



UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA

FACULTAD DE SISTEMAS Y TELECOMUNICACIONES

TRABAJO DE TITULACIÓN

**Implementación de una Red de Sensores
Inalámbricos Usando Tecnología LoRa e IIoT en el
Monitoreo y Control de un Motor para mejorar su
eficiencia operativa.**

MODALIDAD

PROPUESTA TECNOLÓGICA

AUTOR

SANTOS ZALAMEA MARLON LEONEL

CARRERA

ELECTRÓNICA Y TELECOMUNICACIONES

TUTOR

ING. CARLOS ALBERTO SALDAÑA ENDERICA

LA LIBERTAD – ECUADOR

2025-2026

AGRADECIMIENTO

Especialmente doy Gracias a Dios por brindarme salud y vida, a mis Padres, Madrastra y Hermanos por brindarme el apoyo, motivación y oportunidad para poder estudiar esta carrera profesional.

En memoria de mi abuelita Lorenza Merchán y Violeta Soriano, cuya ausencia se convirtió en fortaleza para poder seguir adelante en las dificultades que nos presenta la vida.

DEDICATORIA

Dedico este trabajo a Dios por brindarme salud y vida, por darme el apoyo de no rendirme en situaciones complicadas de la vida, además dedico este trabajo a mis Padres, Madrastra y Hermanos por ser ese pilar fundamental de mi vida y no permitir rendirme, también doy gracias Noelia, Daniela, Cindy y Cesar cuyos compañeros fueron esenciales, el cual se convirtieron en mis amigos motivándome, ayudándome y a superarme para no rendirme en nuestra carrera profesional.

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de titulación denominado: “IMPLEMENTACIÓN DE UNA RED DE SENSORES INALÁMBRICOS USANDO TECNOLOGÍA LORA E IIOT EN EL MONITOREO Y CONTROL DE UN MOTOR PARA MEJORAR SU EFICIENCIA OPERATIVA”, presentado por el estudiante Marlon Leonel Santos Zalamea, de la carrera de Electrónica y Telecomunicaciones de la Universidad Estatal Península de Santa Elena, me permito declarar que luego de haber orientado, estudiado y revisado, lo apruebo en todas sus partes y autorizo al estudiante para que inicie los trámites legales correspondientes.

Se adjunta como anexos los reportes generados por la plataforma COMPILATIO que reportó 4 % de plagio del documento final de tesis.



Ing. Carlos Alberto Saldaña Enderica, Mgtr.
Docente tutor

TRIBUNAL DE GRADO



Validar únicamente en FirmaEC.
Firmado electrónicamente por:
**JOSE MIGUEL SANCHEZ
AQUINO**

**Ing. Washington Torres Guin, Mgtr.
DECANO DE FACULTAD**

**Ing. José Sánchez Aquino, Mgtr.
DIRECTOR DE CARRERA**



**Carlos Alberto
Saldana Enderica**
 Time Stamping
Security Data

**Ing. Carlos Saldaña Enderica, Mgtr.
DOCENTE TUTOR**



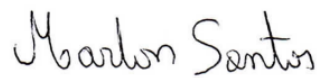
Firmado electrónicamente por:
**OSCAR WLADIMIR
GOMEZ MORALES**
Validar únicamente con FirmaEC

**Ing. Óscar Gómez Morales, Ph.D.
DOCENTE ESPECIALISTA**

**Abg. María Rivera González, Mgtr.
SECRETARIA GENERAL**

DECLARACIÓN DEL AUTOR

Yo, Marlon Leonel Santos Zalamea, el cual soy Egresado de la Universidad Estatal Península de Santa Elena, de la Facultad de Sistemas Y Telecomunicaciones, y de la carrera Electrónica y Telecomunicaciones, declaró que toda responsabilidad de la investigación realizada del presente trabajo de titulación es de mi responsabilidad y el patrimonio del mismo pertenece a la Universidad Estatal Península de Santa Elena.



Marlon Leonel Santos Zalamea
Autor

RESUMEN

En la actualidad, el Ecuador realiza exportaciones de una variedad de productos, uno de ellos son los camarones, el cual se los cría en camaroneras, donde el entorno debe ser ideal para su crecimiento. Para ello, uno de los principales factores que influyen en su producción es la oxigenación, que se lo realiza por Aireadores donde se utiliza motores eléctricos.

Las camaroneras no cuentan con el personal o las tecnologías de bajo coste para abarcar un sistema de monitorización de largas distancias, debido a que algunas emplean solo la visualización de los equipos por medio del personal o no implementan una central donde lleven un registro de los parámetros eléctricos.

Para ello se plantea un sistema de supervisión y monitorización de los parámetros eléctricos de los motores, para ejecutar se utilizará tecnologías de bajo costo como es LoRa, complementado con una red de sensores inalámbricos y una monitorización por medio del HMI, además, se utilizará una central donde se almacenara los datos por medio de una Raspberry pi 4, el cual son ideales para este tipo de entorno debido a sus largas distancias.

El siguiente proyecto que utiliza tecnologías de largo alcance resulta ser favorable para la supervisión de los parámetros eléctricos de los Aireadores en un ambiente controlado del Universidad Península de Santa Elena.

Abstract

At present, Ecuador exports a variety of products, one of them is shrimp, which are raised in shrimp farms where the environment must be ideal for their growth. For this, one of the main factors that influence their production is oxygenation, which is done by aerators where electric motors are used.

Shrimp farms do not have the personnel or low-cost technologies to cover a long-distance monitoring system, because some use only the visualization of the equipment by personnel or do not implement a centralized system for recording electrical parameters.

For this, a system of supervision and monitoring of the electrical parameters of the motors is proposed, to execute it, low-cost technologies such as LoRa will be used, complemented with a network of wireless sensors and monitoring through the HMI, in addition, a central where the data will be stored through a Raspberry pi 4 will be used, which are ideal for this type of environment due to its long distances.

The following project that uses long range technologies is favorable for the monitoring of the electrical parameters of the aerators in a controlled environment of the University Peninsula of Santa Elena.

Índice Base

CAPÍTULO I

1. Fundamentación	1
1.1 Antecedentes	1
1.2 Descripción del proyecto	3
1.3 Objetivos del proyecto	3
1.4 Justificación	3
1.5 Alcance	4
1.5.1. Desarrollo de un sistema de comunicación inalámbrico	4
1.5.2. Integración de las tecnologías Raspberry Pi 4, PLC y HMI.	4
1.5.3. Simulación de diversos escenarios en la gestión del motor en un laboratorio.....	5
1.5.4. Desarrollo de una interfaz HMI para la gestión y control de un motor	5
1.6 Metodología.....	5
1.6.1. Metodología bibliográfica.....	5
1.6.2. Investigación Descriptiva.....	5
1.6.3. Investigación Experimental.....	5
1.6.4 Investigación Aplicada.	6
1.7 Resultados Esperados	6
1.7.1 Desarrollo de un sistema de comunicación inalámbrico.....	6
1.7.2 Integración de las tecnologías Raspberry Pi 4, PLC y HMI.....	6
1.7.3 Simulación de diversas condiciones del motor en un laboratorio ...	6
1.7.4 Desarrollo de una interfaz HMI para el monitoreo y control de un motor.....	6

CAPÍTULO II

2.Marco Referencial.....	7
2.1 Marco Contextual	7
2.2 Marco Conceptual	7
2.2.1 Redes Inalámbricas.....	7
2.2.1.1 Redes LAN	7
2.2.1.2 Redes WPAN.....	8
2.2.1.3 Redes WLAN.....	8
2.2.1.4 Redes WMAN.....	8
2.2.1.5 Redes WWAN.....	8
2.2.2 Comunicación Inalámbrica	8
2.2.2.1 Protocolos de comunicación	8
2.2.3 Tecnología LoRa.....	10
2.2.3.1 Arquitectura de LoRa	11
2.2.4 LoRaWAN.....	11
2.2.4.1 Clases LoRaWAN.....	11
2.2.5 Wireless Sensor Network	12

2.2.6 topología de redes de sensores.....	12
2.2.6.1 Red estrella.....	12
2.2.6.2 Red de Malla.....	12
2.2.6.3 Red Árbol.....	13
2.2.7 Sensores.....	13
2.2.8 Microcontrolador.....	13
2.2.8.1 Microcontrolador ESP32.....	13
2.2.9 Motores eléctricos.....	13
2.2.9.1 Clasificación de motores eléctricos.....	13
2.2.10 Controlador Lógico Programable.....	14
2.2.11 HMI.....	14
2.3 Marco Teórico.....	15

CAPÍTULO III

3. Desarrollo.....	16
3.1 Enfoque.....	16
3.2 Diseño de la Propuesta.....	16
3.3 Componentes Físicos.....	17
3.3.1 PLC s7-1200.....	17
3.3.2 Variador de frecuencia.....	17
3.3.2 HMI Delta.....	17
3.3.3 Raspberry Pi 4.....	17
3.3.4 Sensor de Corriente SCT-013.....	17
3.3.5 Sensor de Temperatura TIPO K.....	17
3.4 Componentes Lógicos.....	17
3.4.1 Tia Portal.....	17
3.4.2 Node-Red.....	18
3.4.3 Maria DB.....	18
3.4.4 IDE Arduino.....	18
3.4.5 DOPsoft.....	18
3.5 Desarrollo de la propuesta.....	18
3.5.1 Red de sensores inalámbricos.....	18
3.5.2 Comunicación inalámbrica de LoRa esp32.....	19
3.5.3 Configuración del LoRa esp32 transmisor.....	19
3.5.4 Diagrama Eléctrico de la conexión del esp32 con el sensor de corriente SCT-013 y el sensor de temperatura tipo K.....	19
3.5.5 Conexión de la Raspberry pi 4 con el esp32.....	19
3.5.6 Gestión de la base de datos.....	20
3.5.7 Arquitectura de la comunicación de la Raspberry pi 4 y el PLC... ..	20
3.5.8 Instalación de Node-red en la Raspberry pi 4.....	20
3.5.8.1 Interfaz de Node-Red.....	20
3.5.9 Programación de bloques en tia portal.....	21
3.5.9.1 Programación del arranque del motor.....	21
3.5.10 Comunicación entre el HMI Y PLC.....	23
3.5.11 Arquitectura de la Interfaz HMI.....	23
3.5.11.1 Pantalla de control del sistema.....	24

3.5.11.2 Pantalla Dashboard para el control de corriente y temperatura	24
3.5.11.3 Pantalla de la Ubicación de los equipos	24
3.5.12 Evaluación preliminar de las métricas de la calidad de señal de LoRaWAN	25

CAPÍTULO IV

4. Análisis de Resultados.....	26
4.1 Pruebas	26
4.2 Resultados.....	27
4.2.1 Precisión de los sensores	27
4.2.2 Análisis de la potencia de la señal	28
4.2.3 Análisis de la Conexión de los Equipos de Control.....	30
4.3 Resultados de los Objetivos	32
Conclusiones	33
Recomendaciones	34
Referencias	

Índice De Figuras

1. Modelo OSI.....	10
2. Capa física.....	11
3. Topología LoRaWAN.....	11
4. Arquitectura.....	16
5. Arquitectura de una red de sensores Inalámbricos	18
6. Tecnología LoRa	19
7. Conexión del esp32 con los sensores	19
8. Conexión de la Raspberry con el esp32.....	20
9. Arquitectura de la comunicación de la Raspberry pi 4 y el PLC.....	20
10. Declaración de variables	21
11. Configuración de node-red.....	21
12. Arranque de un motor con FLIPFLOP.....	22
13. Comparadores para el rango de nuestros sensores.....	22
14. Comunicación entre el HMI Y PLC	23
15. Distribución de datos.....	23
16. Interfaz Principal.....	24
17. Dashboard de los sensores	24
18. Ubicación de los diferentes motores en la piscina	25
19. Tablero de control	26
20. Sistema de Comunicación Inalámbrica.....	27
21. Raspberry Pi 4.....	27
22. Relación de la potencia de señal y la distancia entre los LoRa.....	29
23. Relación de la Distancia y el tiempo de comunicación entre los LoRa ..	30
24. Gráfica del tiempo de respuesta del sistema	31

Índice De Las Tablas

1.Tabla 1 de las bandas de frecuencia.....	10
2.Tabla 2 de Precisión de los sensores	28
3.Tabla 3 de Resultados del LoRa Transmisor y Receptor.....	29
4.Tabla 4 de Resultados del Tiempo de los equipos de control.....	31

Índice De Anexos

1. HMI.....	38
2. Programación de alarmas	38
3. Interconexión de los equipos de control.....	39
4. Movimiento de los datos del PLC al HMI.....	39
5. Programación de las Luces Piloto	39
6. Historial de alarma	40
7. Tiempo de respuesta de Comunicación de node red.....	40
8. Tiempo de respuesta de Comunicación del LoRa Emisor y Receptor.....	41
9. Valor del pírometro en un ambiente controlado.....	41
10. Pírometro y Sensor	42
11. Código LoRa emisor a.....	42
12. Código LoRa emisor b.....	43
13. Código LoRa Receptor a.....	43
14. Código LoRa Receptor b.....	44
15. Código LoRa Receptor c.....	44
16. Código LoRa Receptor d.....	45

Capítulo I

1. FUNDAMENTACIÓN

1.1. Antecedentes

En la actualidad, las tecnologías han permitido la implementación de diversos sistemas de comunicación y control, ofreciendo varias soluciones para los entornos industriales para el área de producción. Un ejemplo claro son los motores, bombas y sistemas de control, ampliamente utilizados tanto en entornos industriales de producción como las camaronearas, debido a su fácil instalación y baja demanda energética. Los motores son máquinas eléctricas que son fundamentales para aplicaciones de baja, media y alta potencia utilizados para procesos industriales, son esenciales para las cadenas de producción, por ello requieren un sistema de monitorización y supervisión para garantizar que su operación sea continua, segura y confiable [1].

Con el paso del tiempo, se han presentado diversos trabajos utilizando tecnología internet industrial de las cosas (IIoT), el cual se refiere a la implementación de la tecnología IOT enfocado en ámbitos industriales, el cual ha dado lugar a sistemas capaces de comunicarse y compartir información de manera rápida y segura, el cual respecta a la incorporación de control de sensores y dispositivos utilizando tecnologías basadas en la nube, permitiendo una mayor automatización, monitoreo y control en tiempo real [2]. La Industria 4.0 ha impulsado una transformación significativa en la manera en que las industrias operan, integrando tecnologías avanzadas que permiten la interconexión de sistemas y máquinas industriales a través de redes inteligentes [3].

