



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA DE MECATRÓNICA

**DISEÑO Y CONSTRUCCIÓN DE UN GERMINADOR DE ESQUEJES Y SEMILLAS
DE PETUNIAS Y CICLAMEN**

Trabajo de titulación previo a la obtención del
Título de Ingeniero en Mecatrónica

AUTORES: OSWALDO WLADIMIR CHICAIZA CALERO

TUTOR: CARMEN JOHANNA CELI SANCHEZ

Quito - Ecuador

2024

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Oswaldo Wladimir Chicaiza Calero, con documento N°1718931999, manifiesto que:

Soy, autor y responsable del presente trabajo; por lo que autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 1 de marzo del 2024

Atentamente,



Oswaldo Wladimir Chicaiza Calero

1718931999

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Yo, Oswaldo Wladimir Chicaiza Calero con documento de identificación No: 1718931999, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del dispositivo tecnológico: “Diseño y construcción de un germinador de esquejes y semillas de petunias y ciclamen”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Mecatrónica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Quito, 1 de marzo del año 2024



Oswaldo Wladimir Chicaiza Calero

1718931999

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Carmen Johanna Celi Sánchez, con documento de identificación N°: 1717437808, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO Y CONSTRUCCIÓN DE UN GERMINADOR DE ESQUEJES Y SEMILLAS DE PETUNIAS Y CICLAMEN, realizado por Oswaldo Wladimir Chicaiza Calero, con documento de identificación N°: 1718931999, obteniendo como resultado final el trabajo de titulación bajo la opción Dispositivo tecnológico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 1 de marzo del año 2024

Atentamente,



Ing. Carmen Johanna Celi Sánchez. Mgtr

1717437808

DEDICATORIA

A Dios por darme vida cada mañana, por haberme dado dones y talentos para usarlos en bien de los demás.

Mi tesis se la dedico con todo mi amor a mis padres Oswaldo y Consuelo por todo el sacrificio y esfuerzo, por darme una carrera para mi futuro y por creer en mi capacidad, aunque no soy perfecto ni el mejor hijo ellos siempre han depositado su confianza y sueños en mí, prometo esforzarme para ser mejor cada día y que se sientan orgullosos de mí.

A mi tía Patricia quien siempre me ha apoyado en diferentes aspectos de mi vida, que ha sido como una madre para mí, con sus consejos y cuidados para ser una mejor persona.

A mis amigos que siempre me han dado palabras de apoyo y con sus oraciones y buenos deseos me han subido el ánimo en mis peores momentos, gracias por compartir alegrías y tristezas durante estos años.

Oswaldo Wladimir Chicaiza Calero

AGRADECIMIENTOS

Gracias a Dios por permitirme tener y disfrutar de mi familia, gracias a mi familia por apoyarme en las decisiones que he tomado a lo largo de estos años y que por ustedes todo esto no sería posible. Gracias por creer en mí y gracias a Dios por permitirme vivir y disfrutar de cada día.

No ha sido sencillo el camino hasta ahora, pero es gracias a su amor, inmensa bondad y apoyo que lo han hecho más fácil. Les agradezco infinitamente y algún día voy a pagarles todo lo bueno que han hecho por mí, y hago presente mi gran amor hacia ustedes, mi familia.

Oswaldo Wladimir Chicaiza Calero

ÍNDICE GENERAL

DEDICATORIA.....	5
AGRADECIMIENTOS	6
ÍNDICE DE TABLAS	11
ÍNDICE DE FIGURAS	12
RESUMEN.....	15
ABSTRACT	16
INTRODUCCIÓN	17
Antecedentes	18
Planteamiento del Problema.....	19
Justificación.....	20
Objetivos.....	20
Objetivo general.....	20
Objetivos específicos	20
Capitulo I.....	21
Conceptos fundamentales.....	21
1.1 Germinación.....	21
1.2 Geminación automatizada.....	21
1.3 Agricultura ornamental.	22
1.4 Esquejes de petunia.....	23
1.5 Semilla de ciclamen.	23
1.6 Variables clave en la germinación de esquejes y semillas.....	24

1.7	Tecnologías y dispositivos utilizados en la germinación de esquejes y semillas.	24
1.7.1	Sistema de germinación automática	24
1.8	Componentes del diseño.	25
1.8.1	Arduino MEGA	26
1.8.2	Sensor de temperatura Dht11	26
1.8.3	Sensor de humedad	25
1.8.4	PWM (modulación por ancho o de pulso)	26
1.8.5	Control difuso	26
1.8.6	HMI	26
2.	Capitulo II	29
2.1	Análisis del diseño	29
2.2	Diseño y construcción	29
2.3	Diseño del soporte del germinador	29
2.4	Análisis estructural de la base	30
2.4.1	Tensión de von mises	31
2.5	Cálculo de fuerzas y momentos	31
2.6	Cálculo del factor de seguridad de la estructura	32
2.7	Diseño de la estructura del germinador	33
2.8	Diseño de las paredes del germinador	34
2.9	Análisis estructural del germinador	35
2.10	Tensión de von mises	36
2.11	Cálculo de fuerzas y momentos	36

2.12	Cálculo del factor de seguridad de la estructura	36
2.13	Cálculo de los remaches.....	37
2.14	Diseño del sistema de riego.....	38
2.15	Diseño de la fuente de calor	39
2.16	Diseño del sistema de ventilación	39
2.17	Bandejas de germinación	40
3.	CAPÍTULO III.....	40
3.1	Componentes a implementar.....	40
3.1.1	Puente H.....	40
3.1.2	Sensor de temperatura dht11	40
3.1.3	Modulo DIMMER de Arduino AC 220V/2 ^a	41
3.1.4	Sensor de humedad FC-28.....	41
3.1.5	Bomba de agua	42
3.1.6	Ventilador de 12 v	42
3.1.7	Pantalla HMI Arduino	42
3.1.8	Manguera de polietileno para riego	43
3.1.9	Elementos para la estructura del germinador.....	43
3.2	DISEÑO ELECTRICO Y ELECTRONICO	44
3.2.1	Diagrama de conexión para el módulo DIMMER.....	44
3.2.2	Diagrama de conexión para bomba de agua	45
3.2.3	Diagrama de conexión para el sensor de humedad.....	45
3.2.4	Diseño electrónico	46
3.2.5	Diagrama de flujo del germinador.....	48
	Capitulo IV.....	53

4.	Análisis y resultados	53
4.1	Programación en Arduino	53
4.1.1	Prueba del control de temperatura	53
4.2	Construcción finalizada del dispositivo	53
4.3	Análisis costos de elaboración	58
4.3.1	Análisis económico de la maquina	58
4.3.2	Construcción eléctrica y de control	59
4.4	Mano de obra.....	60
4.5	Costo total de la construcción de la maquina.....	60
5.	REFERENCIAS.....	64
6.	ANEXOS.....	68
Anexo 1	Etapas de construcción de la estructura.....	68
Anexo 2	Corte de las vigas de madera.....	68
Anexo 3	Ensamble y pintado de la base.....	68
Anexo 4	Cortes conforme las dimensiones especificadas.....	69
Anexo 5	Ensamble del invernadero	69
Anexo 6	Tablero para los componentes electrónicos.....	69
Anexo 7	Prueba de sensores y actuadores.....	70
Anexo 8	Conexiones de control y potencia	70
Anexo 9	Puesta en marcha	70
Anexo 10	Dimensiones generales	71
Anexo 11	Partes de la estructura.....	71
Anexo 12	Código implementado para la programación del dispositivo.....	76

ÍNDICE DE TABLAS

Tabla 1. Sistema agrícola automáticos	25
Tabla 2. Resultados de desplazamiento	55
Tabla 3. Valores obtenidos para el factor de seguridad.....	56
Tabla 4. Proceso de germinación sin controlador.....	56
Tabla 5. Controlador Fuzzy	56
Tabla 6. Balance económico de infraestructura.....	58
Tabla 7. Balance económico de los componentes electrónicos	59
Tabla 8. Presupuesto de la mano de obra	60
Tabla 9. Costo total.....	60

ÍNDICE DE FIGURAS

Figura 1. Proceso de germinación [12].	21
Figura 2. Germinadores automatizados. Autor propio.	22
Figura 3. Agricultura ornamental.	23
Figura 4. Semilla ciclamen [16].	23
Figura 5. Placa de Arduino MEGA.	26
Figura 6. Sensor de temperatura.	26
Figura 7. Modulación por ancho de pulso.	27
Figura 8. Función de membresía.	28
Figura 9. Diseño de soporte 1	29
Figura 10. Diseño de soporte 2	29
Figura 11. Diseño de soporte 3	30
Figura 12. Diseño de soporte 4	30
Figura 13. Análisis de la base.	30
Figura 14. análisis en software inventor.	31
Figura 15. Tensión de VON MISES.	31
Figura 16. Cálculo de la fuerza.	31
Figura 17. Cálculo de seguridad.	32
Figura 18. Resistencia usando el Software.	32
Figura 19. Hallazgo de factor seguridad 13.72.	33
Figura 20. Diseño de estructura del Germinador.	34
Figura 21. Parte frontal del diseño del Germinador.	34
Figura 22. Estructura general del área del Germinador.	34
Figura 23. Diseño del interior del germinador.	35
Figura 24. Análisis de la estructura del germinador.	35

Figura 25. Análisis con software INVENTOR.	35
Figura 26. Tensión de Von Mises.	36
Figura 27. Cálculo de fuerzas y momentos.	36
Figura 28. Factor de seguridad de 5.18.	37
Figura 29. Cálculo de los remaches.	37
Figura 30. Diseño de riego para germinador.	39
Figura 31. Soporte y fuente de calor	39
Figura 32. Ventilador para invernadero	39
Figura 33. Bandeja de germinación de las semillas	40
Figura 34. Componente Puente H [27].	40
Figura 35. Módulo DIMMER AC 220V/2A [29].	41
Figura 36. Sensor de humedad FC28 [30].	41
Figura 37. Bomba de agua [31].....	42
Figura 38. Ventilador o cooler [32].....	42
Figura 39. Pantalla HMI Arduino[33].....	43
Figura 40. Manguera para sistema de riego [34].....	43
Figura 41. Componentes para la estructura física del germinador.	43
Figura 42. conexiones del módulo DIMMER.....	44
Figura 43. Parámetro de conexión del módulo DIMMER.	44
Figura 44. conexión de bomba de agua al IRF520N.....	45
Figura 45. Conexión sensor de tempertatura.....	45
Figura 46. Diseño esquemático de circuito germinador.....	46
Figura 47. Programa Nextion para HMI	47
Figura 48. Pestaña de petunias	47
Figura 49. Pestaña de ciclamen.....	48

Figura 50. diagrama de flujo de germinador automático con Arduino.	49
Figura 51. Funciones de membresía Temperatura	50
Figura 52. Funciones de membresía Humedad del suelo.....	50
Figura 53. Mecanismo de inferencia (reglas).....	51
Figura 54. Funciones de membresía para la potencia del foco	51
Figura 55. Funciones de membresía para el riego de la bomba	52
Figura 56. Funciones de membresía para la ventilación	52
Figura 57. Vista lateral, superior y frontal.	53
Figura 58. Puesta de esquejes y semillas en semilleros.	53
Figura 59. Pantalla HMI en funcionamiento.....	54
Figura 60. Puesta en marcha del dispositivo.	54
Figura 61. Esquejes de petunia enraizados.....	55
Figura 62. Semillas de ciclamen germinadas	55
Figura 63. Comparación tiempo de germinación.....	57
Figura 64. Porcentaje de ahorro	58

RESUMEN

El presente trabajo de titulación se enfoca en el diseño y construcción de un germinador de esquejes y semillas de petunias y ciclamen. Se inició con un estudio previo de las actuales formas de germinación enfocada a plantas ornamentales, tomando en cuenta el diseño de la estructura, la temperatura, humedad y ventilación, todo eso con el fin de obtener las características adecuadas para el diseño de este proyecto. Una vez identificadas las diferentes alternativas, se procedió al diseño de todos los componentes que forman la máquina, incluyendo los sistemas de ventilación, riego y temperatura. Se fabricaron las partes que componen la estructura de la máquina usando un software de diseño mecánico Inventor, para luego proceder con el ensamblaje mecánico, eléctrico y de control. Se utilizó la lógica difusa como herramienta de control para todas las variables del proceso y se realizó la programación de una Human Machine Interface usando Nextion y Arduino, donde se especifica como va a trabajar el sistema dependiendo de la semilla que se seleccione dentro de la pantalla y así monitorear el sistema incluyendo sus respectivas entradas y salidas. Esta interfaz de usuario permitirá controlar la temperatura, humedad y ventilación, supervisando su funcionamiento en tiempo real.

Finalmente, se realizan diversas pruebas de funcionamiento comparando con métodos tradicionales de germinación de semillas. Se corrigen los errores que afectan los resultados del proceso de germinación, con la finalidad de garantizar el correcto funcionamiento del sistema y obtener un proyecto innovador en el ámbito de la floricultura.

ABSTRACT

The present graduation project focuses on the design and construction of a germinator for petunia and cyclamen cuttings and seeds. It began with a preliminary study of current germination methods focused on ornamental plants, taking into account the design of the structure, temperature, humidity, and ventilation, all aimed at obtaining the suitable characteristics for this project's design. Once different alternatives were identified, the design of all machine components was carried out, including ventilation, irrigation, and temperature systems.

The parts composing the machine's structure were manufactured using mechanical design software such as Inventor. Subsequently, mechanical, electrical, and control assembly took place. Additionally, programming of the HMI (Human Machine Interface) control was performed using Nextion and Arduino, specifying how the system will work depending on the selected seed on the screen, ensuring better control over the system, including its respective inputs and outputs. This user interface will allow controlling the system's temperature, humidity, and ventilation while monitoring its real-time operation.

Finally, various functional tests were conducted, comparing them with traditional seed germination methods. Errors that could affect the final germination process results will be corrected. All this is done to ensure the proper functioning of the system and achieve an innovative project in the field of floriculture.

INTRODUCCIÓN

La mecatrónica es una disciplina que combina la ingeniería mecánica, electrónica y los sistemas de control para diseñar y operar sistemas automatizados complejos, el presente desarrollo investigativo se basa precisamente en el diseño y creación de un prototipo utilizando componentes electrónicos y sistemas de control como lo es Arduino, con el propósito de diseñar un germinador de esquejes y semillas de petunias y ciclamen, para ser implementado en una fuente de emprendimiento a nivel de germinación y cultivos de plantas ornamentales, conocidas también como plantas de adornos utilizadas para decoraciones en diferentes espacios y ocasiones. Este tipo de plantas se han convertido entre las principales para su compra dentro del mercado floricultor, ya que a partir de la pandemia por el COVID muchas personas perdieron su trabajo, por lo que, buscan como emprender e innovar en conjunto con la tecnología.

Las plantas ornamentales son cualquier tipo de plantas que permitan embellecer un ambiente arquitectónico o momentos sociales, esto explica, Acosta[1], y según Rivera [2], los germinadores son herramientas que contribuyen al proceso de florecimiento de la planta al mantener las semillas en un ambiente adecuado de humedad. La finalidad de sumergir las semillas es lograr una hidratación óptima para facilitar su germinación. Tomando en cuenta a los profesionales de Agro [3], un germinador de semillas, también conocido como semillero de germinación, es un contenedor utilizado para cultivar semillas y promover su crecimiento. Teniendo claro las diferencias de un germinador, el objetivo del proyecto es mediante diferentes tipos de dispositivos electrónicos como: placas electrónicas (Arduino), sensores de temperatura, humedad, motores entre otros, es diseñar un sistema que permita realizar el proceso y monitoreo de la germinación de semillas para plantas ornamentales como una fuente de emprendimiento personal o propuesta.

Los profesionales en mecatrónica deben tener una comprensión profunda de los sistemas mecánicos, electrónicos y de control, así como habilidades en programación y diseño de software. La capacidad de trabajar en equipo y la capacidad de resolver problemas complejos son esenciales para tener éxito en este campo, como afirma el proyecto educativo de Ingeniería en Mecatrónica[4], un ingeniero mecatrónico es una persona competente para examinar su entorno, posee habilidades creativas y demostrativas de pensamiento crítico y

emocional que le permite idear soluciones tecnológicas innovadoras a problemas en diversos argumentos, y también crear nuevo conocimiento a través de la investigación y el desarrollo de nuevos productos tecnológicos. La investigación tendrá un enfoque mixto el cual permitirá recopilar datos importantes tanto, hipótesis y datos estadísticos, de tipo descriptivo y explicativo, la documentación contará con 4 capítulos, parte introductoria, teórica, metodología y resultados, conclusiones y recomendaciones.

ANTECEDENTES

La germinación es un proceso que se da de forma natural y también en la floricultura, por lo cual, este proceso se lo divide en 3 fases como lo explica el agro comercial Mida[5] la fase de hidratación, la fase de germinación y la fase creciente, menciona también que existen 2 tipos de germinación la epigea e hipogea donde la diferencia es que los cotiledones en la primera quedan encima y en la segunda bajo tierra. Así como existe la diferencia en la germinación, también existe en el método o forma en que se lo realiza, de la forma tradicional o manual y la forma automática, implementando el sistema de riego, monitoreos mediante sensores de humedad y temperatura, los cuales con otros componentes se pueden realizar prototipos a grandes escalas para la germinación de la floricultura, aplicando la tecnología como objeto innovador, como afirma Romero[6] en su prototipo germinador de lechuga, el cual permite aislar las plántulas del entorno, lo que evita que depredadores puedan afectar su producción. Además, este prototipo demuestra la viabilidad de combinar tecnología y agricultura, y aunque su construcción y adaptación fue desafiante, fue posible aplicar las técnicas de control aprendidas en su desarrollo. El MAG[7] menciona lo importante que es la innovación y el uso de la tecnología en el sector agrícola, ya que estas herramientas son capaces de automatizar y reducir la mano de obra para actividades de riego, monitoreo entre otros que se lo realizaban de forma manual.

La tecnología hoy en día permite adaptar componentes para automatizar proceso que se los realizaban anteriormente de la forma tradicional, en el caso de Hernández [8] los germinadores automatizados son herramientas importantes para optimizar los procesos agrícolas. Sin embargo, su costo puede ser alto y puede ser necesario diseñar, construir y operar dispositivos a precios accesibles para pequeños productores. El objetivo de este trabajo fue diseñar, construir y verificar un prototipo de germinador de bajo costo utilizando

materiales disponibles en la región. Este germinador es capaz de controlar automáticamente la temperatura y el fotoperiodo mediante circuitos W1401 y Tem-8 48386. Por lo cual otra forma de implementar componentes electrónicos es mediante la tecnología Arduino la cual es muy asequible para el desarrollo de prototipos utilizados e implementado en muchos sectores de la agricultura, como el sector florícola ornamental, agrícola industrial entre otros, Martínez [8] el Arduino es una plataforma que permite la creación de prototipos escalables de control combinando software y hardware, con el objetivo de automatizar actividades cotidianas. En investigaciones, se puede utilizar la tarjeta Arduino para adquirir mediciones de temperatura y humedad dentro de una cámara de germinación, las cuales son los parámetros esenciales para activar los actuadores necesarios como ventiladores, resistencias térmicas e iluminación.

Las tecnologías automáticas dentro de diferentes áreas cada día se van integrando más a los procesos agrícola y se los reemplaza por métodos automáticos, es el caso de Rivera [9] el cual implementó un sistema germinador para el proceso de semillas orientado al consumo humano y es así que mediante la tecnología se puede desarrollar prototipos adaptados para las necesidades o para generar ingresos económicos, como el desarrollo de un sistema germinador de plantas ornamentales para el desarrollo de un emprendimiento, ya que este tipo de plantas se las utiliza para decorar espacios o áreas de cualquier edificio o zona del hogar.

PLANTEAMIENTO DEL PROBLEMA

Debido a la pandemia del COVID-19, muchas personas han buscado alternativas de emprendimiento para generar ingresos adicionales en sus hogares. La jardinería y el cultivo de plantas son opciones populares en este sentido. Sin embargo, el proceso de germinación de esquejes y semillas puede ser complicado y requiere un monitoreo constante, lo que limita la capacidad de producción y tiempo.

Planteamiento de problema: ¿Cómo se puede diseñar y construir un germinador automático de esquejes y semillas de petunias y ciclamen mediante el uso de tecnología de Arduino para mejorar la eficiencia de producción en un emprendimiento familiar después de la pandemia del COVID-19?

JUSTIFICACIÓN

En los inicios de pandemia se notó un claro despido y cierre de negocios a nivel mundial, según el Banco Mundial[10] en el 2020 la economía se redujo en un 5% donde los pocos ingresos enviaron a muchas personas a vivir bajo pobreza extrema. Con el paso del tiempo y la pandemia se ha notado que ha surgido una gran cantidad de micro empresarios o emprendedores queriendo salir adelante, la UNESCO[11] menciona que post pandemia los emprendedores se han innovado con diferentes métodos y líneas comerciales para mejorar su calidad de vida y salud económica. Por lo cual el presente desarrollo investigativo se basa en integrar todas las variables como: innovación, tecnología, floricultura en un solo prototipo como lo es el diseño de un sistema de germinación automático de plantas ornamentales para un emprendimiento familiar, esto utilizando la tecnología Arduino como instrumentos de desarrollo.

Objetivos

Objetivo general

- Desarrollar un germinador de esquejes y semillas de petunias y ciclamen asistido por una tarjeta de desarrollo electrónica.

Objetivos específicos

- Examinar la germinación de esquejes y semillas ornamentales para la obtención de datos y variables necesarias para el diseño del prototipo, mediante la investigación sobre un proceso agrónomo profesional.
- Construir un dispositivo que permita controlar la temperatura y la humedad para acelerar el proceso de germinación de esquejes y semillas, utilizando un sistema de riego, ventilación y calefacción mediante un control de lógica difusa.
- Automatizar el dispositivo para la supervisión de las diferentes variables durante el proceso de germinado, mediante una tarjeta de desarrollo electrónica y un HMI.
- Realizar las pruebas respectivas para corregir errores, verificar los resultados y el funcionamiento del sistema.

