

Laboratorio 4: BER en señalización de banda base

Sección 2

Camilo Antonio Rios Tenderini—Camilo.rios1@mail.udp.cl

Diego Alexander Hidalgo Gallardo—Diego.hidalgo1@mail.udp.cl

Renato Antonio Yáñez Riveros—Renato.yanez2@mail.udp.cl

Fecha: Junio - 2025

Profesor Asignado:
Marcos Fantoval

Índice

1. Introducción	1
2. Antecedentes	2
2.1. Modulación PSK (Phase Shift Keying)	2
2.2. Variantes de PSK	2
2.2.1. BPSK (2-PSK)	2
2.2.2. QPSK (4-PSK)	2
2.2.3. 8-PSK	2
3. Metodología	3
3.1. Conceptos básicos	3
3.2. Bloques de GNU Radio utilizados	3
3.3. Métodos de la clase <code>gr::digital::constellation</code> . .	4
3.4. Construcción de transmisores PSK.	4
3.4.1. Transmisor BPSK	4
3.4.2. Transmisores QPSK y 8PSK	5
3.4.3. Cálculo BER	6
4. Resultados	7
5. Análisis de Resultados	9
5.1. Transmisores	9
5.2. Bit Error Rate	9
6. Conclusiones	10

1. Introducción

En este laboratorio se explora cómo transmitir datos de manera efectiva a través de un canal, en el cual la presencia de ruido introduce distorsiones que pueden comprometer la integridad de la información transmitida. Para lograrlo, implementamos distintas técnicas de modulación digital, optimizando la eficiencia espectral y la robustez frente a interferencias.

Se analizará su funcionamiento en diferentes aplicaciones y condiciones, comenzando con la modulación por desplazamiento de fase (PSK), debido a su eficiencia y simplicidad. Posteriormente, se estudiará en profundidad el comportamiento de BPSK, QPSK y 8PSK, que difieren en la cantidad de bits transmitidos por símbolo y en su resistencia al ruido.

El objetivo principal fue evaluar el desempeño de cada técnica en función de la tasa de error de bits (BER), que representa la métrica fundamental para cuantificar el rendimiento, al modificar la relación señal-ruido. Para ello, se utilizó la herramienta GNU Radio, que permitió modelar, simular y visualizar estos sistemas de comunicación digital en condiciones cercanas a las reales.

2. Antecedentes

2.1. Modulación PSK (Phase Shift Keying)

La modulación por desplazamiento de fase (PSK) es una técnica donde la información se codifica en cambios de fase de una señal portadora. La cantidad de fases diferentes que puede tomar la señal determina cuántos bits pueden representarse por cada símbolo transmitido. A mayor número de fases, mayor eficiencia (envía mas bits a la vez) pero también mayor probabilidad de verse afectada por el ruido.

2.2. Variantes de PSK

2.2.1. BPSK (2-PSK)

- 2 fases (0° y 180°)
- 1 Bit
- Puede operar con SNR baja

2.2.2. QPSK (4-PSK)

- 4 fases (cada 90°)
- 2 Bits
- Requiere el doble de potencia (3dB) que BPSK para lograr el mismo BER

2.2.3. 8-PSK

- 8 fases (cada 45°)
- 3 Bits
- Requiere el doble de potencia (3dB) que 4-PSK para lograr el mismo BER

3. Metodología

En la experiencia se utiliza GNU Radio para realizar la simulación de un sistema de comunicaciones digitales. A continuación, se explican los conceptos y bloques fundamentales utilizados en el laboratorio:

3.1. Conceptos básicos

Byte Pack y Byte Unpack en GNU Radio:

- **Byte Pack:** Convierte un flujo de bits en bytes (agrupando 8 bits consecutivos en un byte). Es útil para preparar datos para su transmisión o procesamiento en formatos byte-oriented.
- **Byte Unpack:** Realiza la operación inversa, desagrupando bytes en bits individuales. Esto es esencial para procesar datos recibidos a nivel de bits.

