**Diseño de Software realizado sobre Raspberry PI para la Cámara de Niebla Salina de la Universidad Santo Tomas seccional Tunja.**

**Correspondencia:** Se puede enviar un correo formal a la dirección [ing.felipenarvaez017@gmail.com](mailto:ing.felipenarvaez017@gmail.com)

**Abstract:**

La Cámara de Niebla Salina es una herramienta esencial en la investigación industrial y científica para estudiar materiales, superficies y recubrimientos. Funciona exponiendo continuamente las piezas a una niebla ácida corrosiva para simular su comportamiento en ambientes de trabajo o con el paso del tiempo. En este contexto, se desarrolla una Cámara de Niebla Salina en la Universidad Santo Tomás, seccional Tunja, utilizando un compartimiento de vidrio templado y un sistema de control basado en la Raspberry PI 400, sensores sumergibles y actuadores de potencia. Este equipo está equipado con un software de Interfaz Gráfica de Usuario (GUI) que permite al operador iniciar procesos guiados o autónomos. Los procesos guiados activan selectivamente sensores y actuadores durante períodos indefinidos, mientras que los procesos autónomos operan estos componentes dentro de un tiempo especificado, no excediendo las 24 horas. Este enfoque proporciona una plataforma para la investigación en condiciones de corrosión controladas, facilitando el estudio de materiales basado en ciclos de exposición específicos.

**Keywords:** Máquina, Placa de Circuito Impreso PCB, Microcontrolador MCU, Microprocesador MPU, software, Hardware, Diseño, Desarrollo, Monitoreo, Control, Repositorio, Raspberry PI.

**Sobre este Articulo:**

El presente artículo introduce la Máquina de Niebla Salina desarrollada en el Laboratorio de Materiales de la Universidad Santo Tomas, seccional Tunja. El desarrollo de este prototipo se divide en diferentes fases en las que se cuentan: elaboración de planos estructurales, la implementación de la red eléctrica y electrónica, la instrumentación de sensores e incorporación de actuadores electromecánicos, diseño de piezas en 3D, creación de PCB electrónicas y la configuración del sistema de Software, haciendo uso de la Raspberry PI 400 como plataforma central de monitoreo y control. Este artículo proporciona una visión general del proceso de desarrollo, sin embargo, para obtener información más detallada sobre la construcción de este prototipo, se invita al lector a consultar el libro de tesis titulado "Sistema de Monitoreo y Control para el Funcionamiento del Prototipo de la Máquina de Niebla Salina para los Laboratorios de la Facultad de Ingeniería Mecánica".

**Introducción:**

La Máquina de Niebla Salina, también conocida como Cámara de Niebla Salina o Cámara de Niebla Ácida en ciertos ámbitos de la industria y estudio de materiales, constituye una herramienta fundamental para evaluar la resistencia de diversos materiales, recubrimientos y superficies frente a la corrosión en entornos específicos. Este dispositivo opera mediante la creación controlada de una atmósfera corrosiva, reproduciendo las condiciones ambientales a las que los materiales estarán expuestos en su aplicación final.

En este trabajo, se describe y desarrolla una Máquina de Niebla Salina construida en el Laboratorio de Materiales de la Universidad Santo Tomas, seccional Tunja. Se detalla el proceso de diseño estructural, la implementación de la infraestructura eléctrica y electrónica, la instrumentación de sensores y la configuración del sistema informático, destacando el uso de la Raspberry PI 400 como plataforma central para el sistema de monitoreo y control. Esta máquina se concibió con el propósito de facilitar estudios de corrosión y envejecimiento acelerado de materiales, especialmente metales, para la investigación de recubrimientos y superficies metálicas comúnmente utilizadas en la industria. Además, se plantea su utilidad como herramienta educativa en el ámbito académico, proporcionando a estudiantes de ingeniería la oportunidad de realizar prácticas de laboratorio e investigaciones.

