

Sistema de captura de señales FM y reconocimiento de voz para la asistencia de personas con problemas auditivos

1st Miguel Antonio Recinos Galindo
Ingeniería en Electrónica y Telecomunicaciones
Universidad Rafael Landívar
Ciudad de Guatemala, Guatemala
miguelantonio172@gmail.com

2nd Luis Javier Ortíz Abadía
Ingeniería en Electrónica y Telecomunicaciones
Universidad Rafael Landívar
Ciudad de Guatemala, Guatemala
ljortizab@gmail.com

Abstract—En el presente documento se describe un proyecto modelo de un sistema de capturas de señales FM y reconocimiento de voz para la asistencia de personas con problemas auditivos, el objetivo principal es que los usuarios con problemas auditivos puedan sintonizar emisoras de radio de manera visual.

Index Terms—SDR, LimeSDR, GNU Radio, UDP

I. INTRODUCCIÓN

El proyecto consiste en poder implementar y desarrollar una radio FM para personas con distintos problemas auditivos, ya que la discapacidad auditiva es una limitación para muchas personas en el mundo y es un impedimento para que gocen de contenidos multimedia como, por ejemplo, la radio. El objetivo del proyecto descrito en el presente documento es proporcionar un modelo e instrucciones para la implementación de la radio para personas con problemas auditivos, cual debe de capturar y poder recibir las señales FM, donde estas señales son procesadas para poder detectar a los hablantes y, de esta manera almacenar el contenido de la señal, para posteriormente ser visualizada en una página web en tiempo real. Este proyecto brinda fácil acceso a emisoras de radio para que las personas con problemas auditivos puedan gozar del contenido de la radio de una manera eficaz y amigable para el usuario.

II. RADIO DEFINIDA POR SOFTWARE

La radio definida por Software “SDR” (Radio definida por Software, por sus siglas en inglés) es un tipo de sistema que utiliza hardware de radiofrecuencia junto con software para recibir y transmitir señales de radio. El interés por esta tecnología ha llevado a que esta se siga consolidando con diferentes aplicaciones a lo largo de los años y es una alternativa actualizada respecto a la radio tradicional. Un SDR es una herramienta versátil que se puede utilizar para realizar una amplia variedad de tareas relacionadas con las comunicaciones de radio. Estas pueden ser algunas de las tareas comunes que se pueden realizar con un SDR:

- Escuchar y decodificar señales de radio: SDR permite captar, escuchar y decodificar diferentes señales de radio con base a diferentes frecuencias, puede realizar distintas

modulaciones como señales radio FM, AM, SSB, entre otras.

- Transmitir señales de radio: un SDR puede transmitir señales de radio a diferentes frecuencias, de igual forma que la decodificación, también puede realizar distintas modulaciones con base a las señales de radio mencionadas anteriormente.
- Radio cognitiva: Un SDR puede implementar técnicas de radio cognitiva que básicamente es una tecnología de comunicaciones inalámbricas que permite a los dispositivos de radio utilizar el espectro de radio de manera más eficiente y adaptativa, lo que permite un uso más eficiente del espectro de radio y una mejor calidad de las comunicaciones de radio.
- Comunicaciones de emergencia: Un SDR se puede utilizar en situaciones de emergencia para comunicarse con otros dispositivos SDR, puede establecer una comunicación segura y confidencial en caso de que las redes de comunicaciones normales no establezcan la comunicación o fallen.
- Investigación y desarrollo: Un SDR se puede utilizar para investigar y desarrollar nuevos sistemas de comunicaciones de radio, lo que permite experimentar con diferentes configuraciones de hardware y software. Esto permite el avance y evolución de nuevas tecnologías y comunicaciones.

Cabe destacar que para la implementación de nuevos SDR han existido diferentes métodos y técnicas de implementación de SDR (como lo pueden ser el uso de FPGA como implementación de hardware dedicado, y la programación de alto nivel) para poder mejorar la eficiencia y el rendimiento de los sistemas de radio. Existen algunos trabajos sobre el desarrollo de estándares como IEEE 802.22 y protocolos como por ejemplo el protocolo de acceso al medio cognitivo MAC (Control de Acceso al Medio, por sus siglas en inglés), que sirven para mejorar el desarrollo de los estándares y protocolos para la comunicación inalámbrica SDR.

III. METODOLOGÍA DE DESARROLLO

A. Requerimientos para el Hardware

a) *LimeSDR*: es un dispositivo de radio definido por software (SDR) de código abierto que permite la comunicación inalámbrica. LimeSDR es capaz de soportar una variedad de protocolos de radio, como lo pueden ser Bluetooth, Zigbee, LTE, entre otros.

b) *Requisitos mínimos de computadora*: es recomendable utilizar una computadora con los siguientes requisitos mínimos: sistema operativo W10, memoria RAM de 8GB, 4 CPUS y 2 núcleos.

B. Requerimientos para el Software

a) *GNU Radio*: es un software de código abierto que proporciona un entorno de desarrollo para la creación de sistemas de radio definidos por software (SDR). Los sistemas de comunicación se crean a partir de flujos en diagramas de bloques, dichos bloques realizan tareas como la modulación, demodulación, decodificación, filtrado y análisis de señales.

b) *Python*: es un lenguaje de programación interpretado, de alto nivel, que se utiliza en una amplia variedad de aplicaciones de desarrollo de software. A continuación se detallan brevemente las dependencias utilizadas de este lenguaje:

- Reconocimiento de voz: Los modelos de reconocimiento de voz utilizan técnicas de procesamiento de señales y aprendizaje automático para convertir las señales de voz en texto escrito.
- TCP Sockets: son una forma de comunicación entre procesos de software a través de una red de computadoras utilizando el protocolo de comunicación TCP/IP o UDP. Para el proyecto se utilizó UDP debido a que UDP se adecúa para las aplicaciones en tiempo real.
- Threading: es una técnica de programación que permite que un programa realice múltiples tareas al mismo tiempo, conocidas como hilos. Los hilos son una forma de ejecutar varias partes de un programa lo que puede mejorar significativamente la velocidad y la capacidad de respuesta de la aplicación.
- HTML: es el lenguaje utilizado para crear páginas web y aplicaciones web. Junto con CSS y JavaScript, HTML forma la base de la mayoría de las páginas web y aplicaciones web modernas.
- Javascript: Es un lenguaje de programación de alto nivel, interpretado y orientado a objetos que se ejecuta en el navegador del usuario o en el servidor.

IV. IMPLEMENTACIÓN DEL SISTEMA

La implementación y el desarrollo de una radio FM para personas con problemas auditivos constó de tres distintas fases; la primera fue la captura y recepción de señales FM por medio de un diagrama de bloques en la aplicación GNU Radio, en la segunda fase se implementó un script en el lenguaje Python para poder captar y recibir la señal enviada desde el GNU Radio por el protocolo UDP y ser transcrita a un archivo .txt por medio de un reconocimiento de voz. Para la tercera fase se

implementó un script en lenguaje HTML el cual fue utilizado para crear y diseñar la página web y esta hace uso de un script en el lenguaje de Javascript el cual se encuentra en la parte inferior del código HTML, cuya función principal es la de ubicar el archivo de texto, y poderlo cargar a la web por una solicitud http y finalmente, poder mostrar el contenido escrito en tiempo real de la señal FM recibida.

Primera fase: captura y recepción de señales FM. Se procedió a elaborar un nuevo diagrama de bloques (archivo .grc) dentro de la aplicación GNU Radio en donde se modela el proceso para poder recibir y demodular una señal FM, para empezar se utiliza el bloque de recepción de señales en donde se utiliza la tarjeta LimeSDR que es la fuente para la recepción de señales mediante una antena más potente a la que originalmente trae, el segundo bloque es un filtro pasa bajas, los siguientes bloques en paralelo son; el primer bloque de sincronización de frecuencia, y el segundo bloque es un remuestreador racional. Para seguir con la serie, luego del remuestreador racional continúa el bloque WBFM para FM (discriminador de frecuencia). El siguiente bloque fue un filtro pasa banda. En este punto el flujo se bifurca en dos caminos, para el primer camino se procedió a utilizar un bloque que sirve para convertir una señal de entrada en formato de números de punto flotante en una señal compleja de formato de número complejos. Nuevamente el siguiente bloque es un multiplicador de constante y, por último, se utiliza el bloque de sincronización UDP. Para el segundo camino bifurcado se utiliza un multiplicador de constante, y luego el bloque de un sincronizador de archivos .wav.

Segunda fase: se implementó un script en el lenguaje Python para poder captar y recibir la señal enviada desde el GNU Radio por el protocolo UDP a través una dirección ip y un puerto específicos (127.0.0.1 y 5005, respectivamente), en este script la señal es recibida como archivo .wav y por medio de una serie de líneas de código la señal proveniente es sometida a un reconocimiento de voz para poder detectar a los hablantes y, de esta manera poder crear un nuevo archivo .txt en donde se vaya almacenando el texto recibido en la señal de audio.

Tercera fase: se implementó un script en lenguaje HTML el cual fue utilizado para crear y diseñar la página web, HTML es el lenguaje principal para la creación de contenido en la web, se utiliza para definir la estructura y el contenido de la página web. Se utilizó HTML debido a que es la base de muchas tecnologías web como lo es Javascript, que a través del uso de diferentes elementos y atributos es posible crear páginas web interactivas y dinámicas para el usuario. De igual forma, se implementó el lenguaje CSS para dar estilo y diseño a la página web HTML. Un script de Javascript fue utilizado en la implementación del proyecto; el script fue creado en la parte inferior del script HTML para poder ser utilizado dentro de la página web, y su función principal es la de ubicar el archivo de texto en la carpeta donde se iba almacenar luego de haber sido procesada por el script de Python y, de esta manera poderlo cargar a la web por una solicitud http y poder mostrar el contenido escrito en tiempo real de la señal FM

recibida previamente.

V. RESULTADOS

Para poder corroborar la implementación del sistema, se procede a contruir el sistema físico, básicamente conectar el dispositivo limeSDR al ordenador con los aplicativos correspondientes.



Fig. 1. Montaje de equipo.

La señal de FM recibida mediante la limeSDR y el diagrama de bloques para la correcta demodulación. Como último paso



Fig. 2. Diagrama de bloques GNU Radio.

se realizar la implementación del sistema de reconocimiento de voz para poder detectar a los hablantes en la señal FM y se procede a mostrar el texto recibido en la señal en la página web en tiempo real.



Fig. 3. Texto de señal FM.

VI. CONCLUSIONES Y RECOMENDACIONES

Se concluye que se puede diseñar e implementar una radio para personas con problemas auditivos mediante Radio Definida por Software SDR de forma eficiente y de interfaz sencilla para el usuario.

Se recomienda que la recepción de la señal FM se realice en un lugar abiertos y con pocos dispositivos tecnológicos puesto que estos crean interferencia al momento de recibir señales en el dispositivo limeSDR.

Se recomienda que la toma de recepción de la señal sea de manera rápida debido a que la tarjeta se calienta y presenta problemas para la recepción de señales al elevarse la temperatura del circuito integrado que combina las funciones de transmisor y receptor llamada transceptor.

REFERENCES

- [1] Collins, F., Getz, R., Pu, D., Wyglinski, M. (2018). Software-Defined Radio for Engineers. Mobile Communications.
- [2] Stewart, B., Barlee, K., Atkinson, D., Crockett, L., Broadhurst, A. (2022). Software Defined Radio using Matlab and Simulink RTL-SDR (2.a ed.). University of Strathclyde.
- [3] Pérez-Díaz, A., Vargas-Rosales, C., Gómez-García, R. (2020). A comparative study of SDR devices for wireless communication systems. IEEE Access, 8, 214703-214717. Recuperado de <https://ieeexplore.ieee.org/document/9269149>
- [4] Spiceworks. (s.f.). TCP vs UDP: What's the Difference? Recuperado de <https://www.spiceworks.com/tech/networking/articles/tcp-vs-udp/>
- [5] Salim, H., Nizam, M. B., Ismail, M. (2020). Performance analysis of UDP and TCP protocols for transmitting encrypted data. Journal of Physics: Conference Series, 1529(4), 042038. Recuperado de <https://iopscience.iop.org/article/10.1088/1742-6596/1529/4/042038>