3.2. Bloques de GNU Radio utilizados

- **Random Source:** Genera una secuencia aleatoria de bits, bytes o números enteros. En este laboratorio se utiliza para simular la fuente de información binaria que será modulada y transmitida.
- **Chunks to Symbols:** Convierte grupos de bits (chunks) en símbolos de acuerdo a un esquema de modulación específico (BPSK, QPSK, 8-PSK). Mapea cada combinación de bits a un punto específico en la constelación.
- **Noise Source:** Genera ruido blanco gaussiano (AWGN) que se añade a la señal para simular un canal realista. La amplitud del ruido puede ajustarse para variar la relación E_b/N_0 .
- **Constellation Decoder:** Realiza la operación inversa a Chunks to Symbols, decodificando los símbolos recibidos (afectados por ruido) de vuelta a bits. Utiliza decisión por máxima verosimilitud para determinar el símbolo transmitido.
- **BER (Bit Error Rate):** Calcula la tasa de error de bits comparando el flujo de bits transmitidos con los recibidos. Proporciona estadísticas en tiempo real del desempeño del sistema.
- **QT GUI Number Sink:** Muestra valores numéricos en una interfaz gráfica. En este laboratorio se utiliza para visualizar métricas como el BER calculado.
- **Throttle:** Permite limitar la frecuencia de muestreo para no saturar el programa.

3.3. Métodos de la clase `gr::digital::constellation`

- **`arity()`**: Devuelve el número de puntos en la constelación (M para modulación M-aria).
- **`points()`**: Retorna un arreglo con las coordenadas (valores complejos) de los puntos de la constelación.
- **`base()`**: Devuelve el objeto constellation base que puede ser utilizado para crear constelaciones derivadas.
- **`bits_per_symbol()`**: Indica el número de bits representados por cada símbolo de la constelación ($\log_2(M)$).

3.4. Construcción de transmisores PSK.

3.4.1. Transmisor BPSK

Para construir el transmisor BPSK, se utilizó la siguiente configuración en GNU radio.

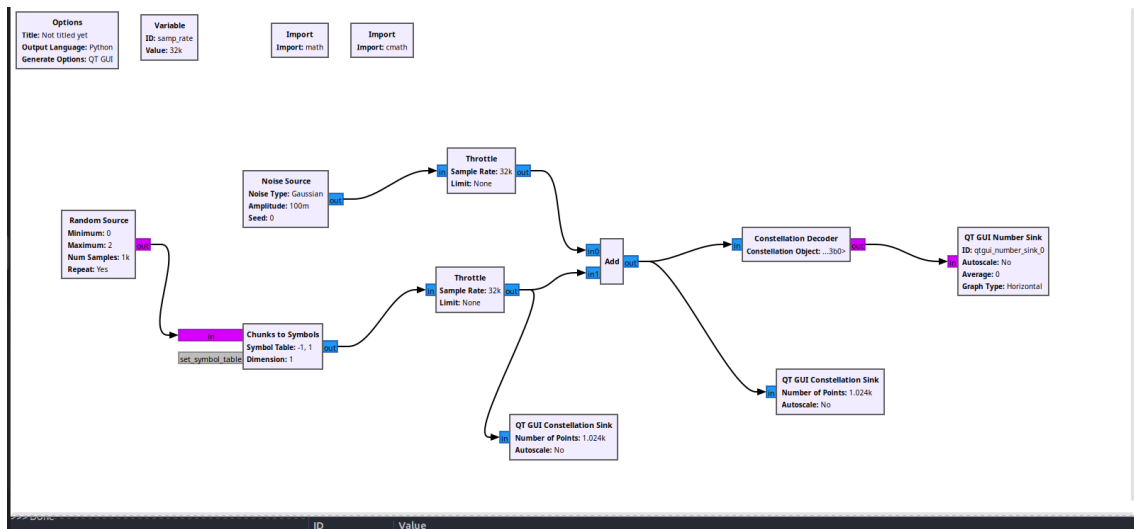


Figura 1: Diagrama de bloques en GNU Radio

El diagrama anterior, se encarga de seguir el siguiente flujo para la correcta implementación del transmisor BPSK:

- **Random Source**

- Genera bits aleatorios para transmitir, simulando ser los datos de entrada.
- Se configura para obtener una secuencia binaria de salida.

- **Chunks to Symbols**

- Convierte los bits generados en símbolos BPSK.
- Define la constelación BPSK asignando una tabla con:
 - Bit 0 → fase de 180 grados.
 - Bit 1 → fase de 0 grados.

- **Noise Source: Ruido Gaussiano**

- Se añade ruido gaussiano al sistema para simular perturbaciones aleatorias típicas en canales reales.
- Este ruido sigue una distribución normal y se configura con una amplitud de 100 milivoltios, lo que permite mantener la prueba lo suficientemente limpia para observar los efectos sin distorsionar excesivamente la señal.

- **Constellation Encoder**

- Codifica los símbolos de entrada según una constelación previamente definida.
- En este caso, basta con configurarlo utilizando `digital.constellation_bpsk()`.