En el transcurso del tiempo, se han desarrollados diversos trabajos y aplicaciones para poder mejorar la gestión de los procesos, entre ellos esta LoRa y LoRaWAN que permiten el monitoreo y gestión de procesos industriales [4]. LoRa (Long Range) se ha posicionado como una solución clave para las redes de área de mediano y largo alcance, además ofrece comunicaciones confiables a largas distancias, con un consumo de energía extremadamente bajo, lo que la hace ideal para aplicaciones IoT e IIoT [5]. Para la implementación del sistema de monitorización, se pueden emplear diversos controladores como la raspberry, ESP32, arduino, etc, debido a su compatibilidad y escalabilidad, el cual con la tecnología LoRa se lo utiliza para acceder a los datos recopilados que transmitan los diversos sensores conectados [6].

La universidad salesiana desarrolló un sistema de monitoreo y control de un motor trifásico, que es accionado por un PLC y un HMI desarrollado por PYTHON, el cual demuestra que la interfaz HMI son esenciales para la gestión en los sistemas de control, permitiendo la supervisión y control del motor desde una estación de monitoreo central, además te permite agregar varias funciones para visualizar los parámetros del motor [7].

Se presentó un artículo que demuestra que los controladores lógicos programables (PLC), han sido utilizados durante mucho tiempo con el fin de poder desarrollar interfaces para mejorar los procesos en la industria, este sistema tiene la facilidad de integrar tecnologías como LoRa, que son capaces de integrar sensores gracias a su escalabilidad de la red obteniendo una gestión remota que pueden ayudar a reducir los costos laborales y aumentar la productividad [8].

La renovación de tecnologías que aplican las industrias es una necesidad nacional como internacional, por ello en la Universidad Nacional Mayor de San Marcos presentó un trabajo que implementaron un sistema utilizando tecnología de bajo costo y de programa abiertos como es el IOT UBIDOTS y un microcontrolador NANO-PLC, obteniendo un sistema de monitoreo remoto sin la necesidad de estar presente en la industria, el cual abarca las necesidades del mercado que provocan la renovación del sistema, con el fin de mejorar la producción y reducir costo [9].

La Universidad Continental desarrolló un prototipo empleando un módulo ESP32 y Atmega que adquiere datos de consumo de corriente, voltaje y temperatura del motor de inducción a través de sensores, el cual envía los datos a un servidor IoT de Ubidots, y permite un monitoreo de los parámetros obtenidos [10], este proyecto abarca puntos considerables para el desarrollo de nuestro proyecto.

En la universidad tecnológica de Perú se desarrolló un sistema para el monitoreo y control de motores monofásicos en aplicaciones de media distancia. Este sistema fue diseñado para operar en entornos industriales, utilizando módulos de comunicación LoRa y sensores, capaces de medir parámetros como la temperatura y el consumo eléctrico, además ofrecía una interfaz “visual Basic” para observar los parámetros del motor y facilitar la toma de decisiones respecto a su control.

En la Escuela Superior Politécnica de Chimborazo se presentó un trabajo de investigación, el cual implementan un sistema remoto para control y monitoreo mediante un sistema de comunicación utilizando un Gateway IIoT, que permite el enlace a una conexión de una red virtual, el cual será esencial para la conexión entre la interfaz HMI programado en HTML [11], este trabajo nos presenta la escalabilidad de la industria 4.0.

Con estos antecedentes tenemos información sólida para poder realizar el presente proyecto, el cual se enfoca en implementar una red de sensores inalámbricos empleando tecnología LoRa mediante un módulo ESP32, el cual se encargará de leer parámetros del sensor de corriente, temperatura, controlador de velocidad y también se utilizará un módulo de entrada y salida, además se empleará un Raspberry pi 4 con el fin de poder procesar la información para transmitir la información a un PLC, que se encargará de poder controlar el sistema leyendo los parámetros obtenidos del motor. Se implementará una interfaz HMI interactiva que va a permitir el monitoreo y control de los datos obtenidos debido a las funciones que se incorporarán, el sistema se lo simulará en un ambiente controlado, el cual será en el laboratorio de la UPSE.

La industria cada vez tiene mayor demanda de producción, por ello está obligado a implementar tecnologías con una mayor optimización en costos y bajo consumo energético, estas tecnologías son escalables para el desarrollo de trabajos futuros empleando tecnologías como controladores inteligentes que es un campo para la innovación e implementación de nuevos proyectos.

1.2. Descripción del Proyecto

En el presente trabajo de investigación, tiene como objetivo desarrollar un sistema capaz de monitorear y controlar parámetros del motor, por ello se implementara una red de sensores inalámbricos mediante un módulo ESP32 LoRa, capaz de recibir la información del sensor de corriente, temperatura y controlador de velocidad que estarán conectados a un módulo de entradas y salidas para la interacción de los sensores y el motor, para poder realizar la interconexión entre modulo LoRa y el controlador PLC se utilizara un microcontrolador “Raspberry Pi 4”, el cual nos permite utilizar un software de código abierto que será necesario para la comunicación con el PLC, además se utilizara una interfaz HMI amigable para el usuario, agregando funciones que permitirán monitorizar y controlar el estado del motor.

En conjunto, este proyecto integrara tecnologías innovadoras de IIoT industrial y telecomunicaciones, proporcionando una solución moderna y segura para el control de motores, con aplicaciones tanto en entornos industriales como en la educación.

1.3. Objetivos del proyecto

Objetivo Principal

- Desarrollar una red de sensores inalámbricos para el monitoreo y control de un motor mediante tecnología LoRa e IIoT.

Objetivos específicos

- Implementar un sistema de comunicación inalámbrica, utilizando un módulo ESP32 para la adquisición de datos de los sensores de corriente, temperatura y control de velocidad para el monitoreo del estado del motor.
- Integrar los equipos de control PLC, Raspberry Pi utilizando un protocolo de comunicación LoRa para el procesamiento y transmisión de datos, y utilizar un HMI para la visualización y gestión del sistema.
- Simular escenarios de la vida real que puedan afectar a los equipos para validar la operatividad del sistema, utilizando equipos de monitoreo y control dentro de un ambiente controlado.
- Desarrollar una interfaz HMI intuitiva, utilizando funciones que permita a las industrias o usuario monitorizar y controlar parámetros del estado del motor.

1.4. Justificación

La reciente acogida de sistemas inalámbricos para el monitoreo en las industrias, enfocados en motores y accionamientos eléctricos el cual consumen alrededor del 45 % de la generación de energía en industrias de producción, pero, debido a que si las maquinarias no mantiene en su consumo energético este valor varia entre un 5 % al 10 % de energía, afectando la productividad e ingresos de la industria [12].

En las industrias de producción como las camaroneras la gestión de monitorear y controlar motores, ya sean monofásicos, trifásicos o bombas es algo imprescindible, ya que son esenciales en diversos procesos industriales y requieren supervisión constante para prevenir fallas eléctricas y mecánicas, que pueden afectar tanto la productividad como la seguridad operativa.

La camaronera Santa Priscila es conocido como uno de los productores y exportadores de camarones del país, el cual cuenta con 58 fincas y más de 18 hectáreas de piscinas y el personal debe movilizarse por grandes distancias para el encendido, apagado y monitorizar el estado de los equipos, esto no solo genera pérdida de tiempo, sino que perjudica el proceso de producción de la camaronera[13].

El uso de tecnologías como LoRa ofrece una solución de bajo consumo energético, segura y económica para la transmisión de datos a largas distancias, lo que es ideal para sistemas de monitoreo remoto en entornos industriales, además, este sistema enfocado en un área extenso como las camaroneras facilita el proceso tanto de la industria como del operador. Este proyecto tiene como finalidad desarrollar un sistema de comunicación

inalámbrica al incorporar el Internet Industrial de las Cosas (IIoT), se facilita la recolección y análisis de datos en tiempo real, lo que permite tomar decisiones preventivas y correctivas en el mantenimiento de los equipos y mejorando la operación de los motores. Además, el proyecto no solo aborda una necesidad industrial, sino que también fomenta el desarrollo de habilidades técnicas en los estudiantes. A través de la configuración y programación de equipos de control y comunicación, los estudiantes adquieren competencias en tecnologías innovadoras como LoRa, IIoT y sistemas de control de motores, integrando conocimientos de electrónica, telecomunicaciones y automatización.

1.5. Alcance

El presente proyecto se enfoca en desarrollar un sistema de comunicación para la gestión y control de un motor, el cual será en un laboratorio de la universidad el que simulará un ambiente controlado, además, los programas que se emplearan son de accesos libres, por ello los siguientes puntos van a definir el alcance del presente proyecto:

1.5.1. Desarrollo de un sistema de comunicación inalámbrico.

- Implementar una arquitectura LoRa utilizando el módulo ESP32, para desarrollar una red de sensores inalámbrico en un ambiente controlado.
- Configurar los sensores de corriente, temperatura y controlador de velocidad que serán parámetros del motor para su gestión y control.

1.5.2. Integración de las tecnologías Raspberry Pi 4, PLC y HMI.

- Configurar y programar un PLC para la recepción de datos enviados por la red de sensores inalámbricos mediante tecnología LoRa.
- Configurar y programar la Raspberry pi 4 para el procesamiento y envío de datos recibidos a la interfaz HMI.

- Configurar la interfaz HMI para visualizar los datos en tiempo real y que permita interacciones manuales.

1.5.3. Simulación de diversos escenarios en la gestión del motor en un laboratorio.

- Realizar pruebas de funcionamiento bajo condiciones que simulen escenarios de las industrias en el laboratorio de la UPSE.
- Documentar la información para analizar la operatividad de los equipos y realizar ajustes del sistema de gestión y control del motor.

1.5.4. Desarrollo de una interfaz HMI para la gestión y control de un motor.

- Diseñar y configurar una interfaz HMI amigable haciéndolo fácil de operar para la gestión de parámetros del motor.
- Implementar funciones de control para las acciones de situaciones que presenten alertas del motor basándose en los datos obtenidos.

1.6. Metodología

Las metodologías empleadas en este proyecto fueron seleccionados para poder facilitar la comprensión del presente proyecto.

1.6.1. Metodología bibliográfica.

Este método es una técnica que se enfoca en poder realizar una investigación con el fin de recopilar información de distintas fuentes bibliográficas como proyectos de titulación, artículos científicos, páginas web y revistas, lo que nos permitirá referenciar el siguiente trabajo de investigación.

1.6.2. Investigación Descriptiva.

El presente proyecto, se enfoca en una implementación de red de sensores inalámbricos, el cual es aplicado a un monitoreo y control de un motor para mejorar la eficiencia de la operatividad, además del uso de tecnología LoRa e IIot, permite a los sensores transmitir información a largas distancias, haciéndolo ideal para situaciones de la industria donde la comunicación de los equipos es esencial.

1.6.3. Investigación Experimental.

En este proyecto, se implementará una red de sensores inalámbricos para el monitoreo y control de un motor en un ambiente controlado, para poder gestionar parámetros esenciales del estado del equipo como la corriente, temperatura y control de velocidad. Además, se implementarán diversas tecnologías para lograr la comunicación de largas distancias como LoRa e IIoT, a este sistema se le realizarán diversas simulaciones, con el fin de ver la efectividad y rendimiento de este sistema al emplear diversos sensores y tecnologías.

1.6.4. Investigación Aplicada. Esta metodología, busca aplicar los conocimientos

del sistema de monitoreo y control en un campo industrial ambientado a un entorno controlado, aplicado a la implementación de una red de sensores inalámbrico para el monitoreo y control de un motor por medio de una interfaz HMI.

1.7. Resultados Esperados.

Un sistema de comunicación capaz de abarcar los puntos considerados en este proyecto, además de obtener un sistema capaz de poder ser escalables para futuros trabajos de investigación.

1.7.1. Desarrollo de un sistema de comunicación inalámbrico

- **Eficiencia en la recopilación de datos:** Los sensores de corriente, temperatura, controlador de velocidad deberán ser capaces de obtener parámetros esenciales del motor.
- **Comunicación Eficaz:** La transmisión de datos deben enviarse sin pérdida de información, debido a que la tecnología LoRa permite la comunicación de larga distancia sin interferencia.

1.7.2. Integración de las tecnologías Raspberry Pi 4, PLC y HMI

- **Integración:** Implementación de un protocolo de comunicación, para la exitosa integración de las tecnologías Raspberry Pi, PLC y HMI con el fin de obtener un intercambio de datos de manera fluida.
- **Optimización:** Se espera obtener un sistema capaz de poder recibir datos de manera automática sin la intervención humana.

1.7.3. Simulación de diversas condiciones del motor en un laboratorio.

- **Simulación:** Se espera un sistema capaz de poder responder de manera eficaz, simulando diversos escenarios que presentaría un motor en un sector industrial en un ambiente controlado.

1.7.4. Desarrollo de una interfaz HMI para el monitoreo y control de un motor

- **Operatividad:** Los usuarios deben poder manejar el sistema, debido a la integración de funciones del monitoreo y control del motor, que serán capaces de hacer al sistema intuitiva y fácil de operar.

CAPITULO II

2. Marco Referencial

2.1 Marco Contextual

Este trabajo proporciona un marco contextual sobre la industria camaronera, donde el uso de tecnologías de la industria 4.0 o también llamado la era de la digitalización son fundamentales, el cual ha sido el avance o crecimiento de las tecnologías de la información y comunicación (TIC). La cuarta revolución industrial integra sistemas físicos, digitales y biólogos para poder crear una red de producción inteligente, además, no solamente estos equipos funcionan de manera independiente, sino que pueden lograr trabajar en conjunto de manera inteligente[14]. Actualmente, las industrias de producción buscan implementar tecnología que involucren la conectividad entre maquinas, sensores e interacción humana[15]. La conectividad entre máquinas y sensores permite recopilar, analizar y compartir datos de manera eficiente, mejorando la toma de decisiones y reduciendo tiempos de inactividad. Por otro lado, la interacción humana sigue siendo esencial, ya sea para supervisar procesos o intervenir en momentos críticos del sistema.

