



Práctica 1: Introducción a GNURADIO

Carlos Fernando Carreño Jerez - 2201729

Juan Esteban Pinto Orozco - 2215585

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Universidad Industrial de Santander

25 de Febrero de 2024

abstract

GNU Radio stands out as a crucial tool for real-time signal manipulation. Through sampling theory, users can grasp the importance of sampling frequency in signal acquisition and processing. This document presents a series of initial steps in GNU Radio, along with some programs that allow demonstrating the use of sampling theory and its applications.

Palabras clave: GNURADIO, , Sampling theory

Resumen GNU Radio se destaca como una herramienta crucial para la manipulación de señales en tiempo real. A través de la teoría de muestreo, los usuarios pueden comprender la importancia de la frecuencia de muestreo en la adquisición y procesamiento de señales. En este documento, se plantea una serie de primeros pasos en GNU Radio, junto con algunos programas que permiten demostrar el uso de la teoría de muestreo y sus aplicaciones.

Palabras clave: GNURADIO, Teoría de muestreo

1. Introducción

En el vasto panorama de las comunicaciones y el procesamiento de señales, GNU Radio emerge como una herramienta indispensable. Esta plataforma de código abierto ofrece una interfaz flexible y poderosa para el diseño y la implementación de sistemas de radio definidos por software. Uno de los aspectos fundamentales en el estudio de GNU Radio es el análisis de frecuencia de muestreo, que constituye un pilar esencial en la comprensión de cómo se manipulan y procesan las señales.



Figura 1: GNU Radio

El objetivo de este trabajo es explorar y comprender tanto los fundamentos de GNU Radio como los principios subyacentes del análisis de frecuencia de muestreo. Este análisis no solo implica entender cómo se realiza el muestreo de una señal, sino también cómo diferentes frecuencias de muestreo afectan la representación y la manipulación de las señales en el dominio digital.

En este contexto, se examinará cómo se pueden utilizar herramientas específicas dentro de GNU Radio para llevar a cabo un análisis exhaustivo de las características espectrales de una señal. Se explorarán técnicas para la visualización de espectros de frecuencia, la detección de picos espectrales, la medición de ancho de banda y la identificación de componentes frecuenciales relevantes en la señal.

En resumen, este trabajo pretende sumergirse en el mundo de GNU Radio y el análisis de frecuencia de muestreo, proporcionando una visión general de cómo esta poderosa herramienta de software puede utilizarse para explorar y comprender las complejidades de las señales en el dominio digital.

2. Metodología

El desarrollo de este laboratorio fue realizado a partir de 2 fases metodológicas, correspondientes a la parte A y B de la práctica en cuestión, en las cuales se trataron temáticas distintas, pero a la vez complementarias para el desarrollo correcto de esta, a continuación, se describe a detalle los pasos llevados a cabo para el correcto cumplimiento de la práctica.

2-A. Parte A

En la primera parte de este laboratorio se enfocó en la creación de habilidades y conceptos referentes al programa de GNU radio, creando la base de conocimiento del mismo para en la parte B poder realizar el desarrollo de la práctica correctamente.

1. Leer el material complementario correspondiente a la práctica vigente, con el objetivo de esclarecer conceptos útiles para el desarrollo correcto del laboratorio, además de comprender el funcionamiento



to del software GNU radio, el cual se implementará como la base para el desarrollo de esta práctica.

Material :

- <https://wiki.gnuradio.org/index.php/Tutorials>
- wireless Communication Systems - From RF Subsystems to 4G Enabling Technologies chapter 12.2. Sampling

2. Iniciar el software GNU radio, familiarizarse con la interfaz y las funcionalidades que brindan los distintos bloques de dicho programa, además de ejecutar por primera vez un diagrama de flujo por medio de los bloques mencionados anteriormente.
3. Se realizar una introducción de las variables básicas que se pueden operar en el software GNU radio, además de comprender la clasificación de los datos en el programa según el código de colores del mismo.
4. Se analiza un nuevo enfoque para las variables anteriormente trabajadas, entendiendo que estas pueden variar tanto al estar en términos de otras variables como durante el tiempo de ejecución del programa
5. Se realiza un enfoque más detallado en el tipo de datos del programa, analizando la salida del diagrama de flujo según el tipo de dato asociado a los bloques.

2-B. Parte B

Esta segunda parte se enfocará más al análisis de la frecuencia de muestreo de las señales, con el fin de comprender su importancia, para ello se trabajará inicialmente con problemas particulares, como son las señales sinusoidales hasta llegar a señales reales como audios o canciones.

1. Se analizará el teorema de frecuencia de Nyquist por medio de un diagrama de flujo planteado en el software de GNU radio, el cual nos permitirá modificar la frecuencia de muestreo y la frecuencia de la señal base, además se trabajará con diferentes tipos de señales con el fin de determinar las ventajas y desventajas de implementar dicho teorema.
2. Se analizará los efectos sobre la forma de onda (representación de la señal en el dominio del tiempo) de una señal muestreada 5 veces la frecuencia fundamental de la señal, esto por medio del software de GNU radio.
3. Se analizarán las ventajas y desventajas presentes al muestrear una señal con una frecuencia 10 veces mayor a la frecuencia fundamental de la señal, además de la forma de onda ilustrada en este proceso.

4. Se demostrará el efecto causado en una señal por medio del proceso de interpolación y decimación a través de GNU radio, proceso semejante a la expansión y compresión en el tratamiento de señales continuas.
5. Se analizarán los efectos ocasionados por la decimación y la interpolación cuando son aplicados a una señal de audio, además de la función del bloque "Multiply Const" y las afectaciones del teorema de Nyquist a esta misma señal.
6. Se implementarán distintos filtros a la misma señal de audio, con el objetivo de poder separar, tonalidades, voces, instrumentos y demás variables de la señal, para esto se implementarán los conocimientos adquiridos en esta práctica.
7. Finalmente se realizará el análisis de los resultados adquiridos en la práctica y se redactará un informe e investigación de la misma.

3. Resultados de la Práctica

3-A. Parte A

1. Estas lecturas nos han proporcionado una comprensión básica del funcionamiento de GNU Radio, así como una comprensión más profunda del papel crucial que desempeña la frecuencia de muestreo en el programa. Mientras que GNU Radio nos ha permitido familiarizarnos con sus operaciones esenciales, la exploración de la frecuencia de muestreo nos ha permitido comprender mejor los principios fundamentales que subyacen en su funcionamiento.
2. Se llevó a cabo el primer programa en GNU Radio, el cual consistió en un análisis de señal en los dominios del tiempo y de la frecuencia. Este programa puede considerarse como el equivalente al clásico 'Hola Mundo' que se realiza en diferentes lenguajes de programación, marcando así un hito inicial en el aprendizaje y la comprensión de las capacidades de GNU Radio.