3.4.2. Transmisores QPSK y 8PSK

Para **QPSK** y **8PSK**, se sigue el mismo diagrama mencionado anteriormente, realizando principalmente cambios en los bloques **Chunks to Symbols** y **Constellation Encoder**.

- **QPSK**

- En **Chunks to Symbols**, se configura la tabla con:

$$\left\{ \frac{1+j}{\sqrt{2}}, \frac{-1+j}{\sqrt{2}}, \frac{-1-j}{\sqrt{2}}, \frac{1-j}{\sqrt{2}} \right\}$$

- *Representando los cuatro símbolos de QPSK distribuidos uniformemente en el plano complejo.*

- Se asigna el codificador de constelación con: `digital.constellation_qpsk()`.

■ 8PSK

- En Chunks to Symbols, se utiliza:

$$\left\{ e^{j \cdot \frac{(2k+1)\pi}{8}} \mid k = 0, 1, \dots, 7 \right\}$$

- Esta expresión genera 8 símbolos equiespaciados en un círculo unitario, representando la constelación 8PSK.
- Se asigna el codificador de constelación con: `digital.constellation_8psk()`.

3.4.3. Cálculo BER

Adicionalmente, se utiliza el bloque BER de GNU para verificar el factor que representa el valor del Bit Error Rate:

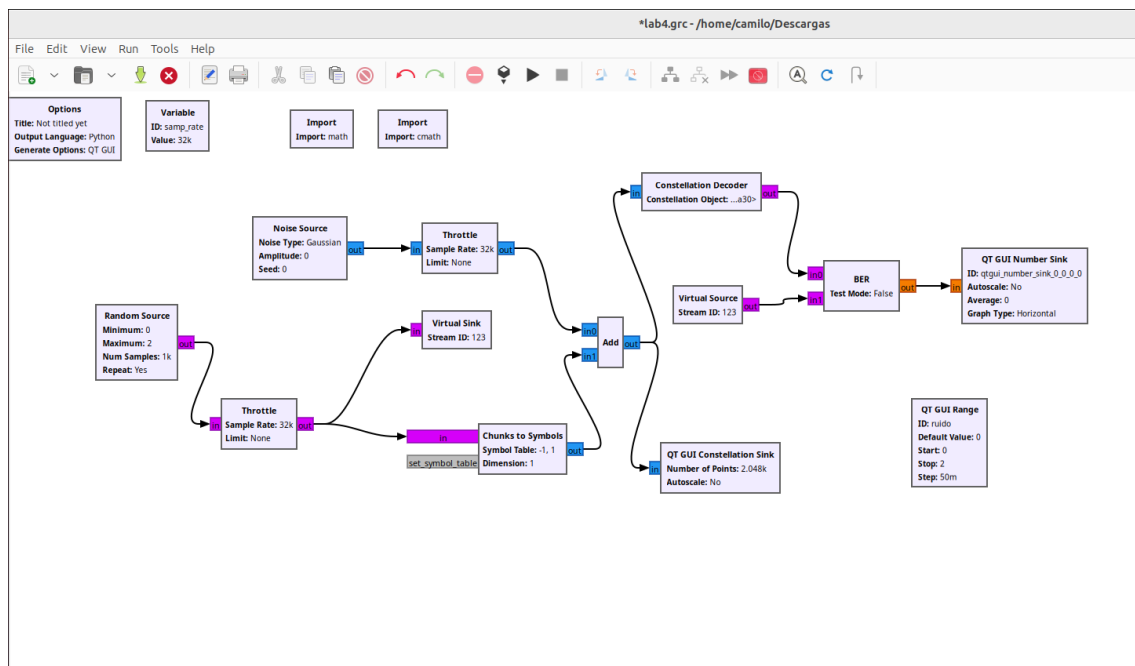


Figura 2: Diagrama de bloques en GNU.

Finalmente, para calcular el BER «experimental», se variaba la amplitud del ruido según el valor de BER a calcular, registrando los resultados manualmente para luego graficarlos y compararlos con el BER teórico.

