

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

SISTEMA DE RIEGO PARA INVERNADEROS CON INTERFAZ WEB UTILIZANDO UN ARDUINO YÚN

MANEJO DE ACTUADORES CON ARDUINO YÚN

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
TELECOMUNICACIONES**

DENNYS FRANCISCO SALAZAR DOMÍNGUEZ

dennys.salazar@epn.edu.ec

DIRECTOR: Ing. RAMIRO EDUARDO MOREJON TOBAR, M.Sc.

ramiro.morejon@epn.edu.ec

Quito, octubre 2022

CERTIFICACIONES

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



DENNYS FRANCISCO SALAZAR DOMÍNGUEZ

Certifico que el presente trabajo de integración curricular fue desarrollado por DENNYS FRANCISCO SALAZAR DOMÍNGUEZ, bajo mi supervisión.



Ing. RAMIRO EDUARDO MOREJÓN TOBAR, M.Sc.
DIRECTOR

DECLARACIÓN DE AUTORÍA

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

DENNYS FRANCISCO SALAZAR DOMÍNGUEZ

Ing. RAMIRO EDUARDO MOREJÓN TOBAR, M.Sc.

DEDICATORIA

Este trabajo y todo logro personal está dedicado a mi padre, madre y hermanos que me han apoyado en todos los sentidos, me han visto crecer, siempre brindando su amor incondicional, en sí, ellos han estado y sé que estarán por siempre en mi vida. Lo son todo para mí, los amo.

A mi familia tanto de la Sierra y Costa Ecuatoriana, cada uno de ustedes son una pieza fundamental que vive en mi corazón.

Al amor de mi vida, te amo con todo mi corazón.

A mis amigos de la niñez, del colegio, de la universidad y a toda persona con la que he compartido un instante fugaz o de varios años.

A la persona que sienta que no va a lograr cumplir sus metas, que ha sufrido depresión, que cree que está solo(a), ánimo que de alguna manera y que se desea con bien desde lo más profundo del corazón se puede alcanzar.

Francisco.

AGRADECIMIENTO

Agradezco a Dios, la Virgen María, Jesucristo y a todos los santos que han estado, están y estarán en el trayecto de mi vida y en el más allá.

A mi padre y super héroe Patricio, su enorme sabiduría y paciencia han invadido mi ser formando en mí el hombre que soy. Un gran hombre que seguir, un ejemplo de padre a aplicar, la historia de un increíble ser para siempre contar.

A mi madre y primer amor Patricia, la mujer que cabe en un alhajero cual zafiro, bella como luna llena, fuerte y radiante como sol de medio día, cálida siempre con su amor de mamá. La base de toda la familia, mi musa por la cual quiero luchar.

A Jeremy, hermano y compañero de aventuras, mi primer gran amigo e influyente artístico que día a día me sorprende e inspira con su gran habilidad. Tu sonrisa ilumina siempre mi caminar.

A Gregory, hermano y consentido de la casa, mi segundo gran amigo, la persona más risueña y comprometida que conozco. Me sacas mil y una risas al igual que con tu compromiso me inspiras a no rendirme, a luchar, a querer siempre mejorar.

A mi segunda madre Soledad, una mujer única, risueña, humilde, soñadora, luchadora, generosa y amorosa. Me ha cuidado, motivado e impulsado desde que he podido andar. Siempre llevo conmigo todo su amor incondicional.

A Julio, José, Vinicio, Paulina, Virgilio, mi familia y amigos que han estado desde siempre apoyándome con su cariño, con sus palabras, con sus acciones.

A Daniel, Alisson, Katherine, Josselyn, Doménica y Pablo, los pequeños que han provocado en mis sonrisas, me han visto llorar y me motivan a quererlos siempre impresionar.

A Nicole quien es y será el amor de mi vida, la mujer que ha provocado un huracán de emociones con su increíble belleza, su inusual sonrisa, lo magnífico de su inteligencia y su gran tenacidad. Tu apoyo, comprensión y amor son mis alas al volar.

A Hernán, amigo, hermano mayor, consejero y guía en muchos aspectos de mi vida. De mis grandes amigos. Su apoyo y cariño incondicional me han hecho llegar muy lejos. Una de las figuras que siempre me va a inspirar.

A Byron sangre de mi sangre, siempre me has apoyado y en ti siempre voy a poder contar.

A Alexander, David, Andrea, Gabriel, Bryan L., Bryan S., grandes amigos desde el inicio de la universidad que, con sus locuras, su inteligencia, sus ganas de superación me motivan día a día y llenan con su amistad muchos aspectos de mi vida.

A mi director del presente trabajo de integración curricular, Ing. Ramiro Morejón por ayudarme, por sus consejos y confianza durante todo el proceso.

A todas las amistades que pude formar y compartir durante mi trayectoria en tan grande institución, en la prestigiosa Escuela Politécnica Nacional.

A la persona que dedicó su tiempo para leer el presente trabajo de integración curricular.

ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
RESUMEN	IX
ABSTRACT	X
1 INTRODUCCIÓN.....	1
1.1 OBJETIVO GENERAL	2
1.2 OBJETIVOS ESPECÍFICOS	2
1.3 ALCANCE	3
1.4 MARCO TEÓRICO.....	6
1.4.1 INVERNADERO	6
1.4.1.1 Parámetros de las condiciones climáticas y del terreno a considerar para el diseño e instalación de los invernaderos	7
1.4.1.2 Efecto invernadero	10
1.4.1.3 Invernadero inteligente	10
1.4.2 ARDUINO YÚN.....	11
1.4.2.1 Alimentación.....	12
1.4.2.2 Interfaces de entrada y salida	12
1.4.2.3 Entorno de desarrollo integrado (IDE).....	13
1.4.3 ACTUADOR.....	16
1.4.3.1 Actuadores eléctricos	16
1.4.3.2 Cilindros hidráulicos	17
1.4.3.3 Cilindros neumáticos	17
1.4.3.4 Servomotores	18
1.4.4 RELÉ	18
1.4.4.1 Módulo relé de 8 canales	18
1.4.5 SISTEMA DE VENTILACIÓN PARA INVERNADERO	19
1.4.5.1 Sistema de ventilación natural o pasiva para invernadero	19
1.4.5.2 Sistema de ventilación forzada para invernadero.....	20
1.4.6 SISTEMA DE CALEFACCIÓN PARA INVERNADERO.....	20
1.4.7 SISTEMA DE RIEGO POR GOTEO.....	21

2	METODOLOGÍA	23
2.1	DETERMINACIÓN DEL PROBLEMA	23
2.2	SELECCIÓN Y APLICACIÓN DE LOS COMPONENTES DE HARDWARE 23	
2.2.1	PLATAFORMA DE DESARROLLO	24
2.2.2	DISPOSITIVOS MECÁNICOS.....	26
2.2.2.1	ACTUADOR PARA EL SISTEMA DE VENTILACIÓN.....	26
2.2.2.2	ACTUADOR PARA EL SISTEMA DE CALEFACCIÓN	28
2.2.2.3	ACTUADOR PARA EL SISTEMA DE RIEGO POR GOTEO	29
2.2.2.4	ACTUADOR PARA EL SISTEMA DE ILUMINACIÓN	30
2.3	DESARROLLO Y APLICACIÓN DE LOS COMPONENTES DE SOFTWARE.....	31
2.3.1	FLUJOGRAMA DE FUNCIONAMIENTO.....	31
2.3.2	CONEXIÓN DEL SISTEMA ARDUINO YÚN Y ACTUADORES.....	38
2.3.3	LIBRERÍAS.....	39
2.3.4	VARIABLES Y OBJETOS.....	39
2.3.5	FUNCIONES.....	45
2.3.6	CONSTRUCCIÓN DE UN INVERNADERO A ESCALA.....	47
3	RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	58
3.1	INICIO DEL SISTEMA, VERIFICACIÓN DE MÓDULO RTC DS3231 Y PRESENTACIÓN DEL SISTEMA DE SOPORTE VITAL.....	58
3.2	SISTEMA DE ILUMINACIÓN Y CONTROL DE LA HUMEDAD RELATIVA 61	
3.3	SISTEMA DE VENTILACIÓN PARA EL CONTROL DE ALTAS TEMPERATURAS Y BAJOS NIVELES DE CO2	69
3.3.1	VALORES EN RANGOS ADECUADOS.....	69
3.3.2	TEMPERATURA ALTA Y NIVELES DE CO2 BAJOS	70
3.3.3	TEMPERATURA ALTA Y NIVELES DE CO2 ADECUADOS	74
3.3.4	TEMPERATURA ADECUADA Y NIVELES DE CO2 BAJOS	76
3.4	SISTEMA DE CALEFACCIÓN PARA EL CONTROL DE TEMPERATURAS BAJAS	78
3.5	SISTEMA DE RIEGO POR GOTEO PARA EL CONTROL DE LA HUMEDAD DEL SUELO.....	80
3.6	CONCLUSIONES.....	83
3.7	RECOMENDACIONES	85
4	REFERENCIAS BIBLIOGRÁFICAS	86

5 ANEXOS.....	93
ANEXO I.....	94

RESUMEN

El presente Trabajo de Integración Curricular contempla el componente relacionado con el manejo de actuadores en un invernadero. La parte fundamental de este proyecto se centra en el desarrollo de un código implementado en la plataforma Arduino YÚN. Esta plataforma permite, mediante sus interfaces de I/O, la manipulación de actuadores encargados de manejar componentes físicos. Esto se consigue mediante la transformación de la energía suministrada en la entrada del dispositivo en otro tipo de energía a la salida para controlar y regular los parámetros del sistema de soporte vital del invernadero.

La plataforma Arduino YÚN permite un correcto diseño de los dispositivos conectados, en este caso, los actuadores los cuales son manipulados gracias a la gran ventaja que presenta esta placa pues combina el sistema operativo Linux con la facilidad de utilización de la plataforma Arduino.

El documento se estructura en cuatro partes. En el capítulo uno se detalla brevemente los conceptos y definiciones generales útiles para el desarrollo del contenido del documento.

Seguidamente, en el capítulo dos se describe el código implementado en la plataforma Arduino YÚN, así como el detalle de construcción del prototipo.

En el tercer capítulo se presentan las pruebas realizadas y el correcto funcionamiento del prototipo. Finalmente se exhiben las conclusiones y recomendaciones derivadas del proyecto.

PALABRAS CLAVE: Actuadores, Arduino YÚN, Invernadero, Linux, Relés.

ABSTRACT

This work involves the practical component related to the handling of actuators in a greenhouse. The main part of this project focuses on the development of a code implemented on the Arduino YÚN platform. This platform allows, through its I/O interfaces, the manipulation of actuators that handle physical components. This is achieved by transforming the supplied energy at the input of the device into another type of energy at the output of it to control and regulate the greenhouse life support system.

The Arduino YÚN platform allows a correct design of connected devices in this case, the actuators which are manipulated thanks to the great advantage that this board offers because it combines the Linux operating system with the ease of use of the Arduino platform.

This document is structured in four parts. Chapter one briefly details the general concepts and definitions useful for the development of the content of the document.

Next, chapter two describes the implemented code in the Arduino YÚN platform, as well as the construction detail of the prototype.

The third chapter presents the tests carried out and the correct functioning of the prototype. Lastly, conclusions and recommendations derived from the project.

KEYWORDS: Actuators, Arduino YÚN, Greenhouse, Linux, Relays.

1 INTRODUCCIÓN

La denominada cuarta revolución industrial está presente y avanza globalmente convirtiendo la vida del hombre para un mejor porvenir. Las tecnologías de la información y de comunicación son la base de esta sublevación que hacen uso de Internet el cual es un factor potenciador necesario que permite la evolución de la nueva era social y económica [1].

Esta revolución lleva consigo entre sus grandes cambios el desarrollo e implementación del Internet de las Cosas (IoT, por su sigla en inglés de Internet of Things), donde cualquier aparato electrónico tiene la capacidad de conectarse a una red que mediante software permite ser manipulado, entregando también información útil de procesos o parámetros a las cuales éste está destinado [2].

El mundo actual presenta una alta demanda de alimentos. En este contexto, los agricultores y empresas que se desarrollan en los sectores rurales en este tipo de actividades buscan aumentar su productividad con alta calidad y ahorro en el proceso. Para lograr estos objetivos se debe ajustar e inspeccionar de manera adecuada los factores climáticos del entorno. Por ejemplo, la humedad relativa del aire es un factor climático importante que influye en el rendimiento final de los cultivos [3]. Por lo cual es necesario hacer empeño en optimizar y mejorar los sistemas que se encargan de proporcionar ventilación, agua, nutrientes y luz solar artificial en los cultivos bajo invernadero.

Los invernaderos inteligentes han tenido un gran impacto pues permiten obtener cultivos de altos atributos que no son autóctonos del lugar o que se encuentren fuera de época dentro de un ambiente controlado y regulado según las características del cultivo. Gracias a estos invernaderos, la productividad aumenta, las plagas son eliminadas y existe un ahorro en agua y fertilizantes. Esta clase de invernaderos funcionan gracias a los sistemas embebidos que son la agrupación de elementos electrónicos que contienen un microprocesador, es decir, son el conjunto de sensores, actuadores y microcontroladores para convertir un sistema convencional en un sistema automatizado. Los datos son recopilados mediante microsensores siendo el primer paso de un procedimiento mecanizado. La creación de modelos algorítmicos sirve para analizar toda la información recopilada que se procesa dentro de las funciones propias del lenguaje, siendo ejecutadas en el microprocesador para tomar decisiones que se reflejan en el trabajo de los actuadores establecidos [1].

El sistema de riego por goteo y la ventilación de tipo forzada en conjunto con el uso de un calefactor e iluminación artificial son una propuesta en este trabajo para solucionar los

problemas presentes debido a los cambios que se produzcan en el clima dentro del invernadero. Se consideran, para la propuesta, dos actuadores posicionados para la ventilación los cuales encenderán dos ventiladores para regular la temperatura proporcionando aire no viciado para mantener el ambiente fresco, controlar la humedad relativa y los niveles de CO₂. Otros actuadores se encargarán del sistema de riego por goteo los cuales permiten el paso de agua proporcionando la cantidad justa y necesaria en un lapso, mejorando los niveles de humedad en el suelo. De igual manera, éstos encienden el sistema de iluminación para aumentar el fotoperiodo del cultivo.

Las variables usadas dentro del código para el conjunto de datos (parámetros definidos) de temperatura máxima y mínima, niveles de CO₂, humedad del suelo, humedad relativa, cantidad de líquido e iluminación varían acorde a los cultivos. No se puede mantener, por ejemplo, un invernadero con cultivos de tomate en las mismas condiciones que un invernadero con cultivos de manzanilla. Es por ello que éstos son ajustables según las necesidades, sin embargo, se hará uso para el presente trabajo técnico de parámetros generales.

1.1 OBJETIVO GENERAL

El objetivo general del presente Trabajo de Integración Curricular es generar código que permita el manejo de actuadores encargados de realizar operaciones mecánicas para permitir el funcionamiento del sistema de riego, ventilación, iluminación artificial y calefacción para el control del sistema de soporte vital del invernadero.

1.2 OBJETIVOS ESPECÍFICOS

Los objetivos específicos del presente Trabajo de Integración Curricular son:

- Establecer la familiarización con la plataforma Arduino YÚN y el manejo de las interfaces I/O para el control de periféricos.
- Elegir los actuadores apropiados a los requerimientos del invernadero.
- Definir una estructura de variables y la asignación de pines necesarios de la plataforma Arduino YÚN para el manejo de los actuadores.
- Implementar el código que permita la operación de los actuadores en base a los parámetros requeridos del invernadero.

1.3 ALCANCE

En el presente trabajo de titulación se desarrolla un algoritmo, para la plataforma Arduino YÚN, que utiliza los parámetros de una estructura de variables para tomar decisiones que mediante los interfaces de I/O permitan el manejo de actuadores que se encargaran de controlar y manipular los sistemas de: riego por goteo, ventilación forzada, calefacción y de iluminación artificial.

En la Figura 1.1 se presenta un esquema del proyecto el cual explica cómo los elementos interactúan entre sí para el funcionamiento en base al análisis de los datos de entrada en el Arduino YÚN. Posteriormente se envía(n) la(s) señal(es) eléctrica(s) a los actuadores para que realicen la actividad respectiva de regulación del sistema de soporte vital. Después cada cinco minutos, se realiza una nueva lectura repitiendo el proceso. Las respectivas pruebas de funcionamiento se ejecutan en un invernadero a escala cubierto por un plástico tipo mica.

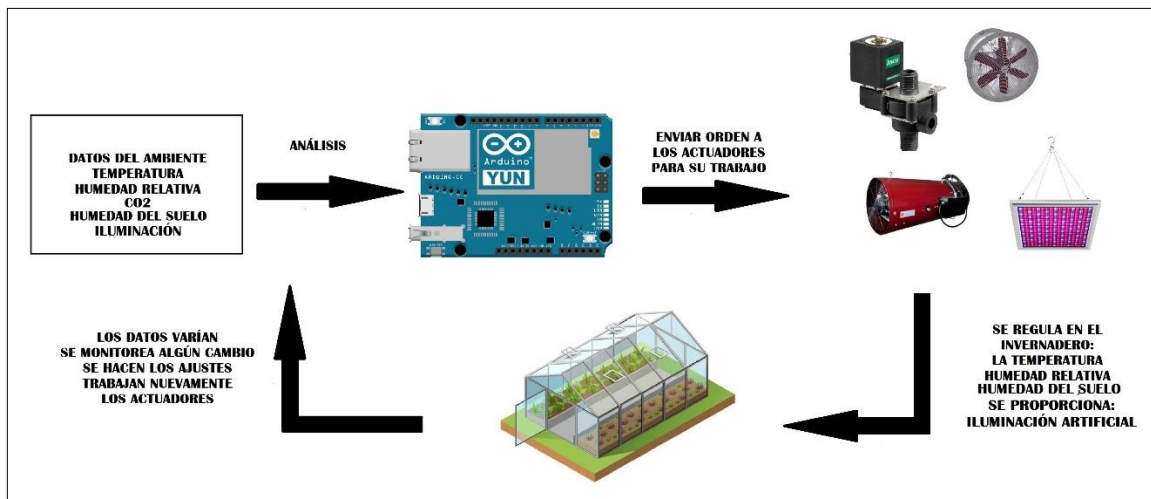


Figura 1.1 Esquema del trabajo que realiza la plataforma Arduino YÚN en conjunto con los actuadores para controlar el ambiente del invernadero.

La estructura de variables, localizada en la memoria SRAM del sistema de la plataforma Arduino YÚN, contiene los datos de temperatura, humedades y CO₂ que servirán para establecer la condición del invernadero y actuar en consecuencia. Cabe señalar que este componente no realizará la adquisición de esta información por lo que se crearán rutinas de prueba que permitan modificar estos datos para desencadenar los procesos que permitan regular el clima del invernadero. La interrupción creada y que se activa manualmente al pulsar un botón indica al microcontrolador que realice una lectura de los datos del ambiente modificando así los valores iniciales cargados en memoria.

El uso de un módulo RTC DS3231 permite el funcionamiento del sistema. Este módulo es un reloj de tiempo real que adquiere la hora y fecha establecida de forma manual o automática proporcionada por una computadora [4]. La ventaja de esta información permite tener en cuenta al algoritmo cuándo funcionar, para ello se ha establecido el siguiente horario de 2 AM a 6 PM, dividido de la siguiente manera:

- De 2 AM a 6 AM funciona principalmente el sistema de iluminación para aumentar el fotoperiodo, el sistema de ventilación y calefacción para regular la humedad relativa y temperaturas bajas.
- De 6 AM a 6 PM funciona el sistema de ventilación controlando las altas temperaturas y niveles de CO₂ en conjunto con el sistema de calefacción para regular las bajas temperaturas.
- De 7 AM a 10 AM y de 3 PM a 5 PM funciona el sistema de riego regulando la humedad relativa del suelo.

Se debe entender cómo los factores externos al invernadero intervienen en la estructura cerrada. En el exterior, es decir, en el ambiente donde se desarrollan los seres vivos, se produce una elevación de temperatura de forma constante la cual es reemplazada gracias a que existen corrientes de convección [5] que se encargan de efectuar el ascenso del material caliente y a su vez el descenso del material frío [6]. Sin embargo, en el ambiente interno del invernadero el suelo produce una elevación en la temperatura del aire y como este es limitado se calienta rápidamente y con mucha facilidad [5]. Esto puede ser corregido con ventilación pasiva o conocida también como natural, pero se busca una mejor distribución del aire fresco el cual en base a estudios [7] se garantiza un flujo más homogéneo haciendo uso del sistema de ventilación forzado en base al trabajo del actuador para el encendido de los ventiladores. El aire fresco ingresa por un extremo y se expulsa por otro extremo el aire viciado, por lo general, por ventilas de aire. Por otra parte, también haciendo uso de los datos de temperatura se debe tener en cuenta que ésta desciende por las noches, en los días fríos o cuando se producen heladas. El encendido de la calefacción funciona gracias al actuador encargado incrementando la temperatura para regular el ambiente.

Los parámetros de la humedad relativa, humedad del suelo y niveles de CO₂ son de consideración. El control de estas medidas se produce gracias al sistema de ventilación forzado y calefacción. La humedad relativa es la humedad que se encuentra presente con respecto al máximo de humedad posible en mismas condiciones de temperatura ambiente [8]. Cuando la temperatura dentro del invernadero aumenta, la humedad relativa

disminuye y viceversa debido a su relación inversamente proporcional. Esto se vigila activando el actuador encargado de la ventilación forzada provocando así que la humedad relativa esté en niveles requeridos entre el 50% y 75%, como también la temperatura entre 15 y 28 °C, y finalmente el CO2 entre 300 a 400 ppm. Pero si se inyecta manualmente este gas incoloro, el máximo valor para el rango recomendado es de 300 a 1200 ppm. Por otra parte, cuando la temperatura interna disminuye, genera un aumento en la humedad relativa. Estos niveles altos se controlan haciendo uso del sistema de ventilación en conjunto con la calefacción [3].

La humedad del suelo que es la cantidad de agua en cierto volumen de tierra permite conocer qué tanto volumen de este líquido vital puede ser absorbido y utilizado para realizar los procesos de crecimiento, germinación, entre otros, por parte de los cultivos [9]. El actuador dispuesto para este trabajo activa el sistema de riego por goteo durante una o más veces por día, uno o varios días a la semana. La recomendación según el tipo de emisor y tiempo de riego se indica en la Tabla 1.1. El tiempo de riego puede variar y se basa en la lectura de la humedad. Si éste se encuentra en niveles muy bajos se activa el riego para distribuir el líquido en la tierra verificándose que se encuentre en un rango de 10 a 50% lo que garantiza que la planta no se marchite y muera, así también evitando la saturación de agua, la propagación de enfermedades y muerte del cultivo.

Tabla 1.1. Emisores de goteo comunes. [10]

Tipo de emisor [galones por hora, gph]	Tiempo de riego [minutos en los días que se riega, min]
Emisor de flujo alto (Hasta 20 gph)	12
Emisor de flujo bajo (Hasta 4 gph)	30
Emisor de flujo bajo (Hasta 2 gph)	60
Emisor de flujo bajo (Hasta 1 gph)	90

El sistema de iluminación proporciona iluminación de tipo artificial el cual hace uso de dos tiras de LEDs RGB direccionable ideales para el manejo de Arduino que permiten alcanzar el espectro de la luz visible que se encuentra entre 400 nm - 700 nm alternando con el encendido de un LED color azul y tres LEDs de color rojo a lo largo de ambas tiras que contienen 19 LEDS cada una. El proceso de fotosíntesis se logra mediante este sistema artificial por que la combinación de estos colores logra alcanzar un espectro similar al de la luz visible como se observa en la Figura 1.2.