La Cámara de Niebla Salina desarrollada para la Universidad Santo Tomás, seccional Tunja, está equipada con un software con Interfaz Gráfica de Usuario (GUI) que permite al operador interactuar para iniciar tanto procesos guiados como autónomos. Aparte a las interfaces de manejo, control y monitoreo con el que el operador interactuar, existen también capas de software en segundo plano y otros dedicados al microcontrolador auxiliar MCU con el que trabaja conjuntamente la Raspberry PI para la lectura de datos del entorno. El proceso guiado implica la activación selectiva o conjunta de los sensores y actuadores del prototipo durante un período indeterminado, mientras que el proceso autónomo implica la operación conjunta de actuadores y sensores dentro de un tiempo especificado, no excediendo las 24 horas en un solo día. Este enfoque requiere un estudio de la corrosión de materiales basado en ciclos de exposición específicos, proporcionando así una plataforma para la investigación en condiciones de corrosión controladas.

**Materiales y Métodos:** El estudio se basa en el libro de tesis que tienen como título: "Sistema de Monitoreo y Control para el Funcionamiento del Prototipo de la Máquina de Niebla Salina para los Laboratorios de la Facultad de Ingeniería Mecánica", elaborada por el Ingeniero Electrónico Luis Felipe Narváez Gómez, disponible en el repositorio CRAI de la Universidad Santo Tomas. Los materiales y métodos utilizados se encuentran documentados en el repositorio del proyecto en Github, bajo el título: "RuisoArt/Project\_MaterialOxidationMachine", propiedad del Ing. Felipe Narváez. Estos recursos proporcionaron la base para el diseño e implementación del sistema de monitoreo y control de la Máquina de Niebla Salina, abarcando la selección de componentes electrónicos, software de desarrollo, así como la metodología de prueba y validación del sistema.

**Resultados**

El software empleado para el control de la Máquina de Niebla Salina, fundamental para la ejecución de investigaciones sobre la degradación de materiales por parte de los operadores que la manejan, presenta una arquitectura multicapa que abarca desde el microcontrolador auxiliar encargado de la adquisición de datos provenientes de los sensores implementados, hasta la interfaz visual que facilita la interacción del operador. Esta interfaz ofrece un tablero de selección de opciones de control y monitoreo, donde en la parte de control tenemos un enfoque individualizado para cada conmutador de forma supervisada y no supervisada por tiempos definidos y no definidos respectivamente, mientras que en el monitoreo tenemos una visualización automática de registros y la generación de gráficos estadísticos correspondientes, optimizando así la comprensión y análisis de los datos obtenidos.

**Software de Microcontrolador Arduino, Lectura de Sensores y enviado a Raspberry PI por comunicación Serial.**

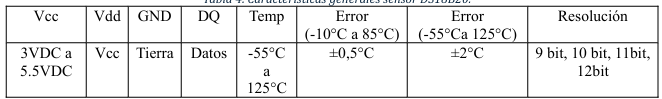
Uno de los primeros softwares que podemos encontrar en el uso de la cámara de niebla salina es aquel

instalado en el MCU Atmega32 del Arduino NANO. Este es el encargado de leer constantemente los

diferentes sensores conectados a él.

Se emplearon dos sensores digitales DS18B20, configurados como termocuplas sumergibles, colocados estratégicamente en la cámara de degradación de la máquina. Dichos sensores operan mediante la lectura de un bus de datos serial, lo que implica que sus lecturas digitales se transmiten dentro de una trama de datos compuesta, permitiendo la conexión de varios sensores (hasta 8) en un único canal de lectura digital, específicamente el pin D2 del Arduino Nano. Para facilitar la lectura de estos sensores, se recurre a dos librerías especializadas: One Wire y DallasTemperature. La primera se encarga del protocolo One Wire, utilizado para la manipulación de datos entrantes, mientras que la segunda, desarrollada por el fabricante del sensor, simplifica la interpretación de la trama de datos, permitiendo almacenar únicamente los valores de temperatura en un arreglo de datos para una extracción eficiente. Algunas características que se tuvieron en cuenta dentro de la programación de lectura de estos sensores son las siguientes:

Características generales de operación del sensor.