4. Resultados

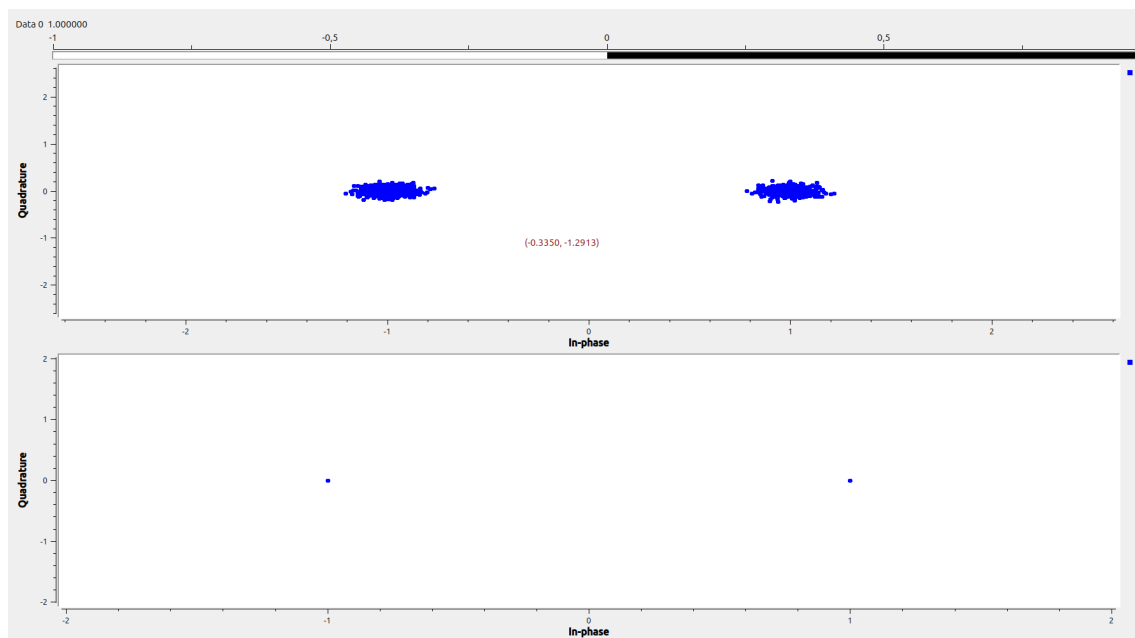


Figura 3: Ejecución de BPSK en GNU radio.

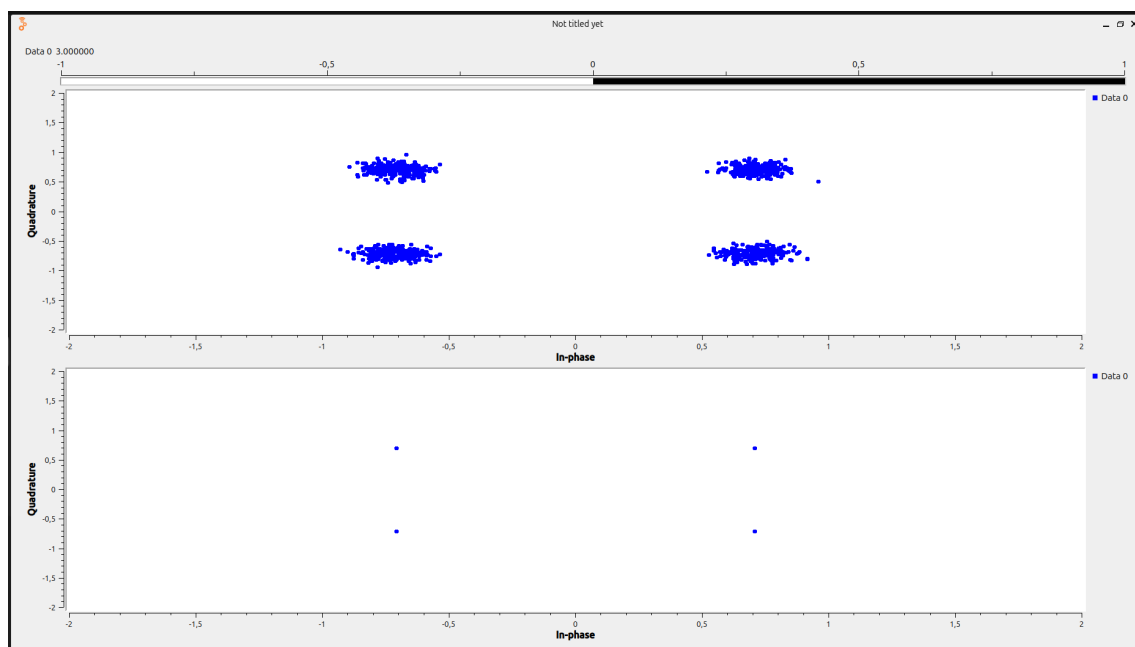


Figura 4: Ejecución de QPSK en GNU radio.

