



Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ingeniería Mecánica Eléctrica

**DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE SONIDOS MIDI A  
LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN UNA  
GUITARRA ELÉCTRICA**

**Sergio Leonel Rodas Quiñonez**

Asesorado por el Ing. Armando Alonso Rivera Carrillo

Guatemala, febrero de 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE SONIDOS  
MIDI A LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN  
UNA GUITARRA ELECTRICA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA  
FACULTAD DE INGENIERÍA  
POR

**SERGIO LEONEL RODAS QUIÑONEZ**  
ASESORADO POR ING. ARMANDO ALONSO RIVERA CARILLO

AL CONFERÍRSELE EL TÍTULO DE

**INGENIERO ELECTRÓNICO**

GUATEMALA, FEBRERO 2023

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA



**NÓMINA DE JUNTA DIRECTIVA**

DECANA	Ing. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Kevin Vladimir Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIA	Ing. Hugo Humberto Rivera Pérez

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANA	Inga. Aurelia Anabela Cordova Estrada
EXAMINADOR	Ing. Cristian Orellana
EXAMINADOR	Ing. Sergio Gómez
EXAMINADOR	Ing. Mario Reyes
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

## **HONORABLE TRIBUNAL EXAMINADOR**

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

### **DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE SONIDOS MIDI A LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN UNA GUITARRA ELECTRICA**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería Mecánica Eléctrica, con fecha 13 de junio de 2022.

**Sergio Leonel Rodas Quiñonez**

Guatemala 15 de noviembre de 2022

Ingeniero  
Julio César Solares Peñate  
Coordinador de Área de Electrónica  
Escuela de Mecánica Eléctrica  
Facultad de Ingeniería  
Universidad de San Carlos de Guatemala

Señor Coordinador:

Por este medio le informo que he concluido con el asesoramiento y revisión del trabajo de graduación con título: **“DISEÑO DE SISTEMA ELECTRONICO QUE ASOCIE SONIDOS MIDI A LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN UNA GUITARRA ELECTRICA”**, desarrollado por el estudiante Sergio Leonel Rodas Quiñonez con carné 201314208.

Considero que este trabajo está bien desarrollado y representa un aporte a la Facultad de Ingeniería y habiendo cumplido con los objetivos del trabajo doy mi entera aprobación al mismo.

Atentamente,



Ing. Armando Alonso Rivera Carrillo

Armando Alonso Rivera Carrillo  
Ingeniero Electrónico  
Colegiado No. 4265

REF. EIME 76.2022.

El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto Bueno del Coordinador de Área , al trabajo de Graduación del estudiante Sergio Leonel Rodas Quiñonez: **“DISEÑO DE SISTEMA ELECTRONICO QUE ASOCIE SONIDOS MIDI A LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN UNA GUITARRA ELECTRICA”** . procede a la autorización del mismo.



Ing. Armando Alonso Rivera Carrillo

Guatemala, 18 de noviembre de 2022.

UNIVERSIDAD DE SAN CARLOS  
DE GUATEMALA



FACULTAD DE INGENIERIA

Guatemala, 16 de noviembre de 2022

**Señor director**  
**Armando Alonso Rivera Carrillo**  
**Escuela de Ingeniería Mecánica Eléctrica**  
**Facultad de Ingeniería, USAC**

Estimado Señor director:

Por este medio me permito dar aprobación al Trabajo de Graduación titulado: **DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE SONIDOS MIDI A LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN UNA GUITARRA ELECTRICA**, desarrollado por el estudiante **Sergio Leonel Rodas Quiñonez**, ya que considero que cumple con los requisitos establecidos.

Sin otro particular, aprovecho la oportunidad para saludarlo.

Atentamente,

**ID Y ENSEÑAD A TODOS**

Una firma manuscrita en tinta azul que parece decir "Julio César Solares Peñate".

**Ing. Julio César Solares Peñate**  
**Coordinador de Electrónica**



Decanato  
Facultad de Ingeniería  
24189101- 24189102  
secretariadecanato@ingenieria.usac.edu.gt

LNG.DECANATO.OI.186.2023

La Decana de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica, al Trabajo de Graduación titulado: **DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE SONIDOS MIDI A LA EJECUCIÓN DE UNA NOTA AL PULSAR UNA CUERDA EN UNA GUITARRA ELÉCTRICA**, presentado por: **Sergio Leonel Rodas Quiñonez**, después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, **autoriza la impresión del mismo.**

IMPRÍMASE:

  
Inga. Aurelia Anabela Cordova Estrada  
Decana



Guatemala, febrero de 2023

AACE/gaoc



## **ACTO QUE DEDICO A:**

- Dios** Por sus bendiciones y orientación en mi carrera universitaria.
- Mis padres** Marcos Rodas y Mercedes Quiñonez por apoyarme incondicionalmente y enseñarme a aplicar los conocimientos en las situaciones requeridas.
- Mis hermanos** Rita y Marcos Rodas por su forma de enseñarme los distintos caminos y ayudarme a caminar en ellos.
- Mis amigos** Moisés Quiroa, José Archila, Edson Valdez, Herbert Arguijo y Miguel Villatoro por su apoyo moral, académico, de vida y los momentos que hemos vivido desde el colegio hasta esta culminación de estudios universitarios.

## **AGRADECIMIENTOS A:**

<b>Universidad de San Carlos de Guatemala</b>	Por formarme profesionalmente, por enseñarme a valorar la educación y orientarme en la resolución de las distintas situaciones.
<b>Facultad de Ingeniería</b>	Por ser la facultad que me aceptó y ayudo en mi formación académica y buen desempeño laboral.
<b>Lic. Erasmo Morales</b>	Por brindarme su amistad, conocimiento, apoyo moral y enseñanza laboral.
<b>José Archila</b>	Por brindarme su apoyo en los momentos en los que se necesitaba una mano amiga.
<b>Daniela Molina</b>	Por apoyarme en los detalles mas pequeños pero importantes en mi carrera y vida.
<b>Diana Carias</b>	Por su tiempo compartido y brindarme motivación y apoyo incondicional en los momentos cruciales de mi vida.

# ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
LISTA DE SÍMBOLOS.....	VII
GLOSARIO.....	IX
RESUMEN.....	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN.....	XV
1. SISTEMAS DE DETERMINACIÓN DE EJECUCIÓN.....	1
1.1. Sistemas de determinación de ejecución de notas musicales .....	1
1.1.1. Frecuencia a la que resuena una nota.....	1
1.1.2. Señales analógicas y discretas.....	3
1.1.3. Conversión de señal analógica a señal Discreta.....	4
1.1.4. Filtración de señal.....	5
1.1.5. Procesamiento de señales discretas.....	8
1.2. Ejemplos de sistemas actuales de determinación de notas.....	9
1.2.1. Afinadores análogos de guitarra.....	10
1.2.2. Afinadores digitales de guitarra.....	11
2. PROCESOS ELECTRÓNICOS PARA ASOCIAR SEÑAL MIDI A LA PULSACIÓN DE UNA CUERDA EN UNA GUITARRA ELÉCTRICA.....	13
2.1. Procesos de análisis de señales.....	13
2.1.1. Procesamiento de la señal de guitarra eléctrica.....	14
2.1.2. Acondicionamiento de señal de guitarra eléctrica para su procesamiento.....	15

2.2.	Procesamiento de señal eléctrica .....	19
2.2.1.	Captación de la señal de guitarra eléctrica directamente a la salida del conector de ¼” .....	19
2.2.2.	Acondicionamiento de la señal eléctrica para su procesamiento .....	20
2.3.	Filtración de señales .....	21
2.3.1.	Transformada de Fourier .....	22
2.3.2.	Análisis de señales en el dominio de la frecuencia .....	23
2.3.3.	Procesamiento digital de señales discretas.....	24
2.4.	Toma de decisiones del sistema .....	25
2.4.1.	Toma de decisiones entorno a la frecuencia de las señales .....	25
2.4.2.	Comunicación del sistema con instrumentos que hagan uso del protocolo MIDI .....	26
3.	PROPUESTA DE DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE NOTAS MIDI A LA EJECUCIÓN DE UNA NOTA EN UNA CUIERDA DE GUITARRA ELÉCTRICA.....	29
3.1.	Descripción del diseño.....	29
3.2.	Análisis del circuito de captación y acondicionamiento de señal eléctrica .....	30
3.3.	Análisis de la tarjeta de procesamiento de señales con mejores prestaciones calidad – precio .....	33
3.4.	Evaluación del diseño del sistema electrónico que asocie notas MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica .....	34
3.5.	Análisis de la asociación de notas MIDI a la pulsación de la cuerda de una guitarra eléctrica .....	38

4.	ANÁLISIS TECNICO-ECONOMICO DEL SISTEMA .....	43
4.1.	Costos generales de la implementación técnica del sistema ...	43
4.2.	Costo de materiales del procesamiento del procesamiento de señal.....	43
4.2.1.	Circuito de adquisición de datos .....	43
4.2.2.	Costos de tarjeta electrónica y componentes auxiliares para el procesamiento de la señal eléctrica .....	47
4.3.	Costos de horas de trabajo .....	49
4.4.	Costo final del diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica .....	51
	CONCLUSIONES .....	55
	RECOMENDACIONES.....	57
	REFERENCIAS .....	59
	APÉNDICES.....	61



# ÍNDICE DE ILUSTRACIONES

## FIGURAS

1.	Diagrama del filtro de paso de bajas frecuencias .....	16
2.	Comportamiento del filtro de paso de bajas frecuencias.....	17
3.	Diagrama del filtro de paso de altas frecuencias .....	18
4.	Filtro de paso de altas frecuencias .....	18
5.	Diagrama del circuito que acondiciona la señal de guitarra eléctrica.....	31
6.	Diagrama de la salida de señal del IC LM 386 amplificada.....	32
7.	Diagrama del acople de la salida de la señal amplificada hacia la tarjeta electrónica programable .....	33
8.	Conversión de señal en el dominio de tiempo a señal en dominio de frecuencia.....	36
9.	Asociación de notas en dominio de la frecuencia a notas MIDI.....	40

## TABLAS

I.	Costos del circuito de adquisición de datos.....	44
II.	Costos de tarjeta electrónica y componentes auxiliares para el procesamiento de la señal eléctrica .....	48
III.	Costos de horas de trabajo .....	49
IV.	Costo final del diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar cuerda en una guitarra eléctrica ....	51





## LISTA DE SÍMBOLOS

<b>Símbolo</b>	<b>Significado</b>
<b>dB</b>	Decibel
<b>F</b>	Faradios
<b>Hz</b>	Hertz
<b>IEM</b>	Interferencia electromagnética
<b>MHz</b>	Mega Hertz
$\mu\text{F}$	Microfaradios
$\Omega$	Ohmios



## GLOSARIO

<b>Circuito Integrado</b>	Chip electrónico que se encapsula de forma estandarizada y con diferentes lógicas de funcionamiento.
<b>Conector de 1/4"</b>	Es un conector universal que sirve para la salida de la señal de guitarra eléctrica.
<b>Filtro RLC</b>	Es un tipo de filtro que utiliza los principales componentes pasivos tales como resistencias, inductancias y capacitancias.
<b>MIDI</b>	Protocolo de comunicación que permite la asociación de notas ejecutadas en la guitarra al lenguaje de comunicación entre instrumentos.
<b>Nyquist-Shannon</b>	El criterio de Nyquist-Shannon es el que se utiliza para muestrear cualquier señal continua por lo menos al doble de la frecuencia para evitar pérdida de información en la señal muestreada.
<b>Transientes</b>	Es un evento que ocurre en las señales eléctricas en el cual suceden fluctuaciones de energía que sobrepasan la intensidad de señal que detecta la señal de guitarra al pulsarse la cuerda.



