

SISTEMAS MODULARES DE COMUNICACIÓN BASADOS EN TÉCNICAS DE MODULACIÓN DE ESPECTRO ENSANCHADO UTILIZANDO GNU RADIO*

Maria Camila Herrera Ramos**

Luis Miguel Capacho Valbuena***

Auramaria Londoño Cano****

Recibido: 18/09/2022 Aceptado: 02/07/2025

<https://doi.org/10.22395/rium.v25n47a1>

RESUMEN

Esta investigación se centra en el desarrollo de sistemas modulares en el software GNU Radio como herramienta de apoyo didáctica para la experimentación en el área de telecomunicaciones. Los sistemas desarrollados utilizan las técnicas de modulación de espectro ensanchado por secuencia directa (*DSSS, Direct Sequence Spread Spectrum*) y por salto de frecuencia (*FHSS, Frequency Hopping Spread Spectrum*). Los módulos configurados en cada sistema permiten la variación de parámetros como el tamaño de la información, el tipo de modulación, el tipo de canal, la amplitud del ruido y características propias de las técnicas implementadas. Los resultados obtenidos son analizados a partir de gráficas de la señal original y la señal modulada en los dominios del tiempo y de la frecuencia, diagramas de constelación, la relación señal a ruido (*SNR, Signal to Noise Ratio*) y la tasa de error binario (*BER, Bit Error Rate*). La modularidad ofrecida por los sistemas desarrollados en este trabajo facilita la generación de diversos escenarios sin la necesidad de crearlos de forma independiente, lo que permite enfocar la experimentación en el análisis de resultados.

Palabras clave: comunicación por radio, procesamiento de señales, radio definida por software, software de código abierto, comunicaciones digitales, análisis de errores, transmisión de datos.

* Resultado del Proyecto de Investigación. Fuente de financiación, Universidad del Quindío.

** Ing. Universidad del Quindío. Armenia, Colombia. Correo electrónico: mcherrerar@uqvirtual.edu.co Orcid: 0000-0002-1453-3612

*** MSc. Universidad del Quindío. Grupo de Procesamiento Digital de Señales y Procesadores (GDSPROC). Armenia, Colombia. Correo electrónico: lmcapacho@uniquindio.edu.co Orcid: 0000-0002-4109-2533

**** MSc. Universidad del Quindío. Grupo de Electrónica Aplicada y Energías (Electrae). Armenia, Colombia. Correo electrónico: amlondonoc@uniquindio.edu.co Orcid: 0000-0002-6045-9425

MODULAR COMMUNICATION SYSTEMS BASED ON SPREAD SPECTRUM MODULATION TECHNIQUES USING GNU RADIO

ABSTRACT

This research focuses on modular system development in GNU Radio software as a teaching aid for telecommunications experimentation. The developed systems use direct sequence spread spectrum (DSSS) and frequency hopping (FHSS) modulation techniques. The modules configured in each system allow for a variation of parameters such as data size, modulation type, channel type, noise amplitude, and characteristics specific to implemented techniques. Results were analyzed using original signal graphs and modulated signal graphs in time and frequency domains, constellation diagrams, signal-to-noise ratio (SNR), and bit error rate (BER). The modularity offered by the systems developed in this work facilitates the generation of diverse scenarios without a need to create them independently, allowing experimentation to focus on analyzing results.

Keywords: Radio communication, signal processing, software-defined radio, open-source software, digital communications, error analysis, data transmission.

INTRODUCCIÓN

En los últimos años la utilidad de las redes inalámbricas ha generado una creciente demanda e implementación de las mismas en todo tipo de entornos, dado el gran abanico de ventajas que ofrecen frente a las tradicionales redes cableadas. Sin embargo, en este tipo de redes el medio de comunicación es el canal radio, el cual, al ser no guiado está expuesto a múltiples interferencias. Sumado a esto, la frecuencia de una señal inalámbrica convencional se mantiene tan constante como se lo permita la técnica de modulación empleada, por lo que el ancho de banda puede mantenerse dentro de ciertos límites y por tanto la señal puede ser interceptada fácilmente por alguien que quiera capturar la información. Estos dos problemas pueden ocurrir bajo ciertas circunstancias. Por ejemplo, cuando otra señal se transmite en la misma frecuencia o muy cerca de la frecuencia de la señal deseada se genera una interferencia que si bien puede ser accidental, como en el caso de un usuario que trata al mismo tiempo de transmitir a través del canal, también puede ser deliberada, como ocurre con una interferencia intencional dirigida directamente al receptor para sabotear la comunicación [1].

Para minimizar los problemas mencionados que pueden derivarse de las vulnerabilidades de las tecnologías de comunicaciones convencionales, surgen las técnicas de modulación de espectro ensanchado (*Spread Spectrum, SS*) para extender la señal dentro de un amplio ancho de banda, de tal modo que primero, se maximice la capacidad de comunicación cuando se tiene un espectro limitado de radio y segundo, se dificulte la interceptación de la señal por receptores no autorizados. Esto último, debido a que, si alguien quiere interceptar una señal de espectro ensanchado, esa persona debe tener un receptor que conozca la función de ensanchamiento y su punto de inicio, así como el tiempo empleado por el transmisor para enviar la información.

El SS es una técnica de modulación muy empleada actualmente en sistemas de comunicaciones que utilizan tecnologías como Bluetooth, ZigBee, 3G y 3.5G del sistema de telefonía celular. Estos sistemas son llamados de espectro ensanchado, precisamente porque la señal a transmitir se expande en una banda muy ancha de frecuencias, incluso más amplia que el ancho de banda mínimo requerido para transmitir la señal original, y con un nivel de potencia similar al ruido de fondo. De este modo, las señales ensanchadas son altamente resistentes al ruido y a la interferencia; y son difíciles de interceptar, ya que suenan como un ruido de corta duración en cualquier receptor no autorizado [2].

Dadas las ventajas y aplicaciones de estas técnicas, cobra importancia la exploración de herramientas de uso gratuito como GNU Radio que permitan realizar la implementación de sistemas de comunicaciones basados en SS a nivel de software, como apoyo en la academia, para la enseñanza y el aprendizaje de estos temas dentro del aula

de clase; y como apoyo en la industria, debido a las ventajas que brinda el diseño y la implementación de sistemas de radio definida por software (*Software Defined Radio*, *SDR*) en plataformas de bajo costo usando GNU Radio. Es por esto que este artículo presenta simulaciones de técnicas de modulación de SS en la herramienta de software libre y de código abierto GNU Radio, el cual provee bloques de procesamiento de señal para simular sistemas de comunicaciones modulares a través de software, en lugar de utilizar circuitos integrados dedicados en hardware, con la posibilidad de variar los parámetros de configuración dentro de cada bloque, así como también de crear bloques nuevos usando el lenguaje de programación Python.