Características de rango de error según la resolución de trabajo del sensor.



Características de implementación del sensor y conexión según distancia de cables de conexión en chasis.



En el presente proyecto, la distancia entre la cámara de niebla salina y el MCU Arduino Nano se mantiene por debajo de los 5 metros, lo que permite la utilización de una resistencia de 4,7 kΩ entre los pines Vdata y Vcc. El sensor de temperatura DS18B20, dotado de tres pines de conexión (GND, Vdata y Vcc), ofrece dos modalidades de conexión con el Arduino: la alimentación normal, que emplea una conexión directa entre los pines correspondientes, y la conexión parasitaria a través del pin Vdata. En este trabajo, se opta por la conexión convencional, en contraposición a la conexión parasitaria.

Otro aspecto que demuestra afectar el valor de la temperatura a registrar es precisamente el lugar de trabajo el cual puede incurrir en presencia de ruido eléctrico por alteraciones de medio físico y la temperatura a la que se encuentra trabajando.

Imagen que contiene Tabla

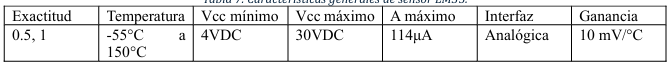
Descripción generada automáticamente

También se implementó el sensor analógico LM35 para la medición de temperatura en el tanque de agua. La peculiaridad de este sensor radica en su capacidad para proporcionar lecturas de temperatura directamente correlacionadas con el voltaje resultante de la variación de su resistencia interna frente a los cambios de temperatura, independientemente de su resolución. Conectado a un pin analógico de la placa Arduino Nano, el sensor utiliza la conversión ADC (Analógico-Digital) con una resolución de 1024 bits para obtener lecturas precisas. La fórmula utilizada para convertir el valor de entrada analógica (n) en grados centígrados permite obtener la temperatura deseada. Los tres pines de conexión del LM35 (Vcc, GND y Vdata) facilitan su integración con la placa Arduino Nano. Dado que su encapsulado no está diseñado para entornos húmedos, se recurre a la protección mediante termoencogible y cableado adicional para su conexión al tanque de agua y al MCU, garantizando así su funcionalidad en condiciones ambientales adversas.

Tabla

Descripción generada automáticamente

Otras características a tener en cuenta en el manejo e este sensor son las siguientes:



El error de medición de este sensor puede darse según el comportamiento de encapsulado a la temperatura de funcionamiento como se muestra en la siguiente imagen.

Gráfico, Gráfico de líneas

Descripción generada automáticamente

En este estudio también se incorpora un interruptor ON/OFF como último sensor conectado a la placa Arduino Nano, diseñado con un encapsulado hueco en una de sus extremidades, funcionando como un balón de aire de nivel. Cuando se sitúa en posición horizontal sobre o entre un líquido, este dispositivo se puede utilizar para detectar el nivel óptimo de agua en el tanque. Su principio de funcionamiento se basa en hacer flotar la parte hueca del encapsulado en presencia de agua, lo que activa una sección magnética pasiva que cierra el circuito interno, permitiendo así el flujo de corriente eléctrica. En contraste, en ausencia de agua, la porción hueca deja de flotar, haciendo caer el imán y abriendo el circuito eléctrico, interrumpiendo el flujo de corriente. El sensor proporciona al Arduino Nano una lectura digital de 1 o 0, indicando la presencia o ausencia de tensión eléctrica, respectivamente, que se interpreta como la presencia o ausencia de agua según la configuración del switch. Para este propósito, se pueden implementar dos configuraciones: PULL-UP y PULL-DOWN, sin necesidad de una PCB electrónica externa, utilizando el soporte universal de la placa Arduino. En este proyecto, se emplea la configuración PULL-UP, donde el cierre del switch provoca una disminución a 0VDC en la entrada analógica, indicando la presencia de agua en el tanque. La resistencia utilizada es de 4,7 kΩ.

