

**IMPLEMENTACIÓN DE UN ENTRENADOR DE RADIO DEFINIDO POR
SOFTWARE PARA REALIZAR PRÁCTICAS EN EL LABORATORIO DE
COMUNICACIONES DEL PROGRAMA DE INGENIERÍA ELECTRÓNICA DE LA
UNIVERSIDAD POPULAR DEL CESAR**

**DIEGO ANDRÉS OÑATE VEGA
RODRIGO ALBERTO SUAREZ CAÑATE**

**UNIVERSIDAD POPULAR DEL CESAR
FACULTAD DE INGENIERÍAS Y TECNOLÓGICAS
PROGRAMA DE INGENIERÍA ELECTRÓNICA
VALLEDUPAR – CESAR**

2025

**IMPLEMENTACIÓN DE UN ENTRENADOR DE RADIO DEFINIDO POR
SOFTWARE PARA REALIZAR PRÁCTICAS EN EL LABORATORIO DE
COMUNICACIONES DEL PROGRAMA DE INGENIERÍA ELECTRÓNICA DE LA
UNIVERSIDAD POPULAR DEL CESAR**

**DIEGO ANDRÉS OÑATE VEGA
RODRIGO ALBERTO SUÁREZ CAÑATE**

**TRABAJO DE GRADO PRESENTADO COMO REQUISITO PARA OPTAR POR
EL TÍTULO DE INGENIERO ELECTRÓNICO**

**DIRECTOR
ING. JORGE ELIECER MOLINA CERVANTES**

**UNIVERSIDAD POPULAR DEL CESAR
FACULTAD DE INGENIERÍAS Y TECNOLÓGICAS
PROGRAMA DE INGENIERÍA ELECTRÓNICA
VALLEDUPAR – CESAR**

2025

Nota de aceptación

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Valledupar DD/MM/AAAA

Dedicatoria

En esta ocasión quiero comenzar agradeciendo a Dios, por darme la sabiduría y fortaleza necesaria para afrontar cada reto y obtener este logro; a mi padre Rodrigo Suarez, a mi madre Libis Cañate y a mi hermana Katy Suarez, por su amor incondicional y apoyo constante, su fe en mí y su aliento inquebrantable han sido fundamentales en cada paso del camino. A mis familiares y amigos por su presencia y respaldo a lo largo de este proceso. Este logro es tanto suyo como mío, por lo que, estoy sumamente agradecido con cada uno de ustedes.

Rodrigo Alberto Suarez Cañate

Quisiera expresar mis sinceros agradecimientos a mi madre, Eleida Luz Vega Montaña, quien siempre me ha apoyado y aconsejado incondicionalmente en todas las etapas de la vida. A mis amigos, compañeros y, hoy en día, colegas de profesión, que me acompañaron en todo este recorrido académico en las aulas de la Universidad Popular del Cesar. A ellos dedico este trabajo.

Diego Andrés Oñate Vega

Agradecimientos

Quisiéramos iniciar expresando nuestro más grande agradecimiento a Dios, por otorgarnos la sabiduría y fortaleza necesaria para sobreponernos a cada reto y lograr finalizar este proyecto.

A la Universidad Popular del Cesar, por brindarnos los recursos y entorno académico necesario para llevar a cabo el desarrollo de este proyecto.

A los docentes e ingenieros, que con su dedicación y enseñanza han enriquecido nuestro proceso de aprendizaje, preparándonos para afrontar los retos que trae consigo esta investigación.

Un agradecimiento especial al ingeniero Jorge Eliecer Molina, por su orientación y apoyo constante, que han sido invaluable para el éxito de este proyecto.

ÍNDICE

Dedicatoria	4
Agradecimientos	5
ÍNDICE DE FIGURAS	10
ÍNDICE DE TABLAS.....	20
RESUMEN	21
ABSTRACT	21
1. INTRODUCCIÓN	23
2. ANTECEDENTES.....	24
2.1. ÁMBITO LOCAL Y REGIONAL	24
2.2. ÁMBITO INTERNACIONAL.....	24
3. LÍNEA Y TEMA DE INVESTIGACIÓN	26
4. FORMULACIÓN DEL PROBLEMA.....	27
5. JUSTIFICACIÓN.....	28
6. OBJETIVOS.....	29
6.1. OBJETIVO GENERAL.....	29
6.2. OBJETIVOS ESPECÍFICOS.....	29
7. MARCO TEÓRICO	30
7.1. Espectro electromagnético.....	30
7.1.1. Espectro radioeléctrico	30
7.1.2. Regulación del espectro radioeléctrico en el Colombia	30
7.2. Sistema básico de comunicaciones	30
7.3. Teorema de Nyquist	31
7.4. Modulación	31
7.5. Técnicas de modulación	32
7.5.1. Modulación de amplitud (AM).....	32
7.5.2. Modulación de frecuencia (FM).....	33
7.5.3. Modulación por desplazamiento de amplitud (ASK).....	34
7.5.3.1. Transmisión en ASK.....	34
7.5.3.2. Ancho banda en ASK	35
7.5.4. Modulación por desplazamiento de frecuencia (FSK).....	35
7.5.4.1. Transmisión en FSK	36

7.5.4.2.	Ancho de banda en FSK	36
7.5.4.3.	Espectro de frecuencias en FSK	36
7.5.4.4.	Recepción en FSK	37
7.6.	Diagrama de constelación	37
7.7.	Modulación por desplazamiento de fase (PSK)	37
7.7.1.	Modulación por desplazamiento binario de fase (BPSK)	38
7.7.1.1.	Transmisión en BPSK	38
7.7.1.2.	Ancho de banda en BPSK	39
7.7.1.3.	Recepción en BPSK	39
7.7.1.4.	Diagrama de constelación en BPSK	39
7.8.	Modulación por desplazamiento de amplitud en cuadratura (QAM)	40
7.9.	Sistema de comunicación SDR	41
7.9.1.	Definición	41
7.9.2.	Esquema básico de SDR	41
7.10.	Hardware SDR disponibles en el mercado	42
7.10.1.	Dongle RTL-SDR Blog V3	43
7.10.2.	Nooelec NESDR SMARt v5	43
7.10.3.	HackRF One	43
7.10.4.	PlutoSDR	43
7.11.	Software SDR	44
7.11.1.	GNU radio	44
7.11.2.	Matlab y Simulink	44
8.	DISEÑO METODOLÓGICO	46
8.1.	Requerimientos del hardware SDR	46
8.2.	Selección del hardware SDR	46
8.3.	Instalación de drivers para reconocimiento del hardware SDR	48
8.4.	Requerimientos de software SDR	49
8.5.	Comparativa y selección del software SDR	50
8.6.	Instalación de GNU Radio	50
8.7.	Desarrollo e implementación de esquemas moduladores y demoduladores en GNU Radio	53
8.7.1.	Modulación de Amplitud (AM)	53
8.7.1.1.	Desarrollo del esquema modulador en GNU Radio	53

8.7.1.2.	Desarrollo del esquema demodulador en GNU Radio.....	61
8.7.1.3.	Implementación del modulador en GNU Radio usando los dispositivos SDR	64
8.7.1.4.	Implementación del demodulador en GNU Radio usando los dispositivos SDR.....	66
8.7.2.	Modulación de frecuencia (FM).....	67
8.7.2.1.	Desarrollo del esquema modulador en GNU Radio.....	67
8.7.2.2.	Desarrollo del esquema demodulador en GNU Radio.....	70
8.7.2.3.	Implementación del modulador en GNU Radio usando los dispositivos SDR	72
8.7.2.4.	Implementación del demodulador en GNU Radio usando los dispositivos SDR.....	73
8.7.3.	Modulación por desplazamiento de amplitud (ASK).....	74
8.7.3.1.	Desarrollo del esquema modulador en GNU Radio.....	74
8.7.3.2.	Desarrollo del esquema demodulador en GNU Radio.....	81
8.7.3.3.	Implementación de modulador en GNU Radio usando los dispositivos SDR	85
8.7.3.4.	Implementación de demodulador en GNU Radio usando los dispositivos SDR.....	86
8.7.4.	Modulación por desplazamiento de frecuencia (FSK).....	87
8.7.4.1.	Desarrollo del esquema modulador en GNU Radio.....	87
8.7.4.2.	Desarrollo del esquema demodulador en GNU Radio.....	92
8.7.4.3.	Implementación del esquema modulador en GNU Radio usando los dispositivos SDR.....	94
8.7.4.4.	Implementación del esquema demodulador en GNU Radio usando los dispositivos.....	95
8.7.5.	Modulación por desplazamiento de fase binaria (BPSK).....	95
8.7.5.1.	Desarrollo del esquema modulador en GNU Radio.....	96
8.7.5.2.	Desarrollo del esquema demodulador en GNU Radio.....	99
8.7.5.3.	Implementación del esquema modulador en GNU Radio usando los dispositivos SDR.....	100
8.7.5.4.	Implementación del esquema demodulador en GNU Radio usando los dispositivos SDR.....	101
8.7.6.	Modulación de amplitud en cuadratura (QAM).....	101
8.7.6.1.	Desarrollo del esquema modulador en GNU Radio.....	101
8.7.6.2.	Desarrollo del esquema demodulador en GNU Radio.....	108

8.7.6.3. Implementación de modulador en GNU Radio usando los dispositivos SDR	111
8.7.6.4. Implementación de demodulador en GNU Radio usando los dispositivos SDR.....	112
9. RESULTADOS.....	114
10. CONCLUSIONES	130
11. REFERENCIAS BIBLIOGRÁFICAS	156

ÍNDICE DE FIGURAS

Figura 1. Espectro electromagnético de frecuencias. Fuente: Tomasi, 2003.	30
Figura 2. Sistema de comunicación básico. Fuente: Basado en [10].....	31
Figura 3. Esquema básico de un sistema modulador. Fuente: Basado en Tomasi, 2003.	32
Figura 4. Esquema de bloques de un sistema de comunicación. Fuente: Basado en Tomasi, 2003.	32
Figura 5. Modulación AM DSBFC (Doble banda lateral con full portadora). Fuente: Tomasi, 2003.	33
Figura 6. Espectro en frecuencia para señal modulada en AM. Fuente: Tomasi, 2003.	33
Figura 7. Modulación FM de una señal portadora senoidal por una señal de onda senoidal: (a) Portadora no modulada; (b) Señal moduladora; (c) Señal modulada con frecuencia. Fuente: Tomasi, 2003.	33
Figura 8. Modulación OOK: (a) Señal binaria de información; (b): Señal modulada ASK. Fuente: Tomasi, 2003.	34
Figura 9. Modulador ASK. Fuente: J. Luque Rodríguez y S. Clavijo Suero, Modulación de señales digitales.	34
Figura 10. Espectro en frecuencia de ASK. Fuente: Basado en Tomasi, 2003.	35
Figura 11. Desviación de frecuencia en FSK. Fuente: Tomasi, 2003.	35
Figura 12. Señal de entrada y salida en la modulación FSK. Fuente: Tomasi, 2003.	36
Figura 13. Esquema simplificado de un modulador FSK. Fuente: Basado en Tomasi, 2003.	36
Figura 14. Espectro de frecuencias en FSK. Fuente: Tomasi, 2003.	37
Figura 15. Demodulador PLL-FSK. Fuente: Basado en Tomasi, 2003.	37
Figura 16. Señal de salida de un modulador BPSK en función de una señal de entrada binaria normalizada. Fuente: Tomasi, 2003.	38
Figura 17. Diagrama simplificado de un modulador BPSK balanceado. Fuente: Basado en Tomasi, 2003.	39
Figura 18. Esquema de bloques simplificado de un receptor BPSK. Fuente: Basado en Tomasi, 2003.	39
Figura 19. Diagrama de constelación. Fuente: Tomasi, 2003.	40
Figura 20. Esquema básico de un modulador NQAM. Fuente: Basado en DANIZIO, P. E. Sistemas de comunicaciones.	40
Figura 21. Diagrama de constelación de la modulación 8-QAM. Fuente: DANIZIO, P. E. Sistemas de comunicaciones.	40
Figura 22. Esquema básico de bloques funcionales de SDR. Fuente: Basado en [15].	42
Figura 23. Receptor Dongle RTL-SDR Blog V3. Fuente: Amazon.	43
Figura 24. Receptor Nooelec NESDR SMARt v5. Fuente: Amazon.	43
Figura 25. Transceptor HackRF One. Fuente: Aliexpress.	43
Figura 26. Transceptor PlutoSDR. Fuente: Amazon.	43
Figura 27. Interfaz gráfica de GNU Radio. Fuente: [21].	44

Figura 28. Interfaz gráfica de Matlab y Simulink. Fuente propia.	45
Figura 29. Detección de dispositivos SDR conectados al sistema. Fuente: propia.	48
Figura 30. Pantalla de inicio del sitio web oficial de Zadig. Fuente:	48
Figura 31. Sección de descarga en sitio web de Zadig. Fuente:	49
Figura 32. Pestaña options y actualización de todos los dispositivos conectados en Zadig. Fuente: propia.	49
Figura 33. Proceso de instalación de Driver para hardware de SDR. Fuente propia	49
Figura 34. Sección destinada a la descarga del archivo de instalación de GNU Radio para Windows. Fuente:	51
Figura 35. Sección de descarga en repositorio de GitHub. Fuente: [23].	51
Figura 36. Pantalla inicial del asistente de instalación de GNU Radio y selección de la opción “Next” para continuar con el proceso. Es recomendable cerrar aplicaciones no relacionadas para evitar reinicios del sistema. Fuente propia.....	52
Figura 37. Pantalla del asistente donde fueron aceptados los términos y condiciones de GNU Radio al hacer clic en “I Agree”. Fuente propia	52
Figura 38. Pantalla de selección del tipo de instalación en la que se seleccionó la opción recomendada y se hizo clic en “Next”. Fuente propia	52
Figura 39. Pantalla de selección de lugar de instalación, en la cual se dejó el valor predeterminado y se hizo clic en “Install”. Fuente propia.	53
Figura 40. Esquema básico de modulador AM. Fuente propia	53
Figura 41. Bloques Options y Variable con sus configuraciones en GNU Radio. Fuente propia.....	54
Figura 42. Ancho de banda para modulador AM. Fuente propia.....	54
Figura 43. Panel de búsqueda en GNU Radio. Fuente propia	55
Figura 44. Bloques Signal Source y sus configuraciones para AM en GNU Radio (señal moduladora y portadora) en un entorno simulado. Fuente propia	55
Figura 45. Bloques QT GUI Dial y QT GUI Level Gauge y sus configuraciones para la variación y visualización de la frecuencia de moduladora para AM en GNU Radio en un entorno simulado. Fuente propia	56
Figura 46. Bloques QT GUI Dial y QT GUI Level Gauge y sus configuraciones para la variación y visualización de la frecuencia de portadora para AM en GNU Radio en un entorno simulado. Fuente propia.	56
Figura 47. Bloque QT GUI Dial y QT GUI Level Gauge para la variación del índice de modulación en AM en GNU Radio en un entorno simulado. Fuente propia.	56
Figura 48. Configuración del bloque QT GUI Dial para la variación del índice de modulación en AM en GNU Radio. Fuente propia.	57
Figura 49. Configuración del bloque QT GUI Level Gauge para la variación del índice de modulación en AM en GNU Radio. Fuente propia.....	57
Figura 50. Bloque Multiply Const y su configuración para AM en GNU Radio en un entorno simulado. Fuente propia.	57
Figura 51. Bloque Add Const para AM en GNU Radio en un entorno simulado. Fuente propia.....	57

Figura 52. Bloque Multiply para AM en GNU Radio en un entorno simulado. Fuente propia	58
Figura 53. Bloque Float To Complex y Constant Source para AM en GNU Radio en un entorno simulado. Fuente propia	58
Figura 54. Bloque Rational Resampler y su configuración para AM en GNU Radio en un entorno simulado. Fuente propia.....	58
Figura 55. Bosquejo de interfaz de modulador. Fuente propia.....	59
Figura 56. Bloques QT GUI Tab Widget y su configuración para la sección de visualización de la señal moduladora en GNU Radio. Fuente propia	59
Figura 57. Bloques QT GUI Tab Widget y su configuración para la sección de visualización de la señal modulada en GNU Radio. Fuente propia	59
Figura 58. Bloque QT GUI Tab Widget para la sección de configuración de parametros de modulación en GNU Radio. Fuente propia.....	59
Figura 59. Configuración bloque QT GUI Tab Widget para la sección de configuración de parámetros de modulación en GNU Radio. Fuente propia.....	60
Figura 60. Espectro en frecuencia de señal modulada en AM en GNU Radio en un entorno simulado. Fuente propia	60
Figura 61. Esquema final de modulador AM para implementación simulada en GNU Radio. Fuente propia.....	60
Figura 62. Configuración de los bloques QT GUI Time Sink y QT GUI Frequency Sink para AM en GNU Radio. Fuente propia	60
Figura 63. Esquema de receptor AM por detección de envolvente. Fuente: Basado en Tomasi, 2003.....	61
Figura 64. Bloque Complex To Float para demodulación AM en GNU Radio en un entorno simulado. Fuente propia.	61
Figura 65. Bloque Rail y su configuración para AM en GNU Radio. Fuente propia.	61
Figura 66. Bloque Rational Resampler y su configuración para AM en GNU Radio. Fuente propia.....	62
Figura 67. Bloque Low Pass Filter y su configuración para AM en GNU Radio. Fuente Propia.	62
Figura 68. Bloques QT GUI Dial y QT GUI Level Gauge para la variación de la frecuencia de corte en la demodulación AM en GNU Radio en un entorno simulado. Fuente propia.....	63
Figura 69. Bosquejo de interfaz de demodulador. Fuente propia.....	63
Figura 70. Bloques QT GUI Widget y su configuración para la visualización de la señal modulada en AM en el dominio del tiempo en GNU Radio. Fuente propia..	63
Figura 71. Bloques QT GUI Widget y su configuración para la visualización de la señal demodulada en AM en el dominio del tiempo en GNU Radio. Fuente propia.	64
Figura 72. Esquema final de demodulador AM en GNU Radio en un entorno simulado. Fuente propia.....	64

Figura 73. Configuración de los bloques QT GUI Time Sink para la visualización en el dominio del tiempo de la señal demodulada en AM (izquierda) y la señal modulada en AM (derecha) en GNU Radio. Fuente propia.....	64
Figura 74. Bloque osm ocom Sink y su configuración para AM en GNU Radio. Fuente propia.....	66
Figura 75. Esquema final de modulador AM en GNU Radio usando SDR. Fuente propia.....	66
Figura 76. Bloque Osm ocom Source y su configuración para AM en GNU Radio. Fuente propia.....	67
Figura 77. Bloque Complex To Float para AM en GNU Radio. Fuente propia	67
Figura 78. Esquema final de demodulador AM en GNU Radio usando SDR. Fuente propia.....	67
Figura 79. Bloque Signal Source y su configuración utilizada para generar la señal moduladora en FM. Fuente propia.....	68
Figura 80. Bloques QT GUI Dial y QT GUI Level Gauge – Ajuste de frecuencia moduladora. Fuente propia	68
Figura 81. Configuración de parámetros de los bloques QT GUI Dial y QT GUI Level Gauge – Frecuencia moduladora. Fuente propia.....	68
Figura 82. Configuración del bloque VCO(Complex), la variable vco_sens almacena el valor de k. Fuente propia.....	69
Figura 83. Esquema modulador FM. Fuente propia	69
Figura 84. Señal modulada en FM con el VCO(Complex) Vs Señal moduladora. Fuente propia.....	70
Figura 85. Espectro de frecuencia de la modulación FM. Fuente propia.....	70
Figura 86. Bloque Quadrature Demod y su configuración para FM. Fuente propia	71
Figura 87. Filtro FTT Low Pass Filter y su configuración para FM en GNU Radio. Fuente propia.....	71
Figura 88. Bloques QT GUI Dial y QT GUI Level Gauge y su configuración para ajuste de frecuencia de corte del filtro en FM. Fuente propia.....	72
Figura 89. Esquema final de demodulador FM en GNU Radio. Fuente propia.	72
Figura 90. Bloque osmocom Sink y su configuración para transmisor FM. Fuente propia.	73
Figura 91. Esquema modulador FM con bloque transmisor osmocom Sink. Fuente propia.	73
Figura 92. Bloque osmocom Source y su configuración para receptor FM. Fuente propia.	74
Figura 93. Esquema final de demodulador FM con bloque receptor osmocom Source en GNU Radio. Fuente propia.....	74
Figura 94. Esquema básico de un modulador ASK. Fuente: Basado en Tomasi, 2003.	75
Figura 95. Bloque Vector Source para OOK en GNU Radio en un entorno simulado. Fuente: propia.....	75
Figura 96. Ancho de banda para ASK. Fuente propia	76

Figura 97. Bloque Vector Source y Repeat para OOK en GNU Radio en un entorno simulado. Fuente propia.....	76
Figura 98. Señal moduladora configurada con una duración de bit de 25us. Fuente propia	76
Figura 99. Configuración del bloque Repeat para OOK en GNU Radio en un entorno simulado. Fuente propia.....	76
Figura 100. Bloque QT GUI Tab Widget y su configuración para la sección de configuración de parámetros en OOK en GNU Radio en un entorno simulado. Fuente propia.....	77
Figura 101. Bloques QT GUI Dial y QT GUI Level Gauge y su configuración para la variación y visualización de la frecuencia de moduladora en OOK en GNU Radio en un entorno simulado. Fuente propia.	77
Figura 102. Conjunto de bloques de conversión de punto flotante a complejo (Float To Complex y Constant Source). Fuente propia.	77
Figura 103. Bloques QT GUI Dial y QT GUI Level Gauge para OOK en GNU Radio en un entorno simulado. Fuente propia.	78
Figura 104. Conexión del bloque Float To Complex con el bloque Virtual Sink para OOK en GNU Radio en un entorno simulado. Fuente propia.....	78
Figura 105. Multiplicación de la señal portadora con la moduladora adecuada para OOK en GNU Radio en un entorno simulado. Fuente propia.....	78
Figura 106. Bloques QT GUI Frequency y QT GUI Time Sink para visualización de señal moduladora en OOK en GNU Radio. Fuente propia.....	79
Figura 107. Configuración bloques QT GUI Frequency y QT GUI Time Sink para visualización de señal moduladora en OOK en GNU Radio. Fuente propia.....	79
Figura 108. Conjunto de bloque para visualización de señales en Transmisor OOK (QT GUI Time Sink, Virtual Sink y Virtual Source). Fuente propia	79
Figura 109. Configuración del bloque QT GUI Time Sink para visualización de señales OOK en GNU Radio. Fuente propia.....	80
Figura 110. Bloques QT GUI Widget para la visualización de la señal moduladora en OOK en GNU Radio en un entorno simulado. Fuente propia.....	80
Figura 111. Bloques QT GUI Widget para la visualización de la señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.....	80
Figura 112. Esquema de modulador ASK para entorno de simulación en GNU Radio. Fuente propia.....	81
Figura 113. Espectro en frecuencia de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.	81
Figura 114. Conexión entre bloque el bloque Complex To Float y Rail. Fuente propia	82
Figura 115. Bloque Low Pass Filter y su configuración en GNU Radio configurado en demodulador OOK en un entorno simulado. Fuente propia	82
Figura 116. Bloques QT GUI Dial y QT GUI Level Gauge y su configuración para variación y visualización de frecuencia de corte en OOK en GNU Radio en un entorno simulado. Fuente propia	82

Figura 117. Bloque Threshold y su configuración para OOK en un entorno simulado en GNU Radio. Fuente propia.....	83
Figura 118. Bloque QT GUI Time Sink y su configuración para visualización de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.	83
Figura 119. Bloque QT GUI Time Sink y su configuración para visualización de señal modulada vs demodulada en OOK en GNU Radio en un entorno simulado. Fuente propia.	84
Figura 120. Bloques QT GUI Tab Widget para visualización de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.....	84
Figura 121. Configuración de bloques QT GUI Tab Widget para visualización de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.	84
Figura 122. Bloques QT GUI Tab Widget y su configuración para visualización de señal demodulada en OOK en GNU Radio en un entorno simulado. Fuente propia.	84
Figura 123. Esquema final del demodulador OOK en un entorno de simulación en GNU Radio. Fuente propia.....	85
Figura 124. Configuración de bloque Osm com Sink y su configuración para modulador OOK en un entorno practico en GNU Radio. Fuente propia.	85
Figura 125. Esquema de modulador OOK utilizando hardware SDR en un entorno real en GNU Radio. Fuente propia.	86
Figura 126. Bloque Osm com Source y su configuración para modulación OOK en un entorno practico en GNU Radio. Fuente propia.....	86
Figura 127. Esquema demodulador OOK utilizando hardware SDR en un entorno practico en GNU Radio. Fuente propia.....	87
Figura 128. Señal generada por el bloque Vector Source. Fuente propia.	87
Figura 129. Señal moduladora a la salida del bloque Repeat. Fuente propia.	88
Figura 130. Señal moduladora con ajustes de tiempo. Fuente propia.	88
Figura 131. Señal moduladora multiplicada por la fm. Fuente propia.	88
Figura 132. Señal resultante de la multiplicación de la señal moduladora por la fm. Fuente propia.....	89
Figura 133. Bloques QT GUI Dial y QT GUI Level Gauge para ajustar la fm. Fuente propia.	89
Figura 134. Ajuste de parámetros para los bloques QT GUI Dial y QT GUI Level Gauge para la fm. Fuente propia.	89
Figura 135. Señal moduladora multiplicada por la fs. Fuente propia.	90
Figura 136. Señal resultante de la multiplicación de la señal moduladora por la fs. Fuente propia.....	90
Figura 137. Bloques QT GUI Dial y QT GUI Level Gauge para ajustar la fs. Fuente propia.	90
Figura 138. Ajuste de parámetros para los bloques QT GUI Dial y QT GUI Level Gauge para la fs. Fuente propia.	90
Figura 139. Esquema modulador FSK. Fuente propia.....	91
Figura 140. Señal modulada FSK. Fuente propia.	91
Figura 141. Espectro de frecuencias para la modulación FSK. Fuente propia	92

Figura 142. Bloque Band Pass Filter y configuración. Fuente propia.	92
Figura 143. Señal modulada a la salida del filtro pasa banda para fm. Fuente propia.	93
Figura 144. Señal rectificada para demodulador de la fm. Fuente propia.	93
Figura 145. Señal modulada a la salida del filtro pasa banda para fs. Fuente propia.	93
Figura 146. Señal rectificada para el demodulador de la fs. Fuente propia.....	94
Figura 147. Esquema demodulador de FSK. Fuente propia.	94
Figura 148. Esquema modulador FSK con el bloque trasmisor osmocomb Sink. Fuente propia.....	95
Figura 149. Esquema demodulador FSK con el bloque receptor osmocomb Source. Fuente propia.....	95
Figura 150. Configuración de bloques generar señal moduladora para BPSK. Fuente propia.....	96
Figura 151. Señal moduladora generada para BPSK. Fuente propia.....	96
Figura 152. Configuración de bloques para generar la inversión de la señal modulada. Fuente propia	97
Figura 153. Señal modulada invertida. Fuente propia	97
Figura 154. Configuración de bloques para transformar los valores 1 en -1. Fuente propia.	97
Figura 155. Señal transformada con magnitud de -1. Fuente propia.....	97
Figura 156. Señal moduladora original sumada a la resultante de las operaciones. Fuente propia.....	98
Figura 157. Esquema modulador de BPSK. Fuente propia.....	98
Figura 158. Espectro de frecuencias BPSK. Fuente propia	98
Figura 159. Bloque Costas Loop para realizar sincronización de fase y frecuencia. Fuente propia.....	99
Figura 160. Esquema del detector de envolvente. Fuente propia.	99
Figura 161. Señal rectificada. Fuente propia.....	100
Figura 162. Esquema final de demodular BPSK en GNU Radio. Fuente propia.	100
Figura 163. Esquema modulador BPSK conectado al transmisor a través del bloque osmocon Sink. Fuente propia.	100
Figura 164. Esquema demodulador BPSK conectado al receptor mediante el bloque osmocomb Source. Fuente propia.....	101
Figura 165. Bloque Vector Source y su configuración para 4QAM en GNU Radio en un entorno simulado. Fuente propia	102
Figura 166. Bloque Stream Demux y su configuración para 4QAM en GNU Radio en un entorno simulado. Fuente propia	102
Figura 167. Conexión entre el bloque Stream Demux y los bloques Chunks To Symbols junto con su configuración para 4QAM en GNU Radio en un entorno de simulación. Fuente propia	103
Figura 168. Conexión entre los bloques Chunks to Symbols y Repeat para 4QAM en GNU Radio en un entorno de simulación. Fuente propia	103

Figura 169. Configuración de los bloques Repeat para 4QAM en GNU Radio en un entorno simulado. Fuente propia	104
Figura 170. Duración de bit de señal moduladora para 4QAM en GNU Radio en un entorno simulado. Fuente propia	104
Figura 171. Bloque QT GUI Tab Widget para sección de configuración de configuración de parámetros en 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	104
Figura 172. Bloques QT GUI Dial y QT GUI Level Gauge para variación y visualización de la señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	104
Figura 173. Configuración de bloques QT GUI Dial y QT GUI Level Gauge para variación y visualización de la señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia	105
Figura 174. Bloques Signal Source y Rational Resampler y su configuración para generar y re muestrear señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	105
Figura 175. Conexiones entre los canales I, Q y la señal portadora en 4QAM en GNU Radio en un entorno simulado. Fuente propia	105
Figura 176. Bloques QT GUI Tab Widget y su configuración para la sección de visualización de señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	106
Figura 177. Conexión de bloque QT GUI Frequency Sink y QT GUI Time Sink para visualización en el dominio del tiempo y espectro en frecuencia en 4QAM en GNU Radio en un entorno simulado. Fuente propia	106
Figura 178. Conexión de bloques para visualización de espectro en frecuencia y dominio del tiempo de la señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	106
Figura 179. Configuración de bloques Rational Resampler y QT GUI Constellation Sink para visualización del diagrama de constelación en 4QAM en GNU Radio en un entorno simulado. Fuente propia	107
Figura 180. Espectro en frecuencia de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia	107
Figura 181. Diagrama de constelación de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia	107
Figura 182. Esquema de modulador 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	108
Figura 183. Bloque Costas Loop y su configuración para demodulación de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia	108
Figura 184. Conexión del bloque Costas Loop y Moving Average para demodulación 4QAM en GNU Radio en un entorno simulado. Fuente propia	109
Figura 185. bloque Complex To Float para demodulación 4QAM en GNU Radio en un entorno simulado. Fuente propia	109
Figura 186. Bloques Threshold para adecuación de señal para demodulación en 4QAM en un entorno simulado en GNU Radio. Fuente propia	109

Figura 187. Conexión de los bloques Integrate y Threshold para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	109
Figura 188. Configuración de los bloques Integrate y Threshold para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia	110
Figura 189. Bloque Stream Mux para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia	110
Figura 190. Bloque Repeat y su configuración para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia	110
Figura 191. Bloques QT GUI Tab Widget y su configuración para visualización de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia	110
Figura 192. Bloques QT GUI Tab Widget y su configuración para visualización de señal demodulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia	111
Figura 193. Configuración de los bloques QT GUI Time Sink para visualización de la señal modulada y de demodulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	111
Figura 194. Esquema final del demodulador 4QAM en GNU Radio en un entorno simulado. Fuente propia.....	111
Figura 195. Bloque osm ocom sink y su configuración para 4QAM en un entorno practico en GNU Radio. Fuente propia.....	112
Figura 196. Bloque Osm ocom Source y su configuración para transmisión de señal 4QAM en un entorno practico en GNU Radio. Fuente propia	113
Figura 197. Esquema final de demodular 4QAM en GNU Radio en un entorno real. Fuente propia.....	113
Figura 198. Interfaz gráfica de transmisor AM. Fuente propia	115
Figura 199. Interfaz gráfica de receptor AM. Fuente propia	115
Figura 200. Prueba realizada de modulador AM. Fuente propia.....	116
Figura 201. Prueba realizada de demodulador AM. Fuente propia.....	116
Figura 202. Interfaz gráfica de modulador FM. Fuente propia	117
Figura 203. Interfaz gráfica de demodulador FM. Fuente propia	117
Figura 204. Prueba realizada de modulador FM. Fuente propia	118
Figura 205. Prueba realizada de demodulador FM. Fuente propia.....	119
Figura 206. Interfaz gráfica de modulador OOK. Fuente propia.....	119
Figura 207. Interfaz gráfica de demodulador OOK. Fuente propia	120
Figura 208. Prueba realizada de modulador OOK. Fuente propia	121
Figura 209. Prueba realizada de demodulador OOK. Fuente propia	121
Figura 210. Interfaz gráfica de modulador FSK. Fuente propia.....	122
Figura 211. Interfaz gráfica de demodulador FSK. Fuente propia	122
Figura 212. Prueba realizada de modulador FSK. Fuente propia	123
Figura 213. Prueba realizada de demodulador FSK. Fuente propia	124
Figura 214. Interfaz gráfica de modulador BPSK. Fuente propia	124
Figura 215. Interfaz gráfica de demodulador BPSK. Fuente propia	125
Figura 216. Prueba realizada de modulador BPSK. Fuente propia	125

Figura 217. Prueba realizada de demodulador BPSK. Fuente propia	126
Figura 218. Interfaz gráfica de modulador 4QAM. Fuente propia	126
Figura 219. Interfaz gráfica de demodulador 4QAM. Fuente propia	127
Figura 220. Prueba realizada de modulador 4QAM. Fuente propia	127
Figura 221. Prueba realizada de demodulador 4QAM. Fuente propia	128
Figura 222. Vista del Entrenador de Radio Definido por Software con hardware SDR integrado en maletín portátil. Fuente propia.....	128
Figura 223. Vista del Entrenador de Radio Definido por Software con hardware SDR integrado en maletín portátil. Fuente propia.....	129
Figura 224. Vista del Entrenador de Radio Definido por Software con hardware SDR integrado en maletín portátil. Fuente propia.....	129

ÍNDICE DE TABLAS

Tabla 1. Cuadro comparativo de las características de los dispositivos SDR.	47
Tabla 2. Cuadro comparativo de las características de los softwares para SDR. .	50

RESUMEN

En el panorama actual, las comunicaciones resultan ser de vital importancia, haciéndose imprescindible que los ingenieros comprendan su funcionamiento, puesto que son quienes deben mostrarse resilientes a los desafíos de una era interconectada. En este contexto, el presente proyecto se enfoca en la implementación de un entrenador de Radio Definido por Software (SDR) para realizar prácticas en el laboratorio de comunicaciones de la Universidad Popular del Cesar. Para esto, se inició con una evaluación exhaustiva de las distintas herramientas computacionales para la configuración de dispositivos SDR, seguidamente el modelado y análisis de las técnicas de modulación AM, FM, ASK, FSK, PSK y QAM en un ambiente simulado, evaluando las capacidades y limitaciones de cada herramienta, la facilidad de uso y compatibilidad con los dispositivos SDR del mercado actual. Posteriormente, fueron implementadas estas simulaciones en un entorno real utilizando dispositivos SDR, verificando la precisión y el rendimiento de las herramientas en condiciones de operación real.

Los resultados mostraron que las herramientas analizadas ofrecen características destacables, sin embargo, GNU Radio destaca en términos de flexibilidad, facilidad de uso y compatibilidad con diversas modulaciones. Se desarrollaron guías prácticas de laboratorio que proporcionan instrucciones detalladas para la configuración y utilización de los dispositivos SDR, así como para la aplicación de las técnicas de modulación mencionadas, proporcionar así una formación práctica, en donde puedan aplicar los conocimientos teóricos y mejorar sus habilidades y destrezas en el manejo de este tipo de sistemas de comunicación.

Palabras clave: SDR, modulaciones, AM, FM, ASK, FSK, PSK, QPSK, GNU Radio.

ABSTRACT

In the current scenario, communications are of vital importance, making it essential for engineers to understand how they work, since they are the ones who must be resilient to the challenges of an interconnected era. In this context, this project focuses on the implementation of a Software Defined Radio (SDR) trainer to carry out practices in the communications laboratory of the Universidad Popular del Cesar. To do this, it began with an exhaustive evaluation of the different computational tools for the configuration of SDR devices, followed by the modeling and analysis of the AM, FM, ASK, FSK, PSK and QAM modulation techniques in a simulated environment, evaluating the capabilities and limitations of each tool, the ease of use and compatibility with the SDR devices on the market today. Subsequently, these simulations were implemented in a real environment using SDR devices, verifying the accuracy and performance of the tools under real operating conditions.

The results showed that the tools analyzed offer notable features, however, GNU Radio stands out in terms of flexibility, ease of use and compatibility with various modulations. Practical laboratory guides were developed that provide detailed

instructions for the configuration and use of SDR devices, as well as for the application of the mentioned modulation techniques, thus providing practical training, where they can apply theoretical knowledge and improve their skills and abilities in the management of this type of communication systems.

Keywords: SDR, modulations, AM, FM, ASK, FSK, PSK, QPSK, GNU Radio.

1. INTRODUCCIÓN

La Radio Definida por Software (SDR) es un sistema de comunicaciones cuyo desarrollo comenzó a principios de la década de los 90 con el proyecto militar SpeakEasy. Este proyecto tenía como objetivo implementar diferentes modelos de comunicación en un equipo programable y ajustable en sus parámetros. El proyecto logró cumplir sus objetivos para el año 1995 y desde entonces ha ido mejorando en aspectos como costos, dimensiones e incremento en su capacidad de procesamiento; detalles significativos que conllevaron a su rápida expansión en todos los niveles de aprendizaje.

Debido a la flexibilidad y fácil implementación de la tecnología SDR, junto al rápido avance de las telecomunicaciones, la SDR ha tomado un gran auge en el mundo y ha evolucionado significativamente. Ha encontrado aplicaciones en diversas áreas, especialmente en la enseñanza de sistemas de comunicaciones, la investigación, la industria y el despliegue práctico de varias tecnologías inalámbricas de vanguardia. Esto hace necesario el conocimiento de estas nuevas tecnologías, el uso de software que permita su configuración y los parámetros que se deben tener en cuenta al momento de su implementación.

En este proyecto se propone crear un espacio de enseñanza didáctico entre profesores y alumnos de la Universidad Popular de Cesar, presentando la tecnología de SDR a través del desarrollo de guías prácticas de laboratorio. Esto permitirá aplicar los conocimientos adquiridos en las asignaturas de Comunicaciones I y Comunicaciones II en un software interactivo como GNU Radio, utilizando los módulos HackRF One y Nooelec NESDR SMARt v5.

2. ANTECEDENTES

En esta sección se aborda el desarrollo e implementación de un sistema de comunicación de Radio Definido por Software desde una perspectiva global hasta una escala local, resaltando los progresos que se han realizado en relación a la temática tratada en el presente proyecto.

2.1. ÁMBITO LOCAL Y REGIONAL

Sanabria Lemos, Nieva Suarez (2018), plantearon como objetivo principal en su tesis el desarrollo de un sistema de comunicación de Radio Definida por Software (SDR) con la finalidad de respaldar el proceso de aprendizaje en el área de sistemas inalámbricos. Para esto, implementaron el sistema haciendo uso de los equipos USRP1, que se encontraban disponibles en el laboratorio de la Universidad Autónoma de Occidente, las técnicas de modulación PSK y QAM de orden variable y un aplicativo de computadora (GNU Radio). El método de investigación utilizado fue el método analítico, empleado con el fin de separar las partes que conforman la modulación digital. Entre sus resultados, se vio reflejado una variación en la ubicación de los puntos en el diagrama de constelación del receptor, en comparación con las ubicaciones en el transmisor. Dicha variación fue debido a la presencia de ruido blanco aditivo gaussiano, el cual produjo pérdida variable de información en cada técnica de modulación aplicada. Como conclusión, identificaron que, en la gran mayoría de la documentación revisada, no se tienen en cuenta ciertos parámetros claves para obtener un funcionamiento óptimo en este tipo de sistemas, por lo que, puede ser necesario agregar o reemplazar algunos componentes. Además, encontraron que un mayor número de muestras por símbolo genera un mayor nivel de ruido, por lo que, recomiendan que el número máximo de muestras sea 4 [1].

En 2016, los autores Forero A., Triana W. y Diaz W. plantearon llevar a cabo la implementación de herramientas de Radio definida por Software (SDR) en el pensum académico del programa de pregrado de telecomunicaciones de la Universidad ECCI. El enfoque de su estudio se centró en las señales AM y FM, utilizando un aplicativo de computadora GNU Radio y equipos USRP. Esto con la finalidad de crear laboratorios prácticos y así facilitar un recurso de aprendizaje tanto para estudiantes como para profesores, permitiendo una mayor comprensión de los distintos conceptos de comunicaciones a través de prácticas en un entorno real [2].

2.2. ÁMBITO INTERNACIONAL

Byron Klever Lisintuña Cuchiparte (2019) realizó su tesis con un enfoque en la aplicación de un sistema SDR (Radio definida por software) en prácticas multidisciplinarias dentro del programa de Telecomunicaciones en la Universidad Israel. El procedimiento de investigación utilizado fue de tipo deductivo, ya que los principios teóricos descritos al principio se llevaron a cabo en un hecho concreto. Este consistió en utilizar el dispositivo RTL-SDR 2832U como sintonizador de

frecuencias para captar señales inalámbricas cuya oscilación está entre los 25 MHz y los 1700 MHz.

Las señales inalámbricas que ingresaban a este sintonizador se procesan en tiempo real en el software Matlab-Simulink, donde se ajustaban los parámetros respectivos para visualizar y analizar el espectro de la señal. Como resultado de la investigación e implementación, se logró llevar a cabo un estudio del funcionamiento de los sistemas SDR. Esto culminó en la creación de un manual de usuario con el propósito de llevar a cabo prácticas de laboratorio que fortalecieran los conceptos fundamentales en el procesamiento y análisis de señales al aplicar las distintas configuraciones de bloques en el entorno de Simulink [3].

Álvaro Gutiérrez Rivera (2021) realizó su trabajo de fin de grado en la Universidad de Sevilla (España) con el objetivo de presentar, de manera teórico-práctica, la tecnología SDR. Esta tecnología es relativamente nueva y presenta numerosos campos de aplicación aún no explorados, lo que permitirá un notable ahorro desde el punto de vista económico. La investigación se centró en proporcionar información sobre cómo funciona la tecnología SDR y profundizar en los aspectos de hardware y software que la envuelven. El propósito de esto era analizar y proyectar los resultados obtenidos en nuevos campos de investigación. El documento expuesto demostró la posibilidad de implementar cualquier tipo de sistema de radiofrecuencia mediante un SDR y conocimientos previos sobre procesamiento de señales y programación [4].

3. LÍNEA Y TEMA DE INVESTIGACIÓN

La línea de investigación en la está centrada este proyecto es “Electrónica, Optoelectrónica y Ciencias Computacionales” del programa de Ingeniería Electrónica; siendo la sublínea Optoelectrónica y Procesamiento de datos, centrándose en los procesos concernientes a la transmisión y recepción de información.

Línea: Electrónica, Optoelectrónica y Ciencias Computacionales.

Sublínea: Optoelectrónica y Procesamiento de datos.

Áreas temáticas: Procesamiento digital de señales e imágenes usando herramientas de hardware y software.

4. FORMULACIÓN DEL PROBLEMA

A lo largo de la historia, hemos visto un crecimiento vertiginoso en los distintos métodos que permiten la transmisión y recepción de información entre múltiples puntos, pasando por sistemas analógicos sencillos (AM y FM) hasta sistemas digitales más complejos como (ASK, FSK, PSK y QAM).

No obstante, se presenta un desafío al utilizar diversas modulaciones, pues el cambio de técnica lleva consigo modificaciones parciales o totales en el Hardware. Por tal razón, surgen los sistemas de Radios Definidos por Software (SDR), que permiten cambiar el tipo de modulación mediante cambios a nivel de software.

Según informes dados por Mordor Intelligence el mercado de la radio definida por software en el campo militar a nivel global, tuvo un valor de 12.45 mil millones de dólares para el año 2020, no obstante, se estima que para el año 2028 este mismo, tenga un valor de 16.20 mil millones de dólares, destacando el aumento en la implementación de sistemas de comunicación SDR en la infantería a fin de obtener sistemas más seguros, por lo que, se verá reflejado un incremento en la demanda de ingenieros con conocimientos referentes a estos sistemas [5].

En la actualidad existen en el mercado una amplia gama de transmisores y receptores SDR, sin embargo, la información sobre su uso y configuración se encuentra dispersa en foros y comunidades en línea, donde la información carece de la claridad necesaria para su fácil reproducción y adquisición.

La Universidad Popular del Cesar en sus laboratorios no cuenta con dispositivos SDR, lo que puede ocasionar que sus Ingenieros Electrónicos en formación presenten falencias en el conocimiento y uso de este tipo de dispositivos. Por tanto, surge el siguiente interrogante ¿Cómo puede la implementación de un entrenador SDR junto al desarrollo de guías prácticas de laboratorio mejorar el proceso de formación de los estudiantes de Ingeniería Electrónica de la Universidad Popular del Cesar en el campo de las comunicaciones?

5. JUSTIFICACIÓN

Los grandes avances en telecomunicaciones a través del tiempo han impulsado cada vez más a que los sistemas sean más refinados y versátiles, dando pie a los sistemas de comunicación de Radio Definido por Software (SDR). Este sistema de procesamiento digital permite tomar una señal banda base y aplicar cualquier modulación a través de software, lo que hace que estos sistemas sean esenciales en una amplia variedad de aplicaciones actuales, tal es el caso de la telefonía celular, siendo utilizada en la implementación de redes 5G y LTE [6]. Así mismo, en el ambiente militar, es aplicada en dispositivos de detección de minas antipersonas, radares y diseño de nuevos sensores [7]. No obstante, en ambos entornos, se requieren cambios continuos en los protocolos de transmisión, volviéndose imprescindible este tipo de sistemas.

Como ingeniero electrónico un reto que se afronta es mantenerse actualizado en un entorno que se encuentra en constante evolución. En este contexto, la necesidad de contar con un dispositivo de entrenamiento SDR con modulaciones variables y guías prácticas de laboratorio se hace aún más urgente. Puesto que, a través de estas herramientas pedagógicas no solo se busca enriquecer el proceso de aprendizaje de los estudiantes de Ingeniería Electrónica de la Universidad Popular del Cesar, sino, también proporcionar una formación práctica, en donde puedan aplicar los conocimientos teóricos en un ambiente controlado, logrando una mejora en las habilidades y destrezas en el manejo de este tipo de sistemas de comunicación, dándoles una mayor competitividad en el campo laboral actual.

6. OBJETIVOS

6.1. OBJETIVO GENERAL

- Implementar un entrenador de Radio Definido por Software para realizar prácticas en el laboratorio de Comunicaciones de la Universidad Popular del Cesar.

6.2. OBJETIVOS ESPECÍFICOS

- Evaluar las herramientas computacionales existentes que permitan la configuración de dispositivos SDR.
- Modelar en una herramienta computacional la transmisión y recepción de información aplicando las modulaciones AM, FM, ASK, FSK, PSK y QAM.
- Implementar en un entorno real las modulaciones AM, FM, ASK, FSK, PSK y QAM utilizando dispositivos SDR.
- Elaborar guías prácticas de laboratorio que detallen la configuración de los dispositivos SDR, los modelos de transmisión y recepción de información para las modulaciones AM, FM, ASK, FSK, PSK y QAM en una herramienta computacional.