## RESUMEN

En el presente trabajo de graduación se desarrollan los conceptos teóricos para diseñar un sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica incluyendo los conceptos técnicos, informáticos y prácticos para el diseño del sistema.

El primer capítulo se analizan los sistemas de determinación de notas existentes para la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica y los conceptos básicos para el entendimiento de este procesamiento de señal.

En el segundo capítulo se aborda el análisis técnico y teórico de los procesos electrónicos para asociar señal MIDI a la pulsación de una cuerda en una guitarra eléctrica.

En el tercer capítulo se presenta la propuesta de diseño de un sistema electrónico fiable en base al procesamiento discreto de la señal analógica a la salida de la guitarra, es decir mientras menos componentes tenga el sistema, menor tendencia a la falla existirá.

En el cuarto capítulo se analizan los costos técnicos-económicos del diseño del sistema electrónico.



# OBJETIVOS

## General

Diseñar un sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica.

## Específicos

1. Analizar los sistemas de determinación existentes para la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica.
2. Analizar los procesos electrónicos para asociar señal MIDI a la pulsación de una cuerda en una guitarra eléctrica.
3. Presentar la propuesta de diseño de un sistema electrónico en base al procesamiento discreto de las señales analógicas.
4. Realizar análisis técnico económico del sistema electrónico.





## INTRODUCCIÓN

Para el desarrollo del tema se propone el diseño de un sistema electrónico de detección de señal eléctrica directamente desde el conector de salida de una guitarra eléctrica, a partir de esta adquisición de datos, se procede al acondicionamiento de señal, filtración y tarjeta de procesamiento de señal. Por lo cual, se realizará la conversión de señal analógica a señal digital, permitiendo la manipulación de datos sin problemas en la tarjeta de procesamiento de señal.

Posteriormente se procederá con la filtración de señales por medio de la transformada rápida de Fourier y la programación lógica de toma de decisiones detectando la frecuencia de mayor presencia en la señal al momento de ejecutar la pulsación de una cuerda en la guitarra eléctrica.

De lo anterior, para ejecutar en el instrumento final el acorde que corresponde a la filtración de señal, se realizará el análisis correspondiente y el respectivo diseño del sistema electrónico que utilizará la tarjeta de procesamiento de señal y sus respectivas ventajas al configurarse como medio de programación y toma de decisiones, así como, de comunicación entre la guitarra y un instrumento que utilice el protocolo MIDI.



# **1. SISTEMAS DE DETERMINACIÓN DE EJECUCIÓN**

## **1.1. Sistemas de determinación de ejecución de notas musicales**

En la actualidad, existen distintos sistemas de determinación de ejecución de notas musicales, desde sencillos sistemas electrónicos hasta complejos sistemas de software. Cabe resaltar que el avance de la tecnología facilita la precisión en la captación de las notas que se ejecutan.

Actualmente existen afinadores de guitarra, que permiten la detección de la frecuencia fundamental que provoca el sonido. Por medio de la filtración de señal, se identifica la frecuencia con mayor presencia en la ejecución de la nota al provocar vibración sobre dos puntos fijos en el instrumento.

También existen complejos sistemas de determinación de notas que involucran procesamiento digital de señal y frecuencias de muestreo en la señal de alta calidad lo que permite gran fidelidad en su desempeño.

### **1.1.1. Frecuencia a la que resuena una nota**

Sears y Semansky (2009) expresan que una cuerda tensa fija entre dos extremos no puede vibrar libremente, lo que da paso al comportamiento de la onda que se produce en la vibración como onda estacionaria, este principio físico indica que las cuerdas que se encuentran bajo tensión vibren en frecuencias determinadas llamadas armónicos, cada armónico es un factor  $n$  multiplicado por la frecuencia fundamental a la que está vibrando la cuerda tensa.

Cuando se realiza la pulsación en una cuerda de guitarra eléctrica todos los armónicos vibran a la misma vez con un rango de distinta amplitud para cada armónico, lo que permite que la frecuencia con mayor presencia, es decir, con la mayor amplitud de señal, será la frecuencia que presente el fenómeno conocido como resonancia.

Resonancia en términos eléctricos, es el fenómeno electromagnético en el cual las señales eléctricas presentan su mayor amplitud en una o varias frecuencias, estas frecuencias provocan que la impedancia en el circuito eléctrico sea equivalente a cero, es decir las admitancias de los componentes del circuito se eliminan, a estas frecuencias se les conoce como frecuencias de resonancia. (Boylestad, 2004)

Las notas musicales en una guitarra por naturaleza siempre resuenan en una frecuencia fundamental y en varios de sus respectivos armónicos que acompañan a la señal, estos armónicos suenan regularmente con menor amplitud que la nota fundamental.

En general las frecuencias de resonancia provocan que se amplifique el volumen de los sonidos que se ejecutan en la pulsación de las cuerdas de guitarra.

De la misma manera en las guitarras eléctricas a la hora de ejecutar una pulsación de una nota en una cuerda, la frecuencia de resonancia será la que tenga mayor amplitud.

### **1.1.2. Señales analógicas y discretas**

En la actualidad la mejor forma de manipular las señales (acústicas, infrarrojas, de audio, video, entre otros) es la utilización de conversiones de señales analógicas a señales digitales y viceversa, de digitales a analógicas, en esta ocasión, se propiciará el uso de la conversión de señal producida por la vibración de la cuerda tensa entre 2 puntos fijos a señal eléctrica y también la conversión de señal eléctrica a señal discreta con la finalidad de procesar la nota ejecutada y asociarla a una o más notas en instrumentos que puedan utilizar el protocolo de comunicación MIDI.

Se analizan este tipo de señales debido a que, el desarrollo de este trabajo depende del entendimiento de los conceptos y procesamiento de estas.

La puesta en práctica de conocimientos de procesamiento de señales permite el análisis con precisión, evitando así, el desperdicio de recursos físicos para el diseño del sistema electrónico que detecte la correcta ejecución de la nota.

- Señal eléctrica: señal que utiliza el fenómeno del movimiento de electrones en la corriente alterna (CA) como medio de transmisión o flujo.
- Señal analógica: señal que en el dominio del tiempo permanece continua y puede ser captada y detectada en cualquier instante y representarse sin ningún fenómeno de alteración o filtro en su transmisión, esto incluye el ruido presente en la señal.
- Señal discreta: señal que utiliza la electrónica digital y que dependiendo de su frecuencia de muestreo y la fidelidad del medio (circuitos

electrónicos encargados de la conversión de señal analógica a señal digital) se pueda procesar de manera discreta y en el dominio de la frecuencia

### **1.1.3. Conversión de señal analógica a señal Discreta**

Para el procesamiento de la señal de la guitarra eléctrica, se analiza la transducción de señal, así como su acondicionamiento a señal eléctrica y su transformación a señal discreta.

La señal eléctrica es la señal captada directamente desde la variación del campo electromagnético al ejecutar la pulsación en la cuerda de la guitarra eléctrica, esta señal se puede captar utilizando un medio de transducción siendo el mismo una pastilla de guitarra que a la salida proporciona señal eléctrica, esta señal se procesa utilizando un preamplificador electrónico basado en un circuito integrado LM386 escogido por su relativo bajo coste y aceptable amplificación sin distorsión de señal.

La pastilla con patrón de captación cardioide directamente debajo de las cuerdas de la guitarra detecta la variación de campos electromagnéticos inducidos que genera la pulsación de las cuerdas de la guitarra, esta variación de campos electromagnéticos detectados por la pastilla se convierten en señal eléctrica a través del principio de transducción de onda electromagnética, es decir, la pastilla de la guitarra actúa como transductor de señal eléctrica, luego se introduce a un circuito que utiliza al integrado LM386 como componente principal para amplificar la pequeña señal que proviene de la pastilla.

Utilizando nociones básicas de circuitos de pequeña señal, se analiza el circuito integrado LM386 como un amplificador operacional dedicado al audio.

El comportamiento de la señal a la salida del circuito electrónico que se encarga de la amplificación se analiza utilizando los conceptos de ganancia del amplificador operacional. (Boylestad, 2004)

La ganancia de este circuito integrado se puede regular utilizando una resistencia entre la patilla 1 y 8 del mismo, pero es de preferencia utilizar la configuración por defecto del chip siendo la misma cerca de los 10 dB.

Se utiliza el integrado LM386 en una configuración que tenga como máximo 10 dB de ganancia con el objetivo de no provocar saturación por amplificación en nuestra señal, con este pequeño circuito se realiza la adquisición de datos

#### **1.1.4. Filtración de señal**

La filtración de señal consiste en el uso de la frecuencia como medio de detección de la señal, la señal eléctrica viaja a través de los distintos circuitos electrónicos a cierta frecuencia lo que permite discernir que nota se está ejecutando.

Se analiza la señal en el dominio de la frecuencia como referencia de la variación de amplitud dependiendo de la frecuencia con mayor presencia en la señal.

Se utiliza el concepto de señal en el dominio de la frecuencia para poder realizar la respectiva filtración, utilizando circuitos electrónicos que permiten la adquisición de señal, procesamiento analógico de filtración de señales no deseadas como el ruido, interferencias electromagnéticas, o alguna otra perturbación que afecte el curso de la señal a través del tiempo.

Para obtener un discernimiento de la señal en curso emitida por el instrumento, se analizan las frecuencias presentes al ejecutar una nota pulsando una cuerda en la guitarra. Para la existencia de la detección en la ejecución de una nota, la guitarra eléctrica tiene que estar en afinación estándar, ya que la mayoría de los guitarristas utilizan este tipo de afinación siendo la convención a 440Hz.

Según la norma ISO 16 (ISO, 1975) La referencia para la afinación estándar de los instrumentos musicales se basa en la nota musical La a 440 Hz.

Desde que se decretó la norma ISO 16, 440hz ha sido la frecuencia estándar para la afinación de todos los instrumentos musicales como: pianos, cellos, violines, entre otros.

Una vez obtenida la afinación estándar en la guitarra eléctrica y conociendo el comportamiento de las ondas sobre las cuerdas tensas, se pueden analizar las notas musicales como señales en el dominio de la frecuencia, utilizando este parámetro se puede realizar filtración de señal.

La filtración de señal ocurre en el circuito electrónico conformado por el integrado LM386, un capacitor electrolítico de 100 microfaradios para filtrar el ruido de la fuente de alimentación del circuito, otro capacitor de 333 nano faradios en serie con un potenciómetro de 15 kilo Ohmios para acople de señal de la entrada de la guitarra al circuito integrado y otro capacitor de 9 nano faradios para filtración en conjunto con una resistencia de 15 ohmios para una salida de señal con frecuencias que se encuentren en el rango de las que alcanzan las notas en la guitarra con algunos de sus armónicos que mayor amplitud tengan en la ejecución de la nota.



Se utiliza un capacitor electrolítico de 220 microfaradios a la salida de todo el circuito con el objetivo de acoplar la señal a la entrada de la tarjeta de programación Arduino Mega.

El circuito se encarga de filtrar las señales indeseadas tales como ruidos causados por la naturaleza de las ondas electromagnéticas y sus medios de transmisión, cables, calidad de la pastilla y componentes internos del circuito electrónico encargado del acondicionamiento de señal.

Los transientes son sonidos de alta amplitud y corta duración en la ejecución de la nota al pulsar una cuerda, generalmente cuando se ejecutan las pulsaciones de cuerdas en una guitarra, se escucha el ataque, el decaimiento, el sustain y la extinción de la nota, en este caso los transientes ocurren en el ataque, cada vez que se pulsan las cuerdas de la guitarra, por tanto, el pico de señal alcanzado por la pulsación de la cuerda introduce ruido en la señal que es inevitable debido a la naturaleza de la forma de ejecución de notas en una guitarra.

Es de gran importancia entender el comportamiento de los transientes ya que dependiendo de la amplitud que tenga esta pulsación, por un pequeño instante de tiempo el volumen del transiente es mayor al volumen de la señal por lo que puede causar interferencia en la filtración y detección de esta.