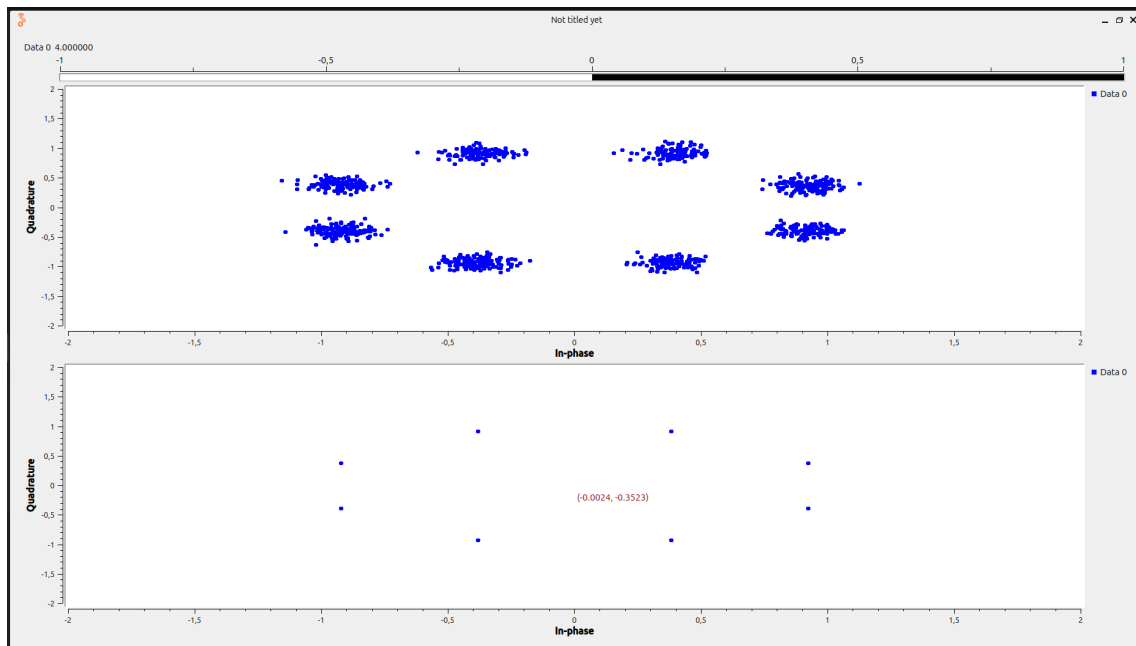


Figura 5: Ejecución de 8PSK en GNU radio.