7. MARCO TEÓRICO

En este capítulo, se proporciona la fundamentación teórica que sustenta la investigación realizada en esta tesis, destacando los conceptos esenciales.

7.1. Espectro electromagnético.

El espectro electromagnético abarca todas las frecuencias en las que se generan radiaciones electromagnéticas, las cuales se propagan a diferentes velocidades y frecuencias, permitiendo clasificarlas según su longitud de onda, tal como se muestra en la figura 1 [8].

Cabe mencionar, que la frecuencia (f) y longitud de onda (λ) se relacionan mediante la ecuación de la velocidad de la luz $c = f \cdot \lambda$; esta relación es inversamente proporcional, lo que quiere decir que, a mayor frecuencia, menor longitud de onda.

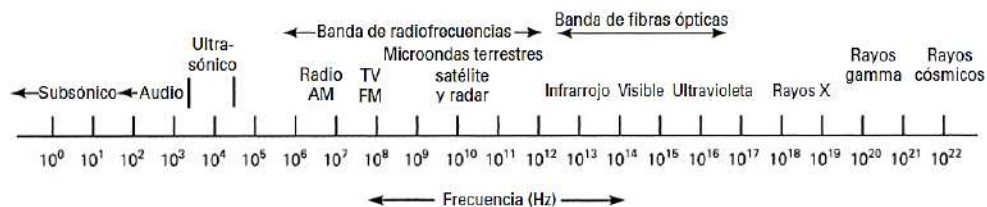


Figura 1. Espectro electromagnético de frecuencias. Fuente: Tomasi, 2003.

7.1.1. Espectro radioeléctrico

El espectro radioeléctrico constituye la porción del espectro electromagnético que es utilizado específicamente para transmitir señales de radio y datos de información por medios no guiados, siendo las frecuencias inferiores a los 3.000 GHz las destinadas para este uso (figura 1).

7.1.2. Regulación del espectro radioeléctrico en el Colombia

En Colombia, la entidad encargada de gestionar el espectro radioeléctrico, es el ministerio de tecnologías de la información y las comunicaciones. Según lo estipulado en el Decreto 4169, su responsabilidad incluye la planificación, asignación del espectro radioeléctrico, la formulación de lineamientos y políticas bajo los cuales se regulará este recurso [9].

7.2. Sistema básico de comunicaciones

Un sistema de comunicaciones se encuentra compuesto por una cantidad de elementos fundamentales que hacen posible la transmisión de la información de forma eficiente, siendo estos los siguientes [10]:

- **Mensaje:** Es la información que se desea transmitir, pudiendo ser una señal análoga o digital (voz, datos o video).
- **Transmisor:** Transforma el mensaje en una señal apropiada para ser transmitida.
- **Medio de transmisión:** Canal o método a través del cual es transmitida la información, pudiendo ser guiado (fibra óptica, cable de cobre) o no guiado (ondas de radio, microondas).
- **Ruido:** Cualquier señal no deseada que interfiera con la comunicación, distorsionando o degradando el mensaje transmitido.
- **Receptor:** Captura la señal transmitida y realiza el proceso de recuperación para obtener la información original.
- **Mensaje recuperado:** Es la información obtenida posteriormente al proceso de demodulación y decodificación.

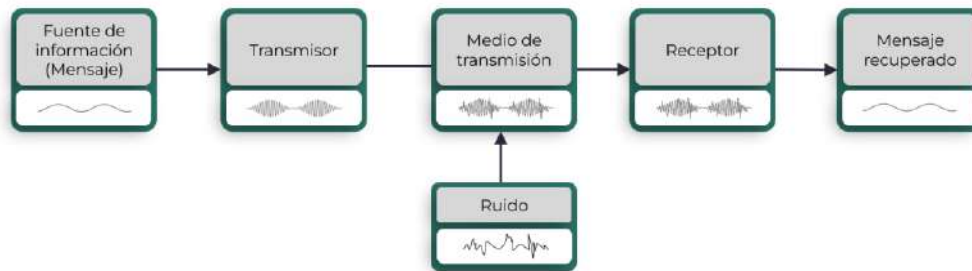


Figura 2. Sistema de comunicación básico. Fuente: Basado en [10].

7.3. Teorema de Nyquist

El teorema de Nyquist establece que la frecuencia mínima de muestreo necesaria para lograr reconstruir de forma correcta una señal analógica a partir de sus muestras digitales, debe ser al menos el doble de la frecuencia máxima de la señal.

$$f_s > 2 \cdot f_{max}$$

Donde:

- f_s es la frecuencia de muestreo en Hertz
- f_{max} es la frecuencia mas alta presente en la señal analógica en Hertz

7.4. Modulación

La modulación hace referencia al proceso mediante el cual se varían características básicas de una señal eléctrica (amplitud, frecuencia y fase), generalmente de mayor frecuencia, conocida como portadora, en función de la señal de información de baja frecuencia, comúnmente llamada moduladora. A la señal resultante de este proceso se le denomina señal modulada.

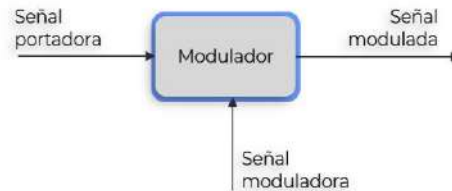


Figura 3. Esquema básico de un sistema modulador. Fuente: Basado en Tomasi, 2003.

La modulación se realiza a través del bloque modulador y su salida es conectada al medio de transmisión elegido, y una vez en el medio, son captadas por un circuito demodulador, el cual realiza el proceso inverso al del modulador y reconvierte la señal modulada en la información original (figura 4) [8].

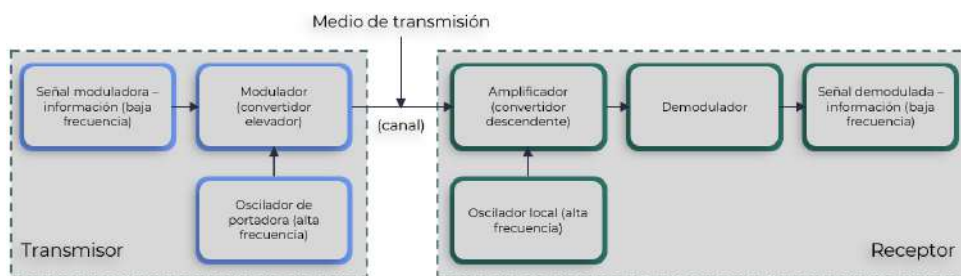


Figura 4. Esquema de bloques de un sistema de comunicación. Fuente: Basado en Tomasi, 2003.

Existen 2 razones principales por la cuales se hace necesario modular:

1. La transmisión señales a través de antenas se vuelve compleja cuando se utilizan bajas frecuencias, por lo que se hace necesario el uso de una señal modulada en alta frecuencia para facilitar su propagación en el medio de transmisión.
2. Cada estación hace uso de una banda de frecuencia diferente, garantizando la calidad de la información y evitando interferencias, lo que asegura un espectro radioeléctrico ordenado.

7.5. Técnicas de modulación

7.5.1. Modulación de amplitud (AM)

La modulación de amplitud (AM) implica variar la amplitud de una señal portadora en función de la amplitud de la señal moduladora, de manera que la señal moduladora se combina con la portadora, provocando que su envolvente refleje las variaciones de la información transmitida (figura 5).

$$v_{am}(t) = [v_c + m(t)]\cos(\omega_c t)$$

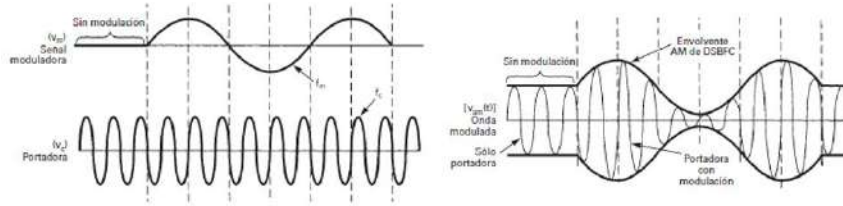


Figura 5. Modulación AM DSBFC (Doble banda lateral con full portadora). Fuente: Tomasi, 2003.

En AM existe un concepto clave, el cual es el coeficiente de modulación, que indica la variación de amplitud de la onda de salida cuando sobre la portadora actúa la señal moduladora. Este debe ser menor o igual a 1 para que la envolvente represente con mayor fidelidad del mensaje [8].

$$m = \frac{E_m}{E_c}, \quad M = \frac{E_m}{E_c} \cdot 100$$

Por último, el espectro de frecuencias para una onda AM, está conformado por una onda portadora y dos bandas laterales (figura 6). El ancho de banda en AM es el resultado de la diferencia entre la frecuencia máxima superior y la frecuencia máxima inferior, o también, dos veces la frecuencia máxima de la señal modulante $B = 2 \cdot (f_{m(max)})$ [8].

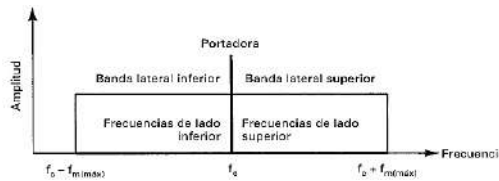


Figura 6. Espectro en frecuencia para señal modulada en AM. Fuente: Tomasi, 2003

7.5.2. Modulación de frecuencia (FM)

En FM, la frecuencia de la señal portadora es directamente proporcional a la amplitud de la señal moduladora. Esto proporciona una mayor resistencia al ruido e interferencias electromagnéticas, ya que la información transmitida es codificada en términos de frecuencia, lo que contribuye a una mayor fidelidad en el proceso de transmisión y recepción de datos.

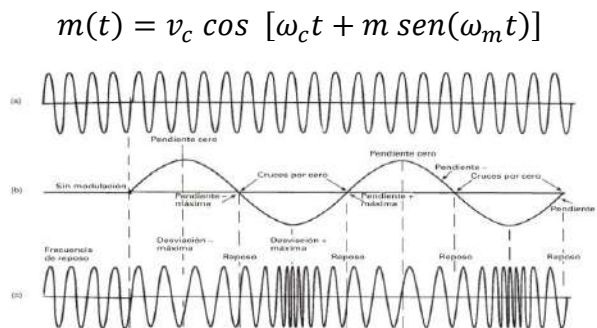


Figura 7. Modulación FM de una señal portadora senoidal por una señal de onda senoidal: (a) Portadora no modulada; (b) Señal moduladora; (c) Señal modulada con frecuencia. Fuente: Tomasi, 2003.

Otro término de mucha importancia en FM, es la desviación de frecuencia, este hace referencia a la medida de cuanto se retira la frecuencia de la señal portadora de su frecuencia inicial en reacción a las variaciones de la señal moduladora. Tal medida, es expresada comúnmente como el valor máximo en que puede desplazarse la frecuencia (Δf) [8].

7.5.3. Modulación por desplazamiento de amplitud (ASK)

La modulación por desplazamiento de amplitud se puede considerar como un caso particular de AM, en el cual la señal que contiene la información es una señal binaria que consta de dos niveles de tensión y una anchura de bit T, y la señal portadora es una onda de tipo sinusoidal con amplitud constante y alta frecuencia.

$$f_{ASK}(t) = m(t) \cdot \cos(\omega_c t)$$

$$\text{Si } m(t) = 0, \quad v \rightarrow f_{ASK}(t) \rightarrow 0; \quad \text{Si } m(t) = A, \quad v \rightarrow f_{ASK}(t) \rightarrow A \cos(\omega_c t)$$

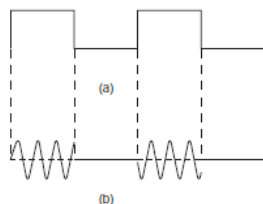


Figura 8. Modulación OOK: (a) Señal binaria de información; (b): Señal modulada ASK. Fuente: Tomasi, 2003.

En la figura 8 se aprecia la modulación On-Off Keying (OOK), un caso particular de ASK, donde la señal portadora permanece sin variaciones de amplitud y fase cuando la señal binaria es 1, sin embargo, cuando la información binaria es 0, se nota la ausencia de la señal portadora [11].

7.5.3.1. Transmisión en ASK

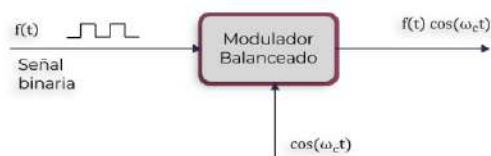


Figura 9. Modulador ASK. Fuente: J. Luque Rodríguez y S. Clavijo Suero, Modulación de señales digitales.

El modulador ASK se puede asimilar a un multiplicador analógico, en el cual la señal de salida es el resultado del producto de las dos señales de entrada. En este caso, se representan las variaciones de amplitud inmersas en una secuencia de datos digitales como variaciones de amplitud de la onda portadora.

7.5.3.2. Ancho banda en ASK

El ancho en la modulación ASK el ancho de banda se encuentra sujeto a la velocidad de transmisión, es decir, la tasa de bits (R_b), y puede ser calculado de la siguiente forma:

$$BW = 2 \cdot R_b$$

Donde:

- BW es el ancho de banda en Hertz (Hz)
- R_b es la tasa de bits en bits por segundo (bps)

Por otro lado, se debe tener en cuenta que en la modulación ASK, la frecuencia de bits (f_b) es igual a la tasa de bits (R_b), puesto que, un símbolo de la señal modulada portará un solo bit de información.

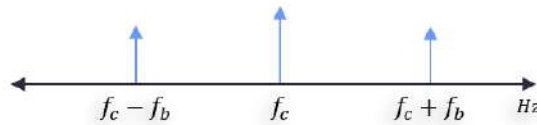


Figura 10. Espectro en frecuencia de ASK. Fuente: Basado en Tomasi, 2003.

7.5.4. Modulación por desplazamiento de frecuencia (FSK)

La Modulación por Desplazamiento de Frecuencia (FSK) posee similitudes con respecto a la modulación de frecuencia convencional; sin embargo, en FSK, la señal moduladora no es una señal analógica con cambios constantes; en este caso, es una señal binaria que se alterna entre dos valores de voltaje.

$$v_{fsk}(t) = v_c \cos \{2\pi[f_c + v_m(t)\Delta f]t\}$$

En FSK, la señal moduladora [$v_m(t)$] toma la forma de una onda binaria normalizada, en donde el valor 1 lógico es igual a 1, y el valor 0 lógico es igual a -1. Esto da lugar a un desplazamiento en la frecuencia de salida, cambiando entre dos frecuencias: frecuencia de marca para cuando la entrada es un 1 lógico (f_m), y frecuencia de espacio para cuando es un 0 lógico (f_s).

Es importante destacar que la frecuencia de la señal portadora se encuentra a una determinada distancia de las frecuencias de marca y de espacio, tal distancia corresponde a la desviación máxima de frecuencia (Δf), tal como se muestra en la figura 11.

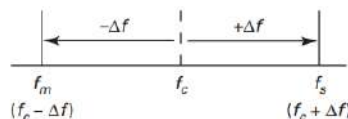


Figura 11. Desviación de frecuencia en FSK. Fuente: Tomasi, 2003.

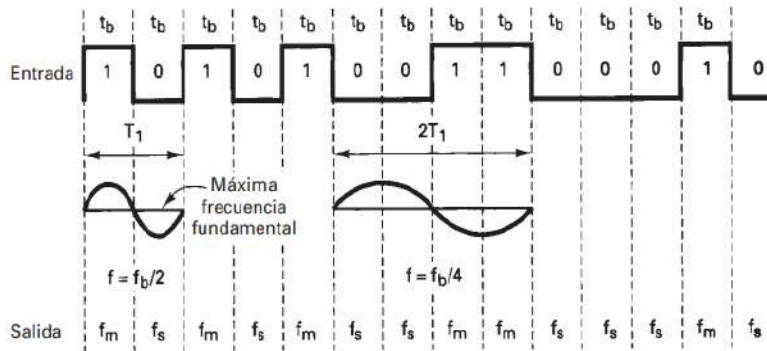


Figura 12. Señal de entrada y salida en la modulación FSK. Fuente: Tomasi, 2003.

En la figura 12, podemos observar el tiempo de bit (t_b), el cual representa el periodo en el que es transmitido un solo bit. Este intervalo es igual al tiempo mínimo necesario en la salida para alternar entre una frecuencia de espacio y de marca, o viceversa [8].

7.5.4.1. Transmisión en FSK

Un modulador FSK convencionalmente resulta ser un oscilador controlado por voltaje (VCO). En este, la frecuencia de reposo es determinada de tal forma que quede distanciada en igual medida tanto de la frecuencia de marca como la de espacio, y una vez ingrese un 1 lógico, la salida del VCO es desplazada hacia la frecuencia de marca, y cuando ingrese un 0 lógico se desplaza hacia la frecuencia de espacio, logrando así alternar entre ambas frecuencias en función del valor lógico de entrada [8].



Figura 13. Esquema simplificado de un modulador FSK. Fuente: Basado en Tomasi, 2003.

7.5.4.2. Ancho de banda en FSK

En FSK, el ancho de banda requerido para transmitir una señal modulada depende directamente de la frecuencia de bits y de la desviación máxima de frecuencia, tal como se muestra en la siguiente expresión: $B = 2 \cdot (\Delta f + f_b)$ [8].

7.5.4.3. Espectro de frecuencias en FSK

En la modulación por desplazamiento de frecuencia (FSK), la señal portadora obtenida tras el proceso de modulación resulta ser una combinación de frecuencias superpuestas, incluyendo la frecuencia de marca, de espacio y la propia frecuencia de portadora. En consecuencia, en el espectro de frecuencias se reflejarán tres

señales de impulso con variaciones de amplitud, señalando la presencia de cada una de estas frecuencias (figura 14).

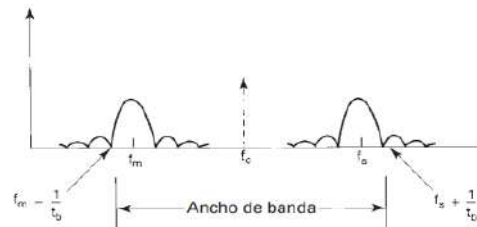


Figura 14. Espectro de frecuencias en FSK. Fuente: Tomasi, 2003.

7.5.4.4. Recepción en FSK

Existen diversas configuraciones de circuitos demoduladores, sin embargo, el más utilizado es el PLL-FSK (figura 15), un demodulador coherente, caracterizado por conservar la fase de la señal de entrada. En este, se tiene como entrada una señal modulada que alterna entre la frecuencia de marca y de espacio, por consiguiente, el VCO comienza oscilando a una frecuencia inicial, mientras que el comparador de fase realiza comparaciones entre la fase del VCO y la fase de la señal de entrada. El filtro de bucle realiza un ajuste de la frecuencia del VCO, a fin de que ésta coincida con la frecuencia de la señal de entrada, dando como resultado la señal demodulada [8].

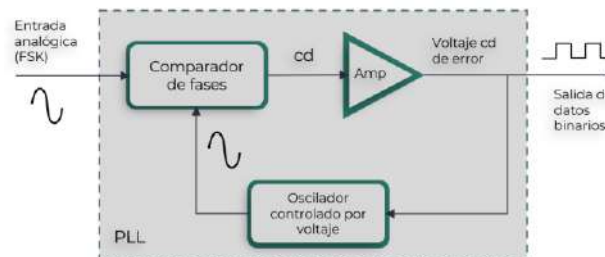


Figura 15. Demodulador PLL-FSK. Fuente: Basado en Tomasi, 2003.

7.6. Diagrama de constelación

El diagrama de constelación consiste en una forma gráfica de representar en el plano complejo los estados posibles, ubicados en función a los valores de amplitud y fase. En este plano, las componentes ubicadas en el eje horizontal, se encuentran en fase con la señal portadora, mientras que las que están situadas sobre el eje vertical, simbolizan las componentes en cuadratura (90°) [12].

7.7. Modulación por desplazamiento de fase (PSK)

En la modulación por desplazamiento de fase (PSK) la fase de la señal portadora varía entre un número limitado de estados en relación con la amplitud de la señal moduladora, recibiendo múltiples denominaciones en función del número de estados posibles que se utilicen.

- Modulación por desplazamiento de fase binaria (BPSK), para 2 estados posibles.
- Modulación por desplazamiento de fase en cuadratura (QPSK), para 4 estados posibles.

Cabe mencionar, que a medida que se aumenta la cantidad de estados posibles, aumentará la cantidad de datos que pueden ser transmitidos dentro del mismo ancho de banda. Sin embargo, al mismo tiempo, aumenta el nivel de susceptibilidad del sistema ante las interferencias y el ruido [8].

7.7.1. Modulación por desplazamiento binario de fase (BPSK)

En BPSK existen dos valores de fases en lo que puede alternar la salida para una única frecuencia de portadora, simbolizando una de las fases un 1 lógico, y otra un 0 lógico. De esta manera, cuando la señal de entrada varíe, la fase de la señal portadora cambia entre dos ángulos que se encuentran desfasados 180° , tal como se muestra en la figura 16.

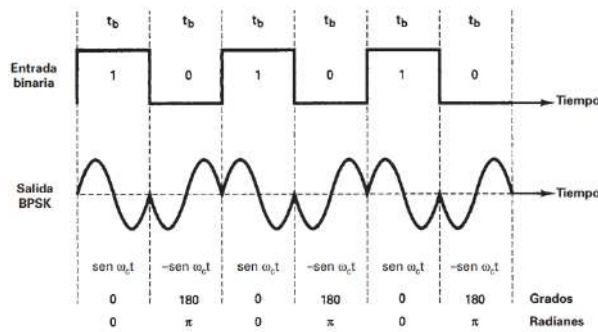


Figura 16. Señal de salida de un modulador BPSK en función de una señal de entrada binaria normalizada. Fuente: Tomasi, 2003.

7.7.1.1. Transmisión en BPSK.

En los transmisores BPSK, es utilizado un modulador balanceado (figura 17), el cual tiene un funcionamiento muy similar al de un interruptor, alternando entre dos estados en función de la condición lógica de entrada. Obteniendo a la salida, una portadora en fase o desfasada 180° [8].

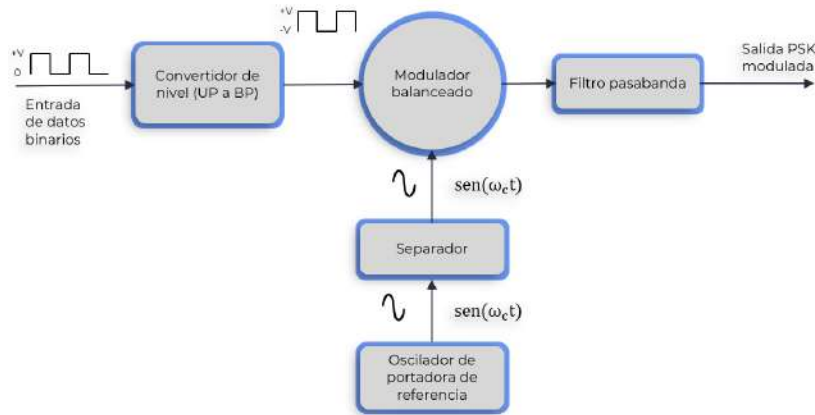


Figura 17. Diagrama simplificado de un modulador BPSK balanceado. Fuente: Basado en Tomasi, 2003.

7.7.1.2. Ancho de banda en BPSK.

En BPSK, la rapidez de cambio en la salida (baudio), es igual a la rapidez de cambio en la entrada (bps), por lo que, el ancho de banda máximo se obtiene cuando en la entrada existe una alternancia continua entre unos y ceros, y puede ser calculado de la siguiente forma: $B = (2 \cdot f_b)/2 = f_b$ [8].

7.7.1.3. Recepción en BPSK.

En BPSK, es utilizado un circuito receptor coherente, por ende, las frecuencias empleadas en el proceso de demodulación se encuentran sincronizadas con las frecuencias establecidas en el transmisor, coincidiendo en fase y frecuencia ambas portadoras (la generada en el transmisor, y la obtenida en el receptor), esto gracias al convertidor de nivel y al bloque de recuperación de reloj [8].

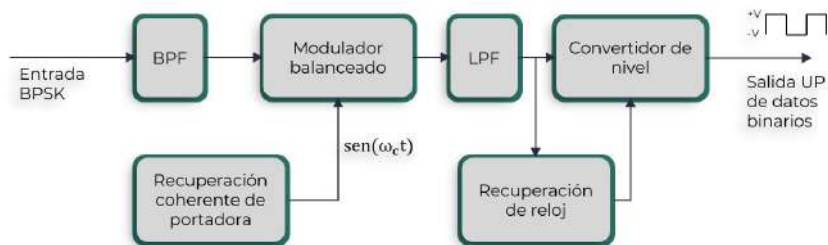


Figura 18. Esquema de bloques simplificado de un receptor BPSK. Fuente: Basado en Tomasi, 2003.

7.7.1.4. Diagrama de constelación en BPSK

El diagrama de constelación en la modulación por desplazamiento binario de fase (BPSK), consta solamente de dos símbolos en el plano complejo, cada símbolo representa una fase de la señal portadora, tal como se muestra en la figura 19.

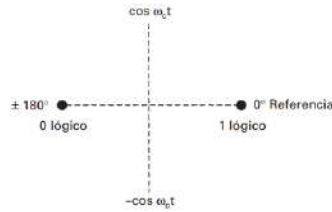


Figura 19. Diagrama de constelación. Fuente: Tomasi, 2003.

7.8. Modulación por desplazamiento de amplitud en cuadratura (QAM)

La modulación de amplitud en cuadratura (QAM), es un tipo de modulación M-aria, en la que para grupos de n bits, podemos obtener $M=2^n$ salidas diferentes.

Esta técnica de modulación digital consiste en que la señal portadora es modulada tanto en amplitud como en fase, para dar lugar a la salida analógica QAM.

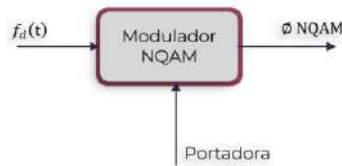


Figura 20. Esquema básico de un modulador NQAM. Fuente: Basado en DANIZIO, P. E. Sistemas de comunicaciones.

La figura 20 presenta el esquema genérico de un modulador QAM, representado como un bloque donde la señal digital y la portadora obtienen a la salida amplitudes y fases variables. Para luego asignar un grupo de bits a una fase y amplitud; el tamaño de la agrupación de bits puede variar dependiendo del valor de M.

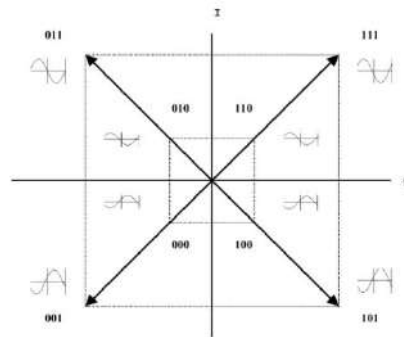


Figura 21. Diagrama de constelación de la modulación 8-QAM. Fuente: DANIZIO, P. E. Sistemas de comunicaciones.

En la figura 21, se aprecia las variaciones de amplitud y fase que pueden ser efectuadas en la señal portadora.

Como se mencionó anteriormente, el valor de M designa la cantidad de salidas diferentes que se pueden obtener, lo que hace posible dividir las diferentes técnicas de modulaciones QAM en función del número de combinaciones posibles de fase y amplitud, conocidas como 4-QAM, 8-QAM, 16-QAM, etc. A medida que aumenta el número de combinaciones posibles, la velocidad de transmisión será más alta, pero también aumentará su susceptibilidad a las perturbaciones [13].

7.9. Sistema de comunicación SDR

7.9.1. Definición

El concepto de Radio Definido por Software se remonta a los años 90, donde su creador Joseph Mitola, ampliamente reconocido como el “padre del SDR”, presentó este término al utilizarlo para referirse al tipo de radios que podían ser reconfiguradas a través de software. Mitola expuso lo que sería una radio definida por Software ideal; en esta, los componentes físicos serían un conversor digital a analógico (DAC) con una antena para el transmisor y un conversor analógico a digital (ADC) con una antena en el receptor, quedando las funciones restantes configuradas a través de procesadores que pueden ser reprogramados. Esta visión innovadora separaría la tecnología de las comunicaciones en dos componentes esenciales: el primer componente es el hardware de radiofrecuencia (RF), encargado de la transmisión y recepción, mientras que el segundo es el software, el cual permite obtener sistemas de comunicación con un alto nivel de flexibilidad con respecto a los parámetros de transmisión, ofreciendo así la capacidad de modificar el rango de frecuencia, la técnica de modulación, el ancho de banda o la potencia máxima de salida, todo esto sin realizar ningún tipo de cambio en el hardware [14].

Desde la introducción de la visión de Mitola, los sistemas de Radio definido por Software (SDR) han estado en continua evolución, convirtiéndose en un elemento indispensable en la revolución de las comunicaciones modernas gracias a su capacidad de adaptarse y ser reconfigurados en respuesta a las cambiantes demandas [15].

7.9.2. Esquema básico de SDR

Con respecto a su arquitectura, la Radio Definida por Software podría ser considerada como un transceptor, en el cual todos los parámetros concernientes a su funcionamiento son establecidos mediante software, gracias a su hardware adaptable. No obstante, su esquema básico se divide en tres secciones: la sección de RF, la sección de IF y la sección de banda base. La unión de la sección RF e IF conforman la parte implementada en hardware, y la sección de banda base es implementada mediante software [16].

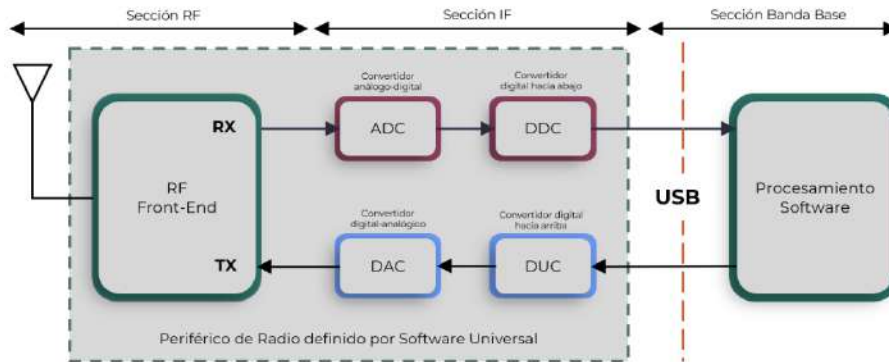


Figura 22. Esquema básico de bloques funcionales de SDR. Fuente: Basado en [15].

La labor que desempeña cada sección en el esquema básico presentado en la figura 22, varía con base al modo en que se encuentre funcionando el dispositivo, ya sea en modo transmisor o receptor [15], tal como se explica a continuación:

Funciones de las secciones en modo transmisor:

- **Sección de banda base:** En esta etapa se efectúa el procesamiento de la señal a transmitir mediante software, ejecutando operaciones como modulación y análisis espectral.
- **Sección de IF:** Se reciben las señales de banda base y, posteriormente, son convertidas a frecuencia intermedia (IF) utilizando un módulo DUC (Convertidor Digital Ascendente) y un DAC (Convertidor Digital a Análogo).
- **Sección RF:** En esta sección final, se transmiten las señales de IF, no sin antes ser amplificadas y moduladas.

Funciones de las secciones en modo receptor:

- **Sección RF:** Se realiza la recepción de las señales de radiofrecuencia, que seguidamente son adecuadas y convertidas a frecuencias intermedias (IF).
- **Sección IF:** Se hace la conversión de las señales IF a banda base, empleando un módulo DDC (Convertidor Descendente Digital) que reduce la tasa de muestreo de la señal. Luego, la señal es digitalizada mediante un conversor análogo a digital (DAC) antes de enviarla a la sección de banda base.
- **Sección de banda base:** En este tramo final, es recibida la señal digitalizada y se procede a efectuar el procesamiento mediante software, implementando diversas operaciones como el filtrado, la demodulación, etc.

7.10. Hardware SDR disponibles en el mercado

Hoy en día, existen una variedad de dispositivos a través de los cuales se hace posible la implementación de un sistema de comunicación basado en Radio Definido por Software (SDR), haciéndose posible encontrar equipos cuyo funcionamiento es solamente como receptores, y otros como transceptores, es decir, pueden ser

configurados ya sea como transmisor, o como receptor. Por lo que, antes de adentrarnos, en los capítulos siguientes, exploraremos los equipos SDR más comunes y sus características.

7.10.1. Dongle RTL-SDR Blog V3



Figura 23. Receptor Dongle RTL-SDR Blog V3. Fuente: Amazon.

En 2017, fue lanzado el RTL-SDR Blog V3, un receptor mejorado y diseñado con el objetivo de satisfacer las necesidades de aquellos apasionados por la Radio Definida por Software (SDR) [17].

7.10.2. Nooelec NESDR SMARt v5



Figura 24. Receptor Nooelec NESDR SMARt v5. Fuente: Amazon.

Este receptor fue desarrollado en el 2021 con el propósito de ser un periférico SDR de alto rendimiento, capaz de captar una gran variedad de señales RF, y ofrecer una amplia versatilidad [18].

7.10.3. HackRF One



Figura 25. Transceptor HackRF One. Fuente: Aliexpress.

En 2014 Great Scott desarrollo HackRF One, un dispositivo transceptor de Radio Definido por Software (SDR) creado con el objetivo de poner a disposición tanto para los entusiastas como para los profesionales una herramienta asequible, que les permitiera experimentar con señales de radiofrecuencia [19].

7.10.4. PlutoSDR



Figura 26. Transceptor PlutoSDR. Fuente: Amazon.

En 2015, Analog Devices desarrolló un dispositivo transceptor de aprendizaje dirigido a los estudiantes de ingeniería, con el propósito de facilitar la comprensión

de los conceptos fundamentales de la Radio Definida por Software (SDR) y las comunicaciones inalámbricas.

7.11. Software SDR

7.11.1. GNU radio

GNU radio es un software libre y de código abierto utilizado para desarrollo de sistemas de radiofrecuencias, más exactamente sistemas de Radio Definido por Software (SDR). Este software cuenta con una interfaz bastante amigable e intuitiva, caracterizada por la codificación cromática, asignando a cada tipo de dato un color específico, lo que facilita la identificación de los tipos de datos de entrada y salida de cada bloque. Los tipos de datos que pueden ser utilizados son: byte (morado), int (verde), short (amarillo), complex (azul) y float (naranja).

Cabe mencionar que, en GNU, los bloques a través de los cuales se procesan las señales son de código abierto y pueden ser modificados mediante C++ o Python. Tal flexibilidad brinda la oportunidad a la comunidad de GNU de compartir consejos y configuraciones a través de distintos foros y páginas web en línea [20].

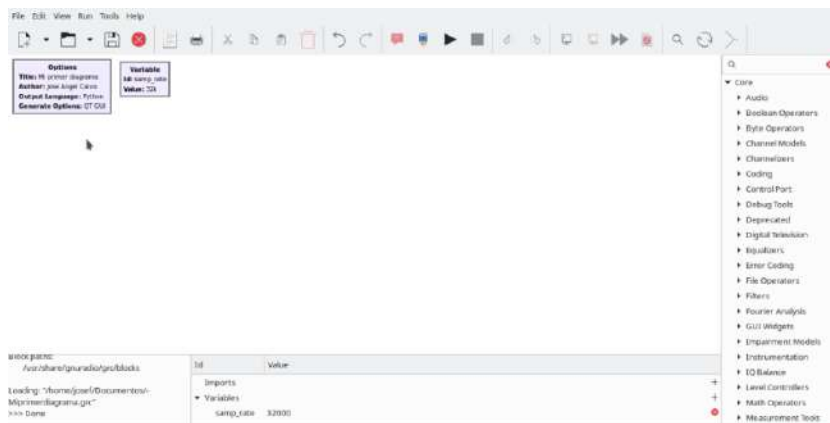


Figura 27. Interfaz gráfica de GNU Radio. Fuente: [21].

7.11.2. Matlab y Simulink

Esta plataforma de programación y cálculo numérico se utiliza comúnmente para el análisis de datos, el desarrollo de algoritmos y modelos. Este software ha tenido un gran impacto en comunidades de ingenieros y científicos en todo el mundo, debido a su diversidad y aplicabilidad en diversos campos.

Esta herramienta integra Toolboxes complementarios para ampliar aún más su variedad de aplicaciones científicas y de ingeniería, como la herramienta de simulación y diseño basada en modelos, llamada Simulink.

Simulink, en su entorno de modelado gráfico y simulación, permite la creación de diagramas de bloques, en los que cada bloque representa un componente físico, un sistema o una función dentro del modelo.

Estas herramientas son utilizadas para la implementación de sistemas SDR, ya que admiten la prueba de diseños inalámbricos en hardware. En otras palabras, permiten la interconexión del hardware SDR con el sistema programado para llevar a cabo los diseños y análisis pertinentes de los diferentes modelos a tratar [22].

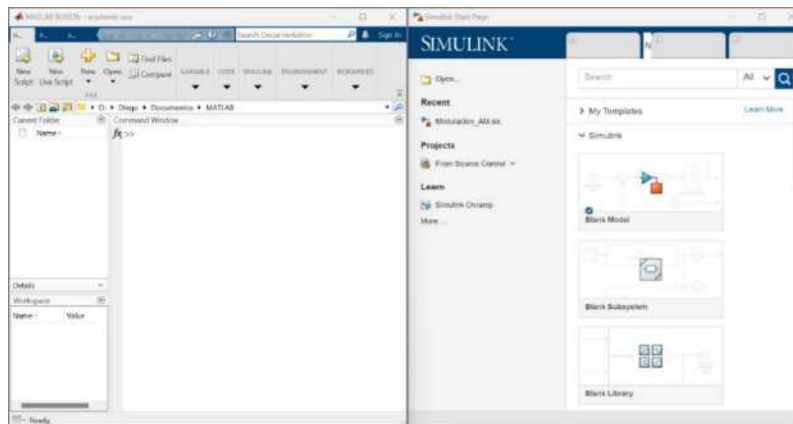


Figura 28. Interfaz gráfica de Matlab y Simulink. Fuente propia.

8. DISEÑO METODOLÓGICO

En esta sección se presenta a detalle el proceso llevado a cabo para el desarrollo e implementación de los esquemas moduladores y demoduladores haciendo uso del hardware SDR y el aplicativo seleccionado.

8.1. Requerimientos del hardware SDR

La selección del hardware SDR adecuado se basó en una serie de requerimientos funcionales que nos permitieron cubrir a cabalidad los objetivos planteados en la presente tesis, así mismo, que garantizaran la eficiencia y precisión en las pruebas y simulaciones, y así asegurar la posibilidad de validar y replicar los resultados obtenidos. Tales requerimientos fueron los siguientes:

8.1.1. Rango de frecuencia: El hardware debe ser capaz de trabajar en los intervalos de frecuencias pertenecientes a cada técnica de modulación:

- **AM:** 530 KHz - 1710 KHz
- **FM:** 88 MHz - 108 MHz
- **ASK:** Variable en función de su aplicación.
- **FSK:** HF (3 MHz - 30 MHz) y VHF (30 MHz - 300 MHz)
- **PSK:** VHF, UHF hasta GHz.
- **QAM:** MHz hasta GHz.

8.1.2. Ancho de banda: Debe contar con un ancho de banda adecuado y ajustable, puesto que, en modulaciones de alta velocidad (QAM y PSK), a un mayor ancho de banda, se tiene una mejora en la calidad de la señal y una mayor tasa de transferencia.

8.1.3. Resolución de Bits: La resolución de los convertidores ADC y DAC deben ser altas para garantizar una alta fidelidad en las conversiones que se lleven a cabo.

8.1.4. Interfaz de conexión: Este parámetro al igual que el resto, es algo a tener muy en cuenta, puesto que, dependiendo de este mismo, si es USB, Ethernet o PCIe, nos ofrecerá una mayor velocidad y facilidad de conexión.

8.1.5. Softwares compatibles: Debe ser compatible con aplicativos SDR, tales como GNU Radio, Matlab/Simulink y SDRSharp.

8.1.6. Capacidad de procesamiento: Debe poseer un procesador potente, siendo apto para manejar las complejidades que conllevan las modulaciones análogas y digitales.

8.2. Selección del hardware SDR

Teniendo en cuenta los requerimientos mencionados anteriormente, la selección adecuada del hardware SDR conlleva una investigación detallada de las opciones disponibles en el mercado, así mismo, ventajas y desventajas de cada una de estas.

Algunos de los dispositivos SDR que cumplen con los requerimientos se relacionan en la tabla 1.

Tabla 1. Cuadro comparativo de las características de los dispositivos SDR.

	Dongle RTL-SDR Blog V3	Nooelec NESDR SMARt v5	HackRF One	PlutoSDR
Rango de frecuencias	De 500 KHz hasta 1766 MHz	De 100 KHz hasta 1750 MHz	De 1 MHz hasta 6 GHz	De 325 MHz hasta 3,8 GHz
Resolución de muestreo	ADC de 8 bits	ADC de 8 bits	ADC y DAC de 8 bits	ADC y DAC de 12 bits
Rendimiento	Chip RTL2832U + Sintonizador R820T2	Chip RTL2832U + Sintonizador R820T3	Procesador LPC4320/4330 + Chips (MAX5864, MAX 2837 y RFFC5072)	Chip AD9363 + FPGA Xilinx Zynq Z-7010
Ancho de banda ajustable	Máximo de 2,4 MHz	Máximo de 2,4 MHz	Máximo de 20 MHz	Máximo 20 MHz
Conexión	SMA hembra y USB estándar	SMA hembra y USB estándar	2 SMA hembra y USB estándar	2 SMA hembra y USB estándar
Software SDR compatible	GNU Radio, HDSDR, Matlab y Simulink	GNU Radio, SDRangel, Matlab y Simulink	GNU Radio, SDR Sharp, Matlab y Simulink	GNU Radio, SDR Sharp, Matlab y Simulink

Tras una evaluación minuciosa de las opciones disponibles en el mercado, y el presupuesto disponible, se seleccionaron dos dispositivos para el desarrollo de las tesis en cuestión: el Hack RF One y el Nooelec NESDR SMARt V5.

El Hack RF One fue seleccionado debido a su versatilidad y capacidad para abarcar un amplio rango de frecuencia y un ancho de banda suficiente para trabajar con las modulaciones de alta frecuencia. Por otro lado, resulta ser compatible con una gran variedad de aplicativos de Radio Definido por Software (SDR), tales como GNU Radio, Matlab/Simulink y SDRSharp. Además, cuenta con un elevado soporte documental que existe en las comunidades de internet que contribuyen en gran medida con la resolución de problemas.

Por otro lado, el Nooelec NEDSDR SMARt V5 fue seleccionado por su accesibilidad y costo reducido, que lo hace una opción interesante en proyecto con presupuestos limitados. No obstante, a pesar de tener un costo relativamente bajo, posee un

excelente desempeño, siendo capaz de manejar las diferentes técnicas de modulación sin problemas. Cabe mencionar que este, al igual que el Hack RF One, resulta ser compatible el mismo grupo de aplicativos de SDR.

8.3. Instalación de drivers para reconocimiento del hardware SDR

En esta sección se detalla el proceso de instalación de los controladores que permiten el reconocimiento y operatividad del hardware SDR con el aplicativo. Cabe mencionar que el software realiza la instalación del driver correspondiente al dispositivo conectado, por lo tanto, basta con realizar el mismo proceso con ambos dispositivos.

Antes de iniciar, es de suma importancia que el dispositivo esté conectado, y verificarlo en el panel de administrador de dispositivos de Windows (figura 29), así mismo, comprobar que se cumple con el único requisito que requiere este software, el cual es contar con Windows 7 o superior, siendo Windows 11 el sistema operativo utilizado en este proyecto.

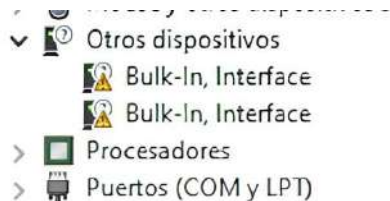


Figura 29. Detección de dispositivos SDR conectados al sistema. Fuente: propia.

8.3.1. Descarga de los drivers

El paso a seguir es la descarga del paquete de instalación del driver, por lo que, inicialmente debe dirigirse al sitio web oficial de Zadig (<https://zadig.akeo.ie/>, figura 30), y en la sección de descarga (Download), descargar la versión más reciente.



Figura 30. Pantalla de inicio del sitio web oficial de Zadig. Fuente:



Figura 31. Sección de descarga en sitio web de Zadig. Fuente:

8.3.2. Instalación del driver

Una vez descargado, se navega hasta la ubicación del archivo descargado y se inicia la instalación seleccionando la opción “si” en la ventana emergente para continuar con el proceso.

Posteriormente para actualizar el listado de dispositivos reconocidos por Zadig, nos dirigimos hacia pestaña “options” y seguidamente seleccionando la opción “List All Devices” (figura 32), a fin de que aparezcan todos los dispositivos conectados a la computadora.

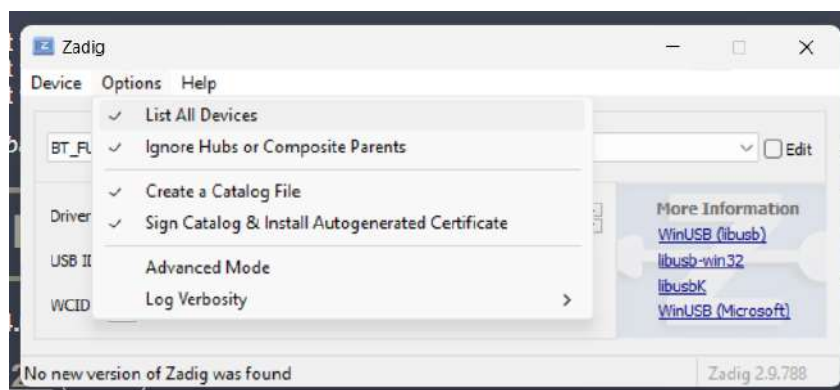


Figura 32. Pestaña options y actualización de todos los dispositivos conectados en Zadig. Fuente: propia.

En menú desplegable se selecciona la opción “Bulk-In, Interface (Interface 0)” e inmediatamente se hace clic sobre el botón “Install Driver”, dejando las otras opciones por defecto (figura 33).

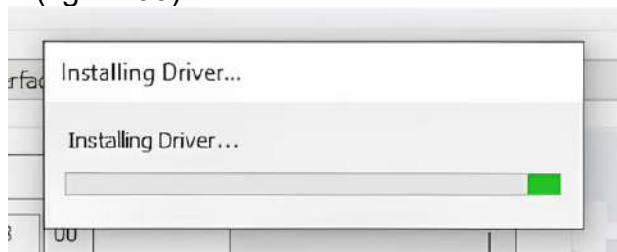


Figura 33. Proceso de instalación de Driver para hardware de SDR. Fuente propia

8.4. Requerimientos de software SDR

Para la implementación de las técnicas de modulación AM, FM, ASK, FSK, PSK y QAM con los dispositivos SDR, es de suma importancia seleccionar un aplicativo que cumpla con los siguientes requerimientos:

8.4.1. Compatibilidad con hardware: Debe ser compatible con los dispositivos SDR seleccionados (HackRF One y Nooelec NESDR SMARt V5).

