

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,
INFORMÁTICA Y MECÁNICA

ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA



CONTROL ADAPTATIVO DE LA POTENCIA TRANSMITIDA POR UN NODO INALÁMBRICO UTILIZANDO ALGORITMOS DE ESTIMACIÓN DE LA RELACIÓN SEÑAL A RUIDO

TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO ELECTRÓNICO

PRESENTADO POR: Bach. MICHEL CHINO QUISPE

ASESOR: Mgt. Ing. JORGE LUIS ARIZACA CUSICUNA

Tesis financiada mediante resolución N°-1033-2019-UNSAAC

Cusco - Perú
2021

DEDICATORIA

Esta tesis es dedicada a mis padres y familia por la constante confianza y apoyo que me han dado.

ÍNDICE

RESUMEN	VIII
INTRODUCCIÓN	IX
I. PLANTEAMIENTO DE LA INVESTIGACIÓN	10
1.1. Planteamiento del Problema.	10
1.1.1. Problemática.	10
1.1.2. Problema General.	11
1.1.3. Problemas específicos.	11
1.1.4. Justificación.	12
1.2. Objetivo.	13
1.2.1. Objetivo General.	13
1.2.2. Objetivos Específicos.	13
1.3. Metodología.	13
1.3.1. Tipo de investigación	14
1.3.2. Población y muestra	14
1.3.3. Variables.	15
1.3.4. Técnicas e instrumentos.	15
1.3.5. Alcance del método.	15
1.3.6. Delimitaciones.	16
II. MARCO TEÓRICO.	17
2.1. Antecedentes.	17
2.2. Redes de sensores.	19
2.2.1. Aplicaciones.	19
2.2.2. Características de la redes de sensores.	20
2.3. Sistemas de Transmisiones Adaptativas	23
2.3.1. Técnicas de tasa variable	25
2.3.2. Técnicas de potencia variable.	25
2.3.3. Probabilidad de error variable	26
2.4. Control PID.	27
2.4.1. Control de tres términos.	27
2.5. Variables aleatorias.	29
2.5.1. Función de distribución acumulativa.	30
2.5.2. Función de densidad de probabilidad.	30
2.5.3. Valor esperado (valor medio).	31
2.5.4. Varianza.	31
2.5.5. Momentos.	31
2.5.6. Función característica.	31
2.5.7. Importantes funciones de densidad de probabilidad.	32

2.6.	GNU Radio Companion (GRC).	34
2.6.1.	Entorno de GNU Radio Companion (GRC).	35
2.6.2.	Diagrama de flujo en el entorno de GRC.	39
2.7.	Radio Definido por Software SDR.	41
2.7.1.	Sistemas de Software Definido por Radio.	42
2.7.2.	Universal Software Radio Peripheral (USRP).	44
III. ANÁLISIS DEL CANAL DE TRANSMISIÓN		49
3.1.	Canal Gaussiano.	49
3.1.1.	Ruido blanco Gaussiano aditivo (AWGN).	50
3.1.2.	Relación señal a ruido (SNR).	51
3.2.	Algoritmos de estimación de la relación señal a ruido.	55
3.2.1.	Modelo de la señal y estimación de la SNR.	56
3.2.2.	Algoritmo de segundo y cuarto momento M_2M_4 .	57
3.2.3.	Algoritmo propuesto por esha Ijaz, Adegbenga B. Awoseyila, and Barry G. Evans.	57
3.2.4.	Estimador de relación señal-variación (SVR).	58
3.2.5.	Algoritmo DDML.	59
3.2.6.	Comparación de Rendimiento.	60
3.2.7.	Diagrama flujo de la implementación de los algoritmos de estimación de la SNR.	62
3.2.8.	Resultado de la simulación de los algoritmos de estimación de la SNR.	62
3.3.	Implementación de los algoritmos de estimación de la SNR en el entorno de GRC.	67
3.3.1.	Diagrama de flujo en el entorno de GRC para las pruebas de los algoritmos de estimación de la SNR.	67
3.3.2.	Resultados de los algoritmos de estimación de la SNR en el entorno de GRC.	68
IV. DISEÑO E IMPLEMENTACIÓN DE LOS SISTEMAS EN PLATAFORMA DE LA SDR		72
4.1.	Implementación del transmisor y receptor en el sistema de comunicación.	72
4.1.1.	Diseño del sistema de comunicación digital.	73
4.1.2.	Implementación del Transmisor QPSK en el entorno de GRC.	73
4.1.3.	Implementación del Receptor QPSK en el entorno de GRC.	75
4.2.	Implementación del algoritmo de estimación de la SNR en el sistema de comunicación.	77
4.2.1.	Diseño para un enlace de transmisión cableada.	77
4.2.2.	Diseño de un enlace de transmisión inalámbrica.	78
4.3.	Implementación del cálculo de la tasa de error de bit en el sistema.	79
4.3.1.	Probabilidad de error y tasa de error de bit.	79
4.3.2.	Implementación del cálculo de la tasa de error de bit.	81
4.4.	Diseño e implementación del sistema de control adaptativo de la potencia transmitida.	82
4.4.1.	Diseño del sistema de control adaptativo de potencia transmitida.	83
4.4.2.	Transmisión full dúplex	83
4.4.3.	Requisitos S/N de la PSK M-aria para tasa de errores fijas.	84
4.4.4.	Implementación del primer escenario.	85

4.4.5. Implementación del segundo escenario.	88
V. PRUEBAS Y RESULTADOS DE LOS SISTEMAS IMPLEMENTADOS.	94
5.1. Transmisor y receptor del sistema de comunicación.	94
5.1.1. Transmisor del sistema de comunicación.	94
5.1.2. Receptor del sistema de comunicación.	95
5.2. Implementación del algoritmo estimador M_2M_4 de la SNR en el sistema de comunicación.	97
5.3. Probabilidad de error y Tasa de errores de bit.	98
5.4. Pruebas y resultados del sistema de Control Adaptativo de potencia transmitida.	99
5.4.1. Primer escenario (Enlace guiado).	99
5.4.2. Segundo escenario (Enlace inalámbrico).	101
Discusiones	107
Análisis e interpretación de los resultados	107
Conclusiones	108
Recomendaciones	110
Referencias	111
Anexos	112
A. Estudio de canal de transmisión	113
1.1. Código para simulación del canal AWGN.	113
1.2. Código para simulación de algoritmos de estimación de la SNR.	114
B. Implementación de los algoritmos de la SNR	122
2.1. Código del algoritmo SVR de estimación de la SNR	122
2.2. Código del algoritmo M_2M_4 de estimación de la SNR	124
2.3. Implementación del sistema de comunicación para modulación BPSK.	126
2.3.1. Transmisor BPSK en plataforma de SDR.	126
2.3.2. Receptor BPSK en plataforma de SDR	128
C. Pruebas y Resultados	130
3.1. Pruebas de la estimación del algoritmo M_2M_4	130
3.1.1. imágenes de Pruebas de estimación de la SNR del canal Guiado.	130
3.1.2. Prueba de estimación de la SNR del canal Enlace inalámbrico.	131
3.2. Código del cálculo de probabilidad de error y tasa de error de bit.	131
3.3. Pruebas del sistema de control adaptativo de potencia transmitida.	133
3.3.1. Código de la asignación de amplitud al nodo sensor en un enlace guiado.	133
3.3.2. Código de la asignación de potencia transmitida al nodo sensor en un enlace inalámbrico.	134
3.3.3. Estimación de la SNR del canal de transmisión en espacio Libre.	137
3.3.4. Sistema de control adaptativo de potencia transmitida para la modulación BPSK.	138

Índice de tablas

2.1. Especificaciones del transmisor de USRP NI 2920.	46
2.2. Especificaciones del receptor de USRP NI 2920.	47
4.1. Requisitos de S/N en PSK M-aria para tasas de errores fijas.	84
5.1. Resultado de la estimación de la SNR y Cálculo de la probabilidad de error. . .	98
5.2. Asignación de amplitud para determinado rango de niveles de SNR	100

Índice de figuras

2.1. Distintos tipos de topologías utilizados en redes de sensores (Vizcaíno, 20114).	20
2.2. Pila de protocolos del modelo OSI (izquierda), y diferentes implementaciones en sistemas de comunicación inalámbrica de alta frecuencia (Vizcaíno, 20114).	21
2.3. Modelo del Sistema de transmisión adaptativa (Goldsmith, 2005).	24
2.4. Entradas y salidas del controlador (Rivera, 2006).	27
2.5. Entradas y salidas del controlador (Rivera, 2006).	29
2.6. Interfaz de GNU Radio Companion (GRC).	35
2.7. Liberia GRC.	36
2.8. Propiedades de bloque de GRC.	37
2.9. Documentación de los bloque de GRC	38
2.10. Gráfico de flujo GRC.	39
2.11. Barra de herramientas GRC.	40
2.12. Señal sinusoidal.	40
2.13. Tipos de datos GRC.	41
2.14. Sistema de un SDR ideal.	42
2.15. Estructura de SDR para un transmisor (Sabate, 2016).	43
2.16. Estructura de SDR para un receptor (Sabate, 2016).	43
2.17. Diagrama de bloque del USRP NI 2920 (Instrument, 2013)	45
3.1. Función de densidad de probabilidad para una variable aleatoria Gaussiana de $N \sim (0, 1)$.	50
3.2. Rango dinámico de la relación señal a ruido.	51
3.3. Diagrama del modelo de canal AWGN.	53
3.4. Señal de información $f(n)$.	53
3.5. Señal no deseada AWGN.	53
3.6. Histograma de la señal no deseada AWGN.	54
3.7. Señal de información afectado por AWGN, para SNR = 2 dB.	54
3.8. Señal de información afectado por AWGN, para SNR = 10 dB.	55
3.9. Señal de información afectado por AWGN, para SNR = 30 dB.	55
3.10. Diagrama de flujo de la simulación de los algoritmos de estimación de la SNR, para determinar el algoritmo de mejor desempeño.	62
3.11. Diagrama de constelación de BPSK.	63
3.12. (a) Diagrama de constelación de BPSK, (b) Diagrama de constelación de BPSK con AWGN.	63
3.13. Rendimiento de los algoritmos de estimación de la SNR para la modulación digital BPSK.	64
3.14. Diagrama de constelación de QPSK	65
3.15. (a) Diagrama de constelación de QPSK, (b) Diagrama de constelación de QPSK con AWGN.	65

3.16. Rendimiento de algoritmo de estimación la SNR de canal AWGN.	66
3.17. Python Block GRC.	67
3.18. Entorno gráfico de GRC, para pruebas de estimación de la SNR.	67
3.19. Estimador de la SNR GRC propio del entorno estima SNR= 2.15 dB	69
3.20. Algoritmo de estimación de SVR , estima SNR = 1.79 dB	69
3.21. Algoritmo de estimación de M_2M_4 , estima SNR = 2.33 dB	69
3.22. Estimador del entorno GRC, estima SNR = 17.81 dB	70
3.23. Algoritmo de estimación de SVR , estima SNR =18.006 dB	70
3.24. Algoritmo de estimación M_2M_4 , estima SNR = 18.14 dB	70
4.1. Diagrama de bloque del transmisor y receptor de un enlace de transmisión inalámbrica	73
4.2. Diagrama de bloque de un transmisor QPSK en GRC.	73
4.3. Diagrama de bloque del receptor QPSK en GRC.	75
4.4. Diagrama de bloque de un enlace cableado para la transferencia de información.	77
4.5. Diagrama de bloque de un enlace inalámbrico para la transferencia de información.	78
4.6. Diagrama de bloque del receptor QPSK en GRC.	78
4.7. Diagrama de bloque del sistema de comunicación.	79
4.8. Probabilidad de error P_e	80
4.9. Diagrama de bloque de trasmisor QPSK en GRC.	81
4.10. Diagrama de bloque de receptor QPSK en GRC.	82
4.11. Diagrama de bloque del control adaptativo de potencia transmitida	83
4.12. Diagrama de bloque del modo de transmisión full dúplex	84
4.13. Primer escenario del sistema de control adaptativo de potencia transmitida.	85
4.14. Diagrama de flujo para la asignación de amplitud al nodo sensor.	86
4.15. Diagrama de bloque en el entorno de GRC que del Nodo sensor.	87
4.16. Diagrama de bloque en el de GRC de la estación base.	88
4.17. Segundo escenario del sistema de control adaptativo de potencia transmitida.	89
4.18. Modulación de OFDM, para generar ruido AWGN.	90
4.19. Esquema del controlador PID del sistema comunicación.	90
4.20. Diagrama de bloque del nodo sensor en el entorno GRC del segundo escenario.	91
4.21. Diagrama de bloque de la estación base en el entorno GRC del segundo escenario.	91
4.22. Esquema de la implementación del segundo escenario dels sistema de control adaptativo de la potencia transmitida.	93
4.23. Lugar de pruebas, distancia entre transmisor y receptor $d = 14$ m para el envío de información de datos de bit aleatorio.	93
5.1. Espectro de la señal transmitida en la banda de 915 MHz.	95
5.2. Diagrama de constelación de la modulación QPSK.	95
5.3. Espectro de la señal recibida.	96
5.4. Constelación de señal demodulada QPSK	96
5.5. Comparación del NMSE para el algoritmo M_2M_4 en tres casos.	97
5.6. Comparación de la probabilidad de error de bit.	98
5.7. Asignación de amplitud A_{Tx} para determinado nivel de SNR estimado.	99
5.8. Espectro del ruido AWGN	101
5.9. Nodo sensor, transmisor BPSK y asignación de potencia transmitida.	102
5.10. Estación base, receptor BPSK y estimación de la SNR del canal.	102

5.11. Entorno gráfico de GRC, del sistema de control adaptativo de potencia transmitida de la segunda prueba.	103
5.12. Entorno gráfico de GRC, del sistema de control adaptativo de potencia transmitida de la tercera prueba.	103
5.13. Nodo sensor, transmisor QPSK y asignación de potencia transmitida	104
5.14. Estación base, receptor QPSK y estimación de la SNR del canal.	105
5.15. Entorno gráfico del entorno GRC, del sistema de control adaptativo de la potencia transmitida, de la segunda prueba en una modulación QPSK	106
5.16. Entorno gráfico del entorno GRC, del sistema de control adaptativo de la potencia transmitida, de la tercera prueba en una modulación QPSK	106
2.1. Diagrama de bloque del transmisor BPSK en GRC.	126
2.2. Constelación de BPSK del transmisor.	127
2.3. Espectro de la información transmitida de BPSK.	127
2.5. Constelación de BPSK del receptor.	128
2.4. Diagrama de bloque de receptor BPSK en GRC.	128
2.6. Espectro de la información recibida de BPSK.	129
3.1. Estimación de la SNR para un enlace guiado.	130
3.2. Distancia entre antena transmisora y receptora $d = 1\text{m}$ del sistema de comunicación.	131
3.3. Información recibida en el dominio del tiempo QPSK, con $G_{tx} = 1\text{dB}$ y $G_{rx} = 10\text{dB}$	137
3.4. Diagrama de constelación y espectro de la señal recibida QPSK.	137
3.5. Diagrama de bloque del nodo sensor para modulación BPSK.	138
3.6. Diagrama de bloque de la estación base para modulación BPSK	138

RESUMEN

En este trabajo se diseña e implementa un sistema de control adaptativo de la potencia transmitida por un nodo inalámbrico, utilizando algoritmos de estimación de la relación señal a ruido (SNR), basado en la tecnología emergente conocida como Radio Definida por Software (SDR).

En primer lugar, se realiza el estudio del comportamiento y los parámetros que intervienen en un canal de transmisión, se sabe que uno de los parámetros importantes de un sistema de comunicación es la relación señal a ruido (SNR), para el cual se examina los algoritmos de estimación de la SNR en las modulaciones digitales BPSK Y QPSK. Para evaluar la relación señal a ruido del canal de transmisión, se selecciona cuatro algoritmos de estimación de la SNR, los cuales se comparan con la finalidad de verificar que algoritmo ofrece mejor desempeño con respecto al otro, posteriormente este algoritmo se implementa en el entorno de GNU Radio Companion (GRC). Seguidamente, se estudia la tecnología radio definida por software (SDR) con el fin de conocer los fundamentos de esta tecnología; donde se indaga sus principios, así como la arquitectura y los componentes físicos que la conforman. Una vez comprendida se hará uso de esta tecnología para diseñar e implementar un sistema de comunicación inalámbrica para las modulaciones BPSK Y QPSK, además en este sistema se implementa el algoritmo de estimación de la SNR en la etapa de recepción.

Finalmente, se diseña e implementa el sistema de control adaptativo de la potencia transmitida en la plataforma de la SDR. Para ello se utiliza la comunicación full dúplex en la adaptación de la potencia transmitida en el nodo sensor. Cabe resaltar que en este trabajo se utiliza las facultades del controlador PID, con el fin de aumentar o disminuir la potencia del nodo sensor, con lo cual se consigue el sistema de adaptación. Además indicar que el termino control adaptativo no se refiere a la técnica utilizada en automatización.

INTRODUCCIÓN

Las redes de sensores inalámbricas es un tema reciente de investigación, actualmente constituyen una tecnología prometedora en el campo de agricultura de precisión, están formados por un conjunto de nodos que realizan una tarea particular. Los nodos a menudo son pequeños dispositivos electrónicos, autónomos, funcionan con batería y pueden comunicarse entre si de forma inalámbrica, las diferentes aplicaciones con estos dispositivos exigen una adecuada gestión de la energía. Con el fin de mejorar su autonomía. Estas redes inalámbricas se caracterizan por utilizar el medio de transmisión dinámico, donde las condiciones del canal de cambian de forma continua, este cambio continuo lleva a diseñar sistemas más inteligentes.

Uno de los inconvenientes más inmediatos del canal, es cuando hay mayor atenuación en el medio de transmisión, bajo esta situación el sistema de comunicación es sobredimensionado en potencia transmitida, con la finalidad de garantizar un enlace fiable de transferencia de datos, esto causa mayor consumo de energía, por lo tanto menor duración en las fuentes recargables. Frente a este enfoque en la presente tesis se diseña e implementa un sistema de control adaptativo de la potencia transmitida por un nodo inalámbrico, utilizando los algoritmos de estimación de la relación señal a ruido, en la plataforma de Radio Definida por Software (SDR); la transmisión adaptativa en potencia permite adecuar la potencia transmitida a las condiciones del canal transmisión. Uno de los parámetros importantes a considerar es la relación señal a ruido (SNR), ya que varios de las otras medidas dependen directamente de su conocimiento para ser evaluadas, en este caso el interés es por la potencia transmitida, una vez obtenida la información del canal, ésta será suministrada al transmisor del sistema, donde el nodo sensor toma la decisión de modificar la potencia transmitida, con el objetivo de mantener la SNR constante en el receptor, de esta manera el sistema podría garantizar un enlace fiable y contribuir en el uso eficiente de la energía.

Capítulo I

PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1. Planteamiento del Problema.

1.1.1. Problemática.

El creciente desarrollo de los dispositivos inalámbricos de comunicación, especialmente en los nodos inalámbricos tiene diversas aplicaciones en los campos de producción y desarrollo, por otra parte la demanda de estos dispositivos en el sector de la agricultura es enorme, es decir, para monitorear las enfermedades en las plantas y cultivos, ataques de insectos y las anomalías climática. De ahí que estos sistemas normalmente están instalados en zonas de difícil acceso, y trabajan de manera independiente y cooperativa. Los nodos sensores tienen una serie de limitaciones, necesitan ser pequeños para ser colocados en cualquier lugar, necesitan poder comunicarse para enviar y recibir información, y su autonomía energética es limitada, además tienen la capacidad de sensado, procesamiento y de comunicación.

Estos sistemas son instalados en una región específica, con una potencia de transmisión definida para el peor caso. Este parámetro fija su consumo energético, lo que puede llegar a limitar su operatividad. En algunos casos donde se requiere tiempos grandes de operación se puede perjudicar su funcionamiento.

1.1.2. Problema General.

Por lo tanto de lo anterior:

Los nodos son instalados en un campo de difícil acceso, transmiten a una potencia fija en canales con alto o bajo nivel de atenuación, esto se traduce en que estos sistemas hacen uso ineficiente de su energía. Si estos sistemas tendrían un sistema de control adaptativo de la potencia transmitida, pueden mejorar su autonomía energética haciendo uso eficiente de su energía.

1.1.3. Problemas específicos.

- Los sistemas de comunicaciones en redes de inalámbricas experimentan problemas energéticos, cuando el canal de transmisión cambia dinámicamente con respecto a las condiciones ambientales. Lo que podría generar las pérdidas de información suministradas por el nodo sensor a la estación base.
- La falta de implementación de diversas técnicas que mejoran el uso óptimo de la energía en las redes inalámbricas, hace que la comunicación inalámbrica sea vulnerable ante cualquier cambio brusco en el canal de transmisión.
- La potencia de transmisión en algunos nodos es fija, debido a la sobredimensión del sistema, en condiciones ambientales variables.
- En las zonas lejanas y de difícil acceso, donde se implementan los nodos sensores, tienen la necesidad de sistemas robustos en la comunicación de los dispositivos, así como las comunicaciones adaptativas, con el fin de monitorear de forma adecuado los parámetros de interés.
- La autonomía energética en los nodos sensores es una limitante, porque algunas aplicaciones requieren que estos sistemas operen el mayor tiempo posible.

1.1.4. Justificación.

De acuerdo con la empresa de investigación de las tecnologías Gartner, en el 2020 se tendrá 20,000 millones de cosas (dispositivos, equipos, etc.) conectadas. Si se considera más de 7,000 millones de habitantes en el planeta, habrá tres veces más dispositivos conectados que personas, un hecho que habla sobre el crecimiento exponencial del internet de las cosas (IoT). El Perú no es ajeno a la tecnología IoT, para potenciar su implementación, la Universidad de Ingeniería y Tecnología (UTEC) y la empresa *Wireless Network Development* (WND) Perú, instalaron la primera antena basada en la tecnología libre *Sigfox* con la cual se desea reducir brechas de desarrollo tecnológico. Si bien los nodos sensores son implementados ampliamente en diversos entornos como la agricultura de precisión, medio ambiente, domótica, salud, etc. la limitación energética hace que la comunicación inalámbrica confiable y eficiente sea una tarea desafiante, ya que estos dispositivos se caracterizan por utilizar un medio de transmisión dinámico, donde uno de los inconvenientes más inmediatos es cuando la potencia del ruido es mayor que la potencia de la señal de información, y como consecuencia hay mayor consumo de energía en estos dispositivos, por lo tanto se tendrá menor duración en las fuentes recargables. Las diferentes aplicaciones con estos dispositivos exigen una adecuada gestión de energía, con el fin de mejorar su autonomía, frente a este enfoque en el presente trabajo se diseñará e implementará un sistema de control adaptativo de potencia transmitida, a través de la estimación de la relación señal a ruido SNR del canal en la plataforma de la Radio Definida por Software (SDR), ya que la adaptación en potencia es uno de los métodos para mejorar la eficiencia energética en estos sistemas.

Las redes inalámbricas como; *Sigfox* y *Lora*, trabajan con dispositivos y sistemas conectados a Internet de las Cosas (IoT), en diversas aplicaciones, se requiere que estos sistemas operen el mayor tiempo posible, porque algunos de estos dispositivos están instalados en zonas de difícil acceso. Entonces para futuras aplicaciones con estos sistemas, su arquitectura y su entorno de programación debe implementar un algoritmo estimador de la SNR del canal, y a partir de esta controlar la potencia transmitida.

1.2. Objetivo.

1.2.1. Objetivo General.

Diseñar e implementar el control adaptativo de la potencia transmitida en un sistema de comunicación inalámbrico, en función de la estimación de la SNR realizada en el receptor.

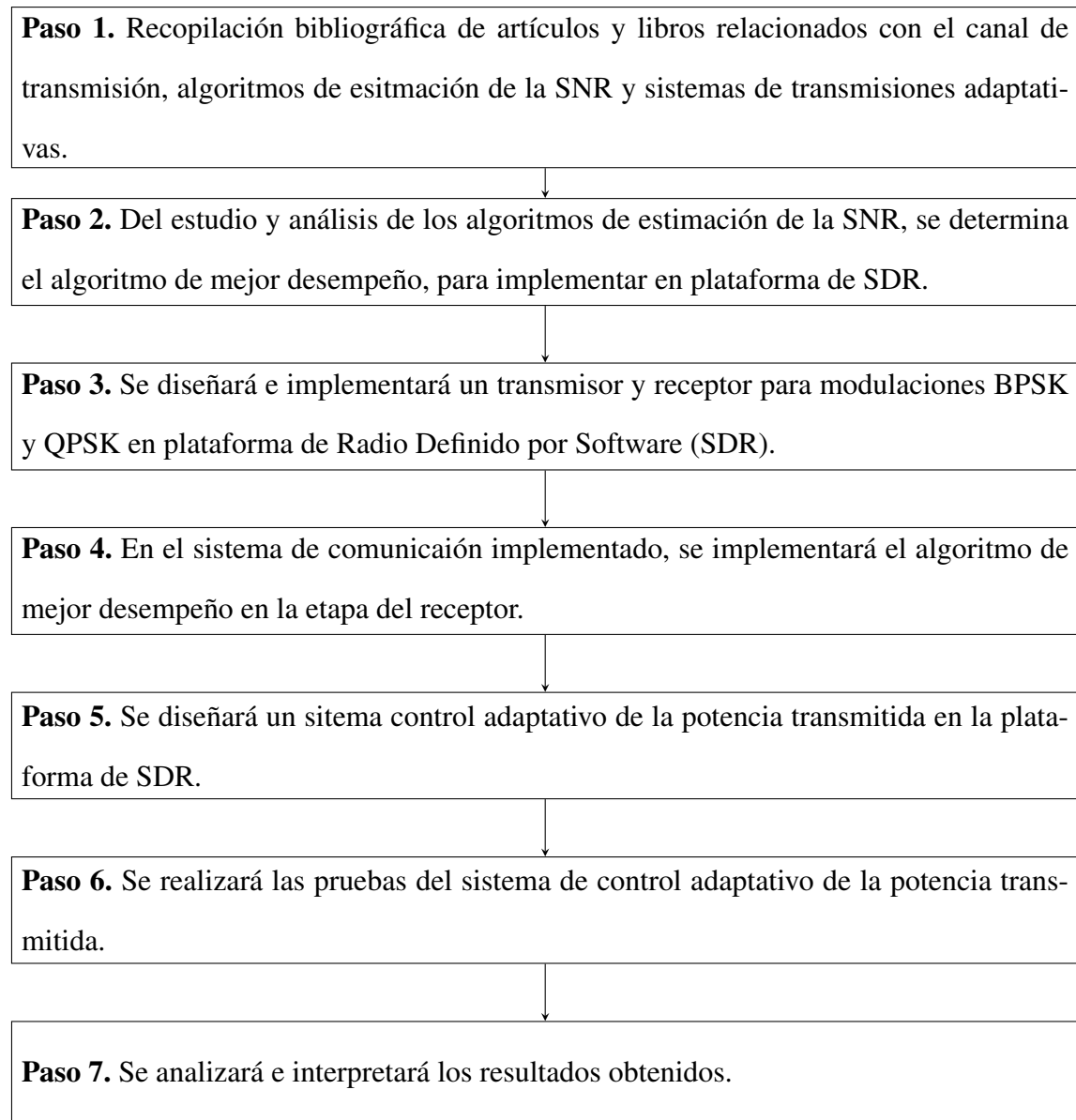
1.2.2. Objetivos Específicos.

- Estudiar los parámetros que intervienen en la estimación de la SNR en un canal de transmisión inalámbrica.
- Determinar el algoritmo de estimación de la SNR de mejor desempeño, para el control adaptativo de la potencia transmitida.
- Diseñar e implementar un transmisor y receptor digital en la plataforma de la Radio Definido por Software (SDR).
- Implementar un sistema de control adaptativo de la potencia transmitida, en la plataforma de SDR.
- Validar la operación del sistema de control adaptativo.

1.3. Metodología.

La presente tesis utiliza metodología aplicada-experimental de enfoque cuantitativo, ya que se vale de datos cuantificables, a las cuales se accede por medio de observaciones y mediciones. Se pretende lograr los objetivos de la investigación realizando pruebas en la plataforma de SDR que comprende el uso del dispositivo USRP (Software Universal Radio Periférico) y el entorno GRC (GNU Radio Companion).

Se tienen los siguientes pasos a seguir para el desarrollo de tesis:



1.3.1. Tipo de investigación

Experimental: Estudia relaciones de causa-efecto a través del sistema de control adaptativo de potencia transmitida, es decir adapta la potencia del transmisor a las condiciones del canal, con el objetivo de mantener un enlace de transmisión fiable a una tasa de error de bit fija.

1.3.2. Población y muestra

- **Población:**

La población comprende los sistemas de transmisiones adaptativas en redes inalámbricas.

- **Muestra:**

Sistema de transmisión adaptativa en potencia para las modulaciones BPSK y QPSK.

- **Tipo de muestreo:** Estratificado, porque se selecciona según ciertas variables de interés y es lo más semejante a la población.

1.3.3. Variables.

$$\rho[i] = 10 \log_{10} \left(\frac{S}{\eta + n[i]} \right) \quad (1.1)$$

- **Variables independientes.**

- η : Ruido complejo AWGN (W: Vatio)
- $n[i]$: Ruido AWGN externo generado por SDR (W:Vatio)

- **Variables dependientes**

- S : Potencia promedio de la señal transmitida (W: Vatio)
- $\rho[i]$: SNR del canal de transmisión (dB: Decibelio)

1.3.4. Técnicas e instrumentos.

- Modulación y demodulación de BPSK y QPSK.
- Algoritmos de estimación de la relación señal a ruido (SNR).
- Control adaptativo de potencia transmitida.
- Dispositivo Software Universal Radio Periférico (USRP).
- Software GNU Radio Companion (GRC).

1.3.5. Alcance del método.

El estudio comprende la implementación de un sistema de control adaptativo de la potencia transmitida a partir de la SNR evaluada en el receptor.

1.3.6. Delimitaciones.

- Los algoritmos de estimación de la SNR son propuestos para modulaciones BPSK y QPSK, porque estas modulaciones son empleadas por los nodos sensores.
- No se considera el efecto de la interferencia cocanal.
- El enlace de comunicación es fija y se realizara pruebas entre puntos fijos determinados.
- Se considera que la SNR del canal de retorno es alto, es decir la potencia de transmisión de la estación base hacia el nodo sensor es mucho mayor que la potencia del ruido.

Capítulo II

MARCO TEÓRICO.

2.1. Antecedentes.

(Lin, 2016): Las redes de sensores inalámbricos se implementan ampliamente en una variedad de entornos, lo que respalda la vigilancia militar, respuestas de emergencias, atención médica y exploración científica. Las limitaciones energéticas de los nodos, hace que la comunicación inalámbrica confiable y eficiente sea una tarea desafiante. Bajo un suministro de energía limitada, la confiabilidad y la eficiencia a menudo están en desacuerdo entre sí. La fiabilidad se puede mejorar transmitiendo paquetes a la máxima potencia de transmisión, pero esta situación introduce un consumo de energía innecesariamente alto. Para proporcionar a los diseñadores de sistemas la capacidad de controlar dinámicamente la potencia de transmisión, el hardware de radio utilizado popularmente, como ofrece un registro para especificar el nivel de potencia de transmisión durante el tiempo de ejecución. Es deseable especificar el nivel de potencia de transmisión mínimo que logre la fiabilidad de comunicación requerida por el bien de ahorrar energía y aumentar la vida útil del sistema.

(Blaszczyk y Lynggaard, 2016): La demanda de dispositivos IoT basados en sistemas inteligentes, son enormes en diversos sectores de producción y desarrollo. Estos sistemas tienen la autonomía computacional y son capaces de comunicarse con un sistema en la nube para informar o descargar datos.

El algoritmo presentado en este trabajo ha sido diseñado y probado en una aplicación de

granja, donde una estación meteorológica envía datos a través de un enlace RF a una puerta de enlace cada 10 minutos, la estación base recopila los datos y los carga a la nube. Para adaptar la potencia transmitida utiliza dos modelos; en el primer modelo calcula la relación señal a ruido (SNR) basándose en los parámetros de una tabla, en el segundo modelo simula el error de bit en una secuencia de tramas transmitidas, que se transfieren a través del canal AWGN que incorpora la SNR del primer modelo. Los módulos RF utilizados en este trabajo proporcionan rangos de potencia de transmisión de: (-11, -6, -3, 0-27) dBm, de acuerdo a la calidad de enlace utiliza este rango de potencia para adaptar el sistema.

(Di Mu y cols., 2017): El control de la potencia de transmisión para una sola radio ha sido ampliamente investigado en la literatura de redes sensores inalámbricos y redes de mallas inalámbricas. Medidas de calidad de enlace indirecto tales como la indicación de la intensidad de la señal recibida (RSSI) el indicador de calidad de enlace (LQI) o medidas de calidad de enlace directo tales como la tasa de recepción de paquetes (PRR) y tasa de error de paquete (PER), se han utilizado para medir calidad de enlace, y enfoque teóricos de control se ha aplicado para lograr la calidad de enlace deseable mediante el control de potencia de transmisión en tiempo de ejecución. Estos enfoques existentes para controlar solo una radio, no son directamente aplicables en este trabajo, ya que el consumo de potencia de múltiples radios deben considerarse conjuntamente. Por el contrario este documento emplea un enfoque pragmático de sistemas integrados para optimizar la potencia transmisión junto con la selección de radio. Los ARTPoS presentado en este artículo se puede controlar su potencia transmisión adaptable, que minimiza proactivamente el consumo de energía sujeto al tráfico de red y condiciones operativas dadas, el rendimiento de las ARTPoS se se demuestra a través de la implementación y experimentación en un hardware real.

(Joaquín y Garsas, 2010): Las redes inalámbricas se caracterizan por utilizar un medio de transmisión dinámico, donde las condiciones de transmisión cambian de forma continua. Este cambio continuo lleva a diseñar los sistemas de comunicaciones desde dos puntos de vista. El más elemental es diseñar el sistema para el peor caso, aunque sencilla, esta solución lleva a un sobredimensionamiento de los recursos empleados; si se considera por ejemplo la potencia de transmisión, este sobredimensionamiento resultará, entre las consecuencias más inmediatas,

en mayores niveles de atenuación, se tendrá mayor consumo de energía (y por lo tanto menor duración de las baterías). Frente al enfoque anterior, la transmisión adaptativa permite adecuar los parámetros de transmisión a las condiciones del canal.

2.2. Redes de sensores.

(Vizcaíno, 2014) Las redes de sensores es un tópico relativamente nuevo y emergente en el campo de la investigación. Se trata de construir pequeños nodos autónomos que se comunican entre si, cada nodo tiene un procesador y una cantidad de memoria que le otorgan cierta capacidad de procesamiento.

Las redes de sensores se comunican entre si de forma inalámbrica, esta comunicación se establece por radiofrecuencia (aunque hay otras, desde ópticas a acústicas) y corto alcance. Otro de los componentes que tiene los nodos es el transceptor de comunicación. Estas dos características (procesamiento y comunicaciones) permiten a las redes de sensores realizar tareas complejas debido a la colaboración entre los que la componen.

Estas tecnologías se pueden distribuir de muchas formas, por ejemplo con comunicación punto a punto, por grupos con un nodo coordinador, con comunicación en todos los vecinos, redes híbridas, etc. Para que los datos lleguen al destino, las redes suelen utilizar una pila de protocolos que incluyan un tipo de enrutamiento multisalto. En la actualidad los protocolos de comunicaciones en WNS es uno de los temas de investigación.

2.2.1. Aplicaciones.

La redes de sensores, como se ha comentado antes, se han usado en ámbito académico para investigar sobre protocolos de redes de comunicación. Además en el terreno de la investigación, ha habido numerosos proyectos para utilizar los conceptos de las redes de sensores en aplicaciones específicas y en algunos casos comerciales.

En el siguiente apartado se mostrarán algunas aplicaciones prácticas en las que se han utilizado las de sensores.

- Medio ambiente.

- Domótica y salud.
- Industriales y vehículos.
- Eficiencia energética.
- Aplicaciones militares y entornos de alta seguridad.

2.2.2. Características de la redes de sensores.

- **Topología de la red de comunicación.** Una de las principales características de las redes sensores es que se comunican entre sí de forma inalámbrica. La forma de comunicación puede utilizarse para clasificar las redes atendiendo a varios parámetros. Como se ha comentado en la introducción, en redes sensores se trabaja con topologías de redes muy diversas. Los tipos de redes pueden ser centralizadas, mallas, en árbol con todos los nodos iguales, con nodos distintos, con nodos móviles, estáticos etc.

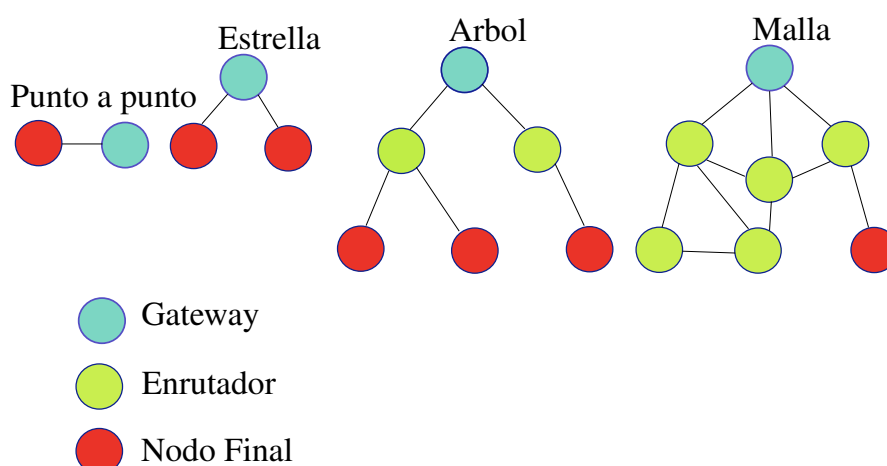


Figura 2.1: Distintos tipos de topologías utilizados en redes de sensores (Vizcaíno, 20114).

La figura 2.1 muestra la arquitectura de las redes de sensores más utilizadas. En primer lugar está el más trivial: comunicación punto a punto; en segundo lugar una red estrella, donde hay un nodo central, que es el único con el que se comunican el resto de los nodo; el tercer tipo es una red en árbol, la cual puede verse como la unión de varias redes en estrella a través de sus respectivos nodos centrales; y finalmente se muestra una red mallada, donde los nodos pueden comunicarse entre sí sin tener que pasar por un nodo central.

Al transmitir información, los datos se generan en el nivel superior del emisor, los datos irán pasando de un nivel al inmediatamente inferior a través de los interfaces que tienen definidos los protocolos. Estos interfaces básicamente son estructuras de datos y funciones que deben reconocer un nivel y el contiguo. Cada nivel realizará las operaciones oportunas con los datos, comúnmente se añadirán cabeceras de control y códigos de detección o corrección de errores. Además, algunos niveles pueden llevar a cabo tareas adicionales como un control del flujo, retransmisiones, etc.

En recepción el proceso es el inverso. Los datos llegan al nivel más bajo y, de ahí, se va pasando a un nivel superior donde se vuelven a realizar ciertas tareas y se desglosan las estructuras de datos.

En la Fig. 2.2 se puede ver varias pilas de protocolos típicas comenzando por el más teórico del modelo OSI, y a continuación tres sistemas de comunicaciones inalámbricas: wifi, bluetooth y zigbee.

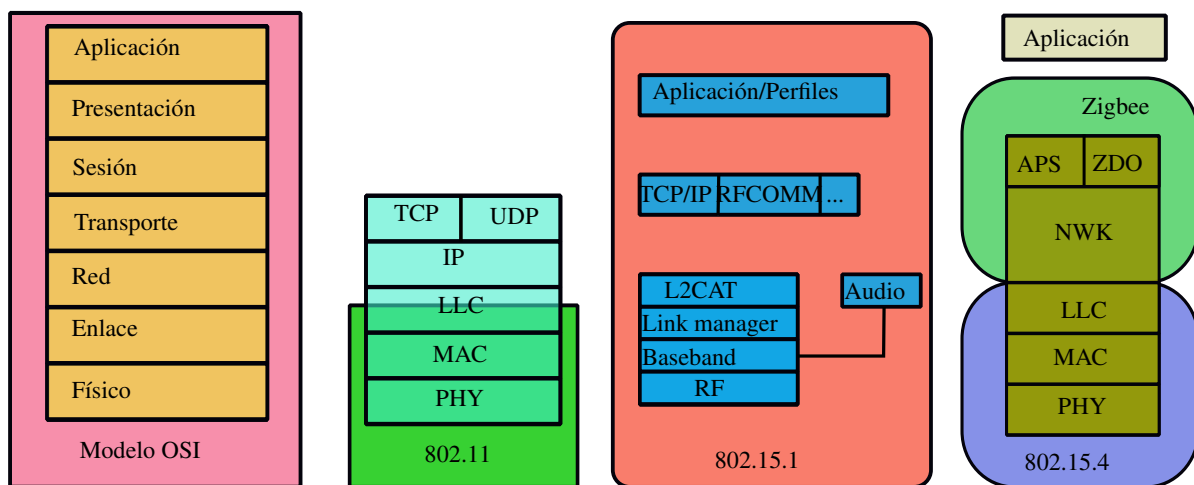


Figura 2.2: Pila de protocolos del modelo OSI (izquierda), y diferentes implementaciones en sistemas de comunicación inalámbrica de alta frecuencia (Vizcaíno, 20114).