Diagrama

Descripción generada automáticamente

**Software de Lectura, desglose y creación de registro de Trama de datos recibida desde MCU ATmega32 con valores de los sensores implementados en la Máquina de Niebla Salina.**

El software empleado en sección del proyecto se encarga de gestionar la lectura de datos provenientes del Arduino Nano mediante comunicación serial por parte de la Raspberry Pi 400. Para ello, es necesario especificar previamente el número del puerto USB al que está conectado el Arduino, considerando que la Raspberry dispone de múltiples puertos USB y no todos admiten comunicación serial. Se utiliza un comando en Raspdebian para identificar los puertos en uso, que generalmente se presenta como ttyACM0 o ttyUSB0. El software se configura para operar a la misma velocidad de transmisión que el Arduino Nano (9600 Baudios), lo que permite una sincronización adecuada en la recepción y lectura de datos. El programa permanece en espera continua a través de un bucle WHILE, interrumpido únicamente por eventos externos o la desconexión del MCU, lo que causaría un error de lectura del puerto USB. Los datos recibidos a través de la comunicación serial se interpretan como tramas independientes, con un bit de inicio y uno de finalización. Estos datos se separan en líneas individuales y se identifican mediante el uso del carácter de punto y coma como separador. Posteriormente, se almacenan de manera independiente en un archivo de texto plano, cuya escritura se realiza con una demora mínima para garantizar un muestreo detallado del puerto. Este enfoque asegura una recopilación precisa de datos para su posterior análisis y procesamiento.