8.4.2. Capacidad de simulación: El aplicativo debe proporcionar los bloques necesarios para modelar y analizar las diferentes técnicas de modulación.

8.4.3. Interfaz de usuario: Debe contar con una interfaz intuitiva y de fácil uso.

8.4.4. Documentación y soporte: Contar con un amplio soporte documental es esencial en la resolución de problemas y el manejo de los diferentes bloques que integra cada aplicativo.

8.4.5. Comunidad: Tener a disposición una comunidad activa (foros), donde encontrar tutoriales e información valiosa para el aprendizaje y solución de problemas.

8.5. Comparativa y selección del software SDR

En el mercado actual, existe una amplia variedad de aplicativos SDR que pueden satisfacer los requisitos establecidos. La tabla 2 muestra una comparación de algunas opciones junto con sus características más relevantes.

Tabla 2. Cuadro comparativo de las características de los softwares para SDR.

Software	Compatibilidad con hardware	Capacidad Simulación	Interfaz de usuario	Documentación y soporte	Comunidad	Costo
GNU Radio	Alta	Alta	Compleja al inicio	Extensa	Activa y amplia	Gratis
Matlab/Simulink	Requiere toolbox de terceros	Media	Intuitiva	Extensa	Moderada	Alta
SDRSharp (SDR#)	Compatible solo con receptores	Baja	Intuitiva	Media	Moderada	Gratis

Se seleccionó GNU Radio, como el aplicativo adecuado para el desarrollo de la presente tesis, debido a las ventajas y beneficios que ofrece respecto a las otras opciones disponibles, tales como:

- Compatibilidad con dispositivos SDR sin requerir de librerías adicionales.
- Amplia biblioteca de bloques para modelar los distintos esquemas de modulación.
- Extenso soporte documental con ejemplos de aplicación.
- Capacidad de modificar datos de transmisión y recepción en tiempo real.

8.6. Instalación de GNU Radio

La correcta instalación de GNU Radio es de suma importancia para el desarrollo de los esquemas de simulación e implementación de pruebas prácticas, por lo que, se inició con la verificación de los requisitos mínimos: un sistema operativo Windows 7 o superior, poseer una arquitectura de 64 bits, siendo Windows 11 de 64 bits, la versión utilizada en este proyecto.

8.6.1. Descarga del instalador

Para descargar el archivo instalador de GNU Radio, se debe acceder al sitio web oficial, que redirige a un artículo contenido en una wiki (<https://wiki.gnuradio.org/index.php?title=InstallingGR>). Allí, nos desplazamos hacia la sección “Quick Start” y hacemos clic en el hipervínculo que lleva como texto “Radioconda” ubicado en el recuadro que señala la versión de Windows (figura 34).

Windows	Download and Install Radioconda by following the instructions at the link and launch "GNU Radio Companion" from the Start menu	v3.10.10.0
macOS	Download and Install Radioconda by following the instructions at the link	

Figura 34. Sección destinada a la descarga del archivo de instalación de GNU Radio para Windows. Fuente:

Al hacer clic en el hipervínculo, será redirigido a un repositorio de GitHub, donde nos desplazaremos hasta la sección “Download”, y posteriormente dar clic sobre el hipervínculo “radioconda-Windows-x86_64.exe” para iniciar la descarga (figura 35).

OS	Architecture	Installer Type	Download
Linux	x86_64 (amd64)	Command-line	radioconda-Linux-x86_64.sh
Linux	aarch64 (arm64)	Command-line	radioconda-Linux-aarch64.sh
Linux	ppc64le (POWER8/9)	Command-line	radioconda-Linux-ppc64le.sh
macOS	x86_64 (Intel)	Command-line	radioconda-MacOSX-x86_64.sh
macOS	x86_64 (Intel)	Graphical	radioconda-MacOSX-x86_64.pkg
macOS	arm64 (Apple Silicon)	Command-line	radioconda-MacOSX-arm64.sh
macOS	arm64 (Apple Silicon)	Graphical	radioconda-MacOSX-arm64.pkg
Windows	x86_64 (amd64)	Graphical	radioconda-Windows-x86_64.exe

Figura 35. Sección de descarga en repositorio de GitHub. Fuente: [23].

8.6.2. Instalación de GNU Radio

En una última instancia, se accede al directorio donde se encuentra el archivo y después de dar clic sobre este, se iniciará el proceso de instalación, en el cual se deben realizar los pasos presentados en la figura 36, figura 37, figura 38, figura 39.

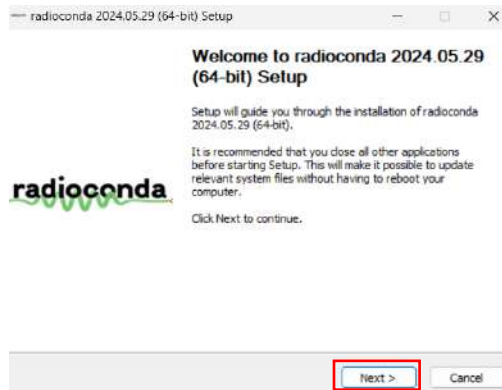


Figura 36. Pantalla inicial del asistente de instalación de GNU Radio y selección de la opción “Next” para continuar con el proceso. Es recomendable cerrar aplicaciones no relacionadas para evitar reinicios del sistema. Fuente propia.

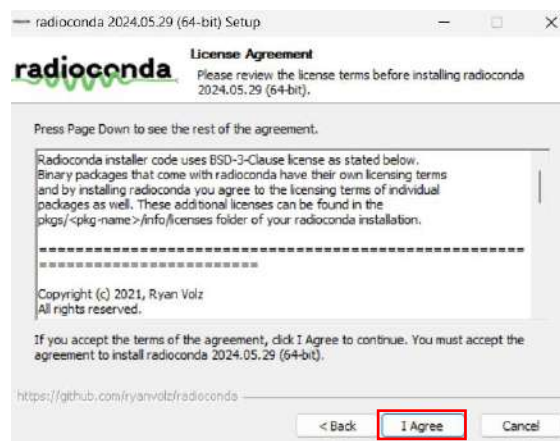


Figura 37. Pantalla del asistente donde fueron aceptados los términos y condiciones de GNU Radio al hacer clic en “I Agree”. Fuente propia

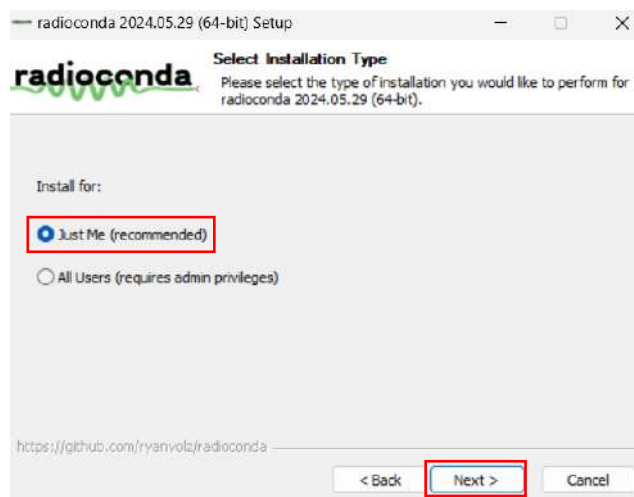


Figura 38. Pantalla de selección del tipo de instalación en la que se seleccionó la opción recomendada y se hizo clic en “Next”. Fuente propia

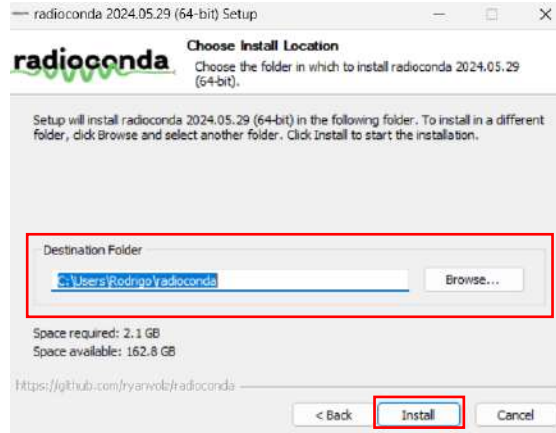


Figura 39. Pantalla de selección de lugar de instalación, en la cual se dejó el valor predeterminado y se hizo clic en "Install". Fuente propia.

8.7. Desarrollo e implementación de esquemas moduladores y demoduladores en GNU Radio

En esta sección se lleva a cabo el desarrollo e implementación en GNU Radio de los distintos esquemas correspondientes a los moduladores y demoduladores de las diferentes técnicas de modulación, por lo que, se hace necesario destacar que todas las implementaciones fueron desarrolladas inicialmente en un entorno simulado, donde tanto el esquema transmisor como el receptor se encontraban en el mismo proyecto (misma computadora), simplificando la fase de pruebas y ajuste de parámetros. No obstante, una vez adaptado a un entorno real utilizando hardware SDR, el transmisor y receptor fueron implementados en proyectos y computadores diferentes, permitiendo la validación del sistema en condiciones reales.

8.7.1. Modulación de Amplitud (AM)

Es una técnica de modulación primordial en las telecomunicaciones, de la cual existen múltiples variantes, no obstante, en este caso nos enfocamos en la doble banda lateral con full portadora (DSBFC).

8.7.1.1. Desarrollo del esquema modulador en GNU Radio

Para generar una señal AM, se emplea un proceso de mezcla no lineal entre la señal moduladora $m(t)$ y la portadora $c(t)$. En esencia, esta modulación modifica la amplitud de la portadora en función de la amplitud de la moduladora. Esto se logra utilizando un sumador y un multiplicador, como se ilustra en la (figura 40).

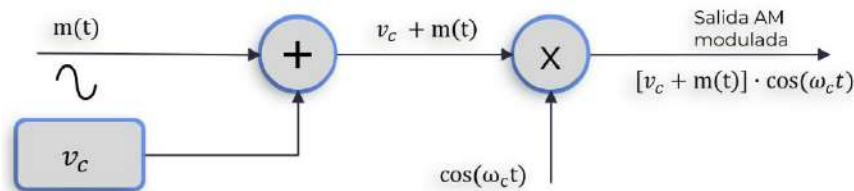


Figura 40. Esquema básico de modulador AM. Fuente propia

Para llevar a cabo la implementación de AM en GNU Radio, iniciamos con la creación de un nuevo proyecto, el cual trae consigo dos bloques de forma predeterminada, el primero llamado “options”, a través del cual se realiza la configuración del proyecto, en el cual se puede modificar el nombre, autor, título, entre otras opciones. El segundo es una variable nombrada “samp_rate”, esta nos permite configurar la frecuencia de muestreo a utilizar durante la simulación.

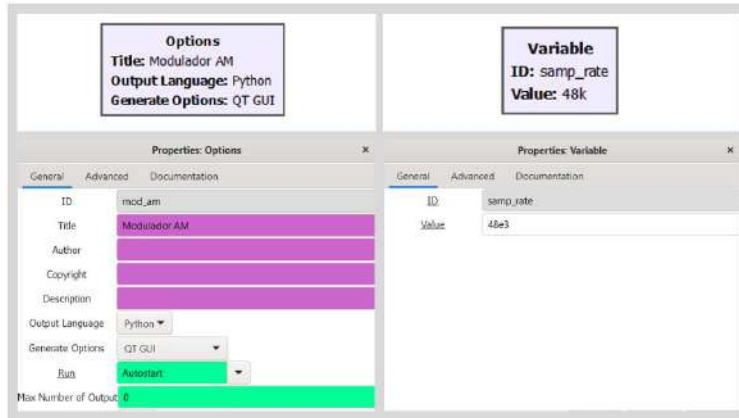


Figura 41. Bloques Options y Variable con sus configuraciones en GNU Radio. Fuente propia.

Posteriormente realizamos los cambios en los valores por defecto que contienen estos bloques, haciendo doble clic sobre cada uno y editando los campos, tal como se muestra en la figura 41. Cabe aclarar que el valor de “samp_rate” fue seleccionado teniendo en cuenta el teorema Nyquist y que en AM las frecuencias presentes en la señal modulada son la frecuencia de portadora (f_c) y las frecuencias de las bandas laterales, siendo la suma y diferencia de la frecuencia de portadora y moduladora ($f_m - f_c$ y $f_c + f_m$), por lo que, el ancho de banda y la frecuencia de muestreo mínima deberían ser:

$$f_N > 2f_{max}, \quad f_c = 10 \text{ KHz}, \quad f_m = 200 \text{ Hz}$$

$$f_N > f_c + f_m \rightarrow f_N > 2(10 \text{ KHz} + 200 \text{ Hz}) \rightarrow f_N > 20.4 \text{ KHz}$$

$$BW = (f_c + f_m) - (f_c - f_m) = (10 \text{ KHz} + 200 \text{ Hz}) - (10 \text{ KHz} - 200 \text{ Hz})$$

$$BW = 10.2 \text{ KHz} - 9.8 \text{ KHz} = 400 \text{ Hz}$$

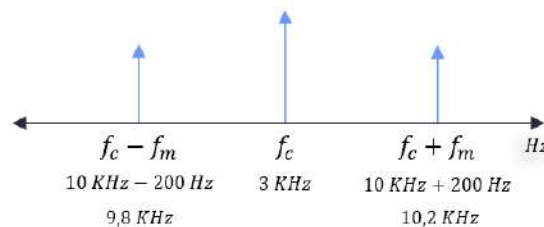


Figura 42. Ancho de banda para modulador AM. Fuente propia.

Una vez configurados los bloques anteriores, se añadieron los bloques necesarios para conformar el esquema del modulador de amplitud, desde el panel ubicado en la parte derecha de la pantalla. En este se pueden realizar búsquedas de dos formas, la primera es haciendo clic sobre el campo de texto ubicado en la parte superior y luego ingresar el nombre del bloque (figura 43) y la segunda forma, es navegando entre las distintas librerías que trae consigo GNU Radio.

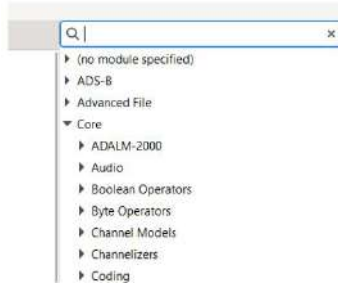


Figura 43. Panel de búsqueda en GNU Radio. Fuente propia

Este esquema de modulación puede ser utilizado para transmitir tonos o señales de audio, siendo en este caso utilizado para la transmisión de un tono, utilizando un generador de señales (*Signal_source*) para generar tanto la señal moduladora, como la señal portadora. La señal moduladora fue configurada como una onda cosenoidal con una frecuencia inicial de 200 Hz, ajustable entre 150 y 300 Hz mediante el bloque *QT GUI Dial*, así mismo, la señal portadora fue establecida como una onda cosenoidal cuya frecuencia puede variar entre 9 KHz y 11 KHz, siendo 10 KHz, el valor establecido como predeterminado (figura 44, figura 45 y figura 46). Además, para visualizar las variaciones de los parámetros anteriormente mencionados, se utilizó un *QT GUI Level Gauge* y así monitorear estos mismos.

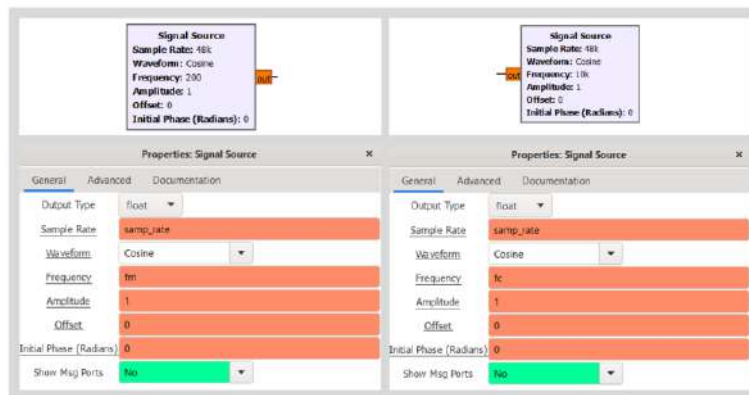


Figura 44. Bloques *Signal Source* y sus configuraciones para AM en GNU Radio (señal moduladora y portadora) en un entorno simulado. Fuente propia

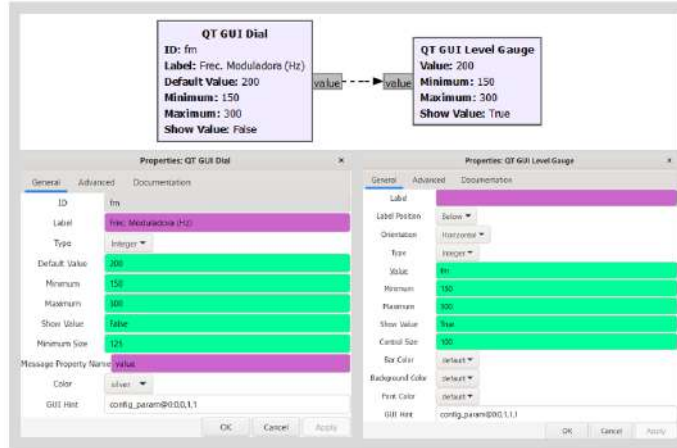


Figura 45. Bloques QT GUI Dial y QT GUI Level Gauge y sus configuraciones para la variación y visualización de la frecuencia de moduladora para AM en GNU Radio en un entorno simulado. Fuente propia



Figura 46. Bloques QT GUI Dial y QT GUI Level Gauge y sus configuraciones para la variación y visualización de la frecuencia de portadora para AM en GNU Radio en un entorno simulado. Fuente propia.

En este caso realizamos la implementación del esquema, aplicando un índice de modulación de 0.5, para esto, tomamos la señal proveniente del generador que representa la señal moduladora y la pasamos a través del bloque *Multiply Const*, el cual fue configurado con un valor ajustable a través de un bloque QT GUI Dial, siendo 0.5 el valor inicial, pero pudiendo variar entre un rango de 0.1 y 1.5 (figura 47, figura 48, figura 49 y figura 50).

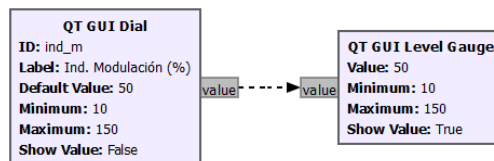


Figura 47. Bloque QT GUI Dial y QT GUI Level Gauge para la variación del índice de modulación en AM en GNU Radio en un entorno simulado. Fuente propia.



Figura 48. Configuración del bloque QT GUI Dial para la variación del índice de modulación en AM en GNU Radio. Fuente propia.



Figura 49. Configuración del bloque QT GUI Level Gauge para la variación del índice de modulación en AM en GNU Radio. Fuente propia.

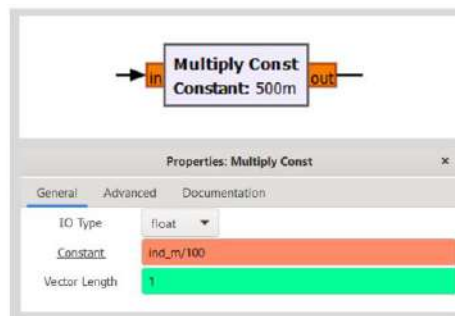


Figura 50. Bloque Multiply Const y su configuración para AM en GNU Radio en un entorno simulado. Fuente propia.

$$m = \frac{A_m}{A_c} = \frac{0.5}{1} = 0.5$$

Una vez acondicionada la señal moduladora, se le suma la amplitud de la señal portadora a través del bloque *Add Const* (figura 51), configurado con un tipo de dato punto flotante y un valor de 1, desplazando la onda sinusoidal en sus cruces por cero.

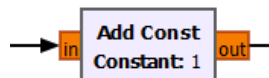


Figura 51. Bloque Add Const para AM en GNU Radio en un entorno simulado. Fuente propia

Para obtener finalmente la señal modulada en amplitud, se utilizó el bloque *Multiply*, siendo este un bloque capaz de multiplicar dos o más señales que ingresen a la entrada de este mismo.

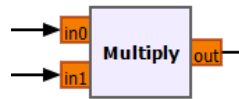


Figura 52. Bloque Multiply para AM en GNU Radio en un entorno simulado. Fuente propia

La señal obtenida a la salida del *Multiply* pasa a la entrada denominada como “re” del bloque *Float To Complex*, convirtiendo el flujo de datos de valores punto flotante a valores de tipo complejo, por lo que, en la entrada “im” de este mismo, se ingresó la salida de un bloque *Constant Source* configurado con un valor de 0 (figura 53).

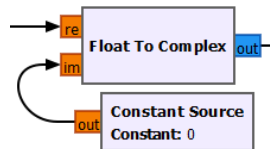


Figura 53. Bloque Float To Complex y Constant Source para AM en GNU Radio en un entorno simulado. Fuente propia

Una vez cambiado el tipo de dato, la señal es ingresada a un bloque *Rational Resampler*, configurado para incrementar la tasa de muestreo, interpolando cada muestra 40 veces, elevando la tasa de 48 KHz a 1.92 MHz, así mismo un valor de Fractional BW de 0.4, siendo este último crucial para evitar la generación de imágenes no deseadas o aliasing. Esta interpolación se hace necesaria para cumplir con la tasa de muestreo mínima de 1Msps que requiere el SDR transmisor.

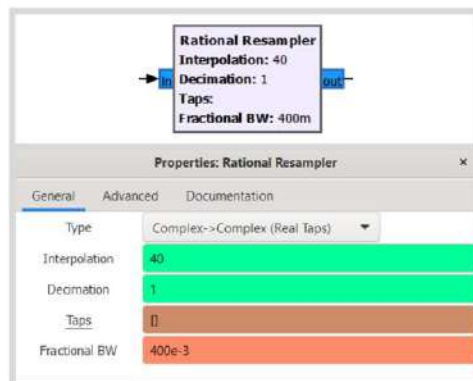


Figura 54. Bloque Rational Resampler y su configuración para AM en GNU Radio en un entorno simulado. Fuente propia.

Para la visualización de las señales tanto en el dominio del tiempo como en el dominio de la frecuencia, se utilizó un conjunto de bloques, como QT GUI Time Frequency y QT GUI Time Sink, además de los bloques QT GUI Dial y QT GUI Level Gauge para la variación de valores. La configuración del parámetro GUI Hint junto con el bloque QT GUI Tab Widget, permitieron organizar cada elemento de forma matricial en la interfaz, aplicando un formato donde se define la fila y columna inicial, además del número de filas y columnas a ocupar en la pantalla, facilitando una distribución estructurada de los controles en distintas secciones, tal como se muestra en la figura 55, figura 56, figura 57, figura 58 y figura 61.

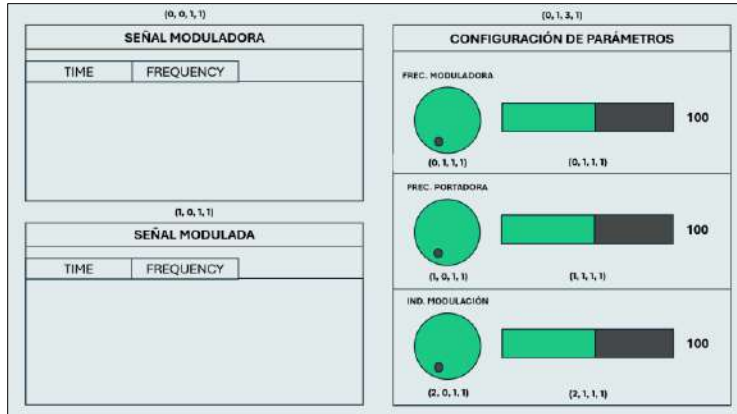


Figura 55. Bosquejo de interfaz de modulador. Fuente propia

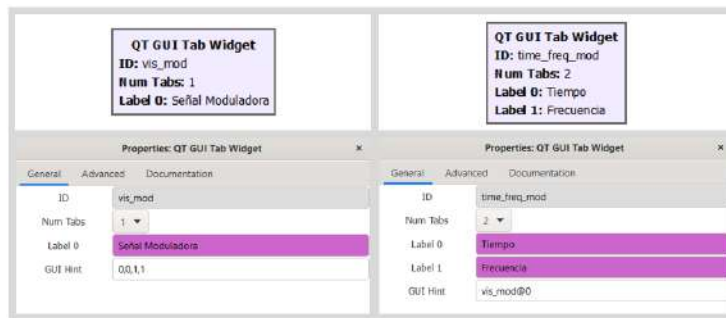


Figura 56. Bloques QT GUI Tab Widget y su configuración para la sección de visualización de la señal moduladora en GNU Radio. Fuente propia

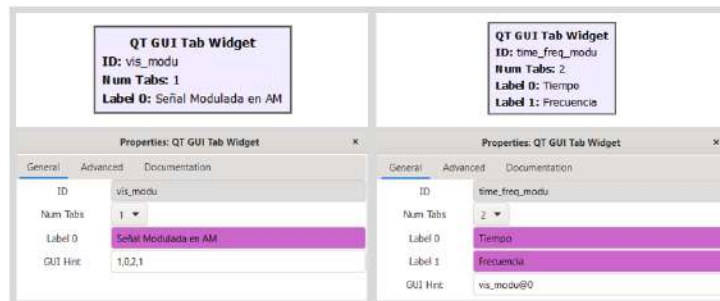


Figura 57. Bloques QT GUI Tab Widget y su configuración para la sección de visualización de la señal modulada en GNU Radio. Fuente propia

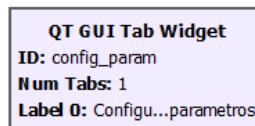


Figura 58. Bloque QT GUI Tab Widget para la sección de configuración de parametros de modulación en GNU Radio. Fuente propia



Figura 59. Configuración bloque QT GUI Tab Widget para la sección de configuración de parámetros de modulación en GNU Radio. Fuente propia

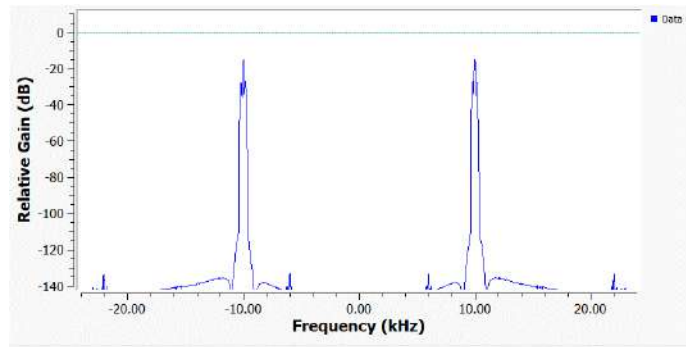


Figura 60. Espectro en frecuencia de señal modulada en AM en GNU Radio en un entorno simulado. Fuente propia

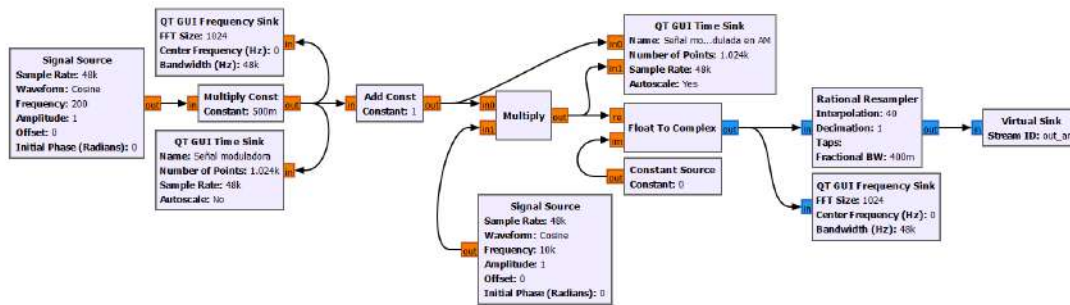


Figura 61. Esquema final de modulador AM para implementación simulada en GNU Radio. Fuente propia

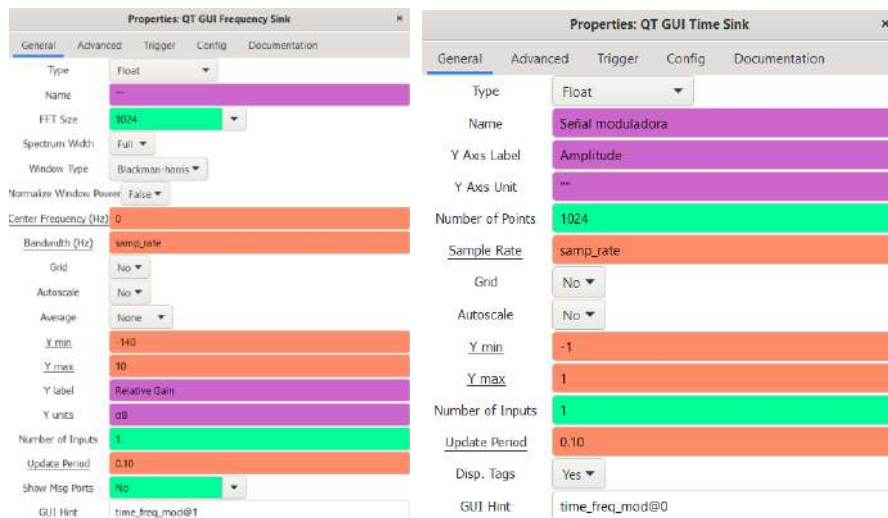


Figura 62. Configuración de los bloques QT GUI Time Sink y QT GUI Frequency Sink para AM en GNU Radio. Fuente propia

8.7.1.2. Desarrollo del esquema demodulador en GNU Radio

En el receptor AM, se aplicó el método de detección de envolvente, siendo este, uno de los métodos más sencillos y comunes, capaz de extraer la onda envolvente, la cual hace referencia a la onda original, la que contiene la información.



Figura 63. Esquema de receptor AM por detección de envolvente. Fuente: Basado en Tomasi, 2003.

Esta implementación fue realizada en un entorno simulado inicialmente, tomamos la señal proveniente del bloque *Rational Resampler* del modulador y se le aplicó un proceso de conversión, pasando de un tipo de dato complejo a un tipo de dato punto flotante, siendo este último, indispensable para el proceso de demodulación (figura 64).



Figura 64. Bloque Complex To Float para demodulación AM en GNU Radio en un entorno simulado. Fuente propia.

A continuación, se realizó la rectificación con el bloque *Rail*, permitiendo el paso únicamente de aquellos valores de la señal que sean mayores a cero y menores a 5, tal como se muestra figura 65, donde se presenta tanto el bloque como su configuración.

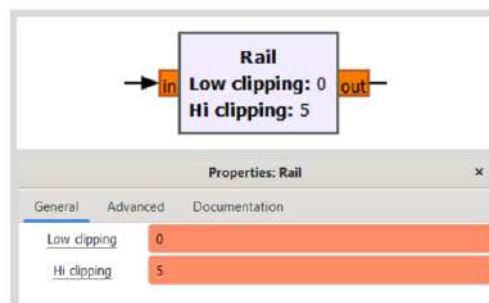


Figura 65. Bloque Rail y su configuración para AM en GNU Radio. Fuente propia.

Seguidamente, se utilizó el bloque *Rational Resampler* con un factor de decimación de 40, a fin de contrarrestar los efectos del bloque *Rational Resampler* utilizado en el modulador, así mismo, con un Fractional BW de 400m.

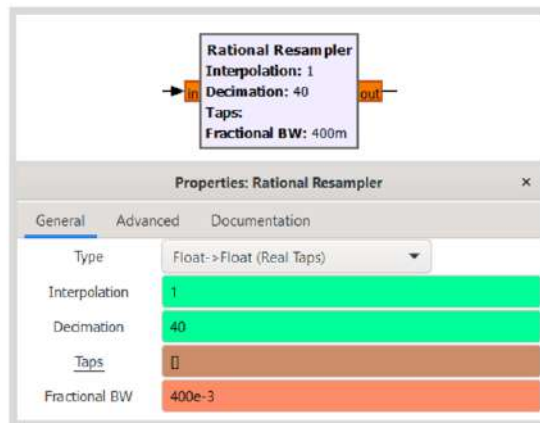


Figura 66. Bloque Rational Resampler y su configuración para AM en GNU Radio. Fuente propia.

En una última instancia, se aplicó un filtro pasa bajo a la señal rectificadora, a fin de eliminar las componentes de alta frecuencia, dejando únicamente la señal envolvente. Para ello, se utilizó el bloque *Low Pass Filter*, configurado una frecuencia de corte inicial de 200 Hz (frecuencia de moduladora), un Transition Width de 10 y dejando el resto de los parámetros por defecto, sin embargo, el valor de la frecuencia de corte puede variar entre 150 Hz y 300 Hz a través de un bloque QT GUI Dial y QT GUI Level Gauge para la visualización de estos cambios.



Figura 67. Bloque Low Pass Filter y su configuración para AM en GNU Radio. Fuente Propia.

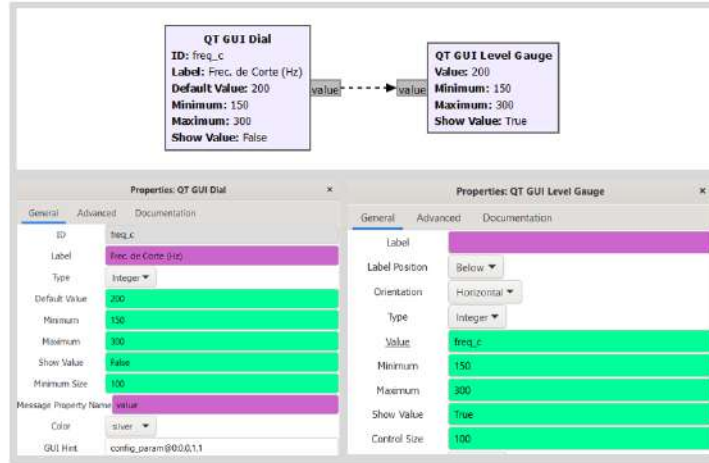


Figura 68. Bloques QT GUI Dial y QT GUI Level Gauge para la variación de la frecuencia de corte en la demodulación AM en GNU Radio en un entorno simulado. Fuente propia.

Para la visualización en el dominio del tiempo tanto de la señal modulada en AM como la demodulada mediante el detector de envolvente, se hizo uso de los bloques QT GUI Widget y QT GUI Time Sink, además de los parámetros “GUI Hint” de estos mismo, tal como se aplicó en el modulador AM (figura 69, figura 70, figura 71 y figura 72).

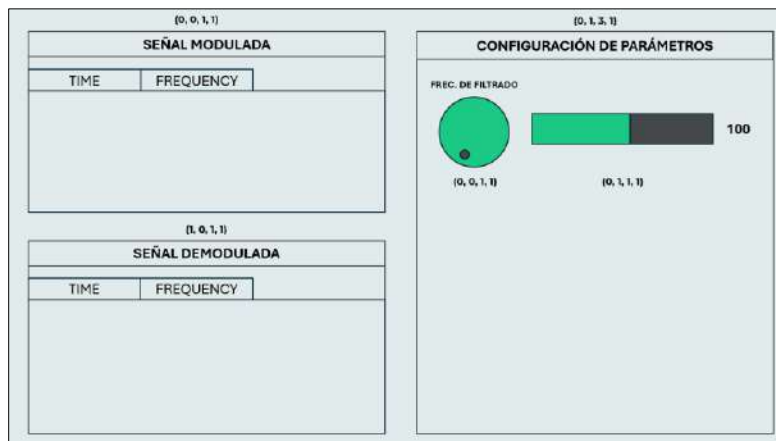


Figura 69. Bosquejo de interfaz de demodulador. Fuente propia

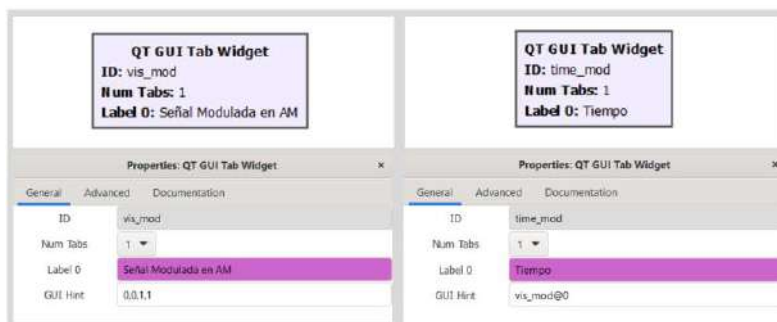


Figura 70. Bloques QT GUI Widget y su configuración para la visualización de la señal modulada en AM en el dominio del tiempo en GNU Radio. Fuente propia.

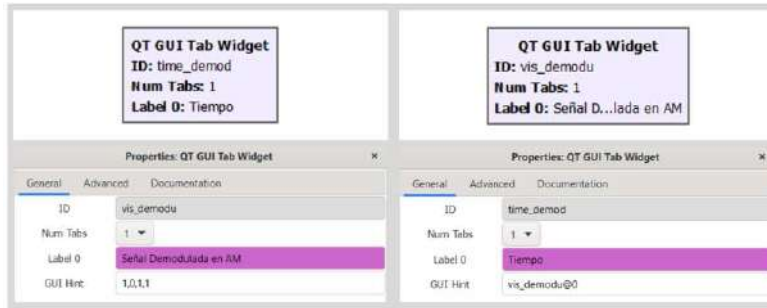


Figura 71. Bloques QT GUI Widget y su configuración para la visualización de la señal demodulada en AM en el dominio del tiempo en GNU Radio. Fuente propia.

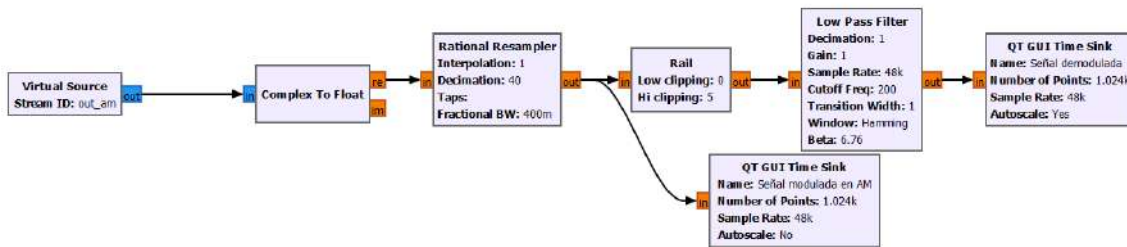


Figura 72. Esquema final de demodulador AM en GNU Radio en un entorno simulado. Fuente propia.

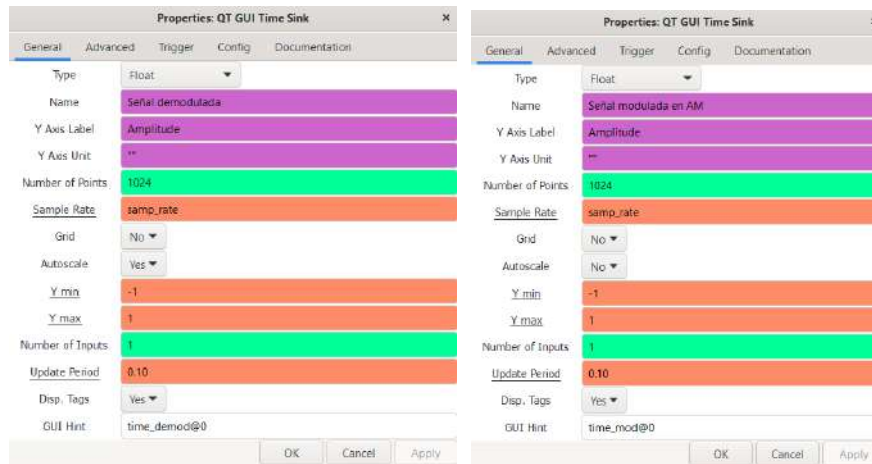


Figura 73. Configuración de los bloques QT GUI Time Sink para la visualización en el dominio del tiempo de la señal demodulada en AM (izquierda) y la señal modulada en AM (derecha) en GNU Radio. Fuente propia.

8.7.1.3. Implementación del modulador en GNU Radio usando los dispositivos SDR

En esta sección se describe la implementación práctica del modulador de amplitud utilizando GNU Radio y el dispositivo SDR anteriormente seleccionado (HackRF One), teniendo en cuenta que anteriormente, fue presentada de forma detallada la instalación de los controladores necesarios para que el aplicativo reconozca los dispositivos SDR, por lo que, la transición de un entorno de simulación a la implementación real se ve facilitada.

Para adaptar el esquema del modulador desarrollado a una implementación práctica utilizando el dispositivo SDR, se añadió un bloque clave: *Osm ocom Sink*, siendo este, el bloque encargado de comunicarse con el hardware SDR transmisor (HackRF One) y así transmitir la señal que es conectada a su entrada, la cual es la señal obtenida a la salida del bloque *Rational Resampler*. Por otro lado, con respecto a la configuración de este mismo, solamente fueron modificados ciertos parámetros, los cuales fueron:

- **Device Arguments:** hackfr=0, indicando que el dispositivo a seleccionar es el HackRF One, no obstante, en el caso de tener más de un HackRF One conectados de forma simultánea, el valor podría variar, pudiendo seleccionar uno u otro. En este caso, el valor es cero, puesto que se tiene solamente un dispositivo conectado.
- **Sample Rate (sps):** 1.92 M. Este parámetro representa la cantidad de muestras por segundo que tomará el dispositivo SDR, por lo tanto, dicho valor fue seleccionado para aumentar el número de muestras por segundo, y así obtener una mayor fidelidad de la señal en el modulador.
- **Ch0: Frequency (Hz):** 930 M. La frecuencia de transmisión es configurada a través de este parámetro, siendo 530 kHz a 1710 kHz el rango de frecuencias en AM, sin embargo, se utilizó 930 MHz, puesto que, en ese rango de frecuencia el espectro se encuentra altamente saturado, provocando demasiada interferencia y ruido en la señal transmitida.
- **Ch0: RF Gain (dB):** 24. La ganancia de RF es un parámetro primordial en la radiodifusión, puesto que, permite configurar la intensidad de la señal en la transmisión, y seleccionar un valor muy alto, nos incurrirá en un incremento tanto de la señal como del ruido adicionado a esta por el entorno, por lo que, en base a múltiples pruebas realizadas, un valor adecuado es el mencionado.
- **Ch0: IF Gain (dB):** 28. Este parámetro representa la ganancia de la frecuencia intermedia, por lo que, se vuelve factor crucial para aumentar la calidad de la señal en el receptor. Ahora bien, en función de las pruebas realizadas un valor adecuado, sería el establecido, permitiendo una calidad de la señal aceptable.

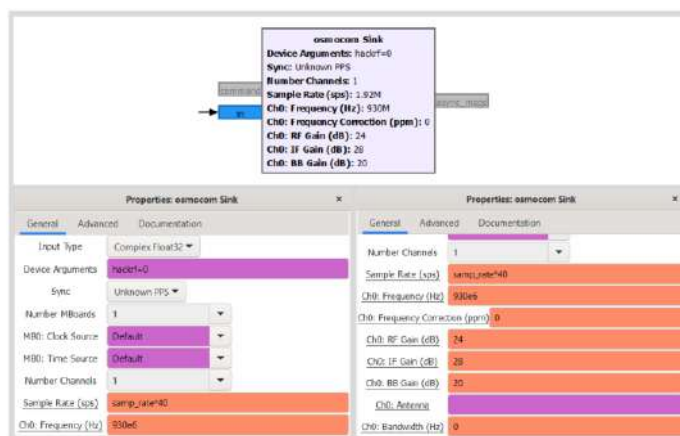


Figura 74. Bloque *osm ocom Sink* y su configuración para AM en GNU Radio. Fuente propia.

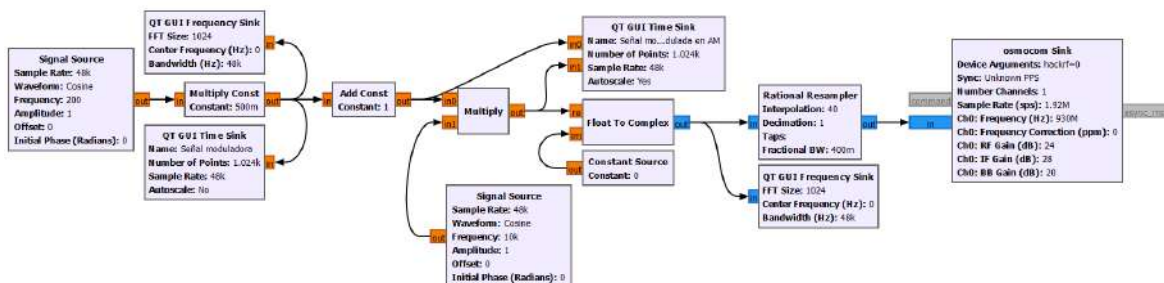


Figura 75. Esquema final de modulador AM en GNU Radio usando SDR. Fuente propia.

8.7.1.4. Implementación del demodulador en GNU Radio usando los dispositivos SDR

Ahora bien, para llevar a cabo la implementación práctica del demodulador de amplitud, al igual que en el modulador, se hizo uso una adaptación del esquema desarrollado en la etapa de simulación, por lo que, se vio facilitada en gran medida la transición de un entorno simulado a uno real.

Para esto, se adicionó un bloque de gran importancia, el cual permitiría tomar la señal captada a través del dispositivo SDR receptor (Nooelec NESDR SMARt V5) y posteriormente demodular esta misma, dicho bloque es el *Osm ocom Source*. La adaptación del esquema inicia con la configuración del bloque *Osm ocom Source*, conservando algunos valores predeterminados y modificando otros con base a los configurados en el modulador, tal como se muestra a continuación:

- **Device Arguments:** rtl=0, indicando que el dispositivo a seleccionar es el Nooelec NESDR SMARt V5, no obstante, al igual en el transmisor, al tener más de un RTL conectado de forma simultánea, el valor podría variar.
- **Sample Rate (sps):** 1.92M
- **Ch0: Frequency (Hz):** 930.048 M. La frecuencia de recepción es configurada a través de este parámetro, siendo el valor establecido en donde se encuentra una de las bandas laterales.
- **Ch0: RF Gain (dB):** 24

- Ch0: IF Gain (dB): 28

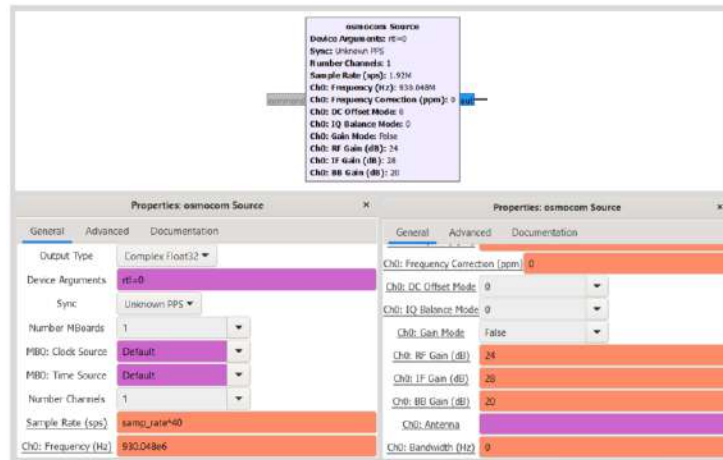


Figura 76. Bloque Osm ocom Source y su configuración para AM en GNU Radio. Fuente propia

La señal compleja obtenida a la salida del bloque receptor, es ingresada al bloque *Complex to Float*, transformando la señal de complejo a punto flotante, en donde esta última, se encuentra compuesta de una parte real y una imaginaria (puerto “re” e “imag”). No obstante, solamente se toma la parte real de señal, y es demodulada mediante el detector de envolvente desarrollado en la etapa de simulación, finalizando con el esquema presentado en la figura 78.



