



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO
CARRERA DE INGENIERÍA ELECTRÓNICA**

**AUTOMATIZACIÓN DE UN SISTEMA DE CULTIVO AEROPÓNICO
EXPERIMENTAL CON MONITOREO REMOTO MEDIANTE SISTEMAS
EMBEBIDOS PARA LA EMPRESA ASEMI**

Trabajo de titulación previo a la obtención del
Título de Ingeniero Electrónico

AUTOR: Bryan Steven Taboada Anchatipan

AUTOR: Víctor Alfonso Yanchapaxi Ganzino

TUTOR: Javier Arturo Rojas Urbano

Quito-Ecuador

2022

**CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE
TITULACIÓN**

Nosotros, Bryan Steven Taboada Anchatipan con documento de identificación N° 1726160292 y Víctor Alfonso Yanchapaxi Ganzino con documento de identificación N° 1727668020; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos 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, 15 de marzo del año 2022

Atentamente,



Bryan Steven Taboada Anchatipan

1726160292



Víctor Alfonso Yanchapaxi Ganzino

1727668020

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

Nosotros, Bryan Steven Taboada Anchatipan con documento de identificación No. 1726160292 y Víctor Alfonso Yanchapaxi Ganzino con documento de identificación No. 1727668020, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Automatización de un sistema de cultivo aeropónico experimental con monitoreo remoto mediante sistemas embebidos para la empresa Asemi”, el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

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

Quito, 15 de marzo del año 2022

Atentamente,



Bryan Steven Taboada Anchatipan

1726160292



Víctor Alfonso Yanchapaxi Ganzino

1727668020

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Javier Arturo Rojas Urbano con documento de identificación N° 0201898020, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: AUTOMATIZACIÓN DE UN SISTEMA DE CULTIVO AEROPÓNICO EXPERIMENTAL CON MONITOREO REMOTO MEDIANTE SISTEMAS EMBEBIDOS PARA LA EMPRESA ASEMI, realizado por Bryan Steven Taboada Anchatipan con documento de identificación N° 1726160292 y por Víctor Alfonso Yanchapaxi Ganzino con documento de identificación N° 1727668020, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 15 de marzo del año 2022

Atentamente,



Ing. Javier Arturo Rojas Urbano MSc.

0201898020

DEDICATORIA Y AGRADECIMIENTO

Primeramente, me gustaría dedicar este trabajo a mis padres Antonio Taboada y Alicia Anchatipan, quienes me han brindado apoyo incondicional no solo a lo largo de mi carrera sino también en otros proyectos de mi vida, quiero decirles que los quiero mucho y que les agradezco por todo. De igual manera a mi hermano Kevin Taboada, de quien admiro muchas cualidades y así mismo motivo a que siga superándose, y a mi novia Elizabeth Vega de quien he recibido mucho apoyo emocional en los momentos que lo necesitaba.

Después, quisiera agradecer a todos quienes han formado parte de mi camino a ser un profesional, a mis amigos más cercanos de la universidad, a mis amigos del colegio; también a todos los profesores que siempre han estado dispuestos a compartir sus conocimientos y ayudarnos cuando lo necesitábamos. Un agradecimiento especial a la Ing. Ana Palaquibay que se ha convertido en participe de este logro al haber patrocinado nuestro proyecto, también a nuestro tutor de tesis el Ing. Javier Rojas quien nos ofreció su apoyo desde el inicio de este trabajo y dio rumbo para que este culminara con éxito. Y finalmente a la Universidad Politécnica Salesiana que me ha otorgado el título de Ingeniero en Electrónica.

Bryan

DEDICATORIA Y AGRADECIMIENTO

Dedico este trabajo a mis padres Víctor Yanchapaxi y Yascila Ganzino por todo el apoyo que me han dado a lo largo de mi vida y de mi carrera, a mi hermana Kelly Yanchapaxi por su apoyo emocional e incondicional.

Agradezco a mis padres por brindarme su tiempo, conocimiento y experiencia para poder culminar con mi proyecto técnico.

Agradezco a mis amigos más cercanos Bryan Taboada, Viví Olalla, Jefferson Campos, Sara B, Juan Escobar y Enver Suarez, por su apoyo incondicional y emocional a lo largo mi carrera.

Agradezco a la Ing. Ana Palaquibay por el apoyo y auspicio que nos ha dado por parte de su empresa para el desarrollo de nuestro proyecto técnico.

Agradezco a nuestro tutor Ing. Javier Rojas, por el apoyo que nos ha dado a lo largo de la realización y culminación de nuestro proyecto técnico.

Agradezco a la Universidad Politécnica Salesiana ya que gracias a sus docentes que nos han guiado en la culminación de la ingeniería he podido adquirir nuevos conocimientos y habilidades.

Finalmente quiero agradecer a todos mis compañeros y amigos que creyeron en mí y mis capacidades.

Víctor

ÍNDICE GENERAL

CAPÍTULO 1

CULTIVOS AEROPÓNICOS.....	4
1.1 Funcionamiento de un cultivo aeropónico	5
1.1.1 Estructura de siembra	6
1.1.2 Tanque	6
1.1.3 Bomba de agua	6
1.1.4 Sistema de tuberías	7
1.1.5 Aspersor	7
1.1.6 Temporizador.....	8
1.2 Estructuras de cultivos aeropónicos	8
1.2.1 Estructura horizontal.....	8
1.2.2 Estructura vertical.....	9
1.3 Automatización de cultivos aeropónicos	10
1.4 Sistemas de monitoreo remoto aplicados a la agricultura.	11

CAPÍTULO 2

DISEÑO DEL SISTEMA AEROPÓNICO VERTICAL	13
2.1 Prototipo aeropónico.....	14
2.2 Controlador	17
2.3 Sistema de riego.....	19
2.3.1 Sensores de humedad.....	19
2.3.2 Sensor de temperatura y humedad relativa.....	22
2.3.3 Servoválvulas.....	24
2.3.4 Circuito de potencia.....	26
2.3.5 Bomba de agua	27
2.3.6 Lógica de operación del sistema	28
2.4 Sistema de iluminación	29

2.4.1	Sensor de luz.....	30
2.4.2	Circuito de potencia.....	32
2.4.3	Lógica de operación del sistema	33
2.5	Sistema de mezcla de nutrientes	34
2.5.1	Sensor de pH.....	34
2.5.2	Sensor de nivel del tanque	35
2.5.3	Sensor de nivel de las soluciones.....	38
2.5.4	Bombas para las soluciones	39
2.5.5	Lógica de operación del sistema	42
2.6	Sistema de vigilancia	45
2.6.1	Cámara	45
2.7	Sistema aeropónico	46
2.8	Base de datos.....	49
2.9	Sistema de monitoreo remoto.....	50
CAPÍTULO 3		
PRUEBAS Y RESULTADOS.....		54
3.1	Prueba del sistema de riego.....	54
3.2	Prueba del sistema de iluminación	56
3.3	Prueba del sistema de mezcla de nutrientes	58

INDICE DE FIGURAS

Figura 1.1: Diagrama de un cultivo aeropónico	5
Figura 1.2: Estructura horizontal para cultivos aeropónicos	9
Figura 1.3: Estructura vertical de cultivos aeropónicos	9
Figura 2.1: Diagrama de bloques del sistema aeropónico	13
Figura 2.2: Estructura del prototipo aeropónico	14
Figura 2.3: Disposición de los tubos de siembra	15
Figura 2.4: Tubos para aeroponía	15
Figura 2.5: Conexiones del sistema de tuberías.....	16
Figura 2.6: Prototipo aeropónico	17
Figura 2.7: Raspberry pi modelo 4	18
Figura 2.8: Conexión entre Raspberry pi y MCP3008	18
Figura 2.9: Diagrama de bloques del sistema de riego	19
Figura 2.10: Sensor de humedad YL-69.....	20
Figura 2.11: Conexión de los sensores YL-69 con Raspberry pi	21
Figura 2.12: Sensor DHT11	22
Figura 2.13: Transmisión de un 0 lógico con el protocolo “Single-wire Two-way”	23
Figura 2.14: Transmisión de un 1 lógico con el protocolo “Single-wire Two-way”	23
Figura 2.15: Conexión del DHT11 con la Raspberry pi	23
Figura 2.16: Señal de entrada del servomotor SG90	24
Figura 2.17: Conexión de las servoválvulas con la Raspberry pi	25
Figura 2.18: Circuito interno del driver ULN2003.....	26
Figura 2.19: Conexión de la bomba al circuito de potencia	27
Figura 2.20: Bomba de agua de 100W y 4500lph	27
Figura 2.21: Diagrama de flujo del sistema de riego	28
Figura 2.22: Diagrama de bloques del sistema de iluminación	29
Figura 2.23: Arreglo de focos para el prototipo aeropónico	30
Figura 2.24: LDR.....	30
Figura 2.25: Conexión del LDR con la Raspberry pi	31
Figura 2.26: Conexión de los focos con la Raspberry pi	32
Figura 2.27: Diagrama de flujo del sistema de iluminación	33
Figura 2.28: Diagrama de bloques del sistema de mezcla de nutrientes	34
Figura 2.29: Sensor de pH	34

Figura 2.30: Conexión del sensor de pH con la Raspberry pi	35
Figura 2.31: Sensor de ultrasonido HC-SR04	36
Figura 2.32: Funcionamiento el sensor HC-SR04.....	36
Figura 2.33: Conexión del sensor HC-SR04 con la Raspberry pi	38
Figura 2.34: Sensor de agua t11592	38
Figura 2.35: Conexión de los sensores de nivel de las soluciones con la Raspberry pi	39
Figura 2.36: Bomba sumergible de 5Vdc.....	40
Figura 2.37: Conexión de las bombas de las soluciones con la Raspberry pi	41
Figura 2.38: Diagrama de flujo del sistema de regulación de pH	42
Figura 2.39: Diagrama de flujo del sistema de mezcla de nutrientes	43
Figura 2.40: Diagrama de flujo del subsistema de mezcla	44
Figura 2.41: Diagrama de bloques del sistema de vigilancia	45
Figura 2.42: Cámara OV5647.....	45
Figura 2.43: Sistema aeropónico	46
Figura 2.44: Datos capturados por la Raspberry pi	47
Figura 2.45: Datos almacenados en Firestore	49
Figura 2.46: Pantalla de autenticación.....	51
Figura 2.47: Pantalla principal para el monitoreo de los sensores	51
Figura 2.48: Pantalla principal con los controles de los actuadores	52
Figura 2.49: Pantalla principal con la representación de los datos del día.....	52
Figura 2.50: Pantalla principal con el menú desplegado	53
Figura 2.51: Pantalla del monitoreo del sistema vía streaming	53
Figura 3.1: Histórico de la bomba de agua	54
Figura 3.2: Gráfica del sensor de temperatura.....	54
Figura 3.3: Gráficas de los sensores de humedad.....	55
Figura 3.4: Gráficas de las servoválvulas	56
Figura 3.5: Histórico de encendido de los focos.....	57
Figura 3.6: Gráfica del sensor de luz	57
Figura 3.7: Gráfica del sensor de nivel del tanque	58
Figura 3.8: Histórico de la bomba de la solución A	58
Figura 3.9: Histórico de la bomba de la solución B	59
Figura 3.10: Histórico de la bomba de la solución C	59
Figura 3.11: Histórico de la bomba del regulador de pH	60
Figura 3.12: Gráfica del sensor de pH.....	60

ÍNDICE DE TABLAS

Tabla 2.1: Especificaciones del sensor de humedad YL-69.....	20
Tabla 2.2: Evaluación de la ecuación Ecu.(2.1)	21
Tabla 2.3 Especificaciones del sensor DHT11	22
Tabla 2.4: Evaluación de la ecuación Ecu.(2.2)	25
Tabla 2.5: Evaluación de la Ecu.(2.5).....	37
Tabla 2.6: Caudal de las bombas para las soluciones	40
Tabla 2.7: Gastos en componentes electrónicos	48
Tabla 2.8: Presupuesto del sistema aeropónico	49
Tabla 2.9 Planes de pago de Firebase	50

RESUMEN

En el presente trabajo de titulación se desarrolla un modelo experimental de cultivo aeropónico vertical, que incluye un sistema de monitoreo remoto, para la empresa ASEMI. El modelo permite visualizar y medir datos de los sensores de humedad, temperatura, pH, nivel e iluminación.

Se diseñó una tarjeta PCB de adquisición de datos utilizando una Raspberry pi y una aplicación con interfaz de monitoreo remoto desarrollada en Android Studio, que se comunican entre sí enviando y recibiendo información a través de la base de datos en tiempo real de Firebase. La aplicación para smartphones permite visualizar las mediciones de los sensores para que el usuario pueda monitorear el funcionamiento del cultivo aeropónico vertical, también dispone de botones de encendido y apagado para el control manual y remoto de los actuadores del sistema, y acceso a una cámara que transmite señal de video en vivo del prototipo implementado. El proyecto controla el riego de forma temporizada, provee de iluminación artificial a las plantas en horarios establecidos, mide y regula el pH en el agua óptimo para el crecimiento de plantas, el caudal en cada tubo PVC del sembrío en base a la humedad, la colocación y mezcla de nutrientes necesarios en el tanque de agua.

Palabras clave: Raspberry Pi, aeroponía, Firebase, Android Studio, monitoreo remoto

ABSTRACT

In this degree work, an experimental model of vertical aeroponic cultivation, including a remote monitoring system, is developed for the company ASEMI. The model allows visualizing and measuring data from humidity, temperature, pH, level and illumination sensors.

A data acquisition PCB board was designed using a Raspberry pi and a remote monitoring interface application developed in Android Studio, which communicate with each other by sending and receiving information through the Firebase real-time database. The smartphone application allows visualizing the sensor measurements so that the user can monitor the operation of the vertical aeroponic crop, it also has on/off buttons for manual and remote control of the system actuators, and access to a camera that transmits live video signal of the implemented prototype. The project controls irrigation on a timed basis, provides artificial lighting to the plants at established times, measures and regulates the pH in the water optimal for plant growth, the flow rate in each PVC pipe of the plantation based on humidity, the placement and mixing of nutrients needed in the water tank.

Keywords: Raspberry Pi, aeroponics, Firebase, Android Studio, remote monitoring.

INTRODUCCIÓN

En el capítulo uno se presenta la investigación realizada sobre la aeroponía vertical, detallando su definición y orígenes, haciendo énfasis en el funcionamiento de este sistema, así como sus partes constitutivas. Además, se exponen los trabajos relacionados que han sido desarrollados en la automatización y el control de forma remota tanto en cultivos hidropónicos como aeropónicos. Toda esta información constituye el fundamento del sistema desarrollado.

Para el capítulo dos, se expone la construcción del modelo aeropónico además de toda la electrónica utilizada para la automatización de su operación, se detallan los elementos utilizados y justifica su elección. También se muestra la lógica de operación del sistema mediante diagramas de flujo.

En el capítulo tres, se detalla los resultados del experimento, el diseño y resultado final de la PCB implementada, la interfaz final de la aplicación, los datos obtenidos de nuestros sensores y actuadores en diferentes tiempos, el correcto funcionamiento del sistema indicado con graficas de los datos obtenidos de los sensores y actuadores durante todo el día.

Para finalizar se detalló las conclusiones y recomendaciones pertinentes al trabajo, así como su respectiva bibliografía.

PLANTEAMIENTO DEL PROBLEMA

Uno de los problemas más importantes que el ser humano debe solucionar en las próximas décadas es el abastecimiento de alimentos debido a la sobrepoblación, ya que se estima que en el 2050 la población mundial será de 9 700 millones de personas (United Nations, 2015), para esto se debe incrementar la producción agrícola en un 70% (Hunter, Smith, Schipanski, Atwood & Mortensen, 2017). Además, las Naciones Unidas prevé que en el año 2030 alrededor del 60% de la población mundial vivirá en la zona urbana (United Nations, 2016), esto conlleva el crecimiento de las ciudades y a su vez la reducción de tierras fértiles para sembrar.

Las recientes innovaciones tecnológicas en cuanto a la agricultura son la hidroponía y la aeroponía que ofrecen múltiples ventajas respecto a cultivos tradicionales como producción en espacios pequeños y sin suelo (tierra), reducción en el consumo de agua de hasta un 85%, cosechas más saludables al no utilizar pesticidas, crecimiento más rápido de las plantas de hasta un 200%, entre otras (Hernández, 2016). Sin embargo, estos métodos de cultivo requieren un control continuo y constante de las condiciones ambientales como temperatura, humedad e iluminación, así como también de la mezcla nutritiva que se suministra a las plantas (Castañares, 2020) (Scaturro, 2019) que generalmente suele ser un proceso manual que requiere la intervención del agricultor (Valles, 2020).

El virus del COVID 19 provocó una pandemia mundial que paralizó muchas de las actividades cotidianas de las personas. Sin embargo, las visitas a tiendas de barrio, mercados y supermercados se mantuvieron por la necesidad primordial de abastecerse de alimentos, de manera que frecuentar estos lugares representa un posible riesgo de contagio (Castro & Zambrano, 2020). Por esta razón, algunas familias han optado por construir huertos caseros dentro de sus domicilios, de hecho, según una encuesta realizada en el 2020 durante la pandemia, el 34% de personas desearían tener un huerto en casa (Navarro, Hernández, Posada, Nepote & Ramirez, 2020), pero no cuentan con el espacio suficiente ni el tiempo necesario para mantener estos cultivos. Otro factor importante que ha motivado a las familias a cultivar dentro de sus hogares ha sido el consumo de productos orgánicos debido a sus propiedades nutritivas que mejorarían el sistema inmunológico (Barrera, 2020).

PROPUESTA DE SOLUCIÓN.

El sistema de automatización permitirá el control de humedad de las raíces de las plantas, temperatura del ambiente, nivel del tanque de la mezcla nutritiva, mezcla de nutrientes, pH de la mezcla nutritiva y luminosidad para el funcionamiento autónomo del cultivo aeropónico, para esto, se dimensionarán los sensores y actuadores que permitan un mejor resultado. El sistema será controlado de forma local mediante una tarjeta embebida como por ejemplo la Raspberry Pi, en la cual se programarán todas las funciones y algoritmos de control utilizando el software interprete de Python. Se va a desarrollar el control remoto del sistema mediante una aplicación móvil que se desarrollará en la herramienta Android Studio.

OBJETIVOS GENERALES Y ESPECÍFICOS

Objetivo General

Automatizar un sistema de cultivo aeropónico experimental con monitoreo remoto mediante sistemas embebidos para la empresa Asemi.