Se resume brevemente las funciones principales de cada nivel.

- Nivel Físico: Se encarga del envío de información entre los nodos a través del medio, establece el tipo de modulación y codificación. En este nivel se establece el código que se utilizara para transmitir la señal, como la codificación Manchester, AMI, NRZ, etc.
- Nivel de enlace de datos: Se encarga de controlar el acceso al medio si éste es compartido, ahorrar energía, minimizar la latencia, proporcionar escalabilidad, y servicios de

transmisión, recepción, confirmación, control de errores, etc. Los protocolos de este nivel se pueden dividir en.

- Ranurados: basados en slots o cuantos de tiempo. Algunos ejemplos son: TDMA, 802.15.4, SMAC, DSMAC, TMAC, TRAMA, etc.
 - De muestreo: buscan periódicamente actividad en el medio. Algunos ejemplos son: ALOHA, BMAC, WiseMAC, CSMA, etc.
 - Otros, como STEM, híbridos como ZMAC, SCP-MAC, etc.
-
- Nivel de Red. Se encarga de encaminar los datos para que lleguen desde su emisor a un receptor a través de una ruta que pueda estar compuesta por otros nodos.
 - Nivel de transporte: Mantiene el flujo y controla los errores que se puedan producir en la comunicación a un nivel más alto que el de enlace de datos (Por ejemplo detectaría información duplicada, orden incorrecto, etc.).
 - Nivel de sesión. Si hay una comunicación bidireccional en este nivel se encarga de regularlo, además de la recuperación de errores.
 - Nivel de presentación: Uniforma la presentación de la información (ASCII, Unicode...).
 - Nivel de aplicación: Define las normas para intercambiar los datos provenientes de la aplicación del usuario.

En las redes de sensores tienen especial importancia los tres primeros, siendo determinantes en la eficiencia energética de la propia red. Existen varias implementaciones de estos tipos de protocolos por ejemplo.

802.15.4 implementa los niveles físicos y de enlace de datos. En el nivel físico utiliza una modulación de ensanchado de espectro y O-QPSk a 2.4 GHz, 250 kb/s; Proporciona unos servicios a la capa de enlace de datos tales como indicación si hay datos, leer y establecer ciertas propiedades, como la potencia, activación y desactivación del transceiver, seleccionar el canal, etc.

Zigbee se sitúa por encima de 802.15.4, en el nivel de red, y establece diferentes topologías

de red: en estrella, árbol, malladas y varios servicios: unirse o abandonar una red, seguridad, rutado de los paquetes, etc.

2.3. Sistemas de Transmisiones Adaptativas

(Goldsmith, 2005) En esta sección se describe sistemas asociados con la transmisiones adaptativas. Se considera que una modulación lineal donde la adaptación tiene lugar en un múltiplo de la velocidad de símbolos $R_s = 1/T_s$, se modela un canal con desvanecimiento plano como un canal de tiempo discreto donde cada uso del canal corresponde a un tiempo de símbolo T_s , tiene una ganancia $\sqrt{g_i}$ estacionaria y ergódica que varia con el tiempo que sigue una distribución $P(g)$ y AWGN $n[i]$ con densidad espectral de potencia $N_o/2$, \bar{S} denota la potencia promedio de la señal transmitida, $B = 1/T_s$ denota el ancho de banda de la señal recibida y \bar{g} ganancia promedio del canal. La SNR instantánea recibida es:

$$\gamma[i] = \frac{\bar{S}g[i]}{(N_oB)} \quad (2.1)$$

, y su valor esperado en todo momento es:

$$\bar{\gamma} = \frac{\bar{S}\bar{g}}{(N_oB)} \quad (2.2)$$

Como $g[i]$ es estacionaria, la distribución de $Y[i]$ es independiente de γi , y denotamos esta distribución $P(\gamma)$. En la transmisión adaptativa, estimamos la ganancia de potencia o la SNR recibida en el momento i y adaptamos los parámetros de modulación y codificación en consecuencia. Los parámetros más comunes para adaptar son la velocidad de datos $R[i]$, la potencia de transmisión $S[i]$ y los parámetros de codificación $C[i]$.

Para la modulación M-ary, la velocidad de datos se calcula en la siguiente ecuación.

$$R[i] = \frac{\log_2 M[i]}{T_s} = B \log_2 M[i] \quad bps. \quad (2.3)$$

La eficiencia espectral de la modulación M-aria es:

$$\frac{R[i]}{B} = \log_2 M[i] \quad \text{bps/Hz} \quad (2.4)$$

Se denota la estimación de SNR como:

$$\hat{\gamma}[i] = \frac{\bar{S}\hat{g}[i]}{(N_o B)} \quad (2.5)$$

La (2.5) se basa en la estimación de ganancia de potencia $\hat{g}[i]$. Suponga que la potencia de transmisión está adaptada en relación con $\hat{\gamma}[i]$. Denotamos esta potencia de transmisión adaptativa en el tiempo i por $S(\hat{\gamma}[i]) = S[i]$ y la potencia recibida en el tiempo i es entonces $\gamma[i] \frac{S(\hat{\gamma}[i])}{S}$. De manera similar, podemos adaptar el velocidad de datos de la modulación $R(\hat{\gamma}[i]) = R[i]$ y/o los parámetros de codificación $C(\hat{\gamma}[i]) = C[i]$ en relación con la estimación $\hat{\gamma}[i]$. Cuando el contexto es claro, omitiremos la referencia de tiempo i relativa a γ , $S(\gamma)$, $R(\gamma)$ y $C(\gamma)$.

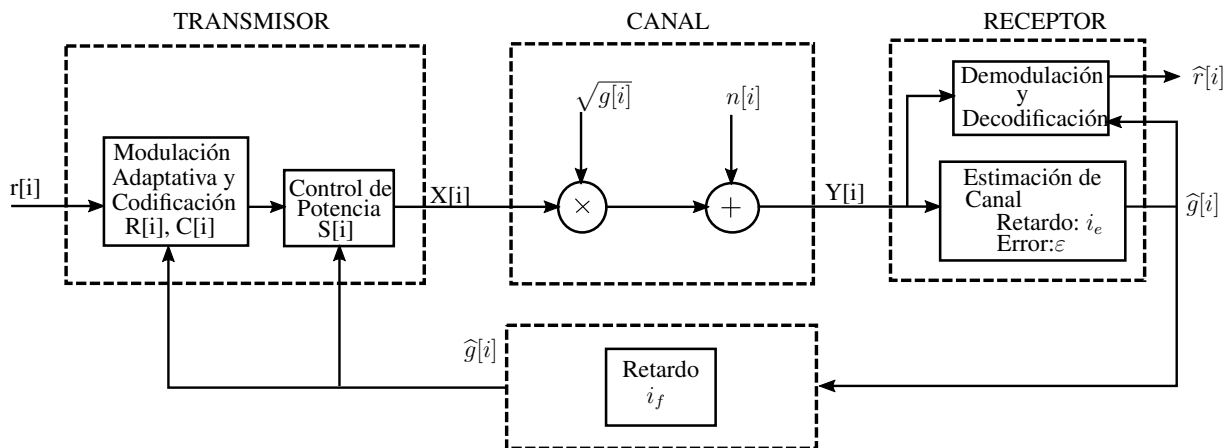


Figura 2.3: Modelo del Sistema de transmisión adaptativa (Goldsmith, 2005).

El modelo del sistema se ilustra en la Fig. 2.3. Suponemos que una estimación $\hat{g}[i]$ de la ganancia de potencia del canal $g[i]$ en el momento i está disponible para el receptor después de un retraso de tiempo de estimación de i_e y que esta misma estimación está disponible para el transmisor después de una ruta combinada de estimación y retroalimentación retraso de $i_d = i_e + i_f$. La disponibilidad de esta información de canal en el transmisor le permite adaptar su esquema de transmisión en relación con la variación del canal. La estrategia de adaptación puede tener en cuenta el error de estimación y el retraso en $\hat{g}[i]$ o puede tratar a $\hat{g}[i]$ como la

ganancia real.

Hay muchos parámetros que se pueden variar en el transmisor relativo a la ganancia del canal γ . En esta sección discutimos técnicas adaptativas asociadas con la variación de los parámetros más comunes: velocidad de datos, potencia, codificación y probabilidad de error.

2.3.1. Técnicas de tasa variable

En la modulación de velocidad variable, la velocidad de datos $R[\gamma]$ varía en relación con la ganancia de canal γ . Esto se puede hacer fijando la velocidad de símbolo $R_s = 1/T_s$ de la modulación y usando múltiples esquemas de modulación o tamaños de constelación, o arreglando la modulación (por ejemplo, BPSK) y cambiando la velocidad de símbolo. La variación de la velocidad de símbolos es difícil de implementar en la práctica ya que un ancho de banda de señal variable no es práctico y complica el intercambio de ancho de banda. Por el contrario, cambiar el tamaño de la constelación o el tipo de modulación con una velocidad de símbolo fija es bastante fácil, y estas técnicas se utilizan en los sistemas actuales. Específicamente, EGPRS para la transmisión de datos en sistemas celulares GSM varía entre la modulación 8PSK y GMSK, y GPRS para la transmisión de datos en sistemas celulares IS 136 TDMA puede usar modulación PSK de 4, 8 y 16 niveles. En general, los parámetros de modulación para variar la velocidad de transmisión se fijan sobre un bloque o cuadro de símbolos, donde el tamaño del cuadro es un parámetro del diseño. Los marcos también pueden incluir símbolos piloto para la estimación de canales y otros controles información. Cuando se utiliza un conjunto discreto de tipos de modulación o tamaños de constelación, cada valor de γ debe asignarse a uno de los posibles esquemas de modulación. Esto a menudo se hace para mantener la probabilidad de error de bit de cada esquema por debajo de un valor dado.

2.3.2. Técnicas de potencia variable.

La adaptación de la potencia de transmisión sola se usa generalmente para compensar la variación de SNR debido al desvanecimiento. El objetivo es mantener una probabilidad de error de bit fija o, de manera equivalente, una SNR recibida constante. La adaptación de potencia invierte así el desvanecimiento del canal para que el canal aparezca como un canal AWGN para

el modulador y demodulador. La adaptación de potencia para la inversión del canal viene dada por.

$$\frac{S(\gamma)}{\bar{S}} = \frac{\sigma}{\gamma} \quad (2.6)$$

Donde σ es igual a la constante SNR recibida. La restricción de potencia media \bar{S} implica que.

$$\int \frac{S(\gamma)}{\bar{S}} p(\gamma) d\gamma = \int \frac{\sigma}{\gamma} P(\gamma) d\gamma = 1 \quad (2.7)$$

Resolviendo (2.7) para σ que produce $\sigma = 1/E[1/\gamma]$, por lo que σ está determinado por $P(\gamma)$ que a su vez depende de la potencia de transmisión promedio \bar{S} a través de γ . Por lo tanto, para una potencia media dada \bar{S} , si el valor de σ requerido para alcanzar el BER objetivo es mayor que $1/E[1/\gamma]$, entonces este objetivo no puede cumplirse. Tenga en cuenta que para el desvanecimiento de Rayleigh donde γ se distribuye exponencialmente, $E[1/\gamma] = \infty$, por lo que no se puede alcanzar el objetivo P_b utilizando la inversión del canal. El desvanecimiento también puede invertirse por encima de un límite dado γ_o , lo que conduce a una inversión de canal truncada para la adaptación de potencia. En este caso, la adaptación de potencia viene dada por.

$$\frac{S(\gamma)}{\bar{S}} = \begin{cases} \frac{\sigma}{\gamma}, & \gamma \geq \gamma_o, \\ 0, & \gamma < \gamma_o \end{cases} \quad (2.8)$$

El valor de corte γ_o puede basarse en una probabilidad de interrupción deseada $P_{out} = P(\gamma < \gamma_o)$ o en una BER objetivo deseada por encima de un límite determinado por la BER objetivo y $P(\gamma)$. Dado que el canal solo se usa cuando $\gamma \geq \gamma_o$, dada una potencia media \bar{S} , tenemos $\sigma = 1/E_{\gamma_o}[1/\gamma]$, donde.

$$E_{\gamma_o}[1/\gamma] \triangleq \int_{\gamma_o}^{\infty} \frac{1}{\gamma} P(\gamma) d\gamma \quad (2.9)$$

2.3.3. Probabilidad de error variable

También podemos adaptar la BER instantánea sujeta a una restricción BER promedio P_b . En los canales de desvanecimiento la probabilidad de error instantánea varía a medida que varía la SNR_{γ} recibida, resultando en una BER promedio de $P_b = \int P_b(\gamma) P(\gamma) d\gamma$. Esto no se considera

una técnica adaptativa ya que el transmisor no se adapta a γ . Así, en la modulación adaptativa, la probabilidad de error se adapta típicamente junto con alguna otra forma de adaptación, como el tamaño de la constelación o el tipo de modulación.

2.4. Control PID.

Los controladores PID suelen utilizar la retroalimentación del bucle de control en aplicaciones industriales y de sistemas de control. El controlador primero calcula un valor de error como la diferencia entre una variable de proceso medida y un punto de ajuste preferido. Luego intenta minimizar el error aumentando o disminuyendo las entradas o salidas de control en el proceso para que la variable del proceso se acerque más al punto de ajuste. Este método es más útil cuando un modelo matemático del proceso o el control es demasiado complicado o desconocido para el sistema. Para aumentar el rendimiento, como por ejemplo aumentando la capacidad de respuesta del sistema, los parámetros del PID deben ser ajustados según la aplicación específica.

2.4.1. Control de tres términos.

El controlador PID esta compuesta por tres elementos que proporcionan una acción proporcional, integral y derivativo que se muestra en la Fig. 2.4 (Rivera, 2006).

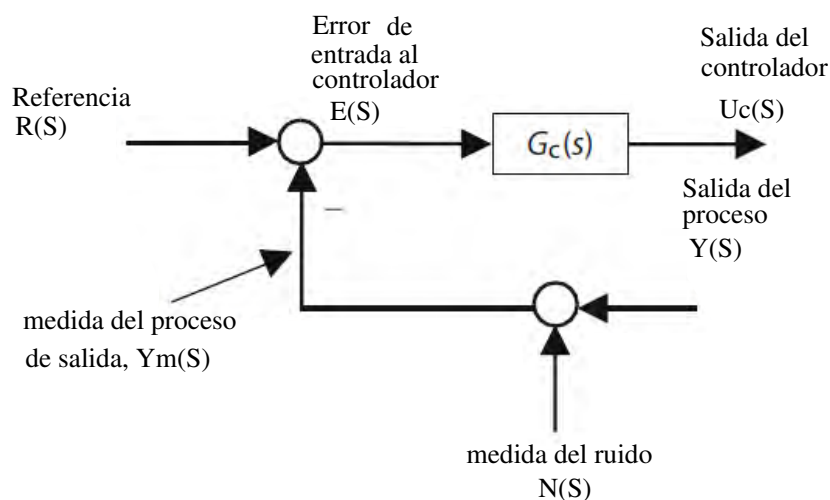


Figura 2.4: Entradas y salidas del controlador (Rivera, 2006).

Enseguida se describe la acciones del controlador.

Acción proporcional.

Se utiliza cuando la acción del controlador debe ser proporcional al tamaño de la señal de error del proceso $e(t) = r(t) - y_m(t)$. Las representaciones del tiempo y del dominio Laplace para el control proporcional se dan como:

- Dominio del tiempo $u_c(t) = k_p e(t)$
- Dominio de Laplace $U_c(S) = k_p E(S)$

Acción integral.

Se utiliza cuando se requiere que el controlador corrija cualquier desviación constante de un valor de señal de referencia constante. El control integral supera la deficiencia del control proporcional eliminando el offset sin el uso de una ganancia de controlador excesivamente grande. Las representaciones del dominio del tiempo y de Laplace para el control integral se dan como:

- Dominio del tiempo $u_c(t) = k_I \int_t e(\tau) d\tau$
- Dominio de Laplace $U_c(S) = \left[\frac{k_I}{S} \right] E(S)$

Acción derivativo.

Si un controlador puede utilizar la tasa de cambio de una señal de error como entrada, entonces esto introduce un elemento de predicción en la acción de control. El control derivado utiliza la tasa de cambio de una señal de error y es el término D en el controlador PID. Las representaciones del dominio del tiempo y de Laplace para el control derivativo se dan como:

- Dominio del tiempo $u_c(t) = k_D \frac{de}{dt}$
- Dominio de Laplace $U_c(S) = [k_D S] E(S)$

Controladores de PID paralelos.

La familia de controladores PID se construye a partir de las combinaciones de los términos proporcional, integral y derivado, según sea necesario para cumplir con los requisitos de

rendimiento específicos. La fórmula del controlador PID paralelo básico es:

$$U_c(S) = \left[k_P + K_I \frac{1}{S} + K_D S \right] E(S) \quad (2.10)$$

El desacoplamiento de los tres términos es una consecuencia de la arquitectura paralela del controlador PID.

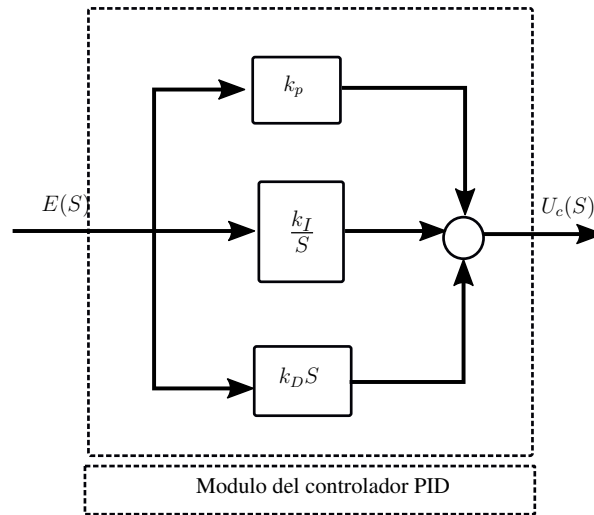


Figura 2.5: Entradas y salidas del controlador (Rivera, 2006).

2.5. Variables aleatorias.

Las variables aleatorias no solo son importantes para las estadísticas sino también para modelar canales inalámbricos. Por lo tanto se revisará algunas definiciones y términos básicos que se usan con frecuencia en relación con las variables aleatorias (Pätzold, 2011).

Un experimento cuyo resultado no se conoce de ante mano se llama un experimento aleatorio. Llamaremos a s puntos que representan los resultados de un experimento aleatorio de puntos de muestra. Un conjunto de posibles resultados de un experimento aleatorio es un evento \mathbf{A} . El evento $\mathbf{A} = \{s\}$ que consiste en un solo elemento s es un evento elemental. El conjunto de todos los resultados posibles de un experimento aleatorio dado se denomina espacio de muestra Q de ese experimento. El espacio de muestras Q se llama el evento determinado y el conjunto vacío o el conjunto nulo se denota por \emptyset , es el evento imposible, sea \mathbf{B} una clase de un subconjuntos de un espacio muestral Q . En la teoría de probabilidad, \mathbf{B} a menudo se llama

campo – σ (o *Algebra* – σ) si y solo si cumple las siguientes propiedades.

- El conjunto vacío $\emptyset \in \mathbf{B}$.
- Si $\mathbf{A} \in \mathbf{B}$, después también, $Q\text{-}\mathbf{A} \in \mathbf{B}$, es decir si el evento \mathbf{A} es un elemento de clase \mathbf{B} entonces también lo es su complemento.
- Si $\mathbf{A}_n \in \mathbf{B}$ ($n = 1, 2, 3, \dots$) después también $\bigcup_{n=1}^{\infty} \mathbf{A}_n \in \mathbf{B}$, es decir, si los eventos \mathbf{A}_n son todo los elementos de la clase \mathbf{B} , también es la unión contable.

2.5.1. Función de distribución acumulativa.

La función F_μ definida por:

$$F_\mu : \mathbb{R} \longrightarrow [0, 1], x \longmapsto F_\mu(x) = P(\mu \leq x) \quad (2.11)$$

Se denomina la función de distribución acumulativa de la variable aleatoria μ . la función de distribución acumulativa $F_\mu(x)$ satisface las siguientes propiedades. a) $F_\mu(-\infty) = 0$; b) $F_\mu(\infty) = 1$ y c) $F_\mu(x)$ no esta disminuyendo, es decir, $F_\mu(x_1) \leq F_\mu(x_2)$.

2.5.2. Función de densidad de probabilidad.

La función P_μ , definido por:

$$P_\mu : \mathbb{R} \longrightarrow \mathbb{R}, x \longmapsto P_\mu(x) = \frac{dF_\mu(x)}{dx} \quad (2.12)$$

Se denomina función de densidad de probabilidad (o densidad de probabilidad) de la variable aleatoria μ , donde se supone que la función de distribución acumulativa F_μ es diferente con respecto a x . La función de densidad de probabilidad $P_\mu(x)$ satisface la siguientes propiedades, a) $P_\mu(x) \geq 0$ para todo x b) $\int_{-\infty}^{\infty} P_\mu(x)dx = 1$ y c) $F_\mu(x) = \int_{-\infty}^x P_\mu(x)dx$.

2.5.3. Valor esperado (valor medio).

$$E\{\mu\} = \int_{-\infty}^{\infty} x P_{\mu}(x) dx \quad (2.13)$$

La cantidad de la ecuación (2.13) se llama valor esperado (valor medio o promedio estadístico) de la variable aleatoria μ , donde $E\{.\}$ denota el operador de valor esperado. El operador de valor esperado $E\{.\}$ es lineal, es decir, la relación $E\{\alpha\mu\} = \alpha E\{\mu\}$ ($\alpha \in \mathbb{R}$) y $E\{\mu_1 + \mu_2\} = E\{\mu_1\} + E\{\mu_2\}$. Sea $f(u)$ una función de variable aleatoria de μ entonces el valor esperado de $f(\mu)$ se puede determinar aplicando la relación fundamental.

$$E\{f(\mu)\} = \int_{-\infty}^{\infty} f(x) P_{\mu}(x) dx \quad (2.14)$$

2.5.4. Varianza.

$$Var\{\mu\} = E\{(\mu - E\{\mu\})^2\} = E\{\mu^2\} - (E\{\mu\})^2 \quad (2.15)$$

El valor de la ecuación (2.15) se denomina varianza de la variable aleatoria μ , donde $Var\{\mu\}$ denota el operador de varianza. La varianza de una variable aleatoria μ es una medida de la concentración de μ cerca de su valor esperado.

2.5.5. Momentos.

El momento de k orden de la variable aleatoria μ se define por:

$$E\{\mu^k\} = \int_{-\infty}^{\infty} x^k P_{\mu}(x) dx, k = 1, 2, 3, \dots \quad (2.16)$$

2.5.6. Función característica.

La función característica de una variable aleatoria μ se define como el valor esperado.

$$\Psi_{\mu}(\nu) = E\{e^{j2\pi\nu\mu}\} = \int_{-\infty}^{\infty} P_{\mu}(x) e^{j2\pi\nu x} dx \quad (2.17)$$

Donde ν es una variable de valor real, cabe señalar que $\Psi_\mu(-\nu)$ es la transformada de Fourier de la función de densidad de probabilidad $P_\mu(x)$. La función característica a menudo proporciona una técnica simple para determinar la función de densidad de probabilidad de una suma de variables aleatorias estadísticamente independientes.

2.5.7. Importantes funciones de densidad de probabilidad.

A continuación, se presentará un resumen de algunas funciones importantes de densidad de probabilidad que se utilizan a menudo en relación con el modelado de canales. Las propiedades estadísticas correspondientes, como el valor esperado y la varianza, también se tratarán.

Distribución uniforme.

Sea θ una variable aleatoria de valor real con la función de densidad de probabilidad .

$$P_\theta(x) = \begin{cases} \frac{1}{2\pi}, & [-\pi, \pi), \\ 0, & \text{si no.} \end{cases} \quad (2.18)$$

Entonces $P_\theta(x)$ se denomina distribución uniforme, se dice que θ se distribuye uniformemente en el intervalo $[-\pi, \pi)$. El valor esperado y la varianza de una variable aleatoria θ distribuida uniformemente son; $E\{\theta\} = 0$ y $Var\{\theta\} = \pi^2/3$ respectivamente.

Distribución Gaussiana (distribución normal).

Sea μ una variable aleatoria de valor real con la función de densidad de probabilidad.

$$P_\mu(x) = \frac{1}{\sqrt{2\pi}\sigma_\mu} e^{-\frac{(x-m_\mu)^2}{2\sigma_\mu^2}} \quad (2.19)$$

Entonces $P_\mu(x)$ se denomina distribución Gaussiana y se dice que μ tiene distribución Gaussiana. En la ecuación anterior, $m_\mu \in \mathbb{R}$ denota el valor esperado y $\sigma_\mu^2 \in (0, \infty)$ es la varianza, es decir, $E\{\mu\} = m_\mu$ y $Var\{\mu\} = E\{\mu^2\} - m_\mu^2 = \sigma_\mu^2$.

Para describir las propiedades de distribución de las variables aleatorias distribuidas Gaussianas μ , a menudo se utiliza la notación corta $\mu \sim N(m_\mu, \sigma_\mu^2)$ en lugar de dar expresión completa

(2.19). Especialmente, para $m_\mu = 0$ y $\sigma_\mu^2 = 1$, $N(0, 1)$ se denomina distribución normal estándar.

Distribución Rayleigh.

Consideramos dos variables aleatorias distribuidas normalmente, estadísticamente independientes μ_1 y μ_2 de media cero, cada una con varianza σ_0^2 , es decir, $\mu_1, \mu_2 \sim N(0, \sigma_0^2)$. Además, derivemos una nueva variable de acuerdo con, $\zeta = \sqrt{\mu_1^2 + \mu_2^2}$. Entonces, ζ representa una variable aleatoria distribuida de Rayleigh. La función de densidad de probabilidad $P_\zeta(x)$ de las variables aleatorias distribuidas de Rayleigh ζ , viene dado por.

$$P_\zeta(x) = \begin{cases} \frac{x}{\sigma_0^2} e^{-\frac{x^2}{2\sigma_0^2}}, & x \geq 0, \\ 0, & x < 0. \end{cases} \quad (2.20)$$

Las variables aleatorias distribuidas de Rayleigh ζ tiene el valor esperado $E\{\zeta\} = \sigma_0 \sqrt{\frac{\pi}{2}}$ y la varianza $Var\{\zeta\} = \sigma_0^2 \left(2 - \frac{\pi}{2}\right)$.

Distribución Rice.

Sea $\mu_1, \mu_2 \sim N(0, \sigma_0^2)$ y $\rho \in \mathbb{R}$ entonces la variable aleatoria $\xi = \sqrt{(\mu_1 + \rho)^2 + \mu_2^2}$ se denomina variable aleatoria de distribución de Rice. La función de densidad de probabilidad $P_\xi(x)$ de las variables aleatorias distribuidas de Rice ξ es.

$$P_\xi(x) = \begin{cases} \frac{x}{\sigma_0^2} e^{-\frac{x^2 + \rho^2}{2\sigma_0^2}} I_0\left(\frac{x\rho}{\sigma_0^2}\right), & x \geq 0, \\ 0, & x < 0. \end{cases} \quad (2.21)$$

Donde $I_0(\cdot)$ denota la función de Bessel modificad de orden cero, para $\rho = 0$, la distribución de Rice $P_\xi(x)$ da como resultado la distribución de Rayleigh $P_\zeta(x)$ descrita anteriormente.

El primer y segundo momento de las variables aleatorias distribuidas de Rice son; $E\{\xi\} = \sigma_0 \sqrt{\frac{\pi}{2}} e^{-\frac{\rho^2}{4\sigma_0^2}} \left\{ \left(1 + \frac{\rho^2}{2\sigma_0^2}\right) I_0\left(\frac{\rho^2}{4\sigma_0^2}\right) + \frac{\rho^2}{4\sigma_0^2} I_1\left(\frac{\rho^2}{4\sigma_0^2}\right) \right\}$ y $E\{\xi^2\} = 2\sigma_0^2 + \rho^2$.

Distribución Lognormal.

Sea μ una variable aleatoria distribuida Gaussiana con el valor esperado m_μ y la varianza σ_μ^2 , es decir, $\mu \sim N(m_\mu, \sigma_\mu^2)$. Entonces, se dice que la variable $\lambda = e^\mu$ se distribuye en lognormal. La función de densidad de probabilidad $P_\lambda(x)$ de las variables aleatorias con distribución lognormal, esta dado por.

$$P_\lambda(x) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma_\mu x}} e^{-\frac{(\ln x - m_\mu)^2}{2\sigma_\mu^2}}, & x \geq 0, \\ 0, & x < 0. \end{cases} \quad (2.22)$$

El valor esperado y la varianza de las variables tiene una distribución lognormal, esta dada por, $E\{\lambda\} = e^{m_\mu + \frac{\sigma_\mu^2}{2}}$ y $Var\{\lambda\} = e^{m_\mu + \frac{\sigma_\mu^2}{2}} (e^{\sigma_\mu^2} - 1)$ respectivamente.

2.6. GNU Radio Companion (GRC).

GNU Radio es un entorno de desarrollo libre y de código abierto, compuesto por archivos y aplicaciones agrupados en librerías. Puede ser usado junto con hardware de RF externo para crear radio definida por software y sin hardware es útil en entornos de simulación. Además, GNU es una herramienta que permite la investigación y desarrollo de sistemas de radio.

Para su funcionamiento, GNU Radio integra los lenguajes Python y C++ mediante una interfaz de compilación llamada SWIG, la cual también es compatible con lenguajes como Perl o Ruby. Esta aplicación permite la construcción y el desarrollo de los bloques en GNU-Radio, atribuyéndole diversas características sobre su funcionamiento tales como:

- Conversión de funciones C/C++ en funciones o comandos Python.
- Convierte variables globales de C/C++ en objetos de Python, haciendo que una asignación de variable sea solo un renombramiento de un objeto (objeto llamado por defecto `cvar`).
- Usa la directiva `%include` para importar código o librerías externas.
- Permite exportación del lenguaje a expresiones "XML".

- Crea una equivalencia de los tipos de datos entre *C/C++* y *Python*.

Una de las ventajas de usar GNU Radio es que al ser un software de código abierto, se puede crear propios bloques según las necesidades con ventaja de programarlos en lenguaje Python. Está diseñado y listo para distribuciones como Mac Os, Windows o Linux, es recomendable usar Linux por las diversas características que este sistema operativo presenta.

2.6.1. Entorno de GNU Radio Companion (GRC).

Se vio anteriormente que GNU Radio es una colección de herramientas que se pueden usar para desarrollar sistemas de radio en software en lugar de completamente en hardware. Ahora comenzamos de manera simple y exploramos cómo usar GNU Radio Companion (GRC), la herramienta gráfica de GNU Radio, para crear diferentes tonos. Debemos tener en cuenta que GRC se creó para simplificar el uso de GNU Radio al permitirnos crear gráficos de Python gráficamente en lugar de crearlos solo en código (GRC, 2019).

Lo primero que se verifica es el interfaz a continuación en la Fig. 2.6.

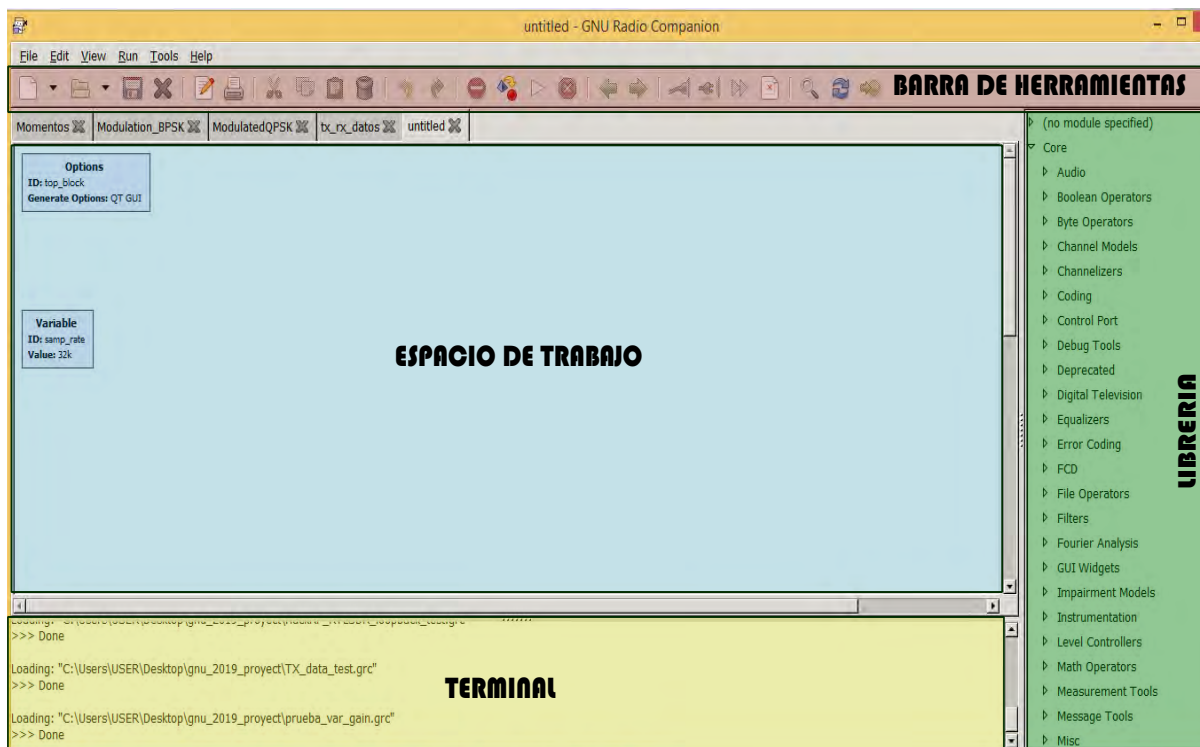


Figura 2.6: Interfaz de GNU Radio Companion (GRC).

Como buscar los bloques de procesado de señal en GRC.

La librería contiene los diferentes bloques instalados en las rutas de bloque GRC. Aquí encontramos bloques que están preinstalados en GNU Radio y bloques que están instalados en el sistema. Al principio parece desalentador buscar bloques. Por ejemplo, si queremos generar una forma de onda, ¿en qué categoría deberíamos buscar? Vemos que hay una categoría de Generadores de forma de onda.

La solución es usar la función de búsqueda haciendo clic en el ícono de la lupa o presionando *Ctrl + f* y luego comenzar a escribir una palabra clave para identificar el bloque. Vemos un cuadro blanco aparecer en la parte superior de la Biblioteca con un cursor. Si escribimos "sink", podemos encontrar todos los bloques que contienen las palabras "sink" y las categorías en las que encontraremos cada bloque.

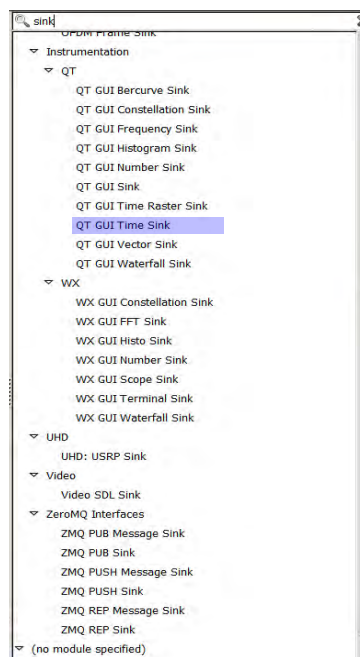


Figura 2.7: Librería GRC.

Por ahora, agreguemos el bloque llamado *QT GUI Time Sink* haciendo clic en su nombre y arrastrándolo al espacio de trabajo o haciendo doble clic en su nombre para que se coloque automáticamente en el espacio de trabajo.

Modificación de propiedades de bloque.

El espacio de trabajo (área principal de la pantalla) contiene todos nuestros bloques que forman nuestro diagrama de flujo, y dentro de cada bloque podemos ver todos los diferentes parámetros de bloque. Sin embargo, hay un bloque especial con el que cada nuevo diagrama de flujo comienza y que se requiere que tenga, llamado **Options Block**. Hagamos doble clic en el Bloque de opciones para examinar sus propiedades. Vemos una ventana como a continuación:

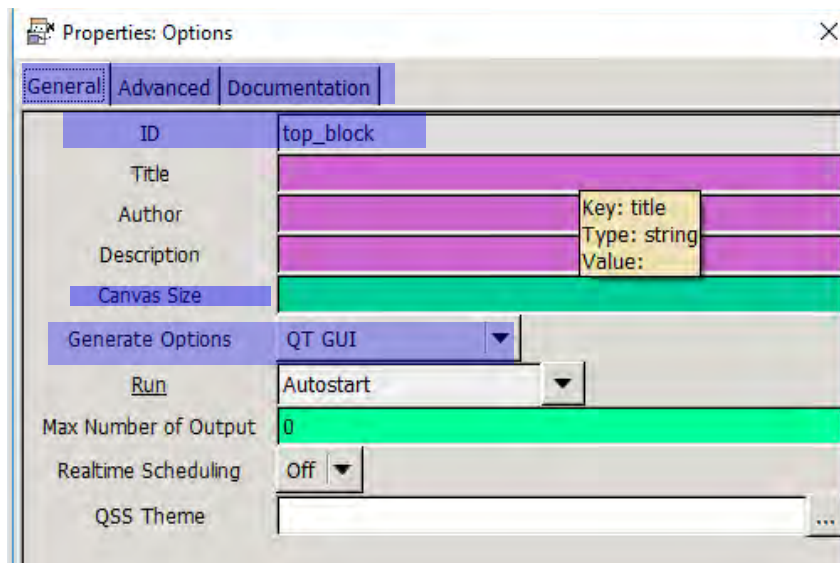


Figura 2.8: Propiedades de bloque de GRC.

Estas propiedades de bloque se pueden cambiar de los valores predeterminados para realizar diferentes tareas. Si volvemos a las propiedades del Bloque de opciones, podemos ver que hay diferentes pestañas y una se titula **Documentation** donde describe las características y funciones que realiza el bloque que se está utilizando como se muestra en la siguiente figura.

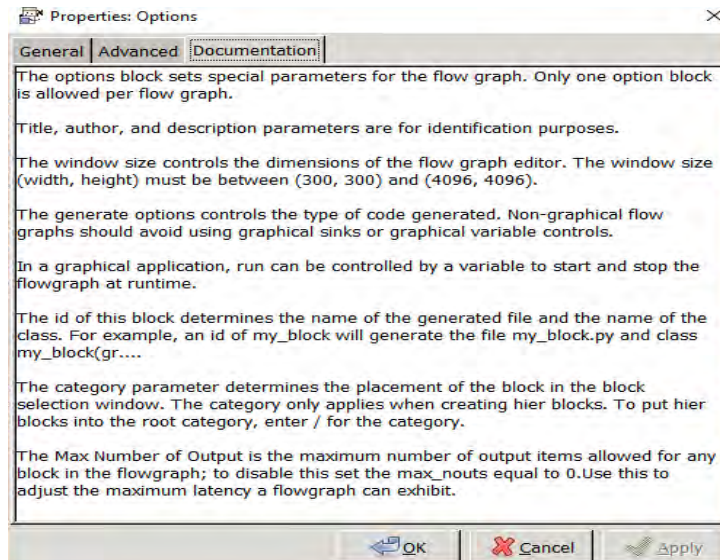


Figura 2.9: Documentación de los bloque de GRC

A continuación se definen algunos de los términos y conceptos utilizados en el entorno de GNU Radio:

- *Scheduler*. En GNU Radio, los diagramas de bloques se ejecutan de forma concurrente, cada bloque tiene su propio hilo. El *scheduler* es el motor interno de GNU Radio. Se ocupa de la coordinación de todos los hilos para que los recursos dedicados a cada uno estén equilibrados.
- *Puerto*. Punto de entrada o salida de un bloque. Los datos manejados pueden ser de varios tipos, de los cuales se muestran los más utilizados durante este trabajo en la Fig. 2.13.
- *Bloque*. Representan un subsistema que realiza una determinada operación. Pueden tener cualquier número de puertos entrada/salida. Pueden ser construidos de forma programática (en *C++* o *Python*) o gráficamente mediante un diagrama de bloques jerárquico. Desde el punto de vista del código, un bloque no es más que un objeto de una clase con un formato determinado en el que se definen las interfaces de entrada/salida y se implementan las operaciones de procesamiento de señal.
- *Diagrama de bloques*. Como su nombre indica, los también llamados flowgraphs interconectan bloques generando un sistema completo listo para ser ejecutado. Aunque existe la

posibilidad de crearlo en C++, normalmente lo veremos como un *script Python* generado automáticamente si estamos utilizando el GRC.

