

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO PARA EL
MONITOREO EN LÍNEA DE LAS TUBERÍAS DE TRANSPORTE
PETROLERO UTILIZANDO UN SISTEMA ACÚSTICO**

**DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE
RECEPCIÓN DE DATOS, DE UN INTERFACE HOMBRE-
MÁQUINA Y DEL SISTEMA DE PROCESAMIENTO DIGITAL DE
LA INFORMACIÓN EN UN COMPUTADOR PORTÁTIL PARA LA
INTERPRETACIÓN DE LOS DATOS RECIBIDOS Y HACER UNA
CORRELACIÓN MATEMÁTICA PARA DETERMINAR LA
LOCALIZACIÓN APROXIMADA DE UN POSIBLE EVENTO DE
PERTURBACIÓN**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
ELECTRÓNICA Y AUTOMATIZACIÓN**

RICARDO XAVIER RAMOS ALBA
ricardo.ramos@epn.edu.ec

DIRECTOR: Dr. FAUSTO EDUARDO AVALOS CASCANTE
eduardo.avalos@epn.edu.ec

DMQ, agosto 2023

CERTIFICACIONES

Yo, RICARDO XAVIER RAMOS ALBA declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

RICARDO XAVIER RAMOS ALBA

Certifico que el presente trabajo de integración curricular fue desarrollado por RICARDO XAVIER RAMOS ALBA, bajo mi supervisión.

Dr. FAUSTO EDUARDO ÁVALOS CASCANTE
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

Sr. Ricardo Xavier Ramos Alba

Dr. Fausto Eduardo Ávalos Cascante

DEDICATORIA

A mi familia: Manuel, Elvia, Fernando, Diana y Tatiana.

AGRADECIMIENTO

A mis padres, Manuel y Elvia que me han ayudado y apoyado en todo momento a lo largo de mi carrera.

A mi hermano, Fernando que me ha apoyado siempre que ha podido.

A mis hermanas, Diana y Tatiana que con su cariño me han ayudado mucho a lo largo de mi carrera.

A mi tía, Rosa que es como mi segunda mamá.

A mi amigo, Willam que más que un amigo es un hermano.

A la Escuela Politécnica Nacional, profesores y compañeros por todo su apoyo a lo largo de todos estos maravillosos años.

Tabla de Contenido

CERTIFICACIONES	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTO	IV
RESUMEN.....	VII
ABSTRACT.....	VIII
1. INTRODUCCIÓN.....	1
1.1 Objetivos.....	2
1.1.2 Objetivo General.....	2
1.1.3 Objetivos Específicos	2
1.2 Alcance	2
1.3 Marco Teórico	3
1.3.1 PROCESAMIENTO DIGITAL DE LA INFORMACIÓN.....	3
1.3.2 TÉCNICAS DE PROCESAMIENTO DIGITAL DE SEÑALES ACÚSTICAS	4
1.3.2.1 Filtrado	5
1.3.2.2 Detección de eventos.....	8
1.3.3 PROBLEMA MAESTRO- ESCLAVO	10
1.3.3.1 Comunicación Serial	10
1.3.3.2 Protocolos de Comunicación Serial.....	11
1.3.3.3 Comunicación Serial Maestro-Esclavo.....	13
1.3.4 MÓDULO HC05.....	15
1.3.5 MÓDULO ESP32.....	17
1.3.5.1 Comunicación serial con el módulo ESP32	18
1.3.6 ROUTER	19
1.3.7 INTERFACE GRÁFICA.....	22
1.3.7.1 Desarrollo de Interfaces en Matlab	22
2. METODOLOGÍA.....	23
2.1 DISEÑO DEL HADWARE	23
2.1.1 RECEPCIÓN DE DATOS	23
2.1.2 CONFIGURACIÓN MÓDULO HC05 MODO ESCLAVO.....	24
2.1.3 FILTRADO DE SEÑALES EN SENSORES DE EMISIÓN ACÚSTICA (ADAPTADO A LA NUEVA APROXIMACIÓN CON EL MÓDULO HC05)	26
2.1.4 TÉCNICAS DE TRANSMISIÓN Y ALGORITMOS PARA RECEPCIÓN E INTERPRETACIÓN DE DATOS (EN EL CONTEXTO DE COMUNICACIÓN A TRAVÉS DEL MÓDULO HC05).....	27

2.2	CONEXIÓN HC05 Y MÓDULO ESP32	28
2.3	ENVIO DE LA SEÑAL CON PROTOCOLO TCP/IP	38
2.3.1	CONEXIÓN WI-FI.....	38
2.3.2	ESTABLECER UN SOCKET TCP	39
2.4	SERVIDOR	39
2.4.1	INTERFAZ GRAFICA.....	43
3.	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	47
3.1	Pruebas y Validación.....	47
3.1.1	Sistema de procesamiento de datos.....	47
3.1.2	Recepción de datos.....	48
3.1.3	Procesamiento de datos e identificación del evento de perturbación. .	48
3.1.3.1	Identificación de la perturbación golpe.	48
3.1.3.2	Identificación de la perturbación perforación.....	52
3.2	CONCLUSIONES.....	54
3.3	RECOMENDACIONES	54
4.	REFERENCIAS BIBLOGRÁFICAS	56
5.	ANEXOS	58

RESUMEN

El enfoque central de este estudio se centra en el procesamiento de datos derivados de perturbaciones que ocurren en una tubería metálica que puede ser parte de un oleoducto. El objetivo principal es discernir con precisión si una perturbación particular se origina por un golpe o por una actividad de taladrado. Para lograr este propósito, se lleva a cabo un análisis meticuloso de las señales capturadas mediante la utilización de un módulo HC05 y un ESP32, ambos interconectados a través de una comunicación serial.

La metodología seguida en este trabajo se basa en la comparación de las amplitudes de las señales registradas en diferentes situaciones. El módulo HC05 y el ESP32 se emplean como dispositivos de captura para recoger datos en tiempo real de las perturbaciones en la tubería metálica. Estos datos son transmitidos a través del router, que actúa como un enlace entre los dispositivos de captura y el sistema de análisis.

Al comparar las amplitudes de las señales capturadas con el módulo HC05 y ESP32, se establece un criterio diferenciador entre las perturbaciones causadas por golpes y aquellas generadas por actividades de taladrado. Esta diferenciación se basa en patrones distintivos de amplitud que surgen en cada tipo de perturbación. Dicha comparación permite detectar diferencias sutiles y evaluar las características específicas de las señales recibidas.

El análisis de las amplitudes de las señales resulta ser una técnica eficiente y precisa para clasificar las perturbaciones en la tubería metálica. La información recopilada a través del módulo HC05 y el ESP32 se procesa y evalúa exhaustivamente para identificar patrones característicos asociados a golpes y perturbaciones producto de actividades de taladrado. Esta clasificación de perturbaciones es crucial para tomar decisiones informadas en la gestión y mantenimiento de las tuberías metálicas permitiendo una respuesta rápida ante situaciones potencialmente problemáticas.

En resumen, este trabajo se enfoca en el procesamiento de datos de perturbaciones en una tubería metálica con el objetivo de distinguir entre golpes y perturbaciones producto de actividades de taladrado. La comparación de las amplitudes de las señales recibidas a través de un módulo HC05 y un ESP32, respaldada por un análisis meticuloso, proporciona un método efectivo y preciso para identificar y clasificar las perturbaciones en la tubería. Esto contribuye significativamente a la toma de decisiones informadas en la gestión y el mantenimiento de la infraestructura de sistemas de tuberías.

ABSTRACT

The central focus of this study revolves around the data processing of disturbances occurring in a metallic pipe that may be part of an oil pipeline. The main objective is to accurately discern whether a particular disturbance originates from a "hit" or drilling activity. To achieve this purpose, a meticulous analysis of the captured signals is conducted using both an HC05 module and an ESP32, both interconnected through a serial communication.

The methodology followed in this work is based on comparing the amplitudes of the signals recorded under different scenarios. The HC05 module and the ESP32 are employed as capture devices to collect real-time data on disturbances in the metallic pipe. This data is transmitted through the router, which acts as a link between the capture devices and the analysis system.

By comparing the amplitudes of the captured signals in both configurations (HC05 module and ESP32), a distinguishing criterion is established between disturbances caused by "hits" and those generated by drilling activities. This differentiation is based on distinctive amplitude patterns that arise in each type of disturbance. This comparison enables the detection of subtle differences and the assessment of specific features in the received signals.

The analysis of signal amplitudes proves to be an efficient and accurate technique for classifying disturbances in the metallic pipe. The information gathered via the HC05 module and the ESP32 is thoroughly processed and evaluated to identify characteristic patterns associated with "hits" and disturbance from drilling activities. This classification of disturbances is crucial for making informed decisions in the management and maintenance of the metallic pipe, allowing for a swift response to potentially problematic situations.

In summary, this work focuses on the data processing of disturbances in a metallic pipe with the goal of distinguishing between "hits" and drilling activities. The comparison of signal amplitudes received through an HC05 module and an ESP32, supported by meticulous analysis, provides an effective and precise method for identifying and classifying disturbances in the pipeline. This significantly contributes to making informed decisions in the management and maintenance of the pipe systems.

1. INTRODUCCIÓN

En la era de la información y la tecnología, el desarrollo de sistemas de interfaz se ha convertido en un campo de investigación crucial. Estos sistemas desempeñan un papel fundamental en una amplia gama de aplicaciones, desde dispositivos móviles hasta sistemas de control industrial. Permiten la interacción entre los seres humanos y las máquinas, facilitando la comunicación y el intercambio de información de manera eficiente y efectiva [1].

El avance de la tecnología ha llevado a una creciente demanda de interfaces hombre-máquina más intuitivas, fáciles de usar y capaces de interpretar la información con precisión. En este contexto, el presente trabajo se enfoca en el desarrollo e implementación de un sistema de recepción de datos de un interfaz hombre-máquina, así como del sistema de procesamiento digital de la información en un computador portátil para la interpretación de los datos recibidos [2].

El propósito central de este trabajo de integración curricular radica en afrontar los retos inherentes a la recepción y análisis de información originada en una interfaz, y en consecuencia, diseñar un sistema capaz de decodificar estos datos con elevada eficacia y exactitud. La esencia del sistema propuesto radica en su enfoque para optimizar la interpretación de los datos obtenidos, aspecto de vital importancia en diversos ámbitos de aplicación. A través de esta investigación, se busca proporcionar soluciones concretas y eficientes que mejoren significativamente la interpretación de la información adquirida en las tuberías de transporte de petróleo [1].

El desarrollo de este sistema no solo se centra en la adquisición y almacenamiento de los datos, sino también en la interpretación y análisis mediante dispositivos como el módulo HC05, ESP32 para el procesamiento digital. Se emplean algoritmos y metodologías eficientes para procesar y extraer información relevante de los datos adquiridos, lo que permitirá obtener resultados precisos y confiables.

Además, se presta especial atención al diseño de una arquitectura robusta y escalable que facilite la integración del sistema en un computador portátil. Esto permite que el sistema sea portátil y accesible en diversos entornos y situaciones, lo cual es fundamental para su aplicabilidad práctica.

1.1 Objetivos

1.1.2 Objetivo General

Desarrollar e implementar un prototipo de sistema para la recepción de datos, el procesamiento digital de la información en una computadora portátil con el fin de interpretar los datos recibidos, realizar una correlación matemática para determinar el tipo de perturbación y mostrar los resultados a través de una interfaz hombre-máquina.

1.1.3 Objetivos Específicos

- Realizar una revisión bibliográfica acerca de las técnicas procesamiento digital de señales acústicas y filtrado de señal para una señal acústica.
- Diseñar e implementar un sistema para la recepción, procesamiento y presentación de la información obtenida por el prototipo.
- Implementar un algoritmo para la interpretación de los datos recibidos y con ello realizar una correlación matemática para determinar el tipo de perturbación.
- Diseñar e implementar un interface hombre-máquina en el cual se muestre el tipo de perturbación detectado.
- Realizar pruebas y correcciones que permitan validar el correcto funcionamiento del sistema de recepción de datos implementado.

1.2 Alcance

Se realizará una revisión bibliográfica acerca de las técnicas procesamiento digital de señales acústicas y filtrado de señal para una señal acústica.

Se realizará una revisión bibliográfica sobre las técnicas de transmisión y los algoritmos necesarios para la recepción e interpretación de datos provenientes de un sensor de emisión acústica.

Se realizará una revisión bibliográfica sobre las técnicas y algoritmos que permitan determinar e identificar el tipo de perturbación.

Se realizará el dimensionamiento y selección de los equipos para la transmisión, recepción y procesamiento de los datos provenientes del sensor de emisión acústica.

Se diseñará e implementará un sistema para la recepción, procesamiento y presentación de la información obtenida por el sensor de emisión acústica.

Se implementará el algoritmo que permita determinar el tipo de perturbación detectado en una tubería metálica que puede ser parte de un oleoducto.

Se implementará el interface Hombre-maquina en el cual se muestre la señal del sensor de emisión acústica en tiempo real.

Se implementará el interface Hombre-maquina en el cual se muestre las alarmas correspondientes al tipo de perturbación.

Se validará del prototipo realizando mediciones y comparando los resultados según los requerimientos de operación.

1.3 Marco Teórico

1.3.1 PROCESAMIENTO DIGITAL DE LA INFORMACIÓN

El procesamiento digital de la información juega un papel fundamental en la interpretación y análisis de los datos adquiridos. En este sentido, se exploran los fundamentos del procesamiento digital de señales y de datos.

Fundamentos del procesamiento digital de datos:

- Tecnologías de Comunicación y Protocolos: Son métodos y normas para la transmisión de datos entre dispositivos, como Bluetooth, Wi-Fi y USB [1].
- Procesadores: Componentes que ejecutan operaciones, con velocidad de reloj, cantidad de núcleos y arquitectura [1].
- Memoria: Es el espacio de almacenamiento temporal con cantidad de RAM y velocidad de transferencia [1].
- Periféricos: Son dispositivos auxiliares como impresoras y cámaras, con compatibilidad [1].
- Unidades de Almacenamiento: Son dispositivos de almacenamiento con capacidad y velocidad de acceso a los datos [1].
- Bluetooth: Es la tecnología inalámbrica para transmisión de datos a corta distancia con compatibilidad [1].
- USB: Es lo estándar para la conexión de dispositivos electrónicos, con versiones y compatibilidad [1].
- Protocolos de Transmisión de Datos: Son reglas y formatos para la transmisión segura y eficiente de información, incluyendo protocolos criptográficos y de detección y corrección [1].

En la imagen de la figura 1.1 se describe el proceso de transmisión y recepción de datos:

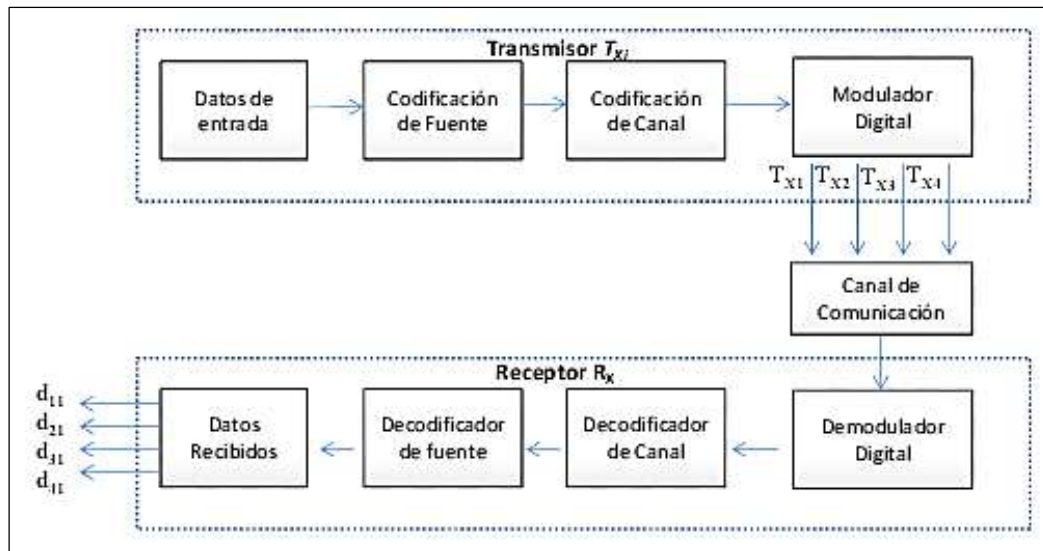


Figura 1.1. Proceso de transmisión y recepción de datos [2].

1.3.2 TÉCNICAS DE PROCESAMIENTO DIGITAL DE SEÑALES ACÚSTICAS

- 1) **Ruido Ambiental:** Las señales acústicas, al ser sensibles a las variaciones en la presión acústica, son propensas a capturar ruido ambiental. Este ruido puede originarse en diversas fuentes, como interferencias electromagnéticas generadas por otros dispositivos electrónicos en el entorno [1]. Estas interferencias introducen fluctuaciones en el nivel de presión acústica detectado por los sensores de emisión acústica, afectando la precisión de las mediciones. Además, vibraciones mecánicas y el ruido térmico también contribuyen al ruido ambiental, incrementando la complejidad de distinguir las señales relevantes. La presencia de este ruido degrada la relación señal-ruido y puede requerir técnicas de filtrado y procesamiento para mitigar su impacto y extraer la información útil con mayor precisión.
- 2) **Reflexiones Múltiples:** Las señales acústicas experimentan reflexiones múltiples cuando se encuentran con interfaces o estructuras dentro del medio. Estas reflexiones generan ecos y reverberaciones en la señal, lo que dificulta la identificación y separación de los componentes individuales. La superposición de las señales reflejadas complica la tarea de discernir la señal original de sus reflexiones, lo que puede llevar a interpretaciones erróneas [1]. La existencia de múltiples caminos de propagación para las ondas acústicas introduce complejidad en el análisis y requiere técnicas avanzadas de procesamiento para desentrañar los componentes de la señal original.

