

Práctica 1: PROGRAMACIÓN EN RADIO DEFINIDA POR SOFTWARE (GNURADIO)

VALENTINA BERNAL PUENTES - 2205556
MANUEL SEBASTIAN SOLANO CANTILLO - 2171518

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones
Universidad Industrial de Santander

26 de febrero de 2023

Resumen

In this laboratory, the construction of blocks in GNU Radio using Python is explored, with a specific focus on developing accumulator and differentiator blocks. Additionally, a set of utilities has been developed in GNU Radio to perform fundamental statistical analysis, such as determining the mean, median, and other relevant measures. An essential part of this initiative involves the remote administration of a GitHub repository through the Linux command line. This establishes a solid foundation for version control management and collaboration in the evolution of blocks and functionalities, ensuring an efficient workflow.

Keywords: GNU radio, Python, GitHub.

En este laboratorio, se explora la construcción de bloques en GNU Radio utilizando Python, con un enfoque específico en la elaboración de bloques acumuladores y diferenciadores. Además, se ha desarrollado un conjunto de utilidades en GNU Radio para realizar análisis estadísticos fundamentales, como la determinación de la media, la mediana y otras medidas relevantes. Una parte esencial de esta iniciativa implica la administración remota de un repositorio en GitHub a través de la línea de comandos de Linux. Esto establece una base sólida para la gestión de versiones y la colaboración en la evolución de bloques y funcionalidades, garantizando así un flujo de trabajo eficiente.

Palabras clave: GNU radio, Python, GitHub.

1. Introducción

El empleo de GNU Radio proporciona una plataforma potente para el procesamiento de señales y comunicaciones, posibilitando la creación de bloques personalizados que se ajusten a una variedad de aplicaciones, en este caso a manejar será con la estadística. Al integrar esta herramienta con la gestión remota de GitHub, se puede

brindar una colaboración en proyectos de desarrollo de software haciendo que este sea un aprendizaje conjunto.

2. Objetivos

■ General

Hacer que el estudiante se familiarice y cree competencias con el programa GNUradio.

■ Específicos

- Conocer el entorno del software GNUradio para la creación de bloques que permitan realizar operaciones estadísticas.
- Implementar los bloques acumulador y diferenciador para comprender como su funcionamiento afecta el análisis de una señal.

3. Marco teorico

GNU Radio es una herramienta de desarrollo de software de código abierto para la implementación de radios definidas por software (SDR). Proporciona bloques de procesamiento de señales que se pueden conectar para crear sistemas de procesamiento de señales complejos. Es compatible con una variedad de hardware SDR y ofrece enlaces Python para facilitar el desarrollo de aplicaciones de procesamiento de señales. Es ampliamente utilizado en investigación, radioaficionados, redes inalámbricas y más. Cuenta con repositorio en GitHub donde se encuentra el código fuente y la comunidad contribuye con mejoras, correcciones de errores y proyectos relacionados con el desarrollo de radios definidas

por software.[1]

Se tiene que mediante el enlace a python se puede programar los siguientes bloques:

- **Bloque acumulador:** Es un componente que suma secuencialmente las muestras de entrada a medida que fluyen a través de él, acumulando así un valor total a lo largo del tiempo.
- **Bloque diferenciador:** Es un componente que calcula la diferencia entre muestras consecutivas de entrada, lo que proporciona una salida que representa la tasa de cambio de la señal de entrada en el dominio del tiempo.

A la hora de calcular los promedios de tiempo para una señal tenemos las siguientes formulas:

- La media de una señal $x(t)$:

$$X_m = \langle x(t) \rangle$$

- La media cuadrática de una señal $x(t)$ es:

$$X_c = \langle x^2(t) \rangle$$

- La Varianza:

$$\sigma_x^2 = \langle [x(t) - X_m]^2 \rangle$$

- Desviación Estándar:

$$\sigma_x = \sqrt{\langle [x(t) - X_m]^2 \rangle}$$

- El valor RMS de una señal $x(t)$:

$$X_{RMS} = \sqrt{\langle [x(t)]^2 \rangle}$$

- La potencia promedio normalizada de una señal:

$$P = X_{RMS}^2 = \langle [x(t)]^2 \rangle$$

4. Procedimiento

Para realizar el laboratorio primero descargamos el libro guía [2], en el cual se encuentran los ejemplos de los codigos en python para realizar los bloques de e Diff (bloque diferenciador) y e Acum (bloque acumulador), tal como se muestra en la figura 1 a demas de el bloque (Pormediado de tiempos) que se encuentra en el texto guía pagina 20.

```
import numpy as np
from gnuradio import gr

class blk(gr.sync_block):
    def __init__(self):
        gr.sync_block.__init__(
            self,
            name='e_diff',
            in_sig=[np.float32],
            out_sig=[np.float32]
        )

    def work(self, input_items, output_items):
        x = input_items[0]
        y0 = output_items[0]
        y0[:] = np.zeros(len(x))
        return len(y)
```