Objetivos específicos

- Analizar el estado del arte de sistemas aeropónicos mediante la investigación científica para la identificación de las variables que requieran ser controladas.
- Desarrollar un sistema de automatización utilizando sensores, actuadores y un sistema embebido con la tarjeta Raspberry Pi para el control de las principales variables de un sistema aeropónico.
- Desarrollar una herramienta de monitoreo remoto con el software Android Studio de desarrollo de aplicaciones para la supervisión del funcionamiento del sistema.
- Construir un prototipo de cultivo aeropónico en base a tubería PVC para la implementación del sistema de monitoreo y automatización.
- Realizar pruebas de funcionamiento de todo el sistema para la obtención de datos por medio de la aplicación para monitoreo remoto que permitan el análisis de su operación.

CAPÍTULO 1

CULTIVOS AEROPÓNICOS

El término Aeroponía proviene de diferentes técnicas de cultivo derivadas de la hidroponía, el cual da como resultado un cultivo vertical en el cual sus raíces se encuentran al aire, dando sentido a su etimología, pues proviene de los términos griegos Aero y Ponos que significa aire y trabajo.

La aeroponía es un método en el cual las plantas crecen sin necesidad del uso de suelo ni de ningún sustrato, las raíces de las plantas se encuentran suspendidas en el aire y colocadas en un lugar oscuro para que su desarrollo se logre de forma óptima, sin plagas ni enfermedades, el crecimiento de la planta se obtiene mediante la aspersion controlada de nutrientes. Este sistema es muy limpio y efectivo, se consigue hasta 60% de ahorro en agua y 90% de ahorro en fertilizantes. Con este método se aprovecha el recurso hídrico, se optimiza el tiempo de producción y crecimiento de la planta y se tiene un mejor manejo de plagas y enfermedades, disminuyendo el uso de agroquímicos dañinos para el suelo y para la salud del ser humano. (Rueda & Pineda, 2012)

La producción aeropónica es sensible al clima por lo que se deben controlar las condiciones de temperatura y humedad del sitio de cultivo, si esto se controla de forma adecuada las cosechas podrán ser secuenciales. Las raíces absorben los nutrientes que provienen de un contenedor opaco, que impida la entrada de luz para evitar el crecimiento de algas en su interior. Además, debe tener un adecuado control de PH óptimo para el crecimiento de la planta que se requiera sembrar

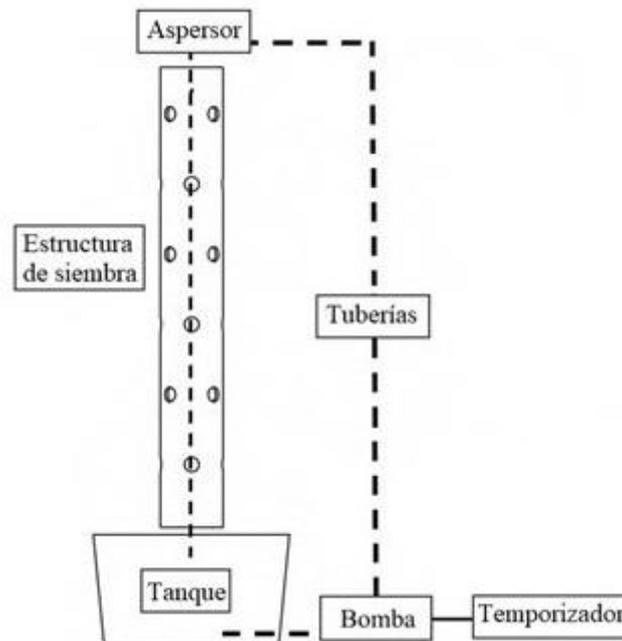
Las primeras investigaciones referentes a la aeroponía se describieron y plantearon por el científico W.A Carter (1942) en su libro: “A method of growing plants in water vapor to facilitate examination of roots”. En el cual expone una teoría referente al crecimiento de plantas en un entorno aéreo y define un método para facilitar el estudio de las raíces de las plantas en vapor de agua.

Según (Salazar, 2021) el primer sistema aeropónico fue desarrollado por el Dr. Franco Massantini, en la Universidad de Pía Italia, por el sistema denominado como columnas de cultivo. Las plantas crecen en la oscuridad y pasan la mayor parte del tiempo expuestas al aire. Al interior del tubo se distribuye la solución nutritiva mediante diferentes técnicas como la pulverización o nebulización.

1.1 Funcionamiento de un cultivo aeropónico

En la Figura 1.1 se muestra un diagrama con los elementos que se consideran básicos para explicar el funcionamiento de un cultivo aeropónico:

Figura 1.1: Diagrama de un cultivo aeropónico



(Aquino, 2020)

Los elementos y componentes que se consideran fundamentales en un cultivo aeropónico son:

- Estructura de siembra
- Tanque
- Bomba de agua
- Sistema de tuberías
- Aspersor
- Temporizador

Se debe tener en cuenta algunas características que estos componentes deberían cumplir para que el sistema funcione adecuadamente, por lo que se pasa a explicar a continuación:

1.1.1 Estructura de siembra

Es el lugar donde se colocan las plantas, protegen las raíces de la luz, y las riegan con la solución nutritiva que circula por el interior. Generalmente se utilizan tubos PVC para fabricar esta estructura de manera casera, o para resultados más prolijos se los puede comprar en plástico a fabricantes de insumos hidropónicos. Estos tubos tienen varios huecos donde se colocan canastillas de plástico que son las que sostienen el peso de las plantas.

Los tubos suelen ser de color blanco para que reflejen la luz, en lugar de absorberla en forma de calor, lo cual no es conveniente. También se debería asegurar que no entre luz al interior del tubo, porque las raíces deben estar en completa oscuridad para un óptimo crecimiento (Scaturro, 2019).

1.1.2 Tanque

Es un componente muy importante porque permite que el sistema sea recirculante. Almacena y recoge la solución nutritiva después de pasar por el sistema de cañerías y por las raíces de las plantas, no se puede utilizar cualquier recipiente, sino que debe ser de plástico y nunca de metal, ya que el metal puede llegar a oxidarse, dañando así la mezcla de nutrientes; también se recomienda que este se encuentre completamente cerrado para que la luz no pueda pasar, evitando el crecimiento de algas u otros microorganismos que podrían llegar a los cultivos a través del sistema de recirculación de nutrientes. (Hernández, 2016) recomienda que el tanque se coloque por debajo de la superficie del suelo, ya que esto ayudará a que el agua nutritiva se mantenga a temperatura ambiente; si el agua estuviera caliente las plantas se podrían marchitar (A. Pérez & Romero, 2019). Otro beneficio de colocar el tanque enterrado en el suelo es que el agua nutritiva tiene más caída libre para generar turbulencias y así facilitar la oxigenación del agua, se aconseja que esta caída libre sea de al menos 0.50 m.

1.1.3 Bomba de agua

Forma el sistema de recirculación, ya que es la encargada de movilizar la mezcla nutritiva desde el tanque hacia el aspersor, para posteriormente regar las raíces de las plantas en forma de lluvia. Es importante que la bomba tenga la potencia suficiente para generar presión en el aspersor, de lo contrario el riego no va a ser adecuado. Sobre todo, cuando se

trata de un cultivo aeropónico vertical, donde la bomba de agua debe ser capaz de empujar el agua varios metros hacia arriba hasta alcanzar los aspersores. Por tal razón se deben revisar muy bien los datos técnicos de la bomba, y adquirir la adecuada para cada proyecto. Otra característica que es muy útil considerar, es que sea una bomba sumergible, de esa manera se puede colocar dentro del tanque haciendo así que el sistema sea más silencioso, ya que el agua absorbe la gran mayoría del ruido que se pueda generar en el funcionamiento de esta.

1.1.4 Sistema de tuberías

Es utilizado para dirigir el agua desde el tanque hacia los aspersores pasando por la bomba, también se encarga de la recolección de vuelta hacia el tanque. Se usan materiales como tubos, codos, tubo en forma de T, acoples. El tipo de tubería generalmente es de ½ pulgada, que es la misma que se usa para el suministro de agua potable en viviendas al ser de fácil obtención, gran disponibilidad y de precios variados. Es muy importante que este sistema de tuberías se encuentre completamente sellado, evitando fugas por donde se pueda escapar parte de la mezcla de nutrientes, para ello, es recomendable utilizar suficiente teflón en cada una de las juntas. Se pueden usar otros materiales de diferentes grosores para el sistema de tuberías, pero se debe tomar en cuenta que un tubo de menor ancho puede generar demasiada presión en las paredes de esta, elevando así la temperatura del agua, o en el peor de los casos romperse, en cambio, tubería de mayor grosor podría no generar suficiente presión en los aspersores.

1.1.5 Aspersor

Es la pieza colocada al final del sistema de tuberías y en la parte superior de la estructura de siembra, se encarga de esparcir la solución nutritiva en forma de gotas para que las raíces puedan absorberlas fácilmente.

En cultivos aeropónicos horizontales se utilizan nebulizadores que rocían el agua en gotas muy pequeñas, en el orden de micras, lo cual es muy beneficioso para las raíces de las plantas a favor de que pueden absorber más agua y en menos tiempo; en cultivos aeropónicos verticales no es tan conveniente usar nebulizadores porque las raíces de las plantas están distribuidas a lo largo de un tubo relativamente delgado, que hace que la nebulización no llegue a todas las raíces, y que más bien, la mayoría de las microgotas

choquen con las paredes del PVC convirtiéndose en gotas más grandes haciendo innecesario el uso de nebulizadores. En cultivos aeropónicos verticales caseros se hace uso de aspersores que bien puede ser un tapón de tubo con algunos huecos pequeños, que en realidad suele ser suficiente para regar las raíces de las plantas.

1.1.6 Temporizador

En un sistema aeropónico, al igual que en cualquier sistema hidropónico, la variable principal que se controla es el riego de los nutrientes hacia las plantas, para esto se utiliza temporizadores mecánicos o electrónicos que se conectan directamente a la bomba haciendo que ésta se active en intervalos de tiempo. Estos tiempos de encendido y apagado varían mucho dependiendo de si la planta tiende o no a absorber mayor cantidad de agua; o factores ambientales como la temperatura, que hacen que la solución se evapore más rápido por lo que se requieren intervalos más largos de riego. Así mismo estos intervalos no van a ser iguales en el día que en la noche, generalmente en el día se riega de 1 a 2 veces por hora mientras que en la noche se debería regar 1 vez cada 2 horas.

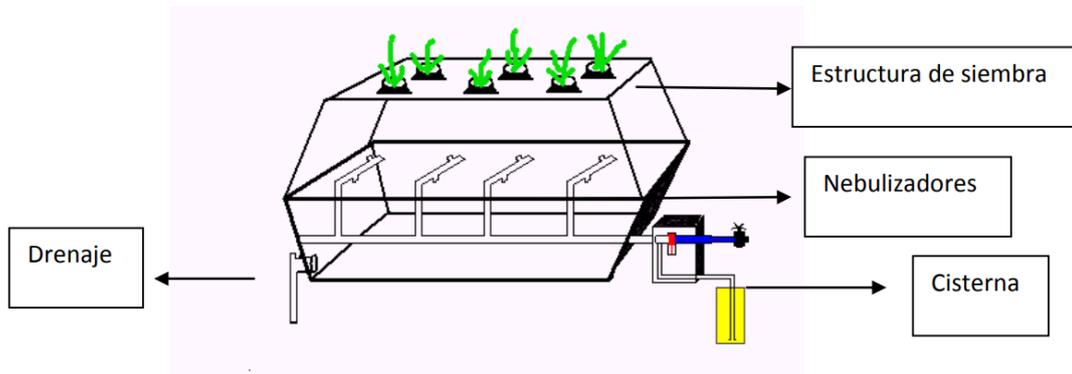
1.2 Estructuras de cultivos aeropónicos

Un cultivo aeropónico, dependiendo de la estructura en que se siembran las plantas, se clasifican de dos maneras: vertical y horizontal.

1.2.1 Estructura horizontal

En una estructura horizontal, como se muestra en la Figura 1.2, las plantas se colocan en una superficie horizontal, lo que ofrece algunas ventajas como: una fácil aspersión de nutrientes, una mejor disposición y acceso a las plantas, y una mejor captación de luz, al estar la estructura perpendicular hacia el sol. (Hernández, 2016)

Figura 1.2: Estructura horizontal para cultivos aeropónicos



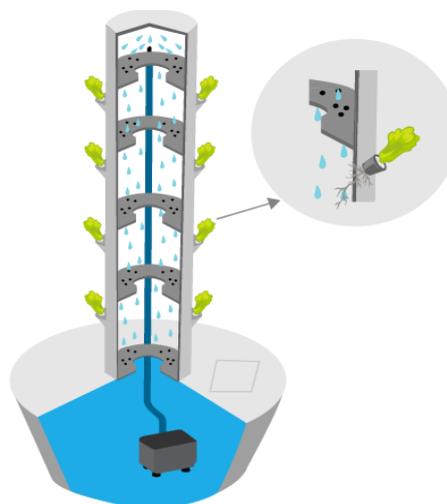
(Hernández, 2016)

1.2.2 Estructura vertical

Una estructura vertical como se observa en la Figura 1.3 tiene una ventaja clave, como es un mejor aprovechamiento del espacio, pudiéndose sembrar una mayor cantidad de plantas por metro cuadrado comparado con una estructura horizontal. Otra de las ventajas es que la solución nutritiva tiende a atrapar mayor cantidad de oxígeno al caer desde una mayor altitud, esto resulta muy beneficioso para las plantas. (Hernández, 2016)

Tanto una estructura para un cultivo aeropónico horizontal como para uno vertical tienen ventajas que deben tomarse en cuenta de acuerdo con las necesidades de cada proyecto, aquí dependerá de los factores que consideren más importantes.

Figura 1.3: Estructura vertical de cultivos aeropónicos



(“Hidroponía vertical”, 2019)

1.3 Automatización de cultivos aeropónicos

En (Bedoya, 2019) se comparó el crecimiento del cilantro utilizando dos tipos de cultivos, el aeropónico y el tradicional los cuales durante dos cosechas se realiza un monitoreo mediante una interfaz gráfica proveniente de una Raspberry, la cual va a permitir controlar, observar y manipular los sensores de pH, temperatura, humedad y conductividad eléctrica siendo enviada la información hacia la Raspberry. Este proyecto consta de un tanque de agua que llega hasta los 250 litros de capacidad, en el mismo se incorporan las soluciones nutritivas, las cuales se las pone según el requerimiento nutricional del cilantro. Se utilizó una bomba de agua que posee 1 HP, la cual es encargada de suministrar agua con nutrientes en el área en la cual se va a sembrar. Se tiene un medidor de presión, una tapa de potencia para controlar los tiempos de riego de la bomba y una tapa de control y automatización, que se va a encargar que el prototipo funcione de manera correcta, un sistema de riego por microaspersión va a permitir que el agua se esparza uniformemente hacia las raíces del cilantro junto con sus nutrientes. Se hizo dos cosechas, en la primera cosecha realizada en el sistema aeropónico se dio una germinación efectiva en 90 de 100 copas, su tiempo de germinación en el suelo fueron de 10 días, dando un total de 435 semillas germinadas.

Según (Aquino, 2020) después de la búsqueda de diferentes fuentes de información se empezó a realizar el primer diseño para crear un modelo digital de un aeropónico el cual va a estar diseñado en SolidWorks, el siguiente paso fue crear los planos del modelo para luego enviarlo a un taller llamado CIBNOR donde se fabricaría la estructura para cada sistema. Para el sistema de irrigación se lo ensambló con una batería de 12v, la bomba utilizada es automática y se utilizó un reloj temporizador comercial, más adelante se adaptó una fuente de poder de computadora para empezar con el desarrollo de los sistemas de irrigación y la medición de parámetros. Para la construcción del sistema aeropónico se utilizó material opaco ya que no permite el paso de luz y ayuda a que no crezcan algas en las paredes de las tuberías. Para el diseño se realizaron diagramas electrónicos para posteriormente empezar con la programación. Las variables a cubrir en la automatización del cultivo aeropónico son: sensor de temperatura, sensor de humedad y sensor de luminosidad. Para la calibración del sensor de pH el cual se desea que tenga un rango de pH 4.01 y 6 se procedió a comparar sus valores con un pH Meter de marca Jenway. La calibración del sensor de conductividad eléctrica se realizó con una solución a 1413 us/cm a 25° C al cual se le comparó con un quipo Oakton modelo ECtester11. Para la irrigación

se utilizó el Arduino Mega que tiene una pantalla y un teclado para que el usuario pueda ingresar y configurar las irrigaciones deseadas de forma inmediata.

(López, 2020) se propuso desarrollar un invernadero con sus mesas de cultivo para aprovechar mejor los recursos naturales y lograr reducir los residuos plásticos, obtención de productos de mayor valor, así como también lograr una infraestructura resistente a los cambios climáticos. Se determinó que la carga eléctrica que va a consumir la maquinaria, así como también las luminarias son de 23084 W que se obtendrán de la red eléctrica pública. Para la automatización se usó sensores de humedad, los cuales son empleados dentro de las mesas de cultivos y reportarán datos de la humedad relativa y temperatura en el interior de las bandejas, se las situó en el invernadero para controlar la humedad del aire. Para el accionamiento del riego se dispone de electroválvulas pilotadas, se accionarán o desconectarán según los datos obtenidos del sensor de humedad. El funcionar de las bombas de agua estará regulado por el nivel de presión, ya que según el requerimiento del caudal se activarán las bombas correspondientes. Las ventanas del invernadero funcionan de forma automatizada, se abrirán y cerrarán mediante un servomotor que funciona a 24V. Todo el sistema de control del invernadero funcionará mediante el equipo de tele gestión SKYREG y un PC, el cual recibirá datos.

1.4 Sistemas de monitoreo remoto aplicados a la agricultura.

(Apaza, 2020) se ha propuesto diseñar e implementar, utilizando red de sensores los cuales estarán basados en el Internet de las Cosas que se aplicarán a la siembra de tomate Cherry. Para ello se utilizó el NodeMCU el cual tiene asociado el SOC WIFI ESP8266 que les permitirá tener conexión Wifi con el proyecto. Con el dispositivo interactuando con internet IoT, se procedió a analizar los factores ambientales de temperatura, humedad, cantidad de luz y el pH del sustrato, estos datos se enviarán hacia la nube. Para empezar con el ensamble del sistema hidropónico hay variables que considerarse como la temperatura, la cual debe estar entre los 7° C y llegar a un máximo de 25 °C, estas son condiciones que serán monitoreadas, otra variable es el pH de la solución nutritiva y del agua el cual debe ir de 6 a 7, para esto se utiliza un tanque el cual tendrá agua y nutrientes en proporciones requeridas para la siembra de tomate Cherry. Para la humedad, el porcentaje mínimo aceptable debe ser de 25%, si la humedad desciende demasiado el tomate Cherry no crecerá de manera óptima. Para el sistema de riego automático se utilizó

un sensor de detección de humedad ubicado en cada sembrío para activar una bomba junto a una electroválvula para el lugar que necesite ser regado. Para los resultados se realizó un análisis de 3 meses, para lo cual se comparó la producción antes y después de la automatización, que después de un análisis de datos y utilizando gráficas se puede observar que existe un aumento en la producción de tomate Cherry en un promedio de 8.15 kg. Posteriormente se realizó un análisis de recuperación de la inversión donde se deduce que se recuperaría la inversión después de 16 meses y posteriormente empezar a obtener ganancias que se estima serán grandes para la empresa.

