

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Electricidad y Computación

Desarrollo de una cámara de secado para optimizar el mantenimiento de
endoscopios en la empresa Perfectech

INGE-2633

Proyecto Integrador

Previo la obtención del Título de:

Ingeniero en Electrónica y Automatización

Presentado por:

Francisco Saul Bastidas Llerena

George Alberto Maquilon Ochoa

Guayaquil - Ecuador

Año: 2024-2025

Dedicatoria

A mis padres, las personas más importantes en mi vida. Este logro es para ustedes. Sin su apoyo, fe y amor incondicional, este sueño no habría sido posible. Gracias por su paciencia infinita, incluso cuando dejaba la mesa del comedor llena de cables y componentes electrónicos, siempre apoyándome en cada paso del camino.

A mi fiel compañero, Toby, mi perrito, que siempre estuvo a mi lado durante largas noches de estudio, deberes y proyectos, esperándome hasta la madrugada y brindándome compañía silenciosa y constante.

Esta tesis también es de ustedes, porque cada logro lleva consigo un pedazo de su amor y apoyo incondicional. Especial nombramiento a los Electrofrénicos como mi especial grupo de amigos de la carrera, que me brindaron fuerzas y risas.

Att: Francisco Saul Bastidas Llerena

Dedicatoria

A mis padres, por mostrarme con el ejemplo que no hay metas que no se puedan alcanzar cuando hay trabajo duro y amor de por medio. Su fe en mí ha sido el pilar que me sostuvo durante los momentos más difíciles y la razón por la que me rendí. Este logro es tanto mío como de todos ustedes; di cada paso apoyado por su apoyo incondicional.

A mis amigos y profesores, que con su apoyo y enseñanzas dejaron una huella imborrable en mi camino académico. En especial, al Ing. Holger Cevallos, por su paciencia y dedicación para guiarnos en este proyecto, y por enseñarnos que el aprendizaje nunca termina. Especial nombramiento a los Electrofrénicos como el mejor grupo de amigos de la carrera que me brindaron un lugar seguro y muchos momentos memorables.

Att: George Alberto Maquilón Ochoa

Agradecimientos

Quiero expresar mi más profundo agradecimiento a mi familia, por siempre estar a mi lado, apoyándome en las buenas y en las malas, siendo el pilar fundamental en este proceso. A todas las personas que han sido parte de este viaje, a mis amigos por su compañía y amistad durante todo el tiempo académico. A mi tutor, por su valiosa guía durante cada etapa del proyecto, y al profesor Dennys Cortez, por sus comentarios, apoyo y ayuda constante, que han sido clave en el desarrollo de este trabajo.

A todos ustedes, muchas gracias por ser parte de este logro.

Att: Francisco Saul Bastidas Llerena

Agradecimientos

Extender mi reconocimiento y agradecimiento a la empresa Perfectech que nos brindó la oportunidad de tomar este proyecto como parte del curso de materia integradora, a mis amigos, cuya compañía y amistad han sido un motor invaluable durante mi etapa académica.

A nuestro tutor el Ing. Holger Cevallos, agradecemos su paciencia, compromiso y orientación a lo largo de cada etapa de este proyecto. Su guía ha sido esencial para llevar nuestras ideas a buen puerto. Asimismo, al Ing. Dennys Cortez, le expresamos nuestra gratitud por sus comentarios, correcciones y constante disposición para ayudarnos a mejorar; su apoyo ha sido clave en el desarrollo exitoso de este trabajo.

Att: George Alberto Maquilón Ochoa

Declaración Expresa

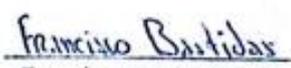
Nosotros Francisco Saul Bastidas Llerena y George Alberto Maquilón Ochoa acordamos y reconocemos que:

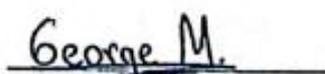
La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por mí/nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique a los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 11 de octubre del 2024.


Francisco Saul Bastidas
Llerena


George Alberto Maquilón
Ochoa

Evaluadores

Msc. Dennys Dick Cortez Álvarez

Profesor de Materia

Ph. D. Holger Ignacio Cevallos

Ulloa

Tutor de proyecto

Resumen

El presente proyecto tiene como objetivo diseñar e implementar una cámara de secado para optimizar el mantenimiento de endoscopios en la empresa Perfectech ubicada en Guayaquil, Ecuador, de tal manera que garantice un proceso eficiente y controlado. Los endoscopios son equipos médicos especializados para procedimientos mínimamente invasivos, utilizados para diagnóstico, terapéuticos y cirugías dentro del cuerpo humano, por medio de orificios naturales los médicos utilizan la endoscopia para tratar enfermedades que afectan el estómago, esófago, etc. Es de suma importancia para esta rama de la medicina que los equipos estén en óptimas condiciones cuando el mismo entre en reparación por varias razones: libre de microorganismos para evitar contaminación cruzada, libre de fugas, etc.

La cámara está diseñada para eliminar la humedad en los canales internos y para el secado del pegamento epóxico en la punta distal, abordando puntos críticos como el tubo de inserción y el anillo de epóxico, áreas comunes de desgaste lo que provoca fuga y humedad en la cámara del endoscopio.

El sistema consta de una ESP32 como el microcontrolador principal, sensores de temperatura, humedad, y una pantalla HMI, además con un controlador en lazo cerrado (PID) para regular las resistencias térmicas. Esta configuración permite monitorear en tiempo real las condiciones de secado. Los resultados demuestran una mejora significativa en la eficiencia y calidad del proceso, superando el método manual tradicional, como el secado bajo un foco incandescente. En conclusión, la cámara de secado representa una solución eficiente y segura para el mantenimiento de endoscopios, contribuyendo a la optimización de procesos en Perfectech.

Palabras clave: Endoscopios, Sistema PID, Temperatura controlada, ESP32, Humedad, Pantalla HMI.

Abstract

The objective of this project is to design and implement a drying chamber to optimize the maintenance of endoscopes in the company Perfectech located in Guayaquil, Ecuador, so as to ensure an efficient and controlled process. Endoscopes are specialized medical equipment for minimally invasive procedures, used for diagnostic, therapeutic and surgical procedures inside the human body. By means of natural orifices, doctors use endoscopy to treat diseases affecting the stomach, esophagus, etc. It is of utmost importance for this branch of medicine that the equipment is in optimal conditions when it enters into repair for several reasons: free of microorganisms to avoid cross contamination, free of leaks, etc.

The chamber is designed to eliminate moisture in the internal channels and for the drying of the epoxy glue in the distal tip, addressing critical points such as the insertion tube and the epoxy ring, common areas of wear that cause leakage and moisture in the endoscope chamber.

The system consists of an ESP32 as the main microcontroller, temperature and humidity sensors, and an HMI display, as well as a closed-loop controller (PID) to regulate the thermal resistances. This configuration allows real-time monitoring of the drying conditions. The results show a significant improvement in the efficiency and quality of the process, surpassing the traditional manual method, such as drying under an incandescent lamp. In conclusion, the drying chamber represents an efficient and safe solution for the maintenance of endoscopes, contributing to process optimization at Perfectech.

Keywords: Endoscopes, PID system, Temperature controlled, ESP32, Humidity, HMI display.

Índice General

RESUMEN.....	I
ABSTRACT.....	II
ÍNDICE GENERAL	III
ABREVIATURAS.....	VI
SIMBOLOGÍA.....	VII
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS.....	IX
ÍNDICE DE PLANOS	X
ÍNDICE DE CÓDIGOS.....	XI
CAPÍTULO I.....	1
1. INTRODUCCIÓN	1
1.1 DESCRIPCIÓN DEL PROBLEMA	1
1.2 JUSTIFICACIÓN DEL PROBLEMA	1
1.3 OBJETIVOS	2
1.3.1 <i>Objetivo General</i>	2
1.3.2 <i>Objetivos Específicos</i>	2
1.4 MARCO TEÓRICO	3
1.4.1 <i>Fundamentos De La Endoscopía</i>	3
1.4.2 <i>Detección De Fugas/Humedad En Endoscopios</i>	4
1.4.3 <i>Sensores De Temperatura Y Humedad (PT100 Y SHT31)</i>	6
1.4.4 <i>Control De Temperatura PID</i>	7
1.4.5 <i>Sensor MAX31865</i>	8
1.4.6 <i>Relé De Estado Solida (SSR-25DA)</i>	9
1.4.7 <i>Normativas y estándares internacionales sobre el reprocesamiento de endoscopios</i>	9
CAPÍTULO 2	11
2. METODOLOGÍA	11
2.1 SELECCIÓN DE MÉTODO Y ESTRATEGIA DE ELABORACIÓN IMPLEMENTADA	11

2.2	EXPERIMENTACIÓN	12
2.3	DIAGRAMA DE FLUJO	13
2.4	DISEÑO CONCEPTUAL	14
2.5	DISEÑO DE INTERFAZ HMI.....	17
2.5.1	<i>Características NEXTION</i>	17
2.5.2	<i>Conexión Entre La HMI Y La ESP32</i>	18
2.6	DISEÑO DETALLADO.....	19
2.6.1	<i>Cálculos De Resistencia Equivalente, Corriente Y Potencia</i>	19
2.6.2	<i>Resistencia equivalente</i>	20
2.6.3	<i>Corriente Total</i>	20
2.6.4	<i>Potencia Total</i>	20
2.6.5	<i>Datos Relevantes Para Cálculos De Materiales</i>	20
2.6.6	<i>Cálculo De Volumen De Materiales</i>	22
2.6.7	<i>Energía Necesaria</i>	23
2.6.8	<i>Tiempo De Calentamiento</i>	26
2.7	CRITERIO DE DISEÑO DEL SISTEMA DE CONTROL PID	27
2.7.1	<i>Determinación De Los Parámetros Térmicos</i>	28
2.7.2	<i>Controlador PID</i>	29
2.7.3	<i>Controlador Ajustado PI</i>	30
2.7.4	<i>Función De Transferencia Del PID en Lazo Cerrado</i>	30
2.7.5	<i>Controlador PID En La ESP32</i>	32
2.7.6	<i>Función De Control PWM</i>	33
2.7.7	<i>Implementación Del Ventilador</i>	33
CAPÍTULO 3		34
3.	RESULTADOS Y ANÁLISIS.....	34
3.1	RESULTADOS DEL ESTUDIO.....	34
3.1.1	<i>Datos Iniciales y Propiedades Físicas</i>	34
3.1.2	<i>Cálculo De Energía Requerida</i>	35
3.1.3	<i>Tiempo De Calentamiento</i>	36
3.2	CONEXIONES Y PRUEBAS	37
3.3	RESULTADO DE PID	38
3.4	VALIDACIÓN Y PRUEBAS DEL EQUIPO.....	39
3.4.1	<i>Caso 1 de Validación</i>	40

3.4.2	<i>Caso 2 de Validación</i>	41
3.5	PRESUPUESTO	41
CAPÍTULO 4		46
4. CONCLUSIONES Y RECOMENDACIONES.....		46
4.1	CONCLUSIONES.....	46
4.2	RECOMENDACIONES	48
BIBLIOGRAFÍA.....		49
APÉNDICES		52
4.3	DISEÑO DE CÁMARA DE SECADO EN AUTOCAD.....	52
4.4	PROCESO DE ELABORACIÓN DE MAQUETA	53
4.5	PANTALLA NEXTION 7”	55
4.6	RELE DE ESTADO SOLIDO SSR-25DA	55
4.7	SENSOR PT100	56
4.8	CÓDIGO ESP32.....	56

Abreviaturas

DEC	Distal End Cap
ESP32	Microcontrolador con Wi-Fi y Bluetooth integrado
ESPOL	Escuela Superior Politécnica del Litoral
HMI	Interfaz Hombre-Máquina
ISA	Insertion Section Assy
PID	Controlador Proporcional Integral Derivativo
UART	Universal Asynchronous Receiver-Transmitter
PWM	Modulación por Ancho de Pulso

Simbología

C_p	Capacidad Calorífica Específica (J/kg·K)
ρ	rho, densidad (kg/m ³).
Q	Energía en Joules (J)
P	Potencia (W)
M	Masa de aire
c	Capacidad térmica
τ	Constante de tiempo
K_p	Ganancia proporcional
K_i	Constante integral
K_d	Constante Derivativo
T_i	Tiempo integral
T_d	Tiempo derivativo
N	Filtro derivativo
Q_{agua}	Calor específico requerido para el agua.
Q_{acero}	Calor requerido para calentar el acero.
Q_{aire}	Calor requerido para calentar el aire.
$P(t)$	Acción Proporcional.
$I(t)$	Acción Integral.
$D(t)$	Acción Derivativa.

Índice de figuras

Figura 1	3
Figura 2	4
Figura 3	5
Figura 4	5
Figura 5	7
Figura 6	8
Figura 7	12
Figura 8	13
Figura 9	14
Figura 10	15
Figura 11	16
Figura 12	17
Figura 13	18
Figura 14	21
Figura 15	21
Figura 16	22
Figura 17	22
Figura 18	27
Figura 19	27
Figura 20	31
Figura 21	32
Figura 22	37
Figura 23	38
Figura 24	40
Figura 25	41
Figura 26	53
Figura 27	54
Figura 29	55
Figura 30	55
Figura 31	56

Índice de tablas

Tabla 1	6
Tabla 2	18
Tabla 3	34
Tabla 4	35
Tabla 5	36
Tabla 6	39
Tabla 7	43

Índice de Planos

Plano 1	52
Plano 2	52
Plano 3	53

Índice de Códigos

Código 1 ESP32 - Arduino IDE.....	73
-----------------------------------	----

Capítulo I

1. Introducción

1.1 Descripción Del Problema

La empresa Perfectech, especializada en reparación y ventas de equipos médicos tanto para el área hospitalaria como el área endoscópica ubicada en la ciudad de Guayaquil, presenta una alta demanda de reparación de endoscopios utilizados en procedimientos mínimamente invasivos en el área de endoscopia en las instituciones médicas. No solo Perfectech sino las empresas en general han identificado un problema después de intervenciones quirúrgicas, los endoscopios pueden presentar acumulaciones de fluidos en su interior, provocado por daños en canales internos por insumos en mal estado, golpes, etc.

Esta acumulación puede provocar daños en los componentes internos del equipo y manchas en la imagen de este. Para solucionar estas complicaciones y proceder a la reparación, es esencial contar con un método eficiente de secado que elimine completamente la humedad. Además, cuando el anillo de epoxi del endoscopio está desgastado o con fisuras, se corre el riesgo de que, entre humedad, lo que requiere reemplazar el pegamento y someterlo a un proceso de secado para asegurar su correcto endurecimiento. Actualmente, el secado manual de los equipos toma muchas horas, generando tiempos muertos que retrasan las reparaciones, afectando la eficiencia.

1.2 Justificación Del Problema

El tiempo prolongado que actualmente toma el secado de los endoscopios durante las reparaciones representa un desafío para la empresa Perfectech, ya que retrasa la reparación de los equipos en el laboratorio y afecta la productividad del equipo técnico. Cada hora que un endoscopio permanece en proceso de secado es tiempo que podría destinarse a reparar otros equipos, lo que se demuestra menos eficiencia y mayores pérdidas económicas para la

empresa. Además, la humedad residual en los endoscopios puede favorecer el crecimiento de microorganismos, lo que pone en riesgo la salud de los pacientes.

Al implementar una cámara de secado, este proceso podrá realizarse de manera mucho más rápida, permitiendo que los equipos estén listos para continuar con su reparación en menor tiempo. Esto no solo optimizará el flujo de trabajo en el área de mantenimiento, sino que también ayudará a que la empresa maneje de mejor manera la alta demanda de reparaciones. El secado seguro del anillo de epoxi y otros componentes críticos evitará daños futuros, garantizando que los equipos se mantengan en condiciones óptimas.

También minimiza el riesgo de infecciones al asegurar que los endoscopios estén completamente secos y libres de bacterias antes de su uso en procedimientos médicos, contribuyendo así a la seguridad de los pacientes y a la eficiencia en las clínicas y hospitales.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar una cámara de secado para endoscopios mediante un microcontrolador que gestione el encendido y apagado automático de las resistencias térmicas por un sistema PID, optimizando el tiempo de secado para eliminar la humedad del equipo, destinado a la empresa Perfectech.

1.3.2 Objetivos Específicos

- Determinar el área óptima para la compatibilidad de la cámara de secado con los diversos modelos de endoscopios manejados por el laboratorio de la empresa.
- Diseñar un sistema de control PID donde las resistencias térmicas se enciendan y se apaguen de manera automática, ajustando la temperatura de manera precisa según los requisitos del usuario.

- Implementar una base de datos para gestionar y almacenar datos del proceso de secado, asegurando un registro de control y mejora enfocado en asegurar la calidad del pegamento y la aplicación.

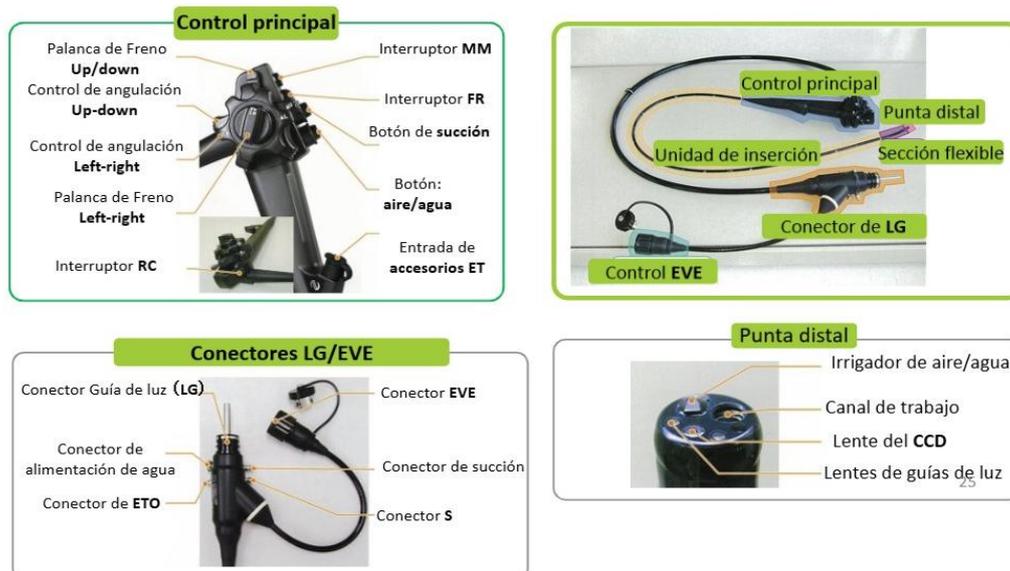
1.4 Marco Teórico

1.4.1 Fundamentos De La Endoscopia

La endoscopia es una técnica médica que permite visualizar dentro del cuerpo humano por medio de un tubo flexible y delgado. Este instrumento se denomina endoscopio, donde en la parte superior se encuentra una pequeña cámara y dos guías de luz en su extremo. Por medio del endoscopio se pueden introducir instrumentos-insumos a través del canal de trabajo o también llamado canal de biopsia para poder realizar procedimientos quirúrgicos, como se puede observar en la siguiente imagen los canales en la punta distal del endoscopio.