Aunque en la literatura existen trabajos e investigaciones que están enfocados en la simulación y evaluación de distintas técnicas de modulaciones digitales en GNU Radio, como en [3], donde se implementan sistemas de radio definida por software para televisión digital terrestre en tiempo real, usando las modulaciones digitales por desplazamiento de fase en cuadratura (*Quadrature Phase Shift Keying*, *QPSK*) y su variante diferencial (*Differential Quadrature Phase Shift Keying*, *DQPSK*), o como en [4], donde se diseña e implementa un bloque modulador/demodulador QAM (*Quadrature Amplitude Modulation*) que ofrece una alternativa con mayores características al bloque configurado por defecto en GNU Radio, persiste la necesidad de profundizar en la implementación de bloques funcionales para técnicas de espectro ensanchado (SS). Estas técnicas, como el espectro ensanchado por secuencia directa (DSSS) o por salto en frecuencia (FHSS), permiten mejorar significativamente la inmunidad de las señales al ruido y la interferencia, además de dificultar su interceptación por parte de receptores no autorizados [5]. En [6] se presenta una plataforma creada con GNU Radio y un dispositivo USRP para la enseñanza de principios de *beamforming*, demostrando cómo modular bloques funcionales en entornos educativos permite explorar técnicas avanzadas de procesamiento de señales en tiempo real. Los resultados obtenidos en estas investigaciones brindan una base comparativa útil tanto para el diseño y evaluación de redes inalámbricas en la industria como para el desarrollo de herramientas educativas y de investigación en telecomunicaciones.

El resto del artículo presenta en la sección 1 la construcción de los módulos necesarios para realizar los escenarios de simulación de DSSS y FHSS, en la sección 2 los resultados obtenidos, y finalmente, en la sección 3 la discusión de los mismos.

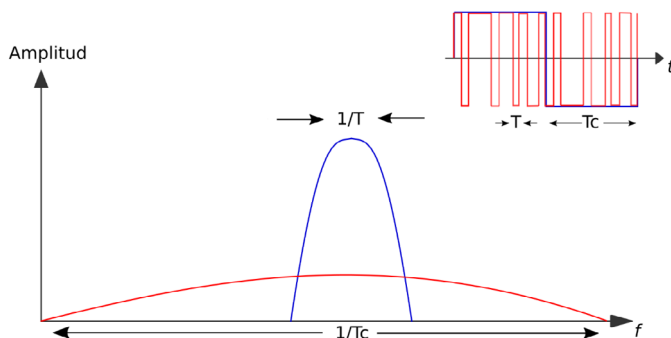
1. MATERIALES Y MÉTODOS

Los sistemas de comunicación implementados en este trabajo se basan en las técnicas de espectro ensanchado por secuencia directa y por saltos de frecuencia sobre la herramienta GNU Radio. En esta sección se explica el funcionamiento de las técnicas usadas y la creación de los bloques necesarios para la implementación en GNU Radio.

1.1. Sistema de espectro ensanchado por secuencia directa (DSSS)

DSSS es una técnica de modulación que consiste en la generación de una secuencia de bits redundante llamados chips para cada uno de los bits que componen la señal original. La secuencia de chips también se conoce como código de dispersión o código de pseudoruido y es el que modula la información para generar una señal ensanchada con un ancho de banda mucho mayor y una densidad espectral de potencia reducida. Así el espectro resultante es muy parecido al del ruido, como se muestra en la figura 1, de tal forma que a todos los receptores les parecerá ruido menos al que va dirigida la señal. Cuanto mayor sea la cantidad de chips del código, mayor será la resistencia de la señal a las interferencias. En la recepción es necesario realizar el proceso inverso para obtener la señal de información original [7].

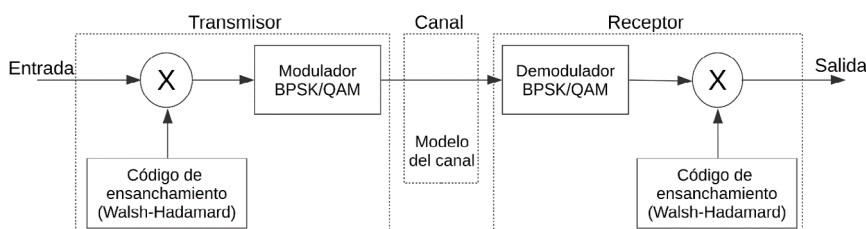
Figura 1. Técnica de modulación DSSS



Fuente: elaboración propia.

El modelo general del sistema de comunicaciones usando la técnica de modulación DSSS mostrado en la figura 2, está compuesto por una etapa de transmisión donde la señal de entrada es multiplicada por el código de dispersión y posteriormente modulada usando QAM o BPSK; la señal en radiofrecuencia es enviada usando un modelo de canal y procesada en el receptor para obtener el mensaje original.

Figura 2. Sistema de comunicaciones usando DSSS

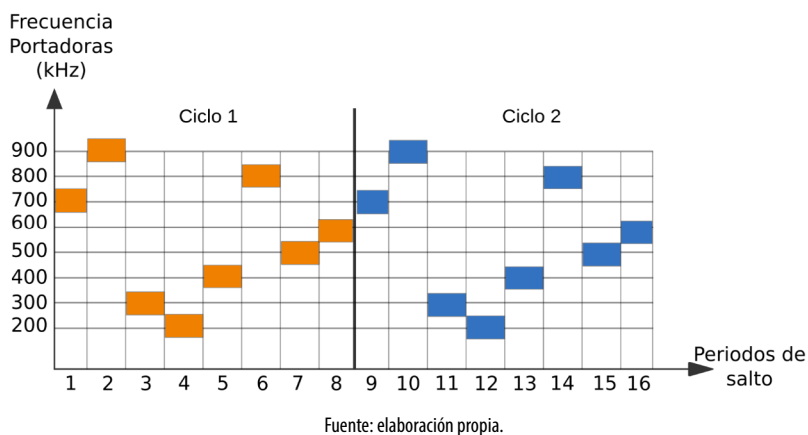


Fuente: elaboración propia.

1.2. Sistema de espectro esparcido por salto de frecuencia FHSS

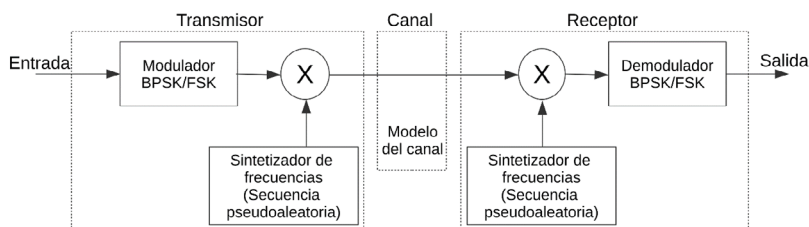
FHSS es una técnica de espectro ensanchado utilizada en los sistemas de transmisión de radio. Consiste en transmitir una parte de la información en una determinada frecuencia durante un intervalo de tiempo inferior a 400 ms. Pasado este tiempo se cambia la frecuencia de emisión y se sigue transmitiendo en otra frecuencia. De esta manera cada tramo de información se va transmitiendo en una frecuencia distinta durante un intervalo muy corto de tiempo, como se muestra en la figura 3. El orden en los saltos de frecuencia se determina según una secuencia pseudoaleatoria almacenada en una tabla que tanto el emisor y el receptor comparten. Así, si se intentara interceptar la señal, solo se conseguiría para unos pocos bits [8].

Figura 3. Técnica de modulación FHSS



El modelo general del sistema de comunicaciones usando la técnica de modulación FHSS, mostrado en la figura 4, consta de un transmisor en donde la señal modulada usando FSK o BPSK es multiplicada por la secuencia pseudoaleatoria que genera los saltos en frecuencia de la señal que se transmite por un canal, que posteriormente se obtiene en el receptor para obtener el mensaje original.

Figura 4. Sistema de comunicaciones usando FHSS



1.3. Bloques implementados en GNU Radio

GNU Radio es un conjunto de herramientas de código abierto y libre, compuesto por una biblioteca de bloques para el procesamiento de señal que permite construir y desarrollar radio definida por software. Los bloques de procesamiento de señal en GNU Radio son escritos en C++ y Python y es posible crear nuevos bloques a partir de los bloques existentes o creándolos desde cero usando la estructura definida para ello. Los tipos de bloques que pueden ser creados en GNU Radio son clasificados de acuerdo al número de elementos de entrada que consume (N) y el número de elementos de salida que produce (M), obteniendo cuatro tipos de bloques diferentes: sincronización (1:1), diezmado (N :1), interpolación (1: M) y general (N : M). También, es posible crear nuevos bloques conectando bloques existentes entre sí, los cuales se conocen como bloques jerárquicos.

Para el desarrollo de los módulos necesarios para la implementación de los sistemas de comunicación basados en técnicas de modulación de espectro ensanchado propuestos en este trabajo, se hizo uso del lenguaje de programación Python para crear los bloques jerárquicos modulares que permiten la variación de parámetros en tiempo real de las características más relevantes en este tipo de modulación, así como la visualización de la señal en cada una de las etapas y la interfaz gráfica de usuario para el ajuste de diferentes parámetros.

Los bloques implementados en este trabajo para la implementación de las técnicas de modulación DSSS y FHSS se dividen en bloques tipo fuente y bloques de procesamiento. Los bloques tipo fuente corresponden a los bloques que poseen salidas, más no entradas, es decir, no dependen de un bloque previo para generar una señal. Los bloques de procesamiento permiten procesar la señal de entrada y obtener una salida con características diferentes conforme al proceso realizado.

1.3.1. Bloques tipo fuente

Se implementó un bloque tipo fuente para generar la secuencia pseudoaleatoria o código de ensanchamiento necesario para DSSS y un bloque tipo fuente para los saltos de frecuencias definidos por una secuencia pseudoaleatoria necesarios para FHSS. El bloque para generar el código de ensanchamiento trabaja en función del número chips por el que se desea multiplicar la información de entrada, generando una secuencia múltiplo de la relación entre el tamaño de datos a procesar y los chips a generar. El bloque encargado de generar la secuencia de saltos pseudoaleatoria depende de los bits por segundo, el tamaño de los datos, la frecuencia central y las variaciones de la frecuencia con respecto a la frecuencia central.

Código de ensanchamiento

El bloque tipo fuente para el código de ensanchamiento usado en DSSS, genera un código Walsh a partir de una matriz Walsh Hadamard H de tamaño $n \times n$, donde n es el orden de la matriz. La matriz Walsh Hadamard se genera una sola vez cuando el parámetro correspondiente al número de chips es modificado con el fin de mejorar el rendimiento del bloque. La matriz es obtenida de forma recursiva a partir de las matrices de la ecuación 1.

$$H_1 = (1)H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} H_{2N} = \begin{pmatrix} H_N & H_N \\ H_N & -H_N \end{pmatrix} \quad (1)$$

El código Walsh se genera a partir de matrices Hadamard donde las filas de la matriz son de orden 2^N para codificar una secuencia de N bits [9]. La secuencia pseudoaleatoria se obtiene usando números aleatorios conforme a una distribución uniforme discreta, para las posiciones de la matriz a tomar. El bloque puede ser configurado con un número de chips equivalente a 2, 4, 8, o 16.

Secuencia de saltos pseudoaleatoria

La secuencia pseudoaleatoria de saltos de frecuencia para FHSS es generada también usando Walsh-Hadamard, a partir de la cual se seleccionan las frecuencias de ensanchamiento deseadas por el usuario, en función de los bits por segundo (bps) ingresados, así como de los valores de frecuencia de portadora (fc) y de frecuencia delta (fd). Estas frecuencias se sintetizan posteriormente para generar la señal en radiofrecuencia ensanchada que será transmitida por el canal.

1.3.2. Bloques de procesamiento

Para el procesamiento de la información se implementó un bloque de procesamiento encargado de realizar la multiplicación entre el código de ensanchamiento y la información a modular y demodular. También se implementaron los módulos necesarios para realizar la modulación encargada de generar la señal en radiofrecuencia.

Multiplicación por código de ensanchamiento

Para el caso de DSSS, los bits provenientes de la fuente son multiplicados uno por uno por todos los chips generados, lo que provoca que a la salida del bloque se tenga la señal ensanchada y codificada por la cadena de bits provenientes del generador de señales pseudoaleatorias. Este proceso es realizado de forma contraria en el demodulador para obtener de nuevo la información.

Moduladores y demoduladores

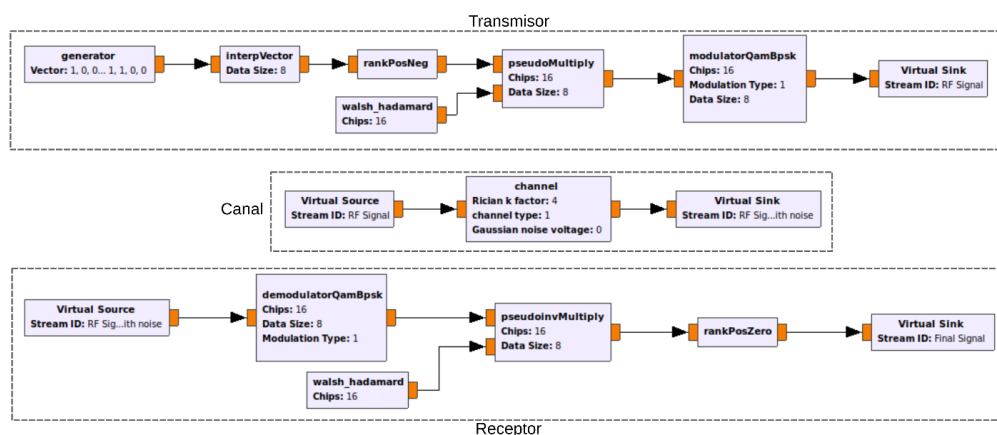
Para realizar la modulación y generar la señal en radiofrecuencia, se implementaron bloques de procesamiento encargados de modular en QAM, BPSK y FSK. Para conservar un sistema modular, se agruparon las modulaciones y demodulaciones usadas en cada técnica obteniendo cuatro bloques de modulación, dos para la modulación y demodulación QAM/BPSK para DSSS y dos para la modulación y demodulación FSK/BPSK para FHSS. Como salida adicional, se tienen las componentes imaginarias y reales que sirven para visualizar la constelación de manera individual.