En el diseño de (Viera, 2019), se desarrolló un sistema de monitoreo remoto que medía ciertas variables que se consideraron indispensables para un cultivo hidropónico para ellos se utiliza tecnología inalámbrica Zigbee. Para el sistema de monitoreo se elaboró una tarjeta que adquiere datos y estará conectado a los sensores de temperatura y humedad. La interfaz de monitoreo se la realizó en Visual Basic donde se puede visualizar los datos recibidos de los sensores. En la tubería PVC se realizaron agujeros donde se colocaron vasos de Tecnopor en el mismo se coloca sustrato y una semilla de aguaymanto. El Arduino Mega será el que envíe, procese y reciba datos obtenidos de los sensores, el módulo Xbee va a estar conectado con los pines Tx y Rx del Arduino que servirá para la comunicación inalámbrica. Se utilizó el sensor DHT11, encargado de medir la temperatura y humedad dentro del sistema hidropónico, para posteriormente ser enviado y procesado en el Arduino Mega y luego enviado hacia Xbee.

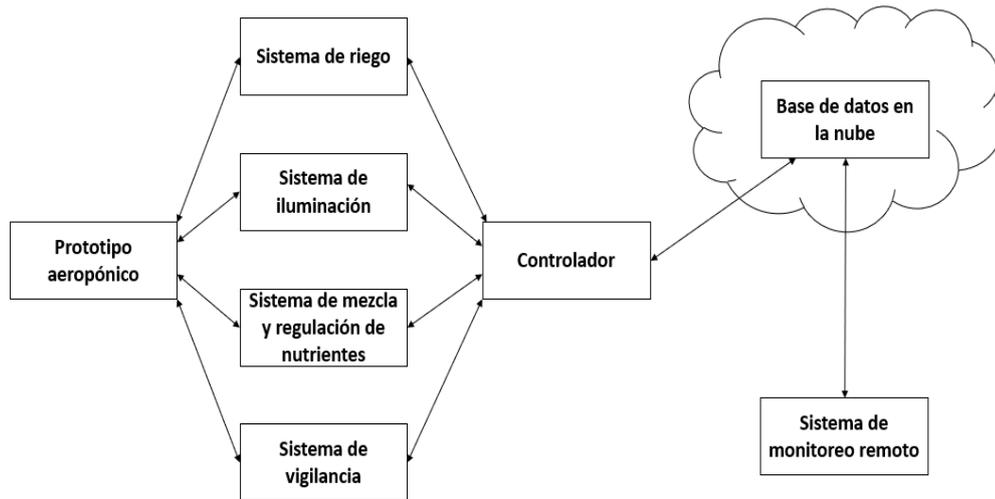
Se comienza con el desarrollo del diseño y arquitectura de operación del sistema, según (A. Pérez & Romero, 2019), se deben especificar los parámetros de control en el hardware, para ello se ocupan los sensores de temperatura LM35 y un sensor de pH con sonda, para los actuadores se utiliza un ventilador con extractor, calentador de aire, bomba de aire y una electrobomba sumergible. Entre los factores importantes está el control de la temperatura en la solución nutritiva, estas proporcionan el alimento que necesita la planta para su desarrollo por lo que es de suma importancia que la solución se encuentre en condiciones adecuadas para que la planta no tenga problema en absorber los nutrientes. Como resultados se puede observar que una vez iniciada la siembra en el hidropónico construido empezaron a trabajar correctamente todos los sensores y el monitoreo mencionado. Se construyó una placa de control la cual se la puso dentro de una carcasa para protegerla de agentes externos que puedan dañar la misma.

CAPÍTULO 2

DISEÑO DEL SISTEMA AEROPÓNICO VERTICAL

En la figura 2.1 se muestra un esquema en diagrama de bloques para el prototipo aeropónico:

Figura 2.1: Diagrama de bloques del sistema aeropónico



Autores: Víctor Yanchapaxi y Bryan Taboada

El esquema consta de varios bloques, los cuales se encargan de controlar variables diferentes:

- **Prototipo aeropónico:** Es la estructura física en la que están montados los demás sistemas, y consta de todos los elementos como tuberías, tanque, soporte y canastillas para colocar las plantas y que estas sean regadas adecuadamente.
- **Controlador:** Es el bloque encargado de adquirir los datos de los sensores y procesarlos para activar los distintos actuadores, también se encarga de establecer la conexión con la base de datos en la nube para el intercambio de información.
- **Sistema de riego:** Es el encargado de determinar los intervalos de riego en el sistema aeropónico, también se realiza un control de caudal de la solución nutritiva.
- **Sistema de iluminación:** Realiza el control del encendido y apagado de los focos que hacen de iluminación artificial para las plantas.
- **Sistema de mezcla y regulación de nutrientes:** Este bloque se encarga de preparar la mezcla de los nutrientes en el agua necesarios para que las plantas se alimenten,

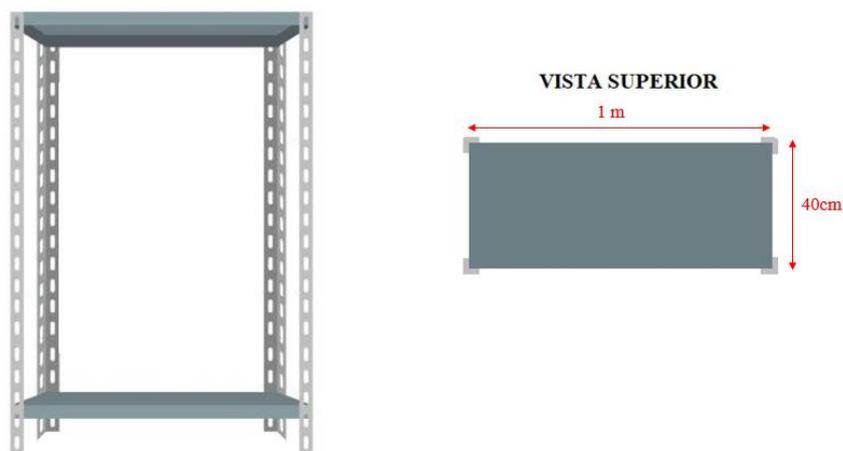
también adecúa el pH de la mezcla para que esta se encuentre en los valores óptimos.

- Sistema de vigilancia: Es un bloque capaz de transmitir datos de imagen y video de la estructura del sistema aeropónico para poder comprobar el estado y funcionamiento del sistema de manera visual.
- Base de datos en la nube: Es la plataforma en la nube que almacena y gestiona los datos entre el controlador y el sistema de monitoreo remoto.
- Sistema de monitoreo remoto: Este bloque lee datos de la nube que corresponden a datos útiles de los sensores y actuadores, y los muestra en una interfaz gráfica para que el usuario revise el estado de todo el sistema de forma sencilla.

2.1 Prototipo aeropónico

Se utiliza una estructura en ángulos de metal que hacen que el prototipo sea robusto y pueda sostener el peso de todos los materiales y elementos que se colocan en él. La estructura de metal se muestra en la figura 2.2.

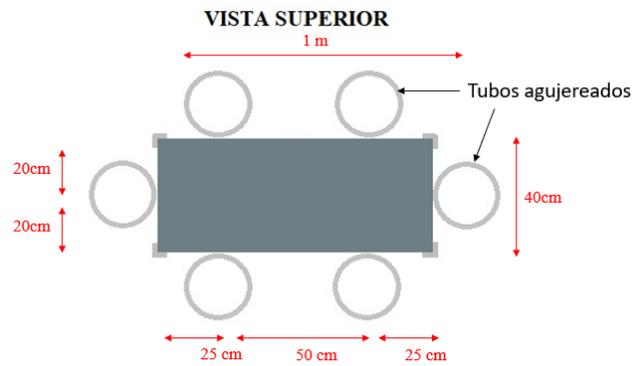
Figura 2.2: Estructura del prototipo aeropónico



Autores: Víctor Yanchapaxi y Bryan Taboada

La estructura tiene una medida de 1 metro de ancho por 40cm de largo, en él se colocan 6 tubos para sostener las plantas. Se utiliza una disposición de 6 tubos, 4 en las partes más anchas y 2 en las más estrechas para que las plantas tengan el suficiente espacio para crecer, en la Figura 2.3 se presenta una vista superior de la disposición de los tubos.

Figura 2.3: Disposición de los tubos de siembra



Autores: Víctor Yanchapaxi y Bryan Taboada

Se usan tubos PVC de 6 pulgadas, de 1.5 m de alto cada uno. Se realizan cortes intercalados de 8 cm, que rodean todo el tubo por los 4 lados, cada 20 cm por los dos lados que son opuestos del tubo, y una separación de 30 cm por los lados restantes. De esa manera se consigue un total de 26 cortes en cada tubo de 1.5 m de largo, 14 cortes por dos de los lados y 12 cortes más por los lados restantes. En cada uno de estos cortes se hacen hendiduras para las canastillas, el resultado debe quedar como la Figura 2.4.

Figura 2.4: Tubos para aeroponía

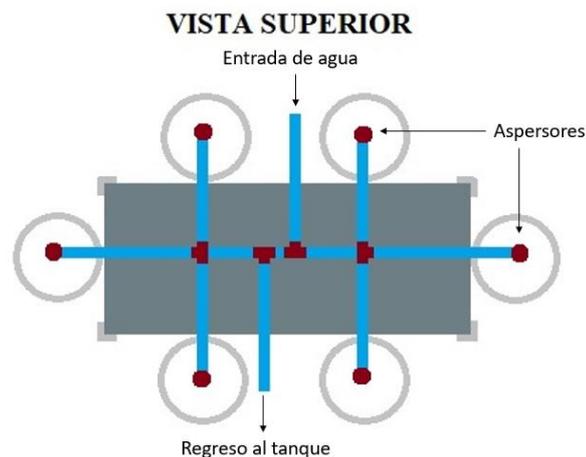


Autores: Víctor Yanchapaxi y Bryan Taboada

El sistema es capaz de sostener a 156 plantas y se puede ubicar en un espacio de 3 m². Esta es justamente una de las grandes ventajas de los sistemas aeropónicos verticales.

Los tubos se sujetan a la estructura de metal mediante rieles de madera, las dimensiones se pueden observar en la Figura 2.3. Las conexiones para el sistema de riego son de tubo de ½ pulgada roscable, esto permite dirigir la mezcla nutritiva hasta arriba de la estructura, desde el tanque, y a partir de ahí distribuirla hacia los 6 tubos PVC agujereados, y para que se disperse en formas de gotas, se conectan aspersores al final de cada tramo. Adicionalmente se conecta un tubo de ½ pulgada para regresar parte del agua con nutrientes hacia el tanque y así controlar el caudal de los líquidos. Una vista superior de las conexiones de los tubos de ½ pulgada para el riego en este prototipo aeropónico se muestran en la Figura 2.5.

Figura 2.5: Conexiones del sistema de tuberías

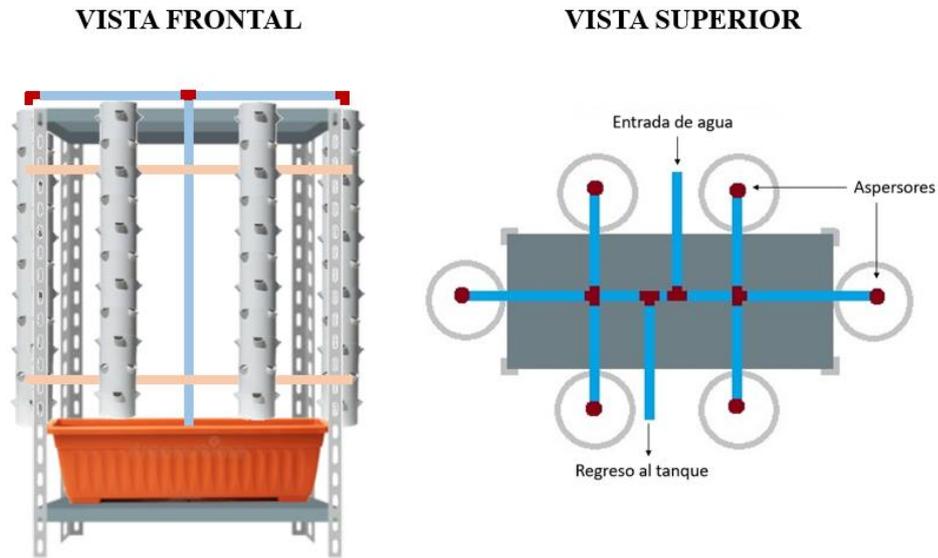


Autores: Víctor Yanchapaxi y Bryan Taboada

Para recoger la mezcla nutritiva, después de que haya pasado por la tubería PVC de siembra, se decidió usar canaletas, que se pueden construir con tubos de 6 pulgadas cortados transversalmente. Finalmente, un tanque recoge y almacena la solución nutritiva hasta que suceda el otro intervalo de riego, las dimensiones de este tanque deben ser similares a las de la base de la estructura, y de plástico para evitar que se oxide debido a la solución nutritiva (Hernández, 2016).

El prototipo propuesto en este trabajo se muestra en la Figura 2.6, que permite la siembra de 156 plantas como: lechuga, albahaca, cilantro, fresas, entre otras, en un espacio de 2 m de ancho por 1.5 m de largo.

Figura 2.6: Prototipo aeropónico



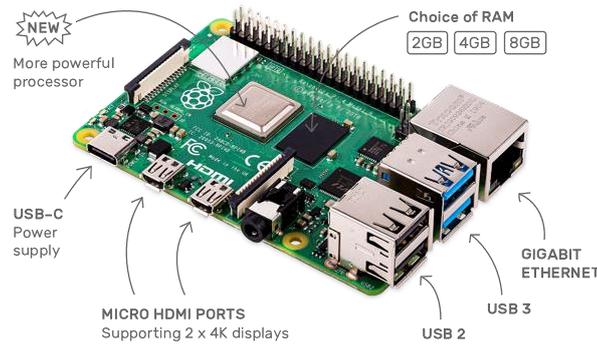
Vista frontal y vista superior del prototipo aeropónico planteado en este proyecto,

Autores: Víctor Yanchapaxi y Bryan Taboada

2.2 Controlador

El controlador que se utilizó para este proyecto es una Raspberry pi modelo 4, la versión que tiene 2 Gb de memoria RAM. Para la elección de este se tomó en cuenta que se van a manejar un gran número de entradas y salidas, también se considera la conexión de una cámara para el monitoreo remoto, adicionalmente se requería que la misma se pueda conectar a Internet, y finalmente altas capacidades de procesamiento para manejar todo lo antes mencionado. De esta manera se optó por utilizar una Raspberry pi, que si bien es cierto es más costosa que otros controladores como Arduino o ESP32, tiene todas las características que se necesitan para este proyecto e incluso para ampliar más funcionalidades en el futuro.

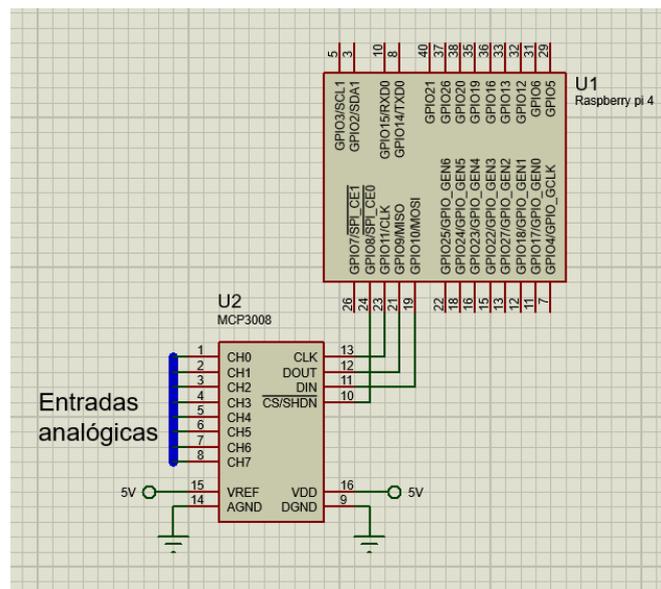
Figura 2.7: Raspberry pi modelo 4



(Raspberry Pi Ltd, s/f)

La Raspberry pi 4, es un sistema embebido que cuenta con procesador, memoria RAM, memoria EEPROM, tarjeta gráfica, chip de audio, y demás periféricos en una misma placa. Cuenta con 40 pines GPIO (General Purpose Input/Output), que pueden ser programados a voluntad, algunos de ellos tienen funciones adicionales como salidas PWM, o protocolos como I2C (Raspberry Pi Ltd, s/f). La desventaja de la Raspberry pi es que no puede leer señales analógicas por lo que se utiliza el chip MCP3008, que es un conversor analógico-digital (ADC) de 8 canales con 10 bits de resolución, a los cuales se conectan 8 sensores analógicos que no requieren de demasiada precisión de lectura. El MCP3008 entrega el valor del sensor en el rango de 0 a 1. El esquema de conexión del MCP3008 con la Raspberry pi se muestra a continuación en la Figura 2.8.

Figura 2.8: Conexión entre Raspberry pi y MCP3008



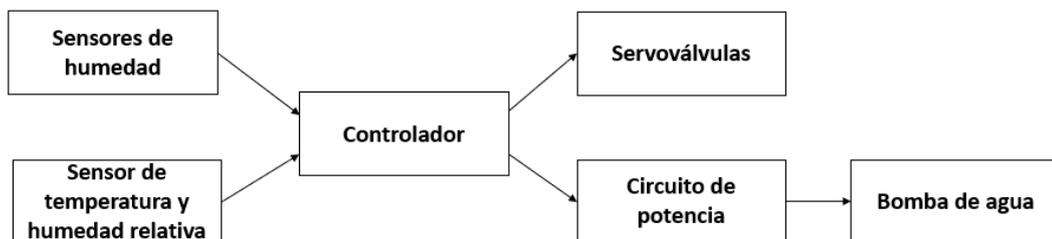
Autores: Víctor Yanchapaxi y Bryan Taboada

El lenguaje de programación que se usa en la Raspberry pi es Python, es uno de los lenguajes más utilizados a nivel mundial y en el, se han hecho muchos desarrollos. Así que, se tienen a disposición una gran cantidad de librerías para la gran mayoría de sensores que hay en el mercado utilizados en este prototipo.

