

Laboratorio 2: Análisis de la Densidad Espectral de Potencia (PSD) en Señales Aleatorias mediante (GNU Radio) Implementación y Caracterización de Señales Binarias, Ruido Blanco y Fuentes Reales

Carlos Stiven Roa Martinez
Universidad Industrial de Santander
Bucaramanga, Colombia
Email: carlos2201948@correo.uis.edu.co

Dana Cotes Cala
Universidad Industrial de Santander
Bucaramanga, Colombia
Email: dana2212256@correo.uis.edu.co

Abstract—This laboratory focused on the analysis of the Power Spectral Density (PSD) of bipolar binary signals, white noise, and real sources (image and audio), using GNU Radio as the experimental platform. The effects of the *Samples per Symbol* (SPS) parameter on spectral shape, bandwidth, and sampling frequency were studied. In addition, the practical limitations of the implementation were contrasted with theoretical models, highlighting the difference between deterministic and random signals.

Index Terms—GNU Radio, Power Spectral Density, Binary Signals, White Noise, Real Sources, Samples per Symbol, Software Defined Radio.

Repositorio: https://github.com/DamaGit/com2_A1_G-9/tree/44d30ff503f0c9157c982b4a54e9ab20a215efc5/Pr%C3%A1ctica_2

I. INTRODUCCIÓN

Este laboratorio se centró en la exploración de conceptos fundamentales relacionados con la programación en radio definida por software (SDR) utilizando GNU Radio. Se estudiaron señales binarias, ruido blanco y fuentes de audio reales, con el fin de comprender sus características en el dominio del tiempo y la frecuencia. Asimismo, se aplicaron técnicas de procesamiento digital de señales para analizar la densidad espectral de potencia (PSD) y modulaciones digitales.

II. OBJETIVOS

- Caracterizar la Densidad Espectral de Potencia (PSD) de señales binarias bipolares, ruido blanco y fuentes reales (imagen y audio), evaluando el impacto de parámetros como Samples per Symbol (SPS) en el ancho de banda, resolución espectral y forma de los lóbulos del espectro.
- Validar teóricamente las limitaciones prácticas en el análisis espectral de señales con ancho de banda teóricamente infinito (como ruido blanco o señales binarias rectangulares), considerando las restricciones de muestreo, filtrado y procesamiento en GNU Radio.

III. METODOLOGÍA

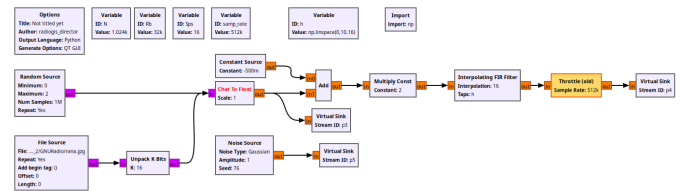


Fig. 1: Flujograma GNU Radio utilizado

Se utilizó el flujograma `randombinayrectsignal.grc` en GNU Radio para:

- Generar señales binarias bipolares con $SPS = 1, 4, 8, 16$.
- Analizar ruido blanco en tiempo y frecuencia.
- Procesar señales de fuentes reales: imagen (`rana.jpg`) y audio (`sonido.wav`).

A. Señal Binaria Bipolar

En el análisis de la figura 2 se observó el comportamiento de la Densidad Espectral de Potencia (PSD) de una señal binaria bipolar con forma rectangular para diferentes valores de *Samples per Symbol* (SPS). Los resultados se obtuvieron mediante simulaciones en GNU Radio.