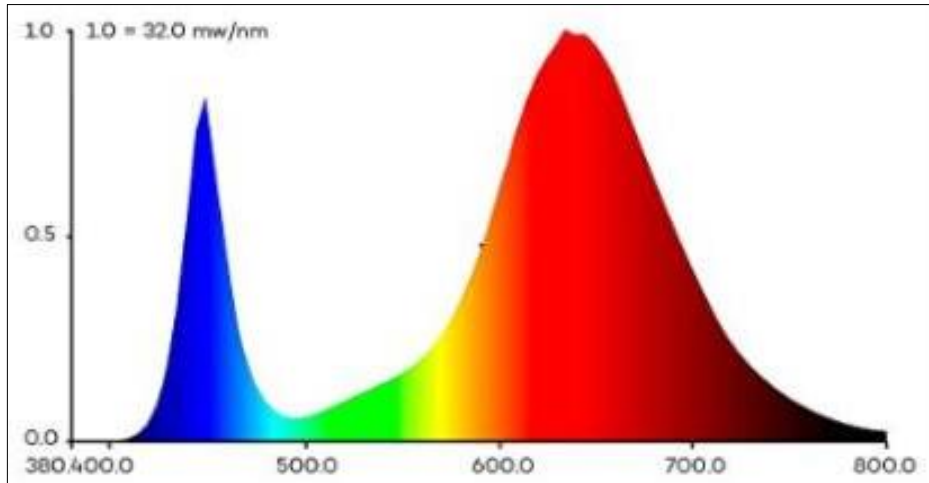


Figura 1.2 Espectro de LED para producción vegetal. [11]

1.4 MARCO TEÓRICO

En el presente componente se hará uso de la tecnología desarrollada por la empresa Arduino, específicamente de la plataforma Arduino YÚN, en conjunto con actuadores, los cuales harán el trabajo mecánico para manipular los sistemas de riego por goteo, ventilación forzada y calefacción para controlar el ambiente dentro del invernadero. Las variables obtenidas del ambiente son el eje principal para el funcionamiento del código, en base a éstas, se tomarán decisiones para controlar y regular el ambiente. En este apartado se dan a conocer conceptos teóricos básicos para la comprensión y desarrollo del documento.

1.4.1 INVERNADERO

Un invernadero se puede definir como:

“...un recinto delimitado por una estructura de madera o de metal, recubierta por vidrio o cualquier material plástico de naturaleza transparente, en cuyo interior suelen cultivarse hortalizas y plantas ornamentales en épocas durante las cuales las condiciones climáticas externas no permitirían obtener el producto deseado.” [12]

La finalidad de construir este tipo de estructuras radica en tres ejes principales [13]:

1. Obtener cultivos fuera de época
2. Aumentar la producción
3. Mejorar la calidad

Estos objetivos se logran gracias a que los cultivos se encuentran bajo cubierta. Los métodos de cuidado, condiciones favorables, el control de los nutrientes y agua permiten alcanzar los estándares establecidos.

Para implementar un invernadero se debe tener en cuenta muchas variables entre ellas está el clima donde se va a ubicar el invernadero (ambiente externo), qué tipo de cultivos se van a plantar, el tipo de material de la estructura, el recubrimiento si es de plástico o vidrio, tipo de ventilación, cubiertas para evitar mucha radiación, sistemas de riego, el suelo y dónde se cultivarán las plantas, el uso de tecnología, entre otros factores. Para esto último se debe considerar varios aspectos ya que la tecnología permite automatizar el proceso tradicional de los cultivos en donde una o varias personas se encargan de forma manual de brindar agua, nutrientes, en sí, de todo lo necesario para el correcto crecimiento y desarrollo del cultivo. El uso de tecnología brinda la facilidad de realizar todo esto gracias al uso de diferentes dispositivos electrónicos y mecánicos. Todo esto debe ser cuestionado de la siguiente forma [13]:

- La rentabilidad de la tecnología con base a las necesidades para así evitar gastos innecesarios y mantener un presupuesto adecuado ya que las inversiones iniciales son fuertes.
- La tecnología usada debe estar acorde a las condiciones de la economía local, esto porque los precios varían en el mercado.
- Sencillas en la tecnología, no se debe exagerar con nada en primera instancia logrando implementar los elementos para una correcta productividad.

1.4.1.1 Parámetros de las condiciones climáticas y del terreno a considerar para el diseño e instalación de los invernaderos

Para ubicar el invernadero en el lugar deseado se debe considerar algunos factores del terreno. La orientación de las estructuras es por donde se debe comenzar ya que esto conlleva ventajas y desventajas con respecto a la influencia de la radiación, el viento, temperatura, precipitación pluvial en el ambiente interno.

Radiación

El efecto de la radiación solar se considera con base a dos diferentes puntos de vista. La transmisión de la radiación fotosintéticamente activa que es la que comprende las longitudes de onda del espectro visible que permite hacer la fotosíntesis y elaborar carbohidratos por parte de las plantas [14]. La disposición del techo provoca que esta radiación ingrese y presente un ángulo de incidencia y por otro también permite una

distribución en los cultivos. La Figura 1.3 es una ilustración de cómo inciden los rayos del sol según la disposición del invernadero con orientación E-O como se indica en la Figura 1.3a, y con orientación N-S como se indica en la Figura 1.3b.

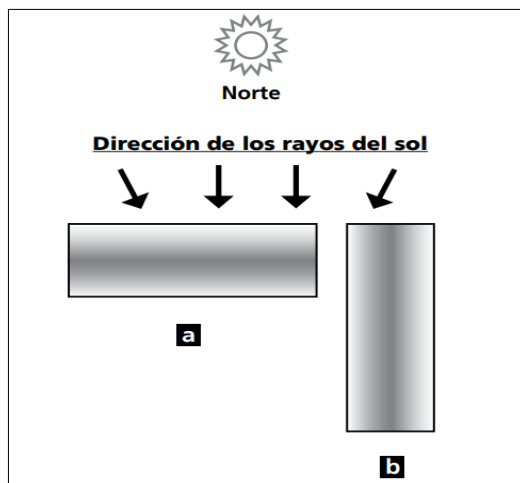


Figura 1.3 Incidencia del sol en invernaderos. a) Invernadero con orientación E-O.
b) Invernadero con orientación N-S [13].

La latitud y época del año influyen en gran medida en los factores mencionados anteriormente. Para los meses de invierno se generan problemas en lugares de gran altitud. En latitudes superiores a 30° se recomienda construcciones de invernaderos en sentido E-O para evitar muchos inconvenientes en los cultivos por causa de la temperatura en épocas de invierno y otoño. Para regiones con clima tropical cambian las consideraciones, se procura construcciones en base a la corriente del viento para el aprovechamiento de este recurso generando así una mejor ventilación y control de la humedad y temperatura [13].

Viento

El viento es un recurso importante en cualquier ambiente donde se tengan cultivos, en especial, dentro de un invernadero. Este recurso tiene sus pros y sus contras, de manera general, éste ayuda a la ventilación, regula la humedad relativa y temperatura. En sistemas de ventilación natural es un factor importante y, dependiendo del lugar, latitud, clima, se debe considerar que puede ser también peligroso. Por ejemplo, si un viento fuerte llega al lugar, éste dañaría a la estructura, afectaría los cultivos y podría traer plagas [13].

La orientación del invernadero debe ser tal que el eje mayor coincida con la dirección de los vientos predominantes, en zonas con vientos fuertes para reducir la presión para evitar daños en las estructuras. El uso de cortinas cortavientos reduce la velocidad del viento lo cual permite una mayor durabilidad en las estructuras del invernadero. La Figura 1.4 indica

la orientación del invernadero con respecto al viento. Si el lugar presenta vientos predominantes del oeste se recomienda usar una orientación E-O para el aprovechamiento adecuado tanto de la luz solar como del viento.

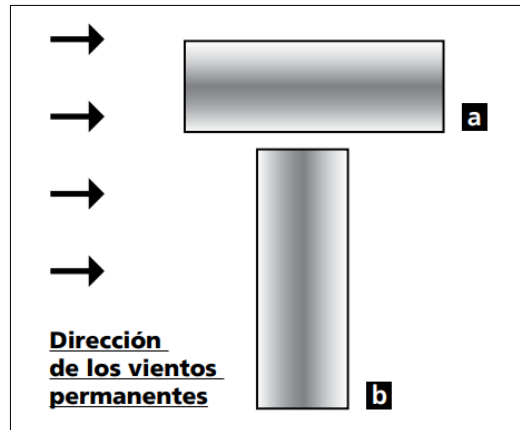


Figura 1.4 Orientación del invernadero con respecto al viento [13].

Precipitación pluvial

Este es un factor externo que, si bien no afecta directamente a los cultivos debido a la protección que presenta el invernadero, si tiene consecuencias a la hora de construir. Las diversas formas de precipitación se presentan a continuación [13].

La lluvia es un factor por considerar. El régimen de caída afecta al material utilizado como recubrimiento del invernadero que debería aguantar las gotas variables en tamaño, así como la cantidad de horas que llueva anualmente. El uso de canaletas y techos con pendientes son de utilidad para evitar que el agua se acumule en una estructura.

El granizo son cuerpos de agua congelada que caen con cierta velocidad y con diferente tamaño. Revisar la frecuencia en las que se produce este fenómeno, ayuda a reforzar la protección del invernadero. En casos extremo se aconseja hacer uso de una cubierta antigranizo.

La nieve es una precipitación en forma de cristales de hielo que se puede acumular en diversas zonas del invernadero, especialmente el techo. Incluso si la nevada es fuerte, ésta puede impedir el ingreso al mismo bajando drásticamente la temperatura interna. Para la construcción de invernaderos, es importante considerar el cálculo de la sobrecarga por nieve en donde se suele utilizar los coeficientes provenientes de los establecidos para la construcción de edificios [15].

Temperatura

Dependiendo de la región donde se ubique el invernadero, en su estructura se debe considerar factores que permitan tanto una correcta ventilación como la retención de calor de ser necesario. Se debe optimizar el uso de la energía que se acumula durante el día evitando gastos innecesarios con el uso del sistema de calefacción. De manera análoga, se busca un correcto sistema de ventilación para mantener niveles adecuados de temperatura cuando se tengan horas con aire caliente.

1.4.1.2 Efecto invernadero

La atmósfera está constituida por diversos gases los cuales tiene un comportamiento distinto, tal que la energía que absorben estos gases se utiliza de forma selectiva en diferentes longitudes de onda siendo incluso para algunos casos transparentes en específicos rangos del espectro. La atmósfera presenta para la parte visible del espectro (400 nm - 700 nm) poca capacidad de absorción y se presenta muy transparente, en cambio, para la radiación ultravioleta o de onda corta tiene un alto nivel de absorción, al igual que para la radiación infrarroja todo esto gracias al ozono, vapor de agua, dióxido de carbono, entre otros [16].

La atmósfera actúa como un filtro radiactivo, permite el paso de los rayos solares que en la superficie terrestre son absorbidos lo que provoca una radiación terrestre, la misma que es absorbida por las nubes y propiamente por la misma atmósfera. Las capas atmosféricas se calientan y envían nuevamente la radiación térmica hacia abajo lo que genera nuevamente el proceso mencionado una y otra vez. Por esta razón "el nombre efecto invernadero proviene de su similitud con las instalaciones construidas para cultivar plantas en un ambiente más cálido que el exterior; dado que el techo de un invernadero tiene la misma propiedad de dejar entrar la radiación solar y bloquear la terrestre generada en su interior" [17].

1.4.1.3 Invernadero inteligente

El invernadero inteligente presenta la característica de hacer uso de sistemas embebidos los cuales son "sistemas electrónicos diseñados específicamente para realizar unas determinadas funciones, habitualmente formando parte de un sistema de mayor entidad" [18]. El uso de sensores, actuadores y controladores permite la implementación de sistemas de tipo mecánico, eléctricos y electrónicos para automatizar el proceso de crecimiento, desarrollo, cuidado y monitoreo de los cultivos bajo techo. Estos sistemas se aplican de manera conjunta para generar sistemas de riego, ventilación, suministro de nutrientes, calefacción, entre otros [19].

Las variables ambientales como la intensidad luminosa, temperatura, humedad relativa, humedad del suelo, pH (acidez) son controladas y reguladas gracias al uso de estos elementos.

1.4.2 ARDUINO YÚN

Arduino YÚN es una placa de desarrollo que se base en el Sistema Operativo Linux que permite una conexión a redes y aplicaciones haciendo que los trabajos desarrollados en conjunto con esta placa entren a la categoría de Internet de las Cosas (IoT). Este primer miembro de esta serie de placas desarrollada por Arduino hace uso de un microchip ATmega32U4 en conjunto con un módulo SOC que permite la ejecución de una distro de Linux conocida como Linino, basada en OpenWRT [20]–[22]. Las Tabla 1.2 y Tabla 1.3 muestran las características de los microprocesadores Linux y AVR de la placa.

Tabla 1.2. Características Microprocesador de Linux. [23]

Procesador	Atheros AR9331
Arquitectura de MIPS	400 MHz
Voltaje de funcionamiento	3,3 V
Ethernet	IEEE 802.3 10 /100Mbit /s
Wifi	IEEE 802.11b/g/n
USB	tipo A 2.0 Host
Ranura de tarjeta de memoria	Solo microSD
RAM	64 Mb DDR2
Memoria flash	16 MB

Tabla 1.3. Características Microcontrolador AVR Arduino. [23]

Microcontrolador	ATMega32u4
Tensión de funcionamiento	5V
Voltaje de entrada	5V
Pines de E/S digitales	20
Canales PWM	7
Pines de entrada analógica	12
Corriente continua a través de los terminales de entrada/salida	40 mA
Corriente continua a través de la salida 3.3V	50 mA
Memoria flash	32 KB, de los cuales 4 KB son usados por el cargador
RAM y SRAM	2.5 KB
EEPROM no volátil	1 KB
Frecuencia de reloj	16 MHz

1.4.2.1 Alimentación

La alimentación se suministra a través de la conexión micro-USB con 5 VDC. Si se alimenta a través del pin Vin, se debe suministrar 5 VCC regulados. También, es compatible con alimentación PoE, pero para ello se debe montar el módulo PoE. La placa no tiene regulador de voltaje incorporado, por lo cual, para voltajes más altos, ésta se dañaría. En la Tabla 1.4 se muestran los pines de alimentación.

Tabla 1.4. Pines de alimentación de la placa Arduino YÚN. [21]

VIN	Voltaje de entrada a la placa Arduino
5V	Fuente de alimentación utilizada para alimentar los microcontroladores y otros componentes de la placa
3V3	Suministro de 3,3 voltios generado por el regulador de a bordo El consumo máximo de corriente es de 50 mA
GND	Pines de tierra
IOREF	El voltaje al que funcionan los pines de E/S de la placa (es decir, VCC para la placa). Esto es 5V.

1.4.2.2 Interfaces de entrada y salida

La Figura 1.5 describe las entradas de los elementos de la placa Arduino YÚN. En ésta se observa el microcontrolador, los pines digitales, los procesadores, la entrada para conexión Ethernet, USB, conexión a WiFi, entre otros.

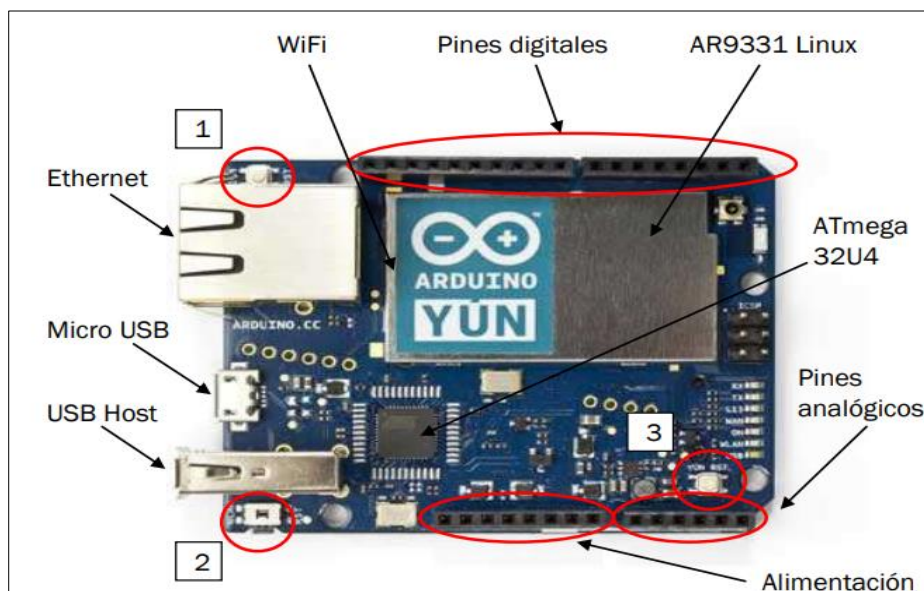


Figura 1.5 Placa Arduino YÚN. [24]

A continuación, se presenta la función de cada PIN [24]:

- Serial: 0 (RX) y 1 (TX). Permite recibir (RX) y transmitir (TX) datos serie TTL.

- TWI: 2 (SDA) y 3 (SCL). Soporta la comunicación TWI.
- Interrupciones externas: 3, 2, 0, 1 y 7. Se pueden configurar para provocar una interrupción.
- PWM: 3, 5, 6, 9, 10, 11 y 13: Proporciona una salida PWM de 8 bits con la función `analogWrite()`.
- SPI: En el conector ICSP. Estos terminales soportan la comunicación SPI a través de la librería SPI
- LED 13: El pin digital 13 lleva conectado un LED integrado en la propia placa (L13). Se encenderá cuando dicho pin se configura como salida y adopte un valor HIGH, mientras que con el valor LOW se apaga.
- Otros indicadores de estado: La placa incorpora otros LEDs para ver el estado de la alimentación, la conexión WLAN, conexión WAN y USB.
- AREF: Voltaje de referencia para las entradas analógicas. Se utiliza `analogReference()`.
- Entradas analógicas: A0 - A5, A6 – A11 (en los pines digitales 4, 6, 8, 9, 10 y 12). El Arduino YÚN tiene 12 entradas analógicas. Cada entrada proporciona 10 bits de resolución (1024 valores).

1.4.2.3 Entorno de desarrollo integrado (IDE)

El Entorno de Desarrollo Integrado IDE (Integrated Development Environment) es la agrupación de diversas herramientas de software que permiten realizar código el cual puede ser cargado a una de las tantas placas desarrolladas por Arduino para la ejecución de éste. El IDE permite escribir, depurar, editar y grabar el programa o también conocido como "sketches" de forma sencilla, gracias a lo amigable de su interfaz gráfica [25]. La Figura 1.6 indica las secciones que presenta el IDE.

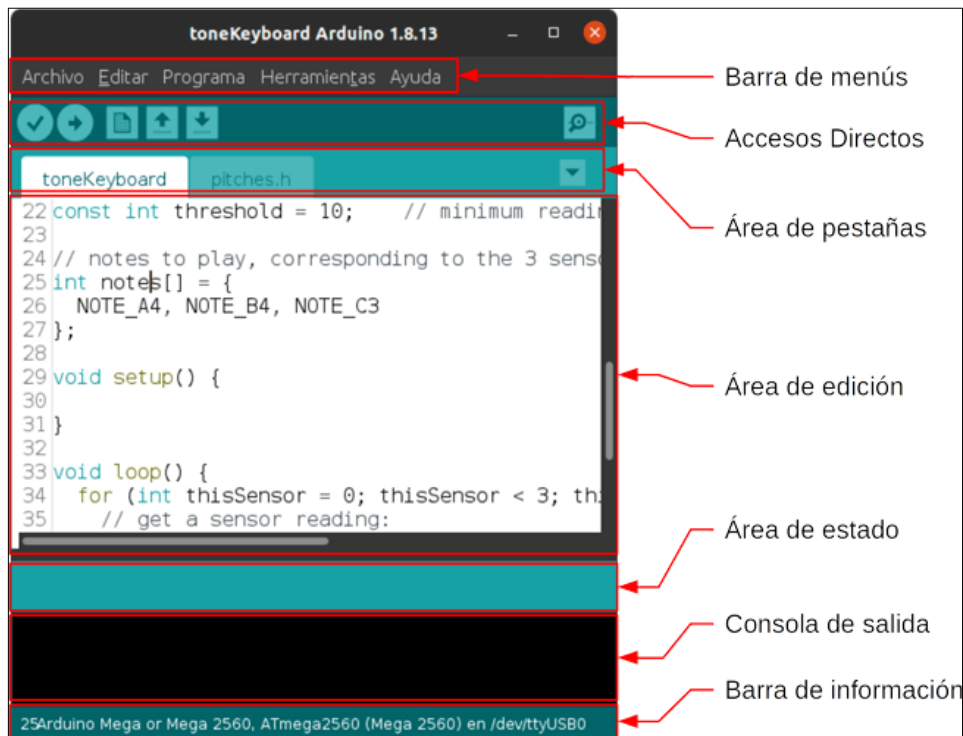


Figura 1.6 Secciones del Entorno de Desarrollo Integrado (IDE) de Arduino. [26]

- Barra de menús: en esta barra se encuentran varios menús que permiten acceder a muchas de las acciones disponibles en el entorno de desarrollo.
- Barra de accesos directos: en esta barra se encuentra un grupo de botones con las acciones más comúnmente realizadas (como compilar y cargar el código al Arduino).
- Área de pestañas: en esta área se muestran los archivos correspondientes al proyecto abierto.
- Área de edición: en esta área es donde se escribe el código.
- Área de estado: en esta área se muestra una barra de progreso cuando se compila o carga el sketch al Arduino.
- Consola de salida: en esta se muestran las salidas generadas por el compilador.
- Barra de información: se muestra información sobre la configuración del IDE.

Los elementos de la sección de Accesos Directos, Área de Pestañas y Barra de menús son los más utilizados, se detallan cada uno de ellos en la Tabla 1.5 y en la Tabla 1.6 respectivamente [25].

Tabla 1.5. Elementos de la sección de Accesos Directos y Áreas de pestañas del IDE de Arduino [25].


Símbolo	Descripción
	Verificar: comprueba si existen errores en el código, si no hay procede a compilar
	Subir: carga en la memoria del microcontrolador el programa desarrollado, se utiliza después de verificar
	Nuevo: crea un nuevo sketch vacío
	Abrir: despliega un menú con todos los sketches disponibles para abrir, sean que estén guardados en la PC o ejemplos proporcionados por Arduino
	Guardar: guarda el código de nuestro sketch en un fichero, el cual tendrá la extensión “.ino”
	Monitor Serial: abre el monitor serial y permite observar información transmitida desde el Arduino por el puerto de comunicación serial
	Menú contextual: permite abrir nuevas pestañas

Tabla 1.6. Elementos de la sección de Barra de menús del IDE de Arduino [25].

