

## Práctica 2: PSD DE SEÑALES ALEATORIAS (GNURADIO)

Adriana Patricia Rodriguez Velandia - 2185524  
Jesse Solomon Blair Camacho - 2191740  
Juan Jose Morales Hernandez - 2191572

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

18 de Marzo de 2024

### Resumen

En este documento, utilizamos el procesamiento de señales en GNU Radio utilizando Python, desarrollando bloques para manejar señales binarias aleatorias bipolares y análisis de ruido blanco. También se ha realizado en GNU Radio para hallar la frecuencia de muestreo, ancho de banda y la tasa de bits. El repositorio relacionado con este laboratorio esta en el siguiente link [https://github.com/adriprv/Comu2\\_B2/tree/Practica\\_2](https://github.com/adriprv/Comu2_B2/tree/Practica_2)

**Palabras clave:** Señales binarias aleatorias bipolares, análisis de ruido blanco

### 1. Introducción

Utilizando GNU Radio para el procesamiento de señales, nos permite colocar bloques y personalizarlos para obtener las diferentes operaciones que se necesitan, con GNU Radio y utilizar GitHub se puede utilizar el trabajo en grupo, colaborando en proyectos de desarrollo.

### 2. Procedimiento

1. Nueva rama practica 2 (Hecho en Git Hub).
2. Descarga de la nueva practica en la estación de cómputo (con git pull).
3. Creación de la nueva carpeta de la practica 2 en la rama.
4. Comprobación del diagrama de flujo propuesto para la practica.

4.1. Para una señal binaria aleatoria bipolar obtuvimos la forma en el tiempo, la PSD y los parámetros principales (rata de bits, frecuencia de muestreo, ancho de banda) para los siguientes valores de Sps.

- $Sps = 4$
- $Bits(Rb) = 32Kbps$
- Frecuencia de muestreo( $F_s$ )= $Rb \cdot Sps = 128KHz$   
Con el criterio de muestreo de Nyquist el ancho de banda en el canal es:  $f_s/2 = 64KHz$

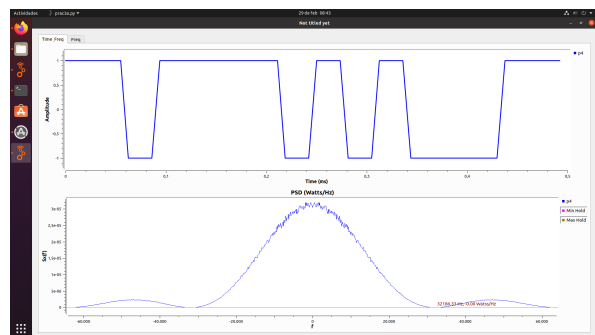


Fig. 1: Señal binaria aleatoria Sps=4

En la reciente figura se puede observar el resultado de la forma en el tiempo, en frecuencia y la PSD de la señal binaria aleatoria con un Sps = 4, es decir que la señal es interpolada 4 veces. Con este valor de Sps se obtuvo los siguientes parámetros:

- $Sps = 8$
- $Bits(Rb) = 32Kbps$

- Frecuencia de muestreo( $F_s$ )= $R_b \cdot S_p_s = 256\text{KHz}$   
Con el criterio de muestreo de Nyquist el ancho de banda en el canal es:  $f_s/2 = 128\text{KHz}$

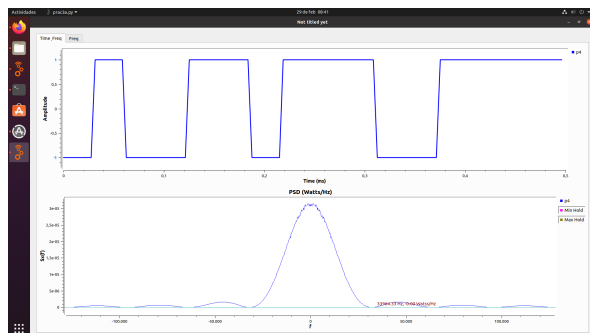


Fig. 2: Señal binaria aleatoria  $S_p_s=8$

En la reciente figura se puede observar el resultado de la forma en el tiempo, en frecuencia y la PSD de la señal binaria aleatoria con un  $S_p_s = 18$ , es decir que la señal es interpolada 18 veces. Con este valor de  $S_p_s$  se obtuvo los siguientes parámetros:

- $S_p_s = 16$
- Bits( $R_b$ )=32Kbps
- Frecuencia de muestreo( $F_s$ )= $R_b \cdot S_p_s = 512\text{KHz}$   
Con el criterio de muestreo de Nyquist el ancho de banda en el canal es:  $f_s/2 = 256\text{KHz}$