- 3) **Atenuación y Dispersión:** A medida que las ondas acústicas se propagan en un medio, enfrentan atenuación y dispersión. La atenuación conlleva a una pérdida gradual de energía a medida que la señal atraviesa el medio, lo que resulta en una disminución de la amplitud de la señal a medida que avanza la distancia. La dispersión provoca cambios en la velocidad de propagación de diferentes frecuencias, lo que lleva a la separación de los componentes de frecuencia de la señal. Estas pérdidas de energía y cambios en la distribución de frecuencias pueden afectar la integridad y la forma de la señal acústica, exigiendo consideraciones sobre la distancia de propagación y técnicas de corrección [1].
- 4) **Problemas de Transducción:** Los sensores acústicos transforman las ondas sonoras en señales eléctricas. Durante este proceso de transducción, pueden surgir imperfecciones que introduzcan artefactos o distorsiones en las señales resultantes. Estas distorsiones pueden manifestarse como cambios en la amplitud, fase o forma de la señal eléctrica, alterando la información que se obtiene de las ondas acústicas [1]. La calidad y precisión de la transducción son aspectos cruciales para evitar introducir errores adicionales en la señal.
- 5) **Propiedades del Medio:** Las propiedades físicas del medio por el cual se propagan las ondas acústicas, como la densidad y la velocidad del sonido, afectan la velocidad de propagación y la atenuación de las ondas. Cambios en estas propiedades pueden introducir variaciones no deseadas en la señal acústica. Por ejemplo, una variación en la densidad puede modificar el índice de refracción y, por ende, afectar el ángulo de propagación de la onda. La comprensión de estas propiedades es esencial para interpretar con precisión las mediciones acústicas y corregir posibles distorsiones causadas por el medio [1].

Algunas de las técnicas más comunes utilizadas en el procesamiento digital de señales acústicas son:

1.3.2.1 Filtrado

Se aplican diversos tipos de filtros para eliminar el ruido y las interferencias no deseadas de la señal. Los filtros pueden ser de diferentes tipos, como filtros pasa bajos, pasa altos, o filtros adaptativos que se ajustan automáticamente a las características de la señal [4].

Filtros pasa bajos

Los filtros pasa bajos permiten el paso de las frecuencias más bajas de la señal, atenuando o eliminando las frecuencias más altas. Son útiles para eliminar el ruido de

alta frecuencia que pueda estar presente en las señales acústicas, lo que facilita la detección de eventos de interés [4].

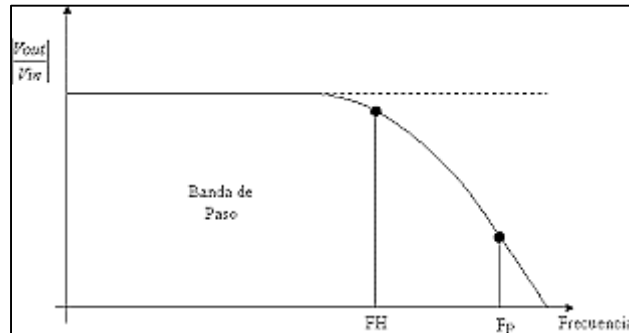


Figura 1.2. Señal de un filtro pasa bajos [4].

Estos filtros actúan como un recurso selectivo que permite el paso de las componentes de frecuencia más bajas, mientras que atenúa las componentes de frecuencia más altas. La función matemática que define la operación de un filtro pasa bajos ideal se expresa de la siguiente manera:

$$H_{LP}(\omega) = \begin{cases} 1, & \text{si } \omega \leq \omega_c \\ 0, & \text{si } \omega > \omega_c \end{cases} \quad (1.1)$$

Donde:

$H_{LP}(\omega)$: es la función de transferencia del filtro pasa bajos.

ω : es la frecuencia angular de la señal de entrada.

ω_c : es la frecuencia de corte, que separa las frecuencias que pasan de las que son atenuadas.

En el monitoreo de tuberías de petróleo y otras tuberías metálicas, estos filtros se utilizan para analizar y enfocarse en las perturbaciones o golpes relevantes, mientras se eliminan las fluctuaciones rápidas o el ruido de alta frecuencia que no son de interés.

Cuando ocurre una perturbación en la tubería, como un golpe o una vibración, se genera una señal que contiene componentes de varias frecuencias. Al aplicar un filtro pasa bajos, se resaltan las partes de la señal asociadas con cambios más lentos o eventos importantes, como los golpes en la tubería, al tiempo que se atenúan las oscilaciones rápidas que podrían no ser relevantes para el análisis [4].

Un ejemplo de filtro pasa bajos podría ser el filtro de media móvil. Este filtro calcula el promedio de los valores de la señal en un intervalo de tiempo determinado. Al hacerlo,

suaviza la señal y elimina las fluctuaciones de alta frecuencia. Esto permite identificar con mayor claridad los momentos en que ocurren los golpes o perturbaciones en la tubería.

Filtros pasa altos

Los filtros pasa altos atenúan o eliminan las frecuencias más bajas de la señal, permitiendo el paso de las frecuencias más altas. Estos filtros son efectivos para eliminar el ruido de baja frecuencia o interferencias que puedan afectar la medición de eventos importantes [4].

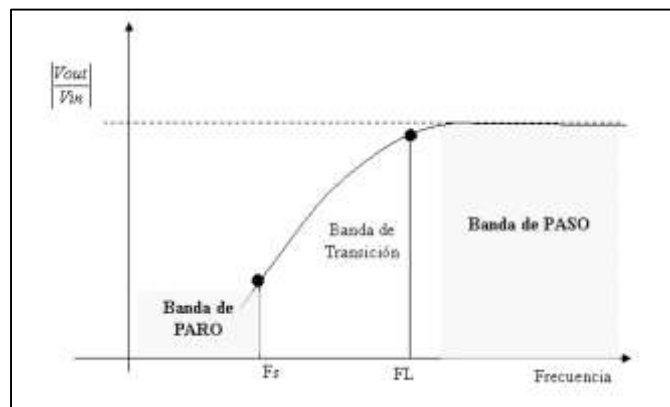


Figura 1.3. Señal de un filtro pasa altos [4].

La función matemática que define la operación de un filtro pasa altos ideal se expresa de la siguiente manera:

$$H_{HP}(\omega) = \begin{cases} 0, & \text{si } \omega \leq \omega_c \\ 1, & \text{si } \omega > \omega_c \end{cases} \quad (1.2)$$

Donde:

$H_{HP}(\omega)$: es la función de transferencia del filtro pasa altos.

ω : es la frecuencia angular de la señal de entrada.

ω_c : es la frecuencia de corte, que separa las frecuencias que pasan de las que son atenuadas.

Filtros adaptativos

Los filtros adaptativos desempeñan un papel avanzado al ajustar sus coeficientes automáticamente en función de las cambiantes características de la señal. Estos filtros se adaptan en tiempo real a las condiciones del entorno y del sistema, lo que resulta en mediciones más precisas y una mayor eliminación del ruido indeseado [4].

La operación matemática fundamental de un filtro adaptativo se expresa mediante la ecuación:

$$w(n+1) = w(n) + \mu * e(n) * x(n) \quad (1.3)$$

En esta ecuación, "w" representa los coeficientes del filtro, "μ" es un parámetro de ajuste conocido como tasa de aprendizaje, "e" es el error entre la señal deseada y la señal filtrada, y "x" es la entrada de la señal. La ecuación muestra cómo los coeficientes del filtro se actualizan iterativamente en función del error y la entrada, lo que permite que el filtro se ajuste para minimizar el error y mejorar la calidad de la señal filtrada.

La función adaptativa se basa en principios matemáticos sólidos y algoritmos de optimización, como el algoritmo de gradiente estocástico. La tasa de aprendizaje "μ" determina la rapidez con la que los coeficientes se ajustan en función del error. Un valor adecuado de "μ" es esencial para lograr una convergencia eficiente y evitar oscilaciones no deseadas en la adaptación [4].

1.3.2.2 Detección de eventos

La detección de eventos implica el uso de algoritmos para identificar patrones o situaciones específicas en la señal de monitoreo, como picos inusuales, cambios bruscos o formas características [4]. Esto es valioso en la vigilancia de tuberías de transporte de petróleo o cualquier otra tubería metálica para detectar fugas u otras perturbaciones. Algunos algoritmos efectivos son:

Algoritmo de Umbral: Este método simple compara la señal con un valor límite previamente establecido. Si la señal supera ese límite, se considera que ha ocurrido un evento relevante, como una fuga. Esta lógica se expresa como:

$$Evento = \begin{cases} Verdadero, & \text{si } x(t) > Umbral \\ Falso & \text{si } x(t) \leq Umbral \end{cases} \quad (1.4)$$

Algoritmo de Derivada: Este enfoque se basa en calcular la tasa de cambio de la señal a lo largo del tiempo. Cambios abruptos en la señal, como los causados por una fuga, resultarán en picos en la derivada de la señal. Esto permite detectar dichos eventos de esta manera:

$$Evento = \begin{cases} Verdadero, & \text{si } \frac{dx(t)}{dt} > Umbral \text{ de derivada} \\ Falso & \text{si } \frac{dx(t)}{dt} \leq Umbral \text{ de derivada} \end{cases} \quad (1.5)$$

El Algoritmo de Transformada de Hilbert: Se basa en la Transformada de Hilbert, una herramienta matemática que permite calcular la envolvente instantánea de una señal. Esta técnica es valiosa para identificar cambios en la amplitud de la señal, como picos, que pueden ser indicativos de eventos relevantes [4]. La Transformada de Hilbert se expresa mediante la siguiente ecuación:

$$Envolvente = |H\{x(t)\}| \quad (1.6)$$

El Algoritmo de Correlación: Es una técnica que busca similitudes entre dos señales. En este caso, se compara la señal adquirida ($x(t)$) con una señal de referencia ($y(t)$) para identificar patrones característicos [4]. Esta comparación se realiza en función del tiempo y permite detectar eventos específicos en la señal acústica, como fugas o cambios en el flujo del petróleo. La correlación se expresa mediante la siguiente ecuación:

$$Correlación = \int_{-\infty}^{+\infty} x(t) * y(t) dt \quad (1.7)$$

Los algoritmos de Machine Learning: Como el aprendizaje supervisado o no supervisado, ofrecen la capacidad de identificar patrones complejos en la señal que podrían señalar eventos particulares. Estos algoritmos tienen la capacidad de ser entrenados utilizando datos históricos, lo que potencia la exactitud de la detección [4]. Un enfoque de Machine Learning ampliamente utilizado es el aprendizaje supervisado, donde el algoritmo se entrena con ejemplos etiquetados para aprender a reconocer

patrones y relaciones en la señal. Por otro lado, el aprendizaje no supervisado busca descubrir patrones intrínsecos en los datos sin la guía de etiquetas predefinidas.

$$Salida = \sigma\left(\sum_{i=1}^n w * xi + b\right) \quad (1.8)$$

1.3.3 PROBLEMA MAESTRO- ESCLAVO

En este modelo, un dispositivo o entidad, llamado "maestro", se encarga de controlar y coordinar uno o más dispositivos o entidades denominadas "esclavos". El maestro actúa como la entidad central que supervisa y dirige las acciones de los esclavos, quienes ejecutan tareas específicas y responden a las solicitudes del maestro [7].

1.3.3.1 Comunicación Serial

La comunicación serial es un método de transferencia de datos entre dispositivos electrónicos mediante la transmisión de bits uno después del otro, a través de un solo canal de comunicación. Es un tipo de comunicación punto a punto, lo que significa que los datos se envían desde un transmisor a un receptor específico [7].

Hay dos modos principales de comunicación serial: el modo asíncrono y el modo síncrono.

1) Modo Asíncrono

Los dispositivos no comparten una señal de reloj común para sincronizar la transmisión. En su lugar, se envía un bit de inicio (start bit) al comienzo de cada byte o carácter para indicar el inicio de una transmisión. Luego, los bits de datos se transmiten uno tras otro, seguidos opcionalmente por bits de paridad y bits de parada (stop bits) para delimitar el byte o carácter. La falta de una señal de reloj común hace que este método sea sencillo y flexible [7].

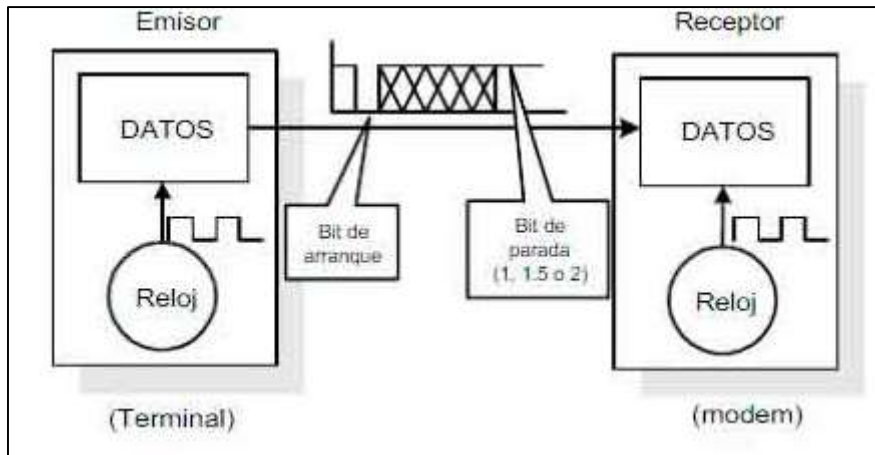


Figura 1.4. Transmisión Asíncrona.

2) Modo Síncrono

Los dispositivos comparten una señal de reloj común para sincronizar la transmisión de datos. Los bits de datos se transmiten en intervalos regulares determinados por el reloj compartido. Esto asegura una sincronización más precisa y permite velocidades de transferencia más altas y mayores distancias de comunicación [7].

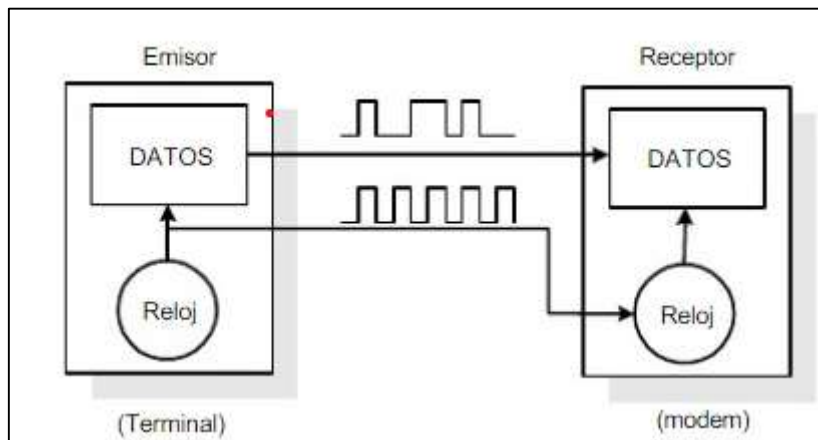


Figura 1.5. Transmisión Síncrona.

1.3.3.2 Protocolos de Comunicación Serial

La comunicación serial también se rige por protocolos específicos que definen las reglas y el formato de la transmisión de datos. Algunos protocolos comunes para la comunicación serial se describen en la tabla 1.1:

Tabla 1.1. Protocolo Comunicación Serial [8]

Protocolo	Ventajas	Desventajas	Aplicaciones
RS-232	Simple, corto alcance, ampliamente soportado	Limitado en distancia, susceptible al ruido	Consolas, periféricos, módems
RS-485	Largo alcance, resistente al ruido	Terminación de línea, mayor complejidad	Sensores industriales, automatización, acceso
UART	Simple, ampliamente usado, bajo consumo	Menor velocidad que protocolos síncronos	Microcontroladores, periféricos
SPI	Alta velocidad, punto a punto o multidrop	Más líneas necesarias, longitud limitada	Sensores, pantallas, memorias
I2C	Solo dos líneas, múltiples dispositivos	Velocidad más baja, distancia limitada	Placas de circuito, interconexión de chips
CAN	Alta inmunidad al ruido, detección errores	Requiere controlador CAN, mayor complejidad	Automoción, sistemas industriales, robótica

- **RS-232 (Recommended Standard 232):**

Es un protocolo de comunicación asincrónico ampliamente utilizado para transmitir datos entre dispositivos a corta distancia. Es común en la conexión de consolas de computadoras y dispositivos periféricos [8].

- **RS-485 (Recommended Standard 485):**

Es un protocolo de comunicación tanto síncrono como asincrónico que permite la transmisión de datos a largas distancias y la conexión de múltiples dispositivos en un mismo bus [8].

- **UART (Universal Asynchronous Receiver-Transmitter):**

Es un componente que facilita la comunicación asincrónica entre dispositivos, como microcontroladores y periféricos [8].

- **SPI (Serial Peripheral Interface):**

Es un protocolo síncrono que permite la transmisión de datos a alta velocidad entre un dispositivo maestro y varios dispositivos esclavos [8].

- **I2C (Inter-Integrated Circuit):**

Es un protocolo síncrono que utiliza solo dos líneas de comunicación para conectar múltiples dispositivos en un bus [8].

- **CAN (Controller Area Network):**

Es un protocolo síncrono con alta inmunidad al ruido, diseñado para aplicaciones en entornos industriales y automotrices [8].

Para este trabajo se utilizará el protocolo UART, opera mediante la transmisión y recepción de datos en forma de bits, donde no se requiere una señal de reloj compartida entre los dispositivos. En su lugar, se utiliza un bit de inicio (start bit) al comienzo de cada byte para indicar el inicio de la transmisión. Luego, los bits de datos se transmiten uno tras otro, seguidos por un bit de paridad (opcional) para verificar la integridad de los datos y uno o más bits de parada (stop bits) que delimitan el byte [8].

Ventajas del Protocolo UART:

- Sencillez: La implementación del protocolo UART es relativamente sencilla, lo que lo hace adecuado para aplicaciones con recursos limitados [8].
- Compatibilidad: Es un protocolo estándar que se encuentra en una amplia gama de microcontroladores y dispositivos, lo que facilita la interconexión entre diferentes componentes [8].
- Asincronía: No se requiere una señal de reloj compartida, lo que permite la comunicación entre dispositivos que pueden tener relojes ligeramente diferentes [8].
- Flexibilidad: Es posible establecer comunicación punto a punto (uno a uno) o punto a multipunto (uno a varios) [8].

1.3.3.3 Comunicación Serial Maestro-Esclavo

La comunicación serial entre un dispositivo maestro y dispositivos esclavos representa un enfoque específico de interacción serial, en el cual un dispositivo central (maestro) tiene el control y coordinación de uno o varios dispositivos secundarios (esclavos) [8].