Figura 77. Bloque Complex To Float para AM en GNU Radio. Fuente propia

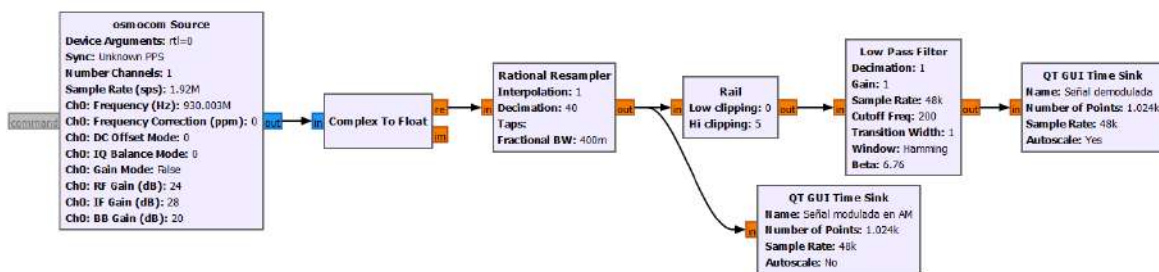


Figura 78. Esquema final de demodulador AM en GNU Radio usando SDR. Fuente propia

8.7.2. Modulación de frecuencia (FM)

La técnica de modulación conocida como frecuencia modulada (FM) consiste en que la frecuencia de la señal portadora, de amplitud constante, sigue las variaciones de amplitud de la señal moduladora. Esta técnica se introdujo como alternativa a la modulación de amplitud (AM), debido principalmente a que ofrece una menor afectación ante las perturbaciones, por lo que es un sistema más eficiente.

8.7.2.1. Desarrollo del esquema modulador en GNU Radio

El esquema modulador está compuesto inicialmente por la señal moduladora, una señal sinusoidal generada mediante el bloque Signal Source, configurado con una frecuencia moduladora de $f_m = 2.5 \text{ KHz}$ y una amplitud de $A = 1$.

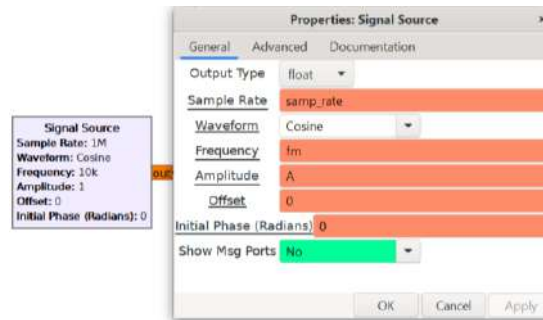


Figura 79. Bloque Signal Source y su configuración utilizada para generar la señal moduladora en FM. Fuente propia.

Para el diseño del modulador FM, es posible realizar la variación de la frecuencia moduladora mediante los bloques QT GUI Dial y QT GUI Level Gauge, que permiten ajustar el valor de dicha frecuencia entre 10 KHz y 25 KHz.

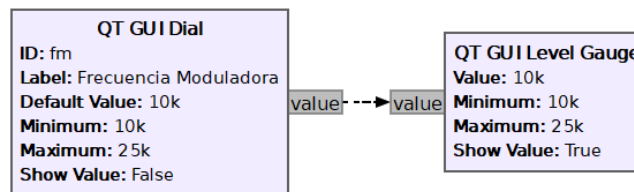


Figura 80. Bloques QT GUI Dial y QT GUI Level Gauge – Ajuste de frecuencia moduladora. Fuente propia

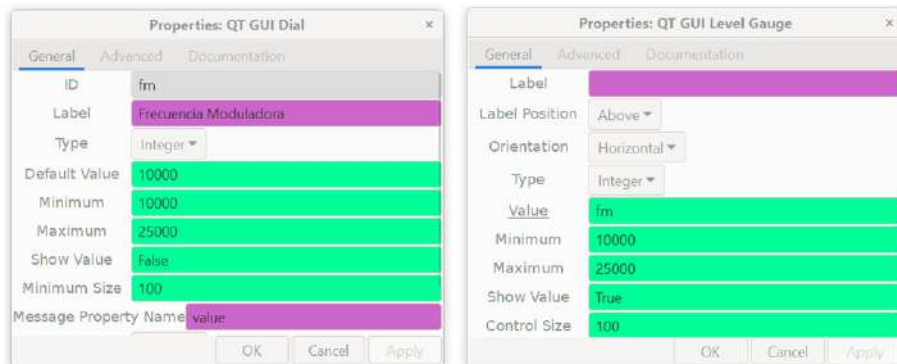


Figura 81. Configuración de parámetros de los bloques QT GUI Dial y QT GUI Level Gauge – Frecuencia moduladora. Fuente propia

Para realizar la modulación FM de forma directa, la frecuencia de la señal modulada debe variar proporcionalmente a la amplitud de la señal moduladora. Para lograrlo, se utiliza un oscilador controlado por tensión a través del bloque VCO (*complex*), que produce una senoide cuya frecuencia depende de la amplitud de la señal de entrada y de la desviación de frecuencia. En este diseño, se seleccionó una desviación de frecuencia de 75 KHz, valor comúnmente utilizado para FM comercial

de banda ancha. Entre los parámetros ajustables del bloque *VCO (complex)* se encuentran:

Sample Rate: Mantiene el valor definido por defecto en la variable *samp_rate*, fijado en 1 MHz.

Sensitivity: Permite ajustar el cambio en la frecuencia en respuesta a un cambio en la amplitud de la señal de entrada, definido por la expresión:

$$k = \frac{2\pi \cdot \Delta f}{A} \leftrightarrow \frac{2\pi \cdot 75 \text{ KHz}}{1} = 471.238 \text{ Krad/s/v}$$

A → Amplitud máxima de la señal de entrada *Δf* → Variación de frecuencia

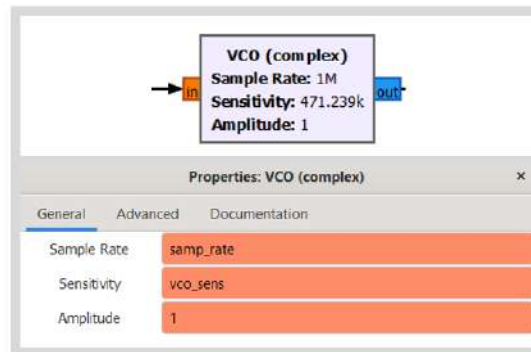


Figura 82. Configuración del bloque *VCO(Complex)*, la variable *vco_sens* almacena el valor de *k*. Fuente propia

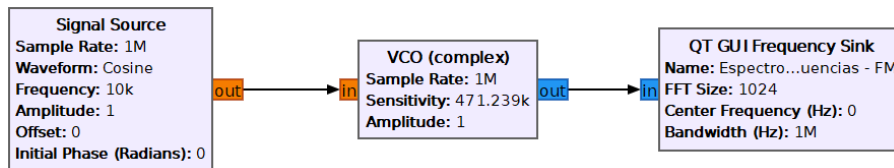


Figura 83. Esquema modulador FM. Fuente propia

Tomando en cuenta los parámetros de diseño mencionados anteriormente, se puede calcular el índice de modulación para FM. Este parámetro representa la desviación máxima de fase, en radianes, de una portadora con fase modulada y está definido por la expresión:

$$m = \frac{\Delta f}{f_m} \leftrightarrow \frac{75 \text{ KHz}}{10 \text{ KHz}} = 7.5$$

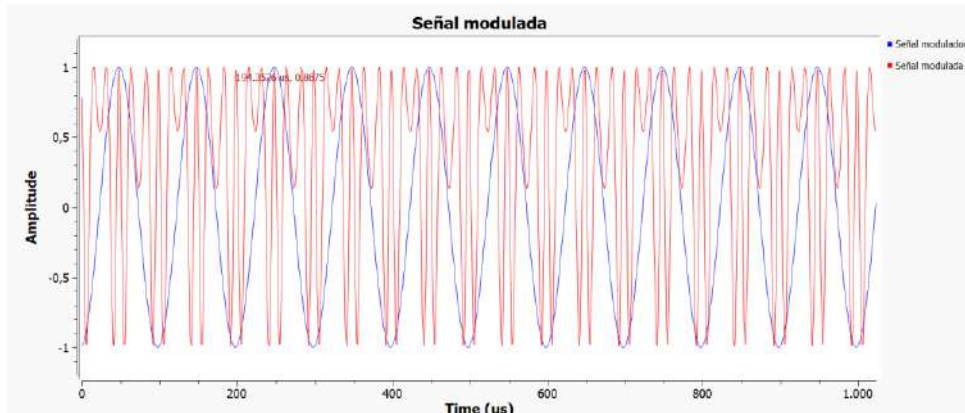


Figura 84. Señal modulada en FM con el VCO(Complex) Vs Señal moduladora. Fuente propia

En la figura 84 se observa la variación de la frecuencia de la señal generada internamente por el VCO (*complex*) en función de la amplitud en valor absoluto de la señal de entrada, esta variación se manifiesta como un aumento de la frecuencia en ambos semiciclos de la señal modulada, mientras que se reduce cuando la señal de entrada cruza el valor cero.

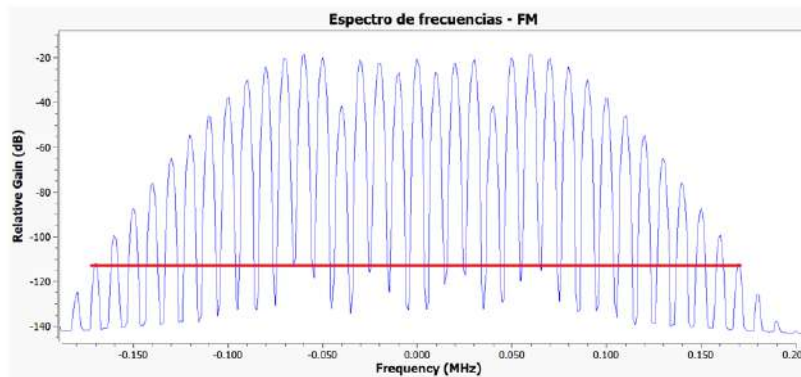


Figura 85. Espectro de frecuencia de la modulación FM. Fuente propia.

El ancho de banda para la modulación FM se puede calcular mediante la regla de Carson, una regla general propuesta por J. R. Carson para estimar el ancho de banda de todos los sistemas con modulación angular. Esta regla indica que el ancho de banda necesario para transmitir una onda modulada en FM es:

$$B = 2(\Delta f + f_m) \rightarrow 2(75 \text{ KHz} + 10 \text{ KHz}) = 170 \text{ KHz}$$

8.7.2.2. Desarrollo del esquema demodulador en GNU Radio

Para realizar el proceso de recepción de datos, se opta por utilizar el bloque *Quadrature Demod*, que se encuentra en la librería de Modulators y está diseñado específicamente para llevar a cabo la demodulación de frecuencia por cuadratura.

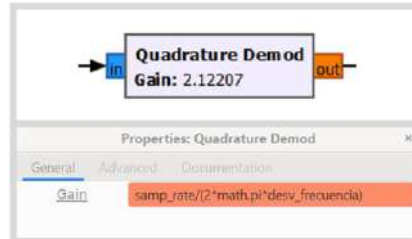


Figura 86. Bloque Quadrature Demod y su configuración para FM. Fuente propia

Internamente, el bloque *Quadrature Demod* determina la fase instantánea de la señal en cuadratura para luego calcular la diferencia de fase entre muestras consecutivas. Esta diferencia de fase es proporcional a la desviación de frecuencia introducida por la señal de información. Finalmente, el bloque aplica un factor de ganancia ajustado por el parámetro:

$$Gain = \frac{sample_rate}{(2\pi \cdot \Delta f)} \leftrightarrow \frac{1 \text{ MHz}}{2\pi \cdot 75 \text{ KHz}} \approx 2.122$$

Este factor escala el resultado a un rango que representa adecuadamente la señal de información.

Para finalizar la demodulación, se emplea un filtro pasa bajas FFT Low Pass Filter, que realiza una transformada rápida de Fourier para luego aplicar el filtro en el dominio de la frecuencia, este ofrece una mayor precisión al limpiar la señal demodulada.



Figura 87. Filtro FTT Low Pass Filter y su configuración para FM en GNU Radio. Fuente propia.

El parámetro Cutoff Freq es variable, ya que fue asignado a una variable controlada mediante los bloques *QT GUI Dial* y *QT GUI Level Gauge*, los cuales permiten ajustar el valor de la frecuencia de corte en un rango de 10 KHz a 25 KHz. Se estableció un valor predeterminado de 10.5 KHz para obtener la señal de información para este ejercicio.

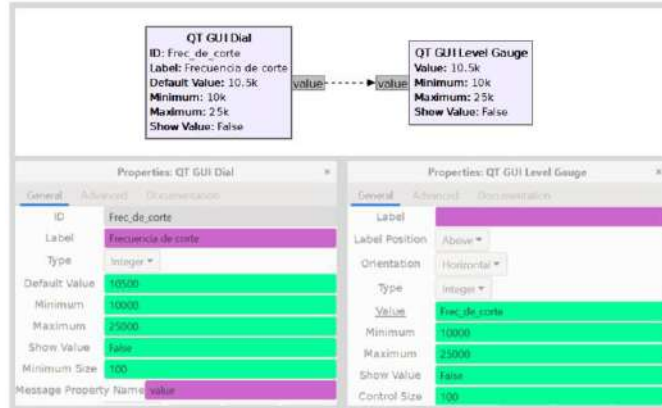


Figura 88. Bloques QT GUI Dial y QT GUI Level Gauge y su configuración para ajuste de frecuencia de corte del filtro en FM. Fuente propia.

En base a las especificaciones descritas previamente, se presenta el esquema final, el cual integra todos los bloques para llevar a cabo la demodulación en FM de forma exitosa.

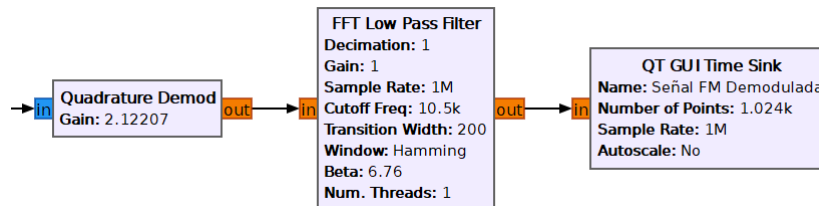


Figura 89. Esquema final de demodulador FM en GNU Radio. Fuente propia.

8.7.2.3. Implementación del modulador en GNU Radio usando los dispositivos SDR

Para llevar a cabo la implementación en un entorno real y hacer uso del hardware SDR en el esquema modulador FM, se requiere adicionar al esquema desarrollado en el entorno simulado, el bloque *osmocom Sink*, debido a que la señal obtenida a la salida del bloque *VCO (complex)* ya es de tipo complejo, no es necesario añadir un bloque convertidor; sin embargo, es necesario ajustar los siguientes parámetros:

- **Device Arguments:** hackrf=0
- **Sample Rate (sps):** 2.5 M
- **Ch0: Frequency (Hz):** La frecuencia de transmisión de la FM comercial en Colombia está en el rango desde los 87,5 MHz hasta los 108 MHz, por lo que se escogió una frecuencia de transmisión de 87 MHz con el fin de evitar interferencias.
- **Ch0: RF Gain (dB):** 24

- **Ch0: IF Gain (dB): 28**

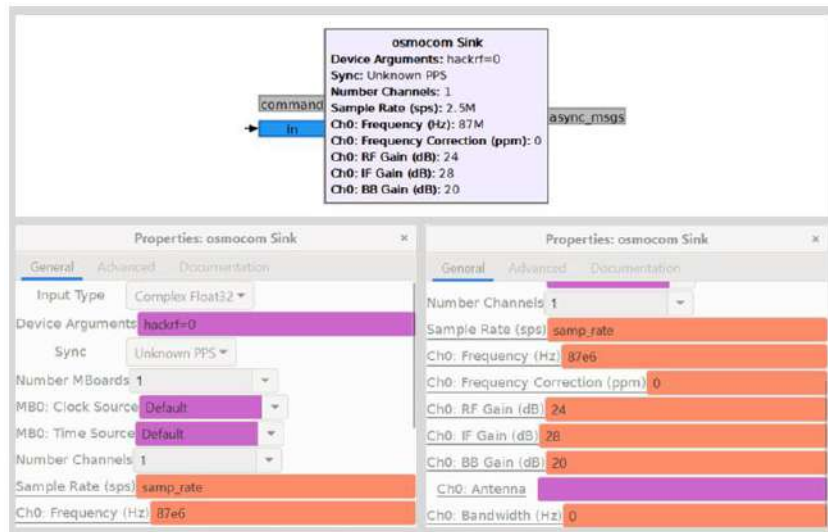


Figura 90. Bloque osmocomblock Sink y su configuración para transmisor FM. Fuente propia.

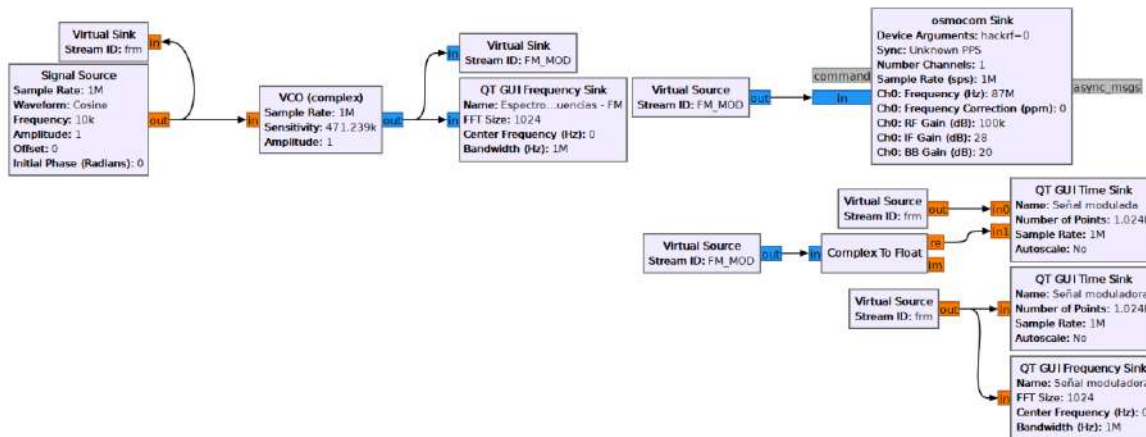


Figura 91. Esquema modulador FM con bloque transmisor osmocomblock Sink. Fuente propia.

8.7.2.4. Implementación del demodulador en GNU Radio usando los dispositivos SDR

Para la implementación del demodulador FM utilizando los dispositivos SDR, se empleó el mismo esquema desarrollado en la sección de simulación, pero al añadir el bloque *Osmocom Source* es posible captar la señal transmitida, configurado con los siguientes parámetros:

- **Device Arguments:** rtl=0
- **Sample Rate (sps):** 2.5 M
- **Ch0: Frequency (Hz):** 87 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

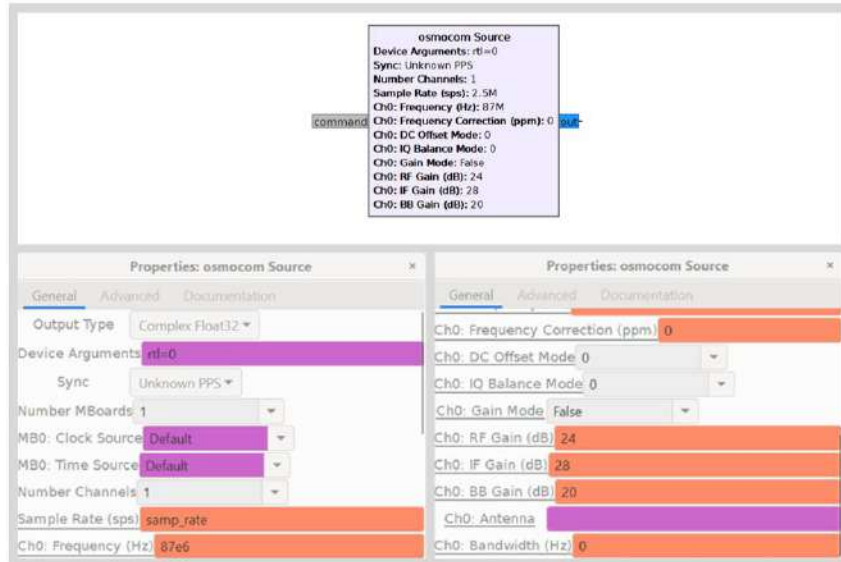


Figura 92. Bloque osmocom Source y su configuración para receptor FM. Fuente propia.

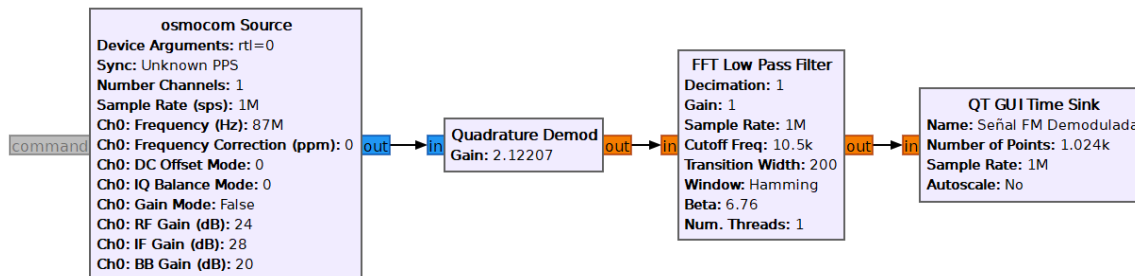


Figura 93. Esquema final de demodulador FM con bloque receptor osmocom Source en GNU Radio. Fuente propia.

8.7.3. Modulación por desplazamiento de amplitud (ASK)

Una de las técnicas de modulación digital que presenta mayor simplicidad es la modulación por desplazamiento de amplitud (ASK), siendo OOK (On-Off Keying), una de las formas más básicas y ampliamente utilizada en aquellos sistemas de comunicación de baja complejidad. En esta sección, nos centraremos en el desarrollo del esquema modulador y demodulador de OOK.

8.7.3.1. Desarrollo del esquema modulador en GNU Radio

Como se mencionó anteriormente, la modulación OOK se basa básicamente en la activación y desactivación de la señal portadora. Esto es debido a que la señal moduladora es digital, compuesta por secuencias de unos y ceros, que, al ser multiplicados con señal portadora, se logra la modulación OOK, tal como se muestra en la figura 94.

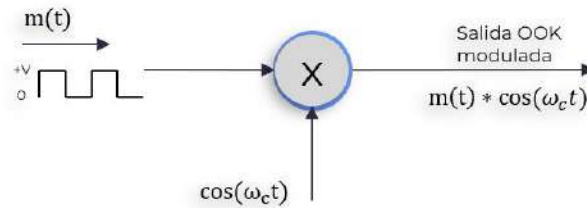


Figura 94. Esquema básico de un modulador ASK. Fuente: Basado en Tomasi, 2003.

Para la implementación del esquema modulador OOK en GNU Radio, se partió de la creación de un nuevo proyecto y configurando los bloques predeterminados de forma similar a las implementaciones anteriores. A partir de esto, se añadió un bloque *Vector Source*, el cual fue configurado para generar la secuencia de unos y ceros que representara la señal moduladora, así como se muestra a continuación:

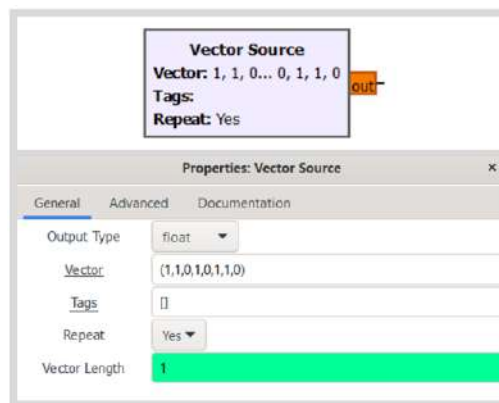


Figura 95. Bloque Vector Source para OOK en GNU Radio en un entorno simulado. Fuente: propia

Dentro de las configuraciones del bloque *Vector Source* aparece el campo para configurar la secuencia, y el tipo de datos que se desea obtener a la salida, siendo punto flotante el establecido en esta ocasión. No obstante, no existe un campo a través del cual se pueda configurar la frecuencia o la duración de cada bit, sin embargo, es posible controlar la duración de cada bit ajustando la tasa de muestreo, y usando el bloque *Repeat* para así repetir cada muestra del vector, alargando o reduciendo cada bit. Por lo tanto, el cálculo de la duración de bit y ancho de banda, es de la siguiente manera:

$$f_b(\text{frecuencia de bit}) = \frac{1}{T_b(\text{duracion de bit})} \rightarrow T_b = \frac{1}{40 \text{ KHz}} = 25 \text{ us}$$

$$T_b = \frac{\text{Número de muestras por bit}}{\text{Tasa de muestreo}}$$

$$\frac{1}{40 \text{ KHz}} = \frac{\text{Número de muestras por bit}}{\text{Tasa de muestreo}}$$

$$\text{Número de muestras por bit} = \frac{\text{Tasa de muestreo}}{40 \text{ KHz}} = \frac{2 \text{ MHz}}{40 \text{ KHz}} = 50$$

$$BW(\text{ancho de banda}) = 2 \cdot R_b = 2 \cdot (40 \text{ KHz}) = 80 \text{ KHz}$$



Figura 100. Bloque QT GUI Tab Widget y su configuración para la sección de configuración de parámetros en OOK en GNU Radio en un entorno simulado. Fuente propia.

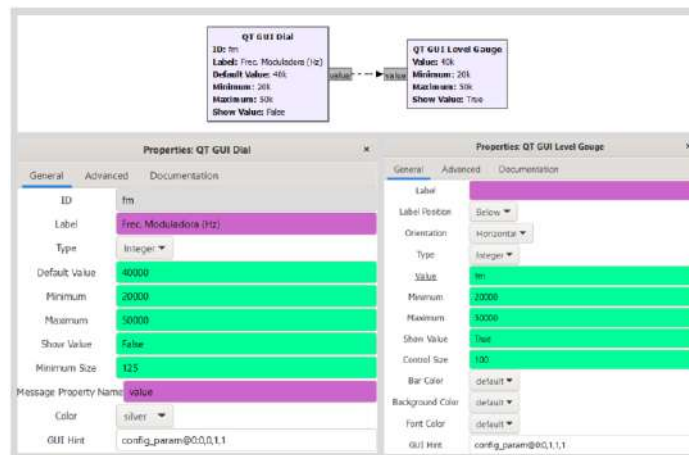


Figura 101. Bloques QT GUI Dial y QT GUI Level Gauge y su configuración para la variación y visualización de la frecuencia de moduladora en OOK en GNU Radio en un entorno simulado. Fuente propia.

Una vez establecido la tasa de bits, la señal obtenida es conectada a la entrada “re” del bloque convertidor de punto flotante a complejo (*Float To Complex*), esto con el fin de obtener el tipo de dato requerido por el bloque transmisor SDR, que, aunque en esta etapa no se lleva a cabo la implementación con el hardware, la señal ya se encontraría adecuada para la implementación con los dispositivos SDR.

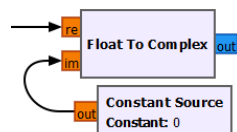


Figura 102. Conjunto de bloques de conversión de punto flotante a complejo (*Float To Complex* y *Constant Source*). Fuente propia.

La salida del bloque *Float To Complex* es conectada a un bloque *Virtual Sink*, este es un bloque a través del cual se hacen posible las conexiones inalámbricas dentro de GNU Radio al trabajar de forma conjunta con el bloque *Virtual Source*, por lo que, fueron configurados con el mismo valor de “Stream ID”, proporcionando dicha funcionalidad.

La señal obtenida a la salida del bloque *Virtual Source*, resulta ser la misma señal resultante del bloque *Float To Complex*, por lo tanto, es conectada a una de las dos

entradas del bloque *Multiply*, el cual a su vez recibe una señal portadora generada por un bloque *Signal Source*, siendo esta última, una onda cosenoidal de frecuencia de 300 KHz. Cabe mencionar que la frecuencia de portadora puede ser variada a través de un bloque QT GUI Dial, y un QT GUI Level Gauge, pudiendo variar entre 150 KHz y 400 KHz (figura 103).

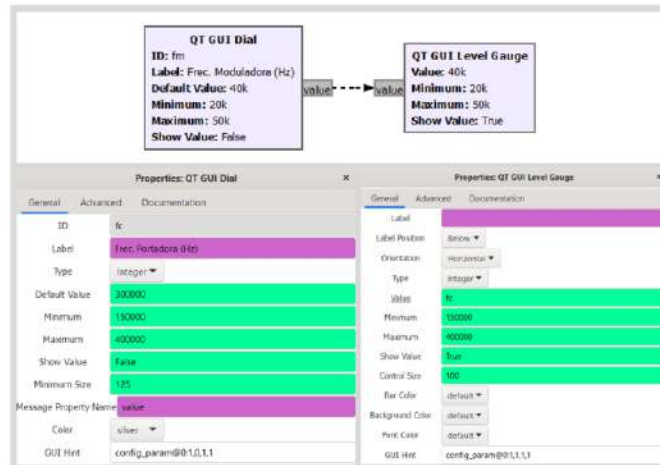


Figura 103. Bloques QT GUI Dial y QT GUI Level Gauge para OOK en GNU Radio en un entorno simulado. Fuente propia.

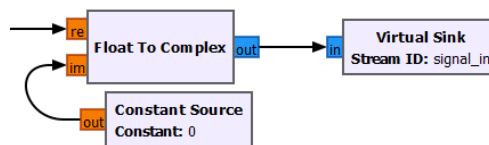


Figura 104. Conexión del bloque Float To Complex con el bloque Virtual Sink para OOK en GNU Radio en un entorno simulado. Fuente propia.

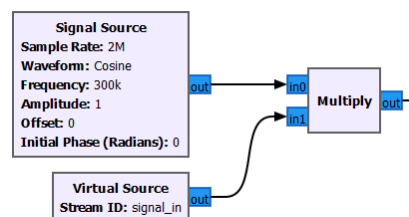


Figura 105. Multiplicación de la señal portadora con la moduladora adecuada para OOK en GNU Radio en un entorno simulado. Fuente propia.

Finalmente, se utilizaron los bloques QT GUI Widget, QT GUI Time Sink y QT GUI Frequency, para visualizar tanto la señal moduladora como la modulada en el dominio del tiempo y de la frecuencia, así mismo, se configuro el parámetro GUI Hint en cada uno de estos, para lograr obtener la interfaz presentada en las modulaciones anteriores (figura 106, figura 108, figura 110 y figura 112). Por otro lado, el bloque QT GUI Time Sink fue configurado en la pestaña “General” con un tipo de dato de entrada “complex” y dos puertos de entrada, así mismo en la pestaña “Config” se estableció el nombre correspondiente a cada señal que ingresa al bloque (“Señal moduladora” y “Señal modulada en OOK”). Cabe mencionar que las señales

que ingresan al bloque *QT GUI Time Sink* fueron conectadas mediante el uso de bloques *Virtual Sink* y *Virtual Source*.

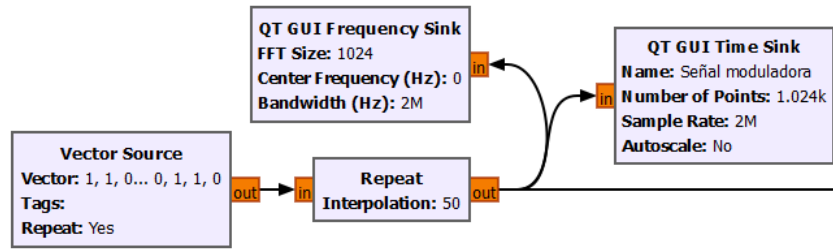


Figura 106. Bloques *QT GUI Frequency* y *QT GUI Time Sink* para visualización de señal moduladora en OOK en GNU Radio. Fuente propia

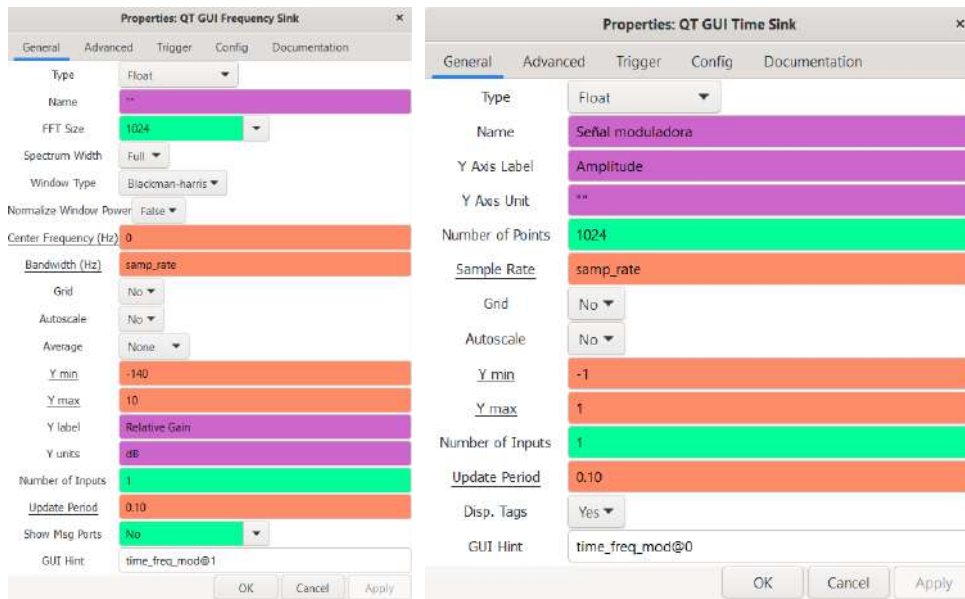


Figura 107. Configuración bloques *QT GUI Frequency* y *QT GUI Time Sink* para visualización de señal moduladora en OOK en GNU Radio. Fuente propia

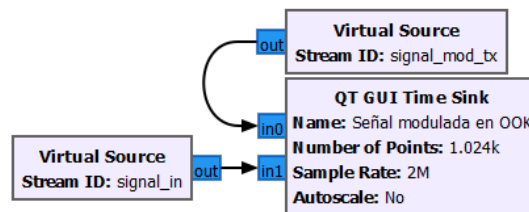


Figura 108. Conjunto de bloque para visualización de señales en Transmisor OOK (*QT GUI Time Sink*, *Virtual Sink* y *Virtual Source*). Fuente propia

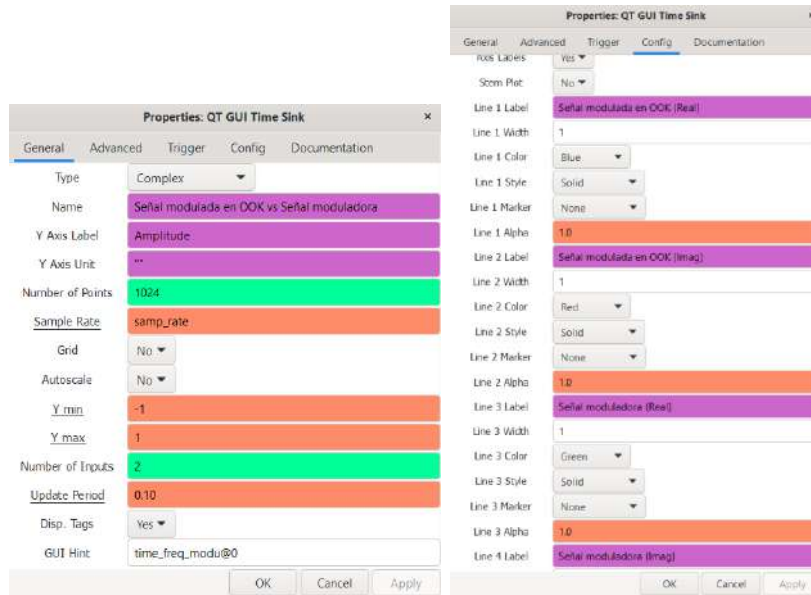


Figura 109. Configuración del bloque QT GUI Time Sink para visualización de señales OOK en GNU Radio. Fuente propia

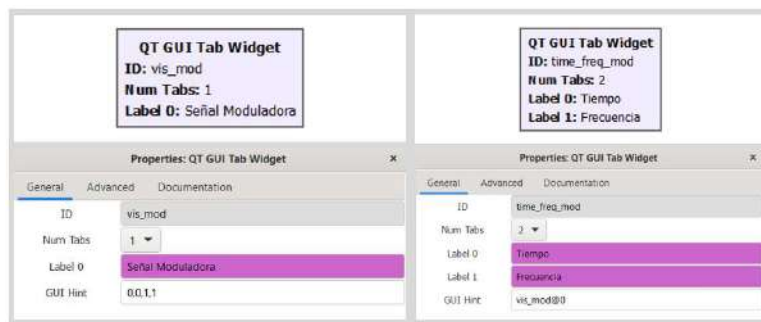


Figura 110. Bloques QT GUI Widget para la visualización de la señal moduladora en OOK en GNU Radio en un entorno simulado. Fuente propia

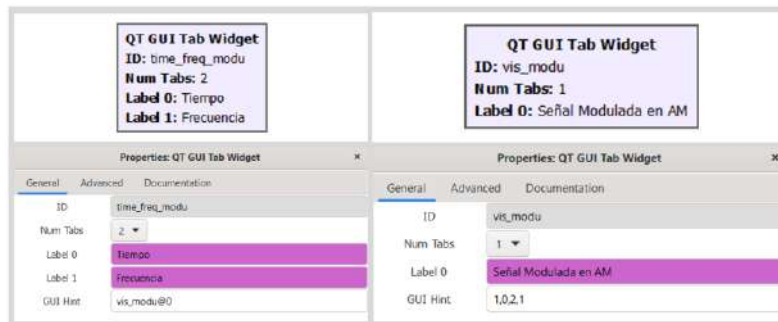


Figura 111. Bloques QT GUI Widget para la visualización de la señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.

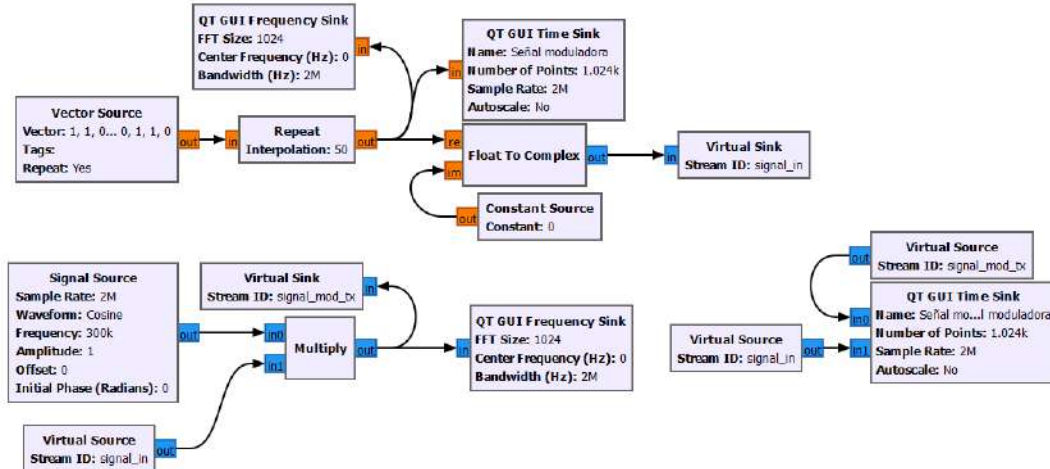


Figura 112. Esquema de modulador ASK para entorno de simulación en GNU Radio. Fuente propia.

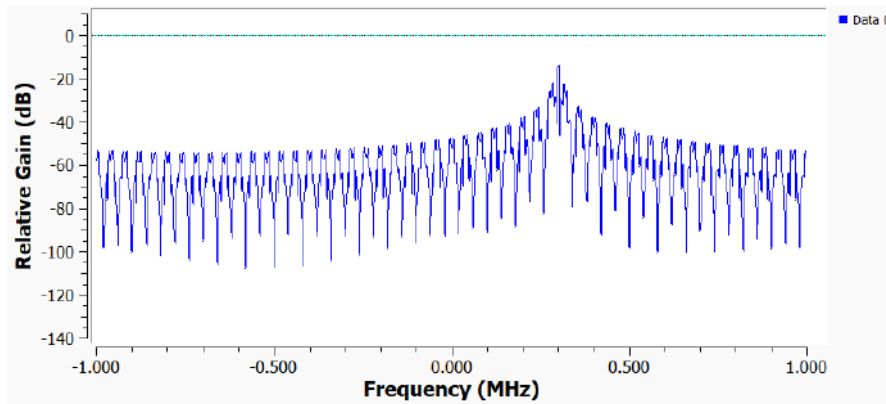


Figura 113. Espectro en frecuencia de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.

8.7.3.2. Desarrollo del esquema demodulador en GNU Radio

Para llevar a cabo la demodulación de una señal modulada en OOK, se utilizó el método de detección de envolvente, siendo este mismo, el aplicado en el receptor AM, por lo que, ambos esquemas comparten una estructura y lógica muy similar, facilitando la reutilización de los bloques anteriormente configurados y una transición fluida entre las diferentes técnicas de modulación.

En el demodulador OOK, la señal proveniente del bloque *Multiply* es transformada de complejo a punto flotante a través del bloque *Complex To Float*, esto debido a que el bloque *Rail* solo permite el ingreso de valores punto flotantes. Posteriormente, se toma parte real ("re") y es transferida al bloque *Rail*, configurado para permitir el paso de aquellos valores de la señal que se encuentren entre 0 y 5, conservando la misma configuración que en AM.

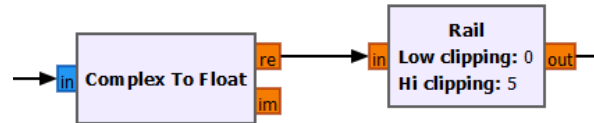


Figura 114. Conexión entre bloque el bloque Complex To Float y Rail. Fuente propia

La señal rectificadora pasa a un filtro pasa bajo (*Low Pass Filter*), configurado para permitir el paso de frecuencias menores a 40 kHz, sin embargo, a diferencia del demodulador AM, en este caso el parámetro de Decimation es establecido en 1, puesto que, no se desea eliminar las muestras añadidas a la señal moduladora. Por otro lado, al igual que en el modulador OOK, se es posible varia la frecuencia de corte a través de un bloque *QT GUI Dial*, *QT GUI Tab Widget* y un *QT GUI Level Gauge*, a fin de sincronizar esta misma con la frecuencia de la señal moduladora transmitida (figura 115 y figura 116).

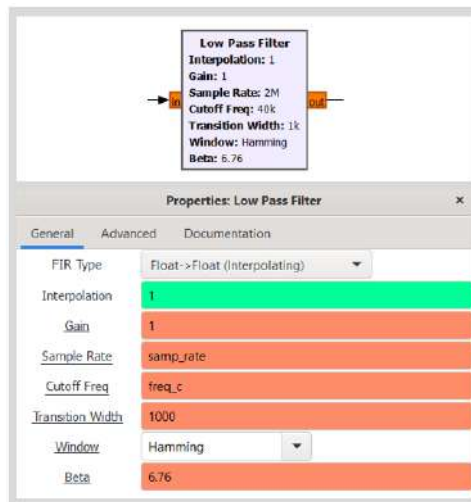


Figura 115. Bloque Low Pass Filter y su configuración en GNU Radio configurado en demodulador OOK en un entorno simulado. Fuente propia

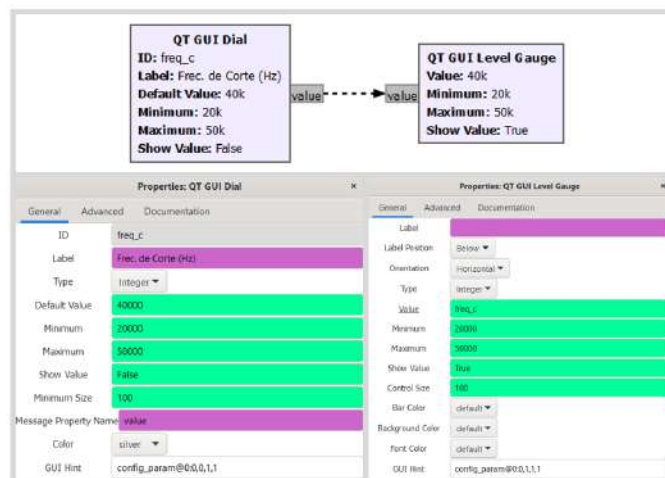


Figura 116. Bloques QT GUI Dial y QT GUI Level Gauge y su configuración para variación y visualización de frecuencia de corte en OOK en GNU Radio en un entorno simulado. Fuente propia

La señal obtenida a la salida del *Low Pass Filter* es ingresada al bloque *Threshold*, el cual compara el valor de entrada con un umbral, generando una señal binaria a la salida en base a la comparación realizada: si la entrada es mayor al umbral, la salida es “1”, y si es menor, la salida es “0”. En este caso, el umbral establecido es 0.165, teniendo en cuenta la señal obtenida del *Low Pass Filter*, la cual tiene un valor máximo aproximadamente de 0.32.

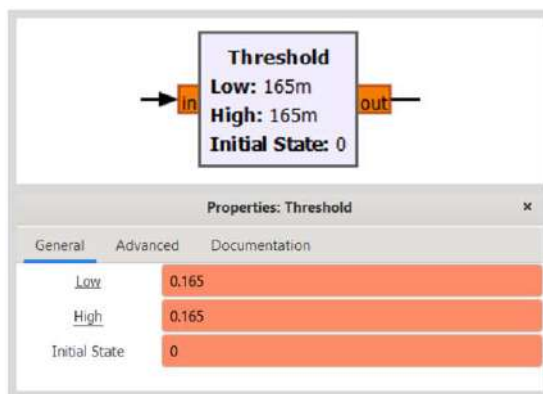


Figura 117. Bloque *Threshold* y su configuración para OOK en un entorno simulado en GNU Radio. Fuente propia

En una última instancia, para visualizar la señal modulada en OOK y la señal demodulada, se utiliza un *QT GUI Time Sink*, *QT GUI Tab Widget*, *Virtual Sink* y *Virtual Source*, configurando estos mismos de tal modo que se pueda visualizar ambas señales de forma simultánea con conexiones inalámbricas virtuales, y siendo organizados tal como se ha presentado anteriormente. Así mismo, se le añade el bloque *Delay*, con el fin corregir el desfase entre ambas señales graficadas, obteniendo el esquema final presentado en la figura 123.