- *Bloque jerárquico*. Son bloques especiales en cuanto a la forma de ser contruidos. En vez de estar definidos por código directamente, se construyen mediante un diagrama de bloques en GRC. Por tanto, su comportamiento depende del de otros bloques. Ayuda a mantener más ordenados los *flowgraphs*.
- *Módulo*. Los bloques en GNU Radio están organizados en paquetes software o módulos, normalmente según las aplicaciones a las que estén destinados, y desarrollados por diferentes agentes. Los módulos son fácilmente identificables por su nombre: todos ellos comienzan por "gr-". Ejemplos de módulos son *gr-osmocom* y *gr-digital*.

2.6.2. Diagrama de flujo en el entorno de GRC.

Ahora que se tiene una mejor comprensión de cómo encontrar bloques, cómo agregarlos al espacio de trabajo y cómo editar las propiedades del bloque, procedamos a construir el siguiente diagrama de flujo de una Fuente de señal que se envía a un Bloque del acelerador y luego a nuestro Time Sink haciendo clic en los puertos / pestañas de colores del tipo de datos uno tras otro para hacer las conexiones:

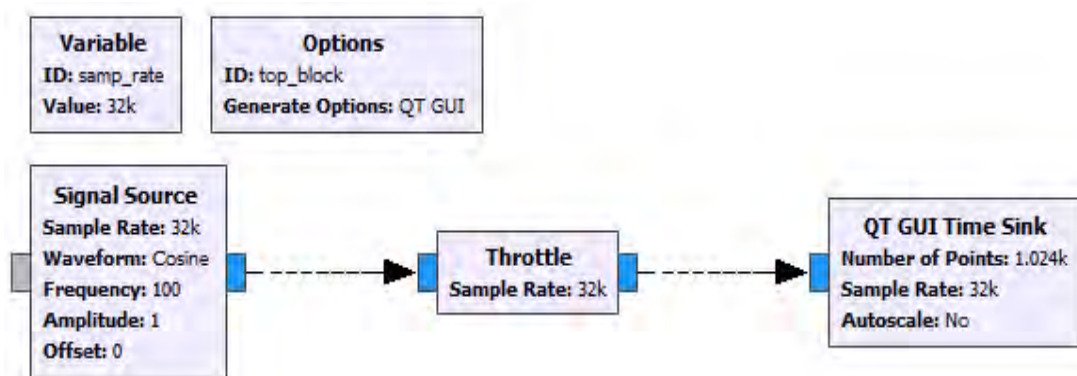


Figura 2.10: Gráfico de flujo GRC.

Una nota sobre el bloque del `throttle`: limita la velocidad del diagrama para que este diagrama trabaje a la velocidad del proceso que se indica en el diagrama de flujo es decir este bloque acelera el diagrama de flujo para asegurarse de que no consuma el 100 % de los ciclos

del CPU y que su computadora no responda.

Con un diagrama de flujo utilizable ahora podemos proceder a discutir la barra de herramientas.



Figura 2.11: Barra de herramientas GRC.

Esta sección de la interfaz contiene comandos presentes en la mayoría de los programas, como nuevo, abrir, guardar, copiar, pegar. Las herramientas importantes aquí son Generar diagrama de flujo, Ejecutar diagrama de flujo y Eliminar diagrama de flujo, todos accesibles a través de F5, F6 y F7 respectivamente.

Se examina la salida.

Al ejecutar el diagrama de flujo, muestra la onda sinusoidal.

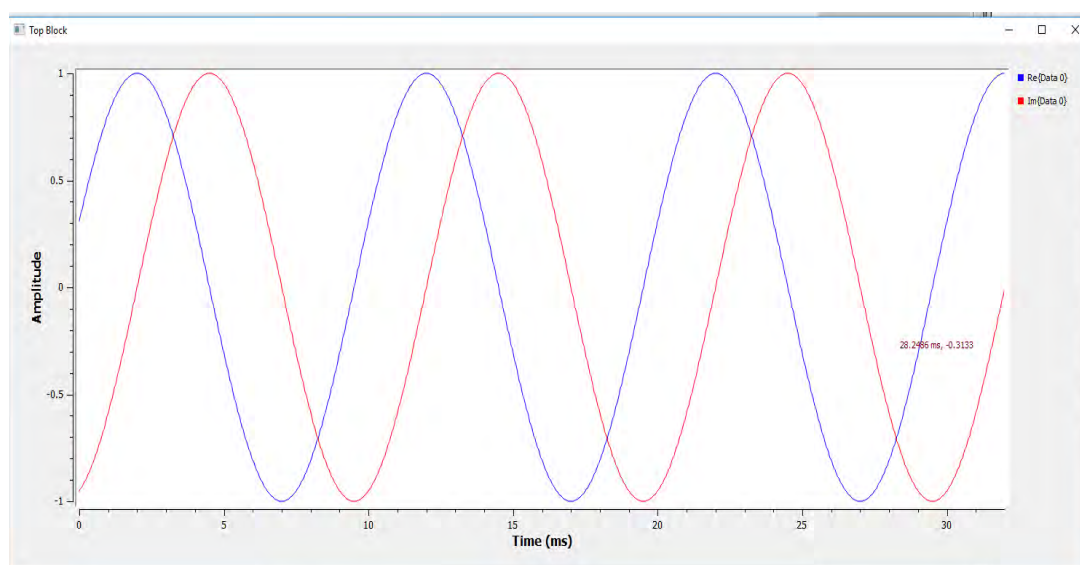


Figura 2.12: Señal sinusoidal.

De lo anterior se obtiene una señal sinusoidal compleja de la forma e^{jw} . Para detener la ejecución del diagrama de flujo se utiliza el botón Kill flowgraph o simplemente cerrando la GUI de Time Sink. Ahora es un buen momento para revisar los tipos de datos en GNU Radio abriendo la ventana help y seleccionamos Types la cual nos muestra la siguiente ventana mostrada en la Fig. 2.13.

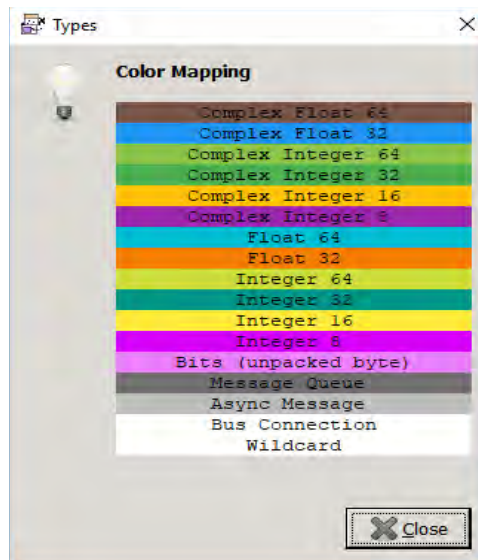


Figura 2.13: Tipos de datos GRC.

De la figura 2.10 se observa que los bloques utilizados tienen puertos azules que son del tipo *Complex Float 32*, lo que significa que contienen una parte real e imaginaria, cada uno de los cuales es del tipo *Float 32*. Podemos razonar que cuando *Time Sink* toma un tipo de datos complejo, genera tanto la parte real como la imaginaria en canales separados.

2.7. Radio Definido por Software SDR.

El software radio es un radio donde las señales portadoras que se van a modular y transmitir por un canal son definidas mediante software. Es decir, las formas de onda son generadas como señales digitales muestreadas, convertidas de digital a analógico a través de un Conversor Digital Analógico (DAC) de banda ancha y con la posibilidad de subir la señal de Frecuencia Intermedia (IF) a Radiofrecuencia (RF). De la misma manera, el receptor emplea un Conversor Analógico Digital (ADC) de banda ancha que captura todos los canales del nodo del software radio. Entonces, el receptor extrae, baja y demodula la señal utilizando software en un ordenador de propósito general.

Por lo tanto, se puede ver el concepto de Software Defined Radio como una tecnología cuyo principal objetivo es que el procesamiento de la señal se realice a través de software y que, además, dicho procesamiento se acerque lo máximo posible a la antena, convirtiendo los problemas que hasta ahora habían recaído en el hardware (mezcladores, filtros, amplificadores, moduladores,

demoduladores, detectores, etc.) en problemas de tipo software.

2.7.1. Sistemas de Software Definido por Radio.

En un sistema SDR ideal la digitalización de la señal se realizaría a pie de antena. Es decir, tanto en el transmisor como en el receptor, todo sería configurado mediante software a excepción de los conversores y de la antena, tal como se muestra en la siguiente figura (Sabate, 2016):

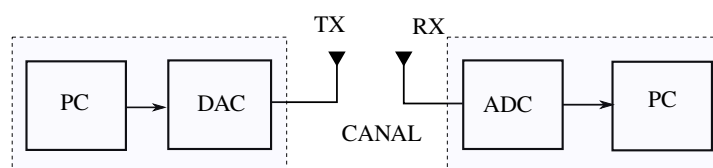


Figura 2.14: Sistema de un SDR ideal.

Por consiguiente, los problemas actuales surgidos del comportamiento no ideal de mezcladores, filtros, amplificadores, etc., se solucionarían y la distorsión introducida en la señal transmitida se vería minimizada en recepción. Sin embargo, el estado actual de la tecnología hace que un sistema SDR ideal no pueda implementarse dado que hay una serie de limitaciones:

- **Limitación en el ADC.** Como se sabe el teorema de *Nyquist* establece que, para evitar el *aliasing* (solapamiento en el espectro) de una señal muestreada, la frecuencia mínima de muestreo ha de ser superior al doble de la frecuencia máxima de la señal a muestrear.

$$f_{Nyquist} \geq 2f_{max} \quad (2.23)$$

Esto implica que, sin filtrado previo, si trasmitiéramos directamente una señal de RF en la banda de 2.5 GHz al conversor analógico digital, este debería soportar un ancho de banda de 5 GHz. Y, a pesar de que los conversores actuales soportan tasas de muestreo altas, de momento los anchos de banda máximo soportados son de cientos de MHz.

Por otra parte, en la práctica, hay que eliminar el ruido y todas aquellas señales indeseadas que se encuentren fuera del espectro de *Nyquist*, ya que si no se suprimen provocan aliasing al realizar el muestreo. Por tanto, es necesario utilizar un filtro paso-bajo analógico de frecuencia de corte $f_{Nyquist}/2$ (filtro *anti-aliasing*) antes de muestrear.

Todo esto es en relación a la etapa de muestreo de un conversor analógico digital. En cuanto a la etapa de cuantificación, también tenemos limitaciones. Si la señal recibida sufre una gran atenuación y los niveles de potencia son muy bajos en comparación al rango dinámico del ADC (bit de resolución), diferentes muestras que correspondían a diferentes niveles podrían ser cuantificados en un mismo estado y, por tanto, la señal no sería cuantificada correctamente.

- **Limitación en los buses.** La velocidad de transferencia máxima de los buses oscila normalmente entre varios cientos de Mbps hasta algún Gbps. Este hecho, junto al hecho de que no todos bits transmitidos son de datos, crean una restricción en la tasa binaria máxima del sistema.
- **Limitación en la PC.** El procesamiento en tiempo real de todos los datos que el sistema genera requiere de ordenadores potentes que sean capaces de realizar cálculos matemáticos avanzados con velocidades elevadas y de forma concurrente. Debido a estas limitaciones y con el objetivo de mejorarlas, a la arquitectura del sistema que se muestra en la figura 2.14 se le introduce unos cuantos bloques de hardware adicional que permiten, entre otras cosas, filtrar en RF, amplificar, bajar la señal a IF y procesarla en banda base. En la siguiente figura se muestra la estructura de un SDR:

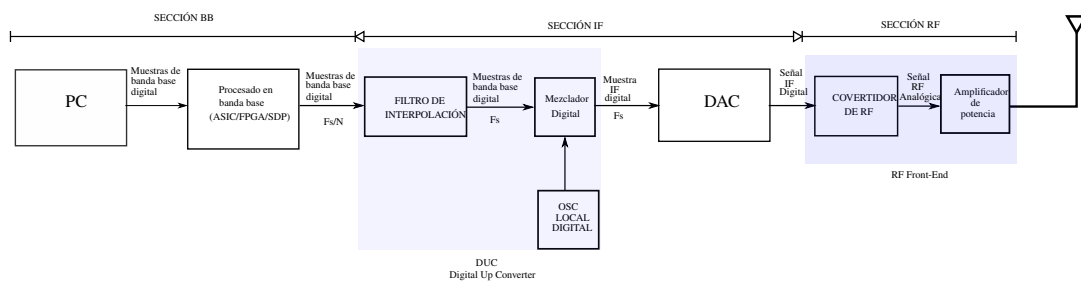


Figura 2.15: Estructura de SDR para un transmisor (Sabate, 2016).

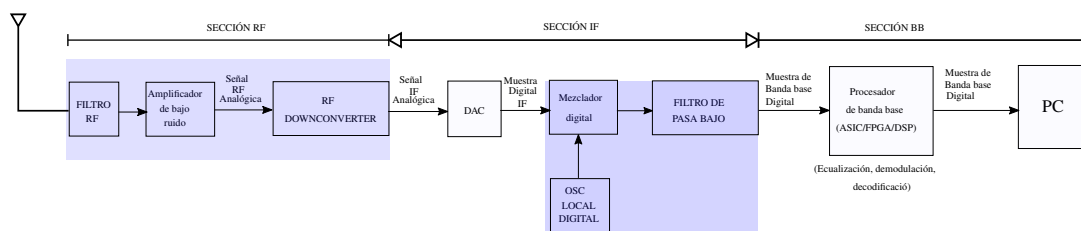


Figura 2.16: Estructura de SDR para un receptor (Sabate, 2016).

la estructura del sistema de SDR de la figura 2.15 y 2.16 comprenden de tres secciones.

■ **Etapa de Radiofrecuencia (RF) tiene como función:**

- Modular y amplificar las señales de IF ajustandoles, de manera que, posteriormente sean transmitidos vía radio, en transmisión.
- Adecuar las señales de RF mediante filtrado y convertirlas a frecuencia intermedia, en recepción.

■ **La etapa de Frecuencia intermedia (IF) se ocupa de:**

- Convertir la señal de banda base a IF (DUC, Digital Up Conversion) y posteriormente convertir dichas señales digitales a señales analógicas (DAC), en transmisión.
- Digitalizar las señales IF (ADC) y después convertirlas a banda base (DDC, Digital Down Conversion), en recepción.

■ **Etapa de Banda Base:** Se encarga, tanto en la transmisión como en la recepción, de todo el procesamiento digital en banda base de las señales.

2.7.2. Universal Software Radio Peripheral (USRP).

El Universal Software Radio Peripheral (USRP) es básicamente un cabezal de radiofrecuencia, de configuración flexible, comercializado por *Ettus Research*, y que está diseñado para trabajar en conjunto con un procesador externo (PC, Workstation, etc.) a través de una FPGA, a la que se conecta mediante un bus Ethernet, y que permite la realización de sistemas de Radio Definido por Software SDR.

USRP NI 2920.

El USRP-2920 es un transceptor de RF ajustable con un convertidor analógico-digital de alta velocidad y un convertidor digital-analógico para la transmisión de señales de banda base I y Q a una PC principal por 1 Gigabit Ethernet. Se puede usar el NI USRP-2920 para las siguientes aplicaciones de comunicaciones: espacio en blanco; emisión FM; seguridad pública;

dispositivos móviles terrestres, sin licencia de baja potencia en bandas industriales, científicas y médicas (ISM); redes de sensor; teléfono celular o GPS.

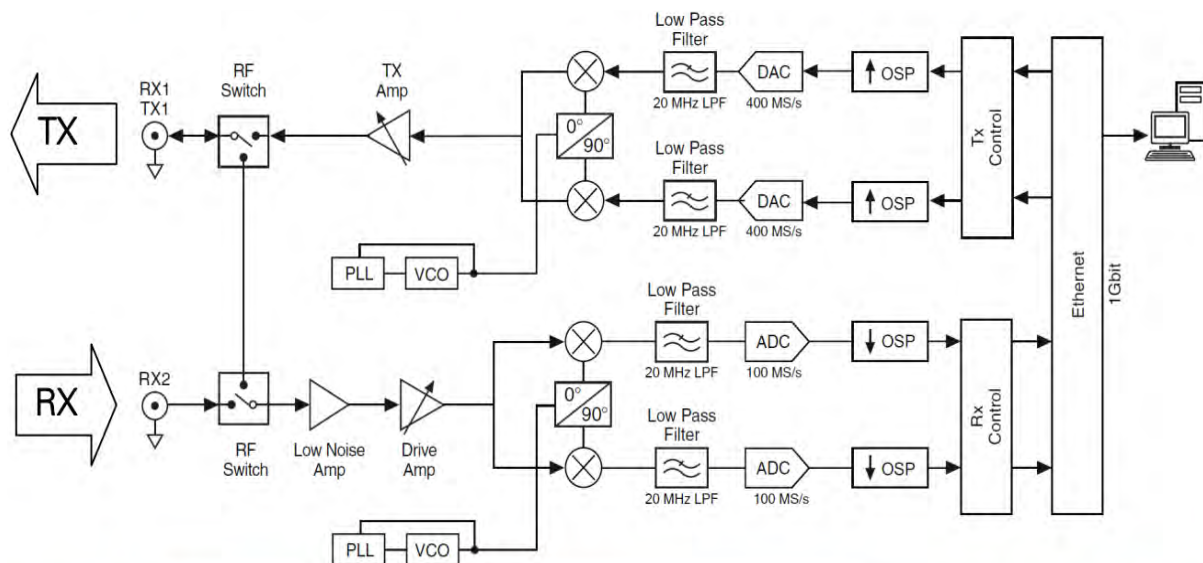


Figura 2.17: Diagrama de bloque del USRP NI 2920 (Instrument, 2013)

De la Fig. 2.17, las señales entrantes se conectan a los conectores SMA estándar se mezclan usando un receptor de conversión directa (DCR) a la banda base I/Q que se muestrean mediante un convertidor analógico/digital (ADC) de 2 canales, 100 MS/s , de 14 bits (Instrument, 2013). Los datos I/Q digitalizados siguen caminos paralelos a través de un proceso de conversión digital descendente (DDC) que mezcla, filtra y diezma la señal de entrada de 100 MS/s a una tasa especificada por el usuario. La conversión descendente las muestras, cuando se representan como números de 32 bits (16 bits cada uno para I y Q), se pasan al huésped computadora a hasta 20 MS/s sobre una conexión Gigabit Ethernet estándar.

Para la transmisión, muestras de señales I/Q de banda base son sintetizadas por el ordenador central y alimentadas al USRP-2920 a hasta 20 MS/s sobre Gigabit Ethernet cuando se representa con 32-bits (16-bits cada uno para el I y Q). El hardware del USRP interpola la señal entrante a 400 MS/s usando un proceso de conversión ascendente (DUC) y luego convierte la señal a analógica con un convertidor digital-analógico (DAC) de 16 bits de doble canal. La señal analógica resultante se mezcla entonces con la frecuencia portadora especificada.

Las especificaciones son validas a 25°C

- Transmisor.

Frequency range	50 MHz to 2.2 GHz
Frequency step	<1 kHz
Maximum output power (P_{out})	
50 MHz to 1.2 GHz	50 mW to 100 mW (17 dBm to 20 dBm)
1.2 GHz to 2.2 GHz	30 mW to 70 mW (15 dBm to 18 dBm)
Gain range ^[1]	0 dB to 31 dB
Gain step	1.0 dB
Frequency accuracy ^[2]	2.5 ppm
Maximum instantaneous real-time bandwidth ^[3]	
16-bit sample width	20 MHz
8-bit sample width	40 MHz
Maximum I/Q sample rate ^[4]	
16-bit sample width	25 MS/s
8-bit sample width	50 MS/s
Digital-to-analog converter (DAC)	2 channels, 400 MS/s, 16 bit
DAC spurious-free dynamic range (sFDR)	80 dB

Tabla 2.1: Especificaciones del transmisor de USRP NI 2920.

■ Receptor

Frequency range	50 MHz to 2.2 GHz
Frequency step	<1 kHz
Gain range ^[5]	0 dB to 31.5 dB
Gain step	0.5 dB
Maximum input power (P_{in})	0 dBm.
Noise figure	5 dB to 7 dB
Frequency accuracy ^[6]	2.5 ppm
Maximum instantaneous real-time bandwidth ^[7]	
16-bit sample width	20 MHz
8-bit sample width	40 MHz
Maximum I/Q sample rate ^[8]	
16-bit sample width	25 MS/s
8-bit sample width	50 MS/s
Analog-to-digital converter (ADC)	2 channels, 100 MS/s, 14 bit
ADC sFDR	88 dB

Tabla 2.2: Especificaciones del receptor de USRP NI 2920.

1. La potencia de salida resultante del ajuste de la ganancia varía a lo largo de la banda de frecuencia y entre los dispositivos.
2. La precisión de la frecuencia se basa en las especificaciones del fabricante del oscilador de cristal compensado por temperatura (TCXO) y no se mide. Alternativamente, puede incorporar una fuente de referencia externa para proporcionar un Reloj de Referencia de frecuencia más preciso y para lograr una mejor precisión de frecuencia.
3. El ancho de banda instantáneo depende de muchos factores, entre los que se incluyen la configuración de la red y el rendimiento de la computadora central. El rendimiento real de los datos puede depender del conjunto de chips.
4. La tasa de muestreo I/Q depende de muchos factores, entre ellos, la configuración de la red y el rendimiento del ordenador central. El rendimiento real de los datos puede depender del conjunto de chips.
5. La amplitud de la señal recibida que resulta del ajuste de ganancia varía en la banda de frecuencia y entre los dispositivos.

6. La precisión de la frecuencia se basa en las especificaciones del proveedor del oscilador de cristal compensado por temperatura (TCXO) y no se mide. Alternativamente, puede incorporar una fuente de referencia externa para proporcionar un reloj de referencia de frecuencia más preciso y lograr una mejor precisión de frecuencia.
7. El ancho de banda instantáneo depende de muchos factores, incluidos, entre otros, la configuración de red y el rendimiento de la computadora host. El rendimiento real de los datos puede depender del conjunto de chips.
8. La tasa de muestreo I/Q depende de muchos factores, entre ellos, la configuración de la red y el rendimiento del ordenador central. El rendimiento real de los datos puede depender del chipset.

Capítulo III

ANÁLISIS DEL CANAL DE TRANSMISIÓN

(Pätzold, 2011) En sistemas de comunicaciones inalámbricas, las ondas electromagnéticas emitidas a menudo no alcanzan la antena receptora directamente debido a los obstáculos que la bloquean en el camino de la línea de visión. De hecho las ondas recibidas son una superposición de ondas provenientes de todas las direcciones debido a la reflexión, difracción y dispersión causadas por edificios, árboles y otros obstáculos esto se conoce como propagación por trayectos múltiples.

Los métodos para modelar canales de transmisión serán útiles para modelar las propiedades estadísticas de los canales del mundo real (medidos) con respecto a la función de densidad de probabilidad. La utilidad y la calidad de un modelo de referencia y el modelo de simulación correspondiente a un canal de transmisión, es evaluada en la última instancia, en función de la forma en que sus parámetros estadísticos individuales puedan adaptarse a las propiedades estadísticas de los canales de referencia, de tal manera que el canal simulado sea lo más semejante a un canal real.

3.1. Canal Gaussiano.

(Wyglinski, Getz, Collins, y Pu, 2018) Las variables aleatorias normales se han usado a menudo para modelar el ruido introducido dentro de un canal de comunicación. De hecho,

muchos análisis de los sistemas de comunicación y su rendimiento a menudo se realizan suponiendo que el canal ruidoso posee un comportamiento aleatorio Gaussiano. En consecuencia, esto hace que la variable aleatoria Gaussiana sea más utilizada en el estudio de los sistemas de comunicación.

3.1.1. Ruido blanco Gaussiano aditivo (AWGN).

El ruido puede provenir tanto de fuentes naturales como de origen humano, como el ruido térmico debido a la agitación térmica de los electrones en las líneas de transmisión, otros transmisores inalámbricos o incluso otros conductores. Se sabe que la combinación de tales fuentes de ruido posee una distribución aproximadamente Gaussiana. La función de densidad de probabilidad Gaussiana se define como.

$$P_{\mu}(x) = \frac{1}{\sqrt{2\pi}\sigma_{\mu}} e^{-\frac{(x-m_{\mu})^2}{2\sigma_{\mu}^2}} \quad (3.1)$$

Donde m_{μ} es la media de la variable aleatoria Gaussiana μ , σ_{μ}^2 es la varianza de la variable μ . En el caso de $m_{\mu} = 0$ y $\sigma_{\mu}^2 = 1$ nos referimos a μ como una variable aleatoria normal estándar como se muestra en la Fig. 3.1.

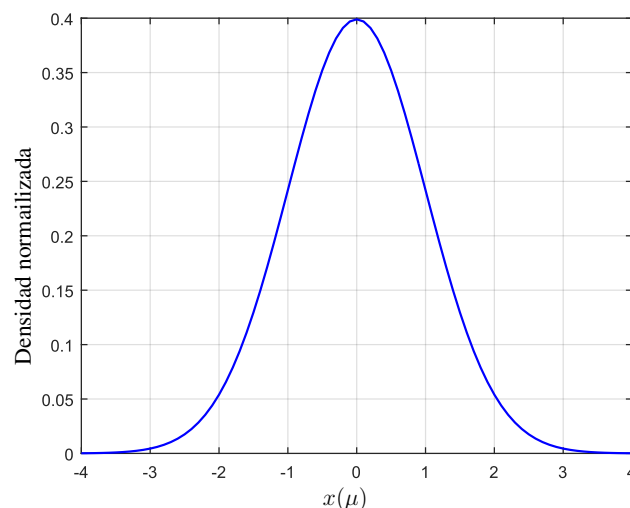


Figura 3.1: Función de densidad de probabilidad para una variable aleatoria Gaussiana de $N \sim (0, 1)$.

3.1.2. Relación señal a ruido (SNR).

La relación señal a ruido se define como la relación entre la potencia de una señal (información significativa) y la potencia del ruido de fondo (señal no deseada) este margen es medido en decibelios como se muestra en la siguiente ecuación:

$$SNR = 10 \log_{10} \left(\frac{S}{N} \right) \quad (dB) \quad (3.2)$$

Donde S potencia de la señal y N potencia de la densidad espectral del ruido. El rango dinámico y relación señal a ruido para referirse a este margen que hay entre el ruido de fondo y nivel de referencia, pueden utilizarse como sinónimos. No ocurre lo mismo, cuando el rango dinámico indica la distancia entre el nivel de pico y el ruido de fondo. Que en las especificaciones técnicas de un equipo aparezca la relación señal/ruido indicada en decibelios no significa nada si no va acompañado por los puntos de referencia utilizados y las ponderaciones como se muestra en Fig. 3.2.

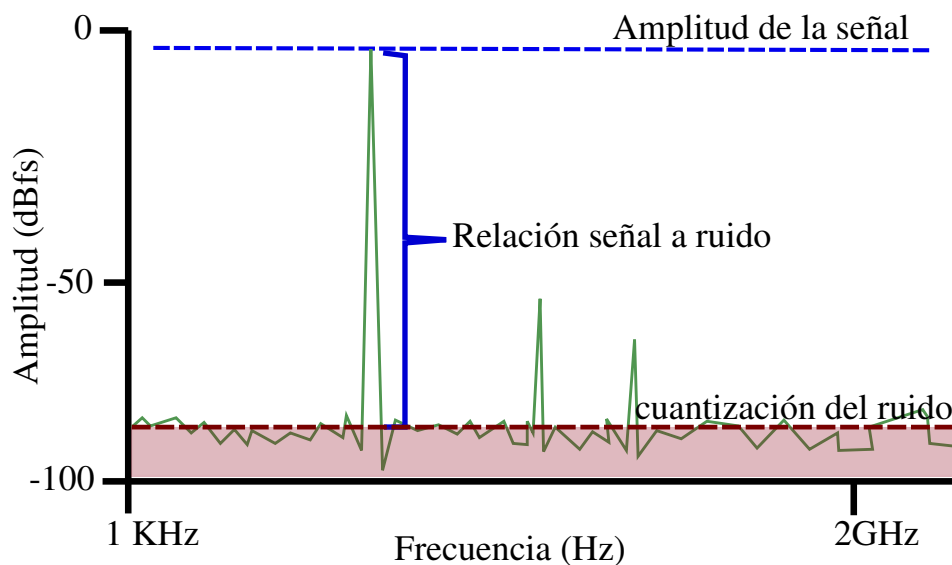


Figura 3.2: Rango dinámico de la relación señal a ruido.

Otra forma de evaluar la relación señal a ruido.

La potencia de la señal como la del ruido deben medirse en el mismo o en puntos equivalentes de un sistema, y dentro del mismo ancho de banda del sistema. Si se conoce la varianza de la señal y del ruido, con media igual a cero, la SNR puede calcular en la siguiente ecuación.

(Guangliang Ren, Yilin Chang, y Hui Zhang, 2005).

$$SNR = \frac{\sigma_{señal}^2}{\sigma_{ruido}^2} \quad (3.3)$$

La ecuación (3.3) se tiene que evaluar en decibelios, como se muestra en la siguiente ecuación (3.4).

$$SNR = 10 \log_{10} \left(\frac{\sigma_{señal}^2}{\sigma_{ruido}^2} \right) \quad (dB) \quad (3.4)$$

La señal que llega hacia el receptor es afectado por el canal Gaussiano, para generar y variar el nivel del ruido despejamos la desviación estándar del ruido de (3.4).

$$\sigma_{noise} = \sqrt{\sigma_{señal}^2 10^{\frac{SNR}{10}}} \quad (3.5)$$

Se hace uso de software Matlab para simular el canal Gaussiano, donde se consideran los siguientes parámetros:

- N : Numero de muestra.
- f_i : Frecuencia de la información.
- $F_s = 50 * f_i$: Frecuencia de muestreo.
- $f(n) = A_m \sin(2\pi n/50)$: Señal de información enviada.

Cabe indicar que la frecuencia de muestreo se selecciona de manera aleatoria, considerando el teorema de muestreo de Nyquist. Con la finalidad de mostrar el comportamiento de una señal a través del canal AWGN. Para generar el ruido AWGN se utiliza el comando de MATLAB `normrnd`.

- $N = \text{normrnd} [\mu_r, \sigma_{ruido}, [1 \quad N]]$
- $S_{recibido} = f(n) + N(n)$, Señal de información recibida.

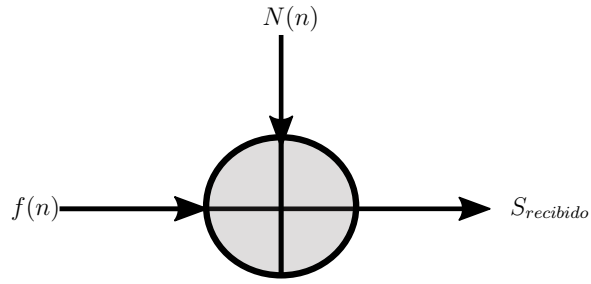


Figura 3.3: Diagrama del modelo de canal AWGN.

A continuación se muestra la simulación, de la información transmitida a través del canal AWGN, en el software de `Matlab`:

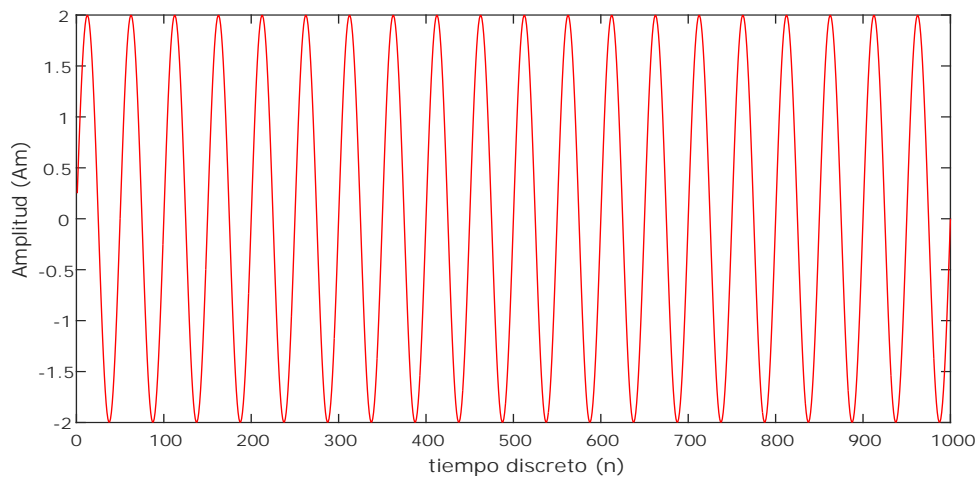


Figura 3.4: Señal de información $f(n)$.

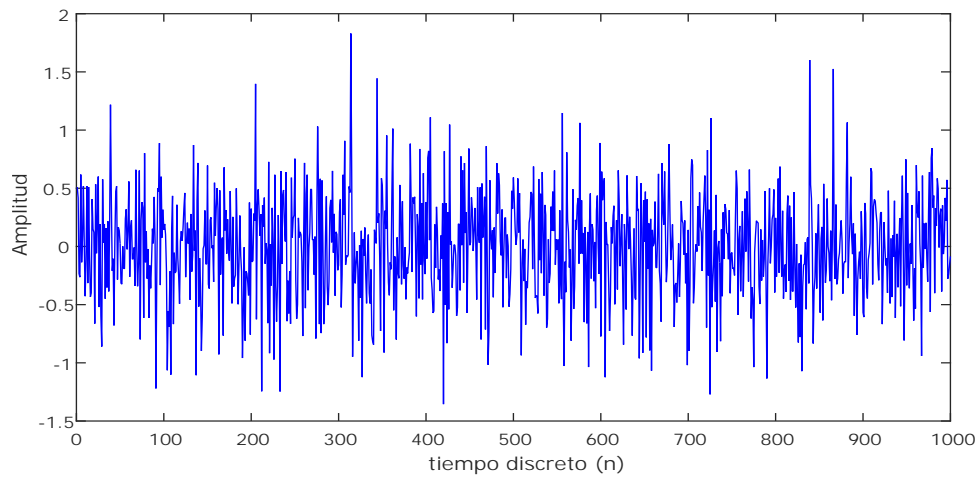


Figura 3.5: Señal no deseada AWGN.

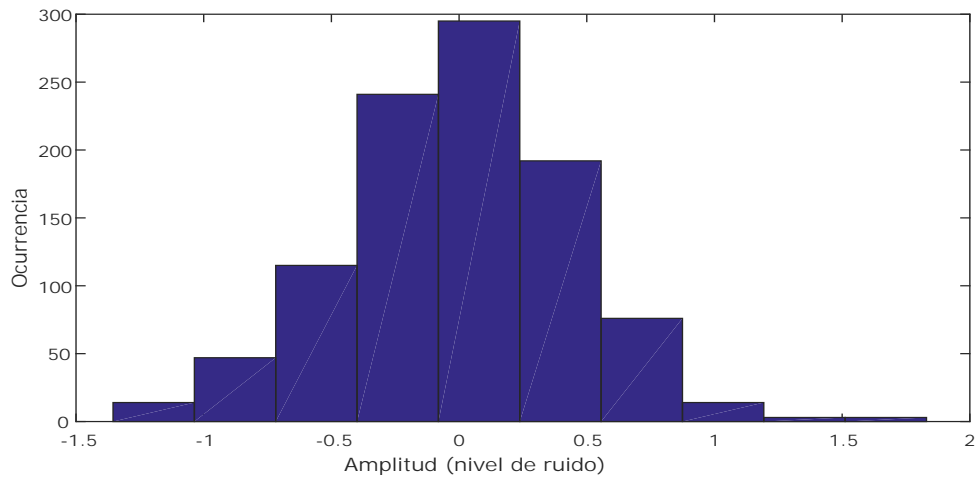


Figura 3.6: Histograma de la señal no deseada AWGN.

Resultados, de la señal de información recibida, para los niveles de; SNR = 2dB, SNR= 10dB y SNR = 30dB.

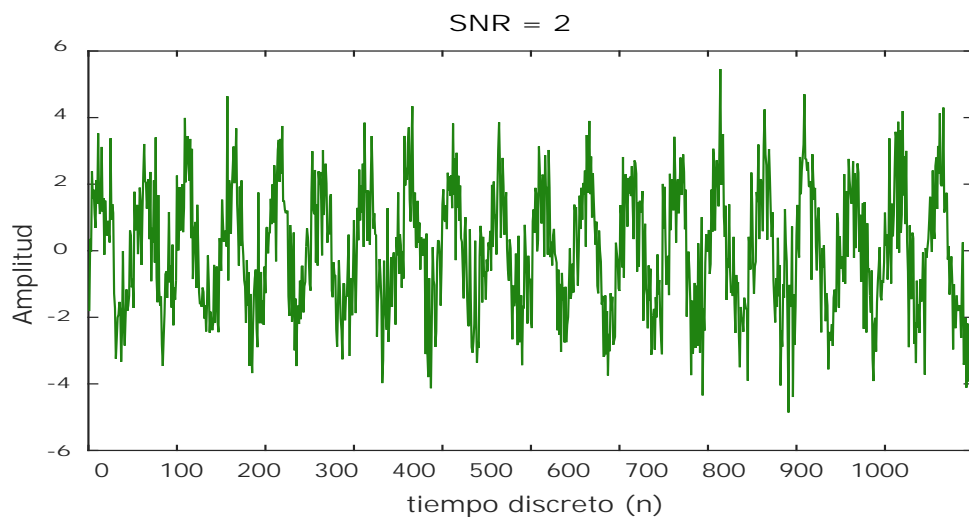


Figura 3.7: Señal de información afectado por AWGN, para SNR = 2 dB.

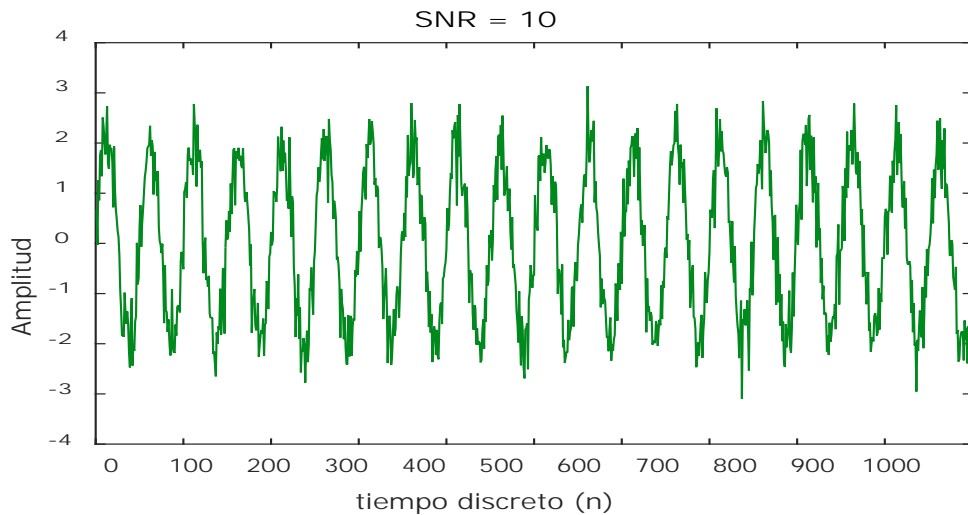


Figura 3.8: Señal de información afectado por AWGN, para SNR = 10 dB.