En este contexto, el rol del maestro implica la iniciación y dirección de la comunicación, mientras que los esclavos responden a las solicitudes del maestro y ejecutan las acciones necesarias. Por lo general, esta comunicación se lleva a cabo mediante dos líneas de comunicación, que pueden ser unidireccionales o bidireccionales. Una de estas líneas permite al maestro enviar datos a los esclavos (línea de transmisión), mientras que la otra facilita la recepción de datos de los esclavos por parte del maestro (línea de recepción) [7].

Los elementos fundamentales en la comunicación serial maestro-esclavo incluyen:

- 1) Protocolo de Comunicación: Para lograr una comunicación eficiente entre el maestro y los esclavos, se emplea un protocolo serial específico. Entre estos protocolos comunes se encuentran RS-232, RS-485, UART, SPI, I2C y CAN, entre otros. La elección del protocolo depende de los requerimientos particulares de la aplicación, tales como la distancia de transmisión, la velocidad requerida y el número de esclavos involucrados [8].
- 2) Identificación de los Esclavos: Cada esclavo en la red debe poseer una dirección única, permitiendo al maestro distinguir y dirigirse a cada dispositivo de forma individual. Esta dirección se utiliza para identificar la respuesta del esclavo y asegurar la correcta entrega de los datos [8].
- 3) Mensajes y Comandos: El maestro emite mensajes o comandos a los esclavos para solicitar información o dirigir su ejecución en tareas específicas. Estos mensajes suelen adherirse a una estructura definida por el protocolo empleado y contienen información como instrucciones, datos, códigos de operación y direcciones [8].
- 4) Respuestas de los Esclavos: Los esclavos reciben los mensajes del maestro y responden con la información o datos solicitados. Estas respuestas también se ajustan a una estructura predefinida por el protocolo. Dependiendo del protocolo, los esclavos pueden transmitir datos de manera espontánea o exclusivamente en respuesta a solicitudes del maestro [8].
- 5) Sincronización y Velocidad de Transmisión: La sincronización entre el maestro y los esclavos es de vital importancia para garantizar una comunicación efectiva. La velocidad de transmisión (baud rate) y los parámetros temporales deben ser configurados adecuadamente para asegurar una transferencia de datos confiable y libre de errores [8].

1.3.4 MÓDULO HC05

El módulo HC05 es un módulo Bluetooth que permite establecer comunicación inalámbrica entre dispositivos a corta distancia. Es ampliamente utilizado en proyectos de electrónica y robótica debido a su facilidad de uso y bajo costo.

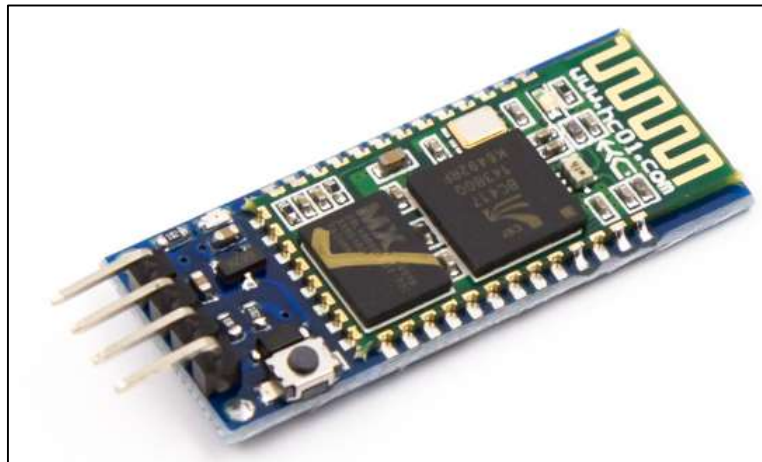


Figura 1.6. Módulo HC05 [9].

Descripción técnica del módulo:

Tabla 1.2. Descripción Técnica del módulo HC05.

Pin	Nombre	Descripción
1	VCC	Alimentación (3.6V - 6V)
2	GND	Tierra (GND)
3	TX	Transmisión UART (Bluetooth al host)
4	RX	Recepción UART (Host al Bluetooth)
5	EN	Enable (Habilitar)
6	STATE	Estado (Conexión Bluetooth)

Características principales del módulo:

- Protocolo Bluetooth: El HC05 utiliza el estándar Bluetooth para establecer una conexión inalámbrica con otros dispositivos compatibles con Bluetooth, como computadoras, smartphones o microcontroladores [9].
- Modos de Operación: El módulo HC05 puede funcionar en diferentes modos, como maestro o esclavo. En el modo maestro, puede iniciar la conexión con otros dispositivos, mientras que en el modo esclavo espera a que otro dispositivo inicie la conexión [9].
- Interfaz Serial: La comunicación con el módulo HC05 se realiza mediante una interfaz serial UART (Universal Asynchronous Receiver/Transmitter), lo que facilita la integración con microcontroladores y otros dispositivos que cuenten con esta interfaz [9].

Proceso de conexión con una computadora:

- Configuración del módulo HC05: Antes de iniciar la conexión, es necesario configurar el módulo HC05 con los parámetros adecuados. Esto implica definir la velocidad de comunicación (baud rate), el modo de operación (maestro o esclavo), el nombre del dispositivo y la clave de acceso, entre otros. La configuración se realiza mediante comandos AT enviados al módulo a través de la interfaz serial [9].
- Emparejamiento con la computadora: Una vez configurado, el módulo HC05 debe emparejarse con la computadora. Esto implica buscar el módulo desde la computadora y proporcionar la clave de acceso previamente configurada. Una vez establecido el emparejamiento, la computadora reconocerá al módulo HC05 y permitirá la comunicación [9].
- Establecimiento de la conexión: Una vez emparejado, se puede establecer la conexión entre el módulo HC05 y la computadora. El módulo actuará como maestro y la computadora como esclava en esta configuración. La comunicación entre ambos dispositivos se realizará a través de la interfaz serial UART [9].
- Transmisión de datos: Con la conexión establecida, el módulo HC05 puede enviar los datos adquiridos por el sensor de emisión acústica a la computadora. Estos datos serán recibidos por un programa o aplicación en la computadora que realizará el procesamiento y análisis en tiempo real [9].

Ventajas del uso del módulo HC05:

- Facilidad de integración: El módulo HC05 es fácil de integrar en proyectos debido a su interfaz serial y su tamaño compacto.
- Conexión inalámbrica: Permite la transmisión de datos sin necesidad de cables, lo que facilita la implementación del sistema de monitoreo en línea en tuberías metálicas.
- Amplia compatibilidad: Es compatible con una amplia gama de dispositivos que soportan el estándar Bluetooth, incluyendo computadoras, smartphones y microcontroladores.

1.3.5 MÓDULO ESP32

El ESP32 es un microcontrolador de la compañía Espressif Systems. Posee dos núcleos (dual-core) con capacidad para conectividad Wi-Fi y Bluetooth integrada.



Figura 1.7. Módulo ESP32 [9].

Descripción técnica del módulo:

Tabla 1.3. Descripción técnica del módulo.

Pin	Nombre	Descripción
1	3V3	Alimentación de 3.3V

Pin	Nombre	Descripción
2	GND	Tierra (GND)
3	EN	Habilitar
19	5V	Alimentación de 5V (pin VIN)
27	3.3V	Alimentación de 3.3V
28	EN	Habilitar
29	3.3V	Alimentación de 3.3V
30	GND	Tierra (GND)

Las características técnicas del módulo ESP32, se describen en la sección de ANEXOS, Anexo No.2: Pines del Módulo ESP32

1.3.5.1 Comunicación serial con el módulo ESP32

La comunicación serial con el módulo ESP32 se refiere a la capacidad de enviar y recibir datos entre el ESP32 y otros dispositivos utilizando el protocolo UART (Universal Asynchronous Receiver/Transmitter). El ESP32 cuenta con pines GPIO que pueden configurarse como puertos serie UART para facilitar esta comunicación.

En la comunicación UART, se utilizan dos pines específicos: TX (Transmitter) y RX (Receiver). A través del pin TX, el ESP32 envía datos a otros dispositivos, mientras que el pin RX le permite recibir datos de otros dispositivos. Para establecer la comunicación serial, es necesario conectar el pin TX del ESP32 al pin RX del dispositivo externo con el que se desea comunicar, y viceversa, conectar el pin RX del ESP32 al pin TX del dispositivo externo. También se debe conectar la referencia de tierra (GND) compartida entre ambos dispositivos.

Una vez establecida la conexión física, el siguiente paso es configurar el ESP32 para habilitar y configurar los puertos serie UART mediante software. Esto implica establecer la velocidad de transmisión (baud rate) y otros parámetros necesarios para una comunicación efectiva.

El ESP32 ofrece ventajas significativas para esta comunicación serial debido a su capacidad de procesamiento, conectividad Wi-Fi y Bluetooth integrada, así como su amplia cantidad de pines GPIO. Además, es posible utilizar la biblioteca SoftwareSerial para crear puertos serie virtuales en pines GPIO adicionales si es necesario.

La comunicación serial UART es ampliamente utilizada en una variedad de proyectos de electrónica y microcontroladores, facilita la transmisión de datos en tiempo real y la interacción con diversos periféricos. Gracias a la flexibilidad y potencia del ESP32, este microcontrolador es una elección popular para implementar comunicaciones UART en proyectos que requieren conectividad inalámbrica y control sofisticado.

1.3.6 ROUTER

El router inalámbrico Huawei Hg 530 es una solución avanzada para la distribución confiable y eficiente de conexiones a Internet a través de redes WiFi. Diseñado con características de alto rendimiento y seguridad.

Especificaciones Técnicas

- Estándares de WiFi: Cumple con los estándares 802.11ac/n/g/b, lo que permite una compatibilidad amplia con dispositivos WiFi-actuales y anteriores.
- Velocidad Máxima: Ofrece una velocidad de hasta 1200 Mbps a través de sus frecuencias de 2.4 GHz y 5 GHz, permitiendo la transmisión de datos fluida y rápida.
- Antenas: Equipado con 4 antenas externas de alto rendimiento que mejoran la cobertura y la calidad de la señal en toda la casa u oficina.
- Puertos Ethernet Gigabit: Dispone de 4 puertos LAN y 1 puerto WAN Ethernet Gigabit para conexiones por cable de alta velocidad.
- Seguridad Avanzada: Incorpora protocolos de seguridad WPA3, WPA2 y WEP para proteger la red contra amenazas cibernéticas y asegurar la privacidad de los usuarios.
- Funciones Adicionales: Incluye funciones como QoS (Quality of Service) para priorizar el tráfico, WPS para configuración rápida, Beamforming para dirigir la señal hacia dispositivos específicos y control parental para administrar el acceso a Internet.
- Potente Procesador: Equipado con un procesador dual-core de 1.2 GHz para manejar múltiples dispositivos y tareas simultáneas sin degradar el rendimiento.
- Amplia Memoria: Cuenta con 256 MB de RAM DDR3 y 128 MB de memoria Flash NAND para almacenamiento de datos y ejecución eficiente de procesos.
- Gestión y Configuración: Ofrece una interfaz de configuración basada en navegador web, lo que facilita la personalización de la red y las opciones de seguridad.

- Alimentación Estable: Viene con un adaptador de corriente de 12V y 2A para asegurar una fuente de energía constante y estable.

Ventajas

- Alto Rendimiento: Una velocidad de hasta 1200 Mbps y antenas externas aseguran una conexión estable y rápida en todas las áreas. Esto es especialmente beneficioso en entornos con alta demanda de ancho de banda, como hogares con múltiples dispositivos conectados o en oficinas donde se realizan tareas que requieren transferencias de datos intensivas.
- Seguridad Mejorada: La compatibilidad con estándares de seguridad modernos (como WPA3) garantiza que la red esté protegida contra amenazas cibernéticas. Esta seguridad mejorada es esencial para salvaguardar la privacidad de los usuarios y sus datos, lo que se vuelve especialmente crítico en la actualidad debido al aumento de los riesgos de seguridad en línea.
- Gestión Eficiente: Las características como la Calidad de Servicio (QoS) permiten priorizar el ancho de banda para aplicaciones críticas, como transmisión de video o juegos en línea, lo que mejora la experiencia del usuario. El control parental ofrece a los padres la capacidad de administrar y supervisar el acceso a Internet de sus hijos, lo que garantiza un uso responsable y seguro.
- Conectividad Versátil: Los puertos Ethernet Gigabit brindan la opción de conexiones por cable para dispositivos que requieren una conexión rápida y confiable, como consolas de juegos, computadoras de escritorio o dispositivos de almacenamiento en red. Esto asegura que todos los dispositivos, independientemente de si son inalámbricos o cableados, puedan aprovechar al máximo el rendimiento de la red.
- Compatibilidad Universal: El soporte de estándares anteriores asegura que dispositivos más antiguos que aún utilizan tecnologías Wi-Fi más antiguas puedan conectarse sin problemas. Esto es útil para garantizar que todos los dispositivos en la red puedan conectarse y comunicarse de manera efectiva.
- Configuración Sencilla: Una interfaz de configuración basada en navegador simplifica la instalación y personalización del router. Esto es especialmente útil para usuarios que pueden no tener experiencia técnica avanzada, lo que les permite poner en marcha la red de manera rápida y sin complicaciones.



Figura 1.8.Router inalámbrico

Nombre: Home Gateway

Modelo: HG 530

Tabla 1.4. Características técnicas del router

Característica	Especificación
Modelo	XYZ-Router 2000
Estándares de WiFi	802.11ac/n/g/b
Velocidad máxima WiFi	Hasta 1200 Mbps (2.4 GHz y 5 GHz)
Banda de frecuencia WiFi	2.4 GHz y 5 GHz
Antenas	4 antenas externas
Puertos LAN	4 puertos Ethernet Gigabit
Puerto WAN	1 puerto Ethernet Gigabit
Seguridad WiFi	WPA3, WPA2, WPA, WEP
Características adicionales	QoS, WPS, Beamforming, Control parental
Procesador	Dual-core a 1.2 GHz
Memoria RAM	256 MB DDR3

Característica	Especificación
Memoria Flash	128 MB NAND Flash
Interfaz de configuración	Web browser basada en GUI
Gestión remota	Sí (a través de aplicación móvil)
Alimentación	Adaptador de corriente 12V, 2A

1.3.7 INTERFACE GRÁFICA

HMI (Interfaz Hombre-Máquina, por sus siglas en inglés) es una interfaz gráfica, un medio a través del cual los seres humanos interactúan con las máquinas y acceden a la información. En este contexto, se explorarán los principios de diseño de interfaces que permitan una interacción intuitiva y efectiva [3].

La HMI desempeña un papel crítico en diversos campos, desde la informática y la electrónica hasta la robótica y la industria. Su diseño adecuado es esencial para mejorar la usabilidad, la productividad y la experiencia del usuario.

1.3.7.1 Desarrollo de Interfaces en Matlab

El "Desarrollo de Interfaces en MATLAB" se trata de crear interfaces gráficas de usuario (GUI) utilizando el software MATLAB. Estas interfaces permiten a los usuarios interactuar visualmente con programas y algoritmos creados en MATLAB, sin necesidad de escribir código en la línea de comandos. Estas interfaces son útiles en diversas aplicaciones, como mostrar datos, controlar sistemas, procesar señales y realizar análisis numéricos [4]. Los usuarios pueden interactuar con elementos como botones, campos de entrada, gráficos y tablas, lo que hace más simple ingresar datos y comprender los resultados.

2. METODOLOGÍA

En este capítulo y acorde al capítulo 1 se describe en detalle el procesamiento digital de las señales detectadas por el sensor en tiempo real.

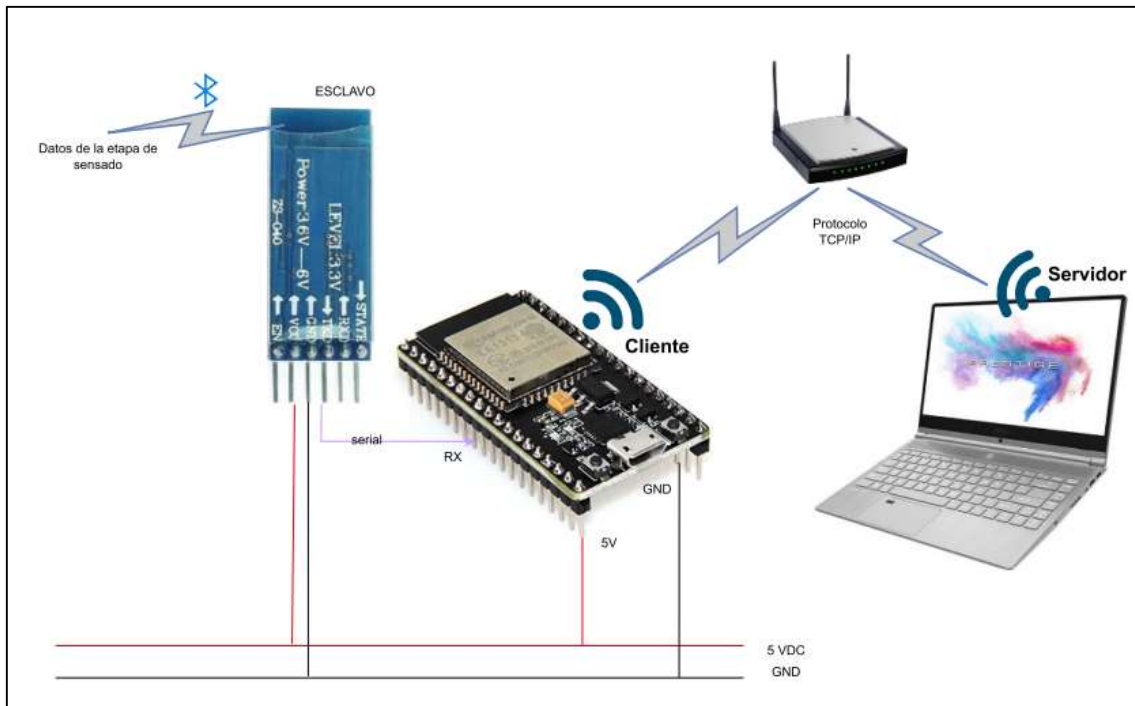


Figura 2.1. Hardware del procesamiento de datos.