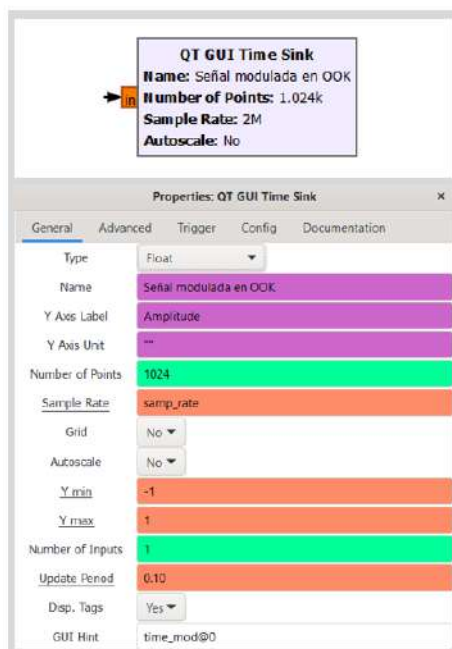


Figura 118. Bloque *QT GUI Time Sink* y su configuración para visualización de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.

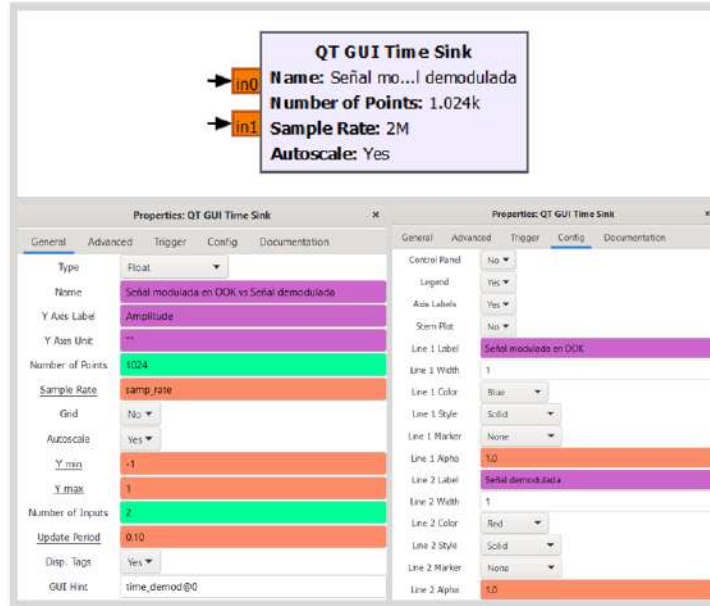


Figura 119. Bloque QT GUI Time Sink y su configuración para visualización de señal modulada vs demodulada en OOK en GNU Radio en un entorno simulado. Fuente propia.

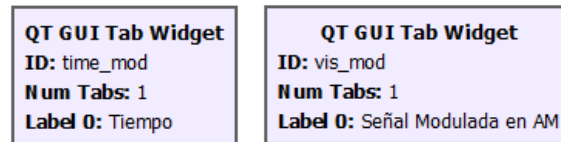


Figura 120. Bloques QT GUI Tab Widget para visualización de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.



Figura 121. Configuración de bloques QT GUI Tab Widget para visualización de señal modulada en OOK en GNU Radio en un entorno simulado. Fuente propia.

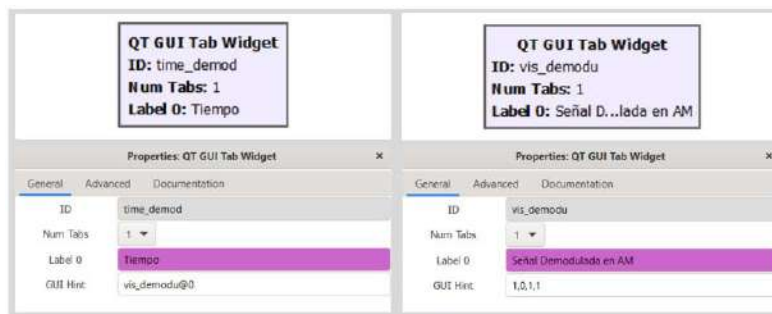


Figura 122. Bloques QT GUI Tab Widget y su configuración para visualización de señal demodulada en OOK en GNU Radio en un entorno simulado. Fuente propia.

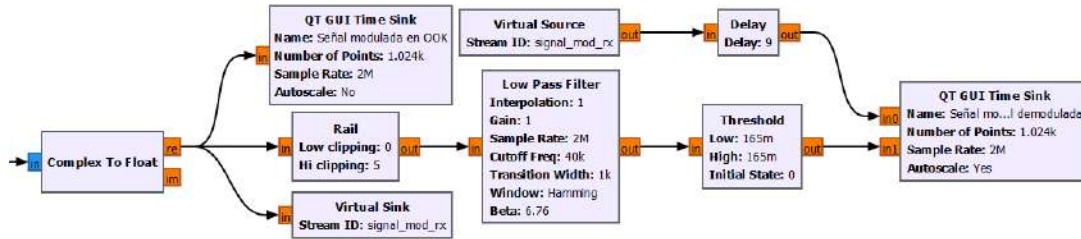


Figura 123. Esquema final del demodulador OOK en un entorno de simulación en GNU Radio. Fuente propia.

8.7.3.3. Implementación de modulador en GNU Radio usando los dispositivos SDR

Al hacer uso del hardware SDR, se requiere adecuar el esquema del modulador desarrollado, para esto, solo bastó con adicionar el bloque *Osm ocom Sink*. Esto debido a que la señal obtenida a la salida del bloque *Multiply* ya es de tipo complejo, por lo que, no se requiere adicionar un bloque convertidor, ya que este proceso fue realizado en una etapa anterior. No obstante, se modificaron ciertos parámetros de la configuración del *Osm ocom Sink*, tales como:

- **Device Arguments:** hackrf=0.
- **Sample Rate (sps):** 2 M
- **Ch0: Frequency (Hz):** 50 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

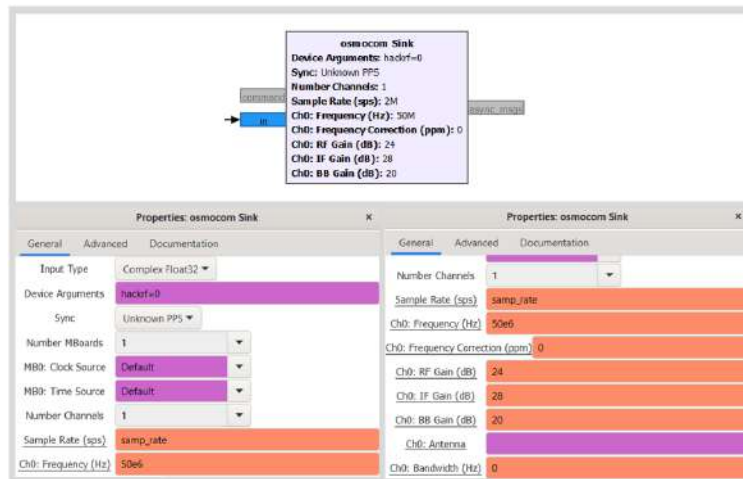


Figura 124. Configuración de bloque *Osm com Sink* y su configuración para modulador OOK en un entorno práctico en GNU Radio. Fuente propia.

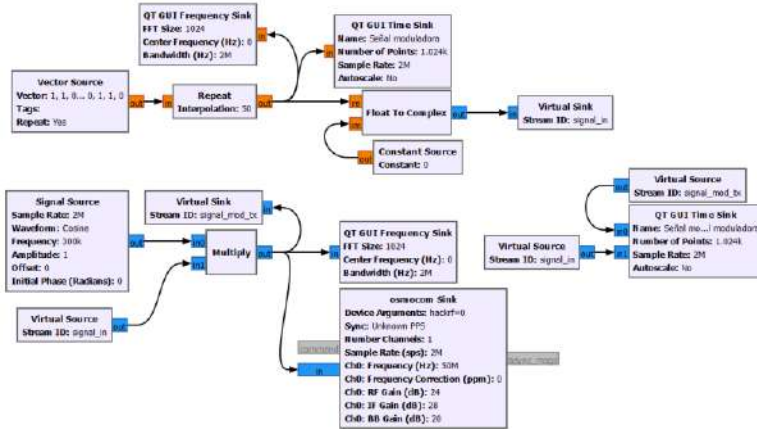


Figura 125. Esquema de modulador OOK utilizando hardware SDR en un entorno real en GNU Radio. Fuente propia.

8.7.3.4. Implementación de demodulador en GNU Radio usando los dispositivos SDR

Para la implementación del demodulador OOK haciendo uso de los dispositivos SDR, se empleó el mismo esquema desarrollado en la sección de simulación. Este esquema fue adaptado aprovechando la configuración previa, y añadiendo solo un bloque, que resulta esencial en este sistema para captar la señal transmitida, siendo este el bloque *Osm com Source*, configurado de la siguiente manera:

- **Device Arguments:** rtl=0.
- **Sample Rate (sps):** 2M
- **Ch0: Frequency (Hz):** 50 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

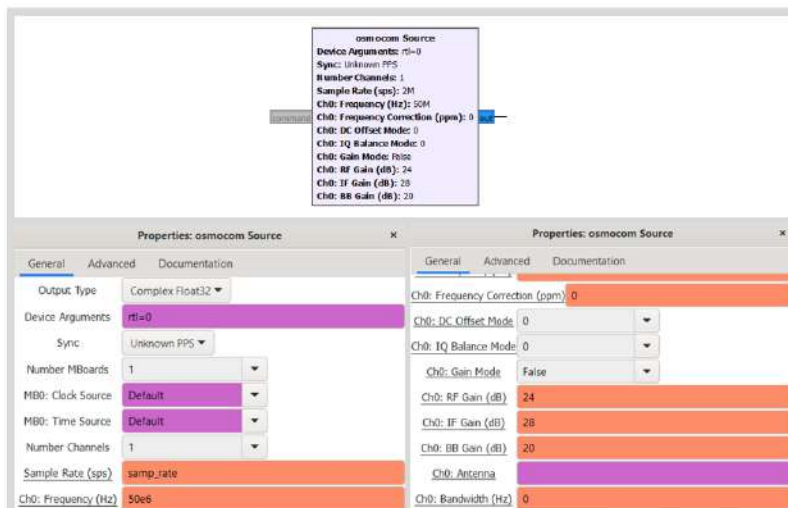


Figura 126. Bloque Osm com Source y su configuración para modulación OOK en un entorno practico en GNU Radio. Fuente propia.

Una vez configurado el *Osm com Source*, la señal obtenida a la salida es transferida al bloque convertidor (*Complex To Float*) para realizar el proceso de demodulación, obteniendo el esquema final presentado en la figura 127.

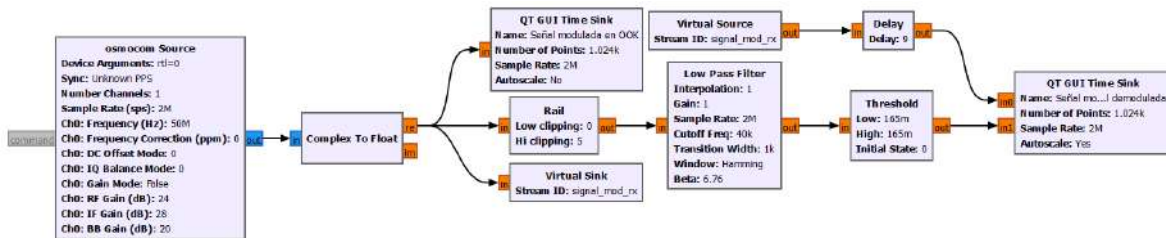


Figura 127. Esquema demodulador OOK utilizando hardware SDR en un entorno practico en GNU Radio. Fuente propia.

8.7.4. Modulación por desplazamiento de frecuencia (FSK)

Esta modulación es muy similar a la FM convencional, pero se diferencian en que la señal moduladora es una señal binaria que varía entre dos valores de amplitud discretos. El método utilizado para llevar a cabo la modulación de FSK es el método de fase discontinua, debido a que la fase es discontinua durante los tiempos conmutación.

8.7.4.1. Desarrollo del esquema modulador en GNU Radio

Inicialmente, se utiliza el bloque Vector Source, en el cual, en el parámetro Vector, se carga la señal de información binaria, en este caso de 11 bits, y se activa la opción de *Repeat* con la finalidad de que el vector se repita indefinidamente. Posteriormente, en el bloque *Repeat*, se ajustan los parámetros de Interpolation a un valor de 400, lo que permite que cada muestra de la señal de información se repita 400 veces y se extienda en el tiempo sin afectar su contenido, por último, se pasa la señal al bloque *Float To Complex*, el cual simplemente convertirá los datos de tipo flotante a complejo, estos ajustes son necesarios antes de operar la señal moduladora.

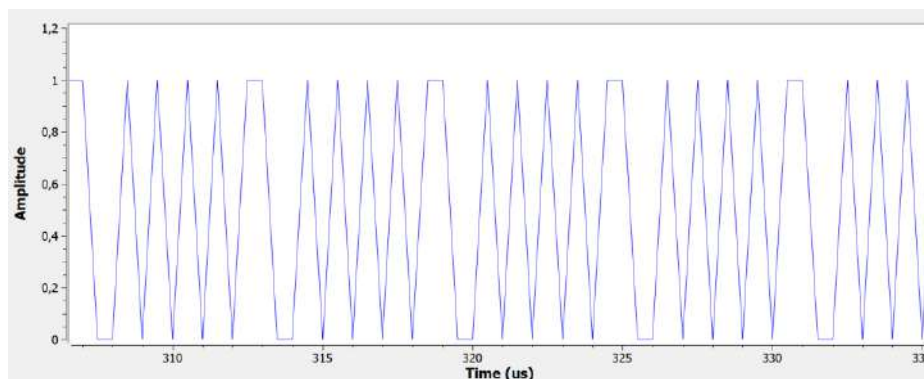


Figura 128. Señal generada por el bloque Vector Source. Fuente propia.

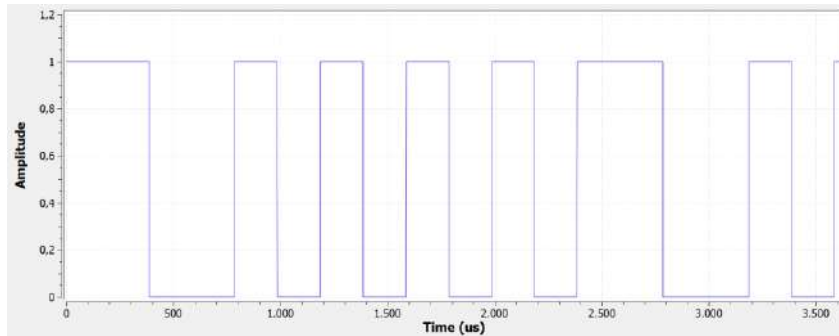


Figura 129. Señal moduladora a la salida del bloque Repeat. Fuente propia.

$$f_b (\text{frecuencia de bit}) = \frac{1}{T_b (\text{duracion de bit})} \rightarrow T_b = \frac{1}{5 \text{ Kbps}} = 200 \text{ us}$$

$$\text{Número de muestras por bit} = \frac{\text{Tasa de muestreo}}{5 \text{ Kbps}} = \frac{2 \text{ MHz}}{5 \text{ Kbps}} = 400$$

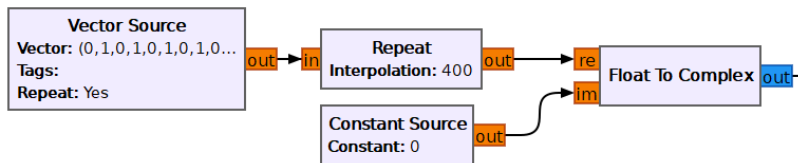


Figura 130. Señal moduladora con ajustes de tiempo. Fuente propia.

Para llevar a cabo la modulación FSK, es importante considerar que, cuando se recibe un 1 lógico de la señal moduladora, esta se multiplica por una frecuencia de marca (f_m) o de trabajo, constituida por un oscilador local implementado mediante el bloque Signal Source, configurado con una frecuencia de 400 KHz y una amplitud de 1.

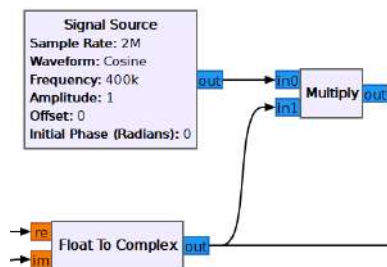


Figura 131. Señal moduladora multiplicada por la f_m . Fuente propia.

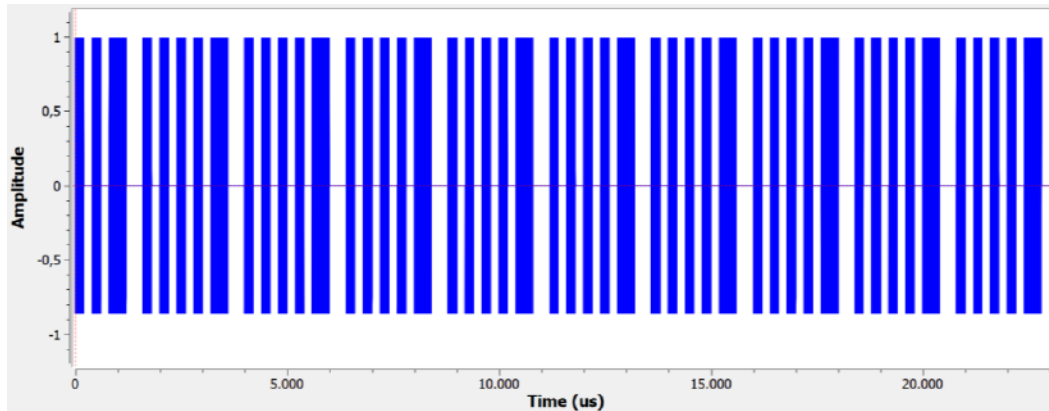


Figura 132. Señal resultante de la multiplicación de la señal moduladora por la fm. Fuente propia.

Para este diseño, la frecuencia de marca se introdujo como una variable mediante los bloques *QT GUI Dial* y *QT GUI Level Gauge*, lo que permite ajustar este parámetro dentro de un rango de 100 KHz a 800 KHz.

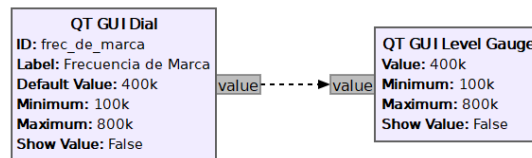


Figura 133. Bloques *QT GUI Dial* y *QT GUI Level Gauge* para ajustar la fm. Fuente propia.

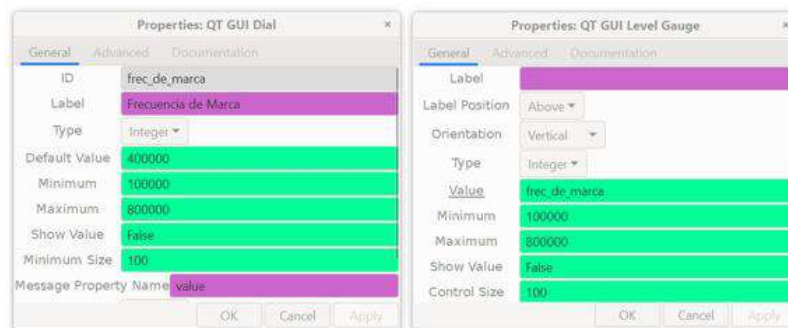


Figura 134. Ajuste de parámetros para los bloques *QT GUI Dial* y *QT GUI Level Gauge* para la fm. Fuente propia.

Cuando se recibe un 0 lógico, primero se realiza la operación $1 - m(t)$ con el fin de obtener la señal invertida, lo que convierte los valores de 1 lógico en 0 lógicos y los 0 lógicos en -1. Posteriormente, se multiplica esta señal por la frecuencia de espacio (f_s), representada mediante el bloque *Signal Source*, configurado como un oscilador con una frecuencia de 200 KHz y una amplitud de 1.

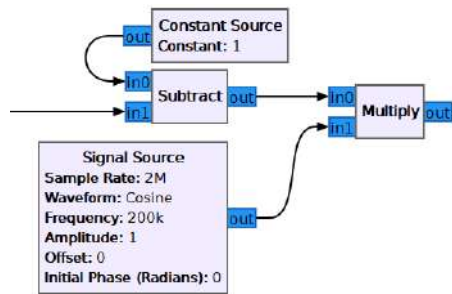


Figura 135. Señal moduladora multiplicada por la fs. Fuente propia.

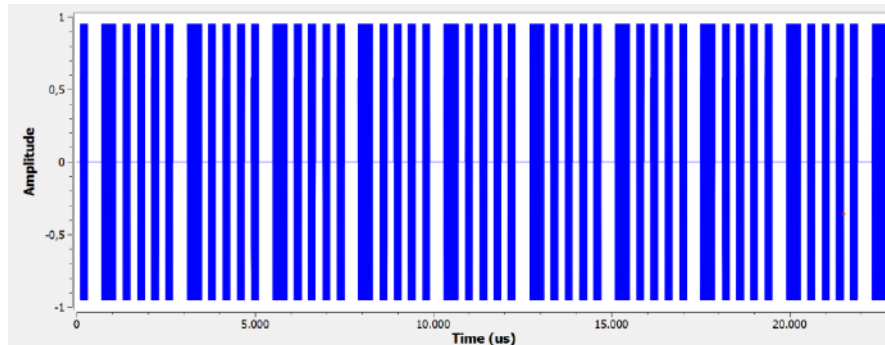


Figura 136. Señal resultante de la multiplicación de la señal moduladora por la fs. Fuente propia.

De manera similar a la frecuencia de marca, la frecuencia de espacio también se introdujo como una variable mediante los bloques *QT GUI Dial* y *QT GUI Level Gauge*, lo que permite ajustar este parámetro entre un valor de 100 KHz y 400 KHz.

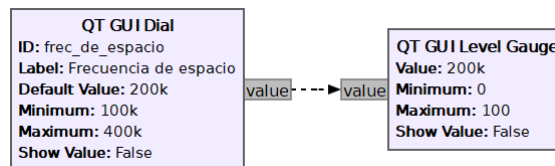


Figura 137. Bloques *QT GUI Dial* y *QT GUI Level Gauge* para ajustar la fs. Fuente propia

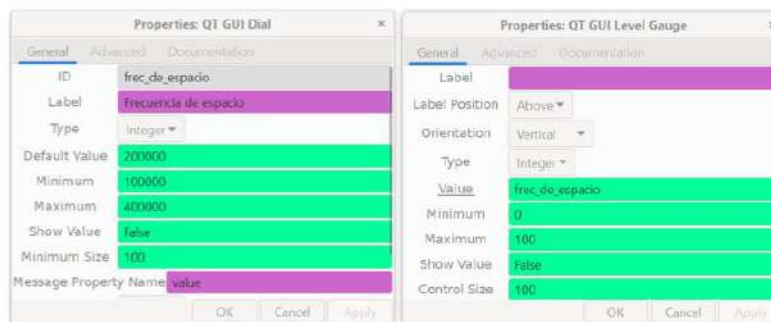


Figura 138. Ajuste de parámetros para los bloques *QT GUI Dial* y *QT GUI Level Gauge* para la fs. Fuente propia.

Los procesos anteriormente realizados tienen como objetivo simular un conmutador electrónico que detecta el nivel lógico y conmuta entre dos osciladores según el dato

lógico recibido. Finalmente, para obtener la señal modulada en FSK, es suficiente con sumar las dos señales resultantes.

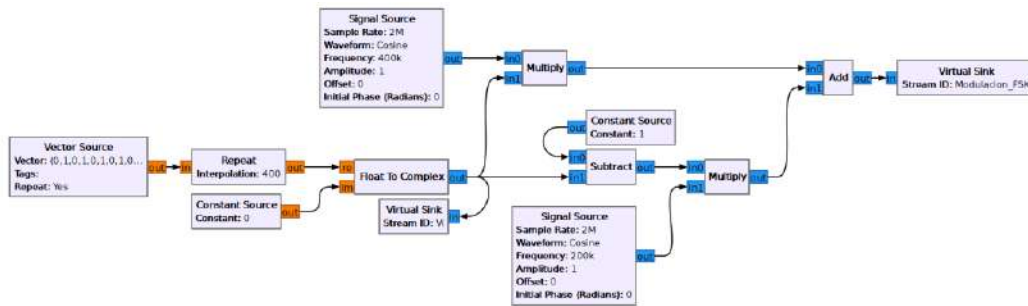


Figura 139. Esquema modulador FSK. Fuente propia.

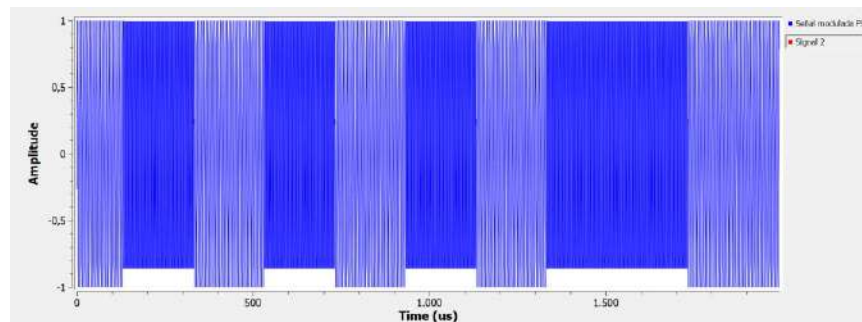


Figura 140. Señal modulada FSK. Fuente propia.

A partir de las configuraciones realizadas para el esquema modulador, se puede calcular la desviación máxima de frecuencia, que corresponde a la desviación máxima de la frecuencia portadora, esta se determina como la diferencia entre las frecuencias de marca y de espacio, dividida entre dos:

$$\Delta f = \frac{|f_m - f_s|}{2} \rightarrow \frac{|400 \text{ KHz} - 200 \text{ KHz}|}{2} = 100 \text{ KHz}$$

Dado que la señal modulada solo puede adoptar uno de los dos valores, la desviación máxima de frecuencia siempre es constante y se mantiene en su valor máximo.

Para el cálculo del mínimo ancho de banda, la ecuación es similar a la regla de Carson para la modulación angular, aunque en este caso se sustituye la frecuencia moduladora por la rapidez de bits.

$$B = 2(\Delta f + f_b) \rightarrow 2(100 \text{ KHz} + 5 \text{ Kbps}) = 210 \text{ KHz}$$

El índice de modulación para FSK, conocido como factor h, es un parámetro que indica la relación entre la separación de las frecuencias de marca y espacio respecto a la rapidez de bits. Mientras mayor sea el factor h, mayor será la separación relativa, lo que mejora la inmunidad al ruido, aunque también incrementa el ancho de banda necesario y está definido por la siguiente expresión.

$$h = \frac{|f_m - f_s|}{f_b} \rightarrow \frac{|400 \text{ KHz} - 200 \text{ KHz}|}{5 \text{ Kbps}} = 40$$

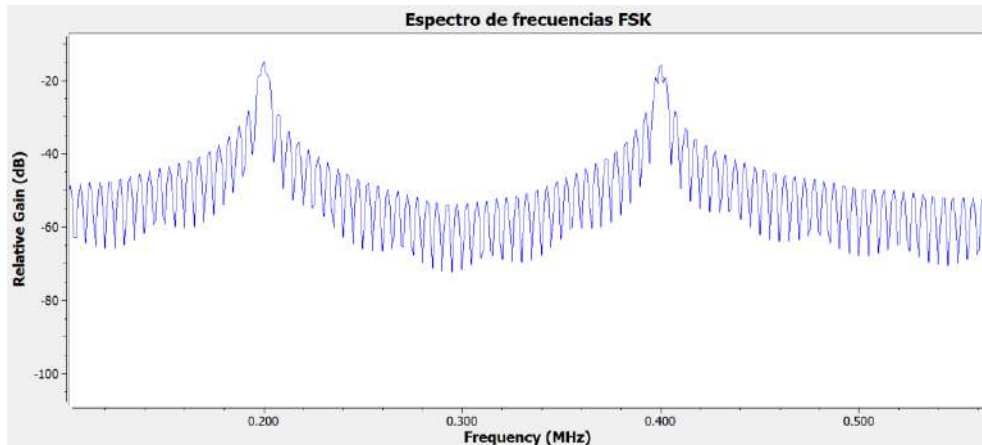


Figura 141. Espectro de frecuencias para la modulación FSK. Fuente propia

8.7.4.2. Desarrollo del esquema demodulador en GNU Radio

Para el demodulador FSK, se consideró el diseño previamente mencionado del demodulador por detección de envolvente, el cual está compuesto, en primer lugar, por dos filtros pasa bandas con el fin de detectar las dos frecuencias por separado, recordando que cada frecuencia está asociada a un estado lógico.



Figura 142. Bloque Band Pass Filter y configuración. Fuente propia.

Se configuró el primer filtro pasa bandas en el rango de 350 KHz a 450 KHz, con el fin de obtener los datos transmitidos en la frecuencia de marca de 400 KHz. Posteriormente, la señal ya filtrada se pasa por el bloque Rail y se rectifica para conservar únicamente el semiciclo positivo de la señal.

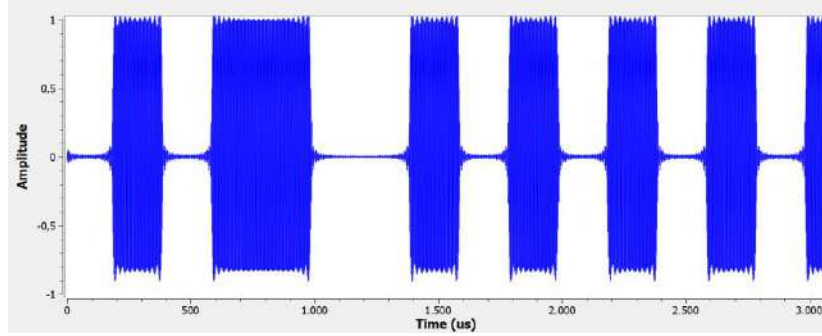


Figura 143. Señal modulada a la salida del filtro pasa banda para fm. Fuente propia.

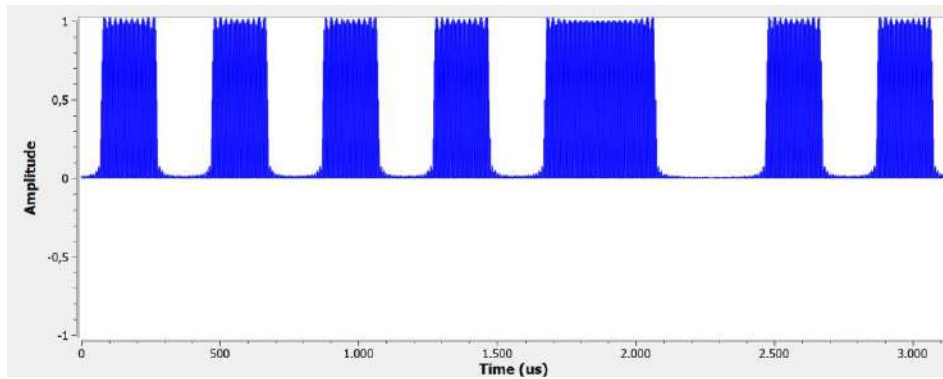


Figura 144. Señal rectificada para demodulador de la fm. Fuente propia.

El segundo de los filtros pasa bandas se configuró en el rango de 150 KHz a 250 KHz, con el fin de obtener los datos transmitidos en la frecuencia de espacio de 200 KHz. Posteriormente, la señal se rectifica de la misma manera que en el caso anterior.

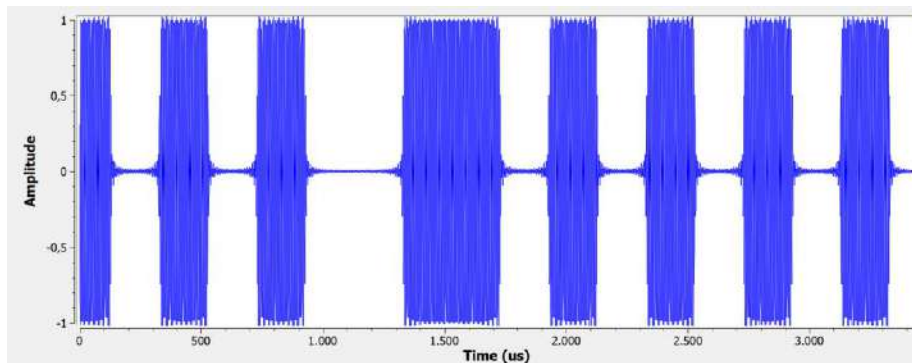


Figura 145. Señal modulada a la salida del filtro pasa banda para fs. Fuente propia.

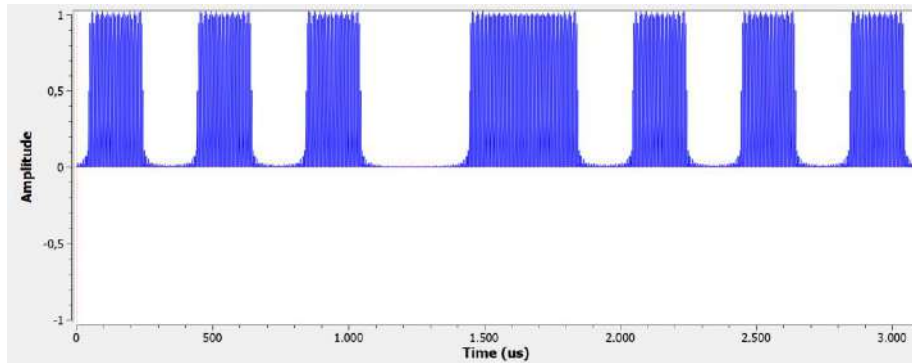


Figura 146. Señal rectificada para el demodulador de la fs. Fuente propia.

Como se explicó anteriormente, después de la rectificación de la onda, se suaviza la señal mediante un filtro pasa bajas para obtener la envolvente. Finalmente, restando ambas señales y aplicando un bloque comparador como el *Threshold*, se determinan los cruces por cero de la señal y obtener la señal correctamente demodulada en FSK.

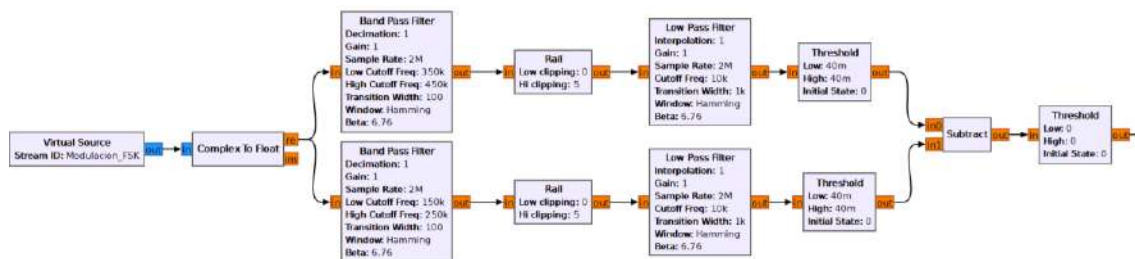


Figura 147. Esquema demodulador de FSK. Fuente propia.

8.7.4.3. Implementación del esquema modulador en GNU Radio usando los dispositivos SDR

Para la implementación del modulador FSK utilizando los dispositivos SDR, se empleó el mismo esquema desarrollado en la sección de simulación, al cual únicamente se añadió el bloque *Osmocom Sink*, que permite transmitir la señal modulada y cuenta con la siguiente configuración:

- **Device Arguments:** rtl=0.
- **Sample Rate (sps):** 2M
- **Ch0: Frequency (Hz):** 50 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

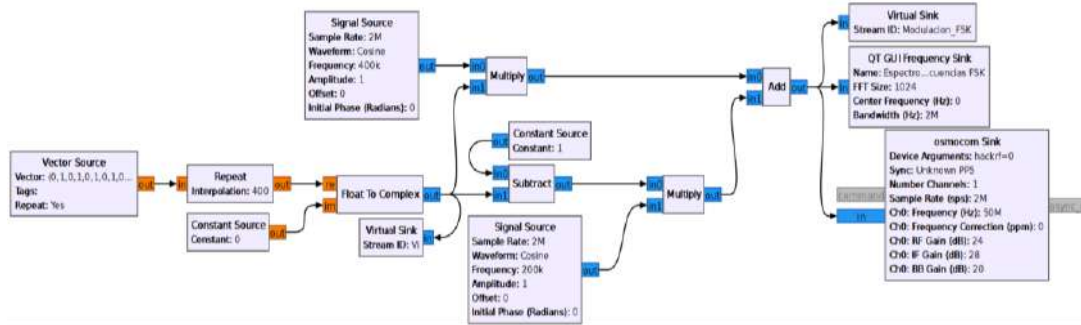


Figura 148. Esquema modulador FSK con el bloque trasmisor osmocom Sink. Fuente propia.

8.7.4.4. Implementación del esquema demodulador en GNU Radio usando los dispositivos

Para la implementación del demodulador FSK utilizando los dispositivos SDR, se empleó el mismo esquema desarrollado en la sección de simulación, al cual se añadió el bloque *Osmocom Source*, que permite la conexión con el módulo receptor para captar la señal modulada, este bloque presenta la siguiente configuración:

- **Device Arguments:** rtl=0.
- **Sample Rate (sps):** 2 M
- **Ch0: Frequency (Hz):** 50 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

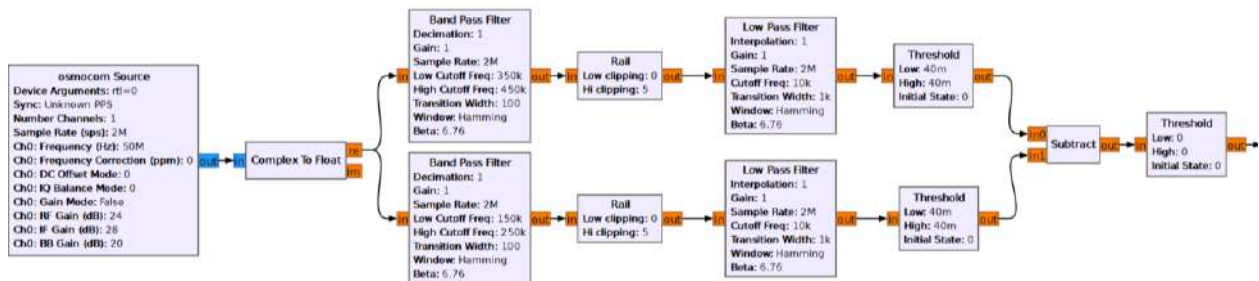


Figura 149. Esquema demodulador FSK con el bloque receptor osmocom Source. Fuente propia.

8.7.5. Modulación por desplazamiento de fase binaria (BPSK)

La modulación BPSK es similar a la modulación por frecuencia, ya que en esta técnica también se produce un desplazamiento de fase, sin embargo, en este caso, la señal moduladora es digital, lo que permite tener dos fases de salida para una sola frecuencia portadora, una de las fases de salida representa el 1 lógico, mientras que la otra corresponde al 0 lógico. Esto implica que, cuando el dato binario varía entre 0 y 1 lógico, se produce un desfase de 180° entre cada estado.

8.7.5.1. Desarrollo del esquema modulador en GNU Radio

Para el diseño, inicialmente se introduce la señal moduladora binaria a través del bloque conocido como Vector Source, activando la opción de Repeat para que el vector se repita indefinidamente. Posteriormente, la señal se conecta al bloque Repeat, donde se ajusta el parámetro de interpolación a un valor de 30, para finalmente realizar la conversión de datos de flotante a complejo utilizando el bloque Float to Complex.

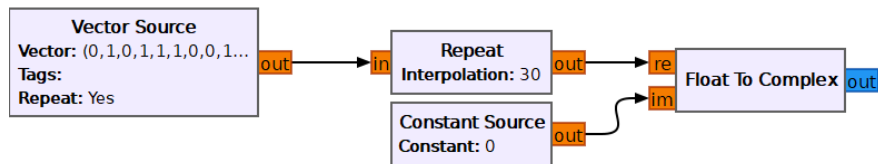


Figura 150. Configuración de bloques generar señal moduladora para BPSK. Fuente propia.

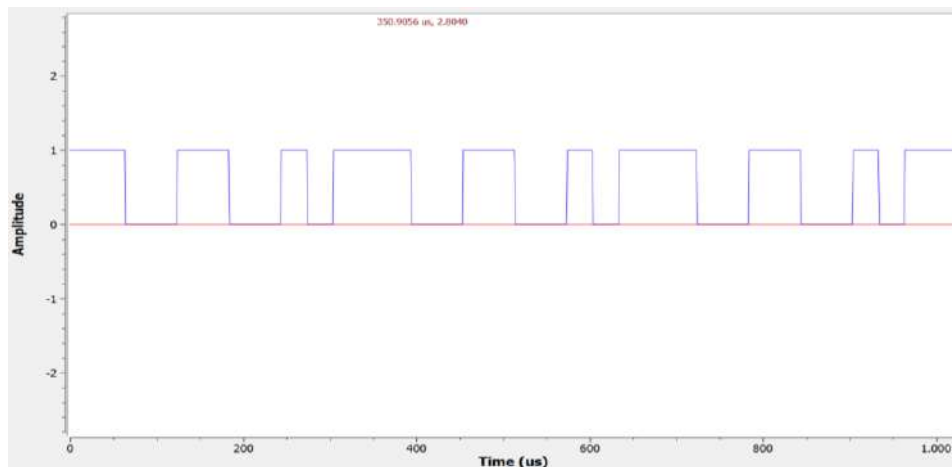


Figura 151. Señal moduladora generada para BPSK. Fuente propia.

$$f_b(\text{frecuencia de bit}) = \frac{1}{T_b(\text{duracion de bit})} \rightarrow T_b = \frac{1}{33.3 \text{ Kbps}} = 30 \text{ us}$$

$$\text{Número de muestras por bit} = \frac{\text{Tasa de muestreo}}{5 \text{ Kbps}} = \frac{2 \text{ MHz}}{33.3 \text{ Kbps}} = 30$$

$$B = f_b \rightarrow 33.33 \text{ KHz}$$

A partir de la frecuencia de bit es posible calcular la frecuencia fundamental máxima de la entrada binaria, descrita por la siguiente ecuación.

$$f_a = \frac{f_b}{2} \rightarrow \frac{33.33 \text{ Kbps}}{2} = 16.66 \text{ KHz}$$

Para realizar la modulación BPSK, se comienza llevando a cabo la operación $1 - m(t)$, con el objetivo de invertir la señal, transformando los ceros en unos y viceversa.

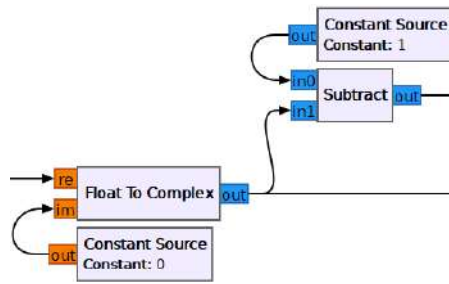


Figura 152. Configuración de bloques para generar la inversión de la señal modulada. Fuente propia

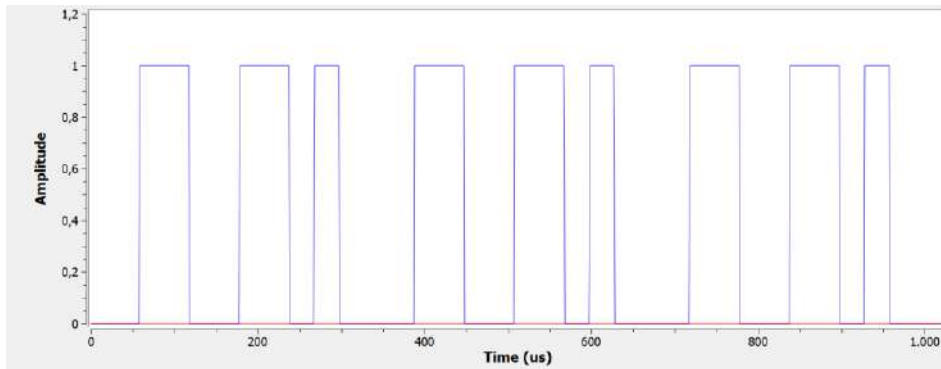


Figura 153. Señal modulada invertida. Fuente propia

Luego, se realiza la operación $-1 \cdot m(t)_{complemento}$ para transformar los valores representados con la magnitud de 1 en -1, como se observa en la figura 154.

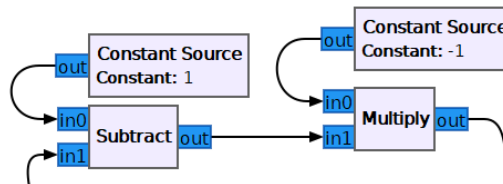


Figura 154. Configuración de bloques para transformar los valores 1 en -1. Fuente propia.

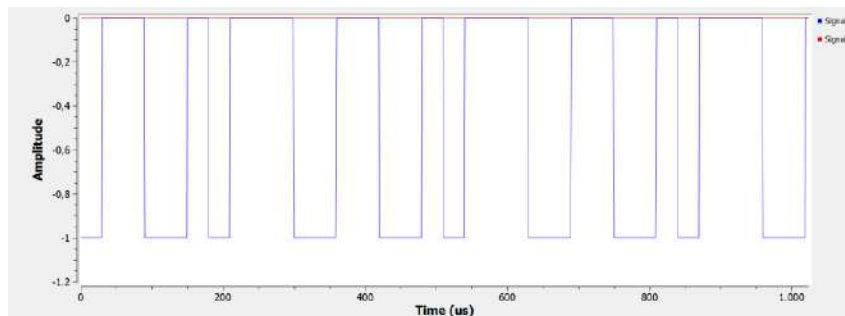


Figura 155. Señal transformada con magnitud de -1. Fuente propia.

Al sumar la señal moduladora original con la resultante de las operaciones anteriores, se obtiene una señal representada por los valores 1 y -1.

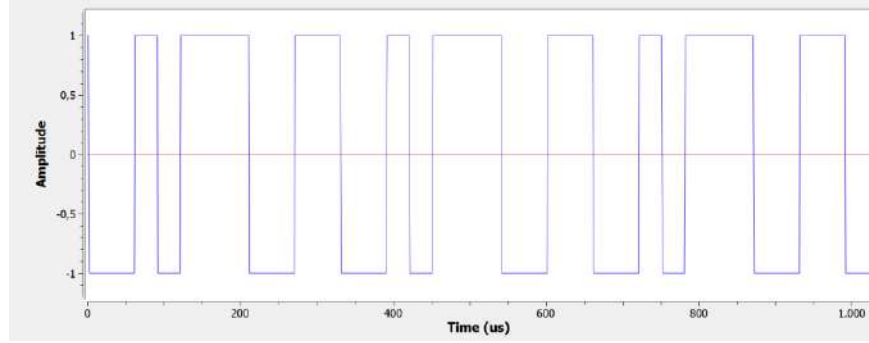


Figura 156. Señal moduladora original sumada a la resultante de las operaciones. Fuente propia.