Ya teniendo los conceptos anteriores, podemos decir que, para la filtración de señal de nuestra guitarra captada directamente del conector de salida, se puede aplicar un filtro con componentes electrónicos utilizando como componente principal el circuito integrado LM386, una vez se utiliza este filtro, se puede acondicionar la señal eléctrica con una relación de señal a ruido en la cual la señal es mucho mayor que el ruido de fondo.

Con la señal eléctrica ya acondicionada se puede abordar el desarrollo del procesamiento de señales discretas.

### **1.1.5. Procesamiento de señales discretas**

Según Pérez Vega et. al. (2007) la relación señal a ruido se define como el cociente de la potencia de la señal entre la potencia de ruido en un punto dado de un sistema y proporciona una medida de la calidad de la señal.

La relación señal a ruido es un parámetro que nos indica cuanta amplitud tiene la señal que se analiza respecto a la amplitud que presenta el ruido derivado a fenómenos electromagnéticos.

El ruido es una señal que se puede percibir en cualquier fenómeno que utilice ondas electromagnéticas como transmisión o transducción en el cual la onda se comporta aleatoriamente y en el ámbito del audio lo podemos percibir con mayor o menor grado de amplitud dependiendo de la relación señal a ruido que tenga la señal que estemos analizando.

Se analiza el parámetro de la relación señal a ruido y su respectivo comportamiento en las señales de audio, porque es importante a la hora de aplicar el procesamiento de la señal de audio a señal en el dominio de la frecuencia.

Para realizar la conversión de la señal de audio a señal discreta se utiliza una tarjeta electrónica de programación Arduino MEGA debido a que, dada su frecuencia de ciclos de trabajo, la tarjeta Arduino puede muestrear nuestra señal sin ningún problema al doble de la frecuencia que tiene la señal eléctrica de la guitarra en su espectro electromagnético.

Según el criterio de Nyquist-Shannon (2008), para que una señal sea muestreada correctamente, se debe de muestrear como mínimo al doble de la frecuencia de la señal que se está muestreando.

Esta frecuencia de muestreo está limitada por la capacidad de la tarjeta de procesamiento de señal Arduino MEGA, en este caso sin ningún problema se puede muestrear a través del puerto analógico que se utilice del Arduino que directamente puede detectar señales analógicas ya que trae sus conversores de señal analógica a digital internos.

Una vez se ingresa nuestra señal eléctrica en el puerto analógico del Arduino MEGA, se puede hacer uso de la transformada rápida de Fourier como método de conversión de señal eléctrica a señal discreta.

Según Yadav (2008) la transformada de Fourier es el método matemático que a través de series trigonométricas (senos y cosenos) puede tomar una señal analógica y analizarla en el dominio de la frecuencia en vez del tiempo, por lo cual la señal eléctrica en el dominio del tiempo se convierte en la señal discreta y digital en el dominio de la frecuencia.

La señal digital se procesa a través de programación que realiza la detección de la frecuencia y su respectivo procesamiento de la señal de forma discreta que asigna para valores en la detección.

## **1.2. Ejemplos de sistemas actuales de determinación de notas**

Actualmente para detectar las notas de una guitarra existen afinadores que detectan la pulsación sobre una cuerda de guitarra con la capacidad de dar parámetros para indicar al usuario si la cuerda se encuentra afinada o no, pero

estos afinadores no pueden asociar notas MIDI en la ejecución de una nota al pulsar una cuerda en el instrumento y procederá a realizar su función de indicar la afinación que tenga mayor presencia en la nota entre todos sus armónicos.

También existen sistemas que hacen uso del procesamiento digital de señal, pero estos resultan bastante caros y necesitan de recursos tanto de hardware como de software de una computadora para poder ejecutarse y utilizar muy bien sus características tales como la precisión en la detección de la nota, el muestreo a una resolución bastante alta con el objetivo de tener la mayor calidad posible en la adquisición de datos de la señal, por lo que resultan costosos y muy difíciles de acceder tomando en cuenta la media de ingresos de los ciudadanos en Guatemala.

### **1.2.1. Afinadores análogos de guitarra**

La tecnología analógica es muy utilizada para la adquisición de datos en los instrumentos musicales, dado a que en el dominio del tiempo se necesitan captar todos los matices, sonidos, armónicos y otras características como el ataque, el sustain, el decaimiento y la extinción en una nota, por lo que es necesario analizar las señales provenientes de los distintos instrumentos como señales continuas en el tiempo, esto con el fin de no perder ningún matiz ni calidad sonora que provenga del instrumento que se está ejecutando.

En la electrónica se considera tecnología analógica a todos los dispositivos que utilicen la adquisición de datos y procesamiento de señal de forma continua, es decir conforme varía la señal en el dominio del tiempo.

Un afinador analógico de guitarra es un dispositivo que utiliza electrónica analógica para captar y procesar la señal, en este caso, estos dispositivos están

diseñados para detectar la frecuencia que mayor presencia tenga al ejecutarse una cuerda de guitarra al aire indicando al usuario que tan lejos o cerca está la cuerda de estar afinada, estos dispositivos hacen uso del estándar de afinación en A2 a 440Hz.

### **1.2.2. Afinadores digitales de guitarra**

Los afinadores digitales de guitarra son aquellos que utilizan el procesamiento de la señal de guitarra de forma discreta, es decir no utilizan directamente la señal analógica para realizar la respectiva discriminación de información, sino que utiliza sus conversores de señal analógica a señal digital para poder interpretar la señal de la guitarra de forma discreta.

Los afinadores digitales utilizan el procesamiento digital de señal para realizar la discriminación de frecuencias, es decir a través de programación de compuertas lógicas en dispositivos conocidos como microcontroladores, se pueden procesar las señales que provengan de una guitarra y por este medio realizar la filtración de señal.

La filtración de señal consiste en el procesamiento de señales discretas, es decir, pulsos en nivel alto o nivel bajo que utilizando a los bits como forma de análisis de información que permite discriminar la señal, por lo que se asocian ciertos parámetros a la hora de realizar la filtración, para cada valor discreto de detección existe otro valor discreto de acción en la programación electrónica y configuración de los distintos tipos de afinadores digitales y sus formas de filtración.

Una vez realizado el procesamiento de la señal con todos los recursos de hardware y software que se involucran para poder tener una filtración de señal

que sea útil al software de los afinadores digitales, se puede mostrar la nota de la cuerda al aire libre en una pantalla o cualquier otro dispositivo de visualización.

## **2. PROCESOS ELECTRÓNICOS PARA ASOCIAR SEÑAL MIDI A LA PULSACIÓN DE UNA CUERDA EN UNA GUITARRA ELÉCTRICA**

### **2.1. Procesos de análisis de señales**

Se analizarán las etapas de las señales, así como su conversión, transducción y manipulación como procesos, esto con el fin de llevar un orden entre etapas, así como mayor facilidad de explicación y desarrollo de la tesis.

En este trabajo se analiza la captación directa de la señal de guitarra eléctrica, por lo tanto, se enfocará en el procesamiento de la señal eléctrica y la señal discreta.

La señal analógica será nuestra señal captada directamente desde la salida del conector de la guitarra eléctrica y se utilizará una pastilla que funciona como un transductor de señal, es decir la variación de campos electromagnéticos al ejecutar la pulsación que induce en la pastilla será la señal eléctrica.

Estos procesos serán analizados con su respectiva electrónica pertinente, es decir el circuito electrónico que utiliza el IC amplificador de bajo ruido LM386, luego se utilizará la tarjeta de programación Arduino MEGA encargada de realizar la conversión de señal eléctrica a señal discreta en el dominio de la frecuencia y su respectiva filtración usando transformada rápida de Fourier.

### **2.1.1. Procesamiento de la señal de guitarra eléctrica**

La señal de la guitarra eléctrica en este caso es la señal audible que el humano puede detectar debido al cambio de presión que existe en el ambiente y es transducida por medio de pastillas hacia el conector de salida. Para Vargas y Burgos (2016) La señal de la guitarra eléctrica resulta de las sumas de las ondas viajeras en toda la extensión de la cuerda, siendo esta la consecuencia de la pulsación de esta.

La señal de guitarra eléctrica que es captada presenta los fenómenos del ataque, sustain, decaimiento y muerte de las notas ejecutadas en una guitarra al pulsar las cuerdas tensas entre el puente y las clavijas.

La pulsación de la cuerda para poder analizarse en términos de captación de notas y para asociar sus respectivas frecuencias a notas MIDI, debe de convertirse a señal eléctrica y la misma debe de convertirse a señal digital, por lo cual existen procesos por los cuales la señal eléctrica es manipulada por electrónica y el software de apoyo que permite convertir la señal análoga en un pulso con cierta frecuencia, de este modo, al obtener la frecuencia que mayor presencia tenga en la pulsación de la cuerda, se pueden asociar una o más notas MIDI que representen la pulsación de la nota.

Para esta transducción de señal, se utilizan las pastillas de la guitarra eléctrica aprovechando el efecto de variación de campos magnéticos que inducen señales electromagnéticas en la pastilla.

Esta forma de captación de señal eléctrica así como la fidelidad y calidad también depende en gran medida de la pastilla con la cual se va a realizar el proceso de captación, en este caso y para nuestro análisis, se utilizará una



pastilla de fabricación americana de alta calidad, cabe mencionar que aunque sean de alta calidad, en las pastillas siempre se introducirá ruido en la señal eléctrica debido a la naturaleza de las señales eléctricas, debido a la calidad de la pastilla, la relación señal a ruido será muy favorable, también se debe de tomar en cuenta el ruido que introducirán los componentes de bajo coste que se utilizan en el acondicionamiento y procesamiento de señal, el ruido es inevitable debido a la calidad de fabricación de estos componentes y este ruido también influirá en la calidad de sonido captado

### **2.1.2. Acondicionamiento de señal de guitarra eléctrica para su procesamiento**

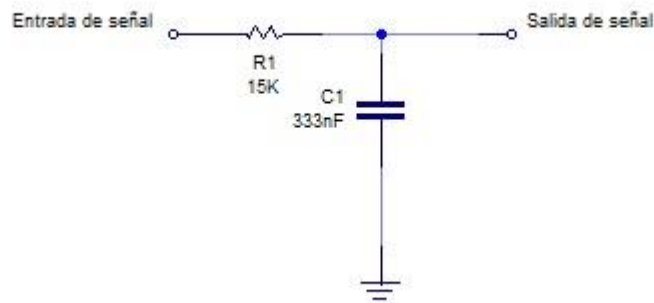
Para realizar la adquisición de datos de la señal eléctrica a la salida del conector se utiliza un circuito electrónico con filtro de entrada en configuración RC pasa bajas con el objetivo de eliminar el ruido de las señales de la red de alimentación y de bajas frecuencias. A. Holt (1989) explica que en el funcionamiento de un filtro de pasa baja (o de paso bajo) la banda transmitida se extiende desde cero, hasta una cierta frecuencia máxima.

Los filtros permiten la transmisión de una o más bandas de frecuencias, rechazando las señales que no se encuentren asociadas en la misma. Un filtro RC se puede configurar como filtro pasa altas o filtro pasa bajas, se propone escoger los valores de los componentes a usar, lo más sencillo es fijar el condensador, y escoger el valor de la resistencia resultado, siendo incluso posible usar resistencias variables para fijar el valor más exacto posible. También es posible usar condensadores variables, pero estos son menos habituales y más costosos.

Como se ha citado previamente, a la frecuencia de corte se produce la resonancia del filtro RC o RL o RLC, es decir, que los valores de la impedancia y de la reactancia capacitiva e inductiva a esa frecuencia se igualan.