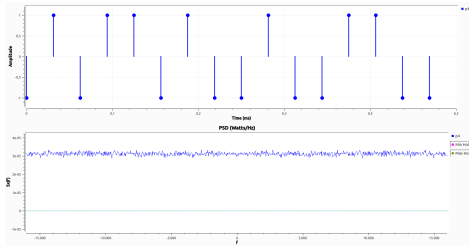
1) Relación entre SPS y la Forma de la PSD

La PSD de una señal binaria bipolar sigue una función $\text{sinc}^2(f)$, característica de pulsos rectangulares. Se observa que:

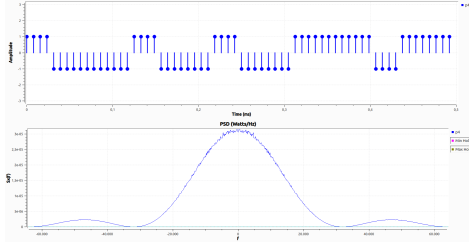
- Al **aumentar el SPS**, el lóbulo principal se estrecha y los lóbulos laterales se acercan entre sí.
- La energía se concentra en frecuencias más bajas con SPS altos.
- La estructura espectral se vuelve más definida con mayores valores de SPS.

2) Ancho de Banda y Frecuencia de Muestreo

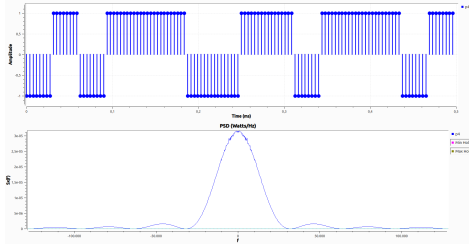
- El **ancho de banda (BW)** se mantiene constante en aproximadamente 32 kHz para todos los valores de SPS,



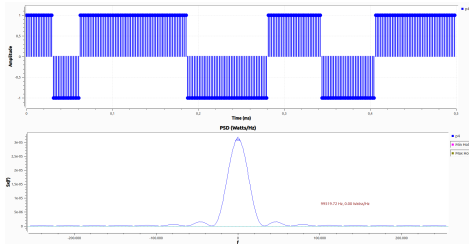
(a) PSD para SPS = 1.



(b) PSD para SPS = 4.



(c) PSD para SPS = 8.



(d) PSD para SPS = 16.

Fig. 2: PSD de una señal binaria bipolar para diferentes valores de SPS.

dado por:

$$BW \approx R_b = 32 \text{ kbps}$$

- La **frecuencia de muestreo** (F_s) aumenta proporcionalmente al SPS:

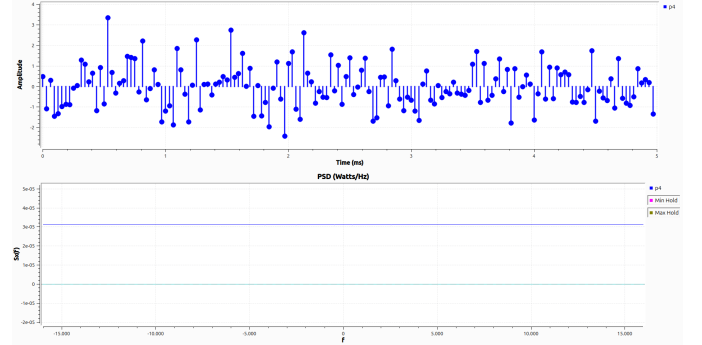
$$F_s = R_b \times \text{SPS}$$

SPS	R_b (kbps)	F_s (kHz)	BW (kHz)
1	32	32	32
4	32	128	32
8	32	256	32
16	32	512	32

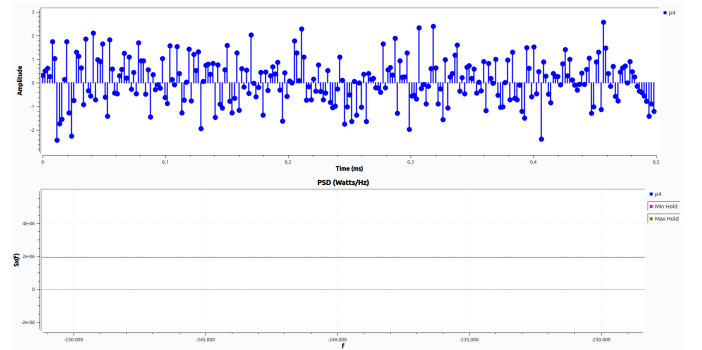
TABLE I: Parámetros para diferentes valores de SPS