CAPITULO I

El presente capítulo menciona los componentes, procesos y materiales sobre el tema a desarrollar combinando la tecnología y los procesos florícolas para mejorar, automatizar la germinación de plantas, su riego y cuidados que se deben dar frecuentemente a este tipo de germinadores automáticos utilizando la tecnología de Arduino.

CONCEPTOS FUNDAMENTALES

1.1 Germinación

Como se observa en la Figura 1, la germinación se refiere al proceso en el cual una semilla o espóra comienza a crecer y desarrollarse en una nueva planta. Esta es la etapa más importante en el ciclo de vida de muchas plantas, puesto que es el momento en que la planta comienza a establecerse y enraizarse en el suelo. Durante la germinación, la semilla o la espóra empieza a absorber agua y nutrientes del suelo, y el embrión dentro de la semilla intenta a crecer y desarrollarse en una nueva planta. La humedad y luz pueden afectar el proceso de germinación [12].

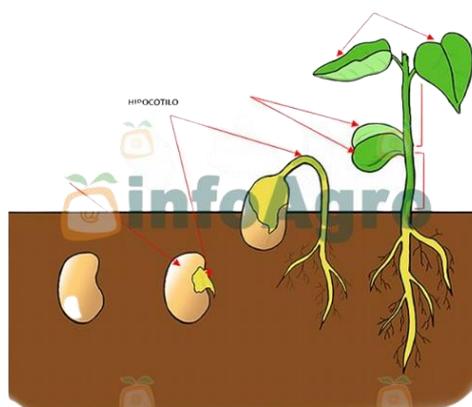


Figura 1. Proceso de germinación [12].

1.2 Geminación automatizada

Según Martínez et al. [13], la germinación automatizada se refiere al proceso de controlar automáticamente las condiciones ambientales y de riego para mejorar el crecimiento y desarrollo durante la etapa de germinación. Esto se logra mediante sistemas automatizados de riego y control de clima en invernaderos o cámaras de germinación. La germinación automatizada puede permitir un mejor control sobre las variables ambientales y reducir la

mano de obra en la etapa de germinación de las plantas. Además, puede aumentar la eficiencia en la germinación de plantas, como se observa en el Figura 2.



Figura 2. Germinadores automatizados. Autor propio.

1.3 Agricultura ornamental.

Agricultura ornamental se refiere a la producción y comercialización de plantas ornamentales, que son cultivadas por sus atractivas características estéticas, como las flores, hojas, tallos y frutos, para su uso en jardines, interiores u otros fines decorativos [14]. Dentro de la producción de plantas ornamentales también puede implicar la utilización de técnicas avanzadas como la micropropagación y la biotecnología para mejorar la calidad y cantidad de la producción. Las plantas ornamentales se utilizan no sólo para embellecer jardines y parques, sino también para la decoración de eventos, como bodas, festivales y otras celebraciones. La agricultura ornamental es un sector en crecimiento que ofrece oportunidades para la innovación y la diversificación de productos agrícolas, ver figura 3.



Figura 3. Agricultura ornamental.

1.4 Esquejes de petunia.

Los esquejes de petunias son pequeñas porciones de tallos y hojas que se cortan de una planta madre y se plantan en un medio de cultivo para que desarrollen sus propias raíces y se conviertan en nuevas plantas [15]. Según Kros Agro [15] los esquejes se cortan preferiblemente en la primavera o en el otoño, y se seleccionan ramas de la planta que estén en buenas condiciones y libres de enfermedades. Una vez seleccionados, los esquejes se cortan y se quitan las hojas de la parte inferior, y luego se plantan en una mezcla de tierra y arena para que enraícen. Las petunias se pueden propagar por semillas o por esquejes, pero los esquejes son una forma más rápida y efectiva de hacer germinar la planta.

1.5 Semilla de ciclamen.

Las semillas de ciclamen se obtienen de la flor, el ciclamen es una planta ornamental que se cultiva por sus flores atractivas y coloridas, estas semillas se pueden usar para propagar nuevas plantas de ciclamen a través de la siembra. Sin embargo, Baessler [16] menciona que las semillas son pequeñas y se pueden encontrar dentro de una cápsula en la base de la flor después de que se ha marchitado. Una vez sembradas, las semillas germinan y crecen para formar nuevas plantas de ciclamen, ver figura 4.



Figura 4. Semilla ciclamen [16].

1.6 Variables clave en la germinación de esquejes y semillas.

Existen algunas variables clave que afectan el proceso de germinación tanto de esquejes como de semillas, como seguimiento a esta actividad Reyna et al. [17] además de los factores principales como temperatura, humedad y oxígeno, también se sabe que la luz es un factor que puede influir en el proceso de germinación de las semillas. A continuación, se mencionan algunas de las variables más importantes a considerar para luego determinar los componentes electrónicos, materiales y demás para la construcción de un germinador automatizado:

Temperatura: La temperatura es un factor crítico para la germinación. Cada planta tiene una temperatura óptima específica para la germinación exitosa. En general, la mayoría de las plantas prefieren temperaturas entre 20-30 °C, pero puede variar según la especie [17].

Temperaturas óptimas para el cultivo de petunias: para enraizar petunias las temperaturas óptimas deben ser de entre 20-25°C, sin embargo, las petunias pueden resistir hasta temperaturas de 35°C.

Temperaturas óptimas para el cultivo de ciclamen: para enraizar semillas de ciclamen es necesario temperaturas de entre 18-22°C.

Humedad: La humedad es esencial para la germinación, ya que ayuda a activar las enzimas y procesos metabólicos que conducen al crecimiento de las raíces y brotes. Por otra parte, Parent [18] el sustrato o el medio de germinación debe mantenerse húmedo, pero no empapado, ya que el exceso de humedad puede propiciar el desarrollo de enfermedades fúngicas.

Luz: La luz juega un papel importante en la germinación de algunas semillas ornamentales como lo son las petunias y ciclamen es por eso que estas necesitan un promedio de 4 y 5 horas de luz a diario. A pesar que pueden crecer en sombra parcial el resultado no viene siendo el mismo puesto que las plantas terminarían con menos flores [19].

Aireación: Las semillas y esquejes requieren una buena circulación de aire para evitar la humedad excesiva y además prevenir la aparición de enfermedades. Es importante proporcionar una ventilación adecuada en el área de germinación.

1.7 Tecnologías y dispositivos utilizados en la germinación de esquejes y semillas.

1.7.1 Sistema de germinación automática

Los sistemas de germinación automático son una herramienta útil para mejorar la germinación de semillas y esquejes. Estos sistemas pueden variar en complejidad y costo,

desde sistemas de riego automático simples hasta sistemas avanzados de control ambiental con tecnología de iluminación y calefacción sofisticada [13]. En la actualidad y con el avance tecnológico muchos de los procesos agrícolas han dejado la forma tradicional de realizarlos para ahora ser automatizados. A continuación, se nombra algunos de los diferentes sistemas automatizados en el área de la agricultura en la Tabla 1:

Tabla 1. Sistema agrícola automáticos

Sistemas agrícolas automatizados	
Sistemas de riego automatizado	Estos sistemas se encargan de distribuir el agua de forma automática a las plantas, basándose en la humedad del suelo o en otros factores como la temperatura y la radiación solar.
Sistemas de cultivo en invernaderos	Los sistemas de cultivo en invernadero pueden estar automatizados en diferentes áreas, como el control de la temperatura, la iluminación y el riego, para maximizar la productividad de los cultivos.
Sistemas de siembra	Estos sistemas pueden automatizar el proceso de siembra, reduciendo la necesidad de trabajo manual y aumentando la precisión y eficiencia en la siembra.
Sistemas de monitoreo de cultivos	Estos sistemas utilizan tecnología avanzada para monitorear los cultivos y proporcionar información importante sobre los niveles de humedad del suelo, el crecimiento de las plantas y otros factores importantes.
Sistemas de control de malezas	Estos sistemas utilizan herramientas de inteligencia artificial y aprendizaje automático para identificar y eliminar las malezas de forma automática, reduciendo la necesidad de trabajo manual y aumentando la eficiencia en el control de malezas.

1.8 Componentes del diseño.

Dentro de este enunciado se conceptualizan los componentes necesarios para el diseño el sistema automatizado para la germinación de plantas ornamentales. A continuación, se conceptualizan los principales componentes electrónicos a utilizar:

1.8.1 Arduino MEGA

Este tipo de tecnología es de código libre, es decir de libre acceso para desarrollar componentes o prototipos de sistemas robotizados, mecanizados o automatizados, este tipo de tecnología al ser de fácil acceso ha permitido desarrollar un sin número de prototipos aplicados a las actividades agrícolas, de seguridad y robótica, en la mecatrónica se realizan procesos mecánicos, electrónicos y de control por lo cual este tipo de herramienta o placas de circuito son esenciales para desarrollar los conocimientos en el área de la mecatrónica, ver figura 5 [22].



Figura 5. Placa de Arduino MEGA.

1.8.2 Sensor de temperatura Dht11.

Son elementos que permiten medir temperatura a través de una señal eléctrica específica, y se clasifican como componentes electrónicos. Esta señal puede ser enviada mediante la variación de la resistencia eléctrica del sensor o de manera directa. Además, también son conocidos como sensores de calor, sondas térmicas o detectores de calor. Estos dispositivos encuentran diversos usos, incluyendo el control de circuitos, ver figura 6 [23].

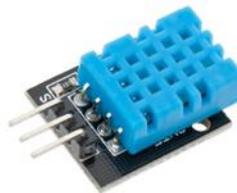


Figura 6. Sensor de temperatura.

1.8.3 PWM (modulación por ancho o de pulso)

Según Guerra [25] PWM significa Modulación por Ancho de Pulso, se utiliza para controlar la cantidad promedio de potencia entregada por una señal eléctrica. PWM trabaja al encender y apagar rápidamente una señal para generar una forma de onda de pulso con un ancho

variable, ver Figura 7.

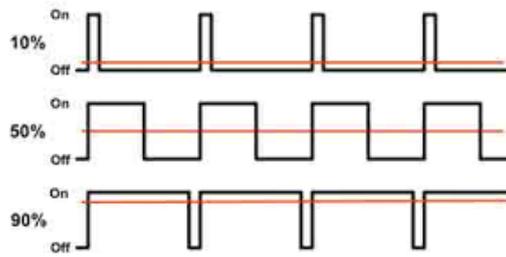


Figura 7. Modulación por ancho de pulso

1.9 Sistema de control difuso

Según Téllez [36] un controlador con lógica difusa es bastante utilizado dentro de la industria puesto que esta clase de control permite capturar conocimientos imprecisos, de expertos en el área a controlar o automatizar ya que al ser un controlador simple y fácil de implementar no requiere obligadamente de un modelo matemático para ejecutarlo, además de ser un sistema robusto.

Al igual que distintos controladores de procesos, esta toma los valores de las variables de entrada y después de un proceso actúa sobre las diferentes salidas para tomar control sobre la planta o sistema.

Cuando se utiliza este tipo de controlador, lo que se procesa ya no son ecuaciones o valores numéricos sino, reglas tales como:

*“SI la temperatura es fría **ENTONCES** aumente la potencia del foco”*

Este tipo de lógica es similar al de una persona ya que toma decisiones mediante el uso de palabras como subir, bajar etc.

1.9.1 Fuzzificación

En esta primera etapa se calcula los grados de pertenencia en las variables del sistema a cada conjunto Fuzzy inputs mediante variables lingüísticas como: *muy baja*, *baja*, *fresco*, *alta* y *muy alta* [37]. Cada una de las variables lingüísticas debe tener asignada su función de membresía, la cual es la que le otorga un valor a la variable.

En la Figura 8 se denota la *función de membresía* para un *Fuzzy input baja*, y un valor dado de 20 grados centígrados.

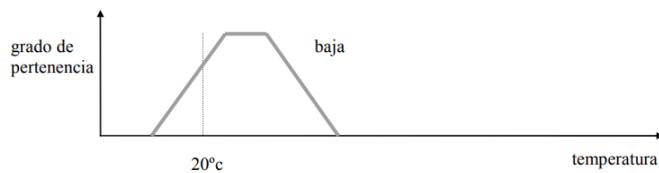


Figura 8. Función de membresía

1.9.2 Evaluación de reglas

Según Gómez [36] la principal función de la base de reglas del controlador Fuzzy es la de representar, de manera precisa, la acción de control con reglas de la forma:

IF (estado de proceso) THEN (salida de control)

El IF es el antecedente y el THEN es el consecuente. Este proceso se hace para cada regla del sistema. Así como en cada una de las variables de entrada, para cada una de las variables de salida le corresponde una función de membresía, representada de forma lingüística. Donde cada uno es un Fuzzy output como puede ser: *subirlo, mantenerlo, bajarlo*.

1.9.3 Defuzzificación

Según Téllez [37] la salida que se genera en el mecanismo de inferencia es una salida difusa la cual no puede ser interpretada por los actuadores, por lo cual se debe transformar la salida difusa a un valor numérico. Este proceso lo realiza el defuzzificador el cual se encarga de transformar el conjunto de valores difusos de salida a un valor interpretable por los elementos de salida del proceso a controlar.

2. CAPITULO II

2.1 Análisis del diseño

En este capítulo se presenta el proceso de construcción del germinador, el cual está técnicamente estructurado, utilizando elementos mixtos como madera y demás para el armado y estructura del área, sistema que más adelante se implementará con el diseño electrónico y los componentes de software y hardware de Arduino.

2.2 Diseño y construcción

Se presenta el diseño CAD de la máquina germinadora, así también los cálculos para su diseño y construcción. Para un mejor análisis se comprobará el diseño en INVENTOR con el fin de corregir los puntos más inestables y así proceder a la construcción.

2.3 Diseño soporte del germinador

Se uso madera de pino de 15 mm de espesor con la cual se construiría la base de 400x500x100 mm, ver Figura 9.

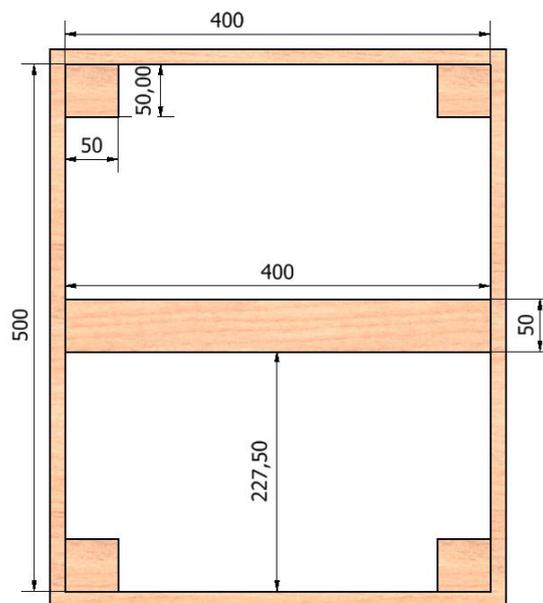


Figura 9. Diseño de soporte 1

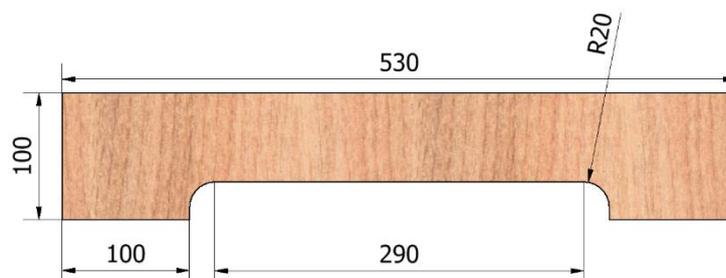


Figura 10. Diseño de soporte 2

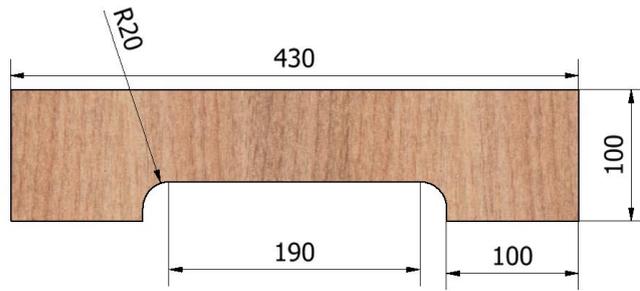


Figura 11. Diseño de soporte 3

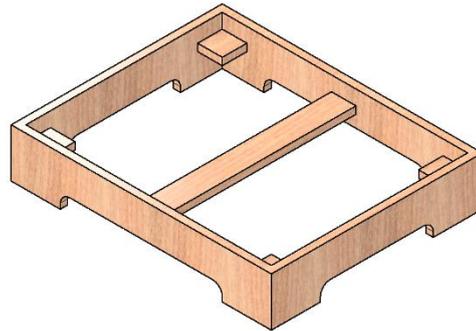


Figura 12. Diseño de soporte 4

2.4 Análisis estructural de la base

La base del germinador deberá soportar los 2kg de la estructura más el peso que tendrá dentro de ella que son exactamente 3kg. Por lo tanto, la base deberá soportar 5kg de peso, ver Figura 13.

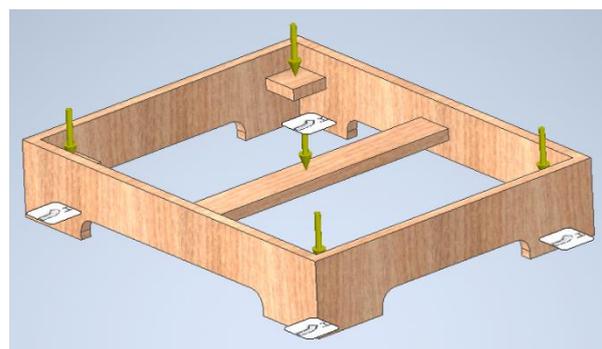


Figura 13. Análisis de la base.

Se hizo el análisis de la estructura utilizando el software INVENTOR, obteniendo así un desplazamiento en la cara inferior de la estructura de 0.06892 mm

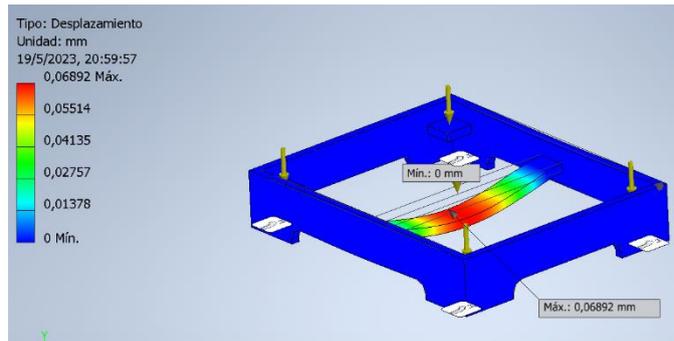


Figura 14. analisis en software inventor.

2.4.1 Tension de von mises

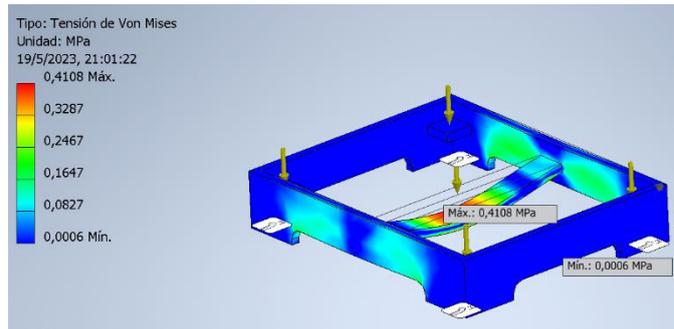


Figura 15. Tension de VON MISES.

2.5 Calculo de fuerzas y momentos

Para realizar estos calculos se utilizo el software DSN Winbeam que se basa en la seccion de la estructura donde se aplicara la fuerza de 50 N, ver Figura 16.

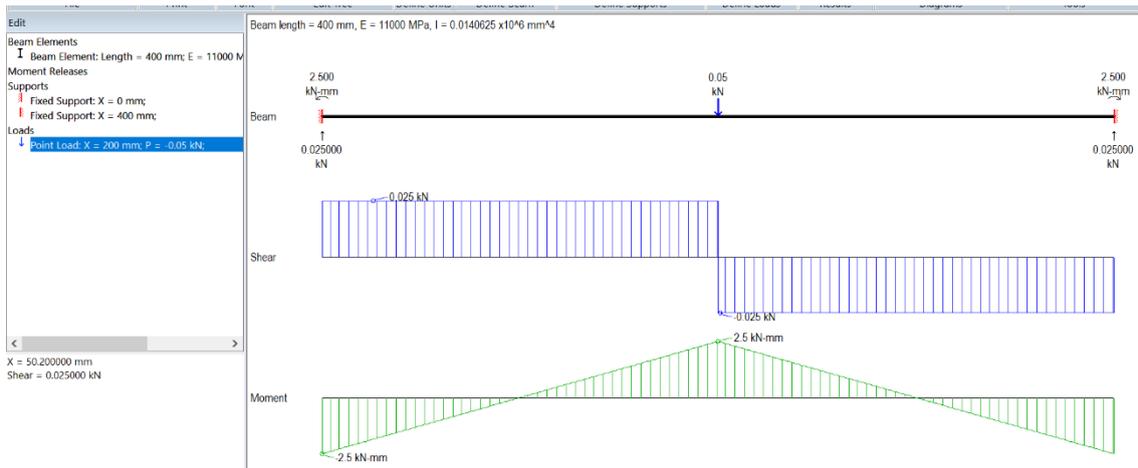


Figura 16. Calculo de la fuerza.