2.3 Sistema de riego

Se constituye de dos lazos de control, en el primero el controlador decide los intervalos de riego en base a los datos del sensor de temperatura y humedad relativa y de un reloj de tiempo real (RTC), con esos datos activa y desactiva la bomba de agua para movilizar la mezcla nutritiva hacia los aspersores. De por medio hay un circuito de potencia debido a que la bomba funciona a 110VAC y la salida del controlador es una señal de 5V de baja potencia. Por otro lado, se tiene un lazo que controla el flujo de la mezcla nutritiva, este control se hace sobre las servoválvulas que se abren y se cierran en diferentes grados permitiendo pasar más o menos solución por las mismas, y a través de los sensores de humedad el controlador calcula cuánto se deben abrir o cerrar cada servoválvula. En la Figura 2.9 se presenta el diagrama de bloques de este sistema.

Figura 2.9: Diagrama de bloques del sistema de riego

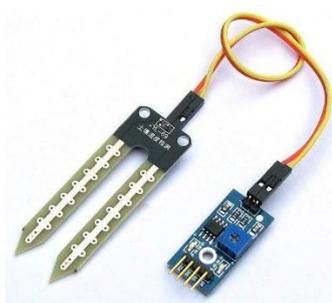


Autores: Víctor Yanchapaxi y Bryan Taboada

2.3.1 Sensores de humedad

Este sistema tiene un total de 6 sensores humedad YL-69, uno para cada tubo del prototipo, los terminales del sensor que miden la humedad tienen colocado en sus puntas una esponja para retener el agua, de esa manera se puede sentir si el riego se está realizando de forma adecuada si los valores de humedad también aumentan o se mantienen.

Figura 2.10: Sensor de humedad YL-69



(ELECFreaks, s/f)

El sensor YL-69 se muestra en la Figura 2.10, cuenta con un circuito acondicionador para la señal que le llega desde las puntas que sensan la humedad. Este acondicionador tiene 4 pines, dos de ellos son para la alimentación del propio circuito que puede ir desde 3.3V hasta 5V, y las dos restantes son una salida digital y una salida analógica. Para la salida digital se debe ajustar el setpoint que acciona dicha salida a 1 lógico a través del potenciómetro de precisión que está en el mismo circuito, pero la señal que más importante es la del pin analógico, que como ya se mencionó antes, se debe conectar a uno de los canales de MCP3008, ya que la Raspberry pi no tiene pines de entrada analógica nativas.

Tabla 2.1: Especificaciones del sensor de humedad YL-69

Ítem	Condición	Min	Max	Unidad
Voltaje	-	3.3	5	V
Corriente	-	0	35	mA
Salida digital	Alimentación: 5V	0	5	V
Salida analógica	En tierra seca	0.8	1	-
	En tierra húmeda	0.5	0.8	-
	En agua	0.5	0.3	-

Valores captados en la Raspberry, Autores: Víctor Yanchapaxi y Bryan Taboada

Cuando entre las puntas del sensor YL-69 no se detecta humedad, la salida analógica de la misma va a tender al valor más alto de referencia, es decir, si se alimenta al circuito acondicionador con 5V, la salida analógica va a ser cercana a esos 5V. En cambio, cuando se detecta mucha humedad entre sus terminales, el valor analógico puede llegar hasta el 50% de valor de referencia, este valor puede bajar aún más si la conductividad del líquido es buena. La señal que es recibida en la Raspberry pi es 1 cuando el sensor no detecta

humedad, y 0.4 aproximadamente cuando se detecta una buena humedad, para representar mejor estos valores se hizo un pequeño ajuste aplicando la Ecu.(2.1).

$$\text{humedad} [\%] = (1 - \text{valor del sensor}) * 100 \quad \text{Ecu.}(2.1)$$

El resultado de esta ecuación da 0 [%] de humedad cuando el sensor no detecta nada de humedad entre sus terminales, y aproximadamente 60 [%] cuando la humedad es buena. Este valor de humedad es proporcional, y se va incrementando paulatinamente conforme la esponja colocada en las puntas del sensor vaya reteniendo más líquido. A continuación, se muestra una tabla evaluando los valores de la Ecu.(2.1).

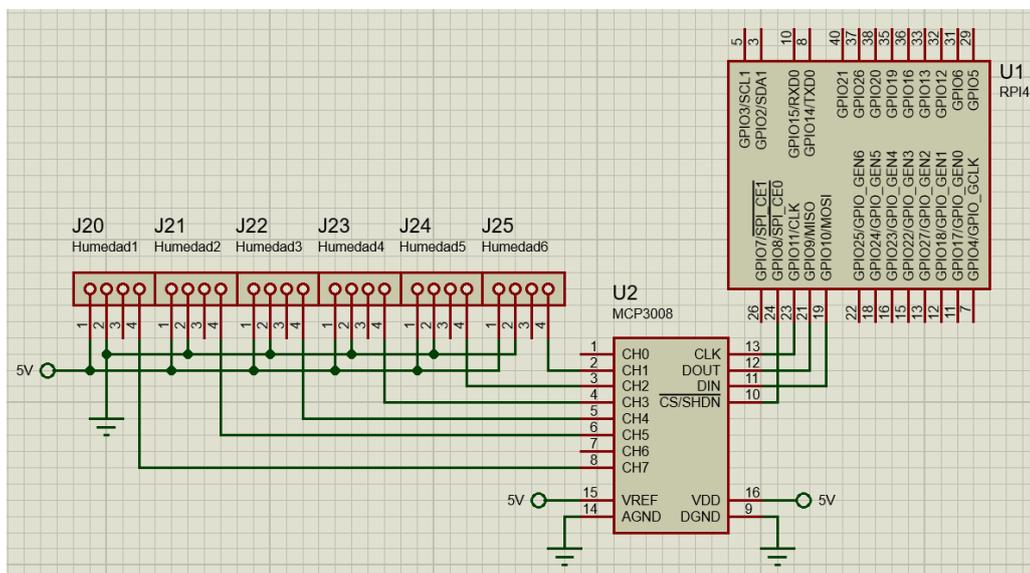
Tabla 2.2: Evaluación de la ecuación Ecu.(2.1)

Sensor [0 a 1]	Estado	Humedad [%]
0	Demasiada humedad	100
0.25	Mucha humedad	75
0.5	Mediana humedad	50
0.75	Poca humedad	25
1	Nada de humedad	0

Transformación del valor de sensor YL-69 a porcentaje de humedad , Autores: Víctor Yanchapaxi y Bryan Taboada

En la Figura 2.11 se muestra el circuito de conexión de los sensores YL-69

Figura 2.11: Conexión de los sensores YL-69 con Raspberry pi

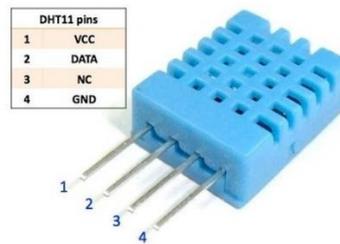


Autores: Víctor Yanchapaxi y Bryan Taboada

2.3.2 Sensor de temperatura y humedad relativa

El sensor que se utiliza comúnmente para esta finalidad es el DHT11, que tiene buena precisión y bajo costo. Mide la temperatura y humedad relativa que existe en el ambiente y, con este parámetro se pueden determinar los intervalos de tiempo en que se enciende la bomba para regar los nutrientes por todo el sistema.

Figura 2.12: Sensor DHT11



(“Cómo usar el sensor DHT11”, s/f)

Este sensor tiene 4 pines, el pin número 1 y pin número 4 visto desde su cara frontal como en la Figura 2.12, son pines de alimentación que pueden ser conectados a 3.3V o a 5V, el pin número 3 no se utiliza, y el pin número 2 transmite en sí la señal del sensor, la misma que requiere de una resistencia pull-up al pin 1, y que (Mouser Electronics, s/f) recomienda sea entre 1Kohm y 10Kohm para distancias menores a 20 metros, y para distancia mayores una resistencia de menor valor.

Tabla 2.3 Especificaciones del sensor DHT11

Ítem	Rango de medición	Precisión	Resolución	Empaquetado
DHT11	20-90%RH	±5%RH	1	4 pines
	0-50°C	±2°C		

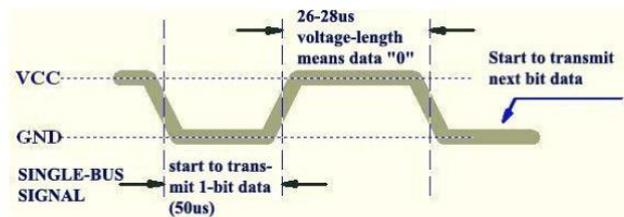
(Mouser Electronics, s/f)

En la Tabla 2.3, se observan las características de medición del sensor DHT11, como por ejemplo los rangos de temperatura que van desde 0 hasta 50°C, respecto a la humedad puede medir desde 20 a 90%Rh.

Para la transmisión de la información, el sensor DHT11 hace uso del protocolo de comunicación denominado “Single-wire Two-way”, en el cual se envía primero una señal de sincronización hacia el controlador esperando una respuesta, cuando se recibe esta señal

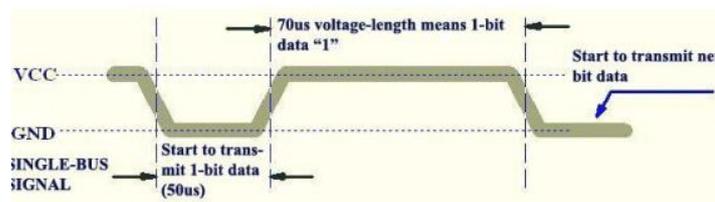
de confirmación el sensor envía los datos de temperatura y humedad relativa codificados en una cadena de 40 bits en un lapso total de 40 ms (Mouser Electronics, s/f). La forma en que se transmiten cada uno de estos bits es primero una señal de 0V que dura 50 us seguido de un pulso de 26 a 28 us si el bit que se transmite es un 0 lógico, Figura 2.13 , o de 70 us si el bit que se transmite es un 1 lógico, Figura 2.14. Esto se repite hasta completar los 40 bits de información que tiene el protocolo.

Figura 2.13: Transmisión de un 0 lógico con el protocolo “Single-wire Two-way”



(Mouser Electronics, s/f)

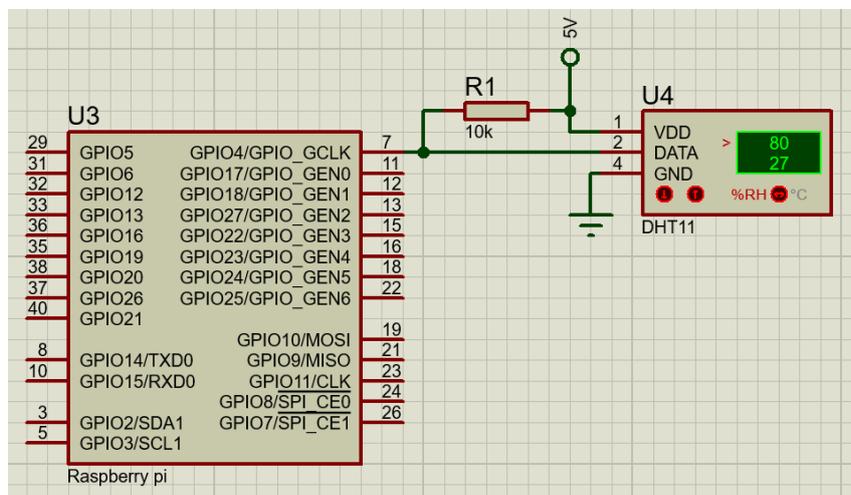
Figura 2.14: Transmisión de un 1 lógico con el protocolo “Single-wire Two-way”



(Mouser Electronics, s/f)

La Figura 2.15 muestra el circuito de conexión del sensor DHT11 con la Raspberry pi.

Figura 2.15: Conexión del DHT11 con la Raspberry pi



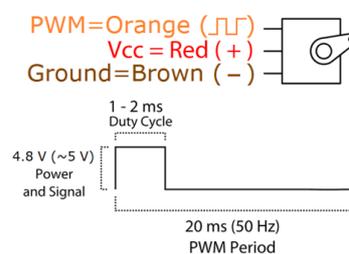
Autores: Víctor Yanchapaxi y Bryan Taboada

2.3.3 Servoválvulas

Las servoválvulas utilizadas en este proyecto están realizadas de forma casera ya que no se encontraron modelos comerciales a la venta en Quito - Ecuador y se fabricaron con llaves de paso y servomotores, cuyo costo es inclusive menor que electroválvulas comerciales que solo permiten o interrumpen el paso de fluidos y que además consumen más energía eléctrica, en cambio las servoválvulas caseras de este prototipo permiten el paso proporcional de los líquidos, así que, se puede controlar el caudal de la mezcla nutritiva hacia los diferentes tubos e incluso impedir completamente el riego en caso de ser necesario. Se fabricaron un total de 7 servoválvulas: 6 servoválvulas regulan el caudal de la mezcla nutritiva en los tubos de siembra, y la servoválvula restante está conectada al tubo de ½ pulgada que regresa la mezcla nutritiva al tanque; por ahí fluirá el resto de líquido que no pasa a través de las 6 servoválvulas de riego cuando estas no están completamente abiertas, esto es para no generar presión excesiva sobre el líquido ya que se sabe que un líquido tiende a aumentar su temperatura si la presión aumenta y si el volumen se mantiene constante. Una mezcla nutritiva caliente no es buena para las raíces de las plantas (A. Pérez & Romero, 2019).

Los servomotores utilizados son los SG90, que tienen un torque de hasta 2.5 kg-cm. Funcionan con 3 pines, dos de ellos para alimentación que van conectados directamente a 5V, y un pin de señal, el cual debe ser una señal PWM, como la Figura 2.16, cuyo ciclo de trabajo puede ir desde 0.5 ms para la posición a 0°, hasta 2.5 ms para llegar a la posición de 180°. El periodo de la señal debe ser de 20 ms.

Figura 2.16: Señal de entrada del servomotor SG90



(“Micro servo miniatura SG90 1.5kg”, 2018)

Para las servoválvulas mencionadas se utiliza el rango de 0° hasta 90° del servomotor, esto es debido a que la llave de paso solo gira en esa amplitud. Así que, el ciclo de trabajo de la

señal de PWM va a variar únicamente entre 0.5 ms y 1.5 ms, pero por facilidad se trabajará estas variaciones en porcentaje, es decir 0% significa completamente cerrado y 100% completamente abierto, con la Ecu.(2.2).

$$\text{Ciclo de trabajo [s]} = -0.00001 * \text{porcentaje} + 0.0015 \quad \text{Ecu.(2.2)}$$

De esta forma, cuando se quiera cerrar completamente la servoválvula, es decir al 0%, la fórmula será igual a 0.0015 s, o 1.5 ms, el servomotor girará hacia los 90° y cerrará la llave de paso completamente. Si se quiere abrir completamente la servoválvula, es decir al 100%, la ecuación será igual a 0.0005 s, ósea 0.5 ms, con el cual el servomotor girará hacia los 0° y la llave de paso se abrirá completamente. También funciona para cualquier otro porcentaje de apertura, por ejemplo, si se requiere abrir la servoválvula al 50%, el ciclo de trabajo será igual a 1ms con lo que el servomotor girará hacia los 45° aproximadamente, y la llave de paso se va a abrir hasta la mitad. Se puede apreciar mejor esta conversión en la Tabla 2.4.

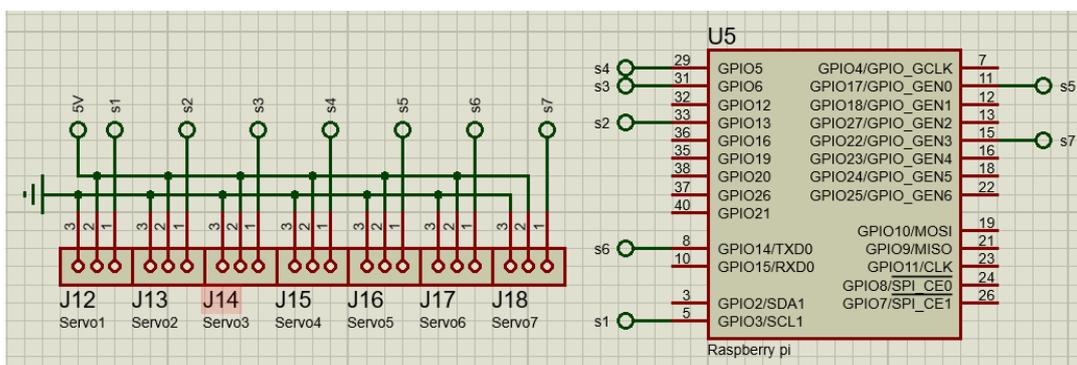
Tabla 2.4: Evaluación de la ecuación Ecu.(2.2)

Porcentaje [%]	Estado de la servoválvula	Ciclo de trabajo [ms]	Grados en el servo [°]
0	Completamente cerrado	1.5	90
25	Poco cerrado	1.25	67.5
50	Medianamente abierto	1	45
75	Poco abierto	0.75	22.5
100	Completamente abierto	0.5	0

Determina el ángulo del servo a partir de porcentaje, Autores: Víctor Yanchapaxi y Bryan Taboada

La conexión de las servoválvulas con la Raspberry pi se muestran en la figura 2.17.

Figura 2.17: Conexión de las servoválvulas con la Raspberry pi

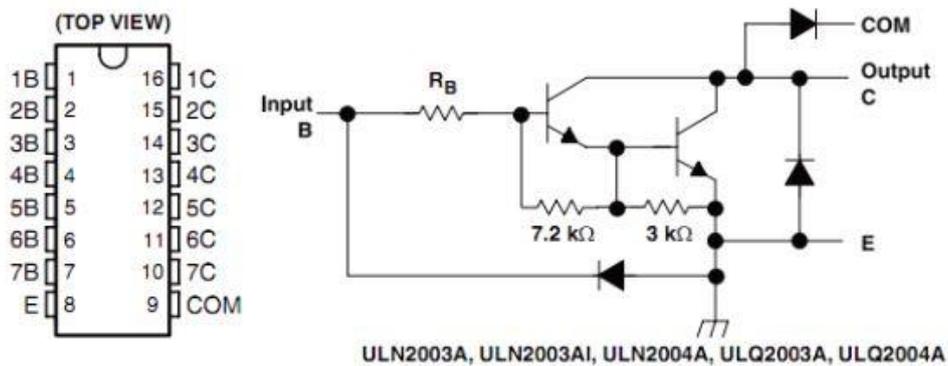


Autores: Víctor Yanchapaxi y Bryan Taboada

2.3.4 Circuito de potencia

Se usa el circuito integrado ULN2003 que es un driver que toma la señal de baja potencia del controlador y la amplifica para elementos eléctricos que requieran de mayor corriente. Internamente el chip contiene un arreglo de 8 transistores Darlington como se ve en la Figura 2.18, que trabajan en modos de corte-saturación, este circuito permite tener 8 entradas con sus respectivas 8 salidas, cada una de estas entradas y salidas reemplazan completamente un circuito de resistencia, transistor y diodo usados comúnmente para alimentar componentes como motores, relés, y demás, que requieren mayor potencia que la salida que un controlador puede otorgar.