El presente proyecto se lo realizará en un laboratorio de la Universidad Estatal Península de Santa Elena simulando un ambiente controlado, el desarrollo e implementación de una red de sensores inalámbricos para un sistema de monitoreo y control de un motor para mejorar su eficiencia operativa, a través de datos obtenidos por los sensores de temperatura, corriente y control de velocidad, con el fin obtener un estudio para aplicaciones relacionadas al área industrial.

2.2 Marco Conceptual

En esta sección se detallan la estructura teórica de los equipos y sistemas que se utilizaran en este proyecto.

2.2.1 Redes Inalámbricas

Las redes inalámbricas tienen el propósito de sustituir las redes cableadas, el cual se divide en las siguientes categorías:

- Redes WPAN
- Redes WLAN
- Redes WMAN
- Redes WWAN

Pero, las redes inalámbricas también se dividen en dos grandes grupos de corto alcance que son las redes de área local (LAN), y las de largo alcance que son las redes de área personal (WMAN).

2.2.1.1 Redes LAN

Las redes inalámbricas LAN, son utilizadas en áreas de corto alcance donde las redes están muy limitadas a su área, como son los edificios, colegios, universidades o casas, de igual manera que las redes de área personal (PAN), que son utilizadas por las laptops

o pc redes donde debe estar cerca para poder comunicarse entre sí. Estas utilizan unas bandas de frecuencias no licenciadas de 2,4 GHz y 5 GHz, estas permiten poder operar redes inalámbricas sin tener una licencia para poder utilizarlas y expandirlas este tipo de redes [16].

2.2.1.2 Redes WPAN

Las redes inalámbricas WPAN, permite una comunicación de corto alcance con una infraestructura de unos 10 metros de distancia. Además, sus principales características son el bajo consumo de energía y una baja velocidad de transmisión. Las tecnologías que utiliza las redes WPAN se basan en Bluetooth, IrDa, ZigBee o UWB [16].

2.2.1.3 Redes WLAN

Las redes inalámbricas de área local o WLAN, se caracterizan en lograr una comunicación de hasta 100 metros un claro ejemplo sería el hogar, las instituciones educativas y oficinas de trabajos. Además, las redes WLAN se basan en el estándar IEEE 802.11 o mas conocidos como las redes Wi-Fi [16].

2.2.1.4 Redes WMAN

Las redes inalámbricas de área metropolitana o WLAN se basa en el estándar IEEE 802.16, el cual es una tecnología con una alta velocidad de transmisión de datos que se rige por una arquitectura de punto a multipunto. El protocolo de comunicación IEEE 802.16 o WiMAX (Worldwide Interoperability for Microwave Access), es parecido al protocolo de Wi-Fi, pero destinado a distancias que abarcan mayores distancias como son las oficinas o Hot spots. Además, trabaja en dos bandas de frecuencias con y sin licencias de 2 GHz, 11 GHz y de 10-66GHz [16].

2.2.1.5 Redes WWAN

Las redes inalámbricas de área amplia o WWAN utilizan frecuencias con licencia en un rango a partir de los 50 Kilómetros, además, la arquitectura de estas redes es a grandes distancias como ciudades incluso países, esto lo logran debido a la comunicación que tienen con los satélites o antenas. Las WWAN se las aplica en dos tecnologías como la telefonía móvil y los satélites [16].

2.2.2 Comunicación Inalámbrica En la actualidad, existe múltiples tecnologías que son inalámbricas, el cual permiten transmisión de datos que pueden ser de corto o larga distancias, por ello se va a describir los protocolos mas importantes de las redes inalámbricas:

2.2.2.1 Protocolos de comunicación Es un sistema que permite la interconexión de dos o más equipos, con el fin de transmitir la información de un lugar a otro. Los protocolos de comunicación pueden ser implementados por hardware o software [17].

ZigBee:

Es un protocolo de comunicación inalámbrica IEEE 802.15.4, el cual opera en redes inalámbricas de cota distancia y baja velocidad de datos, el rango de frecuencia es de 868 MHz, 915 MHz y 2.4 GHz, además, permite el uso de esta tecnología en varias regiones del mundo [18].

Z-wave:

Es una tecnología de red mallada conformada por una comunicación inalámbrica robusta, bidireccional y segura, además, las aplicaciones de Z-Wave incorpora un control y gestión remoto de tecnologías Smart Home, Honeywell y Schneider Electric[19].

Lora:

Es una modulación de radio, esta tecnología esta regulada por LoRa Alliance, su arquitectura está conformado por gateways, estas sirven como acceso para plataformas IoT[20].

Wifi:

El protocolo de comunicación Wi-Fi 802.11 es una tecnología desarrollado por el instituto de ingenieros eléctricos y electrónicos (IEEE), el cual es considerado un estándar para la comunicación de redes inalámbricas, además, Wifi permite integrar topologías ad-hoc, estrella, árbol y además, ofrece una mayor velocidad en la transmisión de datos.

Sigfox:

Es una tecnología basada en el estándar ETSI LTN, opera en bandas de frecuencia sub-GHz libres [20].

Bluetooth:

La tecnología Bluetooth se basa en el estándar IEEE 802.15, el cual está enfocado para comunicaciones omnidireccionales de corto alcance [20].

Modbus TCP

El protocolo de comunicación Modbus, permite la conectividad sobre redes Ethernet, el cual aumenta el grado de conectividad. La transmisión de paquetes TCP/IP (puerto del sistema 502, identificador asa-appl-proto), este protocolo permite el encapsulamiento de la trama del protocolo Modbus, de esta manera Modbus TCP se puede utilizar en internet [21].

PROFIBUS

Es un protocolo de comunicación más utilizada en las industrias, el cual logra cubrir una gran rama de aplicaciones, además, hace posible la interconexión de dispositivos digitales de baja y media potencia con sensores PLC, HMI [22].

MQTT

Es un protocolo que se lo emplea donde se requiera comunicacion Máquina a Máquina o mas conocido como M2M en entornos donde se utiliza IoT, y esta basado en un protocolo TCP/IP, ademas, la arquitectura que emplea es una topologia estrella. La ventaja de utilizar este protocolo es que no demanda demasiada potencia, lo cual lo hace ligero haciendolo ideal para redes IoT. También garantiza la calidad del servicio (QoS) y utiliza TCP/IP en la capa de transporte, este protocolo utiliza la seguridad SSL/TLS, el cual garantiza una entrega confiable de datos[23].

2.2.3 Tecnología LoRa

Lora es una tecnología de radiofrecuencia, el cual permite comunicación de largas distancia con bajo consumo de energía eléctrica, además, esta tecnología su frecuencia de trabajo es por debajo de 1 GHZ como se observa en la siguiente tabla . [24].

Tabla 1 de las bandas de frecuencia

Región	Banda
1 Estados Unidos y America	va desde los 900 MHz a 928 MHz
2 Europa	va desde los 860 MHz a 870 MHz
3 China	va desde los 779 MHz a 787 MHz

La tecnología LoRa permite los siguientes tipos de Flujo de datos:

- **Simplex:** es una comunicación unidireccional, donde un radio solo transmite datos y las otras N radios solo van a recibir datos.
- **Half-dúplex:** es una comunicación bidireccional, pero no de manera simultánea, es decir, las radios pueden transmitir datos, pero no recibirlos en ese periodo de tiempo y una radio puede estar recibiendo, pero no transmitir datos.
- **Full dúplex:** Es una comunicación bidireccional, todas las radios pueden enviar y recibir datos de manera simultánea.

En la siguiente figura, podemos observar la estructura de capas de red, el cual según el modelo OSI se observa tres capas de una trama de datos del protocolo LoRa.

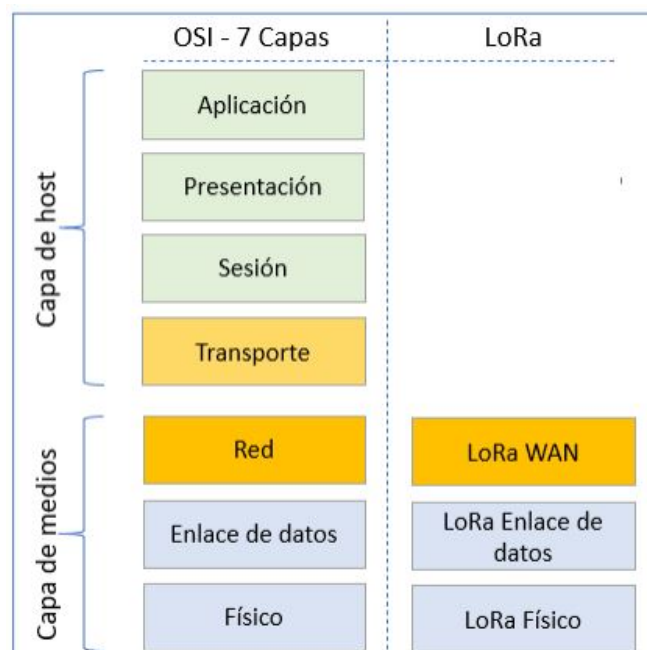


Figura 1: Modelo OSI [24]

El protocolo de comunicación LoRa en su capa física se estructura de la siguiente manera

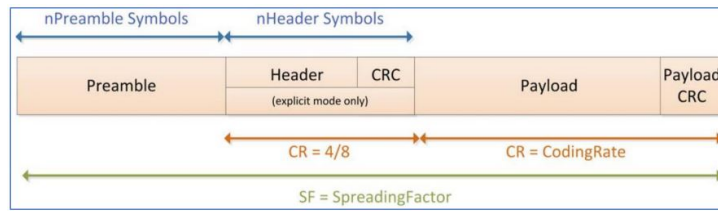


Figura 2: Capa física [24]

2.2.3.1 Arquitectura de LoRa La tecnología LoRa tiene topología de estrella y punto a punto, esto depende de la cantidad de radios LoRa que tiene el enlace.

- **Nodos clientes:** Son los dispositivos que se conectan que son capaces de recolectar información de sensores.
- **Gateway:** Es un dispositivo o estación en una red LoRa el cual es el que recibe la información que es transmitida por los clientes hacia los servidores.
- **Servidores de red:** Son los que se encargan de la recepción y procesamiento de lo que viene del Gateway.

2.2.4 LoRaWAN El protocolo de red LoRaWAN (Long Range Wide Area Network) permite el uso de la capa física de la tecnología LoRa, con el fin de administrar y comunicar dispositivos LoRa, para establecer una comunicación inalámbrica bidireccional de manera segura, bajo costo y consumo energético [25].

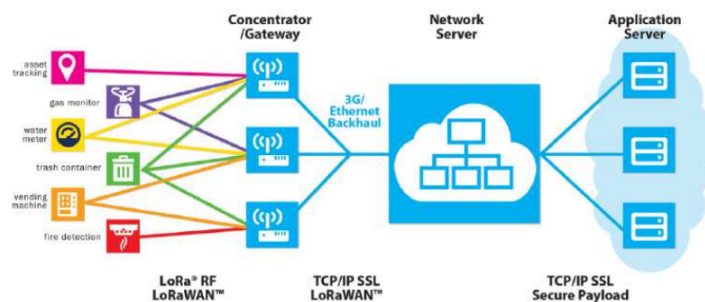


Figura 3: Topología LoRaWAN [25]

2.2.4.1 Clases LoRaWAN La comunicación LoRaWAN se divide en tres clases, esto depende de la funcionalidad entre los dispositivos.

Clase A: En esta clase la transmisión depende del dispositivo final, debido a que la recepción está permitida siempre y cuando la transmisión esté correcta, esta clase está conformada por dos ventanas separadas de recepción, con el fin de que, si la primera ventana recibe completamente los datos transmitidos, entonces la segunda ventana de recepción se cierra. Esta clase está diseñada para aplicaciones que solo envíen datos [26].

Clase B: En esta clase la transmisión no está limitada debido a que permite la creación de varias ventanas de recepción, sin la necesidad de que se haya terminado una transmisión anterior, pero esto produce un aumento en el consumo energético [26].

Clase C: La clase c está abierta al modo de recepción constante, a menos que se produzca una transmisión, esta clase de dispositivos presenta la mejor latencia entre los nodos finales y Gateway, pero con un mayor consumo energético[26].

La seguridad de LoRaWAN se establece en tres claves de seguridad por el cifrado y la autenticación de red los cuales son:

- La clave de sesión de red
- La clave de sesión de aplicación
- La clave de aplicación

2.2.5 Wireless Sensor Network Las redes de sensores inalámbricos o WSN (Wireless Sensor Network), son dispositivos que recolectan información como temperatura, humedad, vibraciones y otras condiciones ambientales, además, estos equipos interconectados se definen como nodos. El desarrollo de las WSN requiere tecnologías de hardware y software, cada nodo de sensores tiene la capacidad de poder recolectar datos para poder enrutarlos hacia un nodo recolector [27].

2.2.6 topología de redes de sensores

Para la industria hay varias arquitecturas que son utilizadas para aplicaciones WSN, en la actualidad, conocemos tres topologías que son: estrella, malla y árbol [27]. Las topologías como todos los sistemas presentan ventajas y desventajas que se los describirá en los siguientes puntos.

2.2.6.1 Red estrella

En la topología estrella, la comunicación entre los nodos y el Gateway o nodo principal es directa, debido a este tipo de red el rendimiento dependerá de la cantidad de nodo que se conecte.

Las ventajas de usar esta topología es la poca latencia y la red es más sencilla, el cual da una facilidad en la implementación. La desventaja mas grande es que si falla el nodo central se cae la red, además de la posibilidad de colisiones entre los nodos [28].

2.2.6.2 Red de Malla

La topología malla o red Mesh, nos permite utilizar una conexión de múltiples nodos, facilitando la interconexión de diferentes caminos de transmisión si se llega a caer un nodo.

Las ventajas de utilizar esta topología es la transmisión de datos que es de manera continua, en caso de tener fallos busca una ruta alterna, a pesar de esto no necesita muchos Gateway para alcanzar una gran escalabilidad. Las desventajas es que presenta demasiada probabilidad de muchas colisiones, alta latencia de la red y tiene un desarrollo costoso [28].

2.2.6.3 Red Árbol

La comunicación entre los nodos depende del nodo coordinador, el cual se encargará de lograr la conexión al nodo principal o Gateway.

Las ventajas de esta red tienen una alta escalabilidad y bajo probabilidad de haber colisiones entre los nodos. Las desventajas son costosas si a la red se le integran routers, además, si un router presenta fallas una gran parte de la red va a caer [28].