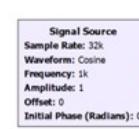
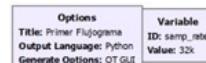


Figura 2: Diagrama de flujo para el análisis de una señal en el dominio del tiempo y la frecuencia.



Dando los siguientes resultados

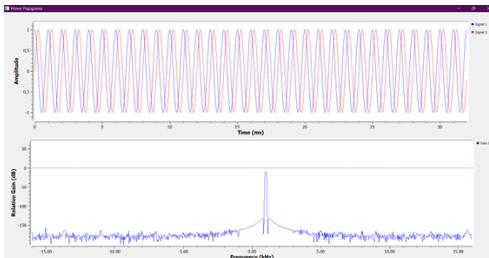


Figura 3: Representación de la señal en el dominio del tiempo y la frecuencia.

3. Se miro como era la clasificacion y el funcionamiento de los diferentes codigos de clasificacion para el cual se realizo la siguiente tarea después de haber visualizado el funcionamiento del sistema, utilizando una señal senoidal ideal a una frecuencia de 100 Hz como referencia. Este enfoque permitió comprender mejor el proceso y garantizar una ejecución precisa de la tarea propuesta.

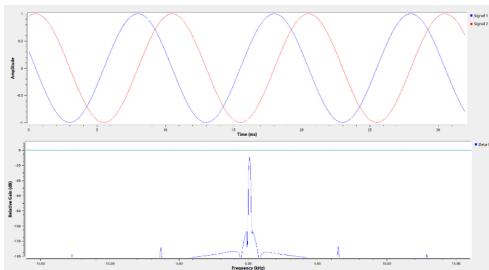


Figura 4: Representación de la señal sen a 100hz en el dominio del tiempo y la frecuencia.

4. Se realizó la adaptación al sistema, lo que nos permitió tener una señal que variara dentro de ciertos valores que especificamos. Como se muestra en la siguiente figura, se ajustó el programa para lograr este objetivo. Dando el siguiente resultado.

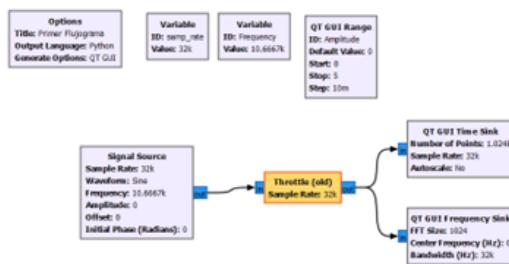


Figura 5: Representación en GNU radio de una señal en Tiempo y Frecuencia

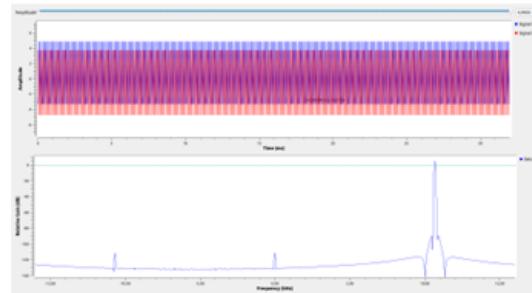


Figura 6: Representación de la señal sen en variacion entre 0 y 5 en su rango en el dominio del tiempo y la frecuencia.

5. En esta sección, se llevó a cabo la unión de todos los conceptos previamente comprendidos, lo que condujo a la generación de la siguiente señal dentro del programa, siguiendo unas características específicas proporcionadas.

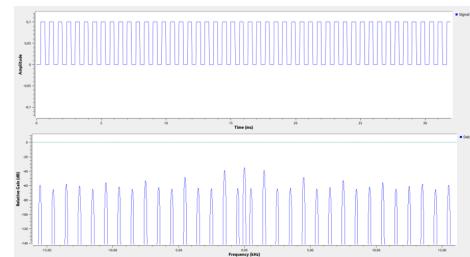


Figura 7: Representación de la señal cuadrada en el dominio del tiempo y la frecuencia.

3-B. Parte B

1. En esta parte, se examinó el teorema de Nyquist, para el cual se obtuvieron los siguientes tres resultados que muestran una señal seínoideal en la frecuencia de Nyquist, cuando está a una frecuencia mayor y cuando está a una frecuencia menor.

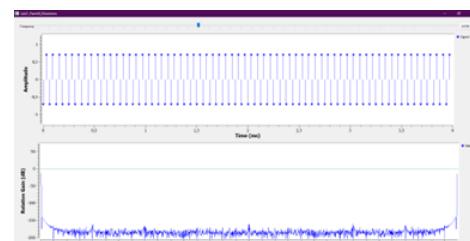


Figura 8: Representación de una señal sinusoidal muesreada con la frecuencia de Nyquist.

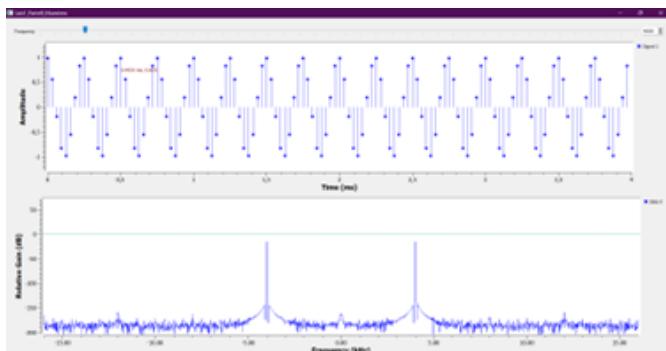


Figura 9: Representación de una señal sinusoidal muestreada a una frecuencia mayor que la frecuencia de Nyquist.

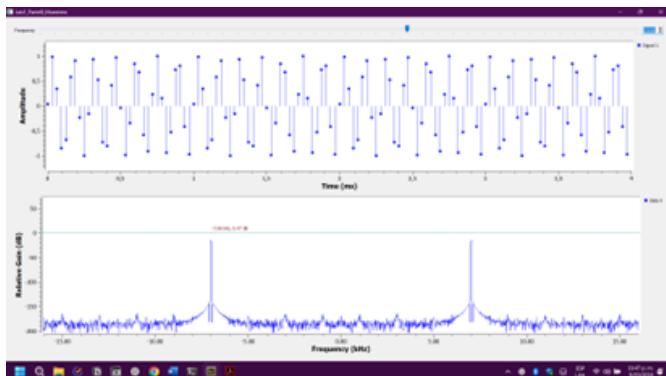


Figura 10: Representación de una señal sinusoidal muestreada con la frecuencia de Nyquist.

Este mismo proceso se aplicó a una señal cuadrática, en la cual se sabe que en el análisis de la frecuencia se observarán más de solo dos picos por punto de frecuencia dando los siguientes resultados.