2.1 DISEÑO DEL HARDWARE

2.1.1 RECEPCIÓN DE DATOS

Los datos recibidos son resultados del producto 2 del presente trabajo de titulación con nombre *“Diseño, desarrollo e implementación del sistema de sensado, del acondicionamiento de señales y del sistema de comunicación en línea y transmisión de datos entre el sistema de sensado y un computador portátil del prototipo para el monitoreo en línea de las tuberías de transporte petrolero utilizando un sistema acústico”*, las señales son recibidas a través del módulo HC05 en configuración esclavo.

El módulo HC-05 maestro desempeña el rol de iniciador en la comunicación, asumiendo la responsabilidad de enviar y recibir datos a través de una interfaz UART (Universal Asynchronous Receiver/Transmitter). Por otro lado, el módulo HC-05 esclavo opera como el receptor en esta relación, recibiendo los datos enviados por el módulo maestro.

La conexión entre ambos módulos se establece mediante un enlace Bluetooth, lo que permite una comunicación serial virtual inalámbrica entre los dispositivos. Este enlace inalámbrico se configura para permitir la transmisión de información en ambos sentidos, lo que facilita el intercambio de datos entre los dispositivos de manera bidireccional.

2.1.2 CONFIGURACIÓN MÓDULO HC05 MODO ESCLAVO

La alimentación para el esclavo que es el módulo HC05 es con $5V_{dc}$, en la imagen de la figura 9 se describe la conexión:

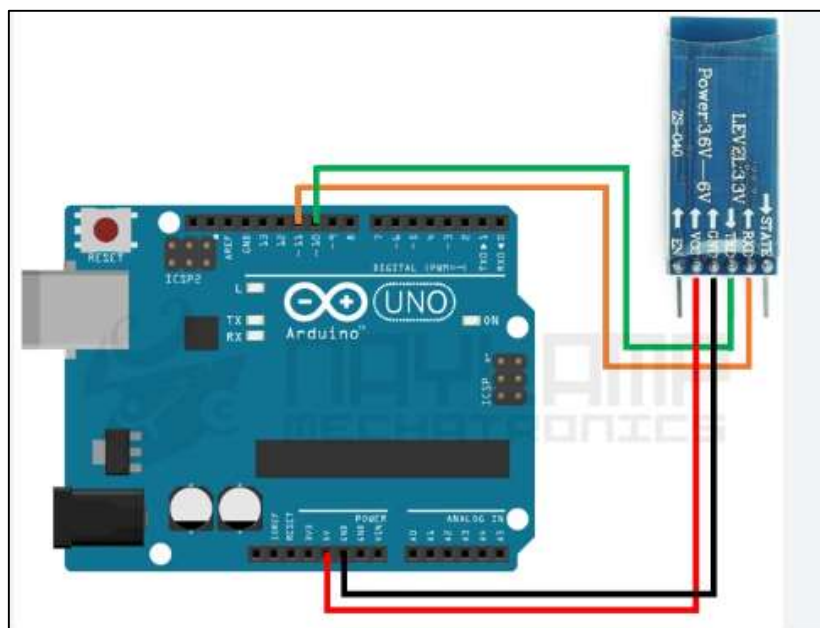


Figura 2.2. Conexión módulo HC05.

La comunicación serial se logra a través de las funciones de comunicación serial propias del entorno de desarrollo de Arduino, lo que facilita la conexión y la comunicación con otros dispositivos a través de la interfaz UART. EL HC05 recibe estos comandos por serial (El HC-05 debe estar conectado rx y tx con rx y tx del Arduino). Los comandos se describen en la siguiente tabla:

Tabla 2.1. Comandos configuración HC05 como esclavo.

Comando	Descripción
AT+NAME=Esclavo	Cambia nombre a "Esclavo".
AT+PSWD="esclavo4321"	Establece contraseña como "esclavo4321".
AT+ROLE=0	Configura modo esclavo (Slave).
AT+CMODE=0	Configura modo Clásico (sin autoconexión).
AT+UART=9600,0,0	Ajusta velocidad a 9600 baudios, sin paridad, 1 bit de parada.

Los comandos utilizados en el Arduino se describen a continuación:

- 1) Comando: AT+NAME=Esclavo
 - Descripción: Este comando se utiliza para cambiar el nombre del módulo HC-05. El nuevo nombre, en este caso "Esclavo", reemplazará el nombre anterior del módulo.
- 2) Comando: AT+PSWD="esclavo4321"
 - Descripción: Este comando permite configurar la contraseña (PIN) de acceso para el módulo HC-05. La contraseña se establece como "esclavo4321" en este caso. Esta contraseña se utilizará para autenticar la conexión con otros dispositivos Bluetooth que intenten conectarse al módulo.
- 3) Comando: AT+ROLE=0
 - Descripción: Este comando establece el rol del módulo HC-05. El valor "0" configura el módulo en modo esclavo (Slave). En este modo, el módulo actuará como un dispositivo pasivo que espera a ser emparejado y conectado por otro dispositivo Bluetooth en modo maestro.
- 4) Comando: AT+CMODE=0
 - Descripción: Con este comando, se configura el modo de conexión del módulo HC-05. El valor "0" establece el modo Clásico, lo que significa que el módulo no se conectará automáticamente a otros dispositivos Bluetooth. En lugar de eso, requerirá una solicitud de conexión desde un dispositivo emparejado.
- 5) Comando: AT+UART=9600,0,0
 - Descripción: Este comando se utiliza para configurar los parámetros de comunicación serial (UART) del módulo HC-05.
 - "9600": Establece la velocidad de transmisión en baudios en 9600. Es la velocidad a la que los bits de datos se transmiten o reciben por segundo.

- "0": Indica que no se usará ningún bit de paridad.
- "0": Configura 1 bit de parada. Indica el número de bits que se agrega después de cada carácter transmitido o recibido para marcar el final de la trama.

La Figura 2.3 muestra la configuración del módulo HC-05 en modo esclavo a través de un diagrama de flujo.

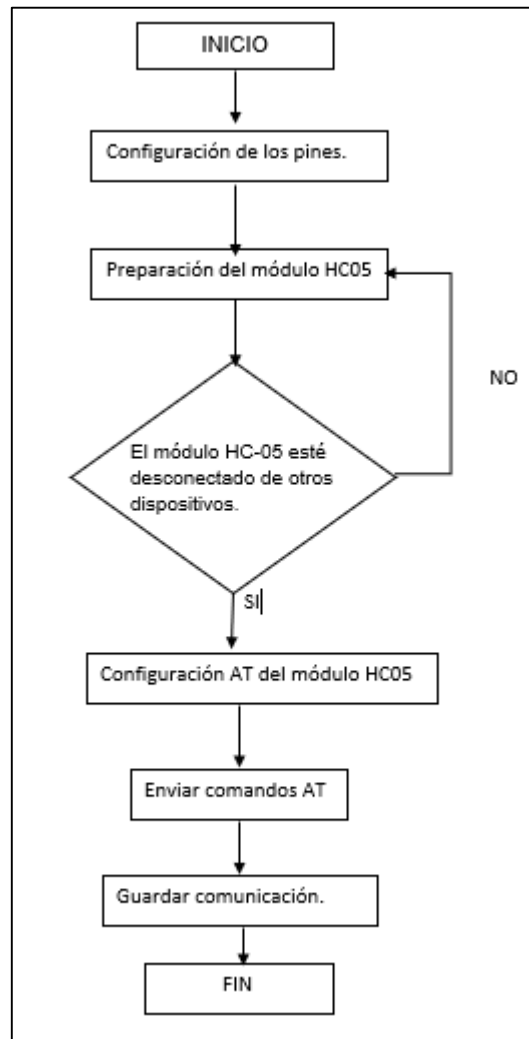


Figura 2.3. Configuración módulo HC05.

2.1.3 FILTRADO DE SEÑALES EN SENSORES DE EMISIÓN ACÚSTICA (ADAPTADO A LA NUEVA APROXIMACIÓN CON EL MÓDULO HC05)

En el contexto de la nueva aproximación que involucra el módulo HC05 en el procesamiento de señales acústicas, se han explorado diversas estrategias de filtrado para mejorar la calidad y precisión de las mediciones obtenidas de estos sensores [10]. Algunas de las investigaciones previas más destacadas en este ámbito son:

- **Filtrado Adaptativo:** Se han propuesto algoritmos de filtrado adaptativo que ajustan automáticamente los coeficientes del filtro en función de las condiciones cambiantes de la señal y el entorno. Esta técnica permite una adaptación eficiente a variaciones y perturbaciones en la señal, mejorando la calidad de las mediciones [10].
- **Transformada de Ondaletas:** La aplicación de transformadas de ondaletas ha demostrado ser eficaz para eliminar el ruido de fondo y mejorar la detección de eventos importantes en las señales acústicas. Esta técnica es especialmente útil para separar componentes de diferentes frecuencias y resaltar características relevantes [10].
- **Reducción de Ruido:** Se han desarrollado enfoques variados para reducir el ruido en señales acústicas. Estos incluyen la supresión de ruido impulsivo y la aplicación de filtros diseñados específicamente para atenuar componentes de baja frecuencia no deseados, mejorando la precisión en la detección de eventos de interés.

La incorporación del módulo HC05 agrega un nuevo aspecto a estas investigaciones al considerar las particularidades de la comunicación inalámbrica y su integración en el proceso de filtrado y procesamiento de señales acústicas. Esto amplía las posibilidades de mejora en la precisión y calidad de las mediciones obtenidas a partir del sensor de emisión acústica en este nuevo enfoque.

2.1.4 TÉCNICAS DE TRANSMISIÓN Y ALGORITMOS PARA RECEPCIÓN E INTERPRETACIÓN DE DATOS (EN EL CONTEXTO DE COMUNICACIÓN A TRAVÉS DEL MÓDULO HC05)

En el contexto de la comunicación a través del módulo HC05, es vital comprender las técnicas de transmisión y los algoritmos necesarios para la recepción e interpretación efectiva de datos. Estos aspectos juegan un papel esencial en asegurar la confiabilidad y eficiencia de la comunicación inalámbrica.

- **Protocolos de Comunicación:** Una de las consideraciones primordiales es la elección de un protocolo de comunicación apropiado. Este protocolo define cómo los datos son transmitidos entre el sistema de monitoreo y la computadora receptora que proporciona los datos de la señal. Pueden emplearse diversos protocolos, como Bluetooth o la comunicación serial. Esta elección depende de

factores como la distancia de transmisión, la velocidad de transferencia y las necesidades específicas del sistema [11].

- **Procesamiento en Tiempo Real:** Al recibir datos en tiempo real a través del módulo HC05, es esencial implementar algoritmos eficientes para procesar y analizar la señal acústica en tiempo real. Esto adquiere una relevancia crítica para identificar rápidamente eventos importantes, como fugas o perturbaciones en las tuberías metálicas. Los algoritmos aplicados en tiempo real permiten un monitoreo constante y la toma de decisiones inmediatas ante situaciones críticas [11].

La comprensión de las características técnicas del módulo HC05 también es esencial para asegurar un funcionamiento óptimo. Para obtener detalles más específicos sobre estas características, se recomienda revisar la sección de Anexos, en el Anexo No. 1: Pines del módulo HC05. Un entendimiento completo de estas características contribuye a optimizar la calidad y confiabilidad de la comunicación inalámbrica, especialmente en el contexto del monitoreo de tuberías de transporte petrolero o metálicas.

2.2 CONEXIÓN HC05 Y MÓDULO ESP32

Al inicio, el dispositivo maestro recibe datos enteros en el rango de 0 a 255, estos datos provienen del acondicionamiento y conversión análoga digital a una velocidad de transmisión de 9600 baudios, el dispositivo HC05 maestro se encuentra en el producto 2 del presente trabajo de titulación con *nombre "Diseño, desarrollo e implementación del sistema de sensado, del acondicionamiento de señales y del sistema de comunicación en línea y transmisión de datos entre el sistema de sensado y un computador portátil del prototipo para el monitoreo en línea de las tuberías de transporte petrolero utilizando un sistema acústico"*

El maestro HC05 envía los datos mediante la interfaz inalámbrica bluetooth al esclavo HC05; los datos recibidos por el módulo HC05 esclavo se transmiten mediante comunicación serial UART haciendo uso de los pines seriales del módulo ESP 32.

En la imagen de la figura 2.4 se muestra la conexión física entre el módulo HC05 (esclavo) y el módulo ESP32.

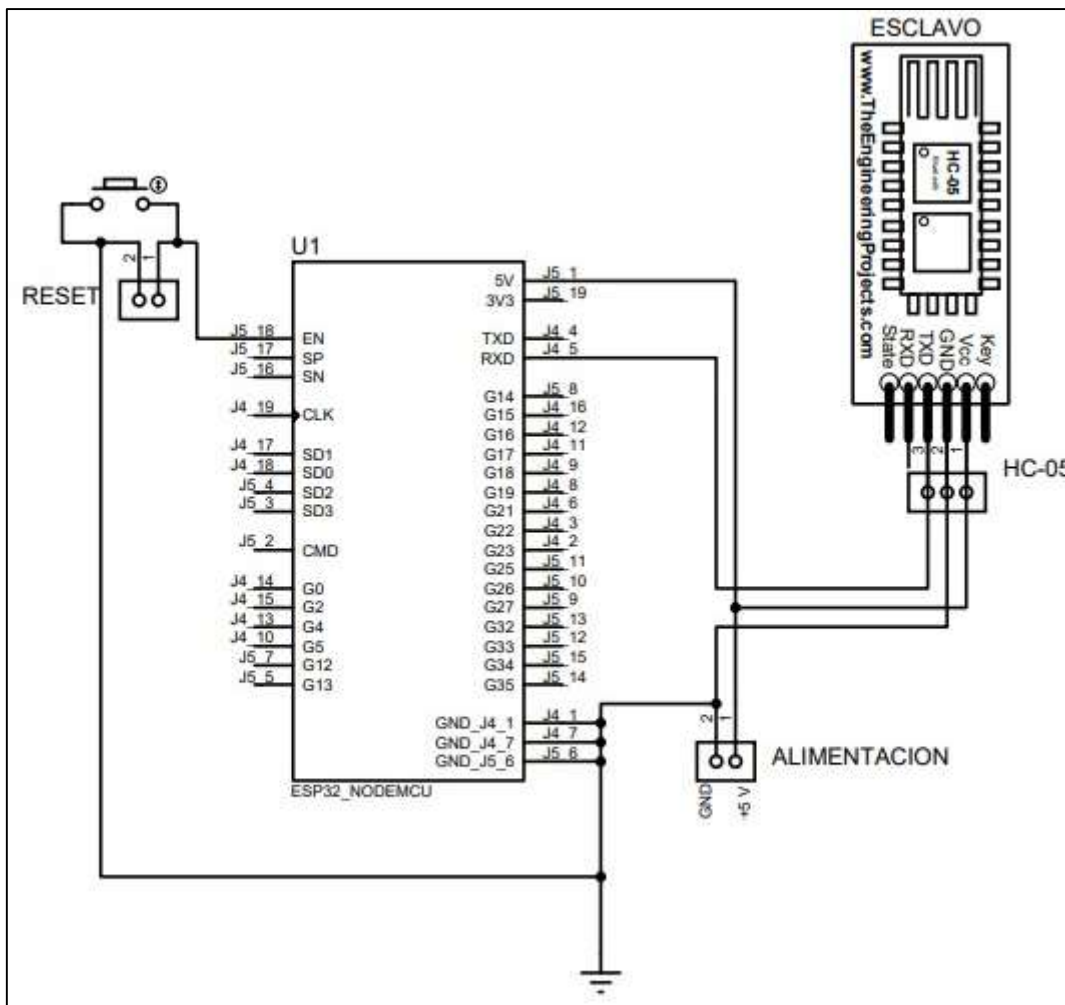


Figura 2.4. Conexión módulo HC05 y ESP32.

De la gráfica se describe el establecimiento de Conexión serial entre el Módulo HC-05 y el Módulo ESP32:

A continuación, se muestran los pasos que permiten la transferencia de datos el HC05 a la ESP 32 y su posterior envío al computador portátil.

Paso 1: Preparación de los módulos

Antes de conectar los módulos, asegúrate de que ambos estén alimentados con una fuente de voltaje adecuada. El módulo HC-05 puede operar entre 3.6V y 6V, mientras que el módulo ESP32 generalmente opera con 3.3V, así que asegúrate de que las tensiones sean compatibles.

Paso 2: Conexiones físicas

Para conectar el módulo HC-05 con el módulo ESP32, sigue estos pasos:

- 1) Asegúrate de que ambos módulos estén apagados antes de conectarlos.

- 2) Conecta el pin VCC del módulo HC-05 al pin de alimentación de 5V del módulo ESP32 o a un regulador de voltaje adecuado si estás usando una fuente de alimentación mayor a 3.3V.
- 3) Conecta el pin GND del módulo HC-05 al pin GND del módulo ESP32 para tener una referencia de tierra común.
- 4) Conecta el pin TX del módulo HC-05 al pin RX del módulo ESP32. Esto permitirá que el módulo HC-05 envíe datos al módulo ESP32.
- 5) Conecta el pin RX del módulo HC-05 al pin TX del módulo ESP32. Esto permitirá que el módulo HC-05 reciba datos del módulo ESP32.

Paso 3: Configuración del módulo HC-05

El módulo HC-05 viene configurado por defecto en modo esclavo.

Para configurar el módulo HC-05 en modo esclavo, sigue estos pasos:

- 1) Asegurarse de que el módulo HC-05 esté desconectado del módulo ESP32 y de cualquier otro dispositivo.
- 2) Conectar el módulo HC-05 a una fuente de alimentación.
- 3) Configurar un Arduino o una interfaz USB-Serial para conectarte al módulo HC-05. Se puede utilizar el monitor serial de Arduino IDE.
- 4) Ingresa al modo de configuración AT del módulo HC-05. Para ello, mantén presionado el botón de enlace (KEY o EN) del módulo y, mientras lo mantienes presionado, enciende el módulo. La luz LED del módulo HC-05 parpadeará lentamente, indicando que está en modo de configuración.
- 5) Abre el monitor serial con una velocidad de 38400 baudios (este es el baud rate de configuración AT del módulo HC-05).
- 6) Envía el comando `AT+ROLE=1` para configurar el módulo HC-05 en modo esclavo. El módulo responderá con "OK" si el comando se ejecutó correctamente.
- 7) Para cambiar el nombre del módulo HC-05, se puede enviar el comando `AT+NAME=NOMBRE`, donde NOMBRE es el nuevo nombre que se desea asignar al módulo. El módulo responderá con "OK" si el comando se ejecutó correctamente.
- 8) Para guardar los cambios y salir del modo de configuración AT, envía el comando `AT+RESET`. El módulo reiniciará y los cambios se guardarán.

