

## RAE

1. **TIPO DE DOCUMENTO:** Trabajo de grado para optar por el título de INGENIERO DE SONIDO.
2. **TÍTULO:** DISEÑO Y CONSTRUCCIÓN DE SISTEMA DE DETECCIÓN DE TEMPO DE BATERÍA Y SINCRONIZACIÓN CON DAW
3. **AUTORES:** Alexander José Ramírez Marquis, Nicolas Rodríguez Zamora.
4. **LUGAR:** Bogotá, DC.
5. **FECHA:** Septiembre de 2017.
6. **PALABRAS CLAVE:** Detección de tempo, BPM, Función de Autocorrelación, Batería, Dispositivo Portátil.
7. **DESCRIPCIÓN DEL TRABAJO:** En este trabajo de investigación se desarrolló un sistema para la detección de tempo de patrones rítmicos interpretados en una batería y la sincronización de pistas musicales dispuestas en un DAW en tiempo real. La investigación se dividió en tres segmentos principales. En la primera fase, se evaluaron diferentes transductores de entrada para adquirir la señal de la batería y determinar el mejor método de captura. En la segunda fase, se desarrolló el software de detección, basado en el método de autocorrelación (ACF) para la estimación de las periodicidades, utilizando las señales obtenidas de la batería. Por último, se evaluó la efectividad del dispositivo y la facilidad de uso del mismo mediante pruebas objetivas y subjetivas.
8. **LÍNEAS DE INVESTIGACIÓN:** El núcleo problemático dentro del cual se sitúa el proyecto de investigación es el de análisis y procesamiento de señales, pues el dispositivo captura señales analógicas y las procesa por medio de métodos electrónicos y por medio del algoritmo de detección de tempo. El alcance de la investigación es de tipo correlacional.  
Este núcleo problemático se encuentra dentro de la línea de investigación de la facultad de ingeniería denominada “Análisis y procesamiento de señales”.
9. **METODOLOGÍA:** Para el desarrollo de este proyecto se parte de un enfoque cuantitativo de investigación, dado que en este caso los fenómenos analizados, el tempo y la sincronización, pertenecen a una realidad objetiva la cual es independiente de las creencias que se tengan acerca de estos. Además, debido a que los datos son producto de mediciones, se representan mediante números y se deben analizar con métodos estadísticos.
10. **CONCLUSIONES:** La disciplina de extracción de información de la música (MIR) es un campo de investigación relativamente pequeño y novedoso para el momento en el que se redactó este documento. La extracción del tempo a partir de señales musicales es una tarea compleja para una máquina, a pesar de ser tan simple para un ser humano, y es por eso que este campo de investigación se encuentra en crecimiento.  
Gracias a los pioneros investigadores en el área, se recolectó gran cantidad de información que facilitó el desarrollo del dispositivo. Se logró implementar un sistema de detección de tempo capaz de grabar, procesar, realizar una estimación de tempo y sincronizar pistas secuenciadas en un DAW.

DISEÑO Y CONSTRUCCIÓN DE SISTEMA PARA DETECCIÓN DE TEMPO  
DE BATERÍA Y SINCRONIZACIÓN CON DAW

Alexander José Ramírez Marquís

Nicolás Rodríguez Zamora

UNIVERSIDAD DE SAN BUENAVENTURA  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE SONIDO

2017

DISEÑO Y CONSTRUCCIÓN DE SISTEMA PARA DETECCIÓN DE TEMPO  
DE BATERÍA Y SINCRONIZACIÓN CON DAW

Alexander José Ramírez Marquís

Nicolás Rodríguez Zamora

PROYECTO DE GRADO

Asesor:

Ing. Belman Jahir Rodríguez Niño

UNIVERSIDAD DE SAN BUENAVENTURA

FACULTAD DE INGENIERÍA

INGENIERÍA DE SONIDO

2017

## CONTENIDO

<b>LISTA DE TABLAS .....</b>	<b>- 12 -</b>
<b>LISTA DE ECUACIONES .....</b>	<b>- 14 -</b>
<b>INTRODUCCIÓN .....</b>	<b>- 15 -</b>
<b>1. PLANTEAMIENTO DEL PROBLEMA.....</b>	<b>- 17 -</b>
<b>1.1 Antecedentes .....</b>	<b>- 17 -</b>
<b>1.2 Descripción y formulación del problema.....</b>	<b>- 19 -</b>
<b>1.3 Justificación.....</b>	<b>- 19 -</b>
<b>1.4 Objetivos de la investigación .....</b>	<b>- 20 -</b>
1.4.1 Objetivo General .....	- 20 -
1.4.2 Objetivos Específicos.....	- 20 -
<b>1.5 Alcances y limitaciones del proyecto.....</b>	<b>- 20 -</b>
1.5.1 Alcances .....	- 20 -
1.5.2 Limitaciones.....	- 21 -
<b>2. METODOLOGÍA .....</b>	<b>- 22 -</b>
<b>3. LÍNEA DE INVESTIGACIÓN DE LA UNIVERSIDAD / LÍNEA DE INVESTIGACIÓN DE LA FACULTAD / NÚCLEOS PROBLÉMICOS .....</b>	<b>- 24 -</b>
<b>4. MARCO DE REFERENCIA .....</b>	<b>- 25 -</b>
<b>4.1 Marco teórico – conceptual.....</b>	<b>- 25 -</b>
4.1.1 Tempo .....	- 25 -
4.1.2 Metrónomo.....	- 26 -
<b>4.1.3 Sistemas de inducción de tiempo.....</b>	<b>- 26 -</b>

4.1.4	Micrófono.....	- 31 -
4.1.5	Sensor Piezoeléctrico.....	- 34 -
4.1.6	Dígitos Binarios .....	- 35 -
4.1.7	Comparador.....	- 36 -
4.1.8	Diodo .....	- 38 -
4.1.9	Raspberry Pi 3.....	- 41 -
4.1.10	Python.....	- 41 -
4.1.11	Programación Multithreading.....	- 41 -
4.1.12	DAW .....	- 42 -
4.1.13	Protocolo MIDI.....	- 43 -
4.1.15	Protocolo OSC.....	- 43 -
<b>5</b>	<b>DESARROLLO INGENIERIL .....</b>	<b>- 45 -</b>
5.1	Diseñar el sistema de adquisición de señales percutivas .....	- 45 -
5.1.1	Determinar transductor de captura .....	- 45 -
5.1.2	Diseño de circuito de adquisición de señales y conversión análogo-digital .....	- 48 -
5.2	Diseñar el algoritmo de detección de tempo e implementarlo en un dispositivo portátil .....	- 61 -
5.2.1	Requerimientos del Software .....	- 62 -
5.2.2	Desarrollo.....	- 64 -
5.2.3	Evaluación para la Sincronización .....	- 75 -
5.3	Evaluar la funcionalidad del dispositivo mediante la comparación de pruebas objetivas y subjetivas .....	- 88 -
<b>6</b>	<b>PRESENTACIÓN Y ANÁLISIS DE RESULTADOS .....</b>	<b>- 96 -</b>

<b>7</b>	<b>CONCLUSIONES.....</b>	<b>- 101 -</b>
<b>8</b>	<b>RECOMENDACIONES.....</b>	<b>- 102 -</b>
<b>9</b>	<b>BIBLIOGRAFÍA.....</b>	<b>- 104 -</b>
<b>10</b>	<b>ANEXOS .....</b>	<b>- 109 -</b>

## LISTA DE FIGURAS

Figura 1. Tempo de 60BPM.....	- 25 -
Figura 2. Bloques de inducción de tempo generales (Fuente propia).....	- 28 -
Figura 3. Sistema de ventanas (Fuente propia).....	- 29 -
Figura 4. Segmentación en partes iguales y comparación (Fuente propia).....	- 30 -
Figura 5. Construcción de un micrófono dinámico ( Fuente: Sound Reinforcement Handbook – Davis).....	- 32 -
Figura 6. Construcción de un micrófono condensador ( Fuente: Sound Reinforcement Handbook – Davis) .....	- 33 -
Figura 7. Construcción de un micrófono piezoeléctrico ( Fuente: Sound Reinforcement Handbook – Davis) .....	- 34 -
Figura 8. Niveles lógicos de voltaje.....	- 35 -
Figura 9. Niveles lógicos de voltaje TTL.....	- 36 -
Figura 10. Niveles lógicos de voltaje CMOS .....	- 36 -
Figura 11. Esquema general de circuito comparador.....	- 37 -
Figura 12. Símbolo eléctrico del diodo.....	- 39 -
Figura 13. Regiones de operación del diodo.....	- 40 -
Figura 14. Símbolo eléctrico del diodo zener .....	- 40 -
Figura 15. Circuito de diodo zener.....	- 40 -
Figura 16. Comparación de la forma de onda de la señal generada por los transductores al recibir una señal de excitación .....	- 46 -
Figura 17. Sistema de adquisición de señales .....	- 49 -



Figura 18. Configuración de pines LM339N .....	50 -
Figura 19. Diagrama de bloques funcional .....	50 -
Figura 20. Tiempo de respuesta vs Voltaje de salida.....	51 -
Figura 21. Diagrama electrónico del LM339N (1 comparador) .....	52 -
Figura 22. Voltaje respecto a corriente y temperatura en diodo zener BZX55C3V3 .....	53 -
Figura 23. Diagrama electrónico BZX55C-3V3.....	54 -
Figura 24. Diagrama de bloques del experimento .....	54 -
Figura 25. Comparador LM339N, señal de entrada: Sinusoidal (amarillo).....	55 -
Figura 26. Comparador LM339N, señal de entrada: Diente de Sierra (amarillo).....	56 -
Figura 27. Diodo Zener 3.3V, señal de entrada: Sinusoidal (amarillo) .....	57 -
Figura 28. Diodo Zener 3.3V, señal de entrada: Diente de Sierra (amarillo) .....	58 -
Figura 29. Circuito LM339N (izquierda) y Diodo Zener 3.3V (derecha) .....	58 -
Figura 30. Pulsos de salida para una señal de entrada Sinusoidal, .....	59 -
Figura 31. Pulsos de salida para una señal de entrada Diente de Sierra, .....	60 -
Figura 32. Pulsos de salida para una señal de entrada Cuadrada, .....	60 -
Figura 33. Esquema de captura y almacenamiento de impulsos.....	63 -
Figura 34. Diagrama de Actividad UML para el Algoritmo de Detección.....	64 -
Figura 35. Sistema de grabación con ventanas ( Fuente propia).....	67 -
Figura 36. División del buffer (Fuente propia) .....	68 -
Figura 37. Desplazamiento de ventana móvil (Región Azul).....	69 -
Figura 38. Desplazamiento completo de la ventana móvil .....	70 -
Figura 39. Diagrama de flujo para clasificación de valores de BPM.....	71 -
Figura 40. Regiones A, B y C para definir tiempos de sincronización.....	76 -

Figura 41. Rango de detección de BPM (Fuente propia).....	76 -
Figura 42. Esquema de sistema OSC (Fuente propia) .....	84 -
Figura 43. Esquema Midi Clock (Fuente propia).....	85 -
Figura 44. Esquema final de sistema de sincronización .....	87 -
Figura 45. Interfaz Gráfica (Fuente propia) .....	87 -
Figura 46. Calificación pregunta No. 1 - Evaluación sincronización .....	92 -
Figura 47. Calificación pregunta No 2 - Evaluación sincronización .....	92 -
Figura 48. Calificación Pregunta No. 1 - Evaluación facilidad de uso .....	93 -
Figura 49. Calificación Pregunta No 2-a - Evaluación facilidad de uso.....	93 -
Figura 50. Calificación Pregunta No. 2-b - Evaluación facilidad de uso.....	94 -
Figura 51. Calificación Pregunta No. 2-c - Evaluación facilidad de uso.....	94 -
Figura 52. Circuito comparador .....	96 -
Figura 53. Referencia.....	97 -
Figura 54. Circuito final del proyecto.....	97 -
Figura 55. Porcentaje de Error de Detección de Tempo Promedio en Prueba Preliminar-	98 -
Figura 56. Porcentaje de Error de Detección de Tempo Promedio en Personas Encuestadas -	98 -
Figura 57. Tiempo de Grabación + Procesamiento Expresado en Compases .....	99 -
Figura 58. Tiempo de Grabación + Procesamiento Expresado en Compases .....	100 -
Figura 59. Tiempo de Grabación + Procesamiento Expresado en Compases .....	100 -
Figura 60. Partitura para batería - Ritmo Rock.....	109 -
Figura 61. Partitura para batería - Ritmo Pop .....	109 -
Figura 62. Partitura para batería - Ritmo Balada .....	109 -

Figura 63. Partitura para batería - Ritmo Funk .....- 110 -

Figura 64. Partitura para batería - Ritmo Reggae .....- 110 -

## LISTA DE TABLAS

Tabla 1. Rangos de tiempo.....	- 26 -
Tabla 2. Algoritmos de detección de tiempo. ....	- 28 -
Tabla 3. Tipos de Diagrama UML .....	- 42 -
Tabla 4. Aspectos de comparación entre transductores (Fuente propia) .....	- 48 -
Tabla 5. Referencias de los comparadores y diodo zener .....	- 50 -
Tabla 6. Comparación de circuito LM339N y Diodo Zener 3.3V .....	- 61 -
Tabla 7. Comparación de tempo fundamental con armónicos y subarmónicos.....	- 73 -
Tabla 8. Puntuación a valores de tempo fundamental.....	- 74 -
Tabla 9. Tiempo de Grabación para un tempo inicial de 60BPM.....	- 79 -
Tabla 10. Tiempo de grabación para un tempo inicial de 200 BPM.....	- 79 -
Tabla 11. Tiempo de grabación para un tempo inicial de 200 BPM.....	- 79 -
Tabla 12. Tiempo de grabación para un tempo inicial de 200 BPM.....	- 80 -
Tabla 13. Tiempo total (grabación + procesamiento) para un tempo inicial de 60 .....	- 80 -
Tabla 14. Tiempo total (grabación + procesamiento) de grabación para un tempoinicial de 200 BPM .....	- 80 -
Tabla 15. Valores de tempo detectado promedio en diferentes géneros.....	- 81 -
Tabla 16. Porcentaje de error promedio en diferentes géneros.....	- 81 -
Tabla 17. Múltiplos enteros encontrados para diferentes figuras musicales para 80 BPM-	82
-	
Tabla 18. Protocolos de Comunicación en DAW (Fuente propia) .....	- 85 -
Tabla 19. Mensajes de Sistema en Tiempo Real.....	- 86 -
Tabla 20. Ritmos que se ejecutan para la evaluación de exactitud del dispositivo.....	- 89 -

Tabla 21. Categorías de calificación subjetiva para la encuesta de opinión de sincronización

..... - 90 -

Tabla 22. Categorías de calificación subjetiva para la encuesta de opinión de facilidad de uso,  
pregunta 1..... - 91 -

Tabla 23. Categorías de calificación subjetiva para la encuesta de opinión de facilidad de uso,  
pregunta 2..... - 91 -

## LISTA DE ECUACIONES

Ecuación 1. Autocorrelación para una señal discreta.....	- 29 -
Ecuación 2. Obtención del BPM.....	- 31 -
Ecuación 3. Cálculo de resistencia para regulación de corriente .....	- 53 -
Ecuación 4. Cálculo de BPM inicial .....	- 65 -
Ecuación 5. Mínimo Tiempo de detección del programa. ....	- 66 -
Ecuación 6. Máximo Tiempo de detección del programa.....	- 66 -
Ecuación 7. Desplazamiento .....	- 70 -
Ecuación 8. Cálculo de valores de tiempo .....	- 70 -
Ecuación 9. Comparación de submúltiplos.....	- 73 -
Ecuación 10. Comparación de múltiplos .....	- 73 -
Ecuación 11. Tamaño de Buffer de Grabación .....	- 77 -
Ecuación 12. Cálculo de porcentaje de error .....	- 81 -
Ecuación 13. Cálculo de diferencia de tiempo entre pulsos.....	- 86 -

## INTRODUCCIÓN

En el contexto de presentaciones musicales en vivo, algunas agrupaciones hacen uso de secuencias para complementar su sonido. Estas secuencias consisten en los audios de una grabación, separados de forma individual, listos para reproducir desde cualquier plataforma de grabación.<sup>1</sup>

Para la sincronización en tempo de estas secuencias con los músicos de la agrupación, es necesario el uso de metrónomo (click), pues este define el BPM (valor que determina la cantidad de golpes por minuto, por lo cual entre mayor sea el valor de BPM, mayor será la velocidad de interpretación de la canción y viceversa) al cual debe ser ejecutada la canción. El metrónomo está sincronizado al tempo de las secuencias y puede ser usado por cada músico de la agrupación o sólo por el baterista.

Sin embargo, cuando no se hace uso de un metrónomo, la referencia de tempo disponible son las secuencias. Esto no asegura que la sincronización en tempo, entre la agrupación y las secuencias, esté basada en algún BPM específico, sino en la percepción subjetiva de tempo de cada músico.

A través del dispositivo de detección de tempo en tiempo real se busca sincronizar las secuencias al tempo del baterista. De esta manera, aunque no se tiene un BPM definido como se hace por medio de un metrónomo, sí se define un método para capturar el BPM y de esta manera sincronizar las secuencias a ese tempo.

---

<sup>1</sup> (Secuencias.com, s.f.)

Ya que la batería puede ser analizada por medio de patrones temporales<sup>2</sup> se escoge este instrumento como la base del desarrollo del proyecto. Sin embargo, agrupaciones musicales que no usen batería no pueden hacer uso del dispositivo.

---

<sup>2</sup> (Goto, 2001)



## 1. PLANTEAMIENTO DEL PROBLEMA

### 1.1 Antecedentes

**Generation of musical scores of percussive un-pitched instruments for automatically detected events.**<sup>3</sup> Se investiga acerca de la generación de partituras musicales de instrumentos sin un pitch determinado, a través de eventos detectados automáticamente.

La detección de eventos hace parte fundamental de este proyecto dado que, para el algoritmo de detección de tempo, se realiza la detección de eventos por medio de un transductor y estos eventos son procesados para estimar el valor de tempo.

**Beat tracking towards automatic musical accompaniment.**<sup>4</sup> Se investiga la posibilidad del uso de un sistema de detección de patrones rítmicos para generar acompañamientos musicales automáticos. Se busca predecir patrones rítmicos que puedan ser coherentes y consistentes para poder acompañar a un intérprete musical en tiempo real.

Para el caso de este proyecto de investigación, el acompañamiento del músico o de la agrupación musical son las secuencias reproducidas desde una estación de trabajo, las cuales se sincronizan al tempo detectado por medio del dispositivo.

**Design of an automatic beat-matching algorithm for portable media devices.**<sup>5</sup> Se define la arquitectura, los métodos de diseño, obstáculos, optimizaciones y resultados de un algoritmo de detección de tempo en tiempo real empleado en dispositivos comerciales portátiles.

---

<sup>3</sup> (Uhle & Dittmar, 2004)

<sup>4</sup> (Davies, Brossier, & Plumbley, 2005)

<sup>5</sup> (Jochelson & Fedigan, 2006)

El algoritmo desarrollado en este proyecto se plantea como un detector de tempo en tiempo real, que se implemente en un dispositivo portátil, el cual permita la sincronización de secuencias al tempo detectado. La reproducción de estas secuencias se puede realizar en diferentes estaciones de trabajo (DAW) comercialmente usadas, siempre y cuando permitan modificar el tempo por medio de MIDI Clock.

**Audio effects for real-time performance using beat tracking.**<sup>6</sup> Se realiza una investigación acerca de la posibilidad de implementar efectos de audio en presentaciones musicales en vivo en donde se utilice una señal de entrada musical capaz de controlar parámetros como: tiempo de delay, frecuencia en LFO, Auto-wah, Tremolo y Vibrato en tiempo real.

El dispositivo propuesto envía el valor de BPM a la estación de trabajo por medio de protocolo OSC y luego por medio de protocolo MIDI, para sincronizar las secuencias al tempo detectado. Dado que el dispositivo detecta el tempo independientemente del protocolo que se use para enviarlo, este valor puede ser usado en diferentes aplicaciones como la sincronización de luces, la implementación en pedales de guitarra, la implementación en consolas, entre otros. Por lo tanto, cualquier aplicación diferente a la sincronización de secuencias se convierte en un alcance del proyecto.

**Improving perceived tempo estimation by statistical modeling of higher-level musical descriptors.**<sup>7</sup> Los algoritmos de estimación de tempo convencionales generalmente operan al detectar eventos de audio significativos y encontrando periodicidad en patrones repetitivos en la señal de audio.

---

<sup>6</sup> (Stark, Plumbley, & Davies, 2007)

<sup>7</sup> (Chen, Cremer, Lee, DiMaria, & Wu, 2009)

El algoritmo que se propone para el desarrollo del proyecto se basa en la captura de eventos significativos y la relación que hay entre patrones repetitivos. Además, se busca generar un algoritmo de mayor exactitud en la detección de tiempo. Sin embargo, es importante tener en cuenta que, dado que el dispositivo será usado por personas, el factor humano hace parte fundamental en la exactitud del dispositivo al detectar el tiempo.