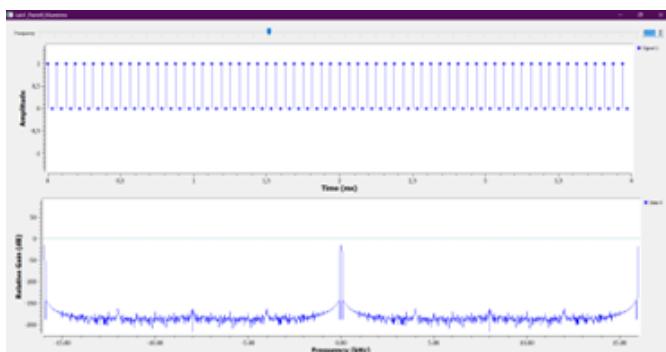


Figura 11: Representación de una señal cuadrada muestreada a la frecuencia de Nyquist.

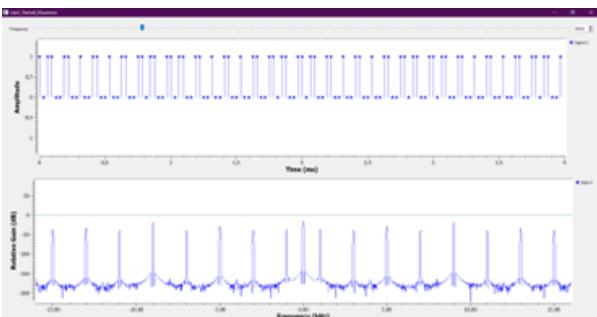


Figura 12: Representación de una señal cuadrada muestreada a una frecuencia mayor que la frecuencia de Nyquist.

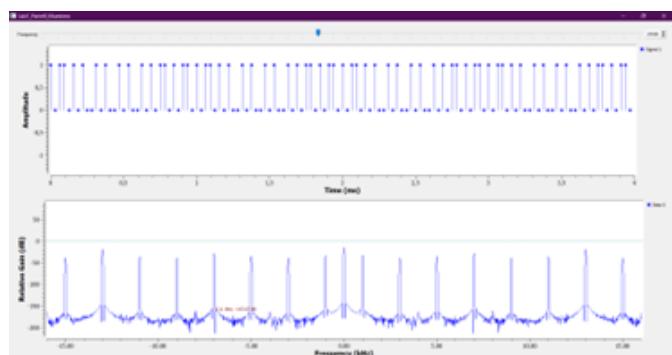


Figura 13: Representación de una señal cuadrada muestreada a una frecuencia menor que la frecuencia de Nyquist.

Este mismo proceso se aplicó a una señal triangular, la cual, en la gráfica de muestreo, genera tres puntos picos por cada corte de frecuencia para la cual igual que las dos anteriores señales se midieron en los puntos de la frecuencia de Nyquist dando los siguientes resultados.

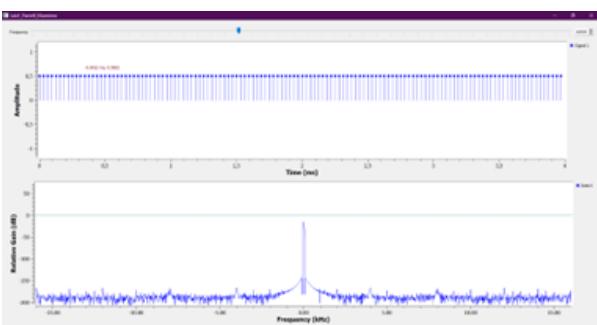


Figura 14: Representación de una señal triangular muestreada a la frecuencia de Nyquist.

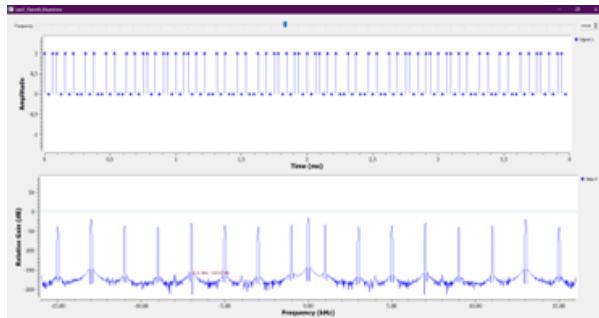


Figura 15: Representación de una señal triangular muestreada a una frecuencia mayor que la frecuencia de Nyquist.

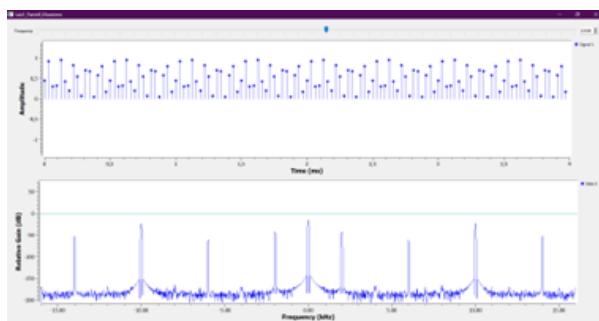


Figura 16: Representación de una señal triangular muestreada a una frecuencia menor que la frecuencia de Nyquist.

2. En este punto, se busca examinar los efectos y beneficios de tener cinco veces el límite de Nyquist. Para ello, se realizó un experimento utilizando una señal senoidal ideal de 6 kHz que fue muestreada a 30 kHz, así como las mismas condiciones pero aplicadas a una señal cuadrada.

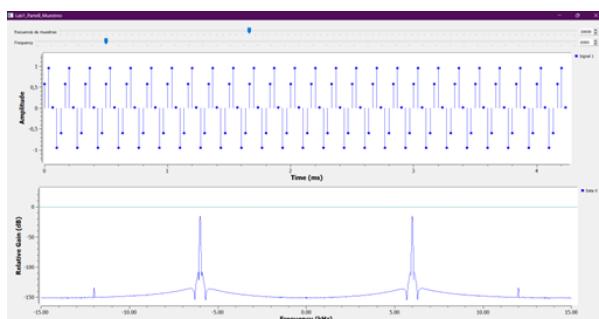


Figura 17: Representación de una señal de sinusoidal muestreada a una frecuencia 5 veces mayor que su frecuencia fundamental.

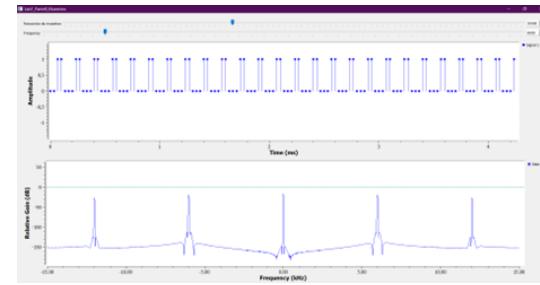


Figura 18: Representación de una señal de cuadrada muestreada a una frecuencia 5 veces mayor que su frecuencia fundamental.

