail/inbox/id/AAQKADI2NzRmOGQ3LTk3NTUtNDUzOS1hMzU1LWI3MDVjMzdIOTIOMwAQALhsezKcDXPvFXDeRp9XII%3D?actSwt=true

Teziutlan, I. T. (2023). *itsteziutlan*. Obtenido de
<http://www.itsteziutlan.edu.mx/index.php/antecedentes-historicos/>

Capítulo VIII Anexo

8.1 Anexo

Ilustración 32 Código de Arduino para las gráficas.

```
1 void setup() {
2   // Inicializar el puerto serial a 9600 baudios
3   Serial.begin(9600);
4 }
5
6 void loop() {
7   // Leer los valores de los sensores
8   int valor_sensor1 = analogRead(A0);
9   int valor_sensor2 = analogRead(A1);
10
11   // Enviar los valores separados por una coma a través del puerto serial
12   Serial.print(valor_sensor1);
13   Serial.print(",");
14   Serial.println(valor_sensor2);
15
16   delay(100); // Ajusta el tiempo de espera según sea necesario
17 }
```

Fuente: Autoría propia, 2023.

Ilustración 33 Código de Arduino para el guardado de audio

```
1 void setup() {
2   Serial.begin(115200); // Inicia la comunicación serial a 115200 baudios
3 }
4
5 void loop() {
6   int sensorValue = analogRead(A0); // Lee el valor del sensor en A0
7   Serial.println(sensorValue); // Envía el valor por el puerto serie
8   delayMicroseconds(1); // Espera un breve tiempo antes de la próxima lectura
9 }
```

Fuente: Autoría propia, 2023.

Ilustración 34 Código de Python para las gráficas parte 1

```
import serial
import matplotlib.pyplot as plt
import numpy as np
from scipy.fft import fft, ifft
from matplotlib.animation import FuncAnimation
import threading
import time

# Configuración del puerto serial y la figura de la gráfica
puerto_serial = 'COM3'
velocidad_serial = 9600

arduino = serial.Serial(port=puerto_serial, baudrate=velocidad_serial, timeout=1)

fig, ((ax1, ax2), (ax3, ax4), (ax5, ax6)) = plt.subplots(3, 2, figsize=(10, 12))
x_data1, y_data1 = [], []
x_data2, y_data2 = [], []
max_points = 1000 # Máximo número de puntos a mostrar
line1, = ax1.plot([], [], label='Sensor 1')
line2, = ax2.plot([], [], label='Señal Filtrada 1')
line3, = ax3.plot([], [], label='Sensor 2')
line4, = ax4.plot([], [], label='Señal Filtrada 2')
line5, = ax5.plot([], [], label='Espectro Sensor 1')
line6, = ax6.plot([], [], label='Espectro Sensor 2')

def actualizar_grafica(frame):
    lectura = arduino.readline().decode().strip() # Lee los datos del puerto serial
```

Fuente: Autoría propia, 2023.

Ilustración 35 Código de Python para las gráficas parte 1

```
if lectura:
    valores = list(map(float, lectura.split(',')))
    if len(valores) == 2:
        valor_sensor1, valor_sensor2 = valores

        if len(x_data1) >= max_points:
            x_data1.pop(0)
            y_data1.pop(0)
        if len(x_data2) >= max_points:
            x_data2.pop(0)
            y_data2.pop(0)

        x_data1.append(frame)
        y_data1.append(valor_sensor1)
        x_data2.append(frame)
        y_data2.append(valor_sensor2)

        line1.set_data(x_data1, y_data1)
        ax1.relim()
        ax1.autoscale_view()

        corte = 0.1 # Frecuencia de corte del filtro pasa bajos
        fft_vals1 = fft(y_data1)
        frecuencia1 = np.fft.fftfreq(len(y_data1))
        filtro1 = np.abs(frecuencia1) <= corte
        fft_vals_filtrados1 = fft_vals1 * filtro1
        signal_filtrada1 = np.real(fft(fft_vals_filtrados1))
```

Fuente: Autoría propio, 2023.