Modelo de canal

Para simular el canal, se implementó un bloque de procesamiento que contiene tres modelos de distribuciones de canal: Gaussiano, Rician y Rayleigh. Para el modelo de canal Gaussiano se utilizó en bloque Noise Source disponible en GNU Radio, el cual permite configurar la amplitud del ruido generado. Para los modelos de canal Rician y Rayleigh, se usó el bloque Fading Model disponible en GNU Radio, el cual permite variar entre estas dos configuraciones; además, permite variar el factor k asociado al modelo Rician que determina la relación de la componente directa y la componente multitrayecto.

El modelo general del sistema de comunicaciones usando DSSS, implementado en GNU Radio, se muestra en la figura 5. El sistema contiene un vector de entrada que es acondicionado, modulado y enviado por un canal que contiene un modelo de canal que puede ser variado por el usuario, para finalmente demodular y obtener el mensaje original.

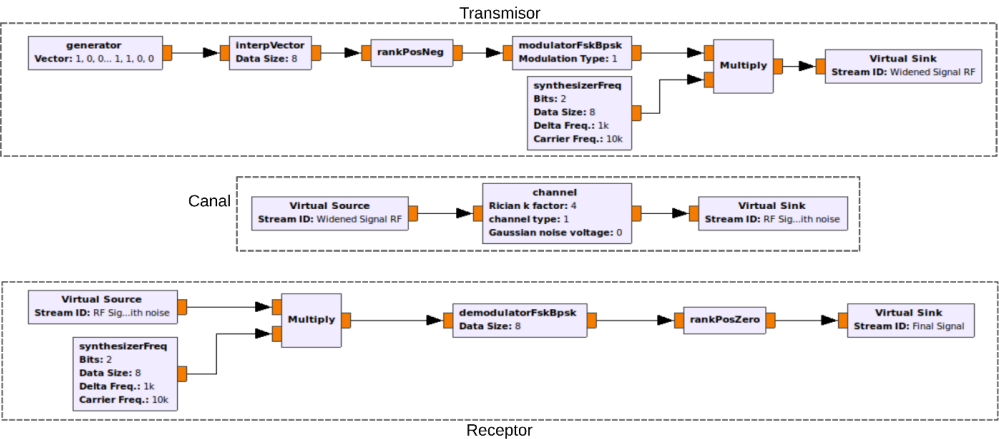
Figura 5. Sistema de comunicaciones usando DSSS en GNU Radio



Fuente: elaboración propia.

El modelo general del sistema de comunicaciones usando FHSS, implementado en GNU Radio, se muestra en la figura 6. El sistema contiene un vector de entrada que es modulado y enviado por un canal que contiene un modelo de canal que puede ser variado por el usuario, para finalmente demodular y obtener el mensaje original.

Figura 6. Sistema de comunicaciones usando FHSS en GNU Radio



Fuente: elaboración propia.

2. RESULTADOS

Después de realizar la implementación de cada uno de los sistemas de comunicaciones, se procede a realizar la verificación de los resultados; primero, a partir de la visualización de las gráficas de la señal al final de cada una de las etapas relevantes del sistema; y segundo, a través de métricas de desempeño como la tasa de error binario (*Bit Error Rate, BER*) y la relación señal a ruido (*Signal Noise Ratio, SNR*). Como valor agregado, se crean variables que pueden ser modificadas en tiempo real, permitiendo observar los resultados ante diferentes configuraciones del sistema. Estas variables son creadas con la ayuda de las herramientas GUI Widgets de QT, estableciendo un bloque QT GUI Tab Widget que permite mostrar en pestañas individuales cada una de las gráficas. Las variables y gráficas configuradas para la visualización de los resultados del sistema DSSS y FHSS se muestran en la tabla 1.

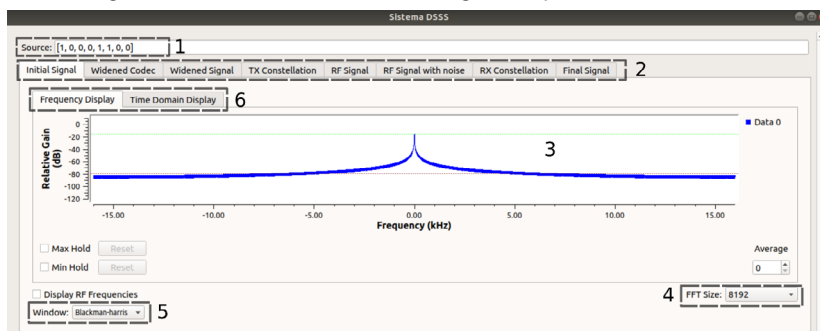
Tabla 1. Variables y gráficas configuradas para los sistemas

Visualización de los sistemas		
Sistema	Variables en tiempo real	Gráficas
DSSS	• Vector de fuente	• Señal original
FHSS	• Tamaño de la información	• Señal ensanchada
	• Tipo de canal	• Señal RF
	• Amplitud del ruido (Canal Gaussiano)	• Señal RF con ruido
	• Variable k (Canal Rician)	• Señal final
	• Tipo de modulación	
DSSS	• Número de chips	• Código de ensanchamiento
		• Constelación en el transmisor
		• Constelación en el receptor
FHSS	• Frecuencia de portadora	• Frecuencias de ensanchamiento
	• Frecuencia delta	• Señal modulada
	• Bits por segundo	• Señal demodulada

Fuente: elaboración propia.

Como ejemplo, en la figura 7 se muestra la ventana de gráficas inicial para la visualización de resultados en el sistema DSSS, en la cual se observa de acuerdo con la tabla 1, la variable que permite modificar el vector de entrada o fuente en tiempo real en el indicador 1, el nombre de las gráficas en pestañas individuales en el indicador 2, la gráfica seleccionada en 3, el tamaño de la FFT (*Fast Fourier Transform*) y el tipo de ventana utilizada en 4 y 5, respectivamente, y finalmente, la posibilidad de observar las gráficas en el dominio de la frecuencia o en el dominio del tiempo en el indicador 6.

Figura 7. Ventana de visualización de gráficas para el sistema DSSS



Fuente: elaboración propia.

La figura 8 corresponde a la parte inferior de la ventana de visualización de resultados donde se muestran las variables que pueden ser configuradas en tiempo real. Dichos parámetros son enunciados a continuación:

Figura 8. Parámetros de configuración para el sistema DSSS

Noise Voltage	1	0,000
K Factor(Rician)	2	4,0
Modulation: 16QAM	3	
chips: 16	4	
Channel type: Gaussian	5	
Data_Size: 8	6	

BER	SNR
Data 0	Data 0
0.000000	96.989594

Fuente: elaboración propia.

1. *Noise Voltage*: este parámetro corresponde a la amplitud del voltaje para un canal tipo Gaussiano configurado en el bloque *Channel*. Este valor es muy importante para determinar la relación señal a ruido del sistema.

2. *K Factor (Rician)*: este parámetro es utilizado bajo la configuración del canal Rician. El factor K representa la razón de potencias de la componente directa de la señal y los componentes del multitrayecto dentro de la distribución de Rician.