## **1.2 Descripción y formulación del problema**

El uso de secuencias por agrupaciones musicales permite complementar el sonido de estas en presentaciones en vivo, ya que son audios que pueden ser reproducidos al mismo tiempo que la agrupación interpreta una canción. Además, son audios grabados de instrumentos que no están presentes físicamente por disponibilidad de los mismos o de músicos, entre otras razones.

Ahora, la sincronización entre las agrupaciones musicales y las secuencias se fundamenta en el uso de metrónomo. Este puede ser usado por todos los músicos de la agrupación musical o, en diversos casos, sólo por el baterista. Sin embargo, cuando no se hace uso de este, la sincronización depende de la percepción subjetiva de tiempo de los músicos y no de un dispositivo encargado de definir el BPM, lo cual da como consecuencia desincronización entre las secuencias y la agrupación musical.

¿Cómo sincronizar secuencias al tiempo de ejecución del baterista en tiempo real?

## **1.3 Justificación**

Con el dispositivo de detección de tiempo en tiempo real se provee una herramienta para el uso de secuencias en presentaciones en vivo sin necesidad de metrónomo, pues

permite sincronizar las secuencias al tempo de ejecución del baterista. Esto reduce notoriamente el problema de desincronización que existe al usar secuencias sin un dispositivo que esté enlazado al BPM de estas y por lo tanto que guíe a los músicos en la velocidad de ejecución de la canción.

El dispositivo permite sincronizar secuencias al tempo del baterista en tiempo real por medio de un algoritmo de detección de tempo junto con un protocolo de comunicación entre el algoritmo y la estación de trabajo (DAW).

## **1.4 Objetivos de la investigación**

### 1.4.1 Objetivo General

- Diseñar y construir un sistema portátil que analice el tempo de una batería para sincronizar pistas secuenciadas en un DAW.

### 1.4.2 Objetivos Específicos

- Diseñar el sistema de adquisición de señales percutivas.
- Diseñar el algoritmo de detección de tempo e implementarlo en un dispositivo portátil.
- Evaluar la funcionalidad del dispositivo mediante la comparación de pruebas objetivas y subjetivas.

## **1.5 Alcances y limitaciones del proyecto**

### 1.5.1 Alcances

Con el dispositivo desarrollado en este proyecto de investigación se proporciona una herramienta que permite el uso de secuencias, por parte de agrupaciones musicales para

presentaciones en vivo, sin necesidad del uso de metrónomo, sino por medio de la sincronización de las secuencias al BPM ejecutado por el baterista.

### 1.5.2 Limitaciones

La agrupación musical que requiera hacer uso del dispositivo debe tener entre sus instrumentos principales la batería acústica.

## 2. METODOLOGÍA

Para el desarrollo de este proyecto se parte de un enfoque cuantitativo de investigación, dado que en este caso los fenómenos analizados, el tempo y la sincronización, pertenecen a una realidad objetiva la cual es independiente de las creencias que se tengan acerca de estos. Además, debido a que los datos son producto de mediciones, se representan mediante números y se deben analizar con métodos estadísticos.<sup>8</sup>

El alcance de la investigación es de tipo correlacional ya que se estudian varios fenómenos para luego analizarlos y establecer la relación entre ellos. Para el caso de este proyecto, como fenómeno principal se estudia el tempo y se desarrolla un dispositivo capaz de capturar las señales percutivas y procesarlas por medio de un algoritmo de detección. Como fenómeno secundario, se estudia la sincronización para luego vincular los dos conceptos en la implementación final del dispositivo y su aplicación en la sincronización de secuencias al tempo del baterista.

El diseño de investigación se compone de un diseño experimental y de otro no experimental. Respecto al diseño experimental, su esencia es que se manipulan intencionalmente una o más variables independientes para analizar las consecuencias que la manipulación tiene sobre una o más variables dependientes dentro de una situación de control para el investigador.<sup>9</sup> Para la aplicación en este proyecto, se manipula de manera intencional el tempo (variable independiente), con el fin de analizar los posibles resultados que se adquieren cuando el dispositivo realiza la respectiva captura del tempo (variable dependiente). La manipulación de la variable independiente se realiza en más de dos grados,

---

<sup>8</sup> (Hernandez Sampieri, 2014)

<sup>9</sup> *Ibíd.*

lo que permite analizar si distintos niveles de la variable independiente producen diferentes efectos

Ahora, respecto al diseño no experimental, consiste en estudios en los que no se varían de forma intencional las variables independientes para ver su efecto sobre otras variables, sino que se observan los fenómenos tal como se dan en su contexto natural, con el fin de analizarlos.<sup>10</sup> La aplicación de este diseño de investigación al presente proyecto, se enfoca en la realización de un experimento junto con una encuesta de opinión, para la evaluación de la funcionalidad del dispositivo, dado que no se puede modificar la experiencia de los sujetos de prueba tocando batería (variable independiente). Además, el tipo de diseño no experimental seleccionado es el transaccional descriptivo, pues se recolectan datos de un solo momento, en un tiempo único, para este caso el momento del experimento y la encuesta, y se indagan la incidencia de los niveles de una o más variables, que para este caso son la intención del sujeto de prueba de mantener el tiempo, acelerar el tiempo y retrasar el tiempo.

---

<sup>10</sup> *Ibíd.*

### **3. LÍNEA DE INVESTIGACIÓN DE LA UNIVERSIDAD / LÍNEA DE INVESTIGACIÓN DE LA FACULTAD / NÚCLEOS PROBLÉMICOS**

El núcleo problémico dentro del cual se sitúa el proyecto de investigación es el de análisis y procesamiento de señales, pues el dispositivo captura señales analógicas y las procesa por medio de métodos electrónicos y por medio del algoritmo de detección de tempo.

Este núcleo problémico se encuentra dentro de la línea de investigación de la facultad de ingeniería denominada “Análisis y procesamiento de señales”.

En último lugar, la línea de investigación de la universidad a la cual pertenece el proyecto de investigación, relacionado con los puntos nombrados anteriormente, es la línea de tecnologías actuales y sociedad, pues se busca dar solución al problema planteado al cual se ven expuestas todas aquellas personas o agrupaciones musicales que hacen uso de secuencias para presentaciones musicales en vivo.



## 4. MARCO DE REFERENCIA

### 4.1 Marco teórico – conceptual

#### 4.1.1 Tempo

En música, el tempo se conoce como la velocidad en la que se interpreta o se reproduce una pieza musical determinada<sup>11</sup> (Kornfeld, 2005). Comúnmente, el tempo es presentado como un número entero, en términos de BPM, el cual hace referencia a la cantidad de “pulsos” que existen por cada minuto (Ej: Un tempo de 60BPM equivale a 1 pulso por segundo o 60 pulsos por minuto). Generalmente, en una partitura musical, el tempo es indicado al inicio de la composición como se muestra en la Imagen 1.



*Figura 1. Tempo de 60BPM*

La duración musical de un “pulso” está definida por el compás de tiempo en la partitura musical, donde el número superior indica la cantidad de pulsos que existen en 1 compas y el número inferior la duración de 1 pulso en términos de figuras musicales<sup>12</sup> (Ej: En la imagen 1, se tiene una partitura musical con 4 pulsos por compas donde cada pulso tendrá una duración de 1/4 de nota o negra). Tradicionalmente, se utilizan palabras en italiano para clasificar los rangos de tempo de muy lento a muy rápido, como se muestra en la Imagen 2.

---

<sup>11</sup> (Kornfeld, 2005)

<sup>12</sup> (Harnum, 2001)

Largo	40 – 60
Larghetto	60 - 66
Adagio	66- 76
Andante	76 – 108
Moderato	108 – 120
Allegro	120 – 169
Presto	169 – 200
Prestissimo	200 +

*Tabla 1. Rangos de tempo.*

#### **4.1.2 Metrónomo**

El metrónomo es un dispositivo mecánico o digital capaz de reproducir pulsos consistentes en forma de sonido o “clicks”, dado un valor de BPM. Este aparato es utilizado ampliamente por los músicos para mejorar sus habilidades rítmicas, usualmente en el rango de 40 y 200 BPM.<sup>13</sup>

#### **4.1.3 Sistemas de inducción de tempo**

Se define como un sistema de inducción de tempo como la serie de procesos requeridos para la extracción de características rítmicas, como lo es el tempo y la detección de pulsos, a partir de una señal de entrada, generalmente un archivo de audio.<sup>14</sup> Estos sistemas se basan en algoritmos computacionales y constituyen de tres partes fundamentales:

---

<sup>13</sup> *Ibíd.*

<sup>14</sup> (Gouyon & Dixon, 2005)

1. **Señal de entrada:** Contiene la información musical que será analizada, generalmente en forma de un archivo de audio. Algunos sistemas computacionales pueden manejar información simbólica como señal de entrada, como el MIDI y partituras musicales transcritas manualmente que contengan únicamente la ubicación de las figuras musicales.<sup>15</sup>
2. **Creación de lista de características:** En esta sección se transforman los datos de la señal de entrada en una serie de características que representan la información rítmica predominante. Para el tempo, se requieren características del inicio de los pulsos, tales como el tiempo, la duración y la amplitud.<sup>16</sup>
3. **Inducción de Pulsos:** Se toma la lista de características rítmicas generada en la sección anterior para la detección de patrones rítmicos mediante un algoritmo determinado. En esta etapa, los programas computacionales pueden proceder utilizando un método de selección de pulsos, el cual evalúa la importancia de un número restringido de posibles periodicidades<sup>17</sup>, o a través de una función de periodicidad, la cual analiza la amplitud de los pulsos versus la frecuencia de los mismos.

---

<sup>15</sup> (Brown, 1993)

<sup>16</sup> (Gouyon & Dixon, 2005)

<sup>17</sup> (Parncutt, 1994)

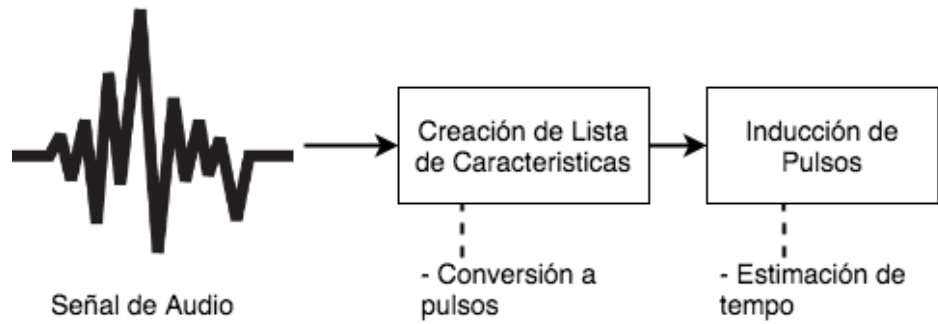


Figura 2. Bloques de inducción de tiempo generales (Fuente propia)

Las funciones de periodicidad son comúnmente calculadas mediante algoritmos estándares de procesamiento de señal, como la transformada de Fourier, en donde Blum et al. (1999) aplicó estas a la lista de pulsos y Pampalk et al. (2002) la utilizó en 20 sub-bandas de frecuencia de la señal de audio. Una de las técnicas más utilizadas para el procesamiento de periodicidad de señales es la función de auto-correlación, la cual es aplicada listas de pulsos representados como funciones delta Dirac o información simbólica.<sup>18</sup>

A continuación, se presentan algunos de los algoritmos de detección de tiempo existentes.

<b>Algoritmo</b>	<b>Tzanetakis</b>	<b>Beatroot</b>	<b>DixonACF</b>	<b>MIRTempo</b>	<b>AlonsoSP</b>
<b>Autor</b>	Tzanetakis	Dixon	Dixon	Lartillot	Alonso
<b>Infraestructura</b>	Marsyas	Java	Matlab	Matlab	Matlab
<b>Inducción de Pulso</b>	ACF	IOI Clustering	ACF	ACF	Producto Espectral

Tabla 2. Algoritmos de detección de tiempo.<sup>19</sup>

<sup>18</sup> (Gouyon & Dixon, 2005)

<sup>19</sup> (Zapata & Gómez, 2011)

#### 4.1.3.1 Función de Autocorrelación

La autocorrelación, se precisa como una herramienta matemática capaz de medir la similitud que existe entre una señal y una versión con un desfase en tiempo de la misma.

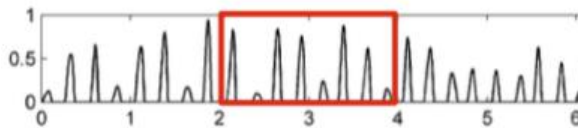
Para una señal discreta, la función de autocorrelación  $R_{xx}$  dependiente de un valor de retraso en el tiempo  $\ell$ , se puede definir como:

$$R_{xx}(\ell) = \sum_{m \in \mathbb{Z}} x(m)x(m - \ell)$$

*Ecuación 1. Autocorrelación para una señal discreta*

Analizando esta ecuación, se encuentra que  $R_{xx}(\ell)$  será máximo en  $\ell = 0$  puesto que se estarán comparando las mismas señales. A medida que el retraso en tiempo  $\ell$  incremente, se calcula un nuevo valor de correlación en donde se podrá observar la relación entre los datos de la misma señal en diferentes instantes de tiempo. En las siguientes figuras se muestra un ejemplo, en donde se utiliza esta herramienta para encontrar periodicidad.

Se tiene una señal discreta con una duración de 6 segundos, conformada por impulsos extraídos de la forma de onda de un archivo de audio.



*Figura 3. Sistema de ventanas (Fuente propia)*

Para encontrar la periodicidad dentro de esta señal, se define una ventana rectangular de longitud finita  $[-L: L]$ , centrada en la serie de datos, como lo muestra

la figura 1 en el recuadro rojo. El tamaño de la ventana es definido como un tercio de la duración total de la tira de datos, para el ejemplo es de 2 segundos. Posteriormente, se calcula la autocorrelación entre los datos incluidos en la ventana definida y la misma señal, el retraso en tiempo  $\ell$  es aplicado a la ventana de datos para nuevamente realizar el cálculo.

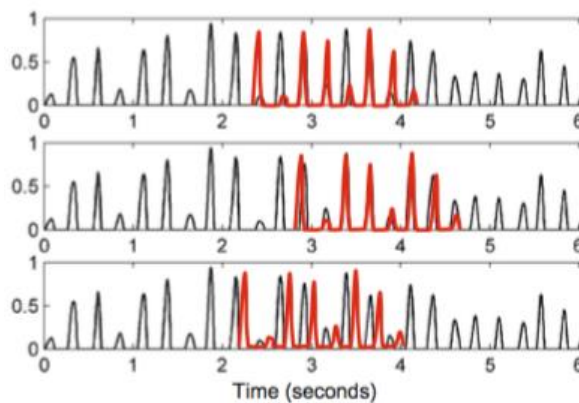


Figura 4. Segmentación en partes iguales y comparación (Fuente propia)

En la figura 2 se muestra el desplazamiento en tiempo de la sección de señal que incluía la ventana y pueden observarse ligeramente los grados de similitud que posee con otras secciones de la señal. Teniendo en cuenta la simetría de la ventana, los valores de retraso en tiempo  $\ell$  que deben considerarse son  $\ell = 0 : 2L$ . Los valores de retraso  $\ell$  donde se encuentren altos niveles de correlación (cerca de 1) serán los indicadores de periodicidad.<sup>20</sup>

Para convertir estos valores a una escala de tiempo, en unidades de BPM, se aplica la siguiente ecuación:

---

<sup>20</sup> (Müller, 2015)

$$\tau = \frac{60}{r * \ell}$$

*Ecuación 2. Obtención del BPM.*

Donde  $\ell$  corresponde al retraso de la señal a comparar dado en muestras y  $r$  el tiempo que exista entre muestras, es decir, el periodo de la frecuencia de muestreo.

Como lo plantean los autores Gouyon y Dixon<sup>21</sup>, algunos métodos son susceptibles a que se encuentren múltiples periodicidades relacionadas entre sí (Ej:  $\tau, 2\tau, 3\tau$ ). El autor Müller<sup>22</sup>, confirma lo anterior en el método de autocorrelación.

En cuanto a la frecuencia de muestreo, los valores típicos se encuentran alrededor de 100Hz y 200Hz.<sup>23</sup>

#### **4.1.4 Micrófono**

Micrófono es un término general usado para referirse a cualquier elemento que transforme energía acústica (sonido) en energía eléctrica (la señal de audio). Se han desarrollado diferentes métodos para la transformación de energía según el uso y propósito.<sup>24</sup> A continuación se explican los tipos de micrófonos usados en el presente documento, según el método para transformar energía acústica en eléctrica:<sup>25</sup>

- Dinámico: es el tipo de micrófono más común, semejante a un altavoz. La figura a continuación muestra el funcionamiento de este.<sup>26</sup>

---

<sup>21</sup> (Gouyon & Dixon, 2005)

<sup>22</sup> (Müller, 2015)

<sup>23</sup> (Gouyon & Dixon, 2005)

<sup>24</sup> (Davis & Jones, 1990)

<sup>25</sup> *Ibíd.*, pág 113

<sup>26</sup> *Ibíd.*, pág. 113

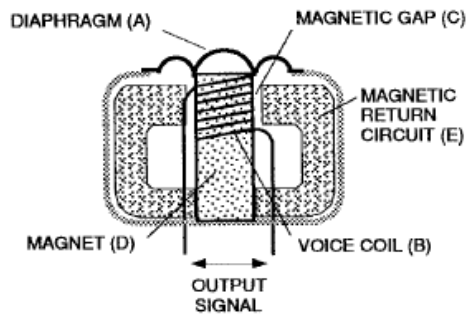


Figura 5. Construcción de un micrófono dinámico ( Fuente: *Sound Reinforcement Handbook – Davis*)

Un diafragma flexible, Figura 5(a), es acoplado a una bobina de alambre fino (b). La bobina está montada en el espacio de aire de un imán (c) de manera que se puede mover libremente hacia delante y hacia atrás.<sup>27</sup>

Cuando el sonido golpea el diafragma, su superficie vibra. Como está acoplado a la bobina, ésta se mueve hacia delante y hacia atrás. Este movimiento dentro del espacio de aire genera una corriente inducida en el alambre. La magnitud y dirección de esta corriente está directamente relacionada con el movimiento de la bobina por lo que es una representación de la onda de sonido incidente.<sup>28</sup>

- Condensador: seguido al dinámico, el micrófono de condensador es el más común. La a continuación muestra la construcción de un elemento condensador.<sup>29</sup>

<sup>27</sup> *Ibíd.*, pág. 113

<sup>28</sup> *Ibíd.*, pág 113

<sup>29</sup> *Ibíd.*, pág. 113



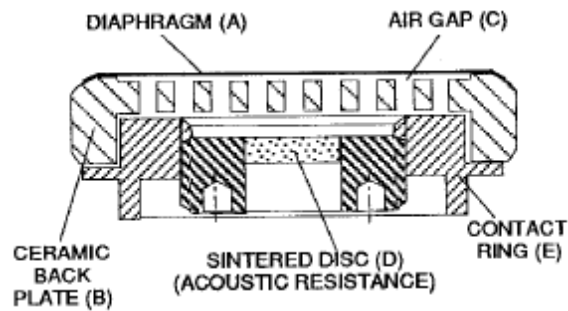


Figura 6. Construcción de un micrófono condensador ( Fuente: Sound Reinforcement Handbook – Davis)

Un diafragma de plástico recubierto de oro, Figura 6(a), es montado sobre una placa posterior conductiva (b), que en ocasiones es hecha de cerámica bañada en oro. El diafragma y la placa posterior, separados por un pequeño espacio de aire (c), forman un componente eléctrico llamado capacitor (o condensador).<sup>30</sup>

Un voltaje de polarización de entre 9 y 48 voltios es aplicado al diafragma por una fuente de poder externa, cargándolo con un voltaje estático.<sup>31</sup>

Cuando el diafragma vibra en respuesta a un sonido, este se mueve cerca y lejos de la placa posterior. Así, la carga eléctrica que se induce en la placa cambia proporcionalmente. Este voltaje fluctuante es la representación del movimiento del diafragma.<sup>32</sup>

Los elementos del micrófono de condensador producen una señal de voltaje muy pequeña, sin casi fuerza. De este modo representan una impedancia muy alta. Es por esto que el micrófono de condensador incorpora un amplificador, lo cual da fuerza a la línea del micrófono. Su función es tanto aumentar el nivel de la señal y aislar el elemento de la baja impedancia de la entrada a la cual el micrófono está conectado.<sup>33</sup>

---

<sup>30</sup> *Ibíd.*, pág 113

<sup>31</sup> *Ibíd.*, pág 113

<sup>32</sup> *Ibíd.*, pág 114

<sup>33</sup> *Ibíd.*, pág 114

#### 4.1.5 Sensor Piezoeléctrico

La figura a continuación muestra el principio de funcionamiento de los micrófonos piezoeléctricos.<sup>34</sup>