3. En esta parte, simplemente se ajustó para que la relación entre la frecuencia de muestreo y la frecuencia de la onda fuera al menos diez veces mayor.

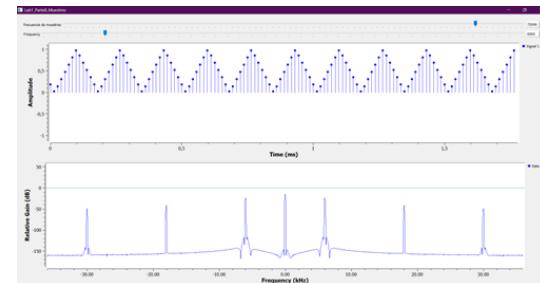


Figura 19: Representación de una señal triangular muestreada a una frecuencia 10 veces mayor que su frecuencia fundamental.

4. En esta sección, se implementó un nuevo bloque que permite realizar decimación e interpolación. Se llevó a cabo una prueba inicial utilizando una señal con una decimación de 4 e interpolación de 4 por separado, con el fin de visualizar el efecto que generaba.

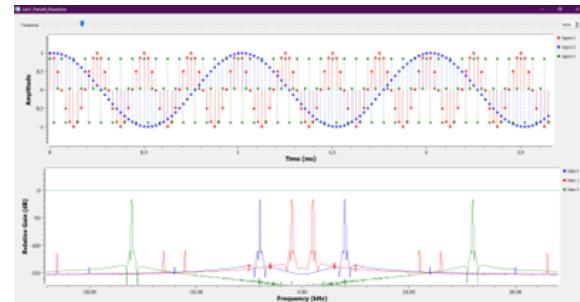


Figura 20: Representación de una señal sinusoidal con una decimación e interpolación igual a 4



Luego de seguir lo solicitado, se determinó que era más apropiado realizar una decimación a un valor menor al ya realizado y otra interpolación a un valor mayor al ya realizado.

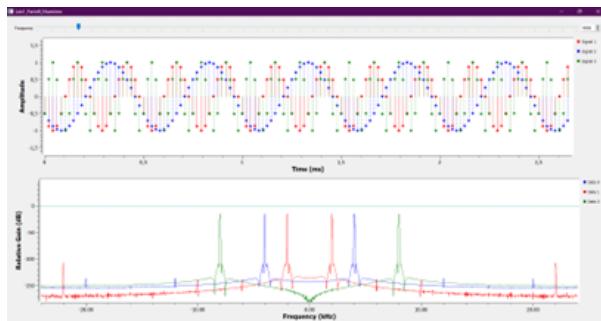


Figura 21: Representación de una señal sinusoidal con una decimación e interpolación igual a 2

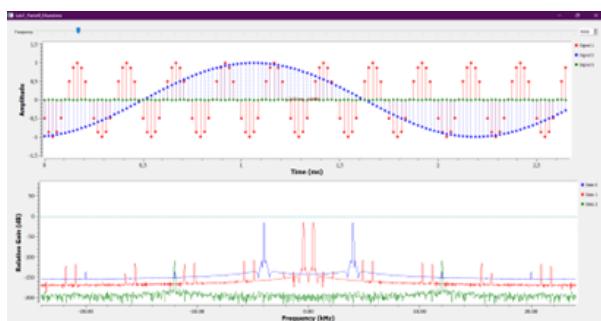


Figura 22: Representación de una señal sinusoidal con una decimación e interpolación igual a 9

- En este punto del informe, se decidió utilizar el audio de la bachata, el cual tiene una frecuencia de 48 [kHz]. Se procedió primero a realizar un ajuste utilizando el bloque de "Multiply Const", lo que permitió observar el efecto resultante en la señal.

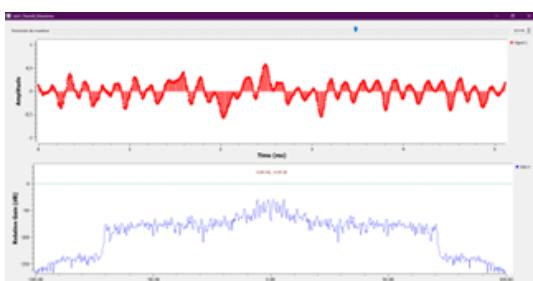


Figura 23: Representación de una señal de audio con un Multiply Const igual a 1.

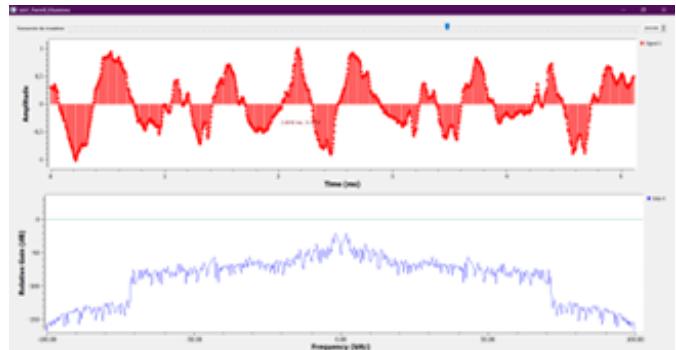


Figura 24: Representación de una señal de audio con un Multiply Const igual a 2.

Tambien se realiza la comprobacion del efecto de la decimación y interpolación.

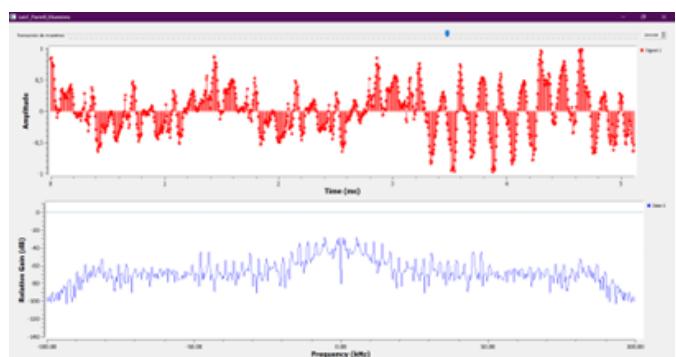


Figura 25: Representación de una señal de audio con decimación igual a 2.

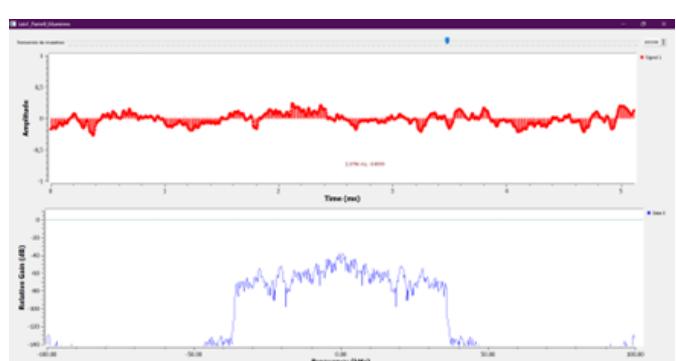


Figura 26: Representación de una señal de audio con interpolación igual a 2.

