

Recibido 08/06/2019

ReCIBE, Año 8 No. 2, Noviembre 2019

Aceptado 02/09/2019

SDR y GNU Radio como plataforma para un laboratorio de comunicaciones digitales

Rogelio Jaimes Rico¹
roger-jr97@hotmail.com

Saul Lazcano Salas¹
slazcanos@uaemex.mx

¹Centro Universitario UAEM Valle de México
Universidad Autónoma del Estado de México

Resumen: SDR (Software Defined Radio) o radio definido por software es una tecnología flexible compuesta por una parte hardware (radio frecuencia) y otra parte basada en software (procesado en banda base). El presente trabajo tiene como fin el análisis de SDR como base para la implementación de un laboratorio de comunicaciones digitales en licenciaturas vinculadas a las telecomunicaciones, tales como Ingeniería en Computación o Ingeniería en Sistemas y Comunicaciones. Para ello, se analizan algunos temas básicos como muestreo, filtrado, modulación y la factibilidad de implementación de los mismos en un esquema SDR, mediante un radio modelo HackRf One en un entorno de GNU Radio.

Palabras clave: GNU Radio, comunicaciones digitales, SDR.

Abstract: SDR (Software Defined Radio) or software-defined radio is a flexible technology composed of a hardware part (radio frequency transceiver) and other software-based part (band pass processing). The purpose of this paper is to analyze the use of SDR as the basis for the implementation of a digital communications laboratory in degrees related to telecommunications, such as Computer Engineering or Systems and Communications Engineering. In this way, some basic topics such as sampling, filtering, modulation and the feasibility of implementing them in an SDR scheme are analyzed, using a HackRF One Radio under a GNU Radio environment.

Keywords: GNU Radio, digital communications, Software Defined Radio.

1. Introducción

GNU Radio (The GNU Radio Foundation, 2019), es una herramienta de desarrollo con licencia tipo GLP (General Public License), que permite, mediante bloques de procesamiento de señales, la construcción y simulación de sistemas de comunicaciones, lo anterior puede ser a nivel simulación o a nivel implementación con la ayuda de radios programables. La tarea de los radios es trabajar a nivel radio frecuencia, mientras el procesamiento en banda base de la señal se realiza mediante el software.

Esta combinación de elementos hardware y software permiten la creación de diferentes esquemas de comunicaciones a partir de un único dispositivo hardware (radio), abriendo la puerta para que en la actualidad, los usuarios de este tipo de tecnología sea muy extensa, desde radio aficionados hasta para realizar investigaciones de frontera en diversas áreas vinculadas a las comunicaciones inalámbricas en general, tales como codificación de canal (Giard, Sarkis, Leroux, & et.al., 2018), redes programables (Macedo, Guedes, Vieira, Vieira, & Nogueira, 2015; Foukas, Nikaein, Kassem, Mahesh, & Kontovasilis, 2016), antenas (Abari, 2016), educación, por citar algunas áreas (Wyglinski, Orofino, Ettus, & Rondeau, 2016).

Por otro lado, existen diversos radios SDR con diferentes características, entre los que destaca, por su relación calidad / precio, el dispositivo HackRF One, que junto al software GNU Radio permite realizar conversiones A/D (Analógico - Digital) y D/A (Digital -Analógico), con una tasa de muestreo de hasta 20 millones de muestras por segundo, trabajar con señales en el rango de 1 MHz – 6 GHz además de capturar, analizar y transmitir frecuencias de radio (Leahy Center for Digital Investigation, 2017).

En conjunto, software (entorno GNU Radio) y hardware (equipo HackRF One), constituyen una herramienta sumamente atractiva, particularmente para fines de docencia dentro del área de sistemas de comunicaciones digitales, ya que permite la construcción y el modelado de elementos tales como conversores analógico-digital y digital analógico, filtros diversos, receptores y emisores completos, entre otros, facilitando enormemente la comprensión de dichos temas por parte de los alumnos.

2. Conceptos básicos

ADC (Convertidor Digital Analógico): transforma una señal eléctrica analógica en un número digital equivalente.

DAC (Convertidor Analógico Digital): transforma un número digital en una señal eléctrica analógica (Huircan, 2003).

Teorema de Nyquist: reconstruye una señal analógica en base a la conversión analógica digital, para que esto se cumpla la señal analógica debe ser limitada en banda y la frecuencia de muestreo debe ser igual o mayor al doble del ancho de banda de la señal, esto si se da por hecho que la señal analógica esta en banda base, este teorema se puede ver reflejado en la conversión analógica-digital y se encuentra representado en la siguiente expresión:

$$F_s \geq 2F_c \quad (1)$$

Donde:

F_s: frecuencia de muestreo (Sampling).

F_c: frecuencia de la información – ancho de banda (Carrier).

GNU Radio: se define como un entorno de desarrollo libre y abierto (The GNU Radio Foundation, 2019); los bloques de procesamiento de señales son de código abierto (C++ o Python). Se enfoca de manera directa en ser una herramienta de apoyo para el trabajo con radios SDR.

Su manejo es simple e intuitivo, esto hace que sea sumamente amigable para todo tipo de usuario, característica particularmente útil para nuevos usuarios. Adicionalmente, cuenta con una comunidad de usuarios que brinda soporte y consejos en foros especializados (FAQ-GNU Radio, 2016).

La interfaz gráfica puede ser dividida en 4 partes, como se muestra en la figura 1:



Figura 1. Interfaz gráfica en GNU Radio con todos sus componentes.

1) Barra de herramientas: espacio configurable para las funciones esenciales, como ejecutar el sistema, guardar proyecto, nuevo espacio de trabajo, copiar, pegar, retroceder (estado previo del proyecto), buscar, habilitar y deshabilitar bloques, por mencionar las más importantes y usadas.

2) Espacio de trabajo: espacio en blanco donde se pueden diseñar sistemas conformados por bloques conectados unos de otros, por defecto, al abrir un nuevo proyecto estarán dos bloques (top_block y samp_rate).

3) Biblioteca de bloques: se encuentran todos los bloques de procesamiento con los que el software está previamente conformado, cada bloque se encuentra ordenado acorde a la función que realiza, basta con acudir a un apartado y desplegar la lista de bloques que contiene.

4) Consola: ubicado en la parte inferior de la interfaz gráfica, en esta se muestra el estado de la ejecución del sistema.

HackRF One: el dispositivo de tipo SDR (Software Defined Radio) HackRF One (Great Scott Gadgets, 2016), tiene la capacidad de digitalizar señales de radio que son recibidas o transmitidas por otros dispositivos, trabaja en un rango de frecuencia desde 1MHz hasta 6GHz, lo cual permite abarcar dispositivos que operan con Bluetooth, radio FM, comunicación de campo cercano (del término en inglés NFC, Near Field Communication) y tecnologías de telefonía móvil.

3. Diseño de sistemas en GNU Radio

Como se ha mencionado, GNU Radio cuenta con una interfaz intuitiva para la creación de sistemas de radiofrecuencias, cada bloque se caracteriza por el tipo de datos de entrada y datos de salida que soporta. Los tipos de datos básicos que GNU Radio soporta son: byte, short, in, floating point y complex, los cuales se distinguen a partir de colores: morado, amarillo, verde, naranja y azul, como se muestra en la figura 2.

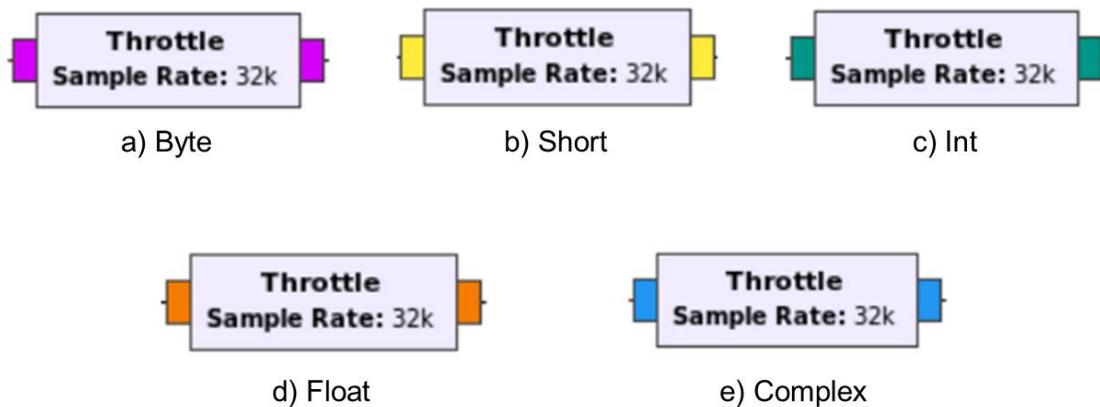


Figura 2. Tipos de datos y sus colores de identificación. Fuente: (FAQ-GNU Radio, 2016).

A continuación se muestran algunos ejemplos de sistemas realizados en GNU Radio en conjunto con el radio HackRF One, cada uno con una función específica:

Convertor analógico-digital: convierte una señal analógica en una señal digital, y es capaz de adaptarse a los dispositivos de procesos involucrados en un sistema de transmisión, se muestra en la figura 3.

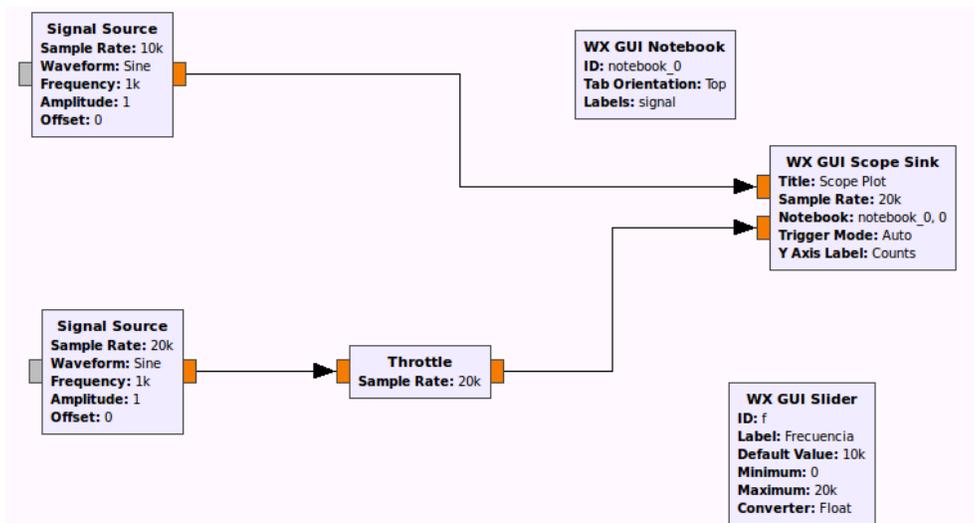


Figura 3. Esquema general de un convertor analógico – digital.

Receptor FM: este diagrama en uso conjunto al dispositivo SDR HackRF One tiene la capacidad de sintonizar estaciones de radio FM y al mismo tiempo graficar la transformada de Fourier de dicha señal a manera de analizador de espectros. Este diagrama se muestra en la figura 4.

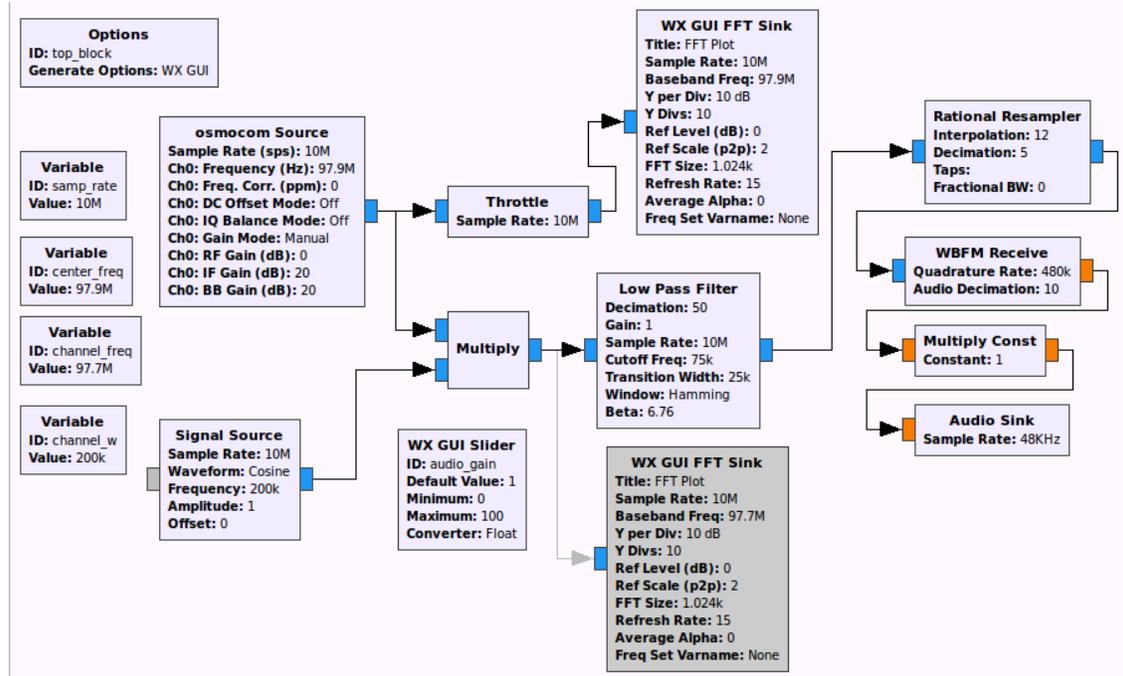


Figura 4. Sistema receptor de señales FM.

Receptor FM de 2 frecuencias: variante del sistema anterior, en donde se añade la recepción de una segunda señal de FM y al igual que el caso anterior, se grafica la transformada de Fourier de la señal a manera de analizador de espectros. El esquema de bloques se muestra en la figura 5.

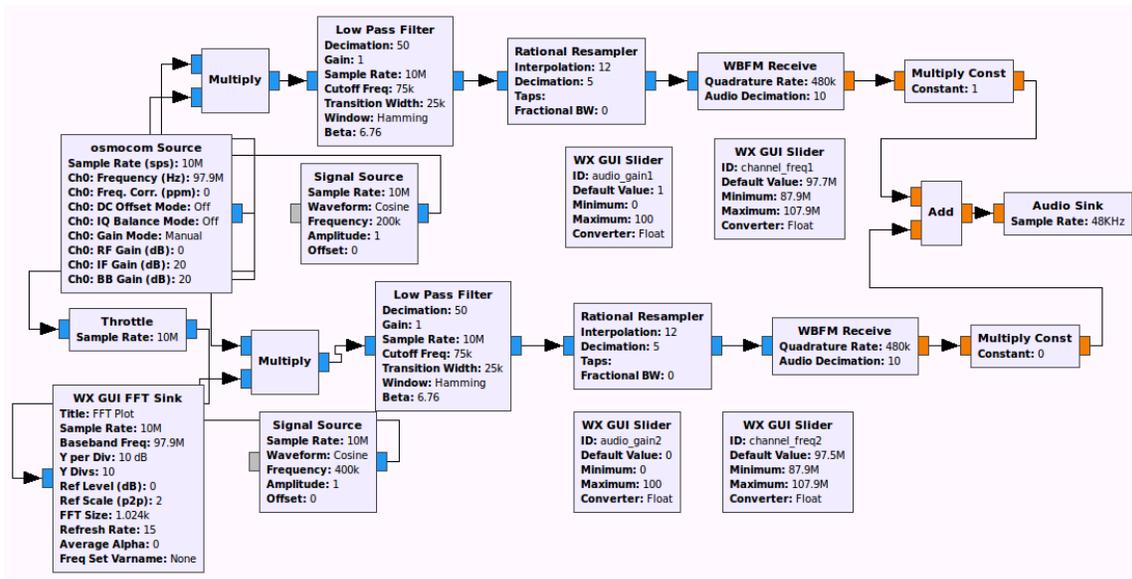


Figura 5. Sistema receptor de dos señales FM.

Filtro pasa bajo: tiene la finalidad de permitir el paso a señales con una frecuencia por debajo de un rango estipulado, por otro lado, rechaza aquellas que superan ese rango. El esquema se muestra en la figura 6.

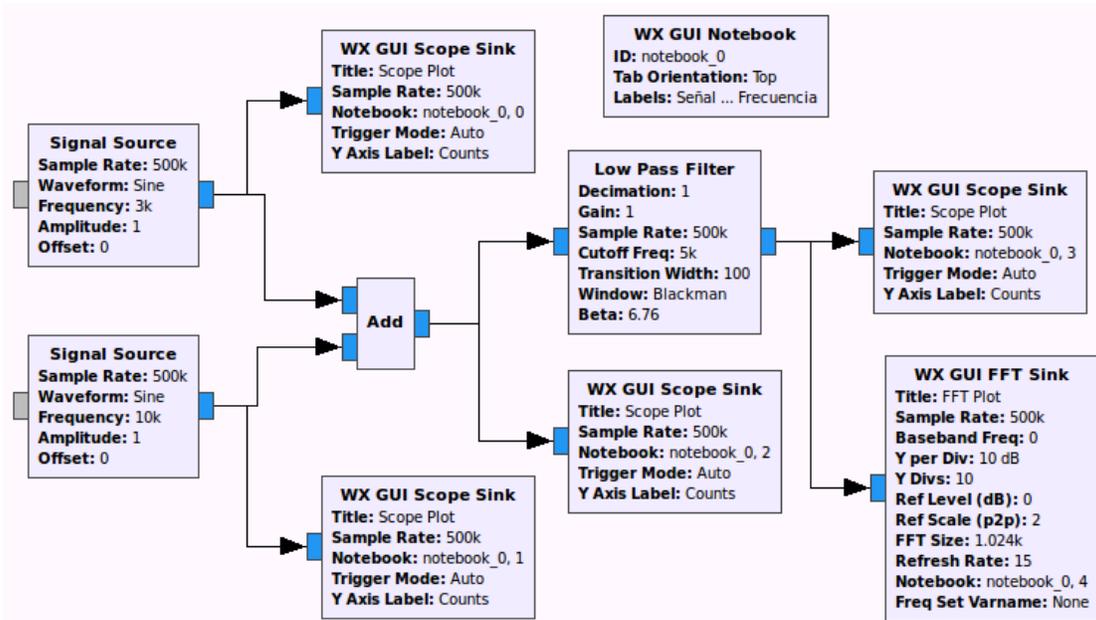


Figura 6. Diseño de un filtro pasa bajo.

La figura 7 define el comportamiento ideal de un filtro paso baja, el cual permite el paso de señales por debajo de la denominada frecuencia de corte y elimina las señales por arriba de dicha frecuencia.

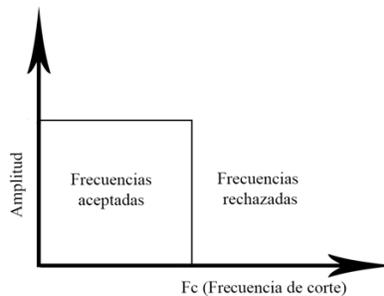


Figura 7. Representación del funcionamiento de un filtro pasa bajo ideal.

Filtro paso alto: de manera opuesta al comportamiento de un filtro paso baja, el filtro paso alto permite el paso de señales cuya frecuencia sea superior a una determinada frecuencia umbral predeterminada; aquellas señales con frecuencia inferior a la frecuencia umbral son atenuadas. El esquema general de la construcción de un filtro paso altas se muestra en la figura 8.

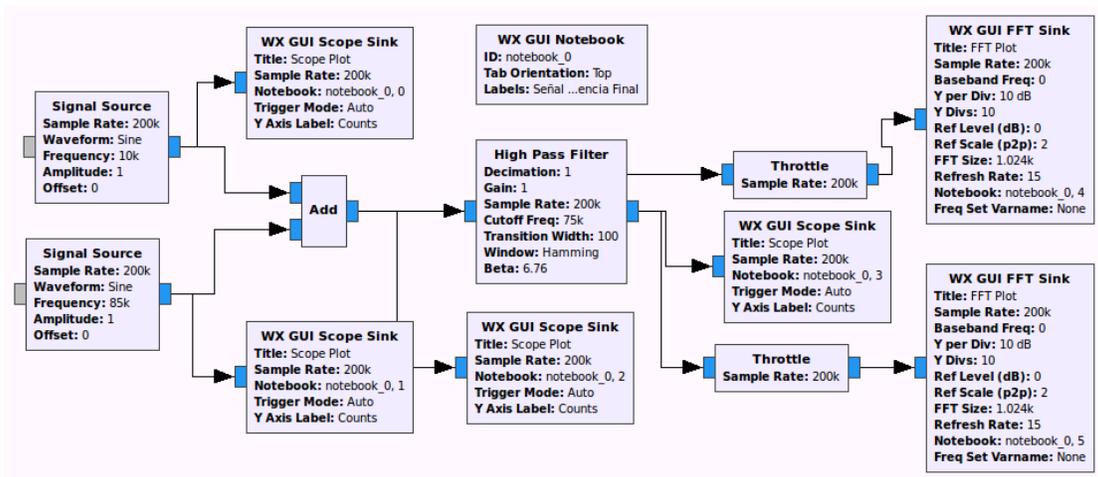


Figura 8. Diseño de un filtro pasa alto.

Como se puede apreciar en la figura 9, la forma en que actúa un filtro pasa alto ideal, dejando pasar solo aquellas frecuencias que están por encima de la frecuencia de corte.

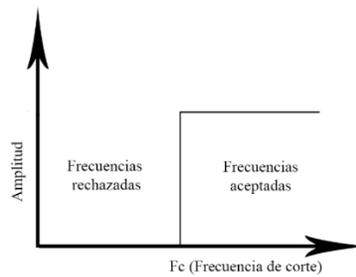


Figura 9. Representación del funcionamiento de un filtro pasa alto ideal.

Filtro pasa banda: Permite el paso únicamente a aquellas señales que se encuentren en un determinado rango, limitado por una frecuencia umbral inferior y una frecuencia de corte superior (el valor más alto que puede aceptar), el diagrama de bloques de la construcción de un filtro paso banda en GNU Radio se muestra en la figura 10.

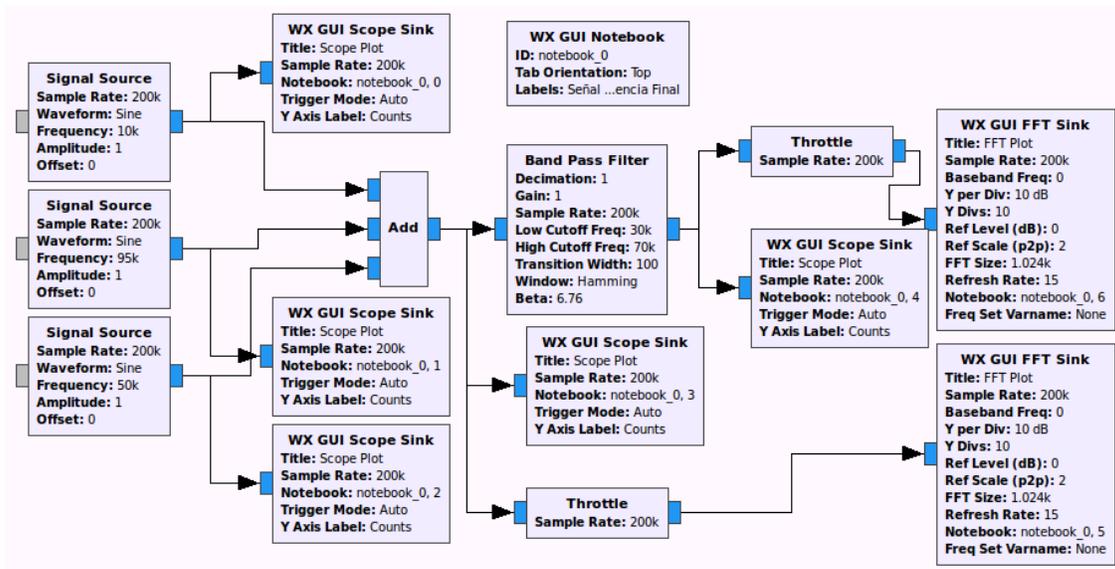


Figura 10. Diseño de un filtro pasa banda.

La figura 11 presenta el comportamiento de un filtro pasa banda ideal, dejando pasar solo aquellas frecuencias que se encuentran entre F1 y F2, rechazando todas aquellas fuera de estos rangos.

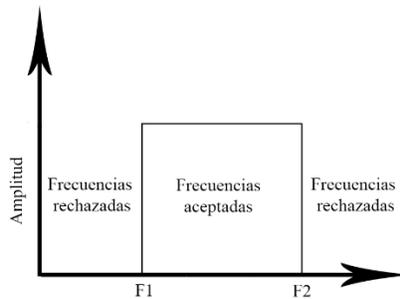


Figura 11. Representación del funcionamiento de un filtro pasa banda.

Filtro rechazada banda: de manera análoga al filtro paso banda, este filtro requiere de igual manera una frecuencia inferior y una superior, sin embargo, este filtro atenúa todas las señales comprendidas en el rango antes definido. El diagrama de bloques en GNU Radio se muestra en la figura 12.

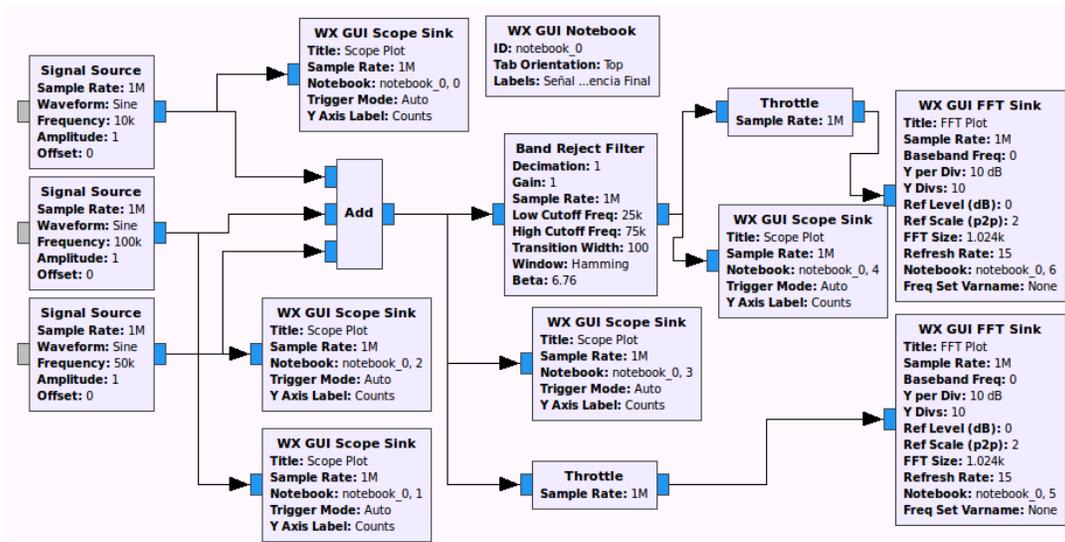


Figura 12. Diseño de un filtro rechaza banda.

Como se puede apreciar en la figura 13, la forma en que actúa un filtro rechaza banda, dejando pasar solo aquellas frecuencias por debajo de F_1 y encima de F_2 , rechazando todas aquellas fuera de estos rangos.

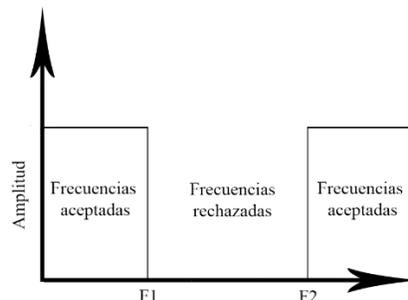


Figura 13. Representación del funcionamiento de un filtro rechaza banda ideal.

4. Conclusiones y trabajo futuro

SDR es una tecnología que presenta un área de oportunidades muy extenso tanto en docencia, aplicaciones a nivel amateur y experimental y por supuesto, en la investigación.

La combinación de radios con una buena relación calidad / precio con un entorno de desarrollo abierto y con una gran comunidad de soporte como lo es GNU Radio, da como resultado una excelente combinación que genere una gran inercia en beneficio de los alumnos, facilitando la comprensión de temas asociados a comunicaciones digitales. En este sentido, un laboratorio completo de comunicaciones digitales basado en tecnología SDR, abriría las puertas no sólo para cubrir lo relacionado a la docencia, sino permitiría generar una masa crítica en los alumnos para ir más allá y apostar por la creación de grupos de trabajo estudiantiles con un perfil muy sólido en comunicaciones inalámbricas en general.

Como trabajo a ser desarrollado, es el estudio de esquemas de codificación de canal enfocados a canales inalámbricos, particularmente códigos polares y su decodificación mediante algoritmos basados en redes neuronales.

REFERENCIAS

- Abari, O. (2016). A millimeter wave software defined radio platform with phased arrays. *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom '16)* (págs. 419-420). New York: ACM. doi:10.1145/2973750.2985258
- FAQ-GNU Radio. (1 de Junio de 2016). *FAQ-GNU Radio*. Recuperado el 10 Agosto de 2019, de FAQ-GNU Radio: <https://wiki.gnuradio.org/index.php/FAQ>
- Foukas, X., Nikaiein, N., Kassem, M., Mahesh, K., & Kontovasilis, K. (2016). FlexRAN: A Flexible and Programmable Platform for Software-Defined Radio Access Networks. *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies (CoNEXT '16)* (págs. 427-441). New York: ACM. doi:10.1145/2999572.2999599
- Giard, P., Sarkis, G., Leroux, C., & et.al. (Mayo de 2018). Low Latency Software Polar Decoders. *Journal of Signal Processing Systems*, 761-775. doi:10.1007/s11265-016-1157-y
- Great Scott Gadgets. (01 de Enero de 2016). *HackRF - Great Scott Gadgets*. Recuperado el 10 Agosto de 2019, de Great Scott Gadgets: <https://greatscottgadgets.com/hackrf/>
- Huircan, J. (1 de Enero de 2003). *Web Quidel Universidad de la Frontera*. Recuperado el Agosto de 2019, de Web Quidel: http://quidel.inele.ufro.cl/~jhuircan/PDF_CTOSII/ad03.pdf
- Leahy Center for Digital Investigation. (1 de Enero de 2017). *Champlain College*. Recuperado el 10 Agosto de 2019, de Leahy Center for Digital Investigation: https://www.champlain.edu/Documents/LCDI/HackRF%20One%20Tutorial_F2017%20-%20Report.docx.pdf
- Macedo, D. F., Guedes, D., Vieira, L., Vieira, M., & Nogueira, M. (2015). Programmable Networks, From Software-Defined Radio to software-Defined Networking. *IEEE Communications Surveys & Tutorials*, 17(2), 1102-1125. doi:10.1109/COMST.2015.2402617
- The GNU Radio Foundation. (1 de 1 de 2019). *GNU Radio - The Free & Open Source Radio Ecosystem*. Recuperado el 10 Agosto de 2019, de GNU Radio: <https://www.gnuradio.org/>
- Wygłinski, A. M., Orofino, D. P., Ettus, M. N., & Rondeau, T. W. (Enero de 2016). Revolutionizing software defined radio: case studies in hardware, software, and education. *IEEE Communications magazine*, 54(1), 68-75. doi:10.1109/MCOM.2016.7378428

Notas biográficas



Rogelio Jaimes Rico estudiante de noveno semestre de la carrera de Ingeniería en Sistemas y Comunicaciones del Centro Universitario UAEM Valle de México de la Universidad Autónoma del Estado de México, México. Su línea de acentuación es telecomunicaciones y está involucrado en trabajos de investigación relacionados a SDR (Software Defined Radio).



Saul Lazcano Salas es Profesor de Tiempo Completo del Centro Universitario UAEM Valle de México de la Universidad Autónoma del Estado de México, México. Egresó de la licenciatura en Ingeniería en Telecomunicaciones por la Universidad Nacional Autónoma de México, México, hizo su Máster en Gestión de las Telecomunicaciones por la Escuela de Organización Industrial en España y su Doctorado en Ingeniería, área de Telecomunicaciones por la Universidad Nacional Autónoma de México, México. Actualmente, desarrolla investigación en el área de codificación de canal y áreas afines. Ha dirigido tesis de maestría y licenciatura, ha sido conferencista ponente en diversos eventos y ha publicado artículos y capítulos de libro en las áreas de Inteligencia Artificial Aplicada e Inteligencia Computacional. Cuenta con el reconocimiento de Perfil Deseable PRODEP de la SEP en México.



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.