Paso 4: Programación del ESP32

Inicio: Comienza el programa.

Preparar el Entorno de Desarrollo:

- 1) Asegurarse de que el entorno de desarrollo Arduino IDE esté configurado correctamente para trabajar con el ESP32.
- 2) Seguir las instrucciones proporcionadas por la documentación oficial del ESP32 para instalar las librerías requeridas.

Definir las Credenciales de Red WiFi:

- 1) Definir el nombre de la red WiFi (ssid) y la contraseña (password) a la que el ESP32 se conectará.

Configurar Parámetros de Red Estáticos:

- 1) Asignar una dirección IP estática al ESP32 definiendo staticIP, gateway y subnet de acuerdo a la configuración de la red.

Crear una Instancia de WiFiClient:

- 1) Crear una instancia de WiFiClient para administrar la conexión WiFi.

Configurar en la Función setup():

- 1) Iniciar la comunicación serial utilizando Serial.begin(9600).
- 2) Establecer el tiempo de espera para la lectura de datos serial mediante Serial.setTimeout(2).
- 3) Llamar a la función connectToWiFi() para conectar el ESP32 a la red WiFi.

Establecer la Conexión WiFi:

- 1) En la función connectToWiFi(), utilizar WiFi.begin(ssid, password) para establecer la conexión a la red WiFi.
- 2) Mientras la conexión no esté establecida (WiFi.status() != WL_CONNECTED), esperar 500 ms y mostrar puntos en la salida serial.
- 3) Una vez conectado, configurar la IP estática si se ha definido y mostrar la IP local del ESP32.

Recibir y Preparar los Datos:

- 1) En la función loop(), verificar si hay datos disponibles en el puerto serial con Serial.available().
- 2) Si hay datos, leer el valor del puerto serial con Serial.parseInt().

- 3) Realizar un prefiltrado si es necesario para ajustar el valor dentro de un rango específico.

Enviar los Datos a MATLAB a través de TCP/IP:

- 1) Llamar a la función `sendToMATLAB(value)` para enviar los datos a un servidor MATLAB.
- 2) Si el cliente WiFi está conectado (`client.connected()`), utilizar `client.write()` para enviar los datos en formato binario.
- 3) Si no está conectado, intentar reconectar utilizando `client.connect()` y luego enviar los datos.

Fin: El programa termina.

En La Figura 2.5 se muestra mediante un diagrama de flujo el programa cargado en la tarjeta de desarrollo ESP 32.

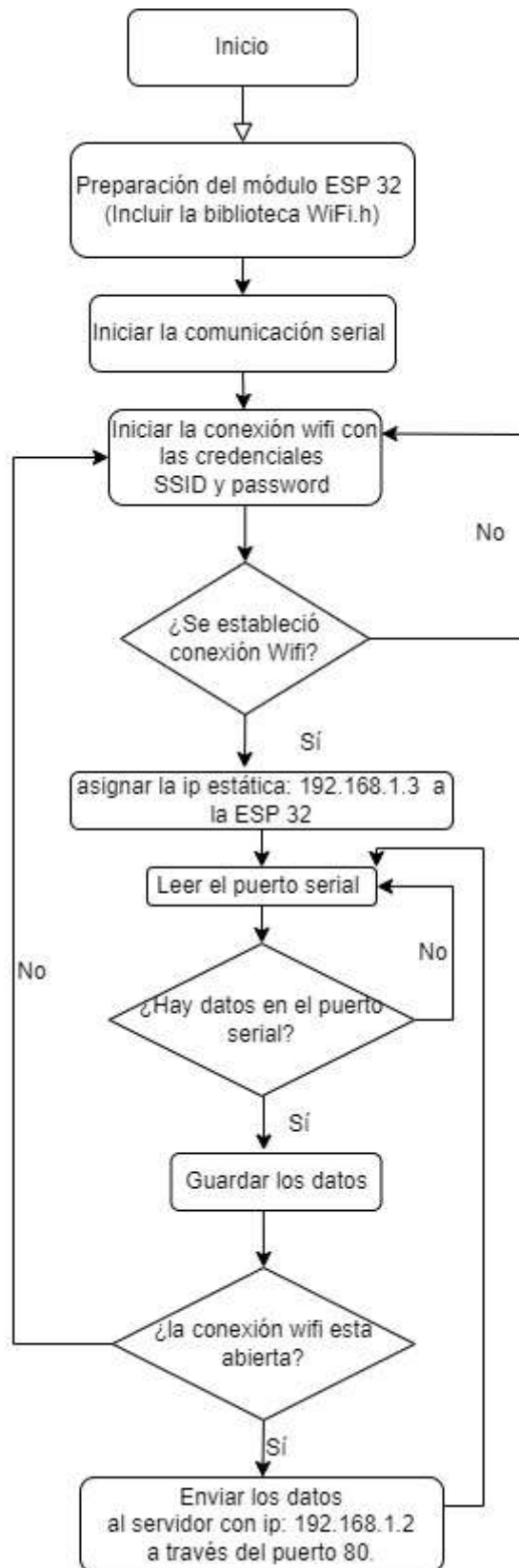


Figura 2.5. Diagrama de flujo del programa de la ESP32.

Paso 5: Verificar la conexión

Cuando ambos módulos están conectados con éxito, se puede empezar a enviar y recibir datos entre ellos.

Se procede al desarrollo del software necesario para el sistema de monitoreo. Esto incluye la asignación de las IP necesarias para las comunicaciones entre los módulos y el diseño de la interfaz, donde se muestran los datos de monitoreo de la tubería de petróleo en tiempo real.

Además, con la implementación de los algoritmos de procesamiento digital de la información se analizan y procesan los datos adquiridos desde el sensor de emisión acústica.

A continuación, se enumeran los pasos correspondientes al proceso descrito:

1) Asignación de IP

IP asignada a la esp 32: 192.168.1.3

IP asignada al servidor (Portatil en donde se tiene la intrfaz): 192.168.1.2

Gateway: 192.168.1.1

2) Asignacion de IP a la ESP 32

En la esp 32 se declara la ip estatica que se le asigna en la red del router.

3) Indicar la IP del servidor a donde envía la esp 32

Dentro del código de la ESP32, se especifica la dirección IP del servidor al que la ESP32 enviará los datos recopilados durante la etapa de censado. Esta dirección IP es una identificación numérica única que se asigna al servidor en la red, permitiendo que los dispositivos se comuniquen con él de manera efectiva.

4) Conexión de la IP a WiFi

Cuando la ESP32 se conecta a la red generada por el router, se indica por monitor serial con el mensaje “ Conectado a WiFi” que la ESP32 se a conectado a la red WiFi del router. Adicionalmente debajo del mensaje anterior se indica la IP estatica asignda a la ESP32.

Luego se obtienen la respuesta que da el monitor serial cuando la ESP32 le fue asignada una IP estática, como se describe en la imagen de la figura 14.



Figura 2.6. Conexión WiFi

Además del mensaje de conexión, se muestra en el monitor serial la dirección IP estática asignada a la ESP32. La dirección IP estática es un valor numérico que identifica de manera única a la ESP32 en la red. Esta dirección permite que otros dispositivos en la misma red se comuniquen con la ESP32 de manera precisa, ya que cada dispositivo tiene una dirección IP única.

La visualización de la dirección IP estática es útil para verificar que la ESP32 ha obtenido correctamente una dirección IP válida de la red WiFi. Esta dirección IP puede ser utilizada para acceder a la ESP32 desde otros dispositivos en la misma red o para dirigir comunicaciones específicas a la ESP32 en la red.

5) Asignación de IP al servidor (portatil en donde se ejecuta la interfaz).

La asignación IP de la pc portatil en donde se ejecuta la interfaz se hace automaticamente mediante DHCP. Cuando la pc portatil se conecta a la red "Acceso Monitoreo" en la opcion propiedades se comprueba que efectivamente la direccion IP que le fue asignada es la: 192.168.1.2

La asignación de una dirección IP al servidor (portátil en el que se ejecuta la interfaz) se realiza de manera automática mediante el protocolo DHCP (Dynamic Host Configuration Protocol). El DHCP es un protocolo utilizado en redes de computadoras para asignar de manera dinámica direcciones IP y otros parámetros de configuración a los dispositivos que se conectan a la red.

Al verificar las propiedades de la conexión de red en la PC portátil, puedes confirmar que la dirección IP asignada es de hecho 192.168.1.2. Esta información es importante para asegurarse de que la PC portátil esté correctamente configurada en la red "Acceso Monitoreo" y pueda comunicarse de manera efectiva con otros dispositivos, como la ESP32 que envía los datos del monitoreo.

Paso 6: Configuración Router

Nombre: Home Gateway

Modelo: HG 530

- 1) Ingreso a la interfaz de configuración del router.
- 2) Reiniciar el router, presionando por 6 segundos en el botón de reset.

Luego de eso aparece entre las redes de Wifi cercanas la red del router, en este caso el nombre de esta red es: INTERNET CNT.

Se conecta a esta red con la contraseña por defecto: 12345678.

Cuando ya se estableció la conexión a la red Wifi del router en el navegador WEB de la PC SE INGRESA la dirección IP: 192.168.1.1



Figura 2.7. Sesión Iniciada

Se debe iniciar sesión con las credenciales:

Nombre de usuario: instalador

Contraseña: .corporacion

Configuración de la red WLAN

En la pestaña Wireless LAN configuramos el nombre de la red y su clave.

SSID: Acceso Monitoreo

Password: Monitoreo_7856

Configuración del DHCP

En la pestaña DHCP, la asignación automática de solo dos direcciones IP.

HG530

- Status
- Basic**
 - ADSL Mode
 - WAN Setting
 - LAN Setting
 - DHCP
 - NAT
 - IP Route
 - Wireless Lan
 - ATM Traffic
- Advanced
- Tools

Wireless Lan

Wireless Setting

Access Point: Enable Disable

Channel ID: ECUADOR
 Channel11 2462MHz Current Channel: 11
(If you select Auto Channel Select, it need to reboot CPE after submitting settings)

SSID Number: 1 2 3 4

SSID Index: 1

SSID: Acceso Monitoreo

Broadcast SSID: Yes No

Use WPS: Yes No

WPS Settings

WPS state: Configured

WPS mode: PIN code PBC

WPS progress: Idle

WMM: Enable Disable

Authentication Type: WPA2-PSK

WPA-PSK

Encryption: AES

Pass-ShareKey: Monitoreo_7856 (8~63 ASCII characters, or 64 Hex string)

Figura 2.8. Asignación de nombre y clave a la red

Se configura el parámetro Client IP Pool Starting Address, con la IP 192.168.1.2 la cual corresponde a la dirección IP de inicio de las IP que DHCP puede asignar a los dispositivos que se conecten a la red WLAN del router.

El otro parámetro que se configura es el parámetro Size of Client Pool, a este parámetro se le asigna el número 2; lo que significa que la cantidad de dispositivos que pueden conectarse a la red WLAN el router son solamente 2.

HG530

- Status
- Basic**
 - ADSL Mode
 - WAN Setting
 - LAN Setting
 - DHCP
 - NAT
 - IP Route
 - Wireless Lan
 - ATM Traffic
- Advanced
- Tools

DHCP

DHCP Settings

DHCP: Server

Client IP Pool Starting Address: 192.168.1.2

Size of Client IP Pool: 2

Primary DNS Server: 0.0.0.0

Secondary DNS Server: 0.0.0.0

Remote DHCP Server: N/A

DHCP lease time: 0 Days 8 Hours 8 Min

WAN Primary DNS Server: 0.0.0.0

WAN Secondary DNS Server: 0.0.0.0

Ethernet: Port 1 Port 2 Port 3 Port 4

WLAN: SSID: 1

DHCP Table

Host Name	IP Address	MAC Address
R12SN	192.168.1.2	38:87:05:07:AD:41

Figura 2.9. Configuración DHCP

Al limitar el número de IPs disponibles aseguramos que tan solo se pueda conectar a la red la esp 32 y el pc portátil.

Mediante el código cargado en la ESP 32 se le asigna a la ESP 32 la IP estática: 192.168.1.3

```
const char* ssid = "Acceso Monitoreo";  
const char* password = "Monitoreo_7856";  
const int serverPort = 80; // Puerto del servidor de MATLAB  
IPAddress staticIP(192, 168, 1, 3);/// ip estatica asignada a la esp 32  
IPAddress gateway(192, 168, 1, 1);/// GATEWAY DEL ROUTER  
IPAddress subnet(255, 255, 255, 0);///mascara de subred
```

Figura 2.10. Asignación a la PC portátil

Mientras que la IP restante la IP: 192.168.1.2 es asignada a la PC portátil en donde se encuentra la interfaz, esta asignación la realiza DHCP.

2.3 ENVIO DE LA SEÑAL CON PROTOCOLO TCP/IP

TCP/IP, el conjunto de protocolos esencial en comunicaciones, es utilizado en aplicaciones de Internet de las cosas (IoT) para asegurar la transmisión segura y confiable de datos. Esto involucra dividir los datos en paquetes, dirigirlos de manera eficiente, establecer conexiones, garantizar la transmisión intacta, controlar el flujo de información, reensamblar y entregar los paquetes en orden y, finalmente, cerrar la conexión. Mediante esta sólida estructura, se logra una comunicación exitosa en redes informáticas e Internet, asegurando que los datos sean entregados confiablemente y de manera segura a través de diversos dispositivos y entornos de red.

El envío de la señal se describe en las siguientes subsecciones:

2.3.1 CONEXIÓN WI-FI

Para utilizar TCP/IP, primero debes asegurarte de que el módulo ESP32 esté conectado a una red Wifi. Puedes configurar el módulo para que se conecte a una red Wifi específica utilizando la biblioteca Wifi.

Una vez que ya se conecta los módulos vía bluetooth y a través del Wifi se procede a enviar los datos a Matlab primero verificando que la conexión esté abierta, es decir se envía los datos a la IP del servidor y a través del mismo puerto para este caso "puerto

80" utilizado inicialmente desde la conexión del módulo HC05 y el módulo ESP32, los datos enviados a Matlab son de tipo flotante. La dirección IP utilizada es: "192.168.68.56".

2.3.2 ESTABLECER UN SOCKET TCP

Después de conectar el módulo ESP32 a la red Wifi, se establece un "socket TCP", que es como un canal de comunicación virtual. Este canal permite que el módulo y otro dispositivo intercambien datos en ambas direcciones, como una conversación telefónica donde ambas partes pueden hablar y escuchar. Este socket es esencial para enviar y recibir información en tiempo real, por ejemplo, para transmitir datos acústicos desde sensores en tuberías a otro dispositivo, como una computadora o una interfaz de software. En resumen, el socket TCP facilita una comunicación bidireccional segura y eficiente entre dispositivos conectados a la red.

El módulo ESP32 inicia una conexión serial con el módulo HC-05 utilizando su dirección IP y el número de puerto adecuado. Una vez establecida la conexión, puedes enviar señales o datos desde el módulo ESP32 al módulo HC-05. El router genera una red para conectar la IP del cliente y la IP del servidor.

2.4 SERVIDOR

En el desarrollo del software para el monitoreo de tuberías de transporte petrolero o metálicas, se implementan varios servicios y configuraciones específicas para asegurar una operación efectiva y continua.

Tcpserver es un servicio esencial dentro del sistema, utilizando el puerto 80 y configurándose con la dirección IP "0.0.0.0". Esto permite al servidor mantener un estado de escucha constante, listo para recibir y manejar solicitudes entrantes a través de dicho puerto. La elección del puerto 80 se alinea con los estándares ampliamente reconocidos para la comunicación HTTP, esencial para la navegación web y transmisión de datos. Al establecer la dirección IP en "0.0.0.0", se abre la puerta a conexiones de diversos orígenes, ampliando el acceso para múltiples usuarios y dispositivos.

La función clave del modo de escucha constante es garantizar que el servidor esté siempre preparado para atender solicitudes entrantes. En el entorno dinámico de la comunicación web, donde los clientes envían peticiones para obtener recursos e información, esta capacidad de respuesta constante resulta fundamental. La configuración en el puerto 80 y el modo de escucha permanente trabajan en conjunto para asegurar que el servidor pueda administrar las solicitudes eficientemente y proporcionar respuestas adecuadas en todo momento.

Por otro lado, el atributo **t.Timeout** desempeña un papel importante al establecer un tiempo de espera para conexiones inactivas. Esta configuración es opcional y se utiliza para controlar el período durante el cual una conexión puede mantenerse inactiva antes de que se desconecte automáticamente. Esta función es especialmente útil en entornos con recursos limitados, ya que libera conexiones no utilizadas y gestiona eficazmente los recursos del servidor. Al asignar un valor a este atributo, se define el intervalo en el cual una conexión puede permanecer inactiva antes de que el sistema tome medidas para cerrarla.

La línea de código **t.BytesAvailable** se ejecuta de manera continua mientras haya datos pendientes de ser leídos en la comunicación serial. Esto significa que la ejecución de esta línea de código está directamente relacionada con la disponibilidad de datos en la comunicación serial. Cuando **t.BytesAvailable** indica la presencia de bytes por procesar, la línea de código correspondiente se activa y lleva a cabo las operaciones necesarias para manejar esos datos.

Por último, **dataSize** hace referencia al tamaño ocupado en memoria por un valor de punto flotante, que en este caso específico es de 4 bytes. El término "dataSize" denota la cantidad de espacio requerido para almacenar un solo valor de punto flotante, un tipo de dato utilizado para representar números reales con fracciones decimales. En esta situación, el valor de 4 bytes indica la cantidad de memoria asignada para almacenar dicho valor de punto flotante.

En conjunto, estos servicios y configuraciones desempeñan un papel crucial en el funcionamiento del software, asegurando la recepción, procesamiento y almacenamiento adecuado de los datos acústicos, y permitiendo la presentación clara y efectiva de los resultados en la interfaz gráfica para su interpretación y toma de decisiones.