Al multiplicarla por una señal portadora, cuando se recibe el valor lógico 1, se mantiene la fase original de la portadora; mientras que, al multiplicarse por el valor lógico -1, la portadora se desfasa 180° , obteniendo así la señal modulada BPSK.

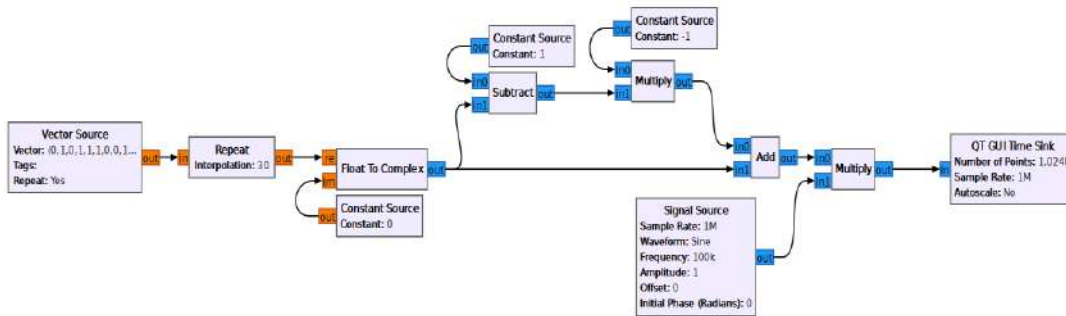


Figura 157. Esquema modulador de BPSK. Fuente propia.

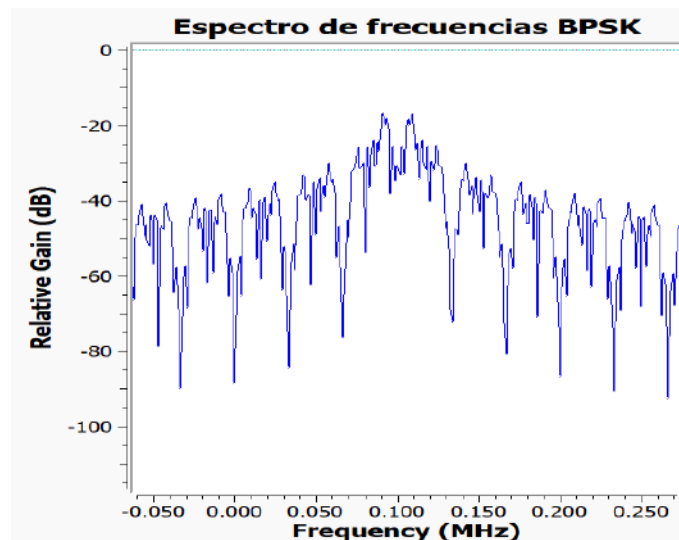


Figura 158. Espectro de frecuencias BPSK. Fuente propia

Para los parámetros configurados en el esquema modulador BPSK, se calcula la banda superior e inferior del espectro de salida.

$$LSF = f_c - f_a \rightarrow 100 \text{ KHz} - 16.66 \text{ KHz} = 83.33 \text{ KHz}$$

$$USF = f_c + f_a \rightarrow 100 \text{ KHz} + 16.66 \text{ KHz} = 116.67 \text{ KHz}$$

8.7.5.2. Desarrollo del esquema demodulador en GNU Radio

Para el diseño del demodulador BPSK, se inició configurando el bloque Costas Loop, el cual permite sincronizar la fase y la frecuencia de una señal portadora modulada, mediante la adecuada configuración de los siguientes parámetros:

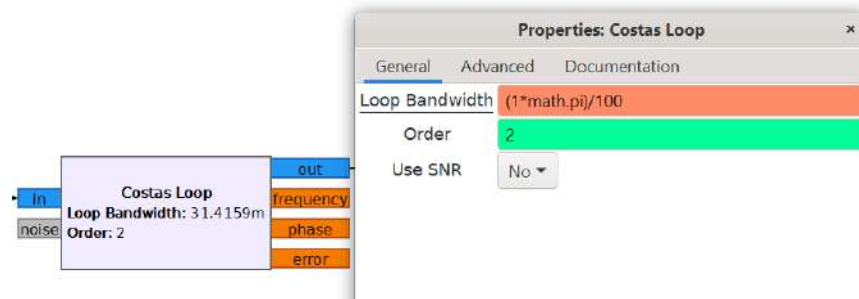


Figura 159. Bloque Costas Loop para realizar sincronización de fase y frecuencia. Fuente propia.

Loop Bandwidth (R): Controla el ancho de banda del bucle interno, el cual determina qué tan rápido o lento debe ajustarse a los cambios de fase y frecuencia. Un valor más alto permite ajustes más rápidos, pero puede producir inestabilidad, mientras que un valor más bajo reacciona más lentamente a los cambios, pero es más estable. Inicialmente, se configuró de forma predeterminada para un ancho de banda interno de segundo orden $\sim 2\pi/100$, pero se ajustó, luego de realizar diferentes pruebas, a un valor más pequeño de $\pi/23.271$.

Order: Este parámetro se configura según la cantidad de niveles de frecuencia que se estén trabajando. Para BPSK, el valor de *order* es 2, ya que se trabaja con dos niveles diferentes de frecuencia.

Para continuar con el proceso de demodulación, se utilizó el esquema del detector de envolvente, para el que, primeramente, se implementó el bloque Complex To Float para convertir los datos a tipo flotante, considerando que es el único tipo de dato que acepta el bloque Rail, el cual realiza la rectificación de la señal. Finalmente, la señal rectificada se suavizó mediante un filtro pasa bajas para obtener la envolvente.

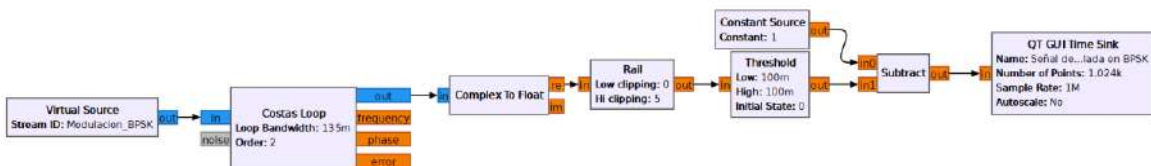


Figura 160. Esquema del detector de envolvente. Fuente propia.

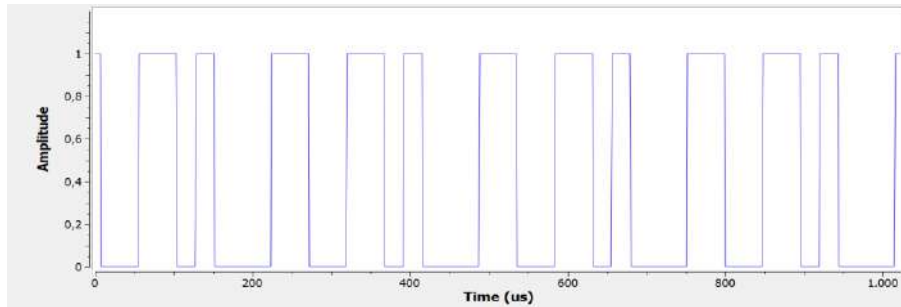


Figura 161. Señal rectificada. Fuente propia.

Para finalizar con el proceso de demodulación, se observa que la señal está centrada en una amplitud de 0.6, por lo que se fija como valor umbral 0.1, al superarse dicho umbral, se obtiene un 1 lógico, y cuando la señal decae, se genera un 0 lógico. Finalmente, se realiza la inversión de la señal para completar el proceso de demodulación de BPSK.

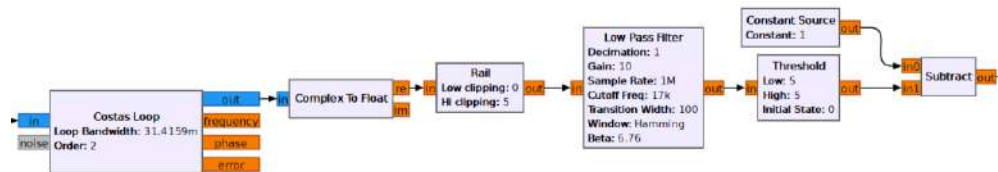


Figura 162. Esquema final de demodular BPSK en GNU Radio. Fuente propia.

8.7.5.3. Implementación del esquema modulador en GNU Radio usando los dispositivos SDR

Para la implementación del modulador BPSK utilizando los dispositivos SDR, se empleó el mismo esquema desarrollado en la sección de simulación, al cual únicamente se agregó el bloque Osmocom Sink para permitir la transmisión de la señal modulada. Este bloque presenta la siguiente configuración:

- **Device Arguments:** rtl=0.
- **Sample Rate (sps):** 2 M
- **Ch0: Frequency (Hz):** 50 M
- **Ch0: RF Gain (dB):** 24

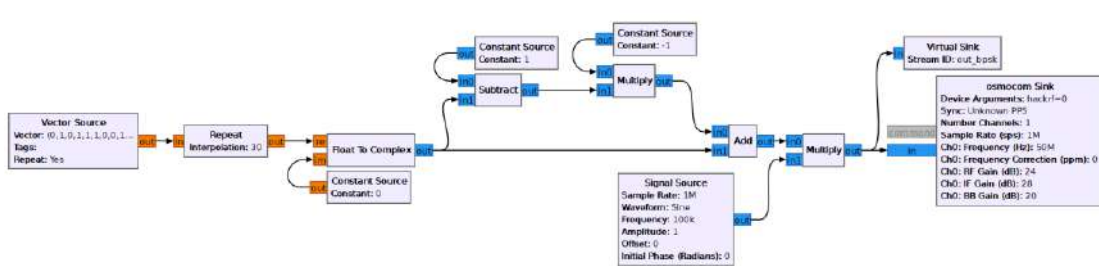


Figura 163. Esquema modulador BPSK conectado al transmisor a través del bloque osmocom Sink. Fuente propia.

8.7.5.4. Implementación del esquema demodulador en GNU Radio usando los dispositivos SDR

Para la implementación del demodulador BPSK utilizando los dispositivos SDR, se empleó el mismo esquema desarrollado en la sección de simulación, al cual se agregó el bloque Osmocom Source, que permite la conexión con el módulo receptor para captar la señal modulada. Este bloque presenta la siguiente configuración:

- **Device Arguments:** rtl=0.
- **Sample Rate (sps):** 2 M
- **Ch0: Frequency (Hz):** 50 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

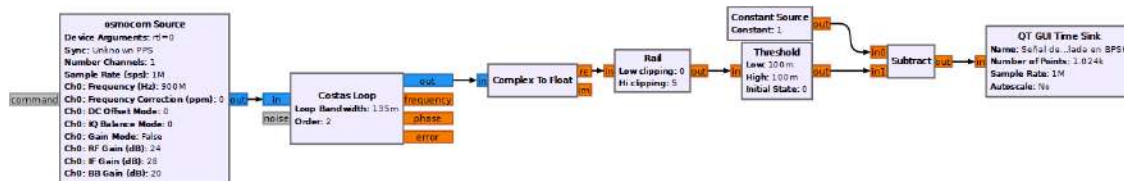


Figura 164. Esquema demodulador BPSK conectado al receptor mediante el bloque osmocom Source. Fuente propia.

8.7.6. Modulación de amplitud en cuadratura (QAM)

La modulación por amplitud en cuadratura resulta ser una técnica de modulación que mezcla tanto la modulación en amplitud como en fase, lo que trae consigo la posibilidad de transmitir mayor cantidad de datos por cada símbolo, a comparación con otras técnicas de modulación. No obstante, en el presente proyecto, nos enfocamos en la implementación de 4QAM, una variante de QAM en la que se cuenta con 4 posibles estados de amplitud y fase para la representación de los tres bits por símbolo.

8.7.6.1. Desarrollo del esquema modulador en GNU Radio

Para la implementación de 4QAM se inició con la creación de un nuevo proyecto y configurando los parámetros de los bloques por defecto, tal como se ha hecho en implementaciones digitales anteriores. Seguidamente, se añade un bloque Vector Source a fin de generar una secuencia que representará a la señal moduladora, y a su vez estableciendo el tipo de dato en byte.

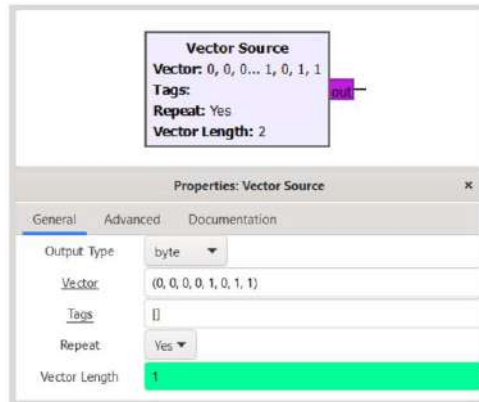


Figura 165. Bloque Vector Source y su configuración para 4QAM en GNU Radio en un entorno simulado. Fuente propia

Una vez configurado el bloque *Vector Source*, la señal obtenida a la salida es ingresada al bloque *Stream Demux*, siendo este último el encargado de separar los valores contenidos en el vector y enviarlos a una salida diferente, por lo que es configurado con un “IO Type” de tipo byte y el número de salidas en 2, tal como se muestra en la figura 166.

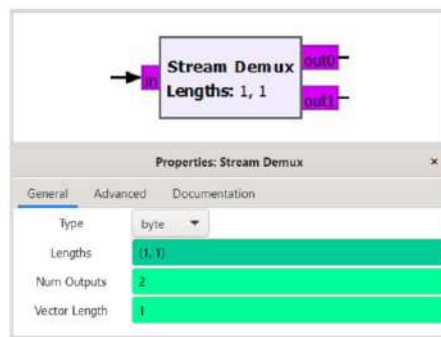


Figura 166. Bloque Stream Demux y su configuración para 4QAM en GNU Radio en un entorno simulado. Fuente propia

Por consiguiente, se conecta un bloque *Chunks to Symbols* a cada salida del bloque *Stream Demux*, configurados tal como se muestra en la figura 167. Todo esto realizado a fin de mapear los valores de los canales, pasando de alternar entre 0 y 1, a 1 y -1.

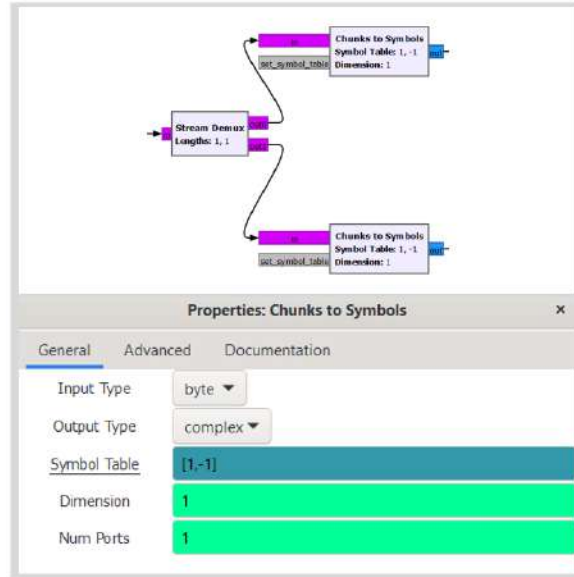


Figura 167. Conexión entre el bloque *Stream Demux* y los bloques *Chunks To Symbols* junto con su configuración para 4QAM en GNU Radio en un entorno de simulación. Fuente propia

Una vez mapeadas las señales, se ingresan a un bloque *Repeat*, para así establecer la frecuencia de la señal moduladora, calculando el valor de interpolación de la siguiente forma:

$$f_b(\text{frecuencia de bit}) = \frac{1}{T_b(\text{duracion de bit})} \rightarrow T_b = \frac{1}{1 \text{ KHz}} = 1000 \text{ us}$$

$$T_b = \frac{\text{Número de muestras por bit}}{\text{Tasa de muestreo}}$$

$$\frac{1}{1 \text{ KHz}} = \frac{\text{Número de muestras por bit}}{\text{Tasa de muestreo}}$$

$$\text{Número de muestras por bit} = \frac{\text{Tasa de muestreo}}{1 \text{ KHz}} = \frac{1 \text{ MHz}}{1 \text{ KHz}} = 1000$$

$$R_b = f_m * \text{número de bits por simbolo} = 1\text{kHz} \cdot 2 = 2 \text{ Kbits/s}$$

$$f_s = \frac{R_b}{2} = \frac{2 \text{ Kbits/s}}{2} = 1 \text{ KHz}$$

$$BW(\text{ancho de banda}) = 2 \cdot f_s = 2 \cdot \frac{1 \text{ Kbits}}{s} = 2\text{KHz}$$

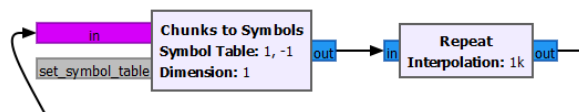


Figura 168. Conexión entre los bloques *Chunks to Symbols* y *Repeat* para 4QAM en GNU Radio en un entorno de simulación. Fuente propia

Cabe mencionar, que al igual que las modulaciones anteriores, la frecuencia de moduladora puede ser ajustada mediante un bloque *QT GUI Dial*, *QT GUI Level Gauge* y el *QT GUI Tab Widget* (figura 171 y figura 173).



Figura 169. Configuración de los bloques Repeat para 4QAM en GNU Radio en un entorno simulado. Fuente propia

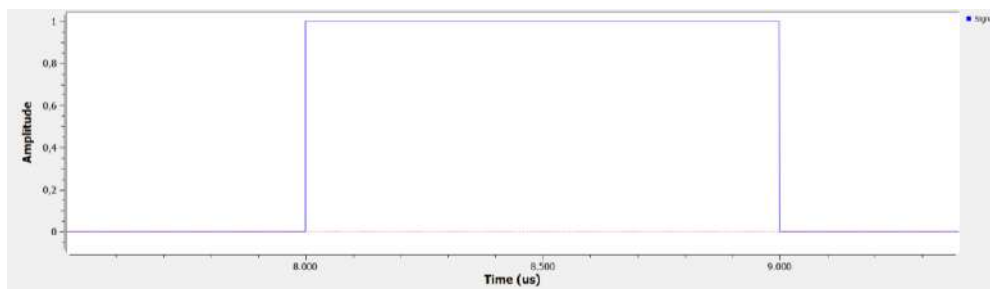


Figura 170. Duración de bit de señal moduladora para 4QAM en GNU Radio en un entorno simulado. Fuente propia



Figura 171. Bloque QT GUI Tab Widget para sección de configuración de configuración de parámetros en 4QAM en GNU Radio en un entorno simulado. Fuente propia

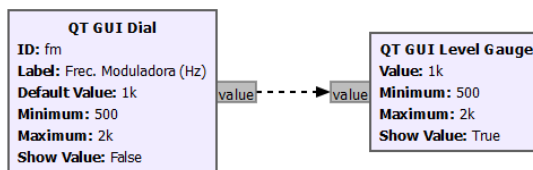


Figura 172. Bloques QT GUI Dial y QT GUI Level Gauge para variación y visualización de la señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia

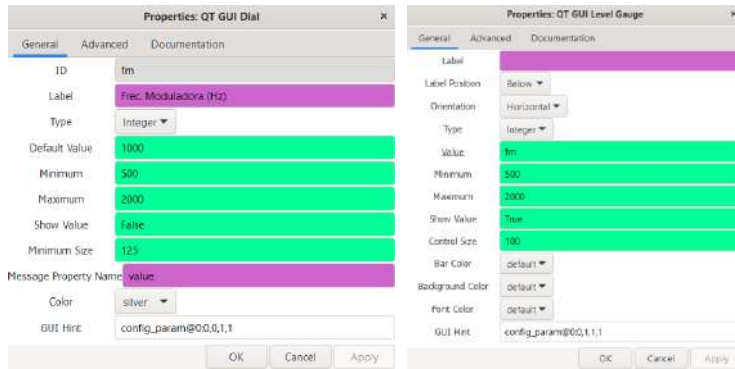


Figura 173. Configuración de bloques QT GUI Dial y QT GUI Level Gauge para variación y visualización de la señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia

Una vez ajustadas las señales de ambos canales (I y Q), se multiplicaron por una señal sinusoidal (portadora) con fase de 0° y 90° respectivamente, siendo generadas a través de un bloque *Signal Source* y un *Phase Shift*. Esta portadora fue configurada con una frecuencia inicial de 250kHz, no obstante, para garantizar una correcta reconstrucción de la moduladora en el demodular, se utilizó un bloque *Rational Resampler* para remuestrear la señal portadora, por lo que, este efecto reduce la frecuencia de la señal, siendo ahora de 68.18 KHz (figura 174, figura 175).

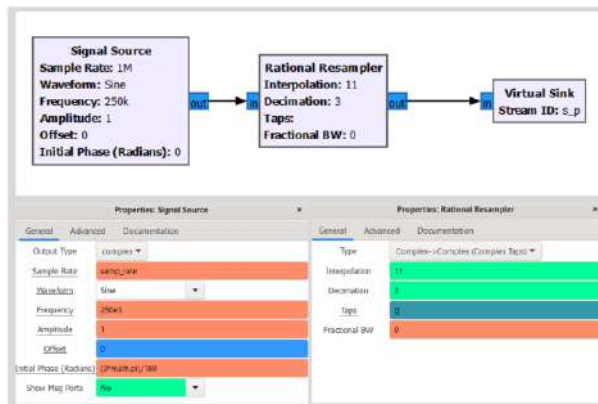


Figura 174. Bloques Signal Source y Rational Resampler y su configuración para generar y re muestrear señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia

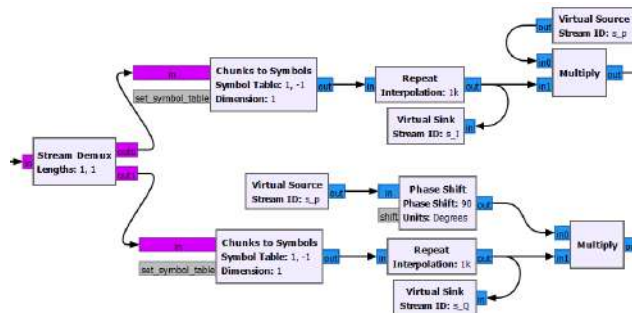


Figura 175. Conexiones entre los canales I, Q y la señal portadora en 4QAM en GNU Radio en un entorno simulado. Fuente propia

Finalmente, para obtener la señal modulada en 4QAM añadimos un bloque *Add* a fin de sumar las señales resultantes de las multiplicaciones realizadas, asimismo, es visualizada mediante el bloque *QT Time Sink*, *QT GUI Tab Widget*, *Virtual Sink* y *Virtual Source* (figura 176, figura 177).

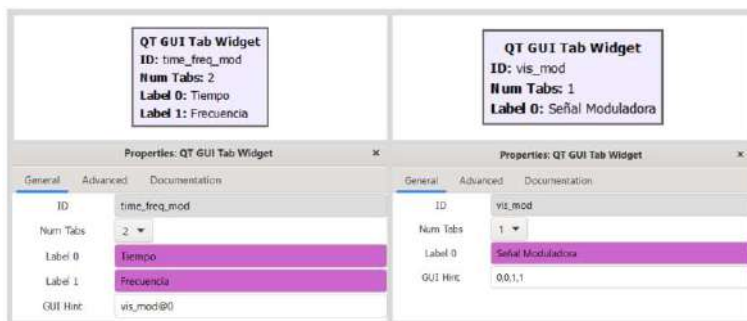


Figura 176. Bloques QT GUI Tab Widget y su configuración para la sección de visualización de señal moduladora en 4QAM en GNU Radio en un entorno simulado. Fuente propia

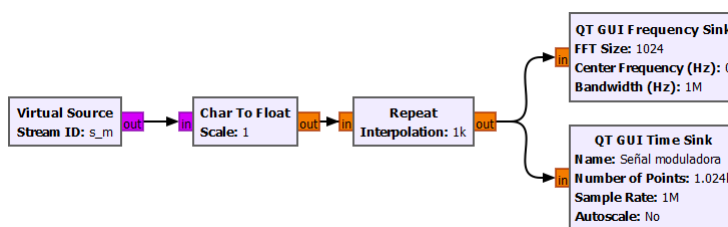


Figura 177. Conexión de bloque QT GUI Frequency Sink y QT GUI Time Sink para visualización en el dominio del tiempo y espectro en frecuencia en 4QAM en GNU Radio en un entorno simulado. Fuente propia

Cabe mencionar que para visualizar de forma correcta el diagrama de constelación se requirió el uso de un bloque *Rational Resampler*, configurado con los valores invertidos a los establecidos en la generación de la onda portadora, presentándose así los 4 puntos correspondientes a los 4 posibles estados de una señal modulada en 4QAM.

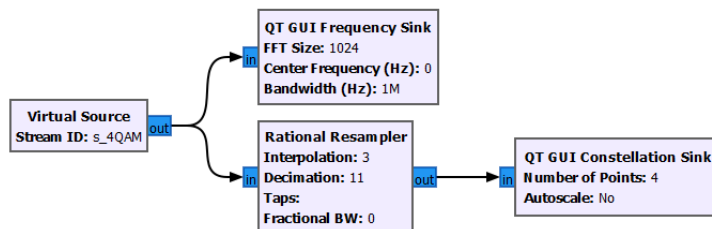


Figura 178. Conexión de bloques para visualización de espectro en frecuencia y dominio del tiempo de la señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

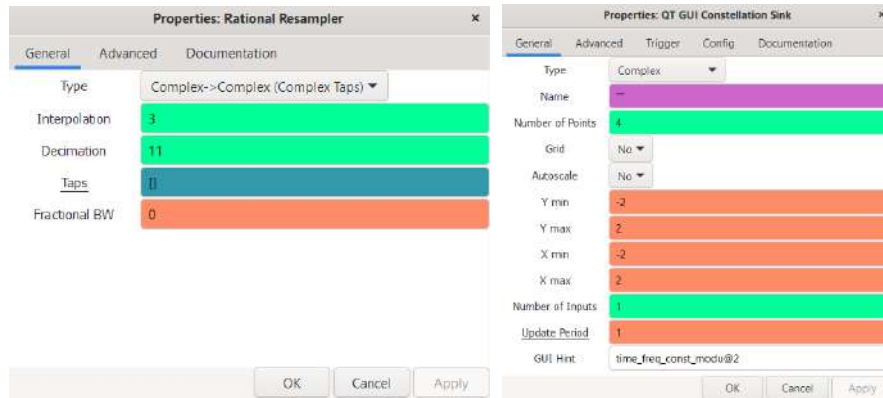


Figura 179. Configuración de bloques Rational Resampler y QT GUI Constellation Sink para visualización del diagrama de constelación en 4QAM en GNU Radio en un entorno simulado. Fuente propia

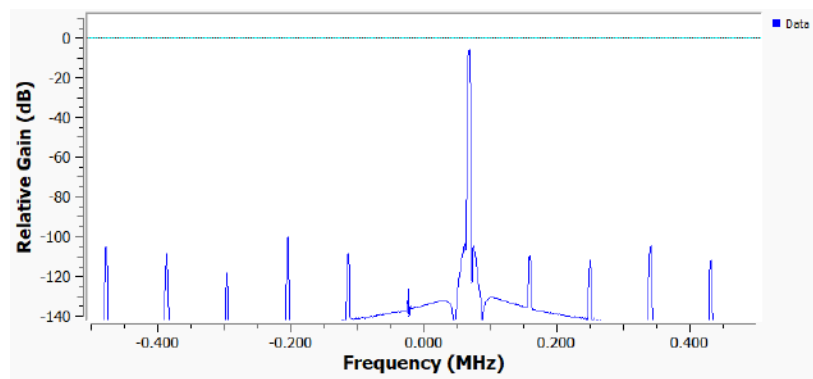


Figura 180. Espectro en frecuencia de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

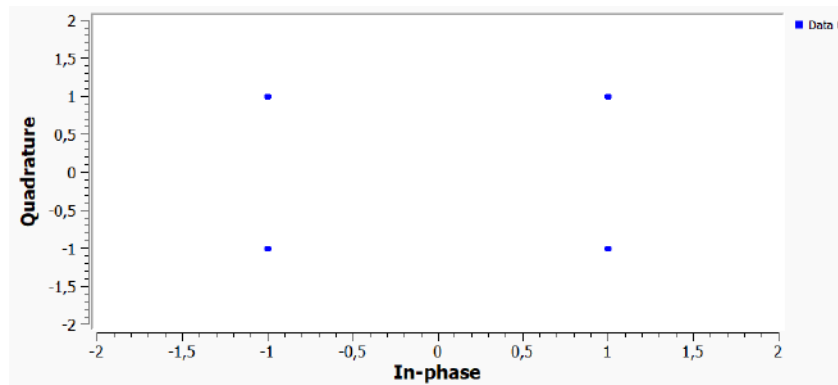


Figura 181. Diagrama de constelación de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

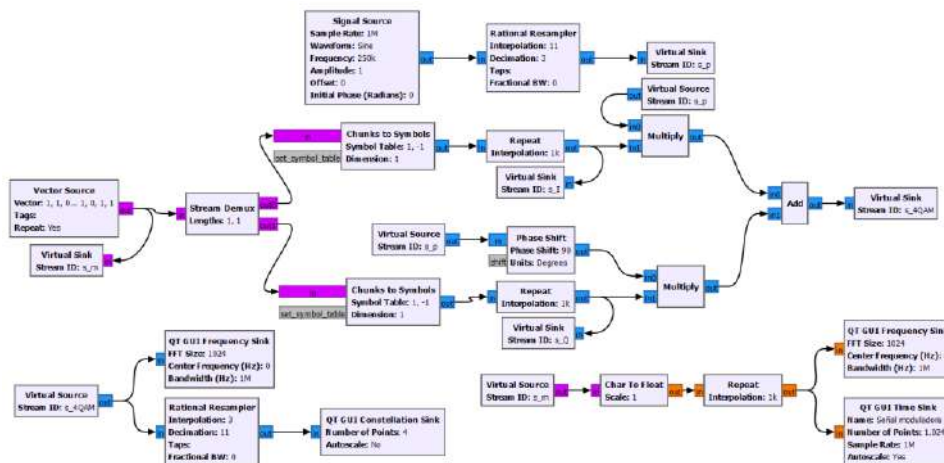


Figura 182. Esquema de modulador 4QAM en GNU Radio en un entorno simulado. Fuente propia

8.7.6.2. Desarrollo del esquema demodulador en GNU Radio

Para la implementación del demodulador 4QAM en GNU Radio, se siguieron una serie de pasos que permitieron la correcta recepción y demodulación de la señal, iniciando una etapa de sincronización y multiplicación de portadora, lo cual fue logrado a través del bloque Costas Loop (figura 183), configurado con un valor de 4 para el parámetro de “order”, puesto que en este caso la técnica de modulación aplicada en el modulador es 4QAM.

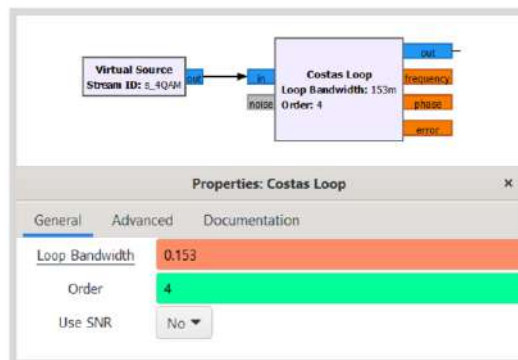


Figura 183. Bloque Costas Loop y su configuración para demodulación de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

Segundamente, se utilizó un bloque Moving Average, para promediar las muestras y eliminar el ruido presente en la señal, logrando demodular de forma correcta la señal (figura 184).

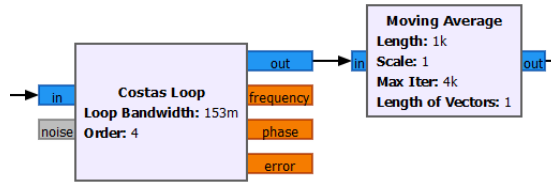


Figura 184. Conexión del bloque Costas Loop y Moving Average para demodulación 4QAM en GNU Radio en un entorno simulado. Fuente propia

Como paso siguiente, se añadió un bloque *Complex To Float* para separar la parte real de la imaginaria (figura 185).



Figura 185. bloque Complex To Float para demodulación 4QAM en GNU Radio en un entorno simulado. Fuente propia

Posteriormente, se aplicó un bloque *Threshold* a fin de adecuar la señal, y seguidamente eliminar las muestras añadidas a cada canal en el modulador 4QAM, a través de los bloques *Integrate* (figura 187), sin embargo, estos bloques generan una suma de las muestras añadidas, aumentando la amplitud de la señal que ingresa al bloque, por lo que, de forma continua se aplica un bloque *Threshold* para limitar la amplitud de las señales de los canales en un rango de 0 a 1 (figura 186).

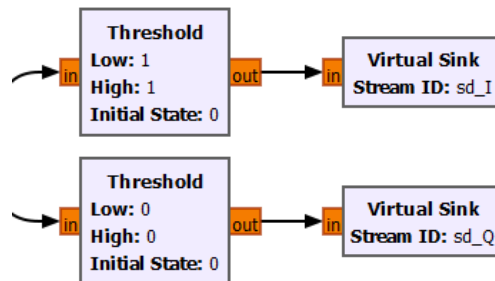


Figura 186. Bloques Threshold para adecuación de señal para demodulación en 4QAM en un entorno simulado en GNU Radio. Fuente propia

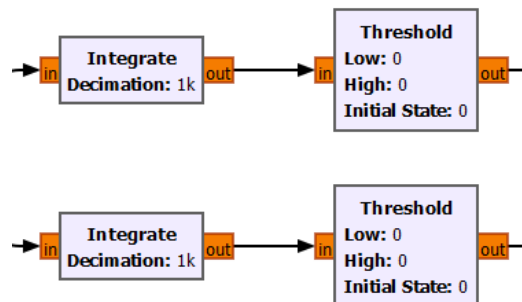


Figura 187. Conexión de los bloques Integrate y Threshold para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia



Figura 188. Configuración de los bloques *Integrate* y *Threshold* para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia

Una vez adecuadas las señales de los canales I y Q, se unieron a través de un bloque *Stream Mux*, siendo la salida de este mismo, ingresada a un bloque *Repeat* configurado con el mismo valor calculado en el modulador (figura 189, figura 190).



Figura 189. Bloque *Stream Mux* para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia

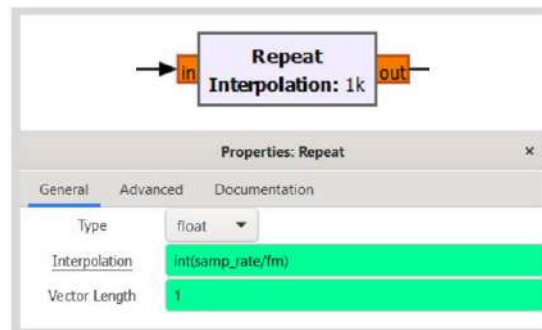


Figura 190. Bloque *Repeat* y su configuración para demodulación de señal 4QAM en GNU Radio en un entorno simulado. Fuente propia

Finalmente, para la visualización tanto de la señal modulada y demodulada en 4QAM, se utilizaron los bloques *QT GUI Tab Widget* y *QT GUI Time Sink* (figura 191, figura 192).



Figura 191. Bloques *QT GUI Tab Widget* y su configuración para visualización de señal modulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

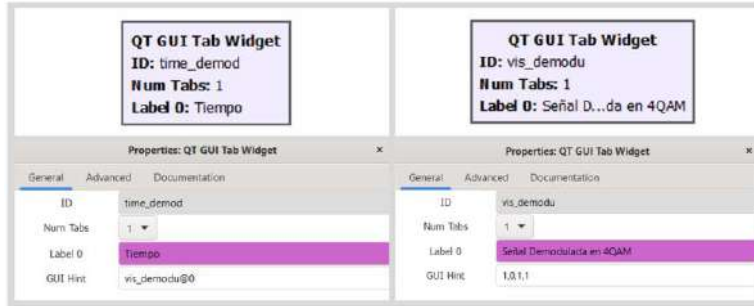


Figura 192. Bloques QT GUI Tab Widget y su configuración para visualización de señal demodulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

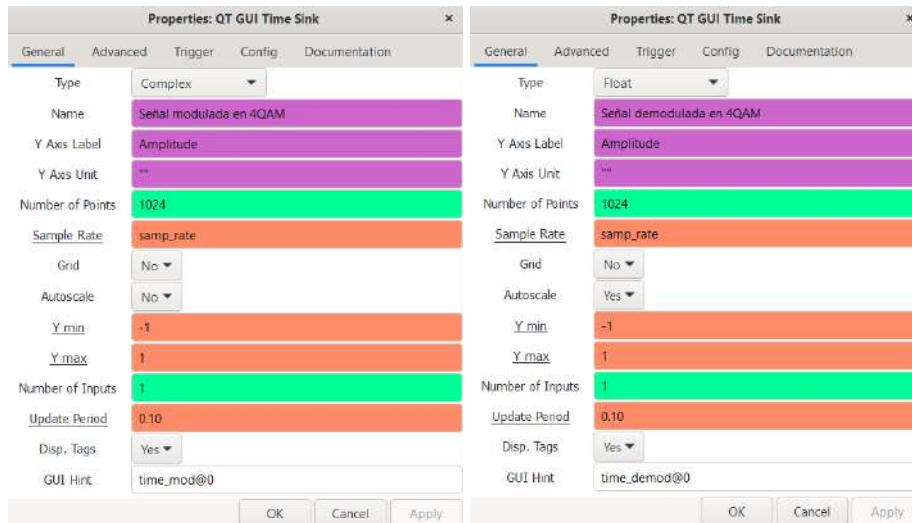


Figura 193. Configuración de los bloques QT GUI Time Sink para visualización de la señal modulada y de demodulada en 4QAM en GNU Radio en un entorno simulado. Fuente propia

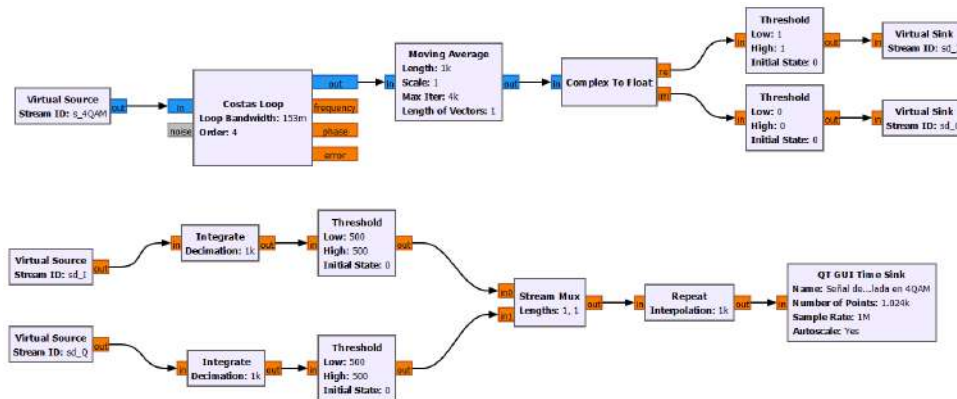


Figura 194. Esquema final del demodulador 4QAM en GNU Radio en un entorno simulado. Fuente propia

8.7.6.3. Implementación de modulador en GNU Radio usando los dispositivos SDR

La implementación del modulador utilizando el hardware SDR fue llevada a cabo siguiendo el esquema anteriormente desarrollado en la sección de simulación, siendo adaptado a la práctica añadiendo únicamente con el bloque *Osm com Sink*

y la configuración de este mismo, por lo que, se mantiene la estructura base del proyecto original y permitiendo la integración del dispositivo SDR transmisor (Hack RF One).

- **Device Arguments:** hackrf=0.
- **Sample Rate (sps):** 1 M
- **Ch0: Frequency (Hz):** 900 M
- **Ch0: RF Gain (dB):** 24
- **Ch0: IF Gain (dB):** 28

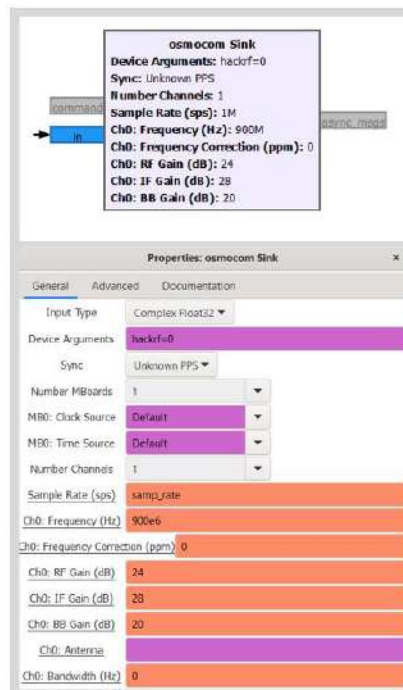


Figura 195. Bloque *osmocomb sink* y su configuración para 4QAM en un entorno práctico en GNU Radio. Fuente propia.

8.7.6.4. Implementación de demodulador en GNU Radio usando los dispositivos SDR

Para la implementación del esquema demodulador 4QAM en GNU Radio haciendo uso del hardware SDR, se reutilizó el mismo esquema desarrollado para la implementación en un entorno simulado, y aplicando pequeñas adaptaciones para el entorno práctico, siendo esta, la incorporación del bloque *Osmocom Source*, configurado con los siguientes parámetros:

- **Device Arguments:** rtl=0.
- **Sample Rate (sps):** 1 M
- **Ch0: Frequency (Hz):** 900 M
- **Ch0: RF Gain (dB):** 24

- Ch0: IF Gain (dB): 28

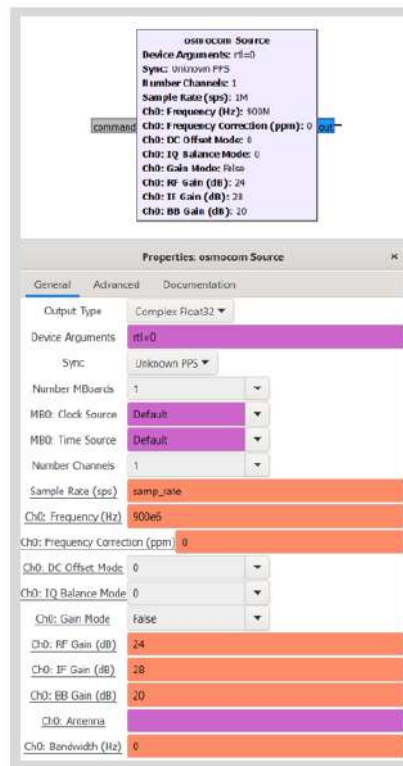


Figura 196. Bloque Osmocom Source y su configuración para transmisión de señal 4QAM en un entorno práctico en GNU Radio. Fuente propia

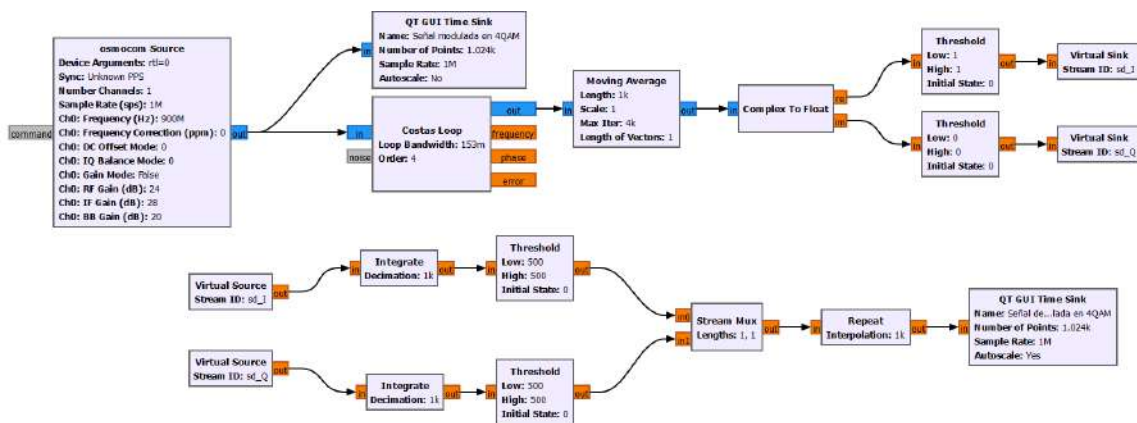


Figura 197. Esquema final de demodular 4QAM en GNU Radio en un entorno real. Fuente propia

9. RESULTADOS

Con la implementación del diseño metodológico propuesto en la sección 8, se logró la correcta transmisión y recepción de señales moduladas en AM, FM, ASK, FSK, BPSK y QAM en un entorno real, garantizando su compatibilidad con el aplicativo de GNU Radio y un desempeño estable. Por otro lado, se ajustaron parámetros como la frecuencia de portadora, frecuencia de moduladora y el índice de modulación, observando así que cada técnica de modulación posee características que influyen en la calidad de la señal transmitida y recibida.

Durante las pruebas, se identificaron factores que afectan en gran medida la calidad de la señal, tales como la presencia de ruido e interferencias. Por lo que, una forma de minimizar estos efectos, fue mediante el ajuste de parámetros de transmisión y aplicación de técnicas de filtrado, obteniendo una mejora altamente notable en la recepción y recuperación de la información transmitida.

Además, se logró desarrollar un manual de usuario y una serie de guías prácticas de laboratorio que facilitaran a los estudiantes de la Universidad Popular del Cesar el aprendizaje de los conceptos relacionados con la modulación y demodulación de señales a través de dispositivos SDR. Así mismo, en esta documentación se proporcionan instrucciones detalladas para la configuración de los dispositivos, la implementación de técnicas de modulación y el análisis de señales, permitiendo integrar un enfoque práctico en el proceso de enseñanza-aprendizaje.

A continuación, son presentados los entornos gráficos desarrollados para cada uno de los sistemas de modulación implementados en GNU Radio, así como los resultados obtenidos a partir de las pruebas realizadas. Estos resultados se han generado bajo los parámetros explicados en la sección anterior y utilizando los módulos de transmisión y recepción disponibles.

9.1. Interfaz gráfica del transmisor y receptor de la modulación en amplitud (AM)

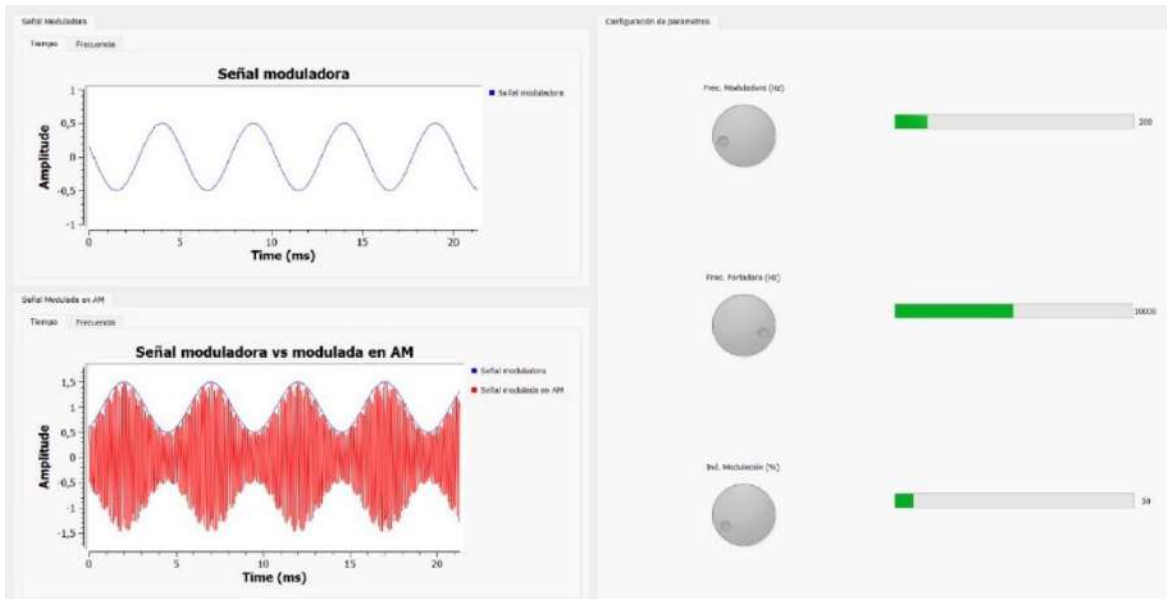


Figura 198. Interfaz gráfica de transmisor AM. Fuente propia

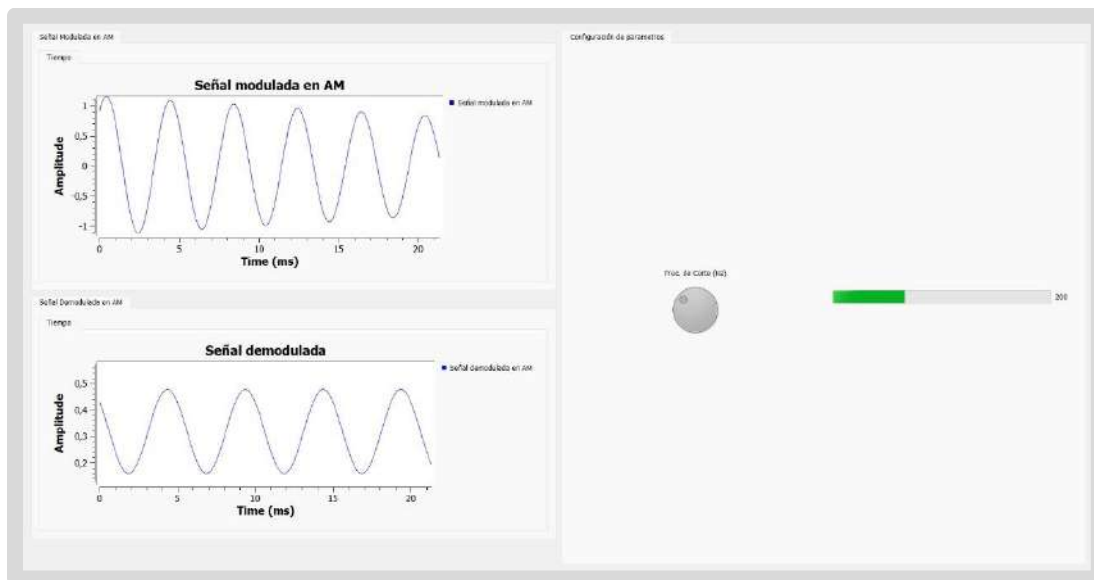


Figura 199. Interfaz gráfica de receptor AM. Fuente propia

9.1.1. Pruebas realizadas



Figura 200. Prueba realizada de modulador AM. Fuente propia

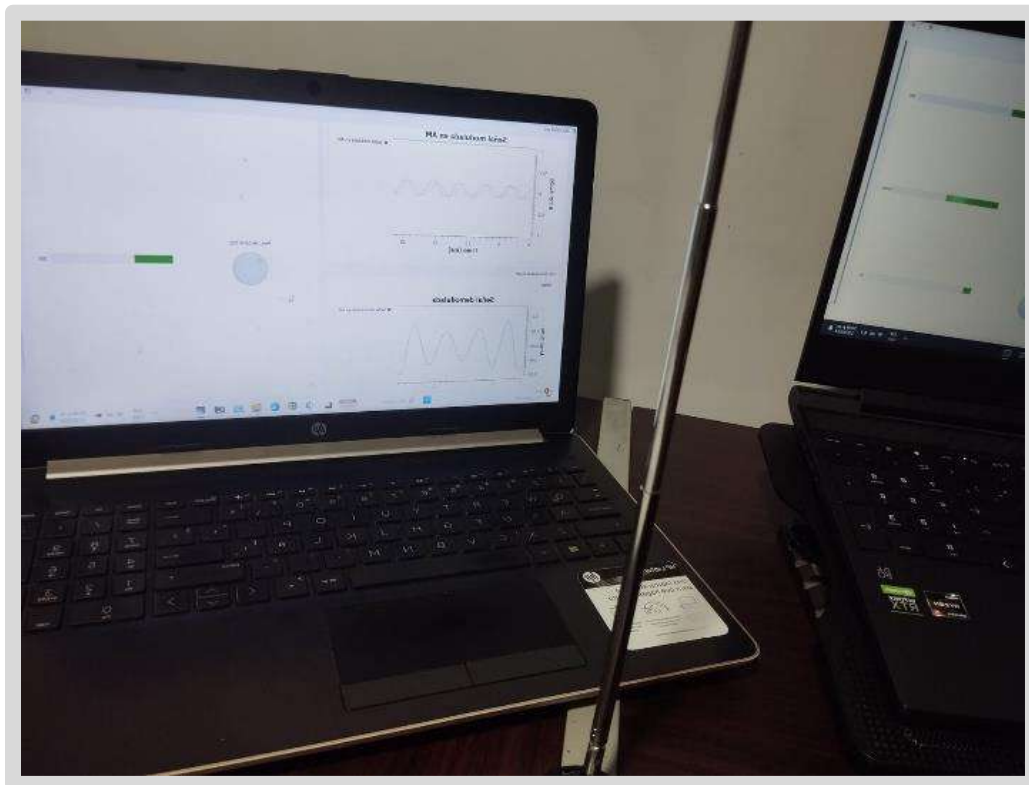


Figura 201. Prueba realizada de demodulador AM. Fuente propia

9.2. Interfaz gráfica del transmisor y receptor de la modulación en frecuencia (FM)

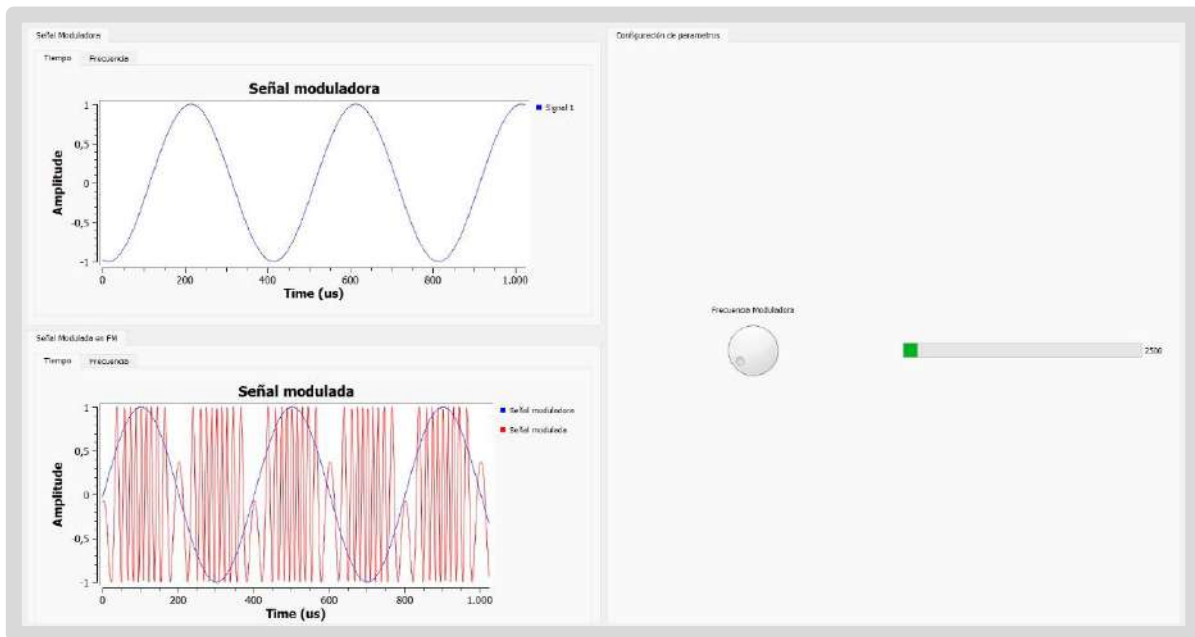


Figura 202. Interfaz gráfica de modulador FM. Fuente propia



Figura 203. Interfaz gráfica de demodulador FM. Fuente propia

9.2.1. Pruebas realizadas

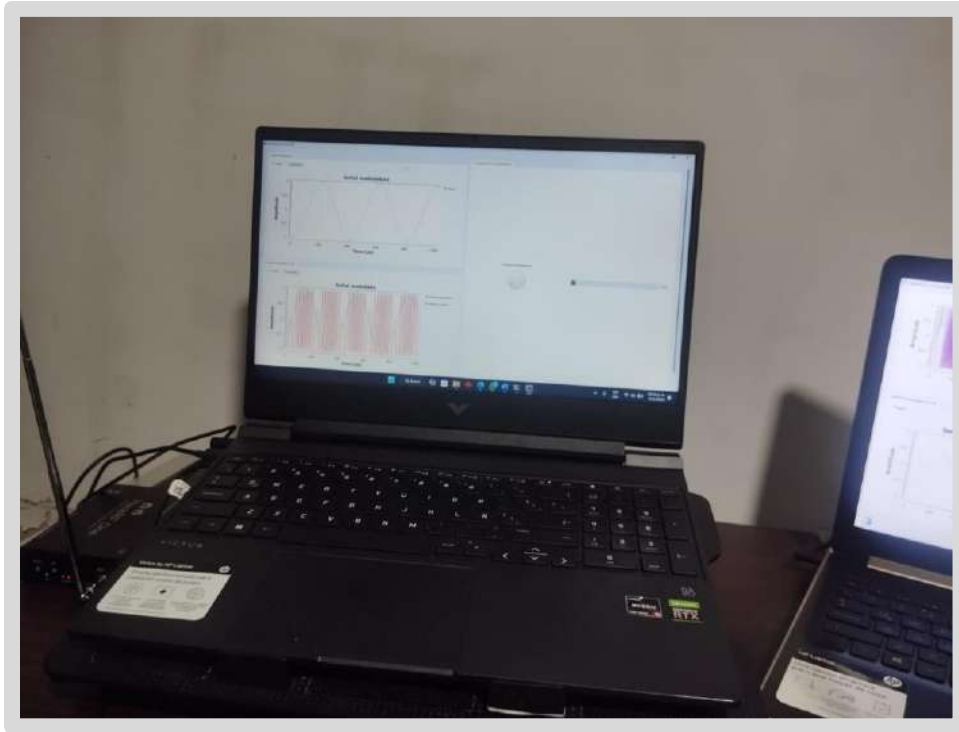


Figura 204. Prueba realizada de modulador FM. Fuente propia

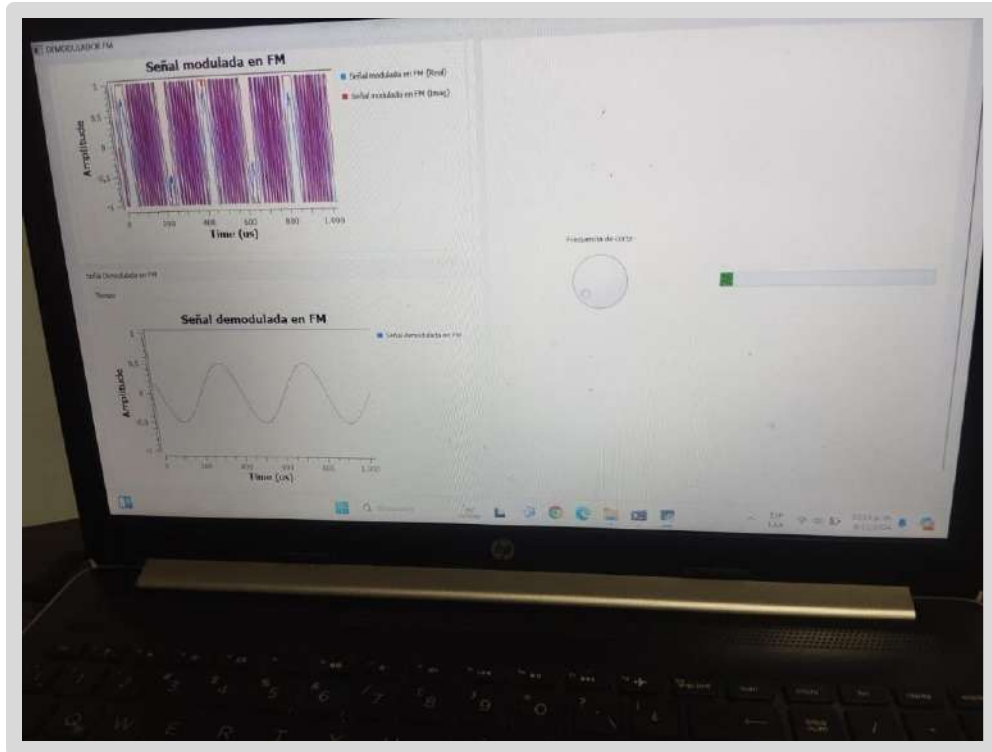


Figura 205. Prueba realizada de demodulador FM. Fuente propia

9.3. Interfaz gráfica del transmisor y receptor de la modulación On-Off Keying (OOK)



Figura 206. Interfaz gráfica de modulador OOK. Fuente propia



Figura 207. Interfaz gráfica de demodulador OOK. Fuente propia

9.3.1. Pruebas realizadas



Figura 208. Prueba realizada de modulador OOK. Fuente propia

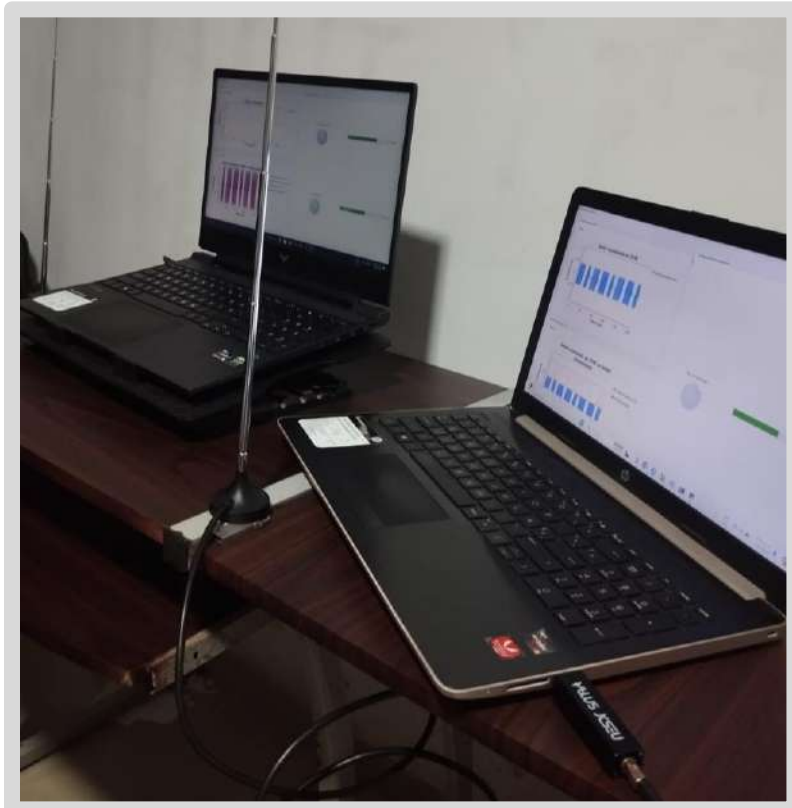


Figura 209. Prueba realizada de demodulador OOK. Fuente propia

9.4. Interfaz gráfica del transmisor y receptor de la modulación por desplazamiento de frecuencia (FSK)



Figura 210. Interfaz gráfica de modulador FSK. Fuente propia

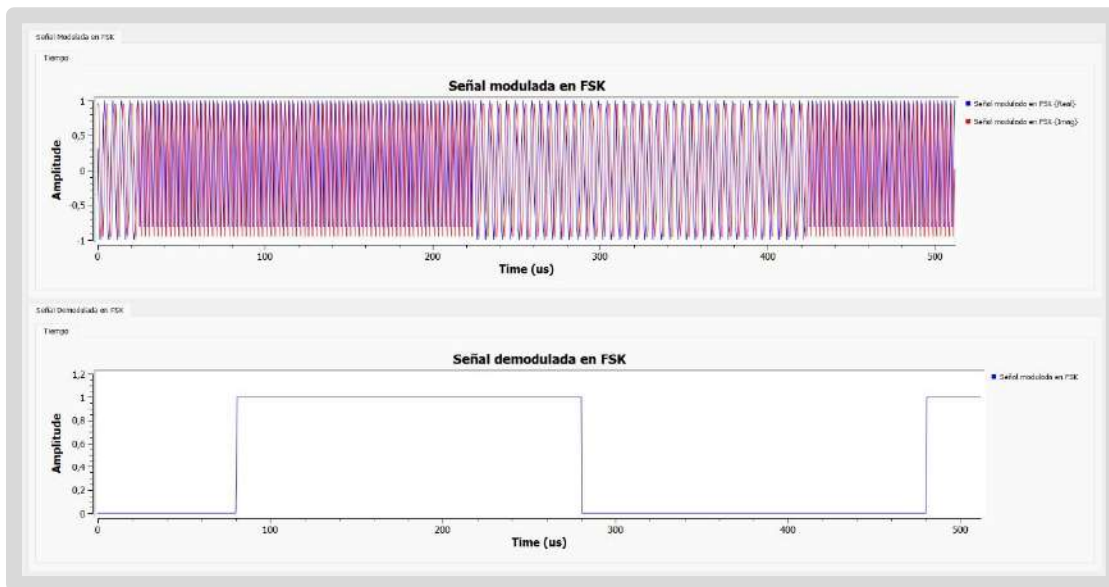


Figura 211. Interfaz gráfica de demodulador FSK. Fuente propia

9.4.1. Pruebas realizadas

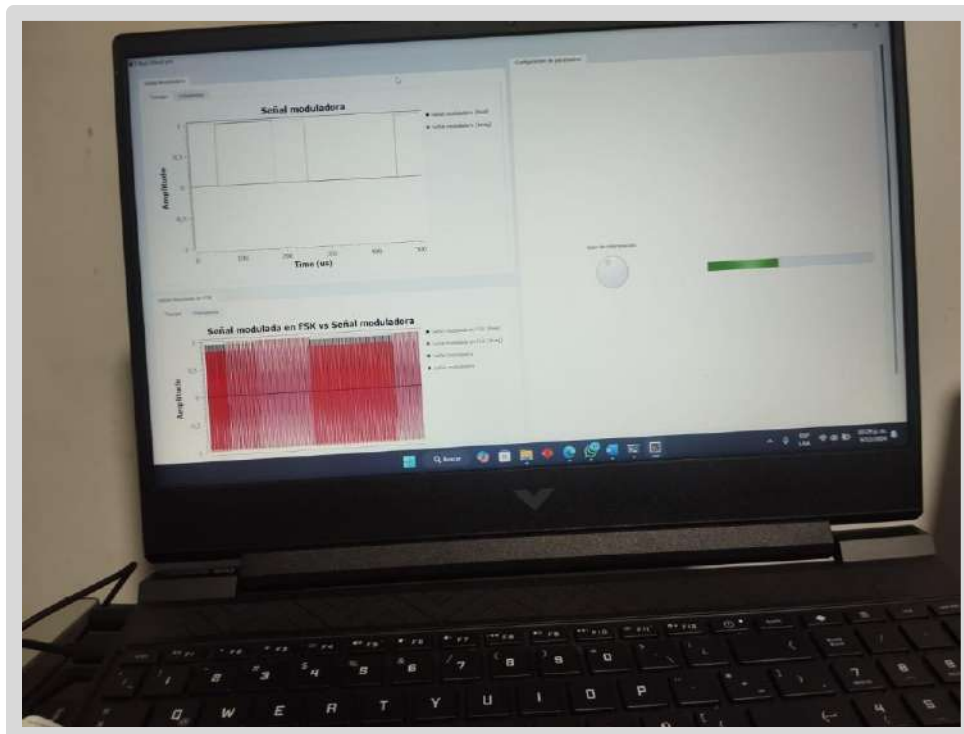


Figura 212. Prueba realizada de modulador FSK. Fuente propia

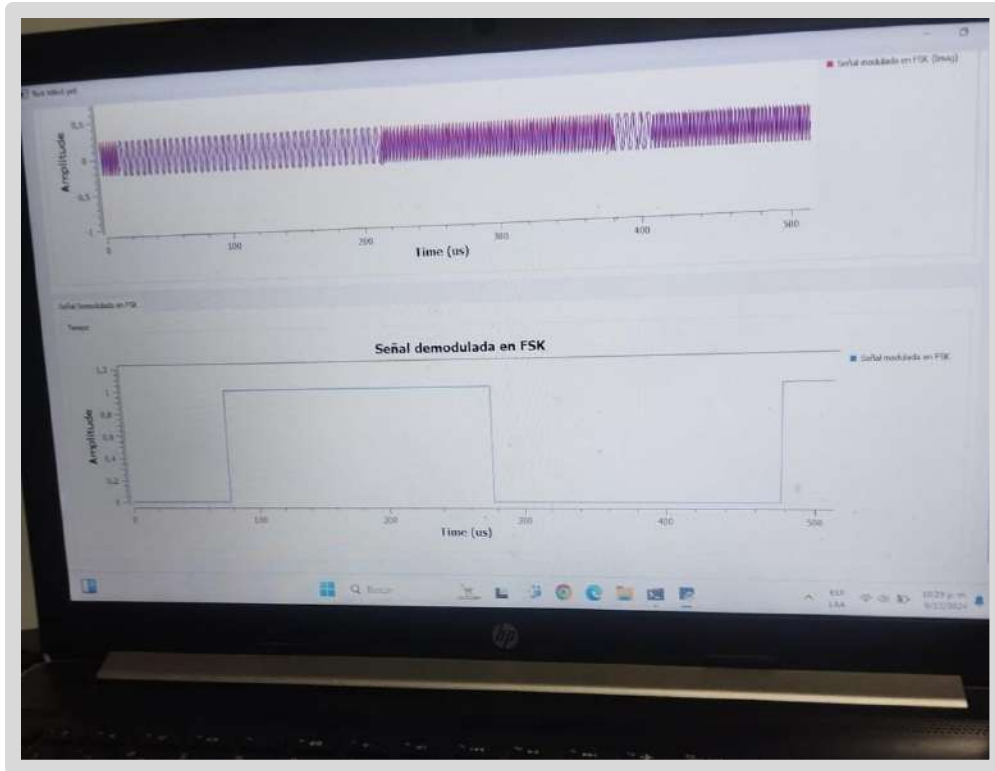


Figura 213. Prueba realizada de demodulador FSK. Fuente propia

9.5. Interfaz gráfica del transmisor y receptor de la modulación por desplazamiento de fase binaria (BPSK)

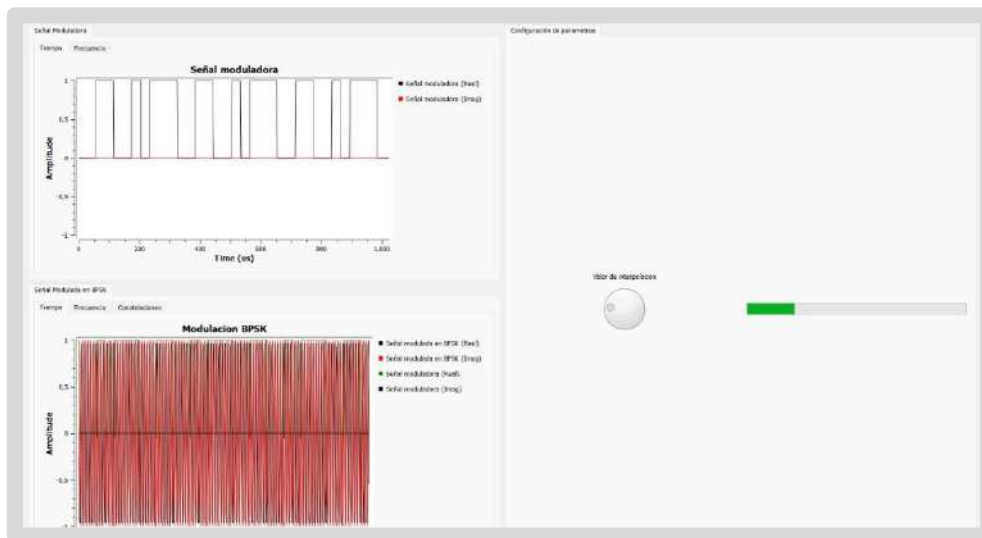


Figura 214. Interfaz gráfica de modulador BPSK. Fuente propia

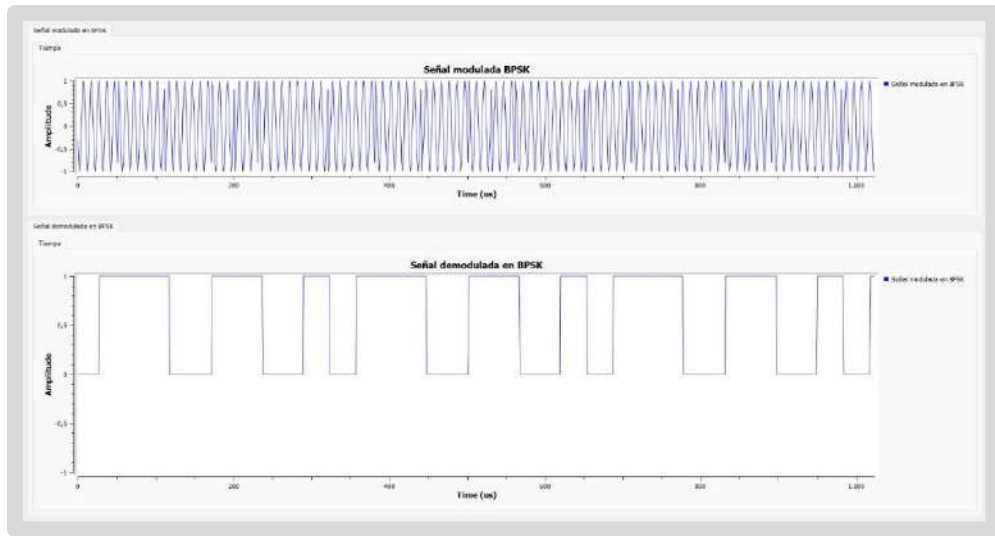


Figura 215. Interfaz gráfica de demodulador BPSK. Fuente propia

9.5.1. Pruebas realizadas

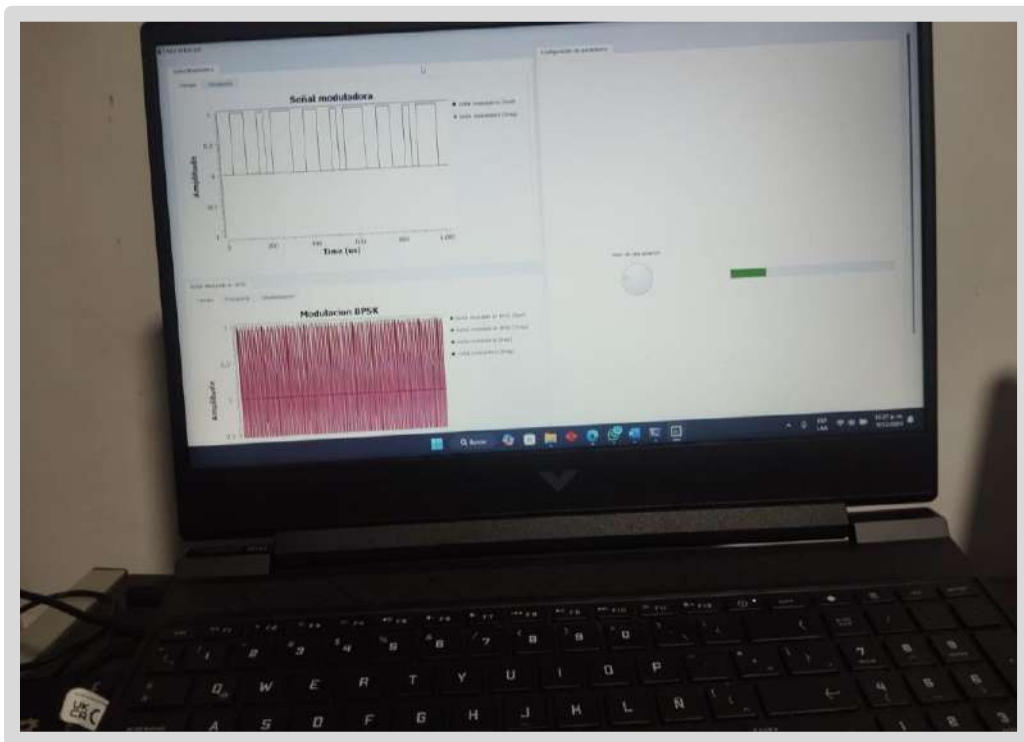


Figura 216. Prueba realizada de modulador BPSK. Fuente propia

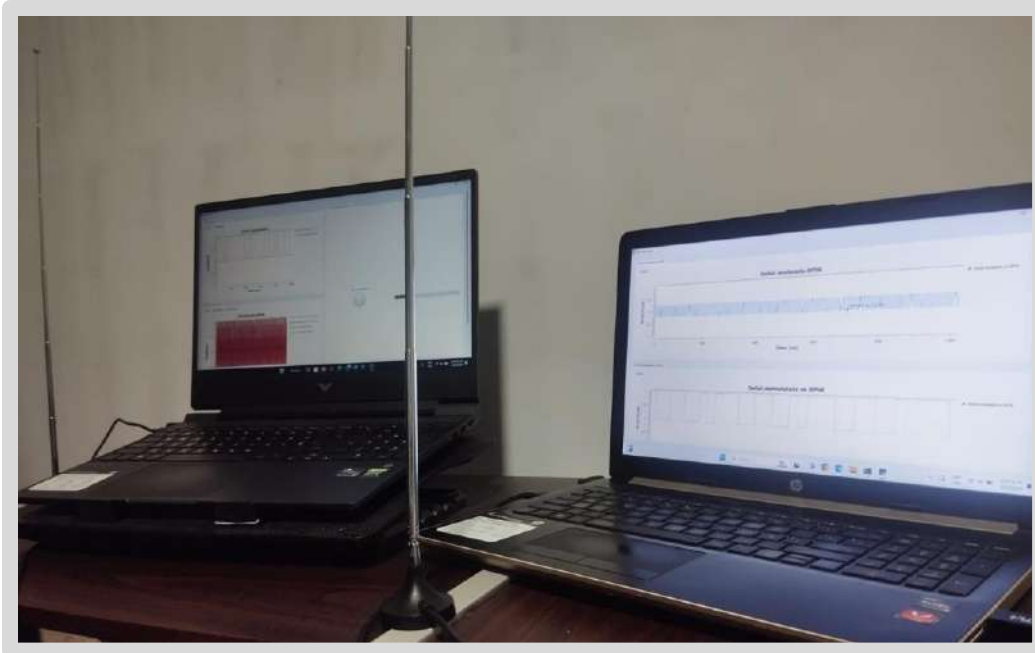


Figura 217. Prueba realizada de demodulador BPSK. Fuente propia

9.6. Interfaz gráfica del transmisor y receptor de la modulación de amplitud en cuadratura (QAM)



Figura 218. Interfaz gráfica de modulador 4QAM. Fuente propia



Figura 219. Interfaz gráfica de demodulador 4QAM. Fuente propia

9.6.1. Pruebas realizadas

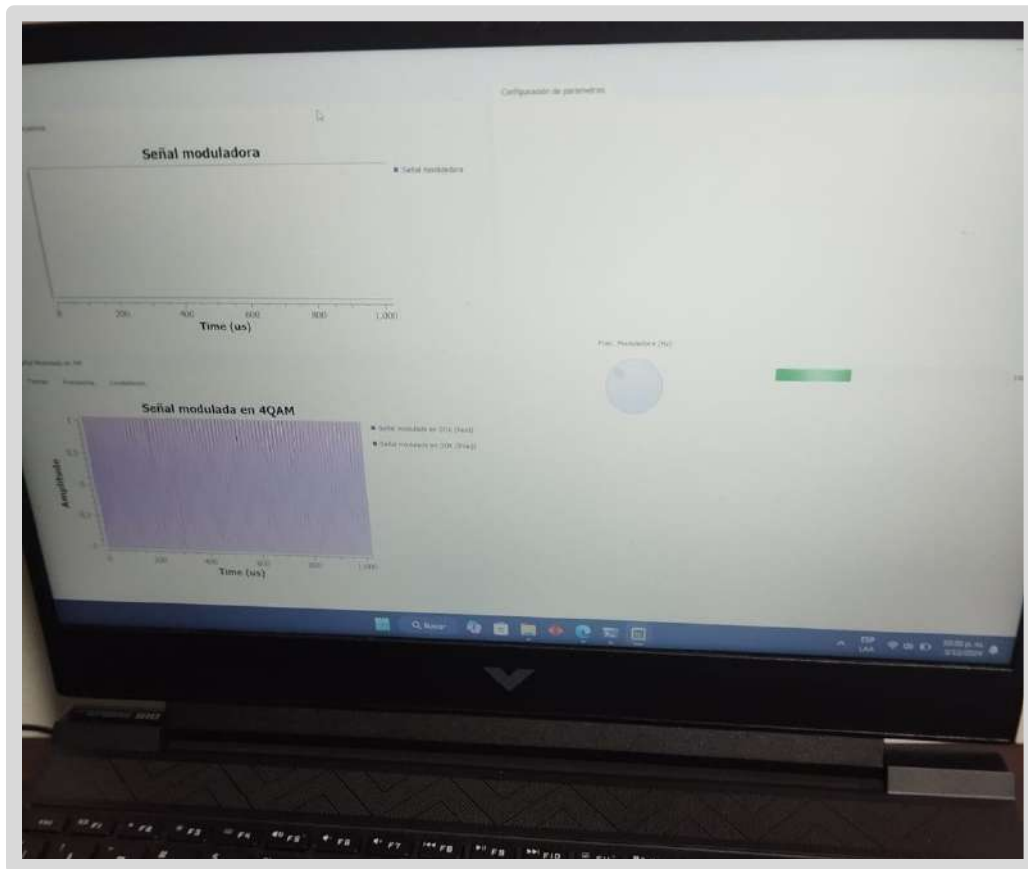


Figura 220. Prueba realizada de modulador 4QAM. Fuente propia

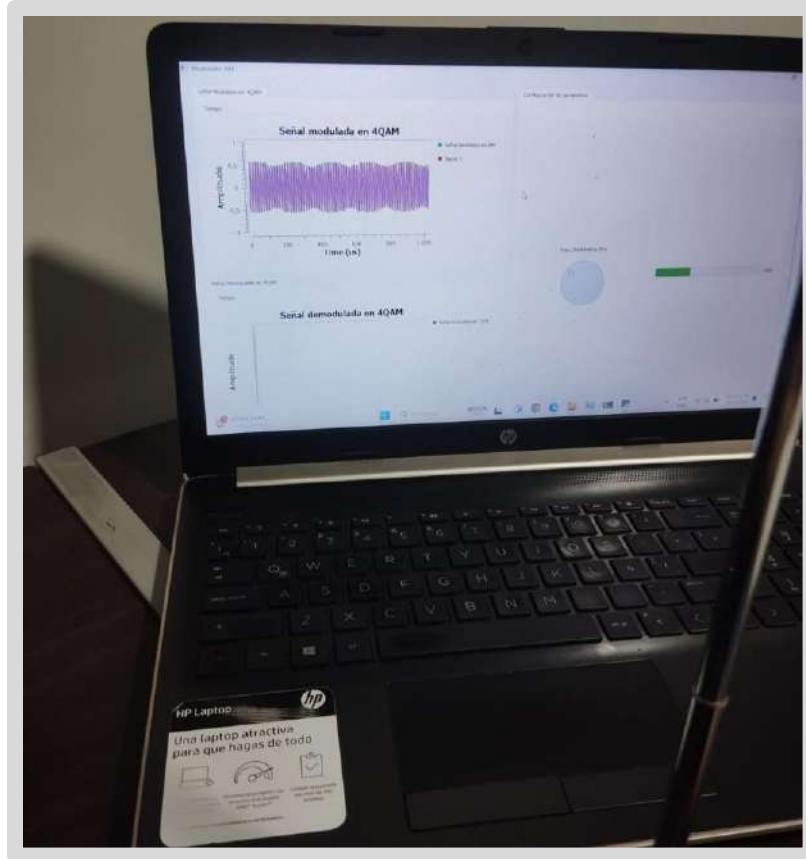


Figura 221. Prueba realizada de demodulador 4QAM. Fuente propia



Figura 222. Vista del Entrenador de Radio Definido por Software con hardware SDR integrado en maletín portátil. Fuente propia



Figura 223. Vista del Entrenador de Radio Definido por Software con hardware SDR integrado en maletín portátil. Fuente propia




Figura 224. Vista del Entrenador de Radio Definido por Software con hardware SDR integrado en maletín portátil. Fuente propia

10. CONCLUSIONES

La evaluación de los distintos aplicativos disponibles para la configuración de dispositivos SDR, evidenció la gran cantidad de opciones que existen hoy día, así mismo, dio a conocer las debilidades y fortalezas que posee cada una de estas. No obstante, todas resultan ser herramientas imprescindibles en el desarrollo e implementación de los sistemas de radio definido por software, puesto que, permiten realizar configuraciones precisas y flexibles basando en las necesidades que requiera el sistema.

Dentro de los aplicativos evaluados, se destacó GNU Radio, debido a su versatilidad y fácil manejo, por lo que el uso de este mismo para el modelado y simulación de las técnicas de modulación AM, FM, ASK, FSK, PSK y QAM, demostró ser una elección acertada. Este aplicativo no solo permite configuraciones precisas y flexibles, si no que a través de su extenso soporte documental proporciona un entendimiento más profundo con respecto a este tipo de sistemas de radiofrecuencia. No obstante, otros aplicativos como SDRSharp (SDR#), brindan una interfaz muy amigable y bastante útil para aquellos usuarios en búsqueda de soluciones rápidas para las recepciones de señales, sin embargo, no cuenta con opciones de transmisión de señales, por lo que, se ve muy limitado en comparación con GNU Radio.

En conclusión, la implementación de un entrenador de Radio definido por Software (SDR) en el laboratorio de comunicaciones de la Universidad Popular del Cesar ofrece una gran oportunidad en la modernización y enriquecimiento de la enseñanza práctica en este campo, permitiéndoles a los estudiantes interactuar con tecnología de vanguardia, y comprender a mayor profundidad las técnicas de modulación y transmisión de señales. Por otro lado, al contar con estos dispositivos, proporcionará un entorno más versátil para experimentar con diversas configuraciones de SDR y técnicas de modulación, mejorando en gran medida la calidad educativa al preparar a los futuros ingenieros electrónicos con habilidades y conocimientos fundamentales en la época actual. disponibilidad de equipos para realizar diferentes prácticas.

	PROGRAMA DE INGENIERÍA ELECTRÓNICA GUÍA DE LABORATORIO	VERSIÓN: 1
		PAG: 1 de #

PRACTICA No. 1

Nombre de la asignatura	
Código de la asignatura	
Título de la practica	
Docente	

CONTEXTUALIZACIÓN

1. Introducción a la modulación de amplitud (AM)

Una técnica de modulación primordial en las telecomunicaciones, es la modulación AM, siendo esta utilizada para la transmisión de información mediante variaciones de amplitud de una señal portadora (señal de alta frecuencia) en forma proporcional a la amplitud de la señal moduladora (señal de baja frecuencia). Existen múltiples variantes de esta modulación, no obstante, en este caso nos enfocamos en la doble banda lateral con full portadora (DSBFC), siendo esta, la aplicada con mayor frecuencia, debido a su fácil implementación.

2. Principios de la modulación

- **Señal portadora:** Señal sinusoidal de alta frecuencia encargada de transportar la información.
- **Señal moduladora:** Señal sinusoidal de baja frecuencia que contiene la información a transmitir.
- **Índice de modulación:** Relación entre la amplitud máxima de la señal moduladora y la amplitud máxima de la señal portadora, definiendo el grado de modulación, siendo este último el que afecta de forma directa el ancho de banda de la señal modulada.
- **Ancho de banda:** Rango de frecuencias que abarca la señal modulada.

2.3. Modulador y demodulador

- **Modulador:** Para obtener la señal modulada en AM, solo basta con aplicar operaciones matemáticas simples, iniciando con la suma de una constante a la moduladora, y posteriormente multiplicando la señal resultante con la portadora.
- **Demodulador:** Proceso inverso a la modulación, donde se recupera la señal que contiene la información.

OBJETIVO GENERAL

Implementar y analizar la modulación AM utilizando dispositivos SDR (HackRF One y Nooelec NESDR V5)

OBJETIVOS ESPECÍFICOS

- Configurar los dispositivos SDR para transmitir y recibir señales AM.
- Implementar un sistema de transmisión y recepción AM en GNU Radio, ajustando frecuencia de portadora y moduladora, y así mismo, el índice de modulación.
- Visualizar señales transmitidas y recibidas, evaluando la fidelidad de la señal demodulada.

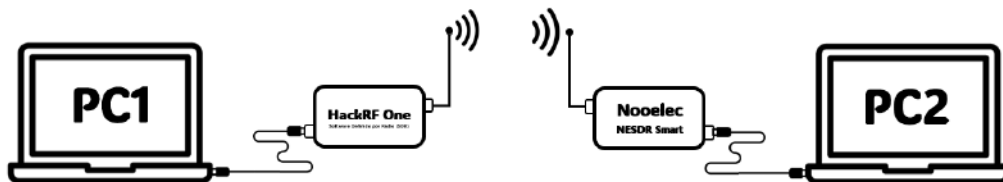
EQUIPOS Y MATERIALES

- 1 SDR transmisor (HackRF One)
- 1 SDR receptor (Nooelec NESDR V5)
- 2 antenas telescópicas
- 2 computadoras con Windows
- Aplicativo GNU Radio
- Drivers y cables de conexión para los dispositivos SDR

PROCEDIMIENTO

1. Preparación del entorno

- **Verificar la instalación de GNU Radio y los drivers:** Asegurarse que GNU Radio este correctamente instalado y el dispositivo SDR (HackRF One y Nooelec NESDR Smart V5) sea reconocidos de forma correcta.
- **Conexión de los dispositivos SDR**

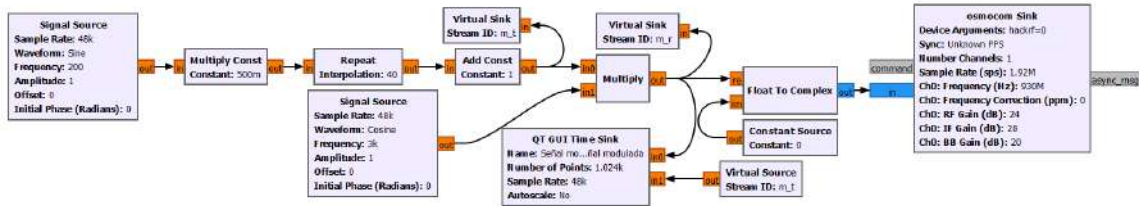


2. Implementación de modulador AM en GNU Radio

- Parámetros de diseño:
 - Frecuencia de portadora: 3 KHz
 - Frecuencia de moduladora: 100 Hz
 - Índice de modulación: 0.5
 - Frecuencia de transmisión: 930 MHz

- Sample rate: a consideración

- Esquema guía

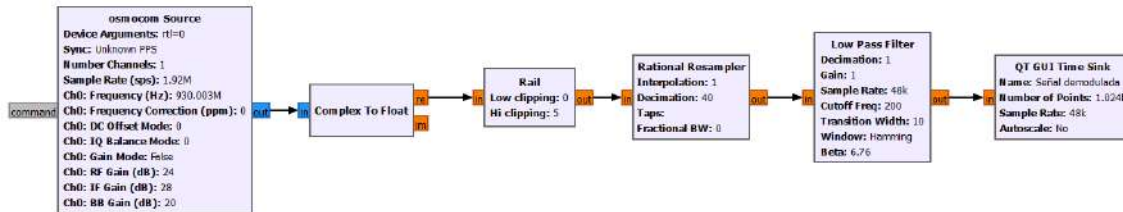


3. Implementación de demodulador en GNU Radio

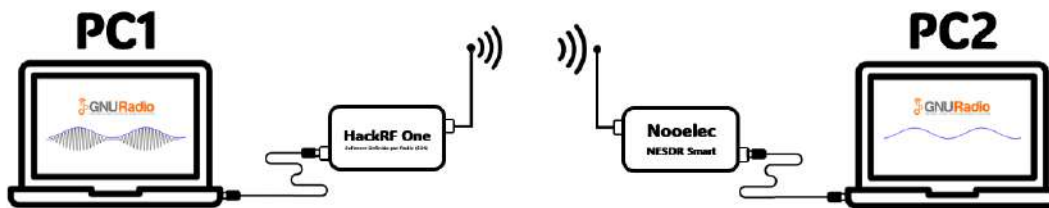
- Parámetros de diseño:

- Frecuencia de transmisión: 930.003 MHz
- Sample rate: A consideración
- Valores de demodulación en base a los establecidos en la modulación

- Esquema guía



4. Transmisión y recepción de la señal modulada en AM



5. Realice un informe escrito que incluya el desarrollo de la actividad, análisis de resultado y conclusiones.

RUBRICAS O PARÁMETROS DE EVALUACIÓN

1. Desarrollo de actividad (60%)

- Cálculos teóricos
- Implementación haciendo uso de los dispositivos SDR

2. Informe escrito (20%)

- Resumen del laboratorio
- Objetivos
- Introducción
- Marco teórico
- Procedimiento, materiales y equipos
- Resultados, interpretación, análisis y discusión de resultados
- Conclusiones
- Bibliografía

3. Sustentación (20%)

- Participación grupal
- Preguntas libres

4. Puntuación


- 0 puntos (no cumple)
- 1 a 5 puntos según el grado de cumplimiento

ENTREGABLES

1. Modulador de amplitud implementado con dispositivo SDR
2. Demodulador por detección de envolvente implementado con dispositivo SDR
3. Informe escrito

REFERENCIAS BIBLIOGRÁFICAS

- [1] https://wiki.gnuradio.org/index.php/Signal_Source
- [2] https://wiki.gnuradio.org/index.php/Multiply_Const
- [3] <https://wiki.gnuradio.org/index.php/Repeat>
- [4] <https://wiki.gnuradio.org/index.php/Multiply>
- [5] https://wiki.gnuradio.org/index.php/Constant_Source
- [6] https://wiki.gnuradio.org/index.php/Low_Pass_Filter
- [7] https://wiki.gnuradio.org/index.php/QT_GUI_Time_Sink
- [8] <https://wiki.gnuradio.org/index.php/Rail>
- [9] https://wiki.gnuradio.org/index.php/Audio_Sink

	UNIVERSIDAD POPULAR DEL CESAR	CÓDIGO:
		VERSIÓN: 1
	PROGRAMA DE INGENIERÍA ELECTRÓNICA GUÍA DE LABORATORIO	PAG: 1 de #

PRACTICA No. 2

Nombre de la asignatura

Código de la asignatura

Título de la practica

Docente

CONTEXTUALIZACIÓN

1. Introducción a la modulación de frecuencia (FM)

La modulación por frecuencia (FM) es una técnica en telecomunicaciones que varía la frecuencia de una señal portadora de acuerdo con la información contenida en la señal moduladora. FM es ampliamente utilizada en radio, televisión, y sistemas de comunicación digital debido a su resistencia al ruido y eficiencia espectral.