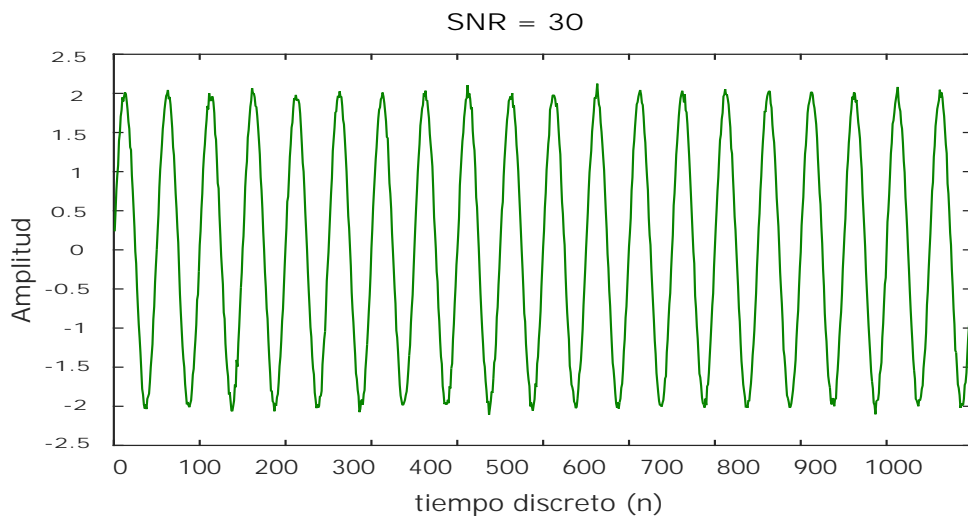


Figura 3.9: Señal de información afectado por AWGN, para SNR = 30 dB.

De las anteriores figuras 3.7, 3.8 y 3.9 se deduce, mientras el nivel de la SNR disminuye la señal de información es vulnerable al ruido esto provocaría la pérdida de la información, sin embargo cuando el nivel de la SNR incrementa la señal podría llegar a ser inmune al ruido.

3.2. Algoritmos de estimación de la relación señal a ruido.

La estimación de la relación señal-ruido (SNR) es un problema esencial en los sistemas de comunicaciones modernos. Muchas aplicaciones requieren que se conozca la SNR para que funcione correctamente, por ejemplo, el control de la potencia en los sistemas de acceso múltiple por división de códigos, la adaptación de los enlaces en los sistemas de modulación adapta-

tiva y codificación, y la decodificación iterativa para los códigos Turbo y de comprobación de paridad de baja densidad (LDPC).

Se han realizado varios estudios de algoritmos de estimación de la SNR para la modulación BPSK y QPSK, que se usa ampliamente en comunicaciones móviles y asignación de canales, pero la mayoría de los algoritmos de estimación de la SNR funcionan muy bien cuando la SNR es alta, pero sus rendimientos no son satisfactorios cuando la SNR es menor. La interferencia y el ruido en el canal a menudo se supone que es el ruido blanco aditivo Gaussiano (AWGN) en muchos casos, en las comunicaciones móviles y satelitales.

En esta sección se realiza estudio y análisis de cuatro algoritmos de estimación de la relación señal a ruido, además estos algoritmos se compararan, para verificar que algoritmo ofrece mejor desempeño, el algoritmo de mejor rendimiento, será implementado en plataforma de SDR.

3.2.1. Modelo de la señal y estimación de la SNR.

En este sistema el modelo de la señal y los algoritmos de estimación de la relación señal a ruido se plantea de la siguiente manera.

Sean S_{Ik} y S_{Qk} , $k = 1, 2, 3, \dots, L$, los componentes en fase y cuadratura de un símbolo pertenecientes a la constelación de una modulación por desplazamiento por fase (PSK) transmitido sobre el canal AWGN. Se supone que los componentes de la señal son variables aleatorias discretas e independientes. Suponiendo que se tome una muestra compleja para cada uno de los símbolos transmitidos y que exista sincronización de portadora, la señal recibida $Z_k = X_k + jY_k$ se puede describir como:

$$X_k = S_{Ik} + n_{Ik} \quad (3.6)$$

$$Y_k = S_{Qk} + n_{Qk} \quad (3.7)$$

Donde, X_k y Y_k representan las componentes en fase y cuadratura respectivamente de Z_k , mientras que n_{Ik} y n_{Qk} representan las componentes en fase y cuadratura de la señal no deseada (ruido), que es independiente y distribuida idénticamente con media igual a cero, la variable aleatoria Gaussiana con varianza $\sigma_I^2 = \sigma_Q^2 = \sigma^2$ respectivamente. La SNR de la señal recibida

ρ esta dada por:

$$\rho = \frac{E\{S_{I_k}^2 + S_{Q_k}^2\}}{E\{n_{I_k}^2 + n_{Q_k}^2\}} \quad (3.8)$$

3.2.2. Algoritmo de segundo y cuarto momento M_2M_4 .

Este algoritmo estima la potencia de la señal y el ruido en función a momentos de segundo y cuarto orden de la señal recibida, como se indica a continuación (Ijaz, Awoseyila, y Evans, 2012).

$$\hat{S}_{M_2M_4} = \sqrt{2M_2^2 - M_4} \quad (3.9)$$

$$\hat{N}_{M_2M_4} = M_2 - \sqrt{2M_2^2 - M_4} \quad (3.10)$$

Donde M_2 y M_4 se determina de la siguiente forma:

$$\hat{M}_2 = \frac{1}{L} \left[\sum_{k=1}^L |X_k|^2 + \sum_{k=1}^L |Y_k|^2 \right] \quad (3.11)$$

$$\hat{M}_4 = \frac{1}{L} \left[\sum_{k=1}^L |X_k|^4 + \sum_{k=1}^L |Y_k|^4 \right] \quad (3.12)$$

Por lo tanto la SNR estimada se calcula en la siguiente ecuación:

$$\hat{\rho} = \frac{\hat{S}_{M_2M_4}}{\hat{N}_{M_2M_4}} \quad (3.13)$$

3.2.3. Algoritmo propuesto por esha Ijaz, Adegbeniga B. Awoseyila, and Barry G. Evans.

Estima la SNR para modulaciones BPSK y QPSK, basado en observaciones de que los valores absolutos de fase y cuadratura de la señal recibida tiene una estrecha relación con la potencia de señal, ya que estos componentes tienen una amplitud constante en la señal de transmisión (Ijaz y cols., 2012).

En consecuencia de la potencia de las muestras de la señal recibida proponen de la siguiente manera.

$$\hat{S} = \frac{1}{L^2} \left[\left(\sum_{k=1}^L |X_k| \right)^2 + \left(\sum_{k=1}^L |Y_k| \right)^2 \right] \quad (3.14)$$

Para calcular la SNR estimada se hace uso del segundo momento M_2 .

$$\hat{\rho} = \frac{\hat{S}}{M_2 - \hat{S}} \quad (3.15)$$

Donde M_2 se calcula en (3.11).

3.2.4. Estimador de relación señal-variación (SVR).

Es un método que se basa en momentos de desarrollo para monitorear la calidad de canal en canales por desvanecimiento por trayectos múltiples, pero también se puede aplicar como una medida del canal AWGN. Este estimador fue diseñado para operar en cualquier señal modulada QPSK. A continuación, esbozamos la derivación del estimador SVR en servicio para canales complejos (Pauluzzi y Beaulieu, 1995).

El estimador SVR es una función del parámetro:

$$\beta = \frac{E\{Z_k Z_k^* Z_{k-1} Z_{k-1}^*\}}{E\{(Z_k Z_k^*)^2\} - E\{Z_k Z_k^* Z_{k-1} Z_{k-1}^*\}} \quad (3.16)$$

El termino $E\{(Z_k Z_k^*)^2\}$ es el momento de cuarto orden, que se muestra en la ecuación (3.12), simplificamos (3.16).

$$E\{Z_k Z_k^* Z_{k-1} Z_{k-1}^*\} = S^2 + 2SN + N^2 \quad (3.17)$$

Aprovechando las independencias de la señal y las muestras del ruido. Escribimos S/N como ρ sustituyendo en (3.17) y en $M_4 = kaS^2 + 6SN + kwN^2$ se obtiene.

$$\beta = \frac{\rho^2 + 2\rho + 1}{(ka - 1)\rho^2 + 2\rho + (kw - 1)} \quad (3.18)$$

El estimador SVR , puede ser resuelto por $\hat{\rho}$ para un canal complejo de la siguiente manera.

$$\hat{\rho}_{SVR,Complejo} = \frac{(\beta - 1) \pm \sqrt{(\beta - 1)^2 - [1 - \beta(ka - 1)][1 - \beta(kw - 1)]}}{1 - \beta(ka - 1)} \quad (3.19)$$

Para la señal PSK *M-aria* para un canal complejo AWGN $ka = 1$ y $kw = 2$ se obtiene la siguiente ecuación.

$$\hat{\rho}_{SVR,complejo} = \beta - 1 + \sqrt{\beta(\beta - 1)} \quad (3.20)$$

En la practica, el parámetro β se estima con su correspondiente promedio de tiempo como:

$$\beta = \frac{\frac{1}{L-1} \sum_{k=1}^{L-1} |Z_k|^2 |Z_{k-1}|^2}{\frac{1}{L-1} \sum_{k=1}^{L-1} |Z_k|^4 - \frac{1}{L-1} \sum_{k=1}^{L-1} |Z_k|^2 |Z_{k-1}|^2} \quad (3.21)$$

Esta estimación se puede aplicar a los canales tanto reales como complejos, se debe tener en cuenta que la recuperación de la fase de portadora no es necesaria para este método, ya que la información de la fase se elimina en el proceso de estimación.

3.2.5. Algoritmo DDML.

Es un algoritmo que se basada en la teoría de estimación de máxima probabilidad. Por lo tanto, se hace necesario usar una combinación de algoritmos de datos (DA) y de decisión (DD) para el uso de datos de tráfico (Ijaz y cols., 2012).

$$\hat{S}_{DDML} = \left[\frac{\frac{1}{L} \sum_{k=1}^L (X_k \hat{S}_{Ik} + Y_k \hat{S}_{Qk})}{\frac{1}{L} \sum_{k=1}^L (\hat{S}_{Ik} + \hat{S}_{Qk})} \right]^2 \quad (3.22)$$

$$\hat{N}_{DDML} = \frac{1}{L} \sum_{k=1}^L (X_k^2 + Y_k^2) - \hat{S}_{DDML} \frac{1}{L} \sum_{k=1}^L (\hat{S}_{Ik} + \hat{S}_{Qk}) \quad (3.23)$$

Donde \hat{S}_{DDML} y \hat{N}_{DDML} son las respectivas estimaciones de la potencia de la señal de información y del ruido, \hat{S}_{Ik} y \hat{S}_{Qk} son las estimaciones de las componentes en fase y cuadratura de la información transmitida.

3.2.6. Comparación de Rendimiento.

El mejor estimador de la SNR es el que exhibe el sesgo más pequeño y tiene la varianza más pequeña. El error cuadrático medio MSE estadístico refleja tanto el sesgo como la varianza de una estimación de la SNR.

Error cuadrático Medio Normalizado (NMSE).

El error cuadrático medio da la medida de las diferencias en promedio entre los valores pronosticados y los observados.

El MSE (error cuadrático medio) estadístico refleja tanto el sesgo como la varianza de la estimación de la SNR y viene dado por (Pauluzzi y Beaulieu, 1995).

$$MSE\{\rho\} = E\{(\hat{\rho} - \rho)^2\} \quad (3.24)$$

Donde $\hat{\rho}$ es una estimación de la SNR, y ρ es el valor verdadero de la SNR. Se calcula la muestra MSE para cada estimador por simulación para una serie de estimaciones ρ de acuerdo con.

$$MSE\{\rho\} = \frac{1}{N_t} \sum_{i=1}^{N_t} (\hat{\rho}_i - \rho)^2 \quad (3.25)$$

El número de ensayos N_t se tiene que elegir suficientemente grande en todo los casos para garantizar un error confiable. La normalización por ρ^2 muestra explícitamente el comportamiento asintótico de los estimadores con la SNR creciente; por lo tanto, los resultados se presentan en términos del MSE normalizado.

$$NMSE\{\rho\} = \frac{MSE\{\rho\}}{\rho^2} = \frac{1}{N_t} \sum_{i=1}^{N_t} \left(\frac{\hat{\rho}_i}{\rho} - 1\right)^2 \quad (3.26)$$

Cramer Rao Bound (CRB).

Es un límite inferior en el error cuadrático medio normalizado NMSE, entonces, para evaluar el rendimiento de los algoritmos de estimación de la SNR, se debe calcular el límite de Cramer-Rao (CRB), donde la distancia entre esta varianza y el CRB permite evaluar la eficiencia del estimador. Para evaluar el rendimiento absoluto de cada estimador, se utiliza el CRB como referencia que se muestra en la siguiente ecuación.

$$CRB_{DA} = \frac{1}{L} \left(\frac{1}{N_{ss}} + \frac{2}{\rho} \right) \quad (3.27)$$

Donde N_{ss} número de muestras por símbolo.

3.2.7. Diagrama flujo de la implementación de los algoritmos de estimación de la SNR.

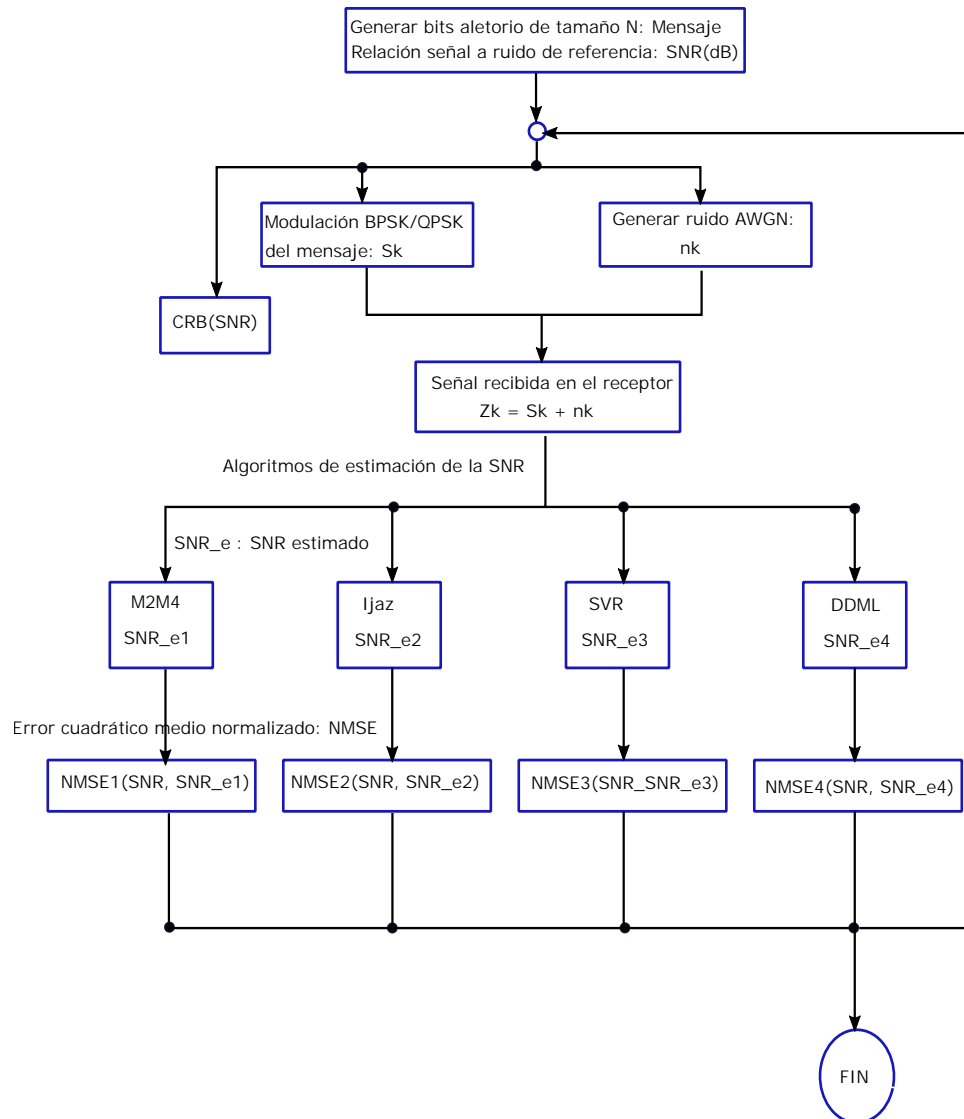


Figura 3.10: Diagrama de flujo de la simulación de los algoritmos de estimación de la SNR, para determinar el algoritmo de mejor desempeño.

3.2.8. Resultado de la simulación de los algoritmos de estimación de la SNR.

Del estudio y análisis de los algoritmos de estimación de la SNR, se utiliza el entorno de Matlab para verificar el desempeño de los algoritmos seleccionados para las modulaciones BPSK y QPSK, el algoritmo estimador de mejor rendimiento con respecto a otros, será implementado en plataforma de SDR.

Modulación BPSK.

La modulación consiste en el desplazamiento de fase para dos símbolos, es la modulación más sencilla por emplear dos símbolos con un bit de información cada uno, los símbolos suelen tener un valor de salto de fase 0° para el 1 y 180° para el cero (-1), como se muestra en el siguiente diagrama de constelación.

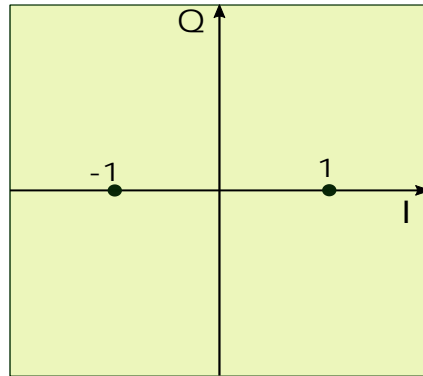


Figura 3.11: Diagrama de constelación de BPSK.

Resultado del desempeño de los algoritmos de estimación para BPSK.

Se genera bit aleatorios de tamaño $L = 5000$, a la cual se aplica la modulación digital BPSK, estos son transmitidos a través del canal AWGN, a continuación se muestra el diagrama de constelación en la Fig. 3.12 para BPSK.

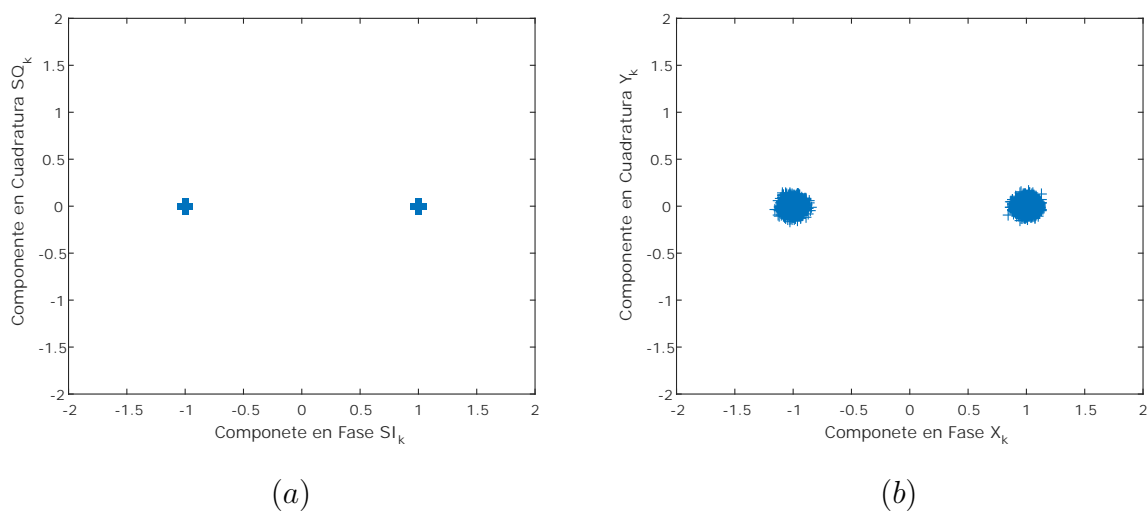


Figura 3.12: (a) Diagrama de constelación de BPSK, (b) Diagrama de constelación de BPSK con AWGN.

En la figura 3.13 se muestra el desempeño de los algoritmos de estimación de la relación señal a ruido con respecto al Cramer Rao Bound (CRB), para la SNR de referencia que varía de 2 dB a 24 dB, y cada valor de \widehat{SNR}_i se estima $N_t = 1000$ veces.

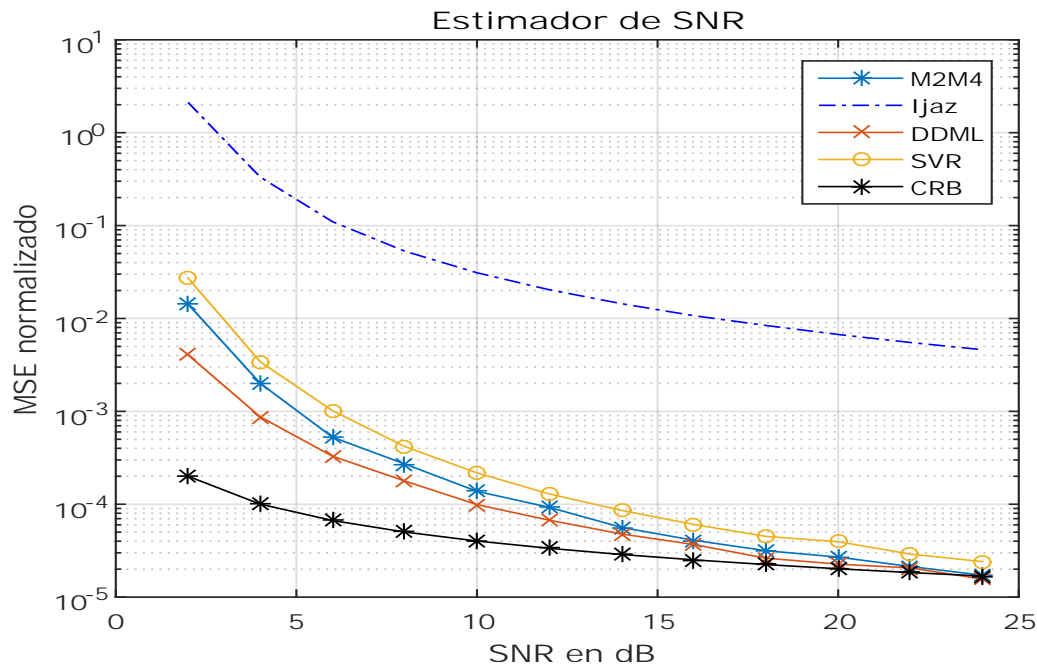


Figura 3.13: Rendimiento de los algoritmos de estimación de la SNR para la modulación digital BPSK.

Modulación QPSK.

Este esquema de modulación se conoce también como Quaternary PSK (PSK Cuaternaria), Quadriphase PSK (PSK Cuadrafásica). Esta modulación digital es representada en el diagrama de constelación por cuatro puntos equidistantes del origen de coordenadas. Con cuatro fases, QPSK puede codificar dos bits por cada símbolo.

- Desfase de 45° : Codifica la secuencia digital "11".
- Desfase de 135° : Codifica la secuencia digital "01".
- Desfase de 225° : Codifica la secuencia digital "00".
- Desfase de 315° : Codifica la secuencia digital "10".

A continuación mostramos el diagrama de constelación de la modulación digital QPSK.

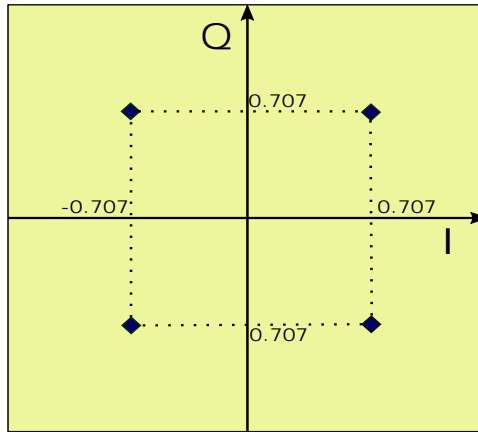


Figura 3.14: Diagrama de constelación de QPSK

Resultado del desempeño de los algoritmos de estimación para QPSK.

Se genera bit aleatorios de tamaño $N = 10000$ a las cuales se aplica la modulación digital QPSK donde se obtiene un tamaño de muestra de $L = 5000$ con la que se evalúa los algoritmos de estimación, estos son transmitidos a través del canal AWGN, a continuación se muestra el diagrama de constelación en la figura 3.15 para QPSK.

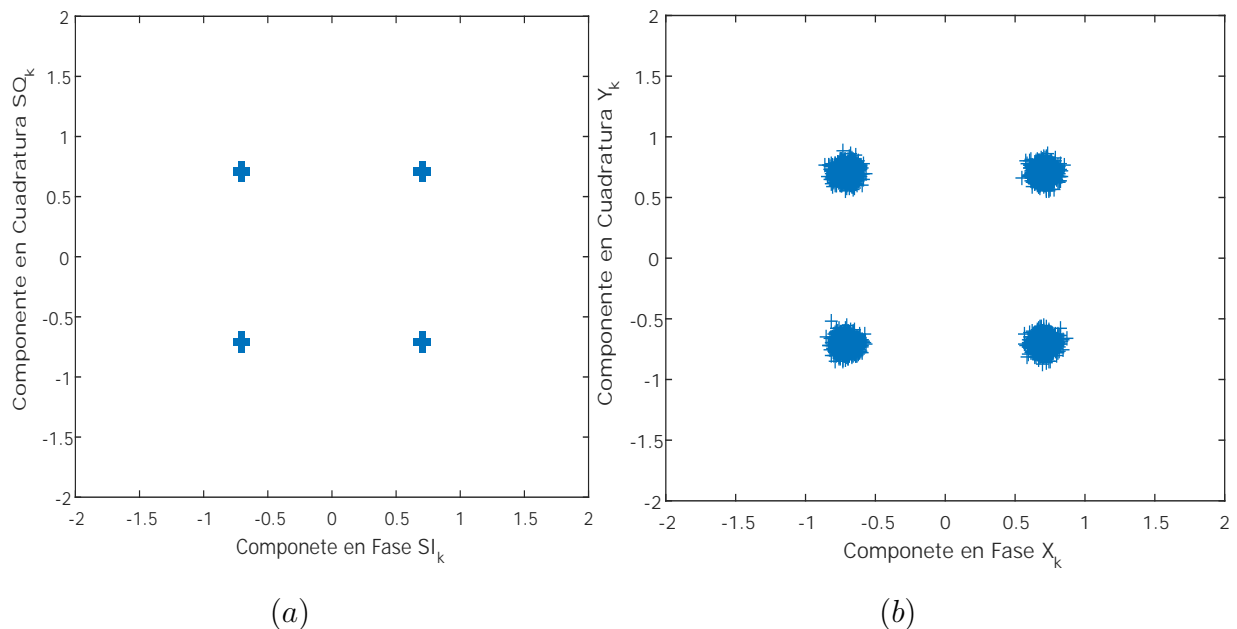


Figura 3.15: (a) Diagrama de constelación de QPSK, (b) Diagrama de constelación de QPSK con AWGN.

De forma similar a la modulación QPSK se evalúa el rendimiento de los algoritmos de estimación de la relación señal a ruido.

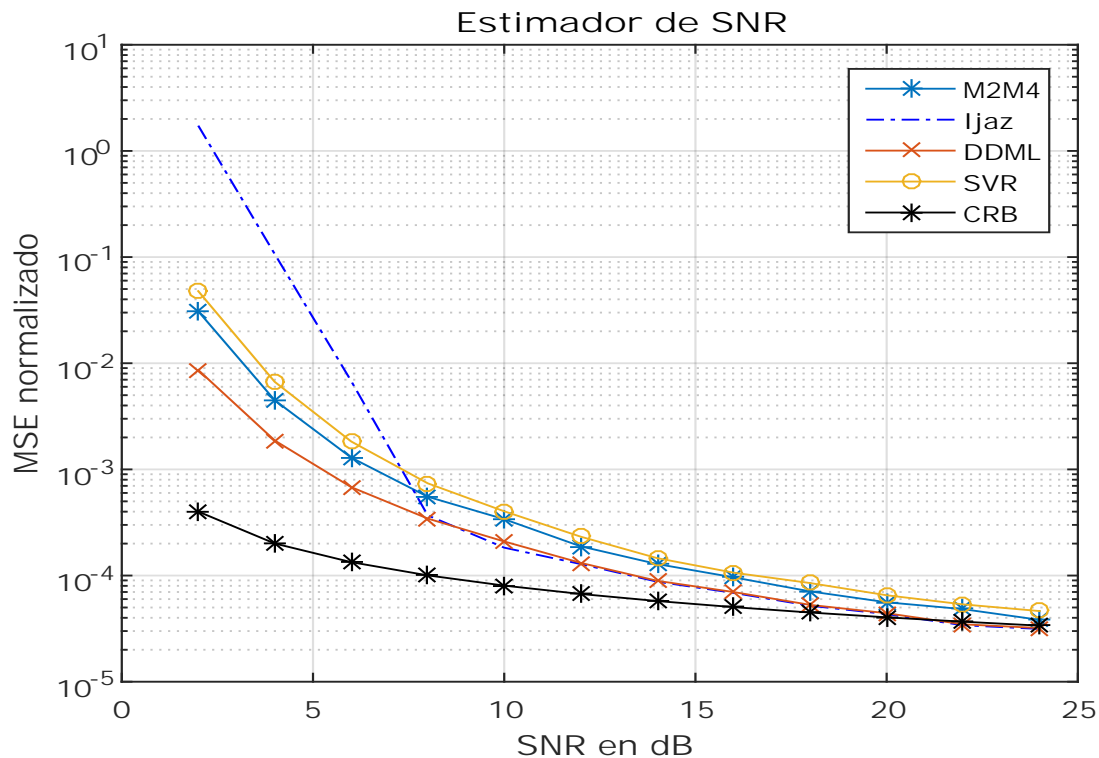


Figura 3.16: Rendimiento de algoritmo de estimación la SNR de canal AWGN.

La Fig. 3.13 y la Fig. 3.16 muestran las curvas de los algoritmos de estimación de la SNR, la curva del algoritmo más próximo al CRB ofrece mejor desempeño con respecto a otros, de la cual se obtiene lo siguiente:

- El algoritmo de máxima probabilidad DDML ofrece mejor desempeño en ambas modulaciones, sin embargo no es posible implementar en plataforma de SDR, debido a que depende de los datos originales o transmitidos, para ser evaluada.
- El algoritmo de Ijaz ofrece mejor desempeño en la modulación QPSK, pero para la modulación BPSK no es de la misma forma, por ende se descarta este algoritmo para la implementación.
- El algoritmo M_2M_4 ofrece mejor desempeño en ambas modulaciones, este algoritmo será implementado en el entorno de GRC, sin embargo el algoritmo SVR también se selecciona, porque para la modulación QPSK su rendimiento es cercano al algoritmo M_2M_4 , esto con el fin de verificar y comparar el rendimiento en el entorno de GRC.

A continuación, en la siguiente sección se implementan los algoritmos estimación de la SNR, seleccionados.

3.3. Implementación de los algoritmos de estimación de la SNR en el entorno de GRC.

En la sección anterior se concluyó, que los algoritmos SVR y M_2M_4 ofrecen mejor desempeño con respecto a los demás estimadores. El código de los algoritmos de estimación de la SNR se embeben en el bloque del entorno GRC llamado “Embedded Python Block”, utilizando el lenguaje de programación Python, como se muestra el Fig. 3.17.

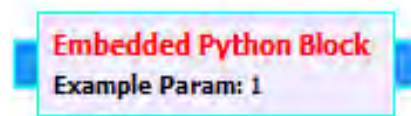


Figura 3.17: Python Block GRC.

Los códigos de estimadores de la SNR, se muestra en el anexo B en las secciones 2.1 y 2.2.

3.3.1. Diagrama de flujo en el entorno de GRC para las pruebas de los algoritmos de estimación de la SNR.

Después de embeber los algoritmos de estimación de la SNR, en Python Block, se debe complementar con algunos bloques, que ayuden a ejecutar el bloque creado, así como un generador de símbolos modulados en QPSK y un canal AWGN, para este análisis, se programa con los siguientes bloques de procesamiento del entorno de GRC, como se muestran a continuación en la Fig.3.18.

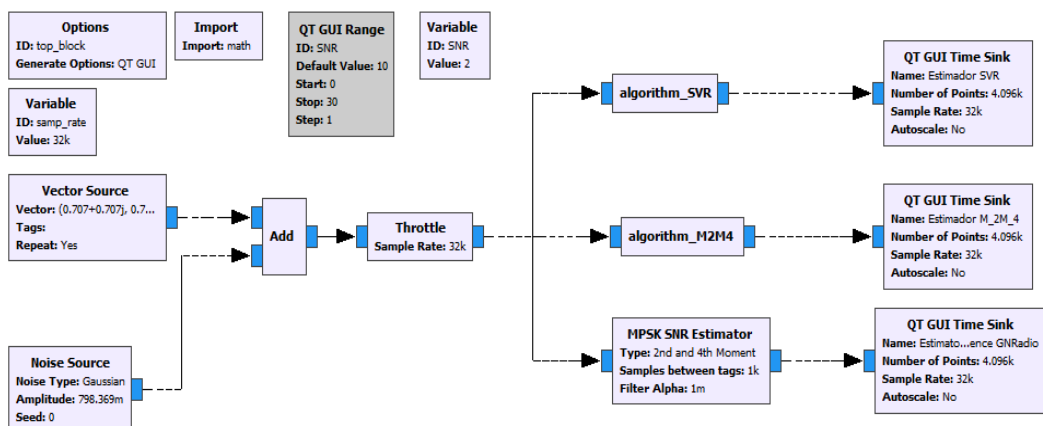


Figura 3.18: Entorno gráfico de GRC, para pruebas de estimación de la SNR.

La Fig. 3.18 muestra el diagrama bloques de GRC, que evalúa los algoritmos de estimación de la SNR, a continuación se describen los bloques de procesamiento:

- El bloque `Vector Source` es una fuente que transmite elementos según el vector de entrada, en este bloque se da un vector de componentes en fase y cuadratura de la modulación QPSK.
- El bloque `Noise Source` produce una señal de ruido que utiliza la distribución Gaussiana, este bloque se añade a la información generada.
- El bloque `Throttle` disminuye el flujo de muestras de modo que la tasa promedio no exceda la tasa específica (en muestras por segundo) del CPU.
- El bloque `Algorithm_SVR` representa la implementación del algoritmo *SVR* que estima la SNR del canal AWGN.
- El bloque `Algorithm_M2M4` representa la implementación del algoritmo M_2M_4 que también estima la SNR del canal AWGN.
- El bloque `MPSK SNR Estimator` es un estimador de la SNR de PSK M-aria del entorno gráfico de GRC, se utiliza para comparar con los algoritmos que se implementaron.
- El bloque `QT GUI Time Sink` es un sumidero gráfico para mostrar múltiples señales en el tiempo, donde el valor de la SNR estimada se muestra etiquetada en la señal de salida.

3.3.2. Resultados de los algoritmos de estimación de la SNR en el entorno de GRC.

Al ejecutar el programa, el bloque `QT GUI Time Sink` muestra la información recibida en el dominio del tiempo, estas señales llevan etiquetadas el valor estimado de la SNR del canal AWGN, además se utilizó el bloque `MPSK SNR Estimator`, para comparar con los algoritmos implementados en el entorno de GRC.

A continuación se muestran los resultados de la estimación de la SNR para los niveles de referencia; SNR = 2dB y SNR = 18 dB.

De las figuras 3.19, 3.20, 3.21, 3.22, 3.23 y 3.24 muestran los resultados de la estimación de la SNR del canal AWGN en el entorno GRC , además se compara con el bloque estimador `MPSK SNR Estimator`, para verificar el margen de error que se tiene al momento de estimar la SNR del canal, de las pruebas realizadas se deduce lo siguiente.

- Las estimaciones para el nivel de la SNR = 2 dB, los algoritmos implementados y el estimador de referencia del entorno GRC no son tan próximos al valor de referencia, cabe indicar que los tres estimadores son parecidas en el margen de error, además se deduce que para niveles bajos de la SNR, las estimación de la SNR carecen de precisión.
- Las estimaciones para el nivel de la SNR = 18 dB, los algoritmos implementados y el estimador de referencia del entorno GRC son muy próximos, quiere decir que cuanto mayor sea el nivel de la SNR la estimación de la SNR del canal es más precisa.
- El algoritmo de estimación *SVR* para niveles de la SNR menores a 2 dB, en la estimación tiene problemas con el entorno GRC, debido a que este estimador en su ecuación matemática opera raíz cuadrada, es decir que para raíz cuadrada de un numero negativo, el sistema no ejecuta esta operación.

De la implementación, se determina que el algoritmo de estimación de M_2M_4 ofrece mejor desempeño tanto en la simulación en el software de Matlab y el entorno gráfico de GNU radio companion GRC, por ende este algoritmo es seleccionado para ser implementado en la plataforma de la SDR, que se explica en el siguiente capítulo.

Capítulo IV

DISEÑO E IMPLEMENTACIÓN DE LOS SISTEMAS EN PLATAFORMA DE LA SDR

En este capítulo se diseña e implementa en la plataforma de SDR los sistemas, así como; un transmisor y receptor para las modulaciones BPSK y QPSK, el algoritmo de estimación de la relación señal a ruido SNR en la etapa del receptor del sistema, el cálculo de la tasa de error de bit, y finalmente el sistema de control adaptativo de la potencia transmitida.

4.1. Implementación del transmisor y receptor en el sistema de comunicación.

Se diseña el transmisor y receptor para las modulaciones BPSK y QPSK del sistema de comunicación, donde se utiliza del entorno de GNU radio companion (GRC) que realiza el procesamiento en banda base de los datos, además se emplea el Universal software radio peripheral (USRP) NI 2920, que convierte las señales digitales a analógicas pasando a la frecuencia intermedia (IF) y posteriormente a la etapa de radiofrecuencia (RF) para emitir ondas de radio en el caso del transmisor, y en el caso del receptor se encarga de decodificar la ondas de radio, se filtra la señal en RF, de este punto se pasa a la frecuencia intermedia (IF), seguidamente se

convierte la señal analógica a digital para entregar a la banda base para el procesamiento de datos.

4.1.1. Diseño del sistema de comunicación digital.

A continuación se muestra el diagrama de bloques del diseño del sistema de comunicación del transmisor y receptor para modulaciones BPSK y QPSK, el cual resume el procesamiento en banda base y la respectiva transferencia de datos por radiofrecuencia (RF).

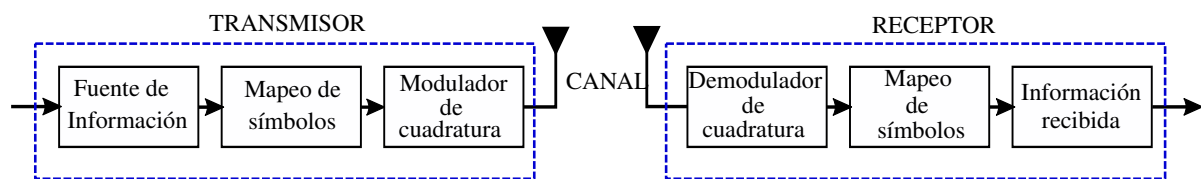


Figura 4.1: Diagrama de bloque del transmisor y receptor de un enlace de transmisión inalámbrica

La Fig. 4.1, inicialmente genera datos de bit aleatorios en el transmisor, estos datos son mapeados de acuerdo al tipo de modulación ya sea BPSK o QPSK, seguidamente son modulados en cuadratura, todo este proceso se lleva a cabo en la banda base, estas señales se convierten a frecuencia intermedia IF y posteriormente se convierten las señales digitales a analógicas las cuales son amplificadas para ser transmitidas por RF.

En la etapa de recepción la señales de RF son adecuadas mediante el filtrado y se convierten a la frecuencia intermedia IF, estas son digitalizadas para ser convertidas en banda base, donde se demodula en cuadratura y mapea dependiendo al tipo de modulación empleada y finalmente se obtiene la información deseada.