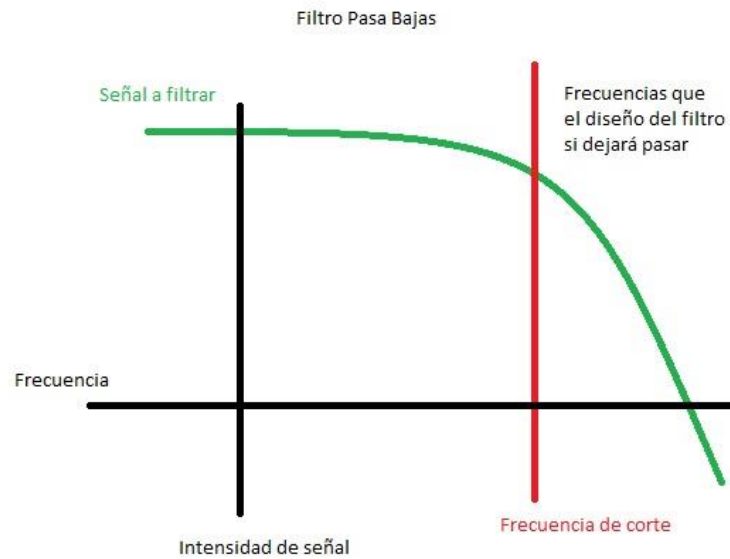
Si se trata de un filtro paso bajo, la resistencia irá en serie con la fuente de alimentación, mientras que el condensador en paralelo tanto con esta como con la carga.

Figura 1. **Diagrama del filtro de paso de bajas frecuencias**



Fuente: elaboración propia, realizado con Circuit Wizard.

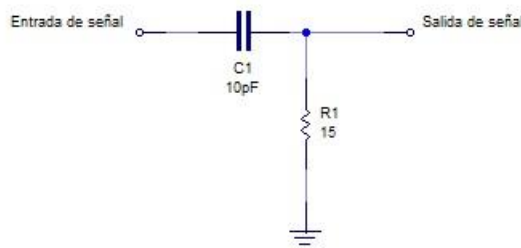
Figura 2. **Comportamiento del filtro de paso de bajas frecuencias**



Fuente: elaboración propia, realizado con Circuit Wizard.

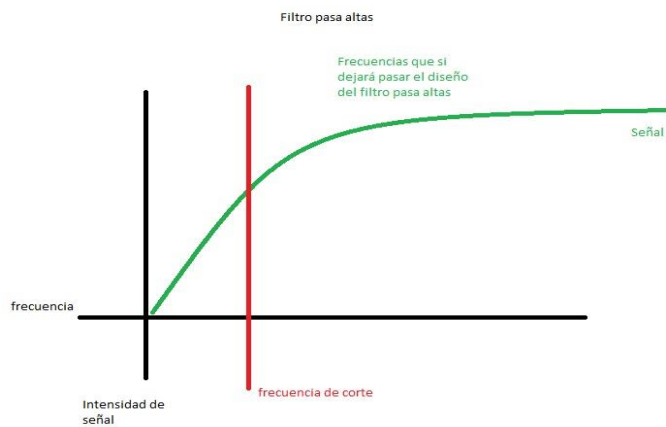
El filtro de paso de altas frecuencias realiza su respectiva atenuación para frecuencias debajo de la frecuencia de corte.

Figura 3. Diagrama del filtro de paso de altas frecuencias



Fuente: elaboración propia, realizado con Circuit Wizard.

Figura 4. Filtro de paso de altas frecuencias



Fuente: elaboración propia, realizado con Circuit Wizard.

Los filtros tipo pasa altas y pasa bajas son el ejemplo básico del funcionamiento de la filtración de señal, en nuestro caso, acondicionamos la señal de la guitarra a través de una resistencia variable y un capacitor conectado

directamente a la entrada no inversora del IC LM386 que funcionan como atenuadores de señal con el objetivo de evitar clípeo y distorsión en la adquisición de datos.

Así mismo, a la salida del amplificador operacional realizamos la respectiva filtración de ruido generado por el IC así como el acople de salida de señal para conectar el circuito de adquisición de datos hacia el pin del Microcontrolador Arduino para su procesamiento de señal.

## **2.2. Procesamiento de señal eléctrica**

El procesamiento es realizado por la electrónica interna de la guitarra y la calidad de este depende de los componentes utilizados, por lo cual también será un factor importante en el análisis del procesamiento de la señal eléctrica, además la fiabilidad de la señal depende de la cantidad de componentes utilizados, mientras más componentes utilice el circuito, mayor tendencia a la falla tendrá el sistema de captación de la señal.

Es por eso por lo que se plantea un procesamiento que utilice la menor cantidad de componentes posibles, esta señal eléctrica se acondicionará a través de un circuito basado en el LM386 con ganancia de 10db con el objetivo de no saturar la señal y distorsionar la frecuencia fundamental en la cual resuena la pulsación de la cuerda de guitarra y algunos de sus armónicos.

### **2.2.1. Captación de la señal de guitarra eléctrica directamente a la salida del conector de ¼"**

Luego de la inducción electromagnética generada debido al cambio de campos eléctricos en la pulsación de una cuerda tensa entre 2 extremos y la

pastilla magnética, se puede manipular la intensidad de la señal a la salida del Jack de una guitarra eléctrica, así mismo, el control de tonos y volumen.

### **2.2.2. Acondicionamiento de la señal eléctrica para su procesamiento**

Para poder analizar los cambios de voltaje en la salida del Jack de la guitarra eléctrica es necesario realizar pruebas para cuantificar el volumen de la señal, es decir si la señal presenta valores de voltaje que no necesiten ser amplificados.

Para tener la señal eléctrica entre los parámetros que no saturen y generen clípeo es decir recorte de señal, es necesario configurar la ganancia de nuestro amplificador de audio o configuración de la amplificación de la señal a través del IC LM386.

La configuración de la ganancia debe ser acorde al cambio de voltaje que genere una guitarra eléctrica en su Jack de  $\frac{1}{4}$  de pulgada.

Esta configuración se consigue de igual forma que la configuración utilizada para el análisis de procesamiento de señal eléctrica, es decir, el circuito electrónico que toma como base al circuito integrado LM386.

El circuito electrónico utiliza filtración de pasa bajos para evitar ruido proveniente de la red de alimentación a bajas frecuencias, el amplificador de audio de bajo ruido para pequeña señal LM386 y el filtro pasa altas en la salida de la señal del IC para filtrar el ruido a altas frecuencias generados por la electrónica interna del chip.

### **2.3. Filtración de señales**

Hermosa (2012) explica que el filtrado de señales es conocido como ruido eléctrico indeseable, este se puede propagar por el aire o por una red electromagnética, y que se controla mediante condensadores debidamente conectados y a veces acompañados de bobinas.

La filtración es el método utilizado para separar datos, en este caso la señal eléctrica que se analiza tiene que ser separada del ruido que se introduce, como ya se ha explicado, el ruido que se introduce en la señal proviene de muchas fuentes externas.

Principalmente se utiliza la filtración en la etapa de adquisición de datos de la señal eléctrica, esta filtración es el acople de la señal de la guitarra eléctrica a la entrada no inversora del IC LM386. Posteriormente se utiliza un capacitor electrolítico para eliminar el ruido proveniente de la fuente de alimentación del circuito.

Luego, se procede con la filtración de la señal a la salida del circuito que amplifica nuestra señal para poder ser procesada y analizada por nuestra tarjeta de programación Arduino MEGA.

En estas filtraciones anteriores, se utilizan los filtros para eliminar ruidos y acondicionar la señal eléctrica para que se encuentre en el rango de las frecuencias que se generan al pulsar las cuerdas de la guitarra.

A la entrada del microcontrolador Arduino MEGA en su pin de recepción de señal analógica la señal eléctrica debe de llegar amplificada, es decir no se puede conectar el micrófono directamente al pin de entrada del microcontrolador

tiene que ser acondicionada la señal, es por eso que se utiliza el circuito electrónico que tiene resistencias y capacitores que forman 3 filtros pasivos y 1 chip LM386.

Para analizar la señal en el microcontrolador se utiliza la filtración que separa la señal de la entrada del pin por medio de su frecuencia que tenga mayor presencia al ejecutarse las notas en la guitarra, por lo cual se utilizan herramientas de programación en el microcontrolador que nos apoyan con la separación de frecuencias.

### **2.3.1. Transformada de Fourier**

Según J. Bobadilla, P. Gomez, & J. Bernal. (1999), la transformada de Fourier (TF) es considerado un método para realizar el traspaso de las señales en el dominio del tiempo al dominio de la frecuencia.

Para comprender el fruncimiento de la conversión de señal en el dominio del tiempo a señal en el dominio de la frecuencia, se utilizan los conceptos de Señales discretas, transformada de Fourier, transformada discreta de Fourier y transformada rápida de Fourier.

Transformada de Fourier: algoritmo utilizado para cuantizar señales continuas en el dominio del tiempo y a través de su respectivo procesamiento digital convertir la señal analógica a señal digital en el dominio de la frecuencia.

Transformada discreta de Fourier: algoritmo que se utiliza para definir el muestreo de la señal acondicionada de la guitarra eléctrica, al definir el número de muestras y la frecuencia máxima que puede alcanzar el sistema de asociación de notas MIDI sin perjudicar la integridad de la señal.



Transformada rápida de Fourier: algoritmo utilizado para tomar en cuenta los valores reales integrables de la señal de guitarra y definir el rango de muestreo que permita utilizar de forma optimizada los recursos de la tarjeta de programación Arduino Mega.

Utilizando la transformada de Fourier se pueden descomponer los componentes espectrales de la frecuencia, en este caso, y para la aplicación del diseño del sistema de asociación de notas MIDI a la ejecución de una nota en una cuerda de guitarra eléctrica, usaremos la transformada rápida de Fourier con el objetivo de maximizar los recursos de la tarjeta de programación Arduino Mega, con esta maximización de recursos podemos obtener tiempos de respuesta mejor acoplados a la realidad, debido a que, al realizar el proceso completo y siguiendo al pie de la letra la metodología de Fourier es muy difícil obtener resultados favorables en el sistema de asociación de notas MIDI a la ejecución de una nota en una cuerda de guitarra eléctrica, ya que la tarjeta de programación enfocaría la mayoría de recursos en el muestreo y conversión de señal y el tiempo de respuesta tendría un retardo muy alto y disfuncional para el sistema.

### **2.3.2. Análisis de señales en el dominio de la frecuencia**

Para entender el análisis de la señal de la guitarra eléctrica acondicionada a la entrada del puerto A0 del Arduino Mega se debe de convertir al dominio de la frecuencia, la razón por la cual se realiza el proceso de conversión es debido a que el procesamiento digital de la señal es mejor si se cuantifican las notas ejecutadas en la guitarra y se asocian a esa nota detectada en la ejecución de la guitarra eléctrica.

La transformada rápida de Fourier utiliza menos recursos para descomponer una señal en sus componentes espectrales, este algoritmo utiliza

de forma optimizada la tasa de muestreo del sistema de asociación de notas MIDI se seleccionan 256.

El número seleccionado de muestras que se quieran hacer de la señal que entra en el puerto A0 del Arduino debe ser potencia de 2 a la n para que sea efectiva la transformada rápida de Fourier.

De lo anterior, la transformada discreta de Fourier nos permite reducir el tiempo de respuesta en ejecución de nuestro sistema y la transformada rápida de Fourier es la que nos permite la detección del pico de señal en el dominio de la frecuencia, es decir cuando la tarjeta Arduino capte la señal analógica en la entrada del puerto A0.

En la filtración de las señales para separar la amplitud de la señal por medio de frecuencias se utiliza la transformada de Fourier, que nos permite analizar una señal continua en el dominio del tiempo y convertirla en señal discreta en el dominio de la frecuencia, esto con el objetivo de identificar las notas en la guitarra y asociarle sus respectivas notas MIDI y ejecutarlas en el instrumento final.