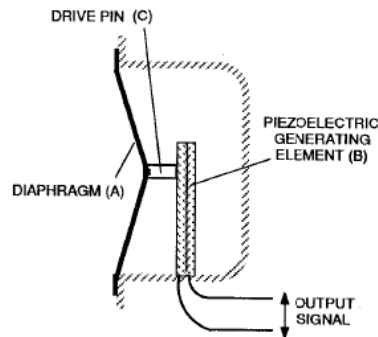


Figura 7. Construcción de un micrófono piezoeléctrico ( Fuente: Sound Reinforcement Handbook – Davis)

Un diagrama flexible, Figura 7(a), es acoplado a un cristal (b), por un perno de conducción (c). El cristal es de un material tal que manifiesta el efecto piezoeléctrico (presión-eléctrica). Cuando es deformado físicamente por presión o torsión, el cristal genera un voltaje eléctrico (potencial) a través de sus caras.<sup>35</sup>

Cuando el sonido golpea el diafragma este vibra, y el cristal, sólo así, es ligeramente deformado. El cristal genera un voltaje en respuesta a esta flexión, y su variación de voltaje es una representación eléctrica del sonido.<sup>36</sup>

Este tipo de micrófono no es reconocido por su calidad en sonido, pero si es bien implementado, el elemento de cristal puede comportarse muy bien. Este método es usado con frecuencia para micrófonos de contacto.<sup>37</sup>

---

<sup>34</sup> *Ibíd.*, pág 114

<sup>35</sup> *Ibíd.*, pág 116

<sup>36</sup> *Ibíd.*, pág 116

<sup>37</sup> *Ibíd.*, pág 116

#### 4.1.6 Dígitos Binarios

En la electrónica digital, los circuitos operan en sistemas en donde solo existen dos posibles estados, “alto” (HIGH) y “bajo” (LOW), conocido como un sistema binario. Este sistema, mediante la combinación de niveles “altos” y “bajos” o 1 y 0, se pueden representar números, símbolos, caracteres alfabéticos, instrucciones, entre otros.<sup>38</sup>

##### 4.1.6.1 Niveles lógicos de voltaje

Los voltajes utilizados para representar los 1 y 0 se denominan voltajes lógicos. Idealmente un nivel de voltaje representará el estado de “alto” y otro el de “bajo”, sin embargo, en la práctica existen rangos de voltaje para representar ambos estados como se muestra en la siguiente figura:<sup>39</sup>

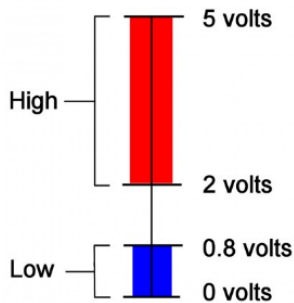


Figura 8. Niveles lógicos de voltaje

Existen múltiples tecnologías de circuitos digitales que manejan niveles de voltaje lógicos diferentes tales como el TTL y CMOS. Para el desarrollo el proyecto es importante tener en cuenta las regiones de voltaje de la tecnología CMOS, puesto que el computador donde se implementará el sistema de

---

<sup>38</sup> (Floyd, 2015)

<sup>39</sup> *Ibíd.*

detección de tiempo (Raspberry Pi) opera bajo estas regiones. A continuación, se presentaran los rangos de voltaje para TTL y CMOS.

#### 4.1.6.2 TTL

Los niveles lógicos de la tecnología TTL se muestran en la figura a continuación.

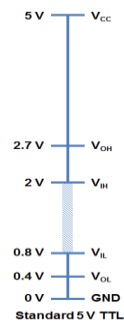


Figura 9. Niveles lógicos de voltaje TTL

#### 4.1.6.3 CMOS

Los niveles lógicos para CMOS se muestran en la siguiente figura.

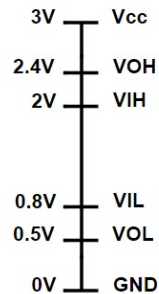


Figura 10. Niveles lógicos de voltaje CMOS

### 4.1.7 Comparador

Se conoce comparador como un circuito integrado diseñado especialmente para la tarea de comparar 2 señales de entrada y emitir un voltaje de salida determinado.

La salida del comparador se encontrará en un estado “alto” cuando  $V_{IN+} > V_{IN-}$  y bajo para  $V_{IN+} < V_{IN-}$

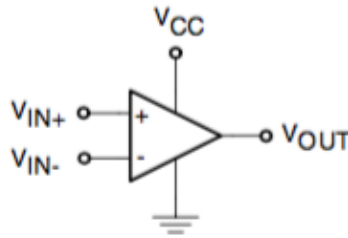


Figura 11. Esquema general de circuito comparador

El amplificador operacional es diseñado con el propósito de ser utilizado con una retroalimentación negativa y por esta razón, para asegurar su estabilidad en esta configuración, parámetros como el slew rate y el ancho de banda máximo son deteriorados.<sup>40</sup> Algunos de las características más importantes al momento de seleccionar un circuito comparador son presentadas a continuación.

#### 4.1.7.1 Delay de propagación:

El delay de propagación  $T_{PD}$  indica la frecuencia máxima que puede ser procesada a la entrada del comparador. Este se define como el intervalo de tiempo entre el momento en el que la señal de entrada supera el nivel de referencia y el estado de la salida cambia.<sup>41</sup>

#### 4.1.7.2 Slew Rate

En un amplificador operacional, el slew rate se define como la razón de cambio del voltaje en la salida por unidad de tiempo, es decir, que tan rápido se dan los

---

<sup>40</sup> (Smat, 2012)

<sup>41</sup> Ibíd.

cambios de estado en la salida del comparador, usualmente medido en voltios/microsegundos. Esta característica es optimizada en los circuitos comparadores puesto que es una propiedad fundamental en las aplicaciones de estos.

#### 4.1.7.3 Tipo de etapa de salida (Colector abierto / Push-Pull)

Existen dos principales tipos de configuración a la salida de los circuitos comparadores, colector abierto y push-pull. En comparación, la salida de colector abierto utiliza una resistencia de pull-up en el circuito, que permite utilizar un voltaje de salida diferente al de alimentación, mientras que la configuración push-pull mantiene un voltaje de salida fijo igual al de la fuente.<sup>42</sup>

#### 4.1.7.4 Voltaje de Overdrive

En comparadores, el voltaje de overdrive se determina como la diferencia del nivel entre la entrada no inversora (señal) y la entrada inversora (referencia). Este valor puede afectar el slew rate del comparador y por esta razón debe ser tomado en cuenta dependiendo del tipo de aplicación.<sup>43</sup>

### 4.1.8 Diodo

El diodo es un componente electrónico semiconductor de estado sólido constituido por una unión de un material tipo p y otro tipo n, el solo permite flujo de corriente en la dirección de Ánodo(+) a Cátodo(-). Para que exista un flujo de corriente en dicha dirección, el voltaje aplicado sobre el diodo debe superar una

---

<sup>42</sup> *Ibíd.*

<sup>43</sup> (Texas Instruments, 2017)

barrera de potencial de acuerdo el material en que se base el diodo, aproximadamente 0.7 para el Silicio y 0.3 para el Germanio.<sup>44</sup>

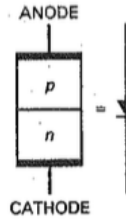


Figura 12. Símbolo eléctrico del diodo

Este componente posee dos regiones de operación definidas por el tipo de polarización:

1. Polarización Directa

La polarización directa es establecida al aplicar una diferencia de potencial positiva en el material tipo-p y negativa en el tipo-n, lo cual permitirá un flujo de corriente a través del diodo tras superar una barrera de potencial. Con este tipo de polarización, el diodo operará en región directa.

2. Polarización Inversa

Al invertir la polarización del diodo, no existirá un flujo de corriente a través del componente hasta que este llegue a la región de breakdown, en donde se producirá una gran corriente inversa que destruirá el diodo.<sup>45</sup> Antes de llegar a la región de breakdown, el diodo operará en región inversa.

---

<sup>44</sup> (Boylestad & Nashelsky, 2013)

<sup>45</sup> (Malvino & Bates)

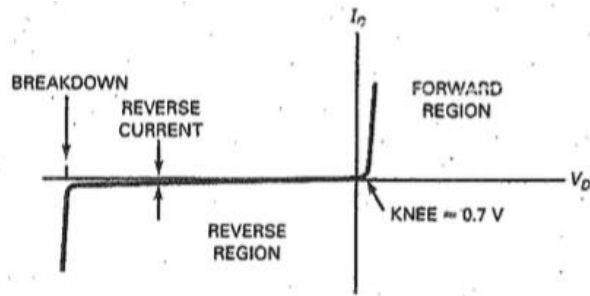


Figura 13. Regiones de operación del diodo

#### 4.1.8.1 Diodo Zener

Se conoce como Zener, a un tipo de diodo preferiblemente de silicon diseñado para operar en la región de breakdown, a diferencia del diodo común. Este tipo de diodo es comúnmente utilizado como regulador de voltaje dada su capacidad de mantener una salida de voltaje constante pese a variaciones de corriente.



Figura 14. Símbolo eléctrico del diodo zener

En un circuito de un diodo zener, debe colocarse una resistencia en serie para proteger al diodo de la corriente que lo atravesará.<sup>46</sup> En la figura 10 se muestra un circuito de regulación de voltaje utilizando un diodo zener, donde la resistencia  $R_S$  representa la resistencia que regula la corriente del circuito.



Figura 15. Circuito de diodo zener

<sup>46</sup> *Ibíd.*



### **4.1.9 Raspberry Pi 3**

La Raspberry Pi 3 es un computador desarrollado por la fundación The Raspberry Pi UK, con el propósito de proveer computadores de bajo costo y software gratuito para estudiantes. Este sistema posee una gran similitud a un computador común en el aspecto de hardware, al integrar 4 puertos USB, 1 puerto HDMI, 1 salida de audio estéreo, 1 puerto ethernet y modulo WIFI. Adicionalmente, cuenta con 40 pines multipropósitos y protocolos de comunicación como UART, I2C y SPI. A nivel de software, cuenta con Raspbian como sistema operativo, basado en Linux.<sup>47</sup>

### **4.1.10 Python**

Lenguaje de programación orientado a objetos de alto nivel, con gran popularidad en el ámbito académico por su sintaxis sencilla, el soporte, y sus extensas librerías distribuidas libremente en la mayoría de plataformas en el mercado. Actualmente es uno de los lenguajes más populares para desarrolladores de backend.<sup>48</sup>

### **4.1.11 Programación Multithreading**

La programación Multithreading o “múltiples hilos” consiste en la implementación de procesos capaces de realizar múltiples tareas de forma simultánea, trabajando por un fin común. En python, estos procesos pueden o no funcionar utilizando las mismas variables, siempre y cuando el programador conserve cierta prudencia al momento de acceder y modificar las variables. Los “threads” son conocidos como procesos ligeros, dado que los

---

<sup>47</sup> (Verry, 2012)

<sup>48</sup> (Python, s.f.)

sistemas generalmente requieren de menos recursos para crear y manipular un “thread”.<sup>49</sup> Para representar graficamente la relación entre los diferentes “threads” se recomiendan los diagramas de actividad UML.<sup>50</sup>

#### 4.1.12 Diagrama UML

Se conoce como UML (Unified Modelling Language) a una serie de simbologías graficas presentadas en un diagrama, que ayudan a describir y diseñar sistemas de software, especialmente programaciones orientadas a objetos. Estos diagramas pueden ser empleados para generar diseños en formas de bocetos, planos y hasta lenguajes de programación. El valor principal de los diagramas UML es la comunicación de un diseño y el entendimiento en un sistema. Existen múltiples tipos de diagramas UML para expresar el funcionamiento de los sistemas desde diferentes perspectivas.<sup>51</sup>

<b>Tipo de Diagrama</b>	<b>Propósito del Diagrama</b>
Diagrama de Clases	Demostrar clases, características y sus relaciones.
Diagrama de Maquina de Estados	Presenta el cambio de los objetos a lo largo de su vida.
Diagrama de Actividad	Utilizado para procedimientos y comportamientos paralelos
Diagrama de Secuencia	Enfatiza una secuencia y presenta detalles sobre la interacción entre objetos

*Tabla 3. Tipos de Diagrama UML*

#### 4.1.12 DAW

Sistema digital de audio diseñado para la grabación, edición y procesamiento de señales de audio y/o mensajes MIDI a nivel profesional a través de un software determinado.<sup>52</sup>

---

<sup>49</sup> (Deitel, 2002)

<sup>50</sup> (Fowler, 2003)

<sup>51</sup> *Ibíd.*

<sup>52</sup> (Diaconescu, 2016)

#### **4.1.13 Protocolo MIDI**

El protocolo MIDI, es un lenguaje digital de comunicación que permite la interacción entre múltiples dispositivos, instrumentos musicales electrónicos y softwares. Este protocolo se utiliza para traducir información como activaciones de nota en un teclado, control de botones, perillas o faders en mensajes digitales que permitan controlar otros dispositivos.<sup>53</sup>

#### **4.1.14 MIDI Clock**

Entre las múltiples posibilidades de control que ofrece el protocolo MIDI se encuentra la sincronización de secuenciadores con otros dispositivos. Esto, se realiza mediante una señal de reloj conocida como MIDI Clock, la cual se encarga de enviar una serie de pulsos desde el dispositivo maestro hacia el esclavo en intervalos regulares, basados en el tempo del dispositivo maestro. Para sincronizar un secuenciador esclavo con un dispositivo maestro externo, este debe enviar 24 pulsos por cada cuarto de compás.

#### **4.1.15 Protocolo OSC**

El protocolo OSC es un sistema de comunicación entre computadoras, sintetizadores de sonido y otros dispositivos multimedia, optimizado para las tecnologías de comunicaciones modernas. Este es implementado en una variedad de aplicaciones como control de procesamiento de audio en tiempo real, sintetizadores de software, dispositivos para medición de sensores, entre otros.<sup>54</sup>

#### **4.1.16 Muestreo por conveniencia**

---

<sup>53</sup> (Miles Huber, 1991)

<sup>54</sup> (OpenSourceControl, s.f.)

Para la selección de la muestra usada para la evaluación de la funcionalidad del dispositivo se eligió el tipo de muestreo no probabilístico denominado muestreo por conveniencia. Este consiste en seleccionar una muestra de la población por el hecho de que sea accesible. Es decir, los individuos empleados en la investigación se seleccionan porque están fácilmente disponibles, no porque hayan sido seleccionados mediante un criterio estadístico.

Esta conveniencia, que se suele traducir en una gran facilidad operativa y en bajos costes de muestreo, tiene como consecuencia la imposibilidad de hacer afirmaciones generales con rigor estadístico sobre la población.<sup>55</sup>

---

<sup>55</sup> (Ochoa, 2015)

## 5 DESARROLLO INGENIERIL

### 5.1 Diseñar el sistema de adquisición de señales percutivas

#### 5.1.1 Determinar transductor de captura

Para determinar el transductor de captura se decidió realizar comparaciones objetivas entre micrófonos comercialmente usados para la batería acústica y sensores piezoeléctricos. A continuación, se describen tres comparaciones: la primera es “voltajes de funcionamiento”, referente a los voltajes generados por los transductores al recibir una señal de excitación; la segunda es “forma de onda”, reflejado en la relación entre amplitud y tiempo de la señal que los transductores generan al recibir una señal de excitación; y la tercera es “costos”, referente a las diferencias en precios entre los transductores.

#### **Voltajes de funcionamiento**

Este aspecto es importante dado que, al saber el voltaje de salida del transductor se puede determinar la manera en la que se transforma la señal analógica en una señal digital. Según los documentos de especificaciones de los micrófonos nombrados en la Tabla # la sensibilidad de estos se encuentra entre 0.6 mV/Pa y 8.9 mV/Pa, mientras que los voltajes de salida de un sensor piezoeléctrico pueden llegar hasta o superar los 20 V.<sup>56</sup> Teniendo esto en cuenta, si se toma el micrófono de mayor sensibilidad y se usa el máximo SPL que puede recibir (según la hoja de especificaciones), en este caso 159 dB SPL, el voltaje generado por el micrófono sería de 15.05 mV. Lo cual no es siquiera la centésima parte del voltaje alcanzado por el sensor piezoeléctrico.

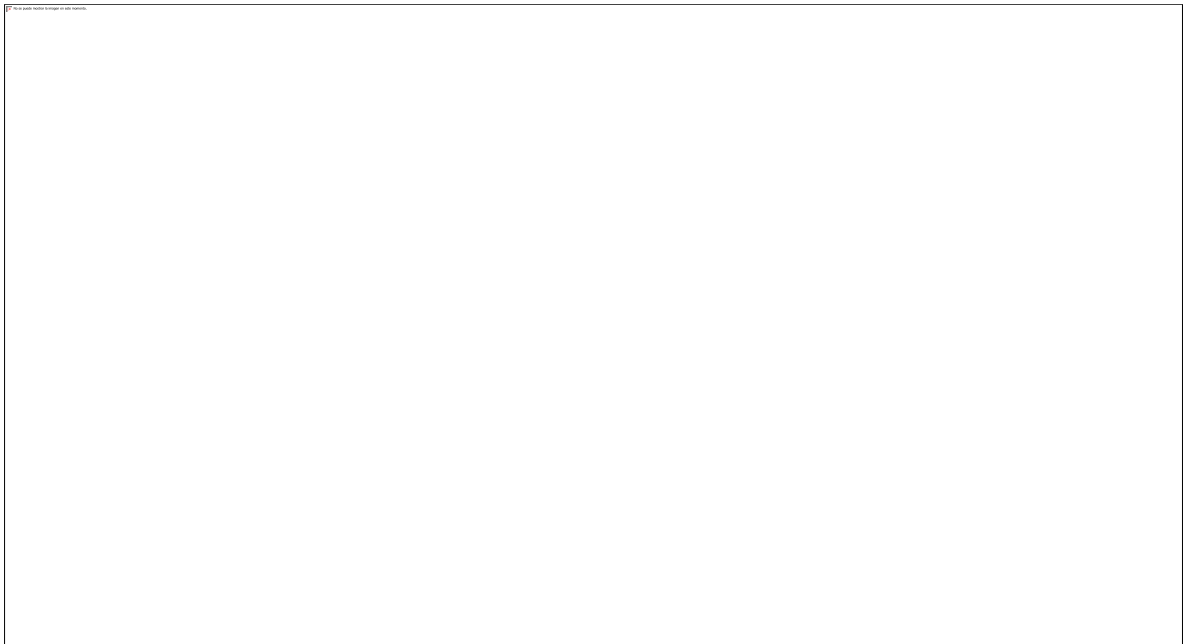
---

<sup>56</sup> (Earnhart, 2012)

## Forma de Onda

Para esta comparación, se realizaron 3 grabaciones en las cuales se registraron golpes en las 3 secciones de la batería escogidas (Hi-Hat, Redoblante y Bombo). En cada sección de la batería se usó un micrófono y un sensor piezoeléctrico, al mismo tiempo. Los micrófonos fueron ubicados según las recomendaciones dadas por Shure en su página oficial de internet y los sensores piezoeléctricos fueron ubicados según las recomendaciones dadas por Roland para la ubicación de triggers, en su página oficial de internet. Las grabaciones fueron registradas en Pro Tools a una frecuencia de muestreo de 44,100 y una profundidad en bits de 24, por medio de la interfaz Focusrite Scarlett 18i8 de primera generación.

A continuación, se pueden observar 3 figuras que comparan las formas de onda capturadas en el Hi-Hat, el Redoblante y el Bombo al recibir una excitación (golpe de baqueta).



*Figura 16. Comparación de la forma de onda de la señal generada por los transductores al recibir una señal de excitación*

Al observar la Figura 16, con las capturas del micrófono en el costado izquierdo de la imagen y las capturas del sensor piezoeléctrico en el costado derecho, visualmente hay semejanza en las formas de onda. Todas las gráficas tienen una duración de dos segundos y su amplitud está normalizada. Además, las señales de excitación (golpes de baqueta) fueron capturadas al mismo tiempo por el micrófono y por el sensor piezoeléctrico para que la comparación sea congruente.

### Costos

Esta última comparación se realiza con el fin de conocer la relación costo beneficio de los transductores. A continuación, se plantea una tabla de comparación que relaciona 3 aspectos de los transductores:

- Marca: fabricante del transductor.
- Uso: parte de la batería en la que generalmente se usa.
- Precio: valor comerciales en pesos colombianos. Para el caso de los micrófonos Shure, los precios fueron tomados de la página oficial de internet en Estados Unidos. Para el caso de los micrófonos Beyerdynamic, los precios fueron tomados de Amazon.com como distribuidor autorizado de la marca. El cambio de dólares a pesos colombianos se hizo con una tasa de cambio de \$2,995.23 COP.