Ilustración 36 Código de Python para las gráficas parte 3

```
line2.set_data(x_data1, signal_filtrada1)
ax2.relim()
ax2.autoscale_view()

fft_vals2 = fft(y_data2)
frecuencia2 = np.fft.fftfreq(len(y_data2))
filtro2 = np.abs(frecuencia2) <= corte
fft_vals_filtrados2 = fft_vals2 * filtro2
signal_filtrada2 = np.real(fft(fft_vals_filtrados2))

line3.set_data(x_data2, y_data2)
ax3.relim()
ax3.autoscale_view()

line4.set_data(x_data2, signal_filtrada2)
ax4.relim()
ax4.autoscale_view()

# Espectro de frecuencia
N = len(y_data1)
T = 1.0 / max_points
xf1 = np.linspace(0.0, 1.0 / (2.0 * T), N // 2)
yf1 = 2.0 / N * np.abs(fft_vals1[:N // 2])

xf2 = np.linspace(0.0, 1.0 / (2.0 * T), N // 2)
yf2 = 2.0 / N * np.abs(fft_vals2[:N // 2])
```

Fuente: Autoría propio, 2023.

Ilustración 37 Código de Python para las gráficas parte 4

```

line5.set_data(xf1, yf1)
ax5.relim()
ax5.autoscale_view()

line6.set_data(xf2, yf2)
ax6.relim()
ax6.autoscale_view()

# Verificación de señal filtrada para enviar mensaje
if max(signal_filtrada1) > 200:
    tiempo_actual = time.strftime("%H:%M:%S", time.localtime())
    mensaje = f"El Sensor 1 superó 200 en la señal filtrada en el tiempo {frame} s."
    print(mensaje) # Puedes reemplazar esto por una acción que desees

if max(signal_filtrada2) > 200:
    tiempo_actual = time.strftime("%H:%M:%S", time.localtime())
    mensaje = f"El Sensor 2 superó 200 en la señal filtrada en el tiempo {frame} s."
    print(mensaje) # Puedes reemplazar esto por una acción que desees

return line1, line2, line3, line4, line5, line6

def leer_puerto_serial():
    while True:
        lectura = arduino.readline().decode().strip()
        valores = list(map(float, lectura.split(',')))
        if len(valores) == 2:

```

Fuente: Autoría propio, 2023.

Ilustración 38 Código de Python para las gráficas parte 5

```

valor_sensor1, valor_sensor2 = valores

if len(x_data1) >= max_points:
    x_data1.pop(0)
    y_data1.pop(0)
if len(x_data2) >= max_points:
    x_data2.pop(0)
    y_data2.pop(0)

x_data1.append(len(x_data1))
y_data1.append(valor_sensor1)
x_data2.append(len(x_data2))
y_data2.append(valor_sensor2)

# Iniciar el hilo para la lectura del puerto serial
thread_lectura_serial = threading.Thread(target=leer_puerto_serial)
thread_lectura_serial.daemon = True # Permite que el hilo se detenga al cerrar el programa principal
thread_lectura_serial.start()

interval = 50 # Intervalo de actualización en milisegundos
ani = FuncAnimation(fig, actualizar_grafica, frames=None, interval=interval)

# Configuración de títulos, etiquetas y leyendas
ax1.set_title('Datos Sensor 1 ')
ax1.set_xlabel('Tiempo')
ax1.set_ylabel('Valor')
ax1.legend()

```

Fuente: Autoría propio, 2023.

Ilustración 39 Código de Python para las gráficas parte 6

```
valor_sensor1, valor_sensor2 = valores

if len(x_data1) >= max_points:
    x_data1.pop(0)
    y_data1.pop(0)
if len(x_data2) >= max_points:
    x_data2.pop(0)
    y_data2.pop(0)

x_data1.append(len(x_data1))
y_data1.append(valor_sensor1)
x_data2.append(len(x_data2))
y_data2.append(valor_sensor2)

# Iniciar el hilo para la lectura del puerto serial
thread_lectura_serial = threading.Thread(target=leer_puerto_serial)
thread_lectura_serial.daemon = True # Permite que el hilo se detenga al cerrar el programa principal
thread_lectura_serial.start()

interval = 50 # Intervalo de actualización en milisegundos
ani = FuncAnimation(fig, actualizar_grafica, frames=None, interval=interval)

# Configuración de títulos, etiquetas y leyendas
ax1.set_title('Datos Sensor 1 ')
ax1.set_xlabel('Tiempo')
ax1.set_ylabel('Valor')
ax1.legend()
```

Fuente: Autoría propio, 2023.