3. *Modulation*: el módulo creado permite elegir la opción de modulación 16QAM o BPSK para el sistema DSSS; y la modulación BPSK y FSK para el sistema FHSSS.

4. *Chips*: esta variable define la cantidad de elementos (chips) que tendrá la señal de esparcimiento, producto de las matrices de Walsh-Hadamard, que para el caso de este sistema en particular está configurado para tener un valor de 2, 4, 8 y 16 chips.

5. *Channel type*: permite escoger el modelo de canal: Gaussiano, Rician o Rayleigh.

6. *Data Size*: representa el tamaño del vector de entrada, es decir, cuantos bits va a tener la señal como fuente. En este caso se definieron solo los tamaños de 8 y 16, con el fin de garantizar la visualización clara de cada etapa, sin saturar visualmente la ventana de gráficas.

Para el caso del sistema FHSS, se pueden modificar, adicionalmente, las siguientes variables:

- *Bps*: esta variable define la cantidad de frecuencias que tendrá el código de ensanchamiento: por ejemplo, para $bps = 2$, el número de frecuencias es $2^{bps} = 4$, y cuatro frecuencias pueden ser representadas por dos bits, así: 00, 01, 10, 11. Las opciones de valores de bps son: 2, 3 y 4.

- *Carrier Freq* (F_c): es el valor de la frecuencia central o frecuencia de portadora.

- *Delta Freq* (F_d): es el valor de la frecuencia delta o desviación de frecuencia.

Finalmente, la interfaz de visualización muestra los resultados del cálculo de la BER y la SNR, parámetros importantes para la evaluación del desempeño de un sistema de comunicaciones digitales.

Como ejemplos, en las secciones 2.1 y 2.2 se muestran las gráficas obtenidas en un escenario con ciertos parámetros configurados para los sistemas DSSS y FHSS. Los resultados obtenidos reflejan el modularidad de los bloques creados, así como las prestaciones de los sistemas implementados para la generación de múltiples configuraciones, lo cual facilita y agiliza el análisis de las técnicas de modulación por SS.

2.1. Escenario configurado para el sistema DSSS

Para verificar el procesamiento aplicado a una señal modulada en DSSS, se muestran las gráficas obtenidas en cada etapa del sistema, un ejemplo de escenario que está configurado con los parámetros iniciales indicados en la tabla 2.

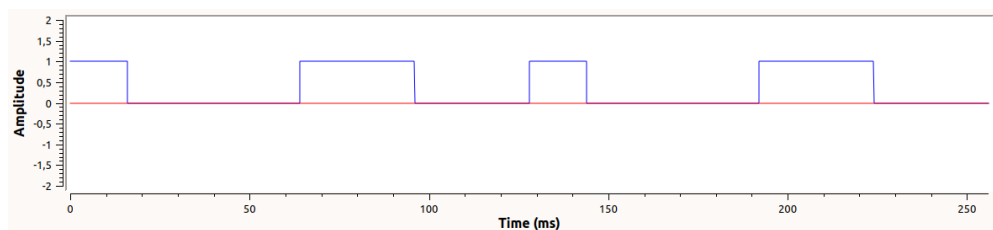
Tabla 2. Parámetros configurados para el sistema DSSS

Parámetros configurados en DSSS	
Vector: [1,0,0,0,1,1,0,0,1,0,1,1,0,0,1,0]	Tamaño de los datos: 16
Tasa de muestreo: 32000 Hz	Chips: 8
Modulación: BPSK	Factor k: 4
Voltaje de ruido: 0	Tipo de canal: Gaussiano
Ventana: Blackman-Harris	Tamaño FFT: 8192

Fuente: elaboración propia.

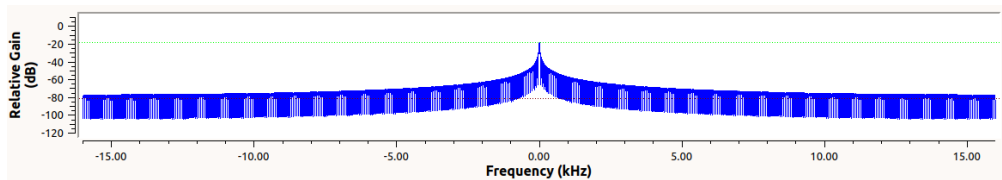
En la figura 9 se puede observar la secuencia de bits del vector fuente de la tabla 2 en el dominio del tiempo. Dado que la ventana de visualización de las gráficas es de 250 ms y la cantidad de datos es de 16 bits, el tiempo de duración de cada bit es de 15,625 ms. En la figura 10 se muestra el espectro de banda estrecha de la señal de origen generada por la fuente.

Figura 9. Señal origen en el dominio del tiempo



Fuente: elaboración propia.

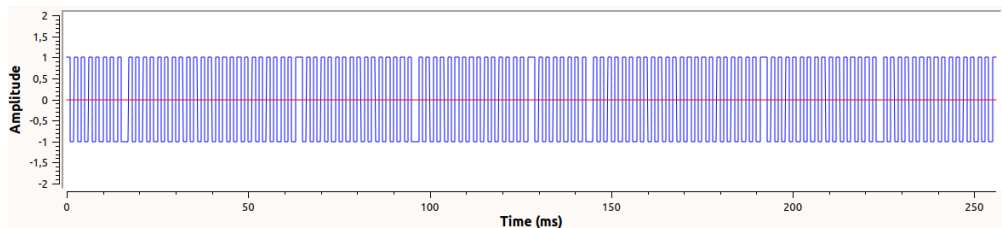
Figura 10. Señal origen en el dominio de la frecuencia



Fuente: elaboración propia.

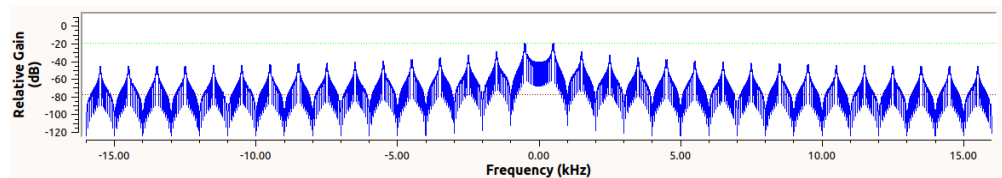
La señal ensanchada en el dominio del tiempo corresponde a la figura 11, en donde que está compuesta por una serie de 128 bits resultantes de la multiplicación de cada uno de los 16 bits por el código de dispersión configurado con 8 chips, según la tabla 2. En este caso, donde inicialmente 1 bit de la señal original tenía una duración de 15,525 ms, ahora en ese mismo tiempo se pueden transmitir 8 bits gracias a la modulación DSSS, con una duración de 1,953125 ms, lo cual genera el efecto de espectro ensanchado debido a que entre más angosto el pulso en el tiempo más ancho de banda ocupa en la frecuencia. La figura 12 muestra la señal ensanchada en el dominio de la frecuencia, donde se evidencia cómo la potencia de la señal es distribuida a lo largo de las componentes espectrales, característica básica de un sistema que usa espectro ensanchado.