Figura 1

Partes de un endoscopio flexible



Nota. Tomado de FujiFilm (2024)

Según el editorial de la Sociedad de Gastroenterología del Perú, el autor Mendoza (2024) explora la evolución de la eco endoscopia, que celebra 50 años de aplicación en el diagnóstico y tratamiento de enfermedades gastrointestinales, destacando su impacto y proyección en la práctica médica.

Del mismo modo, De la Torre (2001) analiza la alucinante evolución de la endoscopia, incitado por avances tecnológicos que han ido mejorando su precisión y efectividad en procedimientos. Este desarrollo no solo ha permitido realizar diagnósticos y tratamientos de manera más eficiente y segura, sino también ampliar el alcance de la endoscopia en la exploración dentro del cuerpo humano. Según el autor, los cambios en esta área han transformado profundamente las capacidades de los procedimientos médicos en el tratamiento de múltiples órganos.

1.4.2 Detección De Fugas/Humedad En Endoscopios

De acuerdo con Sosa et al. (2019) siempre debe realizarse una limpieza previa después de finalizar el procedimiento con el endoscopio. Así, los autores describen esta etapa como un paso muy importante para eliminar materia orgánica y biopelículas de microorganismos incrustados en el endoscopio, recalcando la importancia de pasar un paño húmedo con desinfectante en el tubo de inserción, una medida esencial para mantener la seguridad y el manejo correcto de los equipos. Asimismo, la prueba de fuga posterior a la desinfección del tubo de inserción es importante para garantizar el estado de los canales durante el procedimiento.

Figura 2

Prueba de Fuga



Nota. Elaboración propia

Ante esto, la guía rápida Sinmedica (2021) detalla los casos comunes de presencia de fugas en el endoscopio, como objetos afilados (agujas, pinzas, uñas) y factores donde el riesgo es grande, como el desgaste del tubo de inserción por un mal reprocesamiento, por objetos cortopunzantes e incluso por mordeduras de pacientes en procedimientos de gastroenterología.

Figura 3

Fuga externa por averías en el endoscopio

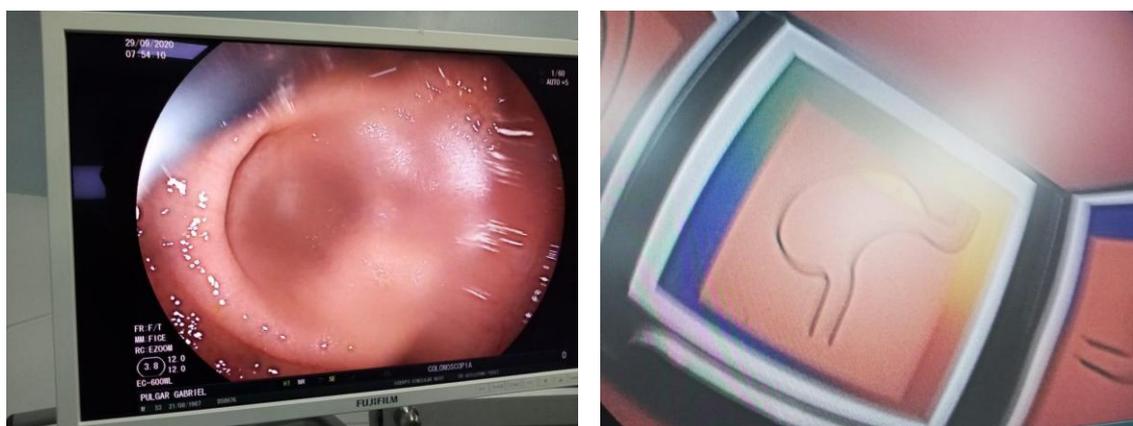


Nota. Elaboración en el departamento técnico de la empresa Perfectech (2024)

Cuando el endoscopio presencia fuga de aire, debe realizarse una prueba de imagen para analizar el caso, a continuación de presencia una prueba de imagen con humedad.

Figura 4

Endoscopio con humedad presentada en canales del Endoscopio



Nota. Elaboración en el departamento técnico de la empresa Perfectech (2024)

1.4.3 Sensores De Temperatura Y Humedad (PT100 Y SHT31)

1.4.3.1 Sensor De Temperatura PT100.

El sensor de temperatura varia con su resistencia a los cambios de temperatura a la que está expuesto. El material del elemento en el que se encuentra envuelto entre capas de material aislante asegura su durabilidad y precisión incluso en condiciones de operación extrema.

Según Fuji electric (2023), las sondas PT100 se clasifican en diferentes clases de precisión, como Clase A y Clase B, cada una con tolerancias específicas en función de la temperatura. A continuación, se muestra una imagen donde se muestra la precisión tanto de la Clase A y la Clase B en función de la temperatura.

Tabla 1

Tolerancias de Clase A y Clase B, fuente: Fuji Electric

Temperatura en °C	Valores básicos en Ω	Errores admisibles (tolerancias)			
		Clase A		Clase B	
		°C	Ω	°C	Ω
-200	18,52	± 0,55	± 0,24	± 1,3	± 0,56
-100	60,26	± 0,35	± 0,14	± 0,8	± 0,32
0	100,00	± 0,15	± 0,06	± 0,3	± 0,12
100	138,51	± 0,35	± 0,13	± 0,8	± 0,30
200	175,86	± 0,55	± 0,20	± 1,3	± 0,48
300	212,05	± 0,75	± 0,27	± 1,8	± 0,64
400	247,09	± 0,95	± 0,33	± 2,3	± 0,79
500	280,98	± 1,15	± 0,38	± 2,8	± 0,93
600	313,71	± 1,35	± 0,43	± 3,3	± 1,06
650	329,64	± 1,45	± 0,46	± 3,6	± 1,13
700	345,28	-	-	± 3,8	± 1,17
800	375,7	-	-	± 4,3	± 1,18
850	390,48	-	-	± 4,6	± 1,34

Nota. Tomado de Universidad de Sevilla (2014)

1.4.3.2 Sensor De Temperatura Y Humedad SHT30-1.

Mañas Torres (2019), investiga el sensor SHT31 donde ofrece una integración sencilla tanto a nivel de hardware como de software, facilitando su uso en proyectos con microcontroladores como Arduino mediante la interfaz I2C. Además, el protocolo I2C

permite la comunicación eficiente entre el sensor y otros componentes en un sistema de múltiples maestros y esclavos.

De acuerdo con las hojas de datos de Sensirion (s.f.), el sensor SHT30-1 cuenta con una salida digital calibrada. Su rango de voltaje de 2.4V a 5.5V, con velocidades de hasta 1MHz, además su precisión estándar es de $\pm 1.5\%$ en humedad relativa y $\pm 0.2\text{ }^{\circ}\text{C}$ en temperatura.

Figura 5

Sensor SHT31



Nota. Tomado de Amazon (2024)

1.4.4 Control De Temperatura PID

En el uso de microcontroladores, el control de temperatura mediante un PID permite mantener un nivel térmico constante empleando control proporcional-integral-derivativo. Con la ayuda de un sensor y un módulo como el MAX31865 para la detección de temperatura, el sistema lee los datos y ajusta la cantidad de energía suministrada para alcanzar la temperatura deseada, evitando oscilaciones excesivas (Arduino, 2018).

En el controlador PID, como describen Zayas et al. (2020), cada sección tiene una función específica, como se describe enseguida:

1.4.4.1 Acción Proporcional $P(t)$

Responde directamente al error a través de la ganancia de proporcionalidad, k . Esta acción, aunque genera oscilaciones si se usa de manera exagerada, ayuda a corregir el error de manera rápida.

1.4.4.2 Acción Integral $I(t)$

Acumula el error según el tiempo, ponderando a través del parámetro de ganancia integral, k_i . Esta acción ayuda a eliminar el error en estado estacionario, asegurando que el sistema pueda alcanzar el valor que tenga como objetivo, estabilizarse.

1.4.4.3 Acción Derivativa $D(t)$

Anticipa el cambio en el error, siempre considerando la tasa de variación mediante la ganancia derivativa, k_d . Esta acción reduce la rapidez con que el sistema responde a cambios en el error, además suaviza las oscilaciones mejorando la estabilidad sin eliminar por completo el error.

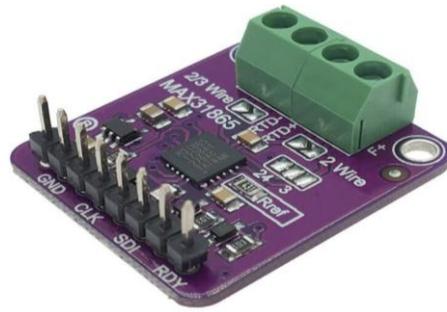
1.4.5 Sensor MAX31865

Como describe U Electronics (s.f.), el detector de temperatura MAX31865 es un dispositivo avanzado diseñado para operar con el PT100 o PT1000. Es ampliamente utilizado en la industria por su precisión y fiabilidad en la medición de temperatura. Este módulo destaca por su capacidad de ofrecer lecturas estables y exactas, lo cual es eficiente en aplicaciones que requieren un control térmico seguro, tales como procesos industriales, proyectos de automatización o en una cámara de secado para endoscopios.

El MAX31865 funciona mediante un sistema de conversión y procesamiento de la señal, capturando las variaciones del sensor PT100 y transformándolas en datos de temperatura precisos. Además, el módulo utiliza una interfaz de comunicación SPI, que permite integrarse fácilmente con microcontroladores y otros dispositivos de control.

Figura 6

Módulo MAX31685



Nota. Tomado de U Electronics (s.f.)

1.4.6 Relé De Estado Solida (SSR-25DA)

Según las especificaciones dadas por Fuji electric (2023), los relés utilizados para conmutar cargas resistivas o inductivas de alta potencia han evolucionado hasta emplear dispositivos semiconductores, como transistores, tiristores o MOSFET, en lugar de los mecanismos electromecánicos. Este cambio permite aumentar la durabilidad y un control más eficiente. La implementación de componentes semiconductores minimiza las interrupciones mejorando y optimizando la estabilidad y eficiencia del sistema de control.

1.4.7 Normativas y estándares internacionales sobre el reprocesamiento de endoscopios

Para asegurar la seguridad de los pacientes y el correcto procedimiento en cirugías es necesario conocer las normativas internacionales de los Ministerios de Salud. Se sabe que los equipos médicos en su mayoría pueden ser reutilizados, estos están sujetos a resguardar microorganismos, residuos los cuales pueden ocasionar problemas de salud a los pacientes si no se los tratan adecuadamente, varias organizaciones han desarrollado instructivos que tratan buenos protocolos en lo que corresponde a limpieza, esterilización y desinsectación. Se detalla recomendaciones de dos organizaciones, exponiendo su análisis.

Sobre estos protocolos, Joselyn Cerezo (2024) sostiene con su investigación lo siguiente:

Según la Organización Mundial de Gastroenterología (WGO siglas en inglés), el procesamiento de los endoscopios debe realizarse en condiciones ambientales,

incluidas la temperatura, humedad, y manteniendo un ambiente estéril. De la misma manera, se recomienda utilizar productos químicos de alta calidad para garantizar la seguridad del sistema y prevenir el crecimiento de microorganismos. (p. 37)

Otra organización donde, Joselyn Cerezo (2024) sostiene nuevamente con su investigación el siguiente protocolo:

La Sociedad Mexicana de Endoscopia Gastrointestinal (AMEG) resalta la importancia de mantener la humedad por debajo del 60% y asegurar 10 cambios de aire por hora en la sala de procesamiento. Estos métodos no solo garantizan un entorno seguro, sino que también respaldan la trazabilidad al documentar cada proceso en detalle. (p. 37)

Capítulo 2

2. Metodología

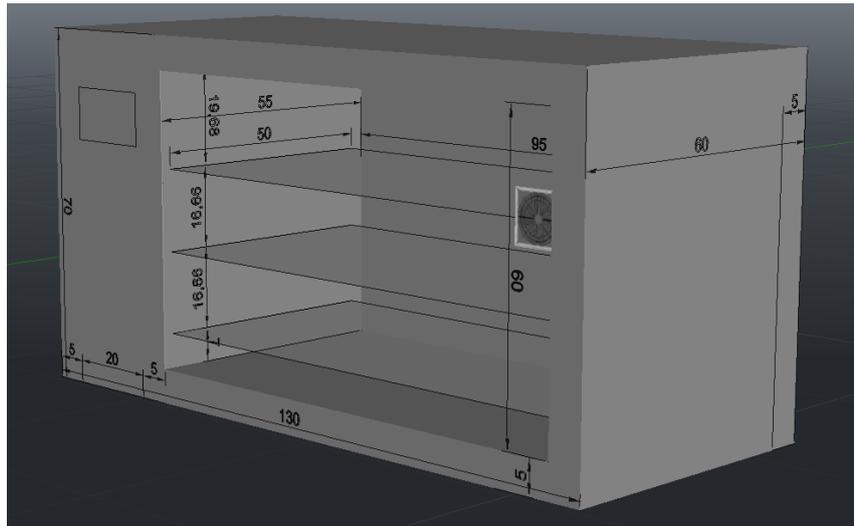
2.1 Selección De Método Y Estrategia De Elaboración Implementada

Frente a la problemática de tener equipos de endoscopia con humedad por presencias de fuga, daños en el tubo de inserción o en la punta distal, posterior a procedimientos, reprocesamientos o golpes, se llevó a cabo una forma de solucionar este inconveniente con el objetivo de optimizar el proceso de reparación y garantizar un flujo mayor de reparaciones en el departamento técnico. La primera opción analizada consistió en utilizar una lámpara con un foco incandescente, donde el equipo de endoscopia se colocaba debajo de la lámpara y se dejaba secar por un periodo de uno a dos días. Esta opción fue evaluada en términos de eficacia del secado, tiempo necesario para lograr el secado completo y los riesgos asociados al estar expuesto a golpes o accidentes. Posterior al análisis, se consideraron criterios como la optimización del tiempo de secado para las reparaciones donde se decidió explorar alternativas que ofrecieran un secado más eficiente, seguro y controlado en tiempos reducidos, pues el uso del foco incandescente no resultó viable para llevar un secado seguro.

El proceso por medio de una cámara de secado donde se evaluaron las medidas pertinentes la cual el diseño consistió en optimizar el tiempo de secado para posteriormente ir a la reparación, teniendo el equipo una capacidad para secar tres endoscopios dentro del equipo a una temperatura de 40 grados. Las medidas internas del equipo son de 1m x 0.6m x 0.55m (ancho x altura x profundidad), con una separación entre bandejas de 16.66 cm. Estas medidas fueron establecidas considerando la comodidad para el ingreso de los endoscopios y los diversos modelos de equipos, donde las dimensiones varían según el tipo y modelo. Se consideró la compatibilidad con modelos estándar como Olympus, Fujifilm y Pentax, cuyos canales de trabajo tienen un diámetro promedio de 2.8 a 3.8 mm.

Figura 7

Diseño en AutoCAD de Cámara de Secado



Nota. Elaboración propia

2.2 Experimentación

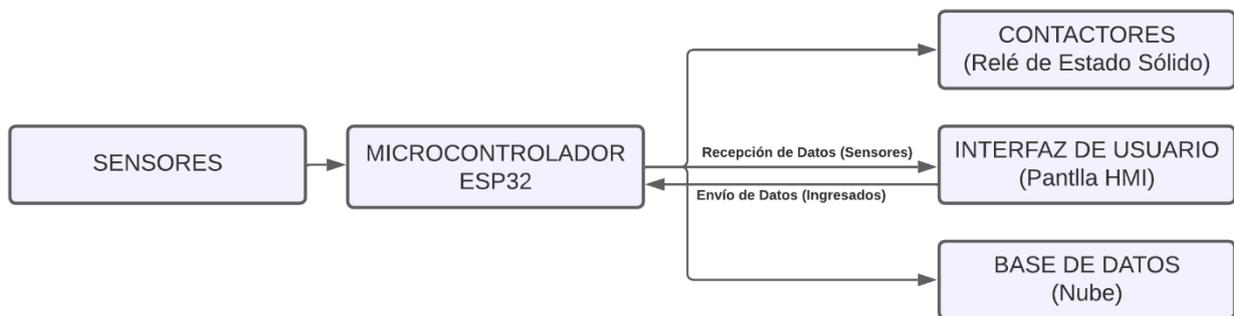
El diagrama de bloques ilustrado en la imagen 6 muestra el sistema de control para la cámara de secado. Este sistema se basó en la recolección de datos mediante los sensores de temperatura para su correcto funcionamiento, garantizando un control eficiente. Los sensores y la pantalla HMI recopilan los datos que son enviados al microcontrolador ESP32. La pantalla HMI no solo envía datos, sino que también funciona como receptor para mostrar la variación de temperatura en todo momento y observar un temporizador, además para realizar una gráfica de temperatura comparándola con el valor seteado. La ESP32 procesa la información y utiliza un modelo PID para garantizar una temperatura estable al valor seteado. Con estos ajustes, la ESP32 acciona los relés de estado sólido que controlan las resistencias térmicas, la cual funcionan como calentadores y el ventilador que funciona como un método para distribuir el aire caliente por toda el área de maneja constante y fluida, asegurando óptimas condiciones dentro del área de trabajo.

La HMI envía comandos hacia la ESP32 para la realización del sistema, la cual permite al usuario ingresar diferentes parámetros claves, como el nombre del cliente, técnico,

modelo, número de serie, cantidad de pegamento utilizado, temperatura y hora de aplicación del pegamento. Esta información es enviada al microcontrolador y registrada en la base de datos dependiendo el tipo de proceso que se desea realizar, proporcionando un historial importante.