ACUMULADOR

```
import numpy as np
from gnuradio import gr

class blk(gr.sync_block):
    def __init__(self):
        gr.sync_block.__init__(
            self,
            name='e_acum',
            in_sig=[np.float32],
            out_sig=[np.float32]
        )

    def work(self, input_items, output_items):
        x = input_items[0]
        y0 = output_items[0]
        y0[:] = x + y0
        return len(y)
```

ACUMULADOR

```
import numpy as np
from gnuradio import gr

class blk(gr.sync_block):
    def __init__(self):
        gr.sync_block.__init__(
            self,
            name='e_diff',
            in_sig=[np.float32],
            out_sig=[np.float32]
        )

    def work(self, input_items, output_items):
        x = input_items[0]
        y0 = output_items[0]
        y0[:] = x - y0
        return len(y)
```

DIFFERENCIADOR

Fig. 1: bloques de diferenciacion y acumulacion

Tal y como nos indica el taller, crearemos un diagrama de bloques que permitira calcular los parametros anteriormente mencionados a partir de la adición de una señal de ruido gaussiana con el fin de observar el comportamiento y ver como puede afectar una señal de entrada sinusoidal como se mostrará en la figura 2.

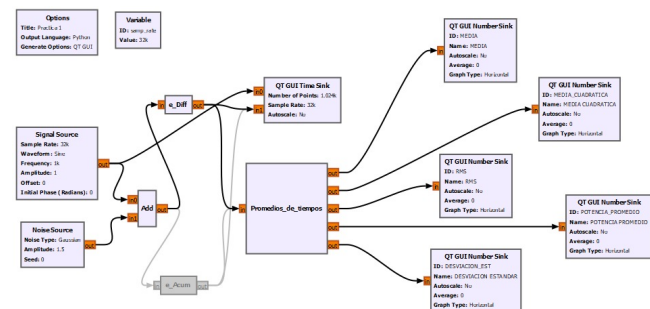


Fig. 2: Diagrama de bloques GNURadio

Final mente procedemos a correr nuestro programa con el fin de evaluar las señales cuando se utiliza un bloque acumulador 3,4 y al utilizar un bloque diferenciador 5,6.

5. Análisis de resultados

al realizar la simulación usando el bloque acumulador, podemos observar que La acumulación de la señal sinusoidal con ruido gaussiano produce una señal con una amplitud mayor que la señal sinusoidal original debido

a una suma de amplitudes entre las 2 señales, a demás que la forma de la señal acumulada dependerá de la frecuencia y la amplitud de la señal sinusoidal, así como de la amplitud y la desviación estándar del ruido gaussiano 3,4.