- SPS=1:** PSD sin lóbulos laterales (apariencia similar al ruido blanco), efecto de un mal muestreo de la señal.
- SPS=4:** Mejor definición de lóbulos, pero aún con solapamiento.
- SPS=8:** Lóbulos bien diferenciados, típico de señales binarias.
- SPS=16:** Máxima resolución espectral con estructura sinc^2 claramente visible.

B. Ruido Blanco



(a) PSD para SPS = 1.



(b) PSD para SPS = 16.

Fig. 3: PSD de una señal WNG diferentes valores de SPS.

En la Figura 3 se presenta el análisis de la Densidad Espectral de Potencia (PSD) de una señal de ruido gaussiano blanco agregado para diferentes valores de *Samples per Symbol* (SPS). Este estudio permite comprender cómo el ruido afecta esta presente en sistemas de comunicación reales.

El ruido gaussiano blanco presenta una PSD teóricamente plana en todas las frecuencias.

- La estructura fundamental de la señal ($\text{sinc}^2(f)$) se mantiene, pero se superpone una componente constante de ruido Figura 3 (a).
- Para **SPS bajos (ej: SPS=1)**, la PSD aparece casi plana, con el ruido enmascarando la estructura de la señal.
- Para **SPS altos (ej: SPS=16)**, la señal permanece constante a pesar de las Variaciones en el tamaño de SPS, indicando que sin importar el tamaño de SPS la señal siempre se mantendrá constante.

C. Fuentes Reales

Al momento de analizar fuentes reales, nos referimos a estímulos del común, en este caso imágenes y sonidos que aunque físicamente se perciben diferentes al momento de transmitir información se evidencia que se pueden descomponer en un sistema vectorial que se acomoda a los medios de transmisión y permite el envío de este tipo de información a partir del espectro.

IV. PREGUNTAS PLANTEADAS EN LA GUÍA

A. Análisis General del Flujograma

- 1) ¿Qué papel juega la siguiente combinación de bloques?
Constant Source → Multiply Const
Estos bloques del flujograma integran la generación, modulación y visualización de la señal binaria aleatoria, permitiendo transformar una secuencia digital en una señal en forma rectangular para su análisis tanto en el dominio del tiempo como en el espectral

B. Bloque Interpolation FIR Filter

- 2) ¿Qué papel juega el bloque Interpolation FIR Filter y cómo funciona?
El bloque Interpolation FIR Filter realiza un proceso de interpolación que aumenta la tasa de muestreo de la señal mediante la inserción de nuevas muestras entre las muestras originales. Este proceso se implementa mediante un filtro FIR (Finite Impulse Response) que cumple dos funciones principales, Aumento de la tasa de muestreo y Filtrado anti-aliasing
- 3) ¿Por qué el parámetro Interpolation vale SPS y qué pasa si se coloca otro valor?
Se utiliza el valor "SPS" en el parámetro de interpolación para mantener la coherencia en la cantidad de muestras por símbolo y la forma original de la señal; si se asigna un valor distinto, se altera la escala temporal y puede llegar a distorsionar la forma de la PSD, lo que dificultaría el análisis.
- 4) Si tuviese que analizar la señal en p3, ¿qué cambios realizaría en la instrumentación?
Tendríamos cambiar el nombre de P4 por P3 en el virtual source lo cual pasaría los Valores del virtual sink a una virtual source pasando la señal P3 por todo el esquema de tratamiento usado anteriormente en P4 y P5.
- 5) ¿Qué fórmula permite conocer el ancho de banda en p4 si se conoce R_b y SPS?
El ancho de banda de la señal en p4 se calcula con la siguiente fórmula:

$$BW_{p4} = \frac{R_b \times SPS}{2}$$

- 6) ¿Qué fórmula permite conocer F_s en p3 si se conoce F_s en p4 y SPS?
La frecuencia de muestreo en p3 se calcula con la fórmula:

$$f_{sp3} = f_{sp4} \times SPS$$

- 7) ¿Por qué la PSD de señales de audio es diferente a la de imágenes siendo ambas bipolares?