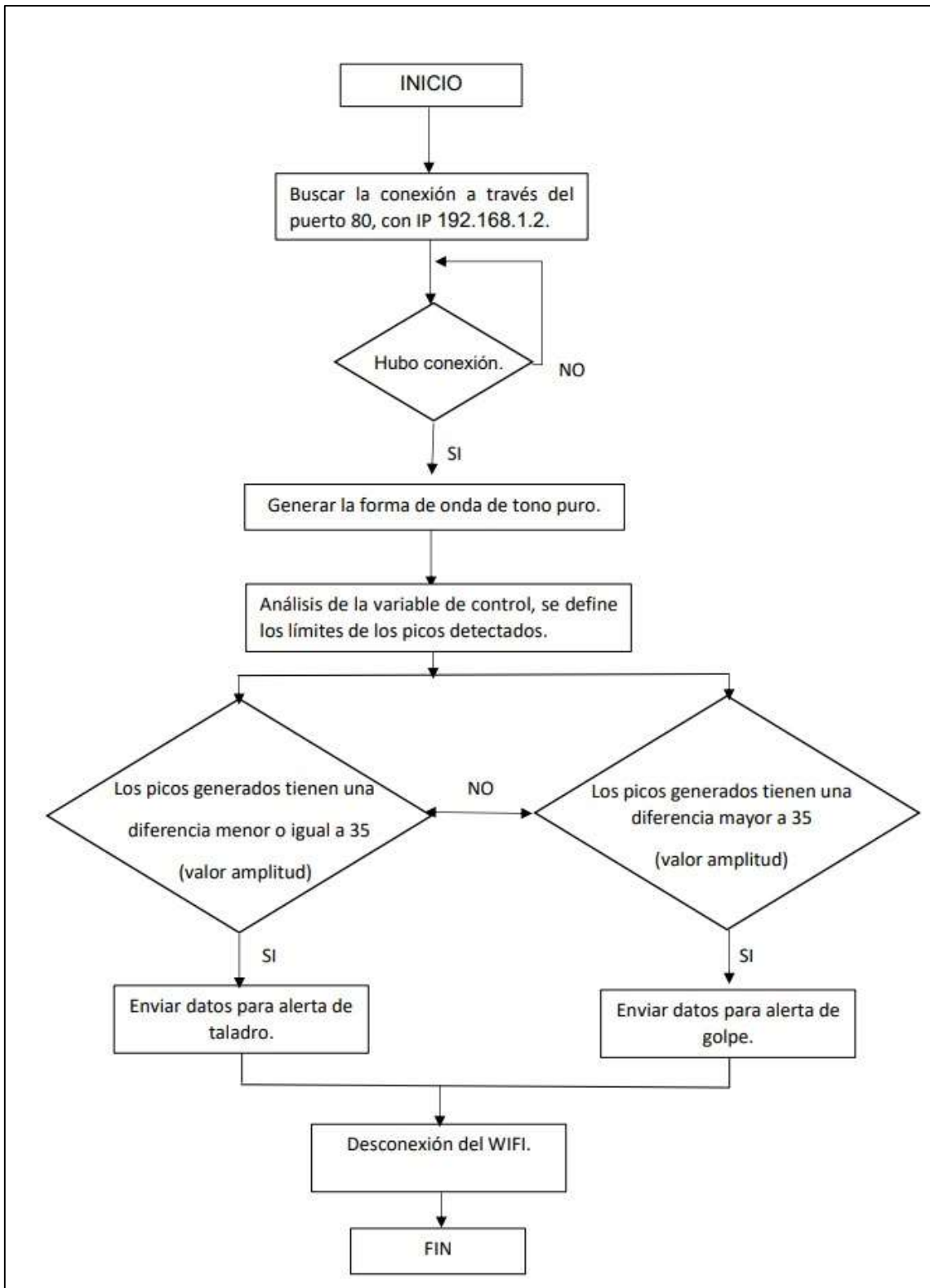


Figura 2.11. Descripción algoritmo software e interfaz gráfica.

Para la conexión con MatLab que es el primer paso como se puede observar en el diagrama de flujo se utiliza los siguientes comandos: `t = tcpserver('0.0.0.0', 80);`
`t.Timeout = 60.`

Se genera una señal de sonido dada por la ecuación:

$$\text{sonido} = \sin(2\pi * \text{frecuencia} * \text{tiempo}) \quad (2.1)$$

Una vez establecida la conexión se procede al análisis de datos, para lo cual se define una variable local que almacena los valores absolutos de los picos de las señales de sonido generadas. El método propuesto utiliza el análisis de picos en una señal acústica para distinguir entre dos tipos de sonidos: aquellos generados por un taladro y los sonidos de golpe.

El procedimiento se basa en la amplitud de los picos y la diferencia entre ellos para tomar decisiones. A continuación, se proporciona una descripción más detallada de cómo funciona este método:

- 1) **Análisis de Picos:** Identificamos los puntos más altos en la señal de sonido, que representan momentos de mayor intensidad.
- 2) **Comparación con Umbral:** se compara la amplitud del pico más alto con un valor predeterminado (143). Si es mayor a este valor, se pasa al siguiente paso.
- 3) **Diferencia entre Picos:** se calcula la diferencia de amplitud entre picos consecutivos para ver si están cercanos o separados.
- 4) **Distinguiendo Taladros y Sonidos de Golpe:**
Caso 1 (Taladro): Si la diferencia entre picos es ≤ 35 , se considera un sonido prolongado, posiblemente de un taladro.
Caso 2 (Sonido de Golpe): Si la diferencia entre picos es > 35 , se interpreta como golpe, sin una señal prolongada.
- 5) **Identificación de Golpes Intensos:** Si el pico más alto es > 150 , se reconoce como un golpe fuerte, indicando un impacto intenso.

La distinción de los sonidos de la perturbación se hace mediante la función **findpeaks** en Matlab, según la documentación en [17], la función "findpeaks" identifica los máximos locales (picos) en el conjunto de datos de entrada llamado "data". Un pico local se refiere a una muestra dentro de los datos que es mayor que las muestras adyacentes o que tiene un valor de infinito. Los picos se enumeran en el orden en que

aparecen en los datos. Además, se omiten los puntos finales de la señal al realizar esta identificación.

Después del envío de datos y procesamiento digital se determina el patrón de golpe o taladro descrito en el paso 4, además del diagrama de flujo de la imagen de la figura 2.11.

2.4.1 INTERFAZ GRAFICA

Componentes Clave del Desarrollo de Interfaces en MATLAB:

- Desarrollo de Interfaz Gráfica en MATLAB para Detección de Sonidos
- Uso de GUIDE (GUI Development Environment): se utiliza GUIDE, una herramienta en MATLAB para crear interfaces gráficas de usuario visualmente.
- Diseña interfaces arrastrando y soltando elementos como botones, gráficos y cajas de texto.

Elementos de la Interfaz:

- La interfaz contiene botones, cajas de texto, gráficos, tablas, etc. Estos elementos permiten la entrada de datos y la visualización de resultados.

Programación y Callbacks:

Aunque GUIDE simplifica la creación, a menudo se requiere programación para definir el comportamiento de la interfaz. Con el uso "callbacks", que son funciones que se ejecutan cuando ocurren eventos específicos, como hacer clic en un botón.

Diseño y Estilo:

GUIDE permite personalizar la apariencia visual de la interfaz, incluidos colores y disposición de elementos.

Para la creación de la Interfaz en MATLAB, se describen los siguientes pasos:

1) Diseño de la Interfaz:

Iniciar una nueva figura utilizando figure.

Agregar elementos gráficos como ejes y botones.

2) Carga de la Señal y Detección:

Cargar la señal de audio desde un archivo usando audioread.

Implementar el algoritmo de detección de taladros y golpes.

3) Representación Gráfica de la Señal:

Utilizar plot para visualizar la señal en los ejes.

4) Indicador Visual (Foco):

Crear un elemento gráfico para el indicador visual.

Modificar BackgroundColor para simular encendido/apagado.

5) Reproducción del Sonido:

Cargar un archivo de sonido para perturbaciones usando audioread.

Usar sound para reproducir sonido tras la detección.

6) Botón de Inicio de Detección:

Agregar un botón con uicontrol.

Establecer un callback para realizar detección, activar el indicador y reproducir sonido.

7) Visualización en Tiempo Real:

Implementar un bucle para actualizar la visualización de la señal en tiempo real.

A continuación, se describe la gráfica de la interfaz de usuario creada:

En la pantalla de inicio se describe los datos de la institución en donde se desarrolló el trabajo de integración curricular, nombre del estudiante y alumno. Una vez que se presentan los datos generales se da clic en Iniciar.



Figura 2.12. Pantalla de Inicio

Luego aparece el recuadro de la figura 2.13 en donde se espera la conexión entre el módulo ESP32, el router y MATLAB, una vez que aparezca el mensaje conectado se da clic en Continuar.



Figura 2.13. Ventana de buscando conexión.

Una vez que se establece la conexión aparece la ventana de la figura 2.14, en esta ventana se puede visualizar en tiempo real la señal que envía el sensor de señales acústicas, en la interfaz se presenta como amplitud el voltaje de la señal y en eje de las x el número de muestras.



Figura 2.14. Alerta de golpe.

En la parte inferior izquierda, cuando se presente una perturbación aparece un mensaje de ALERTA PERTURBACIÓN, esta puede ser de taladro o golpe. En la parte inferior derecha cuando se detecta el tipo de perturbación se encenderá una luz que indica si es taladro o golpe.



Figura 2.15. Alerta de taladro.

Una vez detectada el tipo de perturbación, se da clic en ANÁLISIS y aparecerá la ventana de la figura 2.16, en esta ventana se puede visualizar la tabla de distribución de densidad de radiación y los resultados obtenidos en los que respecta al voltaje de la señal y la duración en segundos. Luego se da clic en Finalizar y vuelve a empezar el programa es decir aparecerá la ventana de inicio la que se describe en la imagen de la figura 20.



Figura 2.16. Ventana de resultados.

En el Anexo No.3 Se describe el montaje del prototipo del trabajo de integración curricular.

3. RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Pruebas y Validación

Una vez que el prototipo es implementado, se llevan a cabo pruebas exhaustivas para validar su funcionamiento y confirmar que cumple con los requisitos establecidos. Esto involucra verificar la adquisición correcta de datos desde los sensores, asegurar la transmisión confiable y segura de los datos, y confirmar el procesamiento adecuado de la información para su presentación en la interfaz.

La detección de "golpes" y "perforaciones" en el monitoreo de tuberías de transporte petrolero o metálicas implica identificar picos seguidos y variaciones notables en los datos acústicos. Estos picos activan un análisis detallado que compara patrones, duración e intensidad con una base de datos de anomalías conocidas para determinar si es un golpe o si se detecta que la tubería está siendo perforada, generando alertas que se muestran en el interfaz desarrollado.

3.1.1 Sistema de procesamiento de datos

Una vez completo el ensamble del prototipo del sistema de procesamiento de datos y con su respectiva corrección de errores relacionados al hardware, se logra obtener el prototipo planificado con las características deseadas. En la figura 3.1 se muestra el resultado final.



Figura 3.1. Prototipo del sistema de procesamiento de datos.

El funcionamiento del sistema se basa en 3 etapas principales las cuales se detallan a continuación.

3.1.2 Recepción de datos.

La recepción de datos se la realiza mediante comunicación serial UART y usando una interfaz inalámbrica, la recepción de datos lo realiza el módulo HC05 en modo esclavo; en mayor detalle este proceso se indica en el apartado 2.1.1.

Luego de que los datos son recibidos son acondicionados y enviados mediante comunicación TCP/IP al computador el cual contiene la interfaz gráfica. En la figura 3.2 se muestra la recepción de datos de una serie de perturbaciones, las cuales posteriormente son procesadas; y esto se indica en el siguiente literal.

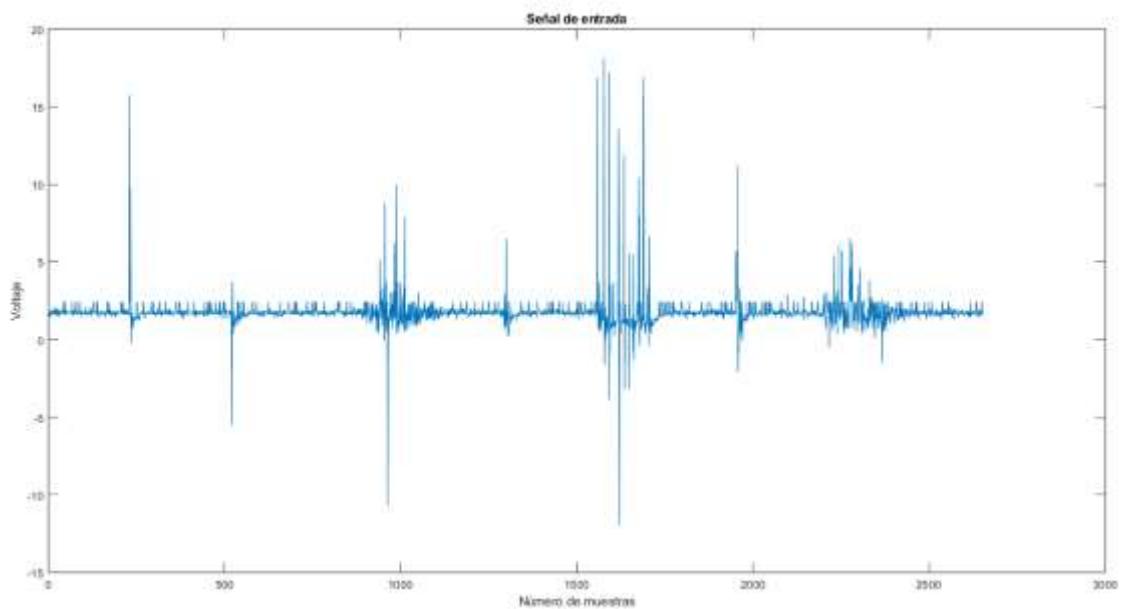


Figura 3.2. Señal con una serie de perturbaciones.

3.1.3 Procesamiento de datos e identificación del evento de perturbación.

Para la identificación del tipo de perturbación se realizó una serie de pruebas con distintas perturbaciones, golpes y perforaciones en el tubo de pruebas. A continuación, se indica el método y los resultados obtenidos para los dos tipos de perturbaciones.

3.1.3.1 Identificación de la perturbación golpe.

Mediante el uso del sonómetro CESVA SC260, se captó el ruido generado por cada una de las perturbaciones de las pruebas realizadas en la tubería de prueba.



Figura 3.3. Sonómetro CESVA SC260.

El sonómetro CESVA SC260 tiene su propio software, el software CESVA capture Studio muestra en una interfaz una serie de datos el dato de interés para la validación de resultados es el:

LAF: Nivel de presión sonora con promediado temporal rápido

El LAF, que es la abreviatura de "Nivel de Presión Sonora Equivalente en el Tiempo A," representa un promedio del nivel de presión acústica durante un período específico. Por lo general, se expresa en decibelios (dB).

En la figura 3.4, se indica los datos tomados de una perturbación de prueba en el interfaz CESVA capture Studio.

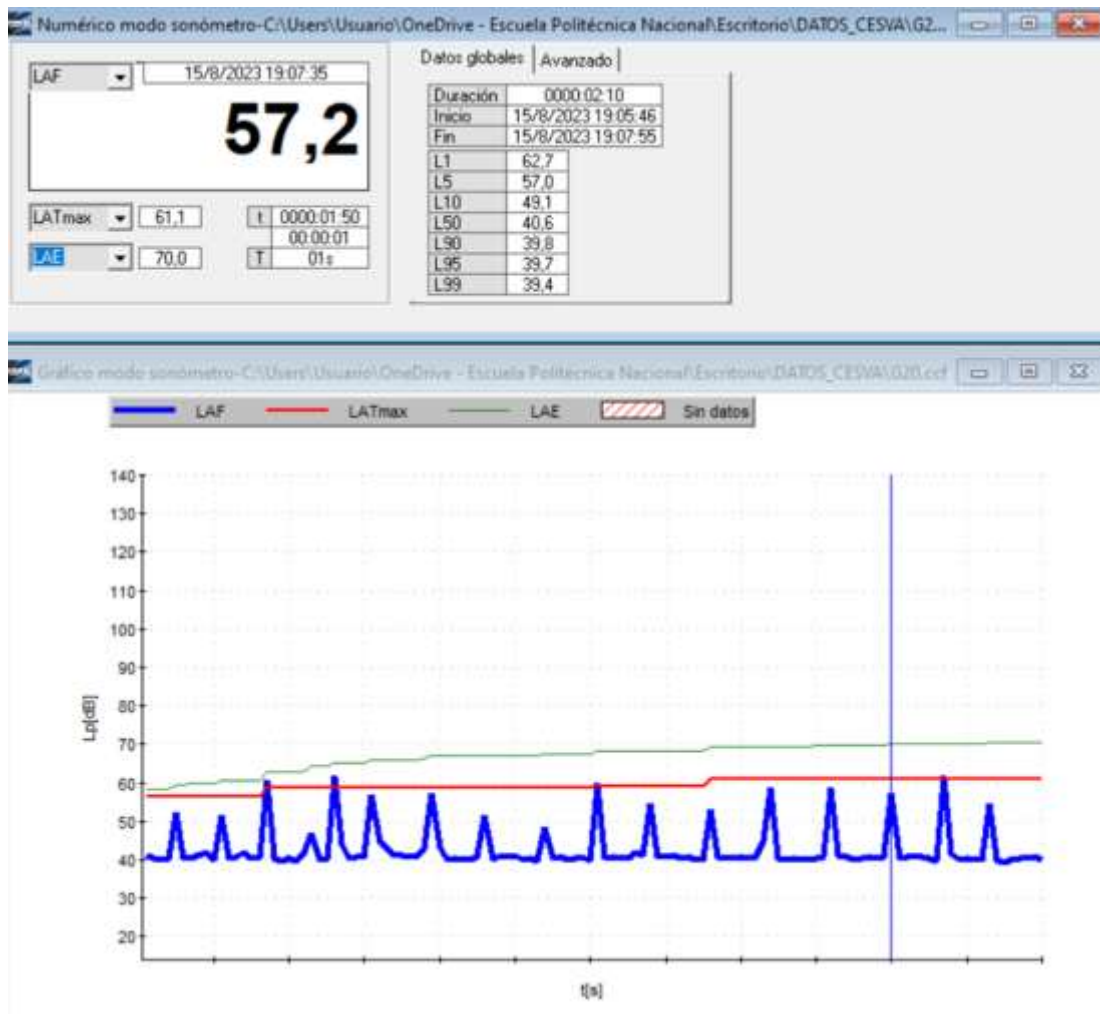


Figura 3.4. Datos mostrados en la interfaz capture Studio del sonómetro CESVA SC260.