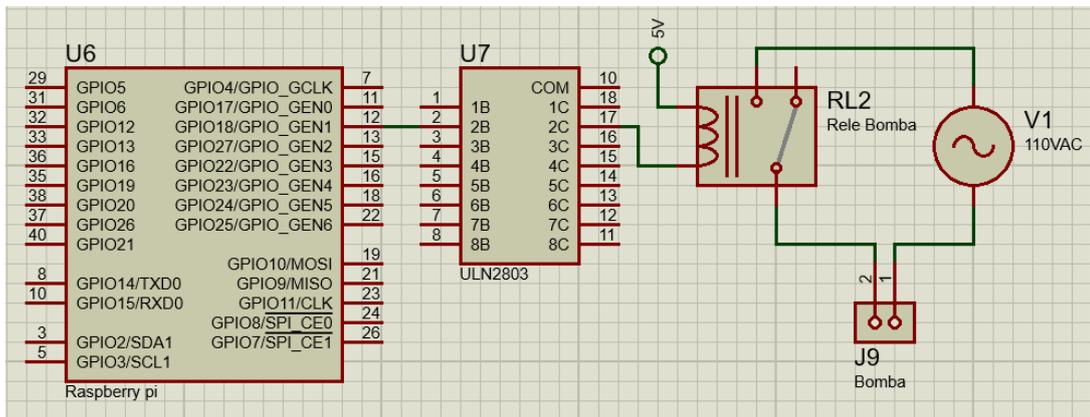
Figura 2.18: Circuito interno del driver ULN2003



(STMicroelectronics, 2021)

La corriente que pueden manejar los pines de salida del ULN2003 son de 500mA a 50V (STMicroelectronics, 2021), que este caso son más que suficientes debido a que solamente se conecta un relé de 5 V que consume una corriente de 40mA aproximadamente. En los contactos normalmente abiertos del relé se conecta la bomba de agua que funciona a 110VAC, y que consume una potencia de 70W, con estos datos ya se aprecia que por este circuito de potencia normalmente circula una corriente de 0.63 A, salvo en el arranque del motor, de todas maneras, el relé de 5V es suficiente al ser capaz de soportar hasta 10 A entre sus contactos. Las conexiones realizadas en el circuito de potencia de la bomba se muestran en la Figura 2.19.

Figura 2.19: Conexión de la bomba al circuito de potencia



Autores: Víctor Yanchapaxi y Bryan Taboada

2.3.5 Bomba de agua

En este prototipo de cultivo aeropónico se utiliza la bomba que se muestra en la Figura 2.20, que funciona con 110VAC a una potencia de 70 W. Esta bomba tiene un caudal de 4500 lph, es decir puede bombear hasta 1.25 litros de agua cada segundo, esta potencia hace que sea capaz de elevar el agua hasta 3.5 m de altura en 2 segundos (“Bomba sumergible SOBO 4500lph”, s/f), la altura a la cual se encuentran los aspersores del experimento aeropónico es de 2 m aproximadamente, pero se requiere que en los aspersores llegue la presión suficiente para que el agua con nutrientes salga con fuerza y sin problemas. Otra característica de la bomba es que es sumergible, por lo que es más silenciosa y ocupa menos espacio al colocarse dentro del tanque. El circuito de potencia enciende y apaga la bomba según los intervalos de tiempo que determine el controlador.

Figura 2.20: Bomba de agua de 100W y 4500lph

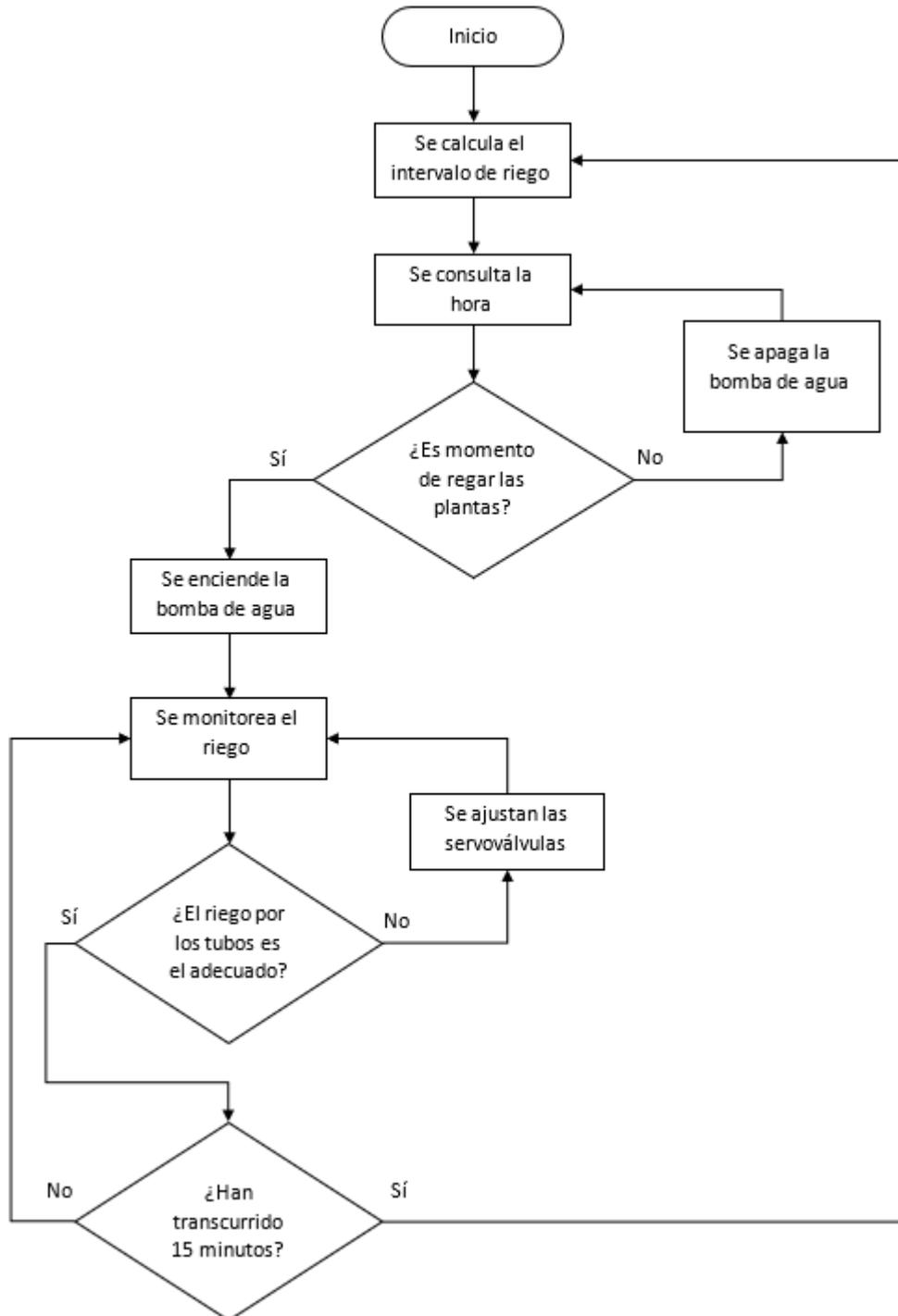


(“Bomba sumergible SOBO 4500lph”, s/f)

2.3.6 Lógica de operación del sistema

El funcionamiento del sistema de riego es controlado por la Raspberry pi, y se puede explicar mediante el diagrama de flujo que se muestra en la Figura 2.21:

Figura 2.21: Diagrama de flujo del sistema de riego

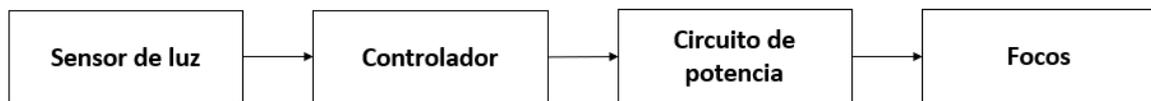


Autores: Víctor Yanchapaxi y Bryan Taboada

2.4 Sistema de iluminación

El control de la iluminación se lo realiza en base al temporizador interno del Raspberry pi al cual se puede acceder mediante programación. Se tienen horarios establecidos para encender la iluminación en la mañana y apagarlo en la noche, el sensor de luz funciona como circuito de realimentación para comprobar que efectivamente han sucedido estos cambios en la iluminación. Un diagrama de bloques para este sistema se presenta en la Figura 2.22.

Figura 2.22: Diagrama de bloques del sistema de iluminación

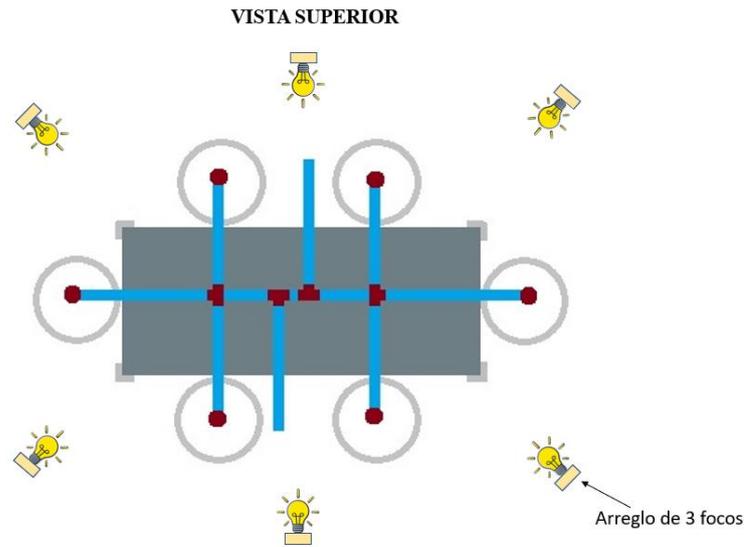


Autores: Víctor Yanchapaxi y Bryan Taboada

Mediante focos se provee de iluminación artificial a las plantas del prototipo aeropónico ante la insuficiencia de luz natural, se utilizan focos tipo LED de 9 W porque proveen luz blanca y no generan calor, lo cual resulta beneficioso para las plantas, pues no tiende a producir quemaduras en las hojas de estas. Además, los focos LED tienen un consumo energético más bajo que los focos fluorescentes e incandescentes, así que resulta en un sistema más eficiente.

Cuando se trata de iluminación artificial en cultivos, lo ideal es que la fuente de luz no se encuentre más allá de 40 cm esto servirá de ayuda para aprovechar la mayor cantidad de luxes que se va a proporcionar, así que se colocaron 6 arreglos de 3 focos en el espacio que queda entre dos tubos de siembra, y un foco más grande dentro de la estructura. De esa manera, se consigue iluminar cada planta que haya sido colocada en el prototipo, como se muestra en la Figura 2.23.

Figura 2.23: Arreglo de focos para el prototipo aeropónico



Autores: Víctor Yanchapaxi y Bryan Taboada

2.4.1 Sensor de luz

El sensor de luz consta de un LDR (Light Dependent Resistor) que se muestra en la Figura 2.24, y una resistencia de 10kohm en serie, donde el LDR o fotorresistor está conectado a 5 V y la resistencia a GND, como se muestra en la Figura 2.25. Estos dos componentes funcionan como un divisor de voltaje, cuando el valor resistivo del LDR se haga más pequeño debido a un cambio en la luz en su parte fotosensible, habrá una caída de voltaje en el punto común entre estos dos componentes, así mismo, si se incide menor cantidad de luz sobre el fotorresistor, su resistencia aumentará y por ende en ese punto el voltaje será mayor. Como se trata de una señal analógica proporcional a la luz, se conecta la salida de este circuito al chip MCP3008, y es este quien envía el valor del sensor a la Raspberry pi a través de un bus de datos.

Figura 2.24: LDR



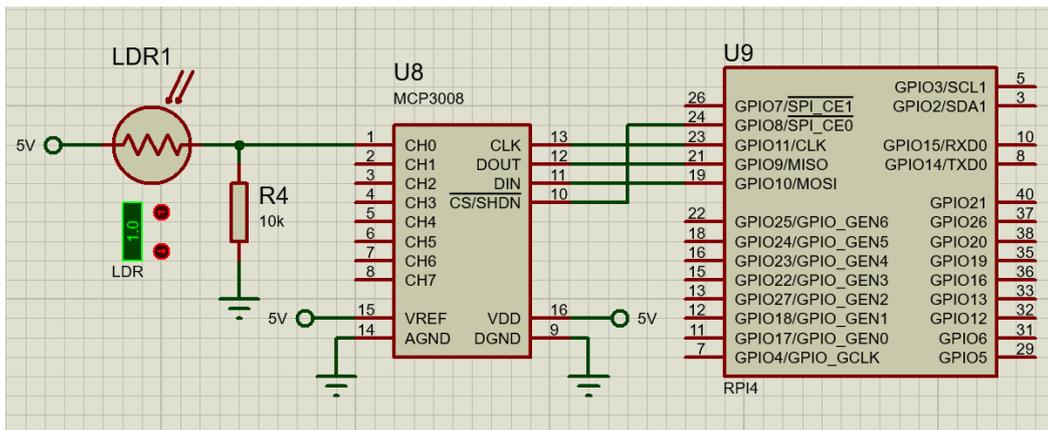
(Embajadores, s/f)

Cuando el sensor de luz detecta oscuridad el valor analógico en el Raspberry pi tiende a ser cercano a 0, y cuando detecta una considerable incidencia de luz, este valor es cercano a 1. Se realizó una conversión de este valor analógico a un aproximado en luxes, para ello se usó un luxómetro y se registraron los valores tanto de la fotorresistencia como del luxómetro, para aplicar una regla de tres y llegar a la ecuación 2.3.

$$Luz[lux] = \frac{220000 * valor\ del\ sensor}{94} \quad \text{Ecu.}(2.3)$$

En la Figura 2.25 se muestra el esquema de conexión del LDR con el actuador:

Figura 2.25: Conexión del LDR con la Raspberry pi

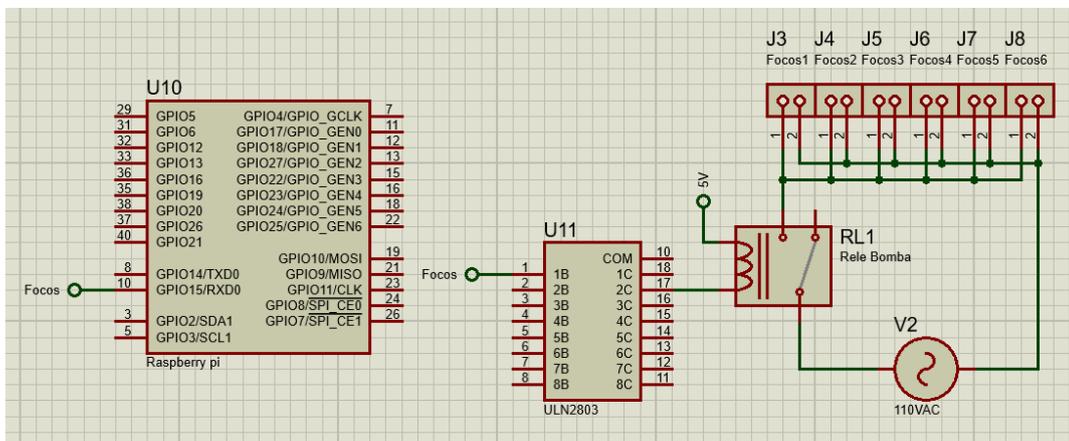


Autores: Víctor Yanchapaxi y Bryan Taboada

2.4.2 Circuito de potencia

El circuito de potencia para el encendido y apagado de las luces del sistema aeropónico es similar al utilizado para el encendido y apagado de la bomba de agua en el sistema de riego de la Figura 2.19. La potencia que se maneja en este circuito es de 192W, que corresponde a la potencia de todos los focos. Los focos se energizan con una fuente AC de 110VAC, por lo que el circuito de potencia del sistema de iluminación maneja una corriente de 1.74 A, que es un poco mayor a la que circula en el circuito de potencia del sistema de riego, sin embargo, también se utiliza un relé de 5V para este bloque. En la Figura 2.26 se muestra el esquema de conexión del arreglo de focos con el circuito de potencia y la Raspberry pi.

Figura 2.26: Conexión de los focos con la Raspberry pi

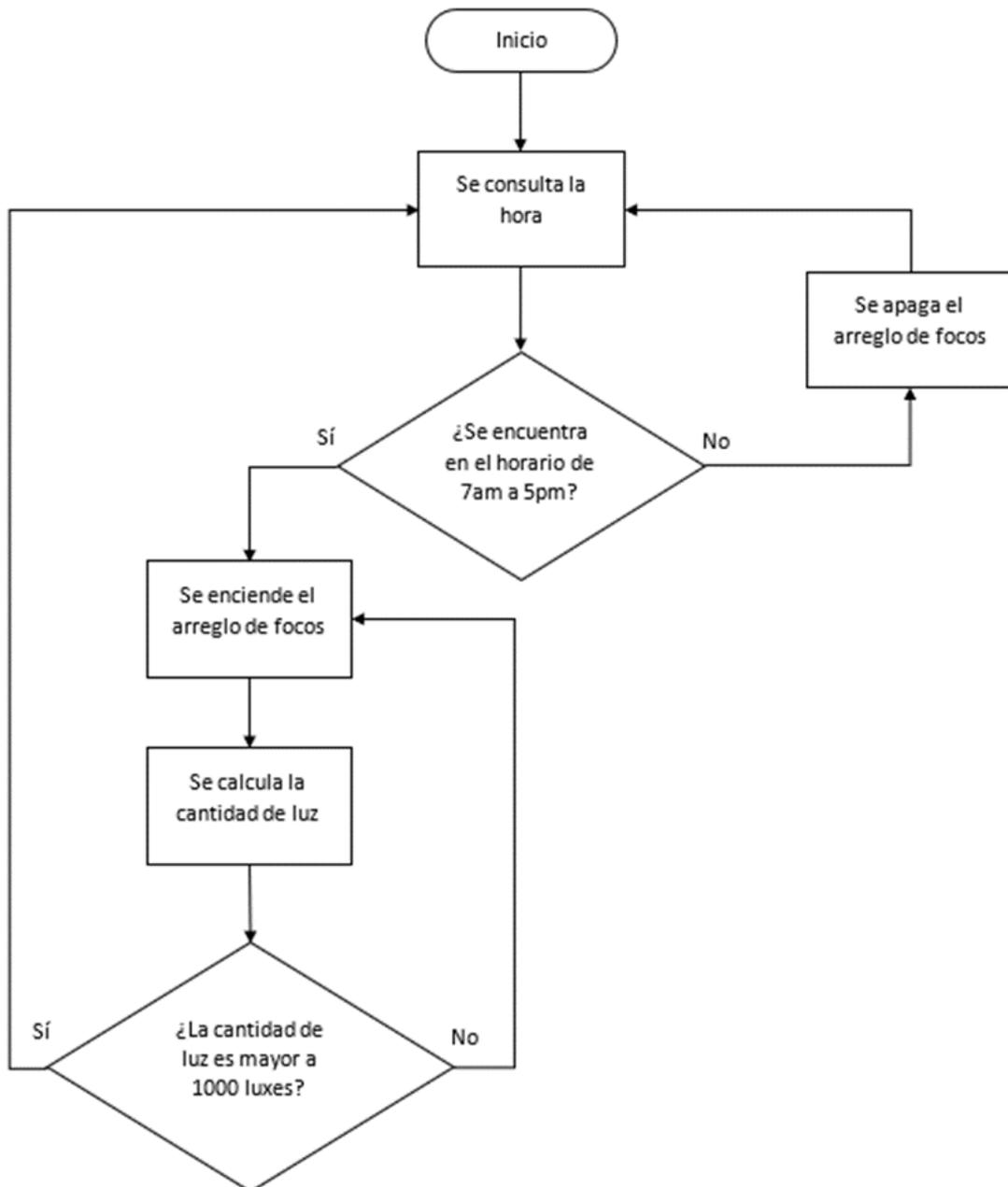


Autores: Víctor Yanchapaxi y Bryan Taboada

2.4.3 Lógica de operación del sistema

El sistema funciona con el RTC (Real Time Clock) de la Raspberry pi para temporizar el encendido y apagado de los focos, con el sensor de luz se comprueba el encendido de estos, como en la Figura 2.27.