Submenú	Descripción
Archivo	Ofrece acciones estándar como crear un nuevo documento sketch, abrir uno existente, guardarlo, cerrarlo, cerrar el IDE, etc.
Editar	Ofrece acciones estándar como deshacer y rehacer, cortar, copiar y pegar texto, seleccionar todo el texto o buscar y reemplazar texto
Programa	Ofrece la acción de verificar/compilar el sketch, la de abrir la capeta donde esta guardado el fichero “.ino” que se está editando en este momento, la de añadir en una nueva pestaña un nuevo fichero de código al sketch y la de importar librerías
Herramientas	Se ofrecen diferentes herramientas variadas, como la posibilidad de auto formatear el código para hacerlo más legible, guardar una copia de todos los sketches del proyecto actual en formato .zip, abrir el monitor serie, etc.
Ayuda	Acceso a varias secciones de la página web oficial de Arduino que contienen diferente artículos, tutoriales y ejemplos de ayuda
Monitor Serial	Permite desde el computador enviar y recibir datos textuales a la placa Arduino usando el cable USB (el cual utiliza la conexión serie). Por otro lado, los datos recibidos provenientes de la placa serán mostrados en la sección central del “monitor serial”. Es importante elegir la misma velocidad de transmisión que se haya especificado en el programa ejecutado en la placa, porque si no, los caracteres transferidos no serán reconocidos correctamente

1.4.3 ACTUADOR

Los actuadores son dispositivos que reciben una energía en su entrada la cual es transformada en otro tipo de energía en su salida para realizar un trabajo. Las magnitudes físicas del entorno gracias al uso de los actuadores aprueban transformarlas en otras magnitudes para provocar un cambio en el estado del sistema en el cual se está usando, esto es beneficioso porque permite mecanizar los procesos en la industria [27].

1.4.3.1 Actuadores eléctricos

El actuador eléctrico se encarga de convertir la energía eléctrica que recibe en energía mecánica. En la Figura 1.7 se observa la acción del campo magnético en una espira, este cambio de energía se logra debido al:

“...efecto que produce una espira conductora que se encuentra dentro de la acción de un campo magnético y por la que circula una corriente eléctrica. En estas condiciones se experimenta una fuerza electromagnética en la espira, la cual induce un desplazamiento perpendicular a las líneas de acción del campo magnético. “ [27]

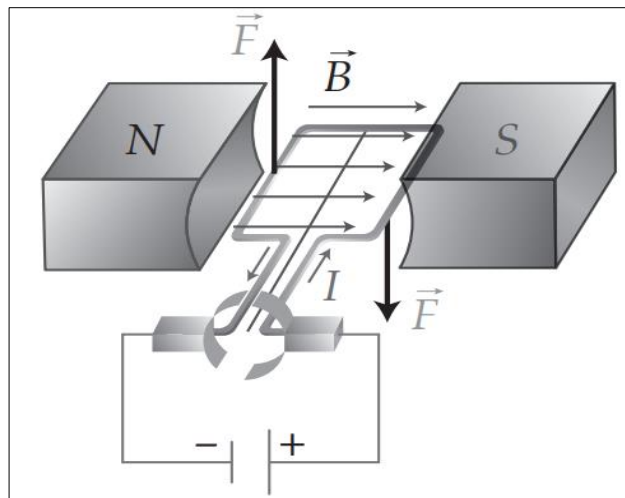


Figura 1.7 Acción del campo magnético en una espira. [27]

Este tipo de actuador hace uso de un motor eléctrico y un reductor para su funcionamiento [28].

1.4.3.1.1 Motor de corriente directa (CD)

Los componentes de este motor son un rotor que se encarga de la parte móvil que brinda fuerza que actúa sobre la carga y estator que da el magnetismo suficiente para inducir la fuerza electromotriz. Como características se tiene que "velocidad de rotación es proporcional al voltaje, mientras que el torque es proporcional a la corriente que circula por su devanado" [27].

1.4.3.1.2 Motor paso a paso

Los motores paso a paso generan la conversión de energía eléctrica a movimiento angular, la conmutación se alcanza debido al uso de un sistema electrónico o tarjeta microcontroladora. Esta tarjeta o sistema provoca pulsos de corrientes modulados que alimentan los devanados internos del motor que son convertidos en desplazamientos angulares discretos en forma directa por los polos del motor. El eje motriz será capaz de avanzar cierto valor en grados, de 1.8° hasta 90° , según la secuencia que se aplique en las entradas de control [27].

1.4.3.1.3 Motor a pasos de imán permanente

El motor a pasos de imán permanente presenta como característica imanes permanentes a diferencia de otros motores que presentan devanados en el rotor. Si el motor no está energizado permite establecer y mantener un par diferente de cero. Dependiendo de la construcción del motor, lo más común es obtener pasos angulares de 7.5° , 11.25° , 15° , 18° , 45° o 90° [28].

1.4.3.2 Cilindros hidráulicos

Los cilindros hidráulicos convierten la energía almacenada de un fluido a presión en trabajo mecánico de movimiento rectilíneo. El tipo de fluido utilizado define características importantes en el funcionamiento del cilindro y es frecuentemente utilizado como fluido el aceite mineral. Como se indica en [27]:

“...la fuerza del movimiento generado por el actuador tiene un bajo valor de compresibilidad, lo que implica un movimiento suave del vástago, que se ve reflejado en una mayor estabilidad frente a cargas estáticas y es capaz de soportar la presión ejercida sobre una superficie, sin aporte de energía (para mover el émbolo de un cilindro sería preciso vaciar este de aceite). Esta misma propiedad del fluido también es la responsable de que en hidráulica se puedan manipular presiones superiores, lo que permite expandir la fuerza de trabajo y que la elevada capacidad de carga y la relación potencia-peso sean destacables, así como sus características de auto lubricación y robustez.”

1.4.3.3 Cilindros neumáticos

Los cilindros neumáticos hacen uso del aire que se considera un tipo de fuerza mecánica. Su principio de funcionamiento se basa en el almacenamiento de aire en un recipiente que esté cerrado y al aplicar una fuerza sobre este aire se genera una compresión forzando las paredes del recipiente, entonces la fuerza que actúa perpendicularmente se aprovecha para generar un desplazamiento lineal de un eje [27].

1.4.3.4 Servomotores

El servomotor consta de un motor con un reductor de velocidad y multiplicador de fuerza, y un circuito de control. Permite un ángulo de giro del eje de 180° que se puede modificar con gran facilidad para obtener un giro libre de 360° . Los componentes del servomotor se indican en la Figura 1.8a, y en la Figura 1.8b el posicionamiento del eje del servomotor.

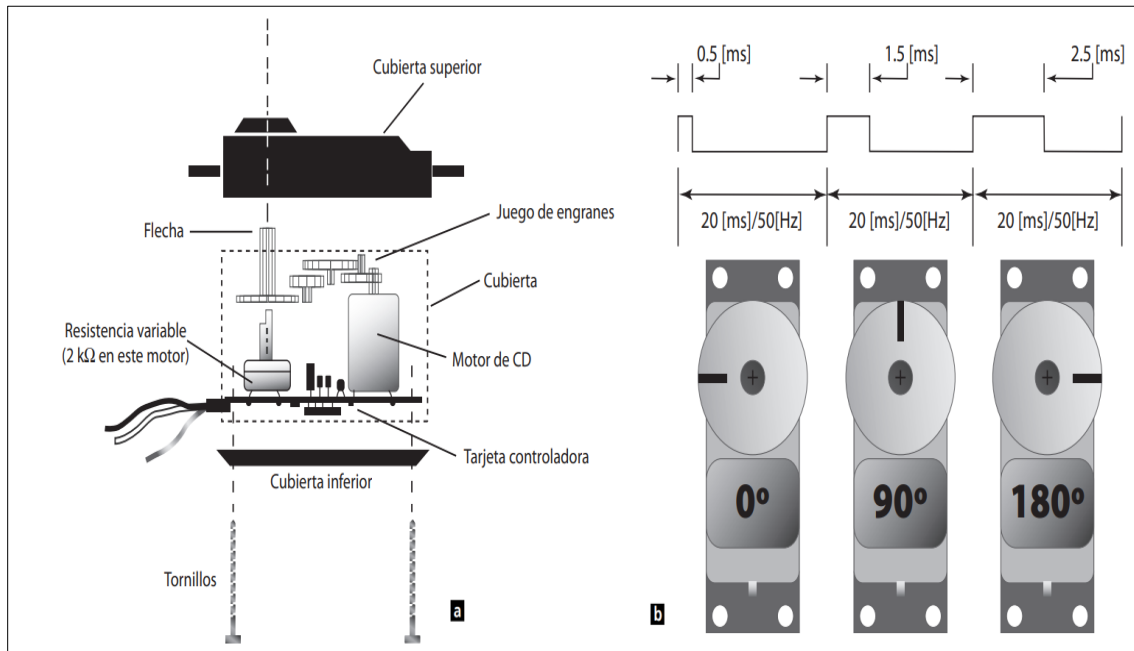


Figura 1.8 Servomotores. a) Componentes. b) Posicionamiento del eje. [27]

1.4.4 RELÉ

El relé es un dispositivo electromagnético que actúa como un interruptor digital el cual es utilizado para controlar valores muy altos que maneja la placa Arduino de voltaje y corriente. Su composición se basa en tres terminales, un terminal común, un terminal normalmente abierto y un terminal normalmente cerrado además de una bobina la cual al estar energizada genera continuidad en el terminal normalmente abierto y en el terminal común [29].

1.4.4.1 Módulo relé de 8 canales

El módulo relé de 8 canales, como se indica en la Figura 1.9, posee 8 relés de alta calidad capaces de manejar cargas de hasta $250\text{V}/10\text{A}$. Los canales presentan aislamiento eléctrico gracias al uso de un optoacoplador y también de un LED que indica el estado del canal. El módulo de relés activa la salida normalmente abierta (NO: Normally Open) al recibir un "0" lógico (0 voltios) y desactiva la salida con un "1" lógico (5 voltios) [30].

El diseño de este módulo relé de 8 canales brinda facilidad para trabajar con tarjetas microcontroladoras como Arduino, Raspberry Pi, ESP8266 (NodeMCU y Wemos), Teensy y PIC; lo que se aprovecha para el manejo de cargas como motores AC, DC, bombillas de luz, luminarias, calentadores de agua, entre otros tipos de actuadores.

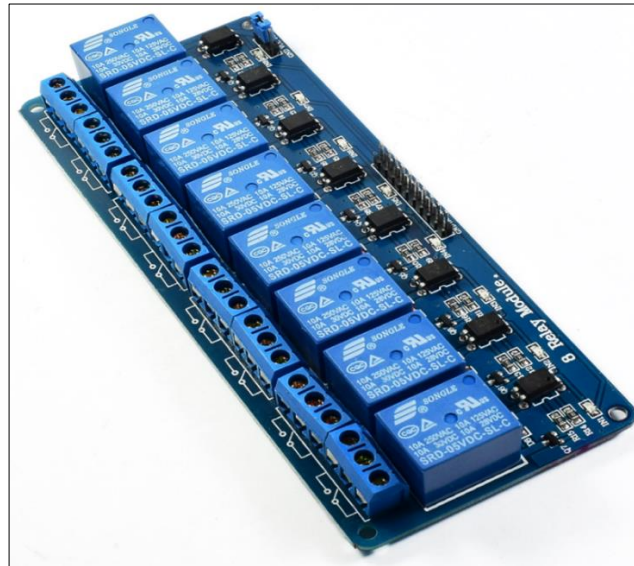


Figura 1.9 Módulo Relé 8CH 5VDC [30].

1.4.5 SISTEMA DE VENTILACIÓN PARA INVERNADERO

El sistema de ventilación es el encargado dentro de un área de regular diferentes factores que se pueden producir debido a los seres vivos y/o maquinaria que se encuentre operando en el espacio confiando. Dentro de un invernadero este sistema de ventilación se encarga fundamentalmente de expulsar el aire viciado por aire fresco sea de manera natural o forzada pues regula la temperatura y humedad que varían dependiendo de los factores climáticos externos, el tipo de material con el que se construye el invernadero, los cultivos y personal existentes dentro del mismo.

1.4.5.1 Sistema de ventilación natural o pasiva para invernadero

La ventilación natural se obtiene mediante sistemas que abren y cierran compuertas en las paredes o en la estructura del invernadero. El ingreso del viento externo y por la diferencia entre la temperatura del aire interior con el aire exterior provoca una diferencia de presión sobre las aberturas. Este proceso permite equilibrar a niveles requeridos el CO₂, controlar la temperatura y humedad del interior, obtener beneficio de la polinización lo cual conlleva a un beneficio también de la fructificación de los cultivos [31]. La Figura 1.10 muestra una foto de un techo de un sistema de ventilación natural para invernadero.



Figura 1.10 Sistema de ventilación natural para invernadero. [32]

1.4.5.2 Sistema de ventilación forzada para invernadero

El sistema de ventilación forzada se genera mediante el uso de ventiladores que aportan aire fresco con una mejor distribución. Existen dos tipos para este sistema: el primer tipo se basa en la admisión de aire y el segundo tipo se basa en la extracción del aire. Los ventiladores implementados aspiran el aire exterior al interior del invernadero, por medio de las aberturas situadas en el lado opuesto de los mismos [31]. La Figura 1.11 presenta una foto de un sistema de ventilación forzada para invernadero.



Figura 1.11 Sistema de ventilación forzada para invernadero. [33]

1.4.6 SISTEMA DE CALEFACCIÓN PARA INVERNADERO

Los sistemas de calefacción hacen uso de generadores de aire caliente para controlar los descensos de temperatura dentro del invernadero. Estos sistemas son de gran utilidad para disponer de equipamiento cuando se presenten heladas lo cual garantiza un adecuado nivel de temperatura en el ambiente interior donde se desarrollan los cultivos [34].

Aquellos sistemas que hacen uso de la combustión interna son generalmente utilizados en los invernaderos y funcionan con gasóleo o gas que permiten el intercambio de calor separando residuos de la combustión llevándolos al ambiente externo. En combinación con un correcto caudal de aire y dardo de alcance de los ventiladores, se genera una correcta

distribución del aire calentado hacia todo el espacio del invernadero. La Figura 1.12 muestra un sistema de calefacción.



Figura 1.12 Sistema de calefacción con generadores de aire caliente. [35]

1.4.7 SISTEMA DE RIEGO POR GOTEO

El riego se puede definir como la “aplicación controlada de agua en áreas de cultivo por medio de sistemas artificiales que abastecen las necesidades de agua de los cultivos” [36]. Los sistemas de riego por goteo están conformados por tuberías de material PVC: PVC-U (policloruro de vinilo no plastificado) de PVC-O (policloruro de vinilo orientado) o de PE (polietileno) [37] que conforman una red que conduce el agua hasta los cultivos y se riega a través de emisores los cuales brindan en cantidades controladas en forma de gota el líquido. A este sistema también se lo conoce como de alta frecuencia (en la agricultura) ya que dependiendo del tipo de suelo utilizada y las necesidades del cultivo se puede regar desde una a dos veces por día, todos los días o algunos días [38].

Los componentes de un equipo de riego presurizado se pueden visualizar en la Figura 1.13 y se detallan a continuación:

- Cabezal principal.
- Tuberías de conducción principales.
- Cabezales de campo.
- Tuberías terciarias.
- Laterales de riego con emisores.

- Subunidad de riego: es el área que se riega con una válvula o cabezal de campo.
- Unidad de riego: es la superficie que se riega simultáneamente tomando un conjunto de subunidades de riego.
- Operación de riego: es la superficie que se riega a la vez en el conjunto de unidades de riego.

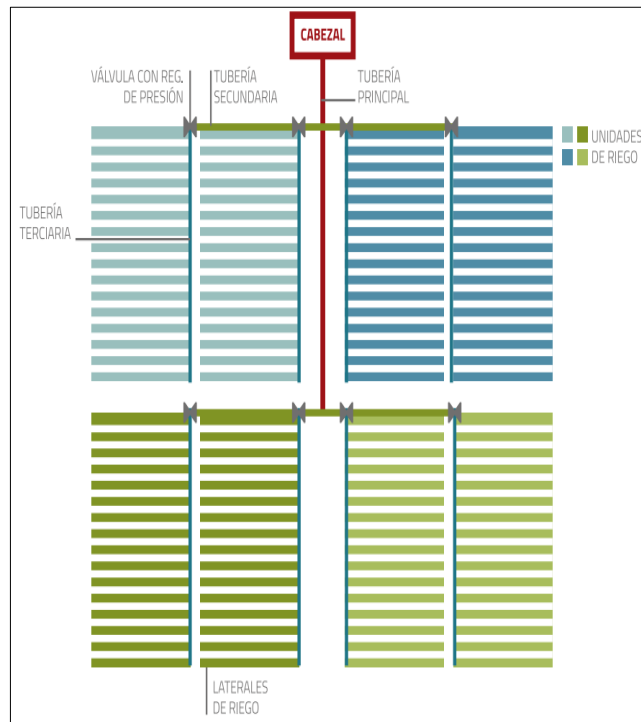


Figura 1.13 Esquema de un sistema de riego presurizado. [38]

2 METODOLOGÍA

En este capítulo se presenta a detalle el proceso realizado para identificar el problema a resolver, los objetivos a cumplir, los métodos realizados, los elementos de hardware y software utilizados, su aplicación y el desarrollo de éstos para llegar a generar el código necesario que cumple con el objetivo principal del presente trabajo de integración curricular.

2.1 DETERMINACIÓN DEL PROBLEMA

Los invernaderos necesitan de automatización para mejorar el control y productividad de los cultivos. Un apartado importante es el uso de diferentes actuadores para cumplir funciones específicas, en este caso, las de regulación de la temperatura a través de la implementación de sistemas de ventilación forzada.

Mediante el uso de sistemas de calefacción se mantiene la temperatura adecuada en las heladas que se pueden presentar porque llegan a provocar una gran disminución de la temperatura, y gracias a este sistema se logra contrarrestar y mantener una temperatura óptima para el desarrollo de los cultivos. De igual forma ambos sistemas normalizan la humedad relativa que se produce en base a la transpiración de la planta evitando así la condensación de este vapor de agua en las paredes del invernadero lo cual genera una lluvia artificial que provoca una sobrehidratación.

El sistema de riego es importante para la correcta humidificación del suelo permitiendo a los cultivos un buen crecimiento y obtención de nutrientes. Para ello, se aplica un actuador que brinda un flujo de agua a la red de tuberías o mangueras según los requerimientos del cultivo las veces que sean necesarias por días o semanas en un intervalo de tiempo determinado. Finalmente, el sistema de iluminación artificial alarga el fotoperiodo de la planta a 16 horas con un descanso de 8 horas para un mejor desarrollo y germinación.

2.2 SELECCIÓN Y APLICACIÓN DE LOS COMPONENTES DE HARDWARE

La elección de cada componente físico necesario para el desarrollo del proyecto se realiza en base a los objetivos planteados. Se explica a continuación cada elemento a implementar en el sistema, así como su funcionamiento, aplicación y beneficios de los cuales se aprovecha para el desarrollo del código.

2.2.1 PLATAFORMA DE DESARROLLO

La plataforma de desarrollo a escoger es la placa Arduino YÚN. La elección se basa en la ventaja que presenta este dispositivo electrónico ya que está diseñado para interconectar el mundo "analógico" (con sensores y actuadores) con el mundo en red (a través de cables o de forma inalámbrica). Para el campo del Internet de las Cosas (IoT) representa un gran avance. El S.O. Linux implementado en la plataforma permite reducir el tiempo de creación de proyectos o prototipos por parte de los usuarios considerablemente esto gracias a los scripts de Shell y Python que son comúnmente usados en sistemas Linux [39].

Para el componente de los actuadores es importante tener en cuenta las interfaces de entrada y salida que presenta la placa para la lectura de los datos (Input Interfaces) y la ejecución de las órdenes enviadas a través de señales eléctricas (Output Interfaces) a cada uno de los actuadores de los diferentes sistemas para la ejecución del respectivo proceso.

Arduino YÚN presenta 20 pines de Entrada y Salida digitales. Cada uno de estos pines se puede usar como entrada o salida mediante el uso de la función `pinMode()`. Los pines trabajan con 5 voltios, pueden recibir o generar una corriente máxima de 40 mA y contienen resistencias pull-up internamente de 20-50 kΩ. Ciertos pines son especiales para propósitos específicos destinados por el fabricante, como se indica en [21]:

- 0 (RX) y 1 (TX): para recepción y transmisión de datos.
- 2 (SDA) y 3 (SCL): comunicación TWI (*Two Wire Interface*).
- Interrupciones externas: 3 (interrupción 0), 2 (interrupción 1), 0 (interrupción 2), 1 (interrupción 3) y 7 (interrupción 4).
- PWM: 3, 5, 6, 9, 10, 11 y 13. Proporciona una salida PWM de 8 bits con la función `analogWrite()`.
- SPI: en el encabezado ICSP. Estos pines admiten la comunicación SPI utilizando la biblioteca SPI.
- LED: 13.
- Entradas analógicas: A0 - A5, A6 - A11 (en los pines digitales 4, 6, 8, 9, 10 y 12).

La Figura 2.1 presenta la distribución de las Interfaces de Entrada y Salida, y en la Figura 2.2 se presenta la asignación de pines que se basa en el microcontrolador ATMEGA32U4, pero con una nomenclatura propia.

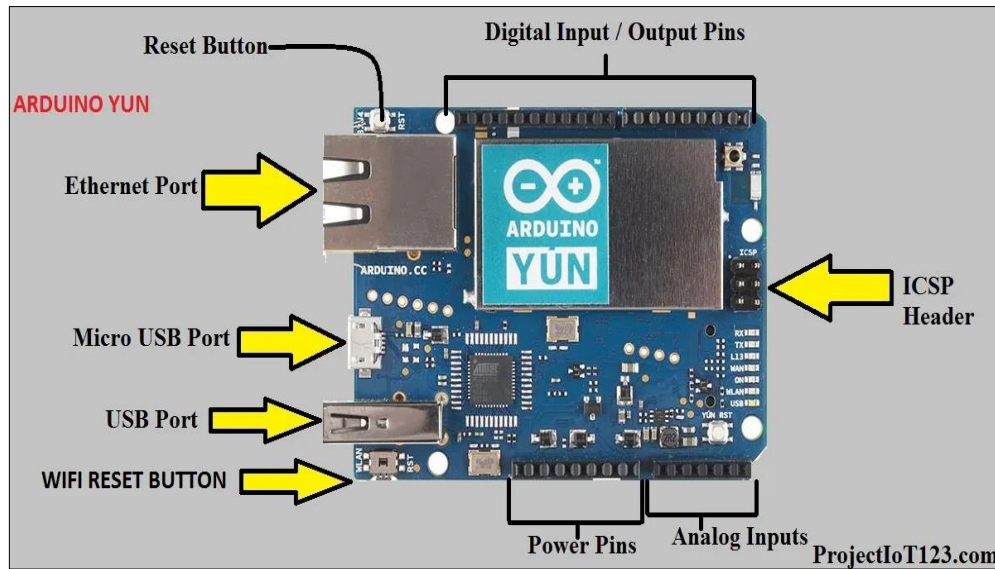


Figura 2.1 Interfaces de Entrada/Salida del Arduino YÚN. [40]

La Tabla 2.1 detalla la distribución de los pines usados para la lectura de los datos y el envío de señales hacia los actuadores. La adquisición de datos del ambiente mediante el uso de sensores no se contempla en el alcance del presente trabajo técnico, sin embargo se designa los pines para obtener las lecturas de las magnitudes físicas del entorno, y también para el envío de las señales en base a los requerimientos de los cultivos para corregir, controlar y regular el sistema de soporte vital.

Tabla 2.1 Distribución de los pines del Arduino YÚN para lectura de datos y envío de señales hacia los actuadores.