4.1.2. Implementación del Transmisor QPSK en el entorno de GRC.

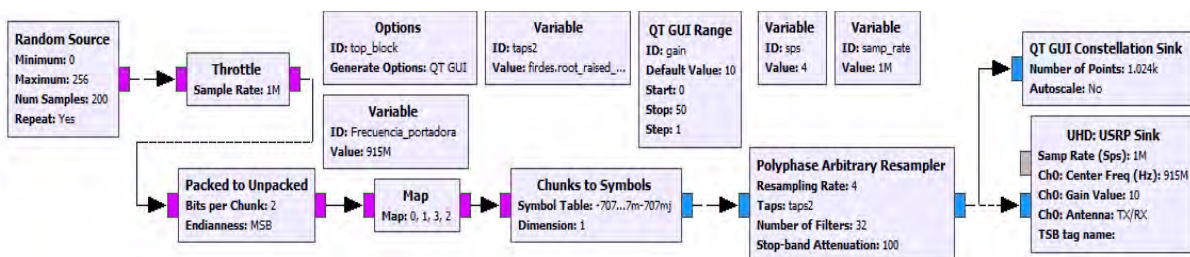


Figura 4.2: Diagrama de bloque de un transmisor QPSK en GRC.

La Fig. 4.2 muestra el diagrama de bloque del transmisor digital en el entorno de GNU radio companion (GRC), que tiene la función de modular en QPSK los datos aleatorios de bit y las procesa utilizando un muestreador arbitrario de banco de filtros polifásicos que disminuye la interferencia entre símbolos (ISI). A continuación se describen los bloques de procesamiento de señal que se ha utilizado para implementar el transmisor digital.

- Bloque `Random Source`. Genera una cantidad de muestras de números aleatorios de $[min, max >$ quiere decir que el valor máximo no se incluye.
- Bloque `Throttle`. Este bloque controla el límite de procesamiento de la máquina durante el procesamiento al ejecutar el programa y por lo general esta se configura a la misma frecuencia de muestreo que el resto de los bloques de simulación.
- Bloque `Packet to Unpacket`. Convierte una secuencia de bytes o cortos empaquetados en una secuencia de bytes o cortos no empaquetados.
- Bloque `Map`. Este bloque asigna una señal entrante al valor en el mapa, para modulación QPSK el mapeo es de la siguiente forma $Map = [0 \ 1 \ 3 \ 2]$.
- Bloque `Chunks to Symbols`. Asigna un flujo de índices de símbolos desempaqueados al flujo de puntos de constelación por ejemplo para la modulación QPSK se asigna $[-0.707-0.707j, -0.707+0.707j, 0.707+0.707j, 0.707-0.707j]$
- Bloque `Polyphase Arbitrary Resampler`. Muestreador arbitrario de banco de filtros polifásicos que evita la interferencia entre símbolos. El parámetro `taps2` se utiliza para describir al filtro conformador de pulso que se va a utilizar, en este caso se trata de un filtro de coseno alzado, a continuación se describe este filtro `firdes.root_raised_cosine(nfilt, nfilt, gain, alpha, ntaps)` donde:
 - `nfilt=32` Número de filtros.
 - `gain= 1.0` Ganancia del filtro.
 - `alpha= 0.35` Factor de exceso de ancho de banda.

- `ntaps = [11*sps*nfiltls]` Numero de tomas, y `sps` muestras por símbolo.
- Bloque UHD: `USRP Sink`. Bloque utilizado para transmitir muestras a un dispositivo USRP (es decir, actuar como transmisor). Para ajustar parámetros como la frecuencia central.

4.1.3. Implementación del Receptor QPSK en el entorno de GRC.

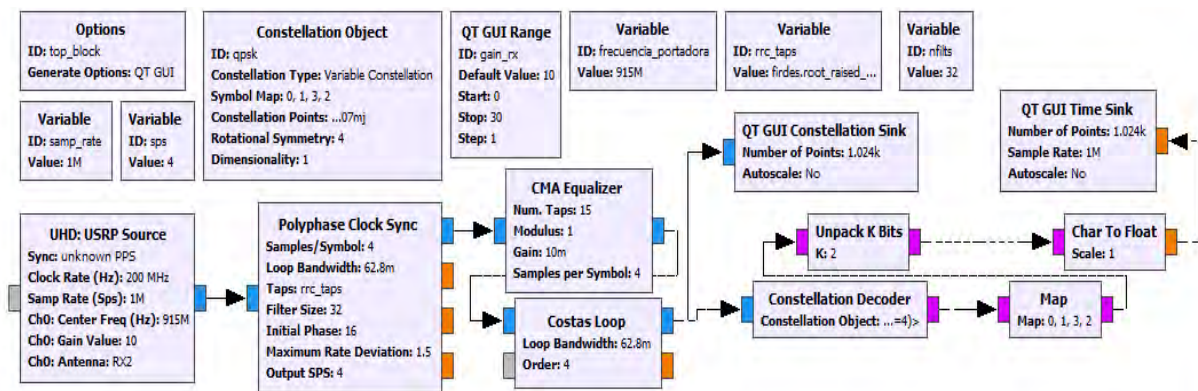


Figura 4.3: Diagrama de bloque del receptor QPSK en GRC.

La Fig. 4.3, muestra el diagrama de bloque del receptor digital en el entorno GNU radio companion (GRC), que tiene la función de demodular en QPSK y recuperar los datos suministrados por el transmisor, a continuación se describe los bloques de procesamiento de señal utilizados.

- Bloque `Polyphase Clock Sync`, este bloque realiza una sincronización de tiempo usando un banco de filtro polifásico. trabaja mediante la creación de dos bancos de filtro; uno de ellos contiene filtro adaptativo de señal (como un coseno alzado) donde cada una de las ramas posee una fase diferente de filtro, este primer banco de filtros tiene la forma de *sinc*. El segundo banco de filtros contiene las derivadas de los filtros en el primer banco. La idea es alinear la señal de salida de manera que esta sea muestreada exactamente en el pico de forma *sinc* se describe:

```
digital.pbf_clock_sync_ccf(sps, loop_bw, taps, filter_size,
init_fase, max_rate_desviation, osps)
```

- `sps = 4` Número de muestras por símbolo de la señal de entrada.

- `loop_bw = 2 * pi/100` Indica el ancho de banda del lazo de control interno.
 - `taps` Filtro prototipo para llenar el banco de filtros:
`firdes.root_raised_cosine(nfilt, nfilt, gain/sps,`
`alpha, ntaps)`
 - `filter_size = 32` Número de filtros.
 - `Init_pashe = 32` Fase inicial del filtro `nfilt/2`.
 - `max_rate_desviation` La tasa de desviación máxima por defecto 1.5.
 - `osps = 4` Número de muestras por símbolo a la salida.
- Bloque `CMA Equalizer` Implementa un filtro adaptativo de módulo constante en una secuencia compleja.
 - `num_taps = nfilt/2` Número de toques en el ecualizador (tamaño del canal).
 - `modulus = 1` módulo de las señales moduladas.
 - `gain = 0.01` ganancia del ciclo de actualización
 - `Samples per Symbol = 4` número de muestras por símbolo de la señal de entrada.
 - Bloque `Costas Loop` Un módulo de recuperación de portadora de bucle Costas, ideal para sincronizar a BPSK, QPSK y 8PSK.
 - Bloque `Constellation Decoder` Decodifica los puntos de una constelación desde un espacio complejo a bits (desempaquetados) según el mapa del objeto.
 - Bloque `unpack k Bits` convierte un byte con k bits relevantes en k bytes de salida con 1 bit cada uno, ubicado en el LSB.
 - Bloque `Char to Float` Convierte la secuencia de caracteres en una secuencia de flotación y aplica un factor de escala (establecido en 1 de forma predeterminada).

En el sistema de comunicación implementado, el transmisor representa al nodo sensor y el receptor a la estación base, el algoritmo M_2M_4 se implementa en la estación base.

4.2. Implementación del algoritmo de estimación de la SNR en el sistema de comunicación.

En esta sección se implementa el algoritmo de estimación de la relación señal a ruido (SNR) de segundo y cuarto momento (M_2M_4) en la plataforma de la radio definida por software (SDR), ya que este algoritmo ofrece mejor desempeño tanto en simulación e implementación en el entorno de GNU radio companion (GRC), cabe indicar que el estimador M_2M_4 se implementa en la etapa del receptor del sistema de comunicación, después de la recuperación de los símbolos. En seguida se muestra el diagrama de bloque que explica la implementación del estimador de la SNR.

4.2.1. Diseño para un enlace de transmisión cableada.

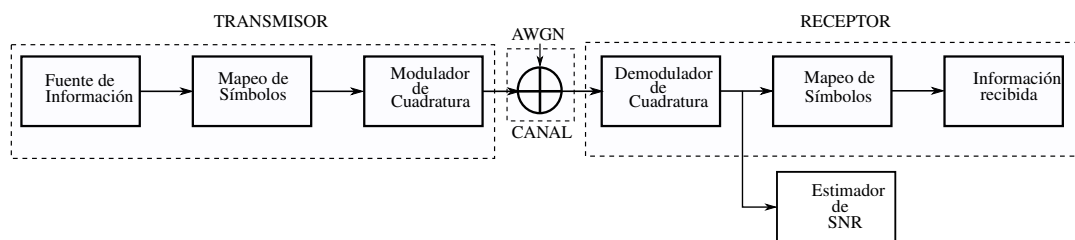


Figura 4.4: Diagrama de bloque de un enlace cableado para la transferencia de información.

La Fig. 4.4 muestra el diagrama de bloque de un enlace guiado del sistema de comunicación de la implementación del primer escenario, donde se suministra el ruido AWGN en banda base en el transmisor con la finalidad de variar el comportamiento del canal y en la etapa del receptor se implementa el algoritmo de estimación M_2M_4 después del demodulador de cuadratura, para que evalúe la relación señal a ruido del canal. Este diseño se implementa utilizando dos radios USRP NI 2920, cable coaxial de conectores de tipo SMA macho en ambos lados, y un atenuador de 30 dB para proteger el equipo USRP en el receptor.

4.2.2. Diseño de un enlace de transmisión inalámbrica.

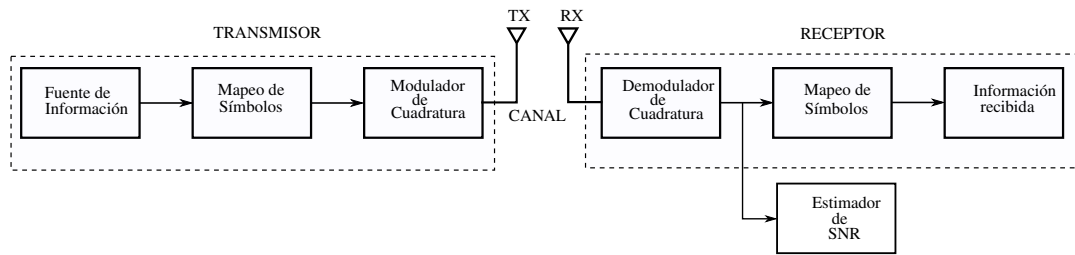


Figura 4.5: Diagrama de bloque de un enlace inalámbrico para la transferencia de información.

La Fig. 4.5, muestra el diagrama de bloque de un enlace inalámbrico de la implementación del segundo escenario, donde las condiciones del canal van a depender del medio en el que este implementado, de forma similar que el primer escenario el estimador de la SNR es implementado en la etapa del receptor. Para implementar este diseño, se utiliza dos radios USRP NI 2920, dos antenas directivas de tipo parche de alta ganancia, en este escenario no es necesario utilizar el atenuador de señal debido a que la información llega atenuada por las condiciones del canal hacia el receptor.

A continuación se muestra la implementación del algoritmo de M_2M_4 , en la etapa de receptor que más adelante representará a la estación base del sistema, en la plataforma de SDR.

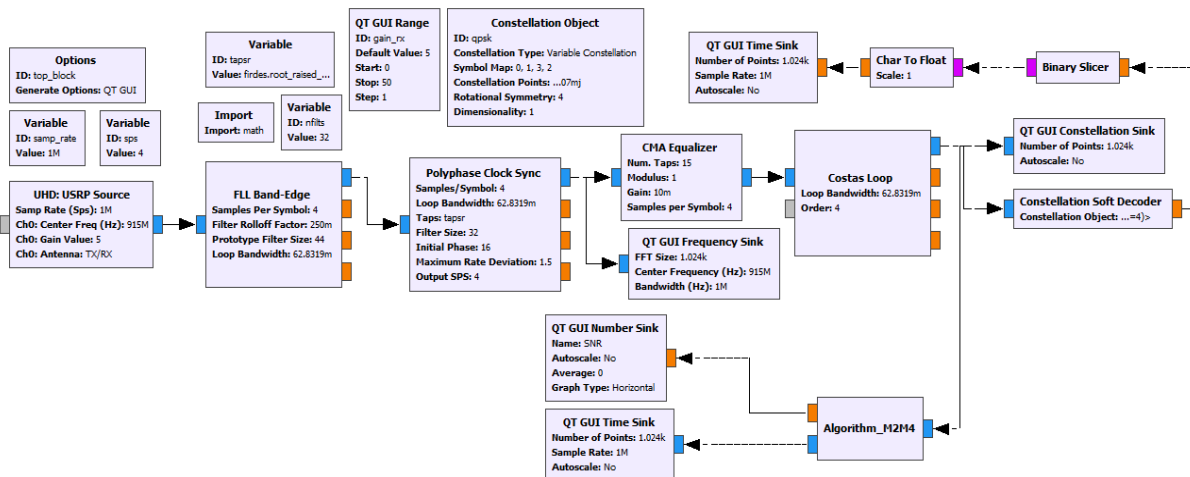


Figura 4.6: Diagrama de bloque del receptor QPSK en GRC.

La Fig. 4.6, muestra el diagrama de bloque de la estación base en GRC, donde el bloque Polyphase Clock Sync sincroniza en el tiempo, utilizando banco de filtros polifásicos y el bloque Costas Loop es un módulo de recuperación de portadora de bucle Costas, ideal

para sincronizar a BPSK, QPSK y 8PSK, después a este bloque se implementa el bloque estimador de la SNR “Algorithm_M2M4”, en la cual se embebe el algoritmo estimador M_2M_4 , además a la salida *output* se le crea dos salidas de tipo `Complejo` (color azul) y `Float` (color anaranjado) como se observa en el bloque estimador.

La salida de tipo `Complejo` muestra la señal recuperada en el dominio del tiempo donde lleva etiquetada el valor de la SNR estimada del canal, y en la salida de tipo `Float`, es creado con el fin de obtener el valor de la SNR de tipo float, este parámetro es importante para implementar el sistema de transmisión adaptativa de potencia.

Los resultados obtenidos de la implementación, en los dos escenarios de prueba, se muestran en el siguiente capítulo.

4.3. Implementación del cálculo de la tasa de error de bit en el sistema.

4.3.1. Probabilidad de error y tasa de error de bit.

Los términos probabilidad de error, P_e y la tasa o frecuencia de errores de bits BER (Bit Error Rate), se usan con frecuencia en forma indistinta, aunque en la práctica sí tienen significados un poco distintos. La P_e es la expectativa teórica, o matemática, de que determinado sistema tenga una tasa de errores. La tasa de errores de bits es un registro empírico (histórico) del funcionamiento real del sistema en cuanto a errores (Tomasi, Hernández, y Pozo, 2003).

Para el diagrama de bloque siguiente, se supone un filtro adaptado en el receptor.

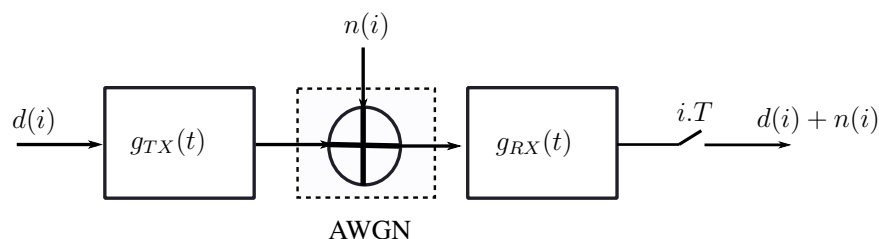


Figura 4.7: Diagrama de bloque del sistema de comunicación.

La Fig. 4.7 muestra el diagrama, que transmite datos binarios a través del canal AWGN, los datos que llegan al receptor han sido atenuadas de la forma siguiente $1+n(i)$ ó $-1+n(i)$ donde

$n(i)$ representa el ruido, a continuación se muestra la función de densidad de probabilidad para el caso de decisión binaria.

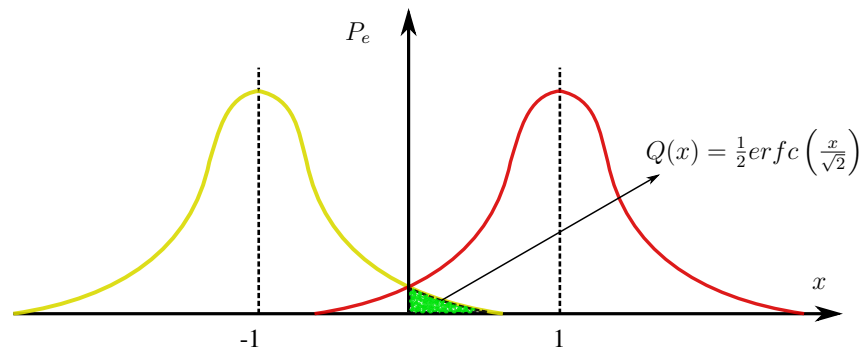


Figura 4.8: Probabilidad de error P_e .

La Fig. 4.8 muestra la intersección de las funciones de densidad de probabilidad, la sección sombreada de color verde representa la probabilidad de error al enviar un dato como -1, que se puede calcular en la siguiente ecuación.

$$P(-1/1) = P(1/1) = P(\eta > \sqrt{E_b}) = Q\left(\frac{\sqrt{E_b}}{\sigma}\right) \quad (4.1)$$

Donde $\sigma = \sqrt{\frac{N_o}{2}}$ la desviación estándar y $\eta = \frac{N_o}{2}$ la densidad espectral de la potencia del ruido (W/Hz).

A continuación se muestra la ecuación del cálculo de la probabilidad de error P_e para la modulación QPSK.

$$P_e = 2Q\left(\sqrt{\frac{E_s}{\eta}}\right) \quad (4.2)$$

Además se puede utilizar las siguientes ecuaciones para poder determinar la P_e , ya sea utilizando una calculadora o un software.

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-t^2/2} dt \quad (4.3)$$

$$\text{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt \quad (4.4)$$

De la ecuación (4.3) y (4.4) se obtiene la siguiente ecuación.

$$Q(x) = \frac{1}{2} \operatorname{erfc} \left(\frac{x}{\sqrt{2}} \right) \quad (4.5)$$

Para evaluar la probabilidad de error de bit de QPSK se tiene la siguiente ecuación donde $M = 4$, representa el orden de modulación digital de PSK.

$$P_{bit} = \frac{2Q \left(\sqrt{\frac{E_s}{\eta}} \right)}{\log_2(M)} \quad (4.6)$$

La ecuación (4.5) ayuda a calcular la probabilidad de error de bit para las modulaciones digitales BPSK y QPSK.

La probabilidad de error de bit para QPSK.

$$P_{bit} = Q \left(\sqrt{\frac{2Eb}{\eta}} \right) \implies P_{bit} = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{Eb}{\eta}} \right) \quad (4.7)$$

La ecuación (4.7) se puede evaluar de la siguiente forma (“Introduction to Detection Theory”, 2014).

$$P_{bit} = \frac{1}{2} \operatorname{erfc} \left(\sqrt{SNR} \right) \quad (4.8)$$

4.3.2. Implementación del cálculo de la tasa de error de bit.

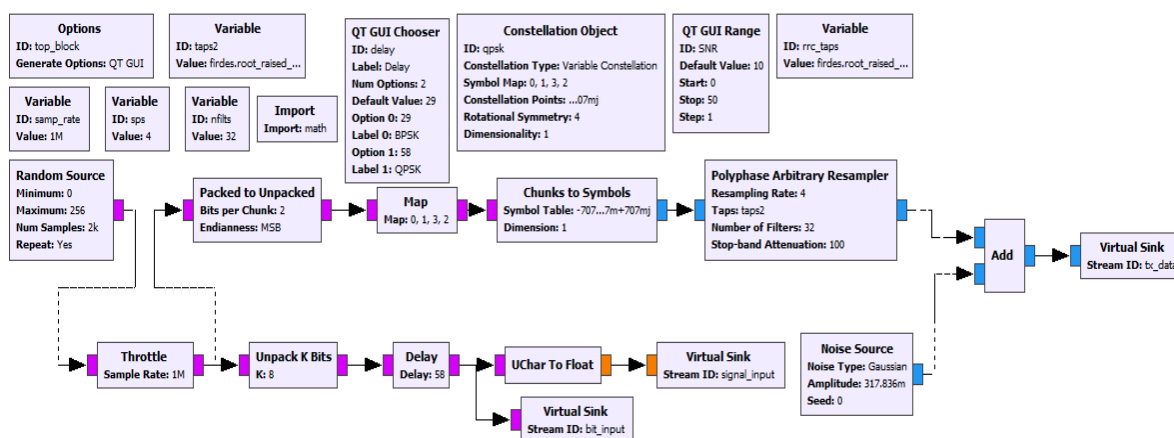


Figura 4.9: Diagrama de bloque de transmisor QPSK en GRC.

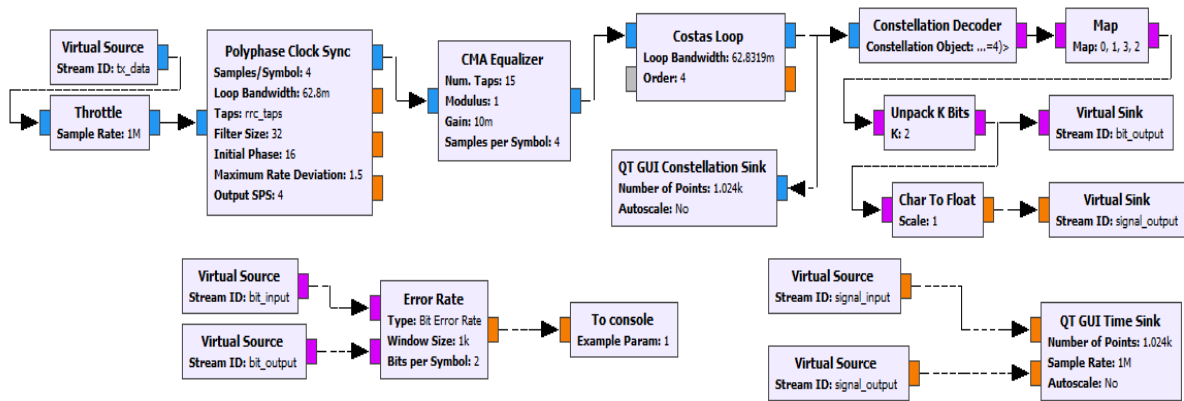


Figura 4.10: Diagrama de bloque de receptor QPSK en GRC.

En el entorno de GNU Radio companion mostrada en la Fig. 4.9 transmite bit aleatorios modulados en QPSK a través del canal AWGN generada en banda base, la Fig. 4.10 recibe los datos codificados en QPSK y recupera la información. Para evaluar la tasa de error de bit del sistema se utiliza el bloque *Error rate*, al cual se le conecta los datos de información transmitidas y los datos la información recuperadas de forma ordenada.

4.4. Diseño e implementación del sistema de control adaptativo de la potencia transmitida.

Los sistemas de comunicaciones inalámbricas se caracterizan por utilizar un medio de transmisión dinámico, en el que las condiciones de transmisión cambian de forma continua debido a las condiciones del canal. Este cambio continuo conlleva a un sobredimensionamiento de los recursos empleados, tal como la potencia de transmisión, ocasionando mayor consumo de energía.

Entonces el sistema de control adaptativo de la potencia transmitida a implementar, tiene como objetivo mantener constante la relación señal a ruido (SNR) en el receptor variando la potencia de transmisión, es decir el transmisor ajusta su potencia a las condiciones del canal, conforme a la información suministrada por el receptor, de esta manera se puede optimizar el consumo energético en la redes inalámbricas instaladas en zonas de difícil acceso.

4.4.1. Diseño del sistema de control adaptativo de potencia transmitida.

En seguida se muestra el diseño del sistema de control adaptativo de la potencia transmitida en la Fig. 4.11, que integra la etapa del transmisor que representara al nodo sensor y la etapa del receptor a la estación base. En la estación base se implementa el algoritmo de segundo y cuarto momento M_2M_4 para estimar el nivel de la SNR del canal, además el diseño emplea un modo de transmisión full dúplex, para la transferencia de datos y canal de retorno para suministrar información del canal al nodo sensor, con esta información el nodo ajusta su potencia de transmisión con la finalidad de mantener el nivel de la SNR constante el receptor.

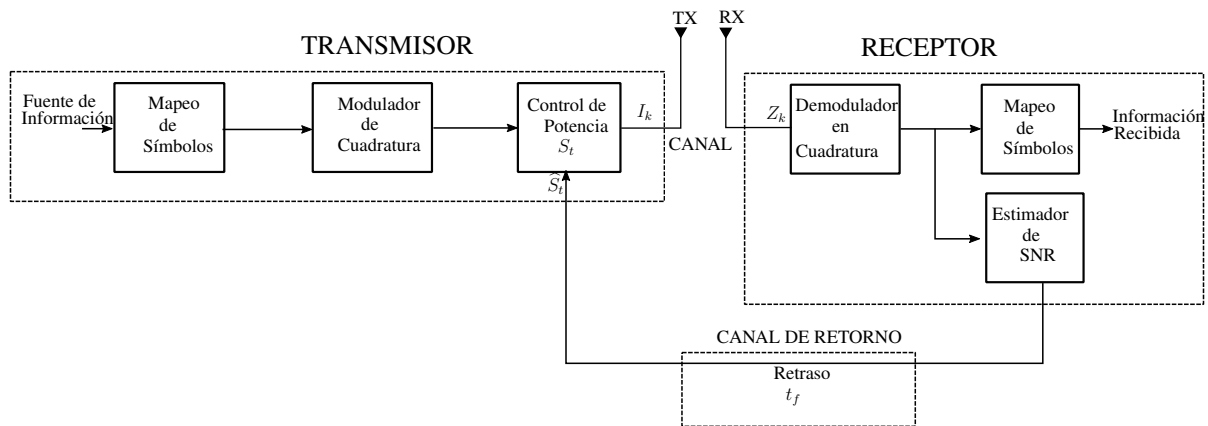


Figura 4.11: Diagrama de bloque del control adaptativo de potencia transmitida

En un sistema de potencia de transmisión constante la relación señal a ruido (SNR) instantánea en el receptor viene dado por .

$$\rho = \frac{\bar{S}g[i]}{N_o B} \quad (4.9)$$

Donde \bar{S} es la potencia promedio (W), $N_o/2$ la densidad espectral de potencia de ruido (W/Hz), B ancho de banda del receptor (Hz) y $g[i]$ es la ganancia variable estacionario y ergódico en el tiempo del canal (Goldsmith, 2005).

4.4.2. Transmisión full dúplex

La mayoría de los sistemas y redes de comunicación modernos funcionan en modo dúplex permitiendo canales de envío y recepción simultáneos. Podemos conseguir esta simultaneidad de varias formas:

- Empleo de frecuencias separadas (multiplexación en el dominio de la frecuencia).
- Cables separados.

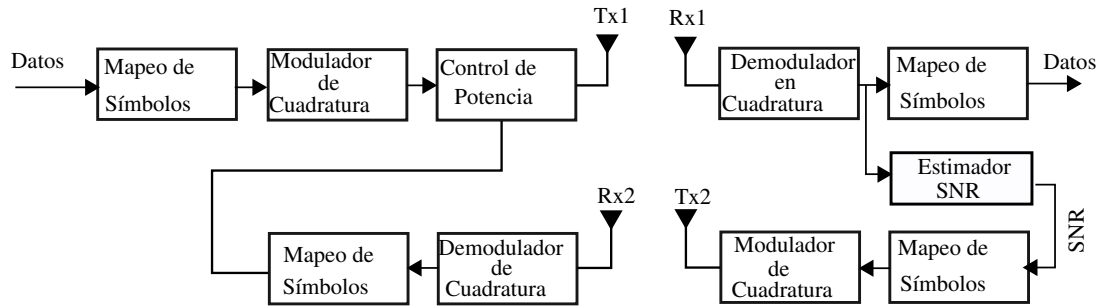


Figura 4.12: Diagrama de bloque del modo de transmisión full dúplex

La Fig. 4.12 representa el diagrama de bloque de la comunicación full dúplex, este diagrama es utilizado para implementar el sistema de control adaptativo de la potencia transmitida en la plataforma de la radio definida por software SDR.

4.4.3. Requisitos S/N de la PSK M-aria para tasa de errores fijas.

La finalidad del sistema de control adaptativo de la potencia transmitida, es ajustar la potencia de transmisión del nodo sensor con el objetivo de mantener la probabilidad de error constante en el receptor, para el cual se basa a la siguiente tabla.

$\log_2 M$	M	$\frac{E_s}{(\eta \log_2 M)}$	
		$P_e = 10^{-4}$	$P_e = 10^{-5}$
1	2	6.92 (8.40 dB)	9.10 (9.59 dB)
2	4	7.57 (8.79 dB)	9.75 (9.89 dB)
3	8	17.2 (12.4 dB)	22.2 (13.5 dB)
4	16	49.7 (17.0 dB)	64.1 (18.1 dB)
5	32	158.0 (22.0 dB)	203.0 (23.1 dB)

Tabla 4.1: Requisitos de S/N en PSK M-aria para tasas de errores fijas.

En este trabajo se emplea las modulaciones BPSK y QPSK, de acuerdo a la tabla 4.1, se tiene la tasa de error $P_e = 9,10 \times 10^{-5}$ para una $SNR = 9,59 \text{ dB}$ en la modulación BPSK y $P_e = 9,75 \times 10^{-5}$ para una $SNR = 9,89 \text{ dB}$ en QPSK, al implementar ambas modulaciones se desea mantener el nivel de la SNR aproximada en $SNR = 10 \text{ dB}$ que nos da la probabilidad de error de 3.8721×10^{-6} para BPSK y 7.744×10^{-6} en QPSK.

4.4.4. Implementación del primer escenario.

En el primer escenario, se implementa el sistema de control adaptativo de la potencia transmitida en un enlace guiado, donde el ruido AWGN se suministra de forma digital en banda base, en la siguiente figura se muestra el esquema del escenario implementado en la plataforma de la radio definida por software SDR.

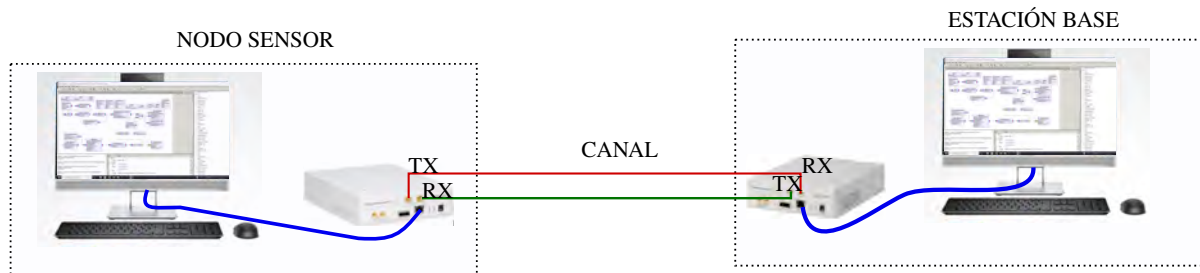


Figura 4.13: Primer escenario del sistema de control adaptativo de potencia transmitida.

De la Fig. 4.13, se describe los dispositivos empleados donde:

- Se utiliza dos ordenares PC y dos radios USRP NI 2920, para el procesamiento de la información en banda base y para la transferencia de datos.
- La conexión de color azul, es un cable ethernet con conectividad de 1 Gigabit, que conecta el ordenador PC y el USRP NI 2920 para la transferencia del procesamiento de banda base a la radio.
- La conexión de color rojo entre las radios, es un cable coaxial con conectores de tipo SMA macho en ambos lados, que enlaza al nodo sensor y la estación base, con la que suministra datos modulados en QPSK.
- La conexión de color verde, también es un cable coaxial de conectores de tipo SMA macho en ambos lados, que conecta la estación base con el nodo sensor, es el canal de retorno por donde se suministra información del canal al nodo sensor, dependiendo del estado del canal el nodo ajustar su potencia transmisión de acuerdo al diagrama de flujo mostrada en la Fig. 4.14, con la finalidad de mantener la tasa de error de bit fija en el receptor.

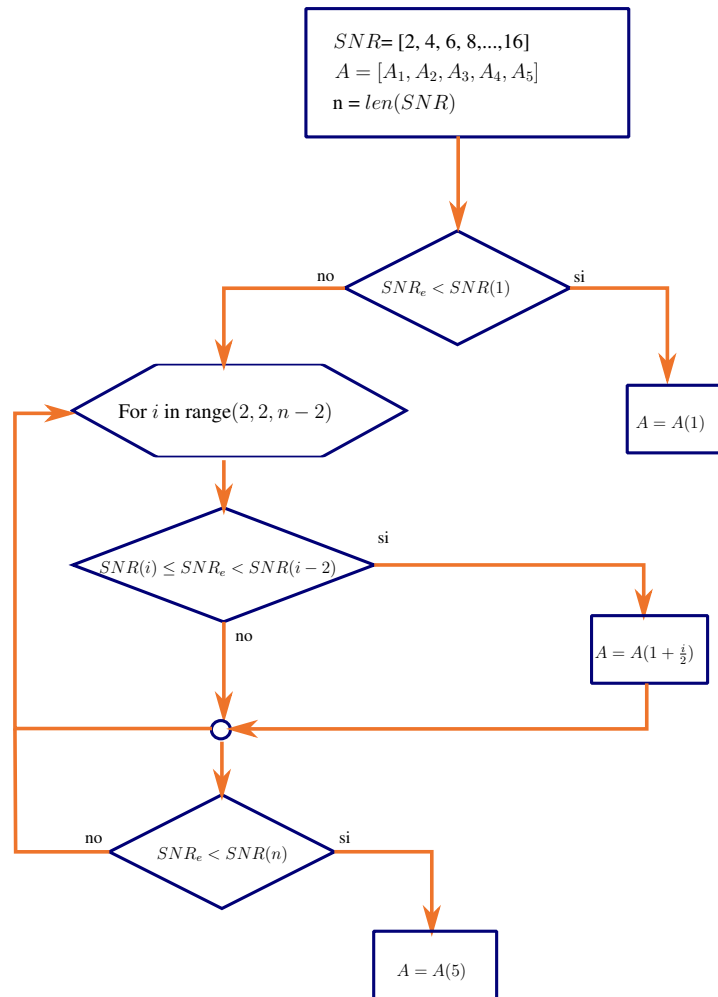


Figura 4.14: Diagrama de flujo para la asignación de amplitud al nodo sensor.

La Fig. 4.14, muestra el diagrama de flujo de la asignación de amplitud, que para un determinado rango de la SNR estimada corresponde una amplitud, este valor deberá multiplicarse con la amplitud constante de la información que transmite el nodo sensor, de esta forma modifica la potencia de transmisión del nodo, con el objetivo de mantener el nivel de la SNR deseado en el receptor.

De acuerdo a las pruebas y experiencia realizadas, al implementar el sistema de control adaptativo de potencia transmitida en un enlace cableado, se debe modificar la amplitud de la señal portadora, para poder aumentar o disminuir la potencia de la señal que se desea transmitir, de esta forma se logra ajustar el nivel de la SNR del canal a las condiciones más favorables.

A continuación se muestra el procesamiento en banda base, en el entorno GRC del nodo sensor y la estación base.

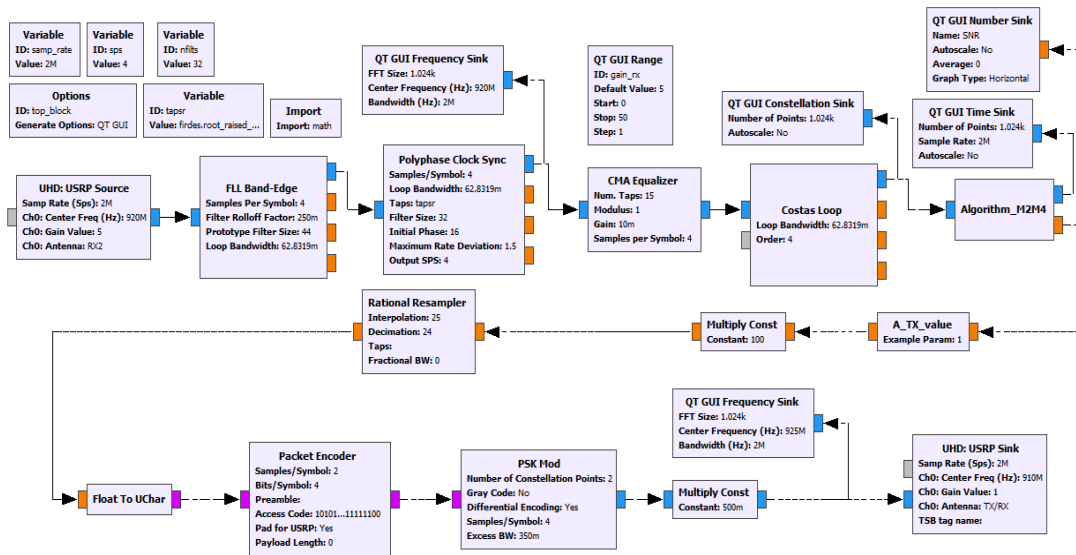


Figura 4.16: Diagrama de bloque en el de GRC de la estación base.

La Fig. 4.16, muestra el diagrama de la estación base en el entorno gráfico de GNU Radio Companion (GRC), el cual opera como receptor de información de bit aleatorios demoduladas en QPSK, después de sincronizar las componentes en fase y cuadratura de la información, el bloque `Algorithm_M2M4` estima el nivel de la SNR del canal y lo transfiere al bloque `A_TX_value` donde se determina la amplitud con la que debe multiplicar la amplitud portadora del transmisor del nodo sensor para ajustar su potencia a las condiciones del canal, esta amplitud es suministra al nodo utilizando la modulación DBPSK . Los resultados obtenidos, se muestran en el siguiente capítulo.

4.4.5. Implementación del segundo escenario.

En el segundo escenario, se implementa el sistema de control adaptativo de potencia transmitida para un enlace inalámbrico en la plataforma de SDR, donde las condiciones del canal depende de las circunstancias temporales y el ruido AWGN generado de forma externa con el USRP NI 2920, como se muestra en la siguiente imagen.

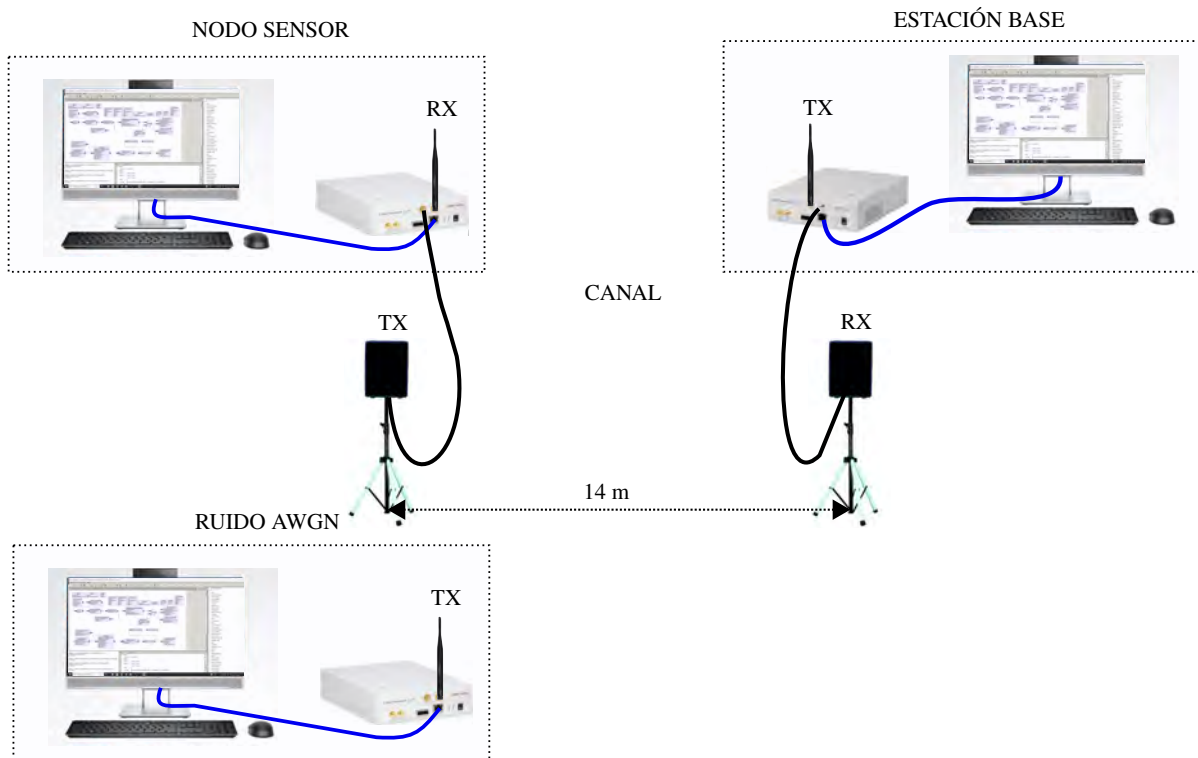


Figura 4.17: Segundo escenario del sistema de control adaptativo de potencia transmitida.