Figura 8

Diagrama de Bloques



Nota. Elaboración propia

2.3 Diagrama De Flujo

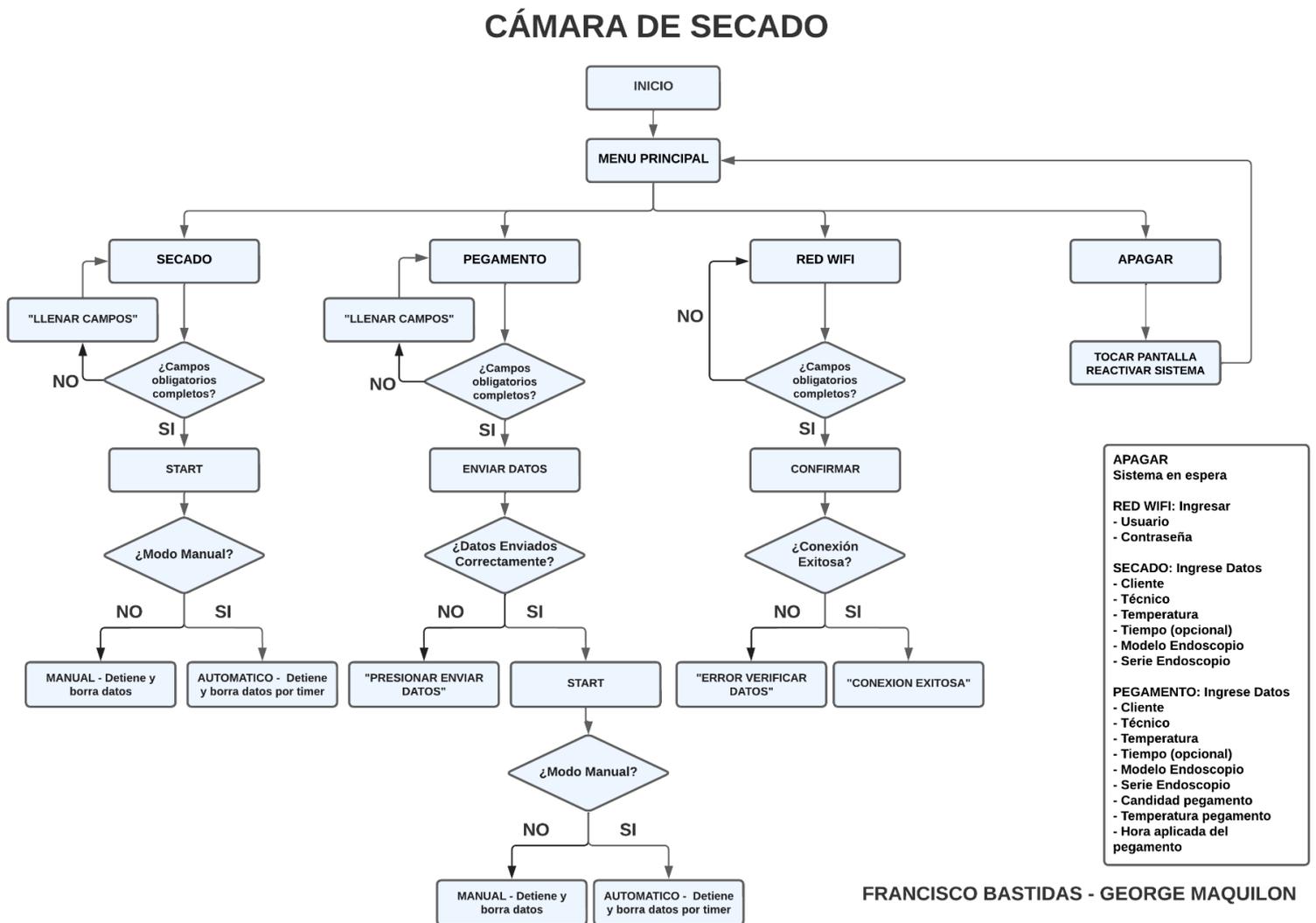
El diagrama de flujo ilustrado en la imagen 9 muestra el proceso del sistema general para la cámara de secado, empezando con el acceso principal de 4 opciones: Secado, Pegamento, Red Wifi y Apagar. En ambos modos de Secado y Pegamento, el usuario ingresa parámetros que son esenciales para el inicio del proceso, campos como la temperatura y opcionalmente, el tiempo. Si faltan datos, el sistema notificara que faltan datos por ingresar. Una vez iniciado, el proceso inicia el PID ajustando los contactores (Resistencias Térmicas y Ventilador) con base a las lecturas de los sensores, finalizando manualmente con STOP para poner en apagado todos los sistemas con excepción del ventilador que estará encendido para ayudar con la ventilación de temperatura.

En el modo Pegamento, se ingresan datos adicionales a comparación del método Secado, además se debe presionar el botón ENVIAR DATOS para que se ingresen a la base de datos, el botón es obligatorio presionarlo para poder presionar START. En el modo Red Wifi, el usuario ingresa el usuario y contraseña de la red y confirma la conexión. Si la

conexión es exitosa, el sistema notifica al usuario; de no ser así, muestra un mensaje indicando error de conexión o clave incorrecta. En la opción Apagar, el sistema entra en modo de espera, apagando los componentes excepto la ESP32, que queda en espera para reactivar el sistema tocando la pantalla.

Figura 9

Diagrama de Flujo



Nota. Elaboración propia

2.4 Diseño Conceptual

El diseño de la cámara de secado se centró en optimizar el proceso de secado de los endoscopios, de tal manera que asegure la distribución de la temperatura de manera controlada.

a) Sistema de calefacción

Compuesto por 10 resistencias térmicas de 84.4Ω , distribuidas en paralelo. Estas son las encargadas de generar el calor necesario para que la temperatura se eleve dentro del área. La elección de conectarlas en paralelo garantiza que la corriente se divida de manera uniforme entre las resistencias

b) Control de temperatura

El control se realizó mediante un sensor MAX31865 (PT100), el cual está conectado a una ESP32. El sensor se encargará de medir la temperatura con precisión dentro del área de trabajo, proporcionando de esta manera una lectura continua que será utilizado para el funcionamiento del PID. Como objetivo para el secado de endoscopios, llegar a una temperatura de 40 grados y tener un control preciso para no dañar otros componentes internos como la cámara, canales o guías de luz.

Figura 10

Ficha Técnica de Control de Temperatura

Operating Environment, Transport Environment and Storage Environment

◆ Operating Environment

Temperature	+10 to +40°C (50 to 104°F)
Humidity	30 to 85%RH (no dew condensation)
Pressure	70 to 106 kPa (10.2 to 15.4 psia or 525 to 795 mmHg) (within range of atmospheric pressure)

Nota. Tomado del Manual FujiFilm (2025)

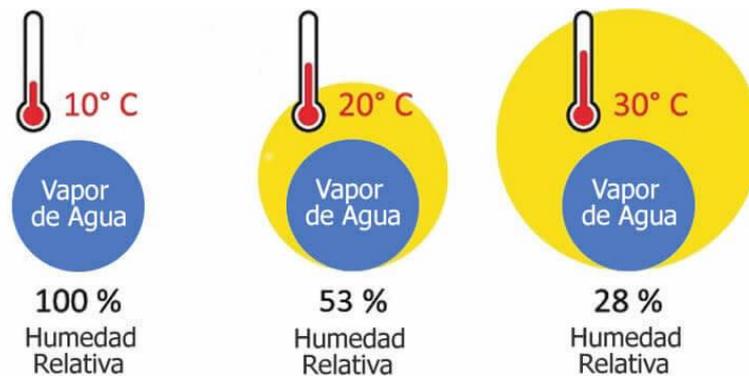
c) Monitoreo de humedad

Para tener garantizado que el proceso de secado sea completo, se integró un sensor de humedad FS304-SHT31, el cual mide las condiciones dentro de la cámara de secado, especialmente en las zonas donde el equipo de endoscopia se encuentra ubicado. Según

Alessandro (2024), la humedad relativa es la relación entre la cantidad que existe de agua en el aire, donde el valor se expresa en porcentaje, entre 0% y 100%.

Figura 11

Comparativos de Humedad



Nota. Tomado de Hogar Seco (s.f)

Según la página Hogar Seco (s.f), la cantidad del vapor de agua depende de la temperatura que se encuentra. Como se puede apreciar en la imagen, a mayor temperatura menos humedad.

d) Sistema de control

El sistema es principalmente gestionado por una ESP32, mediando un algoritmo PID para regular la temperatura. El algoritmo centrado en el ajuste de las resistencias en función de la temperatura medidas por el sensor, asegurando que la temperatura dentro del área se mantenga en el rango de temperatura establecido. El controlador PID actúa de manera continua ajustado por una señal PWM proveniente de la ESP32 en función de la diferencia entre la temperatura actual y la deseada, de esa manera minimizando el error.

e) Distribución del aire

Un aspecto importante en el diseño es priorizar que el aire circule de manera correcta por toda el área de trabajo de la cámara de secado. Para ello, se instaló un ventilador con la capacidad de garantizar la distribución homogénea de la temperatura y controlar la humedad dentro del área de trabajo.

2.5 Diseño De Interfaz HMI

Para el diseño de la interfaz, se utilizó una **pantalla táctil Nextion de 7 pulgadas** y su **Nextion Editor**, una plataforma gráfica que permite crear interfaces visuales de manera eficiente y sencilla. Esta pantalla es compatible con microcontroladores como el **ESP32**, lo que facilita su integración en el sistema.

2.5.1 Características NEXTION

- Funciona con plataformas como **Arduino, ESP32, Raspberry Pi, y FPGA**.
- Permite el control de sensores, comandos y la visualización en gráficos de datos en tiempo real.
- El lenguaje de programación de Nextion es fácil de usar, lo que permite conectar rápidamente la interfaz con la ESP32.
- Los botones, gráficos y controles son fáciles de usar y permiten una visualización clara de los datos.

Figura 12

Diseño de la Interfaz en la HMI



Nota. Elaboración propia

2.5.2 Conexión Entre La HMI Y La ESP32

Una vez diseñada la interfaz en el editor de Nextion, se procedió a establecer la conexión física y lógica entre la pantalla HMI y la ESP32. La comunicación se configuró mediante el protocolo de comunicación **UART (Universal Asynchronous Receiver-Transmitter)**.

En el código de la ESP32, se definieron las variables a cada componente de la HMI, como botones, cuadros de texto y gráficos, utilizando la librería **Nextion.h**. Cada elemento de la interfaz fue integrado a una función específica en la ESP32

Además, se estableció una lista de elementos interactivos (**nex_listen_list[]**) para detectar eventos en tiempo real, como presionar botones o actualizar gráficos. Esta configuración garantizó un flujo continuo de datos entre la pantalla HMI y la ESP32, optimizando el control y monitoreo del sistema.

Tabla 2

Conexión HMI y ESP32

HMI	ESP32
TX	RX
RX	TX
GND	GND
5V	3-5V

Nota. Elaboración propia

Figura 13

Configuración entre la HMI y ESP32

```
// Configuración de la pantalla Nextion

// Declaración de botones de la página 0
NexButton secadoButton = NexButton(0, 1, "SECADO");
NexButton pegamentoButton = NexButton(0, 2, "PEGAMENTO");
NexButton apagarButton = NexButton(0, 4, "APAGAR");
NexText horaFecha = NexText(0, 5, "HORA_FECHA");

// Declaración de elementos en la página 1
NexButton iniciarPage1 = NexButton(1, 2, "INICIO");
NexButton detenerPage1 = NexButton(1, 3, "STOP");
NexText variacionTempPage1 = NexText(1, 21, "VARIACION_TEMP");
NexText variacionHumPage1 = NexText(1, 7, "VARIACION_HUM");
NexWaveform waveformTempPage4 = NexWaveform(3, 3, "WAVEFORM_TEMP");
NexWaveform waveformHumPage3 = NexWaveform(3, 4, "WAVEFORM_HUM");
NexNumber horaPage1 = NexNumber(1, 29, "HORA");
NexNumber minutosPage1 = NexNumber(1, 30, "MINUTOS");
NexButton enviar_datos = NexButton(1, 20, "ENVIAR_DATOS");

// Declaración de elementos en la página 2
NexText inRed = NexText(2, 5, "IN_RED"); // Cuadro para ingresar la nueva red
NexText inPassword = NexText(2, 6, "IN_PASSWORD"); // Cuadro para ingresar la contraseña de la nueva red
NexText wifiActual = NexText(2, 7, "WIFI_ACTUAL"); // Cuadro para mostrar la red WiFi actual
NexText passActual = NexText(2, 10, "PASS_ACTUAL"); // Cuadro para mostrar la contraseña actual
NexButton confirmarButton = NexButton(2, 12, "CONFIRMAR"); // Botón para confirmar nueva conexión
```

Nota. Elaboración propia

a) NexButton

Representa un botón en la interfaz de usuario de la pantalla HMI, que permite detectar cuando se presiona, se la asocia a una acción en el código de la ESP32 para los eventos.

b) NexWaveform

Representa una gráfica de onda, la HMI permite mostrar datos en tiempo real medidos por sensores de forma gráfica.

c) NexText

Representa cuadros de texto en la pantalla HMI, permite mostrar o realizar modificaciones desde el código, además de leer texto ingresado directamente desde la HMI.

2.6 Diseño Detallado

2.6.1 Cálculos De Resistencia Equivalente, Corriente Y Potencia

Se utilizaron 10 resistencias térmicas conectadas en paralelo para distribuir uniformemente el calor.

2.6.2 Resistencia equivalente.

La resistencia equivalente (R_{eq}) se calcula para las 6 resistencias de 84.4 Ω conectadas en paralelo.

$$R_{eq} = \frac{R_{individual}}{N} = \frac{84.4}{10} = 8.44 \text{ } [\Omega] \quad (2.1)$$

2.6.3 Corriente Total.

La corriente total (I_{total}) que fluye a través del sistema se calcula usando la ley de Ohm:

$$I_{total} = \frac{V}{R_{eq}} = \frac{24}{8.44} = 2.84 \text{ } [A] \quad (2.2)$$

2.6.4 Potencia Total.

La potencia total disipada por las resistencias (P_{total}) se determina usando la fórmula:

$$P_{total} = V * I_{total} = 24 * 2.84 = 68.16 \text{ } [W] \quad (2.3)$$

2.6.5 Datos Relevantes Para Cálculos De Materiales

2.6.5.1 Propiedad Del Acero Inoxidable.

Usando un Colonoscopio como modelo de referencia para llevar al máximo los datos, EC-760 VL, con capacidad de canal de trabajo de 3.2 mm y de longitud de 1690 mm, es decir 0.32 cm de diámetro del canal de trabajo y 169 cm de longitud del tubo de inserción.

$$C_p = 468 \text{ } J/Kg \text{ } ^\circ C$$

Con un peso estimado de 5 Kg, se obtiene una masa de 15 Kg dando en cuenta la capacidad de 3 endoscopios en la cámara de secado.

Figura 14*Propiedades termo físicas de la materia*

4. Propiedades termofísicas de la materia

Tabla 4.1: Propiedades termofísicas de sólidos metálicos a 300 K

Descripción	ρ (kg/m ³)	c_p (J/kg·K)	k (W/m·K)	$\alpha \cdot 10^6$ (m ² /s)
Acero puro	7870	447	80.2	23.1
Acero al carbono	7854	434	60.5	17.7
Acero al carbono-silicio	7817	446	51.9	14.9
Acero al carbono-Manganeso-silicio	8131	434	41.0	11.6
Acero con cromo (bajo)	7822	444	37.7	10.9
Acero inoxidable, AISI 302	8055	480	15.1	3.9
Acero inoxidable, AISI 304	7900	477	14.9	3.9
Acero inoxidable, AISI 316	8238	468	13.4	3.5
Acero inoxidable, AISI 347	7978	480	14.2	3.7
Aluminio puro	2702	903	237.0	97.1
Aluminio, aleación 2024-T6	2770	875	177.0	73.0
Aluminio, aleación 195, vaciado	2790	883	168.0	68.2

Nota. Tomado de Universidad de Sevilla (2014)

2.6.5.2 Propiedad Del Aire.

Se describe la siguiente fórmula:

$$\rho = 1.2 \text{ Kg}/\text{m}^3$$

$$C_p = 1000 \text{ J}/\text{Kg} * \text{K}$$

Usando la densidad del aire ($\rho = 1.2 \text{ Kg}/\text{m}^3$):

$$m = \rho \times V \quad (2.4)$$

$$m = 1.2 \times 0.33$$

$$m = 0.396 \text{ [Kg]}$$

Figura 15*Propiedades del Aire*

3.3. Valores aproximados para propiedades del aire y del agua

Fluido	Densidad (kg/m ³)	c_p (kJ/kg·K)	c_v (kJ/kg·K)	Masa molecular (kg/mol)
Aire seco	1.2	1.0	0.717	28.96
Agua líquida	1000	4.18	4.18	18.02

Calor latente de vaporización del agua: $\Delta h_{l-g,w} = 2500 \text{ kJ} / \text{kg}$

Nota. Tomado de Universidad de Sevilla (2014)

2.6.5.3 Propiedad Del Agua.

Volumen dentro de los canales del endoscopio:

$$\rho = 1000 \text{ Kg}/\text{m}^3$$

$$C_p = 4180 \text{ J}/\text{Kg} * \text{K}$$

Figura 16

Propiedades del Agua

3.3. Valores aproximados para propiedades del aire y del agua

Fluido	Densidad (kg/m ³)	c _p (kJ/kg·K)	c _v (kJ/kg·K)	Masa molecular (kg/mol)
Aire seco	1.2	1.0	0.717	28.96
Agua líquida	1000	4.18	4.18	18.02

Calor latente de vaporización del agua: $\Delta h_{l-g,w} = 2500 \text{ kJ} / \text{kg}$

Nota. Tomado de Universidad de Sevilla (2014)

2.6.6 Cálculo De Volumen De Materiales

2.6.6.1 Volumen De La Cámara.

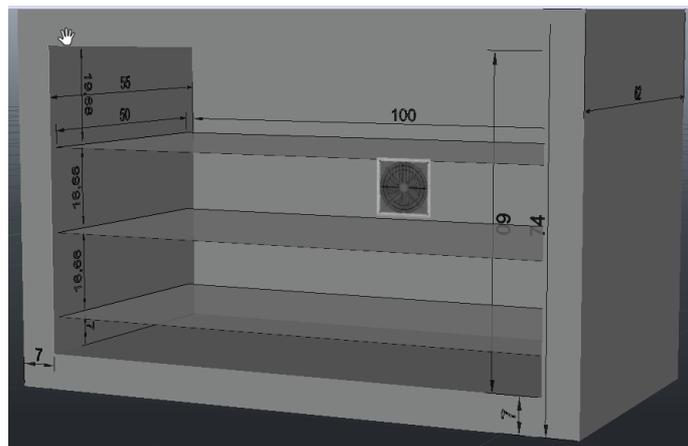
Dados los valores de ancho, altura y profundidad (1m x 0.6m x 0.55m), el volumen es:

$$V = 1m \times 0.6m \times 0.55m \quad (2.5)$$

$$V = 0.33 \text{ [m}^3\text{]}$$

Figura 17

Diseño con mediciones en AutoCAD



Nota. Elaboración propia

2.6.6.2 Volumen Del Canal De Trabajo.

Tomando como referencia el modelo establecido, se conoce su diámetro del canal de trabajo y la longitud del tubo de inserción, se obtiene su volumen.

$$V = \pi \times r^2 \times h \quad (2.6)$$

$$V = \pi \times 0.16^2 \times 169$$

$$V = 13.6 \text{ [cm}^3\text{]}$$

$$V = 13.6 \text{ [ml]}$$

2.6.7 Energía Necesaria

La energía (Q) para calentar el aire se calcula considerando una variación de temperatura ($\Delta T = T_{\text{final}} - T_{\text{inicial}}$)

$T_{\text{final}} \rightarrow$ Temperatura OBJETIVO

$T_{\text{inicial}} \rightarrow$ Temperatura AMBIENTE del equipo

$$\Delta T = T_{\text{final}} - T_{\text{inicial}} = 40 - 20 = 20 \text{ }^\circ\text{C} \quad (2.7)$$

Dado el estudio, se tomó en cuenta tres situaciones diferentes para la obtención de los cálculos, los escenarios establecidos consisten en:

- Peor de los casos, donde todos los canales estén inundados
- Caso promedio, donde solo exista un 50% de humedad en los canales.
- Mejor de los casos, donde exista un 20% de humedad en los canales.