2. Principios de la modulación

- **Señal portadora:** Señal sinusoidal de alta frecuencia encargada de transportar la información.
- **Señal moduladora:** Señal sinusoidal de baja frecuencia que contiene la información a transmitir.
- **Sensibilidad de frecuencia (Kf):** Constante que determina el nivel de modulación.
- **Ancho de banda:** Estimado por la regla de Carson: $B \cong 2(\Delta f + f_m)$ donde Δf es el desvío de frecuencia y f_m es la frecuencia máxima de la señal moduladora.

2.3. Modulador y demodulador

- **Modulador:** Para obtener la señal modulada en FM, basta con utilizar un oscilador controlador por tensión (VCO), el cual produce una senoide cuya frecuencia depende de la amplitud de la señal de entrada y la desviación de frecuencia.
- **Demodulador:** La demodulación por cuadratura utiliza un derivador y un discriminador para recuperar la señal moduladora. En GNU Radio, esto se puede implementar usando el bloque Quadrature Demod.

OBJETIVO GENERAL

Implementar y analizar la modulación FM utilizando dispositivos SDR (HackRF One y Nooelec NESDR V5)

OBJETIVOS ESPECÍFICOS

- Configurar los dispositivos SDR para transmitir y recibir señales FM.
- Implementar un sistema de transmisión y recepción FM en GNU Radio, ajustando frecuencia de portadora y moduladora, y así mismo, el índice de modulación.
- Visualizar señales transmitidas y recibidas, evaluando la fidelidad de la señal demodulada.

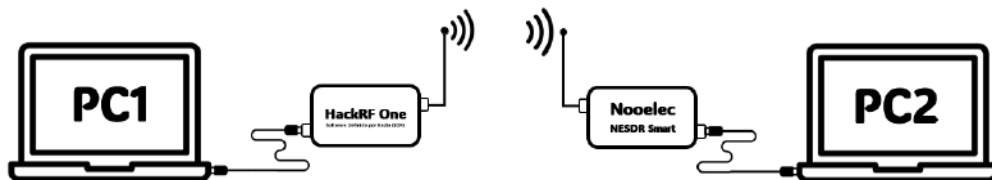
EQUIPOS Y MATERIALES

- 1 SDR transmisor (HackRF One)
- 1 SDR receptor (Nooelec NESDR V5)
- 2 antenas telescópicas
- 2 computadoras con Windows
- Aplicativo GNU Radio
- Drivers y cables de conexión para los dispositivos SDR

PROCEDIMIENTO

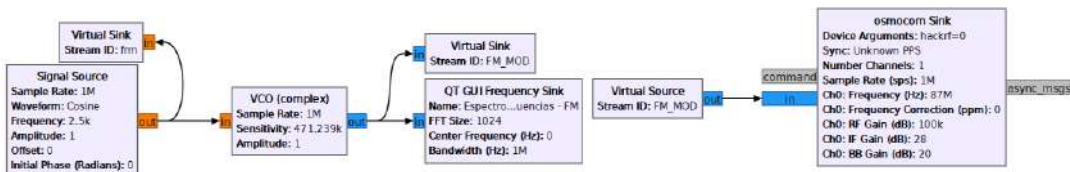
1. Preparación del entorno

- **Verificar la instalación de GNU Radio y los drivers:** asegurarse que GNU Radio este correctamente instalado y el dispositivo SDR (HackRF One y Nooelec NESDR Smart V5) sea reconocidos de forma correcta.
- **Conexión de los dispositivos SDR**



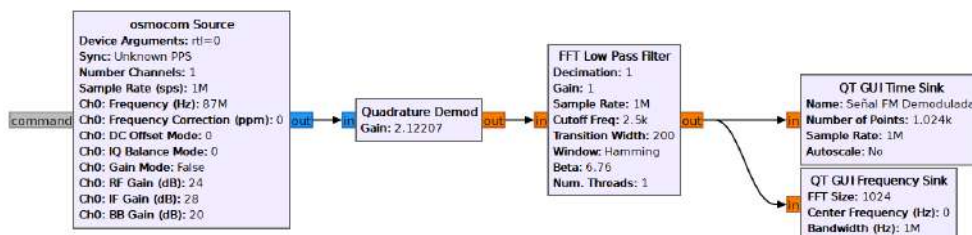
2. Implementación de modulador FM en GNU Radio

- Parámetros de diseño:
 - Frecuencia moduladora: 2.5 KHz
 - Desviación de frecuencia: 75 KHz
 - Frecuencia de transmisión: 87 MHz
 - Sample rate: a consideración
- Esquema guía

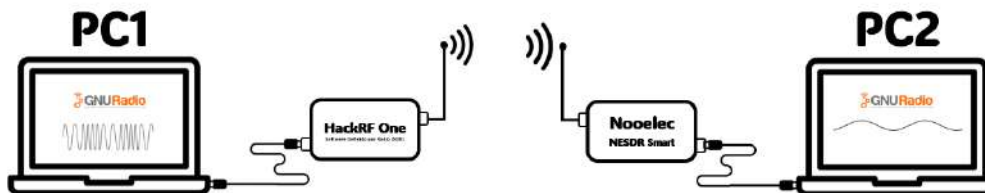


3. Implementación de demodulador en GNU Radio

- Parámetros de diseño:
 - Frecuencia de transmisión: 87 MHz
 - Sample rate: A consideración
 - Valores de demodulación en base a los establecidos en la modulación
- Esquema guía



4. Transmisión y recepción de la señal modulada en FM



5. Realice un informe escrito que incluya el desarrollo de la actividad, análisis de resultado y conclusiones.

RUBRICAS O PARÁMETROS DE EVALUACIÓN

1. Desarrollo de actividad (60%)

- Cálculos teóricos
- Implementación haciendo uso de los dispositivos SDR

2. Informe escrito (20%)

- Resumen del laboratorio
- Objetivos
- Introducción

- Marco teórico
- Procedimiento, materiales y equipos
- Resultados, interpretación, análisis y discusión de resultados
- Conclusiones
- Bibliografía

3. Sustentación (20%)

- Participación grupal
- Preguntas libres

4. Puntuación


- 0 puntos (no cumple)
- 1 a 5 puntos según el grado de cumplimiento

ENTREGABLES

1. Modulador de frecuencia implementado con dispositivo SDR
2. Demodulador por cuadratura implementado con dispositivo SDR
3. Informe escrito

REFERENCIAS BIBLIOGRÁFICAS

- [1] https://wiki.gnuradio.org/index.php/Signal_Source
 [2] https://wiki.gnuradio.org/index.php/Multiply_Const
 [3] <https://wiki.gnuradio.org/index.php/Repeat>
 [4] <https://wiki.gnuradio.org/index.php/Multiply>
 [5] [https://wiki.gnuradio.org/index.php/VCO_\(complex\)](https://wiki.gnuradio.org/index.php/VCO_(complex))
 [6] https://wiki.gnuradio.org/index.php/Low_Pass_Filter
 [7] https://wiki.gnuradio.org/index.php/QT_GUI_Time_Sink
 [8] https://wiki.gnuradio.org/index.php/Quadrature_Demod
 [9] https://wiki.gnuradio.org/index.php/Audio_Sink

	UNIVERSIDAD POPULAR DEL CESAR	CÓDIGO:
		VERSIÓN: 1
	PROGRAMA DE INGENIERÍA ELECTRÓNICA GUÍA DE LABORATORIO	PAG: 1 de #

PRACTICA No. 3

Nombre de la asignatura

Código de la asignatura

Título de la practica

Docente

CONTEXTUALIZACIÓN

1. Introducción a la modulación por desplazamiento de amplitud (ASK)

Una de las técnicas de modulación digital que presenta mayor simplicidad es la modulación por desplazamiento de amplitud (ASK), siendo OOK (On-Off Keying), una de las formas más básicas y ampliamente utilizada en aquellos sistemas de comunicación de baja complejidad. En esta ocasión, nos centraremos en el desarrollo del esquema modulador y demodulador de OOK, donde la presencia o ausencia de la señal portadora es la representación de los bits de datos.

2. Principios de la modulación

- **Señal portadora:** señal sinusoidal de alta frecuencia encargada de transportar los datos digitales.
- **Señal moduladora:** señal digital que varía entre dos niveles, representando los bits mediante dos valores '0' y '1'.

3. Modulador y demodulador

- **Modulador:** se basa básicamente en la activación y desactivación de la señal portadora. Esto es debido a que la señal moduladora es digital, compuesta por secuencias de unos y ceros, que, al ser multiplicados con señal portadora, se logra la modulación OOK.
- **Demodulador:** proceso inverso donde se lleva a cabo la recuperación de la señal digital original a partir de la señal modulada, generalmente aplicando la detección un detector de envolvente.

OBJETIVO GENERAL

Implementar, transmitir, recibir y analizar señales moduladas en OOK utilizando dispositivos SDR (HackRF One y Nooelec NESDR V5) para comprender el proceso y características de esta técnica de modulación.

OBJETIVOS ESPECÍFICOS

- Configurar los dispositivos SDR para transmitir y recibir señales OOK.
- Implementar un sistema de transmisión y recepción OOK en GNU Radio, ajustando frecuencia de portadora y moduladora, y así mismo, el índice de modulación.

- Visualizar señales transmitidas y recibidas, evaluando la fidelidad de la señal demodulada.

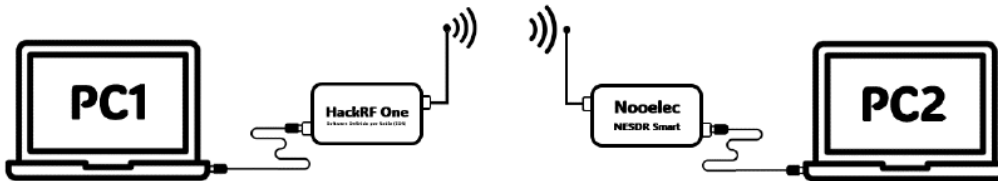
EQUIPOS Y MATERIALES

- 1 SDR transmisor (HackRF One)
- 1 SDR receptor (Nooelec NESDR V5)
- 2 antenas telescópicas
- 2 computadoras con Windows
- Aplicativo GNU Radio
- Drivers y cables para los dispositivos SDR

PROCEDIMIENTO

1. Preparación del entorno

- **Verificar la instalación de GNU Radio y los drivers:** asegurarse que GNU Radio este correctamente instalado y el dispositivo SDR (HackRF One y Nooelec NESDR Smart V5) sea reconocidos de forma correcta.
- **Conexión de los dispositivos SDR**

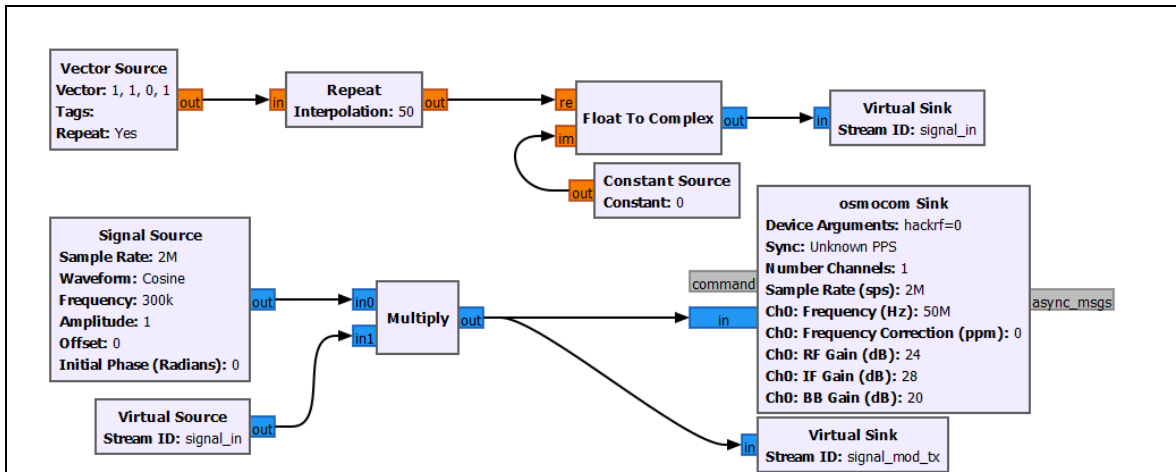


2. Implementación de modulador OOK en GNU Radio

4. Parámetros de diseño:

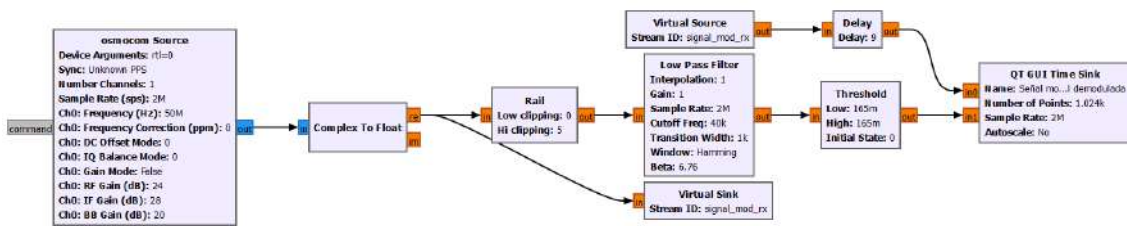
- Frecuencia de portadora: 300 KHz
- Frecuencia de moduladora: 40 KHz
- Frecuencia de transmisión: 50 MHz
- Sample rate: 2 M

5. Esquema guía

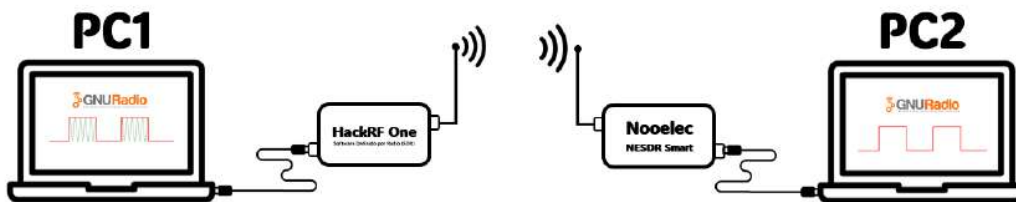


3. Implementación de demodulador OOK en GNU Radio

- Parámetros de diseño:
 - Frecuencia transmisión: 50 MHz
 - Sample rate: A consideración
 - Valores de demodulación en base a los establecidos en la modulación
- Esquema guía



4. Transmisión y recepción de la señal modulada en OOK



5. Realice un informe escrito que incluya el desarrollo de la actividad, análisis de resultado y conclusiones.

RUBRICAS O PARÁMETROS DE EVALUACIÓN

1. Desarrollo de actividad (60%)

- Cálculos teóricos
- Implementación haciendo uso de los dispositivos SDR

2. Informe escrito (20%)

- Resumen del laboratorio
- Objetivos
- Introducción
- Marco teórico
- Procedimiento, materiales y equipos
- Resultados, interpretación, análisis y discusión de resultados
- Conclusiones
- Bibliografía

3. Sustentación (20%)

- Participación grupal
- Preguntas libres

4. Puntuación

- 0 puntos (no cumple)
- 1 a 5 puntos según el grado de cumplimiento

ENTREGABLES

1. Esquema de modulador implementado con dispositivo SDR
2. Esquema de demodulador implementado con dispositivo SDR
3. Informe escrito

REFERENCIAS BIBLIOGRÁFICAS

- [1] https://wiki.gnuradio.org/index.php/Signal_Source
[2] https://wiki.gnuradio.org/index.php/Multiply_Const
[3] <https://wiki.gnuradio.org/index.php/Repeat>
[4] <https://wiki.gnuradio.org/index.php/Multiply>
[5] https://wiki.gnuradio.org/index.php/Constant_Source
[6] https://wiki.gnuradio.org/index.php/Low_Pass_Filter
[7] https://wiki.gnuradio.org/index.php/QT_GUI_Time_Sink
[8] <https://wiki.gnuradio.org/index.php/Rail>
[9] https://wiki.gnuradio.org/index.php/Audio_Sink

	UNIVERSIDAD POPULAR DEL CESAR	CÓDIGO:
		VERSIÓN: 1



PRACTICA No. 4

Nombre de la asignatura	
Código de la asignatura	
Título de la practica	
Docente	

CONTEXTUALIZACIÓN

1. Introducción a la modulación por desplazamiento de frecuencia (FSK)

La modulación por desplazamiento de frecuencia (FSK) es una técnica digital en la que los datos binarios se representan mediante variaciones discretas de frecuencia de una portadora. La técnica propuesta para desarrollar este diseño es la modulación de fase discontinua, donde la fase de la portadora no se mantiene continua durante las transiciones entre las frecuencias de marca y de espacio.

2. Principios de la modulación

- **Señal portadora:** señal sinusoidal de alta frecuencia encargada de transportar los datos digitales.
- **Señal moduladora:** señal digital que varía entre dos niveles, representando los bits mediante dos valores '0' y '1'.

3. Modulador y demodulador

- **Modulador:** Este esquema utiliza una señal binaria para alternar entre dos frecuencias generadas por osciladores, una para el valor 1 y otra para el 0. La modulación se logra mediante operaciones matemáticas simples como suma, resta y multiplicación, codificando los datos digitales en una señal analógica de frecuencia variable.
- **Demodulador:** proceso inverso donde se lleva a cabo la recuperación de la señal digital original a partir de la señal modulada, aplicando el método de detección por envolvente para la frecuencia de espacio y de marca.

OBJETIVO GENERAL

Implementar, transmitir, recibir y analizar señales moduladas en FSK utilizando dispositivos SDR (HackRF One y Nooelec NESDR V5) para comprender el proceso y características de esta técnica de modulación.

OBJETIVOS ESPECÍFICOS

- Configurar los dispositivos SDR para transmitir y recibir señales FSK.
- Implementar un sistema de transmisión y recepción FSK en GNU Radio, ajustando frecuencia de portadora y moduladora, y así mismo, el índice de modulación.
- Visualizar señales transmitidas y recibidas, evaluando la fidelidad de la señal demodulada.

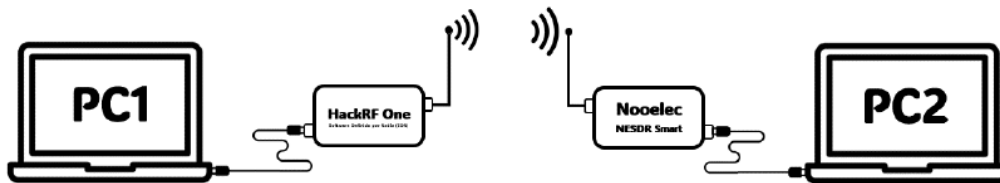
EQUIPOS Y MATERIALES

- 1 SDR transmisor (HackRF One)
- 1 SDR receptor (Nooelec NESDR V5)
- 2 antenas telescópicas
- 2 computadoras con Windows
- Aplicativo GNU Radio
- Drivers y cables para los dispositivos SDR

PROCEDIMIENTO

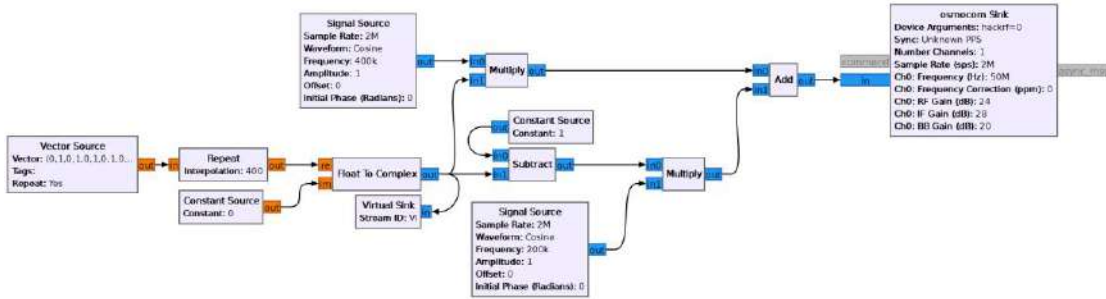
1. Preparación del entorno

- **Verificar la instalación de GNU Radio y los drivers:** asegurarse que GNU Radio este correctamente instalado y el dispositivo SDR (HackRF One y Nooelec NESDR Smart V5) sea reconocidos de forma correcta.
- **Conexión de los dispositivos SDR**



2. Implementación de modulador FSK en GNU Radio

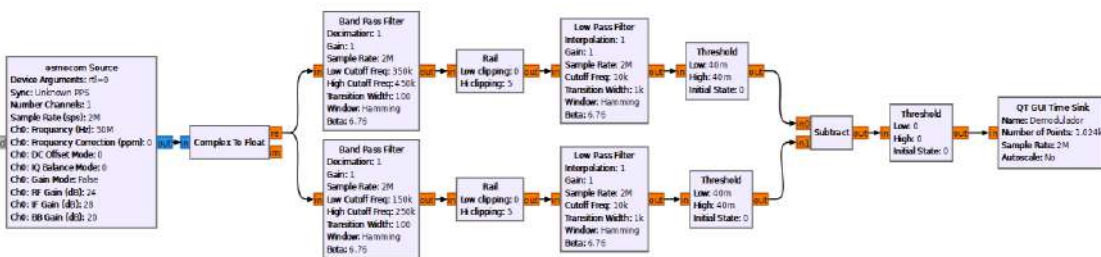
- Parámetros de diseño:
 - Frecuencia de marca: 400 KHz
 - Frecuencia de espacio: 200 KHz
 - Frecuencia de moduladora: 5 KHz
 - Frecuencia de transmisión: 50 MHz
 - Sample rate: 2 M
- Esquema guía



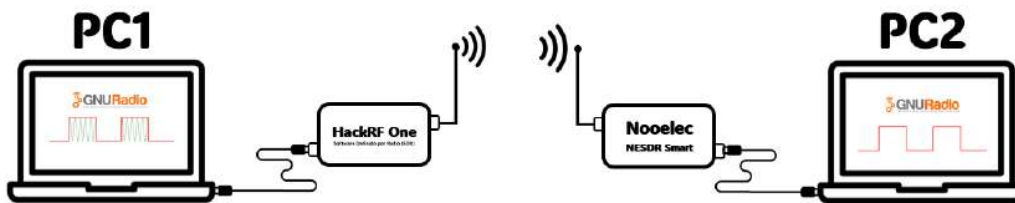
3. Implementación de demodulador FSK en GNU Radio

- Parámetros de diseño:
 - Frecuencia de transmisión: 50 MHz
 - Sample rate: A consideración
 - Valores de demodulación en base a los establecidos en la modulación

4. Esquema guía



5. Transmisión y recepción de la señal modulada en FSK



6. Realice un informe escrito que incluya el desarrollo de la actividad, análisis de resultado y conclusiones.

RUBRICAS O PARÁMETROS DE EVALUACIÓN

1. Desarrollo de actividad (60%)

- Cálculos teóricos
- Implementación haciendo uso de los dispositivos SDR

2. Informe escrito (20%)

- Resumen del laboratorio
- Objetivos
- Introducción
- Marco teórico
- Procedimiento, materiales y equipos
- Resultados, interpretación, análisis y discusión de resultados
- Conclusiones
- Bibliografía

3. Sustentación (20%)

- Participación grupal
- Preguntas libres

4. Puntuación


- 0 puntos (no cumple)
- 1 a 5 puntos según el grado de cumplimiento

ENTREGABLES

1. Esquema de modulador implementado con dispositivo SDR
2. Esquema de demodulador implementado con dispositivo SDR
3. Informe escrito

REFERENCIAS BIBLIOGRÁFICAS

- [1] https://wiki.gnuradio.org/index.php/Signal_Source
- [2] https://wiki.gnuradio.org/index.php/Multiply_Const
- [3] <https://wiki.gnuradio.org/index.php/Repeat>
- [4] <https://wiki.gnuradio.org/index.php/Multiply>
- [5] https://wiki.gnuradio.org/index.php/Constant_Source
- [6] https://wiki.gnuradio.org/index.php/Low_Pass_Filter
- [7] https://wiki.gnuradio.org/index.php/QT_GUI_Time_Sink
- [8] <https://wiki.gnuradio.org/index.php/Rail>
- [9] https://wiki.gnuradio.org/index.php/Audio_Sink

	PROGRAMA DE INGENIERÍA ELECTRÓNICA GUÍA DE LABORATORIO	VERSIÓN: 1
		PAG: 1 de #

PRACTICA No. 5

Nombre de la asignatura	
Código de la asignatura	
Título de la practica	
Docente	

CONTEXTUALIZACIÓN

1. Introducción a la modulación por desplazamiento de fase binaria (BPSK)

La modulación por desplazamiento de fase binaria (BPSK) es una técnica que utiliza dos fases distintas de una portadora para representar los datos binarios 0 y 1. Es ampliamente utilizada en sistemas de comunicación digitales debido a su simplicidad y robustez en entornos ruidosos.

2. Principios de la modulación

- **Señal portadora:** señal sinusoidal de alta frecuencia encargada de transportar los datos digitales.
- **Señal moduladora:** señal digital que varía entre dos niveles, representando los bits mediante dos valores '0' y '1'.

3. Modulador y demodulador

- **Modulación:** el esquema utiliza una señal moduladora binaria donde el bit 1 corresponde a un desfase de 0° y el bit 0 a 180° . Para representar los 0, la señal binaria se transforma a -1, lo que permite que al multiplicarla por la portadora se genere el desfase correspondiente, al combinar estos desfases, se obtiene la señal modulada en BPSK.
- **Demodulador:** proceso inverso donde se lleva a cabo la recuperación de la señal digital original a partir de la señal modulada, en primer lugar, se recupera la fase de la señal moduladora para luego aplicar el método de detección por envolvente.

OBJETIVO GENERAL

Implementar, transmitir, recibir y analizar señales moduladas en BPSK utilizando dispositivos SDR (HackRF One y Nooelec NESDR V5) para comprender el proceso y características de esta técnica de modulación.

OBJETIVOS ESPECÍFICOS

- Configurar los dispositivos SDR para transmitir y recibir señales BPSK.
- Implementar un sistema de transmisión y recepción BPSK en GNU Radio, ajustando frecuencia de portadora y moduladora, y así mismo, el índice de modulación.
- Visualizar señales transmitidas y recibidas, evaluando la fidelidad de la señal demodulada.

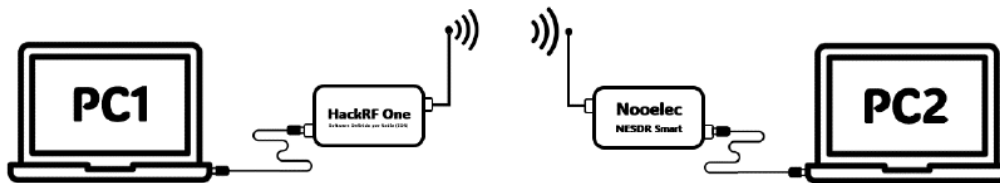
EQUIPOS Y MATERIALES

- 1 SDR transmisor (HackRF One)
- 1 SDR receptor (Nooelec NESDR V5)
- 2 antenas telescópicas
- 2 computadoras con Windows
- Aplicativo GNU Radio
- Drivers y cables para los dispositivos SDR

PROCEDIMIENTO

1. Preparación del entorno

- **Verificar la instalación de GNU Radio y los drivers:** asegurarse que GNU Radio este correctamente instalado y el dispositivo SDR (HackRF One y Nooelec NESDR Smart V5) sea reconocido de forma correcta.
- **Conexión del hardware SDR**



2. Implementación de modulador BPSK en GNU Radio

- Parámetros de diseño:
 - Frecuencia moduladora: 33.3 KHz
 - Frecuencia de portadora: 100 KHz
 - Frecuencia de transmisión: 50 MHz
 - Sample rate: 1 M
- Esquema guía

- Procedimiento, materiales y equipos
- Resultados, interpretación, análisis y discusión de resultados
- Conclusiones
- Bibliografía

3. Sustentación (20%)

- Participación grupal
- Preguntas libres

4. Puntuación

- 0 puntos (no cumple)
- 1 a 5 puntos según el grado de cumplimiento

ENTREGABLES

1. Esquema de modulador implementado con dispositivo SDR
2. Esquema de demodulador implementado con dispositivo SDR
3. Informe escrito

REFERENCIAS BIBLIOGRÁFICAS

- [1] https://wiki.gnuradio.org/index.php/Signal_Source
 [2] https://wiki.gnuradio.org/index.php/Multiply_Const
 [3] <https://wiki.gnuradio.org/index.php/Repeat>
 [4] <https://wiki.gnuradio.org/index.php/Multiply>
 [5] https://wiki.gnuradio.org/index.php/Constant_Source
 [6] https://wiki.gnuradio.org/index.php/Low_Pass_Filter
 [7] https://wiki.gnuradio.org/index.php/QT_GUI_Time_Sink
 [8] <https://wiki.gnuradio.org/index.php/Rail>
 [9] https://wiki.gnuradio.org/index.php/Costas_Loop

	UNIVERSIDAD POPULAR DEL CESAR	CÓDIGO:
		VERSIÓN: 1



PRACTICA No. 6

Nombre de la asignatura	
Código de la asignatura	
Título de la practica	
Docente	

CONTEXTUALIZACIÓN

1. Introducción a la modulación de amplitud en cuadratura (QAM)

La modulación por amplitud en cuadratura resulta ser una técnica de modulación que mezcla tanto la modulación en amplitud como en fase, lo que trae consigo la posibilidad de transmitir mayor cantidad de datos por cada símbolo, a comparación con otras técnicas de modulación. No obstante, en la práctica en cuestión, nos enfocamos en la implementación de 4QAM, una variante de QAM que cuenta con 4 posibles estados de amplitud y fase para la representación de los tres bits por símbolo.

2. Principios de la modulación

- **Señal portadora:** señal sinusoidal de alta frecuencia encargada de transportar los datos digitales.
- **Señal moduladora:** señal digital que varía entre dos niveles, representando los bits mediante dos valores '0' y '1'.

3. Modulador y demodulador

- **Modulador:** La señal moduladora digital es mapeada a puntos en la constelación 4QAM, modulando las señales del canal I en fase, y las del canal Q en cuadratura, para así generar la señal modulada en 4QAM.
- **Demodulador:** proceso inverso donde se lleva a cabo la recuperación de la señal digital original a partir de la señal modulada, detectando las combinaciones de amplitud y fase mediante un detector coherente.

OBJETIVO GENERAL

Implementar, transmitir, recibir y analizar señales moduladas en 4QAM utilizando dispositivos SDR (HackRF One y Nooelec NESDR V5) para comprender el proceso y características de esta técnica de modulación.

OBJETIVOS ESPECÍFICOS

- Configurar los dispositivos SDR para transmitir y recibir señales QAM.
- Implementar un sistema de transmisión y recepción QAM en GNU Radio, ajustando frecuencia de portadora y moduladora, y así mismo, el índice de modulación.
- Visualizar señales transmitidas y recibidas, evaluando la fidelidad de la señal demodulada.

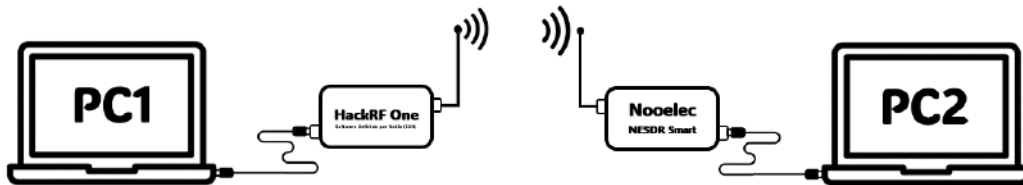
EQUIPOS Y MATERIALES

- 1 SDR transmisor (HackRF One)
- 1 SDR receptor (Nooelec NESDR V5)
- 2 antenas telescópicas
- 2 computadoras con Windows
- Aplicativo GNU Radio
- Drivers y cables para los dispositivos SDR

PROCEDIMIENTO

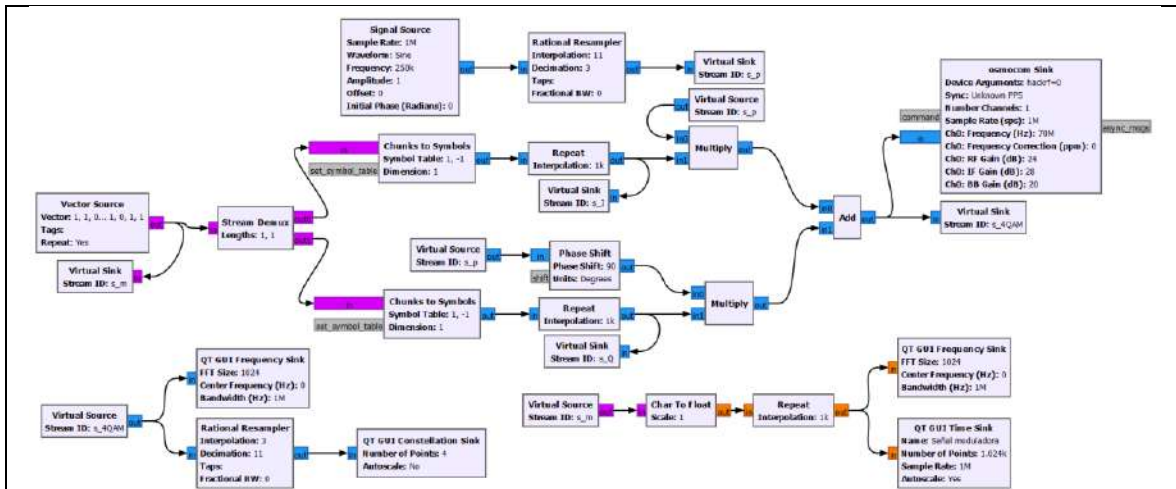
1. Preparación del entorno

- **Verificar la instalación de GNU Radio y los drivers:** asegurarse que GNU Radio este correctamente instalado y el dispositivo SDR (HackRF One y Nooelec NESDR Smart V5) sea reconocido de forma correcta.
- **Conexión de los dispositivos SDR**



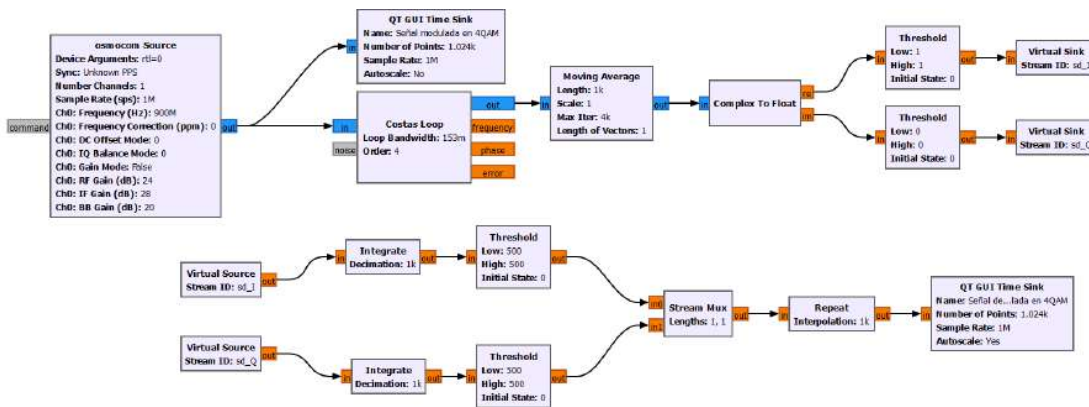
2. Implementación de modulador 4QAM en GNU Radio

- Parámetros de diseño:
 - Frecuencia de portadora: 250 KHz
 - Frecuencia de moduladora: 5 KHz
 - Frecuencia de transmisión: 900 MHz
 - Sample rate: 1 M
- Esquema guía

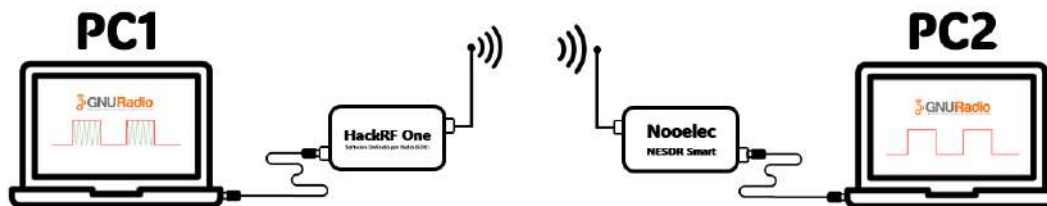


3. Implementación de demodulador 4QAM en GNU Radio

- Parámetros de diseño:
 - Frecuencia de transmisión: 900 MHz
 - Sample rate: A consideración
 - Valores de demodulación en base a los establecidos en la modulación
- Esquema guía



4. Transmisión y recepción de la señal modulada en 4QAM



- 5. Realice un informe escrito que incluya el desarrollo de la actividad, análisis de resultado y conclusiones.

RUBRICAS O PARÁMETROS DE EVALUACIÓN

1. Desarrollo de actividad (60%)

- Cálculos teóricos
- Implementación haciendo uso de los dispositivos SDR

2. Informe escrito (20%)

- Resumen del laboratorio
- Objetivos
- Introducción
- Marco teórico
- Procedimiento, materiales y equipos
- Resultados, interpretación, análisis y discusión de resultados
- Conclusiones
- Bibliografía

3. Sustentación (20%)

- Participación grupal
- Preguntas libres

4. Puntuación

- 0 puntos (no cumple)
- 1 a 5 puntos según el grado de cumplimiento

ENTREGABLES

1. Esquema de modulador implementado con dispositivo SDR
2. Esquema de demodulador implementado con dispositivo SDR
3. Informe escrito

REFERENCIAS BIBLIOGRÁFICAS

- [1] https://wiki.gnuradio.org/index.php/Signal_Source
- [2] https://wiki.gnuradio.org/index.php/Multiply_Const
- [3] <https://wiki.gnuradio.org/index.php/Repeat>
- [4] <https://wiki.gnuradio.org/index.php/Multiply>
- [5] https://wiki.gnuradio.org/index.php/Constant_Source
- [6] https://wiki.gnuradio.org/index.php/Low_Pass_Filter
- [7] https://wiki.gnuradio.org/index.php/QT_GUI_Time_Sink
- [8] <https://wiki.gnuradio.org/index.php/Rail>
- [9] https://wiki.gnuradio.org/index.php/Audio_Sink
- [10] <https://wiki.gnuradio.org/index.php/Subtract>
- [11] <https://wiki.gnuradio.org/index.php/Add>
- [12] https://wiki.gnuradio.org/index.php/Costas_Loop
- [13] https://wiki.gnuradio.org/index.php/Binary_Slicer

- [14] https://wiki.gnuradio.org/index.php/Char_To_Float
- [15] <https://wiki.gnuradio.org/index.php/Integrate>
- [16] https://wiki.gnuradio.org/index.php/Vector_Source
- [17] https://wiki.gnuradio.org/index.php/Vector_to_Stream
- [18] https://wiki.gnuradio.org/index.php/Streams_and_Vectors
- [19] <https://wiki.gnuradio.org/index.php/Threshold>

11. REFERENCIAS BIBLIOGRÁFICAS

- [1] J. R. S. Lemos y D. A. N. Suárez, «Sistema de Comunicación SDR para modulaciones PSK y QAM», 2018. [En línea]. Disponible en: <https://red.uao.edu.co/bitstream/10614/10601/5/T08268.pdf>
- [2] W. E. D. Moreno, A. P. F. Mora, y W. T. Medina, «Inclusión de herramientas SDR en las asignaturas de telecomunicaciones para la universidad ECCI», Repositorio Digital ECCI, ene. 2016, [En línea]. Disponible en: <https://repositorio.ecci.edu.co/handle/001/2194>
- [3] C. L. R. Ernesto, «APLICACIÓN DE UN SISTEMA SDR (RADIO DEFINIDA POR SOFTWARE) PARA PRÁCTICAS MULTIDISCIPLINARIAS EN LA CARRERA DE TELECOMUNICACIONES DE LA UNIVERSIDAD ISRAEL», 2019. <https://repositorio.uisrael.edu.ec/handle/47000/2067>
- [4] B. G. J. Antonio, «Implementación de software Defined Radio en sistemas de comunicaciones actuales», idUS - Depósito de Investigación Universidad de Sevilla, 2021. <https://idus.us.es/handle/11441/109235>
- [5] “Radio definida por software Tamaño del mercado, participación (2022 - 27) | Informe de la industria”. Market Research Company - Mordor Intelligence™. Accedido el 13 de septiembre de 2023. [En línea]. Disponible: <https://www.mordorintelligence.com/es/industry-reports/software-defined-radio-market>.
- [6] MathWorks. “Introducción a Radio definida por software (SDR)”. MathWorks - Creadores de MATLAB y Simulink - MATLAB y Simulink - MATLAB & Simulink. Accedido el 15 de septiembre de 2023. [En línea]. Disponible: <https://la.mathworks.com/discovery/sdr.html>
- [7] J. d. J. Rugeles Uribe, J. C. Martínez Quintero y L. E. Quibano Alarcón, “Multiple frequency continuous wave SDR radar implementation using GNU Radio”, *Ing. Desarro.*, vol. 38, n.º 1, pp. 224–242, enero de 2021. Accedido el 15 de septiembre de 2023. [En línea]. Disponible: <https://doi.org/10.14482/inde.38.1.621.38>
- [8] [1] W. Tomasi, *Sistemas de comunicaciones electrónicas*, 4ta ed. México: Pearson Educación, 2003.
- [9] «Decreto 4169 de 2011 - Gestor normativo», *Función Pública*. <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=44676#:~:text=Asignar%20y%20gestionar%20el%20espectro,est%C3%A9n%20asignadas%20a%20otras%20entidades>.

[10] “Capítulo 1 Introducción | Redes de Computadoras”. Jos Incera. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://jincera.com/NotasRC/capintro.html>

[11] J. Luque Rodríguez y S. Clavijo Suero, Modulación de señales digitales. Universidad de Sevilla, Departamento de Tecnología Electrónica, 1995.

[12] M. E. Diago y W. A. Gomez, «Visualización de diagramas en el tiempo para sistemas de comunicación digitales basados en hardware reconfigurable», *Ciencia y tecnología*, vol. 1, n.º 14, nov. 2014, doi: 10.18682/cyt.v1i14.183.

[13] DANIZIO, P. E. Sistemas de comunicaciones. ed. Córdoba: Jorge Sarmiento Editor - Universitas, 2019. 153 p. Disponible en: <https://elibro.net/es/ereader/biblioupc/175136?page=1>. Consultado en: 05 Nov 2023

[14] «Sistemas de radio Comunicación definida por software | UCSP», *Escuela de Postgrado de la Universidad Católica San Pablo*. <https://postgrado.ucsp.edu.pe/cursos/sistemas-de-radio-comunicacion-definida-por-software/>

[15] Á. Gutiérrez Rivera, “Implementación de Software Defined Radio en sistemas de comunicaciones actuales,” Universidad de Sevilla, 2017.

[16] «Libro “Laboratorio de Comunicaciones Digitales Radio definida por software” – Juan José Murillo-Fuentes’ Web Page». <https://personal.us.es/murillo/librosdr/>

[17] «Elección del dispositivo SDR», *SDR en español*, 5 de febrero de 2023. <https://sdr-es.com/eleccion-sdr/#RTL-SDR-Blog-V3>

[18] «NOOELEC - NOOELEC NESDR SMART V5 SDR - HF/VHF/UHF (100KHz-1.75GHz) RTL-SDR. RTL2832U & R820T2-Based Software Defined Radio - SDR Receivers - Software Defined Radio». <https://www.nooelec.com/store/sdr/sdr-receivers/nesdr-smart-sdr.html>

[19] Lu8mil, «TUTORIAL HackRF ONE en español - TODO-SDR.com», *TODO-SDR.com*, 29 de mayo de 2023. <https://www.todo-sdr.com/hackrf-one/>

[20] R. J. Rico, «SDR y GNU Radio como plataforma para un laboratorio de comunicaciones digitales», 2019. <https://www.redalyc.org/journal/5122/512261374007/html/>

[21] Unantenistaeneltejado, «“Hola Mundo” en GNU Radio», *UN ANTENISTA EN EL TEJADO*, 25 de junio de 2022. <https://unantenistaeneltejado.com/2022/06/25/hola-mundo-en-gnu-radio/>

[22] «Pruebas en hardware», MATLAB & Simulink. <https://es.mathworks.com/solutions/wireless-communications/hardware-testing.html#:~:text=MATLAB%20y%20Simulink%20ayudan%20a,dise%C3%B1o%20continuamente%20con%20pruebas%20inal%C3%A1mblicas>

[23] “GitHub - ryanvolz/radioconda: Software radio distribution and installer for conda”. GitHub. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://github.com/ryanvolz/radioconda>

[24] “About GNU Radio”. GNU Radio. Accedido el 2 de febrero de 2025. [En línea]. Disponible: <https://www.gnuradio.org/about/>

[25] Velasco Ruiz, Johanna Lizeth, Cañizares Carvajal y Stalyn Andrés, “Implementación de un transmisor de radio FM utilizando un dispositivo SDR (Radio Definido por Software) y guía técnica para prácticas de laboratorio de telecomunicaciones en la Universidad de las Fuerzas Armadas ESPE sede Latacunga”, Trabajo de grado, Univ. Fuerzas Armadas ESPE, Sangolquí, 2021. Accedido el 7 de septiembre de 2023. [En línea]. Disponible: <http://repositorio.espe.edu.ec/bitstream/21000/25896/1/M-EPEL-SIT-0119.pdf>

[26] J. L. Ordóñez, «Espectro electromagnético y espectro radioeléctrico», *Dialnet*, 2012. <https://dialnet.unirioja.es/servlet/articulo?codigo=5738106>

[27] «Ley 1341 de 2009 - Gestor Normativo», *Función Pública*. <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=36913>

[28] E. P. C. Andrés, «DISEÑO e IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE COMUNICACIONES PARA UN SATÉLITE TIPO CUBESAT», 11 de noviembre de 2022. <https://noesis.uis.edu.co/items/0b0d0f32-f148-42c9-8b47-b3494029fd3b>

[29] H. F. D. Jhonathan, «Las técnicas de modulación digital», 20 de septiembre de 2021. <https://repositorioslatinoamericanos.uchile.cl/handle/2250/6535569>

[30] «ADALM-PLUTO Evaluation Board | Analog Devices». <https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html#eb-overview>

[31] D. J. Hueyas Florez, "Las técnicas de modulación digital," Monografía de pregrado, Universidad Nacional de Educación Enrique Guzmán y Valle, Lima, Perú, 2021.