Ilustración 40 Código de Python para las gráficas parte 7

```
ax6.set_title('Espectro de Frecuencia Sensor 2')
ax6.set_xlabel('Frecuencia')
ax6.set_ylabel('Amplitud')
ax6.legend()

plt.tight_layout()
plt.show()

arduino.close() # Cierra la conexión serial al cerrar la ventana de la gráfica
```

Fuente: Autoría propio, 2023.

Ilustración 41 código de Python para el guardado de audio Parte 1

```
import serial
import numpy as np
from scipy import signal
import matplotlib.pyplot as plt
import wave

# Conexión al puerto serie donde está conectado Arduino
ser = serial.Serial('COM3', 115200)

fs = 32000 # Frecuencia de muestreo

# Lectura de datos desde Arduino y procesamiento
num_muestras = 16000 # Número de muestras a leer
data = []
for _ in range(num_muestras):
    dato = ser.readline().decode('latin1').strip() # Lee el dato desde Arduino
    dato_numerico = ''.join(filter(lambda x: x.isdigit() or x in '.-', dato))
    try:
        valor = float(dato_numerico)
        data.append(valor)
    except ValueError:
        print(f"Valor no válido: {dato_numerico}")

# Generar el tiempo basado en la longitud final de data después del procesamiento
tiempo = np.linspace(0, (len(data) - 1) / fs, num=len(data))
```

Fuente: Autoría propio, 2023.

Ilustración 42 código de Python para el guardado de audio Parte 2

```
# Aplicación del filtro pasa altas
frecuencia_corte = 0.1 # Frecuencia de corte del filtro pasa altas
b, a = signal.butter(8, frecuencia_corte, 'highpass', fs=fs)

data_filtrada = signal.filtfilt(b, a, data)

# Guardar los datos filtrados en un archivo de audio WAV
data_filtrada_normalizada = np.int16(data_filtrada / np.max(np.abs(data_filtrada)) * 32767)
with wave.open('datos_filtrados.wav', 'w') as archivo_audio:
    archivo_audio.setnchannels(1)
    archivo_audio.setsampwidth(1)
    archivo_audio.setframerate(fs)
    archivo_audio.writeframes(data_filtrada.tobytes())

# Visualización de la señal en el tiempo y en frecuencia
plt.figure(figsize=(10, 6))

plt.subplot(2, 1, 1)
plt.plot(tiempo, data, label='Señal original')
plt.plot(tiempo, data_filtrada, label='Señal filtrada')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')
plt.legend()

# Transformada de Fourier
frecuencias, fft_vals = signal.periodogram(data, fs)
plt.subplot(2, 1, 2)
```

Fuente: Autoría propio, 2023.

```
plt.subplot(2, 1, 1)
plt.plot(tiempo, data, label='Señal original')
plt.plot(tiempo, data_filtrada, label='Señal filtrada')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')
plt.legend()

# Transformada de Fourier
frequencies, fft_vals = signal.periodogram(data, fs)
plt.subplot(2, 1, 2)
plt.semilogy(frequencies, np.sqrt(fft_vals))
plt.xlabel('Frecuencia (Hz)')
plt.ylabel('Amplitud FFT')
plt.title('Espectro de frecuencia')
plt.tight_layout()
plt.show()
```

Fuente: Autoría propio, 2023.

Asignación de Asesor, Comisión Revisadora, Entrega de trabajo Profesional y Dictamen



TECNOLOGÍA
NACIONAL EL NEÓN

Instituto Tecnológico Superior
de Teziutlán

Asunto: Asignación de Asesor(a), Comisión Revisora, Entrega de Trabajo Profesional y Dictamen

Teziutlán, Puebla, 03/enero/2024

Asesor(a): GABRIEL ANGEL RAMIREZ VICENTE
Integrante de Comisión Revisora: LUIS MANUEL GARCIA MARTINEZ
Integrante de Comisión Revisora: LUIS ANGEL MARTINEZ CABRERA
Presentes

Por este medio me permito informar que ha sido asignado como asesor(a) y comisión revisora del trabajo profesional que se convertirá en Tesis de:

Alumno(a): URIBE PEÑA JORGE LUIS
Apellido paterno/maternal/nombre(s)
Número de Control: 19TE0381 Licenciatura o Posgrado: INGENIERIA MECATRONICA
Plan: 2010 Correo Electrónico: L19TE0381@TEZIUTLAN.TECNM.MX

Cuyo tema es: DISEÑO DE ALGORITMO DE PROCESAMIENTO DE SEÑALES ELECTROACUSTICAS RESULTANTES DEL MONITOREO DEL APARATO RESPIRATORIO

25 palabras (máximo)

Se ha enviado a su correo institucional el trabajo profesional o de grado, por lo cual la comisión revisora tendrá 5 días hábiles para realizar las observaciones al alumnado, el/la interesado(a) tendrá igualmente 5 días para corregir y las enviará al correo electrónico institucional de la comisión revisora, agradezco de antemano su valioso apoyo en esta importante actividad para la formación profesional de licenciatura o de posgrado de nuestro alumnado egresado.