**Software de Parada de emergencia de la Máquina de Niebla Salina.**

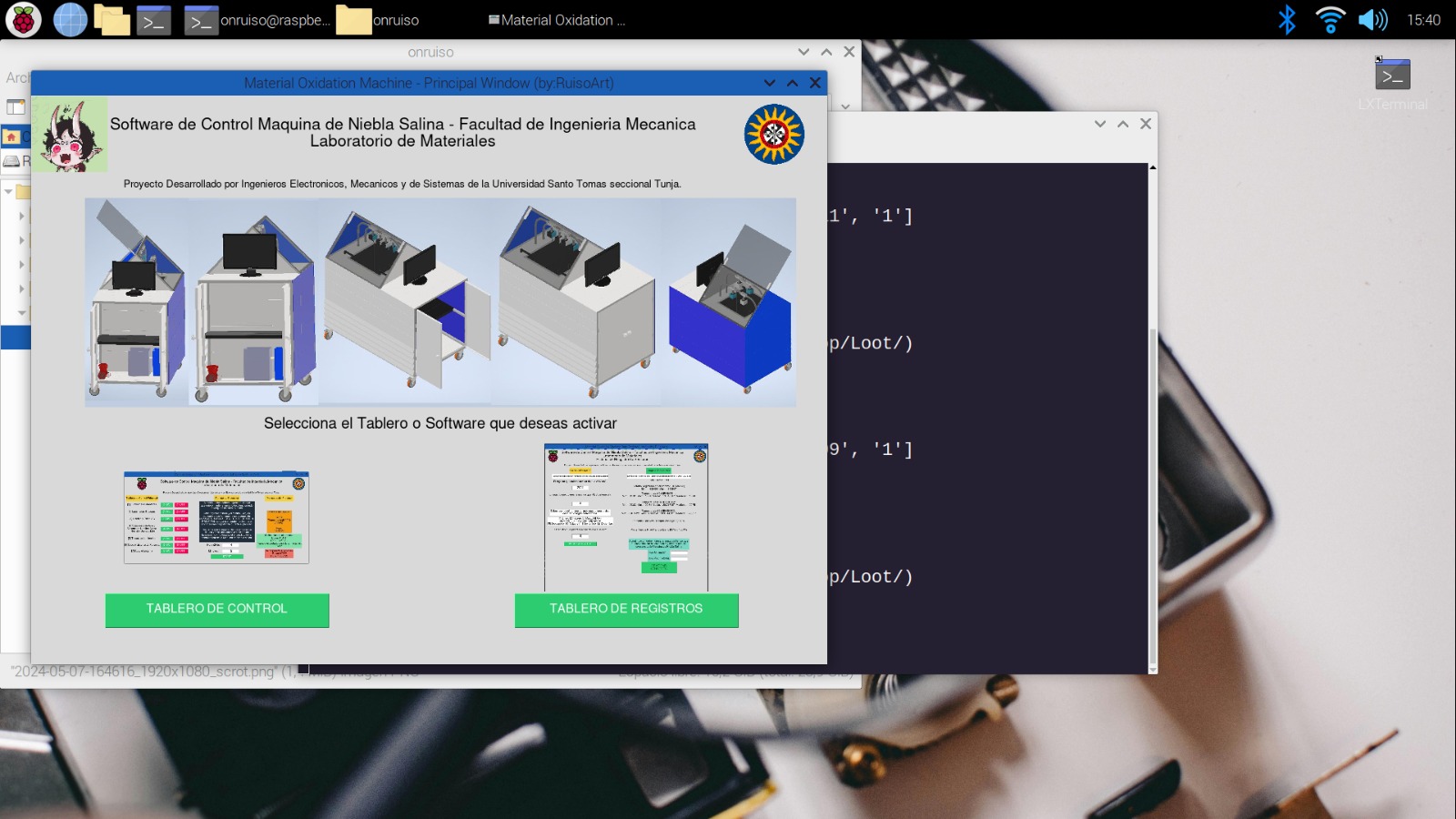
El software diseñado para el paro de emergencia de la Máquina de Niebla Salina opera de manera análoga al anteriormente descrito, pero con un enfoque específico en la protección física del prototipo y sus actuadores, particularmente el reverbero. Dada la naturaleza crítica de la resistencia sumergible, cuya función es hervir agua que puede contener cloruro de cobre o cloruro de sodio, su operación sin líquido presente podría ocasionar sobrecalentamiento, daño al filamento y eventual rotura, con consecuencias potencialmente catastróficas, como el derretimiento del case del actuador o del tanque de agua, lo que podría causar inundaciones o incendios. El software de parada de emergencia monitorea constantemente los datos de los sensores, especialmente el sensor número cuatro (S4), que indica el nivel de líquido en el tanque. Si se detecta un nivel insuficiente de líquido, se ordena el apagado total de los actuadores de la máquina, independientemente de si se encuentra en proceso de degradación.

**Software, Tablero Principal de Arranque Máquina de Niebla Salina.**

Este Software muestra al operario el primer tablero de control con el cual arrancar las funciones desarrolladas para la Maquina de Niebla Salina, su forma de inicializarlo es a través de la consola de comandos CLI, con el comando:



Este comando inicia el archivo de software "Machine.py" mediante la librería "python3" ubicada en la carpeta "bin". Dicho archivo configura la interfaz del tablero de control, así como la ejecución automática de procesos en segundo plano, como el registro de sensores y la activación del botón de parada de emergencia. Además, presenta paneles secundarios de control y registros al operador. La estructura del tablero se muestra a continuación.