2.2.7 Sensores Los sensores son dispositivos encargados de tomar y transmitir medidas cuantitativas, son capaces de captar los cambios de corriente o temperaturas dependiendo del tipo de funcionamiento y transformarlo en señales eléctricas presentándose en forma digital para su envío a otro dispositivo o a una interfaz capaz de procesar sus datos digitales [29].

2.2.8 Microcontrolador

Los Microcontroladores son circuitos integrados que internamente tienen una unidad central de procesamiento, memoria para código y datos, además, su principal desempeño es con funciones de monitoreo y control a través de sensores y actuadores [30]. En la actualidad, los microcontroladores son fundamentales para diversos sistemas digitales, el cual gracias a su versatilidad proporcionan diversas funciones que pueden ser aplicados a diferentes ramas de la ingeniería. [31].

2.2.8.1 Microcontrolador ESP32

En el año 2016, la empresa Espressif Systems desarrolló un microcontrolador ESP32, para ello utilizaron su versión pasada el ESP8266, al integrar un microcontrolador de 32 bits, conexión bluetooth y wifi, obtuvieron un mejor procesamiento y conectividad, el cual permite a los innovadores implementar este dispositivo debido a las ventajas en el rendimiento a adaptarse a proyectos en el área de telecomunicaciones, monitoreo, control y automatización [32].

2.2.9 Motores eléctricos

Los motores son máquinas eléctricas cuya función es convertir la energía mecánica en eléctrica o viceversa, la diferencia es que para convertir energía mecánica a eléctrica el dispositivo se lo llama generador, y para convertir de energía eléctrica a mecánica se llama motor. La mayoría de los generadores y motores convierten su energía a través del campo magnético. Actualmente, estos equipos están en la mayoría de los equipos eléctricos de los hogares y de las industrias, como son las refrigeradoras, ventiladores, las bandas transportadoras, ascensores, etc [33].

2.2.9.1 Clasificación de motores eléctricos

Los motores se pueden clasificar dependiendo del tipo de alimentación, el cual se lo divide de la siguiente manera.

- **Motores DC:** Son motores de corriente directa principalmente denominados Monofásicos.
- **Motores AC:** Son motores de corriente alterna categorizados de inducción asíncronos y síncronos, comúnmente llamados trifásicos y en algunos casos monofásicos.

2.2.10 Controlador Lógico Programable

El controlador Lógico programable o PLC (Programmable Logic Controller), es un equipo electrónico programable cuya función es controlar, sincronización y funciones lógicas en tiempo real en un ambiente industrial. Los PLC tienen una gran ventaja, el cual es modificar la operación sin tener que modificar el sistema de control o las conexiones de los dispositivos de entrada y salida [34].

El controlador lógico programable PLC S7-1200 de la marca de Siemens ofrece múltiples operaciones para el área de automatización y control en diversos procesos industriales como:

- Climatización
- Calefacción
- Tratamientos de agua
- Accionamiento de bombas
- Controladores automáticos

2.2.11 HMI

El interfaz Hombre Maquina (HMI) es un dispositivo de control, el cual permite la interacción o comunicación entre ser humano y máquina, además, las acciones que tiene el HMI es la supervisión, control y visualización de los procesos industriales.

Hardware: Contiene una pantalla táctil para la integración de paneles de control, enfocados en la automatización, además, este equipo es muy robusto capaces de soportar escenarios complejos, como serían las condiciones climáticas o las áreas industriales.

Software: El HMI tiene un software capaz de traducir los datos de las maquinas o sensores en gráficos, para que puedan ser comprensibles y amigables para el usuario.

Características

- **Monitoreo:** Muestra los datos en tiempo real de dispositivos interconectados, capaz de recopilar información de los procesos como equipos, sensores y valores erróneos o alarma.
- **Control:** Los usuarios podrán operar realizando ajustes de los procesos desde la interfaz HMI.
- **Visualización:** La interfaz HMI proporciona diferentes formas de presentar los datos como gráficos, diagramas y otras representaciones visuales, que permite hacer una interfaz amigable con el usuario.

2.3 Marco Teórico

En esta sección se presentan varios trabajos investigativos, que fueron seleccionados con el fin de aportar como guía al presente trabajo de investigación.

En la tesis de “DESARROLLO DE UN TABLERO DEMOSTRATIVO PARA EL CONTROL DE SISTEMAS AIREADORES Y ALIMENTADORES SUPERVISADOS Y MONITOREADOS MEDIANTE RADIOFRECUENCIA POR MEDIO DE SISTEMA SCADA” realizado por J. Lombeida Vásquez y J. Samaniego Reyes en la Universidad Politécnica Salesiana en el año 2022, presentó un sistema supervisado y controlado utilizando un sistema SCADA, con el fin de poder tener una comunicación inalámbrica para controlar accionamientos de motores de forma remota, utilizando un PLC, Variador Danfoss y un Mikrotic STXsq 5 al cual es un equipo de comunicación inalámbrica de 5 GHZ hasta un alcance de 12 KM con un enlace de punto a punto y un protocolo de comunicación 802.11 ac [35].

En la tesis de “DISEÑO E IMPLEMENTACION DE UN PLC BASADO EN RASPBERRY PI Y NODE-RED PARA APLICACIONES INDUSTRIALES IOTT” realizado por Jorge David L., en la Universidad de Ricardo Palma en el año 2021, presenta un sistema con un consumo de 15W, además, un software de control que permite visualizar y controlar las entradas y salidas remotamente desde un dashboard[36].

En el artículo de RENDIMIENTO PARA LA INTEROPERABILIDAD ENTRE RASPBERRY PY, ESP8266 Y PLC CON NODE-RED PARA LA IIOT”, J. Torres Ventura, A. H. Ruelas Puentes, J.R. Herrera García, en la revista de Ciencia y Tecnología en el año 2023, presentó un trabajo llegando a la conclusión que la Raspberry pi y el microcontrolador ESP8266, son dispositivos confiables para el intercambio de datos a nivel industrial, además, el control de flujo de datos en una red distribuida mediante un gestor de protocolos como Node-RED es totalmente confiable y de ser muy intuitivo y amigable con el personal técnico y operativo.[37].

En el trabajo de investigación de “DESARROLLO DE UN PROTOTIPO DE SISTEMA DE TRANSMISIÓN DE BAJA POTENCIA PARA MEDICIÓN DE HUMEDAD Y TEMPERATURA EN UN CENTRO DE DATOS, USANDO EL PROTOCOLO LoRa” realizado por Juan Andrés M. L., en la Escuela Superior Politécnica del Litoral en el año 2020, presentó un sistema capaz de medir datos como la temperatura y humedad, desarrollando un pequeño prototipo que transmite datos usando un sistema de baja potencia empleando tecnología LoRa. [38].

En la tesis de “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL A DISTANCIA PARA EL CONTROL Y GESTIO DE MOTORES MONOFÁSICOS, EMPLEANDO TECNOLOGÍA LORA” realizando por Fabio Ernesto C. A., de la facultad de ingeniería Electrónica en la Universidad Tecnología de Perú en el año 2021, presentó un dispositivo de bajo consumo energético, que mide parámetros eléctricos de un motor utilizando un microcontrolador LoRa E44-TTL-100, obteniendo una interfaz gráfica de en visual Basic, pero en sus pruebas de comunicación de LoRa se obstruyen a partir de los 400 metro de distancia[39].

CAPITULO III

3 Desarrollo

3.1 Enfoque

El presente proyecto tiene un enfoque para una futura implementación en el área de las industrias camaroneras, debido a que es un sector con mucha demanda laboral y producción, por ello el área de mantenimiento debe contar con tecnologías que faciliten este proceso.

Uno de los principales procesos que tiene, es oxigenar la piscina, esto se lo realiza mediante aireadores que llevan motores eléctricos, pero el descuido de monitorizar los parámetros de estos equipos lleva a tener problemas e incluso se dañan debido al exceso de amperaje que consumen ya que son sometidos a una carga de trabajo alta y esto conlleva a un sobrecalentamiento en las bobinas del motor y posteriormente se dañan, remplazar o arreglar estos equipos demandan tiempo de trabajo que perjudican la producción de la camaronera.

3.2 Diseño de la Propuesta

Para ello se realizó el siguiente proyecto que se enfoca en utilizar tecnologías de comunicación inalámbrica de monitoreo en el laboratorio de la upse que será un ambiente controlado.

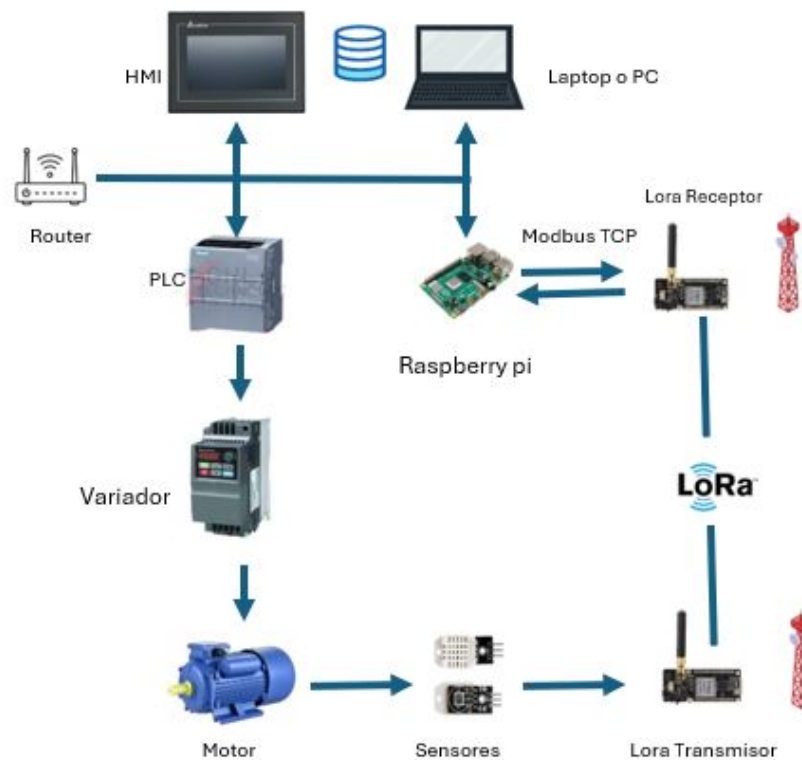


Figura 4: Arquitectura

3.3 Componentes Físicos

3.3.1 PLC S7-1200

El controlador lógico programable PLC S7-1200 de la marca de Siemens ofrece múltiples operaciones para el área de automatización y control en diversos procesos industriales como:

Se utilizó el PLC s7-1200 debido a su flexibilidad, capacidad de comunicación para diversas aplicaciones de comunicación, seguridad y su entorno de programación variada, además este PLC es muy comercializado para tableros de control, el cual lo hace ideal para este proyecto.

3.3.2 Variador de frecuencia

Para el control de velocidad del motor se utilizó un variador que permite modificar o alterar la frecuencia que emplea el motor en su giro rotacional.

3.3.3 HMI Delta

El equipo HMI que se utilizó para este proyecto es del modelo DELTA, el cual nos permite integrar funciones necesarias para nuestro sistema.

Se empleó el HMI Delta debido a su compatibilidad con una gran variedad de PLC, además, ofrece una comunicación flexible y conectividad en redes industriales, facilidad de programación e integra grandes funciones de monitoreo para el fin de este proyecto.

3.3.4 Raspberry Pi 4

La Raspberry es un mini CPU que es utilizada en diversos proyectos más enfocados en la electrónica, la automatización gracias a su versatilidad y escalabilidad en proyectos que requieran desarrollos de software, APIs e interfaces web.

3.3.5 Sensor de Corriente SCT-013

Los sensores de corriente son dispositivos capaces de trabajar como transformadores, el cual lleva la corriente que va a circular por el cable del motor y este actúa como un devanado primario y el sensor cuenta con un devanado internamente, debido a esto podemos obtener una relación de espiras con la corriente entregada, para obtener una carga se coloca una resistencia para obtener los parámetros establecidos además, cuenta con una precisión de medición de ± 1 .

3.3.6 SENSOR DE TEMPERATURA TIPO K

El sensor cuenta con dos modelos que comprenden el estado en seco y sumergible, para este proyecto emplearemos en seco, este sensor es muy versátil ya que su característica permite la medición de 400°C grados, además, cuenta con una precisión de medición de ± 0.5 °C.

3.4 COMPONENTES LÓGICOS

3.4.1 TIA PORTAL Para el desarrollo de la programación del controlador Lógico programable (PLC), se utilizó el software TIA PORTAL de SIEMENS.

3.4.2 NODE-RED

Node-RED es un software de acceso libre, el cual fue desarrollado con el propósito de conectar dispositivos o programas en las aplicaciones IIOT y la automatización en el área industrial. Node-RED en su software utiliza lenguajes de programación de JavaScript, que permite realizar diseños de flujos de datos y servicios mediante una visual web.

3.4.3 MARIA DB

MariaDB es un software de código abierto cliente-servidor, fue creado por Oracle Corporation, el cual es una base de datos, con el fin de poder almacenar y gestionar una gran cantidad de datos de manera eficiente. Además, tiene una compatibilidad con MySQL si se basa en funciones, comandos y APIs, el cual va facilitar la utilización de aplicaciones que utiliza MySQL.

3.4.4 IDE Arduino

Arduino es un software de código abierto compatible con los sistemas operativos de Windows, Linux y MacOS, además, este programa está basado en un lenguaje de programación Java. Arduino permite el envío y recepción de datos utilizando el puerto serial en una función integrada y llamada Monitor serial.

3.4.5 DOPsoft DOPsoft es un software de programación para desarrollar las interfaces de HMI para el control y monitoreo de diferentes procesos industriales, en este caso se empleará para la visualización de los sensores empleados en este proyecto, que serán de corriente y temperatura de un motor.

3.5 Desarrollo de la propuesta

En esta sección se procede a redactar el procedimiento de la propuesta, esto incluye configuración y conexión de los equipos y sensores por medio de los softwares.

3.5.1 Red de sensores inalámbricos

En la siguiente figura, podemos observar cómo se compone una red de sensores inalámbricos conformado por dispositivos y tecnologías que abarca una comunicación de larga distancia.



Figura 5: Arquitectura de una red de sensores Inalámbricos

3.5.2 Comunicación inalámbrica de LoRa esp32

En el presente proyecto se emplearon LoRa esp32, para el sistema de comunicación inalámbrica, se seleccionaron debido a su versatilidad que proporciona su comunicación de grandes distancias y de bajo consumo energético, además, trabajar con Lora esp32 permite la integración de sensores iot e iiot.



Figura 6: Tecnología LoRa

3.5.3 Configuración del LoRa esp32 transmisor

Para la programación del esp32 utilizaremos el software ide Arduino, ya que es un programa de código abierto diseñado especialmente para la configuración de microcontroladores. Arduino IDE gracias a su amplia gama de librerías y compatibilidad en varios sensores y actuadores ya que su lógica de programación es muy versátil y es muy fácil adaptarse a estos equipos.

3.5.4 Diagrama Eléctrico de la conexión del esp32 con el sensor de corriente SCT-013 y el sensor de temperatura tipo K

El Diagrama eléctrico se lo realizó en el software de proteus, debido a ser un programa que permite la interacción con varios componentes electrónicos, gracias a esto nos facilita la simulación y comprobación del funcionamiento de nuestros equipos.

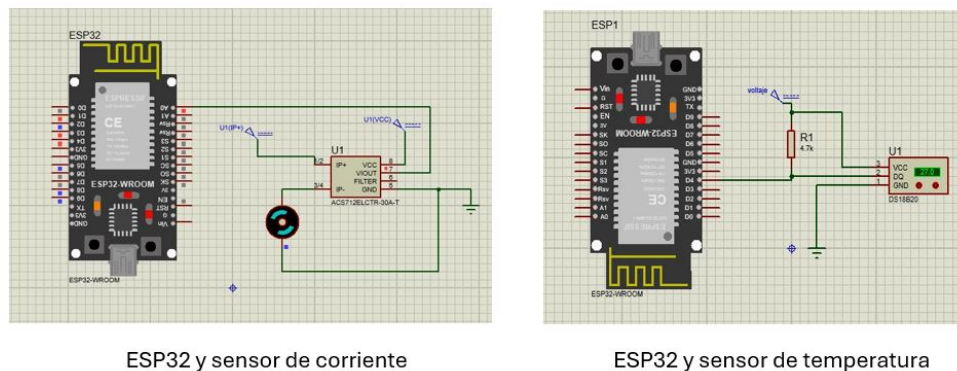


Figura 7: Conexión del esp32 con los sensores

3.5.5 Conexión de la Raspberry pi con el esp32

La comunicación entre la Raspberrypi y el esp32 utiliza el protocolo de comunicación Modbus TCP, debido a que este protocolo permite el envío de datos por una red Ethernet, además para aplicaciones de RTU donde se utiliza la comunicación inalámbrica. El protocolo Modbus RTU es un medio de comunicación para el envío y recibo de datos utilizando un PLC o controlador Lógico y un ordenador.

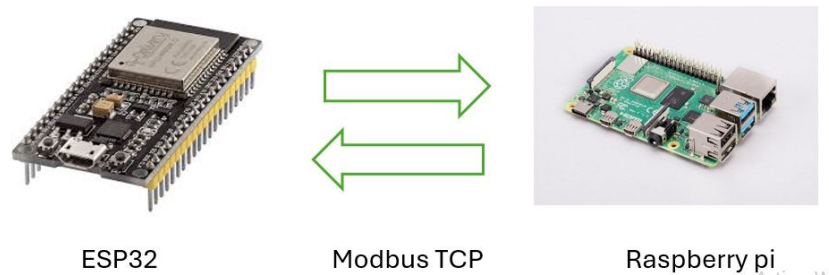


Figura 8: Conexión de la Raspberry con el esp32

3.5.6 Gestión de la base de datos

Se utilizó la base de datos Maria DB ya instalada en la Raspberry Pi 4, con el fin de almacenar datos recopilados de los sensores en una interfaz de comando CMD, esta integración permite llevar una gestión en los datos de manera autónoma para el monitoreo en tiempo real.

3.5.7 Arquitectura de la comunicación de la Raspberry pi 4 y el PLC

Esta arquitectura representa una transferencia de datos en tiempo real, para ello la asignación, validación e identificación de variables debe verificarse para lograr una comunicación fluida.

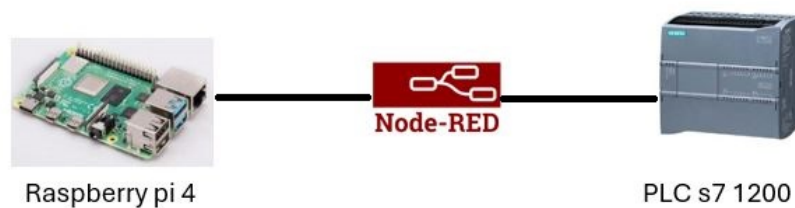


Figura 9: Arquitectura de la comunicación de la Raspberry pi 4 y el PLC

3.5.8 Instalación de Node-red en la Raspberry pi 4 Primero se va a instalar el software Node-red en la Raspberry pi 4, ya que es una herramienta de programación que facilita la interconexión de hardware, APIs e interfaces de página web, el cual se lo utilizara para el desarrollo de la propuesta, además, este servidor es el que se va a encargar de enlazar el controlador lógico PLC y la red de sensores inalámbricos de LoRaWAN conectados por la Raspberry pi 4.

3.5.8.1 Interfaz de Node-Red

La configuración de comunicación inalámbrica utilizaremos bloques o también llamados que permitan la comunicación modbus tcp para la interconexión de los equipos y obtener los datos que se enviaran.

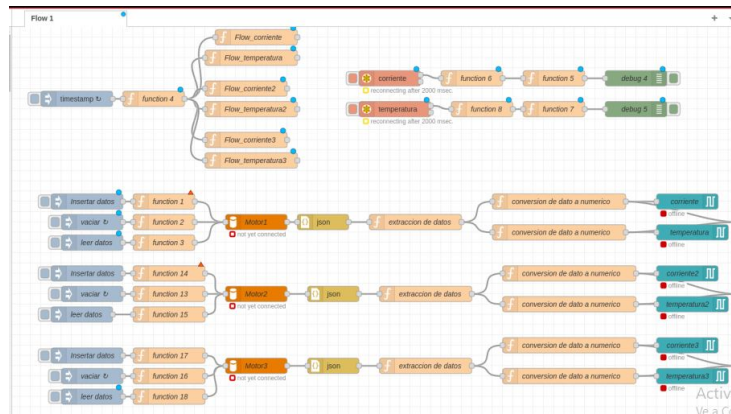


Figura 10: Declaración de variables

3.5.9 Programación de bloques en tia portal

Para el desarrollo del proyecto utilizaremos tia portal V16, para la creación de variables que nos permitirán la transmisión de datos que recibiremos de node-red y posteriormente enviarlo al HMI, además, en este programa realizaremos nuestro lenguaje de programación Ladder que nos garantiza la simulación y funcionalidad adecuada para las respectivas operaciones.

Default tag table							
	Nombre	Tipo de datos	Dirección	Rema...	Acces...	Escrib...	Visibl...
1	encendido_general	Bool	%I0.0				
2	luz_encendido_motor_1	Bool	%M0.0				
3	corriente	Real	%M0.1				
4	temperatura	Real	%M0.2				
5	corriente2	Real	%M0.3				
6	temperatura2	Real	%M0.4				
7	motor_1	Bool	%Q0.0				
8	luz_paro_motor_1	Bool	%Q0.1				
9	motor_2	Bool	%Q0.2				
10	luz_paro_motor_2	Bool	%Q0.3				
11	luz_encendido_motor_2	Bool	%M0.5				
12	datos_sensor_corriente_1	Bool	%M1.4				
13	datos_sensor_corriente_2	Bool	%M1.5				
14	datos_sensor_temperatura_1	Bool	%M1.7				
15	datos_sensor_temperatura_2	Bool	%M2.0				
16	luz_error_corriente_motor_1	Bool	%M2.2				
17	luz_error_corriente_motor_2	Bool	%M2.3				
18	luz_error_temperatura_motor_1	Bool	%M2.5				
19	luz_error_temperatura_motor_2	Bool	%M2.6				
20	m1_encendido	Bool	%I0.1				
21	m2_encendido	Bool	%I0.2				
22	Paro_general	Bool	%I0.3				
23	P_M1_apagado	Bool	%I0.4				

Figura 11: Configuración de node-red

3.5.9.1 Programación del arranque del motor Se realizó la programación escalera para la simulación de un arranque de motor, utilizando FLIPFLOPS con el fin de poder controlar varias operaciones reduciendo las líneas de programación ladder.

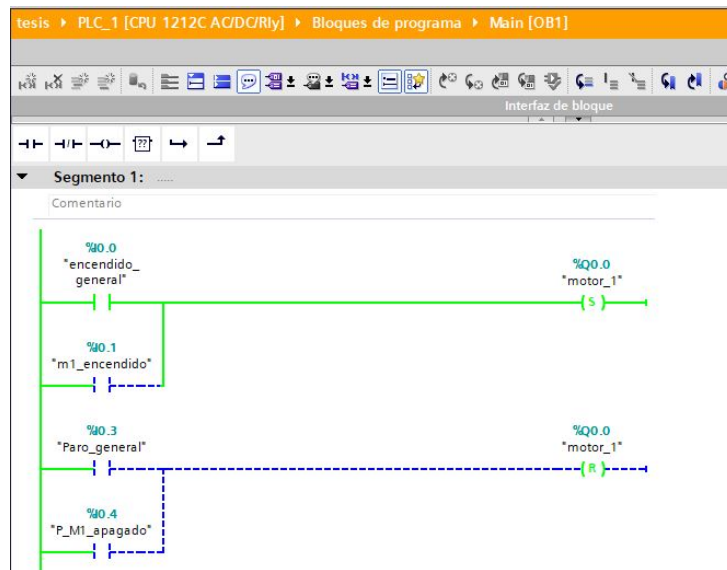


Figura 12: Arranque de un motor con FLIPFLOP

Estos bloques cumplen la función de obtener los datos y poder compararlos. Esta comparación permite un monitoreo de los sensores en los rangos específicos que trabajara el motor, en caso de no cumplirlos se enviara una señal de alarma a la interfaz HMI.

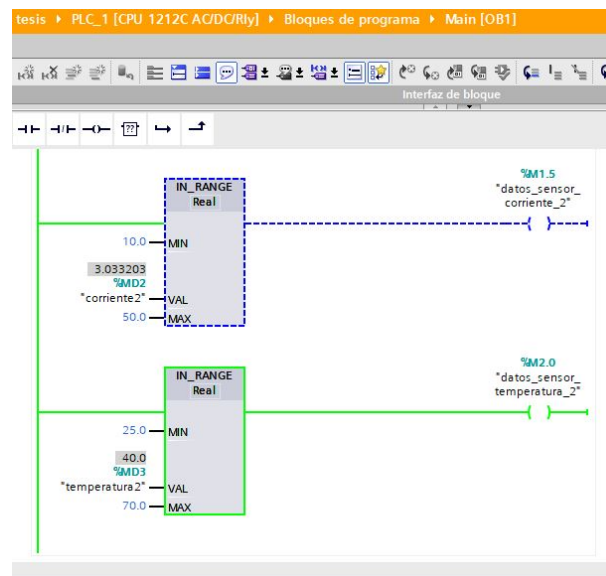


Figura 13: Comparadores para el rango de nuestros sensores

3.5.10 Comunicación entre el HMI Y PLC

Para interconectar los controladores logicos PLC y HMI, se utilizó los softwares DOP-SOFT Y TIA PORTAL, el programa DOPSOFT permite la programación de equipos DELTA desarrollando interfaces para proyectos de automatización industrial, como diseños de control y monitoreo de diversos procesos.



Figura 14: Comunicación entre el HMI Y PLC

Para el envío de datos del PLC al HMI, se realizó una redistribución de datos utilizando un bloque de datos “MOVE”, el cual se lo realizar para la interpretación y análisis de datos recibidos.

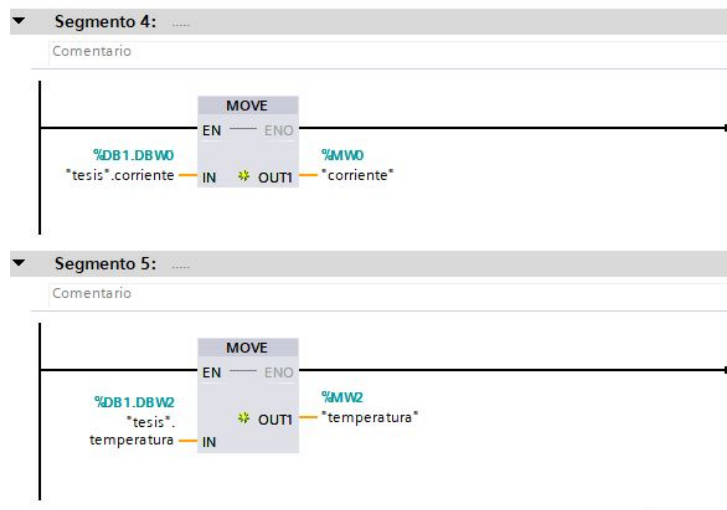


Figura 15: Distribución de datos

3.5.11 Arquitectura de la Interfaz HMI

En los siguientes procesos se puede ilustrar las diferentes interfaces de control y monitoreo de los parámetros del motor.

3.5.11.1 Pantalla de control del sistema

En esta pantalla se muestra la parte de control del sistema general, donde se activa y desactiva todo el sistema y comandos que permitirán la navegación en las diferentes interfaces, además, cuenta con botones de apagado

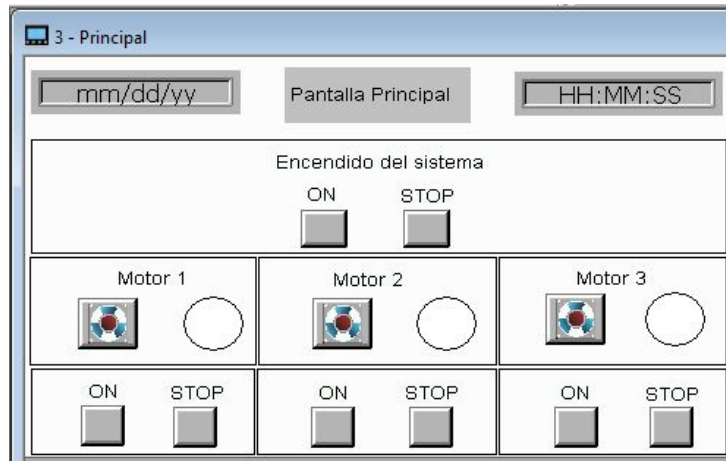


Figura 16: Interfaz Principal

3.5.11.2 Pantalla Dashboard para el control de corriente y temperatura

Aquí se mostrarán dos graficas, el cual representan los valores de los sensores recopilados y almacenados en una grafica para el mejor análisis de los parámetros del motor.

Es necesario llevar un registro de los errores que ocurrieron en horarios y fechas específicas, para poder diagnosticar cual fue la posible causa de este problema.

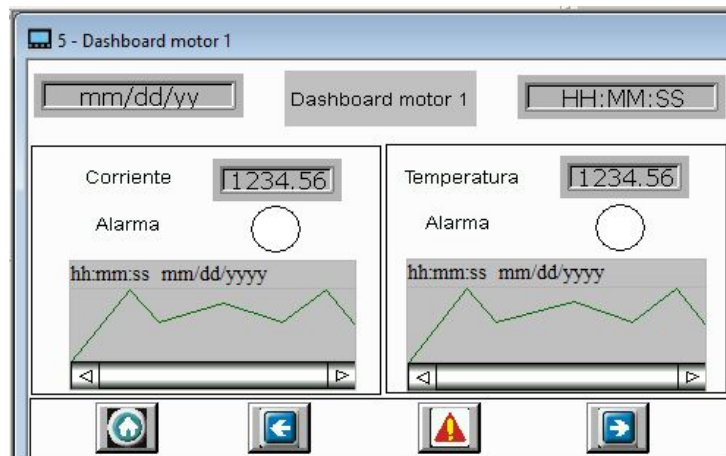


Figura 17: Dashboard de los sensores

3.5.11.3 Pantalla de la Ubicación de los equipos

En un área donde se el área y las condiciones no son fáciles de operar, cuando son demasiados equipos es necesario conocer la ubicación precisa de los motores, por ello se realizo la siguiente interfaz donde se agilizará el tiempo y los procesos de operación.

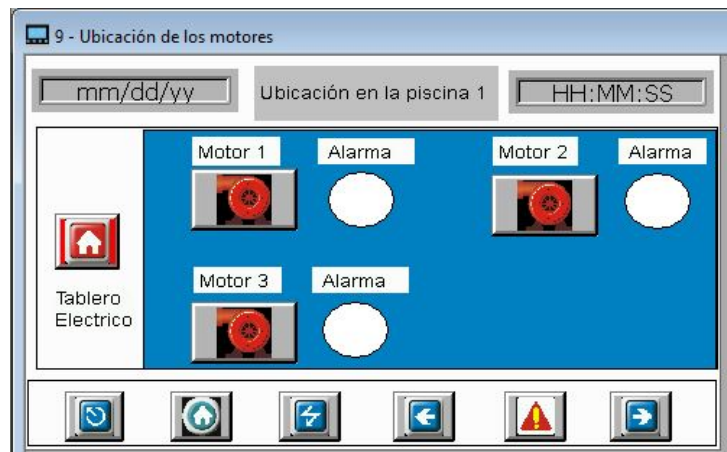


Figura 18: Ubicación de los diferentes motores en la piscina

3.5.12 Evaluación preliminar de las métricas de la calidad de señal de LoRaWAN

Después de completar el desarrollo tanto del hardware como de la programación de nuestro sistema, se abordó la implementación del análisis y calidad de nuestro sistema. Una de las etapas más cruciales en cualquier sistema de comunicación inalámbrica es la validación de la calidad de la transmisión de datos. Para llevar a cabo este proceso se utilizó Arduino, el cual nos permite gestionar y analizar el estado de la red LoRaWAN, se analizaron aspectos clave como la latencia y la tasa de pérdida de paquetes, para poder asegurar el estado de comunicación entre dispositivos de nuestro sistema.

CAPITULO IV

4 Análisis de Resultados

4.1 Pruebas

En la siguiente figura se podrá observar el Tablero de control de nuestro sistema, el cual muestra los elementos de control como el PLC, HMI.



Figura 19: Tablero de control

Los módulos LoRaWAN se colocarán en un tablero donde estarán ubicados nuestro sistema de comunicación inalámbrico, como se muestra en la figura, representando un ambiente controlado, además, se podrá evidenciar la ubicación de los dispositivos realizando los ajustes necesarios para lograr una comunicación eficiente.



Figura 20: Sistema de Comunicación Inalámbrica

En la siguiente figura podemos observar el equipo que lo utilizaremos para nuestro sistema de monitoreo y base de datos.



Figura 21: Raspberry Pi 4

4.2 Resultados

La siguiente sección se presentarán los resultados del presente trabajo de investigación, por ello, se van a detallar métricas siguiendo el orden de nuestros objetivos, además, de la evaluación de la transmisión de datos.

Una de las métricas que analizamos es la intensidad de señal o en sus signos en inglés Received Signal Strength Indicator (RSSI), para evaluar la comunicación entre los dispositivos. También, otra métrica es la precisión de los sensores el cual se la validó con instrumentos de medición.

4.2.1 Precisión de los sensores Se utilizó equipos de alta precisión que se utiliza en la industria para registrar, validar y comparar los sensores y equipos de medición, para poder evaluar la efectividad de nuestros sensores debido a que son dispositivos de pruebas en un ambiente controlado.

Tabla 2 de Precisión de los sensores

Parámetro	Condición	Valor del sensor	Valor con equipo real
1 Temperatura	Sin forzar	28.9°C	28.0°C (Pirómetro)
2 Temperatura	Simulando cambio de temperatura	30.8°C	30.6°C (Pirómetro)
3 Corriente simulado	Sin forzar	3A	3 Multímetro
4 Corriente simulado	Simulando resistencia	3A	3.5A Multímetro

4.2.2 Análisis de la potencia de la señal Se utilizó el programa Arduino para poder registrar y capturar la medición de intensidad de la señal (RSSI), además de evaluar la efectividad de nuestro sistema, debido a que la distancia de los equipos afecta la comunicación, por ello se realizaron varios números de pruebas como se observa en la siguiente tabla.

Para conocer el rango ideal de la intensidad de señal nos regiremos a la siguiente tabla.

- Excelente: Donde la intensidad de señal está a $RSSI > -30$ dBm, debido a que los equipos están muy cerca en línea vista.
- Muy buena: El transmisor está cerca, por ella la señal es de -50 dBm $< RSSI \leq -30$ dBm.
- Muy buena: El transmisor está cerca, por ella la señal es de -50 dBm $< RSSI \leq -30$ dBm.
- Buena: Cuando la comunicación es estable la señal está a -60 dBm $< RSSI \leq -50$ dBm.
- Aceptable: La señal está a -70 dBm $< RSSI \leq -60$ dBm, se define como señal adecuada para la transmisión de datos.
- Baja: Cuando la señal es de -80 dBm $< RSSI \leq -70$ dBm.
- Muy baja: La señal está a -90 dBm $< RSSI \leq -80$ dBm, posiblemente exista pérdida de comunicación.
- Mala: La señal es de $RSSI \leq -90$ dBm, esto produce que la señal sea muy débil y se pueden perder datos.

Con estos datos registrados podemos apreciar la variación de los datos recibidos del LoRa Receptor.

Tabla 3 de Resultados del LoRa Transmisor y Receptor

Objetivo:		Obtener la potencia de comunicación de nuestra red de sensores inalámbrico del motor al HMI		
Pruebas del Motor	Temperatura	Distancia	Tiempo de comunicación del LoRa Transmisor al Receptor	Intensidad de señal (RSSI)
1	28.90°C	1 metros	4 ms	-14 dBm
2	28.90°C	10 metro	20 ms	-30 dBm
3	28.90°C	50 metros	20 ms	-30 dBm
4	28.90°C	100 metros	25 ms	-40 dBm
5	28.90°C	200 metros	30 ms	-50 dBm
6	28.90°C	300 metros	40 ms	-57 dBm
7	28.90°C	500 metros	50 ms	-65 dBm
8	28.90°C	600 metros	55 ms	-66 dBm
9	28.90°C	800 metros	55 ms	-70 dBm
10	28.90°C	1km	80 ms	-75 dBm
Resultado	Datos enviados: 10 Datos recibidos: 10			

Con estos datos obtenidos se puede observar la variación del tiempo, sometiendo a los equipos a diferentes situaciones o distancias.

En la siguiente figura, se puede observar la gráfica de la relación entre la distancia y la potencia de señal del LoRa Transmisor y Receptor, con el fin de poder analizar el comportamiento del mismo.

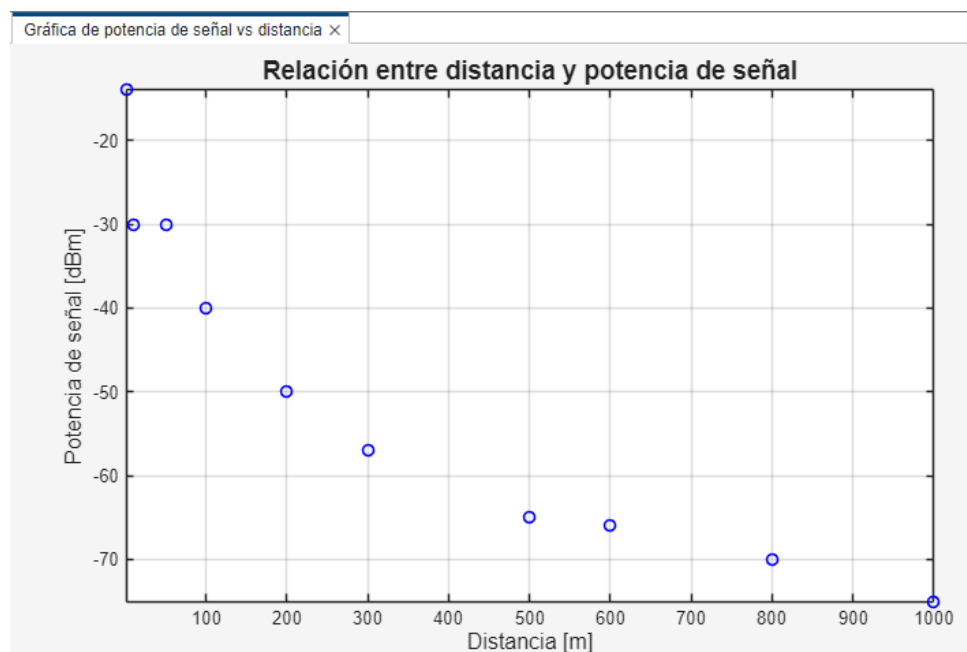


Figura 22: Relación de la potencia de señal y la distancia entre los LoRa.

También, se realizó el análisis de la relación entre el tiempo de transmisión de datos entre la distancia de los loras de transmisor y receptor ya analizada en la tabla 3.

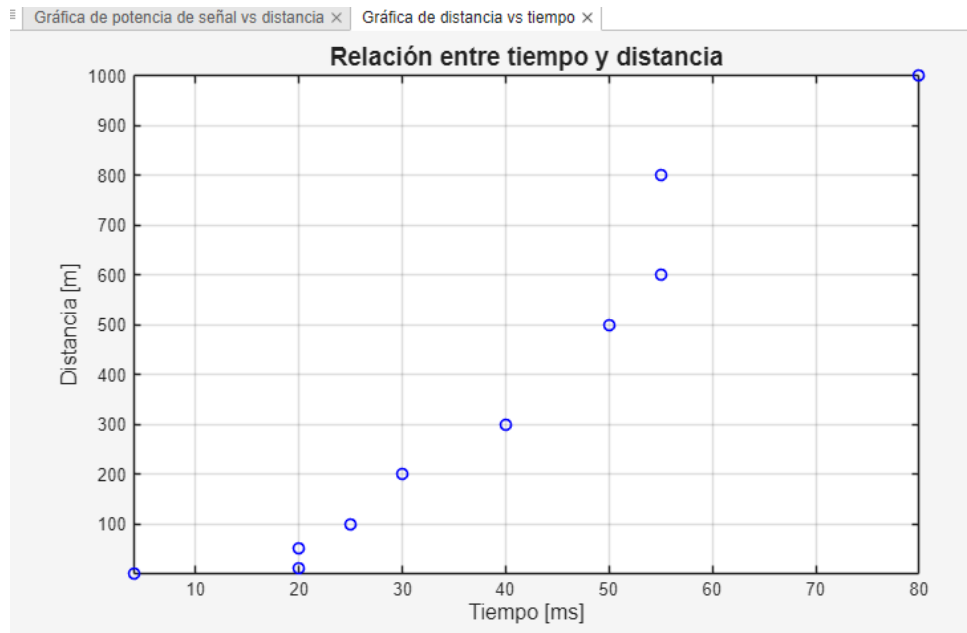


Figura 23: Relación de la Distancia y el tiempo de comunicación entre los LoRa.

4.2.3 Análisis de la Conexión de los Equipos de Control

Se utilizó el software TIA Portal, para capturar y registrar los datos obtenidos de los sensores conectados al sistema, estos datos fueron capturados para medir el tiempo de conexión entre los equipos de control, además facilitó el análisis del comportamiento en diferentes situaciones, como la saturación de dispositivos conectados en un ambiente controlado para verificar la comunicación entre los equipos.

Tabla 4 de Resultados del Tiempo de los equipos de control

Objetivo:		Obtener el tiempo de comunicación de nuestra red de sensores inalámbrico del motor al HMI	
Pruebas del Motor	Temperatura	Tiempo de Respuesta de comunicación de la Raspberry pi 4 al PLC	Tiempo de comunicación del PLC al HMI
1	28.90°C	13 ms	15 ms
2	28.90°C	13 ms	15 ms
3	28.90°C	14 ms	16 ms
4	28.90°C	13 ms	15 ms
5	28.90°C	13 ms	16 ms
6	28.90°C	13 ms	16 ms
7	28.90°C	13 ms	15 ms
8	28.90°C	13 ms	15 ms
9	28.90°C	13 ms	15 ms
10	28.90°C	14 ms	16 ms
Resultado	Datos enviados: 10 Datos recibidos: 10		

Ademas, se realizó el análisis de la gráfica en relación al tiempo con el numero de pruebas que se realizó, como nuestro sistema obtiene datos de nuestros sensores de manera constantes, podemos definir que la siguiente gráfica es el tiempo de respuesta de nuestro sistema .