2.6.7.1 Peor De Los Casos.

Tomando en consideración donde los canales están inundados al 100%, se debe tomar en cuenta la sumatoria del volumen del agua de los canales para la obtención del volumen total.

Canal de Trabajo + Canal de Succión + Canal de Aire + Canal de Agua

$$13.6 \text{ [ml]} + 2 - 5 \text{ [ml]} + 2 - 5 \text{ [ml]} + -5 - 10 \text{ [ml]} = 35 \text{ [ml]}$$

Donde usando 3 colonoscopios: $35 \text{ [ml]} \times 3 = 105 \text{ [ml]} \rightarrow 0.105 \text{ Kg}$

Una vez obtenido los cálculos de volumen necesarios, se procede a obtener la energía necesaria:

$$Q_{\text{agua}} = m \times C_p \times \Delta T \quad (2.8)$$

$$Q_{\text{acero}} = m \times C_p \times \Delta T \quad (2.9)$$

$$Q_{\text{aire}} = m \times C_p \times \Delta T \quad (2.10)$$

$$Q_{\text{agua}} = 0.105 \text{ Kg} \times 4180 \text{ J/Kg} \cdot \text{K} \times 20 \text{ }^\circ\text{C} = 8778 \text{ [J]}$$

$$Q_{\text{acero}} = 15 \text{ Kg} \times 468 \text{ J/Kg} \cdot \text{C} \times 20 \text{ }^\circ\text{C} = 140400 \text{ [J]}$$

$$Q_{\text{aire}} = 0.396 \text{ Kg} \times 1000 \text{ J/Kg} \cdot \text{K} \times 20 \text{ }^\circ\text{C} = 7920 \text{ [J]}$$

$$Q_{\text{TOTAL}} = Q_{\text{agua}} + Q_{\text{acero}} + Q_{\text{aire}} \quad (2.11)$$

$$Q_{\text{TOTAL}} = 8778 \text{ [J]} + 140400 \text{ [J]} + 7920 \text{ [J]}$$

$$Q_{\text{TOTAL}} = 157098 \text{ [J]}$$

2.6.7.2 Caso Promedio.

Una vez obtenido los cálculos de volumen necesarios, se procede a obtener la energía necesaria, tomando en consideración el cambio de la masa con un 50% de humedad.

$$0.105 \text{ Kg} \times 50\% = 0.0525 \text{ Kg}$$

$$Q_{\text{agua}} = m \times C_p \times \Delta T$$

$$Q_{\text{acero}} = m \times C_p \times \Delta T$$

$$Q_{\text{aire}} = m \times C_p \times \Delta T$$

$$Q_{\text{agua}} = 0.0525 \text{ Kg} \times 4180 \text{ J/Kg} \cdot \text{K} \times 20 \text{ }^\circ\text{C} = 4389 \text{ [J]}$$

$$Q_{\text{acero}} = 15 \text{ Kg} \times 468 \text{ J/Kg} \cdot \text{C} \times 20 \text{ }^\circ\text{C} = 140400 \text{ [J]}$$

$$Q_{\text{aire}} = 0.396 \text{ Kg} \times 1000 \text{ J/Kg} \cdot \text{K} \times 20 \text{ }^\circ\text{C} = 7920 \text{ [J]}$$

$$Q_{\text{TOTAL}} = Q_{\text{agua}} + Q_{\text{acero}} + Q_{\text{aire}}$$

$$Q_{\text{TOTAL}} = 4389 \text{ [J]} + 140400 \text{ [J]} + 7920 \text{ [J]}$$

$$Q_{\text{TOTAL}} = 152709 \text{ [J]}$$

2.6.7.3 *Mejor De Los Casos.*

Una vez obtenido los cálculos de volumen necesarios, se procede a obtener la energía necesaria, tomando en consideración el cambio de la masa con un 20% de humedad.

$$0.105 \text{ Kg} \times 20\% = 0.021 \text{ Kg}$$

$$Q_{\text{agua}} = m \times C_p \times \Delta T$$

$$Q_{\text{acero}} = m \times C_p \times \Delta T$$

$$Q_{\text{aire}} = m \times C_p \times \Delta T$$

$$Q_{\text{agua}} = 0.021 \text{ Kg} \times 4180 \text{ J/Kg} \cdot \text{K} \times 20 \text{ }^\circ\text{C} = 1755.6 \text{ [J]}$$

$$Q_{\text{acero}} = 15 \text{ Kg} \times 468 \text{ J/Kg} \cdot \text{C} \times 20 \text{ }^\circ\text{C} = 140400 \text{ [J]}$$

$$Q_{\text{aire}} = 0.396 \text{ Kg} \times 1000 \text{ J/Kg} \cdot \text{K} \times 20 \text{ }^\circ\text{C} = 7920 \text{ [J]}$$

$$Q_{\text{TOTAL}} = Q_{\text{agua}} + Q_{\text{acero}} + Q_{\text{aire}}$$

$$Q_{\text{TOTAL}} = 1755.6 \text{ [J]} + 140400 \text{ [J]} + 7920 \text{ [J]}$$

$$Q_{\text{TOTAL}} = 150075.6 \text{ [J]}$$

2.6.8 Tiempo De Calentamiento

2.6.8.1 Peor De Los Casos.

Dividiendo la energía de los casos establecidos por la potencia del sistema:

$$t = \frac{Q}{P} \quad (2.12)$$

Q → Energía

P → Potencia

$$t = \frac{157098 \text{ J}}{68.16 \text{ W}}$$

$$t = 2304 \text{ segundos} \rightarrow t = \frac{2304}{60}$$

$$t = 38.4 \text{ [minutos]}$$

2.6.8.2 Caso Promedio.

Dividiendo la energía de los casos establecidos por la potencia del sistema:

$$t = \frac{Q}{P}$$

Q → Energía

P → Potencia

$$t = \frac{152709 \text{ J}}{68.16 \text{ W}}$$

$$t = 2240 \text{ segundos} \rightarrow t = \frac{2240}{60}$$

$$t = 37.33 \text{ [minutos]}$$

2.6.8.3 Mejor De Los Casos.

Dividiendo la energía de los casos establecidos por la potencia del sistema:

$$t = \frac{Q}{P}$$

Q → Energía

P → Potencia

$$t = \frac{150075.6 \text{ J}}{68.16 \text{ W}}$$

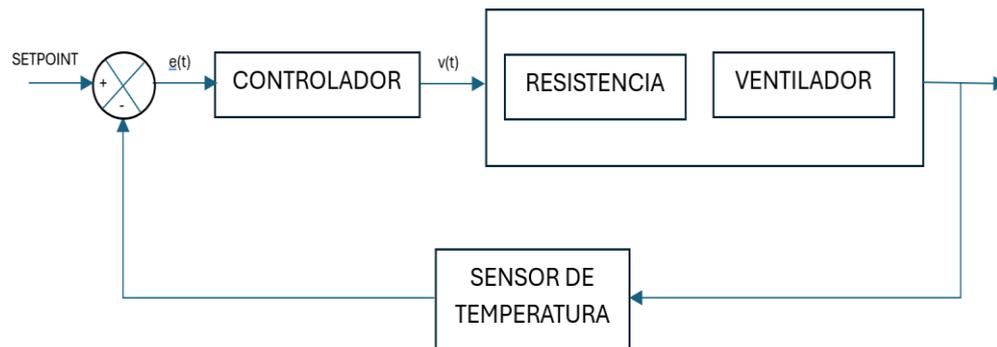
$$t = 2202 \text{ segundos} \rightarrow t = \frac{2202}{60}$$

$$t = 36.7 \text{ [minutos]}$$

2.7 Criterio De Diseño Del Sistema De Control PID

Figura 18

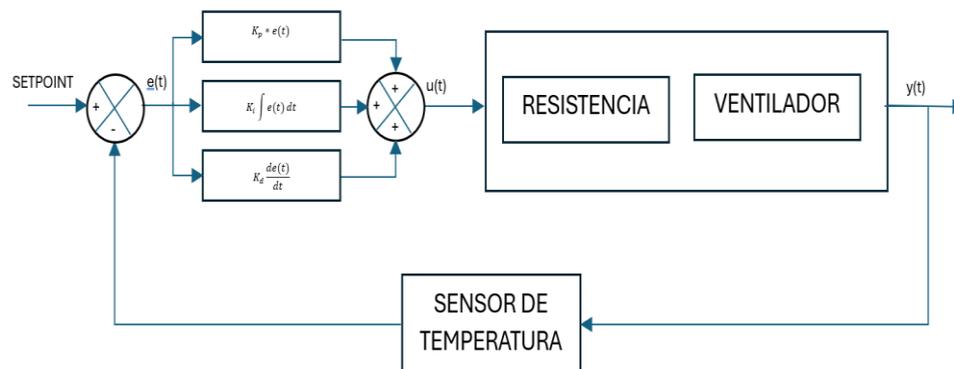
Sistema de Regulación de Temperatura



Nota. Elaboración propia

Figura 19

Sistema PID de lazo Cerrado para Regulación de Temperatura



Nota. Elaboración propia

El control PID implementado para la regulación de la cámara de secado, permitió controlar la temperatura en un rango estable, estableciendo un control preciso de encendido y apagado a los resistores térmicos por medio del relé de estado sólido con una señal PWM

simulando la potencia entregada, asegurando que la temperatura mantenga el valor en la temperatura objetivo.

2.7.1 Determinación De Los Parámetros Térmicos

Los parámetros considerados para el control fueron fundamentales para poder modular el comportamiento de los resistores. Estos parámetros incluyen la capacidad térmica y coeficiente de transferencia de calor.

2.7.1.1 Resistores Térmicos (R).

Los resistores térmicos del sistema dependen directamente de la capacidad térmica de los materiales de la cámara y las dimensiones del sistema, para esto se obtuvo mediante ecuaciones de resistencias en paralelo, para obtener el valor neto del mismo, dando como resultado un valor de 8.44 Ohmios.

2.7.1.2 Capacidad Térmica (C).

La capacidad térmica depende netamente de la masa de aire que habita en el interior de la cámara y el calor específico que hay en el aire. Este dato se obtiene como el producto de la masa del aire y el calor específico:

$$C = m_{total} \times C_p \quad (2.13)$$

$$C = 551.5 \text{ J/K}$$

Donde:

- $m_{total} = m_{tapas} + m_{resistores} + m_{aire}$
- C_p : Capacidad calórica.

2.7.1.3 Constante De Tiempo (τ).

Describe la velocidad que el sistema responde a los cambios de energía que existe en el interior de la cámara.

$$\tau = \frac{C}{P_{total}} \quad (2.14)$$

$$\tau = \frac{551.5}{68.3}$$

$$\tau = 8.07 \text{ segundos}$$

2.7.1.4 Ganancia Estática del sistema (k)

$$k = \frac{P_{total}}{m_{total} * c_p} \quad (2.15)$$

$$k = \frac{68.3}{551.5} = 0.1239^\circ C/W$$

2.7.1.5 Función de Transferencia sin Retardo

$$G(s) = \frac{k}{\tau s + 1} \quad (2.16)$$

$$G(s) = \frac{0.1239}{8.07s + 1}$$

2.7.1.6 Aproximación del Retardo

Usando la aproximación Padé, tiempo muerto de 8s

$$G_{delay}(s) = G(s) * \frac{1 - \frac{\theta}{2}s}{1 + \frac{\theta}{2}s} \quad (2.17)$$

$$G_{delay}(s) = \frac{0.1239}{8.07s + 1} * \frac{1 - 4s}{1 + 4s}$$

2.7.2 Controlador PID

El controlador PID se utiliza para ajustar un sistema de manera controlada, donde los resistores por medio del error entre la temperatura medida por el sensor y la temperatura ajustada, se calcula para mantener una temperatura estable.

2.7.2.1 Ganancia Proporcional.

$$K_p = \frac{1.2 * \tau}{k * \theta} \quad (2.18)$$

$$K_p = \frac{1.2 * 8.07}{0.1239 * 8}$$

$$K_p = 9.77$$

2.7.2.2 Tiempo Integral (Ti)

$$T_i = 2 * \theta \quad (2.19)$$

$$T_i = 2 * 8 = 16 [s]$$

2.7.2.3 *Tiempo Derivativo (Td)*

$$T_d = 0.5 * \theta \quad (2.20)$$

$$T_d = 0.5 * 8 = 4[s]$$

2.7.2.4 *Ganancia Integral (Ki)*

$$K_i = \frac{K_p}{T_i} \quad (2.21)$$

$$K_i = \frac{9.77}{16} = 0.61$$

2.7.2.5 *Ganancia Derivativa (Kd)*

$$K_d = K_p * T_d \quad (2.22)$$

$$K_d = 9.77 * 4 = 39.08$$

2.7.3 **Controlador Ajustado PI**

Con el objetivo de evitar oscilaciones y mejorar la estabilidad del control, se realizaron ajustes en los parámetros del controlador.

2.7.3.1 *Ganancia Proporcional Ajustada*

$$K_p = K_p * 0.1 \quad (2.23)$$

$$K_p = 9.77 * 0.1 = 0.977$$

2.7.3.2 *Tiempo Integral Ajustado*

$$T_i = T_i * 8.7 \quad (2.24)$$

$$T_i = 16 * 8.7 = 139.2[s]$$

2.7.3.3 *Controlador PI sin termino Derivativo*

$$C(s) = K_p \left(1 + \frac{1}{T_i s}\right) \quad (2.25)$$

$$C(s) = 0.977 \left(1 + \frac{1}{139.2s}\right)$$

2.7.4 **Función De Transferencia Del PID en Lazo Cerrado.**

La función de transferencia del PID ajustado es:

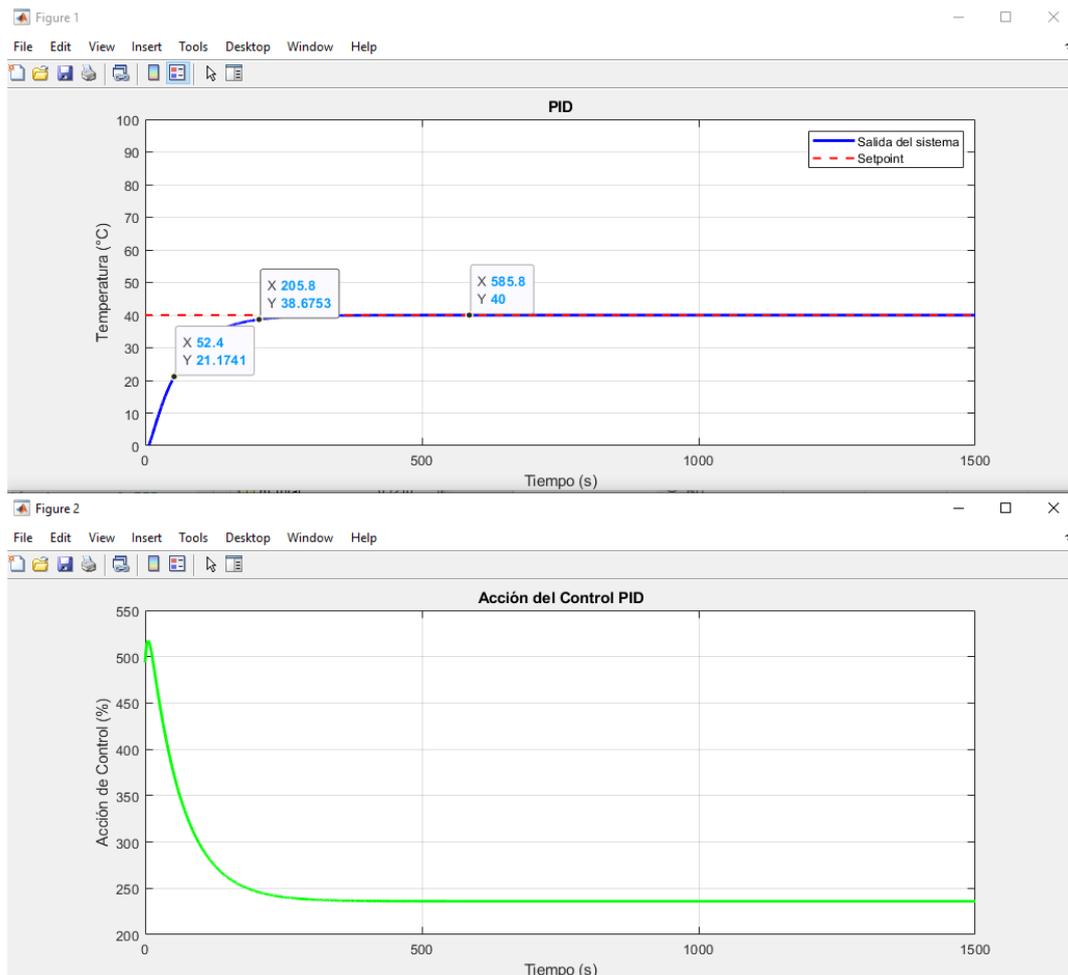
$$T_s(s) = \frac{C(s)*G_{delay}(s)}{1+C(s)*G_{delay}(s)} \quad (2.26)$$

$$T_s(s) = \frac{-291.3 s^2 + 70.13 s + 0.5231}{1.942e^4 s^3 + 4703 s^2 + 105.5 s + 0.5231}$$