2.6 Cálculo del factor de seguridad estructural

Este dato se obtuvo mediante la sección donde tiende al máximo desplazamiento.

Se determina el módulo de la sección transversal S con el software DSN Winbeam, ver Figura 17.

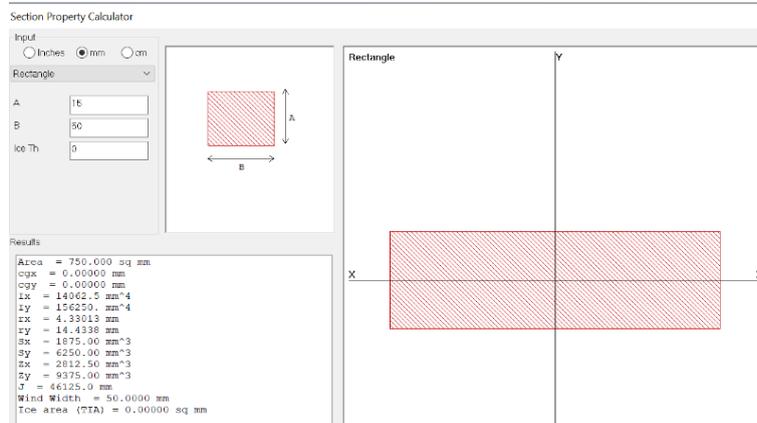


Figura 17. Cálculo de seguridad.

Como el material seleccionado fue madera de pino entonces el cálculo del factor de seguridad es igual a:

$$\sigma_t = \frac{M}{S}$$

$$\sigma_t = \frac{2.5 \text{ N} * \text{m}}{1.875 * 10^{-6} \text{ m}^3}$$

$$\sigma_t = 1.333 \text{ MPa}$$

Se encuentra la resistencia a la cadencia usando el software CES edupack, ver Figura 18.

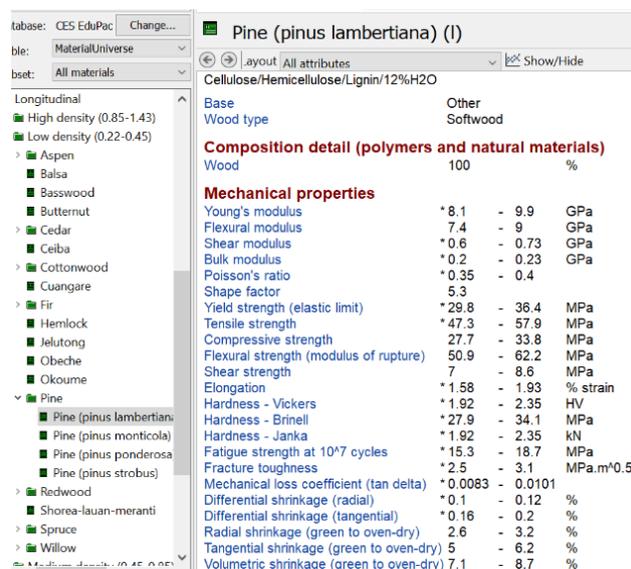


Figura 18. Resistencia usando el Software.

$$f_s = \frac{0.61(S_y)}{\sigma_t}$$

$$f_s = \frac{0.61(29.8MPa)}{1.333MPa}$$

$$f_s = 13.63$$

Donde

σ_t = Esfuerzo del trabajo

M = momento de flexión máximo

S = módulo de la sección transversal

S_y = Resistencia a la cedencia

Se determina un factor de seguridad de **13.72** bastante similar al que entrega el software INVENTOR, ver Figura 19.

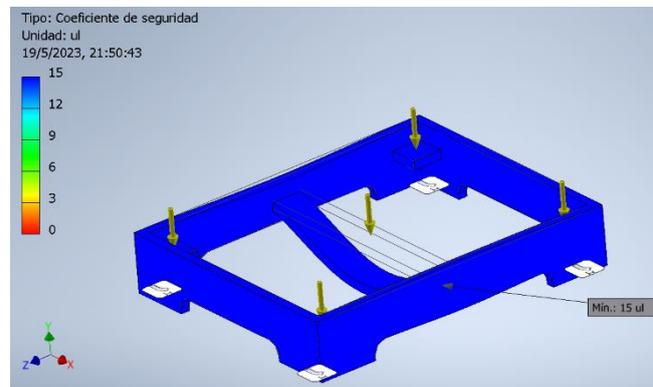


Figura 19. Hallazgo de factor seguridad 13.72.

2.7 Diseño de la estructura del germinador

Para el diseño de la estructura se utilizó un perfil de aluminio estructural 6061 en L de 11x11x1 mm, Figura 20.

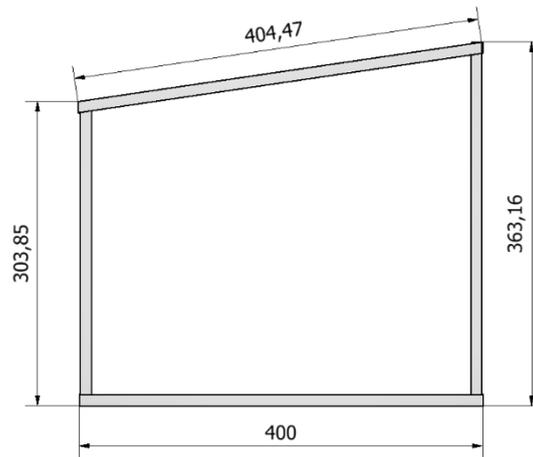


Figura 20. Diseño de estructura del Germinador.

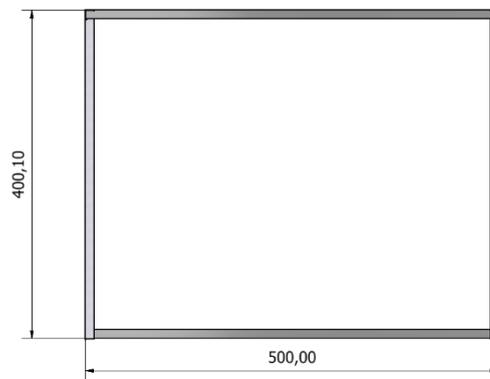


Figura 21. Parte frontal del diseño del Germinador.

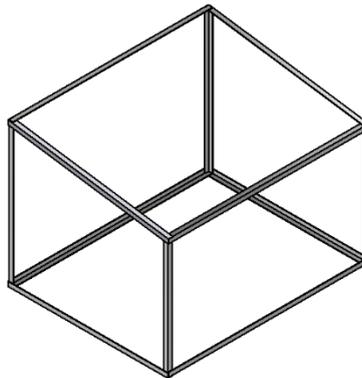


Figura 22. Estructura general del área del Germinador.

2.8 Diseño de las paredes del germinador

Para esto se utilizó acrílico transparente de 3 mm, el cual se cortará a la medida para cubrir la parte interna del germinador y que encaje con la estructura de aluminio. A excepción de la cara inferior la cual será una pared de acrílico de 6 mm de espesor, Figura 23.

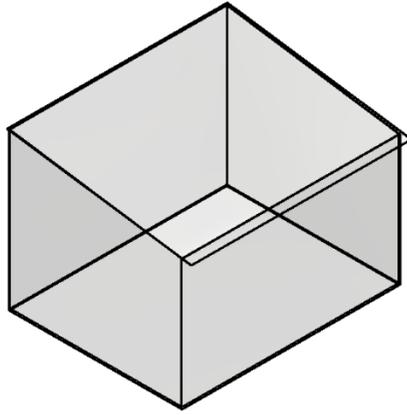


Figura 23. Diseño del interior del germinador.

2.9 Análisis estructural del germinador

La estructura del germinador soportará en total unos 3kg de peso por lo que la fuerza que ejercerá sobre la pared inferior será de 30 N, Figura 24.

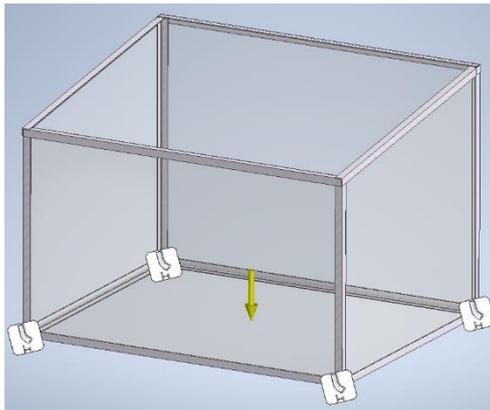


Figura 24. Análisis de la estructura del germinador.

Se realizó el análisis de la estructura utilizando el software INVENTOR, obteniendo así un desplazamiento en la cara inferior de nuestra estructura de 0.1197 mm, Figura 25.

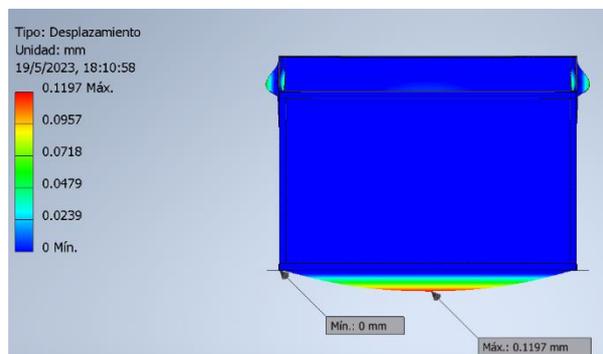


Figura 25. Análisis con software INVENTOR.

2.10 Tensión de von mises

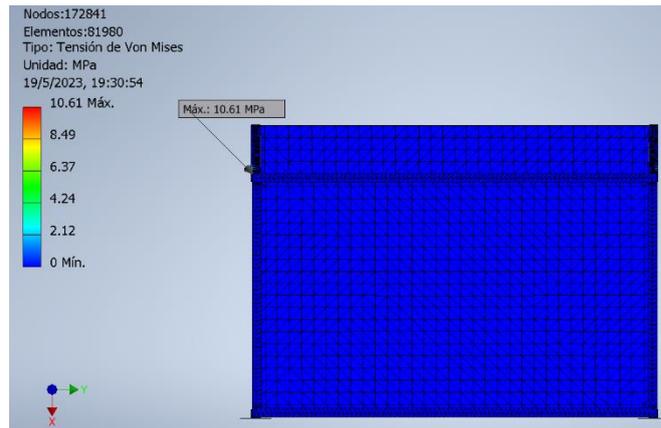


Figura 26. Tensión de Von Mises.

2.11 Cálculo de fuerzas y momentos

Para realizar estos cálculos se utilizó el software DSN Winbeam y se basa en la sección de la estructura donde se aplicará la fuerza de 30 N y tomando la cara inferior como si fuera una viga simplemente apoyada, Figura 27.

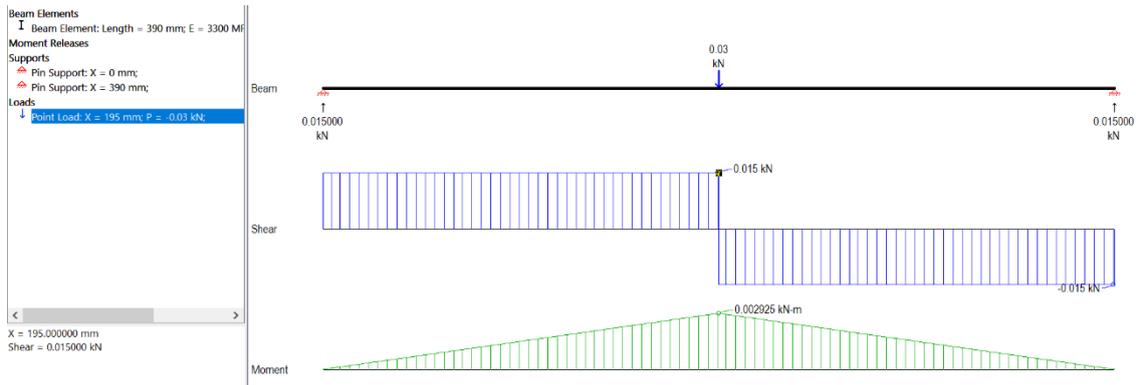


Figura 27. Cálculo de fuerzas y momentos.

2.12 Cálculo del factor de seguridad de la estructura

El factor se obtuvo mediante la tensión de Von Mises con la tensión máxima admitida por el material. Como se escogió aluminio 6061 soldado entonces el cálculo del factor de seguridad sería igual a:

$$fs = \frac{S_y}{\sigma_v}$$

$$fs = \frac{55MPa}{10.61MPa}$$

$$fs = 5.18$$

Donde:

σ_v = tensión de von mises

S_y = Resistencia a la cadencia

Por lo tanto, se obtienen un factor de seguridad de **5.18** el cual es aproximado al obtenido mediante el software INVENTOR, Figura 28.

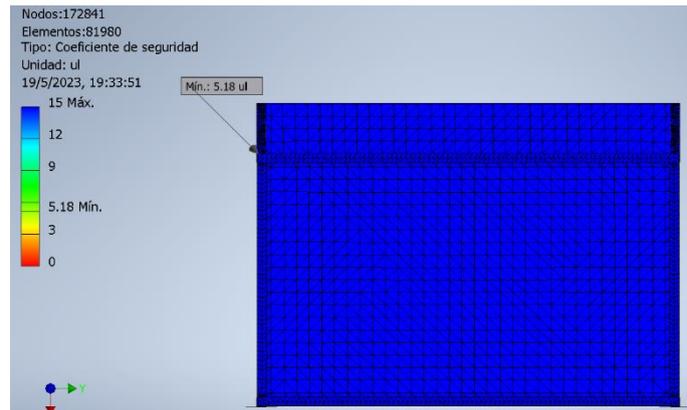


Figura 28. Factor de seguridad de 5.18.

2.13 Cálculo de los remaches

Para asegurar la estructura de aluminio se utilizó remaches de golpe con cabeza plana

Se calculó el diámetro del remache basado en la sección donde se aplicarán los 50N de fuerza, Figura 29.

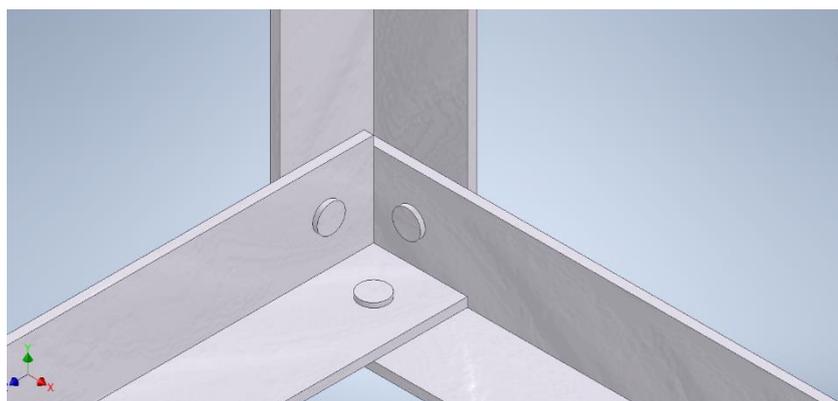


Figura 29. Cálculo de los remaches.

Primero se calcula el esfuerzo cortante de diseño

$$\tau_d = \frac{S_y}{4}$$

$$\tau_d = \frac{55MPa}{4}$$

$$\tau_d = 13.75MPa$$

Este dato se utiliza en la siguiente ecuación y se despeja el diámetro

$$\tau_d = \frac{F}{\frac{\pi * dr^2}{4}}$$

$$13.75MPa = \frac{50N}{\frac{\pi * dr^2}{4}}$$

$$13.75MPa * \frac{\pi}{4} = \frac{50N}{dr^2}$$

$$dr^2 = \frac{50N}{10.799MPa}$$

$$dr^2 = \frac{50N}{10.799 \frac{N}{mm^2}}$$

$$dr^2 = \frac{50N}{10.799 \frac{N}{mm^2}}$$

$$dr^2 = 4.63mm^2$$

$$dr = \sqrt{4.63mm^2}$$

$$dr = 2.15mm \cong 2mm$$

Donde:

τ_d =esfuerzo cortante de diseño

S_y =resistencia a la cadencia del material

F= fuerza aplicada

dr=diámetro del remache

2.14 Diseño del sistema de riego

Se utilizó manguera de avance y acoples de plástico para el sistema de riego por goteo, ver Figura 30.



Figura 30. Diseño de riego para germinador.

2.15 Diseño de la fuente de calor

Se diseñó un soporte y se utilizó una bombilla incandescente para generar la temperatura necesaria dentro del invernadero, ver Figura 31.



Figura 31. Soporte y fuente de calor

2.16 Diseño del sistema de ventilación

Se utilizó un ventilador de 12 voltios para enfriar el invernadero durante el proceso, ver Figura 32.

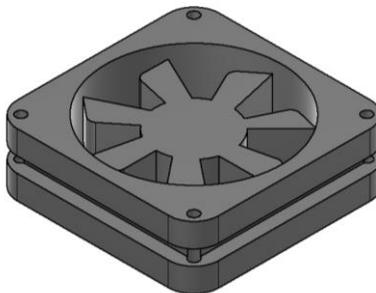


Figura 32. Ventilador para invernadero

2.17 Bandejas de germinación

Se utilizó bandejas plásticas del tipo semilleros con 60 alveolos, ver Figura 33.



Figura 33. Bandeja de germinación de las semillas

3. CAPÍTULO III

3.1 Componentes a implementar

A continuación, en el presente capítulo se presentan y se detallan las funciones que cumplen los componentes electrónicos de Arduino dentro del germinador automatizado, por lo cual se describe el uso y las características de los componentes:

3.1.1 Puente H

Es un circuito el cual pertenece a la familia de Arduino y permite controlar la dirección de la corriente en diferentes sentidos, y esto lo logra por medio de los cuatros transistores, los cuales por su orden permiten hacer girar un motor en diferentes direcciones, se activan por pares dependiente el lado donde necesite girar el motor Abraham[27]. La función que cumple el puente H es controlar los movimientos de los componentes del germinador, Figura 34.

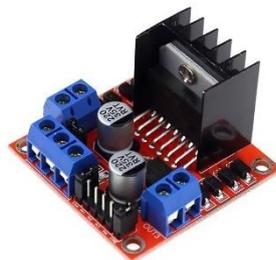


Figura 34. Componente Puente H [27].

3.1.2 Módulo DIMMER de Arduino AC 220V/2ª

Según Max Electrónica [29] este módulo pertenece a la familia Arduino el cual permite controlar la potencia o intensidad de la luz o la velocidad de los motores con corriente alterna (AC) con una carga de 2 amperios a 220 V, este tipo de módulo se los utiliza para proyectos de domótica o robótica y otros. El rol a cumplir dentro del germinador es controlar la cantidad de calor que necesita el proceso de germinación dentro del huerto automático, con el fin de garantizar que las medidas de calor sean las adecuadas para las plantas ornamentales, Figura 35.

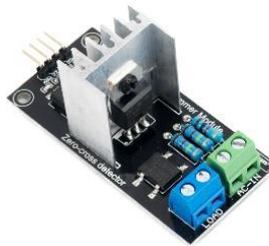


Figura 35. Módulo DIMMER AC 220V/2A [29].

3.1.3 Sensor de humedad FC-28

Este tipo de sensor es conocido por su función de detectar la humedad del suelo FC-28, el cual es un dispositivo electrónico que utiliza la tecnología de resistencia del suelo y determina la cantidad o nivel de humedad que presenta el suelo, está compuesto por sondas metálicas que se incrustan en el suelo y actúan como electrodos [30]. El funcionamiento dentro del germinador es medir la temperatura del suelo, donde se inserta, Figura 36.

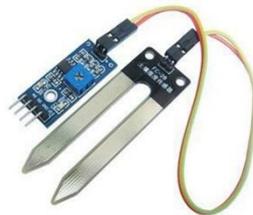


Figura 36. Sensor de humedad FC28 [30].

3.1.4 Bomba hidráulica

Es un componente el cual aumenta la energía cinética dentro de un caudal, estos son elementos ampliamente conocidos en el mercado y empleados en las pequeñas y grandes industrias [31]. La función a cumplir dentro del sistema de riego es que permite extraer el agua de la fuente del germinador, Figura 37.



Figura 37. Bomba de agua [31]

3.1.5 Ventilador de 12 v

Un cooler o ventilador de 12v como se lo conoce comúnmente, se utiliza en sistemas de refrigeración o enfriamientos electrónicos, se los puede encontrar en los case de los pc y en otros dispositivos electrónicos [32]. La función del cooler dentro del germinador es brindar ventilación dentro de la estructura física, Figura 38.



Figura 38. Ventilador o cooler [32].

3.1.6 Pantalla HMI Arduino

Según, atvise [33] un sistema HMI permite la comunicación de la máquina con el ser humano facilitando así los procesos y tareas, además permite manipular la información y datos a recibir. La función que cumple dentro del germinador es presentar la información de humedad, calor, temperatura que el controlador registre, el componente a utilizar se observa en la Figura 39.



Figura 39. Pantalla HMI Arduino[33]

3.1.7 Manguera de polietileno para riego

Este tipo de manguera se la utiliza para tuberías, está fabricada de material polietileno de alta densidad (PEAD) el cual es un material duradero y resistente a la corrosión y muy flexible [34]. Esta manguera permitirá hidratar a las semillas dentro del suelo en el sistema de riego que lleva la estructura del germinador, Figura 40.



Figura 40. Manguera para sistema de riego [34]

3.1.8 Elementos para la estructura del germinador

La estructura del germinador está compuesta por tres materiales principales los cuales son láminas de aluminio, madera Pino para la estructura física, y remaches de aluminio, que conforman la parte estructural del germinador Figura 41.



Figura 41. Componentes para la estructura física del germinador.

3.2 DISEÑO ELECTRONICO

3.2.1 Diagramas para conexión del módulo DIMMER

El principio de funcionamiento consiste en atenuar la radiación lumínica, consta de un TRIAC y un detector de cruce por cero, para sincronizar la variación de tensión y la fase de la corriente AC, como lo indica la Figura 42.