Aunque ambas señales sean binarias bipolares de forma rectangular, sus densidades espectrales de potencia (PSD) difieren debido a las características inherentes de cada una. La señal de audio es continua y varía dentro del rango audible (20 Hz a 20 kHz), mientras que una imagen es bidimensional y esta compuesta por frecuencias espaciales. Esta diferencia en la naturaleza de las señales da lugar a PSD distintas para cada una.

- 8) ¿Qué papel juega el bloque Throttle?

El bloque "Throttle" regula la tasa de muestreo de la señal para limitar el procesamiento y evitar un consumo excesivo de recursos computacionales. Su funcionamiento depende de la tasa de bits (R_b) y el número de símbolos (SPS), asegurando un procesamiento eficiente sin sobrecargar el sistema.

- 9) ¿Qué pasaría con la PSD si no se hace conversión a bipolar (valores 0/1 en lugar de -1/1)?

Si la señal oscila entre 0 y 1, se considera una señal unipolar, lo que resulta en una componente continua DC distinta de cero. Esto hace que la potencia de la señal se concentre principalmente en la frecuencia de 0 Hz, en lugar de distribuirse uniformemente a través de las diferentes frecuencias.

- 10) ¿Coincide el ancho de banda infinito del ruido blanco con lo observado en GNU Radio? ¿Por qué?

En la práctica, el ancho de banda está determinado por las limitaciones del sistema, lo que provoca que la densidad espectral de potencia (PSD) se mantenga constante dentro de un rango específico. Esto se debe a que el sistema no puede procesar frecuencias fuera de su capacidad, lo que restringe la distribución de la potencia a un intervalo determinado.

C. Señal Binaria Rectangular

- 11) ¿Coincide el ancho de banda infinito teórico con lo observado en GNU Radio? ¿Por qué?

Aunque una señal binaria aleatoria de forma rectangular tiene un ancho de banda teóricamente infinito, esto no se refleja en GNU Radio. En la práctica, el ancho de banda está limitado por la frecuencia de muestreo del sistema y las restricciones del software. Como resultado, la densidad espectral de potencia (PSD) de la señal muestra un ancho de banda finito, que agrupa la mayor parte de la potencia en un rango específico. Este efecto se vuelve más evidente cuando los valores del vector h no son todos iguales a 1.

D. Cálculos Espectrales

- 12) ¿Qué fórmula ayuda a calcular el número de lóbulos de la PSD conociendo F_s y SPS?

Para determinar el número de lóbulos de la PSD de una señal binaria aleatoria de forma rectangular, primero se debe calcular la tasa de símbolos (R_s), utilizando la tasa de bits (R_b), la frecuencia de muestreo (f_s) y el valor de

SP S de la siguiente manera:

$$R_s = \frac{f_s}{S_{ps}}$$

Luego, se utiliza la formula siguiente para calcular el numero de lóbulos, incluyendo el lóbulo central:

$$N_{\text{lóbulos}} = \frac{f_s}{R_s}$$

- 13) ¿Cómo se calcula el rango de frecuencias del espectro conociendo R_b y SPS ?

Dado R_b y SPS , la frecuencia de muestreo f_s se obtiene multiplicando la tasa de bits por el numero de muestras por símbolo, de la siguiente manera:

$$f_s = R_b \times SPS$$

El rango de frecuencias sera la mitad de la frecuencia de muestreo. Por lo tanto, el espectro ocupara un rango de frecuencias de 0 kHz a 128 kHz.

- 14) ¿Cómo se calcula la resolución espectral del analizador conociendo N y F_s ?