Número de Pin	Propósito	Detalle
Pin A0 – A2	Adquisición de datos de temperatura	Pin A0 lectura temperatura interna y humedad relativa Pin A1 lectura CO2 Pin A2 lectura humedad del suelo
Pin 2-8	Envío de señal para funcionamiento de actuadores	Pin 2 interrupción Pin 3 actuador ventilador 1 Pin 4 actuador ventilador 2 Pin 5 actuador riego Pin 6 actuador calefactor Pin 7 actuador tiras led Pin 8 señal para configurar RGB en tiras LED
Pin SCL-SDA	Encendido y visualización de información en LCD, adquisición de tiempo mediante RTC DS3231	SCL pulsos de reloj que sincronizan el sistema SDA línea por la que se mueven los datos entre los dispositivos

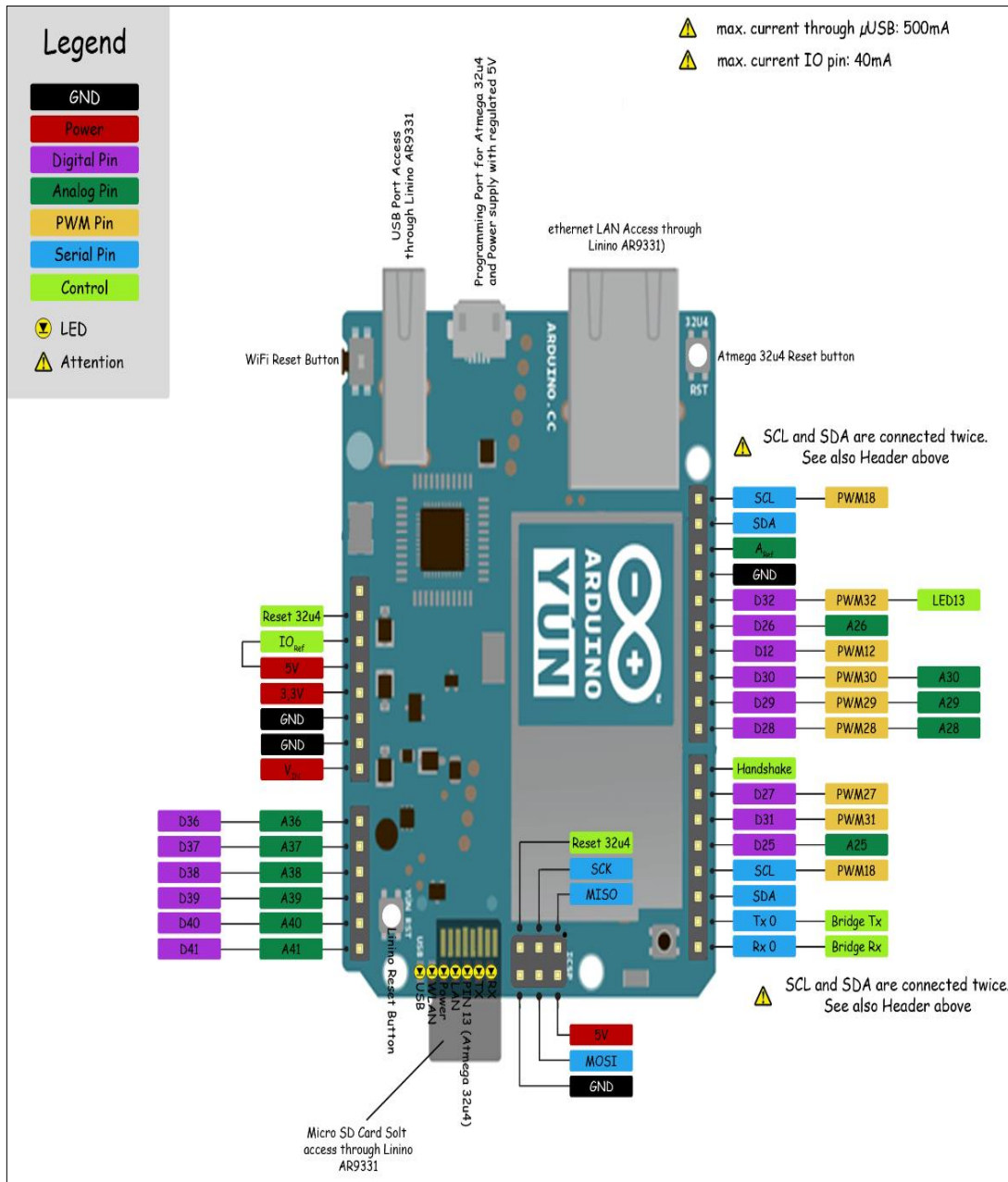


Figura 2.2 Asignación de Pines en el Arduino YUN. [41]

2.2.2 DISPOSITIVOS MECÁNICOS

Esta sección detalla los dispositivos mecánicos utilizados para el funcionamiento del prototipo.

2.2.2.1 ACTUADOR PARA EL SISTEMA DE VENTILACIÓN

El actuador para la ventilación es una máquina de fluido, es decir, se intercambia la energía con el fluido que los atraviesan en este caso el aire del exterior que ingresa al ambiente interno del invernadero. Se presenta un ventilador utilizado generalmente en invernaderos a escala real, y a la par, un ventilador para el prototipo implementado.

En invernaderos con dimensiones considerables se aplican ventiladores de alta potencia y poco ruido. Estos ventiladores presentan características beneficiosas para eliminar el aire viciado o húmedo, olores, entre otros elementos que se encuentran en el espacio interior los cuales no se consideran beneficiosos. En la Figura 2.3 se observa un ventilador convencional cuyas dimensiones varían según el modelo las cuales superan los 50 centímetros hasta 2 o 3 metros.



Figura 2.3 Sistemas de ventilación forzada para la expulsión del aire [42].

La Tabla 2.2 presenta algunas especificaciones técnicas de este tipo de actuadores.

Tabla 2.2. Especificaciones técnicas del ventilador extractor de aire para invernadero MULTIFAN de CONAIRE [43].

Especificación técnica	Descripción
Material	ABS, PP
Velocidad de rotación	2550r / min
Volumen de aire	420-450m ³ / h
Ruido	33dB (A)
Presión	245-305 (Pa)
Frecuencia	60 / 80Hz
Voltaje	220-240 V
Poder	80W
Tipo de enchufe	enchufe del Reino Unido / enchufe de la UE
Diámetro de la cinta	F150 mm / 6 pulgadas
Grado impermeable	IPX4
Peso	3000g / 105,8 oz
Superficie adecuada	18-36m ²

Para el invernadero a escala se hace uso de dos ventiladores de la marca EVL, modelo VN-551, operación a 12 voltios DC con una corriente de 380 mA. La Figura 2.4 presenta las dimensiones del ventilador.



Figura 2.4 Ventilador EVL, Modelo VN-551, 12 VDC, 380 mA. [44]

2.2.2.2 ACTUADOR PARA EL SISTEMA DE CALEFACCIÓN

El calefactor es de tipo portátil que se alimenta con 110 V, consume una potencia de 350 W, sirve para aumentar la temperatura en un radio de 3 metros. Presenta una pantalla LED digital para regular la temperatura en un rango de 15°C a 32°C. La Figura 2.5 indica las dimensiones del calefactor.



Figura 2.5 Calefactor portátil Handy Heater NFJ 03. [45]

2.2.2.3 ACTUADOR PARA EL SISTEMA DE RIEGO POR GOTEO

El actuador de sistema de riego manipula un contactor a través del relé que maneja el Arduino para activar la electroválvula misma que succiona el fluido y lo dirige a la red de distribución de goteo. La Figura 2.6 muestra el contactor electromagnético que se encarga de abrir o cerrar circuitos para la automatización segura del sistema. Además, este dispositivo puede trabajar en vacío o con carga.



Figura 2.6 Contactor Electromagnético [46].

La red de tuberías o mangueras (Figura 2.7) es de material tipo termoplástico obtenido del cloruro de vinilo, que contiene cada cierta distancia agujeros. Este tipo de riego se puede combinar con rociadores (Figura 2.8) para brindar mayor flujo de agua los cuales pueden ser regulables.



Figura 2.7 Manguera utilizada en sistemas de riego por goteo que presenta un orificio cada cierta distancia para un riego tipo gota.



Figura 2.8 Rociador para sistemas de riego por goteo.

En el prototipo implementado se instala una mini bomba de agua sumergible que requiere para su operación de 2,5 V – 6 V; con una elevación máxima de 40 cm – 110 cm. Esta mini bomba proporciona un caudal de 80 l/h – 120 l/h. La Figura 2.9 muestra la mini bomba de agua utilizada.



Figura 2.9 Mini bomba de agua sumergible. [47]

2.2.2.4 ACTUADOR PARA EL SISTEMA DE ILUMINACIÓN

Este sistema funciona con un arreglo de 2 tiras de 19 luces LED cada una las cuales combinan luz cálida (LED color rojo) y luz fría (LED color azul). La mejor combinación se

obtiene al encender un LED con color azul cada tres LEDs de color rojo. En la Figura 2.10 se muestran las dos tiras con los LEDs empotradas en dos canaletas.

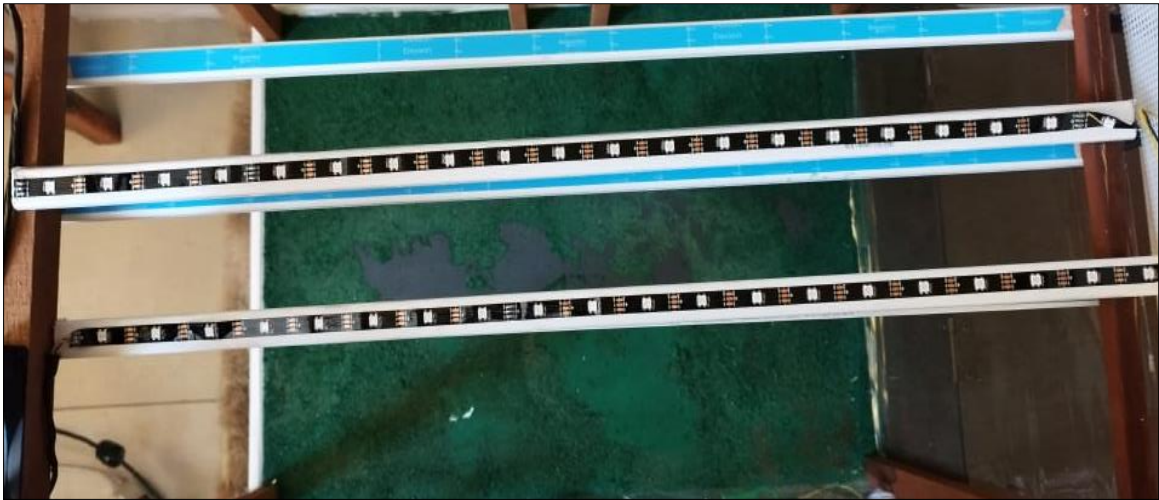


Figura 2.10 Arreglo de 2 x 19 LEDs.

2.3 DESARROLLO Y APLICACIÓN DE LOS COMPONENTES DE SOFTWARE

Esta sección detalla el código desarrollado en el Entorno de Desarrollo integrado (IDE) y almacenado en la memoria SRAM del Arduino.

2.3.1 FLUJOGRAMA DE FUNCIONAMIENTO

La **¡Error! No se encuentra el origen de la referencia.** Figura 2.1 muestra el flujograma de funcionamiento. Se describe cómo el Arduino YÚN se encarga del manejo mediante software de todas las acciones necesarias a realizar para mantener en niveles estables el ecosistema donde se desarrollan los cultivos del invernadero.

Inicio

Esta parte indica el arranque del sistema. Se inicializan las variables a utilizarse en las rutinas implementadas, así como también se verifica que el módulo RTC DS3231 esté activo.

Arranque de variables y puertos de comunicación

Cuando se enciende el Arduino YÚN, éste internamente arranca los procesos asignados en el código cargado: Se designan variables, pines a utilizar para lectura o escritura y la comunicación con el reloj de tiempo real mediante la función `RTC.begin()`. Esto último se utiliza con la finalidad de activar el sistema completo en el horario de 2 am a 6 pm lo cual

se justifica en base a los siguientes parámetros: economía, ligado al consumo de energía debido al funcionamiento continuo del sistema; tiempo de suministro de energía solar, entre las 7 am y 6 pm, 12 horas aproximadamente donde se desarrollan los cultivos debido a que se aporta la energía necesaria para efectuar la fotosíntesis, produciendo materia orgánica para su crecimiento y desarrollo [48]. Para los intervalos de 2 a 6 am ocurre un descenso de temperatura lo que implica que pueden bajar en extremo estos niveles [49], aumentado los horizontes de la humedad relativa, esto se corrige haciendo uso del sistema de calefacción y ventilación de ser necesario. El sistema de iluminación artificial funciona continuamente de 2 a 6 am para aumentar el fotoperiodo de los cultivos para un mejor desarrollo y germinación.

Verificación del funcionamiento del Reloj a Tiempo Real

El uso de un RTC, específicamente del modelo RTC DS3231 es fundamental para poner en marcha el sistema. Se verifica el funcionamiento del módulo RTC de manera repetitiva hasta que este módulo esté funcionando correctamente.

Lectura de la hora actual

Despliega en la pantalla LCD la fecha y hora actuales.

Interrupción

Se genera una interrupción en el PIN2 mediante un cambio de estado de tipo RISING (pasar de un valor bajo a un valor alto) al pulsar el botón correspondiente. Al ser una interrupción, ésta puede modificar los valores del sistema de soporte vital en microsegundos en cualquier momento gracias a que se ha declarado inicialmente que son variables de tipo VOLATILE. Con esta declaración el compilador procede a cargar a la variable en la RAM y no en un registro de almacenamiento.

La lectura de datos es primordial, si bien se hace uso de tres sensores para la lectura de datos de cada una de las magnitudes físicas del entorno, el sistema funciona independiente de éstos. Una vez que se realiza la lectura de los datos, estos valores se someten a la siguiente pregunta: ¿El valor de temperatura, humedad relativa y CO2 están dentro del rango establecido? Si los valores leídos se encuentran fuera de los rangos 15°C y 28°C, 50% y 90%, y 300 ppm a 1200 ppm, respectivamente [50], [51], se ejecutan las correspondientes acciones declaradas en cada una de las rutinas dentro del código.

Los niveles de humedad del suelo dependen de la cantidad de riego, el cultivo y tipo de suelo. Para este último, se tiene en cuenta que "los suelos poseen diferente capacidad para almacenar agua dependiendo de su textura y estructura. El límite superior de

almacenamiento de agua se denomina capacidad de campo (CC), mientras que el límite inferior se denomina punto de marchitamiento permanente (PMP)" [52]. A pesar de esto, la humedad cambia en menores ocasiones con respecto a las otras variables ambientales que necesitan ser monitoreadas constantemente. Por esta razón el sistema de riego se activa durante las mañanas de 6 am a 10 am que es la mejor opción para realizar el riego gracias a que el agua alcanza las raíces de los cultivos evitando la evaporación debido a la radiación solar, y por las tardes de 3 pm a 5 pm cuando la radiación solar es tenue lo cual es provechoso para evitar acumulación de agua que se produciría si se riegan los cultivos en horas donde el sol no se encuentra presente [53].

Lectura y presentación de datos

En este apartado la función procede a leer las variables del ambiente y mostrarlas en pantalla para que se pueda observar el estado del invernadero.

Horario de funcionamiento de 2 am 6 am

En este horario funciona de manera interrumpida el sistema de iluminación artificial con el encendido del arreglo de LEDs. Posteriormente, el sistema de ventilación y calefacción están activos para regular la humedad relativa. Si esto ocurre, se enciende el ventilador hasta alcanzar un nivel de humedad menor al valor máximo del 90% asignado. Luego, puede que baje la temperatura por la acción de los ventiladores. Si es el caso, se enciende la calefacción para alcanzar un nivel adecuado. La calefacción no es programable, y por defecto cada vez que está en funcionamiento brinda aire con temperatura de 25°C hasta un radio de 3 metros.

Si la humedad relativa está en los valores adecuados, se tiene el sistema de calefacción también activo por seguridad a cambios bruscos de temperatura.

Horario de funcionamiento de 6 am a 6 pm

Dentro de este intervalo de tiempo el sistema de ventilación y calefacción están activos. El primer caso de análisis corresponde al cambio de la temperatura superando el valor máximo establecido y de los niveles de CO₂ al descender por debajo del valor mínimo. Se activa la ventilación para ingresar aire del ambiente que contiene entre 300 ppm y 400 ppm. Si no es suficiente se hace uso de un proceso manual para inyectar CO₂ quemando papel reciclado para regular dichos niveles, esto también se realiza en los siguientes casos.

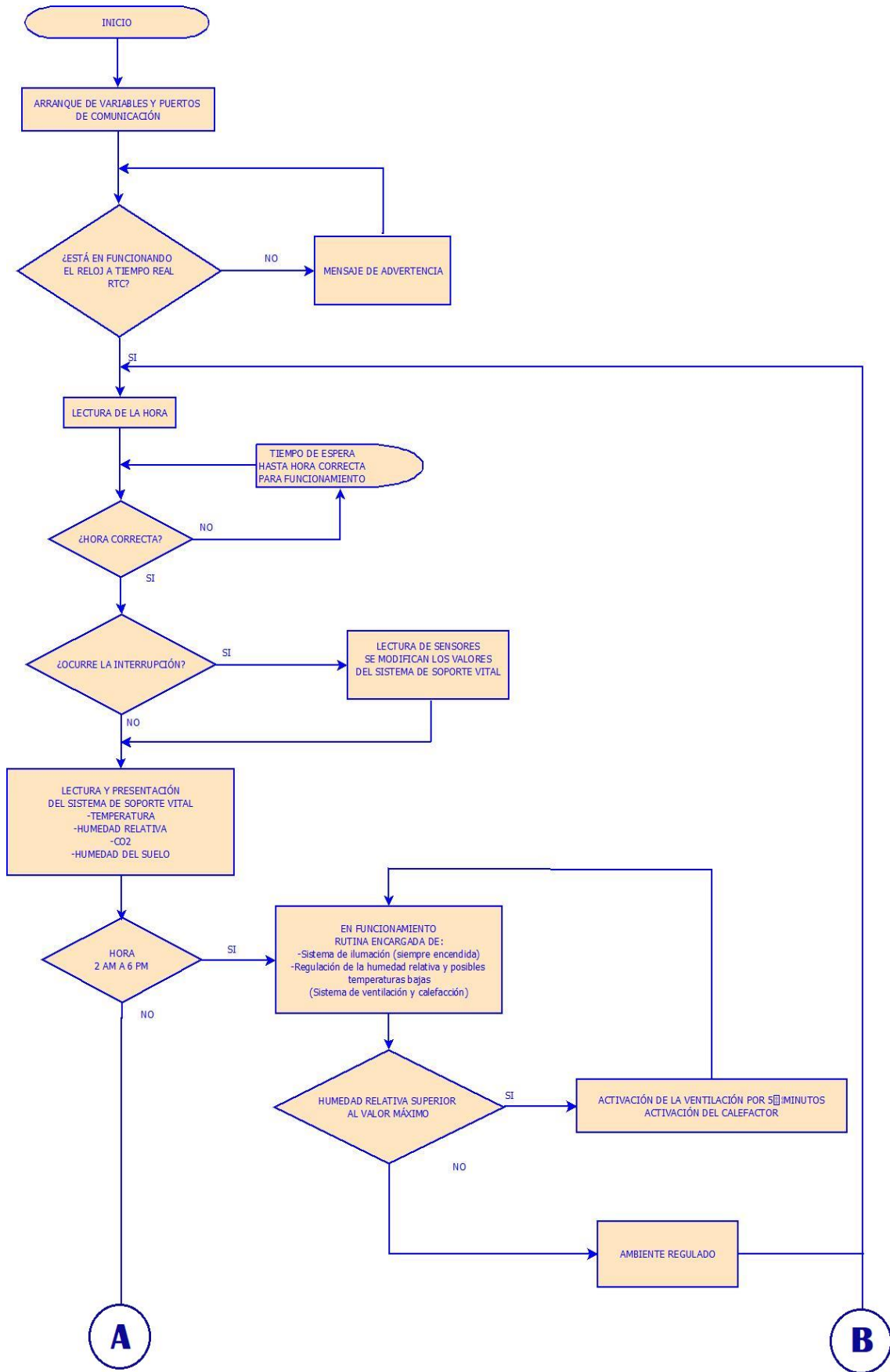
El segundo caso se presenta cuando la temperatura únicamente aumenta sobre el nivel máximo permitido activando los ventiladores para bajar la temperatura. El tercer y último caso inicia al obtener valores de CO₂ por debajo del rango mínimo.

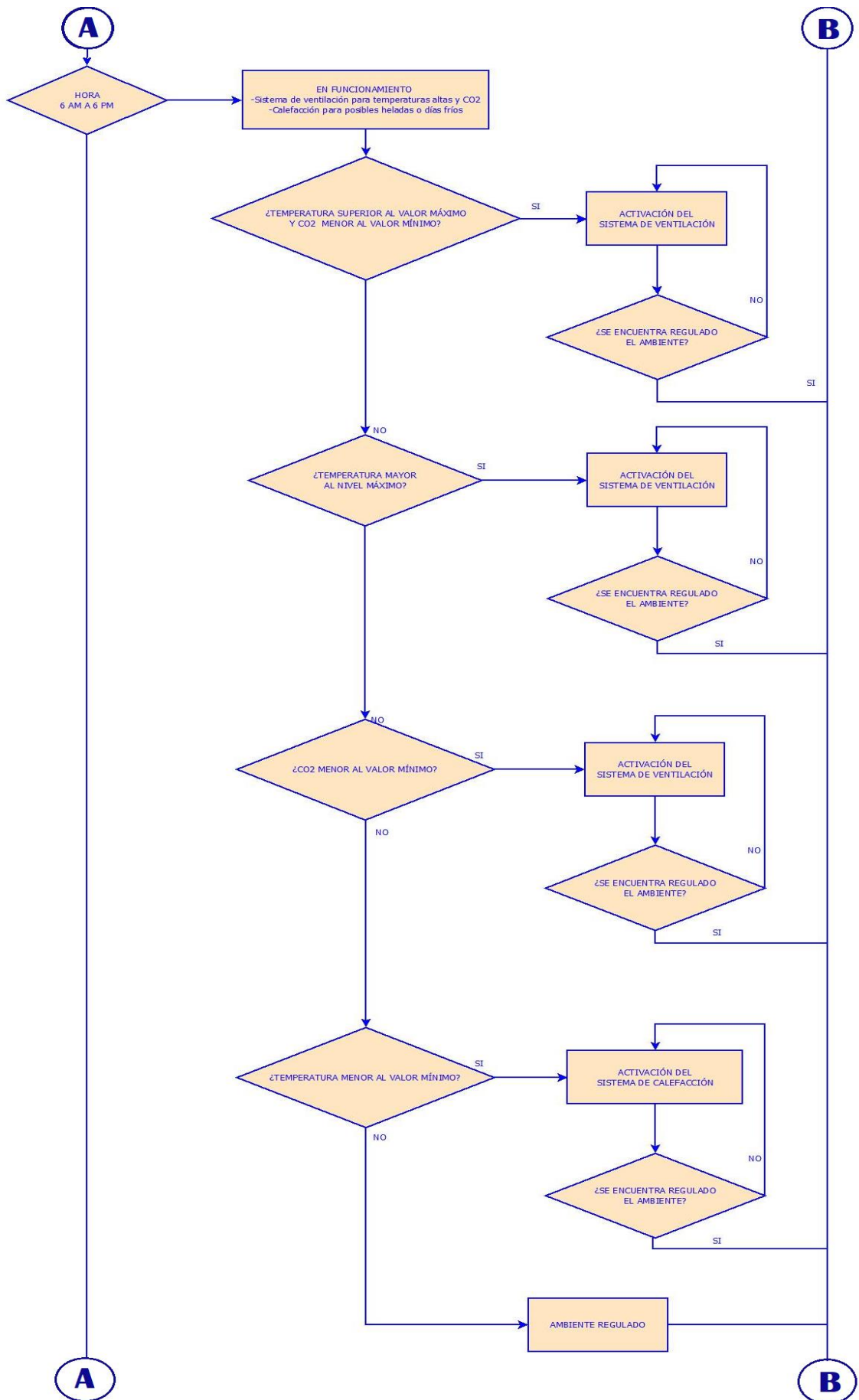
Horario de funcionamiento de 6 am a 10 am y de 3 pm a 5 pm

Si bien este horario está dentro del horario anteriormente detallado, éste no depende de las otras variables ambientales. El sistema de riego proporciona la cantidad de agua necesaria hasta que se encuentre en el rango del 10% al 50%.