Figura 2.27: Diagrama de flujo del sistema de iluminación

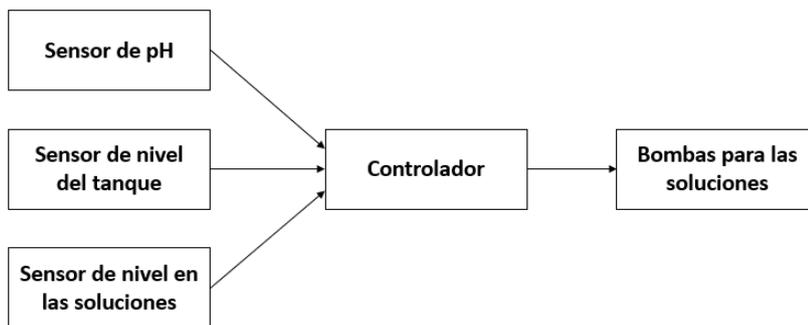


Autores: Víctor Yanchapaxi y Bryan Taboada

2.5 Sistema de mezcla de nutrientes

Se encarga de que el agua que se riega a las plantas tenga los nutrientes necesarios y que el valor de pH se encuentre en los niveles óptimos, estos procesos se realizan de manera periódica y automática. Este sistema tiene 4 botellas, donde 3 contienen soluciones nutritivas: solución A, solución B y solución C respectivamente, y la última botella es un regulador de pH, para este fin se usaron 4 sensores de nivel y 4 bombas para cada envase. En este sistema también se hace uso del sensor de nivel del tanque porque la cantidad de solución A, B o C que se debe adicionar dependerá de la cantidad de agua que haya en el tanque. Para este sistema se diseñó el diagrama de bloques que se muestra en la Figura 2.28.

Figura 2.28: Diagrama de bloques del sistema de mezcla de nutrientes



Autores: Víctor Yanchapaxi y Bryan Taboada

2.5.1 Sensor de pH

El sensor de pH es un instrumento capaz de medir el nivel de alcalinidad o de acidez de un líquido, en este caso se requiere utilizarlo para medir el pH de la solución nutritiva, ya que este valor no puede estar fuera del rango entre 5.5 y 6.5 de acidez, de lo contrario las plantas se marchitarían.

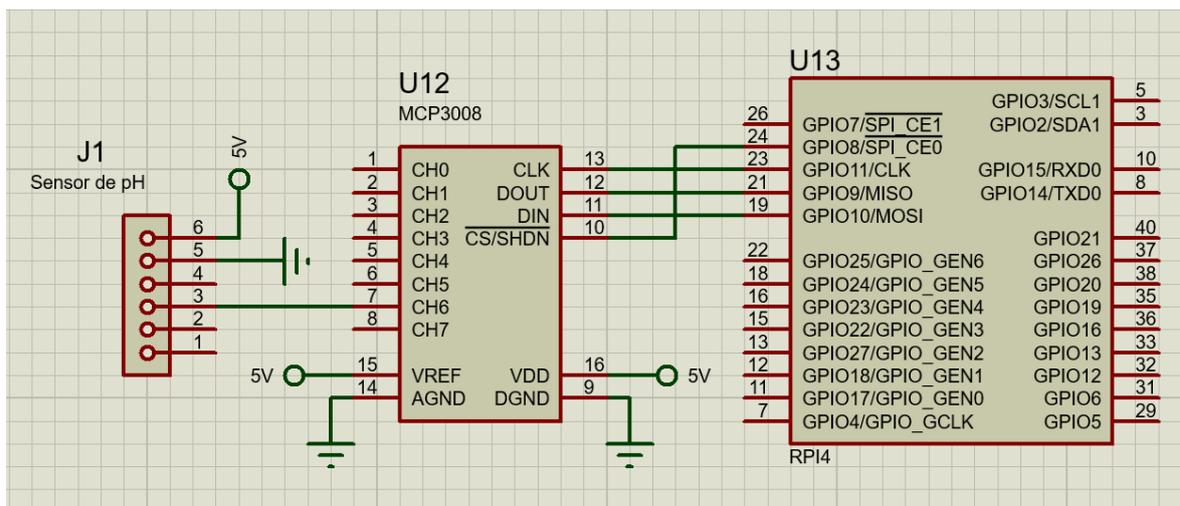
Figura 2.29: Sensor de pH



(“PH-4502C Kit sensor de pH económico”, s/f)

El sensor de pH viene con un circuito acondicionador que toma la señal de la sonda y lo condiciona en dos salidas: una analógica y la otra digital, la señal digital no se usa en este proyecto, pero puede regularse con el potenciómetro que se encuentra al lado contrario del conector de la sonda, cerca de los pines como se ve en la Figura 2.29. La señal analógica está conectada a uno de los canales del MCP3008 y luego a la Raspberry pi, como en la Figura 2.30. Antes de leer los datos del sensor es necesario calibrarlo, para esto se necesita una solución cuyo pH esté definido, o bien un pHmetro ya calibrado del cual tomar referencia. Luego se regula el potenciómetro de precisión que se encuentra cerca del conector de la sonda, Figura 2.29, girándolo en el sentido horario cuando el valor que se muestra está por debajo del valor del cual se toma referencia, y en sentido antihorario cuando el valor esté por encima de este. Esta calibración se debe hacer regularmente, ya que el constante uso hace que el sensor de pH se descalibre.

Figura 2.30: Conexión del sensor de pH con la Raspberry pi

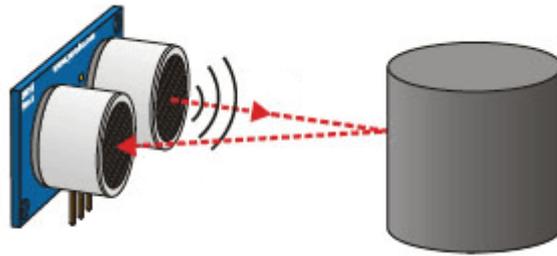


Autores: Víctor Yanchapaxi y Bryan Taboada

2.5.2 Sensor de nivel del tanque

El sistema requiere de un sensor de nivel, ya que la adición de nutrientes se hace en base a la cantidad de agua en litros que haya en el tanque. Una de las formas más fáciles y económicas de medir nivel es usando un sensor de ultrasonido como el HC-SR04 que se puede apreciar en la Figura 2.31. Este sensor usa ondas sonoras por encima del rango audible del ser humano para detectar obstáculos y la distancia a las que estos se encuentran. El rango de distancia en el que este sensor puede medir va desde los 2 cm hasta los 4 m (SparkFun Electronics, s/f).

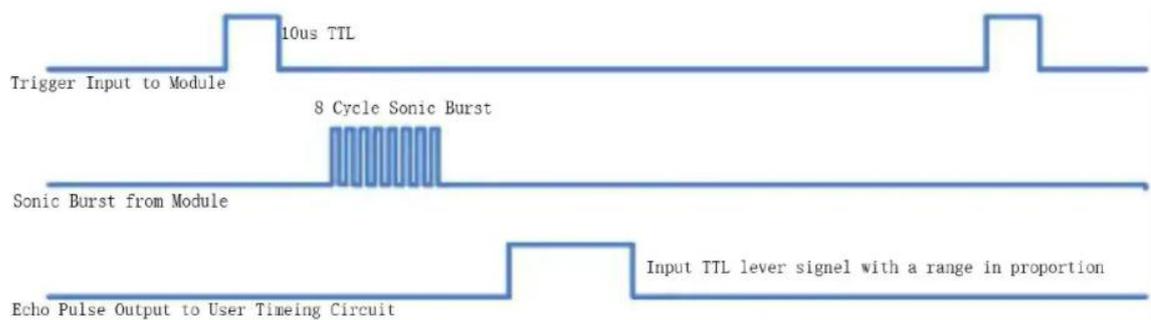
Figura 2.31: Sensor de ultrasonido HC-SR04



(“HC-SR04 sensor ultrasonido con atmega-1”, 2016)

El HC-SR04 tiene cuatro pines, dos de alimentación y dos de señal, los pines de señal son denominados Trigger y Echo, y ambos se pueden conectar a cualquiera de los GPIO de la Raspberry pi, ya que estos pines funcionan con señales digitales que es muestran en la Figura 2.32. Sin embargo, el pin Echo requiere que se conecte un divisor de voltaje antes de conectarse a alguna entrada del controlador, como se puede ver en la Figura 2.33.

Figura 2.32: Funcionamiento el sensor HC-SR04



(SparkFun Electronics, s/f)

Según la información de (SparkFun Electronics, s/f), para hacer que este sensor funcione, se debe enviar un pulso de activación de 10 us a través del Trigger, consecuentemente el sensor de ultrasonido emitirá una onda cuadrada de 40 KHz por su transmisor y, en ese momento el receptor del sensor estará en espera de recibir dicha onda después de que haya rebotado en un obstáculo, cuando esto sucede, el pin Echo se pone a 1 lógico y se lo recibirá por el GPIO al que se encuentre conectado. Por tanto, es importante colocar un temporizador desde que se emite el pulso por el Trigger hasta que se recibe la señal por el Echo, y esto se realiza mediante programación con la ayuda de la función `time.time()` del lenguaje Python en la Raspberry pi. Obtenido este tiempo se puede aplicar la Ecu.(2.4) para pasar el dato de tiempo a distancia en centímetros a la que se encuentra el obstáculo.

$$Distancia [cm] = tiempo * 17150 \quad Ecu.(2.4)$$

En este proyecto no se requiere el nivel en centímetros sino más bien la cantidad en litros de solución nutritiva que hay en el tanque, para ello se toman valores de referencia del nivel del tanque llenándolo con una determinada cantidad de agua, que para este caso fue con el nivel mínimo permisible que son 30 litros, luego se toma otro valor de referencia, en este caso el nivel máximo del tanque que son 60 litros. Se aplicó una regla de 3 y se determinó la ecuación Ecu.(2.5).

$$Nivel[lts] = \frac{30 * (distancia - 34)}{-8} + 30 \quad Ecu.(2.5)$$

Sabiendo la cantidad en litros del tanque se puede calcular la dosis necesaria de nutrientes, y además es posible implementar una alarma cuando el nivel del tanque se acerque al nivel mínimo de 30 litros, porque debajo de este nivel la bomba de agua queda descubierta y, puede sobrecalentarse y dañarse si no tiene agua que mover.

Para comprobar que el dato de distancia se pasa a nivel en litros con la Ecu.(2.5), se la evalúa en la Tabla 2.5.

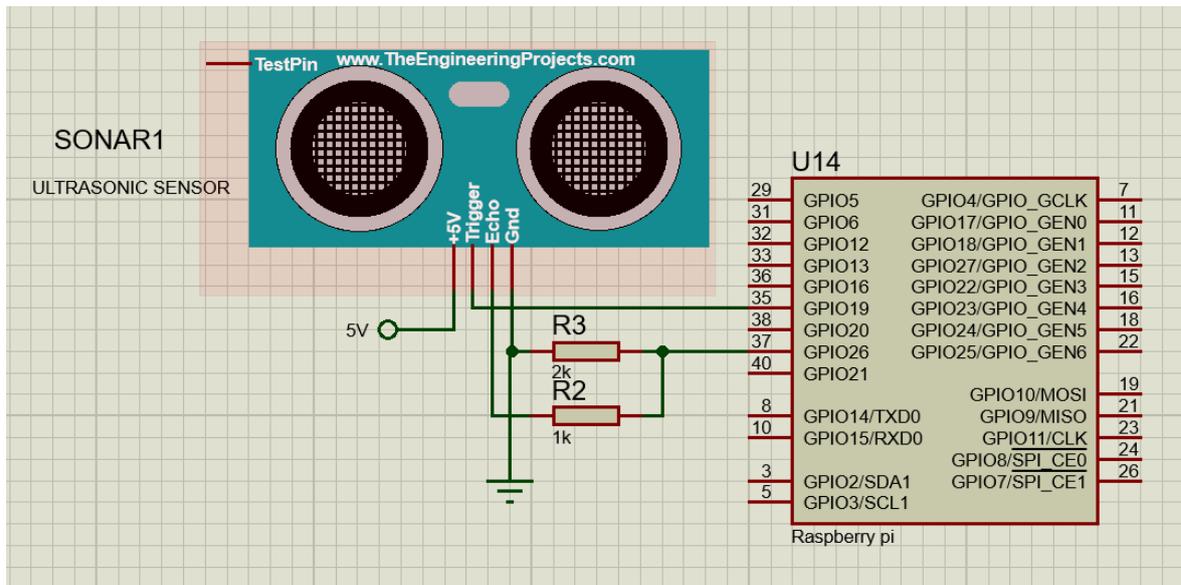
Tabla 2.5: Evaluación de la Ecu.(2.5)

Distancia [cm]	Estado del tanque	Nivel [Litros]
34	Nivel mínimo	30
32	Nivel bajo	37.5
30	Nivel medio	45
28	Nivel alto	52.5
26	Nivel máximo	60

Conversión de distancia del HC-SR04 a nivel en litros, Autores: Víctor Yanchapaxi y Bryan

Taboada

Figura 2.33: Conexión del sensor HC-SR04 con la Raspberry pi



Autores: Víctor Yanchapaxi y Bryan Taboada

2.5.3 Sensor de nivel de las soluciones

Se usó un sensor de agua de modelo t1592 que es el que se muestra en la Figura 2.34, el cual es capaz de medir si entre sus terminales hay agua o no, por esa razón se lo suele usar como detector de lluvia. Sin embargo, en este proyecto se lo utiliza para determinar si existe suficiente cantidad de líquido en las botellas de soluciones nutritivas y en la de regulador de pH, de lo contrario se avisará al operario del sistema de monitoreo remoto que hace falta rellenar los mismos.

Figura 2.34: Sensor de agua t1592

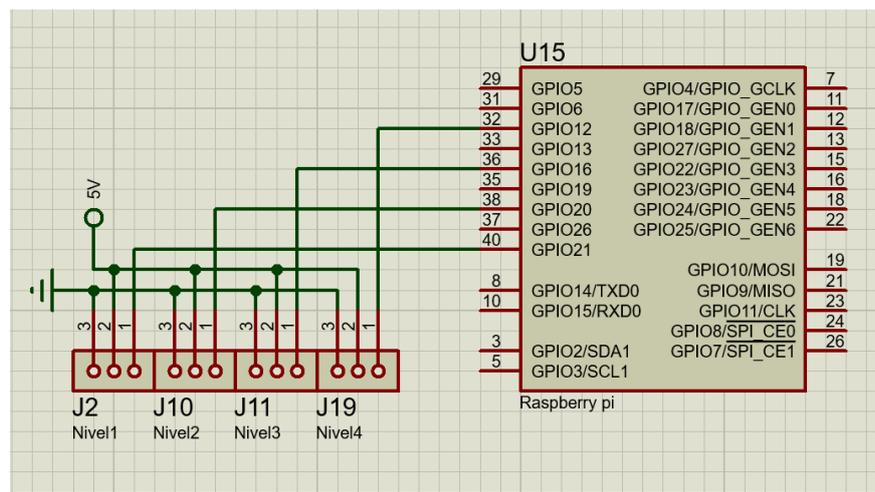


(“Sensor de nivel de lluvia / agua rojo T1592”, s/f)

El sensor tl1592 tiene 3 pines, dos de alimentación y uno de señal, el pin de señal es analógico y proporcional a la cantidad de agua que topan sus contactos descubiertos, pero en este caso, estos pines analógicos son usados como digitales, es decir, están colocados en la parte inferior de las botellas de soluciones, para cuando el nivel esté por encima de este sensor la señal analógica será la máxima posible, porque el sensor está completamente sumergido, y cuando el nivel caiga por debajo de donde se encuentra el sensor, este no será capaz de detectar nada de agua en sus terminales por lo que la salida analógica será cercana a 0.

Los sensores de nivel para los envases de las soluciones A, B, C y regulador de pH, están conectados con la Raspberry pi como se observa en la Figura 2.35.

Figura 2.35: Conexión de los sensores de nivel de las soluciones con la Raspberry pi



Autores: Víctor Yanchapaxi y Bryan Taboada

2.5.4 Bombas para las soluciones

Para colocar las soluciones dentro del tanque se usan mini bombas sumergibles de 5V dentro de cada botella de nutrientes para que cuando sean activadas expulsen el contenido de estas hacia el agua, y posteriormente se mezclen entre sí en el subsistema de mezcla.