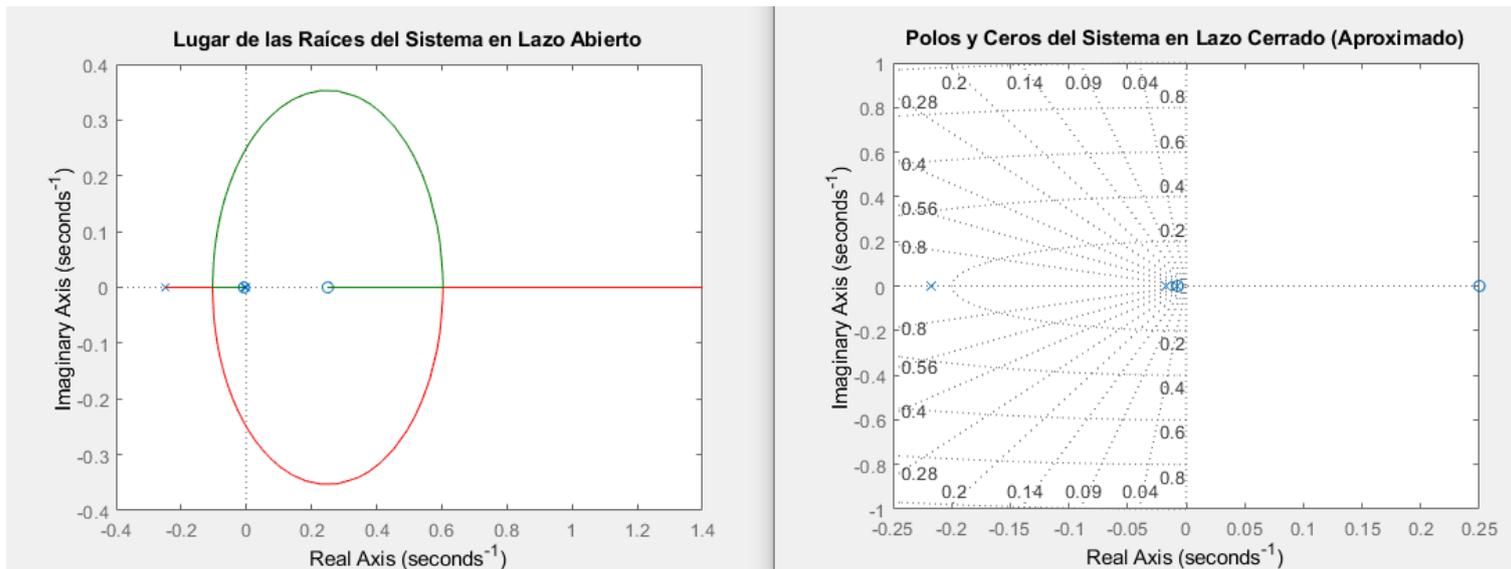
Este controlador ajusta la salida del sistema para mantener la temperatura en el valor ajustado, se realizó la simulación en el software Matlab.

Figura 20

Respuesta Ajustada del Sistema PID



Nota. Elaboración propia

Figura 21*Raíces del Sistema de LA y LC**Nota. Elaboración propia*

En la gráfica presente se puede revisar el comportamiento del sistema, hay dos valores tomados como referencia, (52.4, 21.1741) y (585.8, 40) donde los valores corresponden a (x,y):

X → Eje del tiempo en segundos

Y → Eje de la temperatura en grados Celsius

El primer punto hace referencia a una temperatura ambiente a 20 grados, el segundo punto hace referencia a una temperatura a 40 grados, la cual es el valor ajustado, con un tiempo de 533.4, para obtener el tiempo en minutos se lo divide para 60, dando un tiempo estimado de 9 minutos con 30 segundos.

2.7.5 Controlador PID En La ESP32

El PID realiza tres acciones clave por medio de la ESP32:

$K_p=0.4939$, $K_i=0.0035$, $K_d=19.7544$

- **Proporcional (Kp):** ajusta la salida en función del error actual.
- **Integral (Ki):** ajusta la salida en función del acumulado del error pasado.
- **Derivativo (Kd):** ajusta la salida en función de la tasa de cambio del error

2.7.6 Función De Control PWM

La modulación por ancho de pulso PWM se utiliza para controlar la cantidad de potencia entregada hacia los resistores térmicos por un ciclo de trabajo. Al ajustar el ciclo de trabajo, controlamos la cantidad de tiempo que las resistencias están encendidas a su máxima potencia, se tomó en cuenta los siguientes puntos:

- **Valor máximo de PWM (255):** Significa que las resistencias estarán encendidas todo el tiempo, proporcionando la máxima potencia.
- **Valor mínimo de PWM (0):** Significa que las resistencias estarán apagadas.
- **Valores intermedios (0 a 255):** El ciclo de trabajo del PWM se ajusta, proporcionando potencia proporcionalmente.

2.7.7 Implementación Del Ventilador

El ventilador cumple la función de distribuir el aire caliente por toda el área de trabajo de la cámara de secado, permitiendo tener una temperatura homogénea en cada instante de un proceso. Además, ayuda a enfriar el área cuando se está por encima del valor ajustado, reduciendo el tiempo de enfriamiento, a su vez cuando se termina un proceso, el ventilador queda encendido por un determinado rango de temperatura para expulsar el aire caliente que hay dentro de la cámara de secado a través de rejillas que están implementadas dentro del área.

Capítulo 3

3. Resultados y Análisis

En este capítulo se presentan los resultados obtenidos a lo largo de la implementación y cálculos teóricos del sistema de cámara de secado para endoscopios. Este sistema fue diseñado con el objetivo de abordar las ineficiencias y los problemas asociados con los procesos de secado manual usado por el departamento técnico de la empresa. Se realizaron diferentes análisis enfocados en optimizar los tiempos muertos y brindar un flujo de trabajo mayor para el proceso de reparación. La base de este estudio incluye la eficiencia del secado de humedad y eficiencia del proceso de control para brindar un proceso seguro, además de simulaciones detalladas que sustentan la capacidad del sistema para responder a las demandas del laboratorio técnico.

La implementación del sistema presenta componentes fundamentales como un control PID, que garantiza la precisión de la temperatura, complementando con sensores de alta exactitud que monitorean las condiciones en el interior de la cámara. Estos elementos aseguran un funcionamiento estable y eficiente, minimizando variaciones térmicas y logrando una distribución uniforme del calor.

3.1 Resultados del Estudio

3.1.1 Datos Iniciales y Propiedades Físicas

Las propiedades térmicas de los materiales que intervienen en el sistema son las siguientes:

Tabla 3

Propiedades térmicas de los materiales

Material	Propiedad	Valor
Acero inoxidable	Calor específico (Cp)	468 J/(kg·K)
Agua	Calor específico (Cp)	4180 J/(kg·K)
Aire	Calor específico (Cp)	1000 J/(kg·K)

Aire	Densidad (ρ)	1.2 kg/m ³
Agua	Densidad (ρ)	1000 kg/m ³

Las propiedades térmicas de los materiales presentados en la Tabla 3, ayudaron para la resolución de cálculos y obtener tiempos estimados en que la cámara de secado lograra llegar a la temperatura objetivo general de 40 grados.

3.1.2 Cálculo De Energía Requerida

Se evaluaron tres escenarios de humedad: 100%, 50% y 20%, para comprender cómo varía la energía según la cantidad de humedad presente. Esto proporciona una perspectiva precisa sobre las demandas energéticas del sistema bajo distintas condiciones operativas, permitiendo tener un mayor análisis en el diseño y el consumo del proceso de secado.

Tabla 4

Escenarios de humedad

Escenario	Componente	Masa (kg)	Energía (J)
100% humedad	Agua	0.105	8778.0
	Acero	15.0	140400.0
	Aire	0.396	7920.0
Total		-	157098.0
50% humedad	Agua	0.0525	4389.0
	Acero	15.0	140400.0
	Aire	0.396	7920.0
Total		-	152709.0
20% humedad	Agua	0.021	1755.6
	Acero	15.0	140400.0
	Aire	0.396	7920.0
Total		-	150075.6

Nota. Elaboración propia

3.1.3 *Tiempo De Calentamiento*

El cálculo del tiempo de calentamiento de los componentes del sistema es fundamental para evaluar la eficiencia del diseño en diferentes escenarios de humedad inicial. Tomando casos extremos con 3 endoscopios dentro de la cámara de secado se evaluaron diferentes casos. Para el caso de 100% de humedad, la energía requerida asciende a 157,098 J. Esta cantidad refleja la mayor exigencia del sistema, ya que toda la humedad en el canal debe ser completamente evaporada. Se obtiene un tiempo total de 2304.8 segundos, equivalente a 38.4 minutos, lo que demuestra que el sistema puede operar de manera eficiente incluso en condiciones extremas.

En el escenario intermedio de 50% de humedad, el tiempo de calentamiento disminuye debido a la menor cantidad de agua presente en los endoscopios. En este caso, la energía requerida es de 152,709 J, lo que reduce el tiempo de operación a 2240.4 segundos, o 37.3 minutos. Este resultado representa una condición más común en el uso práctico del sistema, evidenciando su capacidad para manejar escenarios típicos de trabajo con alta eficiencia y estabilidad térmica.

Finalmente, para el escenario más favorable de 20% de humedad, el sistema requiere únicamente 150,075.6 J, lo que resulta en un tiempo de calentamiento de 2201.8 segundos, equivalente a 36.7 minutos. Este caso refleja el óptimo desempeño del sistema en condiciones ideales, donde se aprovechan al máximo las características de los materiales y el control preciso de la temperatura. La consistencia en los tiempos de calentamiento entre los diferentes escenarios evidencia la eficiencia del diseño y la efectividad del control PID para garantizar un secado uniforme y eficiente.

Tabla 5

Escenarios del diseño

Escenario	Energía Total (J)	Tiempo (s)	Tiempo (min)
------------------	--------------------------	-------------------	---------------------

100% humedad	157098.0	2304.8	38.4
50% humedad	152709.0	2240.4	37.3
20% humedad	150075.6	2201.8	36.7

Nota. Elaboración propia

3.2 Conexiones y Pruebas

Figura 2222

Maqueta Funcional



Nota. Elaboración propia

Para verificar el correcto funcionamiento de la maqueta, se llevaron a cabo múltiples fases. Inicialmente, se evaluó el encendido y apagado de cada relé para el control de las resistencias y el ventilador, confirmando su respuesta adecuada ante las señales enviadas desde la HMI. Posteriormente, se probaron los sensores de temperatura y humedad para asegurarse de que las lecturas no fueran erróneas con los valores esperados. Durante estas pruebas, se simuló el comportamiento del sistema en diferentes condiciones operativas, ajustando parámetros como el punto de ajuste de la temperatura y observando la respuesta del control PID implementado. La calibración fue afinada hasta alcanzar una respuesta estable y confiable.

3.3 Resultado de PID

Tras las pruebas pertinentes del sistema de control para mantener una temperatura estable y prueba grafica del PID. Se implementó un control PID utilizando una ESP32 y relé de estado sólido para el encendido y apagado de las resistencias térmicas. En esta configuración, la esp32 fue responsable de recibir las lecturas de los sensores de temperatura PT100 para ejecutar el controlador en tiempo real y mantenerlo en la temperatura fijada ingresada en la pantalla Nextion HMI. Los datos de temperatura se visualizan en la pantalla y generar una gráfica de la temperatura de ajuste vs temperatura medida en tiempo real mostrada en la gráfica 21.

Figura 2323

Control PID durante un Proceso Activo



Nota. Elaboración propia

El algoritmo PID fue configurado con los parámetros $K_p = 0.4939$, $K_i = 0.035$ y $K_d = 19.7544$, la elección de estos parámetros se determinó inicialmente a través de cálculos teóricos y simulaciones realizadas en Matlab. Este análisis permitió obtener valores precisos pero que ofrecían una respuesta no tan eficiente para el caso de equipos médicos. Posteriormente, se realizaron ajustes de manera empírica para acercarse a la estabilidad y precisión del sistema.

- **$K_p = 0.4939$:** La elección de este valor se basó en un proceso para proporcionar una respuesta rápida del sistema ante desviaciones de temperaturas. Durante las pruebas, se verificó que un valor más alto podía generar oscilaciones indeseadas, por lo que se

ajustó para garantizar una respuesta precisa y exacta sin alterar la estabilidad del sistema.

- **Ki = 0.0035:** El parámetro integral se estableció inicialmente para corregir errores acumulativos de manera eficiente. Tras las pruebas, se ajustó a un valor bajo logrando así una corrección progresiva y estable de los errores estacionarios.
- **Kd = 19.7544:** El valor derivativo se calculó para prevenir cambios bruscos en el error, y fue con el objetivo empírico para reducir oscilaciones para alcanzar un equilibrio. Su ajuste final permitió suavizar la transición hacia la temperatura objetivo.

3.4 Validación y Pruebas del Equipo

Detalles sobre las pruebas realizadas y validación del funcionamiento en el departamento técnico de la empresa Perfectech

Tabla 6

Validación de Funcionamiento

Validación de Funcionamiento	
Calidad de Interfaz	La interfaz fue muy didáctica y fácil de manejar, el cambiado de página lo hizo sin ningún problema.
Método Secado y Pegamento	Ambos métodos fueron precisos para solicitar llenar los datos requeridos para poder empezar el proceso.
Conexión Wifi	La conexión wifi fue un éxito permitiendo enviar la información a la base de datos.
Sistema de Control	La posición de los sensores garantizó una temperatura homogénea de la cámara de secado, lo cual garantizó un perfecto secado de todo el endoscopio.
Gráfica	La gráfica de temperatura respondió de manera correcta para visualizar como el valor medido por el sensor alcanza el valor del punto de ajuste.

Nota. Elaboración propia

Se realizaron diferentes pruebas en el laboratorio técnico de la empresa para comprobar la calidad de distribución de temperatura, la prueba se realizó por medio de un

multímetro verificando diferentes casos de temperatura en 35 y 40 grados, como se puede apreciar a continuación:

3.4.1 Caso 1 de Validación

Figura 2424

Verificación con 35 Grados

VERIFICACIÓN DE TEMPERATURA

- Temperatura objetivo: 35C



Nota. Elaboración en el Departamento Técnico (2025)

3.4.2 Caso 2 de Validación

Figura 2525

Verificación con 40 Grados

VERIFICACIÓN DE TEMPERATURA

- Temperatura Objetivo: 40C



Nota. Elaboración en el Departamento Técnico (2025)

3.5 Presupuesto

- **Pantalla Nextion 7" - \$96.90**

La pantalla Nextion NX8048T070 fue seleccionada por su capacidad de ofrecer una interfaz gráfica intuitiva y personalizable, permitiendo ingresar y observar valores, hasta visualizar graficas.

- **Ventilador 3000RPM – 120V - \$17.99**

El ventilador escogido sobre todo por su capacidad de soportar temperaturas normales a altas, hasta 80°C.

- **Modulo MAX31865 PT100 - \$11.88**

Utilizado principalmente para la lectura precisa de las temperaturas a través del sensor PT100.

- **Relé Estado Solido SSR-25DA - \$22.15**

El relé es escogido por su capacidad para manejar altas corrientes y voltajes de manera silenciosa y sin partes móviles.

- **Fuente de Poder 24V - \$22.99**

Suministrado para alimentar las resistencias térmicas y el módulo elevador de Voltaje DC-DC

- **Modulo Elevador de Voltaje DC-DC - \$2.10**

Modulo escogido por el ajuste de voltaje de entrada para proporcionar la alimentación a la ESP32 y Pantalla Nextion.

- **Cableado - \$10.00**

Para la conexión eléctrica del sistema, se utilizaron aproximadamente 10 metros de cable de calibre 16 y 14.

Cable de calibre 16 utilizado para las conexiones de los sensores y componentes electrónicos de bajo consumo. El calibre es adecuado para manejar señales y alimentación de dispositivos como la pantalla Nextion, sensores y módulos.

Cable de calibre 14 utilizado para las conexiones de mayor carga, como el relé SSR-25DA y resistencias térmicas. Escogido por su capacidad de manejar corrientes más altas sin riesgo de sobrecalentamiento.

- **Elaboración de Equipo - \$900.00**

La cámara de secado fue diseñada utilizando materiales de alta calidad que aseguran la durabilidad, resistencia y la funcionalidad. Con materiales como Hierro Negro, Acero Inoxidable, Vidrio Templado, Acrílico.

Incluyendo:

- Pintura especial para protección contra el calor.
- Plancha de Acero Inoxidable de espesor de 1/8".
- Plancha Hierro Negro de 1/8".

- Vidrio Templado 60cm x 30cm.
- Bisagras (2).
- Plancha de Aluminio.
- Seguro de Puerta.
- Manigueta para Puerta.
- Rejillas Inoxidables para escape de aire de calor.
- Soporte de Acero Inoxidable (12).
- Plancha de Acrílico (3) de 10mm de espesor a 48cm x 90cm.
- Trabajo de corte, soldadura y ensamblaje.
- Empaque de Caucho para puerta de 2cm de ancho, 3m de largo.