Tiempo para nueva lectura

Se ha completado el ciclo de lectura y manejo de actuadores. Llegado a este punto se establece un tiempo de 5 minutos para una nueva lectura, es decir, todo el proceso se repite continuamente para el monitoreo constante de los cultivos bajo techo.





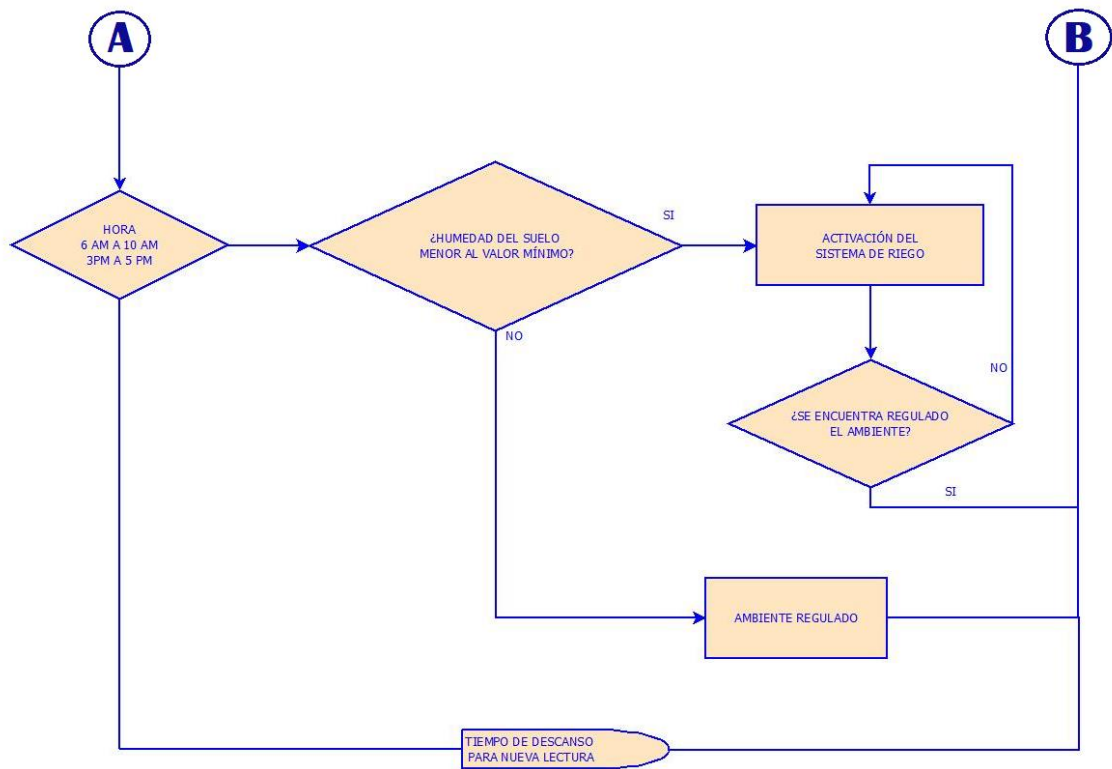


Figura 2.1. Flujograma de funcionamiento del sistema.

2.3.2 CONEXIÓN DEL SISTEMA ARDUINO YÚN Y ACTUADORES

La Figura 2.11 describe la conexión del Arduino YÚN y actuadores, es decir, describe el sistema completo desarrollado.

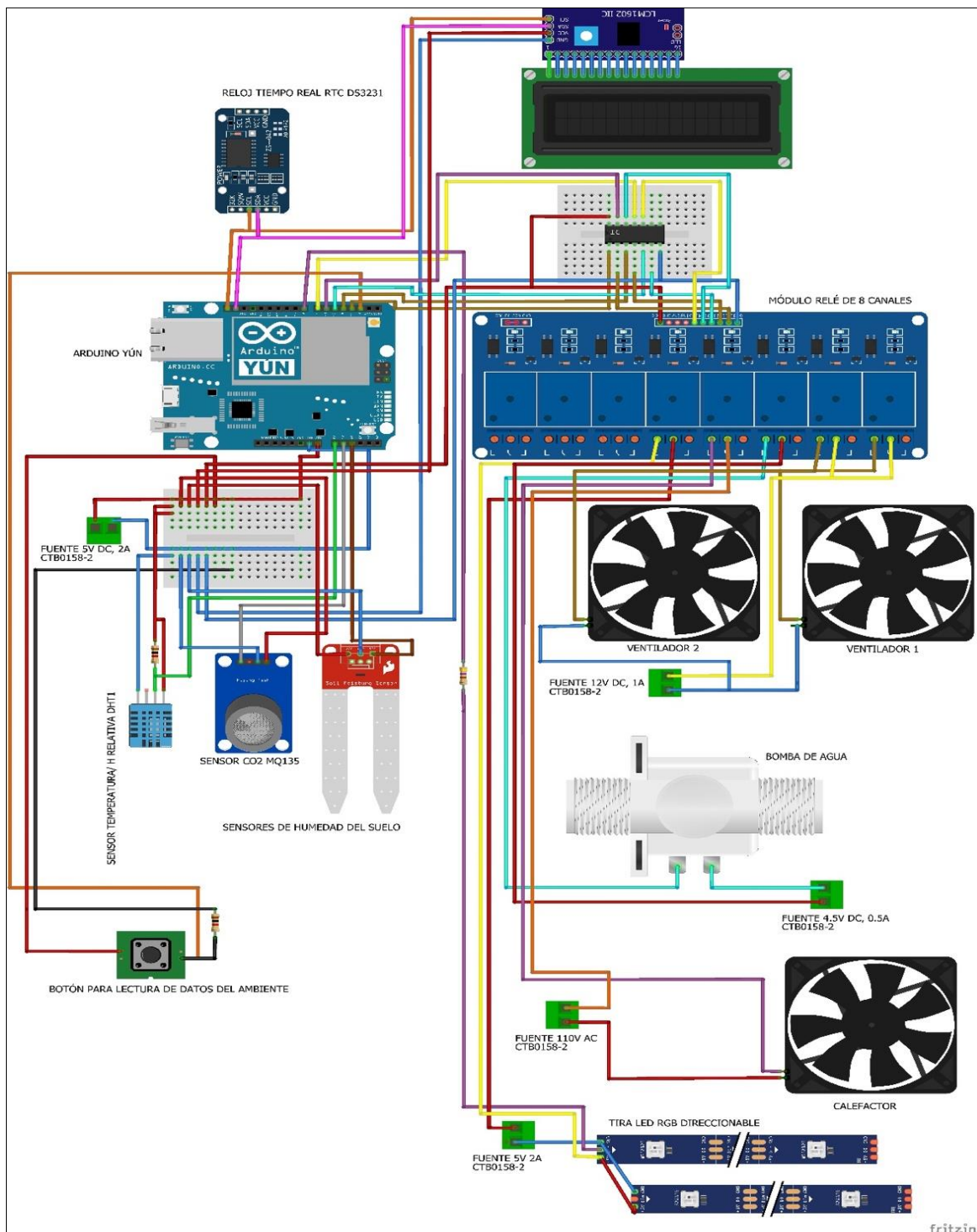


Figura 2.11 Conexión del sistema Arduino YÚN y actuadores.

2.3.3 LIBRERÍAS

La Tabla 2.3 contiene las librerías utilizadas dentro del código.

Tabla 2.3 Librerías utilizadas dentro del código generado para el manejo de actuadores haciendo uso de la plataforma Arduino YÚN. [54]–[58]

Librerías	Descripción
#include <DHT.h>	Librería DHT para el manejo de sensor DHT11, adquisición de los datos de temperatura y humedad relativa
#include <DHT_U.h>	Librería de sensores unificados de temperatura y humedad DHT
#include <Wire.h>	Librería que permite la comunicación con dispositivos I2C/TWI
#include <RTClib.h>	Librería para el funcionamiento del reloj en tiempo real
#include <LiquidCrystal_I2C.h>	Librería que permite manejar pantallas I2C con funciones similares a la librería LiquidCrystal
#include <Adafruit_NeoPixel.h>	librería para controlar tiras y píxeles LED basados en un solo cable

2.3.4 VARIABLES Y OBJETOS

La Tabla 2.4 presenta cada una de las variables y objetos utilizados dentro del código.

Tabla 2.4 Variables y objetos utilizados dentro del código generado para el manejo de actuadores haciendo uso de la plataforma Arduino YÚN.

Sistema/Función	Variable	Tipo	Descripción
Sensor de Temperatura y Humedad Relativa DHT11	tempLec	Flotante (float), 0.0f	Adquiere el valor de la temperatura interna a través del sensor, por defecto en °C
	tempMin	Flotante (float), 0.0f	Valor temperatura mínima en °C

	tempMax	Flotante (float), 0.0f	Valor temperatura máxima en °C
	tempP	Volatile Flotante (float), 0.0f	Valor estándar asignado inicialmente para pruebas de funcionamiento (24 °C), adquiere el valor del sensor una vez se activa la rutina
	humRe	Flotante (float), 0.0f	Adquiere el valor de la humedad relativa interna a través del sensor, por defecto en %
	humRelMin	Flotante (float), 0.0f	Valor mínimo para la humedad relativa interna en porcentaje
	humRelMax	Flotante (float), 0.0f	Valor máximo para la humedad relativa interna en porcentaje
	humeRelP	Volatile Flotante (float), 0.0f	Valor estándar asignado inicialmente para pruebas de funcionamiento (70%), adquiere el valor del sensor una vez se activa la rutina
Sensor de Gas para lectura de CO2 MQ135	CO2	Entero (int)	Adquiere el valor del dióxido de carbono del sensor, por defecto en ppm
	CO2Min	Entero (int)	Valor mínimo de dióxido de carbono en el ambiente en ppm
	CO2Max	Entero (int)	Valor máximo de dióxido de carbono en el ambiente en ppm

	CO2P	Volatile Entero (int)	Valor estándar asignado inicialmente para pruebas de funcionamiento (350 ppm), adquiere el valor del sensor una vez se activa la rutina
Sensor para la Humedad del Suelo	humSue	Flotante (float), 0.0f	Adquiere el de la humedad del suelo del sensor, en Voltaje, con un valor de 0 significa muy húmedo y 1023 es seco
	humSuePor	Flotante (float), 0.0f	Adquiere el valor del sensor de humedad del suelo en porcentaje
	humSuePorMin	Flotante (float), 0.0f	Valor mínimo para la humedad del suelo en porcentaje
	humSuePorMax	Flotante (float), 0.0f	Valor máximo para la humedad del suelo en porcentaje
	humeSueP	Volatile Flotante (float), 0.0f	Valor estándar asignado inicialmente para pruebas de funcionamiento (35%), adquiere el valor del sensor una vez se activa la rutina
Sistema de ventilación	ventilador1	Entero (int)	Pin del actuador que controla el ventilador 1
	ventilador2	Entero (int)	Pin del actuador que controla el ventilador 2
Sistema de riego por goteo (bomba de agua)	bomba_agua	Entero (int)	Pin del actuador que controla la bomba de agua para el sistema de riego por goteo

Sistema de calefacción	calefactor	Entero (int)	Pin del actuador que controla la calefacción
Sistema de iluminación y su respectiva función	int luz = 7; //	Entero (int)	Pin del actuador que controla la tira led para el sistema de iluminación
	PINLEDS 8	Constante (#define)	Pin que controla la secuencia de encendido de la tira LED
	TOTALLEDS 38	Constante (#define)	Total, de LEDs en la tira
	j	Entero (int)	Utilizada para el ciclo for de encendido de LEDs
	azul	Entero (int)	Enciende cada 3 LEDs rojos un LED azul
	residuo	Entero (int)	Almacena el residuo entre j y azul para encender aquellos que cumplan la condición if
Almacenamiento de datos relacionados a fecha y hora	ano	Cadena de caracteres (String)	Almacena el año
	mes	Cadena de caracteres (String)	Almacena el mes
	dia	Cadena de caracteres (String)	Almacena el día
	horas	Cadena de caracteres (String)	Almacena la hora
	minutos	Cadena de caracteres (String)	Almacena los minutos
	segundos	Cadena de caracteres (String)	Almacena los segundos
	fechaActual	Cadena de caracteres (String)	Almacena la fecha actual
	horaActual	Cadena de caracteres (String)	Almacena la hora actual
horasInt	Entero (int)	Almacena el valor de la hora convirtiendo de la	

			cadena de caracteres a un valor entero numérico
Variables utilizadas en la función nueva_lectura	seg	Entero (int)	Cantidad de segundos para una nueva lectura
	aux	Entero (int)	Almacena la variable "seg" como respaldo para el contador regresivo
	t_s	Entero (int)	Almacena la transformación de la variable "seg" en segundos
	t_m	Entero (int)	Almacena la transformación de la variable "seg" en minutos
	t_h	Entero (int)	Almacena la transformación de seg en horas
	i	Entero (int)	Almacena utilizada para el ciclo for
Variables auxiliares	vere	Entero (int)	Permite verificar si el RTC DS3231 está o no conectado
	nt1	Flotante (float), 0.0f	Valor de temperatura mínima de guarda para controlar el ambiente en la funcion temp_co2
	nt2	Flotante (float), 0.0f	Valor de temperatura máxima de guarda para controlar el ambiente en la funcion temp_co2
	nhr1	Flotante (float), 0.0f	Valor de humedad relativa mínima de guarda para controlar el ambiente en la funcion hum_relativa

	nhr2	Flotante (float), 0.0f	Valor de humedad relativa máxima de guarda para controlar el ambiente en la función hum_relativa
	nco2_1	Flotante (float), 0.0f	Valor de co2 mínimo de guarda para controlar el ambiente en la función temp_co2
	nhsue1	Flotante (float), 0.0f	Valor de humedad del suelo mínimo de guarda para controlar la cantidad de líquido en la función riego
	nhsue2	Flotante (float), 0.0f	Valor de humedad del suelo máximo de guarda para controlar la cantidad de líquido en la función riego
	t_restante	Entero (int)	Almacena el valor de tiempo restante cuando se activa la ventilación en la función hum_relativa()
	incremento	Entero (int)	Tiempo en segundos utilizada en la función hum_relativa() para el contador regresivo
	tiempo_vent_hr	Entero (int)	Tiempo en segundos utilizada para la ventilación que despeja el aire húmedo en la función hum_relativa()
Objeto para funcionamiento de RTC	RTC	RTC_DS3231	Creación de objeto RTC DS3231 para el funcionamiento del reloj

Objeto para adquisición de fecha y hora	now	DateTime	Creación del objeto now para obtener valores de fecha y hora
Objeto para adquisición de temperatura y humedad relativa	dht	DHT	Creación de objeto DHT dht para el funcionamiento del sensor DHT11
Objeto para el manejo de LCD I2C	lcd	LiquidCrystal_I2C	Creación de objeto LiquidCrystal_I2C LCD para el funcionamiento del LCD
Objeto para el manejo de tiras LED direccionables	pixels	Adafruit_NeoPixel	Almacena estado y funciones para interactuar con cada LED

2.3.5 FUNCIONES

La Tabla 2.5 contiene cada una de las funciones utilizadas dentro del código.

Tabla 2.5. Funciones utilizadas dentro del código generado para el manejo de actuadores haciendo uso de la plataforma Arduino YÚN. [59]–[80]

Funciones	Descripción
Void	Permite declarar funciones sin devolución desde la función con la que se llamó
setup()	Permite inicializar variables, modos pin, comenzar a usar bibliotecas, etc., las cuales son ejecutadas una sola vez
begin()	Permite iniciar librerías
Wire.begin()	Inicializa la librería Wire y conecta Arduino al bus
RTC.begin()	Inicializa el RTC interno
RTC.begin(DateTime(__DATE__, __TIME__))	Permite establecer la hora en el reloj manualmente o cargar la hora del PC

RTC.now()	Se crea el objeto con la fecha y hora actual
now.year()	Se obtiene únicamente el año
now.month()	Se obtiene únicamente el mes
now.day()	Se obtiene únicamente el día
now.hour()	Se obtiene únicamente la hora
now.minute()	Se obtiene únicamente los minutos
now.second()	Se obtiene únicamente los segundos
lcd.begin(columnas,filas);	Inicia la interfaz de la pantalla LCD y especifica las dimensiones (columnas y filas) de la pantalla
lcd.backlight();	Enciende la luz de fondo
lcd.clear()	Limpia la pantalla
lcd.print()	Imprime un mensaje en el LCD
lcd.setCursor(columna, fila)	Mueve el cursor a una posición designada
dht.begin()	Inicio del sensor DHT
dht.readTemperature()	Leer la temperatura
dht.readHumidity()	Leer la humedad relativa
pixels.begin()	Inicio para el encendido de LEDs
attachInterrupt(digitalPinToInterrupt(pin), ISR, mode)	Crea una interrupción pin: número de pin utilizado en el Arduino ISR: rutina de servicio de interrupción. mode: define cuándo se debe disparar la interrupción. Cuatro constantes están predefinidas como valores válidos: BAJO: estado bajo CHANGE: cuando cambie de valor RISING: cambio de estado de bajo a alto FALLING: cambio de estado de alto a bajo
pinMode()	Configura el pin como entrada o salida
digitalWrite()	Permite escribir un BAJO (0) o un ALTO (1).

	<p>Si el pin está configurado como salida el voltaje se establecerá en el valor correspondiente: 5V (o 3.3V en placas de 3.3V) para ALTO, 0V (tierra) para BAJO</p> <p>Si el pin está configurado como entrada se habilitará (ALTO) o deshabilitará (BAJO) el pullup interno en el pin de entrada</p>
<code>analogRead()</code>	Lee el valor del pin analógico especificado
<code>map()</code>	Asigna un valor de un rango a otro
<code>if (condición) { //declaraciones }</code>	Verifica una condición y ejecuta la siguiente declaración o conjunto de declaraciones si la condición es verdadera
<code>while(condición){ // declaraciones }</code>	Ciclo repetitivo continuo e infinito que se ejecuta hasta que la expresión entre paréntesis sea falsa
<code>for (inicialización; condición; incremento) { // declaraciones; }</code>	Permite repetir un bloque de declaraciones entre llaves. Normalmente se utiliza un contador de incrementos para incrementar y terminar el bucle
<code>delay()</code>	Genera una pausa durante la ejecución del código en milisegundos

2.3.6 CONSTRUCCIÓN DE UN INVERNADERO A ESCALA

Las herramientas y materiales (Figura 2.12) utilizados para la construcción del invernadero a escala son los siguientes:

Materiales:

- Un tablero de fibra de densidad media, de 57 × 80 × 2 cm.
- Tiras de madera de laurel para la estructura del invernadero: cuatro tiras de 2 × 60 × 2 cm, dos tiras de 2 × 69 × 2 cm, cuatro tiras de 2 × 42 × 2 cm, cuatro tiras de 3 × 32 × 3 cm y una tira de 3 × 63 × 3 cm.

- Una tabla de melánico RH de 23 x 68 cm, base utilizada para colocar una caja que contiene los circuitos.
- Madera trípex, dos tablas de 15 x 60 cm, dos tablas de 15 x 15 cm, dos tablas de 17 x 60 cm, con este material se construye la caja que contiene los elementos electrónicos.
- Madera trípex, dos tablas de 17 x 20 cm, una tabla de 18 x 48 cm, dos tablas de 18 x 48 cm, con este material se construye la caja protectora para colocar dentro de ésta los tomacorrientes y las respectivas fuentes para alimentar el circuito.
- Madera trípex, dos tablas de 13 x 23 cm, una tabla de 10 x 13 cm, cuatro tiras de laurel de 2 x 13 x 2 cm, cuatro tiras de 2 x 7 x 2 cm, con este material se construye un soporte para colocar el calefactor portátil.
- Cola blanca para madera.
- Seis bisagras para puerta sin pasador. Éstas sirven para colocar en las cajas mencionadas lo cual permite abrir y cerrar para observar los circuitos y elementos colocados en su interior.
- Plástico tipo mica, tres metros en total para realizar el recubrimiento.
- Una caja de clavos de madera para ajustar el plástico tipo mica
- Dos metros de manguera para parabrisas, es de utilidad porque permite simular el riego por goteo haciendo agujeros a cierta distancia para la caída de agua.
- Canaleta para cables, cinco metros, se dispone de tres tiras empotradas en dos maderas para permitir el paso de la manguera utilizada para el sistema de riego por goteo y ajustar a estas dos tiras LEDs direccionables. También al costado de la estructura y colocados de manera estratégica se colocan canaletas para paso de cables para la alimentación de actuadores, Arduino YÚN y módulo relay de ocho canales.
- Lata de pintura en aerosol color café, para dar detalle y color.
- Broca de 0,8 mm, utilizada para realizar los respectivos agujeros para ajustar el módulo relay de 8 canales, PCB y Arduino.
- 2 m de tira LED direccionable.

- Malla para tejer, esta malla fue seleccionada gracias a su costo barato, fácil manipulación, y que su fin es de colocar al otro extremo de los ventiladores para expulsar el aire viciado.
- Reservorio de agua de 1 litro, aquí se coloca la mini bomba de agua en conjunto con la salida de la manguera para que funcione el sistema de riego por goteo.
- Diez metros de tubo termo retráctil, útil para cubrir las conexiones realizadas brindando seguridad y estética en el diseño.
- Diez metros de cable de timbre, útil para realizar las conexiones entre los actuadores, módulo relay y Arduino YÚN.
- Fuentes de poder, dos fuentes de 12 VDC – 1 A para alimentar a los ventiladores, una fuente de 5 VDC – 1 A para alimentar a la mini bomba de agua, dos metros de cable gemelo calibre 14, un enchufe y toma corriente para alimentar al calefactor, una fuente de 5 VDC – 1 A para alimentar a dos tiras LEDs, una fuente de 5 VDC – 1 A para alimentar externamente al módulo relay de ocho canales y una fuente de 9 VDC – 2 A para alimentar al Arduino YÚN.

Herramientas:

- Cúter o tijeras para recortar.
- Taladro.
- Pistola de silicona.
- Pela cables.
- Martillo.

El corte de las maderas se pide en tiendas de carpintería o que se especialicen en cortes de madera, pues la maquinaria que tienen les permite realizar corte a medida y con gran facilidad. Una vez pintadas cada una de las maderas a utilizar, se procede con el armado de la estructura que se realiza midiendo adecuadamente la posición de cada tira, en conjunto con el uso de clavos para madera y martillo, ajustando así cada una de éstas. Posteriormente el recubrimiento con plástico es externo y hermético para crear un ambiente cerrado. Finalmente se debe colocar cada caja en el respectivo lugar destinado, los actuadores y elementos electrónicos. La Figura 2.13 muestra el estructural del invernadero a escala.



Figura 2.12 Materiales y herramientas utilizados para la construcción del invernadero a escala.



a)



b)



c)



d)



e)

Figura 2.13 Estructura del invernadero a escala. a) Vista frontal; b) Vista lateral derecha; c) Vista lateral izquierda; d) Vista posterior; e) Vista superior.