**Software de visualización de estadística y grafica de Registros de sensores en la Máquina de**

**Niebla Salina en Funcionamiento.**

El software descrito se especializa en la lectura y visualización de registros obtenidos previamente, ofreciendo una interfaz intuitiva para el usuario. Se recomienda su uso después de abrir la interfaz de control del sistema, ya que esta inicializa automáticamente el registro de sensores en un nuevo archivo para su lectura. El programa permite también la visualización de datos de días anteriores, permitiendo al usuario ingresar la fecha de interés a través de tres casillas de entrada para año, mes y día. Estos datos ingresados son validados, asegurando que sean números enteros y que el mes esté en el rango de 1 a 12, convirtiéndolo internamente al nombre del mes en inglés. Similarmente, se verifica que el día esté entre 1 y 31, conforme al calendario gregoriano. En caso de fallo en la validación, se muestra una ventana emergente para corregir el ingreso. Una vez validados los datos, se realiza una búsqueda en la carpeta específica de registros por el nombre del archivo. En caso de encontrarlo, se informa al usuario tanto en la interfaz principal como a través de una ventana emergente, mostrando estadísticas iniciales de los datos y la hora de inicio y finalización de los registros leídos.

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

Además, se destaca que si se revisa el archivo correspondiente a la fecha actual, solo se leerán los datos hasta el momento de la búsqueda, ya que internamente el archivo funciona como un DataFrame de Pandas y solo conserva los datos en memoria temporal al momento de la lectura. Por lo tanto, el operario debe considerar el rango de horas disponibles en el archivo seleccionado para luego ingresar el rango de horas deseadas para la visualización gráfica de los datos.

Nótese que en caso de querer visualizar una sola hora no debe poner el rango del inicio de hora hasta su

finalización si no insertar la misma hora, esto es debido a que se toma la hora final en comparación con la hora final registrada la cual no se hallara en caso de poner la siguiente, por ejemplo:

*Si en el registro tenemos las siguientes horas [15:06:00 – 15:48:13]*

*El software mostrara que tiene un rango de: [15:00:00 – 15:59:59]*

*En caso de querer ver esta hora el operador se tentaría a probar [15 – 16]*

*Lo cual el software lo interpretaría como [15:00:00 - 16:59:59]*

*Así que sacara una ventana emergente en alerta pues no encuentra la hora especificada.*

*Por tanto, deberá ingresar [15 – 15]*

*Lo cual software interpretara como [15:00:00 – 15:59:59]*

*Rango que si existe dentro del software*

*Se procederá a graficar los datos encontrados desde [15:06:00 – 15:48:13]*

La representación gráfica de los sensores se presenta en una ventana emergente con tres gráficas dispuestas horizontalmente, correspondientes a las variables de "temp1", "temp2" y "temp3", que representan los sensores DS18B20-01, DS18B20-02 y LM35, respectivamente. El sensor cuatro, relacionado con la presencia y ausencia de líquido en el tanque de niebla salina, no se incluye en las gráficas debido a que su señal podría distorsionar el escalado de las anteriores gráficas al presentarse como una onda cuadrada. Además, la información sobre el porcentaje de presencia de líquido se detalla en la estadística mostrada en la interfaz principal del programa.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Software de Tablero de Control de conmutación de actuadores en la Máquina de Niebla Salina.**

Este software representa una parte integral y extensa del proyecto, fusionando la inicialización de procesos en segundo plano, las utilidades de las Raspberry Pi para la comunicación GPIO y la conmutación de pines. Su desarrollo implica un estudio detallado del comportamiento de librerías y la respuesta de los actuadores a las señales en los pines de la GPIO. La conmutación de pines es crucial en la computación física, permitiendo encender y apagar distintos dispositivos conectados a la Raspberry Pi, lo que facilita la expansión tecnológica futura de la Máquina de Niebla Salina mediante la adición de sensores o actuadores según las necesidades. Mediante análisis observacional, se ha determinado que el encendido de los pines es casi instantáneo, con un mínimo retardo de decenas de milisegundos, mientras que el apagado presenta una demora notablemente mayor, en el rango de las centenas de milisegundos, un aspecto importante a considerar en la implementación de proyectos.