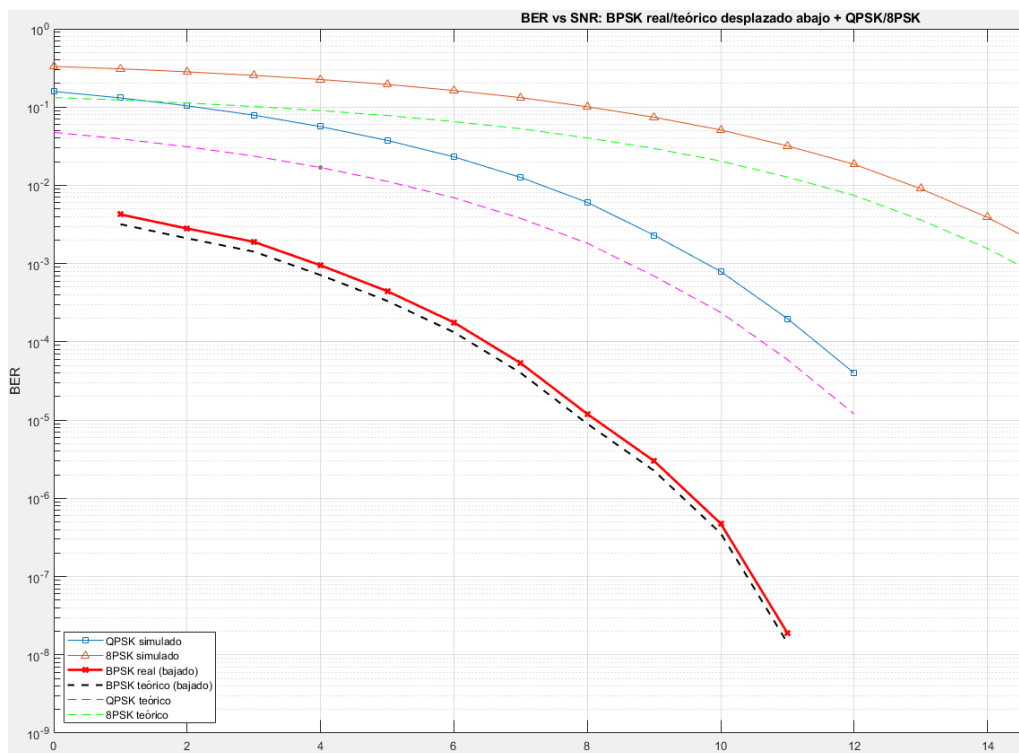


Figura 6: Comparativa entre BER teórico vs BER calculado en GNU.

5. Análisis de Resultados

5.1. Transmisores

En las ejecuciones de los transmisores, se logra apreciar correctamente la constelación de cada caso, reproduciendo con éxito el comportamiento esperado en la realidad.

- **BPSK:** Se observa claramente que solo existen dos ángulos posibles. A medida que se incrementa el ruido, los puntos se agrupan en torno a cada uno de estos, generando dos concentraciones bien definidas.
- **QPSK:** Se repite el mismo fenómeno, esta vez con cuatro ángulos equiespaciados. Los puntos se distribuyen en torno a ellos, lo que facilita distinguir la cuadratura de fase.
- **8PSK:** Del mismo modo, se identifican ocho posiciones angulares. Aunque los puntos están más juntos entre sí que en QPSK, la estructura circular de la constelación se mantiene reconocible incluso con presencia de ruido.

5.2. Bit Error Rate

Como se observa en el gráfico de BER, a medida que se aumenta la cantidad de fases, se necesita más energía para mantener el mismo valor de BER. Esto se debe a la naturaleza del funcionamiento de las modulaciones, ya que al agregar más fases, se disminuye el ángulo entre cada una de las fases, lo que permite enviar una mayor cantidad de información pero también aumenta la probabilidad de que el mensaje enviado se vea afectado por el ruido.

Asimismo, al realizar y registrar en el gráfico los valores de BER experimentales para cada tipo de modulación, los resultados no son completamente idénticos. Una posible explicación para este fenómeno está relacionada con que GNU Radio utiliza métodos internos de generación de ruido y procesamiento que pueden introducir pequeñas variaciones en la simulación, afectando así la precisión y consistencia de los resultados obtenidos.

6. Conclusiones

El laboratorio permitió observar de manera práctica cómo distintas modulaciones por desplazamiento de fase (PSK) ofrecen diferentes niveles de eficiencia y resistencia frente al ruido en sistemas de comunicaciones. Los resultados obtenidos confirmaron las predicciones teóricas sobre el comportamiento de cada esquema de modulación.

BPSK demostró ser la modulación más robusta, manteniendo una clara separación entre símbolos incluso ante niveles significativos de ruido e interferencia. Su simplicidad la convierte en la opción ideal para canales altamente ruidosos, donde la prioridad es minimizar la tasa de error de bits, aunque esto implique sacrificar velocidad de transmisión.

Por su parte, 8PSK mostró la mayor eficiencia espectral al transmitir tres bits por símbolo, pero demostró una mayor vulnerabilidad al ruido debido a la reducida distancia entre puntos de constelación. Esta modulación resulta adecuada únicamente en canales con excelente relación señal-ruido, o cuando se implementan técnicas adicionales de corrección de errores. QPSK, en cambio, demostró ser una alternativa intermedia, al equilibrar confiabilidad y rendimiento.

La plataforma GNU Radio fue una herramienta eficaz para el análisis y comprensión de los sistemas de comunicaciones, ya que facilitó la visualización de los efectos del ruido y la medición precisa de los parámetros de desempeño de cada esquema mediante el cálculo del BER.

Referencias

- [1] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab,
Signals and Systems,
2nd ed., Prentice-Hall, 1996.
- [2] J. G. Proakis and D. G. Manolakis,
Digital Signal Processing: Principles, Algorithms, and Applications,
4th ed., Prentice-Hall, 2006.
- [3] GNU Radio Wiki,
“BER - Bit Error Rate,”
[https://wiki-gnuradio-org.translate.goog/index.php/BER?_x_tr_sl=](https://wiki-gnuradio-org.translate.goog/index.php/BER?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc)
[en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc](https://wiki-gnuradio-org.translate.goog/index.php/BER?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc)