

LABORATORIO EXPERIMENTAL OPERADO CON EL APOYO DE UN ASISTENTE VIRTUAL PARA LA ENSEÑANZA DE TEORÍA DE SEÑALES IMPLEMENTANDO TECNOLOGÍAS DE CÓDIGO ABIERTO

Resumen — *Las dificultades en la enseñanza y el aprendizaje hacen parte del entorno educativo actual y la educación de la asignatura de Teoría de Señales y Sistemas no es ajena de este aspecto. En los cursos dedicados a la enseñanza de esta área suelen darse mayor relevancia a las actividades de tipo teórico y las horas prácticas asignadas están enmarcadas en un concepto completamente de simulación, es decir, no existe una aplicación práctica de los contenidos del currículo y por tanto éstos se enseñan fuera del contexto en el que tienen aplicación. En consecuencia, se ocasiona que los estudiantes alcancen una visión parcial e incompleta de la asignatura Teoría de Señales y Sistemas, y de su importancia en la formación profesional del Ingeniero Electrónico; además, le pierden sentido práctico pues no encuentran relación alguna entre los contenidos desarrollados en clase con el entorno profesional y tecnológico que les rodea, es decir, no confrontan lo teórico con lo práctico. Por lo tanto, se busca verificar si un laboratorio experimental para la enseñanza de la Teoría de Señales y Sistemas logrará potencializar la motivación y el desarrollo de mejores resultados de aprendizajes en los estudiantes que cursan la asignatura. Ahora bien, el presente proyecto integra el diseño y la implementación de instrumentos de medida y de generación de señales necesarios para la experimentación en los contenidos de la asignatura, instrumentos de medida como un osciloscopio y un analizador de espectros e instrumentos productores como un generador de funciones, además, la incorporación de un asistente virtual para el asesoramiento acerca del manejo de los instrumentos y del propio fenómeno físico; todo ello, basado en tecnologías de código abierto, aprovechando su bajo coste y versatilidad haciendo que esta solución sea asequible. La solución proporciona las funcionalidades básicas para trazar ondas periódicas de diferentes amplitudes y frecuencias características de escenarios de la Ingeniería Electrónica y así poder medirlas en el dominio del tiempo y la frecuencia, también presenta la posibilidad de realizar operaciones básicas entre estas señales. Todas estas características están disponibles a través de una interfaz gráfica intuitiva e interactiva. Metodológicamente se definen las fases denominadas: formulación, diseño, construcción, puesta en marcha, evaluación y gestión de cambios; las cuales permitirán el alcance exitoso de los objetivos propuestos en esta investigación.*

Palabras Clave — **Generador De Señales, Osciloscopio, Espectro, Raspberry Pi Pico.**

I. INTRODUCCIÓN

Actualmente, el campo de las telecomunicaciones es una de las ramas de la ingeniería electrónica con más demanda en Colombia [1], [2]. En Norte de Santander por parte de la UFPS

la investigación en telecomunicaciones se realiza a través del Grupo de Investigación y Desarrollo en Electrónica y Telecomunicaciones (GIDET), que cuenta con categoría B en Colciencias. El grupo tiene como objetivo general conformar el grupo de investigación en Electrónica y Telecomunicaciones para capacitación, realización y divulgación de proyectos de investigación en el área de las telecomunicaciones. La vinculación del Semillero de Investigación de Educación de Ingeniería (SIEI) al GIDET hace que la realización y divulgación de los proyectos de investigación del semillero estén enfocados en la educación de disciplinas de Ingeniería y el aprendizaje educativo de los saberes. El SIEI para este proyecto de investigación pretende realizar un laboratorio experimental basado en tecnologías de código abierto, los diferentes instrumentos de medición estarán interconectados por un asistente virtual que será controlado por voz, además, el asistente servirá como guía educativa aportando información sobre el manejo de los equipos y del propio fenómeno físico; se propone con ello una novedosa herramienta didáctica para la asignatura de teoría de señales y sistemas. Los dispositivos indispensables para la enseñanza práctica de la asignatura teoría de señales y sistemas de la UFPS son el generador de funciones, el osciloscopio y analizador de espectros. El primero permite producir los tipos de señales (seno, cuadrada y diente de sierra) que se estudian en la asignatura mientras que el segundo permite la visualización de las señales en el dominio del tiempo provenientes del generador de funciones y el tercero la visualización de las señales en el dominio de la frecuencia. Aunque estos dispositivos en el laboratorio son funcionalmente útiles, no están diseñados para ser productos portátiles o rentables. Es por esta limitación de accesibilidad económica y portabilidad que el laboratorio experimental se desarrolla sobre tecnología de código abierto. Plataformas como Raspberry Pi o Arduino han ido disminuyendo su precio y tamaño a través de los años, haciendo de estas nuevas tecnologías las mejores para el diseño de proyectos electrónicos. Proyectos anteriores han intentado crear dispositivos que, por separado, puedan funcionar como osciloscopio digital y generador de funciones [3]–[4]. Otros proyectos de diseño de osciloscopios digitales han utilizado el periférico de adquisición de señal analógica y digital mixta de bajo costo y alto rendimiento como lo son el Bitscope y el PiScope que es un analizador lógico (visor de forma de onda digital) para la Raspberry Pi. El proyecto de osciloscopio con

BitScope es un hardware especial diseñado para ser un osciloscopio basado en un computador personal programable, analizador lógico, generador de forma de onda, reloj y analizador de espectro. El modelo que se encamina por la misma temática que el presente proyecto es el PiScope; aunque está en una fase de desarrollo temprano y tiene mucho potencial de aplicación no cuenta con las funciones avanzadas que pretende incluir el laboratorio experimental, como: manejar la frecuencia de los gráficos, las mediciones de voltajes y la generación de ondas (seno, cuadrado, diente de sierra). Conociendo los resultados y avances de algunos proyectos que hay en la temática de los osciloscopios digitales y generadores de funciones se tiene una idea más clara de cómo poder aportar un valor agregado con el proyecto. Por ello, para el laboratorio experimental se realiza un dispositivo que funciona como osciloscopio digital y generador de funciones al mismo tiempo que al estar basado en la tecnología de código abierto Raspberry Pi tiene un costo de construcción mucho más asequible. La dualidad osciloscopio - generador del dispositivo se logra gracias a una configuración híbrida entre una Raspberry Pi 3 y un microcontrolador PIC32 [5], [6]. El dispositivo puede, a la vez, trazar y producir señales eléctricas que cambian con el tiempo a lo largo de una escala calibrada. También proporciona la funcionalidad básica del osciloscopio digital para trazar el voltaje variable en un periférico de salida como un monitor de escritorio y la capacidad básica del generador de funciones para producir señales eléctricas periódicas de diferentes amplitudes y frecuencias. Al momento de utilizar el dispositivo, el estudiante que asiste a la práctica del laboratorio podrá aprovechar funciones avanzadas que incluyen: medir el voltaje de pico a pico de una señal generada y manejar el esquema de pasos de tiempo para las señales eléctricas detectadas; logrando así una mejor visualización de la gráfica. Para el generador de funciones se incluyen aspectos como: configurar una frecuencia variable, variar la amplitud y elegir el tipo de señal generada (seno, cuadrado y diente de sierra). Las características del osciloscopio y su función de generación están disponibles a través de una GUIDE intuitiva e interactiva [7]. Las características con las que cuenta el dispositivo hacen que sea una alternativa simple y fácil de usar. Con el tamaño reducido que brindan las tecnologías de código abierto utilizadas y la configuración híbrida que permite tener dos dispositivos en un solo elemento se logra obtener un dispositivo pequeño y práctico fácil de movilizar.

II. PROBLEMA

1.1 PLANTEAMIENTO DEL PROBLEMA

Teoría de señales y sistemas, como asignatura obligatoria en el pensum del programa de Ingeniería Electrónica de la Universidad Francisco de Paula Santander, es fundamental en el ámbito profesional de un ingeniero electrónico, por ello es menester que durante su periodo de formación le sea impartido

todo el conocimiento práctico que sea posible. Actualmente, los laboratorios con los que cuenta el programa de Ingeniería Electrónica para la asignatura Teoría de Señales y Sistemas son de simulación computacional, basados en la herramienta de software Matlab®, contar con un laboratorio de esas características hace que la línea de aprendizaje sea interactiva; no obstante, dificulta un fortalecimiento de lo aprendido que solo se concretiza con la aplicación del conocimiento. Es esa necesidad la que se quiere mitigar con este proyecto el cual planea atacar las falencias prácticas, incentivando al estudiante a la investigación de los fenómenos de las señales y sistemas, aún más, a las situaciones reales con las que se encuentra un profesional de la Ingeniería Electrónica en su campo de desempeño. Por otra parte, la mayoría de los estudiantes que han cursado la asignatura no culminan con resultados académicos satisfactorios. Es por todo lo mencionado que este proyecto se pretende realizar, contribuyendo al perfil profesional del programa donde se enuncia que los egresados deben contar con la capacidad para el aprendizaje autónomo, adquisición y aplicación de la información de forma independiente, valorar los procesos de aprendizaje permanente y trabajar como miembro de un equipo en forma cooperativa, respetuosa, creativa y responsable; perfil que se cumple en la mayoría de los casos. Es cuando surge la pregunta de investigación: ¿Cómo desarrollar e implementar un laboratorio experimental operado con el apoyo de un asistente virtual para la enseñanza de Teoría de Señales implementando tecnologías de código abierto que pueda dar a los estudiantes que cursan la asignatura una herramienta de estudio que sea lo más cercana a la realidad, dando así una nueva alternativa para el aprendizaje de la naturaleza de las señales?

1.2 FORMULACIÓN DEL PROBLEMA.

¿Es posible desarrollar un laboratorio experimental operado con el apoyo de un asistente virtual para la enseñanza de teoría de señales implementando tecnologías de código abierto?

III. JUSTIFICACIÓN

¿Para qué es importante este trabajo de investigación o proyecto? Partiendo de la necesidad presentada en el planteamiento del problema, se hace necesario evaluar cuál podría ser la mejor alternativa para innovar en la práctica docente con los temas relacionados a la teoría de señales, siendo las aplicaciones computacionales las de mejores resultados hoy día, pues estas respaldan las tendencias educativas actuales que van dirigidas a la transformación del rol y la función de los profesores hacia un modelo de enseñanza centrado en el estudiante y basado en el desarrollo de competencias y de aprendizaje autónomo [8]; por ello se busca crear un laboratorio experimental que permita adaptarse a las necesidades de aprendizaje del estudiante, apoyado en las bondades que ofrecen las tecnologías de código abierto como su accesibilidad,

interactividad y flexibilidad, convirtiéndolas en aliadas poderosas de las nuevas estrategias de aprendizaje, especialmente en la educación superior [9]. Por otra parte, en varias investigaciones se ha determinado que la aplicación de tecnologías en la práctica docente influye significativamente en el rendimiento académico de los estudiantes en áreas relacionadas con las matemáticas; experiencias previas [10]–[11] muestran que la utilización de nuevas tecnologías ayuda a los alumnos a aprender las áreas de las ciencias, que les permite mejorar la comprensión, descubrir por si mismos conceptos y por ende, desarrolla en ellos un aprendizaje significativo y las competencias deseadas, además de influir en su motivación lo que les lleva a profundizar en sus estudios de forma independiente, permitiéndoles mejorar sus calificaciones [12], [13]. Simular el comportamiento de las señales sería muy complementario comparándolo a los resultados que se obtengan en el laboratorio experimental donde se manipula el fenómeno real, es decir, permite contrastar el método matemático, método computacional y método práctico, obteniendo así una reducción en el porcentaje de error. El laboratorio de simulación de teoría de señales basado en Matlab® no debe ser excusa para que no se pueda intentar otras formas o maneras de enseñar conocimiento práctico a los estudiantes. Ahora bien, el desarrollo del conocimiento exige una estructura de enseñanza basada en la experimentación, pasando de la observación de efectos, hacia la explicación de los mismos, que permita probar las hipótesis, y finalmente obtener, desde criterios propios, los conceptos teóricos mediante el análisis de los resultados, y así lograr la creación de nuevos conocimientos impulsando la investigación [14]. Es importante resaltar el grado de originalidad que presenta esta propuesta, ya que no se evidencia en la revisión de antecedentes la existencia de una herramienta experimental de tecnologías de código libre orientada a la enseñanza de la teoría de señales, donde se pueda apreciar una aplicación práctica del conocimiento impartido en interacción con el sujeto que aprende, constituyendo una poderosa herramienta para la formación de los estudiantes de ingeniería en la resolución de problemas reales [15].

IV. OBJETIVOS

1.3 OBJETIVO GENERAL

- Desarrollar un laboratorio experimental operado con el apoyo de un asistente virtual para la enseñanza de Teoría de Señales implementando tecnologías de código abierto.

1.4 OBJETIVOS ESPECÍFICOS

- Diseñar la arquitectura tecnológica del laboratorio experimental.

- Crear un generador de funciones, un analizador de espectro y un osciloscopio con tecnologías de código abierto.
- Programar un asistente virtual para brindar apoyo al manejo de los equipos de medición.
- Verificar el correcto funcionamiento del prototipo final.

V. REFERENTE TEORICO

Hay otros proyectos que han intentado crear un modelo de osciloscopio digital, analizador de espectro y generador de funciones por separado. Algunos proyectos similares de osciloscopio digital incluyen Bitscope y PiScope. El proyecto BitScope es un hardware especial diseñado como osciloscopio basado en PC programable, analizador lógico, generador de forma de onda y analizador de espectro. El proyecto similar al sustentado en el presente documento es el PiScope, pero aún está en desarrollo temprano. Además, utiliza una pantalla LCD (en lugar de un monitor) y no tiene algunas de las funciones avanzadas que tendrá disponible el proyecto (como guardar gráficos y mediciones) y no puede generar ondas (seno, cuadrado, diente de sierra). El proyecto de laboratorio experimental proporciona un osciloscopio digital y un analizador de espectro en un solo dispositivo, y se tendrá algunas de las características avanzadas proporcionadas por los osciloscopios de laboratorio. A continuación, se presentan los conceptos claves en el desarrollo de esta propuesta: Teoría de Señales y Sistemas Para cualquier tipo de señal, sin importar su naturaleza, la teoría de señales y sistemas pretende una caracterización de ellas, ya sea, en el dominio del tiempo o en el dominio de la frecuencia. Para los sistemas el estudio se basa en cómo éstos modifican, de múltiples maneras, a las señales. Un sistema, ya sea de filtración, amplificación o atenuación, es un ejemplo claro del tratamiento que éstos le dan a las señales. En base a lo anterior, se puede decir que el objetivo del estudio en la teoría de señales y sistemas es el de conocer y clasificar las señales para tener la capacidad de saber cómo tratarlas, transmitir las o extraer información de ellas [16]. Este objetivo hace que el estudio de la teoría de señales y sistemas sea fundamental para la ingeniería electrónica en todos los niveles de formación, se puede apreciar su aplicación en áreas como procesamiento de señales, sistemas de comunicación y sistemas de control. Por todas estas aplicaciones es esencial que el estudiante tome la asignatura con un entendimiento previo global de las señales y los sistemas. Autores como Alan Victor Oppenheim y Simon S. Haykin han brindado a la teoría de señales y sistemas un tratamiento dirigido al estudiante de una manera práctica e integrada. El abordaje hecho por estos autores refleja la importancia de las señales en tiempo continuo y tiempo discreto en la práctica de la ingeniería. Los dominios en los

cuales puede estar presente una señal o un sistema se tratan a la par en la materia. Esto se hace, específicamente, como una ventaja pedagógica para ayudar al estudiante a ver las similitudes y diferencias fundamentales entre las representaciones en tiempo continuo y en tiempo discreto. Es por esta implicación práctica que un tratamiento integrado puede preparar al estudiante en el uso práctico de los conceptos que imparte la teoría de señales y sistemas [17], [18]. Laboratorios de Teoría de Señales en la Universidad Francisco de Paula Santander Para las prácticas de laboratorio de las asignaturas que emplean la simulación computacional la Universidad Francisco de Paula Santander cuenta con un aula especializada equipada con veinte computadores estándar. En este laboratorio ubicado en el primer piso del edificio Aulas Generales (AG) se imparten las prácticas de laboratorio para asignaturas como: Teoría electromagnética, Teoría de circuitos, Medios de Transmisión, Ecuaciones Diferenciales y Teoría de Señales y Sistemas. Para el caso de los laboratorios prácticos de la asignatura Teoría de Señales y Sistemas se utiliza el software computacional Matlab® para simular las condiciones reales en las que un sistema modifica a una señal. Antecedentes de laboratorios de Teoría de Señales y Sistemas En un contexto de globalización y competencia laboral, es necesario que los estudiantes tengan acceso a tecnología emergente. Es por ello que a los estudiantes en etapa de formación se les debe impartir el conocimiento de la forma más didáctica posible. El conocimiento tanto teórico como práctico debe ser enseñado de forma que el estudiante adquiera las competencias y aptitudes necesarias para desempeñar el rol que le confiere su programa de pregrado. En el caso del entorno pedagógico y metodológico de la ingeniería electrónica se han empleado formas didácticas basadas en conocimiento teórico aplicado en prácticas experimentales que no se asemejan mucho a las condiciones de la vida real. En el estudio titulado: La enseñanza en el Laboratorio. ¿Qué Podemos Aprender en Términos de Conocimiento Práctico y de Actitudes hacia la Ciencia?, la autora Séré, Marie-Geneviève realiza su investigación con el fin de divulgar y destacar los aportes que las prácticas experimentales hacen a la enseñanza del área de conocimiento que se imparte. Haciendo para ello una división esquemática de los diferentes tipos de resultados que se esperan de la enseñanza. Entre estos resultados están: comprender y aprender la teoría, realizar experiencias en base a esa teoría y aprender los procedimientos. Esta forma de preparar experiencias, que empiezan en el aprendizaje de lo conceptual para después pasar a lo práctico, los investigadores la describen como el pasaje «del mundo de los objetos al mundo de las ideas». Este paso de lo conceptual a lo práctico muy menudo se realiza bastante mal [19]. Para la enseñanza de la teoría de señales de sistemas, frecuentemente, los docentes preparan una línea de aprendizaje que empieza en la transferencia del conocimiento conceptual para luego pasar a la comprobación práctica de lo aprendido. Tal es el caso de la implementación de

un laboratorio virtual para el estudio de dispositivos electrónicos que se realizó en la Universidad Nacional de La Plata. Implementación de un laboratorio virtual para que a través de Internet cumpla con objetivos pedagógicos en el estudio de los dispositivos electrónicos básicos usados en la enseñanza de la electrónica en las universidades [20]. El Osciloscopio El osciloscopio es un instrumento que permite visualizar en el dominio del tiempo diferentes tipos de ondas o fenómenos los cuales se presentan en los circuitos eléctricos y electrónicos. Por ejemplo, en el caso de los portátiles, las formas de las ondas encontradas de los distintos puntos de los circuitos están bien definidas, y mediante su análisis podemos diagnosticar con facilidad cuáles son los problemas del funcionamiento. Los osciloscopios son de los instrumentos más versátiles que existen y los utilizan desde técnicos de reparación de televisores hasta médicos. Son dispositivos que visualizan una gráfica que muestra señales eléctricas variables en el tiempo. Un osciloscopio puede medir un gran número de fenómenos, provisto del transductor adecuado (un elemento que convierte una magnitud física en señal eléctrica) será capaz de darnos el valor de una presión, ritmo cardíaco, potencia de sonido, nivel de vibraciones en un coche, etc. El osciloscopio de trazo doble tiene una fuerte ventaja frente al osciloscopio de trazo simple, pues en el primero pueden visualizarse simultáneamente un fenómeno o forma de onda y con una imagen más clara. El funcionamiento del osciloscopio está basado en la posibilidad de desviar un haz de electrones por medio de la creación de campos eléctricos y magnéticos. En casi todos los osciloscopios, la desviación electrónica, también conocida como deflexión, se consigue mediante campos eléctricos. Ello constituye la deflexión electrostática. Raspberry Pi 3 modelo B Raspberry es una serie de ordenadores de placa reducida es decir una placa computadora de bajo coste desarrollada en el Reino Unido por la Fundación Raspberry pi, con el objetivo de estimular la enseñanza de la informática en las escuelas. La placa Raspberry Pi 3 modelo B es la tercera generación de Raspberry Pi y viene con una presentación muy estética, envuelta en un sobre con el logo de Raspberry Pi en el interior de su caja. Se puede implementar como servidor web de bajo consumo para construir un Smart tv, también se puede usar como almacenador de código abierto e incluso como localizador GPS. Generador de Funciones Un generador de funciones sirve para construir señales específicas con determinadas características. Por ejemplo, crear una señal que, de forma de onda cuadrada, sinusoidal o diente de sierra con una determinada frecuencia, y de cualquier amplitud. Con este instrumento se logra aplicar la señal generada a un circuito para analizar su respuesta. e interpretar los cambios que resulten del fenómeno. Un generador de funciones tiene un valor máximo de frecuencia y de amplitud para caracterizar una señal. Los generadores de funciones típicos generan ondas sinusoidales, triangulares y cuadradas. Otros generadores, por su parte, tienen señales programables como por

ejemplo la de un electrocardiograma. Existen también equipos con la función de generar una señal de impulso. Ahora bien, se explicará los bloques de funcionamiento del generador de señales:

- **Regulador de Frecuencia:** En este bloque se puede variar la señal de salida desde 0 Hz (onda continua) hasta el valor máximo permitido por el instrumento de acuerdo a tecnología y fabricación.
 - **Regulador de Ciclo de Trabajo y Offset:** El primero permite regular cada mitad del ciclo de la señal de salida y el Offset realizar un desplazamiento vertical de la onda.
 - En el tercer bloque se ubica el tablero de números con los cuales se especifican los parámetros de la onda de salida y los multiplicadores de frecuencia que pueden estar en unidades de Hz, kHz y MHz.
 - En el cuarto bloque se ubica la salida de la señal con dos conectores de tipo BNC [21].
- Modelo Experiencial de Kolb** El modelo de aprendizaje elaborado por Kolb supone que para aprender algo se debe trabajar o procesar la información que se recibe. Kolb identificó dos dimensiones principales del aprendizaje: la percepción y el procesamiento. Kolb dice que se puede partir de una experiencia directa y concreta, o bien de una experiencia abstracta, que es la que se tiene cuando se lee acerca de algo o cuando se recibe de alguien. El aprendizaje es el resultado de la forma como las personas perciben y luego procesan lo que han percibido [22], [23]. Las experiencias que se tengan, concretas o abstractas, se transforman en conocimiento cuando se elaboran de alguna de dos formas: reflexionando y pensando sobre ellas (alumno reflexivo), o experimentando de forma activa con la información recibida (alumno pragmático). Según el modelo de Kolb un aprendizaje óptimo es el resultado de trabajar la información en cuatro fases: Actuar, Reflexionar, experimentar y Teorizar (Figura 1).



Figura 1. Modelo de David Kolb, aprendizaje basado en experiencias.

formulación, diseño, construcción, puesta en marcha, evaluación y gestión de cambios; las cuales permitirán el alcance exitoso de los objetivos propuestos en esta investigación.

VI. METODOLOGÍA

La presente investigación se sustenta sobre un paradigma pragmático pues aboga por métodos y modelos mixtos o híbridos, donde las relaciones investigador, fenómeno, participantes están determinadas en función de cada estudio en particular y por el contexto [28]. Por tanto, este enfoque es aplicable a la práctica docente, y permite describir el desarrollo de actividades que se emplean para trabajar el currículo, describiendo a su vez el rol del docente. También, se puede evidenciar un diseño de estrategias para un plan de mejoramiento sobre las actividades académicas, donde el docente es quien hace parte de la recolección de información, implementación de las actividades propuestas, analiza y hace seguimiento sobre el proceso que los estudiantes van a realizar con la propuesta del presente trabajo. Teniendo en cuenta la extensión, el objeto de estudio está acotado a los programas de ingeniería de la Universidad Francisco de Paula Santander, donde los sujetos docentes y estudiantes, deben ser observados desde una dimensión objetiva, conceptualizando la naturaleza del objeto deductivamente a través de un proceso de construcción teórico y observacional; todo lo anterior enfatizando en el curso de Teoría de Señales del programa de Ingeniería Electrónica. Por otra parte, se definen las fases denominadas: Diagnóstico, Formulación, Construcción, Gestión de Cambios, Evaluación y Análisis. A continuación, se mencionan las fases o momentos a través de los cuales la metodología propuesta pretende dar cumplimiento a los objetivos específicos de la siguiente manera: la ejecución de la fase de Diagnóstico da cumplimiento al primer objetivo, la ejecución de la fase de Formulación da cumplimiento al segundo objetivo específico, la ejecución de las fases Construcción y Gestión de Cambios dan cumplimiento al tercer objetivo específico y finalmente las fases de Evaluación y Análisis permitirá ejecutar el último objetivo específico.

VII. RESULTADOS ESPERADOS

Con el desarrollo del proyecto se espera generar nuevo conocimiento en producto tecnológico e innovación realizando un registro de software y una socialización en ponencia o un artículo científico con los resultados de la investigación contribuyendo a la apropiación social del conocimiento. Se espera también la formación de recurso humano en Ciencia, tecnología e innovación al desarrollar competencias en los estudiantes investigadores del proyecto. A corto plazo se espera mejorar la práctica docente en el aula en cursos de Teoría de Señales, así como optimizar el aprendizaje de las temáticas desarrolladas en cuanto a señales y sistemas haciendo que la

curva de aprendizaje de dichos conceptos sea más favorable. A largo plazo se espera contribuir en el desempeño de los estudiantes del curso en mención y así mejorar sus competencias profesionales.

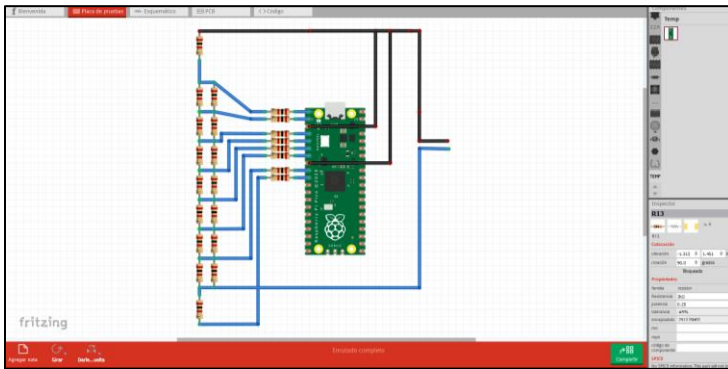
VIII. CONCLUSIONES

- Se puede concluir que el laboratorio experimental operado con el apoyo de un asistente virtual para la enseñanza de teoría de señales implementando tecnologías de código abierto tiene como punto crucial desarrollar un nuevo conocimiento en el área de la tecnología junto a lo teórico, se enfasca que el conocimiento impartido en el aula de clase se lleva un punto donde el estudiante desarrolle la apropiación del mismo, de tal manera que comprenda y desenvuelva dicho conocimiento.
- La Programación de un asistente virtual para brindar apoyo al manejo de los equipos de medición incentivara al estudiante a la investigación de los fenómenos de las señales y sistemas, aún más, a las situaciones reales con las que se encuentra un profesional de la Ingeniería Electrónica en su campo de desempeño. Se espera también la formación de recurso humano en Ciencia, tecnología e innovación al desarrollar competencias en los estudiantes investigadores del proyecto.

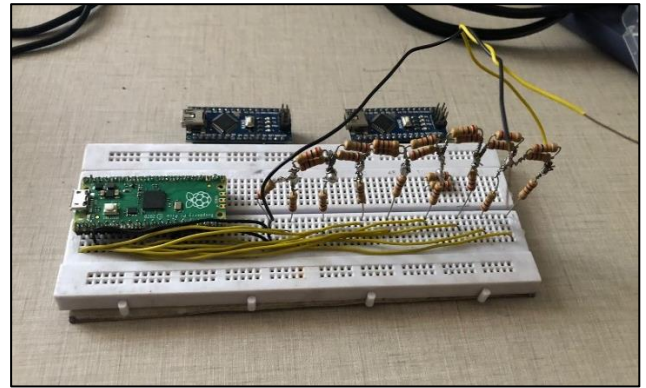
IX. BIBLIOGRAFIA

- [1] M. Correa, D. Ruiz, and V. Oviedo, "Análisis de los activos intangibles - NIC 38 sector de las telecomunicaciones en Colombia," Universidad Cooperativa de Colombia, 2018.
- [2] L. Mannign, "La formación técnico laboral en Colombia, sector de sistemas y telecomunicaciones: Impacto de sus egresados. El caso de Montería Córdoba (1996 - 2015)," Universidad de Cartagena, 2017.
- [3] J. A. C. Osorio, J. A. C. Osorio, and F. A. M. A., "Diseño y construcción de un osciloscopio digital implementado en Matlab," *Sci. Tech.*, vol. 1, no. 34, May 2007.
- [4] B. D. Sandoval Cueva, "Diseño e implementación de un osciloscopio digital con una interfaz gráfica realizada en sistema operativo android," May 2017.
- [5] G. Pasolini, A. Bazzi, and F. Zabini, "A Raspberry Pi-Based Platform for Signal Processing Education [SP Education]," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 151-158, Jul. 2017.
- [6] A. B. Raju, A. Itagi, S. Angadi, V. Sushma, and A. Shettar, "A Low Cost Open Source Hardware Tool for Integrated Learning Experience in Laboratories," *J. Eng. Educ. Transform.*, vol. 0, no. 0, pp. 217-222, Jan. 2015.
- [7] J. Garcillan, "Implementación de un procesador digital de señal con BitScope," Universidad Pontificia Comillas, 2017.
- [8] N. Carr, "The Crisis in Higher Education," *MIT Technology Review*, Sep-2012.
- [9] R. Cózar-Gutiérrez, M. V De Moya-Martínez, J. A. Hernández-Bravo, and J. R. Hernández-Bravo, "Conocimiento Y Uso De Las Tecnologías De La Información Y Las Comunicaciones (TIC) Según El Estilo De Aprendizaje De Los Futuros Maestros," *Form. Univ.*, vol. 9, no. 13, pp. 105-118, 2016.
- [10] G. Vergara, A. Avilez, and J. Romero, "Uso De Matlab Como Herramienta Computacional Para Apoyar La Enseñanza Y El Aprendizaje Del Álgebra Lineal," *MATUA Rev. Matemática La Univ. Del Atlántico*, vol. 3, no. 1, pp. 83-91, 2016.
- [11] I. M. Cruz-Pichardo and Á. Puentes-Puente, "Innovación Educativa: Uso De Las TIC En La Enseñanza De La Matemática Básica," *edmeti - Rev. Educ. Mediática y TIC*, vol. 1, no. 2, pp. 127-147, 2012.
- [12] C. M. Marcilla-De Frutos, "Las TIC En La Didáctica De Las Matemáticas," Universidad De Burgos, 2013.
- [13] E. E. Mantilla-Contreras, "Aplicación Del Modelo De Kolb En La Enseñanza De Las Telecomunicaciones Basadas En Mediciones De Densidad De Potencia Utilizando El Analizador Espectral Narda SRM-3000," Escuela Superior Politécnica Del Litoral, 2009.
- [14] A. Y. Kolb and D. A. Kolb, *Experiential Learning Theory: A Dynamic, Holistic, Approach to Management Learning, Education and Development*. London: Sage Publications, 2008.
- [15] M. L. González, J. Marchueta, and E. A. Vilche, "Modelo De Aprendizaje Experiencial De Kolb Aplicado A Laboratorios Virtuales En Ingeniería En Electrónica," in *I Jornadas Nacionales de TIC e Innovación en el Aula*, 2013, p. 8.
- [16] M. Ruiz Costa-Jussà Helenca and D. Barrobés, "Conceptos de señales y sistemas."
- [17] A. V. Oppenheim and A. S. Willsky, *Señales y Sistemas*, 2da ed. 1998.
- [18] S. Haykin and B. Van Veen, *Señales y Sistemas*, 1ra ed. 2001.
- [19] M.-G. Séré, *La Enseñanza en el laboratorio : ¿qué podemos aprender en términos de conocimiento práctico y de actitudes hacia la ciencia?*, vol. 20, no. 3. Ciències de l'Educació de la Universitat Autònoma de Barcelona, 2002.
- [20] C. A. Ibarra B., S. Medina S., and Á. Bernal N., "Implementación de un laboratorio virtual para el estudio de dispositivos electrónicos," *TE*, vol. no. 2, 2007.
- [21] D. Fajardo, R. Ramón, and F. Vicedo, "Osciloscopio digital mediante FPGA," Universidad Miguel Hernández, p. 10, 2004.
- [22] J. Gómez-Pawelek, "El Aprendizaje Experiencial." Universidad de Buenos Aires, Buenos Aires, Argentina, p. 21, 2011.
- [23] M. C. Sánchez-Gómez, "La Dicotomía Cualitativo - Cuantitativo: Posibilidades De Integración Y Diseños Mixtos," *Campo Abierto*, vol. Monográfico, pp. 11-30, 2015.

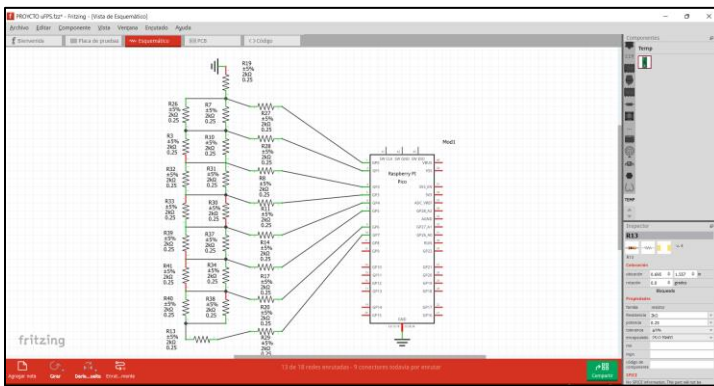
ANEXOS



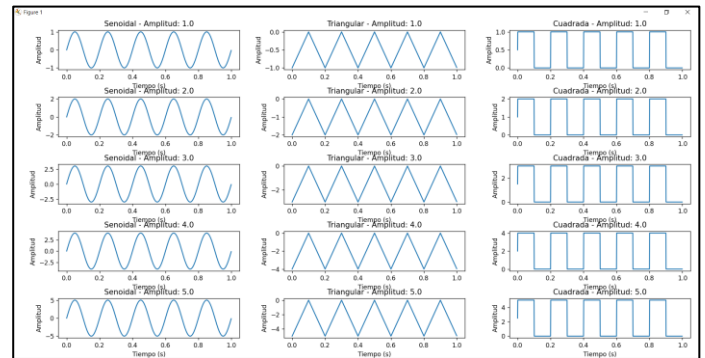
Anexo D.1. Diagrama en Fritzing del generador de señales.



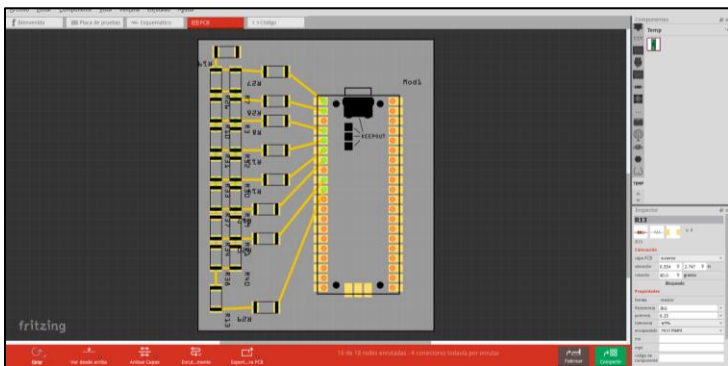
Anexo D.4. Generador de señales prototipo



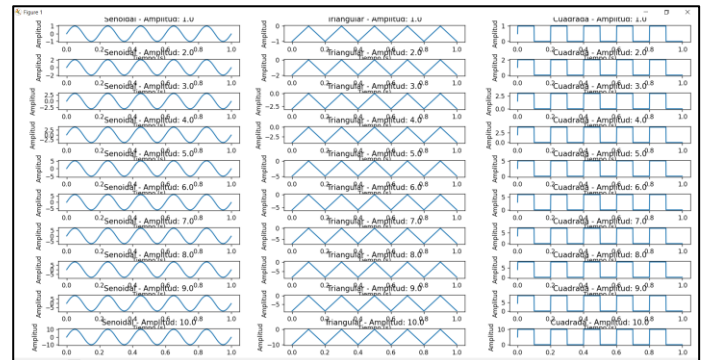
Anexo D.2. Diagrama esquemático en Fritzing del generador de señales.



Anexo D.5. Generador de señales prototipo en python



Anexo D.3. PCB en Fritzing del generador de señales.



Anexo D.6. Generador de señales prototipo en python

XV. APÉNDICES

A. APÉNDICE: CODIGO EN EL PROGRAMA VISUAL STUDIO LABORATORIO EXPERIMENTAL OPERADO CON EL APOYO DE UN ASISTENTE VIRTUAL PARA LA ENSEÑANZA DE TEORÍA DE SEÑALES IMPLEMENTANDO TECNOLOGÍAS DE CÓDIGO ABIERTO

```
import urllib.request
import json
import datetime
import wikipedia
import pyautogui, webbrowser
import AVMSpeechMath as sm
import serial
import sys
import time
import numpy as np
import matplotlib.pyplot as plt
import ipywidgets as widgets
from IPython.display import display
import numpy as np
import matplotlib.pyplot as plt

# UFPS
# NOMBRE:Jhon Jairo VejarCodigo:1161702
# LABORATORIO EXPERIMENTAL OPERADO CON EL APOYO DE UN ASISTENTE VIRTUAL
# PARA LA ENSEÑANZA DE TEORÍA DE SEÑALES IMPLEMENTANDO TECNOLOGÍAS DE CÓDIGO
# ABIERTO "LUCY"
name = 'lucy'
key = 'AIzaSyDhUcZpLfLI-3k31KJmcoqkq2q12TB17yM'
flag = 1

#led=0
#mot=95
#serialArduino = serial.Serial("COM5",9600)

listener = sr.Recognizer()

engine = pyttsx3.init()

voices = engine.getProperty('voices')
```



```

engine.setProperty('voice', voices[0].id)
#pyautogui.displayMousePosition()
engine.setProperty('rate', 178)
engine.setProperty('volume', 0.7)
def talk(text):
    engine.say(text)
    engine.runAndWait()

talk("Bienvenido a casa yon vejar, como puedo ayudarte")
def listen():

    flag = 1
    try:
        with sr.Microphone() as source:
            print("Escuchando...")
            voice = listener.listen(source)
            rec = listener.recognize_google(voice, language='es-ES')
            rec = rec.lower()

            if name in rec:
                print(rec)
                rec = rec.replace(name, ' ')
                flag = run(rec)
            else:
                talk("Vuelve a intentarlo, no reconozco: " + rec)
                print(rec)
    except:
        pass
    return flag

def run(rec):

    if 'reproduce' in rec:
        music = rec.replace('reproduce', '')
        talk('Reproduciendo ' + music)
        pywhatkit.playonyt(music)
    elif 'grafica senal' in rec:
        def crear_grafico(frecuencia, amplitud, tiempo_inicio, tiempo_fin):
            tiempo = np.arange(tiempo_inicio, tiempo_fin, 0.01)
            senal = amplitud * np.sin(2 * np.pi * frecuencia * tiempo)
            fig, ax = plt.subplots()
            linea, = ax.plot(tiempo, senal)
            ax.set_xlabel('Tiempo (s)')
            ax.set_ylabel('Amplitud')
            ax.set_title('Senal sinusoidal de {} Hz'.format(frecuencia))
            return fig, ax, linea

```

```

def actualizar_grafico(frecuencia, amplitud, tiempo_inicio,
tiempo_fin):
    _, _, linea = crear_grafico(frecuencia, amplitud, tiempo_inicio,
tiempo_fin)
    plt.show()

    widgets.interact(
        actualizar_grafico,
        frecuencia=widgets.FloatSlider(value=5, min=0, max=10, step=0.1,
description='Frecuencia'),
        amplitud=widgets.FloatSlider(value=1, min=0, max=2, step=0.1,
description='Amplitud'),
        tiempo_inicio=widgets.FloatSlider(value=0, min=0, max=10,
step=0.1, description='Tiempo inicio'),
        tiempo_fin=widgets.FloatSlider(value=1, min=0, max=10, step=0.1,
description='Tiempo fin')
    )
elif 'grafica senal senoidal' in rec:

    # Parámetros de la señal
    frecuencia_muestreo = 1000 # Frecuencia de muestreo en Hz
    duracion = 1 # Duración de la señal en segundos
    frecuencia_senoidal = 5 # Frecuencia de la señal senoidal en Hz
    amplitud_senoidal = 1 # Amplitud de la señal senoidal
    duty_cycle_cuadrada = 0.5 # Ciclo de trabajo de la señal cuadrada
    (0.5 para una onda cuadrada simétrica)

    # Generar tiempos
    t = np.linspace(0, duracion, int(frecuencia_muestreo * duracion),
endpoint=False)

    # Generar señal senoidal
    senoidal = amplitud_senoidal * np.sin(2 * np.pi * frecuencia_senoidal
* t)

    # Generar señal triangular
    triangular = 2 * amplitud_senoidal * (np.abs(t * frecuencia_senoidal
- np.floor(t * frecuencia_senoidal + 0.5)) - 0.5)

    # Generar señal cuadrada
    cuadrada = amplitud_senoidal * (np.sign(np.sin(2 * np.pi *
frecuencia_senoidal * t)) + 1) * 0.5

    # Graficar las señales
    plt.figure(figsize=(12, 6))

    plt.subplot(3, 1, 1)
    plt.plot(t, senoidal)

```

```

plt.title('Senal Senoidal')

plt.subplot(3, 1, 2)
plt.plot(t, triangular)
plt.title('Senal Triangular')

plt.subplot(3, 1, 3)
plt.plot(t, cuadrada)
plt.title('Senal Cuadrada')

plt.tight_layout()
plt.show()
elif 'Generador de senales' in rec:

    # Parámetros comunes
    frecuencia_muestreo = 1000 # Frecuencia de muestreo en Hz
    duracion = 1 # Duración de la señal en segundos
    frecuencia_senoidal = 5 # Frecuencia de la señal senoidal en Hz

    # Amplitudes para las 5 señales
    amplitudes = np.linspace(1, 5, 5)

    # Generar tiempos
    t = np.linspace(0, duracion, int(frecuencia_muestreo * duracion),
endpoint=False)

    # Graficar 5 señales con diferentes amplitudes y tipos
    plt.figure(figsize=(12, 10))

    for i, amplitud in enumerate(amplitudes, start=1):
        senoidal = amplitud * np.sin(2 * np.pi * frecuencia_senoidal * t)
        triangular = 2 * amplitud * (np.abs(t * frecuencia_senoidal -
np.floor(t * frecuencia_senoidal + 0.5)) - 0.5)
        cuadrada = amplitud * (np.sign(np.sin(2 * np.pi *
frecuencia_senoidal * t)) + 1) * 0.5

        # Graficar señales senoidales
        plt.subplot(5, 3, 3 * i - 2)
        plt.plot(t, senoidal)
        plt.title(f'Senoidal - Amplitud: {amplitud}')
        plt.xlabel('Tiempo (s)')
        plt.ylabel('Amplitud')

        # Graficar señales triangulares
        plt.subplot(5, 3, 3 * i - 1)
        plt.plot(t, triangular)
        plt.title(f'Triangular - Amplitud: {amplitud}')
        plt.xlabel('Tiempo (s)')

```

```

plt.ylabel('Amplitud')

# Graficar señales cuadradas
plt.subplot(5, 3, 3 * i)
plt.plot(t, cuadrada)
plt.title(f'Cuadrada - Amplitud: {amplitud}')
plt.xlabel('Tiempo (s)')
plt.ylabel('Amplitud')

plt.tight_layout()
plt.show()
elif 'enciende invernadero' in rec:
    # import pyautogui
    # pyautogui.displayMousePosition()
    # con el anterior puedo ubicar el mouse en la pantalla lo cual
    # facilita el uso de pyautogui .JJVC.
    # google = 43,271
    # pyautogui.typewrite('prueba 1')
    # pyautogui.press("enter")
    # pyautogui.click(x=100, y=200)
    # abrir(google,click=2)
    # abrir(direc)
    def abrir(pos,click=1):

        pyautogui.moveTo(pos)
        pyautogui.click(clicks=click)
        esc=1979,1070
        abrir(esc)
        time.sleep(1)
        Arduino=1877,70
        abrir(Arduino,2)
        time.sleep(10)
        verificar=16,70
        abrir(verificar)
        pos=87,993
        pyautogui.moveTo(pos)
        time.sleep(10)
        pyautogui.drag(70, 0, duration=1) # move right
        time.sleep(2)
        pos2=145,883
        pyautogui.moveTo(pos2)
        pyautogui.drag(1790, 0, duration=1)
        pyautogui.hotkey("ctrl", "c")
        time.sleep(1)
        esc=1979,1070
        abrir(esc)
        time.sleep(1)
        proteus=1836,234