La Fig. 4.17 muestra el esquema implementado del segundo escenario, donde:

- Se utiliza tres ordenadores PC y tres radios USRP NI 2920 para las pruebas del sistema de control adaptativo de la potencia transmitida.
- Para la transferencia de bit aleatorios modulados en QPSK, del nodo sensor hacia la estación base, se utiliza dos antenas parche directivas de alta ganancia, para un enlace inalámbrico en la banda ISM de 920 MHz.
- Para adaptar la potencia transmitida del nodo sensor se utiliza dos antenas VERT 2450 omnidireccionales, para el canal de retorno en la banda ISM de 910 MHz.
- En el canal de transmisión se suministra ruido AWGN externo en la banda de 920 MHz, para el cual se utiliza el USRP NI 2920 para generar dicho ruido en el espectro.

Para generar el ruido externo, se utiliza la modulación OFDM en la plataforma de la SDR, ya que el espectro de esta modulación se asemeja al ruido AWGN, considerando que las frecuencias portadoras siempre están en transmisión, en la siguiente Fig. 4.18 se muestra el diagrama en el entorno de GRC de la modulación OFDM.

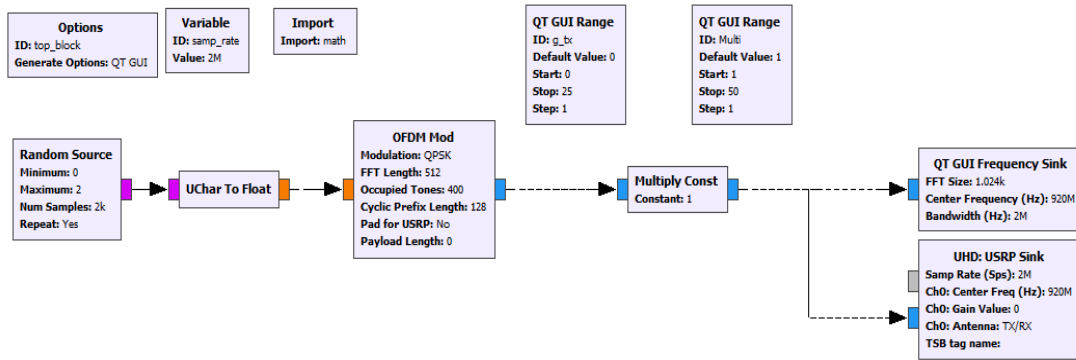


Figura 4.18: Modulación de OFDM, para generar ruido AWGN.

Con respecto a la adaptación de la potencia transmitida, en este escenario se utiliza el PID mostrado en la siguiente figura. Este controlador se emplea con el propósito de adaptar el nivel de potencia del nodo sensor a las condiciones del canal.

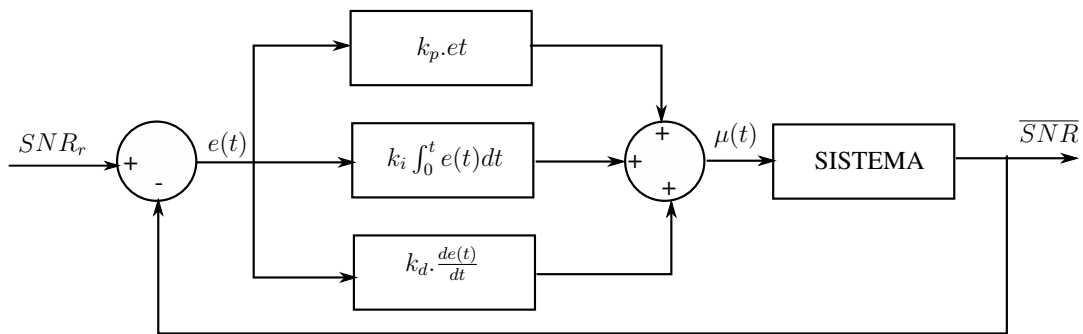


Figura 4.19: Esquema del controlador PID del sistema comunicación.

En la Fig. 4.19, se muestra el control PID implementado en el bloque `Controlador` en el entorno gráfico de GRC en la Fig. 4.21, donde el error se determina $e(t) = SNR_r - \overline{SNR}$ y a partir de esta se implementa el control PID, las constantes proporcional, integral y derivativo se determinan a prueba y error, el resultado de la sumatoria $\mu(t) = k_p \cdot e(t) + k_i \int_0^t e(t) dt + k_d \frac{de(t)}{dt}$, se le suministra al nodo sensor para que aumente o reduzca su potencia transmitida, con la finalidad de mantener constante la SNR de referencia en el receptor.

- Nivel de referencia de la SNR, $SNR_r = 10\text{dB}$.
- Nivel de la SNR estimada, \overline{SNR}

A continuación se muestra la implementación en el entorno gráfico de GNU radio companion (GRC) del nodo sensor y la estación base.

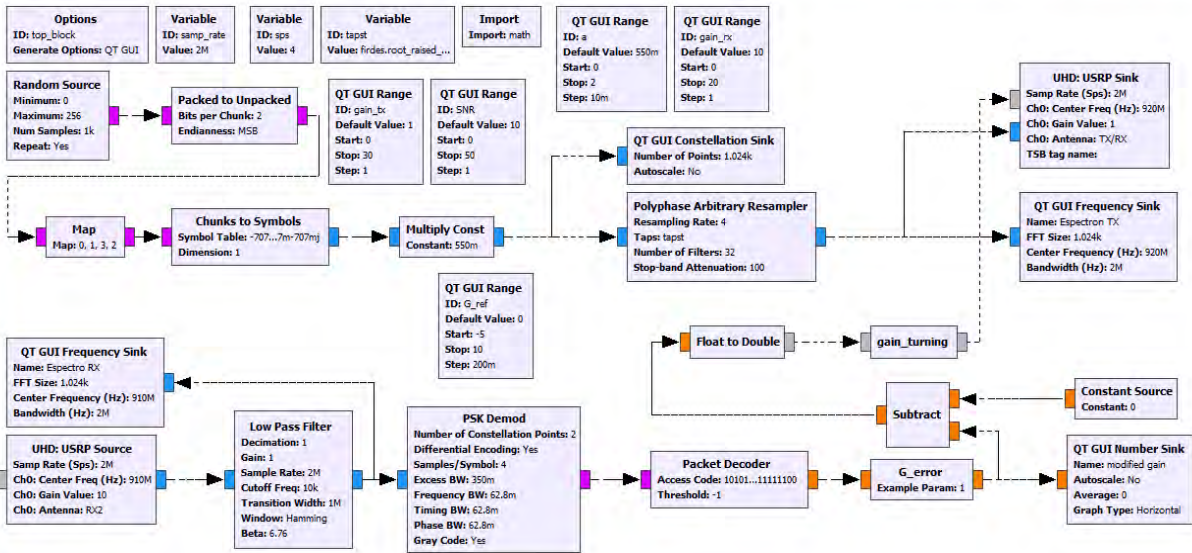


Figura 4.20: Diagrama de bloque del nodo sensor en el entorno GRC del segundo escenario.

La Fig. 4.20, muestra el entorno gráfico que representa al nodo sensor en la plataforma de SDR, donde el nodo transmite información de bit aleatorios modulados en QPSK a la estación base, y el receptor recibe la información de la potencia que es demodulada en BPSK diferencial, con el que modifica la potencia de transmisión del nodo sensor, que es suministrado por la estación base.

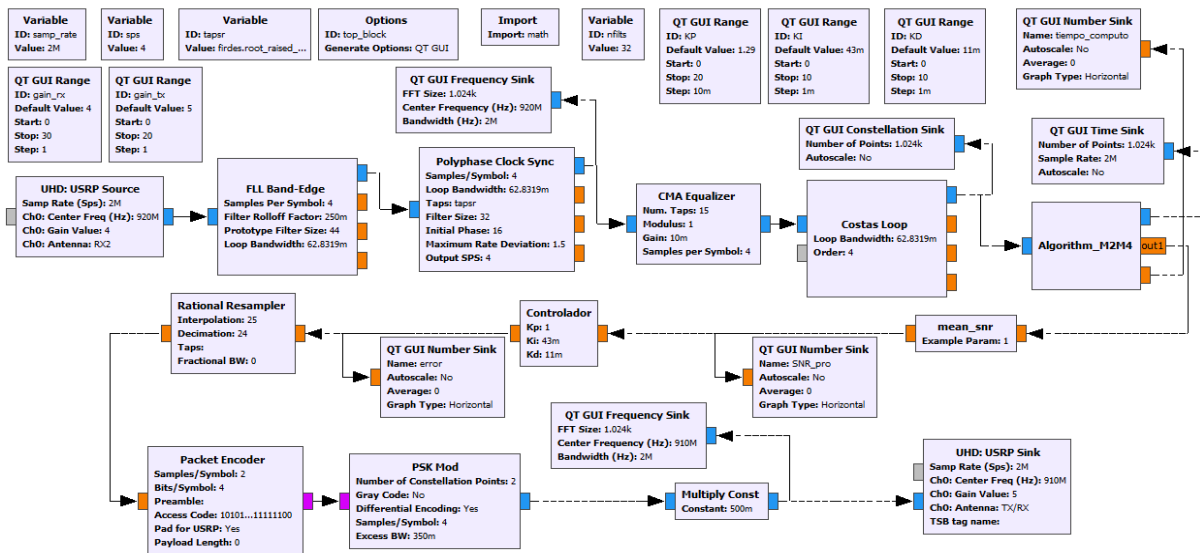


Figura 4.21: Diagrama de bloque de la estación base en el entorno GRC del segundo escenario.

La Fig. 4.21, muestra el entorno gráfico de la estación base en la plataforma de SDR, la estación base demodula en QPSK la información recibida de RF, además estima el nivel de la SNR del canal después de la sincronización y recuperación de los símbolos, y con esta informa-

ción el sistema toma la decisión en cuanto debe aumentar o disminuir la potencia transmitida del nodo sensor, para el cual se implementa control PID en el bloque `controlador`, que se explico con anterioridad.

Configuración de parámetros RF.

Para realizar las pruebas en la plataforma de la radio definida por software, se configura los parámetros de la RF del nodo sensor, la estación base y ruido AWGN generado de forma externo, además se considera algunas condiciones iniciales como la distancia entre el transmisor y receptor, nivel de referencia de la SNR del canal, como se describe en seguida.

- Distancia entre nodo sensor y estación base $d = 14$ m.
- Nivel de referencia de la $SNR = 10$ dB.
- Potencia transmitida del nodo sensor $P_{tx1} = 1$ dB.
- Ganancia del receptor del nodo sensor $P_{rx1} = 1$ dB.
- Potencia transmitida de la estación base $P_{tx2} = 10$ dB.
- Ganancia recibida de la estación base $P_{rx2} = 4$ dB.
- Frecuencia portadora para el envío de información de bit aleatorio del nodo sensor hacia la estación base $f_2 = 920$ MHz.
- Frecuencia portadora del canal de retorno $f_1 = 910$ MHz.
- Ancho de banda $B = 2$ MHz.

A continuación se muestra algunas fotografías de la implementación del segundo escenario en las figuras 4.22 y 4.23.

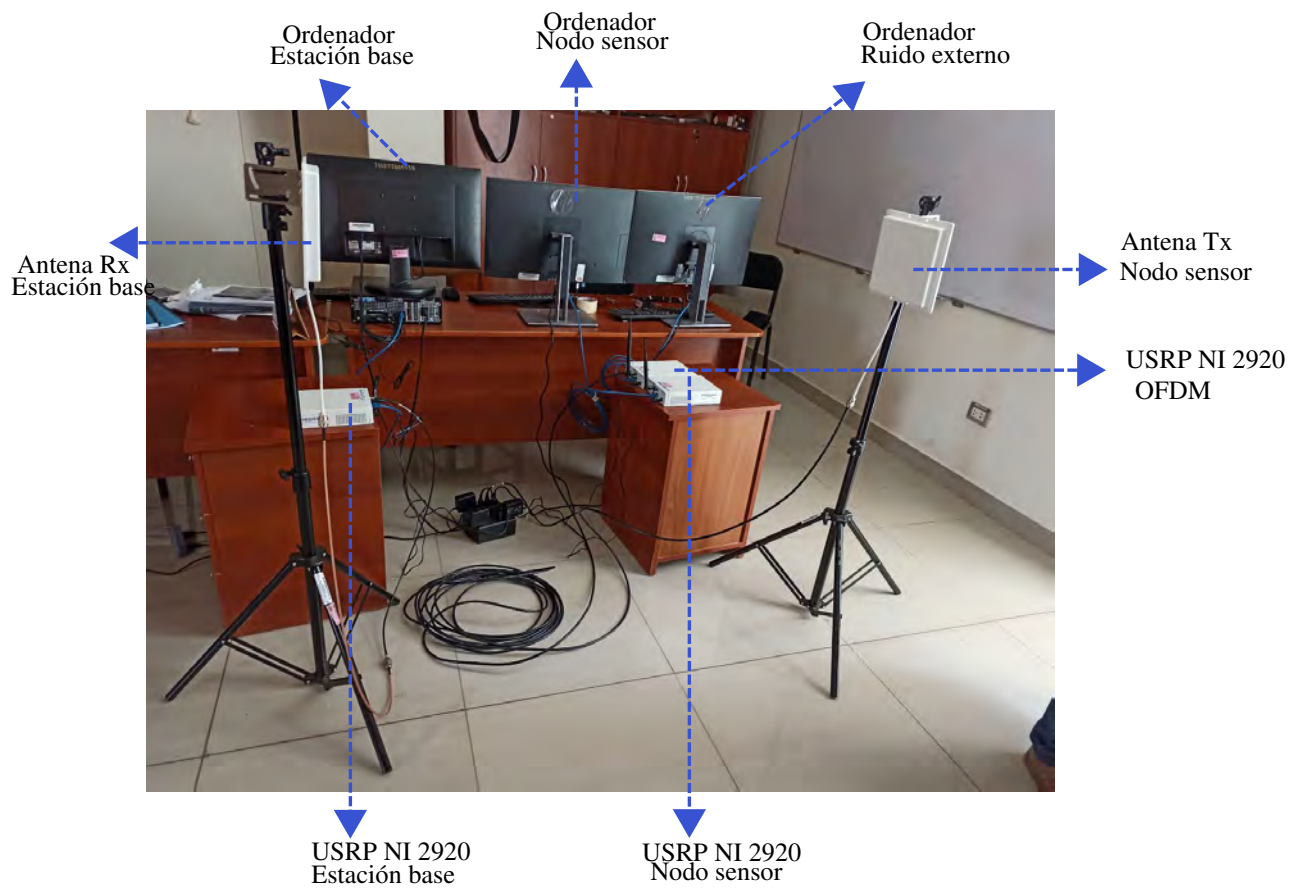


Figura 4.22: Esquema de la implementación del segundo escenario del sistema de control adaptativo de la potencia transmitida.



Figura 4.23: Lugar de pruebas, distancia entre transmisor y receptor $d = 14$ m para el envío de información de datos de bit aleatorio.

Los resultados obtenidos del sistema de control adaptativo de potencia transmitida en los dos escenarios, se muestran en el siguiente capítulo.

Capítulo V

PRUEBAS Y RESULTADOS DE LOS SISTEMAS IMPLEMENTADOS.

A continuación se muestran las pruebas y los resultados obtenidas de la ejecución de los sistemas implementados en la plataforma de la radio definida por software (SDR), mostradas en el interfaz gráfica del entorno de GNU radio Companion (GRC) y las gráficas del software de Matlab.

5.1. Transmisor y receptor del sistema de comunicación.

Las pruebas se realizan en un enlace inalámbrico punto a punto en las bandas de radio industriales, científicas y médicas (ISM), en el cual se utiliza dos sistemas SDR, donde el transmisor (nodo sensor) transmite datos de bit aleatorios modulados en QPSK, y el receptor (estación base) recibe estos datos demoduladas en QPSK, los resultados obtenidos se muestra a continuación.

5.1.1. Transmisor del sistema de comunicación.

Al ejecutar el sistema de transmisor de la Fig. 4.2 en la radio USRP NI 2920, en el entorno gráfico de GNU radio companion (GRC) muestra la visualización del espectro del transmisor centrado en la banda ISM de 915 MHz y el diagrama de constelación de los datos de bit

modulación QPSK, como se observan en las figuras 5.1 y 5.2.

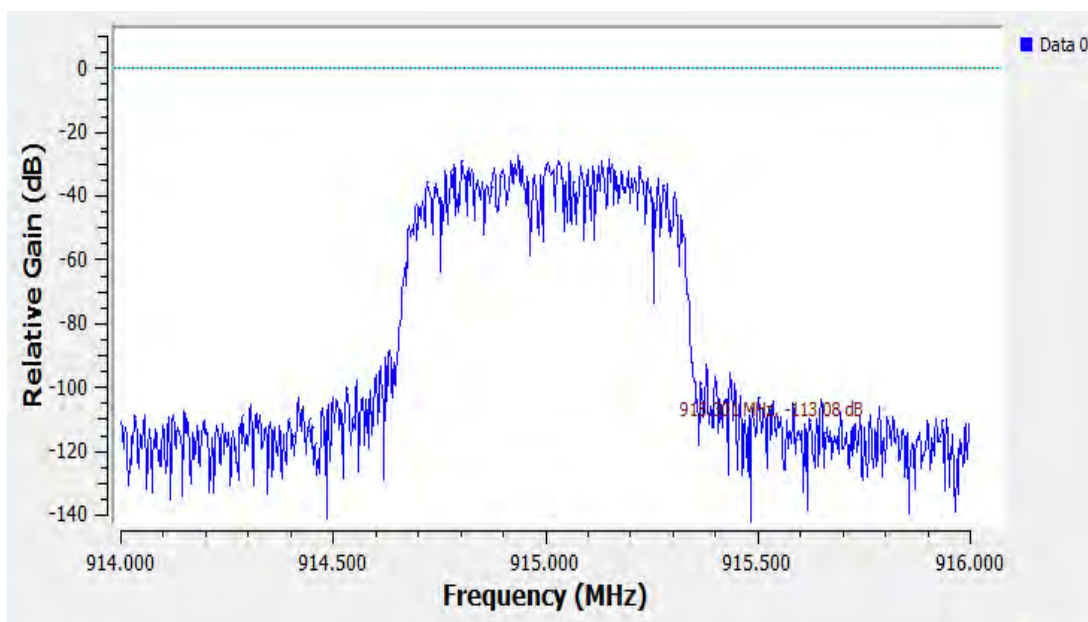


Figura 5.1: Espectro de la señal transmitida en la banda de 915 MHz.

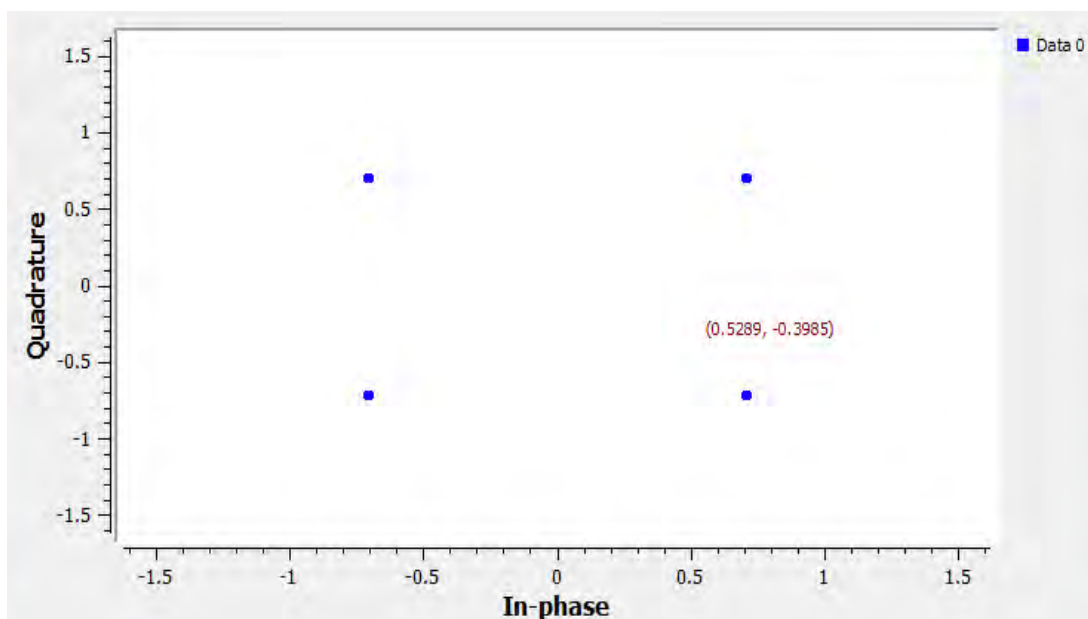


Figura 5.2: Diagrama de constelación de la modulación QPSK.

5.1.2. Receptor del sistema de comunicación.

Casi al mismo tiempo se ejecuta el receptor de la Fig. 4.3 en la otra radio USRP NI 2920, en el entorno gráfico de GRC visualiza el espectro de la señal en la banda de 915 MHz, esta es procesada en banda base para recuperar los datos suministrado por el nodo sensor, el espectro

de la señal de información y el diagrama de constelación demodulado en QPSK se muestran en las figuras 5.3 y 5.4.

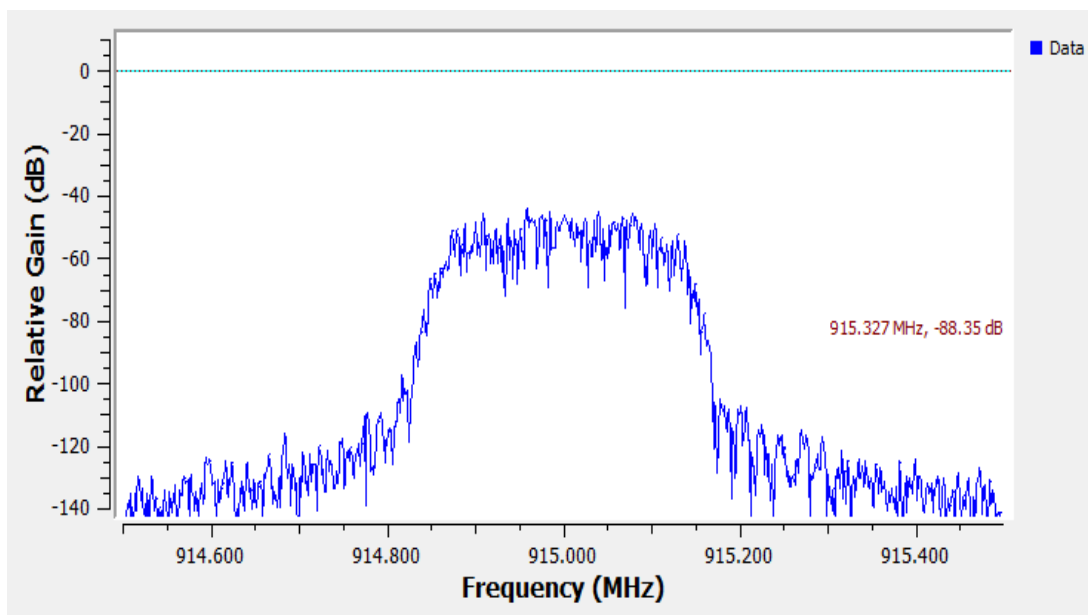


Figura 5.3: Espectro de la señal recibida.

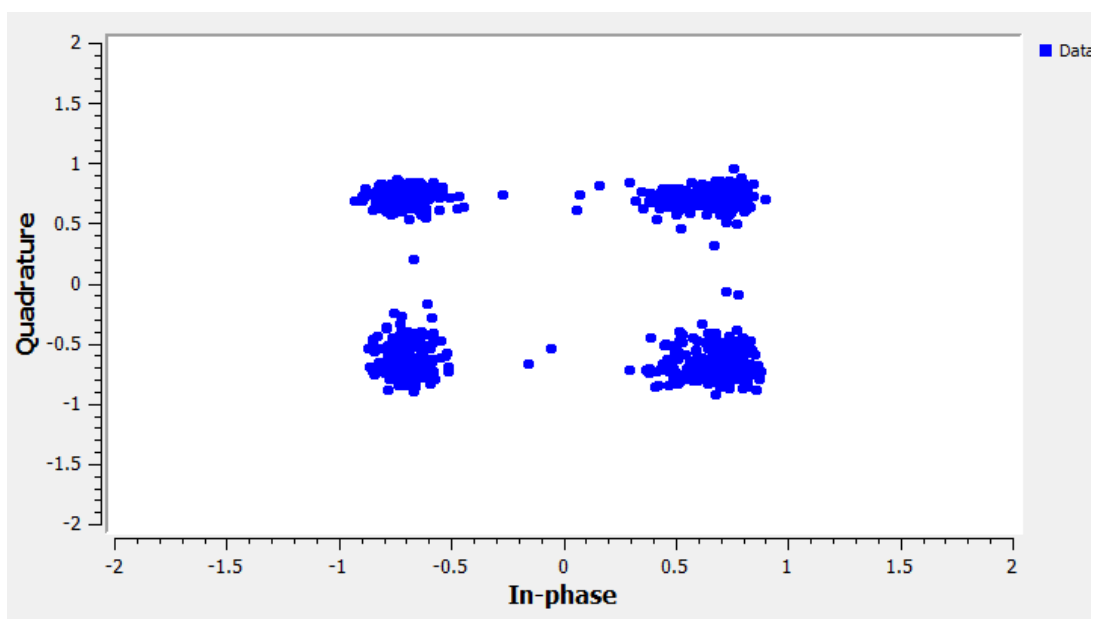


Figura 5.4: Constelación de señal demodulada QPSK

5.2. Implementación del algoritmo estimador M_2M_4 de la SNR en el sistema de comunicación.

Las pruebas de estimación de la relación señal a ruido (SNR) del canal de transmisión, se realizan en dos escenarios; en un enlace guiado con cable coaxial y para un enlace inalámbrico. Al ejecutar el entorno gráfico de GNU radio de la Fig. 4.6 en la radio USRP NI 2920, el bloque `Algorithm_M2M4` estima el nivel de la SNR del canal y los guarda en un archivo `.txt` para diferentes niveles de la SNR de prueba. Para mostrar los resultados, se utiliza el Error Cuadrático Medio Normalizado (NMSE) para comparar los valores obtenidos de la estimación de la SNR en los escenarios implementados, con respecto a los valores estimados de la simulación en `Matlab`, en referencia al Cramer Rao Bound (CRB) como se muestra en la siguiente figura.

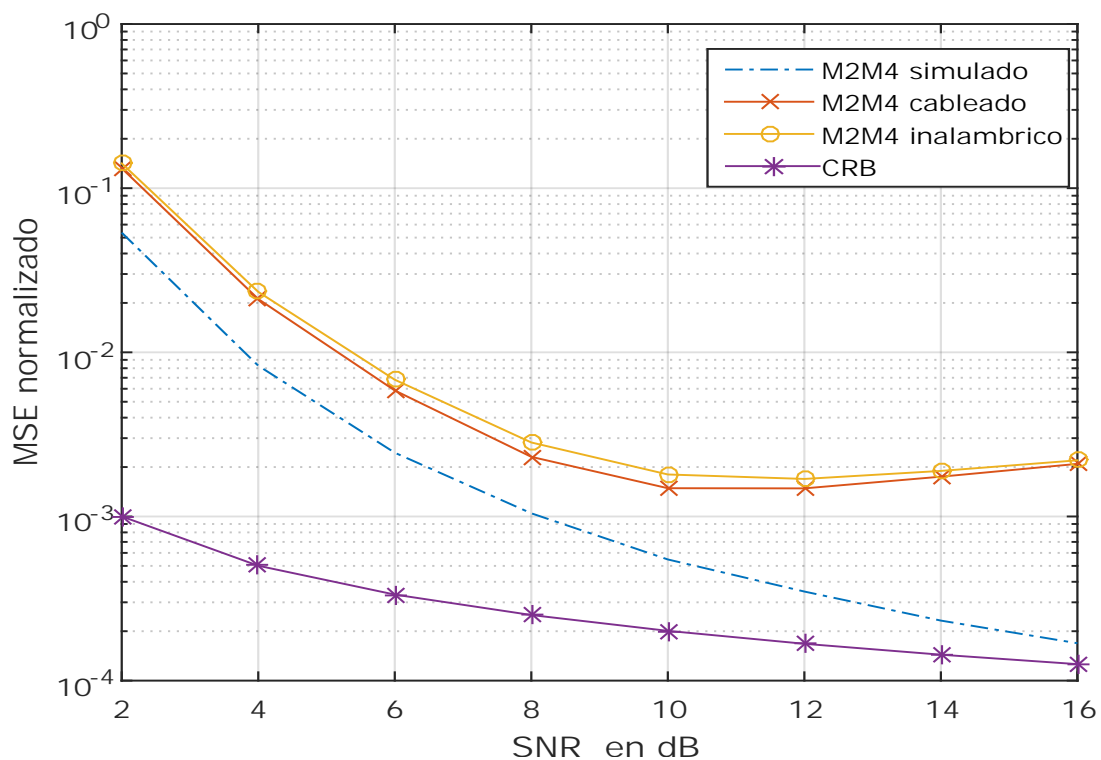


Figura 5.5: Comparación del NMSE para el algoritmo M_2M_4 en tres casos.

En la Fig. 5.5 se observa que el NMSE para la conexión guiada e inalámbrica es mayor con respecto a la simulación, esto se debe al desplazamiento de tiempo y/o espacio que ocurre en la implementación, también se verifica que la conexión por cable tiene mejor desempeño que una conexión inalámbrica.

5.3. Probabilidad de error y Tasa de errores de bit.

Las pruebas para evaluar la tasa de error de bit, se ejecuta el entorno gráfico de GNU radio companion (GRC) de las figuras 4.9 y 4.10, donde el bloque `Error Rate` evalúa la tasa de error de bit en el receptor (estación base), es decir, compara los datos que se transmiten con los datos recibidos, de esta forma determina los bit errados que llegan al receptor para diferentes niveles de la SNR de prueba. A continuación se muestra la tabla de estimación de la SNR, cálculo de la probabilidad de error de bit y la tasa de error de bit.

SNR(dB)	SNR_Estimado(dB)	P_{bit} teórico	BER simulado	BER implementado
2	1.7323	0,0375	0,0382	0,2499
4	3.9751	0,0125	0,0128	0,04765
6	6.062	0,0023	0,0024	0,00999
8	8.1242	1.9091×10^{-4}	2.0710×10^{-4}	5.1131×10^{-4}
10	10.1084	3.8721×10^{-6}	5.3000×10^{-6}	7.9151×10^{-6}
12	12.0817	9.0690×10^{-9}	0	1.7888×10^{-8}
14	14.0252	6.8102×10^{-13}	0	0
16	16.0042	2.2674×10^{-19}	0	0

Tabla 5.1: Resultado de la estimación de la SNR y Cálculo de la probabilidad de error.

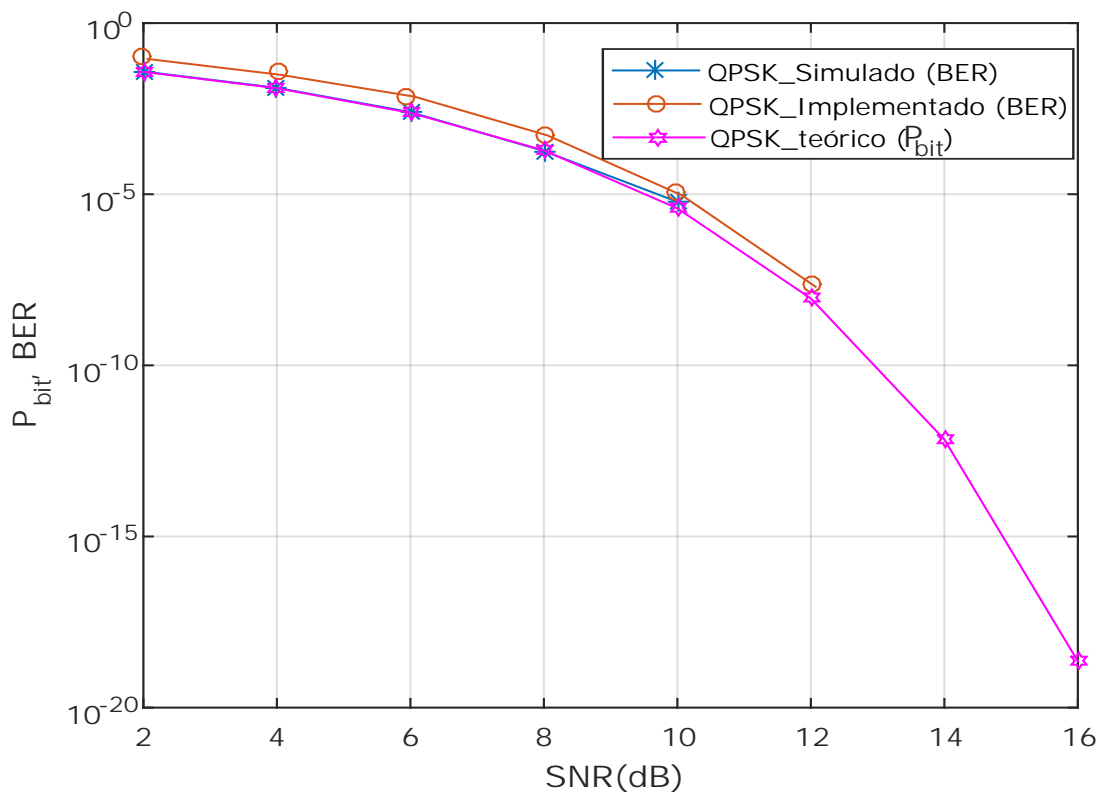


Figura 5.6: Comparación de la probabilidad de error de bit.

En la Fig. 5.6 se observa la gráfica del nivel de la SNR de referencia con respecto a la probabilidad de error de bit P_{bit} , donde se verifica en el rango de la SNR de (2 a 6) dB hay un margen de error notorio en la estimación de la SNR, en el rango (8 a 10) dB la estimación de la SNR es cerna al valor de referencia y finalmente en el rango (12 a 16) dB la tasa de error de bit coincide aproximadamente en los tres casos, es decir, que el algoritmo M_2M_4 estima la SNR del canal con más precisión a mayor nivel de la SNR.

5.4. Pruebas y resultados del sistema de Control Adaptativo de potencia transmitida.

5.4.1. Primer escenario (Enlace guiado).

Con respecto a las pruebas del primer escenario, se ejecuta primero el entorno gráfico GRC del nodo sensor de la Fig. 4.15 y seguidamente el entorno gráfico GRC de la estación base de la Fig. 4.16, para variar el nivel de la SNR del canal; se varía el ruido AWGN suministrado digitalmente en banda base. El nodo sensor adapta la potencia de información a las condiciones del canal de acuerdo a la siguiente figura mostrada.

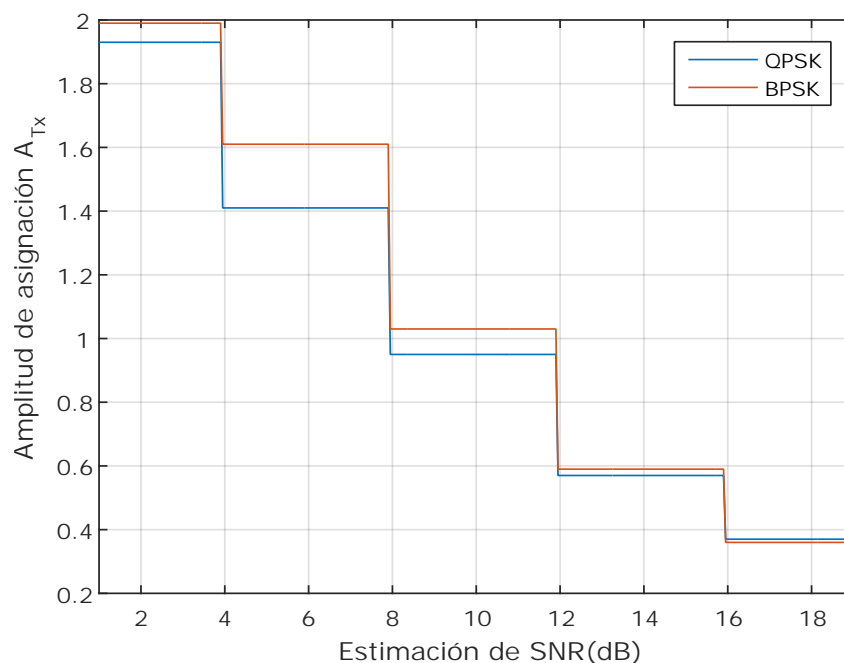


Figura 5.7: Asignación de amplitud A_{Tx} para determinado nivel de SNR estimado.

La Fig. 5.7 muestra los resultados obtenidos de las pruebas y experiencias realizadas en el primer escenario, donde para un determinado rango de la SNR estimada por el algoritmo M_2M_4 corresponde una amplitud constante que debe multiplicar a la señal portadora del nodo sensor, como se detalla en la tabla 5.2. De esta forma el sistema mantiene el nivel de la SNR aproximadamente en 10 dB en el receptor, garantizando una tasa de error de bit confiable en el enlace de transmisión.

$SNR_e(dB)$		Amplitud asignado A_{TX}	
$SNR_e(dB)$	$SNR(dB)$	BPSK	QPSK
$SNR_e < 4$	{..., 1, 2, 3}	1.99	1.93
$4 \leq SNR_e < 8$	{4, 5, 6, 7}	1.61	1.41
$8 \leq SNR_e < 12$	{8, 9, 10, 11}	1.03	0.95
$12 \leq SNR_e < 16$	{12, 13, 14, 15}	0.59	0.57
$SNR_e \geq 16$	{16, 18, 19, ...}	0.36	0.37

Tabla 5.2: Asignación de amplitud para determinado rango de niveles de SNR

Cabe indicar que en el primer escenario, no es viable modificar la potencia de transmisión del equipo USRP NI 2920 para adaptar el sistema a las condiciones del canal, esto es porque el ruido AWGN es suministrada en banda base, es decir, si se aumenta la potencia del equipo, aumenta el nivel de la potencia de la información y del ruido, por ende el nivel de la SNR no varía, sin embargo si se modifica solo la amplitud de la señal transmitida entonces modifica la potencia de la información, de esta manera se logra aumentar o disminuir la intensidad de la señal útil a las condiciones más favorables.

5.4.2. Segundo escenario (Enlace inalámbrico).

las pruebas del segundo escenario, se realiza al ejecutar primero el entorno gráfico de GRC del nodo sensor de la Fig. 4.20 y seguidamente el entorno gráfico de GRC de la estación base de la Fig. 4.21, además se ejecuta el entorno gráfico de GRC de la Fig. 4.18 que transmite la modulación OFDM en la banda 920 MHz con potencia inicial de 0 dB, con el fin de variar el nivel de la SNR del canal, que se muestra en la Fig. 5.8.

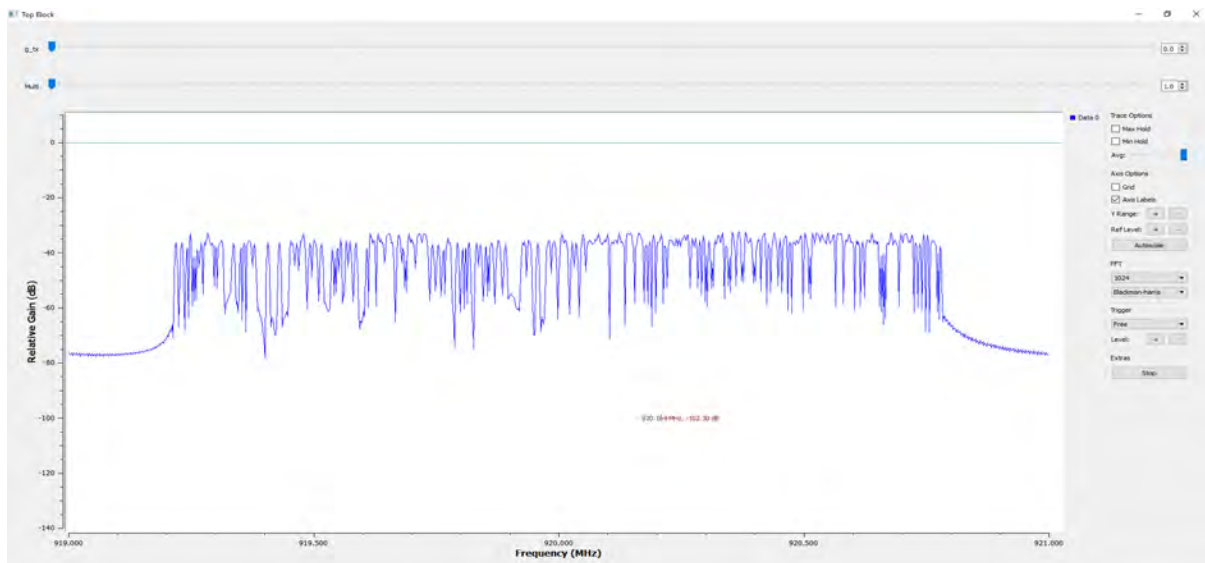


Figura 5.8: Espectro del ruido AWGN

Para adaptar la potencia del nodo sensor a las condiciones del canal, se incrementa el nivel de potencia del equipo USRP NI 2920, a continuación se muestra los resultados.

Resultado del sistema de control adaptativo de potencia transmitida para modulación BPSK.

A continuación se muestra los resultados obtenidos del sistema de control adaptativo de la potencia transmitida en la modulación BPSK a una distancia de 14 m entre el nodo sensor y la estación base, inicialmente se configura el nodo sensor para que transmita a una potencia de 1 dB y la ganancia recibida de la estación base sea de 1 dB, bajo estas consideraciones se tiene un nivel de la SNR en 10 dB aproximadamente, para la primera prueba el ruido externo AWGN se transmite a 0 dB de potencia.

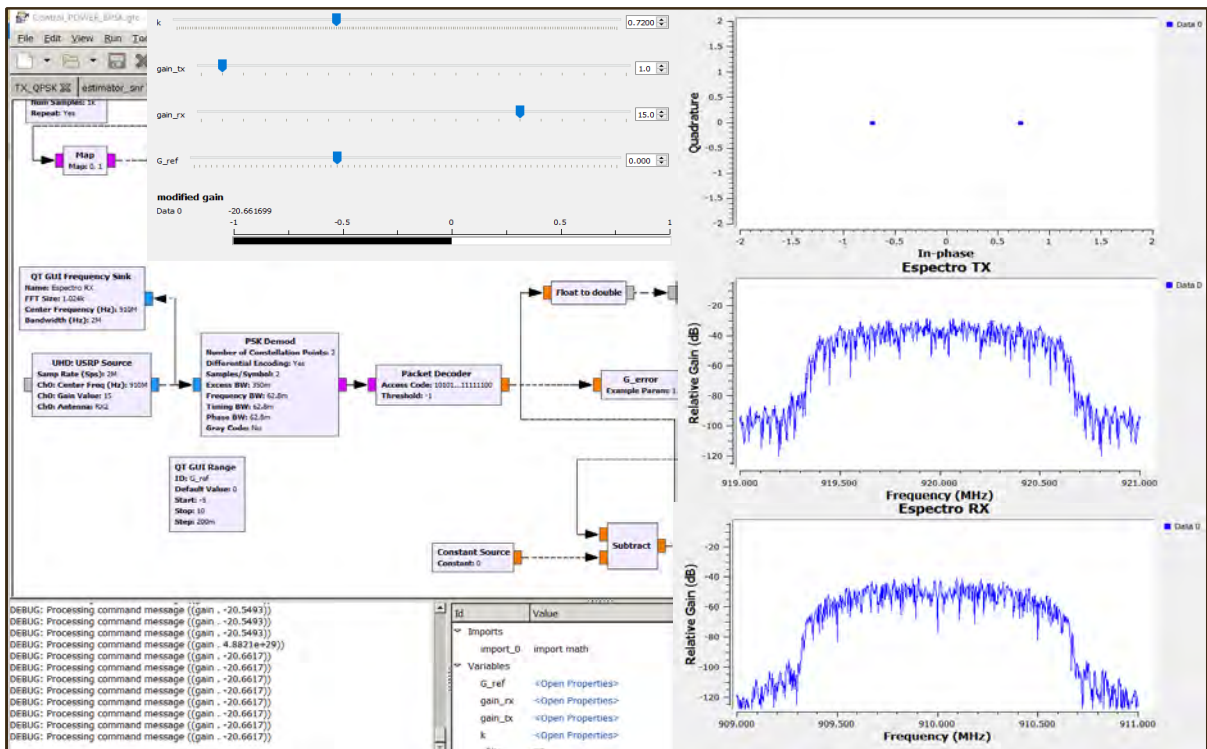


Figura 5.9: Nodo sensor, transmisor BPSK y asignación de potencia transmitida.

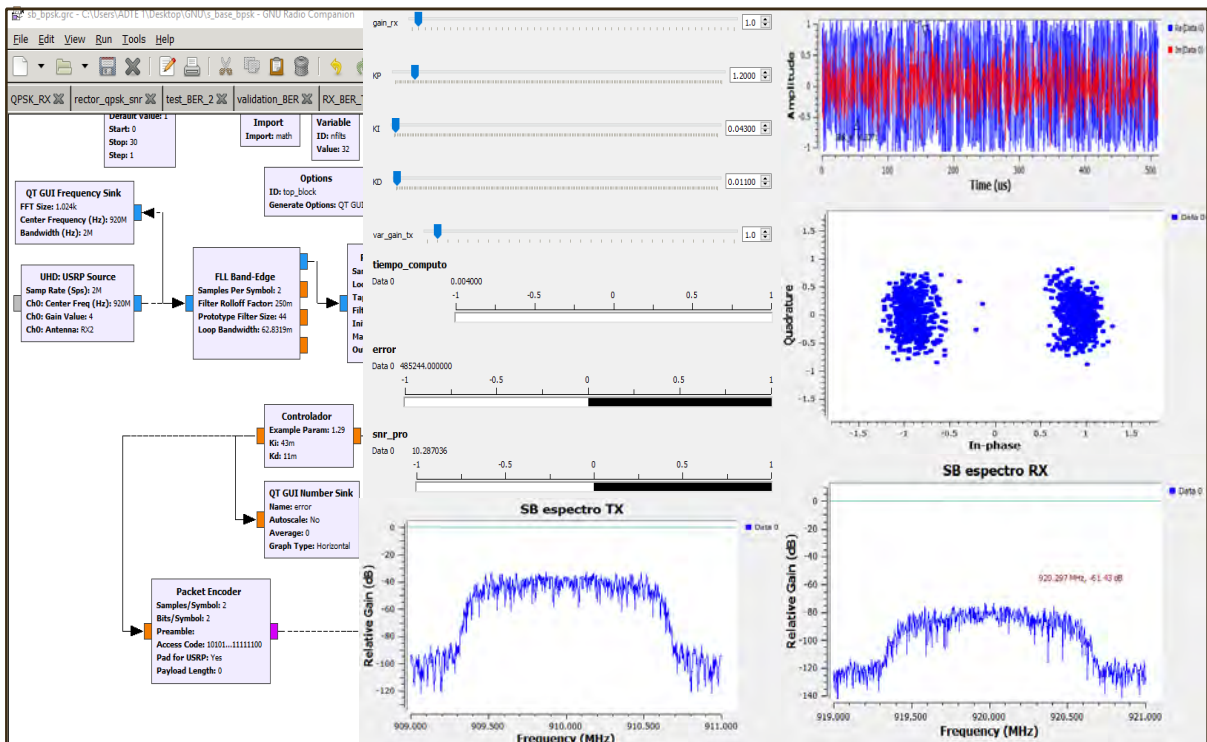


Figura 5.10: Estación base, receptor BPSK y estimación de la SNR del canal.

La Fig. 5.9 muestra la ejecución del entorno GRC del nodo sensor, donde se puede observar el diagrama de constelación de la modulación BPSK, el espectro de transmisión en la banda

de 920 MHz y el espectro del receptor en la banda de 910 MHz, además en la parte de consola se verifica el valor de la potencia transmitida con la que adapta el sistema, en este caso es de -20.5492 dB aproximadamente. La Fig. 5.10 muestra la ejecución del entorno GRC de la estación base, donde se observa la señal recibida en el dominio del tiempo, el diagrama de constelación de la demodulación BPSK, el espectro de la señal recibida y transmitida, los parámetros de las constantes proporcional, integral y derivativo, además se puede verificar que la estimación de la SNR del canal es 10.28 dB. en seguida se muestran más pruebas y resultados.

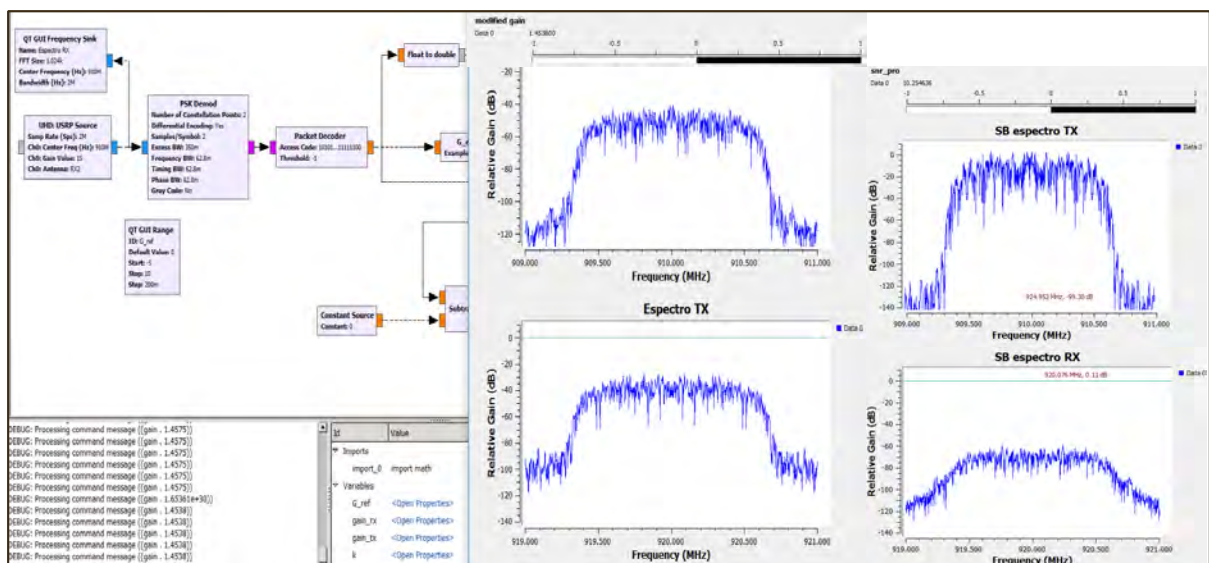


Figura 5.11: Entorno gráfico de GRC, del sistema de control adaptativo de potencia transmitida de la segunda prueba.

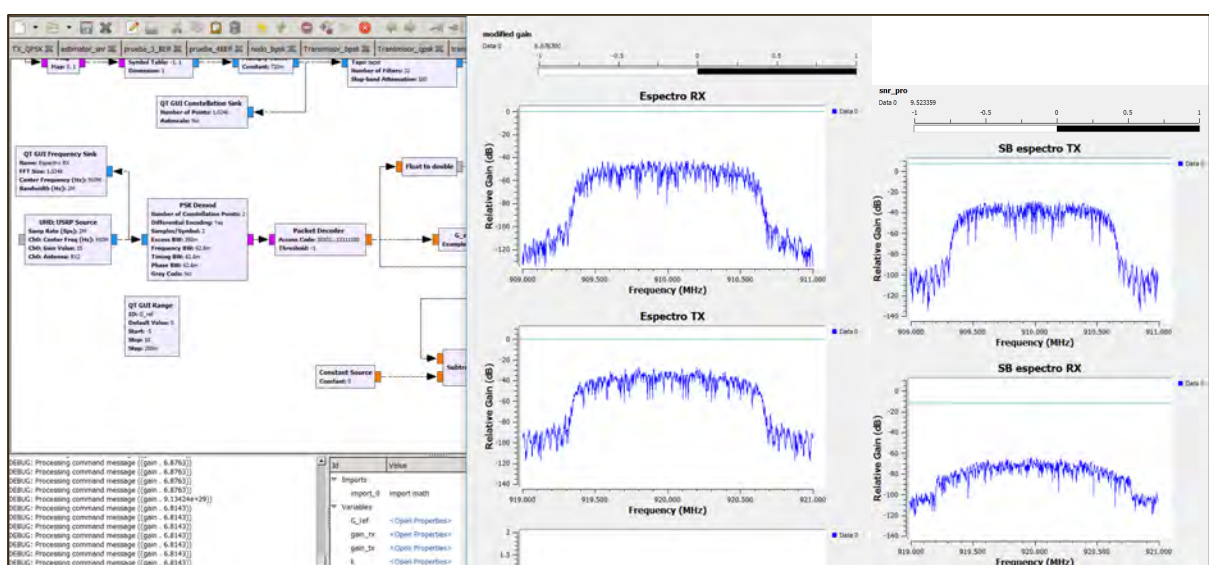


Figura 5.12: Entorno gráfico de GRC, del sistema de control adaptativo de potencia transmitida de la tercera prueba.

La Fig. 5.11 muestra el resultado de la segunda prueba para una potencia transmitida de 2 dB del ruido AWGN externo, la adaptación de la potencia transmitida es de 1.4538 dB del nodo sensor y la SNR estimada se mantiene en 10.2546 dB.

La Fig. 5.12 muestra el resultado de la tercera prueba para una potencia transmitida de 11 dB del ruido AWGN externo, la adaptación de la potencia transmitida es de 6.8763 dB del nodo sensor y la SNR estimada se mantiene en 9.5233 dB.

Resultado del sistema de control adaptativo de potencia transmitida para modulación QPSK.

Para las pruebas del sistema de control adaptativo de la potencia transmitida en la modulación QPSK, considera las mismas condiciones iniciales de la configuración y esquema a la modulación BPSK, con la diferencia de que el ruido AWGN externo no se ejecuta para la primera prueba, en seguida se muestra los resultados en las siguientes imágenes.

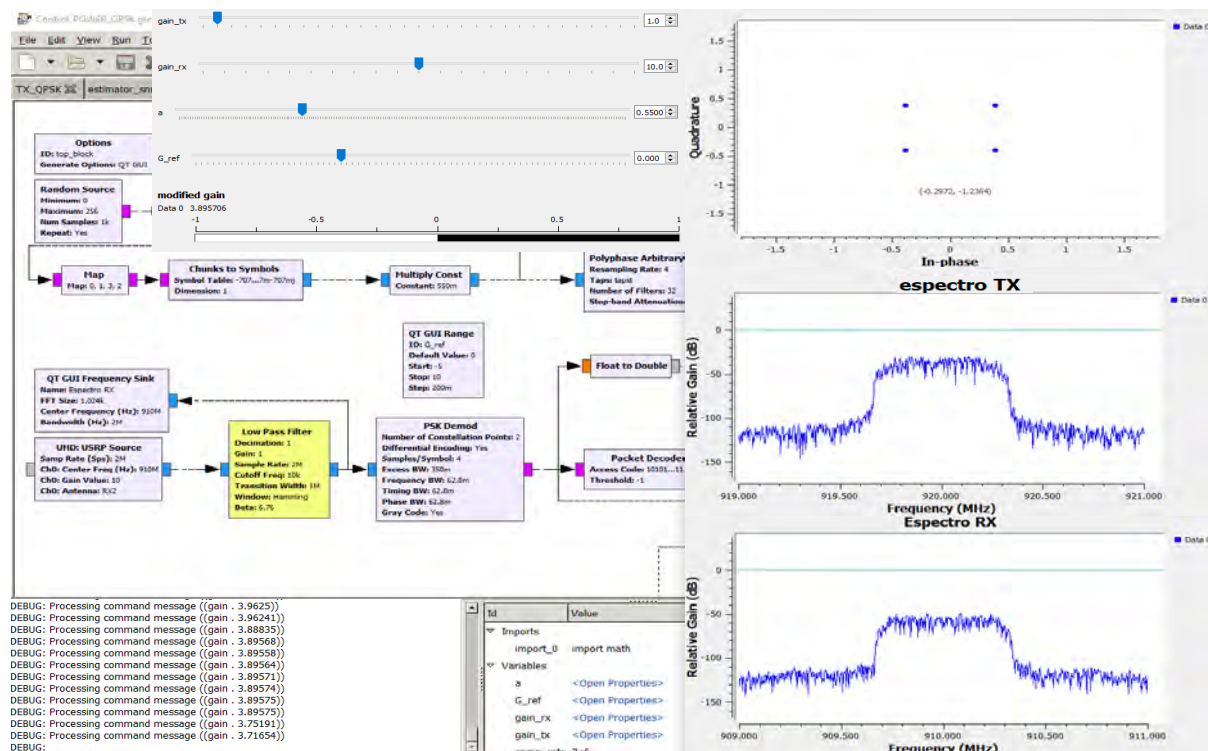


Figura 5.13: Nodo sensor, transmisor QPSK y asignación de potencia transmitida

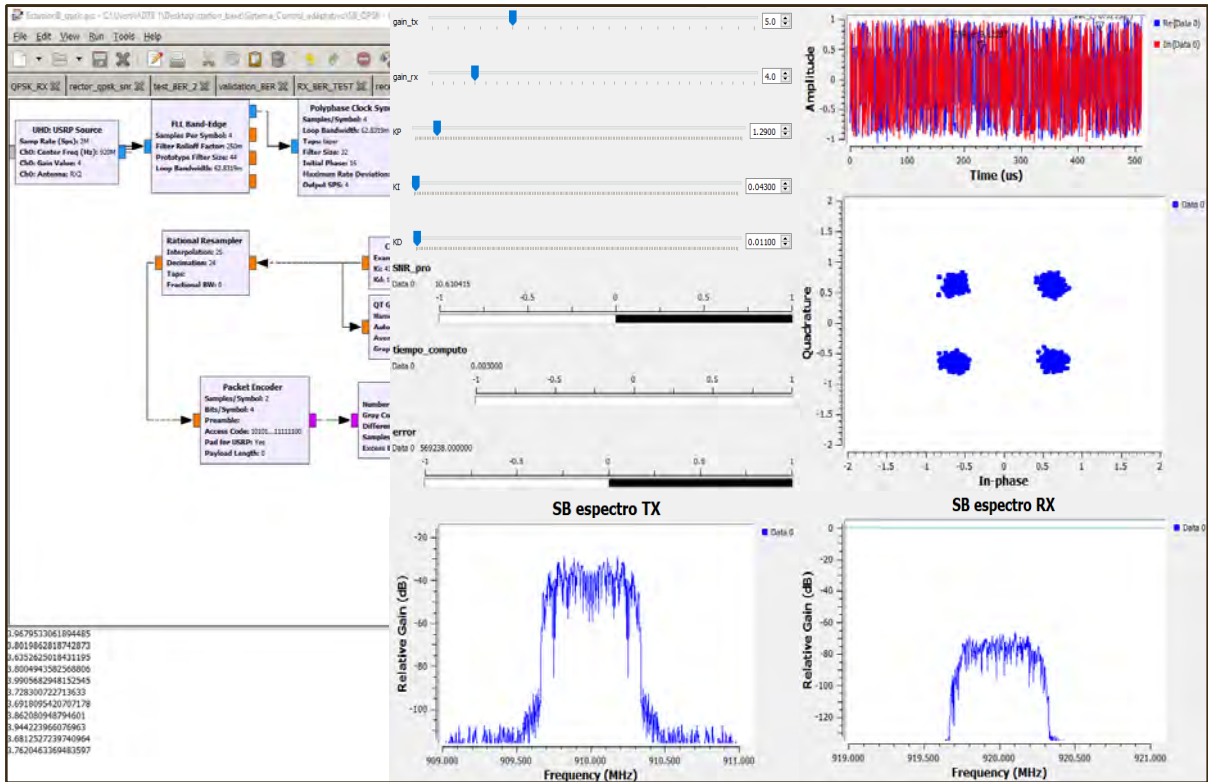


Figura 5.14: Estación base, receptor SB QPSK y estimación de la SNR del canal.

La Fig. 5.13 muestra la ejecución del entorno GRC del nodo sensor, donde se observa el espectro de transmisión de datos en la frecuencia de 920 MHz, el espectro del canal de retorno en 910 MHz, el diagrama de constelación de la modulación QPSK y la consola del entorno GRC, en la cual se verifica el valor de adaptación de la potencia del nodo sensor en 3.8957 dB. La Fig. 5.14 muestra la ejecución del entorno GRC de la estación base, donde se observa la señal recibida en el dominio del tiempo, el diagrama de constelación de la demodulación QPSK, las constantes proporcional, integral y derivativo, y los espectros de transmisión y recepción de los datos, además se puede ver que la SNR estimada del canal es de 10.6104 dB como se espera.

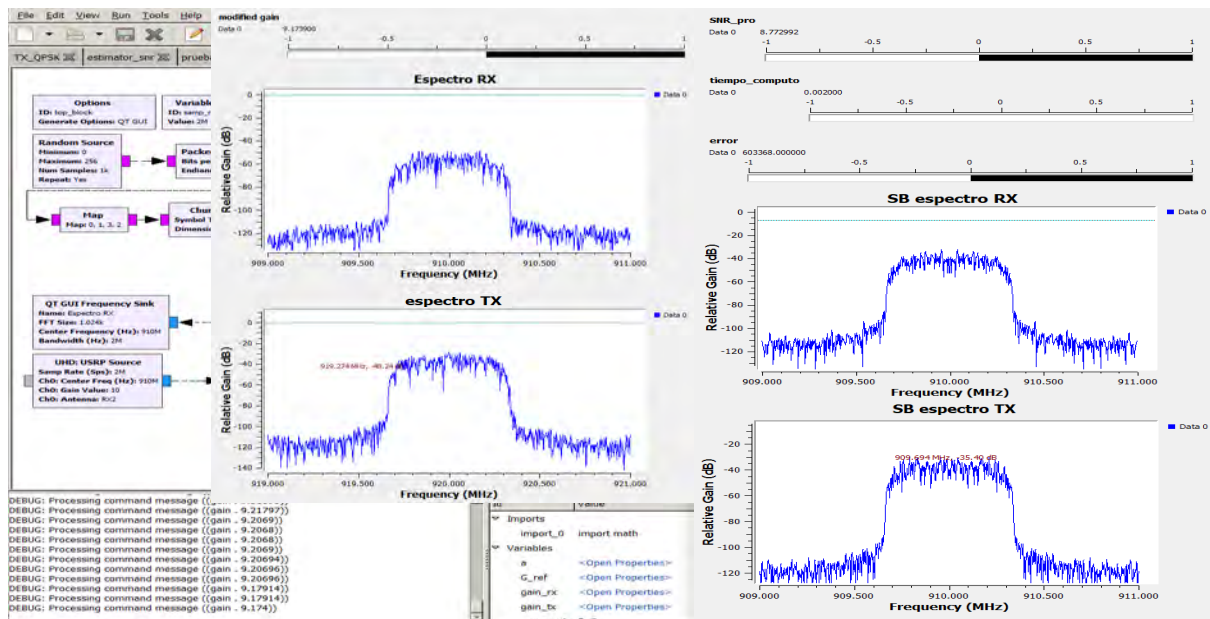


Figura 5.15: Entorno gráfico del entorno GRC, del sistema de control adaptativo de la potencia transmitida, de la segunda prueba en una modulación QPSK

La Fig. 5.15 muestra el resultado de la segunda prueba cuando el ruido AWGN externo transmite a 1dB de potencia y la adaptación de la potencia transmitida es de 9.1739 dB del nodo sensor y la SNR estimada se mantiene en 8.77 dB.

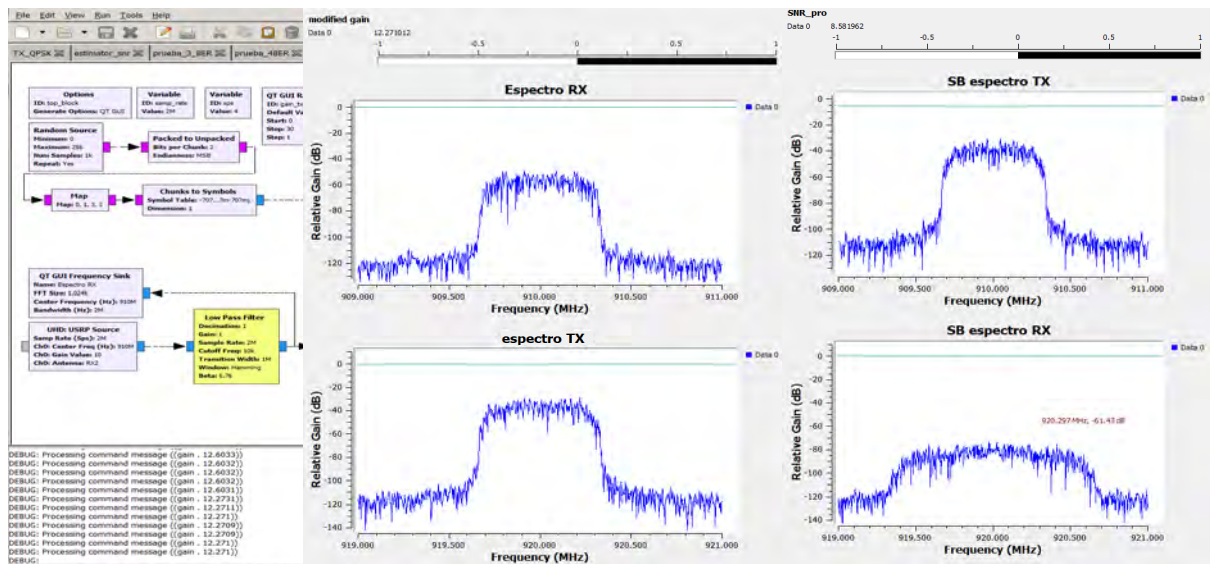


Figura 5.16: Entorno gráfico del entorno GRC, del sistema de control adaptativo de la potencia transmitida, de la tercera prueba en una modulación QPSK

La Fig. 5.15 muestra el resultado de la tercera cuando el ruido AWGN transmite a 5 dB de potencia y la adaptación de la potencia transmitida es de 12.2710 dB del nodo sensor y la SNR estimada se mantiene en 8.5819 dB.

Discusiones

Análisis e interpretación de los resultados.

- La información suministrada por la estación base al nodo sensor, de la gran cantidad de datos recibidos, algún dato es decodificado erróneamente, este al adaptar la potencia transmitida altera el nivel de la SNR, llevándolo a un nivel alto o bajo. Por tal razón el sistema tarda un intervalo de un tiempo mínimo para poder adaptar la potencia transmitida.
- No es posible mantener el nivel de la SNR en 10 dB a una corta distancia entre el nodo sensor y la estación base, ya que para el valor mínimo de potencia transmitida, la SNR estimada es aproximadamente de 30 dB.
- Al transmitir el ruido AWGN externo e incrementar su potencia transmitida mayor a 3 dB, el sistema de control adaptativo de la potencia transmitida en una modulación QPSK adapta el nivel de la SNR aproximadamente en 8.5 dB, esto es debido a que el control PID implementado tiene error en estado estacionario, en este caso el error es aceptable para adaptar la potencia transmitida a las condiciones del canal.

Conclusiones

- Del sistema de control adaptativo de la potencia transmitida implementado en la plataforma de la SDR. La adaptación de la potencia en el nodo sensor del primer escenario, depende de la modificación de la amplitud de la información transmitida, esto es debido a que el ruido AWGN es suministrado en banda base. Por el contrario si se modificará la potencia del dispositivo USRP NI 2920, bajo esta situación aumenta el nivel potencia tanto de la señal y del ruido, por ende el sistema no mantiene el nivel de la SNR como se desea.
- Para modelar el canal transmisión, se emplea la varianza de la información transmitida, a partir de este parámetro se puede determinar la desviación estándar del ruido, para un determinado nivel de la SNR de referencia. El tamaño de muestra y la desviación estándar del ruido, son necesarios para modelar el canal AWGN. Cabe resaltar que los algoritmos de estimación de la SNR, son evaluados en e canal modelado.
- De los algoritmos de estimación de la SNR estudiados para las modulaciones BPSK y QPSK, se concluyó a partir del error cuadrático medio normalizado (NMSE), que el algoritmo DDML ofrece mejor rendimiento con respecto a los demás estimadores de la SNR, en cambio este algoritmo no es implementado en la plataforma de la SDR, debido a que depende de las componentes en fase y cuadratura de la información transmitida. Cabe indicar, que en este trabajo sólo se considera las componentes en fase y cuadratura de la información recibida, para evaluar los algoritmos de estimación de la SNR. Por tal razón los algoritmos seleccionados, son SVR y M_2M_4 , puesto a que son los más próximos a límite inferior de Cramer Rao-Bound.

- Los algoritmos seleccionados, relación señal-variación (SVR) y de segundo-cuarto momento (M_2M_4), son implementados en el entorno de GNU radio companion (GRC). De las pruebas realizadas, se observó que para los valores estimados de la SNR menores a 2 dB se obtuvo un margen de error no deseable con respecto al valor de referencia, y para las estimaciones de la SNR mayores a 2 dB son próximas al valor de referencia. Además se opta por el algoritmo M_2M_4 para implementar en la plataforma de la SDR, ya que este algoritmo ofrece mejor desempeño en el entorno GRC.
- Se diseñó e implementó el sistema de comunicación para un enlace inalámbrico en la plataforma de la radio definido por software (SDR), para lo cual se ha utilizado la radio USRP NI 2920. El procesamiento y la modulación BPSK o QPSK de la información que se desea transmitir, se realiza en banda base en el entorno de GRC, cabe indicar que para el enlace guiado es necesario utilizar un atenuador en la etapa del receptor para proteger el dispositivo empleado.
- Se ha evaluado el desempeño del sistema de control adaptativo de la potencia transmitida en la plataforma de SDR. Este sistema evalúa el nivel de la SNR del canal inalámbrico y con la información adquirida del canal toma la decisión de adaptar la potencia transmitida, si es necesario, con la finalidad de mantener la SNR constante en la estación base. De los resultados obtenidos del sistema de control de la potencia transmitida para la modulación QPSK, se observó, que la SNR estimada varía entre 8.58 dB a 10.6 dB aproximadamente, esto es debido a que, las constantes proporcional, integral y derivativa han sido seleccionadas a prueba y error. Sin embargo la SNR estimada está dentro del intervalo de la tabla de requisitos de la probabilidad de error de bit fijas, en las modulaciones PSK M-aria. De esta forma se tiene un sistema de enlace fiable entre el nodo sensor y la estación base.

Recomendaciones

- Se recomienda que para implementar el sistema de control adaptativo de la potencia transmitida. Se debe garantizar un enlace fiable en el canal de retorno, aumentando la potencia transmitida de la estación base, en caso contrario el sistema no podrá mantener el nivel de la SNR deseado y puede dejar de ejecutar el software.
- Se recomienda implementar el sistema de control adaptativo de potencia transmitida en los siguientes redes inalámbricas como; SigFox (Al-Kashoash y Kemp, 2017), Bluetooth (electronicsnote, s.f.) y Wifi, porque éstos tienen características similares al sistema que se implementó.
- Se recomienda que, para implementar un sistema de enlace guiado, en los dispositivos USRP se debe utilizar un atenuador de radiofrecuencia de 30 dB a la entrada el puerto RF del receptor del sistema para proteger el dispositivo.
- Se recomienda utilizar el entorno de GNU Radio Companion en vista que es un entorno amigable, que ayuda a implementar algunos sistemas de la radio cognitiva.

Referencias

- Al-Kashoash, H., y Kemp, A. (2017, 12). Comparison of 6lowpan and lpwan for the internet of things. *Australian Journal of Electrical and Electronics Engineering*. doi: 10.1080/1448837X.2017.1409920
- Blaszczyk, T., y Lynggaard, P. (2016). An energy-efficient link with adaptive transmit power control for long range networks. *Global Wireless Summit 2016. Proceedings*.
- Di Mu, Yunpeng Ge, Sha, M., Paul, S., Ravichandra, N., y Chowdhury, S. (2017, June). Adaptive radio and transmission power selection for internet of things. En *2017 IEEE/ACM 25th International Symposium on Quality of Service (IWQoS)* (p. 1-10). doi: 10.1109/IWQoS.2017.7969111
- electronicsnote. (s.f.). *Bluetooth radio interface, modulation, and channels*. [urlhttp://www.ti.com/lit/ds/symlink/cc2520.pdf](http://www.ti.com/lit/ds/symlink/cc2520.pdf).
- Goldsmith, A. (2005). *Wireless communications*. Cambridge University Press. doi: 10.1017/CBO9780511841224
- GRC. (2019). *Gnuradiocompanion*. [urlhttps://wiki.gnuradio.org/index.php/GNURadioCompanion](https://wiki.gnuradio.org/index.php/GNURadioCompanion).
- Guangliang Ren, Yilin Chang, y Hui Zhang. (2005, June). A new snr's estimator for qpsk modulations in an awgn channel. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 52(6), 336-338.
- Ijaz, A., Awoseyila, A. B., y Evans, B. G. (2012, Dec). Signal-to-noise ratio estimation algorithm for advanced dvb-rcs systems. *IEEE Transactions on Broadcasting*, 58(4), 603-608.
- Instrument, N. (2013). *Une introduction À la radio dÉfinie par logiciel*. [url](#)

- <https://eduscol.education.fr/sti/sites/eduscol.education.fr.sti/files/ressources/techniques/3801/3801-manuel-tp-usrp-3h.pdf>.
- Instruments, T. (2007). *Cc2520 datasheet*. url<http://www.ti.com/lit/ds/symlink/cc2520.pdf>.
- Introduction to detection theory. (2014). En *Signal processing for cognitive radios* (p. 65-131). John Wiley and Sons, Ltd. doi: 10.1002/9781118824818.ch4
- Joaquín, J., y Garsas, E. (2010, February). Optimización de energía y eficiencia de transmisión con análisis de imparcialidad en comunicaciones inalámbricas adaptativas. , 1-193.
- Lin, S. (2016, March). Adaptive transmission power control for wireless sensor networks. , 12, 1-31.
- Pauluzzi, D. R., y Beaulieu, N. C. (1995, May). A comparison of snr estimation techniques in the awgn channel. En *Ieee pacific rim conference on communications, computers, and signal processing. proceedings* (p. 36-39). doi: 10.1109/PACRIM.1995.519404
- Pätzold, M. (2011). Random variables, stochastic processes, and deterministic signals. En *Mobile radio channels* (p. 17-31). Wiley.
- Rivera, D. E. (2006, Feb). Pid control: New identification and design methods - [book review]. *IEEE Control Systems Magazine*, 26(1), 95-97. doi: 10.1109/MCS.2006.1580159
- Sabate, I. P. (2016). *Implementación de terminales de radio cognitiva en la banda de tv* (Tesis de Master no publicada). Universitat politecnica Catalunya, UPC. (An optional note)
- Tomasi, W., Hernández, G., y Pozo, V. (2003). *Sistemas de comunicaciones electrónicas*. Pearson Educación.
- Vizcaíno, J. R. L. (2014). Redes sensores. En *Modelado y optimización de energía en redes de sensores inalámbricas para la medida de parámetros medioambientales*. (p. 5-11). RiuNet.
- Wyglinski, A., Getz, R., Collins, T., y Pu, D. (2018). *Software-defined radio for engineers*. ArtechHouse. Descargado de <https://books.google.com.pe/books?id=cKR5DwAAQBAJ>

Anexos A

Estudio de canal de transmisión

1.1. Código para simulación del canal AWGN.

```
%% INFORMACION.
% Autor: Michel Chino Quispe.
% Fecha: 9/03/2019.
%% CONFIGURACION
clc;
clear;
%% PARAMETROS
Am = 2;           %Amplitud de la señal portadora.
fi = 2500;       %Frecuencia de la señal de información.
fs = 50*fi;      %Frecuencia de muestreo.
N = 1000;        %Numero de muestras.
SNR = 30;        %Relación señal a ruido.
%% SIMULACION
I = Am*sin(2*pi*fi/fs*(1:N));
x=0:0.000008:0.000008*999;
V_s = var(I);
D_n = sqrt(V_s*10^(-SNR/10));
```

```

N_s = normrnd(0,D_n,[1 N]);
I_r = I + N_s;
% Calculo de voltaje de señal de informacion y ruido.
N_v = mean(abs(N_s));
S_v = mean(abs(I_r));
%Calculo de la relacion señal a ruido.
SNR = 20*log10(S_v/N_v)
%% RESULTADOS Y GRAFICAS
subplot(2,2,1); plot(I,'red')
xlabel('tiempo discreto (n)')
ylabel('Amplitud (Am)')
plot(I_r,'green')
title('SNR = 30 dB')
xlabel('tiempo discreto (n)')
ylabel('Amplitud ')
subplot(2,2,3); plot(N_s,'blue')
xlabel('tiempo discreto (n)')
ylabel('Amplitud ')
subplot(2,2,4); hist(N_s,10)
xlabel('Amplitud (nivel de ruido)')
ylabel('probabilidad de ocurrencia')

```

1.2. Código para simulación de algoritmos de estimación de la SNR.

```

% Simulación de algoritmos de estimacion SNR
% Autor: Michel Chino Quispe
% fecha: 17/05/2019.
%% ESTIMADOR DE SNR PARA MODULACION DE BPSK Y QPSK.

```

```

%% parametros de entrada
clc;
clear;

SNR_max = 24;           % SNR que se estimara.
Nt = 1000;             % Numero de iteraciones.
N = 20                 % Numero de datos.

%% modulación
% message = [1 1 1 0 1 0 1 1 1 0 0 1 1 1 1 0 0 1 0 1 1 0];
message = randi([0 1],1,N);
mod = 'bpsk';
Tx = modulacion(message, mod);
% ===== EVALUANDO PARA SNR VARIABLE =====%
SNR = 2:2:SNR_max;
for j = 1: length(SNR)
for i = 1: Nt;
%% Adición de AWGN al canal de transmision.
Noise = canalawgn(Tx,SNR(j));
%% Informacion a la entrada del receptor.
Zk = signalNoise (Tx,Noise);
%% ALGORITMO DE ESTIMACION MOMENTOS M2M4
SNR_1(i) = M2M4(Zk, Tx);
%% ALGORTIMO DE ESTIMACION PROPUESOTO Ijaz.
SNR_2(i) = Ijaz(Zk,Tx);
%% "ALGORITMO DESICION DIRECTED ALGORITHM BASED ON MAXIMUM
%%LIKELIHOOD ESTIMATION"
SNR_3(i) = DDML (Zk, Tx);

%% ALGORITMO DE ESTIMACION 'SIGNAL TO VARIATION RATIO'

```

```

SNR_4(i) = SVR(Zk,Tx);
end
NMSE_1(j) = sum(((SNR_1-SNR(j))/SNR(j)).^2)/Nt;
NMSE_2(j) = sum(((SNR_2-SNR(j))/SNR(j)).^2)/Nt;
NMSE_3(j) = sum(((SNR_3-SNR(j))/SNR(j)).^2)/Nt;
NMSE_4(j) = sum(((SNR_4-SNR(j))/SNR(j)).^2)/Nt;

CRB(j) = 1*(1+2/SNR(j))/length(Tx);
end
% == Grafica de la variacion de las curvas ====%
semilogy(SNR, NMSE_1, '-+');
hold on;
semilogy(SNR, NMSE_2, 'blue-.');
hold on
semilogy(SNR, NMSE_3, '-x');
hold on
semilogy(SNR, NMSE_4, '-o');
hold on
semilogy(SNR, CRB, '-*black');
grid on
xlabel('SNR en dB')
ylabel('NMSE normalizado')
legend('M2M4', 'Ijaz', 'DDML', 'SVR', 'CRB')
title('Estimador de SNR')

```

Código de función del canal AWGN.

```

%% FUNCION PARA GENERAR EL CANAL AWGN.
function Noise = canalawgn(Tx,SNR_max)
SNR = 2:2: SNR_max; % SNR verdadero.