Figura 11. Señal ensanchada por DSSS en el dominio del tiempo



Fuente: elaboración propia.

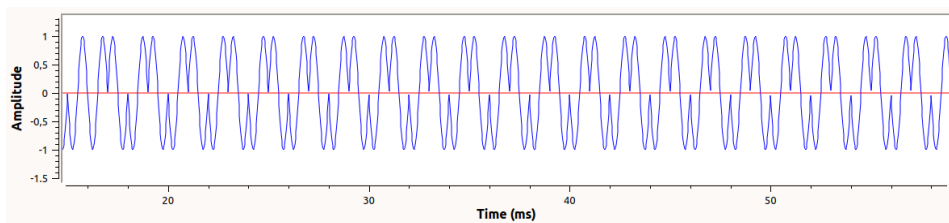
Figura 12. Señal ensanchada por DSSS en el dominio de la frecuencia



Fuente: elaboración propia.

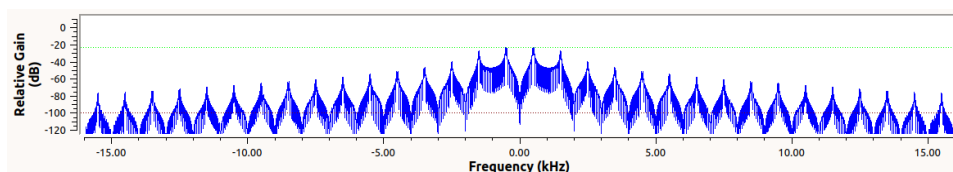
Una vez la señal original ha sido ensanchada es modulada digitalmente usando BPSK (*Binary Phase Shift Keying*). Las señales en el dominio del tiempo y de la frecuencia son mostradas en la figura 13 y figura 14, respectivamente, donde se pueden observar con claridad los cambios de fase de la señal correspondientes a los cambios de bit.

Figura 13. Señal modulada BPSK en el dominio del tiempo



Fuente: elaboración propia.

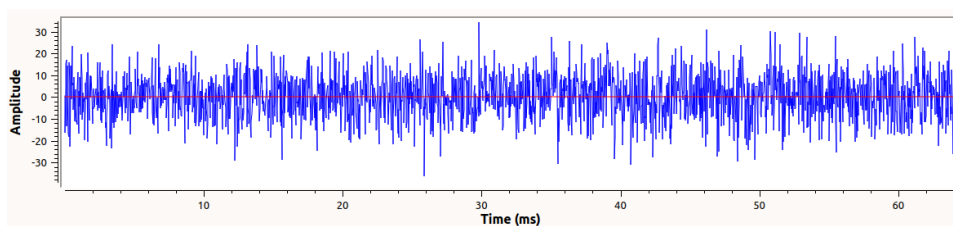
Figura 14. Señal modulada BPSK en el dominio de la frecuencia



Fuente: elaboración propia.

En este punto, la señal está lista para ser transmitida al canal Gaussiano con voltaje de ruido igual a 0 según las configuraciones de la tabla 2, lo cual significa que el ruido es nulo, por lo que la apreciación de la señal modulada con o sin el canal es la misma. Para mostrar el efecto del canal en la señal, se modificó en tiempo real el voltaje del ruido a 10, obteniendo como resultado la señal de la figura 15.

Figura 15. Señal modulada BPSK después de atravesar el canal Gaussiano

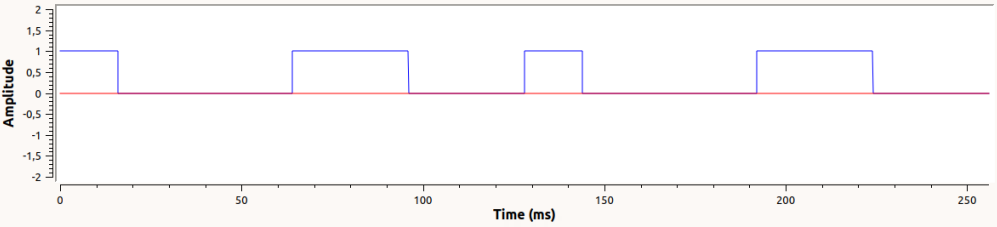


Fuente: elaboración propia.

Cabe resaltar que la SNR (expresada en Decibelios), en el caso de no presentar ningún ruido en el canal (Voltaje de ruido = 0) tiene un valor muy grande, en este caso de 126.96dB, y la BER al no haber interferencias es de 0. Cuando es usado el canal Rician, de acuerdo con el factor K se observan variaciones de la señal, pues a medida que su valor crece, la amplitud de la señal disminuye y viceversa. Para el canal Rayleigh igualmente se observan variaciones en la amplitud, frecuencia y fase de la señal, disminuyendo la SNR y aumentando la BER. Con los resultados de BER y SNR obtenidos ante las variaciones de voltaje del ruido en un canal Gaussiano, se concluye que cuando el voltaje del ruido aumenta es claro que la SNR disminuye, debido a que la potencia de la señal transmitida se divide entre una potencia de ruido cada vez más grande.

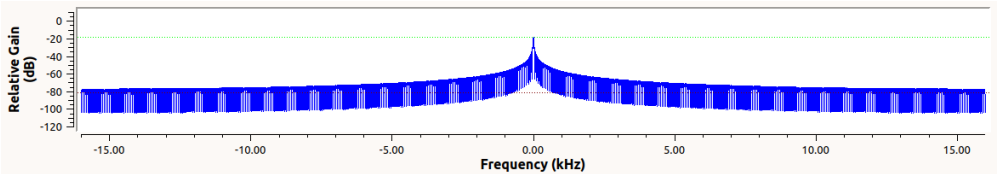
Finalmente, la secuencia de bits recuperada por el receptor se puede observar en la figura 16 y figura 17 en el dominio del tiempo y de la frecuencia, respectivamente, las cuales corresponden a las gráficas de la secuencia de bits generada por la fuente en el transmisor en la figura 9 y figura 10, indicando un procesamiento efectivo de la señal gracias a la técnica de modulación DSSS.

Figura 16. Señal final en el dominio del tiempo



Fuente: elaboración propia.

Figura 17. Señal final en el dominio de la frecuencia



Fuente: elaboración propia.

2.2. Escenario configurado para el sistema FHSS

En esta sección, también se presenta un ejemplo de configuración de los parámetros del sistema usando ahora una modulación FHSS. El objetivo es mostrar nuevamente el procesamiento que se va realizando a la señal original a medida que va atravesando todas las etapas en el transmisor, canal y receptor. En la tabla 3 se indican los valores de los parámetros configurados.