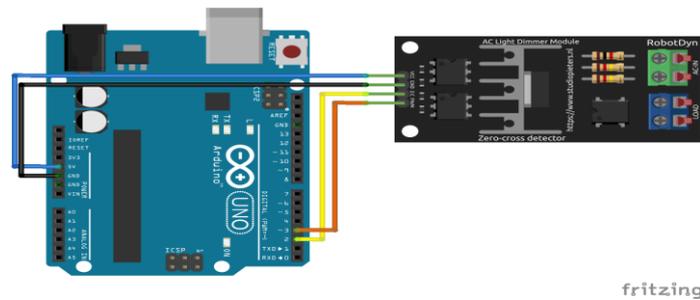


Figura 42. conexiones del módulo DIMMER.

Donde

- Vcc se conecta al pin 5V o 3V del microcontrolador
- GND a tierra del microcontrolador
- Z-C en el pin 2
- PWM en el pin 3

Para la conexión de los pines en el Arduino MEGA se consultó la documentación del módulo RBD DIMMER, revisar Figura 43.

Board	INPUT Pin Zero Cross		OUTPUT Pin
Leonardo	D7 (NOT CHANGABLE)		D0-D6, D8-D13
Mega	D2 (NOT CHANGABLE)		D0-D1, D3-D70
UNO NANO	D2 (NOT CHANGABLE)		D0-D1, D3-D20
ESP8266	<ul style="list-style-type: none"> • D1(I05) • D5(I014) • D7(I013) 	<ul style="list-style-type: none"> • D2(I04) • D6(I012) • D8(I015) 	<ul style="list-style-type: none"> • D0(I016) • D2(I04) • D6(I012) • D8(I015)

Figura 43. Parámetro de conexión del módulo DIMMER.

3.2.2 Diagrama de conexión para bomba de agua

Para el sistema de riego se utilizó un mosfet IRF520N para controlar el encendido del componente, se debe alimentar el módulo, mediante GND y Vin, ver Figura 44.



Figura 44. conexión de bomba de agua al IRF520N

Donde

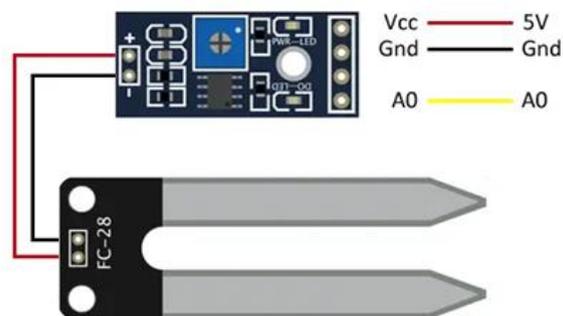
- Vin va alimentada entre 5 y 24V
- SIG se conectará a un pin digital del microcontrolador
- Vcc se conectará a la alimentación del Arduino
- GND a tierra del microcontrolador

3.2.3 Diagrama de conexión para el sensor de humedad

El sensor FC-28 es un componente el cual mide la humedad del suelo, los valores van desde 0 a 1023 en un suelo muy seco o aire. La salida digital dispara LOW cuando el suelo no este húmedo, y HIGH cuando este muy humedo, Figura 45.

Figura 45. Conexión sensor de temperatura

Donde



- Vcc se conecta a la alimentación 5V del microcontrolador
- GND a tierra del microcontrolador
- D0 valor digital

- A0 conectado a la entrada analógica

3.2.4 Diseño electrónico

Se empleó el software Arduino y Fritzing para la programación y simulación de los componentes del dispositivo, presentando el siguiente diagrama de protoboard, cada uno de los componentes deben estar correctamente conectados y realizar pruebas de funcionamiento o simulación, ver Figura 46.

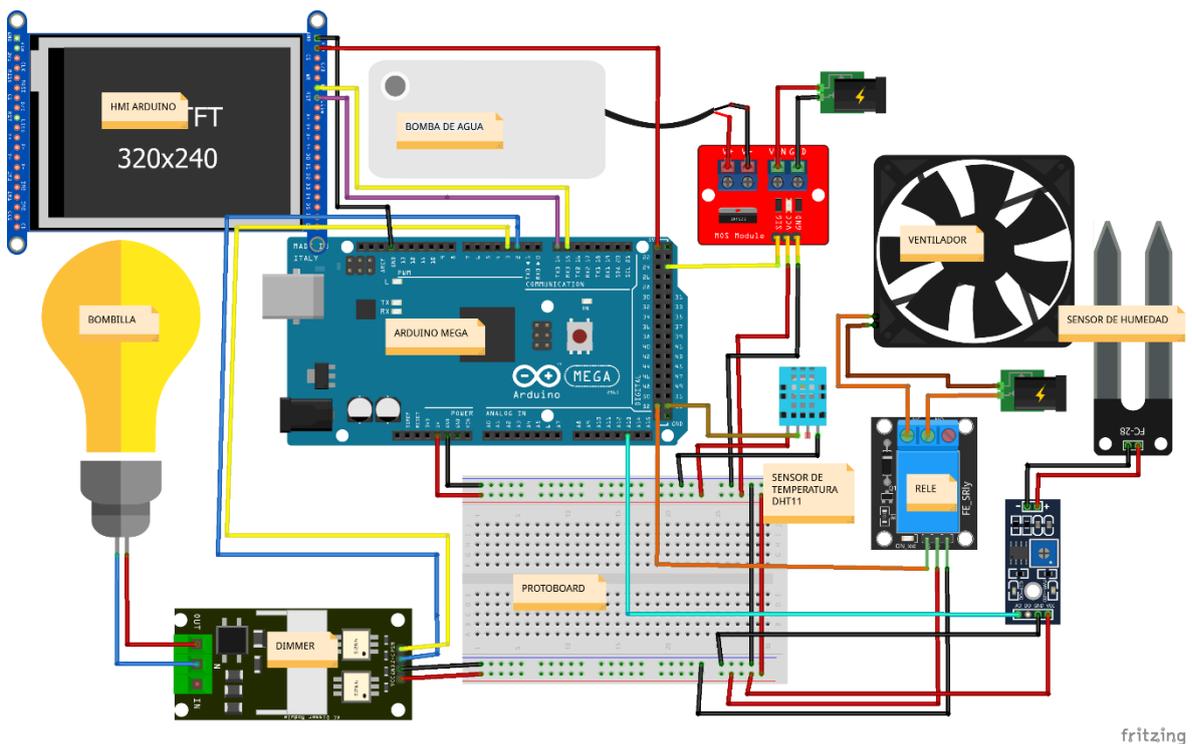


Figura 46. Diseño esquemático de circuito germinador.

La planta posee un controlador difuso para la temperatura y humedad, tanto para las semillas de petunia como para las de ciclamen.

Depende de la semilla seleccionada el controlador configurará un set point, con la ventilación adecuada y el riego cada cierto tiempo del día.

En la pantalla HMI se podrá visualizar las diferentes variables que se presentan durante el funcionamiento de la planta como son: tipo de semilla, temperatura y humedad.

El Arduino proporcionará el voltaje necesario para los sensores de humedad y temperatura, mientras que una fuente externa de 5-24V DC proporcionará el voltaje para los módulos de

relé y para la bomba de agua cuando la humedad del suelo este por debajo del nivel establecido como óptimo.

La fuente de calor o bombilla estará conectada a módulo dimmer, conectado al Arduino en la parte donde se produce el control de la intensidad de la bombilla y del otro lado estará conectado a una fuente de 110V AC para su voltaje necesario.

3.2.5 Diseño de la pantalla HMI

Para el diseño del HMI o interfaz humano maquina se utilizó el software Nextion el cual es un programa el cual nos permite editar diferentes tipos de pantallas HMI ya sean de tipo industrial o convencionales, ver figura 47.

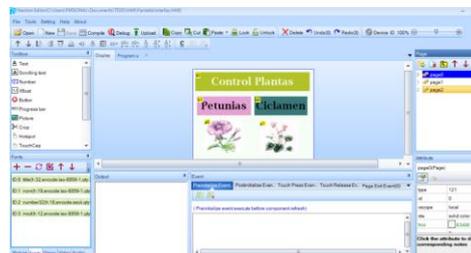


Figura 47. Programa Nextion para HMI

Para el proyecto se diseñó dos pestañas en el HMI en cada una se ubica el tipo de semilla con la que se va a trabajar y las variables de Temperatura y Humedad, tal como se puede observar en las figuras 48 y 49.



Figura 48. Pestaña de petunias



Figura 49. Pestaña de ciclamen

3.2.6 Diagrama de flujo del germinador

En la Figura 50 se denota el proceso para poner en marcha al germinador de petunias y ciclamen, el proceso empieza con la puesta en marcha del sistema, donde la pantalla HMI muestra 2 opciones las cuales son: petunias o ciclamen, luego el control difuso implementado en el sistema se pondrá en marcha primero se establecerá la temperatura adecuada en el ambiente mediante la fuente de calor que en este caso es el foco, el ventilador ayudará a controlar y disipar la temperatura, para la humedad del suelo el sensor indicará si este se encuentra en el porcentaje de humedad adecuado de la semilla caso contrario este enviará la señal para que se encienda la bomba de agua hasta alcanzar el nivel óptimo de humedad en el suelo. Todas estas variables se mostrarán en la pantalla HMI.

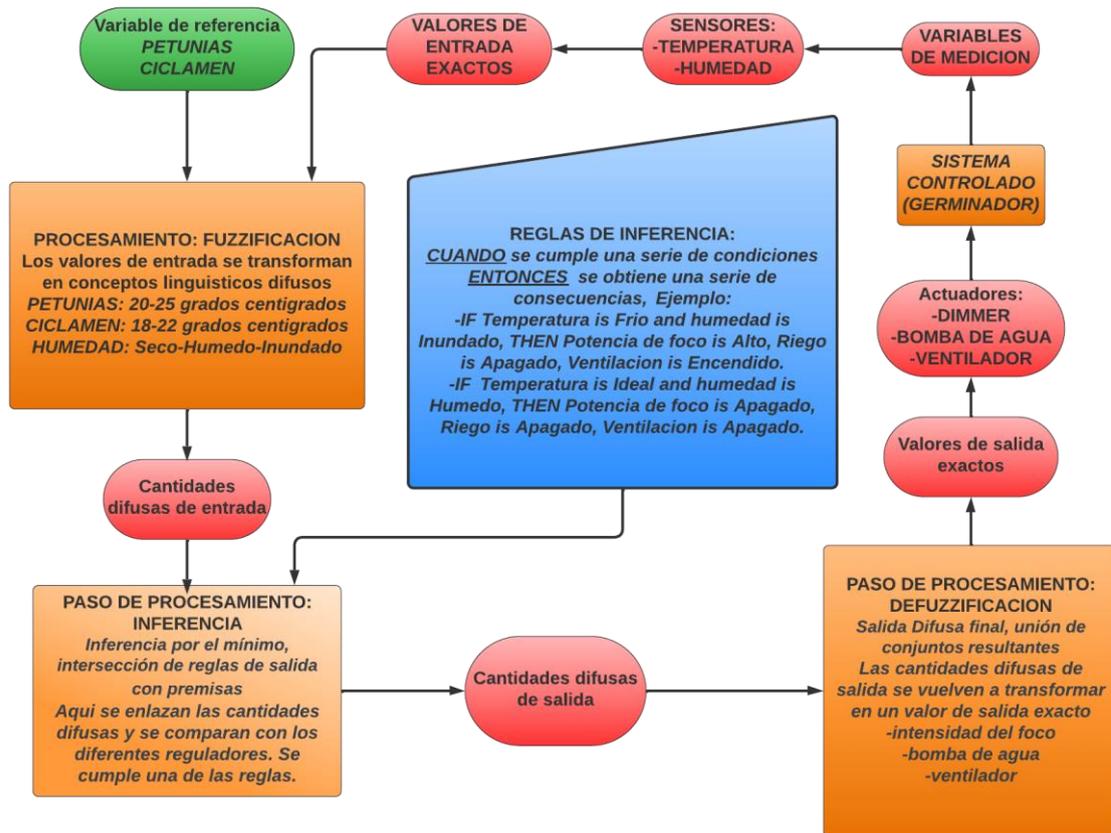


Figura 50. diagrama de flujo de germinador automático con Arduino.

3.3 Fuzzificación del sistema

En esta primera etapa del sistema se convierte las entradas numéricas otorgadas por los sensores de temperatura y humedad para convertirlos en variables lingüísticas.

3.3.1 Funciones de membresía Temperatura y Humedad

En el control del sistema las funciones de membresía son las funciones que describirán como se asignaran los valores de entrada al conjunto difuso en este caso para la temperatura será: Frio, Ideal y Caliente como se indica en la figura 51. Mientras que para la humedad del suelo será: Seco, Húmedo, Inundado como se muestra en la Figura 52.

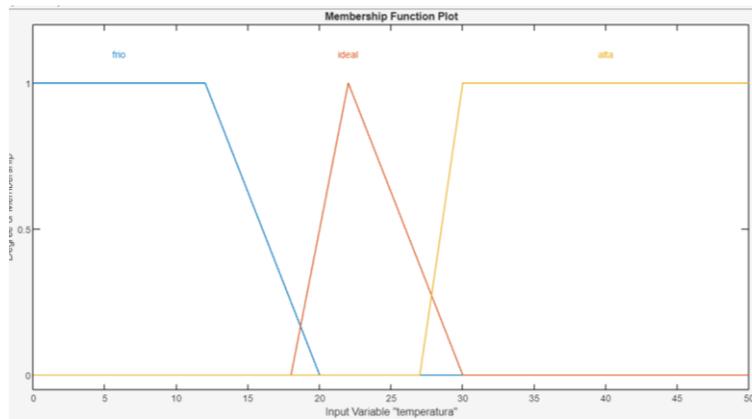


Figura 51. Funciones de membresía Temperatura

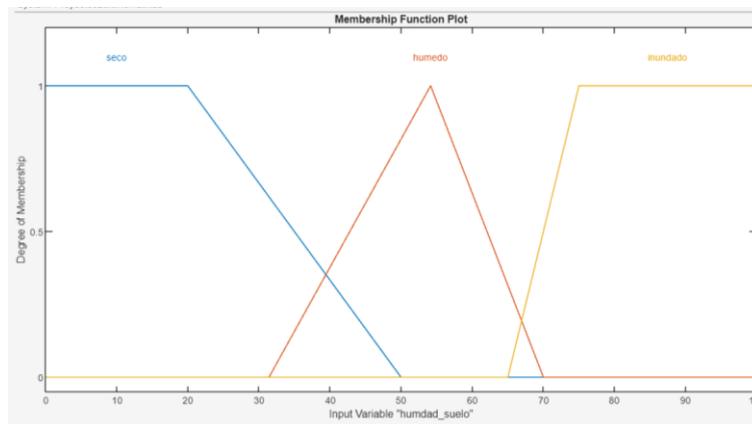


Figura 52. Funciones de membresía Humedad del suelo

3.4 Reglas de inferencia para el controlador difuso

El desarrollo de la programación se la realizó en base a las entradas y salidas que se disponen dentro del sistema, para enlazarlas entre ellas se utilizaron reglas para el proceso de inferencia y así transformar variables de entrada en variables de salida, así como se observa en la figura 53.

	Rule
1	If temperatura is frio and humdad_suelo is seco then potencia_foco is alto, riego is alto, ventilacion is apagado
2	If temperatura is ideal and humdad_suelo is humedo then potencia_foco is medio, riego is medio, ventilacion is apagado
3	If temperatura is alta and humdad_suelo is inundado then potencia_foco is apagado, riego is apagado, ventilacion is encendido
4	If temperatura is frio and humdad_suelo is humedo then potencia_foco is alto, riego is poco, ventilacion is apagado
5	If temperatura is frio and humdad_suelo is inundado then potencia_foco is alto, riego is apagado, ventilacion is apagado
6	If temperatura is frio and humdad_suelo is seco then potencia_foco is alto, riego is alto, ventilacion is apagado
7	If temperatura is ideal and humdad_suelo is seco then potencia_foco is bajo, riego is alto, ventilacion is apagado
8	If temperatura is ideal and humdad_suelo is humedo then potencia_foco is medio, riego is medio, ventilacion is apagado
9	If temperatura is ideal and humdad_suelo is inundado then potencia_foco is medio, riego is apagado, ventilacion is encendido
10	If temperatura is alta and humdad_suelo is seco then potencia_foco is apagado, riego is alto, ventilacion is encendido
11	If temperatura is alta and humdad_suelo is humedo then potencia_foco is apagado, riego is medio, ventilacion is encendido
12	If temperatura is alta and humdad_suelo is inundado then potencia_foco is apagado, riego is apagado, ventilacion is encendido

Figura 53. Mecanismo de inferencia (reglas)

3.5 Defuzzificación del sistema

En esta etapa ya con la lógica difusa implementada se combinan las reglas de inferencia para determinar las salidas del sistema y así obtener una salida final o una acción que se desea realizar en este caso controlar la Temperatura mediante el dimmer, la humedad con la activación de la bomba de agua y la ventilación del lugar.

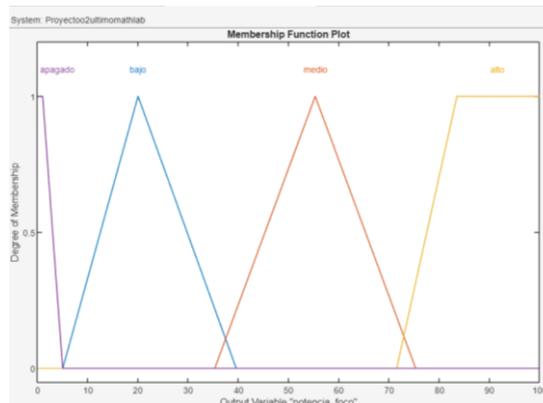


Figura 54. Funciones de membresía para la potencia del foco

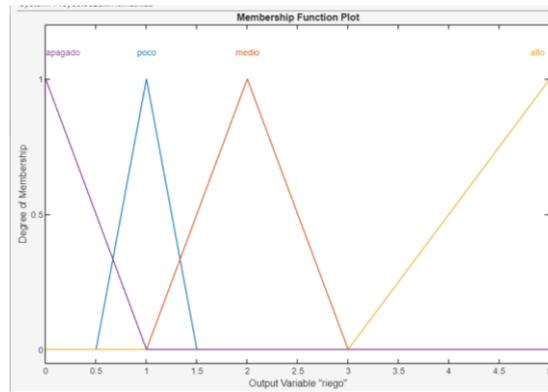


Figura 55. Funciones de membresía para el riego de la bomba

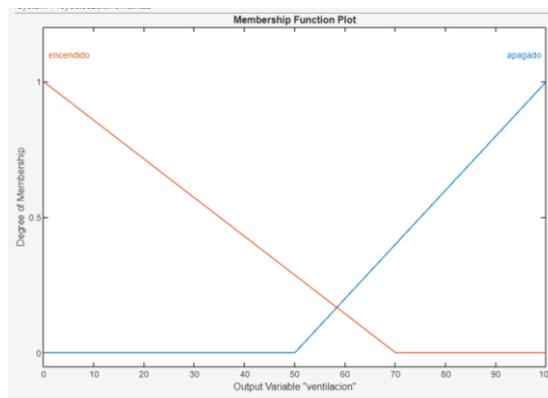


Figura 56. Funciones de membresía para la ventilación

CAPITULO IV

ANÁLISIS Y RESULTADOS

Dentro del presente capítulo se desarrollan los resultados del diseño y construcción de un germinador de esquejes y semillas de petunias y ciclamen, el cual está estructurado y programado con la tecnología Arduino, para el proceso de germinación automática.

4.1 Construcción finalizada del dispositivo

Una vez construida la estructura de la máquina, se dispone a ubicar todos los elementos electrónicos, mecánicos y de control, el dispositivo presenta las siguientes vistas, como se ve en la figura 57.

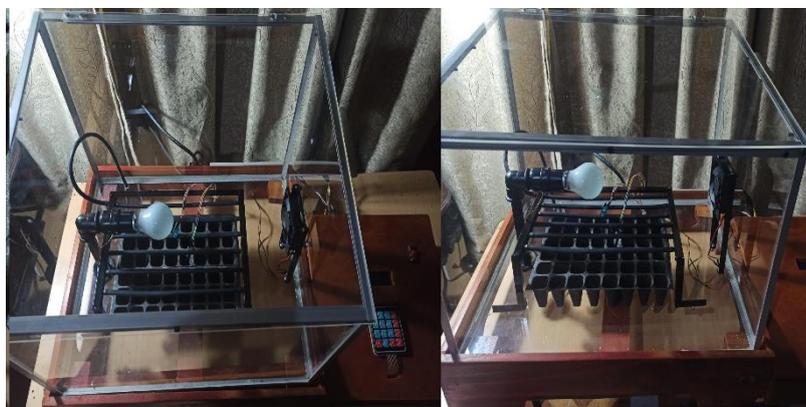


Figura 57. Vista lateral, superior y frontal.

Una vez terminado el dispositivo se procede a colocar las semillas y esquejes de las plantas dentro de los semilleros con el sustrato adecuado, ver figura 58.



Figura 58. Puesta de esquejes y semillas en semilleros.

Se colocarán los semilleros ya listos para poner dentro del invernadero y poner la planta en funcionamiento, como se ve en la figura 60, y una vez colocado los semilleros se procede a

escoger el tipo de semilla con el que se va a trabajar, ver figura 59.