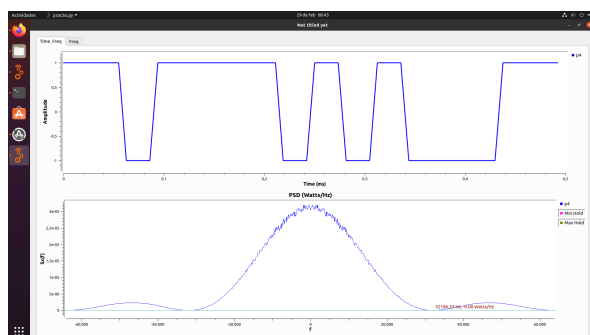


Fig. 3: Señal binaria aleatoria  $S_p_s=16$

En la reciente figura se puede observar el resultado de la forma en el tiempo, en frecuencia y la PSD de la señal binaria aleatoria con un  $S_p_s = 16$ , es decir que la señal es interpolada 16 veces. Con este valor de  $S_p_s$  se obtuvo los siguientes parámetros:

- $S_p_s = 1$
- Bits( $R_b$ )=32Kbps
- Frecuencia de muestreo( $F_s$ )= $R_b \cdot S_p_s = 32\text{KHz}$
- Con el criterio de muestreo de Nyquist el ancho de banda en el canal es:  $f_s/2 = 16\text{KHz}$

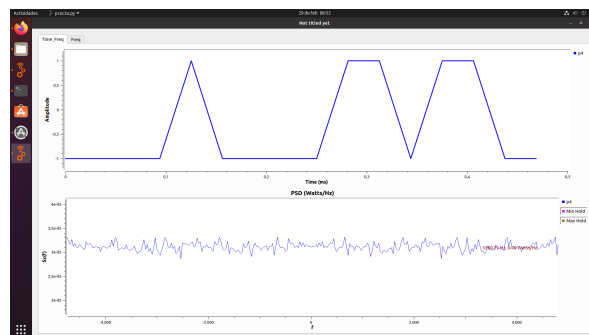


Fig. 4: Señal binaria aleatoria  $S_p_s=1$

En la reciente figura se puede observar el resultado de la forma en el tiempo, en frecuencia y la PSD de la señal binaria aleatoria con un  $S_p_s = 1$ , es decir que la señal es interpolada 1 vez. Con este valor de  $S_p_s$  se obtuvo los siguientes parámetros:

## 5. Comprobación ruido blanco en tiempo y PSD.

5.1. Para este proceso configuramos las "virtual source" del diagrama de flujo intercambiando el nombre de los bloques p4 y p5 entre ellos, además realizamos distintas variaciones en algunos parámetros para observar las diferencias con respecto al resultado del ruido blanco y obtuvimos los siguientes resultados:

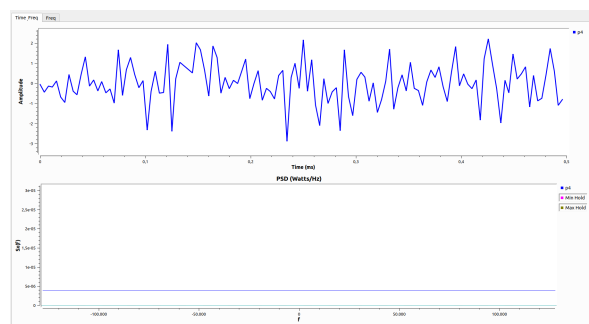


Fig. 5: Ruido Blanco  $S_p_s = 16$

En la anterior imagen se observa el ruido blanco obtenido con un valor determinado de  $S_p_s$ , se puede observar como el ruido blanco impacta la señal

debido a este parámetro, el cual al ser grande el valor de este de dicho parámetro aumenta la susceptibilidad al ruido.

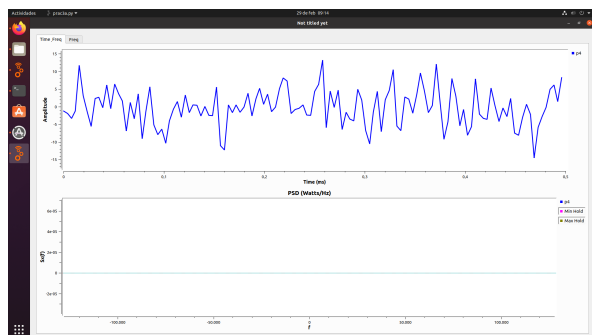


Fig. 6: Ruido Blanco Sps=4

En la anterior imagen se observa el ruido blanco obtenido con un valor determinado de Sps = 4, se puede observar que respecto a la figura anterior (Fig 5), en este caso por ser un valor de Sps menor proporciona una mayor robustez contra el ruido obtenido de la señal binaria.