Las conexiones que van desde la plataforma Arduino YÚN hacia el módulo relay, sensores, compuerta lógica y actuadores, necesitan de un circuito impreso o también conocido como PCB. Las Figura 2.14 y Figura 2.15 muestran el diseño e implementación del circuito realizado en Proteus 8 y una foto del PCB construido, respectivamente.

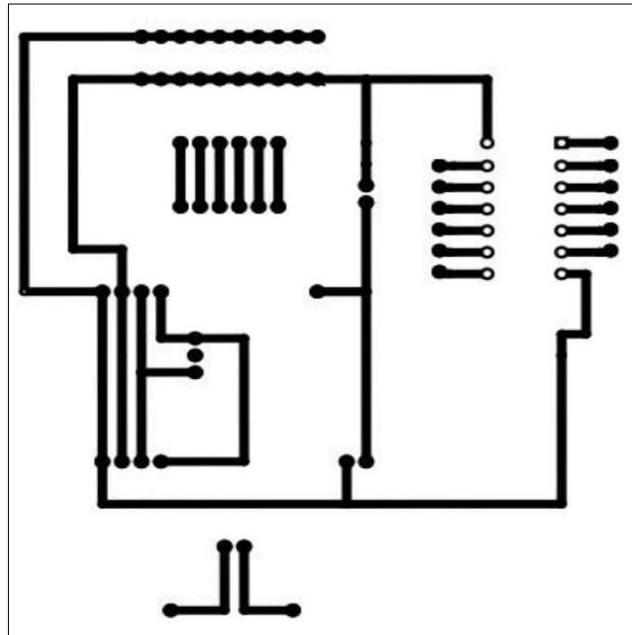


Figura 2.14 Diseño del circuito impreso diseñado en Proteus 8.

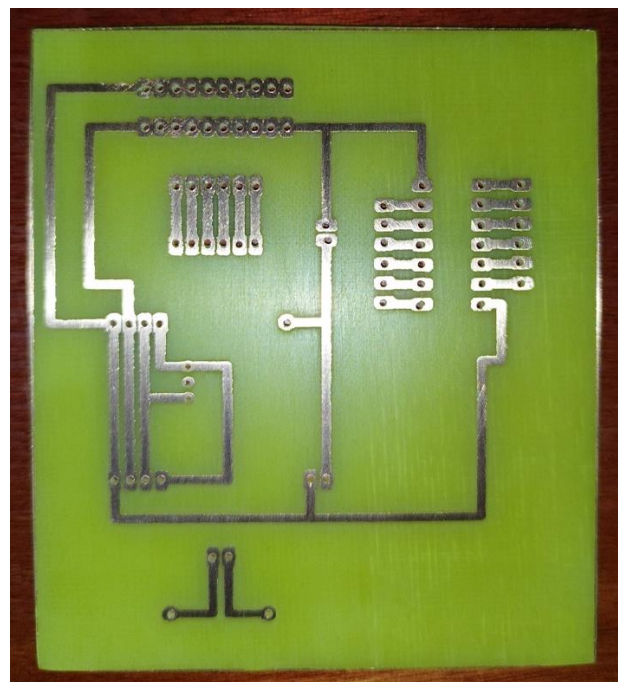


Figura 2.15 Circuito impreso en fibra de vidrio.

Realizada la estructura se forra con el plástico tipo mica cada parte necesaria herméticamente para crear un ambiente cerrado, se acomodan los diferentes elementos ya mencionados. Desde la Figura 2.32 hasta la Figura 2.36 se visualiza el acabado final de los diferentes elementos dispuestos acorde a los requerimientos para generar un ambiente controlado a escala similar al de un invernadero real en el cual se realizan las pruebas de funcionamiento.



a)



b)



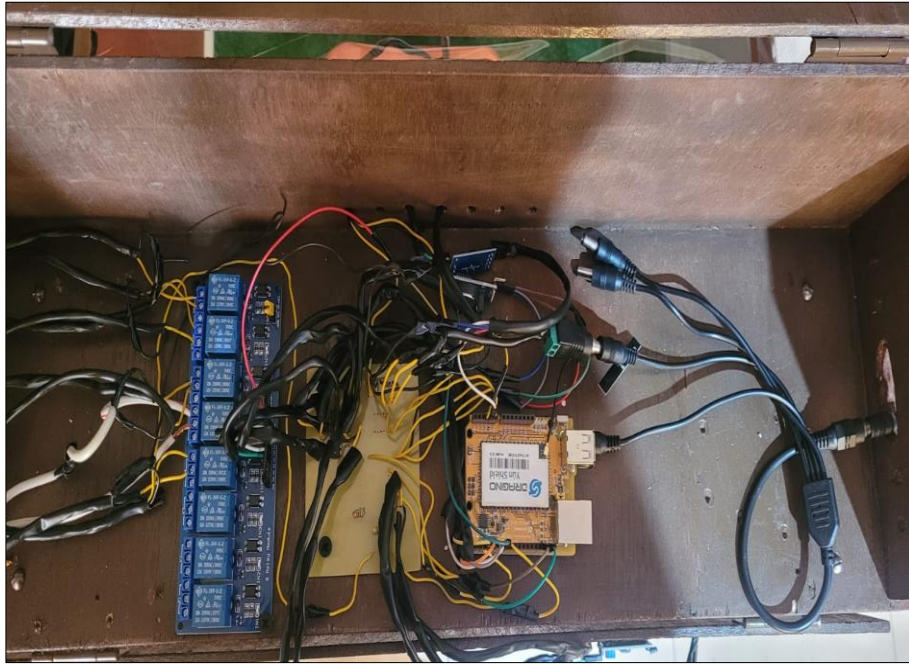
c)



d)



e)



f)



g)

Figura 2.16 Invernadero a escala. a) Vista frontal; b) Vista lateral derecha; c) Vista lateral izquierda; d) Vista posterior; e) Vista superior; Conexiones entre los elementos electrónicos, sensores y actuadores. f) Vista superior; g) Vista frontal.

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

Este capítulo presente los resultados obtenidos de las pruebas de funcionamiento. Asimismo, se presentan conclusiones y recomendaciones derivadas del trabajo.

3.1 INICIO DEL SISTEMA, VERIFICACIÓN DE MÓDULO RTC DS3231 Y PRESENTACIÓN DEL SISTEMA DE SOPORTE VITAL

El inicio del código presenta un mensaje de bienvenida e indica que el sistema se encuentra activo para su funcionamiento. La Figura 3.1 muestra el mensaje desplegado en pantalla al iniciar el sistema.



Figura 3.1 Mensaje de arranque del sistema.

Consecutivamente se verifica si el módulo RTC DS3231 se encuentra activo. La Figura 3.2 muestra una primera prueba de funcionamiento en la cual se desconecta el módulo generando un error que se presenta indefinidamente hasta que el módulo se encuentre conectado.

Figura 3.2 Mensaje de error en el RCT DS3231 debido a que no se encuentra activo.

Al conectar el RTC inmediatamente se logra verificar su funcionamiento y se presenta un mensaje para indicar el estado activo (Figura 3.3). Este paso permite ya poner en marcha al sistema.



Figura 3.3 Mensaje de correcto funcionamiento del RCT DS3231.

Seguidamente, se presenta un mensaje indicando la fecha y hora actual (Figura 3.5a). A continuación, se indica el valor de la temperatura en grados centígrados, humedad relativa en porcentaje, niveles de CO2 en ppm y humedad del suelo en porcentaje (Figura 3.5b). Los valores del sistema de soporte vital son cargados ya en memoria. Éstos se modifican al presionar el botón que se encuentra junto a la pantalla LCD.



Figura 3.4 Mensaje de fecha y hora actual.



a)



b)

Figura 3.5 Sistema de soporte vital. a) Temperatura y humedad relativa; b) CO2 y humedad del suelo.

Como paso final, el sistema indica un mensaje (Figura 3.6) comenzando con el análisis de la información del ambiente que ha sido inicialmente puesto en memoria con valores dentro del rango establecido.

- Temperatura: 24°C
- Humedad relativa: 70%
- Niveles de CO2: 350 ppm
- Humedad del suelo: 35%



Figura 3.6 Análisis del sistema de soporte vital.

3.2 SISTEMA DE ILUMINACIÓN Y CONTROL DE LA HUMEDAD RELATIVA

En este apartado el funcionamiento está programado para el horario de 2 am a 6 am. Se presenta en pantalla la fecha y hora actual (Figura 3.4), lo que corrobora que el sistema trabaja según las interrupciones especificadas dentro del código.

Un mensaje indica que empieza el análisis de las variables, a continuación, se indica que el horario es válido por lo cual la iluminación, ventilación y calefacción están encendidos y listos para operar. La Figura 3.7 muestra los mensajes correspondientes.



a)



b)

Figura 3.7 Mensajes de inicio de análisis. a) Mensaje de horario válido e iluminación;
b) Mensaje de calefacción y ventilación.

El sistema de iluminación comienza entonces su encendido ininterrumpidamente hasta las 6 am. La Figura 3.8 muestra el mensaje de encendido del sistema de iluminación.



Figura 3.8 Encendido del sistema de iluminación artificial.

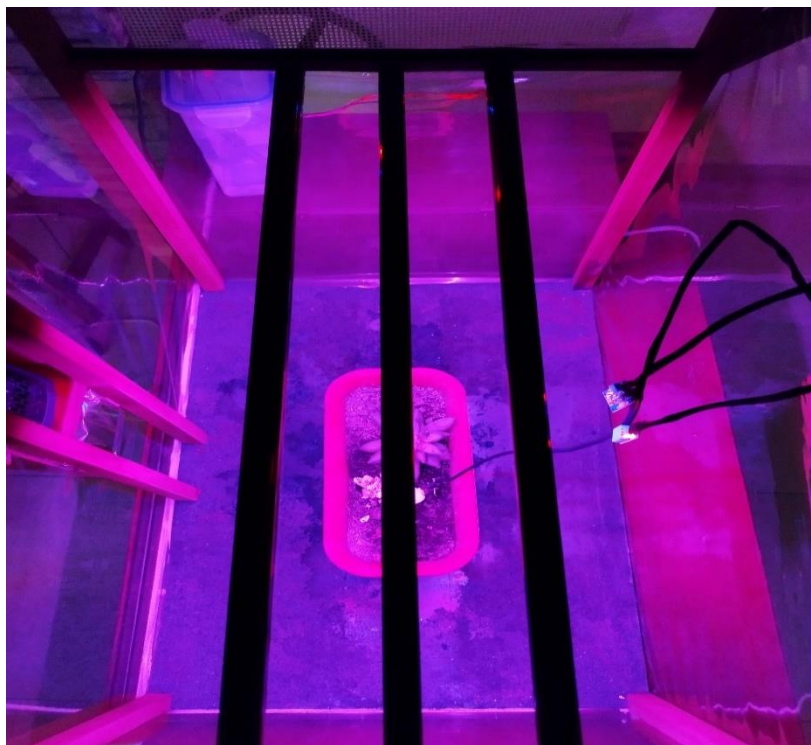
Cabe destacar que el sistema de iluminación trabaja durante todo este lapso y está realizado por medio de LEDs RGB direccionables no genera calor excesivo como es el caso de otras luminarias. De esta manera se evita un gran consumo de energía y desperdicio de ésta en forma de calor. La Figura 3.9 presenta el encendido de las dos tiras LEDs de un arreglo de 2×19 elementos de las cuales se enciende un LED color azul por cada tres LEDs color rojo garantizando emular el espectro de luz visible.



a)



b)



c)

Figura 3.9 Sistema de iluminación. a) Vista posterior; (b) Vista lateral; (c) Vista superior.

La primera parte de la función del código encargada de regular la humedad relativa, "hum_relativa()", analiza los datos. Si éstos se encuentran en el rango adecuado entonces la función no enciende los actuadores y presenta el mensaje indicado en la Figura 3.10.



Figura 3.10 Mensaje indicado cuando el valor de la humedad relativa se encuentra dentro del rango establecido.

Seguidamente, una nueva lectura empieza en un lapso de cinco minutos (Figura 3.11). Algunos de estos procesos desplegados como mensajes en pantalla se repiten constantemente.



Figura 3.11 Nueva lectura mediante un conteo regresivo de cinco minutos.

Para variar la humedad relativa únicamente se rocía agua con un atomizador convencional que permite emular la transpiración de las plantas en el ambiente cerrado. Se esparce una gran cantidad del líquido con mucho cuidado por todo el ambiente hasta conseguir un valor superior al 90%. Finalizado este proceso e iniciada una nueva lectura, el uso de la interrupción correspondiente en el código permite obtener la variación de este parámetro. Está claro que se adquieren también los valores de las otras magnitudes ambientales pero como para este horario solo se contempla la función que controla la humedad relativa no se hacen procesos por parte de los actuadores diferentes al destinado. Cabe mencionar que la temperatura también es controlada debido a la relación inversamente proporcional

que tiene con la humedad relativa, es decir, a mayor temperatura menor humedad relativa y viceversa.

La FUFJDKJHAKDJHFKJHADFJ muestra el mensaje desplegado en pantalla al término de la interrupción. Nótese que los parámetros han cambiado con relación a los mostrados en la Figura 3.5a.



Figura 3.12 Sistema de soporte vital. Mensaje con nuevos valores de temperatura y humedad relativa.

Cuando la humedad relativa sobrepasa el valor máximo configurado de 90% se inicia, como subrutina, el proceso de ventilación para despejar el aire húmedo durante cinco minutos. Posteriormente se enciende la calefacción con un valor por omisión de fábrica de 25°C (manualmente modificable).

El mensaje que inicia el proceso de regulación ambiental indica que el ambiente debe ser regulado lo que enciende el sistema de calefacción. La secuencia de mensajes para este proceso se muestra en la Figura 3.13.



a)



b)



c)

Figura 3.13 Sistema de ventilación. a) Mensaje informativo; b) Parámetro que requiere regulación; c) Encendido del actuador por cinco minutos.

La Figura 3.14 presenta cómo el actuador realiza el trabajo designado para despejar el aire húmedo a través de la ventila de aire. Este actuador se debe detener manualmente, esto se logra pulsando el botón de la interrupción que leerá si la humedad está dentro del rango adecuado caso contrario puede permanecer encendido indefinidamente.



Figura 3.14 Encendido del sistema de ventilación, vista lateral izquierda.

Finalizado el proceso descrito anteriormente se apaga el sistema de ventilación dando paso al sistema de calefacción para regular la temperatura ambiental. Los mensajes respectivos se muestran en la Figura 3.15.



a)



b)

Figura 3.15 Mensajes del sistema de calefacción. a) Apagado del sistema de ventilación; b) Encendido del sistema de calefacción.

El calefactor empieza a funcionar. Se puede conocer la temperatura por defecto de fábrica en la pantalla incorporada en el mismo (Figura 3.16).



Figura 3.16 Sistema de calefacción encendido proporcionado una temperatura de 25 °C.

De igual forma y para casos posteriores, se debe tener en cuenta que es necesario pulsar el botón para activar la interrupción y de esta manera salir de la subrutina una vez que el ambiente haya sido regulado.

La Figura 3.17 muestra el mensaje de que se ha regulado el ambiente y la calefacción se ha apagado terminando así el control de la humedad relativa.



Figura 3.17 Apagado del sistema de calefacción indicando que la regulación del ambiente ha sido exitosa.

Para verificar los nuevos valores de los parámetros se inicia el proceso con una nueva lectura. En la Figura 3.18 se aprecia que la temperatura ahora aumentó a 22°C y la humedad relativa bajó al 72,30 %.



Figura 3.18 Mensaje con nuevos datos del sistema de soporte vital.

3.3 SISTEMA DE VENTILACIÓN PARA EL CONTROL DE ALTAS TEMPERATURAS Y BAJOS NIVELES DE CO2

El horario de funcionamiento para controlar los altos niveles de temperatura y CO2 mediante la ventilación y las bajas temperaturas mediante el uso de la calefacción es de 6 am a 6 pm.

Aquí se consideran tres posibles escenarios (casos): I. Valores en rangos adecuados (no se encienden los actuadores); II. Temperatura alta y niveles de CO2 bajos; III. Temperatura alta y niveles de CO2 adecuados, y; IV. Temperatura adecuada y niveles de CO2 bajos.

3.3.1 VALORES EN RANGOS ADECUADOS

La función del control de la temperatura y humedad "temp_co2()", despliega mensajes indicando que los valores de las magnitudes mencionadas están en el rango adecuado.



a)



b)



c)

Figura 3.19 Secuencia de mensajes del sistema de soporte vital – Caso I. a) Indicación de horario válido y mensaje indicando el estado activo (OK) del sistema de ventilación; b) sistema de calefacción; c) temperatura y CO2. Mensaje de apagado (OFF) del sistema de ventilación.

3.3.2 TEMPERATURA ALTA Y NIVELES DE CO2 BAJOS

La interrupción en una nueva lectura procede a captar la variación de los parámetros de temperatura por encima del valor máximo de 28°C y de nivel de CO2 por debajo de 300 ppm. Para emular este escenario se introducen plantas al invernadero (Figura 3.20) en conjunto con ventilación manual externa. Asimismo, se utiliza un caudín para elevar la temperatura (Figura 3.21).



Figura 3.20 Plantas en el invernadero para el consumo del CO₂.

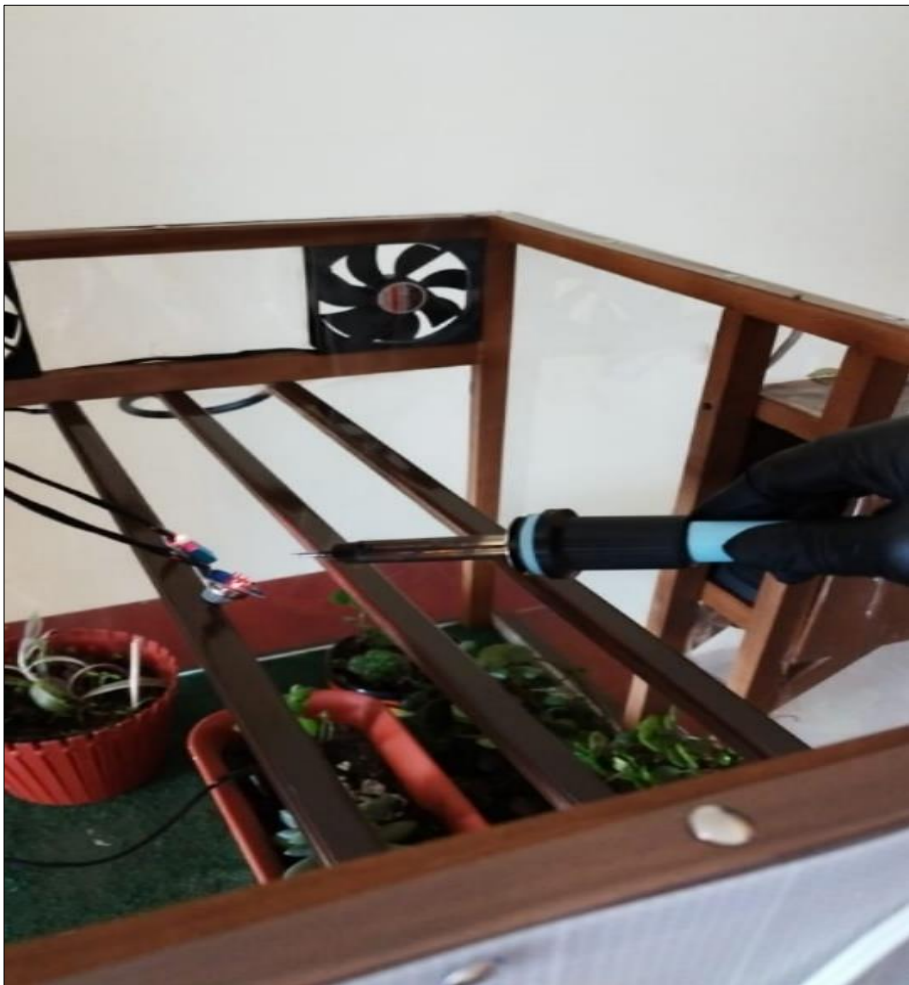


Figura 3.21 Proceso de elevación de la temperatura con un caudín dentro del invernadero.

Se cierra la estructura y se da paso a la interrupción para obtener los datos del sistema de soporte vital tal como se muestra en la Figura 3.22.



a)



b)

Figura 3.22 Mensajes del sistema de soporte vital – Caso II. a) Temperatura y humedad relativa; b) CO2 y humedad del suelo.

Se despliega un mensaje que muestra que el ambiente necesita ser regulado y que las magnitudes a controlar son la temperatura alta y CO2 bajo (Figura 3.23). Se inicia el actuador encargado de la ventilación y es necesario activar la interrupción para monitorear el cambio producido.



a)



b)



c)

Figura 3.23 Secuencia de mensajes del sistema de soporte vital – Caso II. a) Necesidad de regulación; b) Magnitudes a regularse; c) Encendido del sistema de ventilación.

El CO₂ del ambiente externo ingresado no permite alcanzar el rango de valores requerido. En sistemas a escala real, para este escenario, se hace uso de inyectores de CO₂ cuya infraestructura es compleja y costosa. Para el invernadero a escala objeto del presente trabajo se realiza la quema del material para que las emisiones de los gases producidos por la combustión ingresen al ambiente, tal como se muestra en la Figura 3.24. La Figura 3.25 muestra el mensaje de que el sistema ha sido regulado.



Figura 3.24 Inyección manual de CO₂.



Figura 3.25 Ambiente regulado después de la acción del sistema de ventilación e inyección manual.

Al obtener los nuevos valores a través de la interrupción del sistema de soporte vital se alcanza una temperatura ambiente de 23 °C y niveles de CO₂ de 710 ppm, como se muestra en la Figura 3.26.



a)



b)

Figura 3.26 Mensaje que da a conocer los datos del sistema de soporte vital – Caso II. a) Temperatura y humedad relativa; b) CO₂ y humedad del suelo.

3.3.3 TEMPERATURA ALTA Y NIVELES DE CO₂ ADECUADOS

La tercera subrutina se enfoca en el parámetro de la temperatura cuando ésta sobrepasa el límite superior. El mensaje de inicio indica que es necesario regular el ambiente interno. Se procede a indicar qué parámetro del sistema vital de soporte es necesario corregir y luego el actuador, para el caso específico, el de sistema de ventilación se pone en marcha. Esta secuencia de mensajes se presenta en la Figura 3.27.

El uso del cautín para calentar el aire es de utilidad nuevamente, este no afecta gravemente al CO2 y se puede enfocar el análisis en este caso en específico.



a)



b)



c)



d)

Figura 3.27 Secuencia de mensajes del sistema de soporte vital – Caso III. a) Mensaje con datos; b) Mensaje que muestra la necesidad de regular el ambiente; c) Mensaje con el parámetro del sistema de soporte vital a regular; d) Mensaje de encendido del sistema de ventilación.

Al analizar nuevamente el ambiente mediante la interrupción, la ventilación ha regulado el parámetro que ha excedido su límite superior establecido (Figura 3.28).