Figura 2.36: Bomba sumergible de 5Vdc



(Electrostore, s/f)

El caudal de estas mini bombas va a ser pequeño, en el orden de los mililitros, y estos han sido determinados de manera experimental, donde se colocó líquido en los envases de las soluciones, luego se activó cada bomba por un lapso de 10 s y se pesó la cantidad de líquido que estas habían sacado de los envases. Así se determinó que la bomba A, bomba que se encuentra dentro de la botella de la solución A, es capaz de mover 89 ml en 10 s, dividiendo este valor para 10 se obtiene el caudal de litros sobre segundo, es decir, el caudal de la bomba A es 8.9 ml/s, se realizó la misma prueba para la bomba B obteniéndose un caudal de 10 ml/s y para la bomba C se tiene un caudal de 9.8 ml/s. La bomba para el regulador de pH es diferente ya que esta solución no debe ser regada en el orden de los mililitros sino en gotas, por tanto, al final del tubo de la bomba de pH se encuentra un pequeño aspersor, que se opone al caudal y deja caer pequeñas gotas, tras la prueba experimental se ha determinado que esta bomba tiene un caudal aproximado de 5 gotas por segundo. Estos datos se presentan en la Tabla 2.6.

Tabla 2.6: Caudal de las bombas para las soluciones

Mini bomba	Solución	Caudal [ml/s]
Bomba A	Solución A	8,9
Bomba B	Solución B	10
Bomba C	Solución C	9.8
Bomba pH	Regulador de pH	1

Autores: Víctor Yanchapaxi y Bryan Taboada

La programación para la activación de estas bombas se hacen en base a recomendaciones del fabricante de las soluciones, donde se especifica que la solución A debe ser mezclada en el agua todos los días Lunes y en una cantidad de 3ml por cada litro que haya en el

tanque, la solución B los días miércoles en la misma proporción de 3 ml por cada litro de agua, y la solución C los días viernes con 3ml de la misma por cada litro, la adición del regulador de pH dependerá de los datos de pH obtenidos del pHmetro, y puede ser colocado en cualquier momento y en la cantidad que el controlador decida para mantenerlo estable en el rango de 5.5 a 6.5.

Conociendo el valor de tanto del caudal de cada bomba como del nivel del tanque a través del sensor de nivel, se puede determinar por regla de 3 el tiempo de encendido de cada bomba para que se arroje la cantidad de solución necesaria, de esta manera:

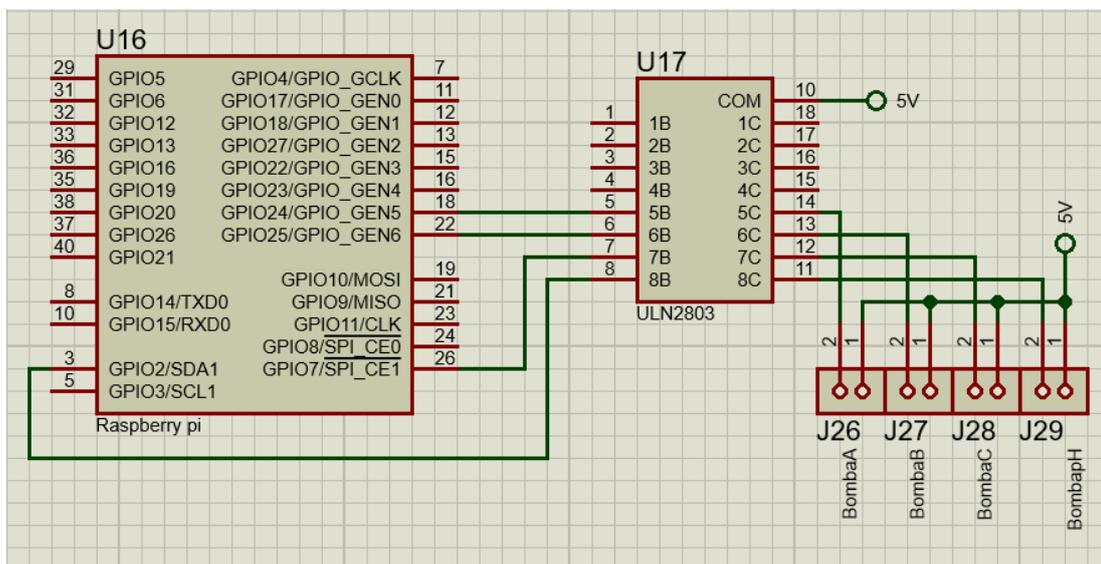
$$Tiempo(Bomba A)[s] = \frac{30 * nivel}{89} \quad \text{Ecu.(2.6)}$$

$$Tiempo(Bomba B)[s] = \frac{30 * nivel}{100} \quad \text{Ecu.(2.7)}$$

$$Tiempo(Bomba C)[s] = \frac{30 * nivel}{98} \quad \text{Ecu.(2.8)}$$

Las bombas estarán conectadas a la Raspberry pi de la forma que se muestra en la Figura 2.37.

Figura 2.37: Conexión de las bombas de las soluciones con la Raspberry pi

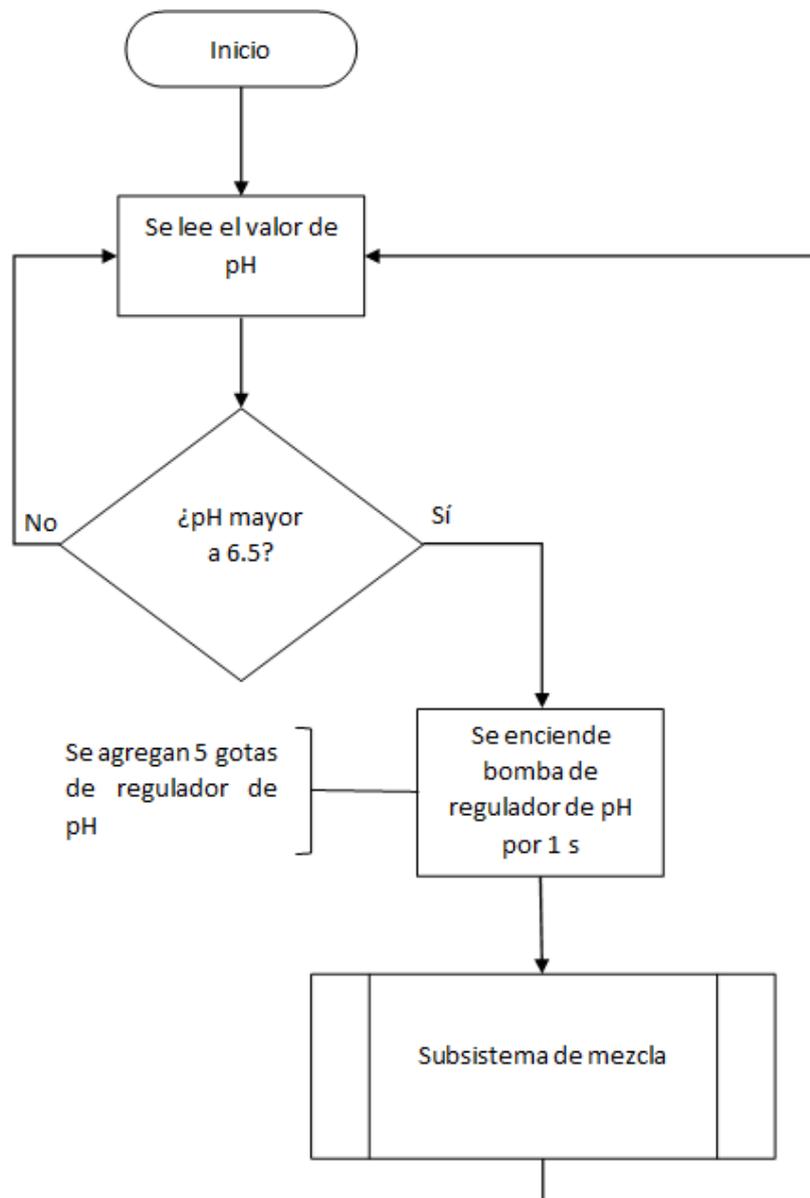


Autores: Víctor Yanchapaxi y Bryan Taboada

2.5.5 Lógica de operación del sistema

Para explicar el funcionamiento del sistema de mezcla de nutrientes y control de pH se han diseñado 3 diagramas de flujo que corresponden a la Figura 2.38, Figura 2.39 y Figura 2.40 respectivamente, donde esta última se trata de un subsistema de mezclado para homogenizar la solución nutritiva después de la adición de nutrientes o del regulador de pH. Los 3 procedimientos se realizan en la Raspberry pi y se muestran a continuación:

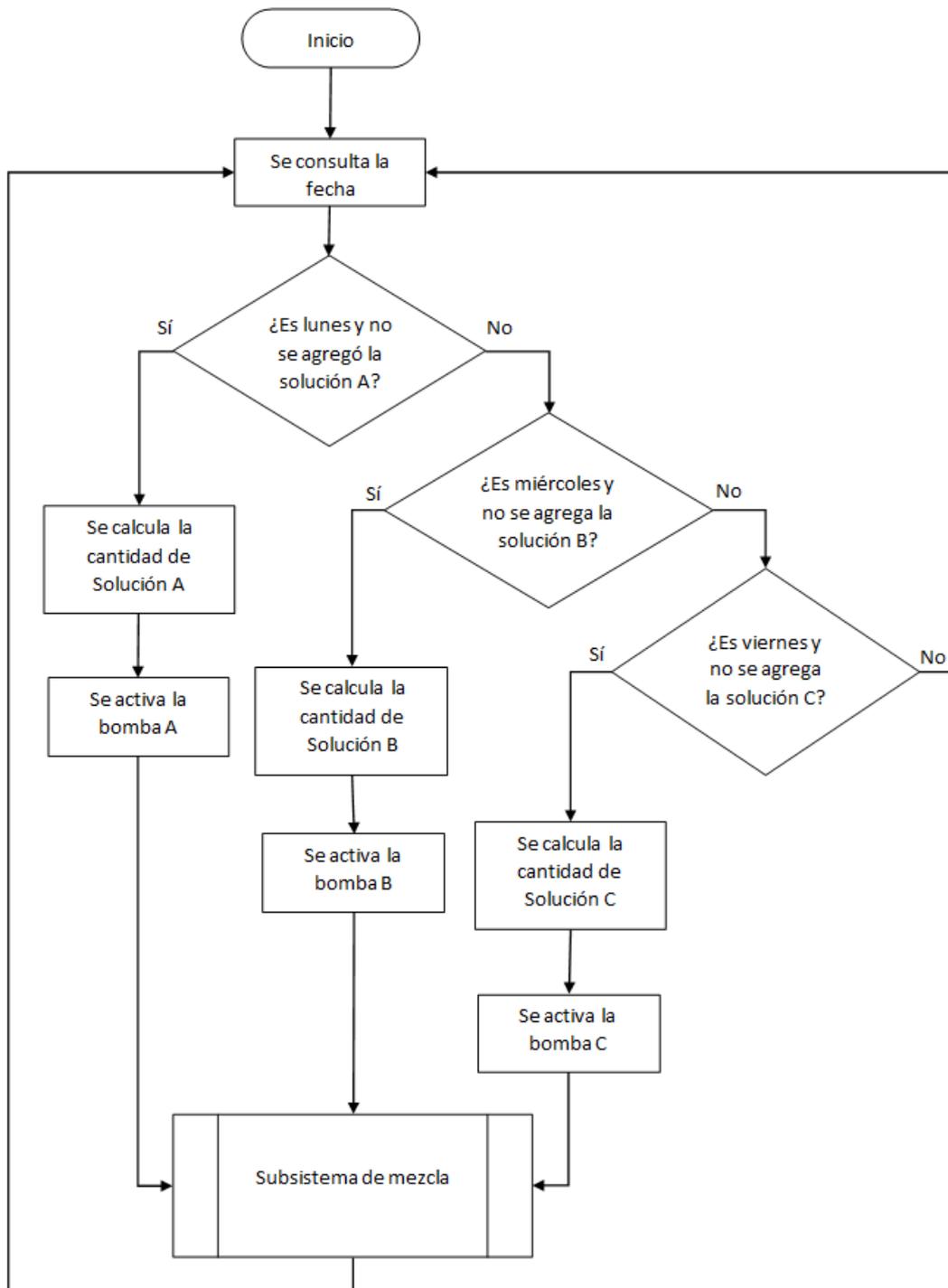
Figura 2.38: Diagrama de flujo del sistema de regulación de pH



Autores: Víctor Yanchapaxi y Bryan Taboada

El otro lazo de control que se maneja en este sistema es el de adición de las soluciones A, B y C a la mezcla de nutrientes, que se puede explicar con la Figura 2.39.

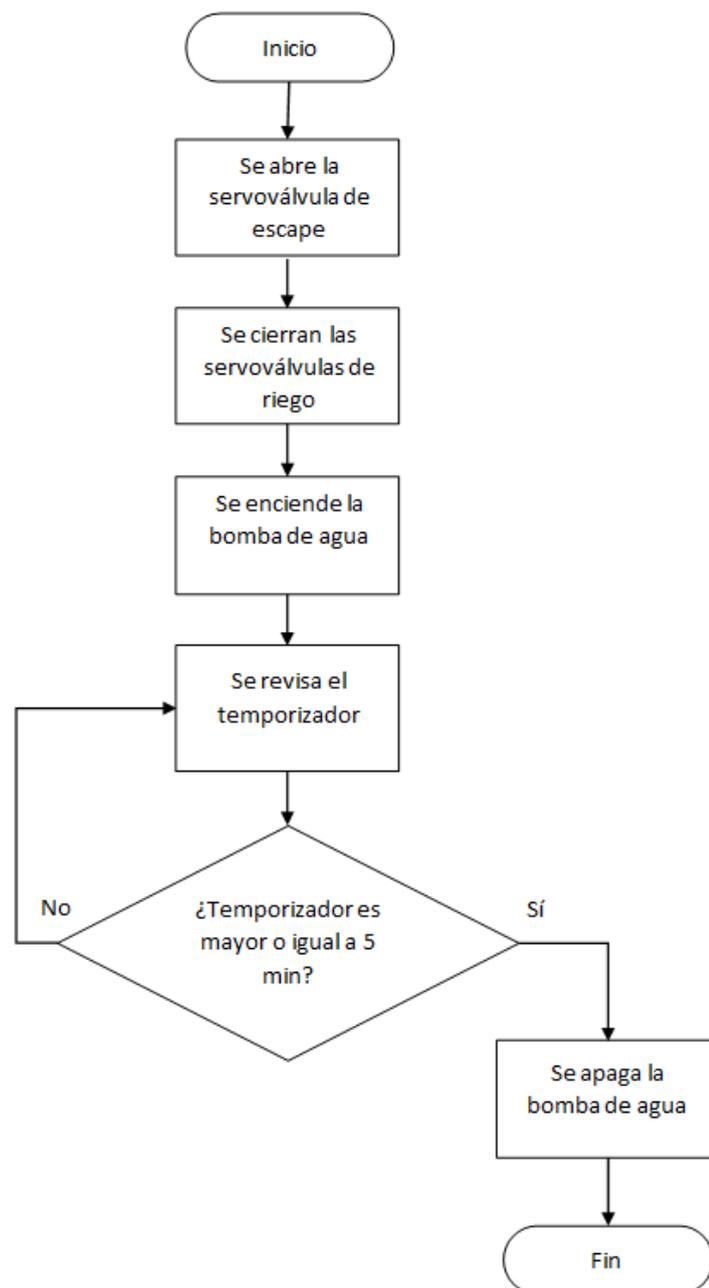
Figura 2.39: Diagrama de flujo del sistema de mezcla de nutrientes



Autores: Víctor Yanchapaxi y Bryan Taboada

En la Figura 2.38 y Figura 2.39 se utiliza un bloque de función que corresponde al subsistema de mezcla que realiza una secuencia de pasos como se muestra en el diagrama de flujo de la Figura 2.40.

Figura 2.40: Diagrama de flujo del subsistema de mezcla

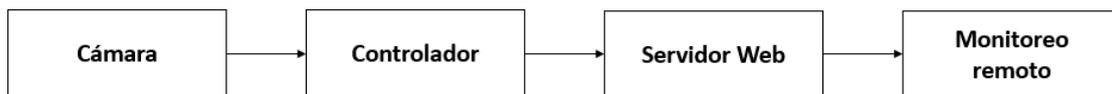


Autores: Víctor Yanchapaxi y Bryan Taboada

2.6 Sistema de vigilancia

Cuenta con una cámara de la cual se obtienen datos de imagen y video, que son capturados por el controlador y colocados en un servidor Web alojado en la misma Raspberry pi para que la aplicación de monitoreo remoto desarrollada en Android Studio pueda acceder a esto datos de video en tiempo real en forma de streaming. En la Figura 2.41 se presenta un diagrama de bloques de este sistema.

Figura 2.41: Diagrama de bloques del sistema de vigilancia

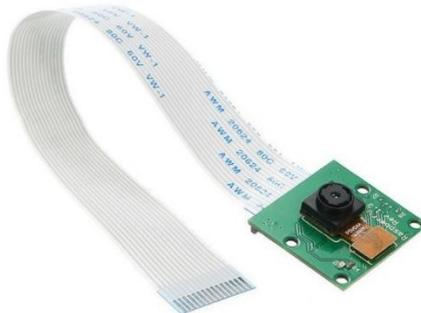


Autores: Víctor Yanchapaxi y Bryan Taboada

2.6.1 Cámara

La cámara utilizada es el modelo OV5647 y se muestra en la Figura 2.42, es de especial y exclusivo uso para la Raspberry pi al tener un tipo de conector especial. La conexión entre esta cámara y el controlador es a través de un cable flexible de 15 vías.