<b>Aspectos de comparación</b>			
<b>Transductor</b>	<b>Marca</b>	<b>Uso</b>	<b>Precio (\$COP)</b>
Sensor Piezoeléctrico	Genérico	Hi-Hat, Redoblante, Bombo	1,000
Micrófono SM137	Shure	Hi-Hat	536,146
Micrófono SM57	Shure	Redoblante	296,528
Micrófono Beta52	Shure	Bombo	566,100
Micrófono M201 TG	Beyerdynamic	Hi-Hat	898,539
Micrófono TG I50d	Beyerdynamic	Redoblante	471,748

Micrófono TG I70d	Beyerdynamic	Bombo	790,590
-------------------	--------------	-------	---------

*Tabla 4. Aspectos de comparación entre transductores (Fuente propia)*

Como se observa en la Tabla 4, el transductor con menor precio es el sensor piezoeléctrico, por una diferencia de mínimo \$295,500. Además, el sensor puede ser usado en las 3 partes de la batería comparadas, lo que lo hace una opción viable debido a la relación costo-beneficio para el proyecto en desarrollo, basados no sólo en el precio sino también en la calidad de la señal generada por el transductor al recibir una señal de excitación, como se mostró en el punto anterior “Formas de Onda”.

Las comparaciones realizadas permiten comprender el funcionamiento de los transductores y, dado que generan señales semejantes al recibir la misma señal de excitación, pero diferentes niveles de voltaje, el sensor piezoeléctrico muestra ser la opción más viable para la captura de las señales de la batería y su uso en un circuito que permite la conversión de señales analógicas en señales digitales, además de ser el transductor más económico. Por esta razón se escoge el sensor piezoeléctrico como el transductor usado para el proyecto.

### **5.1.2 Diseño de circuito de adquisición de señales y conversión análogo-digital**

En esta sección se describen las características de diseño del circuito para la adquisición de las señales de la batería, provenientes del sensor piezoeléctrico. Se requiere de la implementación de un circuito capaz de transformar una señal de voltaje alterno de corta duración, en un pulso de voltaje continuo para ser registrado en la Raspberry Pi. En otras palabras, se requiere de un sistema que convierta señales analógicas en digitales para un mínimo de 3 señales analógicas (Hihat, Redoblante, Bombo).





*Figura 17. Sistema de adquisición de señales*

### **5.1.2.1 Criterios de estudio**

Para esta tarea se plantean dos soluciones que serán evaluadas en cuanto a la facilidad de implementación, calidad de la forma de onda de los pulsos generados y funcionalidad:

1. Comparador
2. Diodo Zener

En cuanto a la facilidad de implementación, se evaluarán la cantidad de componentes y ajustes requeridos para la elaboración de cada método. La funcionalidad indicará el desempeño del método para la aplicación específica y la calidad de forma de onda contrastará las formas de onda de los pulsos generados con un pulso digital ideal, como el que se presenta en la Figura 17.

El pulso digital generado deberá tener un nivel de voltaje entre 2.4V y 3.3V durante su ciclo positivo, puesto que los niveles de voltaje que son operados por la Raspberry Pi 3 se rigen por la tecnología CMOS.

### **5.1.2.2 Planteamiento de circuitos**

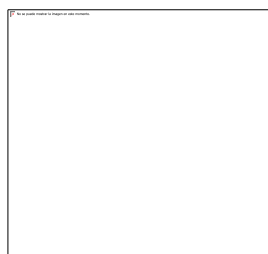
Se seleccionó una referencia de comparador comercial y otra de diodo zener, con las cuales se experimentará para evaluar las características mencionadas anteriormente.

Comparador	LM 339N
Diodo Zener	BZX 55C 3V3

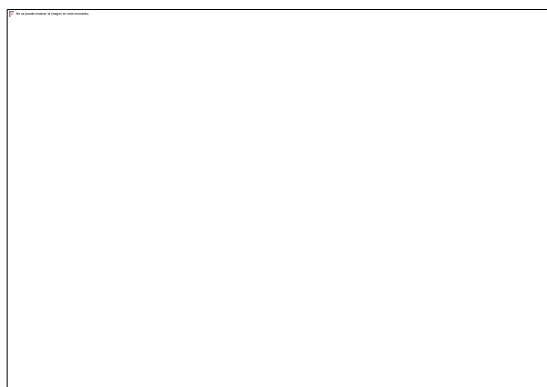
*Tabla 5. Referencias de los comparadores y diodo zener*

### 5.1.2.2.1 Comparador LM339N

El integrado LM339N está compuesto por 4 comparadores de voltaje independientes diseñados para operar a partir de una fuente de alimentación positiva, entre 2V y 36V. A continuación, se presenta la configuración de pines y su diagrama de bloques funcional.



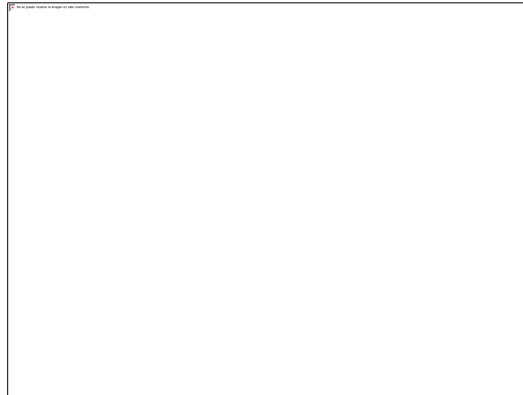
*Figura 18. Configuración de pines LM339N*



*Figura 19. Diagrama de bloques funcional*

En la Figura 19, se observa que el tipo de etapa de salida del comparador LM339N es de colector abierto, lo que permitirá ajustar el voltaje de salida de los pulsos mediante un divisor de voltaje. En cuanto a sus características de velocidad de respuesta, el slew rate,

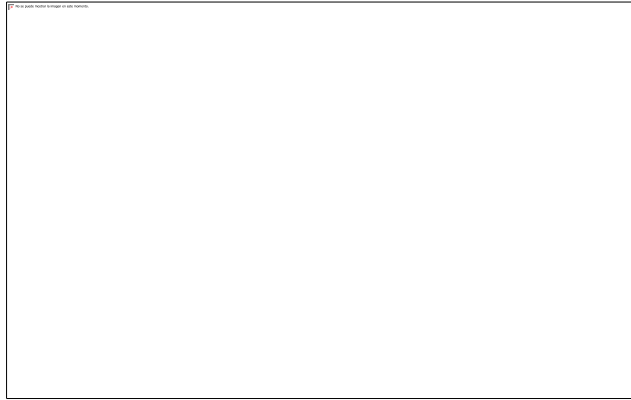
parámetro que determina la razón de cambio del nivel de voltaje de salida con respecto al tiempo, es reemplazado en la ficha técnica de este circuito como tiempo de respuesta, y es descrito por la siguiente figura.



*Figura 20. Tiempo de respuesta vs Voltaje de salida*

En la Figura 20, se observan las diferentes curvas de respuesta con respecto a un valor de overdrive, el cual representa la diferencia de voltajes entre la entrada no inversora (señal) y la inversora (referencia). Según la curva de color rojo, cuando la señal de entrada haya superado la referencia por un nivel de voltaje de 5mV, el pulso de salida será generado aproximadamente 1.25 microsegundos después. Al incrementar la diferencia por la cual la señal de entrada supera a la referencia, los tiempos de respuesta se reducirán.

La siguiente figura presenta el esquema eléctrico del circuito comparador utilizado durante el experimento.

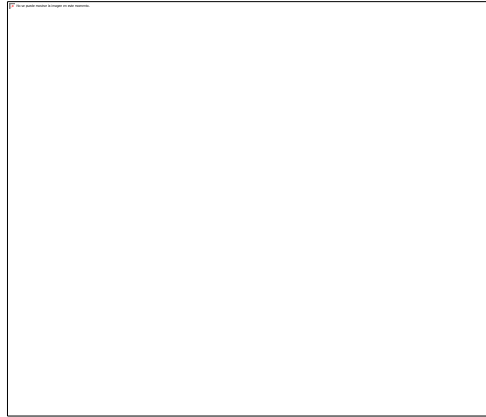


*Figura 21. Diagrama electrónico del LM339N (1 comparador)*

En el circuito de la Figura 21, las resistencias R1 y R2 se encargan de ajustar el voltaje de referencia del comparador, el cual se fijó en 3.3V para ser comparado con el diodo zener que posee el mismo voltaje de regulación. La resistencia R3 es conocida como resistencia de pull-up y es conectada al nivel de voltaje que se desea obtener en el pulso de salida generado por el comparador. Para el experimento se conectó a una fuente de 3.3V, y su valor de 3k Ohm fue recomendado por la hoja de especificaciones del LM339N. La resistencia R4 cierra el circuito, y permite que el voltaje en el pin de entrada del comparador sea 0 cuando no existe una señal de entrada.

#### **5.1.2.2.1 Diodo Zener**

Como una segunda aproximación, se seleccionó un diodo zener de referencia BZX 55C3V3 para ser utilizado como un regulador de voltaje de 3.3V, con el propósito de generar los pulsos cuadrados a partir de la señal alterna de entrada de forma pasiva. Según la hoja de especificaciones, el diodo puede mantener un voltaje de 3.3V con una corriente aproximada de 15mA para una temperatura ambiental de 25 grados centígrados, como se muestra en la Figura 22.



*Figura 22. Voltaje respecto a corriente y temperatura en diodo zener BZX55C3V3*

Es importante destacar que la corriente inversa máxima que puede pasar por este componente es de 115mA antes de su destrucción acorde con las especificaciones del fabricante.

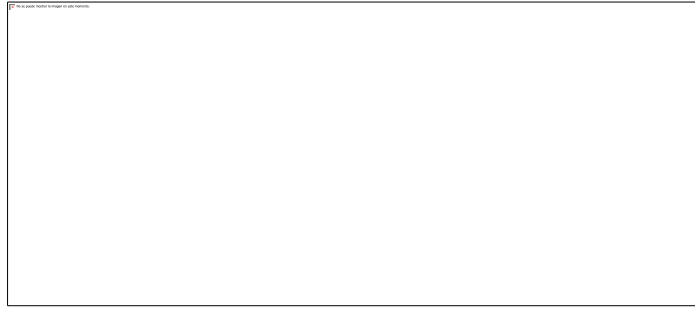
Conociendo el voltaje de regulación y la corriente necesaria, se calcula el valor de la resistencia que regulará la corriente del circuito mediante la ley de ohm.

$$3.3V = 15mA * R$$

$$R = 3.3V/15mA = 220\Omega$$

*Ecuación 3. Cálculo de resistencia para regulación de corriente*

A continuación, se presenta el esquema del circuito zener utilizado durante los experimentos.

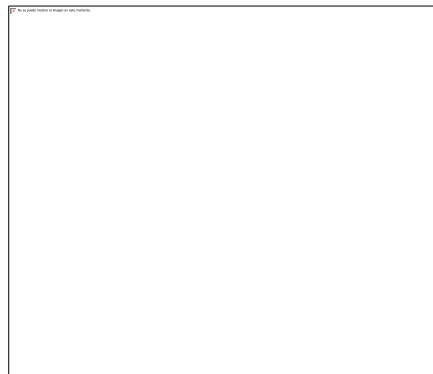


*Figura 23. Diagrama electrónico BZX55C-3V3*

### **5.1.2.3 Experimento**

#### **5.1.2.3.1 Descripción**

Se realizó un experimento para evaluar los circuitos planteados en base a su funcionalidad, facilidad de implementación y forma de onda como se definió al inicio de este capítulo. Para esto, se utilizó un generador de onda como señal de entrada y un osciloscopio como instrumento de medición para observar los pulsos digitales generados, como se muestra en la Figura 24.



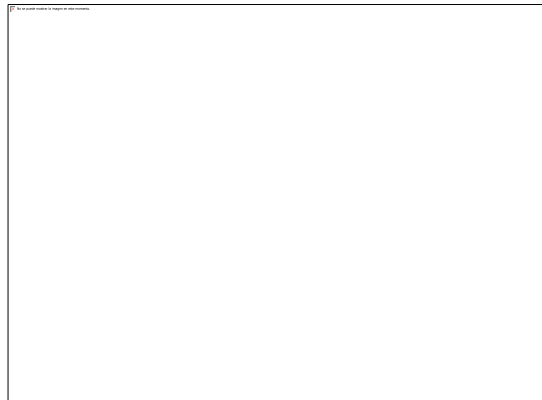
*Figura 24. Diagrama de bloques del experimento*

Se experimentaron los circuitos con 2 formas de onda diferentes para la señal de entrada (Sinusoidal, Diente de Sierra), con una frecuencia de 500Hz y amplitud de 10V pico a pico.

Posteriormente, intentaron capturar los pulsos de salida de cada circuito en la Raspberry Pi para evaluar su funcionalidad en la aplicación requerida.

### 5.1.2.3.2 Resultados del LM339N

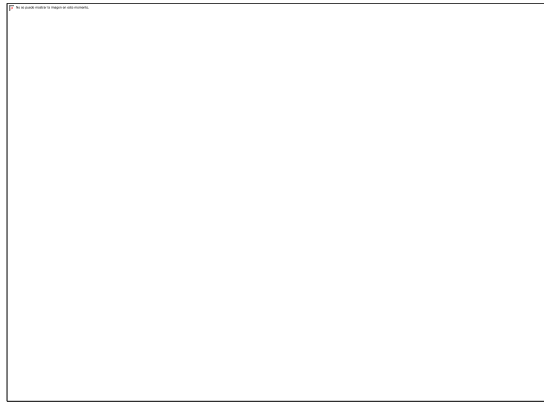
Para el circuito comparador LM 339N se utilizaron dos tipos de forma de onda como señal de entrada, sinusoidal y diente de sierra, con las características de amplitud y frecuencia mencionadas anteriormente. Se midió la entrada del circuito en la Figura 18 en el pin 5 del comparador, y la salida en el pin 2. En las siguientes figuras se presentan os resultados obtenidos.



*Figura 25. Comparador LM339N, señal de entrada: Sinusoidal (amarillo)*

En la Figura 25 se puede observar la señal sinusoidal de entrada (amarillo), junto con el pulso cuadrado generado en la salida (azul). En el osciloscopio, se fijaron dos cursores

horizontales que ayudan a identificar los niveles de voltaje en la gráfica. El cursor “A”, se observa como una línea horizontal segmentada de color blanco a lo largo de la gráfica, indicando el nivel de 5V, mientras que el cursor “B”, de color naranja, indica el nivel de 3.3V. La curva de color azul, correspondiente al pulso digital generado por el comparador, es generada correctamente en el instante que la señal de entrada supera el umbral de 3.3V, indicado en la figura por la línea de color naranja.



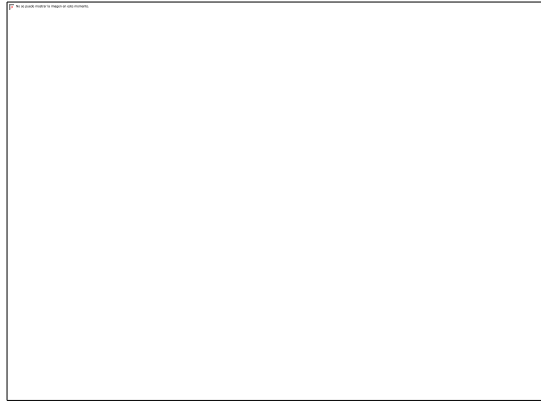
*Figura 26. Comparador LM339N, señal de entrada: Diente de Sierra (amarillo)*

También, en la Figura 26, en donde se utilizó una forma de onda de tipo diente de sierra, se puede observar como el pulso generado por el comparador se mantiene estable, y realiza los cambios de estado en los instantes correctos.

### **5.1.2.3.3 Resultados del Diodo Zener**

En este experimento también se utilizaron las formas de onda de tipo sinusoidal y diente de sierra, con los mismos valores de frecuencia y amplitud que en el experimento del comparador. A continuación, se presentan los resultados obtenidos.





*Figura 27. Diodo Zener 3.3V, señal de entrada: Sinusoidal (amarillo)*

En la Figura 27, se observa como el diodo zener actúa sobre la señal sinusoidal de entrada, restringiendo el nivel de voltaje tras superar los 3.3V como se esperaba. Sin embargo, a diferencia del comparador LM339N, la forma de onda del pulso generado (curva azul), difiere con la de un pulso digital ideal, a pesar de lograr cierta estabilidad en el nivel de voltaje durante la duración del pulso. La forma de onda de entrada, determina la forma de onda del pulso digital, lo cual podría presentar ser una complicación para esta aplicación.

Observando la Figura 28, se encuentra que el zener, es capaz de regular el voltaje de la señal de entrada nuevamente. Por otro lado, la forma de onda del pulso de salida difiere aún más de un pulso ideal, en comparación el pulso obtenido en la Figura 27. Esto, reitera que la forma de onda de los pulsos de salida obtenidos del circuito de diodo zener, dependerán significativamente de la forma de onda de la señal de entrada.

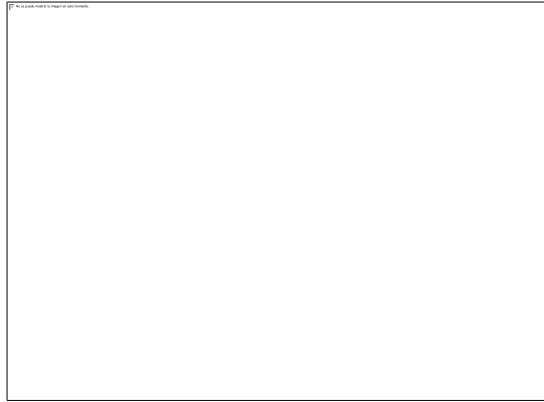


Figura 28. Diodo Zener 3.3V, señal de entrada: Diente de Sierra (amarillo)

#### 5.1.2.3.4 Comparación de LM339N contra Diodo Zener

Comparando los métodos experimentados, se encuentra que cada método posee ciertas ventajas sobre el otro, que podrán definir su escogencia en base a las necesidades de cada aplicación. En cuanto a facilidad de implementación, el circuito del diodo zener requiere de menos componentes y puede ser implementado en espacios más pequeños que el circuito del LM339N. Adicionalmente, al estar compuesto en su totalidad por componentes pasivos, no requiere de un voltaje de polarización para su funcionamiento, lo que le otorga una ventaja en aplicaciones con dichas características.

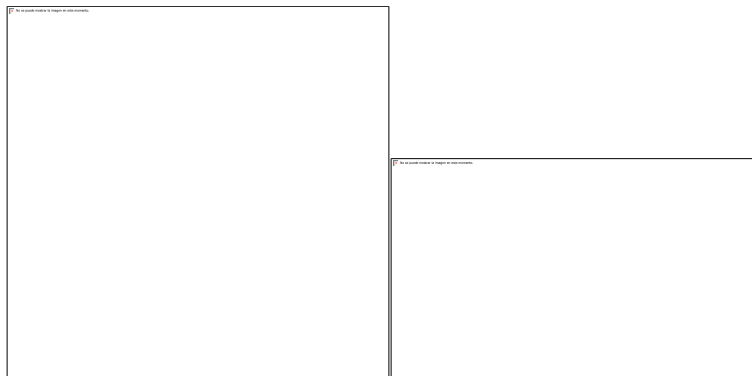
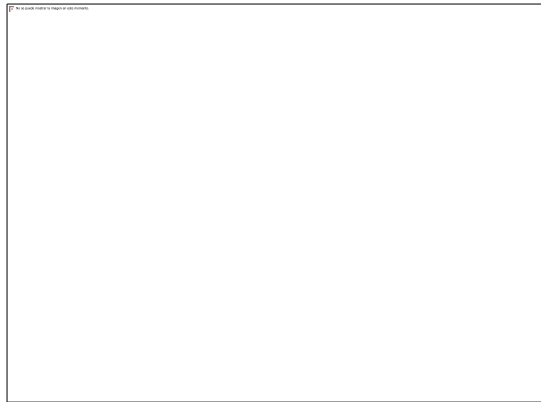


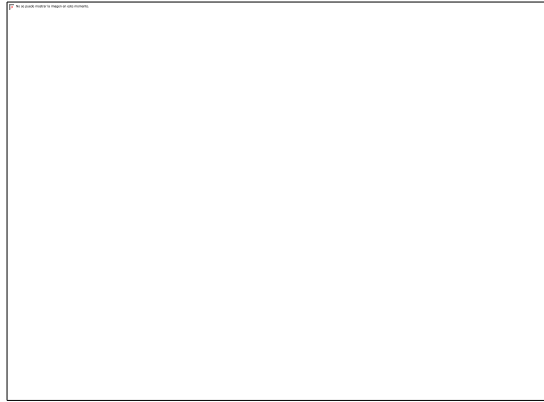
Figura 29. Circuito LM339N (izquierda) y Diodo Zener 3.3V (derecha)

Por otra parte, evaluando la forma de onda de los pulsos de salida, el comparador LM339N logró generar un pulso con mayor estabilidad, independiente de la forma de onda de la señal de entrada. En la Figura 30, se observa como el pulso generado por el circuito comparador, mantiene un voltaje constante a lo largo de la duración del pulso, mientras que el pulso obtenido del diodo zener obedece a las variaciones de la señal de entrada.