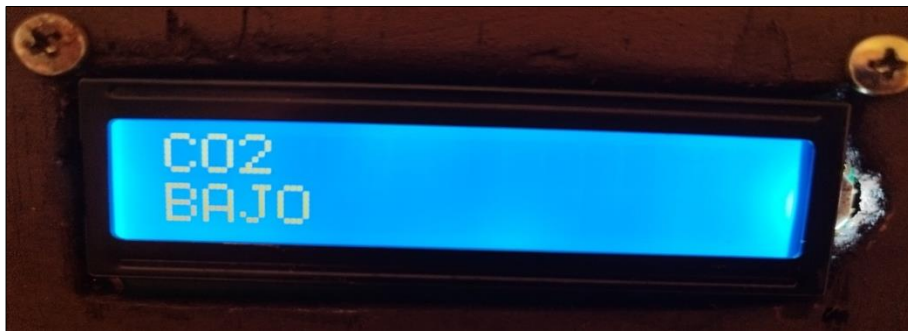
Figura 3.28 Mensaje de ambiente regulado después de la acción del sistema de ventilación – Caso III

3.3.4 TEMPERATURA ADECUADA Y NIVELES DE CO2 BAJOS

La última subrutina ahora se enfoca únicamente cuando el nivel de CO₂ se encuentra por debajo del mínimo. El proceso para bajar los niveles de CO₂ se ejecuta gracias al método mencionado al inicio de esta sección generando un análisis de CO₂ con valor de 287 ppm.



a)



b)



c)

Figura 3.29 Secuencia de mensajes del sistema de soporte vital – Caso IV. a) Mensaje que muestra la necesidad de regular el ambiente; b) Mensaje con el parámetro del sistema de soporte vital a regularse; c) Encendido del sistema de ventilación.

El ambiente se regula con un valor de CO₂ de 312 ppm por lo cual no es necesaria la inyección de CO₂ manualmente (Figura 3.30).



Figura 3.30 Mensaje que presenta los datos del sistema de soporte vital regulado.

3.4 SISTEMA DE CALEFACCIÓN PARA EL CONTROL DE TEMPERATURAS BAJAS

La función calefaccion() brinda seguridad evitando un decremento en los bajos niveles de temperatura.

La primera subrutina rutina en este escenario inicia con un mensaje cuando se encuentra la temperatura en valores adecuados. De igual forma que en procesos anteriores, se repiten los mensajes que indican el estado y proceso a ejecutar (Figura 3.31).



Figura 3.31 Mensaje que indica que el valor de la temperatura está dentro del rango establecido.

La segunda subrutina da inicio al obtener una temperatura inferior al valor mínimo de 15°C. En vista de que el prototipo no cuenta con un sistema de refrigeración, la generación de este escenario resulta bastante compleja. Entonces, para comprobar que el actuador efectivamente encienda el sistema de calefacción, no se utilizó el sensor de temperatura, sino que se configuró directamente la activación de la subrutina estableciendo manualmente la variable de lectura de temperatura en 14°C. La ADSKJFHLAKJDHF muestra la secuencia de mensajes para la activación del sistema de calefacción.

La temperatura real para dicho instante es de 21°C, esto se puede superar gracias a que el actuador puesto en marcha inicia con un valor de 25 °C, durante el lapso que trabaja y

gracias a la estructura hermética el calor generado garantiza entrar al rango establecido siendo mayor a 15° y menor a 28°C.



a)



b)



c)

Figura 3.32 Secuencia de mensajes del sistema de soporte vital – Sistema de calefacción. a) Mensaje que muestra la necesidad de regular el ambiente; b) Mensaje con el parámetro del sistema de soporte vital a regularse; c) Encendido del sistema de calefacción.

La interrupción toma los valores actuales del sistema, la temperatura es de 25 °C, se ha detenido el sistema de calefacción, el despliegue de un mensaje indicando que el ambiente ha sido regulado se ha completado y se procede nuevamente a una lectura de los datos ().



a)



b)

Figura 3.33 Mensaje que presenta los datos del sistema de soporte vital regulado – Sistema de calefacción. a) Ambiente regulado después de la acción del sistema de calefacción. b) Mensaje que presenta los datos del sistema de soporte vital regulado.

3.5 SISTEMA DE RIEGO POR GOTEO PARA EL CONTROL DE LA HUMEDAD DEL SUELO

La humedad del suelo es de suma importancia para un correcto desarrollo de los cultivos. La fotosíntesis no se produciría sin la luz solar, pero tampoco sin tener los niveles adecuados de líquido que son tomados por las raíces el cual no debe ser por debajo de un 10% por lo que podría secarse y morir. De manera similar valores superiores produce un ahogamiento, sobrehidratación, enfermedades e inevitablemente la muerte.

El rango para la humedad del suelo es de 10 – 50 %, así se garantiza una buena hidratación que va de la mano en las horas donde la radiación solar es amigable para la planta. Esto es de 6 am a 10 am y de 3 pm a 5 pm. Un sol de mediodía puede llegar a quemar a las plantas y el riego nocturno a sobre hidratar ya que no se tiene radiación solar para los procesos de crecimiento, germinación, entre otros.

El primer mensaje (Figura 3.34) da a conocer que el sistema se encuentra en el horario de funcionamiento, por lo cual la activación de la bomba se efectuará en caso de ser necesaria.



Figura 3.34 Horario válido para el funcionamiento del sistema de riego por goteo.

Con los valores configurados inicialmente con una humedad relativa del 35% la primera subrutina se ejecuta indicando que no es necesario encender el actuador correspondiente. El mensaje correspondiente se presenta en la Figura 3.35



Figura 3.35 Valor de la humedad del suelo dentro del rango establecido.

La ejecución de la segunda subrutina se produce al utilizar tierra muy seca alcanzando un valor de humedad del 7%, tal como se muestra en la Figura 3.36.



Figura 3.36 Mensaje que da a conocer los datos del sistema de soporte vida – Humedad del suelo baja.

En pantalla se despliega un mensaje para dar a conocer que se requiere riego (Figura 3.37a), el actuador de la mini bomba se enciende (Figura 3.37b) y el uso de la interrupción apagará este sistema una vez que se alcance un valor adecuado.



a)



b)

Figura 3.37 Requerimiento de riego por valores bajos de humedad en el suelo. a) Mensaje informativo; b) Mensaje que indica la activación del riego por goteo.

La Figura Figura 3.38 muestra el sistema de riego por goteo activo.



Figura 3.38 Encendido del sistema de riego por goteo, vista frontal.

Una vez la interrupción detecta el valor requerido de humedad del suelo, apaga inmediatamente el actuador desplegando un sistema indicando que el ambiente se ha regulado (Figura 3.39).



Figura 3.39 Ambiente regulado después de la acción del sistema de riego por goteo.

3.6 CONCLUSIONES

El ambiente interno de un invernadero presenta variaciones que son provocadas por cambios en el clima exterior que dependen también de la estructura, cubierta y ubicación geográfica del invernadero en conjunto con los sistemas encargados de la ventilación, riego, calefacción e iluminación. El funcionamiento de los actuadores mantiene el sistema

de soporte vital acorde a rangos generales establecidos. Cada factor climático influye durante el crecimiento, desarrollo, acumulación de reservas alimenticias, cantidad de líquido y germinación de las plantas. La implementación de un prototipo a escala que actúa de forma semiautomática haciendo uso de los elementos electrónicos realizó el encendido de cada actuador asegurando niveles adecuados en cada caso considerado.

La creación de un código haciendo uso del lenguaje establecido por Arduino a través del manejo del IDE se concreta por su facilidad y entorno amigable. La diversidad de librerías, sencillez para la declaración de variables, estructura y alcance de las funciones propias de Arduino y desarrolladas permiten generar rutinas específicas. Dichas rutinas trabajan en horarios establecidos manipulando mediante el encendido y trabajo conjunto de cada actuador implementado en cada uno de los sistemas. El control y regulación del sistema de soporte vital se ejecutó al mantener a los parámetros inherentes del ecosistema en los rangos instaurados.

Arduino YÚN es una plataforma de desarrollo que combina el uso de Linux con la facilidad de manipular las características que ofrece el dispositivo electrónico como son las interfaces de entrada y salida. Estas brindan puertos que adquirieron un conjunto de datos a través de sus entradas analógicas con el uso de sensores, o enviaron señales PWM para el encendido de actuadores hablando específicamente del enfoque dado al presente trabajo técnico.

Los actuadores utilizados en invernaderos a escala real presentan características provechosas para un control y regulación del ambiente interno, el cual, presenta dimensiones considerablemente grandes. Al implementar un prototipo a escala se emuló un ambiente similar al real, que fue controlado para el desarrollo del cultivo. Los ventiladores, la mini bomba de agua, la manguera de riego, calefactor y luminaria LEDs para llevar a cabo la manipulación de las magnitudes ambientales ejecutaron los procesos de manera adecuada sin ningún tipo de problema.

La plataforma Arduino YÚN cuenta con un total de veinte entradas y salidas digitales de las cuales doce pueden ser utilizadas como entradas analógicas y siete de ellas como salidas PWM. En la implementación del prototipo se optimizó el uso de los pines: tres analógicos para los sensores, siete pines PWM para la interrupción y encendido de actuadores, y finalmente dos pines destinados a dispositivos compatibles con I2C.

La estructura de variables se enfoca en generar un valor máximo y mínimo para el sistema de soporte vital con la combinación de valores de guarda que permiten establecer un rango

seguro al regular cada variable ambiental. La carga en memoria no es pesada ocupando el 54% de la capacidad de ésta.

3.7 RECOMENDACIONES

El jack de la alimentación del Arduino YÚN presenta una caída de voltaje en el pin que debería proporcionar 5 VDC con un valor final de máximo 3 VDC siendo insuficiente para alimentar a los demás elementos. Este problema se puede arreglar al conectar la fuente de alimentación de 5 VDC – 1 A con un splitter de 4 salidas manteniendo el voltaje y misma referencia de tierra para operar la pantalla LCD, módulo relay de ocho canales, sensores y módulo RTC.

El encendido y apagado de cada relay funciona con lógica inversa (LOW enciende el relay, HIGH apaga el relay). Esto se puede controlar haciendo uso de una compuerta NOT para trabajar con lógica directa (HIGH enciende el relay, LOW apaga el relay) en el código.

En implementaciones futuras se sugiere diseñar una rutina para el ingreso manual de un valor mínimo y máximo que configure el rango acorde a las características del cultivo de interés.

4 REFERENCIAS BIBLIOGRÁFICAS

- [1] J. Pérez, “Un invernadero inteligente para optimizar los cultivos.” <https://repository.eafit.edu.co/bitstream/handle/10784/16953/document%20-%202020-06-19T163716.660.pdf?sequence=2>.
- [2] T. Mendieta and A. Jiménez, “La Capacidad del IOT de Transformar el Futuro.” <https://fundacionavenir.net/revista/index.php/avenir/article/view/79/27>.
- [3] T. de Producción, “I INDUSTRIA HORTICOLA.” [Online]. Available: www.agrocomponentes.com
- [4] “DS3231.” <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>.
- [5] “Ambiente invernadero y climatización -.” <https://www.mundohvacr.com.mx/2014/05/ambiente-invernadero-y-climatizacion/>.
- [6] “Corrientes de convección: definición, estudios y réplicas.” <https://www.lifeder.com/corrientes-de-conveccion/>.
- [7] C. Kittas, N. Katsoulas, and A. Baille, “Influence of greenhouse ventilation regime on the microclimate and energy partitioning of a Rose Canopy during summer conditions,” *Journal of Agricultural and Engineering Research*, vol. 79, no. 3, pp. 349–360, 2001, doi: 10.1006/JAER.2000.0653.
- [8] P. I. S. Medrano, “Medición de Humedad Relativa con Psicrómetro Somos su Relevo a la Calidad”. [Online]. Available: www.metas.com.mx
- [9] L. Zotarelli, M. D. Dukes, and K. T. Morgan, “Interpretación del Contenido de la Humedad del Suelo para Determinar Capacidad de Campo y Evitar Riego Excesivo en Suelos Arenosos Utilizando Sensores de Humedad 1 Capacidad del suelo para almacenar agua”. [Online]. Available: <http://edis.ifas.ufl.edu/ae266>
- [10] “Consejos de Riego.” <https://www.snwa.com/espanol/jardines/consejos-de-riego-por-goteo.html>.
- [11] “SIGELEC - Espectro de Luz artificial Led y la fotosíntesis.” <https://sigelec.com.pe/zona-formativa/articulos-tecnicos/espectro-de-luz-artificial-led-y-la-fotosintesis>.
- [12] J. Maroto, *Elementos de Horticultura General*, 3ª. Madrid: Mundi-Prensa.

- [13] Lenscak. Mario and N. Iglesias, “Invernaderos Tecnología apropiada en las regiones productivas del territorio nacional argentino.” https://inta.gob.ar/sites/default/files/inta_-_invernaderos.pdf.
- [14] D. Raffo, “La radiación solar y las plantas: un delicado equilibrio.” https://inta.gob.ar/sites/default/files/script-tmp-inta_fyd_74_la-radiacion-solar-y-las-plantas.pdf.
- [15] A. ALPI and F. TOGNONI, *Cultivo en invernadero*, 3ra ed. Madrid: Ediciones Mundi-Prensa, 1999.
- [16] E. Lab, “INFORMACIÓN TECNICA SOBRE GASES DE EFECTO INVERNADERO Y EL CAMBIO CLIMÁTICO”.
- [17] J. Martínez, A. Fernández, and P. Osnaya, *Cambio climático: una visión desde México*, 1ra ed. Delegación Coyoacán: Secretaría de Medio Ambiente y Recursos Naturales, 2004. [Online]. Available: https://www.ccmss.org.mx/wp-content/uploads/2014/10/Cambio_Climatico_una_vision_desde_Mexico.pdf#page=27
- [18] “INTRODUCCIÓN A LOS SISTEMAS EMBEBIDOS (SSEE) 1.1.-Concepto y aplicaciones 1.1.1.-Concepto”.
- [19] P. Cepeda, H. Ponce, P. Ponce, and A. Molina, “Open Calls View project Artificial Organic Networks View project,” 2010. [Online]. Available: <https://www.researchgate.net/publication/264045356>
- [20] “Arduino Yun | Arduino.cl - Compra tu Arduino en Línea.” <https://arduino.cl/arduino-yun/>.
- [21] “Arduino Yún | Arduino Documentation | Arduino Documentation.” <https://docs.arduino.cc/retired/boards/arduino-yun>.
- [22] “Arduino Yún Arduino A000008 | BricoGeek.com.” <https://tienda.bricogeek.com/descatalogado/570-arduino-yun.html>.
- [23] “▷ ARDUINO YUN.” <https://www.circuitos-electricos.com/arduino-yun/>.
- [24] Estévez. Jesús, “DISEÑO DE UN PROGRAMA EN ANDROID PARA EL CONTROL DE ARDUINO”. [Online]. Available: <https://uvadoc.uva.es/bitstream/handle/10324/14057/TFG-P-302.pdf;jsessionid=A163828019C65F42438BA3C0F5EFF204?sequence=1>

- [25] “Software de Arduino | Arduino.cl - Compra tu Arduino en Línea.”
<https://arduino.cl/programacion/>.
- [26] “Arduino IDE en Windows Linux y Mac.”
<https://programarfacil.com/blog/arduino-blog/arduino-ide/>.
- [27] Leonel. Germán Corona Ramírez, G. S. Abarca Jiménez, and J. Mares Carreño, *Sensores y actuadores*. Larousse - Grupo Editorial Patria, 2014.
- [28] “Actuadores eléctricos: Qué son, cómo funcionan, dónde se utilizan.”
<https://sdindustrial.com.mx/blog/introduccion-a-los-actuadores-electricos-motores-electricos/>.
- [29] “Module Relay Rele 8 Canales OPTOACOPLADO Full Testeado.”
<https://rambal.com/control-de-potencia/774-module-relay-8-canales-optoacoplado.html>.
- [30] “Módulo Relay, 8 canales 5VDC.”
<https://naylampmechatronics.com/drivers/177-modulo-relay-8-canales-5vdc.html>.
- [31] “Ventilación natural y forzada de invernaderos tipo Almería y su relación con el rendimiento de tomate.” https://www.scielo.org.mx/scielo.php?pid=S0187-57792011000400379&script=sci_arttext.
- [32] “Sistemas de Ventilación Natural para Invernaderos - Control del Clima.”
<https://www.novagric.com/es/tecnologia/clima-invernaderos/ventilacion/natural-ventanas>.
- [33] “Ventilación Forzada o Mecánica para Invernaderos - Sistemas.”
<https://www.novagric.com/es/tecnologia/clima-invernaderos/ventilacion/forzada>.
- [34] “Sistemas de Calefacción.” <https://www.invernaderosima.com/es/sistemas-de-calefacci%C3%B3n.html>.
- [35] “Calefacción por Aire en Invernaderos - Control de Temperatura.”
<https://www.novagric.com/es/tecnologia/clima-invernaderos/calefaccion/calefaccion-aire>.
- [36] Mayhua-López *et al.*, “Sistema-de-riego-por-goteo-automático”.
- [37] “Tipos de tuberías para un sistema de riego por goteo agrícola - Prakor.com.”
<https://prakor.com/tipos-de-tuberias-para-un-sistema-de-riego-por-goteo-agricola/>.

- [38] M. Liotta, "Manual de capacitación: Riego por goteo". [Online]. Available: https://repositorio.inta.gov.ar/bitstream/handle/20.500.12123/4528/INTA_EEASanJuan_Liotta_Riego_por_goteo.pdf?sequence=1
- [39] "Features of the new Arduino YÚN | Freedom Embedded." <https://balau82.wordpress.com/2013/09/02/features-of-the-new-arduino-yun/>.
- [40] "Arduino YUN for Beginners - projectiot123 Technology Information Website worldwide." <https://projectiot123.com/2019/05/14/arduino-yun-for-beginners/>.
- [41] "Pinout Yun - Hardware / Arduino Yún - Arduino Forum." <https://forum.arduino.cc/t/pinout-yun/185026>.
- [42] "Ventilación Forzada o Mecánica para Invernaderos - Sistemas." <https://www.novagric.com/es/tecnologia/clima-invernaderos/ventilacion/forzada>.
- [43] "Ventilador extractor de aire para invernadero - CONAIRE - AgrosHOW." <https://agrosHOW.info/productos/hidroponia/ventiladores-y-extractores/ventilador-extractor-de-aire-para-invernadero/>.
- [44] "Ventilador Ensamble 12V 12x12x2.5cm." <https://www.bigtronica.com/elementos-termoelectricos/ventiladores/12v/2790-ventilador-ensamble-12v-12x12x25cm-5053212027902.html>.
- [45] "¡CALEFACTOR ELÉCTRICO PORTÁTIL! – FUNMITPERU." <https://funmitperu.com/products/calefactorportatil?variant=43015091028220>.
- [46] "▷ ¿Qué es un Contactor? Tipos, para qué sirve y cómo elegir." <https://www.electricasas.com/que-es-un-contactor-tipos-y-usos/>.
- [47] "Mini Bomba de Agua Sumergible - 3-6 VDC - 120L/H - Electronilab." <https://electronilab.co/tienda/mini-bomba-de-agua-sumergible-3-6-v-dc-120lh/>.
- [48] "Guía: La luz en tus plantas :. : Hydro Environment :. : Hidroponia en Mexico." https://www.hydroenv.com.mx/catalogo/index.php?main_page=page&id=221.
- [49] "¿Cuál es la hora más fría del día?" <https://www.muyinteresante.es/curiosidades/preguntas-respuestas/cual-es-la-hora-mas-fria-del-dia-131451986179>.
- [50] "Clima de un Invernadero. ¿Cómo conseguir la Temperatura Ideal?" <https://www.novagric.com/es/blog/articulos/clima-invernadero-como-conseguir-temperatura-ideal>.

- [51] “Inyección de Bióxido de Carbono (CO₂) en Invernadero | Intagri S.C.”
[https://www.intagri.com/articulos/horticultura-protegida/inyeccion-de-bioxido-de-carbono-\(CO2\)-en-invernadero](https://www.intagri.com/articulos/horticultura-protegida/inyeccion-de-bioxido-de-carbono-(CO2)-en-invernadero).
- [52] “Uso de Sensores de Humedad para Definir Riego | Intagri S.C.”
<https://www.intagri.com/articulos/agua-riego/uso-de-sensores-de-humedad-para-definir-riego>.
- [53] Á. Egri, Á. Horváth, G. Kriska, and G. Horváth, “Optics of sunlit water drops on leaves: Conditions under which sunburn is possible,” *New Phytologist*, vol. 185, no. 4, pp. 979–987, Mar. 2010, doi: 10.1111/J.1469-8137.2009.03150.X/ABSTRACT.
- [54] “DHT-sensor-library/DHT_U.h at master · adafruit/DHT-sensor-library · GitHub.” https://github.com/adafruit/DHT-sensor-library/blob/master/DHT_U.h.
- [55] “DHT sensor library - Arduino Reference.”
<https://www.arduino.cc/reference/en/libraries/dht-sensor-library/>.
- [56] “Wire - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/communication/wire/>.
- [57] “LiquidCrystal I2C - Arduino Reference.”
<https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>.
- [58] “GitHub - adafruit/Adafruit_NeoPixel: Arduino library for controlling single-wire LED pixels (NeoPixel, WS2812, etc.)”
https://github.com/adafruit/Adafruit_NeoPixel.
- [59] “void - Guía de Referencia de Arduino.”
<https://www.arduino.cc/reference/es/language/variables/data-types/void/>.
- [60] “Wire - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/communication/wire/>.
- [61] “setup() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/structure/sketch/setup/>.
- [62] “loop() - Guía de Referencia de Arduino.”
<https://www.arduino.cc/reference/es/language/structure/sketch/loop/>.
- [63] “begin() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/communication/wire/begin>.

- [64] “Arduino en español: Wire.begin().”
<http://manueldelgadocrespo.blogspot.com/p/wirebegin.html>.
- [65] “RTCZero - begin() - Arduino Reference.”
<https://www.arduino.cc/reference/en/libraries/rtczero/begin/>.
- [66] “Arduino rtc ds1307 |CodigoElectronica.”
<http://codigoelectronica.com/blog/arduino-rtc-ds1307#adjust>.
- [67] “LiquidCrystal - begin() - Arduino Reference.”
<https://www.arduino.cc/reference/en/libraries/liquidcrystal/begin/>.
- [68] “Arduino - LCD I2C | Arduino Tutorial.”
<https://arduinogetstarted.com/tutorials/arduino-lcd-i2c>.
- [69] “DHT11/DHT22 Sensor with Arduino | Random Nerd Tutorials.”
<https://randomnerdtutorials.com/complete-guide-for-dht11-dht22-humidity-and-temperature-sensor-with-arduino/>.
- [70] “NeoPixel: How to Control WS2812 RGB LED w/ Arduino - Arduino Project Hub.”
<https://create.arduino.cc/projecthub/electropeak/neopixel-how-to-control-ws2812-rgb-led-w-arduino-46c08f>.
- [71] “attachInterrupt() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/external-interrupts/attachinterrupt/>.
- [72] “pinMode() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/digital-io/pinmode/>.
- [73] “digitalWrite() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/digital-io/digitalwrite/>.
- [74] “analogRead() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/>.
- [75] “Escalamiento y función map | Arduino.cl - Compra tu Arduino en Línea.”
<https://arduino.cl/escalamiento-y-funcion-map/>.
- [76] “map() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/math/map/>.
- [77] “if - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/structure/control-structure/if/>.

[78] “while - Guía de Referencia de Arduino.”
<https://www.arduino.cc/reference/es/language/structure/control-structure/while/>.

[79] “for - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/structure/control-structure/for/>.

[80] “delay() - Arduino Reference.”
<https://www.arduino.cc/reference/en/language/functions/time/delay/>.