### **2.3.3. Procesamiento digital de señales discretas**

Una vez se capta la frecuencia que mayor amplitud presente al ejecutarse una nota en la guitarra se procede con el procesamiento digital de esta señal, es decir a través del software que se utiliza para programar el microcontrolador Arduino MEGA se manipula la señal.

Esta manipulación de señal se conoce como procesamiento digital, debido a que la señal en el microcontrolador ya no se está analizando en el tiempo

continuo, sino en el dominio de la frecuencia y utilizando la lógica de muestreo de señal.

Para tener un procesamiento de señal aceptable, debe de muestrearse con la mayor cantidad de bits posibles, no se puede exceder de la capacidad del microcontrolador Arduino MEGA porque esto ralentizaría los procesos y el tiempo de respuesta en el dominio de la frecuencia.

Lo contrario a exceder la capacidad del microcontrolador para muestrear nuestra señal es la realización de este mismo proceso con una tasa muy baja de bits, dado a que no se tendría fidelidad en la conversión de datos analógicos a digitales, lo que representa un cálculo erróneo de la frecuencia, es decir nuestra digitalización de señal pierde fidelidad y no se realiza correctamente.

## **2.4. Toma de decisiones del sistema**

Para la toma de decisiones en el sistema, se utiliza la programación embebida en la tarjeta de programación Arduino, es decir para cada captación de nota y su respectiva frecuencia, se asociarán notas MIDI que serán capaces de ser ejecutadas al mismo tiempo utilizando lógica booleana, es decir, si se detecta la frecuencia entre cierto rango, se asociará directamente la emisión de notas MIDI que representarán acordes en este protocolo acondicionando la salida en conectores universales MIDI.

### **2.4.1. Toma de decisiones entorno a la frecuencia de las señales**

La filtración en frecuencia se utilizará para tener una base de comparación respecto a la señal ya procesada con la frecuencia que nuestra tarjeta de

programación tendrá configurada como parámetro al cual debe de obedecer como referencia para detectar a la nota ejecutada en la guitarra como correcta ejecución.

La toma de decisiones entorno a la frecuencia que la tarjeta de programación detecta será discreta, es decir se asociarán valores discretos a la señal que se detecta y se procesará de esta forma en la comparación de la programación.

La programación será en el entorno de desarrollo Arduino y la misma se encargará de realizar a una frecuencia de muestreo aceptable, la detección de la señal analógica que proviene de las captaciones de señal directamente de la salida del conector de guitarra.

Como se explicó anteriormente, la señal eléctrica que ingresa a la tarjeta de programación es introducida en el dominio del tiempo, por lo cual el análisis del cambio de señal en el dominio del tiempo al dominio de la frecuencia permite la manipulación de señal de forma discreta, en este caso para la toma de decisiones necesitamos la señal discreta en el dominio de la frecuencia por lo tanto todos los procesos de manipulación de señal deben de tener alta fidelidad, mientras menos componentes tenga el sistema de adquisición de datos, menor tendencia a la falla presentara el sistema electrónico que asocie sonidos MIDI.

#### **2.4.2. Comunicación del sistema con instrumentos que hagan uso del protocolo MIDI**

MIDI (Musical Instruments Digital Interfaz) es el lenguaje que utilizan actualmente muchos instrumentos para comunicarse entre ellos, enviar, recibir datos y sincronizarse. Nació dentro del mundo de los sintetizadores como

respuesta a una necesidad de los músicos: controlar varios equipos con sus dos manos y hacer capas de varios sonidos entre ellos. (Blanco, 2007)

El protocolo MIDI se puede analizar como el envío de los datos a la salida de nuestra tarjeta ARDUINO, estos datos viajan a través del conector acondicionado para el instrumento al que se vaya a conectar, algunos instrumentos usan el conector MIDI universal, otros instrumentos utilizan como puerto de recepción de señales MIDI los puertos USB, mini USB, USB tipo B, entre otros.

El uso de este protocolo MIDI facilita la comunicación entre instrumentos ya que al ser universal desde su creación, lo que se ha modificado es el tipo de señales que son enviadas sin necesidad de tener un previo enlace o enlace específico, para comunicar pianos entre pianos, guitarras con pianos, o cualquier otra configuración de instrumentos posibles, ya que la guitarra es un instrumento que no posee electrónica activa, para poder comunicar una guitarra con un teclado o cualquier otro instrumento que utilice señales en protocolo MIDI es necesario el procesamiento de su señal.

En nuestro caso para realizar el procesamiento de señal utilizamos primero un circuito de acondicionamiento de datos que permita ingresar la señal de guitarra a la tarjeta de programación Arduino ya con la señal eléctrica en nuestro Arduino, procedemos a realizar la respectiva filtración y procesamiento a señal discreta.

Al realizar el cambio a señal discreta, utilizando bucles de programación para filtración de datos se utilizan los convenios de envío de señal en protocolo MIDI para Arduino.

Para enviar desde el Arduino hacia cualquier instrumento se utiliza el conector universal MIDI con 5 pines, ya si se desea adaptar hacia otro conector, dependiendo del instrumento al que se vaya a conectar, se pueden realizar los respectivos cambios físicos a través de conectores que sirvan como adaptadores del conector de 5 pines hacia el instrumento al cual se desea conectar el sistema electrónico que asocie notas MIDI para la pulsación de una cuerda en una guitarra eléctrica.

### **3. PROPUESTA DE DISEÑO DE SISTEMA ELECTRÓNICO QUE ASOCIE NOTAS MIDI A LA EJECUCIÓN DE UNA NOTA EN UNA CUERDA DE GUITARRA ELÉCTRICA**

#### **3.1. Descripción del diseño**

El diseño del sistema electrónico que asocie notas MIDI a la pulsación de una nota en una cuerda de una guitarra eléctrica se abordará de la siguiente manera.

Tomando en cuenta el análisis de las pastillas de fabricación americana que se encargaran de realizar la transducción de los campos magnéticos a señal eléctrica, así como la calidad del conector de salida de señal eléctrica, se plantea una relación de señal a ruido que no perturbe la captación de señal así como su procesamiento y acondicionamiento a señal discreta.

Una vez la señal de la guitarra es captada, se procesa analógicamente con un chip electrónico, cuya función es la de amplificar la señal eléctrica captada en la entrada de señal del Circuito integrado LM386 cuya entrada utiliza un filtro pasa bajas que se encarga de discriminar las señales de la red eléctrica que alimenta al circuito, esta señal filtrada llega a la patilla 3 del LM386 esta señal se amplifica con ganancia máxima de 10 decibeles, la misma es filtrada a su salida para evitar los ruidos internos a altas frecuencias del LM386, luego pasa por un capacitor de acople de señal y se envía directamente al puerto de señales analógicas A0 del microcontrolador Arduino Mega.

La señal que entra al puerto A0 del microcontrolador es analógica, por lo cual la electrónica interna del Arduino se encarga de realizar la conversión de señal analógica a señal discreta.

La señal discreta se utiliza la transformada rápida de Fourier con el objetivo de procesar la señal en el dominio del tiempo y analizarla en el dominio de la frecuencia, utilizando ciclos de programación embebida, se realiza comparación de la frecuencia que se está detectando al ejecutar la nota en la guitarra electroacústica con el valor de referencia que tiene el ciclo de programación para cada detección de nota.

La nota detectada se convierte en función programable que asociará notas MIDI para cada pulsación de nota individual en las cuerdas de la guitarra.

### **3.2. Análisis del circuito de captación y acondicionamiento de señal eléctrica**

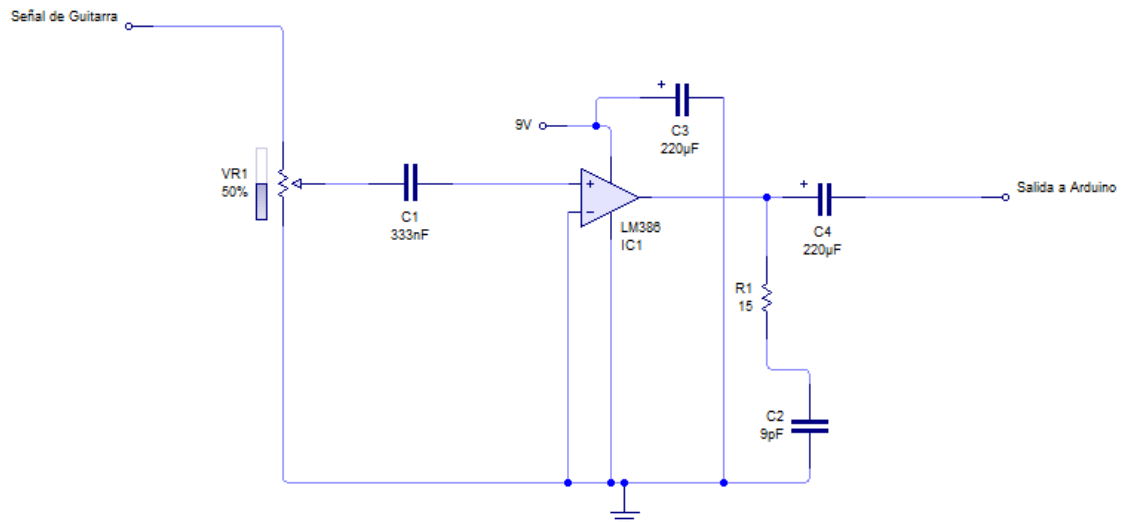
Para Valencia y Londoño (2022) “Un Sistema de Adquisición de Datos es un conjunto de elementos que permiten la medición de variables de un proceso físico, y el muestreo y digitalización de esta información”. (p.13)

Se ha mencionado en los capítulos anteriores que el circuito encargado de la adquisición de datos, así como su acondicionamiento de señal está basado en el IC LM386 que se encarga de amplificar pequeñas señales tales como las provenientes de la guitarra eléctrica.

El circuito que se muestra a continuación es el encargado de acondicionar la señal de guitarra eléctrica para su posterior procesamiento de señal eléctrica a señal discreta.



Figura 5. **Diagrama del circuito que acondiciona la señal de guitarra eléctrica**

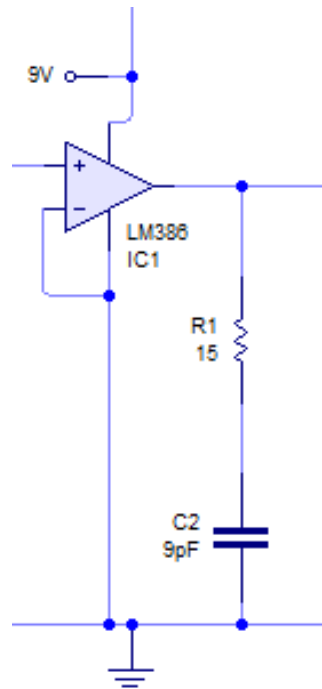


Fuente: elaboración propia, realizado con Circuit Wizard.

La etapa encargada de realizar la primera filtración de señal es la que recibe directamente la señal de guitarra y compuesta por VR1 que representa el potenciómetro de 15000 ohmios y por el capacitor C1 de 333 nanofaradios, encargados de regular la intensidad de señal que llegará a la patilla no inversora del amplificador operacional IC1 siendo el mismo el IC LM386.

La segunda filtración de señal ocurre a la salida del amplificador de bajo ruido LM386 y es llevada a cabo por el filtro paso bajo formado por el capacitor C2 de 47 nanofaradios en serie con la resistencia R1 de 10 ohmios, este es el encargado de filtrar a la salida de la amplificación de la señal de guitarra las frecuencias que no se encuentren en el rango de las que proporciona la guitarra eléctrica.