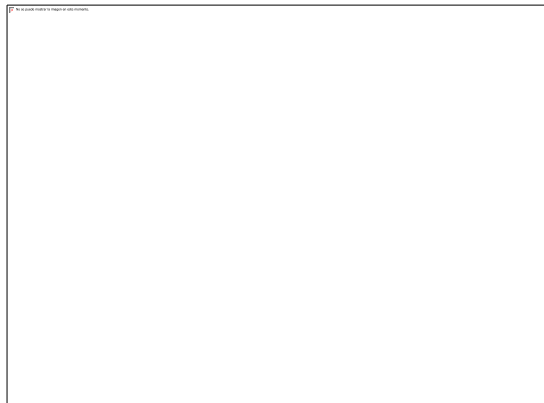
*Figura 30. Pulsos de salida para una señal de entrada Sinusoidal,*

De igual forma, en la Figura 31 se observa como el pulso generado no logra mantener un nivel de voltaje constante en la sección donde la señal de entrada supera los 3.3V. Si se necesitase que el pulso llegara a 3.3V lo antes posible, el circuito del LM339N vencería en tiempo al circuito del zener, y también contaría con una duración del pulso en el voltaje regulado más larga. No obstante, dados los rangos de voltaje permitidos por tecnologías como CMOS y TTL, para la conmutación de los estados, la funcionalidad del circuito del zener no se ve afectada.



*Figura 31. Pulsos de salida para una señal de entrada Diente de Sierra,*

Por último, en la Figura 32 se presenta la comparación entre los pulsos de salida de ambos circuitos para una señal de entrada con tipo de forma de onda cuadrada. En esta gráfica, se puede evidenciar como el comportamiento del circuito zener obedece a la forma de onda de entrada y es capaz de generar un buen pulso digital con una señal de entrada continua.



*Figura 32. Pulsos de salida para una señal de entrada Cuadrada,*

	Facilidad de implementación	Forma de onda de pulso de salida	Funcionalidad
LM339N	<ul style="list-style-type: none"> <li>• Circuito activo, requiere de fuente de alimentación</li> <li>• Cantidad de elementos: 5</li> </ul>	<ul style="list-style-type: none"> <li>• Estable a lo largo del pulso</li> </ul>	<ul style="list-style-type: none"> <li>• Cumple con los requisitos CMOS</li> </ul>
Diodo Zener 3.3V	<ul style="list-style-type: none"> <li>• Circuito pasivo, no requiere de alimentación</li> <li>• Cantidad de elementos: 2</li> </ul>	<ul style="list-style-type: none"> <li>• Inestable, dependiente de la forma de onda de la señal de entrada</li> </ul>	<ul style="list-style-type: none"> <li>• Cumple con los requisitos CMOS</li> </ul>

*Tabla 6. Comparación de circuito LM339N y Diodo Zener 3.3V*

En la Tabla 6, se encuentran las características que fueron evaluadas en cada circuito durante el experimento. Para las aplicaciones del proyecto se seleccionó el circuito comparador dada su estabilidad en los pulsos digitales generados independientemente de la forma de onda de la señal de entrada. Igualmente, los requisitos de polarización del LM339N no presentan ser un problema para el proyecto, puesto que la tarjeta de adquisición de los pulsos digitales, la Raspberry pi 3, cuenta con salidas de voltaje DC que pueden ser utilizadas para alimentar el circuito. Por otro lado, el integrado LM339N, cuenta con 4 comparadores independientes que pueden ser aprovechados para reducir el tamaño del montaje.

## **5.2 Diseñar el algoritmo de detección de tiempo e implementarlo en un dispositivo portátil**

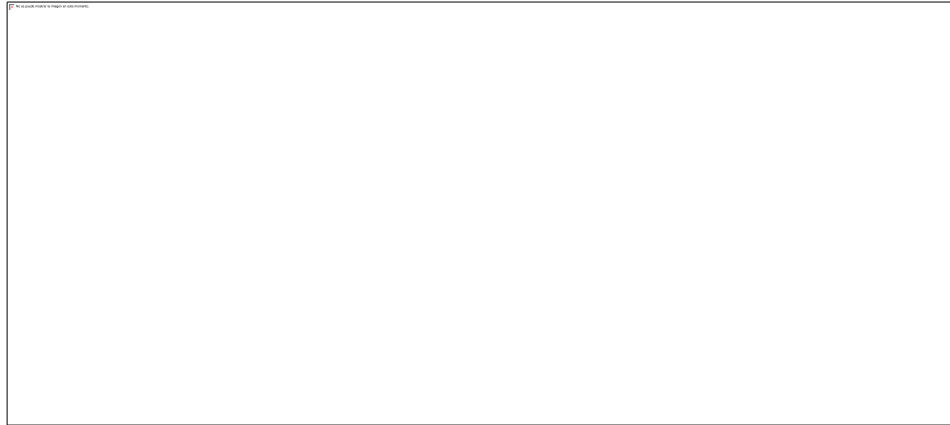
Para la implementación de un algoritmo de detección de tiempo en el circuito de adquisición planteado anteriormente, se requiere de un sistema físico capaz de leer los pulsos digitales, procesar la información rítmica y en lo posible, comunicarse directamente con un DAW para la sincronización. Se propone la utilización de una Raspberry Pi 3 para llevar a

cabo las tareas mencionadas, un computador pequeño que cuenta con pines digitales de entrada y salida que pueden ser manipulados mediante una programación. Para el desarrollo del software se seleccionó Python como lenguaje de programación por ser un lenguaje gratuito, de alto nivel, que cuenta con una amplia variedad de librerías para diferentes aplicaciones, como la que será utilizada para leer los pines GPIO de la tarjeta. Además, es el lenguaje oficial de la Raspberry Pi. Se tuvieron en cuenta los siguientes requerimientos para el desarrollo del software:

## **5.2.1 Requerimientos del Software**

### **5.2.1.1 Captura y almacenamiento de impulsos**

El programa debe tener la capacidad de capturar los impulsos del circuito comparador y almacenar esta información en vectores separados para cada pieza del instrumento. El sistema funcionará como una grabadora digital con una frecuencia de muestreo determinada, y una resolución en bits de 2, lo cual indica, que los datos registrados solo podrán tomar 2 posibles valores, 1 para una detección de un impulso y 0 para una ausencia del mismo. Los datos serán almacenados temporalmente hasta que se haya recolectado suficiente información para obtener una buena estimación del tempo. La cantidad de datos almacenados es definida más adelante.



*Figura 33. Esquema de captura y almacenamiento de impulsos*

### **5.2.1.2 Procesamiento en tiempo real**

Un sistema que funcione en tiempo real no solo depende de su efectividad en la generación de resultados lógicos durante computación, sino también del tiempo que a este le tome producir dichos resultados.<sup>57</sup> Lo anterior, no necesariamente determina que dicho tiempo sea instantáneo, pero si adecuado para las características del entorno. Para el proyecto, se evaluará el tiempo en base a la cantidad de compases musicales requeridos para estimar un valor de tiempo y realizar la sincronización.

### **5.2.1.3 Precisión**

La precisión de detección de tiempo en unidades de BPM está definida dentro de un rango en donde se evalúa si el valor detectado es válido o no. Actualmente, en las publicaciones de algunos de los algoritmos de tempo más reconocidos, el porcentaje de tolerancia de detección de BPM varía entre 4% (Dixon, 2004) y el 20% (Hainsworth, 2003). Los autores Rossiter y Tsang estudian la gran diferencia que existe entre diferentes artículos

---

<sup>57</sup> (Buttazzo, 2011)

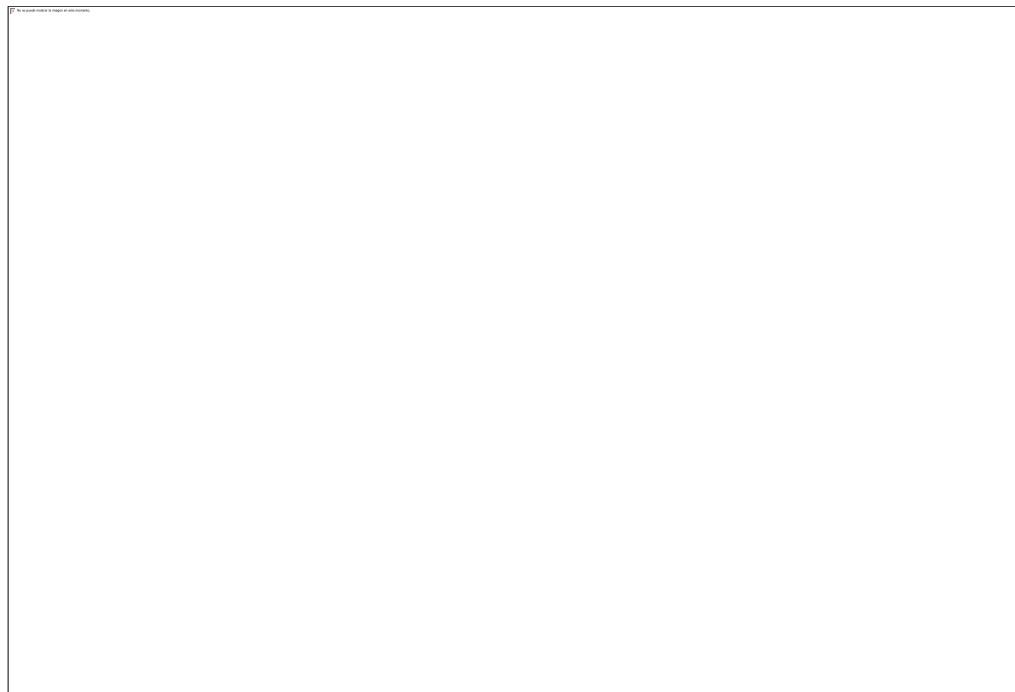
y proponen utilizar una tolerancia de detección del 10%, la cual será aplicada para esta investigación.

#### **5.2.1.4 Sincronización con DAW**

Por último, se requiere de una comunicación entre el dispositivo final y un DAW para la sincronización de pistas. Esta será establecida a través del protocolo MIDI, por ser uno de los más populares encontrados en DAW. Para modificar el tempo de las pistas se estudiarán dos aproximaciones, la primera, a través de mensajes de tipo Control Change, y la segunda mediante una señal de reloj MIDI externa, generada desde la Raspberry Pi.

### **5.2.2 Desarrollo**

#### **5.2.2.1 Diagrama general**



*Figura 34. Diagrama de Actividad UML para el Algoritmo de Detección.*



### 5.2.2.2 Parámetros iniciales (Frecuencia de Muestreo y Tamaño de Grabación)

En primer lugar, se desea detectar un valor de tiempo inicial que determinará algunos de los parámetros iniciales del algoritmo y el tiempo de sincronización inicial. Para esto, se construyó un tambor mudo de goma suave con un piezoeléctrico adherido al parche capaz de detectar los golpes de las baquetas con precisión, el cual será nombrado como “pad” en el desarrollo del proyecto.

Se pretende que el usuario marque en el pad, con 4 golpes de negra, el tiempo inicial en el cual comenzará la secuencia tal y como se hace tradicionalmente para indicar a los músicos el inicio de una interpretación. El pad estará conectado a la Raspberry mediante el circuito comparador y sus golpes serán registrados. Para calcular el tiempo inicial, se medirá la diferencia de tiempo entre cada golpe de marcación y se promediará el resultado. Luego se transforma el promedio de tiempo en segundos a unidades de BPM mediante la Ecuación 4.

$$BPM \text{ Inicial} = \frac{60}{\text{tiempo promediado entre golpes}}$$

*Ecuación 4. Cálculo de BPM inicial*

Entre los primeros parámetros a definir se encuentran la frecuencia de muestreo, y la duración de cada buffer de grabación. Estos se encuentran directamente relacionados con el rango de tempos que el programa pueda detectar. El tamaño del buffer de grabación, define el mínimo valor de tiempo que el programa podrá detectar. La fórmula para conocer el mínimo tiempo que podrá ser detectado, dado un tamaño del buffer de grabación, se muestra a continuación:

$$\text{Mínimo Tiempo a Detectar (BPM)} = \frac{60}{\text{Tamaño de Buffer en Segundos}}$$

*Ecuación 5. Mínimo Tiempo de detección del programa.*

De modo que, para un tamaño de buffer de 1 segundo, el programa podrá detectar un tiempo mínimo de 60BPM. Por otro lado, la frecuencia de muestreo determinará el máximo tiempo que podrá ser detectado, acorde con la siguiente formula:

$$\text{Máximo Tiempo a Detectar (BPM)} = \frac{60}{\frac{1}{\text{Frecuencia de Muestreo}}}$$

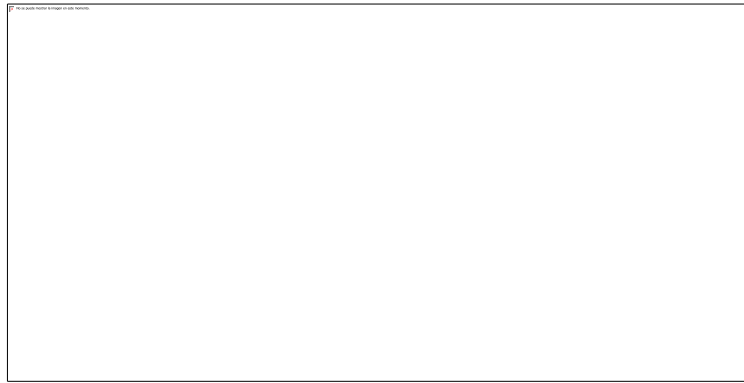
*Ecuación 6. Máximo Tiempo de detección del programa.*

Por tanto, para una frecuencia de 50Hz, el programa podrá detectar un valor de tiempo máximo de 3000BPM. Estos parámetros de diseño también afectaran la cantidad de procesos que deberá ejecutar el programa y el tiempo que este se demorará en entregar resultados.

### **5.2.2.3 Grabación**

Para registrar los impulsos, se programaron 3 ciclos infinitos que constantemente leen el nivel de entrada de los pines de la Raspberry Pi. Estos ciclos son ejecutados de manera simultánea, mediante un tipo de programación “Multithreading” o múltiples hilos que, incluso, permitirá añadir tiempos de retraso en la lectura de cada transductor, para obtener un pulso limpio sin efectos de rebote. Los impulsos obtenidos del circuito comparador son registrados dentro de una matriz de 3 filas y 1 columna, donde en cada fila se encuentra la información de cada transductor por separado. Cuando el sistema detecte un impulso, ingresará un número 1 en la fila correspondiente, de lo contrario, ingresará un número 0. Cada buffer de grabación contendrá una matriz 3x1 con la información de los pulsos

detectados durante la interpretación del baterista. Al completar todos los datos del buffer se realizará el proceso de detección de tempo mediante el método de autocorrelación y el buffer será desocupado y preparado para registrar un nuevo paquete de datos. La siguiente imagen demuestra gráficamente el proceso de grabación.



*Figura 35. Sistema de grabación con ventanas ( Fuente propia)*

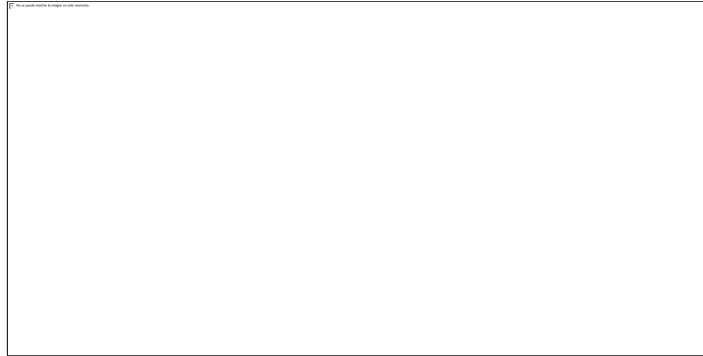
La región de color verde representa la señal rítmica que será capturada por el sistema cada determinado tiempo. Se puede observar como las ventanas de grabación se solapan en el proceso y se realiza un nuevo cálculo de tempo en cada finalización de estas, manteniendo así, una relación entre los bloques de datos adyacentes e incrementando la frecuencia con la que el programa generará nuevos resultados.

#### **5.2.2.4 Procesamiento (Detección del tempo)**

##### **5.2.2.4.1 Cálculo de Coeficientes de Autocorrelación**

Para encontrar la periodicidad en la información capturada mediante el método de autocorrelación propuesto por el investigador Müller, primero se toma el

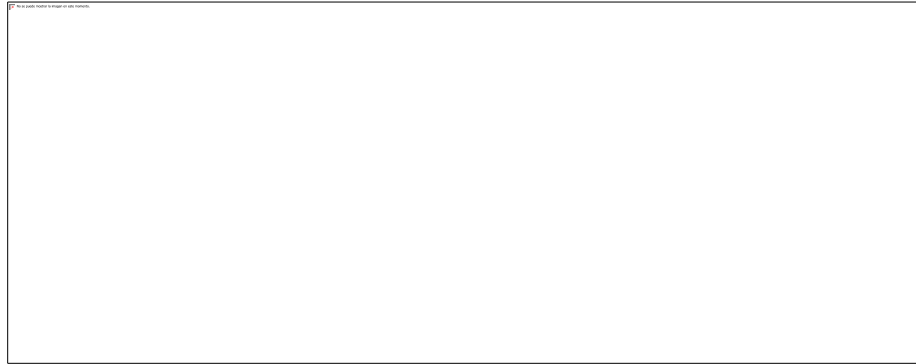
buffer de datos y se divide en 3 secciones de una misma longitud como se muestra en la figura 3.



*Figura 36. División del buffer (Fuente propia)*

Luego, se define una ventana de datos fija y otra móvil, con las cuales se realizarán las correlaciones. Para la ventana de datos fija, se toman los datos de la sección central (sección 2) como aparece en la Figura 36, en la región sombreada de color rojo. La ventana de datos móvil se define inicialmente como la misma región que la ventana fija, a diferencia que esta se desplazará a lo largo del buffer de datos. Ahora, se calcula el coeficiente de correlación entre la ventana de datos fija y la ventana de datos móvil fila por fila. Por ejemplo, los datos del bombo de la ventana fija serán correlacionados con los datos de bombo de la ventana móvil. En esta primera correlación se alcanzarán los valores máximos, puesto que ambas ventanas contienen la misma información.

Posteriormente se desplaza la ventana móvil una muestra hacia la izquierda y se almacenan los datos que se encuentren dentro de ella.



*Figura 37. Desplazamiento de ventana móvil (Región Azul)*

Nuevamente se calcula el coeficiente de correlación entre la ventana fija y los datos que se encuentren dentro de la ventana móvil. Se almacena el coeficiente de correlación y el número de muestras que se desplazó la ventana móvil. El mismo procedimiento es repetido hasta que la ventana móvil se haya desplazado a lo largo del buffer de datos, como aparece en la Figura 38. Como resultado, se obtendrán tres matrices (1 por cada pieza de batería: hi-hat, redoblante y bombo) de dimensión  $2 \times 1$ , donde la primera fila contendrá los coeficientes de correlación encontrados y la otra los desplazamientos en muestras de cada cálculo. Conociendo la frecuencia de muestreo, se pueden convertir los desplazamientos en términos de muestras a unidades de tiempo, al multiplicar el número de muestras por el inverso de la frecuencia de muestreo, como se muestra en la Ecuación 7.