5 ANEXOS

ANEXO I. Código fuente implementado para manejo de actuadores a través del uso de la plataforma Arduino YÚN

ANEXO I

```
/*
*****
*****

* LIBRERÍAS PARA EL USO DE SENSORES, COMUNICACIÓN I2C, RELOJ

*
*****
*****/

#include <DHT.h> //librería DHT para el manejo de sensor DHT11,
temperatura/humedad relativa

#include <DHT_U.h> //librería de sensores unificados de temperatura y
humedad DHT

#include <Wire.h> //protocolo de comunicación I2C, permite comunicarse con
dispositivos I2C/TWI

#include <RTCLib.h> //librería para el funcionamiento del reloj en tiempo
real

#include <LiquidCrystal_I2C.h> // librería para el funcionamiento de LCD
16x2

#include <Adafruit_NeoPixel.h> // libreria para controlar tiras y píxeles
LED basados en un solo cable

/*
*****
*****

* DECLARACIÓN DE VARIABLES PARA LECTURA DE SENSORES, RELOJ, ESCRITURA EN
LOS ACTUADORES E INTERRUPCION

*
*****
*****/

// Sensor de Temperatura y Humedad Relativa DHT11

float tempLec = 0.0f; //variable para tomar el valor de la temperatura
interna del sensor, por defecto en °C

float tempMin = 15.00f; //valor temperatura mínima en °C

float tempMax = 28.00f; //valor temperatura maxima en °C

volatile float tempP = 25.00f; //valor estandar asignado inicialmente para
pruebas de funcionamiento

float humRel = 0.0f; //variable para tomar el valor de la humedad relativa
interna del sensor, por defecto en %

float humRelMin = 50.00f; //valor minimo para la humedad relativa interna
en porcentaje

float humRelMax = 90.00f; //valor maximo para la humedad relativa interna
en porcentaje

volatile float humeRelP = 60.00f; //valor estandar asignado inicialmente
para pruebas de funcionamiento

// Sensor de CO2

int CO2 = A1; //variable para tomar el valor del dióxido de carbono del
sensor, por defecto en ppm
```

```

int CO2Min = 300; //valor mínimo de dióxido de carbono en el ambiente en
ppm
int CO2Max = 1200; //valor máximo de dióxido de carbono en el ambiente en
ppm
volatile int CO2P = 312; //valor estandar asignado inicialmente para
pruebas de funcionamiento

// Sensor de Humedad del Suelo
float humSue = A2; //variable para tomar el valor de la humedad del suelo
del sensor, en voltaje con 0 muy humedo y 1023 seco
float humSuePor; //variable para el sensor de humedad del suelo en
porcentaje
float humSuePorMin = 10; //valor mínimo para la humedad del suelo en
porcentaje
float humSuePorMax = 50; //valor máximo para la humedad del suelo en
porcentaje
volatile float humeSueP = 41; //valor estandar asignado inicialmente para
pruebas de funcionamiento

// Sistema de ventilación
int ventilador1 = 3; //pin del actuador que controla el ventilador 1
int ventilador2 = 4; //pin del actuador que controla el ventilador 2

// Sistema de riego por goteo (bomba de agua)
int bomba_agua = 5; //pin del actuador que controla la bomba de agua para
el sistema de riego por goteo

// Sistema de calefacción
int calefactor = 6; //pin del actuador que controla la calefacción

// Sistema de iluminacion y su respectiva funcion
int luz = 7; //pin del actuador que controla la tira led para el sistema
de iluminacion
#define PINLEDS 8 //pin que controla la secuencia de encendido de la tira
LED
#define TOTALLEDS 38 //total de LEDs en la tira
int j = 0; // variable utilizada para el ciclo for de encendido de LEDs
int azul = 4; //variable que enciende cada 3 LEDs rojos un LED azul
int residuo; //almacena el residuo entre j y azul para encender aquellos
que cumplan la condicion if

// Variables para almacenar datos relacionados a fecha y hora
String ano; //almacena el año
String mes; //almacena el mes
String dia; //almacena el día
String horas; //almacena la hora

```

```

String minutos; //almacena los minutos
String segundos; //almacena los segundos
String fechaActual; //almacena la fecha actual
String horaActual; //almacena la hora actual
int horasInt; //almacena el valor de la hora convirtiendo de string a int

// Variables para la funcion nueva_lectura
int seg = 5; //variable para una nueva lectura en segundos (ajustar segun
se requiera)
int aux = seg; //variable que almacena seg como respaldo para el contador
regreviso
int t_s; //almacena la transformacion de seg en segundos
int t_m; //almacena la transformacion de seg en minutos
int t_h; //almacena la transformacion de seg en horas
int i; //variable utilizada para el ciclo for

// Variables auxiliares
int vere = 0; //sirve para verificar si el RTC DS3231 esta o no conectado
float nt1 = 16; //valor de temperatura minima de guarda para controlar el
ambiente en la funcion temp_co2
float nt2 = 27; //valor de temperatura maxima de guarda para controlar el
ambiente en la funcion temp_co2
float nhr1 = 55; //valor de humedad relativa minima de guarda para controlar
el ambiente en la funcion hum_relativa
float nhr2 = 85; //valor de humedad relativa maxima de guarda para controlar
el ambiente en la funcion hum_relativa
float nco2_1 = 310; //valor de co2 minimo de guarda para controlar el
ambiente en la funcion temp_co2
float nhsue1 = 20; //valor de humedad del suelo minimo de guarda para
controlar la cantidad de liquido en la funcion riego
float nhsue2 = 40; //valor de humedad del suelo maximo de guarda para
controlar la cantidad de liquido en la funcion riego
int t_restante; //almacena el valor de tiempo restante cuando se activa la
ventilacion en la funcion hum_relativa
int incremento = 1; //tiempo en segundos utilizada en la funcion
hum_relativa para el contador regresivo
int tiempo_vent_hr = 300; //tiempo en segundos utilizada para la
ventilacion que despeja el aire humedo en la funcion hum_relativa
/*
*****
*****

* INICIALIZACIÓN DE SENRORES, LCD Y RELOJ (OBJETOS)

*
*****
*****/
RTC_DS3231 RTC; //creación de objeto RTC DS3231 para el funcionamiento del
reloj
DateTime now; //creacion del objeto now para obtener valores de fecha y
hora

```

```

DHT dht (A0, DHT11); // creación de objeto DHT dht para el funcionamiento
del sensor DHT11

LiquidCrystal_I2C lcd(0x27,16,2); //creación de objeto LiquidCrystal_I2C
LCD para el funcionamiento del LCD

Adafruit_NeoPixel pixels(TOTALLEDS, PINLEDS, NEO_GRB + NEO_KHZ800);
//almacena estado y funciones para interactuar con Adafruit NeoPixels

/*
*****
*****

*   SETEO DE FUNCIONES UTILIZADAS POR EL MICROCONTROLADOR

*
*****
*****/

void setup() {
  // Inicialización del sensor de temperatura, humedad relativa y de los
  actuadores

  pinMode(tempLec, INPUT); //configurado como entrada
  pinMode(ventilador1, OUTPUT); //configurado como salida
  pinMode(ventilador2, OUTPUT); //configurado como salida
  digitalWrite(ventilador1, LOW); //se envía una señal de bajo para no
  activar el relay
  digitalWrite(ventilador1, LOW); //se envía una señal de bajo para no
  activar el relay

  // Inicialización del sensor de CO2
  pinMode(CO2, INPUT); //configurado como entrada

  // Inicialización del sensor de humedad del suelo
  pinMode(humSue, INPUT); //configurado como entrada
  pinMode(bomba_agua, OUTPUT); //configurado como salida
  digitalWrite(bomba_agua, LOW); //se envía una señal de bajo para no
  activar el relay

  // Inicialización de la calefacciom
  pinMode(calefactor, OUTPUT); //configurado como salida
  digitalWrite(calefactor, LOW); //se envía una señal de bajo para no
  activar el relay

  // Inicialización iluminacion
  pinMode(luz, OUTPUT); //configurado como salida
  digitalWrite(luz, LOW); //se envía una señal de bajo para no activar el
  relay

  // Inicialización del reloj en tiempo real
  Wire.begin(); //se inicia el puerto I2C

```

```

RTC.begin(); //se Inicia la comunicación con el RTC DS3231
RTC.adjust(DateTime(__DATE__, __TIME__)); // Establece la fecha y hora
//cargar la primera vez sin comentar, posteriormente comentar una vez
establecida la hora

// Inicialización del objeto para adquisición de datos de temperatura y
humedad relativa
dht.begin();

//Inicialización de LCD
lcd.begin(16,2); //inicia LCD de 16 columnas y 2 filas
lcd.backlight(); //encendido de luz led en LCD
lcd.clear(); //limpia LCD
lcd.print("BIENVENIDO"); //imprime un mensaje
lcd.setCursor(0, 1); //ubica el curso en la columna cero, fila 1
lcd.print(" SISTEMA OK ");
delay(10000); //retarde de 10 segundos

//Inicialización del Sistema de Iluminacion
pixels.begin(); // inicialización

// Configuración de los pines para la interrupción de cambio de bajo a
alto para lectura de todos los sensores
// si entran en el horario establecido
attachInterrupt(digitalPinToInterrupt(2), lecturas, RISING);

//Verificación del funcionamiento del reloj de tiempo real
while(vere == 0){ //si la variable vere es cero sigue el ciclo hasta
tener conectado el modulo RTC
    if (RTC.begin()){ //sale del ciclo si trabaja el RTC, vere = 1
        lcd.clear();
        lcd.print("OK!");
        lcd.setCursor(0, 1);
        lcd.print("RTC activo");
        delay(10000);
        vere = 1;
    }else{ //sigue en el ciclo si no trabaja el RTC, vere = 0
        lcd.clear();
        lcd.print("ERROR!");
        lcd.setCursor(0, 1);
        lcd.print("RTC inactivo");
        vere = 0;
    }
}

```

```

        delay(10000);
    }
}

void loop(){
    tempLec = dht.readTemperature(); //lectura del dato de temperatura en C
    del objeto dht, utilizado en la interrupción lecturas()

    humRel = dht.readHumidity(); //lectura del dato de humedad relativa en
    % del objeto dht, utilizado en la interrupción lecturas()

    // RELOJ - Obtención de datos de fecha y hora actuales
    DateTime now = RTC.now();
    ano = now.year();
    mes = now.month();
    dia= now.day();
    horas = now.hour();
    horasInt = horas.toInt();
    minutos = now.minute();
    segundos = now.second();
    fechaActual = ano + "/" + mes + "/" + dia;
    horaActual = horas + ":" + minutos + ":" + segundos;
    lcd.clear();
    lcd.print("FECHA: " + fechaActual );
    lcd.setCursor(0, 1);
    lcd.print("HORA: " + horaActual );

    delay(5000);

    presentacionDatos();

    //se establece las horas de funcionamiento de los sistemas
    if((now.hour() >= 2) && (now.hour() <= 5)){
        lcd.clear();
        lcd.print("HORARIO VALIDO");
        lcd.setCursor(0, 1);
        lcd.print("LUZ OK");
        delay(3000);
        lcd.clear();
        lcd.print("CALEFACCION OK");
        lcd.setCursor(0, 1);
    }
}

```

```

    lcd.print("VENTILACION OK");
    delay(3000);
    //sistemaIluminacion();

    lcd.clear();
    lcd.println("SISTEMA DE ");
    lcd.setCursor(0, 1);
    lcd.print("ILUMINACION ON");
    delay(3000);

    sistemaIluminacion();
    hum_relativa();
}

if (((now.hour() >= 6) && (now.hour() <= 17))) {
    lcd.clear();
    lcd.print("HORARIO VALIDO");
    lcd.setCursor(0, 1);
    lcd.print("VENTILACION OK");
    delay(3000);
    lcd.clear();
    lcd.print("CALEFACCION OK");
    delay(3000);
    temp_co2();
    calefaccion();
    digitalWrite(luz, LOW);
}

if (((now.hour() >= 6) && (now.hour() <= 9)) || (((now.hour() >= 15) &&
(now.hour() <= 16)))) {
    lcd.clear();
    lcd.print("HORARIO VALIDO");
    lcd.setCursor(0, 1);
    lcd.print("RIEGO OK");
    delay(5000);
    riego();
    digitalWrite(luz, LOW);
}

nueva_lectura();
}

```

```

/*
*****
*****

* FUNCIONES PARA EL CONTROL DEL SISTEMA DE TEMPERATURA, HUMEDADES, CO2,
  RIEGO E ILUMINACIÓN

*
*****
*****/

void presentacionDatos(){
    // la siguiente funcion indica en el LCD los valores de temperatura,
    humedad relativa, CO2, humedad del suelo

    lcd.clear();
    lcd.print("Temp "+String(tempP)+" C");
    lcd.setCursor(0, 1);
    lcd.print("HumR "+String(humeRelP)+" %");
    delay(7000);
    lcd.clear();
    lcd.print("CO2 "+String(CO2P)+" ppm");
    lcd.setCursor(0, 1);
    lcd.print("H Suelo "+String(humeSueP)+" %");
    delay(5000);

    lcd.clear();
    lcd.print("ANALIZANDO");
    lcd.setCursor(0, 1);
    lcd.print("AMBIENTE");
    delay(3000);
}

void lecturas(){
    // la siguiente funcion es una interrupcion para analizar mediante
    sensores las variables del ambiente

    // con ello se pueden realizar pruebas de funcionamiento segun se
    requiera manipulandó manualmente si es necesario

    // la temperatura, humedades, CO2 del ambiente para la activacion de
    actuadores

    int valorCO2 = analogRead(CO2); // lectura del dato del sensor CO2
    almacenado en valorCO2

    float hume = analogRead(humSue); //lectura del dato de la humedad del
    suelo

    humSuePor = map(hume, 0, 1023, 100, 0); //conversión de la humedad del
    suelo en %

    tempP = tempLec; //asignacion del valor leido del sensor de temperatura

```



```

    humeRelP = humRel; //asignacion del valor leido del sensor de humedad
relativa
    CO2P = valorCO2; //asignacion del valor leido del sensor de CO2
    humeSueP = humSuePor; //asignacion del valor leido del sensor de humedad
del suelo
}

void temp_co2(){
    //la siguiente funcion permite la regulaci3n de la temperatura y CO2
cuando
    //los valores se encuentran fuera del rango establecido

    if(((tempP > tempMin) && (tempP < tempMax)) && (CO2P > CO2Min)){
        //condicion que verifica la lectura dentro del rango de los valores
para la temperatura y CO2
        lcd.clear();
        lcd.print("TEMP Y CO2 OK!");
        lcd.setCursor(0, 1);
        lcd.print("VENTILACION OFF");
        delay(3000);
    }else if((tempP > tempMax) && (CO2P < CO2Min)){
        //condicion que verifica la lectura fuera del rango de los valores
para la temperatura y CO2
        lcd.clear();
        lcd.print("SE NECESITA");
        lcd.setCursor(0, 1);
        lcd.print("REGULAR AMBIENTE");
        delay(3000);

        lcd.clear();
        lcd.print("TEMP ALTA");
        lcd.setCursor(0, 1);
        lcd.print("CO2 BAJO");
        delay(3000);

        while((tempP > nt2) && (CO2P < nco2_1)){
            //ciclo repetitivo para regular la temperatura y CO2 dentro de los
valores de guarda
            lcd.clear(); // borra el contenido, indica una posici3n (0,0)
            lcd.println("VENTILACION ON");
            digitalWrite(ventilador1, HIGH);
            digitalWrite(ventilador2, HIGH);
            delay(1000);
        }
    }
}

```

```

    }
    digitalWrite(ventilador1, LOW);
    digitalWrite(ventilador2, LOW);
    lcd.clear();
    lcd.print("AMBIENTE REGULADO");
    lcd.setCursor(0, 1);
    lcd.print("VENTILACION  OFF");
    delay(3000);
}else if(tempP > tempMax){
    //condicion que verificar si la temperatura esta fuera del valor maximo
    lcd.clear();
    lcd.print("SE NECESITA");
    lcd.setCursor(0, 1);
    lcd.print("REGULAR AMBIENTE");
    delay(3000);

    lcd.clear();
    lcd.print("TEMPERATURA");
    lcd.setCursor(0, 1);
    lcd.print("ALTA");
    delay(3000);

    while(tempP > nt2){
        //ciclo repetitivo para regular la temperatura hasta el valor de
guarda
        lcd.clear(); // borra el contenido, indica una posición (0,0)
        lcd.println("VENTILACION  ON");
        digitalWrite(ventilador1, HIGH);
        digitalWrite(ventilador2, HIGH);
        delay(1000);
    }

    digitalWrite(ventilador1, LOW);
    digitalWrite(ventilador2, LOW);
    lcd.clear();
    lcd.print("AMBIENTE REGULADO");
    lcd.setCursor(0, 1);
    lcd.print("VENTILACION  OFF");
    delay(3000);
}else if(CO2P < CO2Min){
    //condicion que verifica si el CO2 esta abajo del valor minimo

```

```

lcd.clear();
lcd.print("SE NECESITA");
lcd.setCursor(0, 1);
lcd.print("REGULAR AMBIENTE");
delay(3000);

lcd.clear();
lcd.print("CO2");
lcd.setCursor(0, 1);
lcd.print("BAJO");
delay(3000);

while(CO2P < nco2_1){
    //ciclo repetitivo para regular el CO2 hasta el valor de guarda
    lcd.clear();
    lcd.println("VENTILACION  ON");
    digitalWrite(ventilador1, HIGH);
    digitalWrite(ventilador2, HIGH);
    delay(1000);
}

digitalWrite(ventilador1, LOW);
digitalWrite(ventilador2, LOW);
lcd.clear();
lcd.print("AMBIENTE REGULADO");
lcd.setCursor(0, 1);
lcd.print("VENTILACION  OFF");
delay(3000);
}
}

void hum_relativa(){
    //la siguiente funcion permite la regulaci3n de la humedad relativa
    //cuando los valores se encuentran fuera del rango establecido

    if((humeRelP > humRelMin) && (humeRelP < humRelMax)){
        //condicion que verifica si la humedad relativa se encuentra dentro
del rango establecido
        lcd.clear();
        lcd.print("H RELATIVA OK!");
        lcd.setCursor(0, 1);

```

```

    lcd.print("VENT CALE OFF");
    delay(3000);
}else if(humeRelP > humRelMax){
    //condicion que verifica si la humedad relativa es mayor al valor
    maximo establecido
    lcd.clear();
    lcd.print("SE NECESITA");
    lcd.setCursor(0, 1);
    lcd.print("REGULAR AMBIENTE");
    delay(3000);

    lcd.clear();
    lcd.print("H RELATIVA");
    lcd.setCursor(0, 1);
    lcd.print("ALTA");
    delay(3000);

    while(incremento < tiempo_vent_hr){
        //ciclo repetitivo para regular la humedad relativa hasta el valor
        de guarda adecuado
        t_restante = tiempo_vent_hr - incremento; //almacena el valor maximo
        con el minimo para generar un contador regreviso
        lcd.clear();
        lcd.println("VENTILACION ON");
        lcd.setCursor(0, 1);
        lcd.print(String(t_restante)+ " segundos");
        digitalWrite(ventilador1, HIGH);
        digitalWrite(ventilador2, HIGH);
        delay(1000);
        incremento++; //incrementa su valor en uno para salir del ciclo
    }

    digitalWrite(ventilador1, LOW);
    digitalWrite(ventilador2, LOW);
    lcd.clear();
    lcd.print("VENTILACION");
    lcd.setCursor(0, 1);
    lcd.print("OFF");
    delay(3000);

    while(tempP > nt1){

```

```
    //ciclo repetitivo para regular el descenso de temperatura y humedad
    relativa por su relación inversamente proporcional hasta el valor maximo
    de guarda
```

```
    lcd.clear();
    lcd.println("CALEFACTOR  ON");
    digitalWrite(calefactor, HIGH);
    delay(1000);
}
```

```
digitalWrite(calefactor, LOW);
lcd.clear();
lcd.print("AMBIENTE REGULADO");
lcd.setCursor(0, 1);
lcd.print("CALEFACTOR  OFF");
delay(3000);
```

```
}
```

```
void calefaccion(){
```

```
    //la siguiente funcion permite la regulación de la temperatura
    //cuando los valores se encuentran por debajo del rango establecido
```

```
    if((tempP > tempMin) && (tempP < tempMax)){
```

```
        //condicion que verifica la lectura dentro del rango de los valores
        para la temperatura y CO2
```

```
        lcd.clear();
        lcd.print("TEMP OK!");
        lcd.setCursor(0, 1);
        lcd.print("VENTILACION OFF");
        delay(3000);
```

```
    }else if(tempP < tempMin){
```

```
        //condicion que verificar si la temperatura esta por debajo del valor
        de la temperatura minima
```

```
        lcd.clear();
        lcd.print("SE NECESITA");
        lcd.setCursor(0, 1);
        lcd.print("REGULAR AMBIENTE");
        delay(3000);
```

```
        lcd.clear();
        lcd.print("TEMPERATURA");
        lcd.setCursor(0, 1);
```

```

    lcd.print("BAJA");
    delay(3000);

    while(tempP < nt1){
        //ciclo repetitivo que regula la temperatura hasta el valor de guarda
        minimo
        lcd.clear();
        lcd.println("CALEFACTOR  ON");
        digitalWrite(calefactor, HIGH);
        delay(1000);
    }

    digitalWrite(calefactor, LOW);
    lcd.clear();
    lcd.print("AMBIENTE REGULADO");
    lcd.setCursor(0, 1);
    lcd.print("CALEFACTOR  OFF");
    delay(3000);
}
}

void riego(){
    //la siguiente funcion permite la regulaci3n de la humedad del suelo
    //cuando los valores se encuentran por debajo del rango establecido

    if((humeSueP > humSuePorMin) && (humeSueP < humSuePorMax)){
        //condicion que verifica que el valor se encuentre dentro del rango
        establecido
        lcd.clear();
        lcd.print("HUM SUELO OK!");
        lcd.setCursor(0, 1);
        lcd.print("RIEGO OFF");
        delay(3000);
    }else if(humeSueP < humSuePorMin){
        //condicion que verifica que el valor se encuentra por debajo del rango
        establecido
        lcd.clear();
        lcd.print("SE REQUIERE");
        lcd.setCursor(0, 1);
        lcd.print("RIEGO");
        delay(3000);
        while(humeSueP < nhsuel){

```

```

        //ciclo repetitivo que regula la humedad del suelo hasta el valor de
guarda minimo
        lcd.clear();
        lcd.print("RIEGO ON");
        digitalWrite(bomba_agua, HIGH);
        delay(1000);
    }
    lcd.clear();
    lcd.println("AMBIENTE REGULADO!");
    lcd.setCursor(0, 1);
    lcd.println("BOMBA OFF");
    digitalWrite(bomba_agua, LOW);
    delay(3000);
}
}

void sistemaIluminacion(){
    lcd.clear();
    lcd.println("SISTEMA DE ");
    lcd.setCursor(0, 1);
    lcd.print("ILUMINACION ON");
    delay(3000);

    digitalWrite(luz, HIGH);

    for(j=0;j<TOTALLEDS;j++){
        //Encender led
        pixels.setPixelColor(j, 200, 0, 0); // encendido de LED rojo con un
brillo de 78.4%
        pixels.show(); // muestra los valores aplicados

        residuo = j % azul; //calcula cada cuantos pines encender un LED azul
        if(residuo == 0){
            pixels.setPixelColor(j, 0, 0, 255); // encendido de LED azul con un
brillo de 100%
            pixels.show(); // encendido
        }
    }
}

void nueva_lectura(){

```

```

//funcion que genera una nueva lectura en un tiempo establecido por el
usuario
for(i=seg; i>=0; i--){
  t_h = seg / 3600; //convierte los segundos en horas
  t_m = (seg - (t_h * 3600))/60; //convierte los segundos en minutos
  t_s = seg - ((t_h * 3600) + (t_m * 60)); //almacena los segundos
  seg = seg - 1;
  lcd.clear();
  lcd.print("NUEVA LECTURA EN");
  lcd.setCursor(0, 1);
  lcd.print(String(t_h) + ":" + String(t_m) + ":" + String(t_s));
  delay(1000);
}
segundos = aux; //se almacena los segundos
}

```