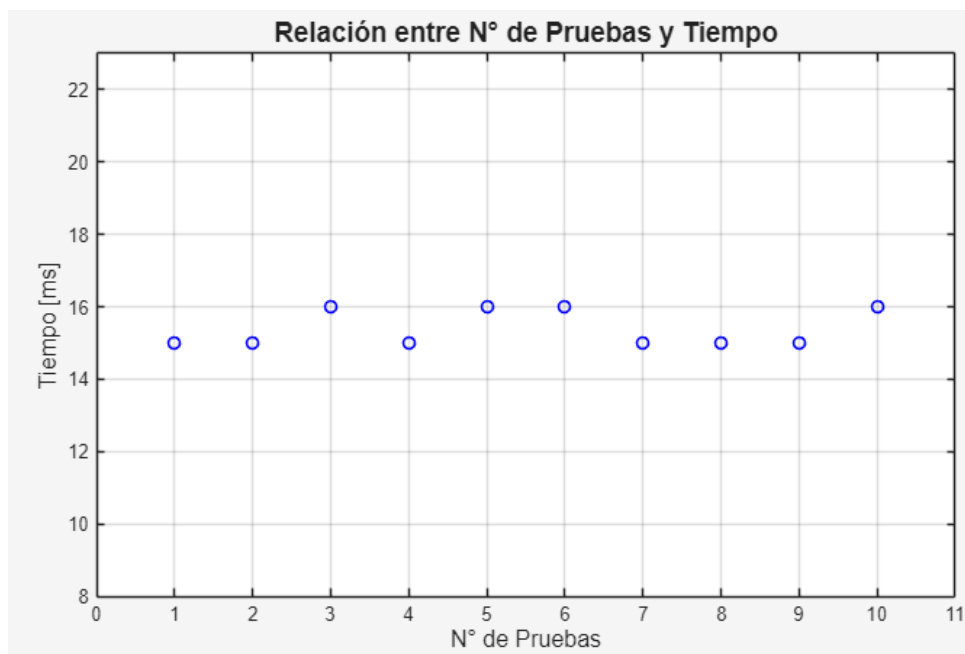


Figura 24: Gráfica del tiempo de respuesta del sistema.

4.3 Resultados de los Objetivos

Una vez realizadas las respectivas pruebas para el monitoreo de los parámetros del motor mediante un ambiente controlado.

Con respecto al objetivo 1, el desarrollo de un sistema de comunicación inalámbrica con el LoRa ESP32 para la adquisición de datos, se observa en la tabla 3 que los valores obtenidos por los sensores presentan un error del 0,2 %, y demuestra ser preciso a los datos enviados y capturados dando una transmisión de datos del 100 % y respuesta dando un promedio de tiempo de 38 ms, debido a este tiempo lo hace ideal para el objetivo de este proyecto.

En base a las pruebas realizadas y registradas en la tabla 4, podemos definir que el intercambio de datos de los equipos Raspberry Pi 4, PLC y HMI fue exitoso, obteniendo un 100 % de transmisión de datos por TCP/IP, por ello no se obtuvieron ninguna pérdida de datos en el ambiente controlado.

Por el Objetivo 3 de simular escenarios de la vida real, se opero el sistema en diversas distancias para validar la operatividad y rapidez del sistema, obteniendo valores aceptables para el fin del proyecto.

El sistema cuenta una interfaz muy intuitiva de diversas pantallas para la mayor comprensión del usuario, con el fin de poder realizar el monitoreo y supervisión de los parámetros eléctricos del motor.

Conclusión

La implementación de una red de sensores inalámbricos utilizando tecnología IIoT en conjunto con la tecnología LoRa, representa una solución altamente eficiente y adaptable para el monitoreo y control de un motor. Además, esta combinación tecnológica es especialmente adecuada para aplicaciones que requieren comunicación donde se deben abarcar grandes distancias.

LoRa permite una transmisión de datos robusta y de bajo consumo energético donde la comunicación continua y confiable entre los sensores y los sistemas de control es indispensable. Además, esta tecnología se caracteriza por su compatibilidad con múltiples dispositivos de automatización y control.

Los sistemas de control PLC y HMI permiten ejecutar tareas de control en tiempo real, el cual proporcionan una interfaz visual intuitiva para el monitoreo, diagnóstico y configuración del sistema.

Gracias a la implementación de este sistema de monitoreo de largo alcance, es posible optimizar recursos humanos y materiales, ya que se reduce la necesidad de supervisión física constante. Además, permite una respuesta más rápida por parte del personal operativo ante señales de alarma.

Recomendaciones

Para garantizar una transmisión de datos estable los LoRas emisor y receptor deben estar ubicados con línea de vista directa, debido a que la presencia de obstáculos puede afectar la transmisión de datos.

Para una implementación a escala real, se requiere instalar sensores industriales para operar de forma confiable en condiciones adversas. Además, estos sensores ofrecen mayor precisión y durabilidad.

Es fundamental implementar tiempos específicos para el mantenimiento de los equipos de control como los microcontroladores, PLCs y LoRa, y limpiar los datos de su memoria para evitar su saturación de datos.

Dado que el entorno donde sería implementado presenta altos niveles de salinidad, es esencial proteger el sistema completo mediante recubrimientos anticorrosivos y sellado adecuado de conexiones.

Referencias

- [1] S. C. Esquivia, A. F. S. Prisco, J. M. R. Serna y M. d. J. B. Durando, “Sistema inalámbrico basado en IoT para la medición temperatura y velocidad en un motor de inducción,” *Revista CINTEX*, vol. 27, n.º 2, págs. 32-42, 2022.
- [2] P. E. Maldonado Jara, “Diseño e implementación de un sistema SCADA-IIOT de monitoreo de paros en máquinas de tejido, para reducir tiempos de corte,” B.S. thesis, Riobamba, Universidad Nacional de Chimborazo, 2023.
- [3] E. H. Rivadeneira y P. L. Andrade, *Diseño e implementación de una red inalámbrica de sensores con tecnología LoRa para monitoreo industrial orientado a OPC de arquitectura unificada*, 2021.
- [4] J. A. G. Vasquez, M. E. C. Arroyo, M. M. H. Maguiña, M. K. L. Asto, L. M. S. Quincho y A. F. V. Cabrera, “TECNOLOGÍAS DEL INTERNET DE LAS COSAS APLICADAS EN LA CADENA DE SUMINISTRO. UNA REVISIÓN SISTEMÁTICA,” *Gestión de Operaciones Industriales*, vol. 2, n.º 01, págs. 8-26, 2023.
- [5] C. Sun, F. Zheng, G. Zhou y K. Guo, “Design and Implementation of Cloud-based Single-channel LoRa IIoT Gateway Using Raspberry Pi,” en *2020 39th Chinese Control Conference (CCC)*, 2020, págs. 5259-5263. DOI: 10.23919/CCC50068.2020.9189480.
- [6] C. L. Aguilar Diaz y A. L. Cieza De Los Santos, “Propuesta de Implementación para la Automatización y Monitoreo de una Planta de Tratamiento de Agua Residual (PTAR),”
- [7] J. H. Hwang Cárdenas y J. A. Iñiguez Avila, “Sistema de comunicación inalámbrico y HMI basado en PYTHON para la operación y monitoreo remoto de un motor trifásico de inducción,” B.S. thesis, 2023.
- [8] M. Saban et al., “A Smart Agricultural System Based on PLC and a Cloud Computing Web Application Using LoRa and LoRaWan,” *Sensors*, vol. 23, n.º 5, 2023, ISSN: 1424-8220. DOI: 10.3390/s23052725. dirección: <https://www.mdpi.com/1424-8220/23/5/2725>.
- [9] J. A. R. GAMBOA, “Modelamiento e implementación de una arquitectura de control y monitoreo de bajo costo basado en microcontroladores para un entorno IIOT,” *Repos. Tesis-UNMSM*, 2023.
- [10] W. Huacho Ichpas, “Diseño de un módulo de control basado en plataforma IOT para el monitoreo remoto de motores de inducción de baja potencia. Huancayo-2022,” 2022.
- [11] D. J. Guevara Hidalgo y S. E. Tibanquiza Chuncho, “Implementación de un sistema remoto para control y monitoreo de procesos utilizando un PLC con Gateway IIOT,” 2021.
- [12] M. Intelligence, *Análisis de participación y tamaño del mercado de monitoreo de motores tendencias de crecimiento y pronósticos (2024-2029)*, 2024. dirección: <https://www.mordorintelligence.com/es/industry-reports/motor-monitoring-market>.
- [13] D. A. Aldaz Chacho, K. F. Quinte Alcoser et al., “Sistema de monitoreo y control remoto de una bomba sumergible para el llenado de un reservorio en un sector camaronero de Santa Priscila,” *ESPOL. FIEC*, 2023.

- [14] S. V. E. Desiderio y M. A. N. Cornejo, “Análisis exhaustivo de las tecnologías clave en la Industria 4.0,” *Polo del Conocimiento*, vol. 9, n.º 8, págs. 945-965, 2024.
- [15] W. A. Khan, L. Wisniewski, D. Lang y J. Jasperneite, “Analysis of the requirements for offering industrie 4.0 applications as a cloud service,” en *2017 IEEE 26th international symposium on industrial electronics (ISIE)*, IEEE, 2017, págs. 1181-1188.
- [16] O. M. Núñez Meléndez, “Análisis comparativo de la tecnología WiFi y LiFi para la selección adecuada en la Facultad de Ciencias Administrativas, Gestión Empresarial e Informática de la Universidad Estatal de Bolívar, año 2016–2017,” 2017.
- [17] Z. S. OMAÑA, “Automatización del Proceso de Frutado y Batido de Yogurt Empleando un Controlador Lógico Programable y una Interface Hombre-Máquina,” Tesis doct., Universidad Autónoma del Estado de Hidalgo, 2014.
- [18] J. P. Dignani, “Análisis del protocolo ZigBee,” Tesis doct., Universidad Nacional de la Plata, 2012.
- [19] C. Serrano Cano, “Instalación de un sistema domótico Z-Wave en entorno residencial,” 2017.
- [20] P. Pardal Garcés, “Redes de Área Extensa para aplicaciones de IoT: modelado de comunicaciones Sigfox,” 2017.
- [21] J. Piña, *Modbus: Representación de datos*, 2013.
- [22] I. L. Cevallos Polo y D. S. Santo Proaño, “Diseño e Implementación de un Módulo Didáctico para prácticas de Red Profibus y Profinet Con Plcs 71500 Aplicado a Motores Trifásicos,” B.S. thesis, 2021.
- [23] N. Aguirre, N. Aranda y N. Balich, “Seguridad en el envío de mensajes mediante protocolo MQTT en IoT,” *INNOVA UNTREF. Revista Argentina de Ciencia y Tecnología*, 2019.
- [24] P. Bertoleti, *Proyectos con ESP32 y LoRa*. Editora NCB, 2019.
- [25] J. C. Merino Polidura et al., “Despliegue y Evaluación de una Red OnSite LoRa-WAN basada en The Things Network Stack versión 3,” 2019.
- [26] I. Ordóñez Monfort, “Estudio de la arquitectura y el nivel de desarrollo de la red LoRaWAN y de los dispositivos LoRa,” 2017.
- [27] J. Pérez, E. Urdaneta y Á. Custodio, “Metodología para el diseño de una red de sensores inalámbricos,” *Universidad, Ciencia y Tecnología*, vol. 18, n.º 70, págs. 12-22, 2014.
- [28] E. J. Guaña Moya, “Diseño de una Red de Sensores Inalámbricos (WSN) para monitorear parámetros relacionados con la agricultura,” B.S. thesis, Quito, 2016., 2016.
- [29] J. Martínez Jimeno et al., “Desarrollo de un sistema de monitorización de temperaturas en tiempo real para intercambiador de calor de doble tubo con sondas sumergibles de temperatura DS18B20 usando el miconcontrolador Arduino,” 2018.
- [30] F. Santiago Espinosa et al., *Los microcontroladores AVR de ATMEL*. Universidad Tecnológica de la Mixteca, 2012.
- [31] J. A. Suárez Briones, “Prototipo de dispositivo electrónico Iot para la adquisición de diversos parámetros físicos en la transportación pública,” B.S. thesis, La Libertad: Universidad Estatal Península de Santa Elena, 2019., 2019.