6. Señal en tiempo y frecuencia de fuente de mundo real (cámara fotográfica).

6.1. Para este proceso configuramos y editamos el diagrama de flujo requerido para esta parte de la practica, en donde configuramos algunos parámetros para leer un archivo de imagen digital .jpg (rana.jpg) e hicimos algunos experimentos variando algunos de los parámetros del diagrama de flujo y obtuvimos los siguientes resultados:

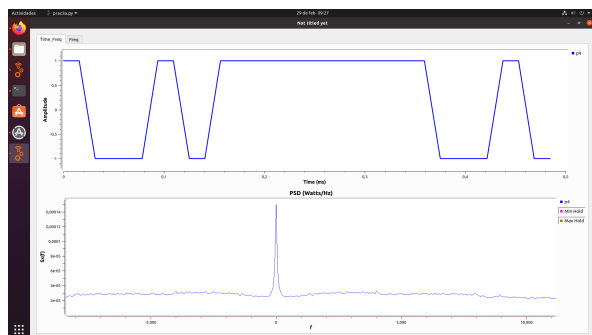


Fig. 7: Señal resultante Sps=2

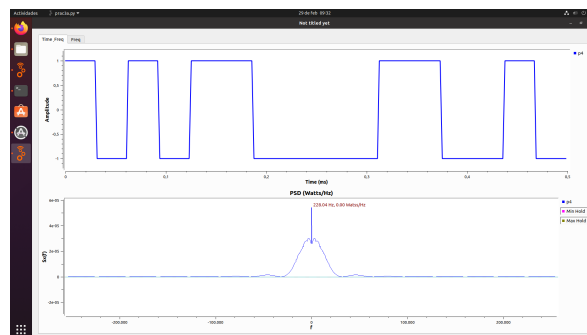


Fig. 8: Señal resultante Sps=16

7. Señal en tiempo y frecuencia de fuente de mundo real (Micrófono).

7.1. Para este proceso realizamos los respectivos cambios requeridos en el diagrama de flujo estipulados en la practica, en donde configuramos un parámetro para que leyera un un archivo de audio .wav (sonido.wav). Con este archivo de audio hicimos algunos experimentos y realizamos algunas variaciones de parámetros del diagrama de flujo para observar lo que pasaba con la señal en tiempo y en frecuencia, y obtuvimos lo siguiente.

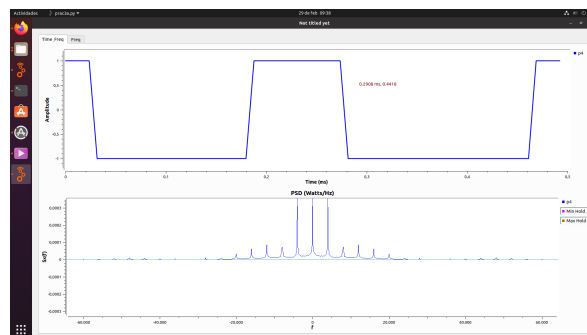


Fig. 9: Señal resultante Sps=4

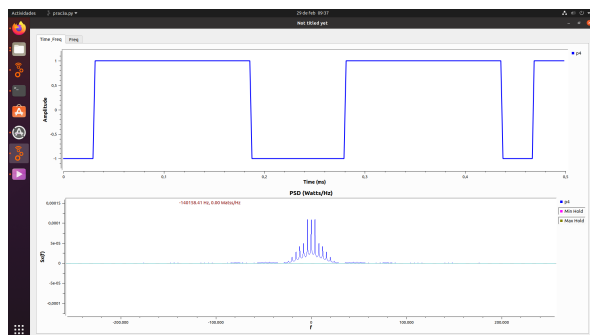


Fig. 10: Señal resultante Sps=16

## 8. Preguntas de control.

### 8.1. Preguntas de auto control sobre el flujograma randombinayrectsinal.grc:

#### a. Se encuentran tres bloques:

-Constant Source(Fuente Constante)

Constant=-500m Este bloque genera una señal constante de potencia con valor de -500mw

-Add

in1 Toma la señal de entrada generada por la fuente constante y le permite sumar una señal adicional.