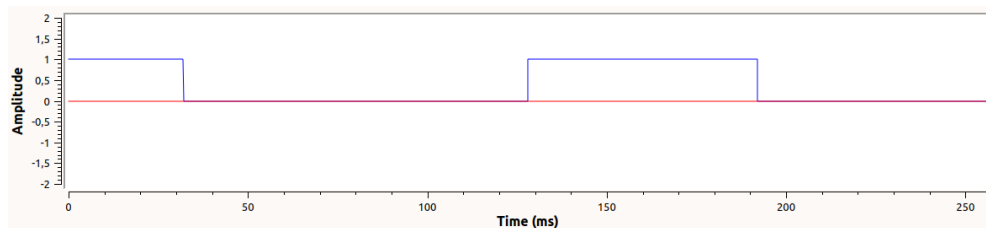
Tabla 3. Parámetros configurados para el sistema FHSS

Parámetros configurados en FHSS	
Vector: [1,0,0,0,1,1,0,0]	Tamaño de los datos: 8
Tasa de muestreo: 32000 Hz	Bps: 4
Modulación: FSK	Factor k: 4
Fc: 10000	Fd: 1000
Voltaje de ruido: 5	Tipo de canal: Gaussiano
Ventana: Blackman-Harris	Tamaño FFT: 8192

Fuente: elaboración propia.

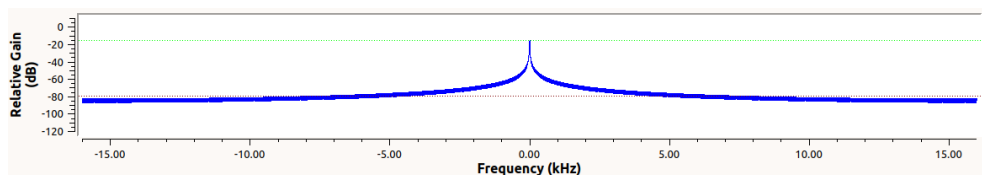
En la figura 18 y figura 19 se muestra la secuencia de 8 bits del vector fuente en el dominio del tiempo y de la frecuencia, respectivamente.

Figura 18. Señal origen en el dominio del tiempo



Fuente: elaboración propia.

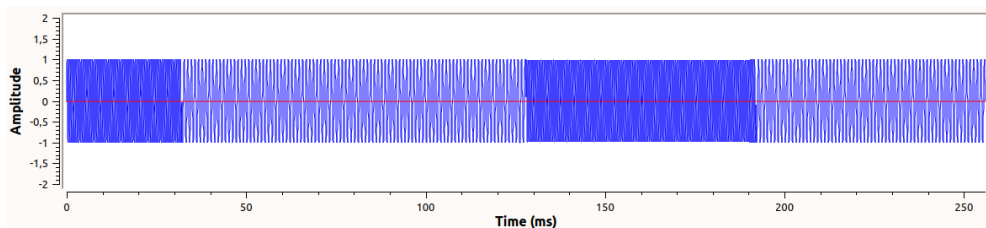
Figura 19. Señal origen en el dominio de la frecuencia



Fuente: elaboración propia.

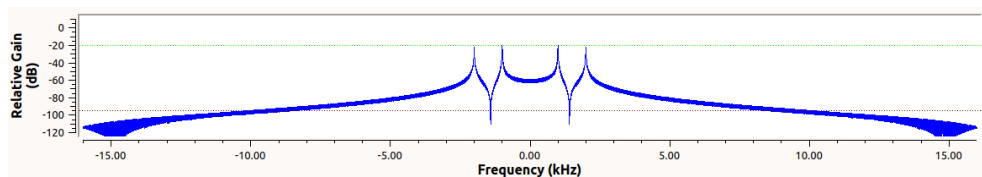
La señal modulada digitalmente usando FSK está en la figura 20 y figura 21, donde se ven dos frecuencias diferentes: 1kHz y 2kHz, representando cada una los bits en 1 y en 0.

Figura 20. Señal modulada FSK en el dominio del tiempo



Fuente: elaboración propia.

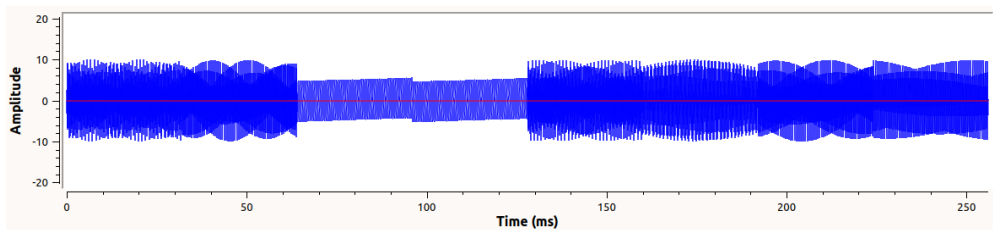
Figura 21. Señal modulada FSK en el dominio de la frecuencia



Fuente: elaboración propia.

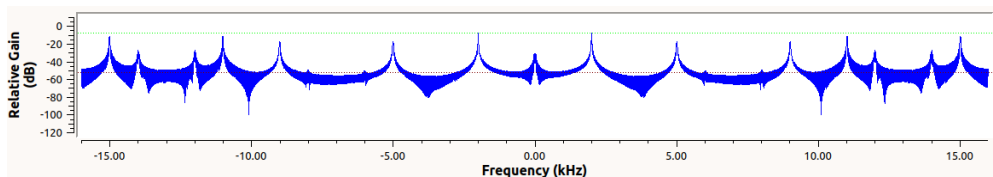
En la figura 22 y figura 23 se encuentra la señal ensanchada con FHSS, donde se claramente algunos saltos o cambios de frecuencias en la señal, generados de acuerdo con la secuencia pseudoaleatoria. En el tiempo, de la figura 22 se puede apreciar que la señal toma un comportamiento similar a una señal de ruido, ya que no posee un comportamiento periódico y posee distintas frecuencias con diferentes tiempos cada una a lo largo de la señal. El espectro de la señal ensanchada de la figura 23, si se compara la señal original (figura 19), es mucho más amplio y la potencia de la señal es distribuida a lo largo de las componentes espectrales, característica básica de un sistema que usa espectro esparcido.

Figura 22. Señal ensanchada por FHSS en el dominio del tiempo



Fuente: elaboración propia.

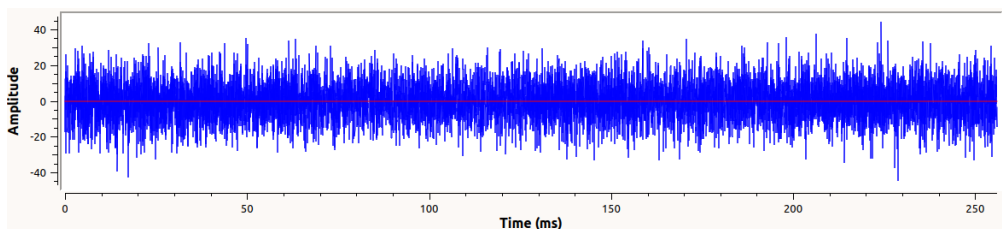
Figura 23. Señal ensanchada por FHSS en el dominio de la frecuencia



Fuente: elaboración propia.

Bajo un canal Gaussiano con voltaje del ruido igual a 5, según la configuración de la tabla 3, la señal observada en la figura 24 se parece a una señal de ruido, generando un efecto de “camuflaje” de la información para evitar interceptaciones por parte de receptores no autorizados.