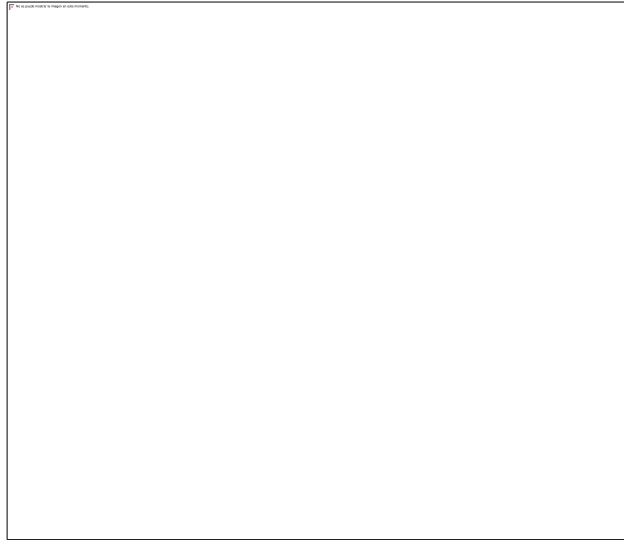


Figura 38. Desplazamiento completo de la ventana móvil

$$\text{Desplazamiento en Tiempo (segundos)} = \text{Desplazamiento en Muestras} * \frac{1}{F. \text{Muestreo}}$$

Ecuación 7. Desplazamiento

#### 5.2.2.4.2 Filtrado de datos

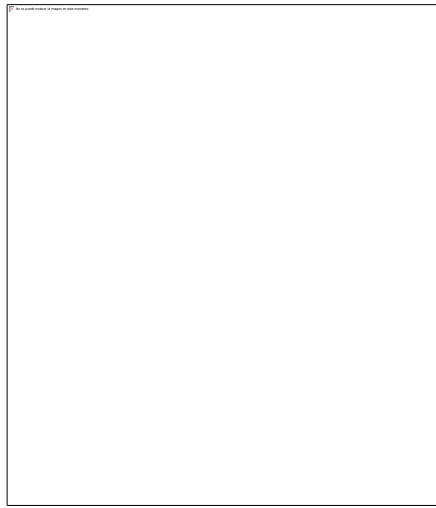
A continuación, se filtran las parejas de datos (Coeficiente, Desplazamiento) para extraer los valores de desplazamiento que obtuvieron los coeficientes más altos de autocorrelación. Para esto se crea una nueva matriz que evaluará cada uno de los coeficientes de autocorrelación obtenidos y desechará aquellos datos que se encuentren por debajo de un umbral determinado. Luego, los datos de desplazamiento en tiempo son transformados a valores de tiempo mediante la siguiente ecuación.

$$BPM = \frac{60}{\text{Desplazamiento en tiempo (segundos)}}$$

Ecuación 8. Cálculo de valores de tiempo

Finalmente, se obtiene una matriz de datos 2x1 por cada transductor, con valores de tiempo en unidades de BPM en una fila y coeficientes de autocorrelación en otra. Ahora, se clasifican las parejas de datos según su valor de tiempo en dos categorías, valores de tiempo fundamentales y valores de tiempo armónicos.

Esta clasificación se da utilizando el valor de tiempo inicial que es dado por el usuario, y es aplicada en base al diagrama de flujo en la Figura 39.



*Figura 39. Diagrama de flujo para clasificación de valores de BPM*

Una vez clasificados los valores de tiempo detectados en cada transductor, es posible emitir un valor de tiempo detectado de acuerdo con la lista de valores de tiempo fundamentales obtenidos, donde pueden ocurrir cualquiera de los siguientes 3 escenarios.

- a) Escenario 1: No se encontraron valores de tiempo fundamental

En el caso de no encontrar ningún valor de tiempo fundamental el programa no retornará ningún resultado y esperará a la próxima captura de datos.

b) Escenario 2: Se encontró un único valor de tiempo fundamental

Al encontrar un único valor de tiempo fundamental, el programa emitirá dicho valor como el tiempo detectado en la captura.

c) Escenario 3: Se encontraron 2 o más valores de tiempo fundamental

Este escenario es el más común de los 3, en donde el programa detecta múltiples valores de tiempo fundamental y debe realizar un procedimiento de comparación entre los valores de tiempo fundamentales y armónicos, el cual se explica a continuación.

#### 5.2.2.4.3 Comparación de tiempos Fundamentales y Armónicos

Se requiere de un método que relacione los valores fundamentales y armónicos con la finalidad de generar un único dato de tiempo. En el contexto de la investigación de algoritmos de estimación de tiempo, se conoce que estos pueden detectar valores como la mitad, el doble, el triple y hasta un tercio del valor de tiempo verdadero. Sin embargo, también es común encontrar que incluso músicos profesionales perciben el tiempo de manera diferente entre ellos.<sup>58</sup>

Estos valores, múltiplos y submúltiplos del tiempo original, serán descritos durante la investigación como valores de tiempo armónicos, haciendo una analogía con la frecuencia. La comparación se basa en conocer qué tan enteros son los múltiplos y submúltiplos de los armónicos con cada uno de los valores fundamentales. Para esto, se toman cada uno de los valores de tiempo fundamental

---

<sup>58</sup> (Rossiter, Tsang, & So, 2007)



encontrados, se comparan con cada armónico, y se les da una puntuación en caso de encontrar que ambos valores son múltiplos o submúltiplos enteros.

Se utiliza la Ecuación 9 para comparar un valor subarmónico y la Ecuación 10 para valores armónicos.

$$S = \left| \frac{Fundamental}{Armónico} - round\left(\frac{Fundamental}{Armónico}\right) \right|$$

*Ecuación 9. Comparación de submúltiplos*

$$A = \left| \frac{Armónico}{Fundamental} - round\left(\frac{Armónico}{Fundamental}\right) \right|$$

*Ecuación 10. Comparación de múltiplos*

Los resultados de las ecuaciones son filtrados bajo un umbral definido, que determina la tolerancia para calificar los múltiplos y submúltiplos como enteros. En las siguientes tablas se encuentra un ejemplo del procedimiento de comparación.

		Fundamental					Fundamental				
		Tempo	90	100	110			Tempo	90	100	110
Armónicos	52	0.27	0.08	0.12	Subarmónicos	200	0.22	0	0.18		
	48	0.13	0.08	0.29		225	0.5	0.25	0.05		
	45	0	0.22	0.44		160	0.22	0.4	0.45		
	51	0.24	0.04	0.16		195	0.17	0.05	0.23		
	33	0.27	0.03	0.33		183	0.03	0.17	0.34		

*Tabla 7. Comparación de tempo fundamental con armónicos y subarmónicos*

En la Tabla 7 se presentan una serie de valores de tempo fundamental (color rojo) definidos arbitrariamente en base a un tempo real de 100 BPM junto con sus armónicos y subarmónicos (color azul). En la tabla se encuentran los resultados de

las comparaciones, calculados con la Ecuación 9 para los subarmónicos y la Ecuación 10 para los armónicos. Ahora, se define un umbral para determinar la puntuación que obtendrá cada valor de tempo fundamental, para el ejemplo será de 0.05. En la siguiente tabla se muestran las puntuaciones obtenidas por cada valor de tempo fundamental.

		Fundamental		
		Tempo	90	100
Armónicos	52	0	0	0
	48	0	0	0
	45	1	0	0
	51	0	1	0
	33	0	1	0
		1	2	0

		Fundamental		
		Tempo	90	100
Subarmónicos	200	0	1	0
	225	0	0	1
	160	0	0	0
	195	0	1	0
	183	1	0	0
		1	2	1

Tabla 8. Puntuación a valores de tempo fundamental

Los datos en la Tabla 7 que se encuentren por debajo del umbral de 0.05, sumaran 1 punto a la valoración del tempo fundamental correspondiente como aparece en la Tabla 8. El valor de tempo detectado que será emitido será el tempo fundamental que obtenga el puntaje más alto. En caso de obtener dos tempos fundamentales con la puntuación más alta, el tempo detectado que será emitido corresponderá al promedio de los fundamentales.

### 5.2.3 Evaluación para la Sincronización

Por último, el programa deberá comunicarse con el reloj MIDI que se encargará de controlar el tempo de reproducción de las pistas. El desarrollo del reloj MIDI será explicado más adelante. Con el propósito de asegurar que las transiciones entre los cambios de tempo ocurran de manera fluida, los tempos detectados serán evaluados previamente para determinar los cambios de tempo adecuados para la ejecución del baterista. El programa enviará un valor de tempo de sincronización diferente luego de clasificar el tempo detectado en alguna de las siguientes 3 regiones.

- a) Región A: La diferencia entre el tempo detectado y el tempo actual es muy pequeña, y puede deberse a las imperfecciones humanas al interpretar un ritmo. Cuando el tempo detectado se encuentra en esta región se mantendrá el tempo actual para la sincronización.
- b) Región B: La diferencia entre el tempo detectado y el tempo actual es lo suficientemente amplia para determinar que el baterista intenta acelerar o desacelerar el ritmo, respectivamente. En este caso, el tempo de sincronización será modificado al tempo detectado.
- c) Región C: La diferencia entre el tempo detectado y el tempo actual es grande, y un cambio de tempo de sincronización brusco puede desorientar al músico. En esta región, el programa modifica la velocidad del reloj MIDI por un valor constante.

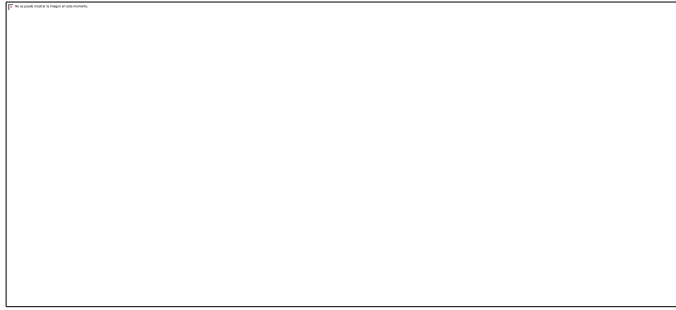


Figura 40. Regiones A, B y C para definir tiempos de sincronización

En la Figura 40, se muestra la curva que representa los cambios del tiempo de sincronización de acuerdo con la región del tiempo detectado.

Se definió que, para asegurar la máxima velocidad de detección, es preferible calcular el tamaño de buffer al inicio de cada interpretación, basándose en el valor de tiempo inicial producto del conteo en el pad.

Se propone el siguiente procedimiento para el cálculo del tamaño del buffer de grabación, teniendo en cuenta cómo este puede afectar el rango de tempos que pueden ser detectados. Tomando el tiempo inicial como valor central del rango de detección, se definen los límites superior e inferior a un 20% de este, como se muestra en el siguiente ejemplo, tomando 100 BPM como tiempo central del rango. Esto es realizado para reducir el tamaño del buffer al mínimo requerido y así optimizar el tiempo de procesamiento de datos. La zona de color azul representa el rango de tempos que podrán ser detectados.



Figura 41. Rango de detección de BPM (Fuente propia)

Al conocer los límites superior e inferior, se plantea que, para una detección eficiente de todos los valores dentro del rango establecido, debe poder capturarse por lo menos un sub-

armónico del mínimo valor de tiempo del rango de detección. Dado que el tamaño del buffer afecta el tiempo mínimo que puede detectarse, este es calculado para obtener el primer sub-armónico del tiempo mínimo dentro del rango establecido, para el ejemplo 80 BPM. El tamaño del buffer de grabación se regirá por la siguiente ecuación:

$$\text{Tamaño de Buffer (Segundos)} = \frac{60}{\frac{\text{Limite Inferior de Detección}}{2}}$$

*Ecuación 11. Tamaño de Buffer de Grabación*

Tomando como ejemplo la Figura 41, con un tempo inicial de 100 BPM y límite inferior de detección de 80 BPM, el cálculo de la Ecuación 11 retornaría un tamaño de buffer de 1.5 segundos. Recordando que el sistema de grabación está compuesto por 3 secciones, se obtiene un tiempo total de 4.5 segundos de grabación para un tempo inicial de 100 BPM.

Los autores Gouyon y Dixon<sup>59</sup> proponen sugieren utilizar una frecuencia de muestreo entre 100Hz y 200Hz cuando la estimación del tiempo es realizada a partir de información simbólica.

## **Pruebas del Sistema**

### **Tiempo Real**

Para evaluar la característica de tiempo real en el algoritmo, se midieron los intervalos de tiempo en los procesos de grabación y detección de tiempo durante su funcionamiento, modificando factores influyentes como el tiempo de detección, la frecuencia de muestreo y el rango de detección. Los tiempos medidos fueron dividido en dos categorías.

---

<sup>59</sup> (Gouyon & Dixon, 2005)

- **Tiempo de Grabación:** Este representa el intervalo de tiempo que le toma al algoritmo en registrar todos los datos de un buffer de grabación.
- **Tiempo de Procesamiento:** Este intervalo representa la cantidad de tiempo necesaria para realizar los procesos de estimación de tiempo, como la autocorrelación, la comparación y el filtrado de datos.

En este experimento se estudiaron 3 variables que influyen directamente en los tiempos de grabación y procesamiento del sistema como lo son:

- **Frecuencia de Muestreo:** Este parámetro determinará la cantidad de datos que serán registrados en un periodo de tiempo determinado. Una frecuencia de muestreo mayor representará un aumento en la cantidad de datos en el buffer y un mayor número de operaciones que deberán ser realizadas en la etapa de procesamiento. Se estudiarán las frecuencias 100Hz y 200Hz.
- **Rango de Detección de Tempo:** El rango de detección de tempo define directamente el tamaño de buffer requerido para la correcta detección del tempo, lo cual incrementará el tiempo de grabación. Los rangos evaluados serán 30%, 20% y 10% del tempo inicial.
- **Tempo de Detección Inicial:** El tempo de detección inicial se encuentra directamente relacionado con el tiempo de grabación, puesto que definirá el tamaño del buffer necesario para la detección del rango establecido. A mayor tempo, pueden utilizarse intervalos de grabación más pequeños, dado que la

distancia entre cada golpe es menor. Se evaluarán los valores de 60 BPM y 200 BPM como los casos extremos en los cuales será utilizado el algoritmo.

En las siguientes tablas se presentan los resultados obtenidos para los intervalos de tiempo de grabación y procesamiento promediados por 10 muestras. La primera columna contendrá las frecuencias de muestreo ( $F_s$ ) estudiadas mientras que la primera fila presentará los rangos de detección.

T. Grabación 60 BPM	RD = 30%	RD = 20%	RD = 10%
$F_s = 100\text{Hz}$	9.47 segundos	7.97 segundos	7.12 segundos
$F_s = 200\text{Hz}$	9.99 segundos	8.03 segundos	7.32 segundos

Tabla 9. Tiempo de Grabación para un tiempo inicial de 60BPM

T. Grabación 200 BPM	RD = 30%	RD = 20%	RD = 10%
$F_s = 100\text{Hz}$	3.08 segundos	2.75 segundos	2.45 segundos
$F_s = 200\text{Hz}$	3.26 segundos	2.95 segundos	2.59 segundos

Tabla 10. Tiempo de grabación para un tiempo inicial de 200 BPM

En las Tabla 9 y Tabla 10 se presentan los resultados correspondientes a los tiempos de grabación para 60 BPM y 200 BPM. En cada tabla se observa como los cambios en la frecuencia de muestreo no afectan el tiempo de grabación significativamente puesto que este solo puede ser influenciado por el tiempo inicial de detección como se mencionó anteriormente.

T. Procesamiento 60 BPM	RD = 30%	RD = 20%	RD = 10%
$F_s = 100\text{Hz}$	1.49 segundos	1.07 segundos	0.87 segundos
$F_s = 200\text{Hz}$	4.70 segundos	3.38 segundos	2.60 segundos

Tabla 11. Tiempo de grabación para un tiempo inicial de 200 BPM

T. Procesamiento 200 BPM	RD = 30%	RD = 20%	RD = 10%
$F_s = 100\text{Hz}$	0.28 segundos	0.22 segundos	0.18 segundos
$F_s = 200\text{Hz}$	0.69 segundos	0.53 segundos	0.44 segundos

*Tabla 12. Tiempo de grabación para un tempo inicial de 200 BPM*

En las Tabla 11 y Tabla 12, se presentan los tiempos de procesamiento del sistema en los diferentes escenarios. Al comparar los tiempos de procesamiento con los de grabación, es claro que la mayor parte del tiempo del proceso de detección el sistema se encuentra grabando la información necesaria para una correcta estimación.

También, se encuentra que, para una frecuencia de muestreo de 200 Hz, los tiempos de procesamiento aumentan considerablemente y pueden llegar a perjudicar la dinámica de detección y sincronización del sistema.

Para comprender los intervalos de tiempo detectados en un contexto musical, los resultados son transformados de segundos a compases musicales en las Tabla 13 y Tabla 14.

T. Total 60 BPM	RD = 30%	RD = 20%	RD = 10%
Fs =100Hz	2.74 compases	2.26 compases	2.00 compases
Fs =200Hz	3.67 compases	2.85 compases	2.48 compases

*Tabla 13. Tiempo total (grabación + procesamiento) para un tempo inicial de 60*

**BPM**

T. Total 200 BPM	RD = 30%	RD = 20%	RD = 10%
Fs =100Hz	2.80 compases	2.48 compases	2.19 compases
Fs =200Hz	3.30 compases	2.89 compases	2.52 compases

*Tabla 14. Tiempo total (grabación + procesamiento) de grabación para un tempoinicial de 200 BPM*

En cuanto a la cantidad de compases que el sistema requiere para estimar y sincronizar un valor de tiempo, se encuentra en el rango de 60BPM hasta 200 BPM se mantiene un aproximado de 2.3 compases para un rango de detección del 20%. Dado que el algoritmo funciona con dos buffers de grabación paralelos, el sistema es capaz de generar un resultado en la mitad del tiempo medido en el experimento.



El factor más influyente en los intervalos de tiempo medidos es el rango de detección de tiempo, el cual logra disminuir hasta 0.7 compases reduciendo el rango de un 30% a un 10%. La relevancia de estos resultados para el usuario será evaluada en el objetivo 3, junto con las pruebas subjetivas realizadas.

### Pruebas de Precisión

Previo a la evaluación del sistema con los sujetos de prueba, se realizaron ensayos para verificar la precisión de detección de tiempo del dispositivo. Con la ayuda de un metrónomo, se interpretaron los ritmos de los géneros del experimento anterior y se registraron 30 capturas de valores de tiempo. Posteriormente, se promediaron los valores de tiempo obtenidos y se calculó el porcentaje de error de detección mediante la siguiente ecuación.

$$\% \text{ Error} = \frac{|BPM \text{ de Metrónomo} - BPM \text{ Detectado}|}{BPM \text{ de Metrónomo}} * 100$$

*Ecuación 12. Cálculo de porcentaje de error*

En las siguientes tablas se presentan los valores de tiempo detectados promediados y los porcentajes de error obtenidos para cada género.

BPM Detectado	Rock	Pop	Balada	Funk	Reggae
80 BPM	81.04	80.86	80.92	78.19	79.87
120 BPM	121.26	121.41	120.79	113.67	N/A

*Tabla 15. Valores de tiempo detectado promedio en diferentes géneros*

% de Error	Rock	Pop	Balada	Funk	Reggae
80 BPM	1.39	1.17	1.24	6.02	2.09
120 BPM	1.90	1.82	3.22	7.55	N/A

*Tabla 16. Porcentaje de error promedio en diferentes géneros*

En la Tabla 16 se encuentra el porcentaje de error de detección promedio calculado para cada uno de los géneros propuestos. En los ritmos de Rock, Pop y Balada el sistema logró los mejores resultados de precisión, obteniendo un porcentaje de error aproximado de 1.24% para un tempo de 80 BPM y de 2.31% para un tempo de 120 BPM. Para ritmos más complejos, como el funk y el reggae, donde el hi-hat es interpretado en una figura musical de menor duración, el porcentaje de error fue mayor.

Esto, puede deberse a que el método de autocorrelación expresa los sub-múltiplos enteros del tempo detectado y suprime los múltiplos enteros. Por tanto, al identificar un valor de tempo elevado como sucede en el hi-hat, la cantidad de sub-múltiplos enteros detectados es mucho mayor y pueden entorpecer la estimación del tempo.

En la siguiente tabla se muestran los sub-múltiplos enteros que son detectados para diferentes figuras musicales en el hi-hat en un tempo de 80 BPM.

Figura	1er M.	2do M.	3er M.	4to M.	5to M.	6to M.	7mo M.	8vo M.
Negra	80 BPM	40 BPM	-	-	-	-	-	-
Corchea	160 BPM	80 BPM	53 BPM	40 BPM	-	-	-	-
Semi-Corchea	320 BPM	160 BPM	106 BPM	80 BPM	64 BPM	53 BPM	45 BPM	40 BPM

*Tabla 17. Múltiplos enteros encontrados para diferentes figuras musicales para 80 BPM*

En un escenario donde se tiene un tempo inicial de 80 BPM y un rango de detección de 20%, el sistema puede detectar cambios de tempo en el rango de 64 BPM hasta 96 BPM. Si en el hi-hat se interpretase un patrón rítmico correspondiente a una figura musical de semicorchea, el algoritmo podría detectar hasta 8 armónicos de los cuales 2 se categorizan dentro del rango de detección (80 BPM y 64 BPM), lo cual perjudicaría la precisión del algoritmo.

Para solventar esta falla, se puede reducir el rango de detección de tempo a un 10%, incrementando la precisión del algoritmo a cambio de una reducción en el rango de movilidad del tempo.

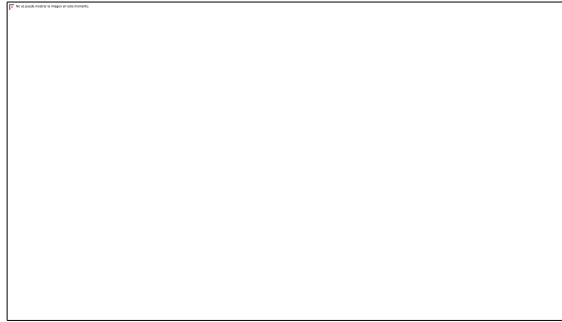
## **Sincronización**

Para la sincronización de tempo con un DAW, se examinaron dos aproximaciones que permitiesen establecer una comunicación estable y estandarizada entre el dispositivo portátil y el software de secuencias. La sincronización será evaluada en algunos de los DAW más comunes para la implementación de secuencias.