Dictamen de Comisión Revisora, Aprobación de Impresión o Grabación y Autorización para subirla al Repositorio del TecNM

Siendo el día: 19 de febrero de 2024 se reunieron los miembros de la comisión para revisar el trabajo asignado y una vez analizado se decidió liberarlo y aprobarlo para su grabación y programación de examen profesional.

GABRIEL ANGEL RAMIREZ VICENTE
Nombre y Firma del(la) Asesor(a)

LUIS MANUEL GARCIA MARTINEZ
Nombre y Firma de integrante de la Comisión Revisora

LUIS ANGEL MARTINEZ CABRERA
Nombre y Firma de integrante de la Comisión Revisora

MYRIAM SANCHEZ PEREZ
Subdirección Académica

R08/06/2023
Folio: 41

GOBIERNO DEL ESTADO
S.E.P.
INSTITUTO TECNOLÓGICO
SUPERIOR DE TEZIUTLÁN
SUBDIRECCIÓN
ACADÉMICA

F-SAC-18



Fracción 1 y II sin Aire Libre, Teziutlán, Puebla, C.P. 73960 Tels. 231 311 4000 / 4001 / 4002 / 4003
e-mail: itsteziutlan@hotmail.com | www.teziutlan.tecnm.mx



2024
Felipe Carrillo
PUERTO

Formato de Licencia de uso

LICENCIA DE USO OTORGADA POR Jorge Luis Uribe Peña, de nacionalidad Mexicana, mayor de edad, con domicilio ubicado en la calle Valdez, colonia centro en la ciudad de Tlapacoyan, Veracruz, en mi calidad de titular de los derechos patrimoniales y morales y autor(a) de la tesis denominada "Diseño de algoritmo de procesamiento de señales electroacústicas resultantes del monitoreo del aparato respiratorio" en adelante "LA OBRA" quien para todos los fines del presente documento se denominará "EL(LA) AUTOR(A) Y/O EL(LA) TITULAR", a favor del Instituto Tecnológico Superior de Teziutlán del Tecnológico Nacional de México, la cual se regirá por las cláusulas siguientes:

PRIMERA –OBJETO: "EL(LA) AUTOR(A) Y/O EL(LA) TITULAR", mediante el presente documento otorga al Instituto Tecnológico Superior de Teziutlán del Tecnológico Nacional de México, licencia de uso gratuita e indefinida respecto de "LA OBRA", para almacenar, preservar, publicar, reproducir y/o divulgar la misma, con fines académicos, por cualquier medio en forma física y a través del repositorio institucional y del repositorio nacional, éste último consultable en la página: (<https://www.repositorionacionalcti.mx/>).

SEGUNDA - TERRITORIO: La presente licencia se otorga, de manera no exclusiva, sin limitación geográfica o territorial alguna, de manera gratuita e indefinida.

TERCERA -ALCANCE: La presente licencia contempla la autorización para formato uso de "LA OBRA" en cualquier formato o soporte material y se extiende a la utilización, de manera enunciativa más no limitativa a los siguientes medios: óptico, magnético, electrónico, virtual (en red), mensaje de datos o similar, conocido o por conocerse.

CUARTA – EXCLUSIVIDAD: La presente licencia de uso aquí establecida no implica exclusividad en favor del Instituto Tecnológico Superior de Teziutlán; por lo tanto, "EL(LA) AUTOR(A) Y/O EL(LA) TITULAR" conserva los derechos patrimoniales y morales de "LA OBRA", objeto del presente documento.