-Multiply Constant

constant=2 El bloque multiplica la señal resultante de la suma por un valor constante de 2.

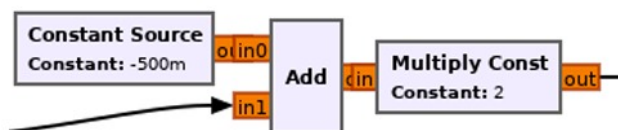


Fig. 11: Combinación de bloques de Gnu Radio

#### b. ¿Qué papel juega el bloque interpolationg FIR Filter", ¿Como funciona?

1. Teniendo en cuenta que la interpolación es usado para saber cuantas muestras se generan entre dos muestras de entrada. El SPS, controla la tasa de muestreo del interpolador para la relación entre la velocidad de bits y la tasa de símbolos,

haciendo tambien una recuperación de los datos de forma más facil. En el caso de colocar otro valor, se debe ajustar la interpolación esto generaría un aumento en la tasa de muestreo.

2. Se observa un cambio en las graficas, analizando que aumenta la densidad espectral de potencia, se reduce el ancho de banda y esto ocasiona que la amplitud de los lobulos cambie.

$$3. \text{ Ancho de banda } = \frac{Rb}{FsP4}$$

4. Se debe conocer la frecuencia de muestreo

$$\text{Muestreo en p3}(FsP3)=Sps*FsP4$$

- c. Con los análisis espectrales de señales procesadas a la imagen y el audio, se observo que aunque sus parámetros eran exactamente iguales, los resultados varían en su representación de DSP, debido a que una señal de audio es mucho mas compleja en las variaciones de tiempo.
- d. Este es un bloque de control de flujo el cual es muy importante para la velocidad de procesamiento de la señal y también para evitar el procesamiento excesivo ya que esto evita consumir recursos de forma innecesaria ajustando el ajuste de tasa de bits.
- e. Se verá afectada por la falta de simetría, la falta de esta puede ocasionar un espectro de frecuencia asimétrico y un impacto en el ancho de banda
- f. Se puede observar que el ruido blanco tiene un ancho de banda muy grande al ancho de banda de la señal que nos interesa, ya que el ruido blanco teóricamente tiene un ancho de banda infinito.
- g. Una señal binaria aleatoria de forma rectangular se puede representar como una SINC, esta señal tiene componentes con todas las frecuencias, reduciendo la amplitud a medida que se aleja de la frecuencia central, es decir,



no se podría observar un ancho de banda infinito por limitaciones en el software.

h. En la función sinc.

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$$

$$R_s = \frac{f_s}{Sps}$$

$f_s = \text{Frecuenciademuestreo}$

Sps = número de muestras por símbolo

Con esto hallamos el ancho del pulso rectangular en términos de la tasa de símbolos:

$$T = \frac{1}{R_s}$$

La distancia entre los ceros seguidos en el espectro:

$$f = R_s$$

$$N_{\text{merodelbulos}} = \frac{f_s}{2R_s}$$

Solamente se halla este número de lóbulos hasta la frecuencia de Nyquist

i. Para calcular este espectro de una señal binaria aleatoria de forma rectangular, se debe considerar la tasa de bits  $R_s$  y el Sps, al determinar la tasa de símbolos  $R_s$ , dependiendo de la modulación, en este caso asumimos una modulación binaria, en esta cada símbolo representa un solo bit, es decir,  $R_s = R_b$ .

j. Cuando se toma un analizador de espectros en la transformada de Fourier (FFT) se determinará por una resolución de frecuencia FFT, esta directamente relacionada con el número de puntos  $N$  de FFT y la frecuencia de muestreo  $f_s$ .

A medida que aumenta el número de  $N$ , la resolución espectral disminuye, ocasionando una mejora, ya que esto ayuda a distinguir componentes de frecuencia mucho más cercanos entre ellos, pero se debe tener en cuenta que aumentar  $N$  nos implica una mayor memoria.

k. Esto ocasiona que 16 bits por byte sean entregados, colocando una distorsión en la señal, aumentando el ancho de banda y también errores en la señal.

l. Determinar el ancho de banda x lobulo

$$\text{Anchodebandaxlobulo} = \frac{\text{Anchodebandatotal}}{\text{delobulos}}$$

Con esto ahora se determina la frecuencia de muestreo

$$f_s = R_s \times Sps$$

m. Se calcula con:

Frecuencia de muestreo en la entrada = Es la frecuencia de muestreo del flujo de bits que entra al bloque "Unpack K bits"