```

```

    abrir(proteus,2)
    time.sleep(8)
    pos3=817,382
    pyautogui.moveTo(pos3)
    pyautogui.click(button='right')
    pos4=826,382
    abrir(pos4)
    time.sleep(2)
    pos5=866,405
    abrir(pos5)
    time.sleep(1)
    pyautogui.hotkey("ctrl","a")
    time.sleep(2)
    pyautogui.hotkey("ctrl","v")
    time.sleep(2)
    pos5=1212,309
    abrir(pos5)
    time.sleep(2)
    pos6=32,1010
    abrir(pos6)
    time.sleep(1)
    time.sleep(10000)
elif 'realiza un dibujo' in rec:
    for i in range (101):
        time.sleep(0.05)
        sys.stdout.write("\r%d %" % i)
        sys.stdout.flush()
    # import pyautogui
    # pyautogui.displayMousePosition()
    # con el anterior puedo ubicar el mouse en la pantalla lo cual
    # facilita el uso de pyautogui .JJVC.
    # google = 43,271
    # pyautogui.typewrite('prueba 1')
    # pyautogui.press("enter")
    # pyautogui.click(x=100, y=200)
    # abrir(google,click=2)
    # abrir(direc)
def abrir(pos,click=1):
    pyautogui.moveTo(pos)
    pyautogui.click(clicks=click)
    esc=1979,1070
    abrir(esc)
    bsc=150,1070
    abrir(bsc)
    pyautogui.typewrite('paint')
    time.sleep(1)
    pyautogui.press("enter")
    time.sleep(1)

```

```

pyautogui.hotkey("alt", "space")
time.sleep(0.5)
pyautogui.typewrite("x")
time.sleep(1)
pn=845,531
abrir(pn)
distance = 200
while distance > 0:
    pyautogui.drag(200, 0, duration=0.5) # move right distance -= 5
    pyautogui.drag(0, 200, duration=0.5) # move down
    pyautogui.drag(- 200, 0, duration=0.5) # move left distance -= 5
    pyautogui.drag(0, - 200, duration=0.5)
elif 'cuantos suscriptores tiene' in rec:
    name_subs = rec.replace('cuantos suscriptores tiene', '')
    data =
urllib.request.urlopen(f'https://www.googleapis.com/youtube/v3/channels?part=statistics&forUsername={name_subs.strip()}&key={key}').read()
    subs = json.loads(data) ["items"] [0] ["statistics"] ["subscriberCount"]
    talk(name_subs + " tiene {:,d}".format(int(subs)) + " suscriptores!")
elif 'enviar mensaje por whatsapp' in rec:
    webbrowser.open(f'https://web.whatsapp.com/send?phone={+573105768010}')
)
time.sleep(45)
pyautogui.typewrite('prueba 1')
pyautogui.press("enter")
elif 'hora' in rec:
    hora = datetime.datetime.now().strftime('%I:%M %p')
    talk("Son las " + hora)
elif 'busca' in rec:
    order = rec.replace('busca', '')
    wikipedia.set_lang("es")
    info = wikipedia.summary(order, 1)
    talk(info)
elif 'busca en google' in rec:
    pywhatkit.search(rec)
    talk('Estoy Buscando en Google')
elif 'cuentame un chiste' in rec:
    talk("no soy graciosa pero me se un chiste, ay va, Por que lloraba el
libro de matematicas, Porque tenia muchos problemas, jaja no te lo
esperabas cierto")
elif 'que eres' in rec:
    talk('soy una asistente virtual, disenada con el fin de facilitar la
vida cotidiana, fui disenada por yon vejar')
    #Autor Jhon Jairo Vejar
elif 'cuentame un trabalengua' in rec:
    talk('no se muchos pero me se un trabalengua, ay va,Tres tristes
tigres,comen trigo en un trival,un tigre, dos tigres, tres tigres.')
elif 'cuentame un poema' in rec:

```

```

talk('ay va, entrando a modo poeta, Cada vez que pienso en ti, mis
ojos rompen en llanto, y muy triste me pregunto, por que te quiero tanto')
elif 'cuanto es' in rec:
    res = sm.getResult(rec)
    talk(res)
elif 'enviar mensaje por voz' in rec:
    talk("Que deseas enviarle")
    listen
    print(rec)
    time.sleep(10)
    talk(rec)
elif 'encender led' in rec:
    led = 1
    mot = 95
    cad = str(led) + "," + str(mot)
    serialArduino.write(cad.encode('ascii'))
    talk('Bueno yon!')
elif 'abrir youtube' in rec:
    #import pyautogui
    # pyautogui.displayMousePosition()
    # con el anterior puedo ubicar el mouse en la pantalla lo cual
    # facilita el uso pyautogui .JJVC.
    canciones = ["kurt sonreir", "kurt la mujer perfecta"]
    def abrir(pos, click=1):

        pyautogui.moveTo(pos)
        pyautogui.click(clicks=click)
        abrir(google, click=2)
        time.sleep(2)
        pyautogui.hotkey("alt", "space")
        time.sleep(0.5)
        pyautogui.typewrite("x")
        time.sleep(1)
        abrir(direc)
        pyautogui.typewrite("www.youtube.com")
        pyautogui.hotkey("enter")
        time.sleep(30)
    for i in range(len(canciones)):
        print(i)
        abrir(buscar)
        pyautogui.typewrite(canciones[i])
        pyautogui.hotkey("enter")
        time.sleep(10)
        abrir(cancion)
        time.sleep(15)
        abrir(pausa)
        print("programa terminado Jhon Vejar")
elif 'abrir instagram' in rec:

```



```

import pyautogui
# pyautogui.displayMousePosition()
# con el anterior puedo ubicar el mouse en la pantalla lo cual
facilita el uso de pyautogui .JJVC.
def abrir(pos,click=1):

    pyautogui.moveTo(pos)
    pyautogui.click(clicks=click)
    abrir(43,271,click=2)
    time.sleep(2)
    pyautogui.hotkey("alt","space")
    time.sleep(0.5)
    pyautogui.typewrite("x")
    time.sleep(1)
elif 'apagar luz' in rec:
    talk('bueno yon!')
    led=0
    mot = 95
    cad = str(led) + "," + str(mot)
    serialArduino.write(cad.encode('ascii'))
elif 'abrir puerta' in rec:
    talk('bueno yon!')
    led = 1
    mot=0
    cad = str(led) + "," + str(mot)
    serialArduino.write(cad.encode('ascii'))
elif 'tapar puerta' in rec:
    talk('bueno yon!, jaja no te lo esperabas cierto')
    led = 1
    mot = 95
    cad = str(led) + "," + str(mot)
    serialArduino.write(cad.encode('ascii'))
elif 'deten el servicio' in rec:
    flag = 0
    talk("Saliendo...")
else:
    talk("Vuelve a intentarlo, no reconozco: " + rec)
    cad = str(led) + "," + str(mot)
    serialArduino.write(cad.encode('ascii'))
return flag
google = 43,271
direc = 198,50
buscar = 360,136
cancion = 424,322
pausa = 131,567
while flag:
    flag = listen()

```

B. APÉNDICE : ANIMACION EN BLENDER DE LUCY