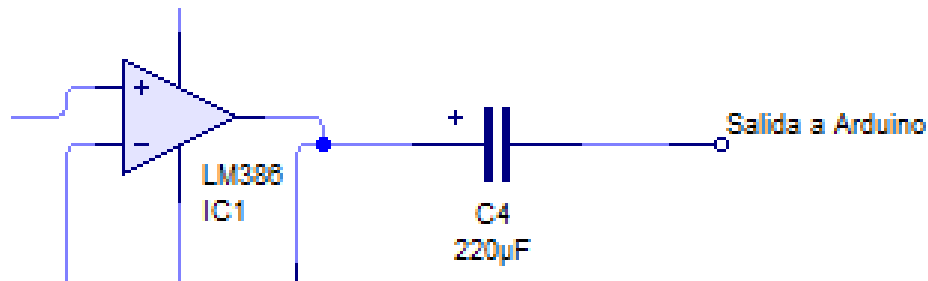
Figura 6. Diagrama de la salida de señal del IC LM 386 amplificada



Fuente: elaboración propia, realizado con Circuit Wizard.

A la salida de nuestro amplificador LM386 se utiliza un capacitor de 220 microfaradios como acoplador de señal ya que se conecta directamente al puerto A0 de señales analógicas del Arduino Mega.

Figura 7. **Diagrama del acople de la salida de la señal amplificada hacia la tarjeta electrónica programable**



Fuente: elaboración propia, realizado con Circuit Wizard.

### 3.3. **Análisis de la tarjeta de procesamiento de señales con mejores prestaciones calidad – precio**

Para Pérez vega et. al. (2007) “El proceso de muestreo es común a todos los sistemas de modulación de pulsos y por lo general, su descripción se hace en el dominio del tiempo, se convierte en una secuencia de frecuencias discretas de la señal, a intervalos regulares”. (p.237)

Para escoger una tarjeta que nos apoye con el procesamiento de señales, se analiza la capacidad de realizar muestreo de señal, este análisis depende en gran medida de los bits que dediquemos para el muestreo de la señal.

Seguidamente se analiza la capacidad de realizar conversiones de señales analógicas a señales digitales, esto debido a la naturaleza de la señal eléctrica proporcionada por el circuito que se encarga de acondicionar la señal de guitarra, así como su respectiva conversión a señal discreta para poder detectar la frecuencia con mayor presencia en la nota ejecutada al pulsar la cuerda de la guitarra eléctrica.

Las características anteriores son las más importantes debido a que como mínimo, la tarjeta de programación debe de procesar una frecuencia de muestreo que no afecte la integridad de la señal. Sumado a lo anterior, el precio de la tarjeta de programación que nos apoya en el sistema de asociación de notas MIDI para la pulsación de una cuerda en la guitarra eléctrica debe ser de bajo coste, debido a los objetivos de este trabajo.

En aras de ser congruente con este estudio, se utiliza para el diseño del sistema electrónico que asocie notas MIDI la tarjeta de programación Arduino Mega con su frecuencia de reloj a 16 MHz es decir puede ejecutar instrucciones en aproximadamente 3 nanosegundos por instrucción, esto nos permite configurar una frecuencia de muestreo que se mantenga estable en el rango de las frecuencias que alcanzan las notas ejecutadas en la guitarra eléctrica.

#### **3.4. Evaluación del diseño del sistema electrónico que asocie notas MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica**

Para la adquisición de datos de la señal de guitarra se utiliza el circuito electrónico de detección basado en el amplificador de bajo ruido LM386 anteriormente explicado.

En conjunto con la tarjeta de programación podemos obtener la señal en el dominio de la frecuencia con el fin de manipular con lógica de programación nuestra señal y asociarle notas MIDI que serán las encargadas de comunicarse con otros instrumentos.

Para que una nota MIDI sea asociada, lo único que se necesita es detectar que frecuencia es la que tiene mayor presencia al momento de ejecutar la

pulsación en la guitarra eléctrica. Para este fin se utiliza el acondicionamiento de señal analógica a señal discreta, este proceso se realiza internamente en los circuitos DAC que tiene la tarjeta Arduino Mega. Según Martin et. al. (2007) los convertidores tienen como misión la obtención de una representación digital de la magnitud analógica que se presenta a su entrada. Los procesos que deben llevar a cabo son los de la cuantificación, por el que la señal analógica continua de entrada se transforma en un conjunto discreto de estados de salida. (p.315).

Media vez la señal es convertida a señal discreta, podemos aplicar la transformada rápida de Fourier, es un método que permite la detección de la frecuencia que mayor presencia tenga la pulsación de la nota en la guitarra al ser ejecutada.

Se muestra el código de programación en Arduino Mega utilizado el análisis de la detección de frecuencia que mayor presencia tenga en la señal al pulsar una cuerda de guitarra eléctrica.

Figura 8. **Conversión de señal en el dominio de tiempo a señal en dominio de frecuencia**

```
include "arduinoFFT.h" // incluye la librería que apoyará con la
transformada rápida de Fourier

#define muestras 256 // Debe de ser potencia de 2 para su
implementación

#define frecuencia_muestreo 9000 // debe de ser menor a 10000 Hz
y no superar la capacidad del ADC del puerto A0 del Arduino

arduinoFFT FFT = arduinoFFT(); // inicialización de la función de
transformada rápida

unsigned int periodo_muestreo; // se designa como integer el periodo
de muestreo en microsegundos

unsigned long microsegundos; // se designa como long a
microsegundos double vReal[muestras]; // se define el arreglo vReal que
procesara la parte real de la señal de la guitarra que es acondicionada en el
puerto A0

double vImag[muestras]; // defina el arreglo VImag para la parte
imaginaria

void setup() {

Serial.begin(9600); // se inicializa comunicación con el puerto serial a
9600 baudios
```

Continuación de figura 8.

```
    periodo_muestreo = round(1000000*(1.0/frecuencia_muestreo)); // se
define el periodo al que se realizará el muestreo de señal

}

void loop() {

  /*Muestreo*/

  for(int i=0; i<muestras; i++) // ciclo for que realiza el muestreo de señal
leyendo la señal que entra en el puerto A0 del Arduino

  {

    microsegundos = micros(); // indicamos que la función microsegundos
usara la funcion micros(); interna de la logica Arduino

    vReal[i] = analogRead(0); // se declara todo lo que leamos en el puerto A0
como parte real de la función de la transformada rápida de Fourier

    vImag[i] = 0;

    while(micros() < (microsegundos + periodo_muestreo)){

    }

  }

  /*transformada Rápida de fourier*/

  FFT.Windowing(vReal,      muestras,      FFT_WIN_TYP_HAMMING,
FFT_FORWARD); // procesamiento de las lecturas en el arreglo que almacena los
valores del puerto A0 en vReal

  FFT.Compute(vReal, vImag, muestras, FFT_FORWARD);

  FFT.ComplexToMagnitude(vReal, vImag, muestras);
```

Continuación de figura 8.

```
double pico_frecuencia = FFT.MajorPeak(vReal, muestras,
frecuencia_muestreo); // Detección de la frecuencia que mayor presencia
tenga a la hora de pulsar la cuerda en la guitarra

/*Imprimir resultados en consola*/

Serial.println(pico_frecuencia); //Imprime en consola el valor de
frecuencia detectado

delay(100); //repetir el proceso de detección cada cien milisegundos
milisegundos

}
```

Fuente: elaboración propia, realizado con Arcuino IDE.

### 3.5. Análisis de la asociación de notas MIDI a la pulsación de la cuerda de una guitarra eléctrica

Para poder asociar notas MIDI a la programación de Arduino es de vital importancia conocer la lógica de comparación y asociación.

Para la comparación se plantea almacenar la nota que se está ejecutando en tiempo real y procesarla en una variable en la cual utilizando rangos de valores que se encuentren en el rango de detección para cada frecuencia asociada al traste que en el que se ejecuta la pulsación de la cuerda.



De lo anterior, lo que se plantea es la utilización de la variable que detecta en todo momento las lecturas del puerto analógico A0 y para cada rango determinado de frecuencias comprendidas en la ejecución de la nota, asignarle valores en protocolo MIDI, en este caso queda a discreción del diseñador si se pueden asociar 2 o más notas MIDI para cada pulsación de guitarra.

En lo que concierne a este trabajo se utilizarán las notas que se ejecuten en la sexta cuerda como notas principales en nuestro sistema, y para cada nota que se detecte entre este rango de frecuencias, se le asociara un acorde con 3 o 4 notas MIDI, es decir, si se detecta una nota ejecutada en un traste de la guitarra en la sexta cuerda, se asociará un acorde compuesto por 3 o 4 notas en protocolo MIDI y se ejecutará en el instrumento final, en este caso un teclado que utilice el protocolo MIDI. Dependiendo si el acorde a ejecutar en el instrumento final es mayor, se utilizarán 4 notas, y si el acorde es menor, se utilizarán 3 acordes.

Se muestra a continuación la lógica de programación que permite el uso del protocolo MIDI en la tarjeta de Programación Arduino MEGA 2560.

En el lenguaje de programación que utiliza la tarjeta Arduino Mega, se pueden importar librerías que nos permiten el uso del protocolo de comunicación MIDI así como las instancias que nos permiten inicializar la comunicación de la tarjeta Arduino con el instrumento final.

Se utilizan convenios estandarizados para su respectivo uso en la comunicación MIDI para la asignación de números decimales que representan la nota en su respectiva octava, la intensidad o volumen a la que la nota se ejecutará y el canal que se utiliza para el envío de información sin que exista redundancia.

De esta forma el instrumento final con el que se quiere comunicar la tarjeta también pueda entender las instrucciones, por lo tanto, ejecute sin problemas las notas asociadas a la pulsación de la cuerda de la guitarra eléctrica.

Figura 9. **Asociación de notas en dominio de la frecuencia a notas MIDI**

```
#include <MIDI.h> // agrega al programa la librería
MIDIMIDI_CREATE_DEFAULT_INSTANCE(); // Crea la instancia para manipular objetos
MIDI

void setup() {
  pinMode(A1, OUTPUT); // se declara el Pin A1 como salida para las señales MIDI
  MIDI.begin(MIDI_CHANNEL_OFF); // se inicializa el protocolo MIDI en el arduino
  indicando que solo será salida
}

void loop() {
  for(int i=0; i<muestras; i++) // ciclo for que realiza el muestreo de señal leyendo
  la señal que entra en el puerto A0 del Arduino
  {
    microsegundos = micros(); // indicamos que la función microsegundos usara la
    funcion micros(); interna de la logica Arduino

    vReal[i] = analogRead(0); // se declara todo lo que leamos en el puerto A0 como
    parte real de la función de la transformada rápida de Fourier

    vImag[i] = 0;
    while(micros() < (microsegundos + periodo_muestreo)){
    }
  }
}
```

Continuación de figura 9.

```
    /*transformada Rápida de fourier*/
    FFT.Windowing(vReal, muestras, FFT_WIN_TYP_HAMMING, FFT_FORWARD);
// procesamiento de las lecturas en el arreglo que almacena los valores del puerto A0 en
vReal
    FFT.Compute(vReal, vImag, muestras, FFT_FORWARD);
    FFT.ComplexToMagnitude(vReal, vImag, muestras);
    double pico_frecuencia = FFT.MajorPeak(vReal, muestras, frecuencia_muestreo);
// Detección de la frecuencia que mayor presencia tenga a la hora de pulsar la cuerda en la
guitarra
    /*Imprimir resultados en consola*/
    Serial.println(pico_frecuencia); //Imprime en consola el valor de frecuencia
detectado
    delay(100); //repetir el proceso de detección cada cien milisegundos milisegundos
}
    if (pico_frecuencia >= 232hz && pico_frecuencia <=235hz ) // Rango de frecuencia
para la detección de la nota
        MIDI.sendNoteOn(46,120,1); // se envia en protocolo MIDI la nota que se
ejecutaran el en instrumento final
        MIDI.sendNoteOn(38,120,1); // se asocian las notas que forman el acorde La# en
el instrumento final
        MIDI.sendNoteOn(41,120,1);
        delay(100);
        MIDI.sendNoteOff(46,0,1);
        MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente para que
no se ejecute perpetuamente
```

Continuación figura 9.

```
MIDI.sendNoteOff(41,0,1);  
    delay(100);  
    else if (pico_frecuencia >= 382hz && pico_frecuencia <=385hz ) // Rango de  
        frecuencia para la detección de la nota  
        MIDI.sendNoteOn(46,120,1); // se envia en protocolo MIDI la nota que se  
        ejecutaran el en instrumento final  
MIDI.sendNoteOn(38,120,1); // se asocian las notas que forman el acorde La# en  
        el instrumento final  
        MIDI.sendNoteOn(41,120,1);  
        delay(100);  
        MIDI.sendNoteOff(46,0,1);  
MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente para  
        que no se ejecute perpetuamente  
        MIDI.sendNoteOff(41,0,1);  
        delay(100);  
        else{  
        MIDI.sendNoteOff(46,0,1);  
MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente para  
        que no se ejecute perpetuamente  
        MIDI.sendNoteOff(41,0,1);  
        delay(100);  
        }  
    delay(100); //repetir el proceso de detección cada cien milisegundos  
        milisegundos
```

Fuente: elaboración propia, realizado con Arcuino IDE.