- En esta etapa del informe, se llevó a cabo el proceso de análisis de la pista de audio utilizada en la sección anterior. Esto se logró mediante el uso de filtros pasa bajas, pasa altas y un pasabandas, los



cuales se emplearon para enfocarse en diferentes partes específicas del audio.

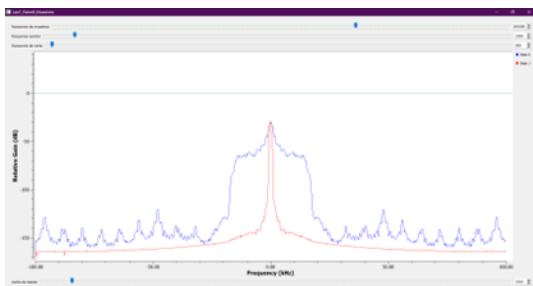


Figura 27: Representación de una señal de audio bajo un filtro pasa bajos.

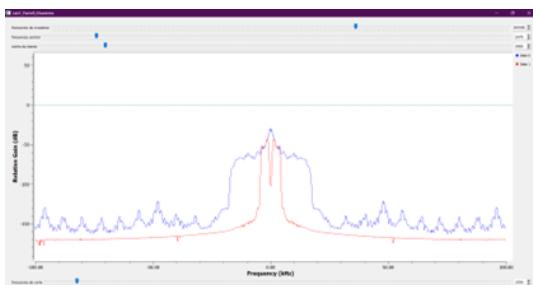


Figura 28: Representación de una señal de audio bajo un filtro pasabanda.

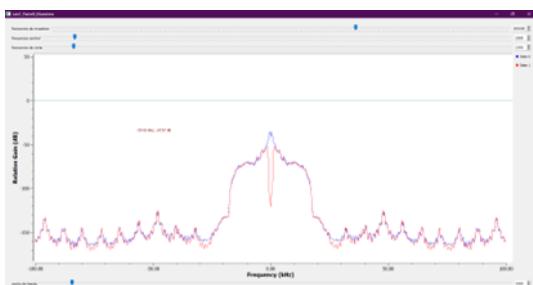


Figura 29: Representación de una señal de audio bajo un filtro pasa altas.

4. Análisis de Resultados

4-A. Parte A

Puntos 1 y 2.

Esta práctica comenzó con la familiarización del estudiante con el software GNU radio, programa el cual será la base para la construcción del conocimiento a partir del análisis de las diversas problemáticas planteadas. Como primer acercamiento al programa se nos pidió recrear el diagrama ilustrado en la 2. esto con el

objetivo de poder representar tanto en el dominio del tiempo como en el de la frecuencia una señal sinusoidal compleja, para ello utilizamos los bloques, Variable, QT GUI Frequency Sink, QT GUI Time Sink, Signal Source y Throttle, el resultado de este diagrama de flujo al momento de su ejecución se puede visualizar en la 3.

Basándonos en lo observado en la práctica y las guías suministradas (ya enunciadas en las partes A, 1) podemos describir correctamente el funcionamiento de cada bloque.

1. Variable: Crea una manipulación dinámica de parámetros dentro de un flujo de señal., mediante el proceso de la asignación de un valor que estaría bajo cierto indicador.

Variable
ID: samp_rate
Value: 32k

Figura 30: Bloque correspondiente a “Variable” en GNU radio [1]

2. QT GUI Frequency Sink: Actúa como una interfaz gráfica, lo que permite visualizar múltiples señales en el dominio de la frecuencia, es decir, nos permite mostrar en el programa la magnitud del espectro de la señal en cuestión.

QT GUI Time Sink
Number of Points: 1.024k
Sample Rate: 32k
Autoscale: No

Figura 31: Bloque correspondiente a “Signal Source” en GNU radio [1]

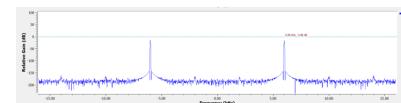


Figura 32: Representación de una señal coseno con el bloque QT GUI Frequency Sink en GNU radio.

3. QT GUI Time Sink: Tiene la capacidad de tomar conjuntos de flujos flotantes o complejos y trazarlos en el dominio del tiempo. Cada señal se traza con



un color diferente y a partir de las opciones del bloque se pueden modificar la etiqueta y el color de un número de entrada determinado. En Conclusión, cumple la misma función que el bloque QT GUI Frequency Sink pero en este caso trabajando en el dominio del tiempo.

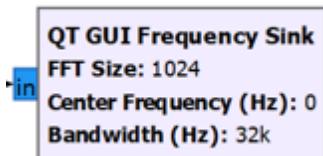


Figura 33: Bloque correspondiente a “QT GUI Time Sink” en GNU radio [1]

4. Signal Source: Es el generador de señales, el encargado de generar una variedad de formas de onda, como triangulares, cuadradas, sinusoidales y diente de sierra. Admite una salida de tipo complex, float, int y short.

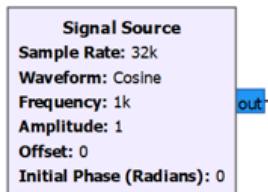


Figura 34: Bloque correspondiente a “Signal Source” en GNU radio [1]

5. Throttle: Regular el flujo de muestras de manera que la tasa promedio no exceda la tasa específica (en muestras por segundo). En otras palabras, el bloque Throttle nos permite regular la velocidad promedio a la que se consumen las muestras de entrada, es decir que al existir tantos datos siendo enviados al programa este se encarga de regular el flujo al que estos mismos pasan, con el objetivo de que el programa los procese de forma correcta, por eso es esencial considerarlo en el desarrollo de esta práctica.

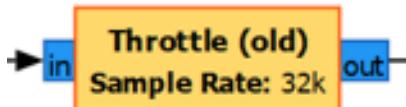


Figura 35: Bloque correspondiente a “Throttle” en GNU radio [1]

Puntos 3.

En la siguiente actividad de la practica se presentaron dos nuevos conceptos, el primero siendo las propiedades de los colores en el software de GNU radio y la asignación de variables en el diagrama de Flujo.