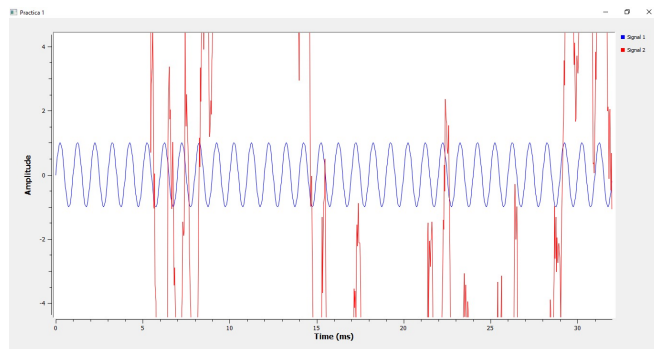


Fig. 3: señal con bloque acumulador

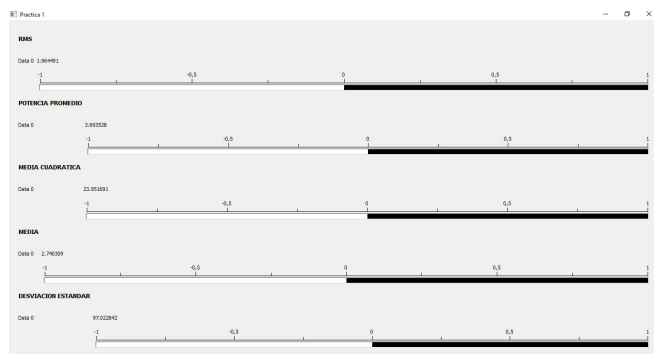


Fig. 4: parámetros señal con bloque acumulador

al realizar la simulación usando el bloque diferenciador, podemos observar que La diferenciación de la señal sinusoidal con ruido gaussiano produce una señal con una amplitud mucho mayor que la señal sinusoidal original debido a una suma de amplitudes entre las 2 señales ya que la amplitud del ruido gaussiano se ve amplificada debido a la diferenciación del mismo 5,6.

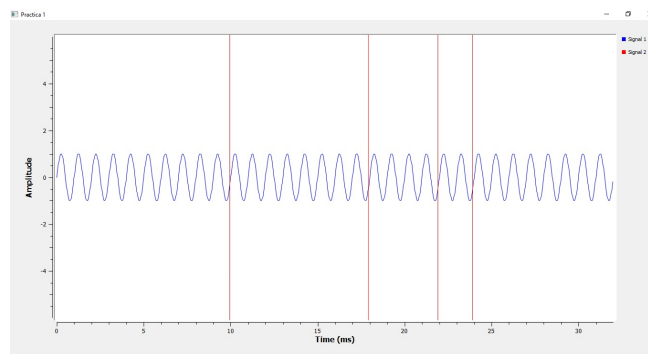


Fig. 5: señal con bloque diferenciador

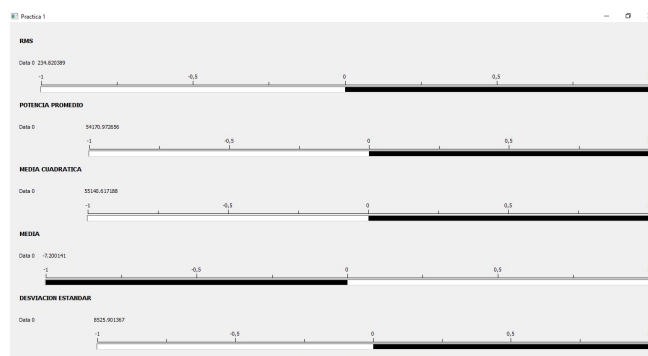


Fig. 6: parámetros con bloque diferenciador

6. Conclusiones

- El bloque acumulador en GNURadio es crucial en el ámbito del procesamiento de señales y las comunicaciones, ya que posibilita la acumulación progresiva de valores a lo largo del tiempo. Esta funcionalidad es fundamental para diversas aplicaciones, desde la integración de señales hasta el cálculo de promedios móviles. Esto resalta la versatilidad del bloque y su relevancia en diversos ambitos.
- La creación de un bloque que muestre estadísticas de promedios temporales en señales continuas y discretas dentro de GNURadio ofrece una herramienta indispensable y flexible para el análisis de señales en tiempo real. Esta capacidad proporciona una comprensión más detallada de las tendencias y comportamientos de las señales, lo cual es crucial en diversas áreas como las telecomunicaciones, las finanzas y la ingeniería. Este tipo de herramienta facilita la toma de decisiones fundamentadas y la optimización de sistemas, lo que la convierte en



una valiosa contribución para la investigación y el desarrollo en una variedad de campos.

- El funcionamiento del bloque acumulador permite integrar las señales de entrada y crear un efecto de filtro pasa bajas para mejorar la relación señal a ruido (SNR).
- El funcionamiento del bloque difereciador permite derivar las señales de entrada y crear un efecto de filtro pasa altas con el fin de detectar cambios abruptos en la señal.

Referencias

- [1] “Sistemas de comunicaciones utilizando gnu radio,” 2015. [Online]. Available: <https://www.boletin.upiita.ipn.mx/index.php/ciencia/639-cyt-numero-51/1191-sistemas-de-comunicaciones-utilizando-gnu-radio>
- [2] O. B. y O. M. Reyes Torres, “Comunicaciones digitales basadas en sdr,” 2019. [Online]. Available: <https://sites.google.com/saber.uis.edu.co/comdig>