## **4. ANÁLISIS TECNICO-ECONOMICO DEL SISTEMA**

### **4.1. Costos generales de la implementación técnica del sistema**

Para estimar la viabilidad técnica - económica de la fabricación del sistema en función de la fiabilidad, respuesta en frecuencia, correcto funcionamiento y el bajo costo que representa el hardware y software a utilizar, se realizan análisis de costo de materiales, implementación de conocimientos, horas de trabajo y parámetros que influyan en el costo final del diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica.

Los costos conforme el tiempo pueden variar, debido a las alzas en los precios, por lo que este trabajo se enfocará en los materiales y costos actuales de los componentes y servicios requeridos para el diseño del sistema electrónico.

### **4.2. Costo de materiales del procesamiento del procesamiento de señal**

Se analizan los costos de los materiales para el diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica.

#### **4.2.1. Circuito de adquisición de datos**

Se presentan los costos de los componentes a utilizar en el diseño para el circuito de adquisición de datos desde la recepción de la señal de guitarra,

filtración, amplificación y respectivo acople y adecuación para la salida de señal de nuestro circuito.

A continuación, se muestran en la siguiente tabla los componentes utilizados en el circuito de adquisición de datos, su función en el circuito y su costo en quetzales.

Tabla I. **Costos del circuito de adquisición de datos**

<b>Componente</b>	<b>Función en el diseño del circuito</b>	<b>Precio</b>
Potenciómetro logarítmico de 10 kilohmios	Filtración de entrada de señal	Q.40.00
Capacitor de 10 nanofaradios	Filtración de entrada de señal	Q.1.50
Capacitor de 100 microfaradios	Filtración de ruido de la fuente de	Q.1.50

	alimentación	
Circuito Integrado LM 386	Amplificación de la señal de la guitarra filtrada	Q.9.00

Continuación tabla I.

<b>Componente</b>	<b>Función en el diseño del circuito</b>	<b>Precio</b>
Capacitor de 47 nanofaradios	Filtración de ruido debidocomponentes internos del CI LM386	Q.1.50
Resistencia de 10 ohmios	Filtración de ruido debido a componentes	Q.1.00

	internos del CI LM386	
Capacitor de 220 microfaradios	Acople de señal para conectarse al Arduino Mega	Q.1 .50
Conector Jack de ¼ de pulgada	Entrada de señal del circuito	Q.8 .00
Terminal Block	Salida de señal del circuito	Q.2 .00
Placa de fibra de vidrio con pistas de cobre	Ensamblar el circuito de adquisición de datos	Q. 15. 00
Estaño	Soldar componentes	Q.5 .00
Cable	Conexiones de conectores de entrada y salida	Q.1 0.0 0
	Costo total	Q.9 6.0 0

Fuente: elaboración propia realizado con Microsoft Word.



#### **4.2.2. Costos de tarjeta electrónica y componentes auxiliares para el procesamiento de la señal eléctrica**

Se analizan los costos de la tarjeta Arduino Mega 2560, Así como los componentes auxiliares necesarios en el procesamiento de la señal discreta hasta su respectiva salida a señal MIDI.

En la actualidad se utilizan conectores USB para conectar los dispositivos MIDI entre sí, sin embargo, no se puede generalizar esta conexión para todos los dispositivos, por lo tanto en el diseño del circuito que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica, para la salida de la señal que se comunica con los dispositivos de ejecución final se determina el uso del conector universal de 5 pines del protocolo MIDI y salida con conector de USB tipo B.

Tabla II. **Costos de tarjeta electrónica y componentes auxiliares para el procesamiento de la señal eléctrica**

<b>Componete</b>	<b>Función en el diseño del circuito</b>	<b>Precio</b>
Tarjeta de programación Arduino Mega 2560	Dispositivo encargado del procesamiento discreto de señal así como la programación de la asociación de sonidos MIDI	Q.430.00
Conector MIDI 5 pines	Conector de salida	Q.15.00
Conector USB tipo B	Conector de salida	Q.20.00
Cable	Conexiones	Q.20.00

	Total	Q.485. 00
--	-------	--------------

Fuente: elaboración propia, realizado con Microsoft Word.

#### **4.3. Costos de horas de trabajo**

La hora de trabajo de personal capacitado para el diseño del sistema electrónico se valora en 50 quetzales. Para la elaboración del diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica, se dividirán los costos en módulos, con el objetivo de simplificar la presentación de los costos de horas de trabajo.

Tabla III. **Costos de horas de trabajo**

<b>Modulo</b>	<b>Horas de trabajo</b>	<b>Precio</b>
Adquisición de datos	1 5 horas	Q.75 0.00
Procesamiento de señal analógica a señal discreta	1 7 horas	Q.85 0.00
Lógica del Programa de diseño del sistema electrónico	2 4 horas	Q120 0.00
Lógica de programación del	1 7 horas	Q.85 0.00

protocolo MIDI		
	T	Q.36
	total	50

Fuente: elaboración propia, realizado con Microsoft Word.

**4.4. Costo final del diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar una cuerda en una guitarra eléctrica**

Se realiza la sumatoria de costos del diseño de sistema electrónico, en este caso se toman en cuenta las variaciones en los precios al año 2022 en el consumo de materiales electrónicos y eléctricos.

Tabla IV. **Costo final del diseño del sistema electrónico que asocie sonidos MIDI a la ejecución de una nota al pulsar cuerda en una guitarra eléctrica**

<b>Etapa del diseño del sistema electrónico</b>	<b>Función en el diseño del sistema electrónico</b>	<b>Costo</b>
Circuito de Adquisición	Acondicionar la señal	Q.96 .00

ón de Datos	para la tarjeta de procesa miento.	
Tarjeta electrónica de procesa miento de señal y conector es de salida hacia el instrume nto de ejecución final	Procesa miento de la señal a dominio de la frecuenci a, programa ción lógica de la asociació n de sonidos MIDI a la ejecución de una nota en una guitarra eléctrica	Q.48  5.00
Etapas del diseño del	Función en el diseño del sistema	Cost  o

sistema electrónico	electrónico.	
Costo de horas de trabajo	Tiempo de trabajo utilizado para el diseño del sistema electrónico.	Q. 3650
	Costo Total	Q.4, 231.00

Fuente: elaboración propia realizado con Microsoft Word.





## CONCLUSIONES

1. Analizar los sistemas de determinación existentes para la ejecución de notas en una guitarra se determinó que el protocolo de comunicación entre instrumentos MIDI es el que mejores prestaciones puede brindar para el desarrollo del sistema electrónico que detecte la nota de una cuerda ejecutada en una guitarra eléctrica y la asocie a demás notas en un instrumento de ejecución final.
2. Al analizar los procesos electrónicos para asociar señales midi a la pulsación de una cuerda en una guitarra eléctrica se concluye que el acondicionamiento de la señal y su respectiva filtración permite una mejor manipulación de señal cuando es ingresada a la tarjeta de programación Arduino Mega, así mismo, la transformada rápida de Fourier permite el aprovechamiento de los recursos de la tarjeta evitando sobre procesamiento y retraso en la respuesta a la ejecución de la nota en la guitarra y asociación a notas MIDI en el instrumento final.
3. Luego de los análisis respectivos, se determinó que para el diseño final del sistema electrónico se utilicen componentes de bajo coste pero con la mejor calidad posible, es decir utilizar componentes pasivos de alta calidad, así mismo, la tarjeta Arduino Mega original para evitar fallas.
4. En el análisis técnico-económico del sistema se determinó que los gastos de los materiales electrónicos, tanto como los demás, son de gran importancia ya que el diseño del sistema electrónico que asocie notas

MIDI a la pulsación de una cuerda en una guitarra eléctrica para que sea agradable al consumidor, debe tener una muy buena presentación.

## RECOMENDACIONES

1. Implementar el diseño del circuito electrónico que asocie notas MIDI a la pulsación de una cuerda en una guitarra eléctrica se deben de tener conocimientos musicales, electrónicos, eléctricos y de programación, la integración de estos conocimientos, permiten el ahorro de recursos físicos, dinero y tiempo.
2. Prolongar la vida del sistema electrónico se recomienda el mantenimiento respectivo a sus placas de adquisición de datos, conectores de señal, conectores de alimentación, cableado y limpieza de polvo en caso se llegue acumular en los componentes.
3. Entender los conceptos del funcionamiento del sistema electrónico, así mismo se pruebe con diferentes instrumentos que utilicen el protocolo MIDI, para la experimentación musical con el dispositivo.



## REFERENCIAS

1. Blanco, X. (2007). El Protocolo MIDI. *Hispanic*. Recuperado de <https://analfatecnicos.net/pregunta.php?id=56>.
2. Boylestad, R. L. (2004). *Introducción al análisis de circuitos*. México: Pearson.
3. Burgos, J. S. (2016). Análisis de sistemas procesadores de Guitarra. *Ingenium* Vol.17, 76-89. Recuperado de: <https://revistas.usb.edu.co/index.php/Ingenium/citationstylelanguage/get/chicago-author-date?submissionId=2740&publicationId=2154>.
4. Cl, A. (2022). *Arduino Cl*. Recuperado de: <https://arduino.cl/arduino-mega-2560/>.
5. Constantino Perez Vega, J. M. (2007). *Sistemas de Telecomunicaciones*. Santander: Universidad de Cantabria .
6. Donate, A. H. (2012). *Electrónica Aplicada*. Barcelona: Marcombo.
7. Holt, C. A. (1989). *Circuitos Electronicos: digitales y analogicos*. New York: Reverte.
8. ISO. (1975). *iso.org*. Obtenido de [iso.org: https://www.iso.org/standard/3601.html](https://www.iso.org/standard/3601.html).

9. Jesus Bobadilla, P. G. (1999). *LA TRANSFORMADA DE FOURIER. UNA*. Madrid: Departamento de Informatica Aplicada.
10. Jose L. Martin, J. A. (2007). *Problemas Resueltos de la electrónica*. Madrid: Delta.
11. Marcela V. Valencia, A. A. (2022). *Introducción a la adquisición y acondicionamiento de señales*. Medellin: Instituto Tecnológico Metropolitano.
12. Ortega, M. (2014). *MATLAB Aplicado a telecomunicaciones*. Ciudad de Mexico: Alfaomega.
13. Robredo, G. R. (2001). *Electrónica Básica para Ingenieros* . Santander: Universidad de Cantabria.
14. Semansky, S. y. (2009). *Física Universitaria Volumen 1*. Mexico: Pearson.
15. Yadav, A. (2008). *Digital Communication*. New Delhi: University Science Press.