La primera aproximación, consiste en la utilización del protocolo de comunicación OSC. Esta propuesta tiene como ventaja, la eliminación de medios físicos para la transmisión de los mensajes, dado que estos son enviados a través de una red inalámbrica, con gran precisión.

Por otra parte, se requiere que el DAW cuente con este protocolo de comunicación y la capacidad de configurar el parámetro de tempo para ser controlado por estos mensajes. De lo contrario, se implementará un programa por fuera del DAW que interprete los mensajes OSC y se comunique con el software de secuencias mediante MIDI, al ser un protocolo más estandarizado. El programa intérprete será desarrollado en el software Pure Data.

Para la transmisión de los mensajes OSC de forma inalámbrica se requiere de una red Wi-Fi establecida, a la cual deberán conectarse tanto el dispositivo portátil como la maquina en donde se encuentre DAW. A continuación, se presenta un esquema de hardware y software requerido para la implementación del sistema explicado anteriormente.



*Figura 42. Esquema de sistema OSC (Fuente propia)*

La segunda aproximación, se basa en la implementación de un reloj MIDI externo que controlará directamente la velocidad de reproducción del DAW, modificando la frecuencia de los pulsos desde el dispositivo portátil. Esta comunicación requiere de una mayor velocidad de transmisión de datos para su funcionamiento, por tanto, no puede ser empleada a través de medios inalámbricos como el protocolo OSC.

Esta propuesta, simplifica las configuraciones que deben realizarse en el DAW para comunicar los dispositivos, en vista de ser un sistema estandarizado, utilizado globalmente para la sincronización de múltiples dispositivos en el mercado.

Adicionalmente, empleando el reloj MIDI, pueden conocerse detalles adicionales con respecto a la reproducción de las secuencias, como los beats y el número de barra en el que se encuentre la interpretación en determinado momento. Esta información puede ser utilizada para un futuro desarrollo del dispositivo, en la sincronización de efectos de audio y/o sistemas audiovisuales.

Para controlar el DAW desde el dispositivo portátil, se requiere de una interfaz MIDI capaz de recibir los mensajes externos y direccionarlos hacia software de secuencias.

A continuación, se presentará un esquema indicando los requerimientos de hardware y software para la implementación de este sistema.

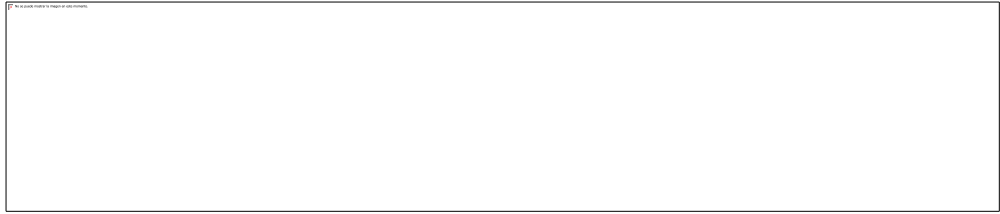


Figura 43. Esquema Midi Clock (Fuente propia)

Al trabajar directamente con mensajes MIDI, es descartada la necesidad de un programa adicional que funcione como interprete entre los dispositivos. A continuación, se presenta una lista con diferentes DAW junto con sus protocolos de comunicación incluidos.

DAW	Protocolo OSC	Protocolo MIDI
Live	✗	✓
Reason	✗	✓
Logic Pro X	✗	✓
FL Studio	✗	✓
Cubase	✗	✓
Pro Tools	✗	✓
Reaper	✓	✓

Tabla 18. Protocolos de Comunicación en DAW (Fuente propia)

Ambas aproximaciones fueron desarrolladas, con el fin de encontrar aquella cuyas propiedades cumplan con las necesidades del proyecto y permitan el futuro desarrollo del mismo. Este desarrollo no será profundizado en el documento, puesto no es el objetivo de la sección. Dado que la mayoría de DAW cuentan únicamente con un protocolo de comunicación MIDI, se seleccionó la aproximación de control por reloj MIDI para la sincronización de secuencias, la cual será desarrollada a continuación. Adicionalmente, se aprovecharán las ventajas del sistema OSC para comunicar la Raspberry Pi con el reloj MIDI implementado en Pure Data.

Utilizando el software de programación Pure Data, se construyó un programa de reloj MIDI, capaz de generar los pulsos necesarios para la sincronización, en base a un valor de tempo obtenido desde la Raspberry mediante OSC. Los mensajes MIDI utilizados pertenecen a la categoría de “System Real-Time Messages” o mensajes de sistema en tiempo real, en la cual, además de contener la sincronización MIDI, también dispone de algunos mensajes simples como inicio y fin de reproducción, que entregarán un mayor control sobre las secuencias desde el dispositivo. En la siguiente tabla se presentarán los mensajes que se utilizarán para la sincronización.

Nombre	Descripción	Mensaje (Sistema Decimal)
Pulso de Reloj	Enviado 24 veces por cada cuarto de nota cuando se requiere de sincronización	248
Inicio de Secuencia	Inicia la reproducción de la secuencia actual	250
Fin de Secuencia	Detiene la reproducción de la secuencia actual	252

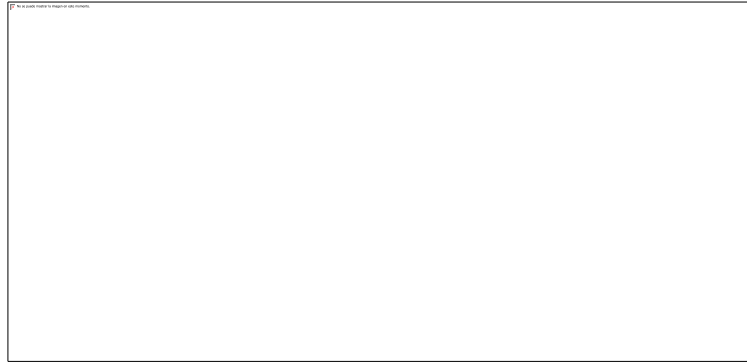
*Tabla 19. Mensajes de Sistema en Tiempo Real*

Para calcular el tiempo de espera entre cada pulso en base a un tempo dado, se realiza el siguiente cálculo.

$$\text{Tiempo entre Pulsos (segundos)} = \frac{\left(\frac{60}{\text{Tempo a sincronizar}}\right)}{24}$$

*Ecuación 13. Cálculo de diferencia de tiempo entre pulsos*

Finalmente, se tiene el siguiente esquema general de la etapa de sincronización.

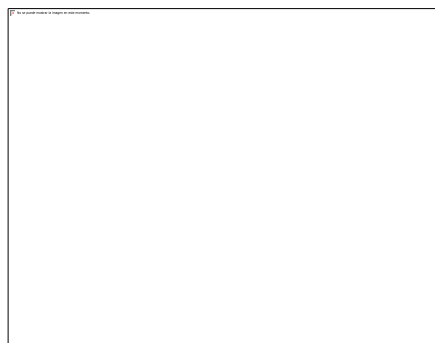


*Figura 44. Esquema final de sistema de sincronización*

### **Interfaz Gráfica**

El dispositivo contará con una interfaz gráfica que será presentada en una pantalla táctil, para que el usuario pueda interactuar con el dispositivo. En la Figura 45 se encuentra una captura de la interfaz que observará el usuario. Esta cuenta con un único botón que le permitirá iniciar el conteo en el pad para detectar el tiempo inicial e iniciar la reproducción de secuencias. Debajo de este se encontrará información sobre el tiempo que detecta el dispositivo y el tiempo de sincronización de las secuencias.

Por último, esta cuenta con una barra que le indicará al usuario el rango de tiempo de detección del sistema durante la interpretación y en que parte de este se encuentra.



*Figura 45. Interfaz Gráfica (Fuente propia)*

### **5.3 Evaluar la funcionalidad del dispositivo mediante la comparación de pruebas objetivas y subjetivas**

Para evaluar la funcionalidad del dispositivo se realiza un experimento que incluye tres pruebas. La primera evalúa la exactitud en la detección de tempo por parte del dispositivo; la segunda evalúa la sincronización entre el tempo ejecutado por el baterista y el tempo al cual la secuencia se reproduce; y la tercera evalúa la facilidad de uso del dispositivo. Es importante tener en cuenta que la primera prueba se evalúa de manera objetiva, mientras que las siguientes dos se evalúan de manera subjetiva. Estas tres categorías de análisis componen la definición de funcionalidad para este objetivo.

Para la selección de la población que es evaluada se hace uso de la técnica de muestreo por conveniencia, en la cual los sujetos son seleccionados dada la conveniente accesibilidad y proximidad de los sujetos para el investigador. Debido a factores como el tiempo y la disponibilidad de sujetos de prueba se determina que la muestra se compone de 5 personas.

#### **Evaluación de exactitud**

Entiéndase por exactitud la definición que el diccionario de la RAE (Real Academia Española) proporciona: “igual o que se asemeja en un grado muy alto a algo o alguien que es tomado como modelo”. En el caso de esta investigación, el modelo tomado es un metrónomo que marca el tempo guía al cual deberán ser ejecutados diferentes ritmos. Lo que se busca evaluar es qué tan semejante es el tempo capturado por el dispositivo en comparación con el tempo guía del metrónomo.

La prueba consiste en la ejecución de 5 ritmos a dos tempos diferentes, como se muestra en la siguiente tabla:



No. Prueba	Ritmo	Tempo (BPM)
1	Rock	80
2		120
3	Pop	80
4		120
5	Balada	80
6		120
7	Funk	80
8		120
9	Reggae	80

*Tabla 20. Ritmos que se ejecutan para la evaluación de exactitud del dispositivo*

Los sujetos de prueba ejecutarán cada ritmo durante el tiempo necesario para que el dispositivo registre 10 capturas de tiempo. Las partituras de los ritmos que se les presenta a los bateristas se encuentran en el Anexo A.

### **Evaluación de la sincronización**

Entiéndase por sincronización la definición que el diccionario de la RAE proporciona: “hacer que coincidan en el tiempo dos o más movimientos o fenómenos”. Para la aplicación de ese concepto a este proyecto, los fenómenos que deben coincidir en el tiempo son el tiempo de ejecución del baterista y el tiempo de reproducción de las secuencias.

Para evaluar lo anterior, se realiza un experimento que consta de tres partes. En la primera, el sujeto de prueba interpreta el ritmo Rock, mostrado en el Anexo A, e intenta mantener el tempo mientras la secuencia que se está reproduciendo se sincroniza a este. En la segunda, el sujeto de prueba interpreta el ritmo Rock e intenta acelerar progresivamente el tempo, mientras la secuencia que se está reproduciendo se sincroniza a este tempo. Por último, el sujeto de prueba interpreta el ritmo Rock e intenta retrasar progresivamente el tempo, mientras la secuencia que se está reproduciendo se sincroniza a este.

Luego, el sujeto da una calificación a las siguientes afirmaciones:

Según su percepción, califique qué tan adecuado fue el tiempo que le tomó al dispositivo sincronizar las secuencias a su tiempo, cuando:

- a. Intentó acelerar el tiempo
- b. Intentó retrasar el tiempo

La calificación se dará con una valoración de 1 a 5 de la siguiente manera:

<b>Calificación</b>	<b>Equivalencia Subjetiva</b>
1	Muy bueno
2	Bueno
3	Neutral
4	Malo
5	Muy malo

*Tabla 21. Categorías de calificación subjetiva para la encuesta de opinión de sincronización*

### **Evaluación de facilidad de uso**

La facilidad de uso, para este proyecto, se evalúa a partir de dos condiciones bajo las cuales el sujeto de prueba da su opinión acerca del uso del dispositivo. La primera es la interacción entre la interfaz y el usuario, y la segunda es la información proporcionada por el dispositivo a través de la pantalla. Para esto se toma en cuenta la definición que el diccionario de la RAE proporciona para el término “facilidad”: “condiciones especiales que permiten lograr algo o alcanzar un fin con menor esfuerzo”.

Para esta prueba se tiene en cuenta el experimento de sincronización realizado.

1. Según su percepción, califique qué tan fácil fue la interacción entre el dispositivo y el usuario.
2. En su opinión, fue de ayuda la información proporcionada por el dispositivo a través de la pantalla para:
  - a. Mantener el tiempo
  - b. Acelerar el tiempo

c. Retrasar el tiempo

Para responder la primera pregunta, los sujetos de prueba deben seleccionar un valor entre 1 y 5, de la siguiente manera:

<b>Calificación</b>	<b>Equivalencia Subjetiva</b>
1	Muy fácil
2	Fácil
3	Neutral
4	Difícil
5	Muy difícil

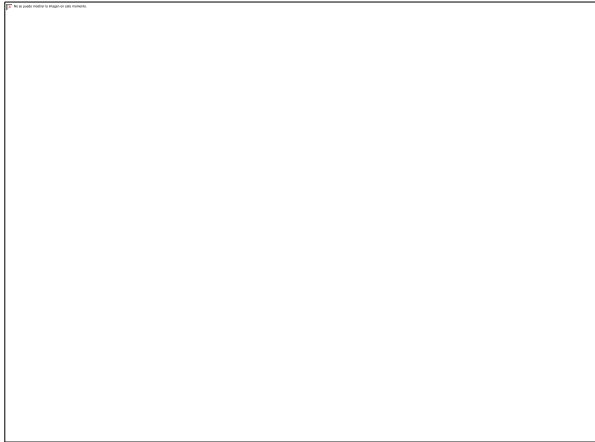
*Tabla 22. Categorías de calificación subjetiva para la encuesta de opinión de facilidad de uso, pregunta 1*

Para la segunda pregunta, los sujetos de prueba deben seleccionar un valor entre 1 y 5, de la siguiente manera:

<b>Calificación</b>	<b>Equivalencia Subjetiva</b>
1	Muy poco
2	Poco
3	Neutral
4	Mucho
5	Bastante

*Tabla 23. Categorías de calificación subjetiva para la encuesta de opinión de facilidad de uso, pregunta 2*

Conforme a las categorías de análisis planteadas se obtuvieron los siguientes resultados, con los cuales se evalúa la funcionalidad del dispositivo. Para comenzar, se presentan los resultados de la encuesta de opinión realizada para evaluar el tiempo que le tomó al dispositivo sincronizar la secuencia al tiempo detectado.



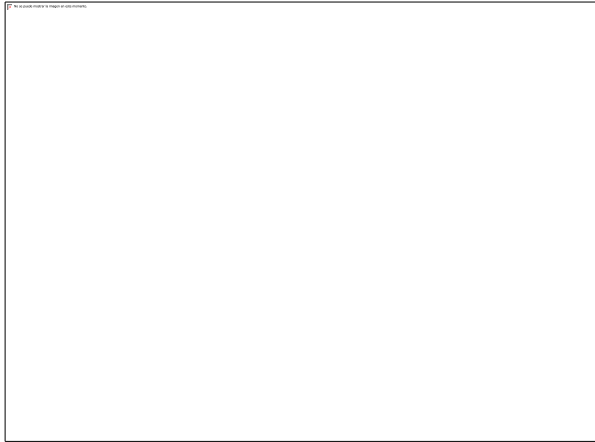
*Figura 46. Calificación pregunta No. 1 - Evaluación sincronización*



*Figura 47. Calificación pregunta No 2 - Evaluación sincronización*

La Figura 46 permite describir que, según la percepción de los sujetos de prueba, el tiempo que le tomó al dispositivo cuando intentaron acelerar el tempo fue neutral en un 80%, mientras que cuando intentaron retrasar el tempo, el tiempo fue bueno en un 60% como se observa en la Figura 47.

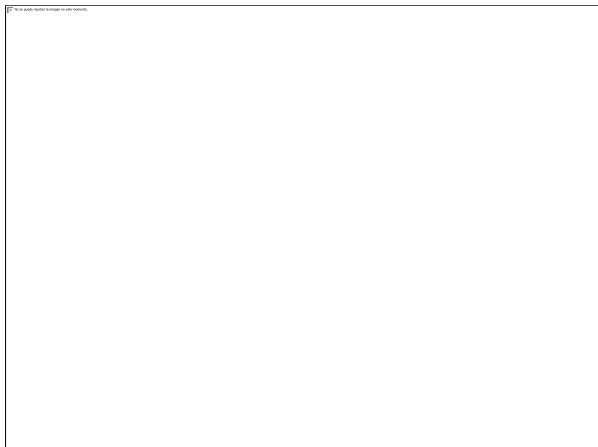
Ahora, se presentan los resultados de la encuesta de opinión realizada para evaluar la facilidad de uso del dispositivo. Para esto se tienen en cuenta las dos condiciones que permiten evaluar esta categoría de análisis de forma más detallada. La primera es la interacción entre el usuario y el dispositivo.



*Figura 48. Calificación Pregunta No. 1 - Evaluación facilidad de uso*

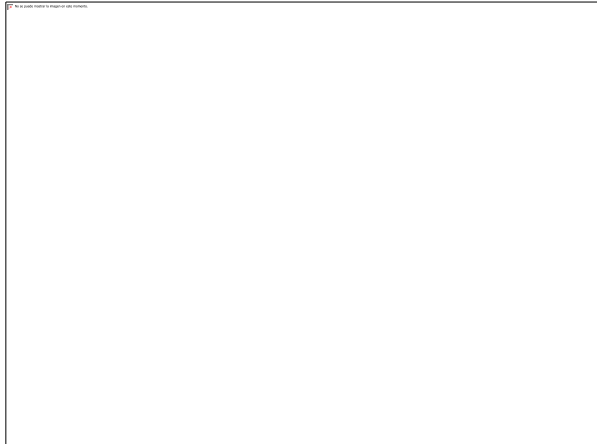
Los resultados que se pueden observar por medio de la Figura 48 permiten describir que la interacción entre el usuario y el dispositivo tiende a ser fácil, pues, aunque sólo el 20% de la muestra opinó que fue muy fácil la interacción, este porcentaje beneficia la calificación denominada “fácil” al generar una tendencia creciente y no al contrario.

La segunda condición que permite evaluar la facilidad de uso del dispositivo es la información proporcionada por el dispositivo a través de la pantalla.



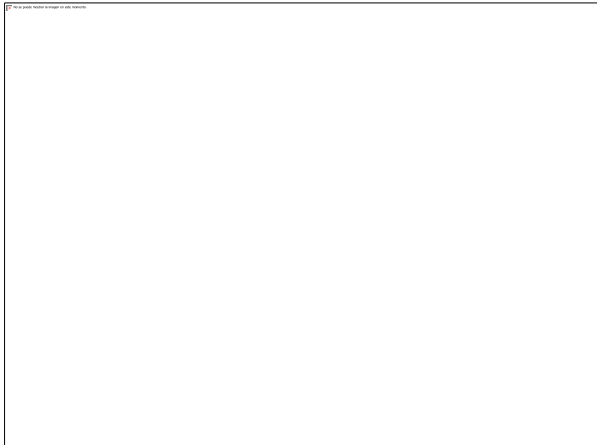
*Figura 49. Calificación Pregunta No 2-a - Evaluación facilidad de uso*

La Figura 49 permite observar que cuando los sujetos de prueba intentaron mantener el tiempo, la información proporcionada a través de la pantalla les fue de bastante ayuda.



*Figura 50. Calificación Pregunta No. 2-b - Evaluación facilidad de uso*

Ahora, cuando intentaron acelerar el tempo, la información proporcionada por la pantalla fue de bastante ayuda. Sin embargo, el porcentaje fue del 60%, esto es un 20% menos que al intentar mantener el tiempo.



*Figura 51. Calificación Pregunta No. 2-c - Evaluación facilidad de uso*

Por último, cuando los sujetos de prueba intentaron retrasar el tempo, la información proporcionada por medio de la pantalla fue de bastante ayuda en un 60%. Pero, al contrario del caso anterior, para el 20% de la muestra fue de mucha ayuda, lo que revela que la intención de retrasar el tempo está por encima de la intención de acelerarlo en un 20%.

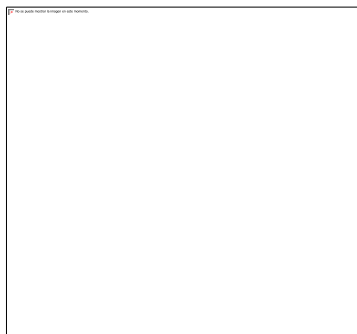
Es importante tener en cuenta que, aunque las respuestas a las encuestas de opinión fueron medidas por medio de porcentajes, estas son respuestas subjetivas, que dependen de la percepción de cada sujeto de prueba, por lo tanto no deben ser tomadas como factores definitivos sino como tendencias correspondientes a la muestra tomada.