El valor en decibelios fue tomado para cada perturbación de prueba realizada y posteriormente tabulado como se indica en las tablas 3.1 y 3.2.

En la interfaz gráfica en el apartado de análisis se obtuvo cada uno de los VRMS (voltaje RMS), correspondiente a cada perturbación.



Figura 3.5. Ventana de análisis de la última perturbación.

Como se observa en la figura 3.5 correspondiente al análisis de la última perturbación, uno de los resultados del análisis que se nos indica es el voltaje RMS.

Finalmente se tabulan estos datos junto al resultado de la identificación del evento de perturbación, el tipo de perturbación correspondiente se muestra en la interfaz gráfica y esto se indica en la figura 3.6. La tabla de los resultados obtenidos se muestran en la tabla 3.1.



Figura 3.6. Identificación del tipo de perturbación golpe en la interfaz gráfica.

Como se muestra en la figura 3.6 el tipo de señal corresponde a un evento de perturbación “golpe”, luego de unos instantes el tipo de perturbación se nos muestra en la interfaz gráfica con una alerta.

Tabla 3.1. Resultados de la identificación de la perturbación golpe.

Perturbación de prueba	Amplitud en decibelios	Voltaje RMS	Detección perturbación golpe
Golpe 1	30.4 dB	1.84 [V]	Sí detectó
Golpe 2	46.7 dB	1.947 [V]	Sí detectó
Golpe 3	47.4 dB	1.95 [V]	Sí detectó
Golpe 4	47.6 dB	1.931 [V]	Sí detectó
Golpe 5	45.0 dB	1.891[V]	Sí detectó
Golpe 6	46.3 dB	1.907 [V]	Sí detectó
Golpe 7	72.6 dB	2.911[V]	Sí detectó
Golpe 8	49.9 dB	1.943 [V]	Sí detectó
Golpe 9	63.0 dB	2.627 [V]	Sí detectó
Golpe 10	39.5 dB	1.813 [V]	Sí detectó

Durante las pruebas y con los resultados obtenidos se puede constatar que la identificación del evento de perturbación tipo golpe es correcta.

3.1.3.2 Identificación de la perturbación perforación.

Para la identificación del tipo de perturbación perforación se hizo uso también de un sonómetro instalado en un dispositivo móvil, al igual que para la identificación de la perturbación denominada golpe. Se captó el ruido generado por cada una de las distintas perturbaciones producto de taladrar la tubería.

En la interfaz gráfica en el apartado de análisis se obtuvo cada uno de los VRMS (voltaje RMS), correspondiente a cada una de las perturbaciones generadas al taladrar la tubería de pruebas.

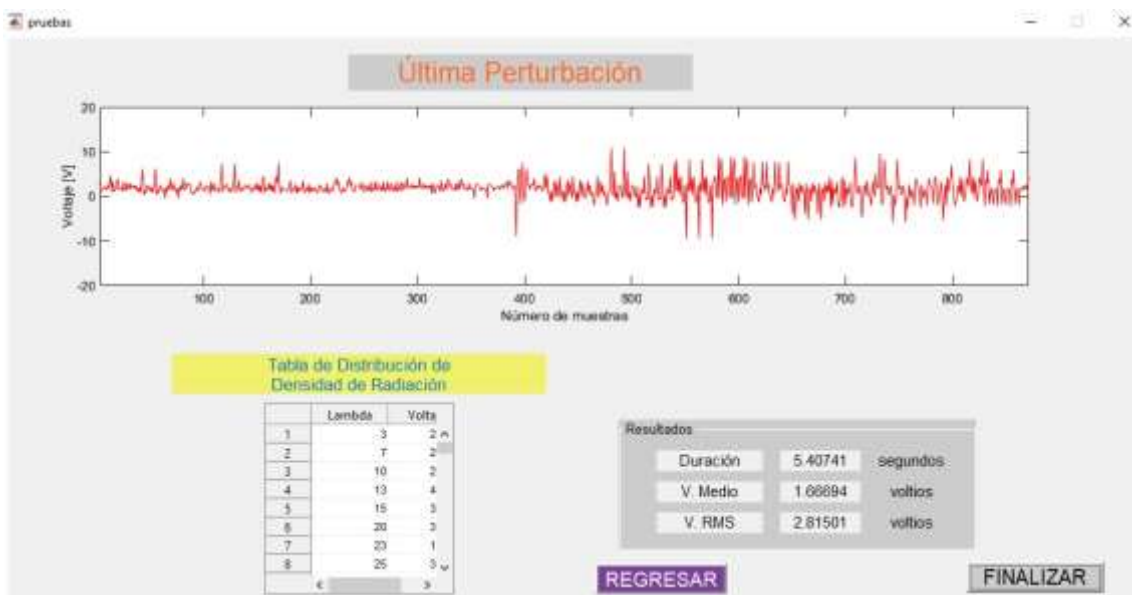


Figura 3.7. Ventana de análisis de la última perturbación.

Como se observa en la figura 3.7 correspondiente al análisis de la última perturbación, uno de los resultados del análisis que se nos indica es el voltaje RMS.

Finalmente se tabularon estos datos junto al resultado de la identificación del evento de perturbación, el tipo de perturbación correspondiente se muestra en la interfaz gráfica y esto se indica en la figura 3.7. La tabla de los resultados obtenidos se muestra en la tabla 3.2.



Figura 3.8. Identificación del tipo de perturbación taladro en la interfaz gráfica.

Como se puede observar en la figura 3.8, la señal de la perturbación corresponde a una perturbación generada al taladrar la tubería de prueba, lo que en la interfaz gráfica se nos indica con una alerta de perturbación y se enciende el tipo de perturbación taladro, que nos indica que se está produciendo una perforación en la tubería.

Tabla 3.2. Resultados de la identificación de la perturbación taladro.

Perturbación de prueba	Amplitud en decibelios	Voltaje RMS	Perturbación taladro (perforación)
Taladrado superficial 1	78.9 dB	1.98 [V]	Sí detectó
Taladrado superficial 2	74.6 dB	1.83 [V]	Sí detectó
Taladrado superficial 3	79.4 dB	2.03 [V]	Sí detectó
Taladrado superficial 4	78.9 dB	2.102 [V]	Sí detectó
Taladrado superficial 5	79.3 dB	2.27 [V]	Sí detectó
Taladrado superficial 6	79.5 dB	2.311 [V]	Sí detectó
Taladrado superficial 7	78.1 dB	2.21 [V]	Sí detectó
Taladrado superficial 8	79.7 dB	2.182 [V]	Sí detectó
Taladrado superficial 9	77.4 dB	1.982 [V]	Sí detectó
Taladrado superficial 10	78.2 dB	2.077 [V]	Sí detectó
Perforación 1	79.3 dB	2.128 [V]	Sí detectó
Perforación 2	80.1 dB	2.143 [V]	Sí detectó
Perforación 3	78.7 dB	2.012[V]	Sí detectó
Perforación 4	79.4 dB	2.419 [V]	Sí detectó
Perforación 5	80.1 dB	2.127 [V]	Sí detectó

Perforación 6	80.3 dB	2.119 [V]	Sí detectó
Perforación 7	78.7 dB	2.012[V]	Sí detectó
Perforación 8	78.4 dB	1.982[V]	Sí detectó
Perforación 9	79.3 dB	2.301 [V]	Sí detectó
Perforación 10	79.2 dB	2.183[V]	Sí detectó

Durante las pruebas y con los resultados obtenidos se puede constatar que la identificación del evento de perturbación es correcta, ya que se detecta correctamente el tipo de perturbación correspondiente a un taladrado en la tubería de prueba.

3.2 CONCLUSIONES

Se ha logrado un hito significativo al desarrollar y poner en marcha un prototipo funcional de un sistema de recepción de datos, una interfaz intuitiva y un sólido sistema de procesamiento digital de información. Este logro se construyó sobre una base sólida, ya que se llevó a cabo una minuciosa revisión de la literatura que proporcionó un profundo entendimiento de las complejas técnicas de procesamiento de señales acústicas y el filtrado de datos de sensores de emisión acústica.

El enfoque metodológico adoptado abarcó todas las etapas desde la concepción hasta la validación y optimización del sistema, lo que permitió un desarrollo integral y completo. En particular, se diseñó y desarrolló una interfaz de usuario que proporciona una visualización clara y accesible de los datos, mediciones provenientes del sensor y las alertas correspondiente al tipo de perturbación detectada.

La implementación del algoritmo para interpretar los datos recibidos permitió analizar y extraer información relevante de las señales acústicas. Con la información recibida se logró hacer una correlación matemática y determinar el tipo de perturbación.

La implementación exitosa del algoritmo de interpretación de datos ha llevado a la capacidad de analizar y extraer información crítica de las señales acústicas recibidas. Los resultados obtenidos hasta ahora son alentadores, ya que han incluido la detección de perturbaciones.

3.3 RECOMENDACIONES

Para futuros proyectos, se recomienda considerar la posibilidad de realizar pruebas y validaciones exhaustivas del prototipo en un entorno de laboratorio y en condiciones reales de operación de la tubería de transporte petrolero. Esto ayudará a garantizar las recomendaciones que se han formulado en función de los desafíos y oportunidades

identificados durante el desarrollo del proyecto actual. Estas sugerencias están destinadas a mejorar y fortalecer futuros proyectos en el mismo ámbito. A continuación, se explica por qué se han propuesto estas recomendaciones:

Pruebas y Validaciones Exhaustivas: La implementación de pruebas tanto en un entorno de laboratorio controlado como en condiciones reales de operación permitirá verificar la confiabilidad y precisión del sistema en situaciones prácticas. Esto garantizará que el prototipo funcione de manera efectiva y precisa en condiciones reales, lo que es esencial para su adopción y éxito en aplicaciones del mundo real.

Mantenerse Actualizado: Mantenerse informado sobre los últimos avances en el campo es crucial para asegurarse de que el sistema se beneficie de las últimas técnicas y enfoques. La tecnología de procesamiento de señales y sensores es un área en constante evolución, y mantenerse al día permitirá la incorporación de métodos más avanzados en futuros proyectos.

Mejora de la Interfaz: Una interfaz usuario amigable y efectiva es esencial para que los operadores puedan interpretar y tomar decisiones basadas en los datos. La implementación de técnicas de visualización más avanzadas permitirá una comunicación más eficiente de la información y facilitará la toma de decisiones informadas.

Técnicas de Procesamiento Avanzadas: El aprendizaje automático y la inteligencia artificial ofrecen el potencial de llevar el análisis de datos a un nivel superior. Estas técnicas pueden permitir una detección más precisa de patrones y anomalías en las señales, así como una mayor automatización en la interpretación de los datos, lo que podría resultar en una mayor eficiencia y precisión en la toma de decisiones.

4. REFERENCIAS BIBLOGRÁFICAS

- [1] S. Wu, J. Wang, and Y. Sun, "Real-Time Monitoring System for Pipeline Leakage Based on Acoustic Emission," *Sensors*, vol. 20, no. 6, p. 1649, 2020.
- [2] E. Martinez et al., "Leak Detection in Pipelines Using Acoustic Emission Technique and Machine Learning Algorithms," *Journal of Loss Prevention in the Process Industries*, vol. 58, pp. 191-199, 2019.
- [3] S. Al-Bugharbee et al., "Pipeline Leakage Monitoring System Using Acoustic Emission Technique and Internet of Things," *Journal of Natural Gas Science and Engineering*, vol. 49, pp. 214-223, 2018.
- [4] L. Liu et al., "A Novel Pipeline Leakage Detection Method Based on Acoustic Emission and Transient Temperature Analysis," *IEEE Access*, vol. 6, pp. 53067-53077, 2018.
- [5] S. R. Silva et al., "A Comparative Study of Acoustic Emission Techniques for Leak Detection in Pipelines," *Journal of Loss Prevention in the Process Industries*, vol. 47, pp. 80-88, 2017.
- [6] D. Meng et al., "Leak Detection in Pipelines Based on the Hybrid Method of Acoustic Emission and Wavelet Transform," *Journal of Loss Prevention in the Process Industries*, vol. 46, pp. 143-153, 2017.
- [7] P. C. Omenzetter, "Monitoring of Pipe Leaks Using Acoustic Emission: Part 1 - Experimental and Numerical Modeling," *Journal of Pipeline Systems Engineering and Practice*, vol. 8, no. 2, p. 04017018, 2017.
- [8] Z. Chen et al., "An Acoustic Emission-Based Pipeline Leakage Localization System with Small Unmanned Aerial Vehicles," *IEEE Access*, vol. 5, pp. 11884-11894, 2017.
- [9] J. T. C. W. Tang et al., "Machine Learning Techniques for Acoustic Emission Based Pipeline Leakage Detection," *Journal of Loss Prevention in the Process Industries*, vol. 44, pp. 242-252, 2016.
- [10] A. Suárez et al., "Acoustic Emission Signal Processing Techniques for Pipeline Leak Detection," *Sensors*, vol. 16, no. 8, p. 1314, 2016.
- [11] L. Chen et al., "Leakage Detection in Oil Pipelines Based on Ensemble Empirical Mode Decomposition and Acoustic Emission Techniques," *Sensors*, vol. 16, no. 1, p. 103, 2016.
- [12] M. Wang et al., "Real-Time Pipeline Leakage Detection Based on Acoustic Emission and Genetic Algorithm," *Sensors*, vol. 15, no. 6, pp. 13091-13106, 2015.
- [13] R. Li et al., "Pipeline Leakage Detection Method Based on Acoustic Emission Technique and Time-Frequency Analysis," *Mathematical Problems in Engineering*, vol. 2014, Article ID 840849, 2014.
- [14] K. Wang et al., "A Pipeline Leakage Monitoring System Using an Acoustic Emission Technique," *Sensors*, vol. 13, no. 2, pp. 2038-2053, 2013.
- [15] J. López et al., "A Pipeline Leakage Detection Method Based on Acoustic Emission and Principal Component Analysis," *Sensors*, vol. 12, no. 7, pp. 9038-9051, 2012.

[16] A. V. Oppenheim and R. W. Schaffer, "Discrete-Time Fourier Analysis," in Discrete-Time Signal Processing, 3rd ed., Upper Saddle River, NJ, USA: Prentice-Hall, 2010, pp. 79-127.

[17] MathWorks. "findpeaks Function - MATLAB & Simulink". MathWorks. [En línea]. Disponible en: <https://la.mathworks.com/help/signal/ref/findpeaks.html>. Acceso: Agosto 16, 2023.

5. ANEXOS

ANEXO I.

Tabla A.1. Pines del módulo HC05

No. Pin	Nombre	Descripción	Funcionalidad
1	VCC	Alimentación (3.6V - 6V)	Alimentación del módulo HC-05
2	GND	Tierra (GND)	Conexión a tierra (0V)
3	TX	Transmisión UART (Bluetooth al host)	Envía datos desde el HC-05 al otro dispositivo a través de UART
4	RX	Recepción UART (Host al Bluetooth)	Recibe datos desde el otro dispositivo y los envía al HC-05 a través de UART
5	EN	Enable (Habilitar)	Activa el modo de configuración del HC-05 (AT mode)
6	STATE	Estado (Conexión Bluetooth)	Indicador del estado de conexión Bluetooth

ANEXO II.

Tabla A.2. Pines del módulo ESP32.

No. Pin	Nombre	Descripción	Funcionalidad
1	3V3	Alimentación de 3.3V	Alimentación del ESP32 con 3.3V
2	GND	Tierra (GND)	Conexión a tierra (0V)
3	EN	Enable (Habilitar)	Activa el módulo ESP32 cuando se establece a HIGH
4	34	GPIO 34	Propósito general Input/Output (GPIO)
5	35	GPIO 35	Propósito general Input/Output (GPIO)
6	32	GPIO 32	Propósito general Input/Output (GPIO)
7	33	GPIO 33	Propósito general Input/Output (GPIO)
8	25	GPIO 25	Propósito general Input/Output (GPIO)
9	26	GPIO 26	Propósito general Input/Output (GPIO)
10	27	GPIO 27	Propósito general Input/Output (GPIO)
11	14	GPIO 14	Propósito general Input/Output (GPIO)
12	12	GPIO 12	Propósito general Input/Output (GPIO)
13	13	GPIO 13	Propósito general Input/Output (GPIO)
14	23	GPIO 23	Propósito general Input/Output (GPIO)
15	22	GPIO 22	Propósito general Input/Output (GPIO)
16	21	GPIO 21	Propósito general Input/Output (GPIO)
17	19	GPIO 19	Propósito general Input/Output (GPIO)
18	18	GPIO 18	Propósito general Input/Output (GPIO)
19	5V	Alimentación de 5V	Alimentación del ESP32 con 5V (pin VIN)
20	GND	Tierra (GND)	Conexión a tierra (0V)
21	17	GPIO 17	Propósito general Input/Output (GPIO)
22	16	GPIO 16	Propósito general Input/Output (GPIO)

No. Pin	Nombre	Descripción	Funcionalidad
23	4	GPIO 4	Propósito general Input/Output (GPIO)
24	0	GPIO 0	Propósito general Input/Output (GPIO)
25	2	GPIO 2	Propósito general Input/Output (GPIO)
26	15	GPIO 15	Propósito general Input/Output (GPIO)
27	3.3V	Alimentación de 3.3V	Alimentación del ESP32 con 3.3V
28	EN	Enable (Habilitar)	Activa el módulo ESP32 cuando se establece a HIGH
29	3.3V	Alimentación de 3.3V	Alimentación del ESP32 con 3.3V
30	GND	Tierra (GND)	Conexión a tierra (0V)

ANEXO III.

Prototipo

Circuito impreso PCB: Una vez realizado el diseño del prototipo de recepción de datos se obtiene el circuito impreso que se muestra en la siguiente figura.