# APÉNDICES

## Apéndice 1. **Manual de Implementación**

Diseño del circuito de adquisición de datos

- Realizar en un software de apoyo (Proteus, Simulink, entre otros.) el diagrama del circuito.
- Tener a la mano Cautin de baja potencia, estaño, placa de fibra de vidrio, acido para quemar el cobre de placas, alcohol, recipiente, papel, lija, plancha y componentes electrónicos para la realización del circuito.
- Imprimir el diseño del circuito electrónico en tinta LASER para mejor transferencia.
- Lijar la parte de cobre de la placa de fibra de vidrio.
- Ajustar la impresión del circuito PCB a la parte de cobre placa de fibra de vidrio.
- Planchar durante 7 minutos sobre el diseño en la placa de fibra de vidrio para tener una excelente transferencia de calor.
- Asegurarse de que el diseño del circuito esté bien pegado a la parte de cobre de la placa.
- Vertir un poco de ácido en un recipiente y colocar la placa de fibra de vidrio para remover el cobre que no utilizaremos en el circuito.
- Limpiar la placa con acetona o algún solvente para que remueva los restos de tinta y acido de la placa.
- Barrenar los agujeros en donde irán los componentes.

### Continuación de apéndice 1.

- Como consejo se pueden colocar los componentes en sus respectivos agujeros con el objetivo de ordenar y dejar lo mejor posible el circuito.
- Una buena práctica de soldadura en chips de circuitos integrados tales como el LM386 suele ser utilizando socket de 8 pines, de esta forma el calor de soldadura lo recibe el socket y no el IC, luego de soldar el socket se procede a la colocación del IC LM386.
- Repasar el diseño del circuito y asegurarse que cada componente se encuentre en el orden indicado para el correcto funcionamiento del circuito.
- Soldar componente por componente.
- Luego de soldar todos los componentes verificar que exista continuidad en todas las pistas de cobre.
- Verificar el funcionamiento del circuito de adquisición de datos.

### Conexiones entre el circuito de adquisición de datos y el Arduino Mega.

- Realizar los respectivos cableados entre circuitos y sus respectivas alimentaciones de voltaje.
- Alimentar circuito de adquisición de datos y Arduino con 5 voltios en corriente directa.
- Conectar los comunes GND de ambos circuitos.
- De la salida de señal del circuito de adquisición de datos, conectar a la tarjeta de programación la señal utilizando cable calibre 20 awg hacia el puerto A0 del Arduino.



Continuación apéndice 1.

### Programación lógica en la tarjeta de programación Arduino MEGA

- Conseguir el entorno de desarrollo Arduino IDE.
- Descargar librerías MIDI.h y Arduinofft.h e instalarlas en el Arduino IDE.
- Realizar la definición de variables para el procesamiento de señal análoga a señal discreta y señales MIDI.

```
- #include "arduinoFFT.h" // incluye la librería que apoyará con la
transformada rápida de fourier

#include <MIDI.h> // agrega al programa la libreria MIDI

#define SAMPLES 256 // Debe de ser potencia de 2 para su
implementación en el uso de Bits

#define SAMPLING_FREQUENCY 9000 // debe de ser menor a 10000 Hz
en orden de no producir aliasing y no superar la capacidad delmADC del puerto
de analogicos
```

Continuación apéndice 1.

```
arduinoFFT FFT = arduinoFFT(); // inicialización de la función de transformada rápida
unsigned int sampling_period_us;

unsigned long microseconds; // se asignan parámetros sin signo, para que la función
de transformada rápida los pueda procesar sin problemas

double vReal[SAMPLES]; // se asigna el vector real para el muestreo de la señal
double vImag[SAMPLES]; // se asigna el vector imaginario para que la función fft realice su
secuencia, ya que estos datos no serán de utilidad para el proyecto

MIDI_CREATE_DEFAULT_INSTANCE(); // Crea la instancia para manipular objetos
MIDI
```

Desarrollo la clase que nos permitirá configurar las condiciones iniciales de nuestro proyecto

```
void setup() {

pinMode(A1, OUTPUT); // se declara el Pin A1 como salida para las señales MIDI

MIDI.begin(MIDI_CHANNEL_OFF); // se inicializa el protocolo MIDI en el arduino
indicando que solo será salida

Serial.begin(9600);

sampling_period_us = round(1000000*(1.0/SAMPLING_FREQUENCY)); // periodo al
que se realizará el muestreo de señal

}
```

Continuación apéndice 1.

```
        FFT.Windowing(vReal,  muestras,  FFT_WIN_TYP_HAMMING,
FFT_FORWARD); // procesamiento de las lecturas en el arreglo que almacena
los valores del puerto A0 en vReal

        FFT.Compute(vReal, vImag, muestras, FFT_FORWARD);

        FFT.ComplexToMagnitude(vReal, vImag, muestras);

        double  pico_frecuencia  =  FFT.MajorPeak(vReal,  muestras,
frecuencia_muestreo); // Detección de la frecuencia que mayor presencia tenga
a la hora de pulsar la cuerda en la guitarra

        /*Imprimir resultados en consola*/

        Serial.println(pico_frecuencia); //Imprime en consola el valor de
frecuencia detectado

        delay(100); //repetir el proceso de detección cada cien milisegundos
milisegundos
    }

    if (pico_frecuencia >= 232hz && pico_frecuencia <=235hz ) // Rango de
frecuencia para la detección de la nota

        MIDI.sendNoteOn(46,120,1); // se envia en protocolo MIDI la nota que se
ejecutaran el en instrumento final
```

Continuación apéndice 1.

```
MIDI.sendNoteOn(38,120,1); // se asocian las notas que forman el acorde
La# en el instrumento final

MIDI.sendNoteOn(41,120,1);

delay(100);

MIDI.sendNoteOff(46,0,1);

MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente
para que no se ejecute perpetuamente

MIDI.sendNoteOff(41,0,1);

delay(100);

else if (pico_frecuencia >= 382hz && pico_frecuencia <=385hz ) // Rango
de frecuencia para la detección de la nota

MIDI.sendNoteOn(46,120,1); // se envia en protocolo MIDI la nota que se
ejecutaran el en instrumento final

MIDI.sendNoteOn(38,120,1); // se asocian las notas que forman el acorde
La# en el instrumento final

MIDI.sendNoteOn(41,120,1);

delay(100);

MIDI.sendNoteOff(46,0,1);
```

Continuación apéndice 1.

```
MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente
para que no se ejecute perpetuamente

MIDI.sendNoteOff(41,0,1);

delay(100);

else if (pico_frecuencia >= 523hz && pico_frecuencia <=527hz ) // Rango
de frecuencia para la detección de la nota

MIDI.sendNoteOn(46,120,1); // se envia en protocolo MIDI la nota que se
ejecutaran el en instrumento final

MIDI.sendNoteOn(38,120,1); // se asocian las notas que forman el acorde
La# en el instrumento final

MIDI.sendNoteOn(41,120,1);

delay(100);

MIDI.sendNoteOff(46,0,1);

MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente
para que no se ejecute perpetuamente

MIDI.sendNoteOff(41,0,1);

delay(100);

else if (pico_frecuencia >= 438hz && pico_frecuencia <=443hz ) // Rango
de frecuencia para la detección de la nota
```

## Continuación apéndice 1.

```
        MIDI.sendNoteOn(46,120,1); // se envia en protocolo MIDI la nota que se ejecutaran el
en instrumento final

        MIDI.sendNoteOn(38,120,1); // se asocian las notas que forman el acorde La# en el
instrumento final

        MIDI.sendNoteOn(41,120,1);

        delay(100);

        MIDI.sendNoteOff(46,0,1);

        MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente para que no
se ejecute perpetuamente

        MIDI.sendNoteOff(41,0,1);

        delay(100);

        else{

        MIDI.sendNoteOff(46,0,1);

        MIDI.sendNoteOff(38,0,1); // se apaga la nota que se envió anteriormente para que no
se ejecute perpetuamente

        MIDI.sendNoteOff(41,0,1);

        delay(100);

        }

        delay(100); //repetir el proceso de detección cada cien milisegundos milisegundos

    }
```

## Continuación apéndice 1.

Desarrollo de la clase que se estará ejecutando mientras nuestro sistema electrónico se encuentre encendido, se encarga de realizar el procesamiento de la señal en el dominio del tiempo a señal discreta en el dominio de la frecuencia y a su vez asociar notas MIDI para cada pulsación de nota ejecutada en una cuerda de guitarra eléctrica.

```
void loop() {  
    for(int i=0; i<muestras; i++) // ciclo for que realiza el muestreo de señal leyendo la  
    señal que entra en el puerto A0 del arduino  
    {  
        microsegundos = micros(); // indicamos que la función microsegundos usara la  
        funcion micros(); interna de la logica Arduino  
  
        vReal[i] = analogRead(0); // se declara todo lo que leamos en el puerto A0 como  
        parte real de la función de la transformada rápida de fourier  
  
        vImag[i] = 0;  
  
        while(micros() < (microsegundos + periodo_muestreo)){  
        }  
    }  
  
    /*transformada Rápida de fourier*/  
}
```

Continuación apéndice 1.

- Después de la implementación de la programación se procede a enviar la señal de salida de nuestro sistema a través de 2 salidas, la MIDI USB tipo B y la MIDI estándar de 5 pines
- Se realiza la conexión entre el sistema electrónico y el instrumento de ejecución final que utilice el protocolo MIDI.

Fuente: elaboración propia, realizado con Microsoft Word y Arduino IDE.



## Apéndice 2. **Manual de usuario**

Encender el sistema electrónico que asocia sonidos MIDI a la ejecución de una nota en una cuerda de guitarra eléctrica.

- Asegurarse que la guitarra se encuentre bien conectada a nuestro sistema electrónico.
- Asegurarse que el sistema electrónico se encuentre correctamente conectado al instrumento de ejecución final.
- Encender el instrumento de ejecución final que utiliza el protocolo MIDI.
- Dependiendo de la nota ejecutada en la cuerda de guitarra, el sistema electrónico asociará para cada nota 1 acorde en el instrumento final.
- El sistema electrónico está diseñado para que se ejecute en el instrumento final que utiliza protocolo MIDI 1 acorde formado por 4 notas si es acorde mayor o 3 notas en el caso que sea acorde menor.
- Ejecutar una nota en una cuerda de guitarra eléctrica.

Fuente: elaboración propia, realizado con Microsoft Word y Arduino IDE.

### Apéndice 3. **Manual para resolver fallas**

- Si no enciende el sistema, asegurarse que la alimentación de corriente directa que alimenta los circuitos este llegando a todas las etapas del sistema.
- Si no se detecta la pulsación de la cuerda en la guitarra eléctrica, asegurarse que la pastilla funcione, que el switch de cambio de bobinas funcione y que el conector de salida de señal se encuentre en buen estado.
- Si no detecta señal la tarjeta Arduino, asegurarse que nuestro circuito de adquisición de datos se encuentre en buen funcionamiento y tengamos voltaje a la salida de la señal.
- Si no asocia la pulsación de la cuerda de guitarra a un sonido MIDI, asegurarse que la comunicación entre el Arduino y el instrumento final se realice correctamente, conectando a otro puerto MIDI en el instrumento final.
- En caso de que el sistema presente ejecuciones erróneas, volver a cargar el código a la tarjeta ARDUINO y corroborar su funcionamiento.
- En caso de que el sistema no sea compatible con el instrumento final, asegurarse de que el instrumento final utilice el protocolo de comunicación MIDI.
- En caso de que algún componente del circuito de adquisición de datos se encuentre en fallo, medir continuidad en los componentes pasivos para verificar su funcionamiento o si se encuentra en corto.
- En caso de que falle el CHIP LM386 medir voltajes de alimentación y de entrada y de salida para asegurarse de que es el chip y procede al cambio.

Fuente: elaboración propia, realizado con Microsoft Word y Arduino IDE.