Tabla 7*Presupuesto Empleado*

PARTE ELECTRONICA		
Producto	Cantidad	Precio
Módulo de alimentación 3,3V 5V	1	\$4,90
ESP32	1	\$17,50
Sensor Humedad -Temperatura	1	\$21,99
Sensor Temperatura PT100	2	\$25,00
Pantalla Nextion 7 inch	1	\$96,90
Ventilador 3000RPM - 120V	1	\$17,99
Modulo MAX31865 PT100	1	\$11,88
Relé Estado Solido SSR-25DA	2	\$25,18
Fuente de poder 24V 10A 110VAC	1	\$22,99
Modulo Elevador de Voltaje DC-DC	1	\$2,10
Cable #14	5[m]	\$4,00
Cable #20	10 [m]	\$7,00
Envio Amazon		\$80,00
Total		\$337,43

PARTE ESTRUCTURAL		
Elaboración Cámara de Secado	Pintura especial para protección contra el calor.	\$900,00
	Plancha de Acero Inoxidable de espesor de 1/8".	
	Plancha Hierro Negro de 1/8".	
	Vidrio Templado 60cm x 30cm.	
	Bisagras (2).	
	Plancha de Aluminio.	
	Seguro de Puerta.	
	Manigueta para Puerta.	
	Rejillas Inoxidables para escape de aire de calor.	
	Soporte de Acero Inoxidable (12).	
	Plancha de Acrílico (3) de 10mm de espesor a 48cm x 90cm.	
	Trabajo de corte, soldadura y ensamblaje.	
	Empaque de Caucho para puerta de 2cm de ancho, 3m de largo.	

Presupuesto empleado para el proyecto		
Mano de Obra		\$400,00
TOTAL DE PRESUPUESTO		\$1.637,43
Adm. e Ingeniería del Proyecto	35%	573,10
COSTO DEL EQUIPO		\$2.210,53

Nota. Elaboración propia.

El análisis de esta investigación revela un avance para el departamento técnico en la reparación de endoscopios. La implementación de un sistema de control basado en PID no solo demostró ser eficiente para mantener la temperatura en un punto estable, sino que también logró una mejora en el flujo de trabajo, dando la facilidad de poder realizar el secado de hasta 3 endoscopios al mismo tiempo.

Ofreciendo beneficios como:

- Optimización del Tiempo, aportando para que todos los escenarios de secado presentan tiempos inferiores a 40 minutos en condiciones extremas, garantizando un flujo de trabajo continuo en el laboratorio.
- Impacto en la Productividad, La capacidad para procesar tres endoscopios simultáneamente incrementa significativamente la productividad del laboratorio. Este diseño no solo optimiza el flujo de trabajo, sino que también

asegura uniformidad en el secado, lo cual es crucial para evitar problemas asociados a residuos de humedad.

- El diseño de la cámara se adapta a diferentes modelos de endoscopios, asegurando su funcionalidad para una variedad de la marca.
- La eficiencia del sistema permite reducir los tiempos muertos global del laboratorio en un aumento del rendimiento general.

En resumen, esta investigación no solo presenta resultados beneficiosos, sino que también presenta un análisis para mejorar las metas del año en el departamento técnico, donde el proceso más eficiente de secado permite retornar los equipos a las instituciones médicas ya reparados reparado por el personal.

Capítulo 4

4. Conclusiones y Recomendaciones

4.1 Conclusiones

Tras la implementación de las fases correspondientes en el desarrollo de este proyecto, se observó que los endoscopios, al ser equipos médicos frágiles, necesitan herramientas especiales para su cuidado y reparación. Por ello, se creó una cámara de secado especializada que garantiza un secado eficaz, minimizando peligros y asegurando que los endoscopios queden en óptimas condiciones para su uso.

El diseño y la implementación de esta cámara de secado para endoscopios gestionada mediante un sistema PID, permitió sustituir el método artesanal que empleaban anteriormente los técnicos, basado en el uso de focos incandescentes por un sistema automatizado. Esta mejora redujo drásticamente los tiempos de secado, pasando de 1 a 3 días a solo 2 a 3 horas, mejorando la eficiencia en el mantenimiento de los endoscopios para la empresa Perfectech.

El sistema PID desarrollado fue implementado con el propósito de regular de manera constante y precisa el funcionamiento del relé de estado sólido, encargado de controlar el encendido y apagado de las resistencias térmicas. Se añadió un ventilador para mejorar la circulación de aire, garantizando así una distribución homogénea del calor. Este diseño garantizó un secado uniforme y minimizó las oscilaciones térmicas, mejorando el control del proceso.

La cámara de secado fue diseñada con dimensiones y capacidades que la hicieran adecuada para los endoscopios de FujiFilm usados en la compañía. Además, por su extensión y ajuste de temperatura flexible, se verificó su aptitud para utilizarse con endoscopios de la marca Olympus y Pentax, junto con otras marcas. Esto plantea la opción de utilizar este sistema en un entorno de alta demanda para diferentes tipos de equipos médicos.

Con el objetivo de optimizar la supervisión y seguimiento de los procedimientos, se implementó una base de datos asociada tanto al método de secado como al de aplicación de pegamento en las puntas distales de los endoscopios. Esta medida se llevó a cabo con el propósito de simplificar el registro y evaluación de datos relevantes, permitiendo un monitoreo más detallado y favoreciendo la constante mejora de dichos procesos. Además, esta herramienta asegura una aplicación más eficiente del pegamento y contribuye al desarrollo de materiales de mayor calidad para futuros procedimientos.

Finalmente, los resultados de validación del equipo se obtuvieron a través de pruebas realizadas con diferentes modelos de endoscopios, incluyendo aquellos utilizados para colonoscopia, endoscopia alta, colangiopancreatografía retrógrada endoscópica (CPRE) y eco endoscopia. Estas pruebas confirmaron lo adaptable y eficaz que es el sistema, evidenciando su cumplimiento con las necesidades operativas y los estándares requeridos para diversos modelos de equipos médicos.

4.2 Recomendaciones

Posterior a las pruebas de validación realizadas en el equipo, se identificaron áreas que pueden ser mejoradas para una gestión de trabajo más eficiente en trabajos futuros, se detallan a continuación.

- Tomar en consideración la implementación de un sistema de aplicación web o móvil que permita llevar un control a larga distancia. Integrando la posibilidad de modificar parámetros de temperatura y tiempo de secado, así como tener la capacidad de encender y apagar el equipo de manera remota.
- Considerar la integración de módulos adicionales, como sistemas de desinfección avanzada por luz ultravioleta (UV), que complementen el proceso de secado y mejore la calidad ayudando a una desinfección reduciendo la carga microbiana en las superficies externas e internas del equipo.
- Se recomienda desarrollar e implementar un sistema avanzado que permita ampliar la funcionalidad de la cámara de secado, de manera que se pueda manejar simultáneamente el secado de diferentes endoscopios. Esto implicaría una interfaz más robusta que permita el ingreso y registro de datos para cada endoscopio en proceso, así como un cronometro individual para cada endoscopio.

Bibliografía

- Alessandro. (2024, 6 marzo). *Come Funziona il Sistema di Controllo Umidità*. FDM - Environment Makers. <https://www.dellamarca.it/es/como-funciona-el-sistema-de-control-de-humedad/>
- Amazon.com: *DIYmalls Nextion 7 inch HMI Display Resistive Touch Screen 5V TFT LCD 800x480 for Arduino ESP32 Development Board (NX8048T070) : Industrial & Scientific. (s. f.)*.
https://www.amazon.com/dp/B0B9GRBWV7?ref=ppx_yo2ov_dt_b_fed_asin_title
- Amazon.com: *HiLeTGO 2PCs SSR-25DA 25A 250V Solid State Relay Module : Industrial & Scientific. (s. f.)*.
https://www.amazon.com/dp/B01N1MMSKI?ref=ppx_yo2ov_dt_b_fed_asin_title
- Amazon.com: *ALITOVE DC 24V 10A 240W Power Supply Adapter Transformer Switch AC 110V / 220V to DC 24V 10amp Universal Regulated Switching Converter for LED Strip Light CCTV Camera Security System : Electronics. (s. f.)*.
https://www.amazon.com/dp/B078RYWZMH?ref=ppx_yo2ov_dt_b_fed_asin_title
- Amazon.com: *T-PRO RTD PT100 Temperature Sensors Three-wire System,Stainless Steel Probe(4×30MM) Range:-50°C-200°C (6.6Feet) : Industrial & Scientific. (s. f.)*.
https://www.amazon.com/dp/B072LSRB22?ref=ppx_yo2ov_dt_b_fed_asin_title
- Amazon.com: *Temperature Humidity Sensor, I2C Interface FS304 SHT30 FS304 SHT31 FS304 SHT35 Waterproof Digital Thermal Soil Probe Module Stainless Steel Room Thermometer Hygrometer with Extension Cable(SHT31) : Industrial & Scientific. (s. f.)*.
https://www.amazon.com/dp/B09Z72CXX2?ref=ppx_yo2ov_dt_b_fed_asin_title

Amazon.com: *T-PRO RTD PT100 Temperature Sensors Three-wire System,Stainless Steel Probe(4×30MM) Range:-50°C-200°C (3.3Feet) : Industrial & Scientific. (s. f).*

https://www.amazon.com/dp/B0711WRM4X?ref=ppx_yo2ov_dt_b_fed_asin_title

Amazon.com: *T-PRO RTD PT100 Temperature Sensors Three-wire System,Stainless Steel Probe(4×30MM) Range:-50°C-200°C (3.3Feet) : Industrial & Scientific. (s. f.-b).*

https://www.amazon.com/dp/B0711WRM4X?ref=ppx_yo2ov_dt_b_fed_asin_title

Amazon.com: *GUIANDJING 120mm x 38mm Cooling Fan 110V 120V Dual Ball Bearing High CFM 3000RPM with AC Plug and Aluminium Alloy Frame : Electronics. (s. f.).*

https://www.amazon.com/dp/B0CXHJHSRB?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1

Arduino. (2018). *PID temperature control with Arduino.*

<https://blog.arduino.cc/2018/04/16/pid-temperature-control-with-arduino/>.

Bravo, A. (2001). La alucinante evolución de la endoscopia. *Revista Portuguesa De Pneumologia*, 66, 58–59. <https://www.revistagastroenterologiamexico.org/es-pdf-X0375090601252936>

Cerezo Loor, J. J. (2024). *Análisis del reprocesamiento de endoscopios en un centro gastroenterológico de Guayaquil: Desarrollo de una guía estandarizada para mejorar la seguridad del paciente* (Bachelor's thesis).

Forumeiros. (2017). *As Partes Comuns de um Endoscópio*. <https://cpma-endoscopios.directorioforuns.com/t8-as-partes-comuns-de-um-endoscopia>

Fuji Electric. (2023). *Relé de estado sólido*. <https://www.fujielectric.fr/es/tecnologias/rele-estatico/>

Marketing Simmedica-PENTAX. (2021). *Guía para prevenir daños en endoscopios*. <https://blog.simmedica.com/blog/guia-prevenir-da%C3%B1os-en-endoscopios>

Mañas, A. (2019.). *Desarrollo e implementación de un invernadero inteligente controlado con Arduino* [Tesis de licenciatura, Universidad de Jaén].

<https://crea.ujaen.es/bitstream/10953.1/14003/1/MEMORIA%20TFG%20ANTONIO%20JESUS%20MANAS.pdf>.

Mendoza, A. (2024). Ecoendoscopía: pasado, presente y futuro de un viaje al espacio interior. *Rev Gastroenterol Peru*, 44(1), 5-7.

<https://revistagastroperu.com/index.php/rgp/article/view/1675>

Ministerio de Salud Pública de Ecuador. (2020). Guía de reprocesamiento de dispositivos médicos reutilizables. Quito: Ministerio de Salud Pública de Ecuador.

Organización Mundial de la Salud (OMS) (2016). Guidelines on Core Components of Infection Prevention and Control Programmes at the National and Acute Health Care Facility Level. <https://www.who.int/publications/i/item/9789241549929>

Organización Mundial de la Salud (OMS) (2020). Manual de seguridad en el reprocesamiento de dispositivos médicos. Ginebra: OMS.

Sensirion. (s.f.). *Datasheet SHT3x-DIS: Humidity and Temperature Sensor*.

<https://www.alldatasheet.es/datasheet-pdf/view/897975/ETC2/SHT31.html>.

Fuji electric. (2023). *Sensor de temperatura PT100*.

<https://www.fujielectric.fr/es/tecnologias/sensor-de-temperatura-pt100/>

Sosa, Ó., & Campos, R. (2019). Relevancia del reprocesamiento de los endoscopios en el control de infecciones. *Endoscopia*, 31(4), 163–167.

<https://doi.org/10.24875/end.19000023>

Zayas, F., Quintián, H., Jove, E., Casteleiro, J., & Calvo, J. (2020). Diseño de controladores PID. [Tesis de maestría, Universidade da Coruña].

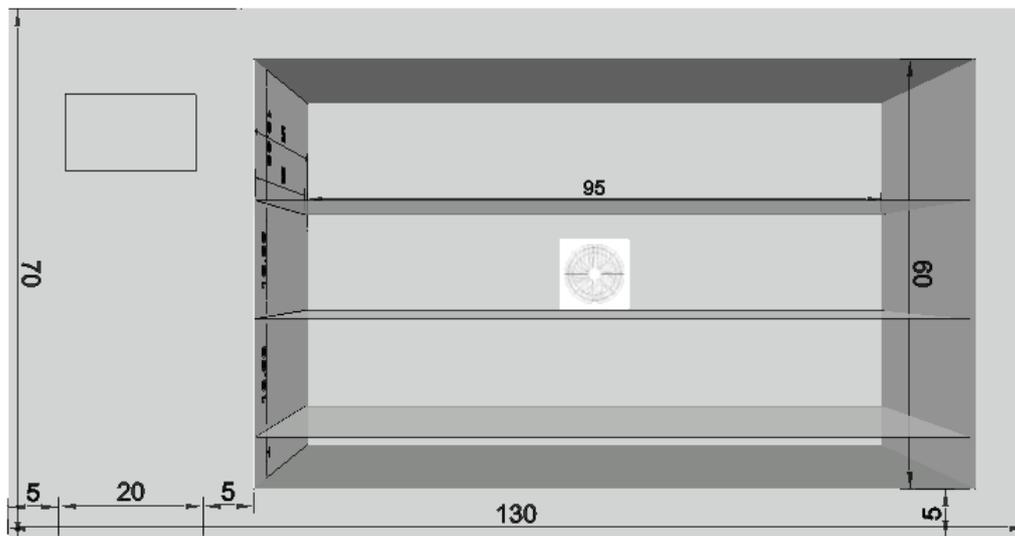
https://ruc.udc.es/dspace/bitstream/handle/2183/25824/Zayas_Gato_2020_Dise%c3%bl0_de_controladores_PID.pdf?sequence=2&isAllowed=y

Apéndices

4.3 Diseño De Cámara De Secado En AutoCad

Plano 1

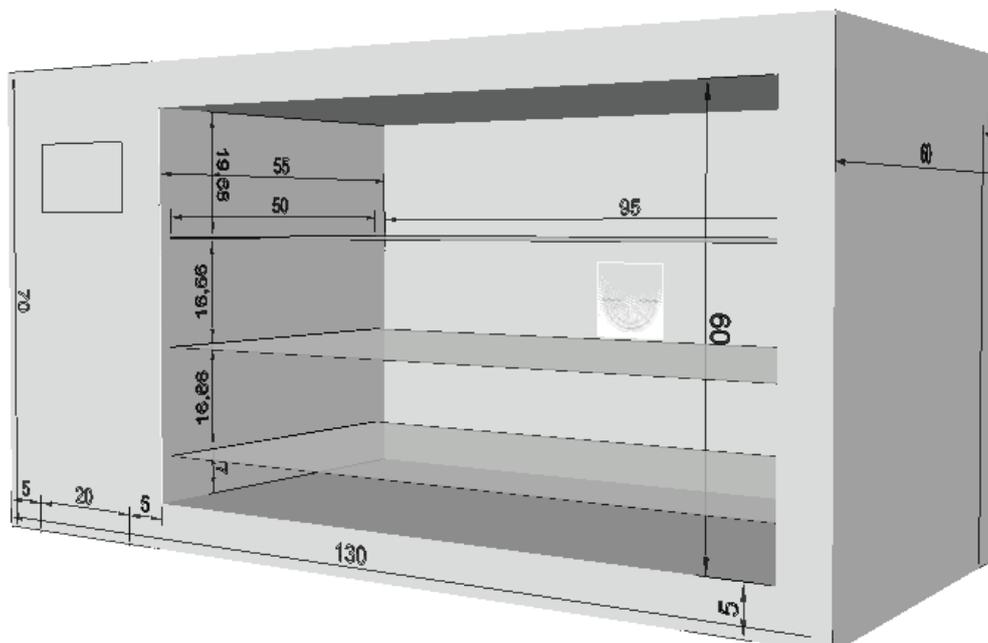
Vista frontal



Nota. Elaboración propia.