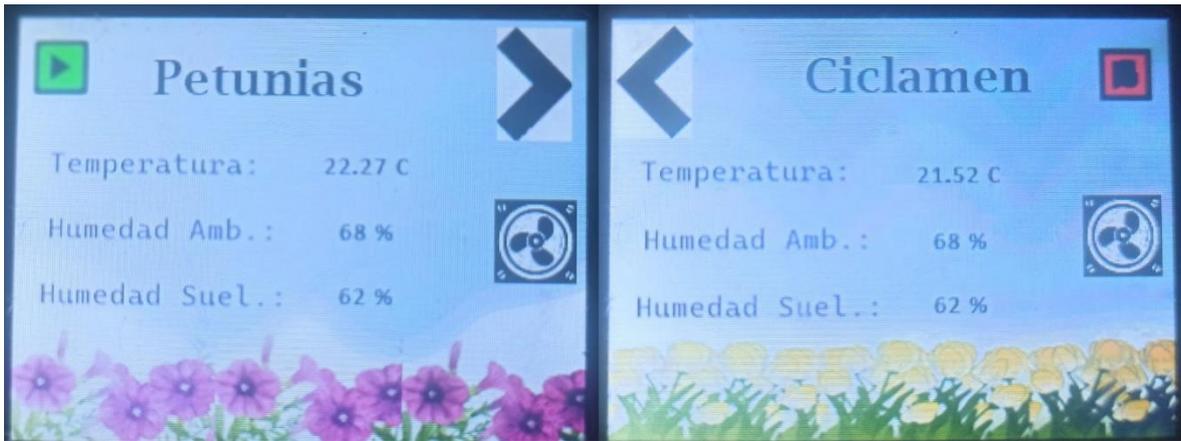


Figura 59. Pantalla HMI en funcionamiento



Figura 60. Puesta en marcha del dispositivo.

Los primeros resultados para los esquejes de petunias se dan a partir de la segunda semana donde los esquejes ya han soltado raíz, ya con esto se puede proceder a sembrar en maceteros más grandes e individuales, ver figura 61.



Figura 61. Esquejes de petunia enraizados

Los resultados para las semillas de ciclamen se dan a partir de los 20 días, para este punto ya habrán salido pequeñas plántulas con raíz por lo que ya se puede proceder a llevarlas a una maceta más grande, ver figura 62.



Figura 62. Semillas de ciclamen germinadas

4.2 Resultados de desplazamiento y factor de seguridad

Tabla 2. Resultados de desplazamiento

Desplazamiento	Valor	Descripción
Base de madera	0.06892 mm	Se obtuvo un valor muy pequeño casi inexistente en el desplazamiento de la base de madera por lo que la estructura no se verá con deformaciones considerables.
Estructura de aluminio y acrílico	0.1197 mm	Se obtuvo un valor de desplazamiento muy pequeño en toda la estructura compuesta de perfiles de aluminio y láminas de acrílico, además al ser este último un material plástico no sufrirá deformaciones.

Fuente: Autores propios.

En la tabla número 2 se muestran los resultados obtenidos de desplazamiento de las dos principales estructuras que componen el germinador.

Tabla 3. Valores obtenidos para el factor de seguridad

Factor de seguridad	Valor	Descripción
Base de madera	13.72	El factor de seguridad obtenido es de 13.72 a 1 por lo que la estructura de madera es capaz de soportar la carga aplicada sin ningún problema.
Estructura de aluminio y acrílico	5.18	El factor de seguridad de la estructura de aluminio y acrílico es de 5.18 a 1, lo cual nos indica que el germinador podrá soportar la carga real sin fallar.

Fuente: Autores propios.

En la tabla número 3 se muestran los resultados obtenidos del factor de seguridad de ambas estructuras indicando que el germinador soportara las cargas aplicadas en ella sin fallar.

4.3 Pruebas de los parámetros de la germinación sin control y el controlador Fuzzy

Tabla 4. Proceso de germinación sin controlador

Germinación sin control					
Descripción	Temperatura	Humedad	Ventilación	Semillas y esquejes plantados	Semillas y esquejes germinados
Mes 1	25°C	70%	Sin ventilación	48	31
Mes 2	24°C	73%	Sin ventilación	48	28
Mes 3	25°C	50%	Sin ventilación	48	29

Fuente: Autores propios.

En la tabla 4 se muestran los datos recolectados con el sistema tradicional para controlar la temperatura y humedad del invernadero, cabe destacar que en este método no se disponía de ventilación para el sistema. El resultado no es muy eficaz puesto que por cada 10 semillas solo se obtuvieron 6 plántulas.

Tabla 5. Controlador Fuzzy

Controlador Fuzzy					
Descripción	Temperatura	Humedad	Ventilación	Semillas y esquejes plantados	Semillas y esquejes germinados
Mes 1	22°C	60%	Normal	48	46
Mes 2	22°C	60%	Normal	48	48
Mes 3	22°C	60%	Normal	48	47

Fuente: Autores propios.

En la tabla 5 se muestran los datos recolectados con el sistema de control Fuzzy en la cual se obtiene una mayor producción en la germinación de semillas y esquejes. Teniendo como resultado de por cada 10 semillas o esquejes se obtenía 9 a 10 plántulas.

4.4 Análisis del tiempo de germinación entre el método tradicional y Fuzzy

Al realizar la comparación se puede observar que existe un ahorro de tiempo bastante significativo en el tiempo de germinación de las semillas y esquejes. En la figura 63 se tienen los datos del tiempo de cada controlador.

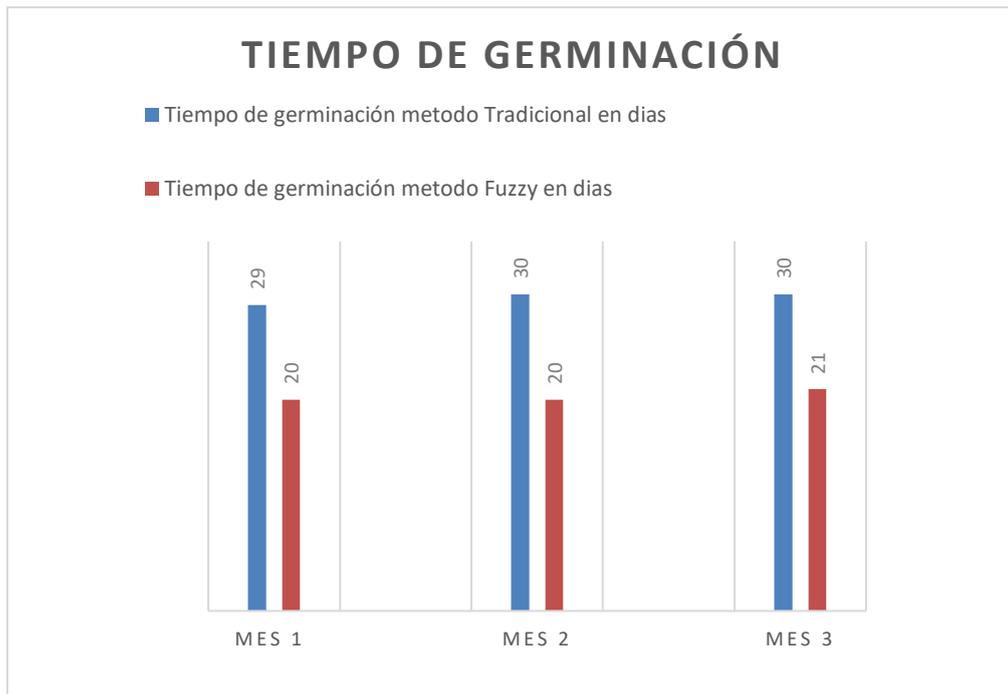


Figura 63. Comparación tiempo de germinación

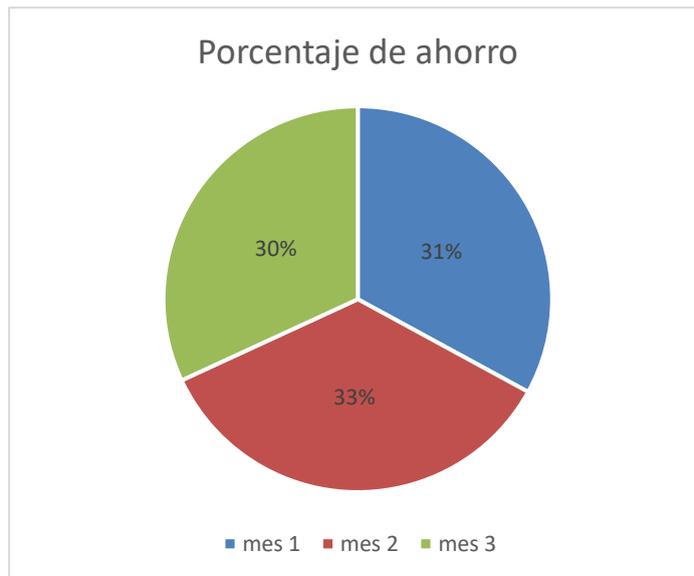


Figura 64. Porcentaje de ahorro

De los datos obtenidos en la figura 63 se obtiene un porcentaje de ahorro en el tiempo de germinación para cada mes como se observa en la figura 64, obteniendo como resultado un promedio de 31,33% en el tiempo total del proceso. Acortar el tiempo de germinación en forma significativa, indica la eficacia que existe en el uso del sistema implementado, además que ayuda a aumentar el volumen de producción al año, mostrando así la rentabilidad y eficacia de la máquina.

4.5 Análisis costos de elaboración

Los análisis redactados muestran un presupuesto a baja escala, ya que se realiza con el fin de germinar semillas, ya que un huerto es diferente desde su esquema y funcionamiento.

4.5.1 Análisis económico de la máquina

Se hizo un análisis de los gastos para la construcción de la máquina, donde se incluyen mano de obra, componentes y materiales.

Construcción mecánica

Se detalla la lista de precios.

Tabla 6. Balance económico

Cantidad	Descripción	Precio por unidad (\$)	Precio total (\$)
-----------------	--------------------	-------------------------------	--------------------------

1	Tabla de madera de pino	5,50	5,50
1	Perfil de aluminio en ángulo	6,50	6,50
1	Plancha de acrílico de 3mm	40,00	40,00
100	Remaches de aluminio de cabeza plana	0,70	0,70
2	Bisagras	0,50	1,00
1	Semilleros	1,00	1,00
5	Tubo para riego	0,80	4,00
16	Acoples para manguera	0,80	12,80
	Total		71,50 \$

Fuente: Autores propios.

4.5.2 Construcción eléctrica y de control

Se detalla la lista con su costo correspondiente.

Tabla 7. Balance económico de los componentes electrónicos

Cantidad	Descripción	Precio por unidad (\$)	Precio total (\$)
1	Arduino MEGA	25,00	25,00
1	Sensor de temperatura Dht11	2,00	2,00
1	Modulo dimmer Arduino	22,00	22,00
1	Sensor de humedad FC-28	2,50	2,50
1	Bomba de agua	10,00	10,00
1	Ventilador de 12v	3,00	3,00
1	Pantalla HMI	45,00	45,00
1	Cables para protoboard	4,00	4,00
1	Protoboard	8,00	8,00
1	Bombilla	1,00	1,00
1	Key pad 4x4	2,75	2,75
1	Módulo de 2 relés	3,00	3,00
2	Cargador variable	15,00	30,00
	Total		158,35

Fuente: Autores propios.

4.6 Mano de obra de la maquina

En el Ecuador la mano de obra ronda los 693,00\$ al mes, luego se analizó el valor diario para obtener la mano de obra descrita en la tabla

Tabla 8. Presupuesto de la mano de obra

Horas	Descripción	Precio-unidad \$	Total \$
240-300	Mano de obra Ing.	3,93	943,20
	Total		943,20

Fuente: Autores propios.

4.7 Costo total de la máquina

Se realizó la sumatoria de cada costo mencionado con anterioridad

Tabla 9. Costo total

Tipo de costo	Valor (\$)
Construcción mecánica	71,50
Construcción eléctrica y control	121,35
Mano de obra	943,20
Total	1136,05

Fuente: Autores propios.

CONCLUSIONES

- Se hizo el respectivo estado del arte con referencia a cada uno de los métodos para la germinación de semillas y esquejes de plantas ornamentales, obteniendo un buen diseño a partir de criterios de evaluación propuestos, con el objetivo de conseguir un proceso eficiente durante la germinación, además de obtener mejores brotes al final del ciclo.
- Se utilizó un software de diseño CAD para crear todas las partes que componen el dispositivo, todo esto para mejorar la calidad de diseño mediante el análisis estructural, este análisis ayuda a solucionar posibles problemas que se presenten en el proceso de construcción de la máquina. Dando como resultado un factor de seguridad de 5.18 para la estructura principal, ya con estos valores se garantiza un óptimo desempeño en la estructura de la máquina para su construcción.
- Se diseñó el sistema eléctrico de la máquina utilizando un simulador de circuitos electrónicos, en donde se colocaron todos los sensores y actuadores que funcionaran en conjunto para el control de temperatura, humedad y ventilación.
- Se realizó la respectiva programación para la máquina con una tarjeta Arduino MEGA y su sistema HMI a través de una pantalla touch compatible y se realizó las conexiones para la válvula de riego, ventilador y bombilla eléctrica.
- La lógica difusa es un sistema de control el cual no necesita de un modelo matemático que describa cómo se comporta la planta, por lo que es más fácil manejar los datos de entrada, utilizando reglas y conjuntos difusos para tomar decisiones basadas en datos difusos y generar salidas precisas de control, obteniendo excelentes resultados de control acortando el tiempo de germinación de 30 días a 20 días en promedio.
- Se tuvo un mejor resultado de producción en la germinación, aumentando en un 40% el número de semillas que terminaron enraizando ya que al implementar el controlador Fuzzy, este puede manejar de mejor manera la imprecisión e incertidumbre de los datos de entrada.

- En la forma tradicional de germinación, el tiempo de enraizamiento era de 30 días y aplicando el método Fuzzy se logró obtener un tiempo de germinación de 20 días por lo que hubo un porcentaje de ahorro del 31,33% demostrando así la eficacia del sistema al implementar este tipo de controlador.
- Para la construcción de este proyecto se tuvo una inversión de casi 200 dólares fuera de gastos de mano de obra, por lo que se puede concluir que es un proyecto rentable a largo plazo y con muchos beneficios para los pequeños emprendimientos florícolas.

RECOMENDACIONES

- Para obtener mejores resultados es preferible ajustar la temperatura a un promedio entre ambas donde los esquejes y semillas germinen de la mejor manera.
- El proyecto podría no solo germinar semillas de petunias y ciclamen por lo que si se necesitara hacer crecer otro tipo de semillas simplemente se optaría por configurar la interfaz y establecer una temperatura adecuada para la germinación.
- Se puede optar por un mejor método para el riego dentro de los semilleros, por ejemplo, la pulverización y así optimizar la cantidad de agua que se ocupa durante el proceso.
- Para crear un sistema aún más sólido se puede implementar el uso de un PLC de tipo industrial volviéndolo más preciso, estable y aumentando así el tiempo de vida útil de los actuadores y HMI.

5. REFERENCIAS

- [1] Acosta Belen, “+25 plantas ornamentales: qué son, tipos, nombres, imágenes y vídeos,” Apr. 03, 2023. <https://www.ecologiaverde.com/plantas-ornamentales-que-son-tipos-nombres-e-imagenes-2642.html> (accessed May 11, 2023).
- [2] Walter Rivera, “DISEÑO Y CONSTRUCCIÓN DE UN GERMINADOR DE SEMILLAS PARA CONSUMO HUMANO WALTER RIVERA LOZADA Trabajo de Grado para optar el título de,” 2016.
- [3] Agroentornos, “¿Qué es un germinador? – Agroentornos,” Mar. 02, 2021. <https://agroentornos.com/que-es-un-germinador/> (accessed May 11, 2023).
- [4] INGENIERÍA MECATRÓNICA, “Proyecto educativo del programa INGENIERÍA MECATRÓNICA Contenido,” 2020.
- [5] Huerta Y Jardin, “Semillas: Definición, tipos, germinación y almacenamiento – Comercial Mida,” Jun. 15, 2018. <https://comercialmida.es/blogs/blog/semillas> (accessed May 11, 2023).
- [6] Romero Luis and Garcia Heiler, “INFORME TÉCNICO FINAL DE PROYECTOS DE INVESTIGACIÓN,” Nov. 2019.
- [7] Ministerio de Agricultura y Ganadería, “Ecuador presenta actividades tecnológicas que implementa para desarrollar el agro – Ministerio de Agricultura y Ganadería,” Mar. 2021. Accessed: May 11, 2023. [Online]. Available: <https://www.agricultura.gob.ec/ecuador-presenta-actividades-tecnologicas-que-implementa-para-desarrollar-el-agro/>
- [8] A. Hernández López, A. Xochil Avila Alejandro, N. Mendoza Francisco, and H. Hernández López, “Diseño, construcción, verificación y prueba de un germinador de bajo costo”.
- [9] Walter Rivera, “diseño y construcción de un germinador de semillas para consumo,” 2016. <https://studylib.es/doc/6824449/dise%C3%B1o-y-construcci%C3%B3n-de-un-germinador-de-semillas-para-c...> (accessed May 24, 2023).
- [10] Banco Mundial, “La COVID-19 (coronavirus) hunde a la economía mundial en la peor recesión desde la Segunda Guerra Mundial,” Jun. 08, 2020. <https://www.bancomundial.org/es/news/press-release/2020/06/08/covid-19-to-plunge-global-economy-into-worst-recession-since-world-war-ii> (accessed May 11, 2023).
- [11] I.-A. G. S. C. M. of I.-A. S. for E. S. and C.-A. D. B. authorCorporate:UNESCO

- Office Montevideo and Regional Bureau for Science in Latin America and the Caribbean, “Evaluación del impacto del COVID-19 en las industrias culturales y creativas,” 2021. Accessed: May 11, 2023. [Online]. Available: <https://unesdoc.unesco.org/ark:/48223/pf0000380185?1=null&queryId=N-a3e3a6bd-cf60-40da-b09d-c604a0430750>
- [12] Info Agro, “El proceso de la germinación de semillas. Etapas,” 2022. https://www.infoagro.com/documentos/el_proceso_germinacion_semillas__etapas.asp (accessed May 24, 2023).
- [13] M. Romero *et al.*, “Automatización de un sistema de germinación para plántula,” *Am. J. Sci. Educ*, vol. 6, p. 22004, 2019.
- [14] Pineda jose, “Tecnología de los alimentos, Qué es la tecnología alimentaria,” 2022. <https://encolombia.com/economia/agroindustria/agronomia/tecnologia-de-los-alimentos/> (accessed May 24, 2023).
- [15] kros Agro, “Cómo cultivar petunias a partir de esquejes de la mejor manera,” 2021. <https://krosagro.com/es/tuneles-de-plastico/como-cultivar-petunias-a-partir-de-esquejes-de-la-mejor-manera/> (accessed May 24, 2023).
- [16] Liz Baessler, “¿Se puede cultivar ciclamen a partir de semillas? Cómo cultivar ciclamen a partir de semillas,” 2020. <https://www.diversegarden.com/houseplants/cyclamen/cyclamen-seed-propagation.htm> (accessed May 24, 2023).
- [17] H. Reyna, O. Fernández, A. Marisa, and A. Fierro Álvarez, “Manual de propagación de plantas superiores”.
- [18] Susan Parent, “¿Cómo influye la humedad en la calidad de los cultivos? | PRO-MIX,” 2022. <https://www.pthorticulture.com/es/centro-de-formacion/como-influye-la-humedad-en-la-calidad-de-los-cultivos/> (accessed May 24, 2023).
- [19] H. Encias, “Como cuidar las petunias guia definitiva” https://www.elmueble.com/plantas-flores/como-cuidar-petunias_45097. (Último acceso: Enero 1, 2023).
- [20] K. Jamilex and S. Ortiz, “Previo a la obtención del título de: INGENIERA AGROPECUARIA”.
- [21] Maher, “Sistema de Riego Automático: Qué es y Cómo Funciona.” <https://www.maherelectronica.com/sistema-riego-automatico/> (accessed May 24,

- 2023).
- [22] Alejandro Hernández López¹, Alma Xochil Avila Alejandre, Noé Mendoza Francisco, and Héctor Hernández López, “Diseño, construcción, verificación y prueba de un germinador de bajo costo,” 2019, Accessed: May 24, 2023. [Online]. Available: <https://www.scielo.org.mx/pdf/remexca/v10n2/2007-0934-remexca-10-02-431.pdf>
 - [23] “El sensor de temperatura - Rechner Sensors.” <https://www.rechner-sensors.com/es/documentacion/knowledge/el-sensor-de-temperatura> (accessed May 24, 2023).
 - [24] Vasyly Cherlinka, “Sensores De Humedad Del Suelo Para Usos Agrícolas,” Aug. 03, 2022. <https://eos.com/es/blog/sensores-de-humedad-del-suelo/> (accessed May 24, 2023).
 - [25] José Guerra, “Señal PWM con Arduino y analogWrite.” <https://programarfacil.com/blog/arduino-blog/pwm-con-arduino-analogico/> (accessed May 24, 2023).
 - [26] “Diseño de un controlador difuso basado en el esquema del compensador paralelo distribuido (PDC).” <https://ve.scielo.org/> (accessed Nov 06, 2023).
 - [27] AbrahamG, “Cómo utilizar puente H con Arduino Controlar motor de CD,” 2021. <https://www.automatizacionparatodos.com/puente-h-arduino/> (accessed Jun. 08, 2023).
 - [28] Luis del Valle, “Cómo utilizar el DHT11 para medir la temperatura y humedad con Arduino,” 2017. <https://programarfacil.com/blog/arduino-blog/sensor-dht11-temperatura-humedad-arduino/> (accessed Jun. 08, 2023).
 - [29] “Módulo Dimmer AC Optoacoplado 2CH BTA16600B Lógica Control 3.3-5V,” 2019. <https://maxelectronica.cl/prototipo/660-modulo-dimmer-ac-optoacoplado-2ch-bta16600b-logica-control-33-5v.html> (accessed Jun. 08, 2023).
 - [30] Victor Arrieta, “Sensor de humedad de suelo FC28 - Higrómetro Arduino – Arca Electrónica,” 2018. <https://www.arcaelectronica.com/blogs/tutoriales/sensor-de-humedad-de-suelo-fc28-higrometro-arduino> (accessed Jun. 08, 2023).
 - [31] Luis Llamas, “Encender una bomba de agua con Arduino,” 2016. <https://www.luisllamas.es/bomba-de-agua-con-arduino/> (accessed Jun. 08, 2023).
 - [32] Bosser, “Turbina Cooler 12V. 120 x 120 x 38mm Buje,” 2023. <https://bossar.com.ar/productos/coolers/25-turbina-cooler-12v-120-x-120-x-38mm-buje> (accessed Jun. 08, 2023).

- [33] ATVISE, «atvise,» 25 Abril 2021. [En línea]. Available: <https://atvise.vesterbusiness.com/news/sistema-hmi/>. [Último acceso: 30 11 2023].
- [34] “Todo sobre las mangueras de polietileno - Cauchos Malaca.” <https://www.cauchosmalaca.com/mangueras-de-polietileno/> (accessed Jun. 08, 2023).
- [35] COPADATA, «COPADATA,» 29 Noviembre 2023. [En línea]. Available: <https://www.copadata.com>. [Último acceso: 29 Noviembre 2023].
- [36] F. O. Tellez, «dep.fie.umich.mx,» 11 Diciembre 2023. [En línea]. Available: https://dep.fie.umich.mx/~fornelas/data/uploads/pres_controldifuso.pdf. [Último acceso: 11 Diciembre 2023].
- [37] J. C. Gomez, «frba.utn.edu.ar,» 12 Diciembre 2023. [En línea]. Available: <https://www.frba.utn.edu.ar/wp-content/uploads/2021/02/Fuzzy-Control.pdf>. [Último acceso: 12 Diciembre 2023].

6. ANEXOS



Anexo 1 Etapas de construcción de la estructura

Previo a la compra de los perfiles de aluminio en ángulo de 11x11 mm, plancha de acrílico y tablas de pino se procede a realizar los respectivos cortes según las medidas establecidas en los planos y a remachar las esquinas para formar la estructura.



Anexo 2 Corte de las vigas de madera

Luego de los cortes se procede a el ensamble de las partes para ir estructurando y dando forma a la arquitectura del germinador.



Anexo 3 Ensamble y pintado de la base

Remachado de las piezas cortadas y el pegado de las paredes del marco del germinador, cabe mencionar que los cortes y dimensiones se los puede observar en el capítulo VI.



Anexo 4 Cortes conforme las dimensiones especificadas.

El ensamble del germinador se lo realiza agregando los componentes electrónicos una vez terminado el armado de la estructura que edificara el proyecto.



Anexo 5 Ensamble del invernadero

El diseño del tablero de control y conexiones donde se colocarán los componentes dejando de manera estética y funcional dentro del germinador.



Anexo 6 Tablero para los componentes electrónicos.



Anexo 7 Prueba de sensores y actuadores

En este punto se carga el programa en el Arduino el cual se encuentra alimentado por la computadora, posteriormente se añaden los sensores, actuadores y se les da la respectiva alimentación para su funcionamiento.

Ensamble y conexiones



Anexo 8 Conexiones de control y potencia

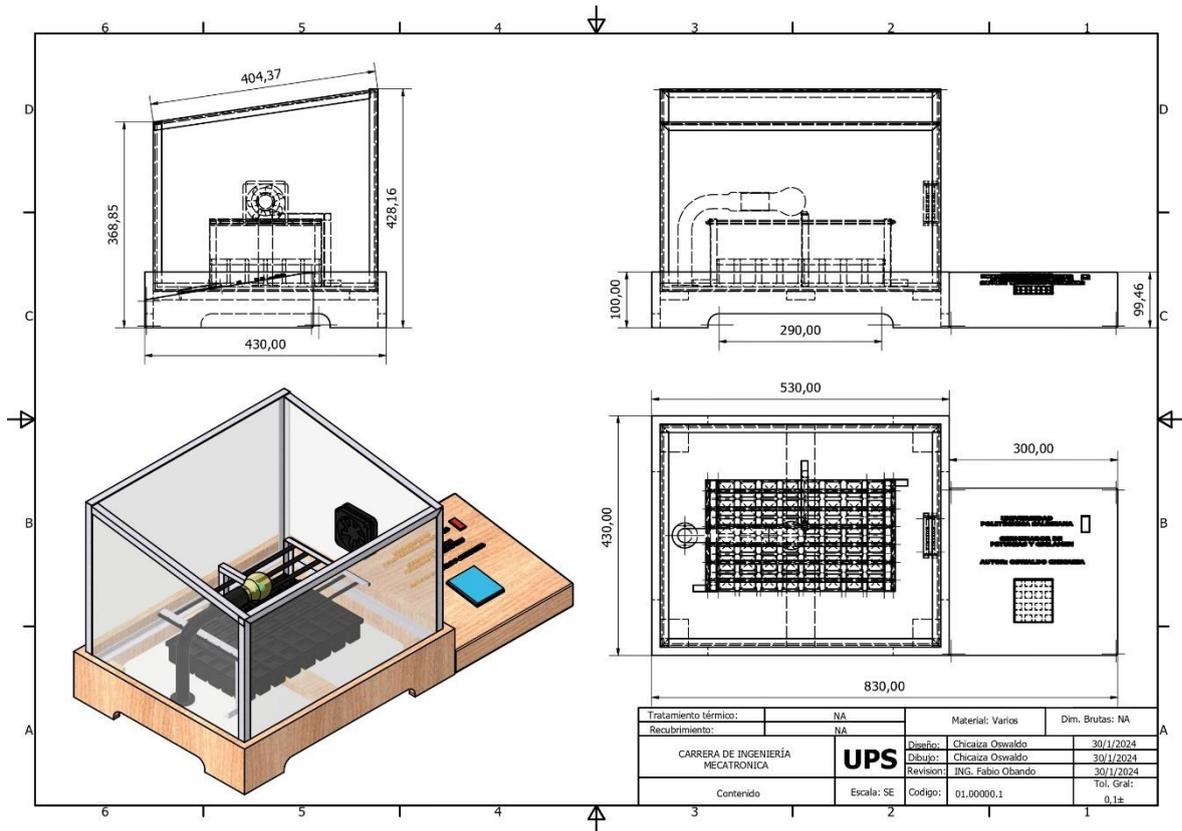


Anexo 9 Puesta en marcha

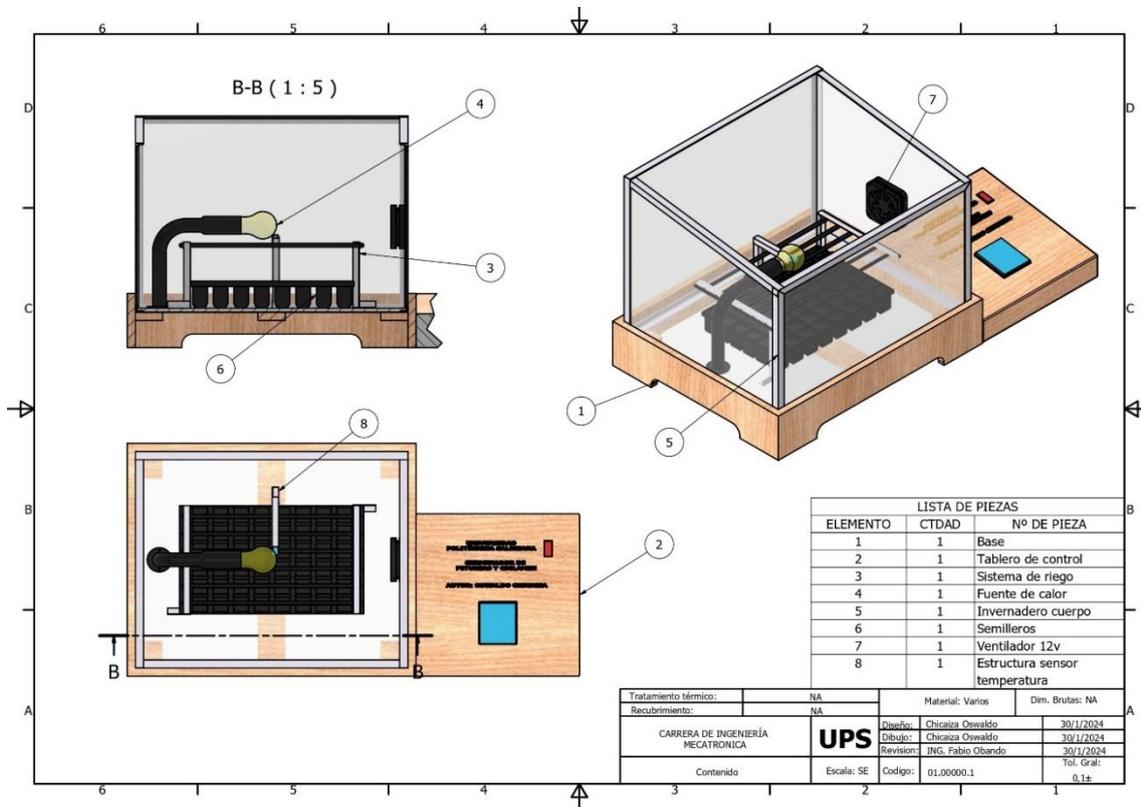
Para este apartado iniciamos con las semillas de petunias colocando en las bandejas junto con el sustrato, ubicamos y nos aseguramos que todo se encuentre en orden para dar inicio al procedimiento de germinación.

Planos de la estructura

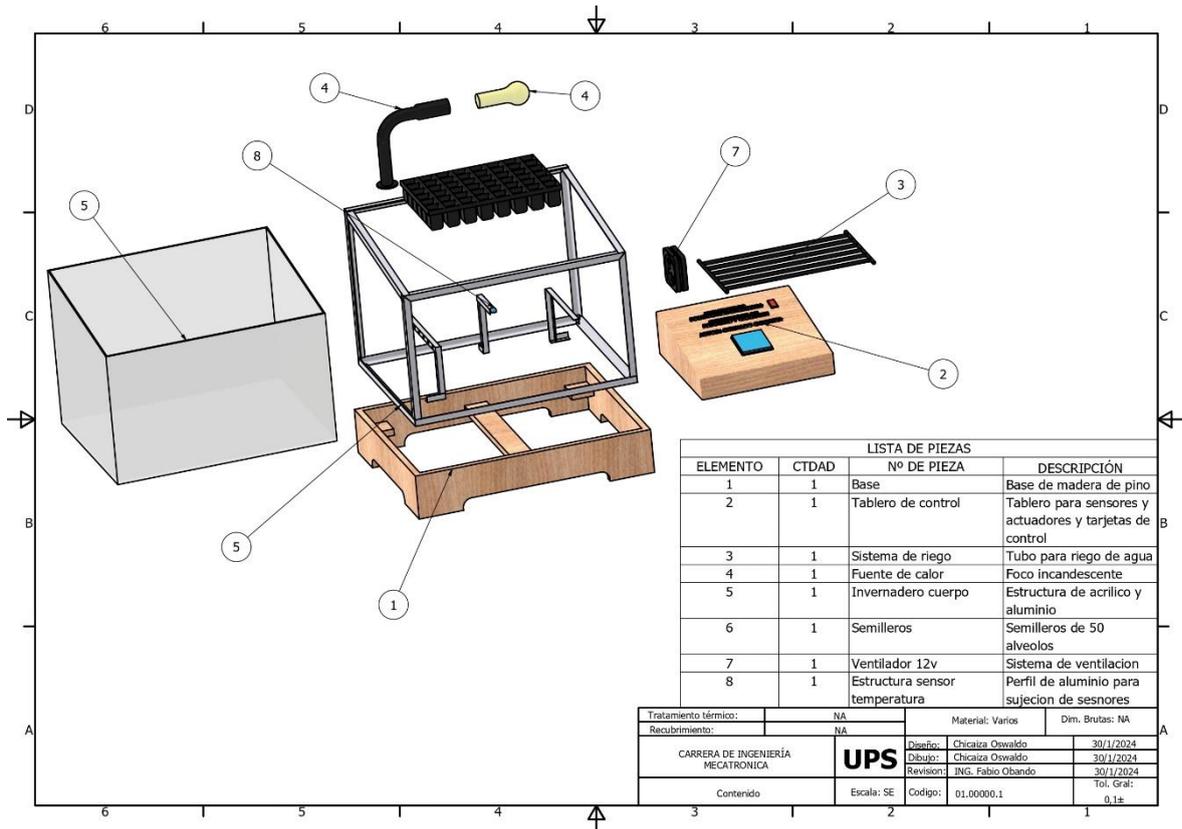
Según los planos cuenta las dimensiones que llevara el germinador y las conexiones indicadas.