La resolución espectral (RE) de un analizador de espectros se puede determinar si se conoce la frecuencia de muestreo (f_s) y el numero de puntos en la FFT (N), utilizando la siguiente ecuación:

$$RE = \frac{f_s}{N}$$

E. Bloques de Procesamiento

- 15) ¿Qué pasa si en Unpack K Bits se configura $K = 16$?

Al aumentar el parametro K a 16 en el bloque "Unpack K Bits", se procesan 16 bits por cada byte de entrada. Esto genera grupos de 16 bits, lo que se refleja en la PSD al aparecer impulsos distribuidos a lo largo del espectro. Además, se observan armónicos y una mayor distribución de la energía, lo cual incrementa el ancho de banda.

- 16) ¿Cómo calcular F_s a la entrada de Unpack K Bits conociendo lóbulos de PSD y BW?

Para determinar la frecuencia de muestreo del bloque "Unpack K Bits", se toma el ancho de banda de la señal (BW) junto con el numero de lóbulos (N) en la PSD. La frecuencia de muestreo necesaria para capturar todos los lóbulos en la PSD se calcula mediante la siguiente formula:

$$f_s = 2 \times BW \times N$$

- 17) ¿Cómo calcular F_s a la salida de Unpack K Bits conociendo F_s de entrada?

Si se conoce la frecuencia de muestreo de la entrada (f_{in}), se puede calcular la frecuencia de salida (f_{out}) utilizando la relación:

$$f_{out} = f_{in} \times K$$

- 18) ¿Cómo calcular F_s a la salida de Char to Float conociendo F_s de entrada?

El bloque "Char to float" no cambia la frecuencia de muestreo, sino que ajusta la resolución de la entrada. Al convertir los caracteres en flotantes, la salida mantiene la misma frecuencia de muestreo, pero con una mayor precisión. Esto permite representar los datos con mayor exactitud en las etapas siguientes del sistema.

F. Análisis Comparativo

- 19) ¿Para qué valor de SPS la PSD de señal bipolar es similar al ruido blanco?

Cuando el valor de SPS es igual a 1, la PSD presenta una distribución plana, lo que indica que la energía de la señal se distribuye de manera uniforme a lo largo de todas las frecuencias.

G. Modificaciones del Flujoograma

- 20) ¿Qué cambios hacer para forma de dientes de sierra?

Para sintetizar la señal diente de sierra, se modifica únicamente el vector h , asignándole los valores $\{0, 0.2, 0.5, 0.75, 1\}$. Con ello se obtiene un tramo lineal de pendiente 1/4, lo que permite visualizar adecuadamente la forma diente de sierra. El único inconveniente observado es el cambio de bit, que introduce una ligera irregularidad en la señal, pues en ese punto se toman dos muestras idénticas.

- 21) ¿Qué cambios hacer para codificación Unipolar RZ?

Para generar una señal Unipolar RZ, ajuste la constante del bloque Constant Source a cero, eliminando así el offset y produciendo una señal unipolar. Luego, modifique la variable h de modo que la mitad de sus componentes sean 1 y la otra mitad sean 0. Por ejemplo, con $h = [1, 1, 1, 1, 0, 0, 0, 0]$. Al incrementar el número de valores en esta secuencia, la precisión de la codificación Unipolar RZ aumentará.

- 22) ¿Qué cambios hacer para codificación Manchester NRZ?

En el flujoograma, para generar una señal binaria Manchester NRZ, se ajusta el filtro FIR para permitir transiciones rápidas entre los niveles -1 y 1.

- 23) ¿Qué cambios hacer para forma de señal OOK?

Para la modulación OOK, se ajustó el vector h para generar una forma de señal senoidal, repitiendo este conjunto de datos cuatro veces. Como resultado, en la PSD de la señal de salida se observa un desplazamiento del contenido espectral hacia frecuencias más altas. Cabe aclarar que, a medida que aumenta la cantidad de muestras en h , se obtiene una mejor representación de la señal.