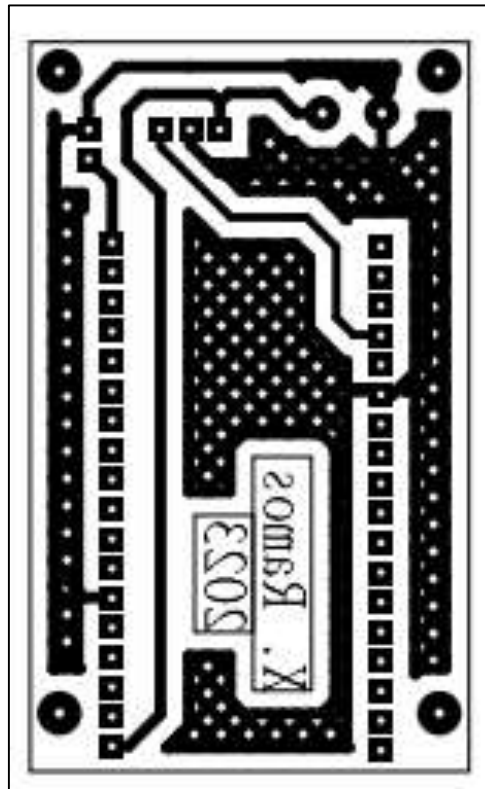


Figura A.1. Circuito Impreso diseñado en el software Proteus.

Esta placa se imprime en una baquelita, como se muestra en la siguiente figura.



Figura A.2. Circuito Impreso.

Montaje del módulo: Se imprime en 3D el prototipo del hardware en donde se encuentra, el módulo ESP32, Hc05, cables para comunicación serial y el circuito impreso.



Figura A.3. Prototipo del módulo de recepción de Datos

Montaje del hardware completo: Se presenta el módulo Wifi y el circuito montaje del módulo antes descrito.



Figura A.4. Hardware completo del prototipo de sistema.

ANEXO IV.

Manual de Usuario.

A continuación, se describe los pasos a realizarse para un correcto funcionamiento del prototipo del sistema de procesamiento de datos.

Alimentación del prototipo.

Paso 1: Se debe alimentar el módulo mediante el conector USB, el cual debe suministra un voltaje de 5 VDC y una corriente de al menos 300 mA.



Figura A.5. Alimentación del módulo de recepción.

Paso 2: Se debe también alimentar el router Huawei mediante una fuente de 12 VDC que suministre al menos 500 mA.



Figura A.6. Alimentación del router.

Paso 3: Una vez que el módulo y el router son alimentados estos encenderán, y se puede continuar con el siguiente paso.

Instalación del programa

Paso 1: Descargar el instalador del programa Monitoreo_Perturbacion 1.0, el instalador se encuentra en la siguiente carpeta compartida: https://epnecuador-my.sharepoint.com/:f:/g/personal/ricardo_ramos_epn_edu_ec/EpSOPP37U6IPsm7fnnvHD4EB-5dzm78AxdHo8Qp4eZ0OeQ?e=WBtOkG

Paso 2: Ejecutar el instalador MyApplnStaller_mcr.exe y empezará la instalación.

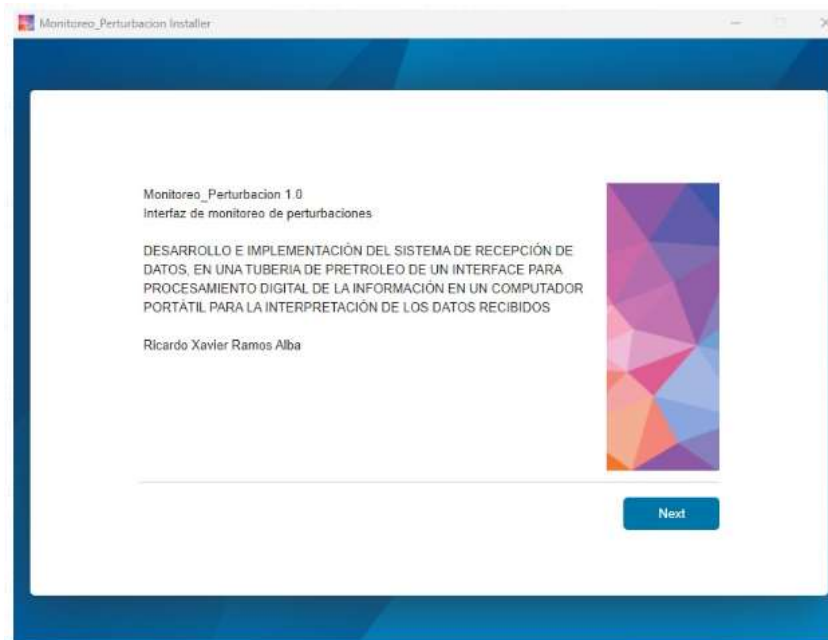


Figura A.7. Ventana de inicio de instalación.

Paso 3: En la ventana que se muestra arriba cliquee en next para continuar con la instalación.

Paso 4: En la ventana que aparece seleccione la carpeta de destino en donde va a guardarse el programa y seleccione la casilla “Add shortcut to deskpot”, si quiere generar un acceso directo en el escritorio.

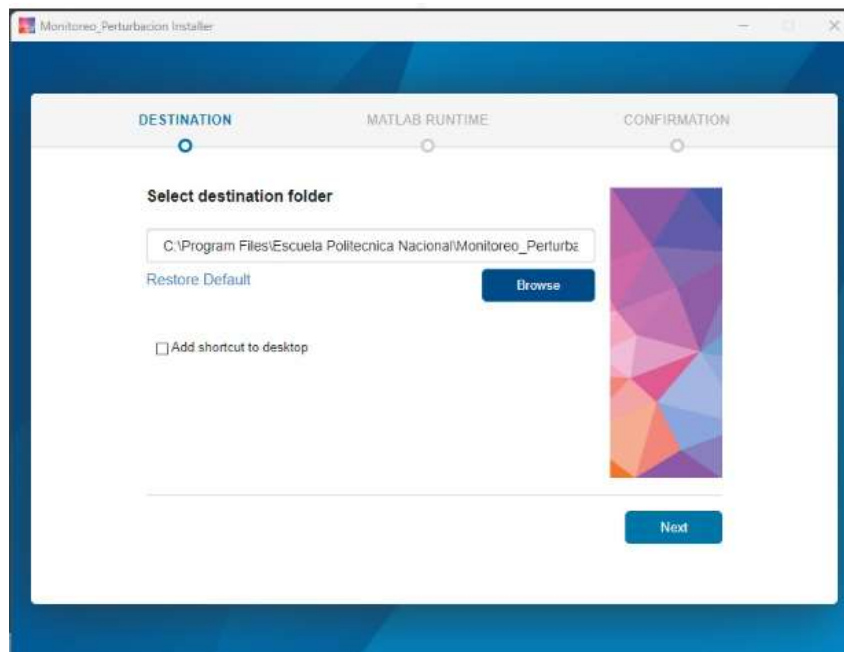


Figura A.8. Ventana de carpeta de destino de los archivos del programa.

Paso 5: En la siguiente ventana se debe seleccionar la ubicación en donde se va a guardar el compilador de MATLAB, el archivo necesario para ejecutar MATLAB.

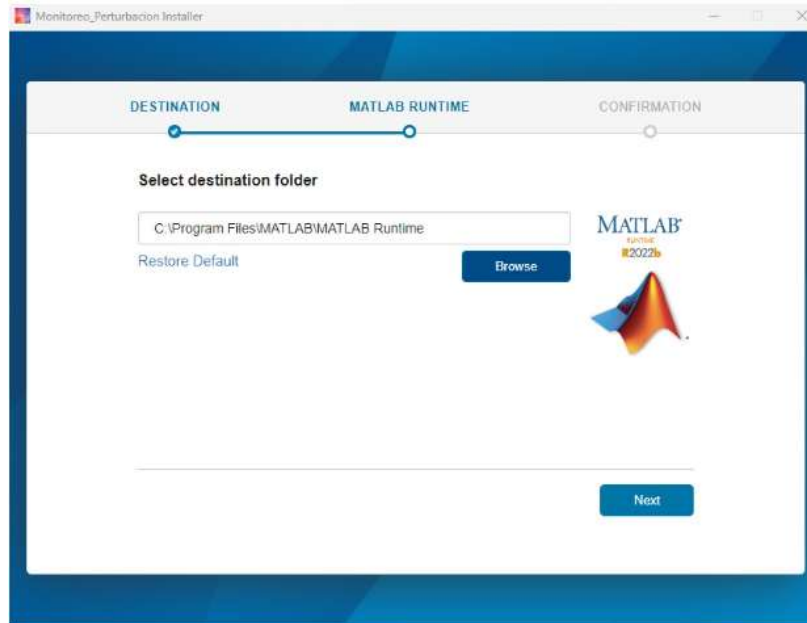


Figura A.9. Tercera ventana de instalación.

Paso 6: En la siguiente ventana se debe aceptar los términos y condiciones de la licencia.

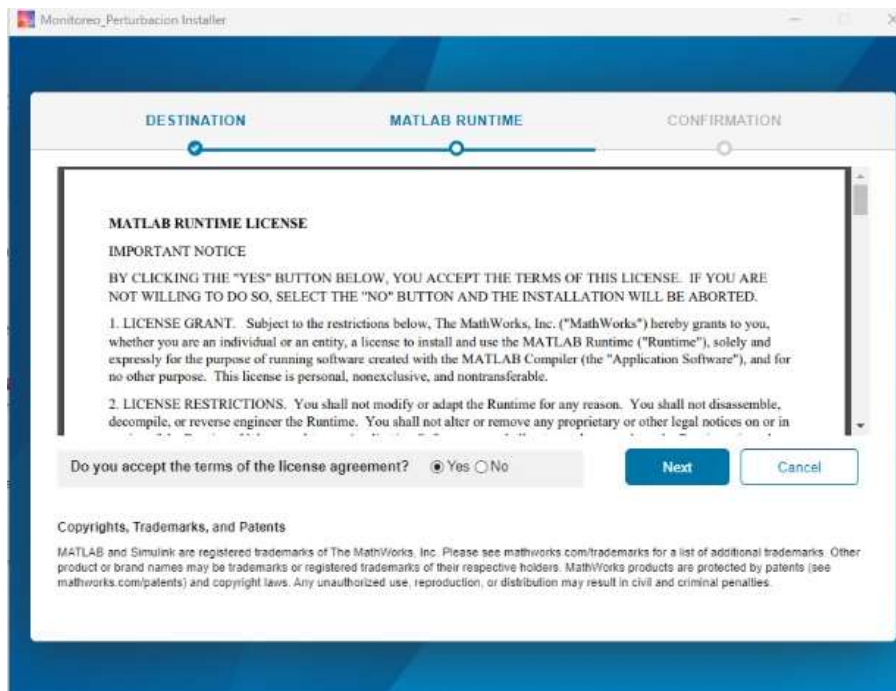


Figura A.10. Ventana de términos y condiciones.

Paso 7: Luego de aceptar los términos cliquee en “Next” y aparecerá una ventana para confirmar que se desea instalar el programa Monitoreo_Perturbacion y el complemento de Matlab.

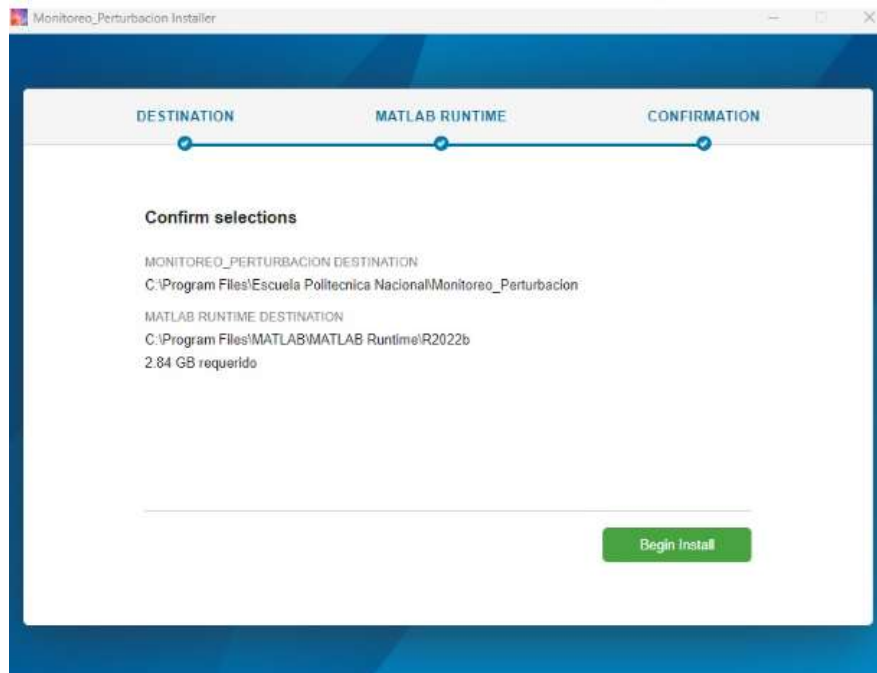


Figura A.11. Inicio de instalación.

Paso 8: Al clicar en el botón “Begin Install” empezará la instalación.

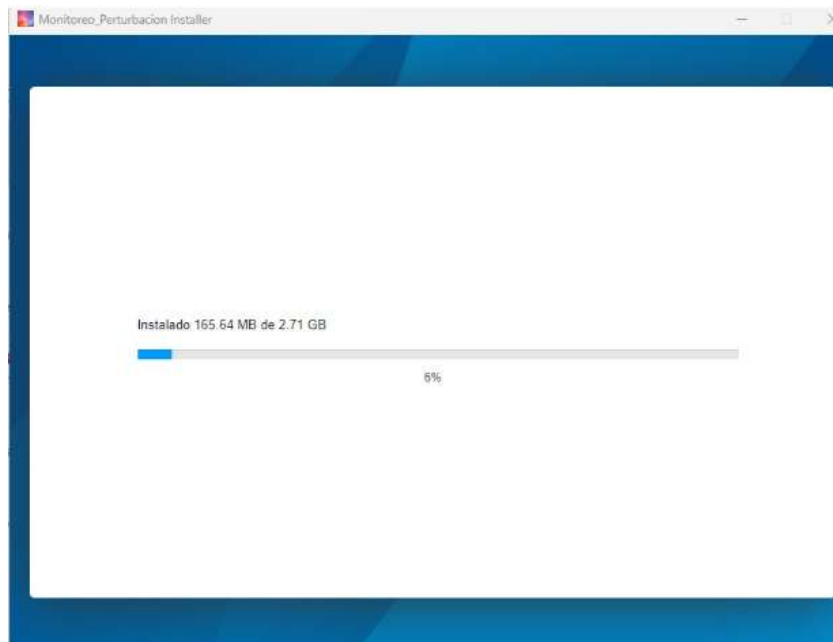


Figura A.12. Ventana de progreso de instalación.

Paso 10: Una vez finalizada la instalación se puede seguir con el siguiente ítem.

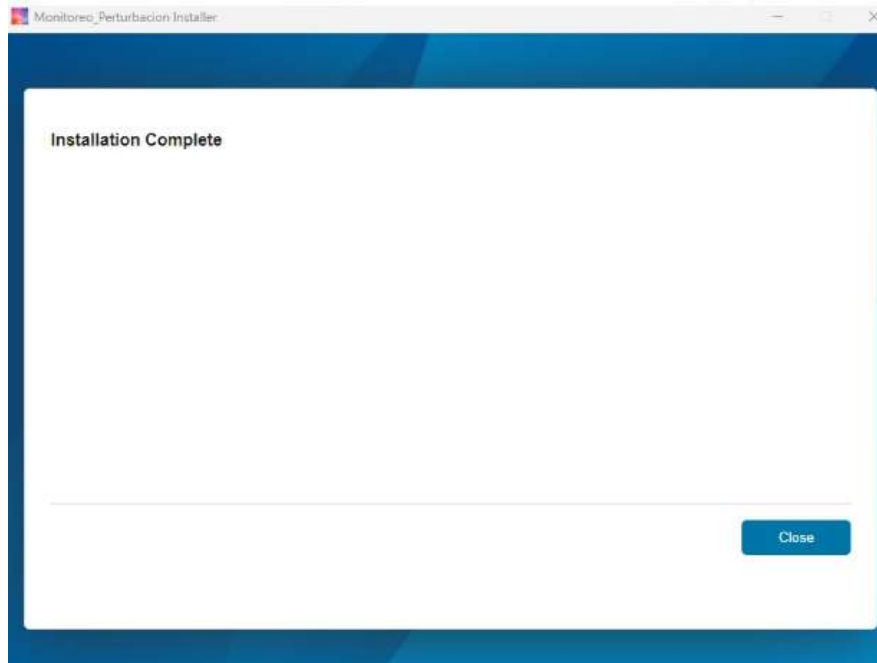


Figura A.13. Ventana de finalización de la instalación.

Inicialización de la interfaz gráfica.

Paso 1: Se debe ejecutar el programa, Monitoreo_Perturbacion en el computador portátil en el cual se va a ejecutar el monitoreo de la tubería.

Paso 2: Antes de dar inicio al programa, la computadora portátil debe conectarse a la red "Acceso Monitoreo" con la contraseña: Monitoreo_7856

Paso 3: Una vez conectado a la red "Acceso Monitoreo" hay que verificar que el router nos haya asignado la ip: 192.168.1.2, si no lo hizo hay que asignarle esta ip al computador manualmente.

Paso 4: Se debe iniciar el programa y luego en la pantalla de inicio se debe presionar el botón de iniciar.



Figura A.14. Ventana de inicio de la interfaz.

Paso 5: Luego de esto muestra la siguiente pantalla.



Figura A.15. Ventana de conexión.

Paso 6: Luego de que en la anterior pantalla haya aparecido el mensaje continuar se debe presionar el botón del módulo para que este se sincronice con el computador y empiece a enviar los datos.



Figura A.16. Sincronización modulo de recepción con interfaz.

Paso 7: Luego de esto se nos desplegara la interfaz gráfica para el monitoreo de la tubería



Figura A.17. Ventana principal de la interfaz.

Manejo de la interfaz gráfica

- a) En la interfaz gráfica se nos indica por defecto la señal de entrada en tiempo real y la detección del evento de perturbación ya sea golpe o taladro.
- b) Además, en la interfaz gráfica se tiene el botón de análisis el cual al presionarlo lleva a la ventana de análisis.



Figura A.18. Ventana de análisis.

- c) Para regresar a la ventana principal se debe presionar el botón de regresar.

Cierre del programa

Para cerrar el programa se debe presionar el botón finalizar, el botón finalizar se encuentra tanto en la ventana principal como en la ventana de análisis.