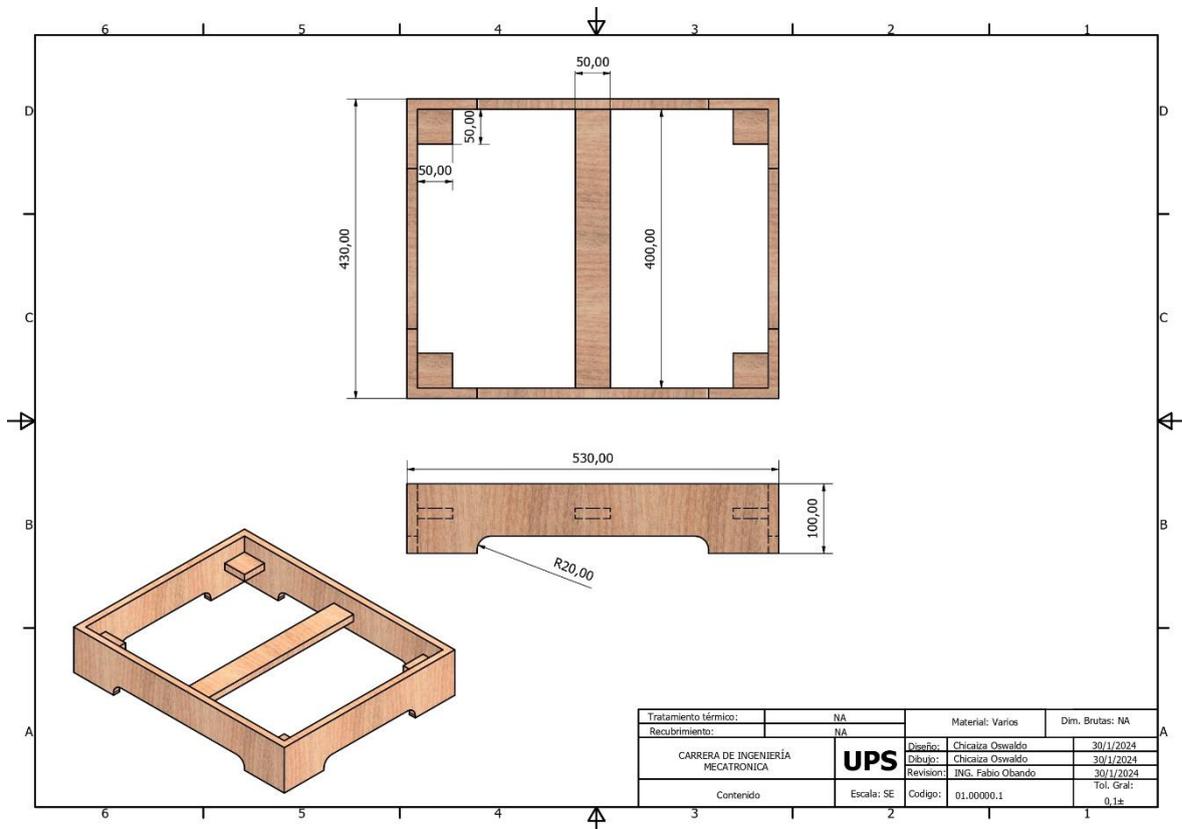
Anexo 10 Dimensiones generales



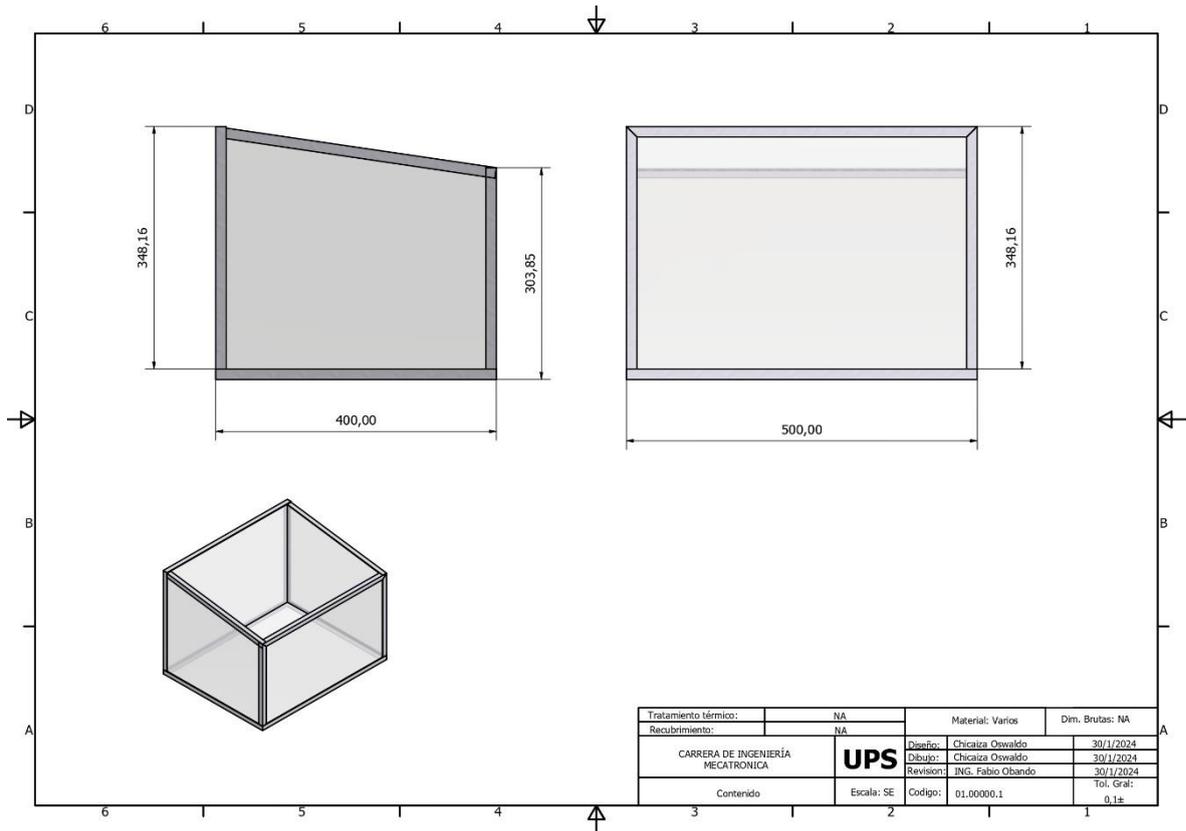
Anexo 11 Partes de la estructura



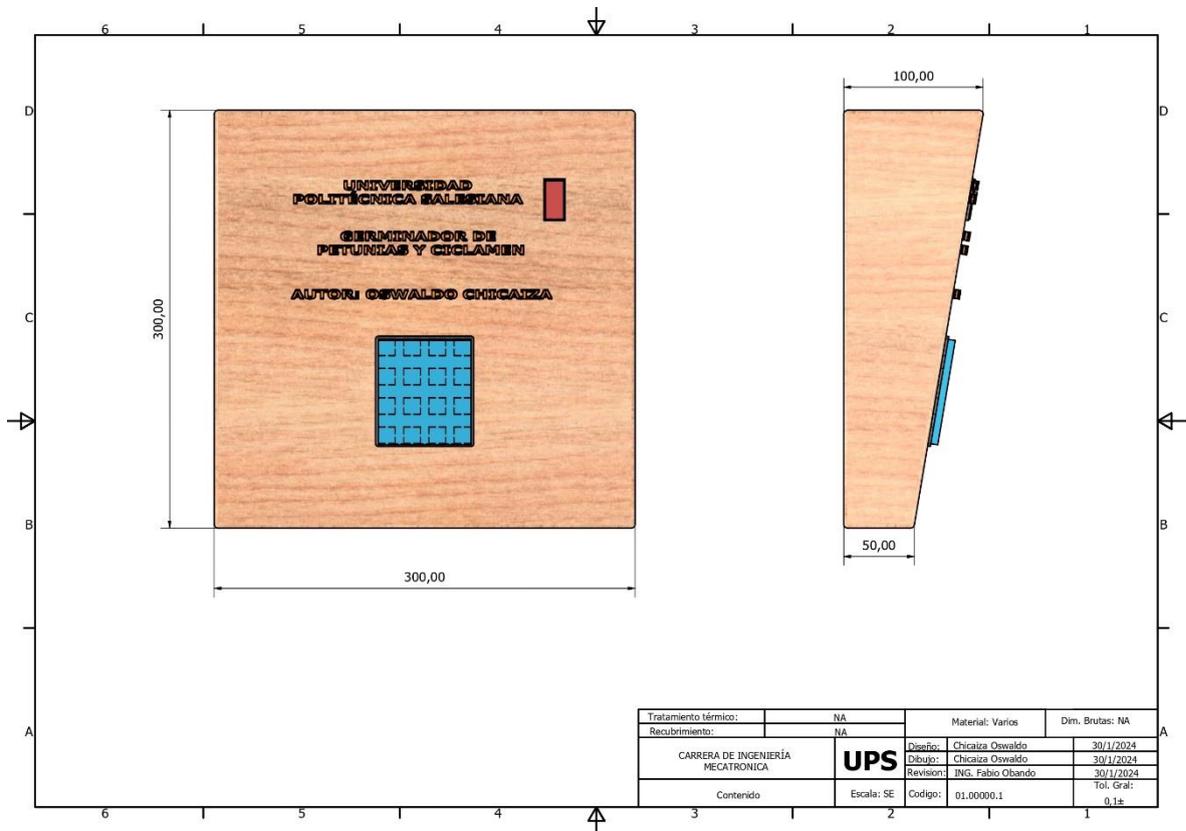
Anexo 12 Planos de despiece



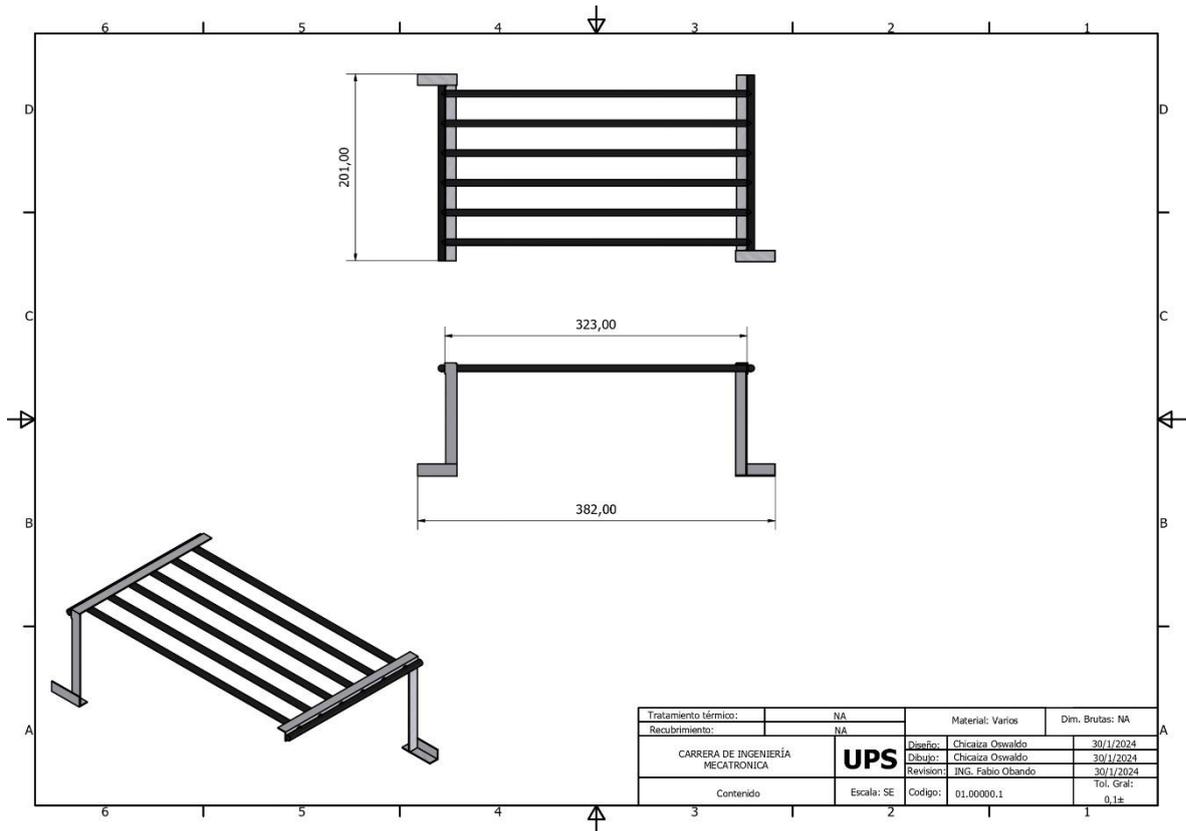
Anexo 13 Base de la estructura



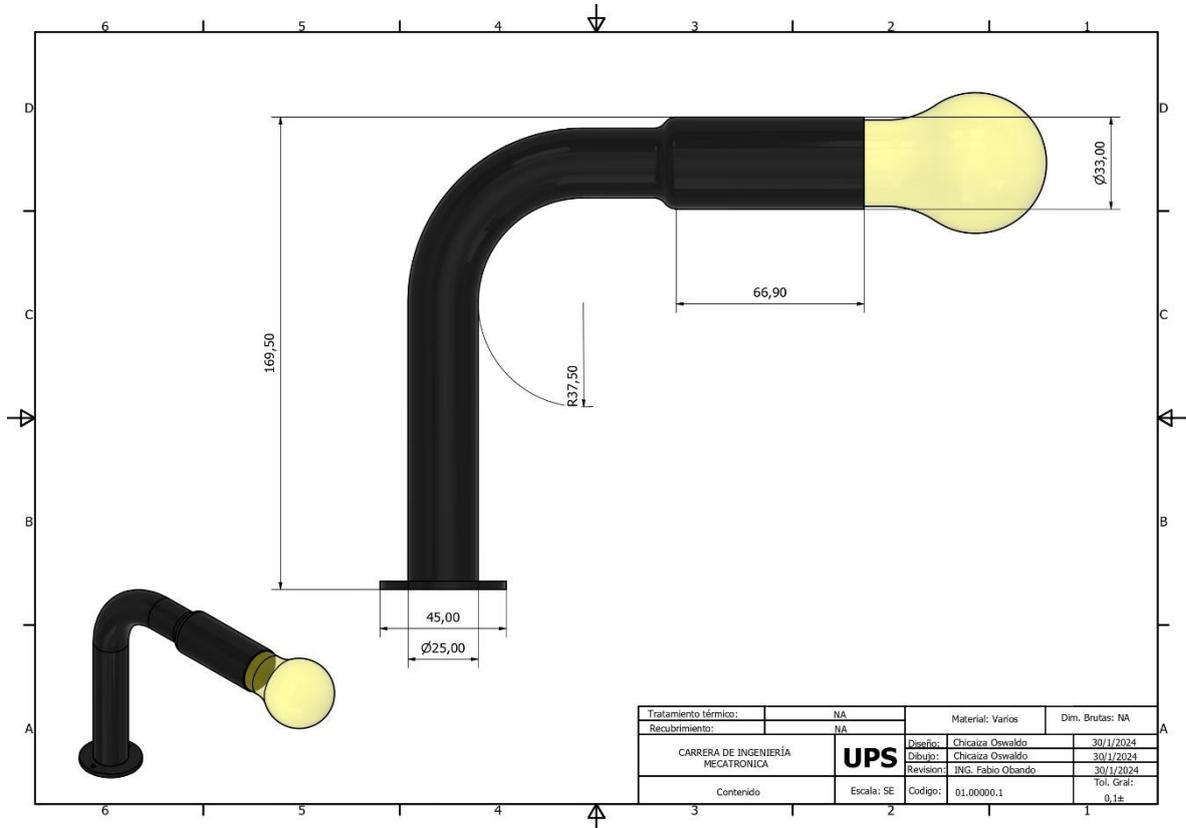
Anexo 14 Estructura del invernadero



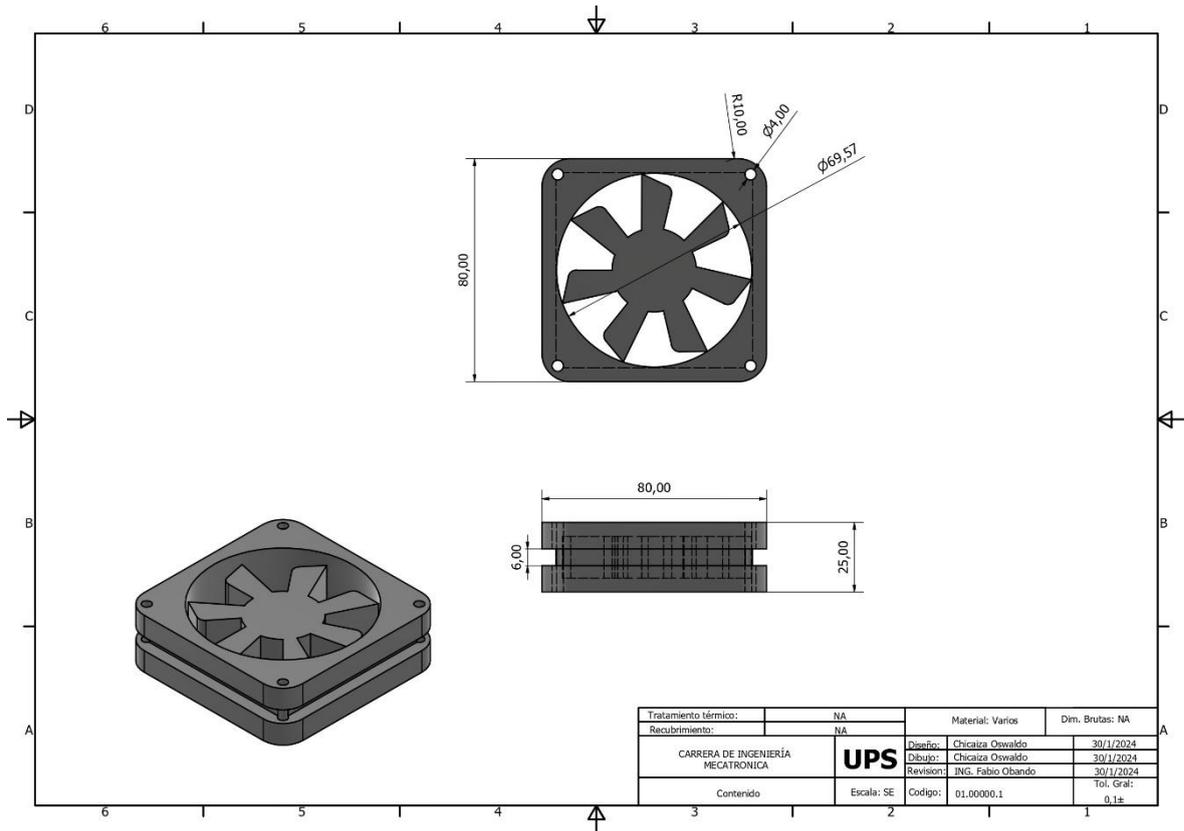
Anexo 15 Panel de control



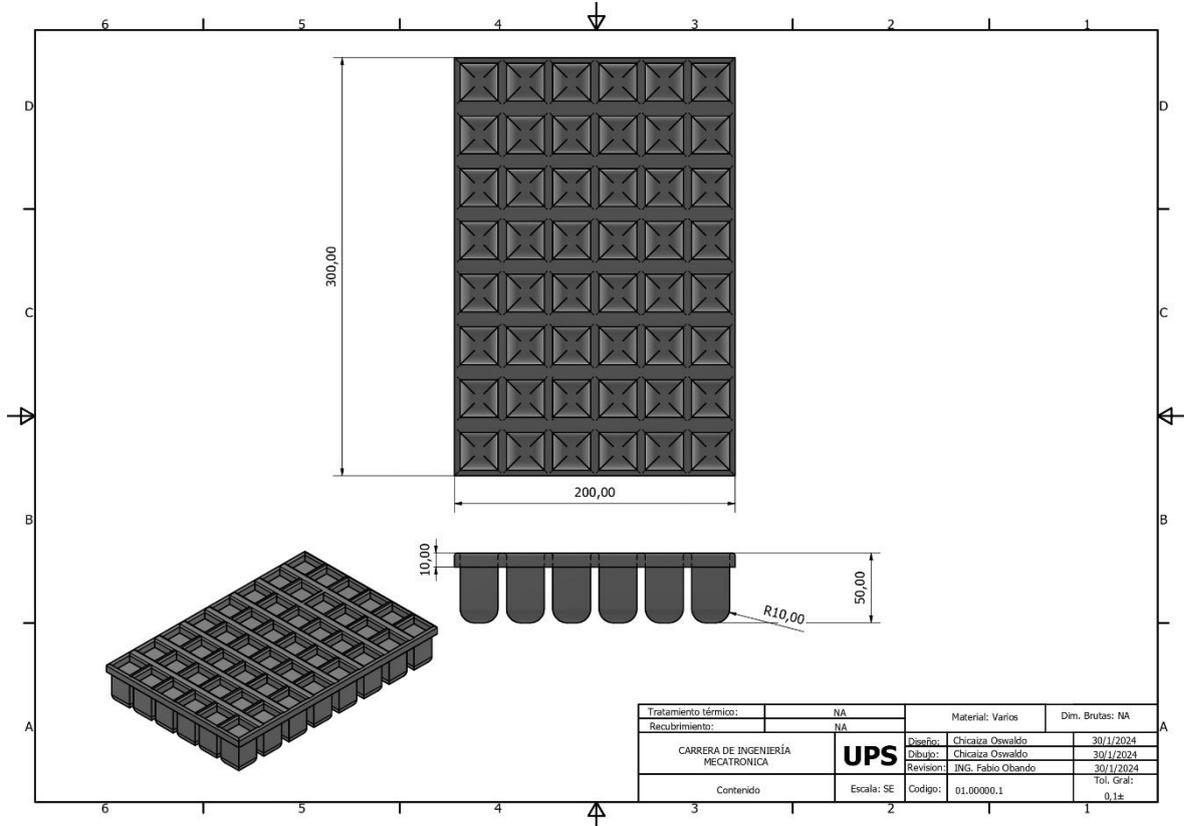
Anexo 16 Sistema de goteo



Anexo 17 Fuente de calor



Anexo 18 Ventilador



Anexo 19 Semilleros

Anexo 20 Código implementado para la programación del dispositivo

```

#include "fis_header.h"
#include "Nextion.h"
#include "max6675.h"
#include "DHT.h"
#include "Dimmer.h"

#define FIS_TYPE float
#define FIS_RESOLUTION 101
#define FIS_MIN -3.4028235E+38
#define FIS_MAX 3.4028235E+38
typedef FIS_TYPE(*_FIS_MF)(FIS_TYPE, FIS_TYPE*);
typedef FIS_TYPE(*_FIS_ARR_OP)(FIS_TYPE, FIS_TYPE);
typedef FIS_TYPE(*_FIS_ARR)(FIS_TYPE*, int, _FIS_ARR_OP);

// Number of inputs to the fuzzy inference system
const int fis_gcI = 3;
// Number of outputs to the fuzzy inference system
const int fis_gcO = 6;
// Number of rules to the fuzzy inference system
const int fis_gcR = 22;

FIS_TYPE g_fisInput[fis_gcI];
FIS_TYPE g_fisOutput[fis_gcO];

//240x320 pantalla
/*   Componente (Editor Nextion)           Objeto (Biblioteca Arduino)
      Texto                               NexText
      Número                             NexNumber
      Botón                               NexButton
      Barra de progreso                   NexProgressBar
      Corte                               NexCorte
      Deslizador                           NexDeslizador
      Botón de doble estado               NexDSButton
      Casilla de verificación             NexCheckbox */
/** +-----+-----PUERTOS dimmer1-----+-----+
-----+
* | Board      | INPUT Pin      | OUTPUT Pin      |
* |            | Zero-Cross     |                  |
* +-----+-----+-----+-----+
* | Mega       | D2 (NOT CHANGABLE) | D0-D1, D3-D70  |
* +-----+-----+-----+-----+
* | Uno        | D2 (NOT CHANGABLE) | D0-D1, D3-D20  |
* +-----+-----+-----+-----+
+*/

//-----
unsigned long previousMillis = 0;
const long interval = 1000; //1 segundo
unsigned long previousMillis2 = 0;
unsigned long interval2 = 0;
unsigned long previousMillis3 = 0;
unsigned long interval3 = 3000;

```

```

//---SALIDAS-----
#define BOMBARELE 10
#define dimmerPIN 12
#define ventiladorPIN 9

//DHT11(Sensor 1)
#define DHTPIN 3 // Pin de datos para el DHT11
#define DHTTYPE DHT11 // Tipo de sensor, puedes cambiar por
DHT22
DHT sensorDHT(DHTPIN, DHTTYPE); // Crear el objeto de tipo DHT

//hw-50(Sensor 2)
#define thermoSO 4 // Pin
#define thermoCS 5 // Pin
#define thermoSCK 6 // Pin

int humedad;
float temperaturatC;
float temperaturatF;
float sensacionC;
float sensacionF;

//hw-50(Sensor 2)
float temperatura2C;
float temperatura2F;
MAX6675 thermocouple(thermoSCK, thermoCS, thermoSO);

Dimmer dimmer(dimmerPIN, DIMMER_RAMP, 4);
int outDimVal = 0;
bool iniDimmer = true;

// Declaramos las variables
int tiempoRiego = 10000; // Tiempo de riego en milisegundos
int contadorRiegos = 0; // Contador de riegos
bool estBomba = false; //false= apagado, true= encendido
int horaActual; // Hora actual
int humedadTierra; //++% humedad tierra
int porcentajeRequerido = 60; /**Conf. humedad requerida en el suelo
float temperaturaProm;
int modo = 1; //1 petunia(defecto), 0 ciclamen
float potencFoco, riego, ventOut;

uint32 t camb1=0,camb2=0,onpage1=0,onpage2=0;
// Declare Nextion objects // (page id, component id, component name)
NexText temp1 = NexText(1, 6, "t4");
NexText humeAmb1 = NexText(1, 7, "t5");
NexText humeSuel = NexText(1, 8, "t6");
NexText temp2 = NexText(2, 6, "t42");
NexText humeAmb2 = NexText(2, 7, "t52");
NexText humeSue2 = NexText(2, 8, "t62");
NexButton modPetuIni = NexButton(0, 3, "b1");
NexButton modCiclIni = NexButton(0, 2, "b0");
NexDSButton cambDePaC = NexDSButton(1, 10, "btAct1");
NexDSButton cambDeCaP = NexDSButton(2, 10, "btAct2");

```

```

NexDSButton bt0 = NexDSButton(1, 9, "bt0");
NexDSButton bt02 = NexDSButton(2, 9, "bt02");
NexDSButton n0 = NexDSButton(1, 12, "n0");
NexDSButton n02 = NexDSButton(2, 12, "n0");

// Register objects to the touch event list
NexTouch* nex_listen_list[] = {
    &modPetuIni, &modCiclIni, &cambDePaC, &cambDeCaP,
    NULL
};

void modoInicio(void* ptr);
void modoCambio(void* ptr);
void modoInicio2(void* ptr);
void modoCambio2(void* ptr);

void leerDatos();

void setup() {
    Serial.begin(9600);
    sensorDHT.begin();

    nexInit();
    modPetuIni.attachPop(modoInicio);
    modCiclIni.attachPop(modoInicio2);
    cambDePaC.attachPop(modoCambio, &cambDePaC);
    cambDeCaP.attachPop(modoCambio2, &cambDeCaP);

    pinMode(ventiladorPIN, OUTPUT);
    digitalWrite(ventiladorPIN, HIGH);
    pinMode(BOMBARELE, OUTPUT);
    digitalWrite(BOMBARELE, LOW);
    dimmer.begin();
    dimmer.setMinimum(0);
    delay(1000);
}

// Loop routine runs over and over again forever:
void loop() {

    nexLoop(nex_listen_list);

    cambDePaC.getValue(&camb1);n0.getValue(&onpage1);
    cambDeCaP.getValue(&camb2);n02.getValue(&onpage2);

    if((camb1==1&&(onpage1==1)){
modo=1;
    }else if((camb2==1)&&(onpage2==1)){
modo=0;
    }

    unsigned long currentMillis = millis();

```

```

if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;

    leerDatos();
    //Input
    g_fisInput[0] = temperatura2C;
    g_fisInput[1] = humedadTierra;
    g_fisInput[2] = humedad;

    if (modo == 1) {
        //Output1 Pet
        g_fisOutput[0] = 0;
        g_fisOutput[1] = 0;
        g_fisOutput[2] = 0;
        fis_evaluate(); //Fuzzy petunia
        potencFoco = g_fisOutput[0];
        riego = g_fisOutput[1];
        ventOut = g_fisOutput[2];
//Serial.println(".....CAMBIADO A MODO PETUNIA CON
EXITO.....");
    } else {
        //Output2 Cicl
        g_fisOutput[3] = 0;
        g_fisOutput[4] = 0;
        g_fisOutput[5] = 0;
        fis_evaluate(); //Fuzzy ciclamen
        potencFoco = g_fisOutput[3];
        riego = g_fisOutput[4];
        ventOut = g_fisOutput[5];
// Serial.println(".....CAMBIADO A MODO CICLAMEN
CON EXITO.....");
    }

    datosSerialMathlab();

    dimmer1Ctrl();
    ventilacion();
    riegoSuelo();
    enviarDatosPantalla();
}
}

/*****/
void modoInicio(void* ptr) {
    modo = 1; //petunia
}
void modoInicio2(void* ptr) {
    modo = 0; //ciclamen
}
void modoCambio(void* ptr) {
    uint32_t Estado_dual;

    if (Estado_dual) {modo = 0;}
    else {modo = 1;}
}

```

```

}
void modoCambio2(void* ptr) {
    uint32_t Estado_dual;

    if (Estado_dual) {modo = 1;}
    else {modo = 0;}
}
/*****/

void leerDatos() {
    int ValorEnviar = 0;
    //Sensor1
    humedad = sensorDHT.readHumidity();
    temperatur1C = sensorDHT.readTemperature();
    temperatur1F = sensorDHT.readTemperature(true);

    sensacionC = sensorDHT.computeHeatIndex(temperatur1C, humedad, false);
    sensacionF = sensorDHT.computeHeatIndex(temperatur1F, humedad);

    //Sensor2
    temperatura2C = thermocouple.readCelsius();
    temperatura2F = thermocouple.readFahrenheit();

    //SensorTierraHumedad
    humedadTierra = analogRead(A0);
    humedadTierra = map(humedadTierra, 1023, 0, 0, 100);

    temperaturaProm = (temperatura2C + temperatur1C) / 2;
}

void datosSerialMathlab() {
    //Entradas
    //Serial.print("temperatura:");
    Serial.println(temperaturaProm);
    //Serial.print("humedadTierra:");
    Serial.println(humedadTierra);
    //Serial.print("humedadAmbiente:");
    Serial.println(humedad);
    //Serial.println("-----");
    // Set output vlaue: potencia_foco
    Serial.println(potencFoco);
    // Set output vlaue: riego
    //Serial.print("riego:");
    Serial.println(riego);
    // Set output vlaue: ventilacion
    // Serial.print("ventilacion:");
    Serial.println(ventOut);
}

void enviarDatosPantalla() {
    //-----*****
    String ValorTempl = String(/*temperatur1C*/ temperaturaProm) + " C";
    int str_len = ValorTempl.length() + 1;
}