QUINTA – CRÉDITOS: El Instituto Tecnológico Superior de Teziutlán y/o el Tecnológico Nacional de México reconoce que "EL(LA) AUTOR(A) Y/O EL(LA) TITULAR" es el(la) único(a), primigenio(a) y perpetuo(a) titular de los derechos morales sobre "LA OBRA"; por lo tanto, siempre deberá otorgarle los créditos correspondientes por la autoría de la misma.

SEXTA – AUTORÍA: "EL(LA) AUTOR(A) Y/O EL(LA) TITULAR" manifiesta ser el(la) único(a) titular de los derechos de autor que derivan de "LA OBRA" y declara que el material objeto del presente fue realizado por él(ella), sin violentar o usurpar derechos de propiedad intelectual de terceros; por lo tanto, en caso de controversia sobre los mismos, se obliga a ser el(la) único(a) responsable. Dado en la Ciudad de Teziutlán, Puebla, a los **once** días del mes de **febrero** de dos mil **veinticuatro**.

"EL(LA) AUTOR(A) Y/O
EL(LA) TITULAR",


Jorge Luis Uribe Peña
Nombre y Firma

"EL INSTITUTO TECNOLÓGICO
SUPERIOR DE TEZIUTLÁN"


Arminda Juárez Arroyo
Directora General
Nombre, Firma y Sello

Carta de Autorización, Consulta y Publicación

Tecnológico Nacional de México
Instituto Tecnológico Superior de Teziutlán

CARTA DE AUTORIZACIÓN DEL(LA) AUTOR(A) PARA LA CONSULTA Y PUBLICACIÓN ELECTRÓNICA DEL TRABAJO DE INVESTIGACIÓN

El que suscribe: **JORGE LUIS URIBE PEÑA**

Con Número de Control: **19TE0381**

Perteneciente al Programa Educativo: **INGENIERÍA MECATRÓNICA**

Por este conducto me permito informar que he dado mi autorización para la consulta y publicación electrónica del trabajo de investigación en los repositorios académicos.

Registrado con el producto: **TESIS**

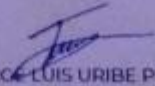
Cuyo Tema es:

DISEÑO DE ALGORITMO DE PROCESAMIENTO DE SEÑALES ELECTROACÚSTICAS RESULTANTES DEL MONITOREO DEL APARATO RESPIRATORIO

Correspondiente al periodo:
AGOSTO-DICIEMBRE 2023

Y cuyo(a) director(a) de tesis es:
M.S.C GABRIEL ANGEL RAMÍREZ VICENTE

ATENTAMENTE


JORGE LUIS URIBE PEÑA

Nombre y firma

Fecha de emisión: **13/02/2024**
c.c.p. Subdirección Académica

8.1.1 Índice de figuras

ILUSTRACIÓN 1 UNA SEÑAL DE TIEMPO QUE CONSTA DE COMPONENTES SINUSOIDALES (IZQUIERDA) SE CONVIERTE MEDIANTE ALGORITMOS FFT AL DOMINIO DE LA FRECUENCIA (DERECHA).....	20
ILUSTRACIÓN 2 EJEMPLO DE UN ESPECTRO EN EL DOMINIO DE LA FRECUENCIA DE UNA SEÑAL DE PRESIÓN SONORA MEDIDA EN UN ENTORNO RUIDOSO	21
ILUSTRACIÓN 3 EJEMPLO DE ANÁLISIS FFT EN MÚLTIPLES INSTANCIAS DE TIEMPO ILUSTRADO EN UNA PANTALLA 3D	23
ILUSTRACIÓN 4 SE MUESTRAN TRES REPRESENTACIONES DIFERENTES EN UNA PANTALLA TÍPICA DE DEWESOFTX: CASCADA FFT FRENTE A SEÑAL DE REFERENCIA, RMS GENERAL FRENTE A SEÑAL DE REFERENCIA Y ARMÓNICOS EN EL DOMINIO DEL TIEMPO	23
ILUSTRACIÓN 5 VISUALIZACIÓN DE DATOS DE VIBRACIÓN EN LOS EJES DE TIEMPO Y FRECUENCIA SIMULTÁNEAMENTE.....	23
ILUSTRACIÓN 6 VISUALIZACIÓN DE DATOS DE VIBRACIÓN EN LOS EJES DE TIEMPO Y FRECUENCIA SIMULTÁNEAMENTE.....	24
ILUSTRACIÓN 7 SE UTILIZA UN FILTRO DE PASO BAJO PARA ELIMINAR EL RUIDO DE ALTA FRECUENCIA QUE NO SE DESEA DE UNA SEÑAL	26
ILUSTRACIÓN 8 TIPOS DE FILTRO	26
ILUSTRACIÓN 9 DISEÑO DEL FILTRO EN EL SOFTWARE DEWESOFTX.....	28
ILUSTRACIÓN 10 LA PANTALLA DE CONFIGURACIÓN DEL FILTRO FIR MUESTRA EL CAMBIO DE FASE CERO (LÍNEA VERDE) Y LA DISTORSIÓN (LÍNEA ROJA) DENTRO DE LA BANDA DE PASO (LÍNEA GRIS VERTICAL)	29
ILUSTRACIÓN 11 COMPARACIÓN EL CAMBIO DE FASE DE LOS FILTROS IIR Y FIR.....	
ILUSTRACIÓN 12 UNA GRABACIÓN DE UNA FORMA DE ONDA COMPLEJA	30
ILUSTRACIÓN 13 LA MISMA SEÑAL SE MUESTRA EN EL DOMINIO DE LA FRECUENCIA (FRECUENCIA FRENTE A MAGNITUD).....	30
ILUSTRACIÓN 14 LOUDNESS SENSOR Y SENSOR KY-038.	38
ILUSTRACIÓN 15 MICRÓFONO ELECTRET	39
ILUSTRACIÓN 16 CIRCUITO AMPLIFICADOR EN PROTEUS	39
Ilustración 17 conexión de sensores con Arduino.....	40
Ilustración 18 Conexión de un módulo sd a Arduino uno.....	43
Ilustración 19 Diagrama.....	48
ILUSTRACIÓN 20 SERIAL PLOTTER Y MONITOR SERIE DE ARDUINO	51
ILUSTRACIÓN 21 DATOS DEL SENSOR 1 Y SENSOR 2 SOBREPUESTOS	51
Ilustración 22 Grafica de similitud.....	52
ILUSTRACIÓN 23 DATOS RECIBIDOS POR PYTHON Y MOSTRADOS EN GRAFICAS.....	53
ILUSTRACIÓN 24 ESPECTRO DE FRECUENCIA DE SENSOR 1 Y 2.	54
ILUSTRACIÓN 25 SEÑAL FILTRADA	55
ILUSTRACIÓN 26 GRAFICAS DE LO SENSORES (SEÑAL CRUDA A LA IZQUIERDA, SEÑAL FILTRADA A LA DERECHA).	55
ILUSTRACIÓN 27 ARCHIVOS DE AUDIO GUARDADOS.....	56

ILUSTRACIÓN 28 CONEXIÓN DE SENSORES CON ARDUINO.....	56
Ilustración 29 Conexión de sensor y modulo sd.....	57
Ilustración 30 Conexión de sensor y modulo sd.....	57
ILUSTRACIÓN 31 PUNTOS EN LOS QUE SE PUSIERON LOS MICRÓFONOS PARA LA RECEPCIÓN DE DATOS.....	58
ILUSTRACIÓN 32 CÓDIGO DE ARDUINO PARA LAS GRÁFICAS	71
ILUSTRACIÓN 33 CÓDIGO DE ARDUINO PARA EL GUARDADO DE AUDIO	71
ILUSTRACIÓN 34 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 1.....	71
ILUSTRACIÓN 35 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 1.....	72
ILUSTRACIÓN 36 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 3.....	72
ILUSTRACIÓN 37 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 4.....	73
ILUSTRACIÓN 38 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 5.....	73
ILUSTRACIÓN 39 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 6.....	74
ILUSTRACIÓN 40 CÓDIGO DE PYTHON PARA LAS GRÁFICAS PARTE 7.....	74
ILUSTRACIÓN 41 CÓDIGO DE PYTHON PARA EL GUARDADO DE AUDIO PARTE 1.....	75
ILUSTRACIÓN 42 CÓDIGO DE PYTHON PARA EL GUARDADO DE AUDIO PARTE 2.....	75
ILUSTRACIÓN 43 CÓDIGO DE PYTHON PARA EL GUARDADO DE AUDIO PARTE 3.....	76

8.1.2 Índice de Tabla

<u>TABLA 1 CRONOGRAMA DE ACTIVIDADES.....</u>	37
<u>TABLA 2 COMPARACIÓN ENTRE DISTINTOS MICROCONTROLADORES.....</u>	41