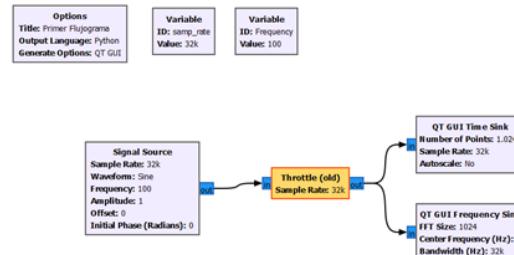


Figura 36: Diagrama de flujo para el análisis de una señal en el dominio del tiempo y la frecuencia utilizando bloques de variables

Para esto se recreo el diagrama de flujo correspondiente a la 36. En el cual se asignaron dos bloques correspondientes a variables, asignándole a estas el nombre de "Samp rate = 32kHz" y "Frequency = 100Hz", siendo así los parámetros que modificarán los demás bloques presentes en el esquema, ya que estos son implementados a modo de funciones, dando como resultado las graficas observadas en la 4. (Seno complejo en el dominio del tiempo y la frecuencia).

A continuación, se puede visualizar el significado explícito de cada color en GNU radio además de una visualización de estas mismas en la asignación de los parámetros del bloque QT GUI Frequency Sink:

Color	Parameter
Orange	Samp rate
Purple	Name, Frequency
Green	FFT Size, Window Type, Normal Window Power, Center Frequency (Hz), Bandwidth (Hz), Grid, Autoscale, Average, Xmin, Xmax, Ymin, Ymax, Ylabel, Yunits, Number of Inputs, Update Period, GUI Hint, Show Msg Ports

- Punto flotante: naranja
- Entero: verde
- Cuerda: violeta

Figura 37: Código de colores y asignación de parámetros en el bloque QT GUI Frequency Sink

De esto se puede concluir que al momento de la asignación de un parámetro específico se debe considerar el tipo de dato del mismo, con el fin de evitar errores en la ejecución del código.



Puntos 4 En la sección 4 de la parte A del correspondiente laboratorio, nos enfocamos en la implementación de variables modificables durante la ejecución del programa.

Para el análisis de esto se implemento el diagrama ilustrado en la 5. En el cual se incorporó el bloque “QT GUI Range”, el cual permite la creación de una variable modificable durante el proceso de ejecución.

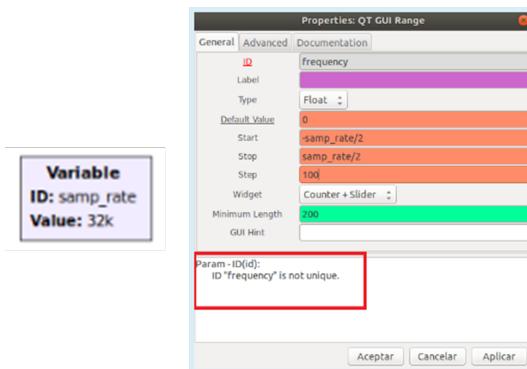


Figura 38: Asignación de parámetros para el bloque QT GUI Frequency Sink.

Esta nueva herramienta proporciona una gran utilidad al momento de analizar los diferentes modelos planteados en la práctica, ya que permite visualizar como cambia cada sistema el variar los parámetros presentes en el mismo, en la fig8. Se puede observar la implementación de una señal seno la cual permite variar su amplitud entre 0 y 5 mediante una “scrollbar”, permitiendo observar cambios en tiempo real tanto en su representación en el dominio del tiempo como en el dominio de la frecuencia.

Puntos 5 En la parte final de la práctica A se indicó los diferentes tipos de datos correspondientes a las entradas y salidas de los bloques.

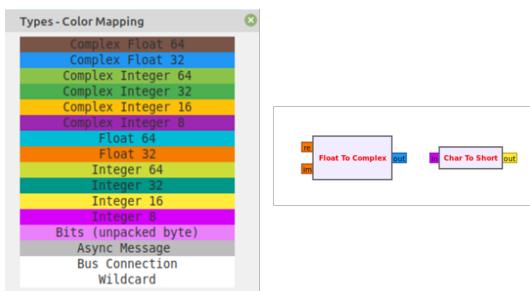


Figura 39: tipos de datos y colores asociados

A partir de estas se logro visualizar que hay afectaciones en la presentación en el dominio del tiempo, ya

que por medio del esquema trazado en la fig9. Se logro visualizar el comportamiento que presenta al momento de su ejecución cuando se trabaja con una variable compleja en contraste con una flotante.

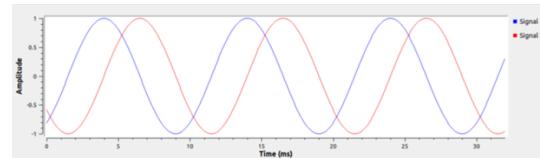


Figura 40: Representación de una señal coseno tipo Complex float 32.

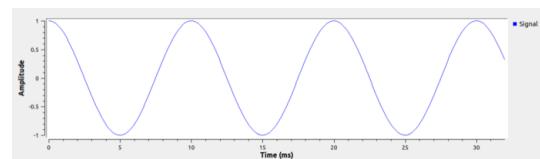


Figura 41: Representación de una señal coseno tipo float.

Apreciando que en Complex 32 se utiliza un flotante de 32 bits para el componente real y un flotante de 32 bits para el componente imaginario, en cambio en el tipo flotante la señal coseno es una salida real y única en el dominio del tiempo. Se puede apreciar en la fig10. El tipo de dato flotante para una señal cuadrada.

4-B. Parte B

Puntos 1 Al muestrear una señal exactamente en el límite de la frecuencia e Nyquist, es decir al doble de su ancho de banda (para una reconstrucción sin pérdidas), podemos obtener una representación básica de la señal original (2 muestras por Periodo), al ser muestreada de esta manera evitamos causar una alteración en la frecuencia original de dicha señal al momento de su reconstrucción y de esta forma evitar el efecto de Aliasing, fenómeno el cual se da al muestrear la señal de origen con una frecuencia menor a la de Nyquist, ocasionando que la frecuencia original de dicha señal cambie, causando así una superposición de la señal muestreada al momento de ser reconstruida y por ende una representación incorrecta de la misma (Esta se puede evidenciar al realizar un contraste de la señal representada en el tiempo con su espectro). Por otro lado, al muestrear una señal en el punto exacto de la frecuencia de Nyquist obtenemos una representación más fácil de procesar, pero menos precisa que si la muestreamos a una frecuencia mucho mayor, pero si nos declinamos por usar una frecuencia mínimamente menor a la de Nyquist es decir por debajo de este límite, nos



arriesgaríamos a presentar una pérdida de información debido a la incapacidad que se presentaría de captar algunas de las frecuencias más altas de la señal original (ya entrando en el efecto de Aliasing anteriormente descrito). Lo descrito anteriormente se puede apreciar con mas detalle en las figuras 8,9,10,11,12,13,14,15,16, las cuales fueron elaboradas variando los parámetros del esquema ilustrado en la 42.