- [32] R. A. Salvador y C. B. Prieto, “Sistema de seguridad con reconocimiento facial en módulo ESP32,” *Mare Ingenii*, vol. 4, n.º 1, págs. 43-52, 2022.
- [33] S. Chapman, *Máquinas eléctricas*. Mc Graw Hill, 2012.
- [34] J. B. Gamarra Zarate, “Sistema De Bombeo Automatizado Mediante Controlador Lógico Programable, Como Alternativa Para La Reducción De Los Tiempos De Desabastecimiento De Agua, En El AH Portada De Manchay Ii, Distrito De Pachacamac,” 2017.
- [35] J. N. Lombeida Vásquez y J. J. Samaniego Reyes, “Desarrollo de un tablero demostrativo para el Control de Sistemas de aireadores y alimentadores supervisados y monitoreados mediante radiofrecuencia por medio de Sistema Scada,” B.S. thesis, 2022.
- [36] J. D. León Callupe, “Diseño e implementación de un PLC basado en Raspberry Pi & Node-RED para aplicaciones industriales IoT,” 2021.
- [37] J. Torres Ventura, A. Ruelas Puente y J. Herrera García, “Rendimiento para la interoperabilidad entre Raspberry pi, ESP8266 y PLC con Node-RED para el IIoT,” *Ingenius. Revista de Ciencia y Tecnología*, n.º 29, págs. 90-97, 2023.
- [38] J. A. Maroto Lema, A. Nuñez Unda et al., “Desarrollo de un Prototipo de Sistema de Transmisión de Baja Potencia para Medición de Humedad y Temperatura en un Centro de Datos, Usando el Protocolo Lora,” Tesis doct., ESPOL. FIEC., 2020.
- [39] F. E. Cunza Asencios, “Diseño e implementación de un sistema de control a distancia para el control y gestión de motores monofásicos, empleando tecnología LoRa,” 2021.

labelsec5.3

Anexos



Figura 25: HMI

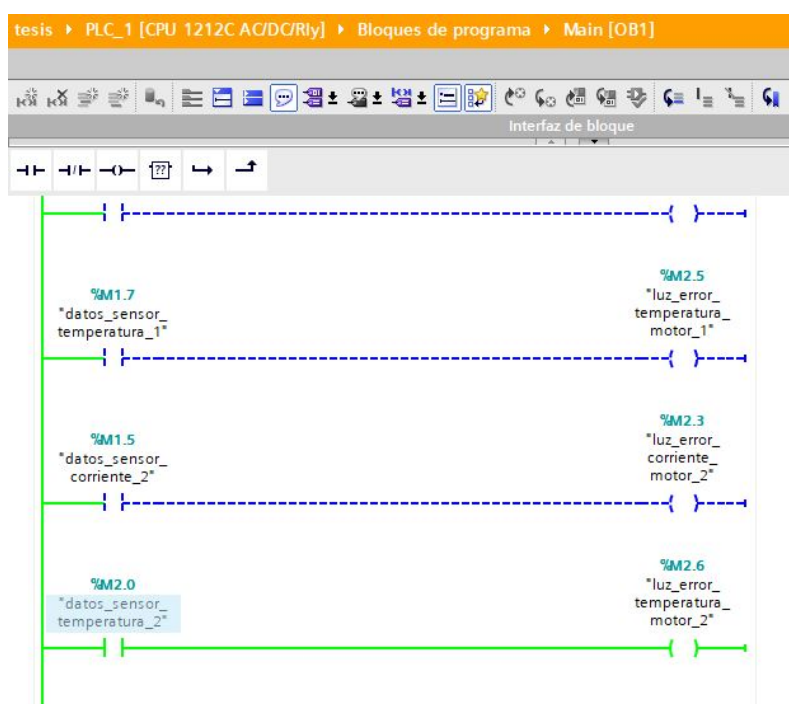


Figura 26: Programación de alarmas

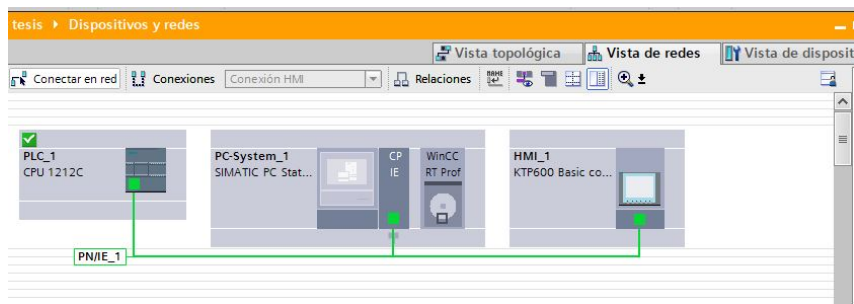


Figura 27: Interconexión de los equipos de control

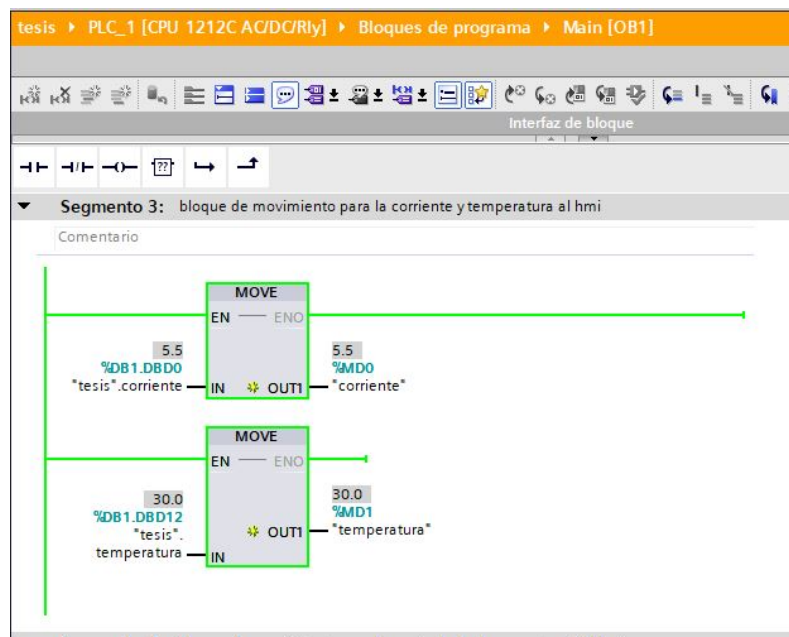


Figura 28: Movimiento de los datos del PLC al HMI

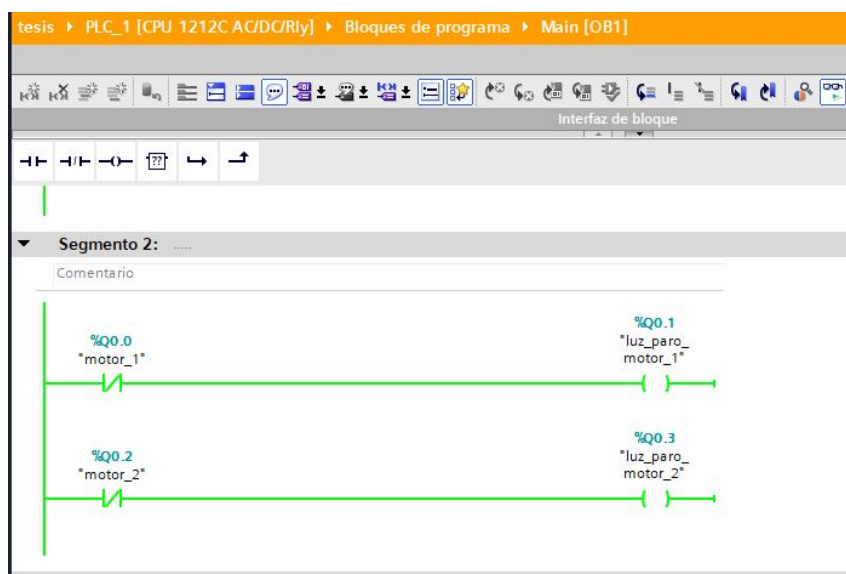


Figura 29: Programación de las Luces Piloto



Figura 30: Historial de alarmas

```

auto@raspberrypi: ~
Archivo  Editar  Pestañas  Ayuda
64 bytes from 172.17.12.180: icmp_seq=13 ttl=64 time=208 ms
64 bytes from 172.17.12.180: icmp_seq=14 ttl=64 time=15.0 ms
64 bytes from 172.17.12.180: icmp_seq=15 ttl=64 time=40.3 ms
64 bytes from 172.17.12.180: icmp_seq=16 ttl=64 time=59.0 ms
64 bytes from 172.17.12.180: icmp_seq=17 ttl=64 time=161 ms
64 bytes from 172.17.12.180: icmp_seq=18 ttl=64 time=109 ms
64 bytes from 172.17.12.180: icmp_seq=19 ttl=64 time=29.9 ms
64 bytes from 172.17.12.180: icmp_seq=20 ttl=64 time=49.9 ms
64 bytes from 172.17.12.180: icmp_seq=21 ttl=64 time=280 ms
64 bytes from 172.17.12.180: icmp_seq=22 ttl=64 time=88.3 ms
64 bytes from 172.17.12.180: icmp_seq=23 ttl=64 time=110 ms
64 bytes from 172.17.12.180: icmp_seq=24 ttl=64 time=272 ms
64 bytes from 172.17.12.180: icmp_seq=25 ttl=64 time=54.5 ms
64 bytes from 172.17.12.180: icmp_seq=26 ttl=64 time=208 ms
64 bytes from 172.17.12.180: icmp_seq=27 ttl=64 time=92.8 ms
64 bytes from 172.17.12.180: icmp_seq=28 ttl=64 time=182 ms
64 bytes from 172.17.12.180: icmp_seq=29 ttl=64 time=51.2 ms
64 bytes from 172.17.12.180: icmp_seq=30 ttl=64 time=51.8 ms
64 bytes from 172.17.12.180: icmp_seq=31 ttl=64 time=78.6 ms
^C
--- 172.17.12.180 ping statistics ---
31 packets transmitted, 31 received, 0% packet loss, time 30059ms
rtt min/avg/max/mdev = 14.951/95.611/279.548/69.350 ms
auto@raspberrypi:~ $

```

Figura 31: Tiempo de respuesta de Comunicación de node red

```
Salida Monitor Serie x
Mensaje (Intro para mandar el mensaje de 'TTGO LoRa32-OLED' a 'COM4')
-
81 bytesDatos recibidos: Sin movimiento,Temperatura: 28.90 C,Humedad: 40.00 %,Gas D0: Sin gas,Gas A0: 2467
RSSI: -105
Tiempo de comunicación LoRa: 4 ms
*****
Monitoreo de los parametros del motor
Paquete recibido:
81 bytesDatos recibidos: Sin movimiento,Temperatura: 28.90 C,Humedad: 40.00 %,Gas D0: Sin gas,Gas A0: 2463
RSSI: -106
Tiempo de comunicación LoRa: 5 ms
*****
Monitoreo de los parametros del motor
Paquete recibido:
81 bytesDatos recibidos: Sin movimiento,Temperatura: 28.90 C,Humedad: 40.00 %,Gas D0: Sin gas,Gas A0: 2465
RSSI: -105
Tiempo de comunicación LoRa: 5 ms
*****
Monitoreo de los parametros del motor
Paquete recibido:
81 bytesDatos recibidos: Sin movimiento,Temperatura: 28.90 C,Humedad: 40.00 %,Gas D0: Sin gas,Gas A0: 2465
RSSI: -105
Tiempo de comunicación LoRa: 5 ms
*****
Monitoreo de los parametros del motor
Paquete recibido:
81 bytesDatos recibidos: Sin movimiento,Temperatura: 28.90 C,Humedad: 40.00 %,Gas D0: Sin gas,Gas A0: 2466
RSSI: -105
Tiempo de comunicación LoRa: 4 ms
```

Figura 32: Tiempo de respuesta de Comunicación del LoRa Emisor y Receptor



Figura 33: Valor del pírometro en un ambiente controlado



Figura 34: Pírometro y Sensor

```
LoraEmisor.ino
1  #include <SPI.h>
2  #include <LoRa.h>
3  #include <DHT.h> // Librería para el sensor DHT11
4
5  // Definición de pines
6  #define SS 5 // Pin NSS (CS) para LoRa
7  #define RST 14 // Pin RESET para LoRa
8  #define DIO0 2 // Pin DIO0 para LoRa
9  #define PIR_PIN 27 // Pin del sensor PIR
10 #define DHT_PIN 32 // Pin del sensor DHT11
11 #define MQ2_AO_PIN 34 // Pin A0 del sensor de gas MQ-2 (salida analógica)
12 #define MQ2_DO_PIN 26
13
14 // Pin D0 del sensor de gas MQ-2 (salida digital)
15
16 // Configuración del sensor DHT
17 #define DHTTYPE DHT11 // Tipo de sensor (DHT11)
18 DHT dht(DHT_PIN, DHTTYPE);
19
20 void setup() {
21   Serial.begin(115200);
22   pinMode(PIR_PIN, INPUT);
23   pinMode(MQ2_DO_PIN, INPUT);
24   dht.begin();
25
26   LoRa.setPins(SS, RST, DIO0);
27   if (!LoRa.begin(433E6)) {
28     Serial.println("Error al inicializar LoRa. Verifica las conexiones.");
```

Figura 35: Código LoRa emisor a.


```

receptor_Lora2_modbus.ino
30
31 void setup() {
32     Serial.begin(9600);
33
34     // Inicialización de la pantalla OLED
35     if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
36         Serial.println(F("Error al iniciar pantalla OLED"));
37         for (;;);
38     }
39     display.clearDisplay();
40     display.setTextSize(1);
41     display.setTextColor(SSD1306_WHITE);
42     display.setCursor(0, 10);
43     display.println("Iniciando LoRa...");
44     display.display();
45
46     // Configuración LoRa
47     LoRa.setPins(SS, RST, DIO0);
48     if (!LoRa.begin(433E6)) {
49         Serial.println("Error al iniciar LoRa");
50         display.clearDisplay();
51         display.setCursor(0, 10);
52         display.println("Error LoRa");
53         display.display();
54         while (1);
55     }
56
57     Serial.println("LoRa iniciado correctamente");
58     display.clearDisplay();

```

Figura 38: Código LoRa Receptor b.

```

receptor_Lora2_modbus.ino
59 display.clearDisplay();
60 display.setCursor(0, 10);
61 display.println("LoRa iniciado");
62 display.display();
63
64 // Configuración de red WiFi para Modbus TCP/IP
65 WiFi.config(ip, gateway, subnet);
66 WiFi.begin("Linlin", "Holi12345");
67 while (WiFi.status() != WL_CONNECTED) {
68     delay(500);
69     Serial.print(".");
70 }
71
72 Serial.println("\nWiFi conectado");
73 Serial.println("Dirección IP: ");
74 Serial.println(WiFi.localIP());
75
76 // Configuración del servidor Modbus
77 mb.server();
78 mb.addHreg(RegTemperatura); // 40002
79 }
80
81 void loop() {
82     mb.task(); // Tarea Modbus
83
84     // Recepción de datos por LoRa
85     int packetSize = LoRa.parsePacket();
86     if (packetSize) {

```

Figura 39: Código LoRa Receptor c.

```

receptor_Lora2_modbus.ino
99     float temperatura = obtenerValor(message, "Temperatura: ", " C");
100
101
102     // Mostrar datos en la pantalla OLED
103     display.clearDisplay();
104     display.setCursor(0, 0);
105     display.println("Datos recibidos:");
106     display.setCursor(0, 10);
107     display.println("Temp: " + String(temperatura) + " C");
108     display.setCursor(0, 30);
109     // Enviar datos a los registros Modbus
110     mb.Hreg(RegTemperatura, (int)temperatura);
111
112
113
114     delay(200);
115 }
116
117 // Función para obtener valores específicos de un mensaje
118 float obtenerValor(String mensaje, String inicio, String fin) {
119     int startIndex = mensaje.indexOf(inicio) + inicio.length();
120     int endIndex = mensaje.indexOf(fin, startIndex);
121     if (startIndex != -1 && endIndex != -1) {
122         return mensaje.substring(startIndex, endIndex).toFloat();
123     }
124     return 0.0;
125 }
126

```

Figura 40: Código LoRa Receptor d.