Plano 2

Vista Lateral

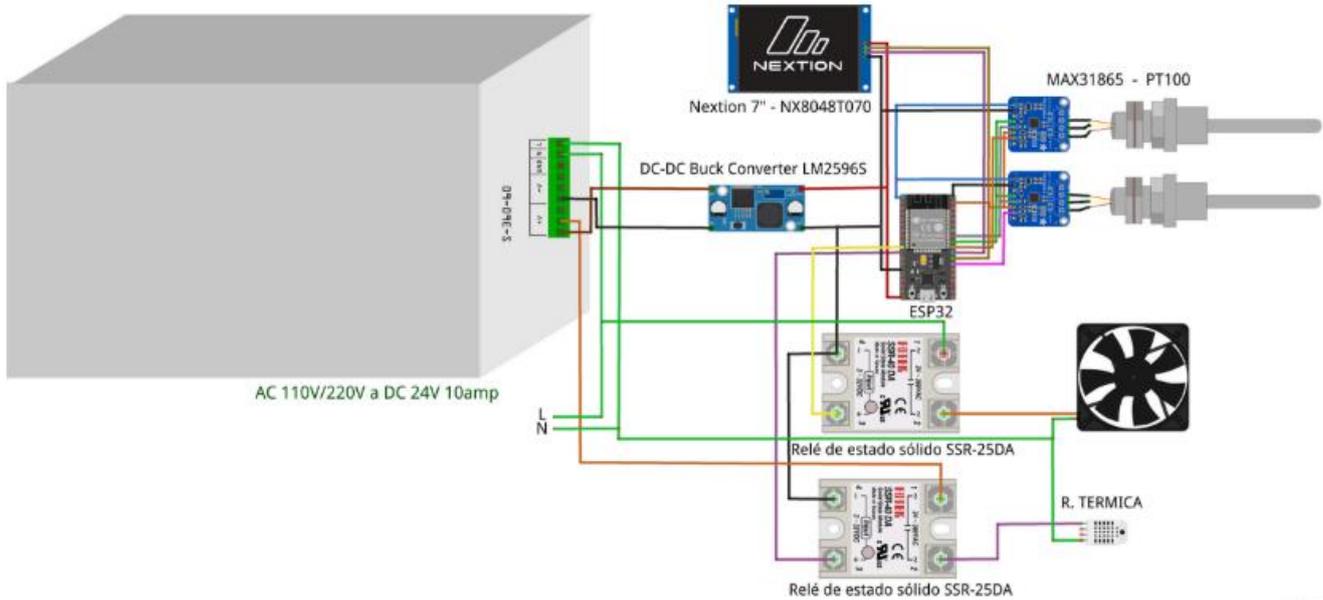


Nota. Elaboración propia

Plano 3

Plano esquemático

DIAGRAMA DE CONEXIONES



Nota. Elaboración propia

4.4 Proceso De Elaboración De Maqueta

Figura 2626

Vista lateral con tapa



Nota. Elaboración propia

Figura 2727*Vista lateral sin tapa**Nota. Elaboración propia.***Figura 28***Vista frontal**Nota. Elaboración propia.*

4.5 Pantalla Nextion 7"

Figura 2928

Especificaciones Técnicas Pantalla Nextion 7"

Nextion Models

Nextion Type	Basic Series
Nextion Models	NX3048T070

Specifications

	Data	Description
Color	64K 65536 colors	16 bit 565, 5R-6G-5B
Layout size	181(L)×108(W)×9(H)	
Visual Area (V.A.)	164.9mm(L)×100mm(W)	
Active Area (A.A.)	154.08mm(L)×85.92mm(W)	
Resolution	800×480 pixel	Also can be set as 480×800
Touch type	Resistive	
Touches	> 1 million	
Backlight	LED	
Backlight lifetime (Average)	>30,000 Hours	
Brightness	230 nit	0% to 100%, the interval of adjustment is 1%
Weight	268g	

Nota. Tomado de Amazon (2024)

4.6 Rele de Estado Solido SSR-25DA

Figura 3029

Especificaciones Técnicas SSR-25DA

Product information

Brand	T-PRO
Item dimensions L x W x H	9.41 x 6.3 x 0.2 inches
Material	PFA, Stainless steel 316
Style	Modern
Measurement Accuracy	±1.5%
Measuring Range	-50°C - 200°C
Mounting Type	Flange Mount
Output Type	Wire
Specific Uses For Product	temperature measurement
Upper Temperature Rating	2E+2 Degrees Celsius
UPC	759974532719
Manufacturer	T-PRO
Product Dimensions	9.41 x 6.3 x 0.2 inches
Item Weight	0.634 ounces
ASIN	B0711WRM4X
Country of Origin	China
Item model number	TD0030
Customer Reviews	4.5       102 ratings 4.5 out of 5 stars
Best Sellers Rank	#47,131 in Industrial & Scientific (See Top 100 in Industrial & Scientific) #51 in Temperature Probes & Sensors
Is Discontinued By Manufacturer	No
Date First Available	May 21, 2017

Nota. Tomado de Amazon (2024)30

4.7 Sensor PT100

Figura 3131

Especificaciones Técnicas PT100

Technical Details

Contact Type	Normally Open
Current Rating	60 Amps
Mounting Type	Screw Mount
Brand	HiLetgo
Operation Mode	Automatic
Coil Voltage	380 Volts
Contact Current Rating	25 Amps
Maximum Switching Current	25 Amps
Maximum Switching Voltage	250 Volts
Number of Terminals	2
Manufacturer	HiLetgo
Part Number	3-01-1247-25A
Item Weight	6.7 ounces
Package Dimensions	4.21 x 3.31 x 1.3 inches
Item model number	3-01-1247-25A
Is Discontinued By Manufacturer	No
Size	SSR-25DA
Material	FR4
Measurement System	Metric, Imperial
Batteries Included?	No
Batteries Required?	No

Nota. Tomado de Amazon (2024)

4.8 Código ESP32

```

#include <WiFi.h>
#include <HTTPClient.h>
#include <Nextion.h>
#include <Adafruit_MAX31865.h>
#include <PID_v1.h>
#include <Ticker.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <Wire.h>
#include <Adafruit_SHT31.h>
#include <Preferences.h>
Preferences preferences; // Crear
instancia de Preferences
// Configuración WiFi
//const char* ssid = "Netlife-Bastidas";
//const char* password = "1202775837";
// Variables para la conexión WiFi
char ssid[50] = ""; // Almacena el
SSID ingresado por el usuario
char password[50] = ""; // Almacena
la contraseña ingresada por el usuario
bool conectadoWiFi = false; // Estado
de la conexión WiFi
// Link de Google Apps Script para la
Base de Datos
const char* serverName =
"https://script.google.com/macros/s/AKfy
cbyiZuINKg-
bVgT25Ky7KIL9C1u5xaCFflotMI4xIL-
kD5N5U6RJb8kDeab1qUOTihjUzg/exec
";

// Configuración para dos sensores
MAX31865m
Adafruit_MAX31865 thermo1 =
Adafruit_MAX31865(5, 23, 19, 18); //
Sensor 1: CS=5, SDI=23, SDO=19,
CLK=18
Adafruit_MAX31865 thermo2 =
Adafruit_MAX31865(4, 23, 19, 18); //
Sensor 2: CS=4, SDI=23,
SDO=19,CLK=18
// Configuración del cliente NTP
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP,
"pool.ntp.org", -5 * 3600, 60000); //
Ecuador está en UTC-5
// Sensor de Humedad
Adafruit_SHT31 sht31 =
Adafruit_SHT31(); // Instancia para el
sensor SHT31
#define RREF 430.0
#define RNOMINAL 100.0
// Pines y variables del sistema
#define PIN_RELAY_RESISTENCIAS
27
#define PIN_RELAY_VENTILADOR 26
// Pin del ventilador
// Pines RX y TX para Serial2
#define RX2 16
#define TX2 17
// Pin Buzzer

```

```

#define BUZZER_PIN 25 // Cambia el
número de pin según tu configuración
// Variables PID
bool procesoActivo = false;
double Temperatura; // Entrada al PID
(temperatura actual)
double Output; // Salida del PID
(control del relé)
double Setpoint = 0.0; // Temperatura
objetivo ingresada por el usuario
// Valores PID ajustados
double Kp = 11.31726;
double Ki = 0.15; // Calculado como el
inverso del tiempo de integración
double Kd = 3.5; // Tiempo derivativo
PID myPID(&Temperatura, &Output,
&Setpoint, Kp, Ki, Kd, DIRECT);
// Configuración para Time Proportional
Control (TPC)
const unsigned long Period = 5000; //
Período en milisegundos (5 segundos)
unsigned long lastTime = 0;
// Variables para el temporizador
int timerSegundos = 0; // Segundos
totales restantes
Ticker temporizadorTicker; // Ticker para
el temporizador
// Configuración de la pantalla Nextion
// Página de Standby
NexText standbyHoraFecha =
NexText(6, 1, "F_H_STANDBY"); //

```

```

Cuadro de texto para la fecha y hora en
Standby
// Declaración de botones de la página 0
NexButton secadoButton =
NexButton(0, 1, "SECADO");
NexButton pegamentoButton =
NexButton(0, 2, "PEGAMENTO");
NexButton redesButton = NexButton(0,
6, "REDES");
NexButton apagarButton = NexButton(0,
4, "APAGAR");
NexText horaFecha = NexText(0, 5,
"HORA_FECHA");
// Declaración de elementos en la
página 1
NexButton iniciarPage1 = NexButton(1,
2, "INICIO");
NexButton detenerPage1 =
NexButton(1, 3, "STOP");
NexText variacionTempPage1 =
NexText(1, 19, "VARIACION_TEMP");
NexText ingresoTempPage1 =
NexText(1, 33, "INGRESO_TEMP");
NexText variacionHumPage1 =
NexText(1, 31, "VARIACION_HUM");
NexWaveform waveformTempPage4 =
NexWaveform(3, 3,
"WAVEFORM_TEMP");
NexWaveform waveformHumPage3 =
NexWaveform(3, 4,
"WAVEFORM_HUM");

```

```

NexNumber horaPage1 =
NexNumber(1, 26, "HORA");
NexNumber minutosPage1 =
NexNumber(1, 27, "MINUTOS");
NexNumber segundosPage1 =
NexNumber(1, 34, "SEGUNDOS");
NexButton enviar_datos = NexButton(1,
18, "ENVIAR_DATOS");
// Declaración de elementos en la
página 2
NexText inRed = NexText(2, 5,
"IN_RED"); // Cuadro para
ingresar la nueva red
NexText inPassword = NexText(2, 6,
"IN_PASSWORD"); // Cuadro para
ingresar la contraseña de la nueva red
NexText wifiActual = NexText(2, 7,
"WIFI_ACTUAL"); // Cuadro para
mostrar la red WiFi actual
NexText passActual = NexText(2, 10,
"PASS_ACTUAL");// Cuadro para
mostrar la contraseña actual
NexButton confirmarButton =
NexButton(2, 10, "CONFIRMAR"); //
Botón para confirmar nueva conexión
// Cuadros de texto globales para base
de datos
NexText nombre_insti_txt(1, 15,
"NOMBRE_INSTI");
NexText ingresar_pu_txt(1, 20,
"INGRESAR_P_U");
NexText ingresar_ta_txt(1, 21,
"INGRESAR_T_A");
NexText ingresar_h_txt(1, 23,
"INGRESAR_H"); // Cuadro de texto
para la hora
NexText ingresar_m_txt(1, 25,
"INGRESAR_M"); // Cuadro de texto
para los minutos
NexText ingresar_model_txt(1, 16,
"INGRESAR_MODEL");
NexText ingresar_serie_txt(1, 17,
"INGRESAR_SERIE");
NexText nombre_tecn_txt(1, 29,
"NOMBRE_TECN");
NexText textoPagina7(7, 5, "TEXTO");
NexTouch *nex_listen_list[] = {
&redesButton, &secadoButton,
&pegamentoButton, &apagarButton,
&iniciarPage1, &detenerPage1,
&enviar_datos, &confirmarButton, NULL
};
// Definir variables globales
bool modoPegamento = false;
bool modoSecado = false;
bool modoManual = false; // Variable
global para modo manual
bool modoAutomatico = true; // Variable
global para modo automático
bool datosEnviados = false; // Variable
para rastrear si se presionó el botón
ENVIAR DATOS

```

```

// PARA EL LOOP
unsigned long lastLoopTime = 0; //
Variable para almacenar el tiempo de la
última ejecución
const unsigned long loopInterval = 100;
// Intervalo deseado en milisegundos
(100 ms)
// Función para leer texto de un objeto
Nextion
String leerTexto(NexText &obj) {
    char buffer[100] = {0};
    obj.getText(buffer, sizeof(buffer));
    return String(buffer);
}
// Función para validar los campos
obligatorios
bool validarCampos() {
    String mensajeError = "ERROR
COMPLETAR PARAMETROS: ";
    // Validar campos comunes
    if
(leerTexto(nombre_insti_txt).isEmpty()) {
        mensajeError += "NOMBRE
INSTITUCION - ";
    }
    if
(leerTexto(nombre_tecn_txt).isEmpty())
{
        mensajeError += "NOMBRE
TECNICO - ";
    }
}

```

```

if
(leerTexto(ingresoTempPage1).isEmpty
()) { // Usar el cuadro de texto nuevo
para ingresar el setpoint
    mensajeError += "SETPOINT - ";
}
if
(leerTexto(ingresar_model_txt).isEmpty(
)) {
    mensajeError += "MODELO - ";
}
if
(leerTexto(ingresar_serie_txt).isEmpty())
{
    mensajeError += "SERIE - ";
}
// Validar campos adicionales para
PEGAMENTO
if (modoPegamento) {
    if
(leerTexto(ingresar_pu_txt).isEmpty()) {
        mensajeError += "PEGAMENTO
USADO - ";
    }
    if
(leerTexto(ingresar_ta_txt).isEmpty()) {
        mensajeError += "TEMP DEL
PEGAMENTO - ";
    }
    if
(leerTexto(ingresar_h_txt).isEmpty()) {

```

```

        mensajeError += "HORA DEL
PEGAMENTO - ";
    }
    if
(LeerTexto(ingresar_m_txt).isEmpty()) {
        mensajeError += "MINUTOS
DEL PEGAMENTO - ";
    }
}
// Elimina el último " - " si hay errores
if (mensajeError.endsWith(" - ")) {
    mensajeError =
mensajeError.substring(0,
mensajeError.length() - 3);
}
// Si hay errores, cambiar a la página
7 y mostrar el mensaje
if (mensajeError != "ERROR
COMPLETAR PARAMETROS: ") {
    sendCommand("page
MENSAJES"); // Cambiar a la
página de mensajes
textoPagina7.setText(mensajeError.c_st
r()); // Mostrar el mensaje en el cuadro
de texto
    sendCommand("ref TEXTO");
// Refrescar el objeto de texto en la
pantalla
    return false; //
Indicar que faltan datos
}

```

```

    return true; // Indicar que la validación
fue exitosa
}
void sonidoBoton() {
    tone(BUZZER_PIN, 1000, 100); //
Genera un tono de 1000 Hz durante 100
ms
}
// Callback para el botón SECADO
void botonSecadoCallback(void *ptr) {
    sonidoBoton(); // Reproducir sonido
modoSecado = true;
modoPegamento = false;
}
void botonRedesCallback(void *ptr) {
    sonidoBoton(); // Reproducir sonido
}
// Callback para el botón PEGAMENTO
void botonPegamentoCallback(void *ptr)
{
    sonidoBoton(); // Reproducir sonido
modoPegamento = true;
modoSecado = false;
}
// Callback para el botón APAGAR
void botonApagarCallback(void *ptr) {
    sonidoBoton(); // Reproducir sonido
procesoActivo = false;
digitalWrite(PIN_RELAY_RESISTENCI
AS, LOW); // Apagar el relé

```

```

digitalWrite(PIN_RELAY_VENTILADOR
, LOW);
}
// Callback para el botón INICIO
void botonIniciarPopCallback(void *ptr) {
    sonidoBoton(); // Reproducir sonido
    Serial.println("Botón INICIO
presionado");
    // Verificar si estamos en modo
PEGAMENTO
    if (modoPegamento) {
        // Validar si los datos han sido
enviados
        if (!datosEnviados) {
            sendCommand("page
MENSAJES"); // Cambiar a la página de
mensajes
textoPagina7.setText("ERROR:
PRESIONAR ENVIAR DATOS."); //
Mostrar mensaje de error
            return; // Salir si no se han
enviado los datos
        }
        // Validar campos específicos de
PEGAMENTO
        if (!validarCampos()) {
            return; // Salir si faltan datos
        }
    }
    // Verificar si estamos en modo
SECADO

```

```

    if (modoSecado) {
        // Validar los campos necesarios
para SECADO
        if (!validarCampos()) {
            Serial.println("Campos
obligatorios no completos para
SECADO");
            return; // Salir si faltan datos
        }
    }
    // Leer la temperatura objetivo
(Setpoint)
    char buffer[10] = {0};
    ingresoTempPage1.getText(buffer,
sizeof(buffer));
    Setpoint = atof(buffer);
    // Verificar si la temperatura es válida
    if (Setpoint <= 0) {
        Serial.println("Temperatura inválida
o no ingresada");
        sendCommand("page
MENSAJES"); // Cambiar a la página de
mensajes
        textoPagina7.setText("Temperatura
inválida o no ingresada."); // Mostrar
mensaje de error
        return;
    }
    // Leer las horas, minutos y segundos
del temporizador

```

```

uint32_t tempValue = 0;
horaPage1.getValue(&tempValue);
int timerHoras = tempValue;
minutosPage1.getValue(&tempValue);
int timerMinutos = tempValue;
segundosPage1.getValue(&tempValue);
int timerSegundosInput = tempValue;
// Calcular los segundos totales del
temporizador
timerSegundos = (timerHoras * 3600)
+ (timerMinutos * 60) +
timerSegundosInput;
// Configurar los modos (manual o
automático) según el temporizador
if (timerSegundos <= 0) {
modoManual = true;
modoAutomatico = false;
} else {
modoManual = false;
modoAutomatico = true;
temporizadorTicker.attach(1,
decrementarTemporizador); // Iniciar el
temporizador
}
// Activar el proceso
procesoActivo = true;
myPID.SetMode(AUTOMATIC);
}
void controlBuzzer() {
// Verificar el tiempo restante y activar
el buzzer según corresponda

```

```

if (timerSegundos == 60) {
// A 1 minuto de finalizar: dos
pitidos
tone(BUZZER_PIN, 1000, 200); //
Pitido de 200 ms
delay(300); // Pausa
entre pitidos
tone(BUZZER_PIN, 1000, 200); //
Segundo pitido
} else if (timerSegundos <= 10 &&
timerSegundos > 0) {
// Durante los últimos 10 segundos:
un pitido por segundo
tone(BUZZER_PIN, 1000, 500); //
Pitido de 500 ms
} else if (timerSegundos == 0) {
// Cuando el temporizador llega a
0: pitido largo de 5 segundos
tone(BUZZER_PIN, 1000, 3000); //
Pitido largo
}
}
void reiniciarTemporizador() {
// Reiniciar las variables de hora,
minutos y segundos
horaPage1.setValue(0);
minutosPage1.setValue(0);
segundosPage1.setValue(0);
// Reiniciar el temporizador en 0
timerSegundos = 0;
}

```

```

// Función para actualizar el
temporizador
void decrementarTemporizador() {
    static unsigned long lastUpdate = 0; //
    Último tiempo de actualización
    static int ultimaHoraMostrada = -1; //
    Última hora mostrada
    static int ultimoMinutoMostrado = -1;
    // Último minuto mostrado
    static int ultimoSegundoMostrado = -
    1; // Último segundo mostrado
    // Verificar si es tiempo de actualizar
    (cada 1000 ms)
    if (millis() - lastUpdate >= 1000 &&
    procesoActivo && modoAutomatico) {
        lastUpdate = millis(); // Actualizar el
    tiempo de la última ejecución
        if (timerSegundos > 0) {
            timerSegundos--;
            controlBuzzer(); // Llama a la
    función del buzzer
        } else {
            // Cuando el temporizador llega a
    0, reinicia las variables
            reiniciarTemporizador();
        }
        // Convertir segundos restantes a
    horas, minutos y segundos
        int horasRestantes =
    timerSegundos / 3600;