- 24) ¿Qué cambios hacer para forma de señal BPSK?

Para la modulación BPSK, se mantuvo el vector h para sintetizar la portadora senoidal y se ajustó el bloque Constant Source para que la señal resultante fuera claramente visible, en particular al observar la inversión de fase (π) cuando el bit de datos cambia entre 0 y 1. Este ajuste garantiza que las transiciones entre símbolos modulados se distingan con nitidez, facilitando su análisis e interpretación.

- 25) ¿Qué cambios hacer para forma de señal ASK?
Para obtener la señal ASK, se conservó el vector h empleado para generar la señal senoidal y se ajustaron los bloques `Constant Source` y `Multiply Constant` con el fin de imponer amplitudes distintas cuando la entrada es 1 y cuando es 0. De este modo, la portadora mantiene su forma senoidal y su amplitud conmuta según el bit de datos, implementando la modulación por desplazamiento de amplitud (ASK).
- 26) ¿Qué cambios hacer para forma de latidos del corazón?
Se modifica el vector h de tal forma que se construya punto a punto la señal de latido de corazón.
- 27) ¿Qué cambios hacer para forma personalizada?
Se construye un vector con puntos detallados con el fin de construir la curvatura en la señal.

V. CONCLUSIONES

- 1) Contenido espectral por tipo de señal. Las señales de audio concentraron su PSD en bandas bajas y medias, coherente con la naturaleza de la voz/sonidos; en contraste, las señales de imagen codificadas con $K = 8$ exhibieron un espectro más disperso debido a sus transiciones rápidas y mayor complejidad frecuencial.
- 2) Efecto de SPS en tiempo y espectro. Al incrementar las muestras por símbolo (SPS), se mejoró la resolución temporal y la calidad de la forma de pulso (transiciones más suaves), reduciendo el contenido de alta frecuencia por símbolo; sin embargo, al aumentar $f_s = R_b \cdot \text{SPS}$ se ensanchó el ancho de banda observado y apareció un mayor número de lóbulos en la PSD .
- 3) Sensibilidad a parámetros del sistema. Cambios en SPS, tasa de bits R_b y el factor de interpolación h redistribuyeron la energía espectral de la señal. La inclusión de portadora y la selección de modulación (p.ej., OOK, BPSK) desplazaron y moldearon el espectro de acuerdo con la técnica empleada.
- 4) Codificaciones de línea: Unipolar RZ vs. Manchester NRZ. La señal *Unipolar RZ* concentró potencia en bajas frecuencias (cerca de 0 Hz), mientras que *Manchester NRZ* presentó media nula y mayor concentración en frecuencias medias, mitigando la componente en DC y facilitando el acoplamiento AC.
- 5) Ruido blanco. El ruido blanco presentó PSD aproximadamente constante en todo el rango de frecuencias, a diferencia de las señales determinísticas o codificadas cuya PSD se concentró en bandas específicas.
- 6) Limitaciones prácticas del entorno. Aunque una secuencia binaria rectangular ideal tuvo ancho de banda teóricamente infinito, en la práctica (p.ej., GNU Radio) el sistema/software limitó el ancho de banda utilizable, por lo que la PSD se confinó a un rango finito. Estas restricciones se acentuaron cuando los valores del vector h difirieron de la unidad.
- 7) Control de tasa de muestreo. El bloque `Throttle` reguló f_s para evitar un uso excesivo de recursos computacionales. Su ajuste fue consistente con R_b y SPS,

garantizando un procesamiento eficiente sin sobrecargar el sistema.

REFERENCES

- [1] H. Ortega, B. Bodá, O. Mauricio, and R. Torres, "Comunicaciones digitales basadas en radio definida por software," [Online]. Available: <https://sites.google.com/saber.uis.edu.co/comdig>.
- [2] "Generación y análisis de señales en GNU Radio," YouTube, [Online]. Available: <https://www.youtube.com/watch?v=qfOhnqlhmas>. [Accessed: Sep. 12, 2025].