## 6 PRESENTACIÓN Y ANÁLISIS DE RESULTADOS

Una vez culminado el proceso de desarrollo del proyecto, se obtuvo como resultado un sistema de detección de BPM y sincronización de pistas de forma inalámbrica, implementado en un dispositivo portátil acompañado por una pantalla táctil que presenta una interfaz gráfica con información importante para el usuario e interacción directa con el algoritmo.

Este prototipo fue diseñado específicamente para registrar y procesar señales provenientes de una batería acústica, obteniendo la información de los golpes del hi-hat, redoblante y bombo mediante sensores piezoeléctricos.

Para procesar la señal de manera digital, se implementó un circuito comparador que generase un pulso digital a partir de una señal análoga de entrada. La ventaja del comparador yace en la posibilidad de ajustar el umbral de voltaje a partir del cual se generará un pulso digital. Esta sección del circuito fue implementada con un divisor de voltaje en donde una de las resistencias es variable, con el propósito de ajustar el umbral de detección a conveniencia.



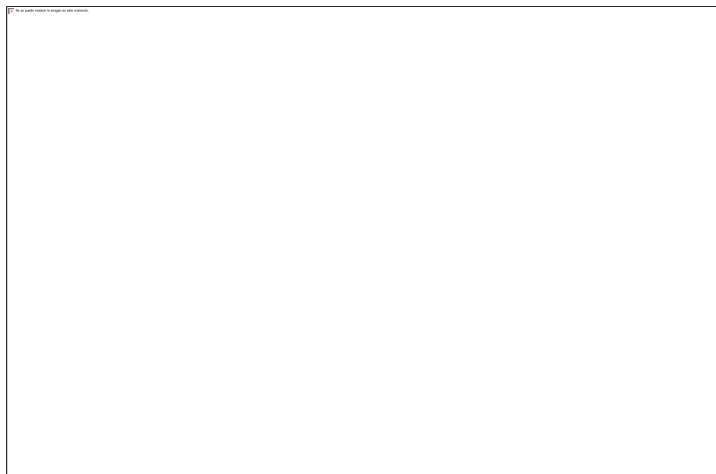
*Figura 52. Circuito comparador*





*Figura 53. Referencia*

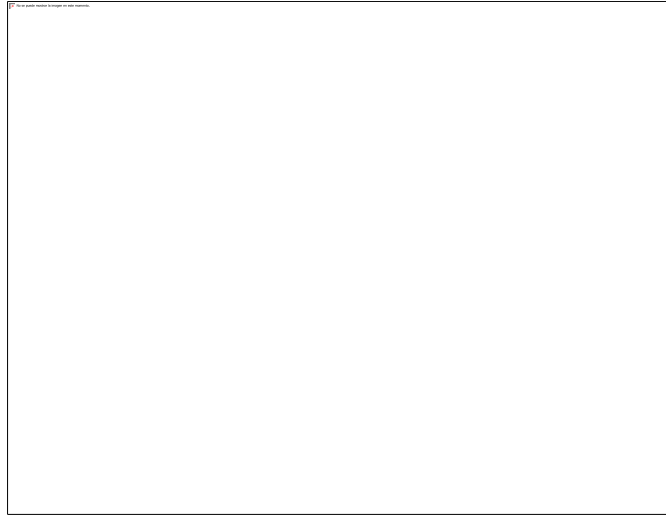
El circuito comparador presentado anteriormente es repetido 4 veces, para cubrir las 3 piezas de la batería y un pad de conteo inicial. Se utilizó la misma referencia variable para cada comparador excepto para el pad, el cual cuenta con su propia referencia variable independiente.



*Figura 54. Circuito final del proyecto*

El circuito presentado en la Figura 54, corresponde al diseño final del sistema de adquisición de señales. Las salidas de los comparadores fueron conectadas directamente a la tarjeta Raspberry Pi, la cual también se utilizó como fuente de voltaje para el circuito de adquisición.

El algoritmo de detección de tempo es evaluado en dos aspectos fundamentales que definen la efectividad del dispositivo, la precisión de detección y el procesamiento en tiempo real. En cuanto a la precisión, se compararon las pruebas preliminares junto con los resultados de los sujetos encuestados.



*Figura 55. Porcentaje de Error de Detección de Tempo Promedio en Prueba Preliminar*

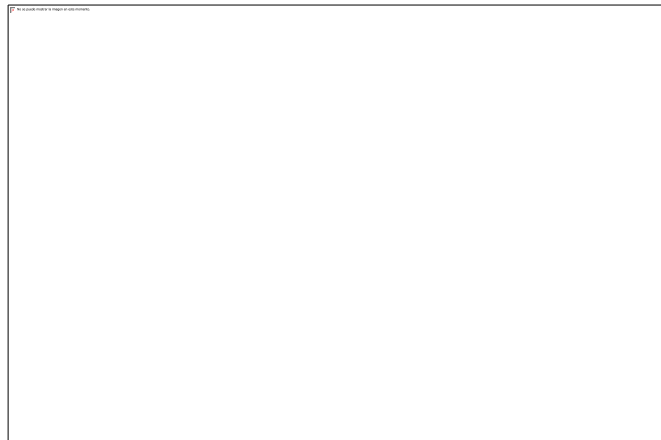


*Figura 56. Porcentaje de Error de Detección de Tempo Promedio en Personas Encuestadas*

En base a los resultados obtenidos, se encontró que para ninguno de los ritmos evaluados se obtuvo un porcentaje de error superior al 10%, lo cual cumple con el

requerimiento de precisión planteado en el objetivo 2. También, como se explicó en el objetivo 2, el porcentaje de error aumentó en los géneros de funk y reggae dado la cantidad de sub-múltiplos enteros detectados por el sistema, en ritmos donde el hi-hat interpreta figuras musicales de semi-corchea.

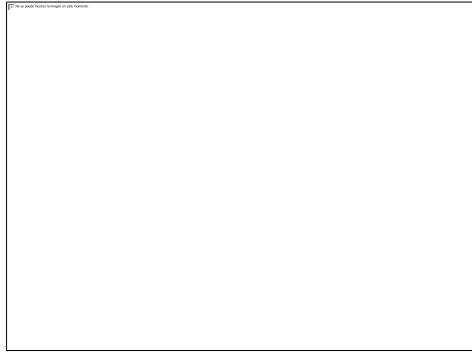
En el segundo aspecto, correspondiente al tiempo de procesamiento del algoritmo, se observaron los parámetros influyentes y se compararon con los resultados de las pruebas de percepción de tiempos de sincronización. Se encontró que en un rango de valores de tempo de detección de 60 BPM hasta 200 BPM, los tiempos de grabación y sincronización se mantienen estables, teniendo en cuenta la duración en compases como se muestra en la siguiente gráfica.



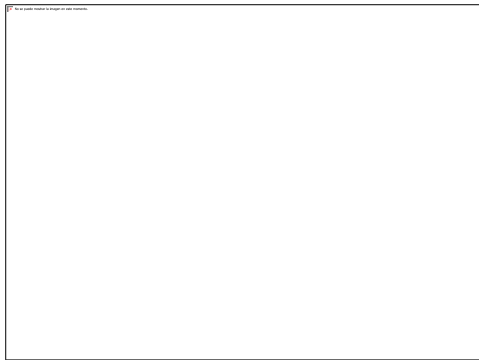
*Figura 57. Tiempo de Grabación + Procesamiento Expresado en Compases*

En promedio, el algoritmo se demora alrededor de 2.1 compases para generar una estimación de tiempo, como se presenta en la Figura 57.

Al analizar los resultados de las preguntas subjetivas sobre los tiempos de sincronización de tempo, se observa que la mayoría de los sujetos de prueba percibieron este intervalo de tiempo como apropiado, sin embargo, no ideal.



*Figura 58. Tiempo de Grabación + Procesamiento Expresado en Compases*



*Figura 59. Tiempo de Grabación + Procesamiento Expresado en Compases*

Para mejorar los intervalos de tiempo, del sistema se propone el ajuste de los parámetros influyentes estudiados en el objetivo 2, como el rango de detección. Aun así, el sistema pudo ser empleado por los participantes del experimento correctamente para cumplir con las tareas propuestas.

## 7 CONCLUSIONES

La disciplina de extracción de información de la música (MIR) es un campo de investigación relativamente pequeño y novedoso para el momento en el que se redactó este documento. La extracción del tempo a partir de señales musicales es una tarea compleja para una máquina, a pesar de ser tan simple para un ser humano, y es por eso que este campo de investigación se encuentra en crecimiento.

Gracias a los pioneros investigadores en el área, se recolectó gran cantidad de información que facilitó el desarrollo del dispositivo. Se logró implementar un sistema de detección de tempo capaz de grabar, procesar, realizar una estimación de tempo y sincronizar pistas secuenciadas en un DAW.

Para la detección, se empleó el método de autocorrelación, aprovechando las características de la herramienta para el tipo de señal de entrada que se obtuvo del instrumento percutivo.

El algoritmo de detección cumple con los requisitos de precisión para los diferentes géneros populares experimentados al registrar porcentajes de error inferiores al 10% propuesto en las competencias de algoritmos de detección de tempo. Como falencia, se encuentra que, para la interpretación de figuras musicales de corta duración como la semi-corchea, la precisión disminuye considerablemente y se deben ajustar parámetros como el rango de detección para compensar la deficiencia.

La sincronización de pistas mediante el reloj MIDI, y la comunicación entre la Raspberry y el DAW a través de OSC, fueron implementados de manera que se aprovechan las propiedades de cada protocolo de comunicación.

## 8 RECOMENDACIONES

Las recomendaciones a continuación serán desarrolladas según el orden de los objetivos. Para comenzar, respecto al método de captura de señales se sugiere investigar acerca de otra forma para realizar la adquisición. Es importante que al realizar comparaciones entre los diferentes transductores se analicen los mismos impulsos, es decir, se capture la misma fuente al mismo tiempo y con los transductores capturando al mismo tiempo.

Además, se debe tener en cuenta si la señal de salida de los transductores es análoga o digital dado que, al ser digital, podría entrar directamente a la tarjeta de adquisición sin la necesidad de pasar por un proceso de conversión a señales de dos estados, que, para el caso de este proyecto, fue el comparador.

Ahora, al referirse a la sección de programación se recomienda usar un lenguaje de programación más ligero y comercial como Java, C o C++, usando un microcontrolador no tan costoso como la tarjeta Raspberry Pi3. Y en relación con el método teórico usado para la detección del tiempo merece el trabajo analizar el Tempograma de Fourier y el Tempograma Cíclico, métodos propuestos en el libro “Fundamentals of Music Processing” escrito por Meinard Müller, texto guía usado en el desarrollo de este objetivo.

Para optimizar el dispositivo a partir de la sección electrónica, se recomienda hacer un procesamiento más profundo a la señal de entrada por medio de procesos análogos como filtros, de manera que las señales usadas para la detección de tiempo abarquen sólo la información necesaria para el correcto procesamiento digital. Además, es importante tener en cuenta que si se van a usar comparadores se debe asegurar que estos contengan trigger de Schmitt, de lo contrario se aconseja su implementación para que el rizado, producto del voltaje alterno del piezoeléctrico, no perjudique la fiabilidad de la captura.

Por último, se recomienda la implementación de un detector de Beats junto al detector de BPM dado que, para una una sincronización precisa, se requiere no sólo de la sincronización al tempo del baterista sino también de la sincronización a los beats del compás. Este aspecto es muy importante pues dos audios pueden estar sincronizados en tempo pero desfasados en beats, generando la sensación de desincronización entre el baterista y las secuencias implementadas.

## 9 BIBLIOGRAFÍA

- Battenberg, E., Huang, V., & Wessel, D. (Marzo de 2012). Toward Live Drum Separation Using Probabilistic Spectral Clustering Based on the Itakura-Saito Divergence. *Audio Engineering Society*, 10.
- Boylestad, R. L., & Nashelsky, L. (2013). *Electronic devices and circuit theory*. U.S.A: Pearson Education.
- Brown, J. C. (1993). Determination of the meter of musical scores by autocorrelation. 5.
- Buttazzo, G. C. (2011). *Hard Real-Time Computing Systems: Predictable scheduling algorithms and applications*. Springer.
- Chen, C.-W., Cremer, M., Lee, K., DiMaria, P., & Wu, H.-H. (2009). Improving perceived tempo estimation by statistical modeling of higher-level musical descriptors. *Audio Engineering Society*, 8.
- Chen, C.-W., Cremer, M., Lee, K., DiMaria, P., & Wu, H.-H. (Mayo de 2009). Improving Perceived Tempo Estimation by Statistical Modeling of Higher-Level Musical Descriptors. *Audio Engineering Society*, 8.
- Davies, M. E., Brossier, P. M., & Plumbley, M. D. (2005). Beat tracking towards automatic musical accompaniment. *Audio Engineering Society*, 7.
- Davis, G., & Jones, R. (1990). *Sound Reinforcement Handbook*. U.S.A: Hal Leonard Publishing Corporation.
- Davis, G., & Jones, R. (1990). *Sound Reinforcement Handbook*. Estados Unidos: Yamaha.
- Davis, M. E., Brossier, P. M., & Plumbley, M. D. (Mayo de 2005). Beat Tracking Towards Automatic Musical Accompaniment. *Audio Engineering Society*, 7.



- Deitel. (2002). *Python: How to program*. Deitel & Associates, Inc.
- Diaconescu. (2016).
- Earnhart, A. (2012). *Using piezoelectric technology to harvest energy from drums and inspire and engaging high school classroom experience*. Austin: The University of Texas .
- Electronics, R. (s.f.). *Using the I2C Bus*. Obtenido de Robot Electronics: <http://www.robot-electronics.co.uk/i2c-tutorial>
- Evans, M. J. (Octubre de 2007). Interactive Beat Tracking for Assited Annotation of Percussive Music. *Audio Engineering Society*, 14.
- Floyd, T. L. (2015). *Digital Fundamentals*. Londres: Pearson Education.
- Fowler, M. (2003). *UML Distilled: A brief guide to the standard object modeling language*. Pearson Education.
- Gärtner, D. (Enero de 2014). Unsupervised Learning of the Downbeat in Drum Patterns. *Audio Engineering Society*, 10.
- Goto, M. (2001). An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of new music research*, 159-171.
- Gouyon, F., & Dixon, S. (2005). A review of automatic rythm description systems. *Computer Musical Journal*, 23.
- Harnum, J. (2001). *Basic Music Theory*. Sol-Ut Press.
- Harnum, J. (2001). *Basic Music Theory: how to read, write and understand written music*. Sol-Ut Press.
- Hawley, S. (2013). *Fourier Transforms, Audio Engineering and the Quantum Nature of Reality*. Estados Unidos.

Hernandez Sampieri, R. (2014). *Metodología de la investigación*. México: Mc Graw Hill Education.

I2C-bus.org. (s.f.). *I2C - What's That?* Obtenido de I2C Bus: <http://www.i2c-bus.org/i2c-bus/>

Jochelson, D., & Fedigan, S. (2006). Design of an automatic beat-matching algorithm for portable media devices. *Audio Engineering Society*, 10.

Jochelson, D., & Fedigan, S. (Octubre de 2006). Design of An Automatic Beat-Matching Algorithm for Portable Media Devices. *Audio Engineering Society*, 10.

Kornfeld, J. (2005). *Music notation and theory for intelligent beginners*. Artwork.

Kornfeld, J. (2005). *Music Notation and Theory for Intelligent Beginners*.

Last Name, F. M. (Year). Article Title. *Journal Title*, Pages From - To.

Last Name, F. M. (Year). *Book Title*. City Name: Publisher Name.

Müller, M. (2015). *Fundamentals of music processing*. Suiza: Springer.

Müller, M. (2015). *Fundamentals of Music Processing*. Erlangen, Alemania: Springer.

Malvino, A., & Bates, D. J. (s.f.). *Electronic Principles*.

Miles Huber, D. (1991). *The MIDI Manual*. Focal Press.

Miles, D. (s.f.). *The MIDI Manual*. Focal Press.

Ochoa, C. (2015). *Netquest*. Obtenido de Muestreo no probabilístico: muestreo por conveniencia: [www.netquest.com](http://www.netquest.com)

OpenSourceControl. (s.f.). *Introduction to OSC*. Obtenido de [Opensoundcontrol.org](http://Opensoundcontrol.org).

Parncutt, R. (1994). A Perceptual Model of Pulse Saliency and Metrical Accent in Musical Rhythms. *European Journal of Neuroscience*.

Perron, M. (Octubre de 1991). How Steady Is Your Click Track? *Audio Engineering Society*, 15.

Perron, M. (November de 1994). Checking Tempo Stability of MIDI Sequencers. *Audio Engineering Society*, 16.

Python. (s.f.). *What is Python? Executive Summary*. Obtenido de Python: [www.python.org](http://www.python.org)

Python. (s.f.). *What is Python? Executive Summary*. Obtenido de Python Software Foundation: <https://www.python.org/doc/essays/blurb/>

Raspberry Pi Foundation. (s.f.). *What is a Raspberry Pi?* Obtenido de Raspberry Pi: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>

Rodet, X., & Jaillet, F. (s.f.). *Detection and Modeling of Fast Attack Transients*. Paris.

Rossiter, D., & Tsang, R. (Noviembre de 2007). Beat Deviation for Tempo Estimation Algorithms. *Audio Engineering Society*, 14.

Rossiter, D., Tsang, R., & So, R. H. (Noviembre de 2007). Beat Deviation for Tempo Estimation Algorithms. *Audio Engineering Society*.

Secuencias.com. (s.f.). *Secuencias*. Obtenido de Secuencias.com: [www.secuencias.com](http://www.secuencias.com)

Smat, R. (2012). Introduction to comparators, their parameters and basic applications. ST Microelectronics.

Sparkfun. (s.f.). *3.3 V CMOS Logic Levels*. Obtenido de Sparkfun: <https://learn.sparkfun.com/tutorials/logic-levels/33-v-cmos-logic-levels>

Sparkfun. (s.f.). *I2C*. Obtenido de Sparkfun: [https://learn.sparkfun.com/tutorials/i2c?\\_ga=2.263081740.1406484493.1496341865-1746785494.1496249448](https://learn.sparkfun.com/tutorials/i2c?_ga=2.263081740.1406484493.1496341865-1746785494.1496249448)

- Sparkfun. (s.f.). *Introduction to Operational Amplifiers with LTSpice*. Obtenido de learn.sparkfun.com: [https://learn.sparkfun.com/tutorials/introduction-to-operational-amplifiers-with-ltspice?\\_ga=2.190345954.857476032.1496249448-1746785494.1496249448](https://learn.sparkfun.com/tutorials/introduction-to-operational-amplifiers-with-ltspice?_ga=2.190345954.857476032.1496249448-1746785494.1496249448)
- Sparkfun. (s.f.). *TTL Logic Levels*. Obtenido de Sparkfun: <https://learn.sparkfun.com/tutorials/logic-levels/ttl-logic-levels>
- Sparkfun. (s.f.). *Voltage Dividers*. Obtenido de Sparkfun: <https://learn.sparkfun.com/tutorials/voltage-dividers>
- Stark, A. M., Plumbley, M. D., & Davies, M. E. (2007). Audio effects for real-time performance using beat tracking. *Audio Engineering Society*, 7.
- Stark, A., Plumbley, M., & Davies, M. (Mayo de 2007). Audio Effects for Real-time Performance Using Beat Tracking. *AES*, 7.
- Texas Instruments. (2017). LMx39-N, LM2901-N, LM3302-N Low-Power Low-Offset Voltage Quad Comparators.
- Uhle, C., & Dittmar, C. (Junio de 2004). Drum Pattern Based Genre Classification of Popular Music. *Audio Engineering Society*, 7.
- Uhle, C., & Dittmar, C. (2004). Generation of musical scores of percussive un-pitched instruments for automatically detected events. *Audio Engineering Society*, 9.
- Uhle, C., & Dittmar, C. (Mayo de 2004). Generation of Musical Scores of Percussive Un-Pitched Instruments from Automatically Detected Events. *Audio Engineering Society*, 9.
- Verry, T. (2012). *What is a raspberry?* Obtenido de Extremetech: [www.extremetech.com](http://www.extremetech.com)

Zapata, J. R., & Gómez, E. (2011). Comparative evaluation and combination of audio tempo estimation approaches. *Music Technology Group*, 10.

## 10 ANEXOS

### ANEXO A

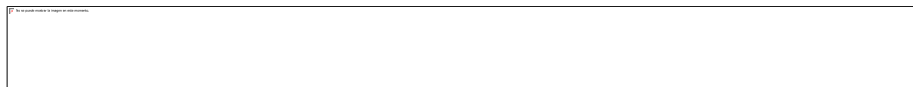
Partituras de ritmos usados en las pruebas de exactitud.



*Figura 60. Partitura para batería - Ritmo Rock*



*Figura 61. Partitura para batería - Ritmo Pop*



*Figura 62. Partitura para batería - Ritmo Balada*



*Figura 63. Partitura para batería - Ritmo Funk*



*Figura 64. Partitura para batería - Ritmo Reggae*