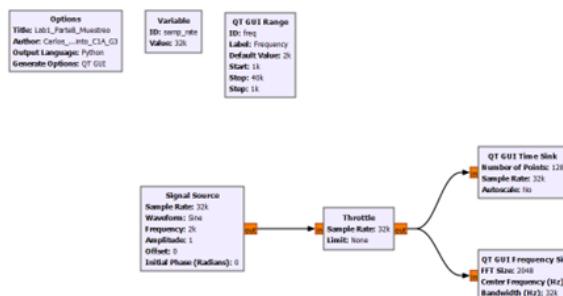


Figura 42: Representación de una señal coseno tipo float.

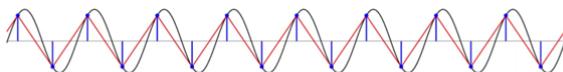


Figura 43: Seno muestreado exactamente con la frecuencia de Nyquist [2]

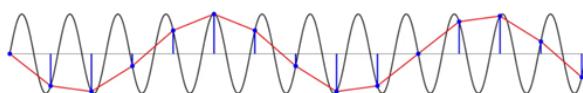


Figura 44: Alteración en la frecuencia de la señal que se reconstruirá debido a muestrearla con una frecuencia menor a la frecuencia de Nyquist [2]

Visualizando el comportamiento de cada uno de los tipos de señales ilustrados en las imágenes referenciadas y expuestas en esta misma sección, podemos concluir que le teorema de Nyquist es válido para cualquier tipo de señal continua, independientemente de su forma de onda, ya sea una señal cuadrada, triangular o de sierra. Por ende, para una representación más fidedigna de la señal muestreada es recomendable muestrearla a una frecuencia mayor a la de Nyquist considerando a sobre carga en el análisis computacional de la misma señal.

Puntos 2

Cuando el muestreo es cinco veces mayor que la frecuencia de la señal, se asegura una representación

precisa y fiel de la misma, evitando el aliasing y preservando adecuadamente la información de alta frecuencia, tal y como se puede apreciar en la forma de onda de las señales ilustradas anteriormente (representación de la señal en el dominio del tiempo), ya que visualizando dicho muestreo se puede deducir a simple vista la señal de origen a la que corresponden. Esta mayor tasa de muestreo permite una mejor resolución temporal, capturando con precisión los cambios rápidos en la señal a lo largo del tiempo, lo que facilita el procesamiento digital y garantiza una mayor calidad en la representación digital de la señal. Las señales muestreadas a esta frecuencia se muestran en las figuras 17 y 18, esquemático 45.

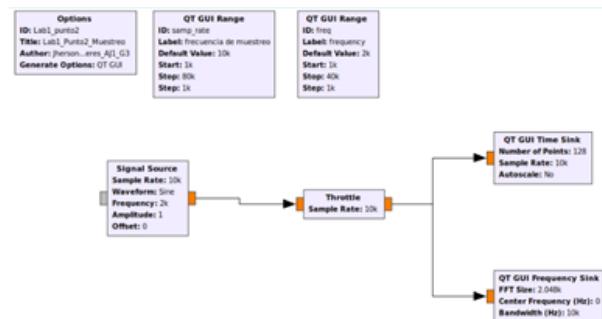


Figura 45: Diagrama de flujo correspondiente al muestreo de una señal con una frecuencia 5 veces mayor a la fundamental

Puntos 3

Cuando el muestreo supera diez veces la frecuencia de la señal, se logra una representación excepcionalmente precisa y detallada de la misma. Esto garantiza una eliminación efectiva del aliasing, preservando completamente el contenido espectral, incluso en frecuencias más altas. La alta tasa de muestreo resulta en una fidelidad mejorada y una resolución temporal y frecuencial excepcional, lo que reduce el ruido y la distorsión para una reproducción más clara y precisa en el dominio digital. Además, facilita la implementación de técnicas de procesamiento avanzadas, permitiendo un tratamiento más sofisticado de la señal con una calidad óptima, sin embargo, al ser tan elevada dicha tasa de muestreo ocasiona a su vez un mayor consumo de recursos en su procesamiento digital. 19

Puntos 4

Tras el desarrollo de esta simulación con varios valores tanto para la interpolación como la decimación de la señal coseno, ilustradas en las figuras 20, 21 y 22 , podemos concluir que, en el caso de la interpolación, esta sería el caso equivalente a la expansión en señales



continuas, ya que cuanto mayor es la interpolación aplicada en la señal muestreada más parecida es a la señal original, esto se debe a que la interpolación aumenta los puntos de muestreo de la señal, creando valores entre muestras, dando así una representación más cercana a la señal base muestreada, es decir que logra aumentar la frecuencia de muestreo de dicha señal, en el caso de la decimación, esta se comporta como la compresión en señales continuas, sin embargo en este caso como la señal digital solo trabaja con valores enteros, al momento de “comprimir” la señal, se eliminaran muestras de la señal (Por ejemplo $3/2$, no daría como resultado un entero por ende sería una muestra eliminada), haciendo así que el número de muestras de la señal se reduzca. [3][4]

Puntos 5 Partiendo del análisis del audio, podemos concluir que:

Basándonos en el análisis de las figuras 23 y 24, El bloque Multiply Const como su nombre indica es un bloque el cual nos permite multiplicar por una constante (escalar o vectorial) la señal de entrada, por lo cual para este caso específico lo que nos permite es realizar un control de volumen en la señal saliente del audio.

En el caso de la interpolación 26, se aprecia como la canción en cuestión se reproduce a una velocidad menor, causándole un efecto de ralentización, sin embargo, también se pierde algo de nitidez en la letra de la canción, por ende, se puede deducir que al aumentar el muestreo tarda más la reproducción, y al crear muestras entre estos valores ocasiona una interferencia en el sonido que presenta la canción.

En el caso de la decimación 25, se presenta el efecto opuesto a la interpolación, ya que la velocidad de reproducción de la canción aumenta. Si se obtienen los resultados esperados, en el caso de la interpolación al existir más muestras en la reproducción es normal que su velocidad de reproducción se vea reducida y al ser estas encontradas basándose en las muestras existentes y no periódicas, es normal la presencia de interferencia o ruido en la señal, en el caso de la decimación, al reducir y eliminar muestras hará que la producción de la misma sea más veloz, al tener que recorrer menos muestras que antes, sin embargo, al eliminar estas, se pierde información, reflejándose en que ciertas partes de la canción no se comprendan con todo detalle.

¿Qué pasa si se disminuye la frecuencia de muestreo (samp rate) por debajo de la frecuencia audio rate?