Diagrama

Descripción generada automáticamente

Durante el desarrollo de la conmutación con la Raspberry Pi, se identificaron diversas librerías que cumplían con el propósito requerido, sin embargo, se observaron variaciones en su comportamiento respecto a la GPIO, lo que afectaba la fiabilidad de las operaciones. Se optó por una programación física más precisa, similar a la utilizada en microcontroladores como el PIC 16F877A, para garantizar la estabilidad en la comunicación de altos y bajos voltajes. Por ejemplo, la librería GPIOZERO, ampliamente recomendada y utilizada en la comunidad Raspberry Pi, presentó comportamientos no deseables al interactuar con la GPIO. Se comprobó mediante multímetro y osciloscopio que los pines de la GPIO, una vez activados en el código, no regresaban a un estado de apagado de 0VDC, sino que mantenían un voltaje mínimo de aproximadamente 0.20V DC a 0.40V DC, lo que podría interpretarse erróneamente como un estado lógico 1 y provocar el inicio de rutinas no deseadas en la Máquina de Niebla Salina.

Imagen que contiene Diagrama

Descripción generada automáticamente

Para este proyecto en el que se manejan gases corrosivos, es un comportamiento sumamente peligroso, por lo que se decidió trabajar con otras librerías, siendo la actual RPi.GPIO en uso en el presente prototipo, la cual ha demostrados mayor confiabilidad con los datos entregados por los pines. De la siguiente imagen la línea en seguida corresponde al comportamiento del voltaje con la librería GPIOZERO mientras que la línea punteada corresponde al voltaje en pines con la librería RPI.GPIO.

Imagen en blanco y negro

Descripción generada automáticamente con confianza baja

Para el operador de la Máquina de Niebla Salina, el software de control presenta una interfaz con dos tableros: uno que muestra los actuadores con botones de encendido y apagado, y otro con estadísticas del software y opciones para iniciar un proceso no supervisado con apagado automático. Al abrir el software, se inicia automáticamente la lectura y registro de sensores, así como el software de parada de emergencia. Cada botón del primer tablero se corresponde con un pin único de la GPIO de la Raspberry Pi, y el segundo tablero permite al operador ingresar una hora para que la máquina funcione hasta apagarse automáticamente. Se realizan comprobaciones de entrada para asegurar la validez de los datos ingresados por el usuario, y se advierte que el botón de parada de emergencia detendrá el ciclo de trabajo si detecta falta de líquido. Una vez iniciado el proceso no supervisado, el software se congelará hasta la hora predeterminada de apagado. Se recomienda reiniciar el software después de utilizar el proceso no supervisado para evitar posibles problemas de cache. Los registros de sensores realizados mientras el software esté abierto se encuentran en la carpeta "LOOT" para su revisión detallada.

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente

**Discusión:**

El software desarrollado para el funcionamiento de la Máquina de Niebla Salina esta subdividido en diferentes capas comprendidas entre el MCU, software en segundo plano de la Raspberry PI y software en primer plano con interfaz grafica de usuario GUI con la cual el operador puede manejar el prototipo. Cada una de estas piezas fue diseñada con el uso de librerías que no significaran un gasto de recursos de procesamiento según la plataforma en que fueron hechas.

Sin embargo, aunque hay una adecuación importante del software a su entorno operativo, este ultimo representa un limite de lo que se puede o no desarrollar, asi como el uso de ciertas tecnologías o el acoplamiento de nuevos componentes electrónicos durante un proceso de escalado tecnológico del proyecto.

Esto presupone una futura actualización del núcleo principal del sistema de computo, raspberry Pi, por una verison superior u otro modulo que cumpla la función requerida. El problema con esto es el posible rediseño de varias de las capas de software diseñadas actualmente.

Conclusiones

**Referencias**

Luis Felipe Narváez Gómez. Sistema de Monitoreo y Control para el Funcionamiento del Prototipo de la Maquina de Niebla Salina para los Laboratorios de la Facultad de Ingeniería Mecánica. Abril 2024. Libro de Tesis Pregrado, Facultad de Ingeniería de Sistemas, Universidad Santo Tomas, Trabajos a Futuro, pagina 80.