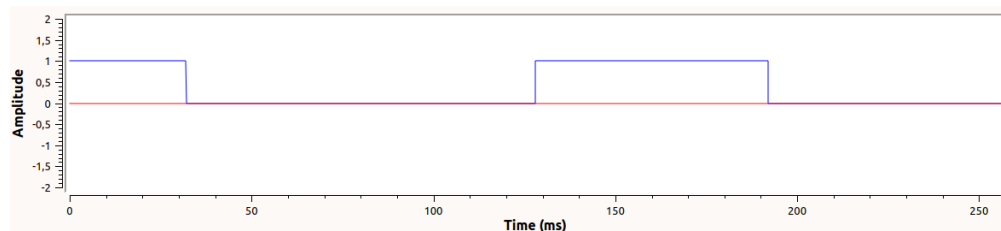
Figura 24. Señal ensanchada después de atravesar el canal Gaussiano



Fuente: elaboración propia.

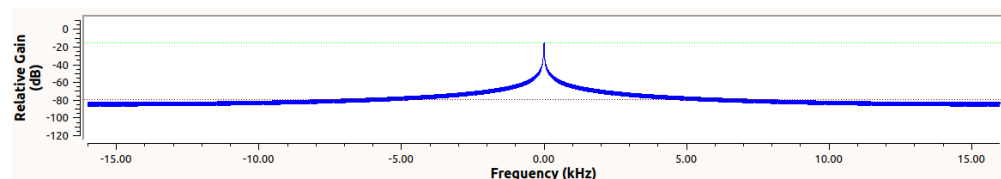
Finalmente, en el receptor después de demodular la señal para devolver los procesos realizados por el transmisor, se obtiene la secuencia de bits de salida (figura 25 y figura 26) exacta a la secuencia de bits (figura 18 y figura 19) generada por la fuente.

Figura 25. Señal final en el dominio del tiempo



Fuente: elaboración propia.

Figura 26. Señal final en el dominio de la frecuencia



Fuente: elaboración propia.

Cabe resaltar que la SNR (en dB), en el caso de no presentar ningún ruido en el canal (Voltaje de ruido = 0), tiene un valor muy grande de 133.40 dB, y la BER al no haber interferencias es de 0. Para un ruido gaussiano con voltaje de 5, la SNR es de -0,5 dB.

3. DISCUSIÓN

GNU Radio proporciona una plataforma flexible y modular para el diseño y evaluación de sistemas de comunicación digitales, permitiendo el desarrollo de bloques personalizados en Python y la modificación dinámica de parámetros durante la simulación. Esta característica fue fundamental en el desarrollo de los módulos presentados, ya que permitió observar en tiempo real el comportamiento espectral de las técnicas DSSS y FHSS, así como la influencia de parámetros como el tipo de modulación digital, la cantidad de chips por bit, la longitud del mensaje y el modelo de canal empleado.

Los bloques desarrollados permitieron visualizar claramente el ensanchamiento del espectro en ambas técnicas: en DSSS debido a la secuencia pseudoaleatoria (*spread code*) y en FHSS como resultado de los saltos de frecuencia gobernados por una tabla compartida. Esta visualización facilita la comprensión de los mecanismos de resistencia al ruido e interferencias, uno de los principales beneficios del espectro ensanchado.

Además, mediante el análisis de la tasa de error de bit (BER) en función de la relación señal a ruido (SNR), se evidenció cómo la robustez de cada técnica varía según la configuración utilizada. En escenarios con alta interferencia, DSSS mostró mejor desempeño cuando se aumentaba la longitud de la secuencia de chips, mientras que FHSS resultó más efectivo cuando el número de canales de salto y la velocidad de cambio eran adecuados frente a la tasa de datos.

La BER se utilizó como métrica integral para evaluar el sistema completo, incluyendo transmisor, canal y receptor. Esto permite identificar configuraciones más adecuadas para ambientes hostiles, así como entender el compromiso entre robustez y eficiencia espectral. En particular, se observó que DSSS puede ser más sensible a la sincronización de secuencia, mientras que FHSS presenta mayores requerimientos de ancho de banda en función de los saltos.

Finalmente, este trabajo demuestra que GNU Radio es una herramienta eficaz no solo para evaluar el comportamiento técnico de sistemas de espectro ensanchado, sino también como recurso didáctico en la formación en comunicaciones digitales. Su carácter libre y abierto lo convierte en una alternativa accesible frente a plataformas comerciales como MATLAB o LabVIEW, ofreciendo beneficios en términos de replicabilidad, extensión y aplicación en escenarios reales de radio definida por software (SDR).

REFERENCIAS

- [1] School of Electrical Engineering, Zhengzhou University, Zhengzhou, P.R. China, Z. Ma, T. Chen, M. Zhang, P. D. Kecerski, y S. Dang, “Literature Review of Spread Spectrum Signaling: Performance, Applications and Implementation”, *J. Commun.*, vol. 10, no 12, pp. 932-938, 2015, doi: 10.12720/jcm.10.12.932-938.
- [2] N. Stojanović, B. Todorović, V. Ristić, y I. Stojanović, “Direct sequence spread spectrum: History, principles and modern applications”, *Vojnoteh. Glas.*, vol. 72, n° 2, pp. 790–813, 2024, doi: 10.5937/vojtehg72-49325.
- [3] L. Shavanthi, K. Nagamani, y H. B. Sandya, “Implementation of DVB Standards using Software-Defined Radio”, en 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India: *IEEE*, may 2018, pp. 860–864. doi: 10.1109/RTEICT42901.2018.9012564.
- [4] Arboleda, Molina. O. Diseño e implementación de un módulo eficiente modulador/ demodulador QAM para GNU Radio. En R, Llamosa Villalba (Ed.). *Revista Gerencia Tecnológica Informática*, 15(43), 63-77, 2016.

- [5] F. Wunsch, K. Maier, H. Jäkel, y F. K. Jondral, “Implementation and Performance Evaluation of IEEE 802.15.4 LECIM DSSS PHY at 2.4 GHz”, *Proc. GNU Radio Conf.*, vol. 2, n° 1, Art. núm. 1, sep. 2017.
- [6] A. Sarbu, R. Papa, A. Digulescu, y C. Ioana, “A Software-Defined Radio Platform for Teaching Beamforming Principles”, *Appl. Sci.*, vol. 14, n° 22, p. 10386, nov. 2024, doi: 10.3390/app142210386.
- [7] S. Zhang, F. Liu, Y. Huang, y X. Meng, “Adaptive Detection of Direct-Sequence Spread-Spectrum Signals Based on Knowledge-Enhanced Compressive Measurements and Artificial Neural Networks”, *Sensors*, vol. 21, n° 7, Art. núm. 7, ene. 2021, doi: 10.3390/s21072538.
- [8] D. J. Torrieri, *Principles of Spread-Spectrum Communication Systems*, 5th ed. Cham, Switzerland: Springer, 2022.
- [9] A. Artés Rodríguez, F. Pérez González, J. Cid Sueiro, R. López Valcarce, C. Mosquera Nartallo, y F. Pérez Cruz, *Comunicaciones digitales*. Madrid, España: Pearson Educación, 2007.