Cuando no se cumple con el teorema de muestreo de Nyquist se visualiza una pérdida de información, ya que existen momentos en que la canción se escucha “trabada” es decir que se detiene constantemente, como si la señal se cortara y se recuperara cada cierto tiempo, esto se

debe al efecto de aliasing ya enunciado y explicado anteriormente (Sección B1). (Un mal muestreo crea interferencias y perdidas de información debido a la sobreposición)[5]

Puntos 6 En el último ítem de la parte B del laboratorio, se realizó una variación de frecuencia para cada uno de los filtros presentes en el diagrama de flujo de la figura 46. Utilizando como muestra la canción “La bachata de Manuel Turizo”, esto con la finalidad de lograr suprimir elementos de dicha señal de audio.

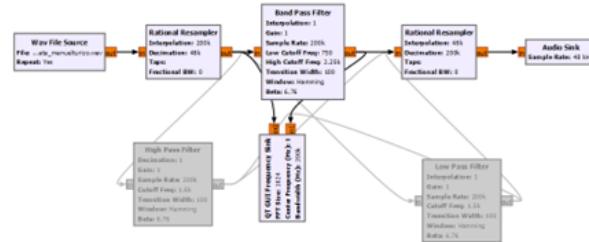


Figura 46: . Diagrama de flujo correspondiente a un ecualizador

Primeramente, se aplicó un filtro de pasa Bajas como se evidencia en la 27. En este caso solamente se varió la frecuencia de corte del filtro, llegando a estimar que a un valor de frecuencia de corte de 380[Hz] logramos suprimir la voz en la canción, dejando solamente la música de fondo, aunque se redujo significativamente la intensidad de esta, cabe aclarar que a esta señal le corresponde una frecuencia de 48[KHz] y esta siendo muestreada a una frecuencia de 200[KHz].

En relación al filtro Pasa bandas, la señal siguió con su correspondiente muestreo de 200[KHz], en este se encontró que el filtro en cuestión con la configuración de frecuencia central en 2.470[KHz] y un ancho de banda correspondiente a 2.980[KHz] se puede enfocar casi por completo a la voz del cantante, aunque se escucha el sonido de algunos instrumentos ya que están en dicho rango de frecuencias, 28.

Finalmente, en el caso del filtro pasa Altas, se trabajó con la misma frecuencia de muestreo, y variando la frecuencia de corte a 1.350[KHz] se logra eliminar parte de la instrumental, y atenuar un poco la voz del cantante, ya que se pierden algunas frecuencias bajas del tono de su voz y de la instrumental que compone a esta canción. 30.



5. Conclusiones

- El teorema de Nyquist es una herramienta fundamental al momento de muestrear señales, ya que a partir de esta podemos determinar la mínima frecuencia de muestreo con la que una señal puede ser correctamente reconstruida, por el contrario, al ignorar este teorema y trabajar con frecuencias menores nos veremos afectados por el efecto de Aliasing, fenómeno que ocasiona que la frecuencia original de dicha señal cambie, causando así una superposición de la señal muestreada al momento de ser reconstruida y por ende una representación incorrecta de la misma.
- El software GNU radio es una herramienta muy práctica y versátil para la creación y apropiación de nuevos conocimientos, ya que por medio de esta se pueden realizar análisis bastante detallados de diferentes problemáticas y/o escenarios, como los expuestos en esta práctica, ya que su amplia documentación lo convierten en una herramienta valiosa para la educación en ingeniería de comunicaciones. Permite explorar conceptos teóricos en un entorno práctico, lo que facilita la comprensión y aplicación de los principios fundamentales.
- Por medio del Software GNU radio se logró realizar el análisis correspondiente de la Decimación e interpolación en las señales, siendo estas una contraparte de la compresión y la expansión en señales continuas. Se puede concluir entonces que la interpolación aumenta los puntos de muestreo de la señal, creando valores entre muestras, dando así una representación más cercana a la señal base muestreada, es decir que logra aumentar la frecuencia de muestreo de dicha señal, por otro lado, en la decimación al trabajar con señales muestreadas al momento de “comprimir” estas señales se eliminaran varias muestras, haciendo así que el número de muestras de dicha señal se reduzca.
- Al trabajar con frecuencias elevadas de muestreo, varias veces superiores a la de Nyquist, podemos obtener una representación más acertada de la señal original, incluso logrando apreciar la forma de onda característica de la misma, es decir que a mayor numero de muestreo mejor análisis se le podrá realizar a la señal y por ende se evitara por completo el efecto de Aliasing (ya que está por encima de la frecuencia de Nyquist), sin embargo, al ser tan elevada dicha tasa de muestreo ocasiona a su vez un mayor consumo de recursos en su procesamiento digital.
- Se logro adquirir el conocimiento básico del Pro-
- grama GNU radio, aprendiendo a crear diagramas de flujo a partir de los bloques implementados en la guía y comprender el funcionamiento de estos.
- Multiply Const en GNU Radio ofrece opciones de configuración avanzadas, permitiendo un control fino sobre la constante de multiplicación. Esta característica facilita la adaptación precisa de la señal a los requisitos específicos de cada aplicación, lo que proporciona a los ingenieros un alto grado de flexibilidad y control sobre el procesamiento de señales dentro del entorno de GNU Radio. Esta capacidad de control nos permitió comprenderlo en esta práctica mediante su aplicación en la música, lo que nos permitió modificarlo según fuera necesario.
- La decimación e interpolación son técnicas fundamentales en el procesamiento de señales que permiten la reducción o expansión de la tasa de muestreo de una señal. Esto nos permitió visualizar en esta aplicación que ocasiona una ralentización y aceleración en la canción, respectivamente.
- Se logró discriminar ciertas componentes de la señal de la canción mediante un ecualizador simulado en el software GNU Radio. Esto se logró utilizando varios filtros, incluyendo filtros pasa bajas, pasa altas y pasa bandas, en los cuales se variaron la frecuencia de corte, el ancho de banda y la frecuencia central. Estos ajustes permitieron visualizar su aplicación en este campo al limitar la audición a las notas más altas, las más bajas o incluso enfocarse únicamente en la voz del cantante o en uno de los instrumentos que participaban.

Referencias

- [1] G. RADIO. Tutorial gnu radio. [Online]. Available: <https://wiki.gnuradio.org/index.php/Tutorials>
- [2] A. en Español. ¿qué es el efecto alias (aliasing) y el teorema de nyquist? [Online]. Available: <https://www.youtube.com/watch?v=NH84nQYhi-0>
- [3] SacaTuCarrera. Compresion/expansion y diezmado/interpolación en señales. [Online]. Available: https://www.youtube.com/watch?v=r0ffHMJKtl0w&ab_channel=SacaTuCarrera
- [4] U. P. de València UPV. Diezmado e interpolacion | 30/51 | upv. [Online]. Available: https://www.youtube.com/watch?v=LJ77_0D2YAk&ab_channel=UniversitatPolit%C3%A8cnicaVal%C3%A8ncia-UPV
- [5] G. Radio. Gnu radio. [Online]. Available: https://wiki.gnuradio.org/index.php/Multiply_Const