```

```

for j = 1: length(SNR)
N = length(Tx); % Tamaño de datos transmitidos.
e = sqrt(0.505*10.^(-SNR(j)/10)); % Varianza del Ruido.
nI = normrnd(0,e,[1 N]); % Ruido aleatorio AWGN en Fase.
nQ = normrnd(0,e,[1 N]); % Ruido aleatorio AWGN en Cuadratura.
Noise = (nI + 1j*nQ); % Ruido en fase y cuadratura.
end

```

Código de función Componentes de la información recibida.

```

function Zk = signalNoise (Tx,Noise)
SIk = real(Tx); % Componente en fase
SQk = imag(Tx); % Componente en Cuadratura.
nIk = real(Noise); % ruido en fase.
nQk = imag(Noise); % ruido en cuadratura.
Xk = SIk + nIk; % Componente en fase señal mas ruido
Yk = SQk + nQk; % Componente en cuadratura señal mas ruido
Zk = Xk + 1j*Yk; % Informacion a la entrada del receptor.
plot(Zk,'+') % Grafica de constelacion
xlim ([-2 2])
ylim ([-2 2])
xlabel('Componete en Fase X_k')
ylabel('Componente en Cuadratura Y_k')
end

```

Código de modulación BPSK y QPSK.

```

%% MODULACION BPSK Y QPSK
function Tx = modulacion(message, mod)

switch mod

```

```

case 'bpsk'
L = length(message);           % Tamaño de simbolos ingresados.
SIk = message*2-1;             % Componente en fase.
SQk = zeros(1,L);             % Componente en cuadratura.
Tx = complex(SIk,SQk);         % Datos transmitidos

plot(Tx,'+', 'linewidth',4) % Diagrama de constelacion
xlim ([-2 2])
ylim ([-2 2])
xlabel('Componete en Fase SI_k')
ylabel('Componente en Cuadratura SQ_k')
case 'qpsk'

L = 0.5*length(message);      % Tamaño de simbolos ingresados.
S_k1 = message(2:2:end);      % Posiciones pares del mensaje.
S_k2 = message(1:2:end);      % Posiciones impares del mensaje.

vector = zeros(1 , L); % Vector de inicialización.

for k = 1 : L
if S_k1(k) >S_k2(k) % par de bit asigando en +0.7071 - 0.7071i
vector(k)=(7*pi)/4;
end
if S_k1(k)<S_k2(k) % par de bit asigando en -0.7071 + 0.7071i
vector(k)=(3*pi)/4;
end
if S_k1(k)>0&S_k2(k)>0 % par de bit asigando en +0.7071 + 0.7071i
vector(k)= pi/4;

```



```

end
if S_k1(k)<1 & S_k2(k)<1 % par de bit asigando en -0.7071 - 0.7071i
vector(k)= 5*pi/4;
end
end
% información
Sk = vector; % Información en fase y cuadratura.
SIk = cos(Sk); % Información en fase.
SQk = sin(Sk); % Información en cuadratura.
Tx = SIk + 1j*SQk; % Información transmitido.
plot(Tx,'+', 'linewidth',4) % Diagrama de constelacion
xlim ([-2 2])
ylim ([-2 2])
xlabel('Componente en Fase SI_k')
ylabel('Componente en Cuadratura SQ_k')
otherwise
disp('Unknown modulation')
end

```

Código del algoritmo DDML.

```

%% "ALGORITMO DESICION DIRECTED ALGORITHM
%% BASED ON MAXIMUM LIKELIHOOD ESTIMATION"

%% "Signal-to-Noise Ratio Estimation Algorithm for
%% Advanced DVB-RCS Systems"
function SNR_3 = DDML (Zk, Tx)
L = length(Tx); % Tamaño de simbolos ingresados
Xk = real(Zk); % componentes en fase.
Yk = imag(Zk); % Componentes en cuadratura.

```

```

SIk = real(Tx); % Componentes en fase al transmitir.
SQk = imag(Tx); % componentes en cuadratura al transmitir.
% Calculo del algoritmo propuesto.
F = sum(SIk.^2 + SQk.^2
S_DDML=((sum(Xk.*SIk + Yk.*SQk))/L)/((F))/L).^2;
N_DDML=(sum(Xk.^2 + Yk.^2))/L-S_DDML*((F))/L);
SNR_3 = 10*log10(S_DDML/N_DDML);
end

```

Código del algoritmo M_2M_4 .

```

%% ALGORITMO DE ESTIMACION MOMENTOS M2M4
function SNR_1 = M2M4(Zk, Tx)

L = length(Tx); % Tamaño de simbolos ingresados.
Xk = real(Zk); % componentes en fase en el receptor.
Yk = imag(Zk); % Componentes en cuadratura en el receptor.
% Segundo Momento
M2 = (sum(abs(Xk).^2)+sum(abs(Yk).^2))/L;
% Cuarto Momento
D = 2*sum((Xk).^2.*(Yk).^2)
M4=(sum(abs(Xk).^4)+sum(abs(Yk).^4)+D)/L;
S1 = sqrt(2*M2.^2-M4); % intensidad de la Señal.
N1 = M2-S1; % Intensidad del Ruido.
SNR_1 = 10*log10(S1/N1); % Valor estimado en dB
end

```

Código del algoritmo Ijaz.

```

%% ALGORITMO DE ESTIMACION PROPUESOTO Ijaz.
%% "Signal-to-Noise Ratio Estimation Algorithm

```

```

%%for Advanced DVB-RCS Systems"
function SNR_2 = Ijaz(Zk,Tx)
L = length(Tx); % Tamaño de estos simbolos ingresados.

Xk = real(Zk); % componente en fase.
Yk = imag(Zk); % Componente en cuadratura.
% Calculo del 2do momento M2
M2 = (sum(abs(Xk).^2)+sum(abs(Yk).^2))/L; 2
% Calculo del algoritmo que se propuso en el paper.
S_propuesto = ((sum(abs(Xk)))^2 + (sum(abs(Yk)))^2)/(L^2);
SNR_2 = 10*log10(S_propuesto/(M2-S_propuesto));
end

```

Código del algoritmo SVR.

```

%% ALGORITMO DE ESTIMACION 'SIGNAL TO VARIATION RATIO'
%% A Comparison of SNR Estimation Techniques for the AWGN Channel
function SNR_4 = SVR(Zk,Tx)
L = length(Tx); % Tamaño de simbolos ingresados.
E = abs(Zk); % Valor absoluto de la informacion en el receptor.
E(end) = []; % elimina el ultimo componente de la matrix.
Yn_12 = E.^2; % Eleva al cuadrado sin la ultima componente.
% Calculo del algoritmo.
G= (sum(abs(Zk).^4)/L - ((sum(abs(Zk).^2)/L)*((sum(Yn_12))/(L-1))))
B = ((sum(abs(Zk).^2)/L)*((sum(Yn_12))/(L-1))) /G;

SNR_4 = (10*log10((B-1) + sqrt(B*(B-1))));
end

```

Anexos B

Implementación de los algoritmos de la SNR

2.1. Código del algoritmo *SVR* de estimación de la SNR

```
import numpy as np
from gnuradio import gr
import math
import pmt

class blk(gr.sync_block):
    def __init__(self): # only default arguments here
        """arguments to this function show up as parameters in GRC"""
        gr.sync_block.__init__(
            self,
            name='SVR', # will show up in GRC
            in_sig=[np.complex64],
            out_sig=[np.complex64]
        )

    def work(self, input_items, output_items):
```

```

"""example: multiply with constant"""
L = 64 # Numero de simbolos
k = 0
in1 = input_items[0][:] # informacion recibida
Zn = np.delete(in1,0)
Zn_1 = np.delete(in1,-1)
vector1 = np.absolute(Zn)**2
vector2 = vector1**2
vector3 = np.absolute(Zn_1)**2
SNR_3 = np.zeros((1,64)) # Iniliza vector de la SNR

for i in range(0, len(input_items[0])):

    if 64*i + L <= 4094:

        b1 = vector1[64*i:64*i+L]
        b2 = vector2[64*i:64*i+L]
        b3 = vector3[64*i:64*i+L]

        A = math.fsum(b1*b3)/L
        B = math.fsum(b2)/L
        if B > A:
            beta = A/(B-A)
            if beta > 1:
                # Evalua la SNR por ventanas de 64
                snr_w = np.absolute( beta - 1 + math.sqrt(beta*(beta-1)))
                SNR_3[0,k] = 10*math.log10 (snr_w)
                k = k+1

SNR = np.sum(SNR_3)/k # Promedio de SNR estimada

```

```

for j in range(0,4096,1024):
    item_index = j
    # Etiqueta la SNR estimda en la Señal de salida
    offset = self.nitems_written(0)+item_index
    key = pmt.intern("SNR")
    value = pmt.from_double(np.double(SNR))
    self.add_item_tag(0, offset, key, value)
    output_items[0][:] = input_items[0][:]
return len(output_items[0])

```

2.2. Código del algoritmo M_2M_4 de estimación de la SNR

```

import numpy as np
from gnuradio import gr
import math
import pmt
class blk(gr.sync_block):
    def __init__(self): # only default arguments here
        """arguments to this function show up as parameters in GRC"""
        gr.sync_block.__init__(
            self,
            name='M2M4', # will show up in GRC
            in_sig=[np.complex64],
            out_sig=[np.complex64]
        )
    def work(self, input_items, output_items):
        """example: multiply with constant"""
        L = 64 # numero de simbolos
        in0 = input_items[0][:] # Entrada de simbolos al buffer.

```

```

Xk = in0.real # parte real de la informacion recibida.
Yk = in0.imag # parte imaginaria de la informacion recibida.

v1 = (np.absolute(Xk))**2
v2 = (np.absolute(Yk))**2
v3 = v1**2
v4 = v2**2
v5 = 2*v1*v2

k = 0
SNR_2 = np.zeros((1,64))
# Calcular la SNR con el algoritmo M2M4
for i in range(0, 64):
    if 64*i+L <= 4096:

        vector1 = v1[64*i:64*i+L]
        vector2 = v2[64*i:64*i+L]
        vector3 = v3[64*i:64*i+L]
        vector4 = v4[64*i:64*i+L]
        vector5 = v5[64*i:64*i+L]

        # Evalua el segundo momento M2
        M2 = math.fsum(vector1 + vector2)/L}

        # Evalua el cuarto momento M4
        M4 = math.fsum(vector3 + vector4 +vector5)/L

        Signal = float (math.sqrt(np.absolute(2*M2**2 - M4)))

        Noise = float(np.absolute(M2 - Signal))

    if Noise > 0:

        SNR_2[0,k] = 10*math.log10((Signal/Noise))

        k = k +1

```

```
SNR = np.sum(SNR_2)/k # SNR promediada en ventana de 64
```

```
for j in range (0,4096,2048):
```

```
    item_index = j
```

```
    # Etiqueta el valor de la SNR en señal de salida
```

```
    offset = self.nitems_written(0) + item_index
```

```
    key = pmt.intern("SNR")
```

```
    value = pmt.from_double(np.double(SNR))
```

```
    self.add_item_tag(0, offset, key, value)
```

```
    output_items[0][:] = input_items[0][:]
```

```
    return len(output_items[0])
```

2.3. Implementación del sistema de comunicación para modulación BPSK.

2.3.1. Transmisor BPSK en plataforma de SDR.

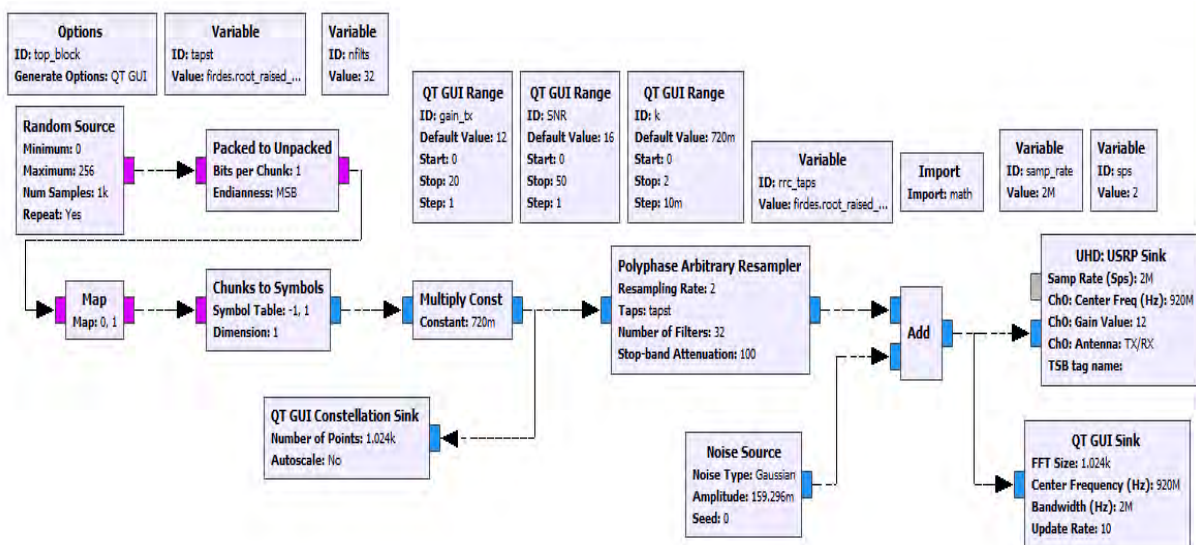


Figura 2.1: Diagrama de bloque del transmisor BPSK en GRC.

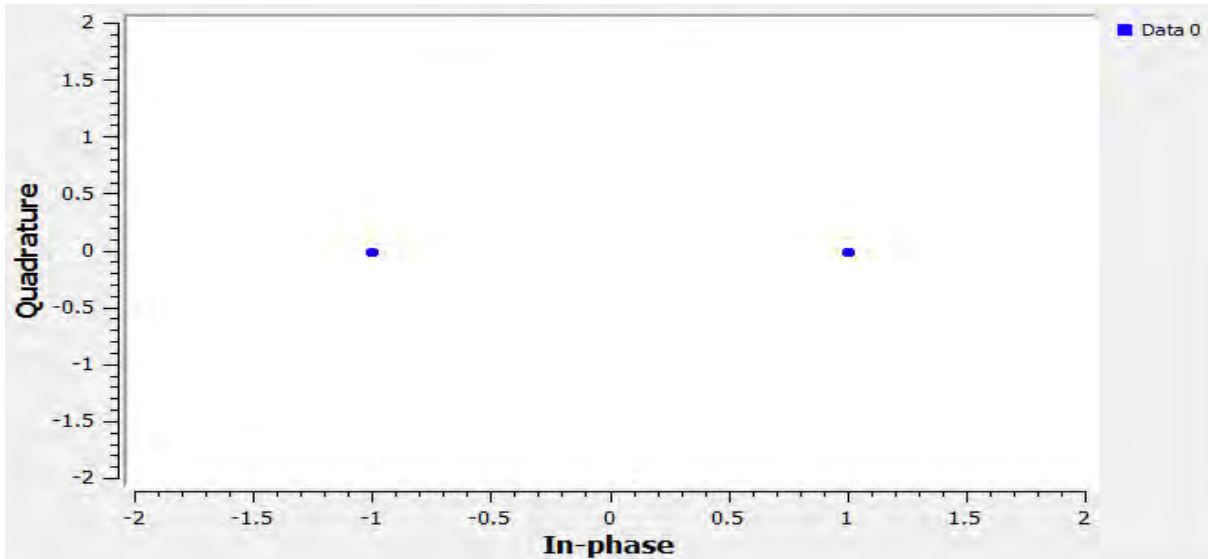


Figura 2.2: Constelación de BPSK del transmisor.

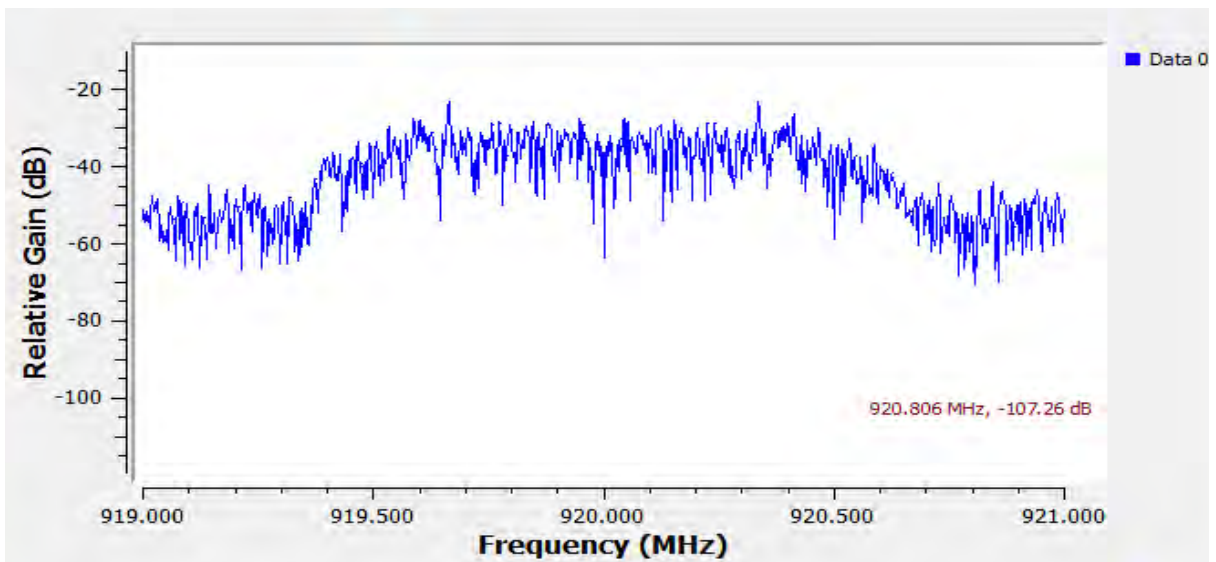


Figura 2.3: Espectro de la información transmitida de BPSK.

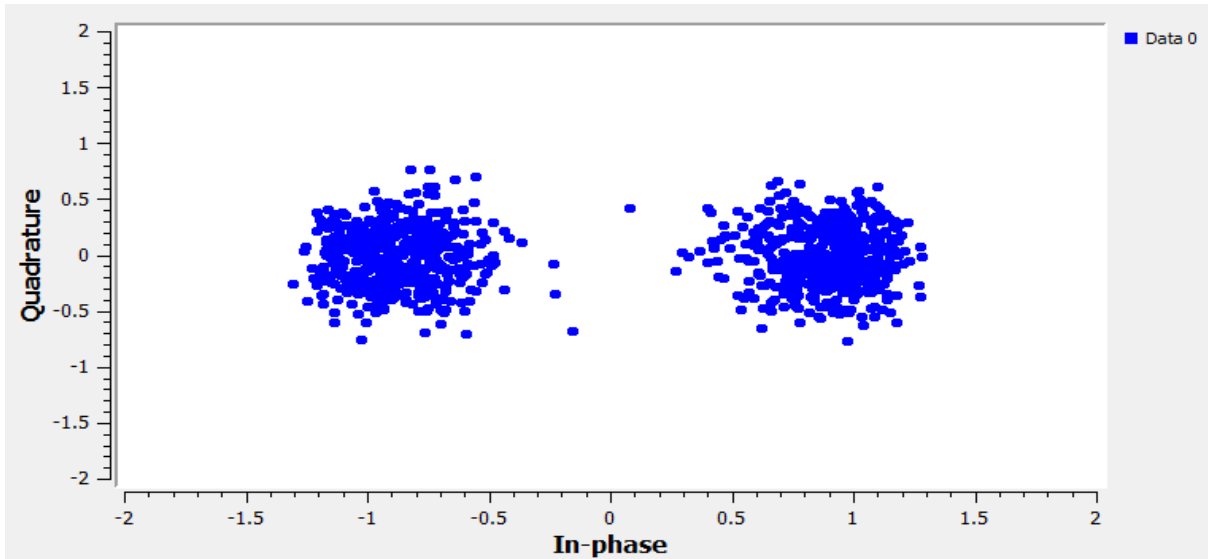


Figura 2.5: Cosntelación de BPSK del receptor.

2.3.2. Receptor BPSK en plataforma de SDR

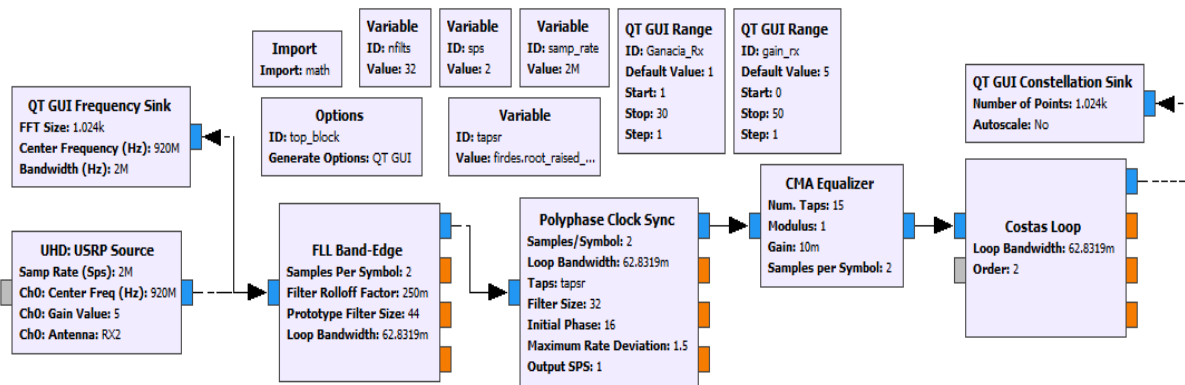


Figura 2.4: Diagrama de bloque de receptor BPSK en GRC.

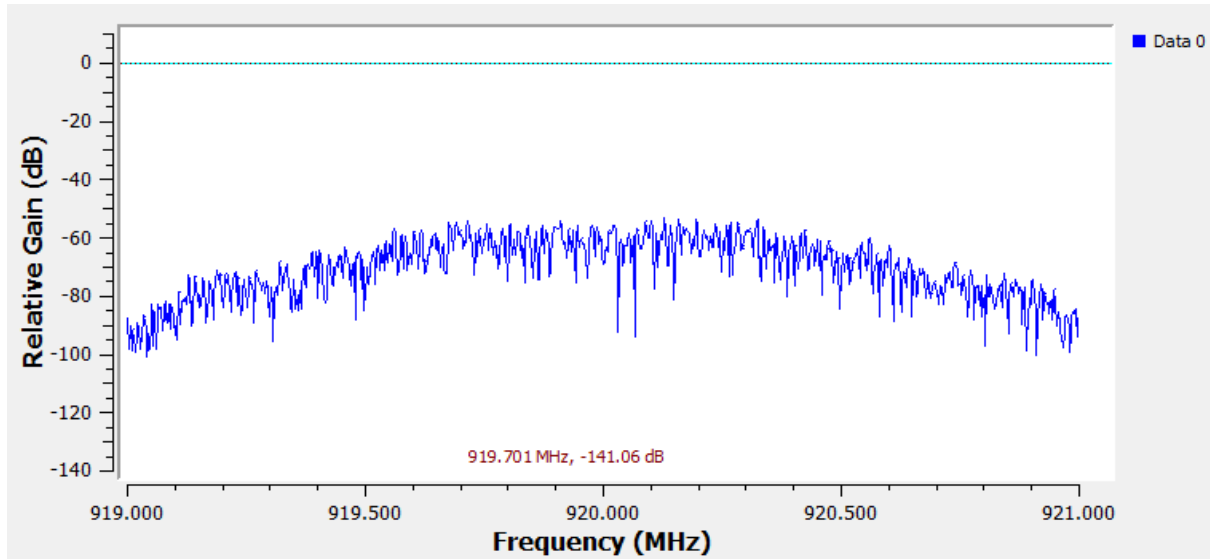


Figura 2.6: Espectro de la información recibida de BPSK.

Anexos C

Pruebas y Resultados

3.1. Pruebas de la estimación del algoritmo M_2M_4 .

3.1.1. imágenes de Pruebas de estimación de la SNR del canal Guiado.

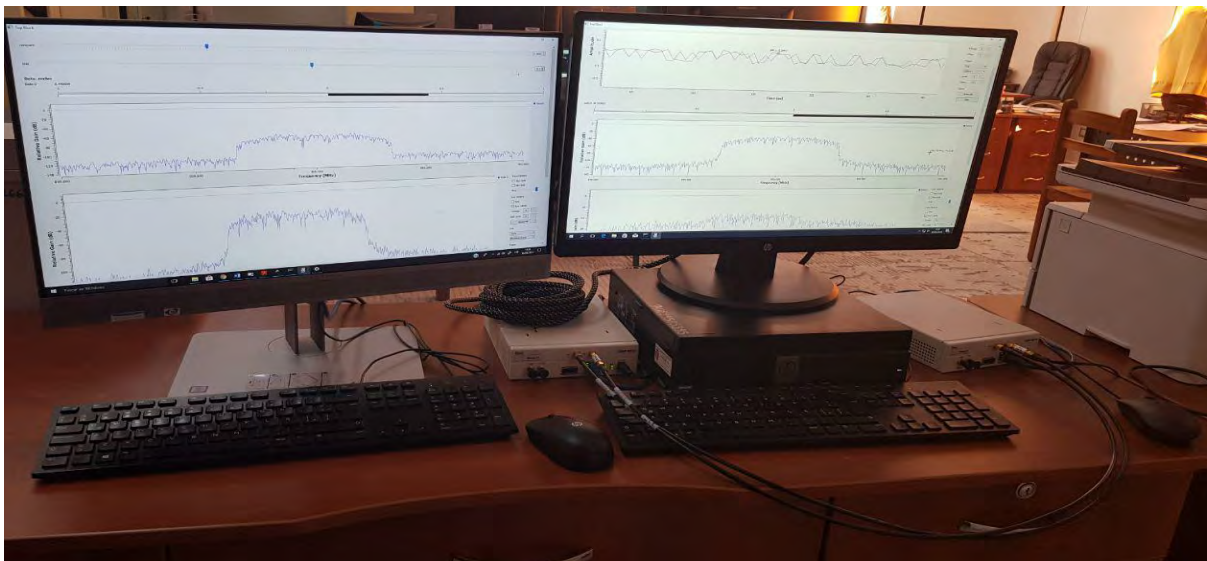


Figura 3.1: Estimación de la SNR para un enlace guiado.

3.1.2. Prueba de estimación de la SNR del canal Enlace inalámbrico.



Figura 3.2: Distancia entre antena transmisora y receptora $d = 1\text{ m}$ del sistema de comunicación.

3.2. Código del cálculo de probabilidad de error y tasa de error de bit.

```
% calculo de probabilidad de error de bit
% fecha = 27/07/2019
% Autor: Michel Chino Quispe
clear all;
close all;
L = 10000000; % Tamaño de muestra.
SNR_t = 2:2:16; % SNR en dB a evaluar.
for i = 1 : length(SNR_t)
    si=2*(round(rand(1,L))-0.5);
    sq=2*(round(rand(1,L))-0.5);
    s=si+j*sq;
    e = sqrt(0.505*10.^(-SNR_t(i)/10));
    ni= normrnd(0,e,[1 L]);
    nq= normrnd(0,e,[1 L]);
```

```

n = ni + j*nq;
z = s + n;
si_=sign(real(z));
sq_=sign(imag(z));
ber1=(L-sum(si==si_))/L;
ber2=(L-sum(sq==sq_))/L;
format long;
% Calculo de BER simulada.
ber(i)=mean([ber1 ber2]);
y = length(ber);
end
% Calculo de BER en plataforma de SDR
BER_sdr= [0.249973325      0.047655692      0.0099953231
0.0005113409      7.96685E-06      1.7888E-08      0      0];
SNR_te = 10.^(SNR_t./10);
% Calculo de BER de forma teorico
BER_te = 0.5.*erfc(sqrt(SNR_te));
semilogy(SNR_t, ber, '*-')
hold on
semilogy(SNR_t, BER_sdr, 'o-')
hold on
semilogy(SNR_t, BER_te, '-mh')
xlabel('SNR (dB)')
ylabel('P_{be}')
legend('QPSK\teorico', 'QPSK\simulado', 'QPSK\implementado')
grid on

```

3.3. Pruebas del sistema de control adaptativo de potencia transmitida.

3.3.1. Código de la asignación de amplitud al nodo sensor en un enlace guiado.

```
"""  
Embedded Python Blocks:  
  
"""  
  
import numpy as np  
from gnuradio import gr  
  
class blk(gr.sync_block):  
    """Embedded Python Block example - a simple multiply const"""  
  
    def __init__(self, example_param=1.0):  
        """arguments to this function show up as parameters in GRC"""  
        gr.sync_block.__init__(  
            self,  
            name='Gain_asignement', # will show up in GRC  
            in_sig=[np.float32],  
            out_sig=[np.float32]  
        )  
  
        # if an attribute with the same name as a parameter is found,  
        # a callback is registered (properties work, too).  
        self.example_param = example_param  
  
    def work(self, input_items, output_items):  
        """example: multiply with constant"""
```

```

ino = np.mean(input_items[0][:])
SNR_est = round(ino)
one = np.ones(len(input_items[0][:]))
A_tx = 0
if SNR_est < 4:
A_tx = 1.93*one
if SNR_est >= 4 and SNR_est < 8:
A_tx = 1.41*one
if SNR_est >= 8 and SNR_est < 12:
A_tx = 0.95*one
if SNR_est >= 12 and SNR_est < 16:
A_tx = 0.57*one
if SNR_est >= 16:
A_tx = 0.37*one
output_items[0][:] = A_tx * self.example_param
return len(output_items[0])

```

3.3.2. Código de la asignación de potencia transmitida al nodo sensor en un enlace inalámbrico.

```

import numpy as np
from gnuradio import gr
import time

global EAnt
EAnt = 0 # error anterior
global e_I
e_I = 0 # error inicial
global A_I
A_I = 0 # Accion integral

```



```

global A_P
A_p = 1 # Acción proporcional

global A_D
A_D =0 # Acción Derivativo

global T_anterior
T_anterior = time.time() # tiempo inicial

class blk(gr.sync_block):
    """Embedded Python Block example - a simple multiply const"""

    def __init__(self, Kp=1.0, Ki=1.0, Kd=1.0): #
        """arguments to this function show up as parameters in GRC"""
        gr.sync_block.__init__(
            self,
            name='Controlador', # will show up in GRC
            in_sig=[np.float32],
            out_sig=[np.float32]
        )
        # if an attribute with the same name as a parameter is found,
        # a callback is registered (properties work, too).
        self.Kp = Kp # Constante proporcional
        self.Ki = Ki # Constante Integral
        self.Kd = Kd # Constante derivativo
    def work(self, input_items, output_items):
        """example: multiply with constant"""

        global T_anterior

        global A_I

        global e_I

        global A_P

```

```

global EAnt
global A_D
SNR_m = np.mean(input_items[0][:])

e = 10 -SNR_m

## =====

T_actual = time.time()
T = T_actual - T_anterior
if T > 0.00001:
A_p = self.Kp*(e) # acción proporcional
e_I = e*T + e_I
A_I = e_I*self.Ki # acción integral
E_D = (e-EAnt)/T
A_D = E_D*self.Kd # acción derivativo
EAnt = e
T_anterior = T_actual
A = A_p + A_I + A_D
er = A*10000+500000 # error de salida
print A
output_items[0][:] = np.around(er,0)
return len(output_items[0])

```

3.3.3. Estimación de la SNR del canal de transmisión en espacio Libre.

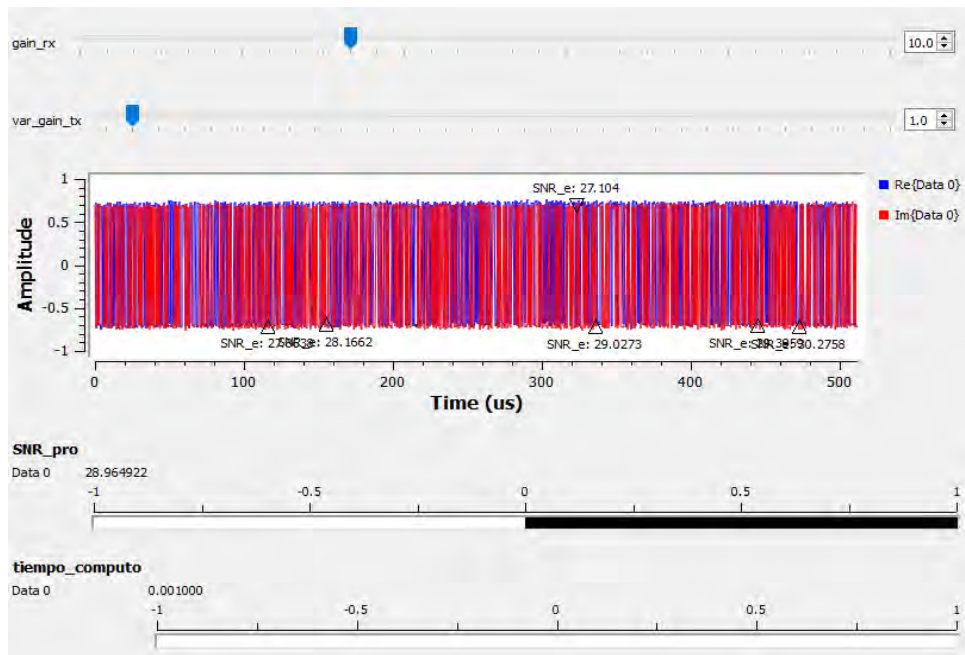


Figura 3.3: Información recibida en el dominio del tiempo QPSK, con $G_{tx} = 1dB$ y $G_{rx} = 10dB$

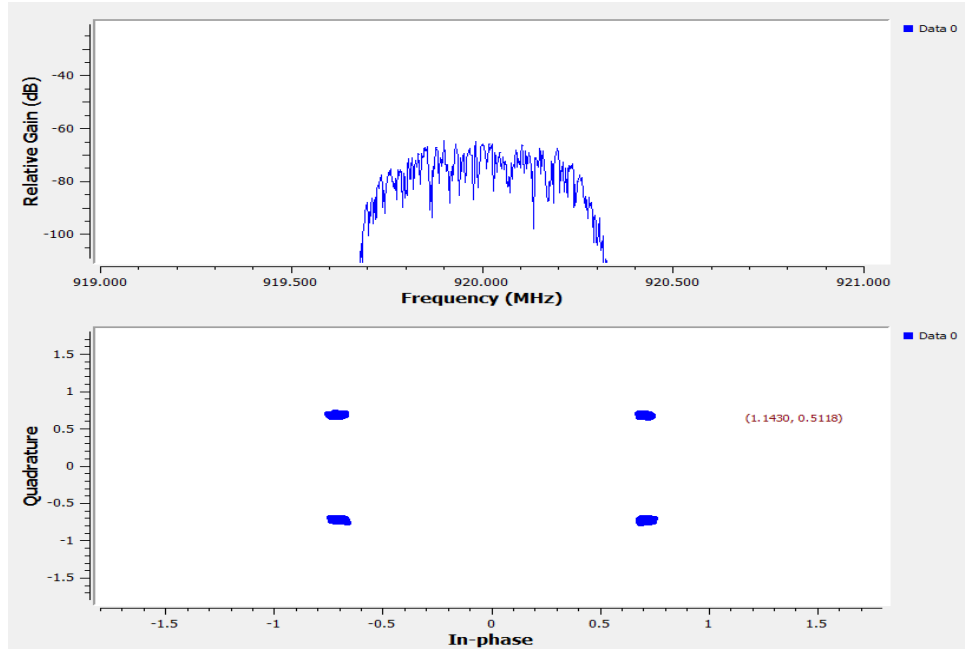


Figura 3.4: Diagrama de constelación y espectro de la señal recibida QPSK.

3.3.4. Sistema de control adaptativo de potencia transmitida para la modulación BPSK.

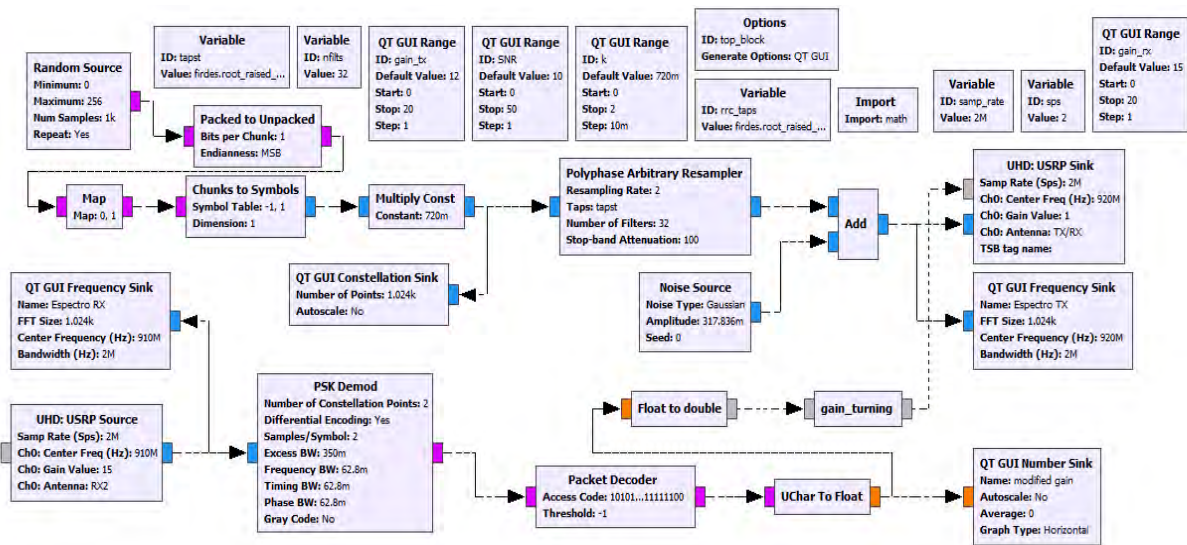


Figura 3.5: Diagrama de bloque del nodo sensor para modulación BPSK.

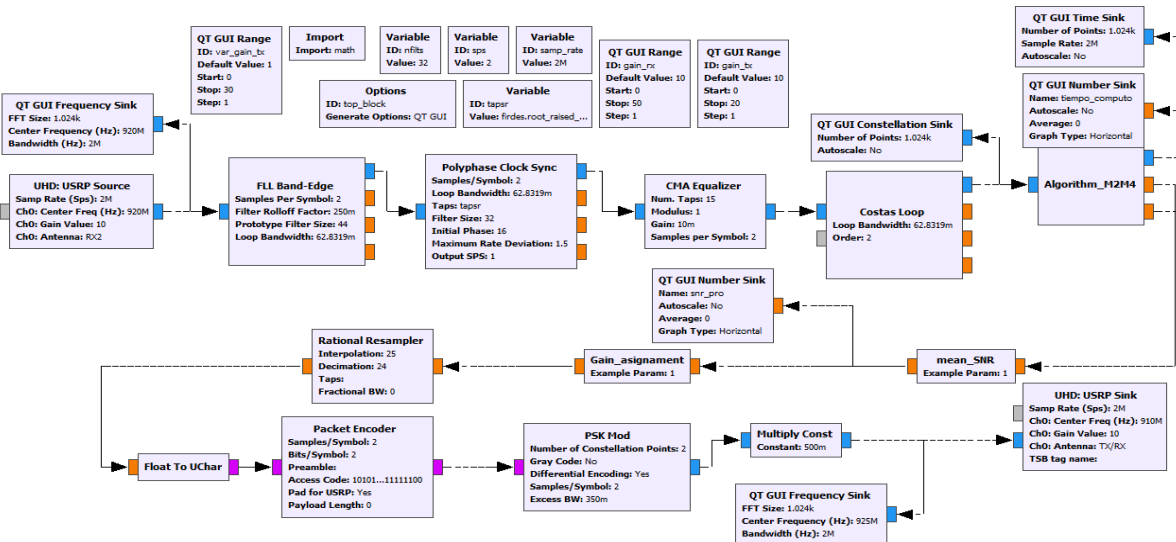


Figura 3.6: Diagrama de bloque de la estación base para modulación BPSK