```

```

        int minutosRestantes =
    (timerSegundos % 3600) / 60;
        int segundosRestantes =
    timerSegundos % 60;
        // Actualizar el cuadro de horas
    solo si cambia la hora
        if (horasRestantes !=
    ultimaHoraMostrada) {
            horaPage1.setValue(horasRestantes);
            ultimaHoraMostrada =
    horasRestantes;
        }
        // Actualizar el cuadro de minutos
    solo si cambia el minuto
        if (minutosRestantes !=
    ultimoMinutoMostrado) {
            minutosPage1.setValue(minutosRestant
    es);
            ultimoMinutoMostrado =
    minutosRestantes;
        }
        // Actualizar el cuadro de segundos
    solo si cambia el segundo
        if (segundosRestantes !=
    ultimoSegundoMostrado) {
            segundosPage1.setValue(segundosRes
    tantes);
            ultimoSegundoMostrado =
    segundosRestantes;
        }

```

```

    }
}
// Callback para el botón STOP
void botonDetenerPopCallback(void
*ptr) {
    sonidoBoton(); // Reproducir sonido
    // Detener el proceso
    procesoActivo = false; // Marcar el
proceso como inactivo
    timerSegundos = 0; // Reiniciar el
temporizador
    // Detener el temporizador y reiniciar
variables
    temporizadorTicker.detach();
    reiniciarTemporizador();
    // Apagar los dispositivos conectados
digitalWrite(PIN_RELAY_RESISTENCI
AS, LOW);
digitalWrite(PIN_RELAY_VENTILADOR
, LOW);
    // Reiniciar el PID
    myPID.SetMode(MANUAL); //
Detener el PID
    Output = 0; // Reiniciar la
salida del PID
    myPID.SetMode(AUTOMATIC); //
Configurar el PID para su próxima
ejecución
    // Limpiar cuadros de texto en la
pantalla Nextion

```

```

    ingresoTempPage1.setText("");
    variacionTempPage1.setText("");
    variacionHumPage1.setText("");
    horaPage1.setValue(0);
    minutosPage1.setValue(0);
    segundosPage1.setValue(0);
    nombre_insti_txt.setText("");
    ingresar_pu_txt.setText("");
    ingresar_ta_txt.setText("");
    ingresar_h_txt.setText("");
    ingresar_m_txt.setText("");
    ingresar_model_txt.setText("");
    ingresar_serie_txt.setText("");
    nombre_tecn_txt.setText("");
    /* // Borrar la gráfica de temperaturas
for (int i = 0; i < 100; i++) {
    waveformTempPage4.addValue(0, 0);
    waveformTempPage4.addValue(1, 0);
} */
    // Borrar los datos del waveform
usando el comando 'cle'
    sendCommand("WAVEFORM_TEMP.cl
e");
    sendCommand("ref
WAVEFORM_TEMP"); // Refrescar el
objeto gráfico para aplicar el cambio
}
// Callback para el botón
ENVIAR_DATOS
void botonEnviarDatosPopCallback(void
*ptr) {

```

```

sonidoBoton(); // Reproducir sonido
// Verificar los campos obligatorios
para PEGAMENTO
if (modoPegamento) {
    if (validarCampos()) {
        String nombreInsti =
leerTexto(nombre_insti_txt);
        String pegUsado =
leerTexto(ingresar_pu_txt);
        String tempAplicada =
leerTexto(ingresar_ta_txt);
        String hora =
leerTexto(ingresar_h_txt);
        String minutos =
leerTexto(ingresar_m_txt);
        String modeloEquipo =
leerTexto(ingresar_model_txt);
        String serieEquipo =
leerTexto(ingresar_serie_txt);
        String nombreTecn =
leerTexto(nombre_tecn_txt);
        // Concatenar hora y minutos en
formato HORA:MINUTOS
        String horaAplicada = hora + ":"
+ minutos;
        // Enviar los datos a la base de
datos
        enviarDatos(nombreInsti,
pegUsado, tempAplicada, horaAplicada,
modeloEquipo, serieEquipo,
nombreTecn);

```

```

// Marcar como enviado
datosEnviados = true;
Serial.println("Datos enviados
correctamente");
    } else {
        Serial.println("Faltan campos
obligatorios para PEGAMENTO");
    }
}
// Función para enviar los datos a la
base de datos
void enviarDatos(String nombreInsti,
String pegUsado, String tempAplicada,
String horaAplicada, String
modeloEquipo, String serieEquipo,
String nombreTecn) {
    if (WiFi.status() ==
WL_CONNECTED) {
        HTTPClient http;
        http.begin(serverName);
        http.addHeader("Content-Type",
"application/json");
        // Crear el JSON con los datos a
enviar
        String jsonData =
"{\"nombreInsti\": \"\" + nombreInsti +
        "\", \"pegUsado\": \"\" +
pegUsado +
        "\", \"tempAplicada\": \"\"
+ tempAplicada +

```

```

        "\", \"horaAplicada\":\":" +
horaAplicada +
        "\", \"modeloEquipo\":\":"
+ modeloEquipo +
        "\", \"serieEquipo\":\":" +
serieEquipo +
        "\", \"nombreTecn\":\":" +
nombreTecn + "\}";
        http.POST(jsonData);
        http.end();
    }
}
// Tarea para manejar la conexión WiFi
void conectarWiFi() {
    if (WiFi.status() != WL_CONNECTED)
    {
        WiFi.begin(ssid, password);
        unsigned long startAttemptTime =
millis();
        while (WiFi.status() !=
WL_CONNECTED && millis() -
startAttemptTime < 10000) {
            delay(500);
        }
        if (WiFi.status() ==
WL_CONNECTED) {
            conectadoWiFi = true;
            wifiActual.setText(ssid); //
Actualizar en pantalla
        } else {
            conectadoWiFi = false;

```

```

        wifiActual.setText("SIN
CONEXION");
    }
}
// Tarea para actualizar la fecha y hora
void actualizarHoraFecha() {
    if (WiFi.status() ==
WL_CONNECTED) {
        timeClient.update();
        unsigned long epochTime =
timeClient.getEpochTime();
        if (epochTime > 0) {
            struct tm *ptm =
gmtime((time_t *)&epochTime);
            int day = ptm->tm_mday;
            int month = ptm->tm_mon + 1;
            int year = ptm->tm_year +
1900;
            String formattedTime =
timeClient.getFormattedTime();
            String fechaHora = String(day)
+ "/" + String(month) + "/" + String(year)
+ " " + formattedTime;
            // Actualizar en la pantalla
            horaFecha.setText(fechaHora.c_str());
            //standbyHoraFecha.setText(fechaHora.
c_str());
        }
    }
}

```

```

void confirmarButtonCallback(void *ptr) {
    sonidoBoton(); // Reproducir sonido
    Serial.println("Botón CONFIRMAR
presionado.");
    // Leer las credenciales ingresadas
    String ssidIngresado =
leerTexto(inRed);
    String passwordIngresado =
leerTexto(inPassword);
    // Validar que ambos campos no estén
vacíos
    if (ssidIngresado.isEmpty() ||
passwordIngresado.isEmpty()) {
        wifiActual.setText("Error");
        passActual.setText("Vacíos");
        return;
    }
    // Mostrar mensaje de conexión en
pantalla
wifiActual.setText("CONECTANDO...");
    // Guardar las credenciales en las
variables globales
    ssidIngresado.toCharArray(ssid,
sizeof(ssid));
passwordIngresado.toCharArray(passw
ord, sizeof(password));
    // Intentar conectar a la red WiFi
conectarWiFi();
    // Actualizar el estado en la pantalla
dependiendo del resultado
    if (conectadoWiFi) {

```

```

        wifiActual.setText(ssid); // Mostrar
el SSID conectado
    } else {
        wifiActual.setText("SIN
CONEXION");
    }
}
void setup() {
    // Configurar pines I2C (SDA: GPIO
21, SCL: GPIO 22)
    Wire.begin(21, 22);
    // Inicializar el sensor
    if (!sht31.begin(0x44)) { // Dirección
predeterminada 0x44
        Serial.println("Error: No se pudo
inicializar el sensor SHT31.");
    }
    /*
    // Inicializar la memoria y recuperar
las credenciales
    preferences.begin("WiFiCreds", true);
    // Abrir en modo lectura
    String savedSSID =
preferences.getString("ssid", ""); //
Recuperar el SSID guardado
    String savedPassword =
preferences.getString("password", ""); //
Recuperar la contraseña guardada
    preferences.end(); // Cerrar la
memoria

```

```

    if (!savedSSID.isEmpty() &&
!savedPassword.isEmpty()) {
        Serial.println("Credenciales
recuperadas:");
        Serial.println("SSID: " +
savedSSID);
        Serial.println("Password: " +
savedPassword);
        // Convertir las credenciales a char
array
savedSSID.toCharArray(ssid,
sizeof(ssid));
savedPassword.toCharArray(password,
sizeof(password));
// Intentar conectar automáticamente a
la red guardada
        conectarWiFi();
    } else {
        Serial.println("No se encontraron
credenciales guardadas.");
        wifiActual.setText("SIN
CONEXION"); // Actualizar en pantalla
Nextion
    }
*/
// Configuración de pines de salida
pinMode(PIN_RELAY_RESISTENCIAS,
OUTPUT); // Relé para controlar las
resistencias
pinMode(PIN_RELAY_VENTILADOR,
OUTPUT); // Relé para controlar el

```

```

ventilador
digitalWrite(PIN_RELAY_RESISTENCI
AS, LOW); // Inicialmente apagado
digitalWrite(PIN_RELAY_VENTILADOR
, LOW); // Inicialmente apagado
        pinMode(BUZZER_PIN, OUTPUT);
// Pin para el buzzer
        digitalWrite(BUZZER_PIN, LOW);
// Asegurar que el buzzer esté apagado
        // Configurar pines RX y TX para
Serial2
        Serial2.begin(9600, SERIAL_8N1,
RX2, TX2);
        // Inicializar pantalla Nextion
        nexInit();
// Conectar a WiFi
        conectarWiFi();
// Inicializar cliente NTP
        timeClient.begin();
        // Configurar botones de la pantalla
redesButton.attachPop(botonRedesCall
back, &redesButton); // Botón
REDES
secadoButton.attachPop(botonSecadoC
allback, &secadoButton); // Botón
SECADO
pegamentoButton.attachPop(botonPega
mentoCallback, &pegamentoButton); //
Botón PEGAMENTO
apagarButton.attachPop(botonApagarC
allback, &apagarButton); // Botón

```

```

APAGAR
iniciarPage1.attachPop(botonIniciarPop
Callback, &iniciarPage1); // Botón
INICIO
detenerPage1.attachPop(botonDetener
PopCallback, &detenerPage1); //
Botón STOP
enviar_datos.attachPop(botonEnviarDat
osPopCallback, &enviar_datos); // Botón
ENVIAR_DATOS
confirmarButton.attachPop(confirmarBut
tonCallback, &confirmarButton);
    // Inicializar sensores MAX31865
thermo1.begin(MAX31865_2WIRE); //
Configuración del sensor 1
thermo2.begin(MAX31865_2WIRE); //
Configuración del sensor 2
    // Inicializar PID
myPID.SetMode(AUTOMATIC);
myPID.SetOutputLimits(0, 255);
myPID.SetSampleTime(100); //
Tiempo de muestreo en milisegundos
    // Inicializar cuadros de texto en la
pantalla
    horaPage1.setValue(0);    // Cuadro
HORA en Página 1
    minutosPage1.setValue(0); //
Cuadro MINUTOS en Página 1
    segundosPage1.setValue(0);
//variacionTempPage1.setText(""); //
Cuadro para temperatura objetivo

```

```

    // Iniciar sin conexión
WiFi.disconnect();
    conectadoWiFi = false;
}
void loop() {
    nexLoop(nex_listen_list);
    // Obtener el tiempo actual
    unsigned long currentMillis = millis();
    unsigned long now = millis(); // Obtén
el tiempo actual en milisegundos
    // Leer la temperatura y la humedad
del SHT31
    float temperaturaSHT31 = NAN;
    float humedadSHT31 = NAN;
    float offsetTemp = -10.0; // Ajuste
para corregir la desviación del SHT31
    if (sht31.begin()) { // Verificar si el
sensor está conectado
        temperaturaSHT31 =
sht31.readTemperature() + offsetTemp;
        humedadSHT31 =
sht31.readHumidity();
    }
    // Leer el comando enviado por
Nextion
    if (Serial2.available()) { // Cambiado a
Serial2 para la comunicación con
Nextion
        String comando =
Serial2.readStringUntil('\n'); // Leer
hasta un salto de línea

```

```

    if (comando == "SECADO") {
    botonSecadoCallback(nullptr); // Llama
al callback de SECADO
        } else if (comando ==
"PEGAMENTO") {
    botonPegamentoCallback(nullptr); //
Llama al callback de PEGAMENTO
        } else if (comando == "APAGAR") {
    botonApagarCallback(nullptr); // Usar la
función Standby en lugar de reiniciar
        }
    }
    if (procesoActivo) {
        // Leer las temperaturas de ambos
sensores con el ajuste de -10
        //double temp1 =
thermo1.temperature(RNOMINAL,
RREF) - 10.3;
        //double temp2 =
thermo2.temperature(RNOMINAL,
RREF);
        //double temp1 =
thermo1.temperature(RNOMINAL,
RREF);
        // Calcular el promedio de las
temperaturas
        //double promedioTemp = (temp1 +
temp2) / 2.0;
        // Combinar con la temperatura del
SHT31 si está disponible

```

```

        //double promedioTotal =
promedioTemp; // Inicializar con el
promedio actual
        //if (!isnan(temperaturaSHT31)) {
        // promedioTotal =
(promedioTemp + temperaturaSHT31) /
2.0;
        //}
        double promedioTotal =
temperaturaSHT31;
        static unsigned long
lastSwitchTime = 0;
        static bool resistenciasEncendidas
= false;
        // Si la temperatura está dentro del
rango de control On/Off
        if (temperaturaSHT31 >= Setpoint
&& temperaturaSHT31 <= (Setpoint +
0.7)) {
            if (now - lastSwitchTime >=
4000) { // Cada 4 segundos cambia el
estado
                resistenciasEncendidas =
!resistenciasEncendidas; // Alternar
estado
                digitalWrite(PIN_RELAY_RESISTENCI
AS, resistenciasEncendidas ? HIGH :
LOW); // Encender o apagar
                lastSwitchTime = now; //
Actualizar tiempo del último cambio
                Serial.println(resistenciasEncendidas ?

```

```

"Resistencias ENCENDIDAS" :
"Resistencias APAGADAS");
    }
    } else if (temperaturaSHT31 <
Setpoint) {
    // Control PID cuando la
temperatura está por debajo del setpoint
    Temperatura =
temperaturaSHT31;
    myPID.Compute();
    if ((now - lastTime) >= Period) {
        lastTime = now;
    }
    if ((now - lastTime) < (Output /
100.0) * Period) {
digitalWrite(PIN_RELAY_RESISTENCI
AS, HIGH); // Calentamiento
proporcional
    } else {
digitalWrite(PIN_RELAY_RESISTENCI
AS, LOW); // Apagar resistencias
    }
    } else if (temperaturaSHT31 >
(Setpoint + 0.7)) {
    // Si la temperatura excede el
rango, apagar las resistencias
digitalWrite(PIN_RELAY_RESISTENCI
AS, LOW);
    Serial.println("Temperatura
excede el rango, resistencias
APAGADAS.");
    }
    // Mantener el ventilador siempre
encendido
digitalWrite(PIN_RELAY_VENTILADOR
, HIGH);
    // Manejo del buzzer para notificar
cuando la temperatura alcance el rango
deseado
    static bool buzzerNotificado = false;
    if (!buzzerNotificado &&
temperaturaSHT31 >= Setpoint &&
temperaturaSHT31 < (Setpoint + 1.0)) {
        tone(BUZZER_PIN, 1000, 500);
    // Emitir un pitido
        buzzerNotificado = true; //
Marcar como notificado
    } else if (temperaturaSHT31 <
Setpoint || temperaturaSHT31 >=
(Setpoint + 1.0)) {
        buzzerNotificado = false; //
Reiniciar la notificación si sale del rango
    }
    // Mostrar la humedad en el
monitor serie y pantalla Nextion
    if (!isnan(humedadSHT31)) {
    // Actualizar la humedad en la pantalla
Nextion
    variacionHumPage1.setText(String((int)
humedadSHT31).c_str());
    }

```

```

        // Mostrar temperatura en la
pantalla
variacionTempPage1.setText(String((int
)temperaturaSHT31).c_str());
/* //Graficar temperaturas
    int tempEscalada =
(int)(Temperatura * 10); // Escalar la
temperatura con decimales
    int setpointEscalado =
(int)(Setpoint * 10);
waveformTempPage4.addValue(0,
tempEscalada % 256); // Temperatura
escalada
waveformTempPage4.addValue(1,
setpointEscalado % 256); // Setpoint
escalado
*/
    // Enviar temperatura y setpoint al
waveform sin escalar
    waveformTempPage4.addValue(0,
(uint8_t)temperaturaSHT31); // Canal 0
para la temperatura
waveformTempPage4.addValue(1,
(uint8_t)Setpoint); // Canal 1 para el
setpoint
}

        else {
digitalWrite(PIN_RELAY_RESISTENCI
AS, LOW); // Apaga el relé si el proceso
no está activo
digitalWrite(PIN_RELAY_VENTILADOR
, LOW);
        }
        // Si ya estamos conectados,
verificar el estado periódicamente
        if (conectadoWiFi && WiFi.status()
!= WL_CONNECTED) {
            Serial.println("Conexión WiFi
perdida.");
            conectadoWiFi = false;
            wifiActual.setText("Sin conexión");
        }
        // Actualizar la fecha y hora cada
segundo
        static unsigned long
lastTimeUpdate = 0;
        if (millis() - lastTimeUpdate > 1000)
        {
            lastTimeUpdate = millis();
            actualizarHoraFecha();
        }
        delay (300);
    }
}

```

Código 1. ESP32 - Arduino IDE