```

```

char char_array[str_len];
ValorTemp1.toCharArray(char_array, str_len);
templ.setText(char_array);

String ValorTemp2 = String(temperaturaProm) + " C";
int str_len2 = ValorTemp2.length() + 1;
char char_array2[str_len2];
ValorTemp2.toCharArray(char_array2, str_len2);
temp2.setText(char_array2);

String ValorhumeAmb1 = String(humedad) + " %";
int str_lenh = ValorhumeAmb1.length() + 1;
char char_arrayh[str_lenh];
ValorhumeAmb1.toCharArray(char_arrayh, str_lenh);
humeAmb1.setText(char_arrayh);

String ValorhumeAmb2 = String(humedad) + " %";
int str_lenh2 = ValorhumeAmb2.length() + 1;
char char_arrayh2[str_lenh2];
ValorhumeAmb2.toCharArray(char_arrayh2, str_lenh2);
humeAmb2.setText(char_arrayh2);

String ValorhumeSue1 = String(humedadTierra) + " %";
int str_lenhs = ValorhumeSue1.length() + 1;
char char_arrayhs[str_lenhs];
ValorhumeSue1.toCharArray(char_arrayhs, str_lenhs);
humeSue1.setText(char_arrayhs);

String ValorhumeSue2 = String(humedadTierra) + " %";
int str_lenhs2 = ValorhumeSue2.length() + 1;
char char_arrayhs2[str_lenhs2];
ValorhumeSue2.toCharArray(char_arrayhs2, str_lenhs2);
humeSue2.setText(char_arrayhs2);

}

void dimmer1Ctrl() {
    int potencia = potencFoco;

    if (iniDimmer) {
        iniDimmer = false;
        dimmer.set(potencia);
    } else {
        if (potencia != dimmer.getValue()) {
            dimmer.set(potencia);
        }
    }
    if (potencia < 5) {
        dimmer.off();
    } else {
        dimmer.on();
    }
}
}

```

```

void riegoSuelo() {
    // Actualizamos la hora actual
    horaActual = millis() / 3600000;
    interval2 = riego;
    // Si el suelo está seco, riegos la planta
    if (humedadTierra < porcentajeRequerido && contadorRiegos < 3 &&
estBomba == false) {
        estBomba = true;
        contadorRiegos = contadorRiegos + 1;
        //Serial.println("El suelo está seco. Regando...");
        // Abre el relé
        digitalWrite(BOMBARELE, HIGH);
        delay(interval2);
        //Serial.println(contadorRiegos);

    }
    if (humedadTierra > porcentajeRequerido) {
        estBomba = false;
        // Cierra el relé
        digitalWrite(BOMBARELE, LOW);
    }
    // Si la hora actual es la medianoche y si el contador de riegos es
mayor que 0. Si ambas condiciones son verdaderas, entonces reiniciamos el
contador a 0.
    if (horaActual % 24 == 0 && contadorRiegos > 3) {
        contadorRiegos = 0;
    }
}

void ventilacion() {
    if (ventOut < 50) {
        digitalWrite(ventiladorPIN, HIGH);
        bt0.setValue(0);
        bt02.setValue(0);
        //Serial.println("Apagado");
    } else {
        digitalWrite(ventiladorPIN, LOW);

        bt0.setValue(1);
        bt02.setValue(1);
        //Serial.println("Encendido");
    }
}

/*****
// Support functions for Fuzzy Inference System
/*****
// Trapezoidal Member Function
FIS_TYPE fis_trapmf(FIS_TYPE x, FIS_TYPE* p)
{
    FIS_TYPE a = p[0], b = p[1], c = p[2], d = p[3];

```

```

    FIS_TYPE t1 = ((x <= c) ? 1 : ((d < x) ? 0 : ((c != d) ? ((d - x) /
(d - c) : 0)));
    FIS_TYPE t2 = ((b <= x) ? 1 : ((x < a) ? 0 : ((a != b) ? ((x - a) /
(b - a) : 0)));
    return (FIS_TYPE) min(t1, t2);
}

// Triangular Member Function
FIS_TYPE fis_trmf(FIS_TYPE x, FIS_TYPE* p)
{
    FIS_TYPE a = p[0], b = p[1], c = p[2];
    FIS_TYPE t1 = (x - a) / (b - a);
    FIS_TYPE t2 = (c - x) / (c - b);
    if ((a == b) && (b == c)) return (FIS_TYPE) (x == a);
    if (a == b) return (FIS_TYPE) (t2*(b <= x)*(x <= c));
    if (b == c) return (FIS_TYPE) (t1*(a <= x)*(x <= b));
    t1 = min(t1, t2);
    return (FIS_TYPE) max(t1, 0);
}

FIS_TYPE fis_min(FIS_TYPE a, FIS_TYPE b)
{
    return min(a, b);
}

FIS_TYPE fis_max(FIS_TYPE a, FIS_TYPE b)
{
    return max(a, b);
}

FIS_TYPE fis_array_operation(FIS_TYPE *array, int size, _FIS_ARR_OP
pfnOp)
{
    int i;
    FIS_TYPE ret = 0;

    if (size == 0) return ret;
    if (size == 1) return array[0];

    ret = array[0];
    for (i = 1; i < size; i++)
    {
        ret = (*pfnOp)(ret, array[i]);
    }

    return ret;
}

//*****
// Data for Fuzzy Inference System
//*****
// Pointers to the implementations of member functions
_FIS_MF fis_gMF[] =

```

```

{
    fis_trapmf, fis_trimf
};

// Count of member function for each Input
int fis_gIMFCount[] = { 3, 3, 3 };

// Count of member function for each Output
int fis_gOMFCount[] = { 4, 4, 2, 4, 4, 2 };

// Coefficients for the Input Member Functions
FIS_TYPE fis_gMFI0Coeff1[] = { 0, 0, 12, 20 };
FIS_TYPE fis_gMFI0Coeff2[] = { 18, 22, 30 };
FIS_TYPE fis_gMFI0Coeff3[] = { 27, 30, 50, 50 };
FIS_TYPE* fis_gMFI0Coeff[] = { fis_gMFI0Coeff1, fis_gMFI0Coeff2,
    fis_gMFI0Coeff3 };
FIS_TYPE fis_gMFI1Coeff1[] = { -10.0352112676056, -10.0352112676056,
    19.9647887323944, 49.9647887323944 };
FIS_TYPE fis_gMFI1Coeff2[] = { 31.4, 54.137323943662, 70 };
FIS_TYPE fis_gMFI1Coeff3[] = { 65, 75, 100, 100 };
FIS_TYPE* fis_gMFI1Coeff[] = { fis_gMFI1Coeff1, fis_gMFI1Coeff2,
    fis_gMFI1Coeff3 };
FIS_TYPE fis_gMFI2Coeff1[] = { 0, 0, 30, 50 };
FIS_TYPE fis_gMFI2Coeff2[] = { 45, 60, 75 };
FIS_TYPE fis_gMFI2Coeff3[] = { 70, 85, 100, 100 };
FIS_TYPE* fis_gMFI2Coeff[] = { fis_gMFI2Coeff1, fis_gMFI2Coeff2,
    fis_gMFI2Coeff3 };
FIS_TYPE* fis_gMFI0Coeff[] = { fis_gMFI0Coeff, fis_gMFI1Coeff,
    fis_gMFI2Coeff };

// Coefficients for the Output Member Functions
FIS_TYPE fis_gMFO0Coeff1[] = { 5, 20, 39.55 };
FIS_TYPE fis_gMFO0Coeff2[] = { 35.2816901408451, 55.2816901408451,
    75.2816901408451 };
FIS_TYPE fis_gMFO0Coeff3[] = { 71.5, 83.5, 100, 100 };
FIS_TYPE fis_gMFO0Coeff4[] = { 0, 0, 1, 5 };
FIS_TYPE* fis_gMFO0Coeff[] = { fis_gMFO0Coeff1, fis_gMFO0Coeff2,
    fis_gMFO0Coeff3, fis_gMFO0Coeff4 };
FIS_TYPE fis_gMFO1Coeff1[] = { 0.5, 1, 1.5 };
FIS_TYPE fis_gMFO1Coeff2[] = { 1, 2, 3 };
FIS_TYPE fis_gMFO1Coeff3[] = { 3, 5, 5 };
FIS_TYPE fis_gMFO1Coeff4[] = { 0, 0, 1 };
FIS_TYPE* fis_gMFO1Coeff[] = { fis_gMFO1Coeff1, fis_gMFO1Coeff2,
    fis_gMFO1Coeff3, fis_gMFO1Coeff4 };
FIS_TYPE fis_gMFO2Coeff1[] = { 50, 100, 100 };
FIS_TYPE fis_gMFO2Coeff2[] = { 0, 0, 70 };
FIS_TYPE* fis_gMFO2Coeff[] = { fis_gMFO2Coeff1, fis_gMFO2Coeff2 };
FIS_TYPE fis_gMFO3Coeff1[] = { 5, 10, 20 };
FIS_TYPE fis_gMFO3Coeff2[] = { 20, 40, 60 };
FIS_TYPE fis_gMFO3Coeff3[] = { 71.5, 83.5, 100, 100 };
FIS_TYPE fis_gMFO3Coeff4[] = { 0, 0, 1, 5 };
FIS_TYPE* fis_gMFO3Coeff[] = { fis_gMFO3Coeff1, fis_gMFO3Coeff2,
    fis_gMFO3Coeff3, fis_gMFO3Coeff4 };
FIS_TYPE fis_gMFO4Coeff1[] = { 0.5, 1, 1.5 };

```

```

FIS_TYPE fis_gMFO4Coeff2[] = { 1, 2, 3 };
FIS_TYPE fis_gMFO4Coeff3[] = { 3, 5, 5 };
FIS_TYPE fis_gMFO4Coeff4[] = { 0, 0, 1 };
FIS_TYPE* fis_gMFO4Coeff[] = { fis_gMFO4Coeff1, fis_gMFO4Coeff2,
fis_gMFO4Coeff3, fis_gMFO4Coeff4 };
FIS_TYPE fis_gMFO5Coeff1[] = { 50, 100, 100 };
FIS_TYPE fis_gMFO5Coeff2[] = { 0, 0, 70 };
FIS_TYPE* fis_gMFO5Coeff[] = { fis_gMFO5Coeff1, fis_gMFO5Coeff2 };
FIS_TYPE** fis_gMFOCoeff[] = { fis_gMFO0Coeff, fis_gMFO1Coeff,
fis_gMFO2Coeff, fis_gMFO3Coeff, fis_gMFO4Coeff, fis_gMFO5Coeff };

// Input membership function set
int fis_gMFI0[] = { 0, 1, 0 };
int fis_gMFI1[] = { 0, 1, 0 };
int fis_gMFI2[] = { 0, 1, 0 };
int* fis_gMFI[] = { fis_gMFI0, fis_gMFI1, fis_gMFI2};

// Output membership function set
int fis_gMFO0[] = { 1, 1, 0, 0 };
int fis_gMFO1[] = { 1, 1, 1, 1 };
int fis_gMFO2[] = { 1, 1 };
int fis_gMFO3[] = { 1, 1, 0, 0 };
int fis_gMFO4[] = { 1, 1, 1, 1 };
int fis_gMFO5[] = { 1, 1 };
int* fis_gMFO[] = { fis_gMFO0, fis_gMFO1, fis_gMFO2, fis_gMFO3,
fis_gMFO4, fis_gMFO5};

// Rule Weights
FIS_TYPE fis_gRWeight[] = { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1 };

// Rule Type
int fis_gRType[] = { 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1 };

// Rule Inputs
int fis_gRI0[] = { 3, 0, 0 };
int fis_gRI1[] = { 0, 1, 0 };
int fis_gRI2[] = { 0, 0, 1 };
int fis_gRI3[] = { 1, 0, 0 };
int fis_gRI4[] = { 1, 3, 0 };
int fis_gRI5[] = { 2, 2, 2 };
int fis_gRI6[] = { 3, 1, 2 };
int fis_gRI7[] = { 1, 3, 3 };
int fis_gRI8[] = { 2, -3, 0 };
int fis_gRI9[] = { -3, 3, 0 };
int fis_gRI10[] = { 2, 2, 3 };
int fis_gRI11[] = { 3, 0, 0 };
int fis_gRI12[] = { 0, 1, 0 };
int fis_gRI13[] = { 0, 0, 1 };
int fis_gRI14[] = { 1, 0, 0 };
int fis_gRI15[] = { 1, 3, 0 };
int fis_gRI16[] = { 2, 2, 2 };
int fis_gRI17[] = { 3, 1, 2 };

```

```

int fis_gRI18[] = { 1, 3, 3 };
int fis_gRI19[] = { 2, -3, 0 };
int fis_gRI20[] = { -3, 3, 0 };
int fis_gRI21[] = { 2, 2, 3 };
int* fis_gRI[] = { fis_gRI0, fis_gRI1, fis_gRI2, fis_gRI3, fis_gRI4,
fis_gRI5, fis_gRI6, fis_gRI7, fis_gRI8, fis_gRI9, fis_gRI10, fis_gRI11,
fis_gRI12, fis_gRI13, fis_gRI14, fis_gRI15, fis_gRI16, fis_gRI17,
fis_gRI18, fis_gRI19, fis_gRI20, fis_gRI21 };

// Rule Outputs
int fis_gRO0[] = { 4, 2, 2, 0, 0, 0 };
int fis_gRO1[] = { 1, 3, 1, 0, 0, 0 };
int fis_gRO2[] = { 4, 2, 1, 0, 0, 0 };
int fis_gRO3[] = { 3, 2, 1, 0, 0, 0 };
int fis_gRO4[] = { 3, 4, 2, 0, 0, 0 };
int fis_gRO5[] = { 4, 4, 1, 0, 0, 0 };
int fis_gRO6[] = { 2, 3, 2, 0, 0, 0 };
int fis_gRO7[] = { 3, 4, 2, 0, 0, 0 };
int fis_gRO8[] = { 4, 4, 2, 0, 0, 0 };
int fis_gRO9[] = { 4, 4, 1, 0, 0, 0 };
int fis_gRO10[] = { 4, 0, 0, 0, 0, 0 };
int fis_gRO11[] = { 0, 0, 0, 4, 2, 2 };
int fis_gRO12[] = { 0, 0, 0, 1, 3, 1 };
int fis_gRO13[] = { 0, 0, 0, 4, 2, 1 };
int fis_gRO14[] = { 0, 0, 0, 3, 2, 1 };
int fis_gRO15[] = { 0, 0, 0, 3, 4, 2 };
int fis_gRO16[] = { 0, 0, 0, 4, 4, 1 };
int fis_gRO17[] = { 0, 0, 0, 2, 3, 2 };
int fis_gRO18[] = { 0, 0, 0, 3, 4, 2 };
int fis_gRO19[] = { 0, 0, 0, 4, 4, 2 };
int fis_gRO20[] = { 0, 0, 0, 4, 4, 1 };
int fis_gRO21[] = { 0, 0, 0, 4, 0, 0 };
int* fis_gRO[] = { fis_gRO0, fis_gRO1, fis_gRO2, fis_gRO3, fis_gRO4,
fis_gRO5, fis_gRO6, fis_gRO7, fis_gRO8, fis_gRO9, fis_gRO10, fis_gRO11,
fis_gRO12, fis_gRO13, fis_gRO14, fis_gRO15, fis_gRO16, fis_gRO17,
fis_gRO18, fis_gRO19, fis_gRO20, fis_gRO21 };

// Input range Min
FIS_TYPE fis_gIMin[] = { 0, 0, 0 };

// Input range Max
FIS_TYPE fis_gIMax[] = { 50, 100, 100 };

// Output range Min
FIS_TYPE fis_gOMin[] = { 0, 0, 0, 0, 0, 0 };

// Output range Max
FIS_TYPE fis_gOMax[] = { 100, 5, 100, 100, 5, 100 };

/*****
// Data dependent support functions for Fuzzy Inference System
*****/
FIS_TYPE fis_MF_out(FIS_TYPE** fuzzyRuleSet, FIS_TYPE x, int o)
{

```

```

FIS_TYPE mfOut;
int r;

for (r = 0; r < fis_gcR; ++r)
{
    int index = fis_gRO[r][o];
    if (index > 0)
    {
        index = index - 1;
        mfOut = (fis_gMF[fis_gMFO[o][index]])(x,
fis_gMFOcoeff[o][index]);
    }
    else if (index < 0)
    {
        index = -index - 1;
        mfOut = 1 - (fis_gMF[fis_gMFO[o][index]])(x,
fis_gMFOcoeff[o][index]);
    }
    else
    {
        mfOut = 0;
    }

    fuzzyRuleSet[0][r] = fis_min(mfOut, fuzzyRuleSet[1][r]);
}
return fis_array_operation(fuzzyRuleSet[0], fis_gcR, fis_max);
}

FIS_TYPE fis_defuzz_centroid(FIS_TYPE** fuzzyRuleSet, int o)
{
    FIS_TYPE step = (fis_gOMax[o] - fis_gOMin[o]) / (FIS_RESOLUTION - 1);
    FIS_TYPE area = 0;
    FIS_TYPE momentum = 0;
    FIS_TYPE dist, slice;
    int i;

    // calculate the area under the curve formed by the MF outputs
    for (i = 0; i < FIS_RESOLUTION; ++i){
        dist = fis_gOMin[o] + (step * i);
        slice = step * fis_MF_out(fuzzyRuleSet, dist, o);
        area += slice;
        momentum += slice*dist;
    }

    return ((area == 0) ? ((fis_gOMax[o] + fis_gOMin[o]) / 2) : (momentum
/ area));
}

/*****
// Fuzzy Inference System
*****/
void fis_evaluate()
{
    FIS_TYPE fuzzyInput0[] = { 0, 0, 0 };

```

```

FIS_TYPE fuzzyInput1[] = { 0, 0, 0 };
FIS_TYPE fuzzyInput2[] = { 0, 0, 0 };
FIS_TYPE* fuzzyInput[fis_gcI] = { fuzzyInput0, fuzzyInput1,
fuzzyInput2, };
FIS_TYPE fuzzyOutput0[] = { 0, 0, 0, 0 };
FIS_TYPE fuzzyOutput1[] = { 0, 0, 0, 0 };
FIS_TYPE fuzzyOutput2[] = { 0, 0 };
FIS_TYPE fuzzyOutput3[] = { 0, 0, 0, 0 };
FIS_TYPE fuzzyOutput4[] = { 0, 0, 0, 0 };
FIS_TYPE fuzzyOutput5[] = { 0, 0 };
FIS_TYPE* fuzzyOutput[fis_gcO] = { fuzzyOutput0, fuzzyOutput1,
fuzzyOutput2, fuzzyOutput3, fuzzyOutput4, fuzzyOutput5, };
FIS_TYPE fuzzyRules[fis_gcR] = { 0 };
FIS_TYPE fuzzyFires[fis_gcR] = { 0 };
FIS_TYPE* fuzzyRuleSet[] = { fuzzyRules, fuzzyFires };
FIS_TYPE sW = 0;

// Transforming input to fuzzy Input
int i, j, r, o;
for (i = 0; i < fis_gcI; ++i)
{
    for (j = 0; j < fis_gIMFCount[i]; ++j)
    {
        fuzzyInput[i][j] =
            (fis_gMF[fis_gMFI[i][j]])(g_fisInput[i],
fis_gMFICoeff[i][j]);
    }
}

int index = 0;
for (r = 0; r < fis_gcR; ++r)
{
    if (fis_gRType[r] == 1)
    {
        fuzzyFires[r] = FIS_MAX;
        for (i = 0; i < fis_gcI; ++i)
        {
            index = fis_gRI[r][i];
            if (index > 0)
                fuzzyFires[r] = fis_min(fuzzyFires[r],
fuzzyInput[i][index - 1]);
            else if (index < 0)
                fuzzyFires[r] = fis_min(fuzzyFires[r], 1 -
fuzzyInput[i][-index - 1]);
            else
                fuzzyFires[r] = fis_min(fuzzyFires[r], 1);
        }
    }
    else
    {
        fuzzyFires[r] = FIS_MIN;
        for (i = 0; i < fis_gcI; ++i)
        {
            index = fis_gRI[r][i];

```

```

        if (index > 0)
            fuzzyFires[r] = fis_max(fuzzyFires[r],
fuzzyInput[i][index - 1]);
        else if (index < 0)
            fuzzyFires[r] = fis_max(fuzzyFires[r], 1 -
fuzzyInput[i][-index - 1]);
        else
            fuzzyFires[r] = fis_max(fuzzyFires[r], 0);
    }
}

fuzzyFires[r] = fis_gRWeight[r] * fuzzyFires[r];
sW += fuzzyFires[r];
}

if (sW == 0)
{
    for (o = 0; o < fis_gcO; ++o)
    {
        g_fisOutput[o] = ((fis_gOMax[o] + fis_gOMin[o]) / 2);
    }
}
else
{
    for (o = 0; o < fis_gcO; ++o)
    {
        g_fisOutput[o] = fis_defuzz_centroid(fuzzyRuleSet, o);
    }
}
}
}

```