$K$  = Longitud del paquete en bits

Entonces:

$$\text{Frecuenciademuestreoalasalida} = \frac{\text{Frecuenciademuestreoalaentrada}}{K}$$

n. Este bloque se utiliza para cambiar caracteres en puntos flotantes, no puede cambiar la frecuencia de muestreo pero si la representación de los datos. En este caso como el bloque no tiene nada que ver con la frecuencia de muestreo, podemos decir que la velocidad de muestreo en la salida es igual a la velocidad de muestreo en la entrada.

o. Se debe utilizar una tasa de símbolos por segundo (Sps) mucho más alta con respecto al ancho de banda, para que los cambios en la señal pasen rápido y de manera aleatoria para que el PSD se pueda parecer a la del ruido blanco.

p. Función de manipulación (h): Esto determina como se cambia la señal binaria aleatoria, para poder tomar forma de dientes de sierra, la función es:

$$h(t) = \frac{2t}{T}$$



Duración de Bits: Se ajusta para que coincida con el período (T).

- q. Definir la función de manipulación (h) para generar la forma de onda que queremos, la Unipolar RZ quiere decir que los bits 1 se van a mostrar como puntos positivos y volviendo a cero, entonces los bits 0 solamente serían puntos en cero.

T = Período de un bit

Para un bit en 1:

$$h(t) = 1 \\ 0 = t < \frac{T}{2}$$

Para un bit en 0:

$$h(t) = 0 \\ 0 = t < T$$

Configurar tiempo y duración de los bits:  
Esto debe coincidir con la representación Unipolar RZ.

- r. Definir la función de manipulación (h):  
Definiendo una función h que pueda generar la forma de la onda que se desea, esta codificación coloca los bits 1 en representación de transiciones de Tensión en el medio del bit, pero los bits 0, se colocan como la ausencia de transiciones, entonces.

Para un bit 1:

T = Período de un bit

A = Amplitud

$$h(t) = +A \\ \text{Para } T \leq t < T$$

Para un bit 0:

$$h(t) = -A \\ \text{Para } \frac{T}{2} \leq t < T$$

Esto genera la representación "Manchester NRZ"

Configuración del tiempo y duración de los bits: Estos dos deben coincidir con la configuración "Manchester NRZ" el período de los bits tiene que ser la mitad del período original.

- s. Añadiendo el bloque "Char to float" (Convierte datos de tipo carácter en datos tipo flotante) seguido con el bloque "Random source", el bloque "rfvco" (Genera señales de RF y manipula la frecuencia en el dominio de radiofrecuencia).

- t. - Añadiendo "Char to Float" más "Random source".

- Con el bloque "fvco ff" (Genera una modulación BPSK)

- Con el bloque "Multiply Const", solo se debe configurar la constante que se quiere multiplicar y conectar la señal de entrada.

Entonces la señal de salida será el resultado de la multiplicación de la señal de entrada por la constante que se quiere.

- u. Se deberá añadir

- "Char to Float"

- "Signal Source" (Genera señales simuladas)

- "Multiply" (Realiza multiplicaciones entre dos señales)

- "Throttle" (Controla la velocidad de procesamiento en los datos)

- "Float to complex"

- v. Se debe colocar:

- "Constellation Modulator" (Genera una secuencia de símbolos que representa el patrón de un latido cardíaco) Los símbolos están de los rangos de 0 a 1, donde 0 representa reposo y 1 un pico de un latido cardíaco.

- "Throttle", esto ajustará la señal de muestreo, tienen que ser alta para que se pueda representar con los latidos.

- "Multiply", para multiplicar la señal binaria por una constante de esta manera serán valores reales en vez de 0 y 1.

- "FIR Interpolating Filter", para aumentar la densidad de las muestras obteniendo una señal más sofisticada, iguales a los latidos cardíacos.



- x. La diferencia esta en la distribución de las amplitudes de los símbolos, las señales binarias bipolares tienen componentes de frecuencia bajas por la alternancia entre valores positivos y negativos, en cambio la señal binaria unipolar tiene un pico más definido en la frecuencia de la señal portadora debido a que tiene solamente amplitudes positivas

puede ver que entre mucho más grande esto, mayor será la frecuencia de muestreo.

- Se puede ver que el ruido blanco enre mayor sea el Sps, el ruido va a aumentar.
- El ancho de banda de un ruido blanco es infinita y esta siempre presente
- La frecuencia de muestreo es proporcional a los  $R_b$  y Sps.

### 3. Conclusiones

- En la Densidad Espectral de Potencia (PSD), al analizar los cambios, cambiando los valores de Sps se

### Referencias