Figura 2.42: Cámara OV5647



(“Cámara Raspberry Pi v2 - 8 Megapixels”, s/f)

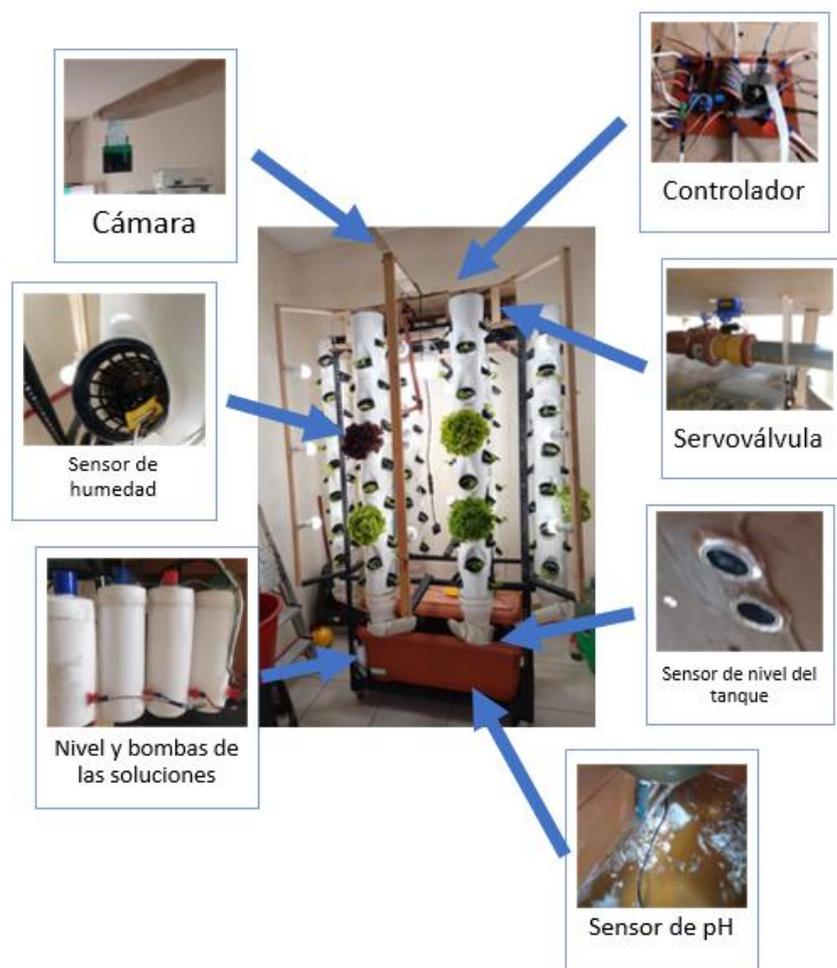
La cámara OV5647 tiene un lente de 5 megapíxeles a una resolución de 2592x1944 (“5MP OV5647”, s/f), cuya calidad de imagen no es tan buena, lo cual es aceptable debido al costo que esta tiene, sin embargo, es suficiente para ver el estado de este sistema porque el prototipo se encuentra a poca distancia y no tiene mayores detalles.

A través de una librería de Python se crea un servidor web que transmite los datos de video de la cámara en tiempo real, similar a un streaming. El sistema de monitoreo remoto deberá tener un visualizador de páginas web para que el usuario pueda ver esta transmisión de video.

2.7 Sistema aeropónico

La Figura 2.43 muestra el sistema aeropónico implementado en este trabajo y la ubicación de los sensores y actuadores.

Figura 2.43: Sistema aeropónico



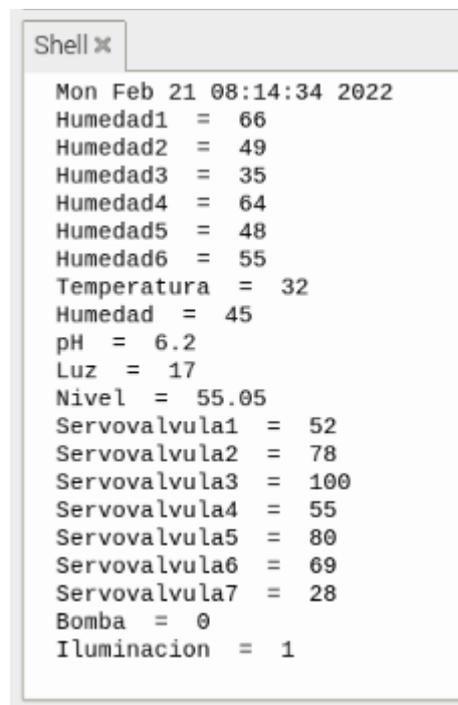
Autores: Víctor Yanchapaxi y Bryan Taboada

El controlador se encuentra en la parte superior del sistema aeropónico donde se sabe que no va a ser afectado por gotas que puedan caer del sistema de riego, ni de manipulación accidental o intencionada de personas, mientras que los sensores de humedad han sido

colocados dentro de las canastillas de cada uno de los tubos de siembra; las servoválvulas están conectadas en el sistema de cañerías justo antes de los aspersores, el sensor de nivel se encuentra debajo de la tapa del tanque, el sensor de pH se encuentra dentro del tanque a la altura que le permita mantenerse siempre en contacto con el agua; los sensores de nivel al igual que las bombas se encuentran instaladas dentro de las 4 botellas para las soluciones: A, B, C y regulador de pH. Finalmente, la cámara ha sido colocada con una extensión de 1.5 m hacia afuera del sistema aeropónico para un mejor ángulo de visión.

Los datos de los sensores capturados por la programación de la Raspberry pi se muestran en la Figura 2.44.

Figura 2.44: Datos capturados por la Raspberry pi



```
Shell x
Mon Feb 21 08:14:34 2022
Humedad1 = 66
Humedad2 = 49
Humedad3 = 35
Humedad4 = 64
Humedad5 = 48
Humedad6 = 55
Temperatura = 32
Humedad = 45
pH = 6.2
Luz = 17
Nivel = 55.05
Servovalvula1 = 52
Servovalvula2 = 78
Servovalvula3 = 100
Servovalvula4 = 55
Servovalvula5 = 80
Servovalvula6 = 69
Servovalvula7 = 28
Bomba = 0
Iluminacion = 1
```

Autores: Víctor Yanchapaxi y Bryan Taboada

En el desarrollo del prototipo de la Figura 2.43 se emplearon componentes eléctricos y electrónicos que se detallan en la Tabla 2.7.

Tabla 2.7: Gastos en componentes electrónicos

Cantidad	Descripción	Valor
1	Raspberry Pi	\$89
1	Cámara	\$10,50
1	Fotocelda LDR	\$0,50
1	Sensor de temperatura DHT11	\$3,50
1	Sensor de pH	\$45
1	Sensor de nivel HC-SR04	\$2,75
6	Sensor de humedad YL-69	\$15,00
7	Servomotores	\$24,50
7	Válvulas	\$1
1	Bomba de agua sumergible	\$90
4	Bombas sumergibles de pecera	\$18
4	Sensores de nivel TI1592	\$5
1	MCP3008	\$6
1	ULN2003	\$3
2	Relés de 5V	\$2
8	Borneras	\$2,80
2	Tiras porta pines macho.	\$2
2	Tiras porta pines hembra	\$2
4	Resistencia	\$0,10
2	Socket	\$0
19	Focos	\$34,20
19	Boquilla	\$9,50
6	Metros alambre UTP	\$3,00
1	Baquelita	\$2
	Total	\$372

Autores: Víctor Yanchapaxi y Bryan Taboada

El presupuesto aproximado para construir el sistema aeropónico propuesto en la Figura 2.43 se obtuvo tras sumar el total de los gastos en material eléctrico y electrónico, material estructural y costos de producción que incluyen mano de obra y costos de transporte, como se muestra en la Tabla 2.8.

Tabla 2.8: Presupuesto del sistema aeropónico

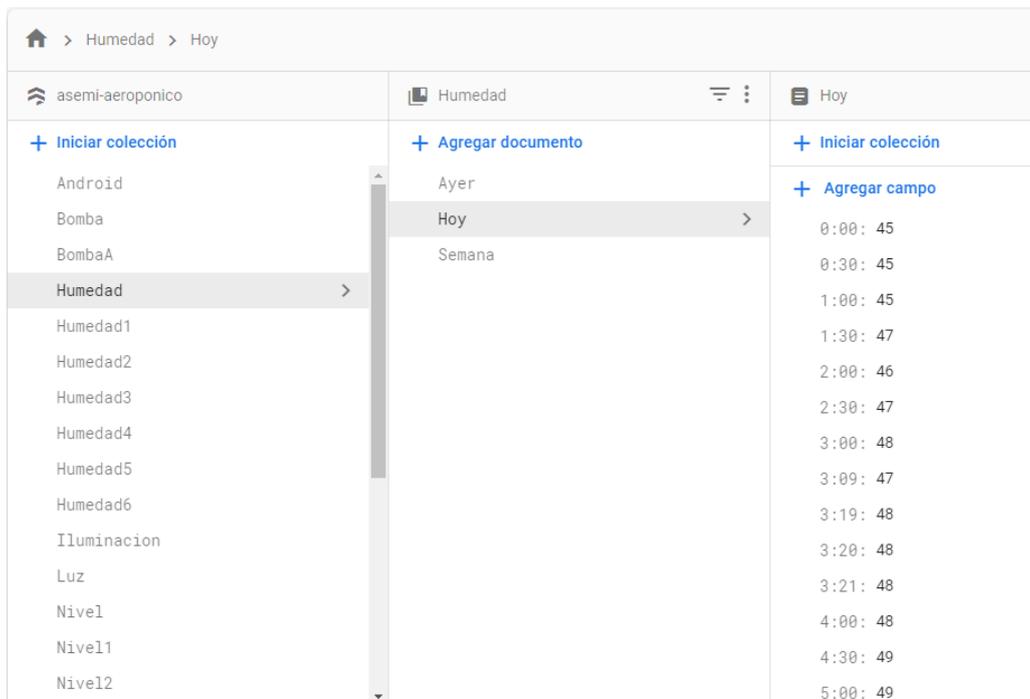
Gastos en material electrónico.	\$372
Gastos en material estructural	\$250
Costos de producción	\$250
Total	\$872

Autores: Víctor Yanchapaxi y Bryan Taboada

2.8 Base de datos

Se hace uso de la base de datos Firestore de Firebase para almacenar, actualizar y leer los datos generados por los sensores y actuadores del sistema aeropónico organizados en colecciones, luego en documentos y guardados en un formato de clave-valor. Esta forma de representar la información facilita la escritura de las mismas ya que se tiene un directorio predeterminado para cada sensor, y dentro de este se puede clasificar los datos a conveniencia, lo cual fue útil para este proyecto porque se hace posible guardar los datos del día de ayer en un documento denominado “Ayer”, y de toda la semana en un documento “Semana”, al mismo tiempo que se pueden seguir generando datos en el documento “Hoy”, como en la Figura 2.45.

Figura 2.45: Datos almacenados en Firestore



Autores: Víctor Yanchapaxi y Bryan Taboada

Respeto a los costos por el uso de este servicio, Firebase maneja dos planes de precios: el plan Spark que es gratuito y el plan Blaze donde se paga por uso adicional de servicios. En este caso, el plan Spark es más que suficiente, ya que permite realizar hasta 20000 operaciones de escritura, 50000 operaciones de lectura y 20000 eliminaciones cada día en la base de datos de Firestore, como se muestra en la Tabla 2.9. El plan de pago Blaze en cambio, está diseñada para el desarrollo de aplicaciones de gran escala, que no es este caso.

Tabla 2.9 Planes de pago de Firebase

Productos	Plan Spark (Gratis)	Plan Blaze (Prepago)
Autenticación de usuarios	10 000 por mes	\$0.01 por verificación adicional
Escritura de datos en Firestore	20 000 por día	\$0.18 por cada 100 000 escrituras adicionales
Lectura de datos en Firestore	50 000 por día	\$0.06 por cada 100 000 lecturas adicionales
Almacenamiento	5GB	\$0.026 por cada GB adicional

FUENTE: Documentación de Firebase

2.9 Sistema de monitoreo remoto

Es una interfaz gráfica desarrollada en Android Studio para que el usuario pueda acceder a los datos e información del sistema aeropónico desde un smartphone con sistema operativo Android.

En primer lugar, la aplicación muestra una pantalla de inicio de sesión como en la Figura 2.46, donde el usuario tendrá que autenticarse mediante Firebase Authentication con un correo electrónico y contraseña que solo el operador u operadores deberían poseer, ya que, una vez autenticados tendrán la supervisión y control total de todo el sistema aeropónico.

Figura 2.46: Pantalla de autenticación



Autores: Víctor Yanchapaxi y Bryan Taboada

La segunda pantalla solo se inicia cuando se ha ingresado un correo y una contraseña válida, aquí se muestra la información de todos los sensores y actuadores del sistema como: sensores de humedad, sensor de temperatura, sensor de nivel, sensor de pH, control de la bomba de agua, etc. La información será mostrada en un indicador de progreso, que tiene el nombre del sensor y el último valor que este haya sentido, junto a cada indicador se encuentra un gráfico que se visualizará en color verde si el dato se encuentra dentro de los límites correctos de funcionamiento, como se puede ver en la Figura 2.47.

Figura 2.47: Pantalla principal para el monitoreo de los sensores



Autores: Víctor Yanchapaxi y Bryan Taboada

Para los actuadores se tienen botones para el control manual o automático de los mismos, como se aprecia en la Figura 2.48, cuando se selecciona el control manual, el usuario podrá cambiar el estado de los actuadores a su voluntad para el caso que se considere necesario, los actuadores que se pueden controlar son: el encendido y apagado de las luces, el encendido y apagado de la bomba de agua, y el control porcentual de las servoválvulas.

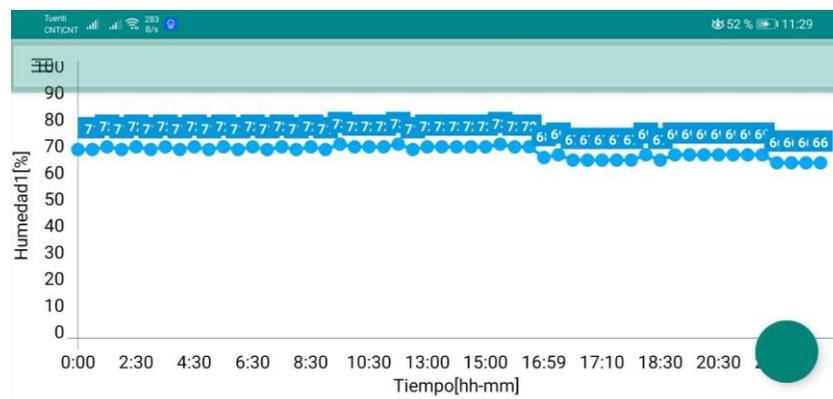
Figura 2.48: Pantalla principal con los controles de los actuadores



Autores: Víctor Yanchapaxi y Bryan Taboada

Sin embargo, para una visión más detallada del comportamiento del sensor se puede girar el smartphone para que quede en posición horizontal, entonces se muestran los gráficos de los valores capturados por cada sensor y cada actuador durante el día, de esta manera es posible detectar fallas o tendencias, como en la Figura 2.49.

Figura 2.49: Pantalla principal con la representación de los datos del día



Autores: Víctor Yanchapaxi y Bryan Taboada

En la esquina superior izquierda de la pantalla se encuentra un menú desplegable, Figura 2.50, donde se muestran algunas opciones como: el origen de los datos, es decir, se puede seleccionar de que día se desea visualizar la información de los sensores y actuadores, de momento se consideró necesario los gráficos de ese mismo día, del día anterior, y un resumen de toda la semana, y así comparar si hubo cambios significativos entre un día u otro. La otra opción que tiene el menú es la de filtrar los datos para cada sensor, es decir, se puede elegir solo uno de ellos para que el monitoreo sea más rápido.

Figura 2.50: Pantalla principal con el menú desplegado



Autores: Víctor Yanchapaxi y Bryan Taboada

En la esquina inferior derecha se encuentra un botón flotante que tras ser accionado permite activar la cámara de la Raspberry y empezar la transmisión en video de lo que está enfocando la cámara. De esta manera, el usuario podrá visualizar de que se hayan producido los cambios que se hayan realizado, o ver que todo esté funcionando correctamente, como en la Figura 2.51. Desde esa pantalla se puede regresar a la pantalla principal de la Figura 2.47 presionando el botón de regresar.

Figura 2.51: Pantalla del monitoreo del sistema vía streaming



Autores: Víctor Yanchapaxi y Bryan Taboada

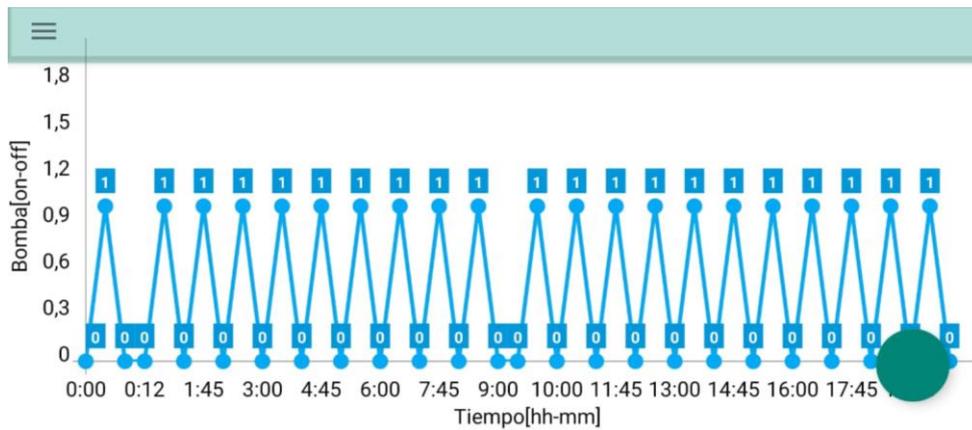
CAPÍTULO 3

PRUEBAS Y RESULTADOS

3.1 Prueba del sistema de riego

Para comprobar el funcionamiento del sistema de riego se utilizan las gráficas de la aplicación de monitoreo remoto con los datos recopilados y almacenados durante el día en Firestore. Para eso se toma como referencia el gráfico de encendido y apagado de la bomba de agua que se muestra en la Figura 3.1.

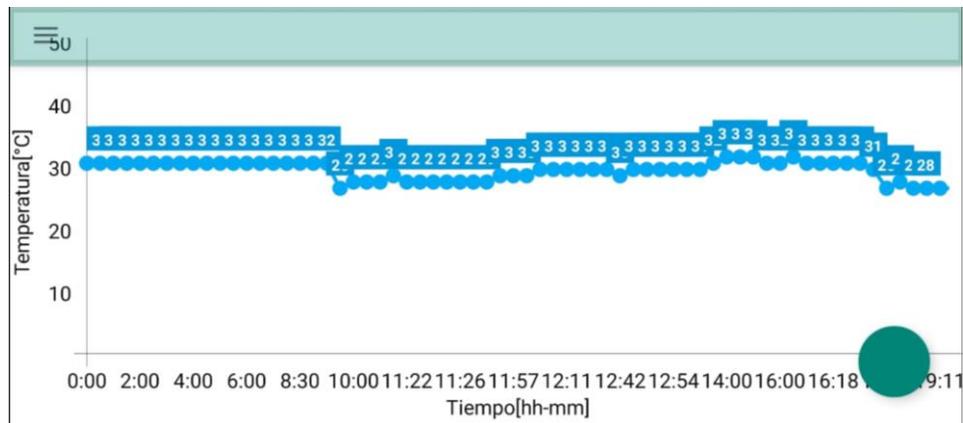
Figura 3.1: Histórico de la bomba de agua



Autores: Víctor Yanchapaxi y Bryan Taboada

Como se puede apreciar la bomba de agua se enciende y apaga varias veces al día para mantener la humedad en las raíces de las plantas, y la forma de comprobar que esto suceda es a través de los sensores de humedad. En la Figura 3.2 se presentan las gráficas del sensor de temperatura.

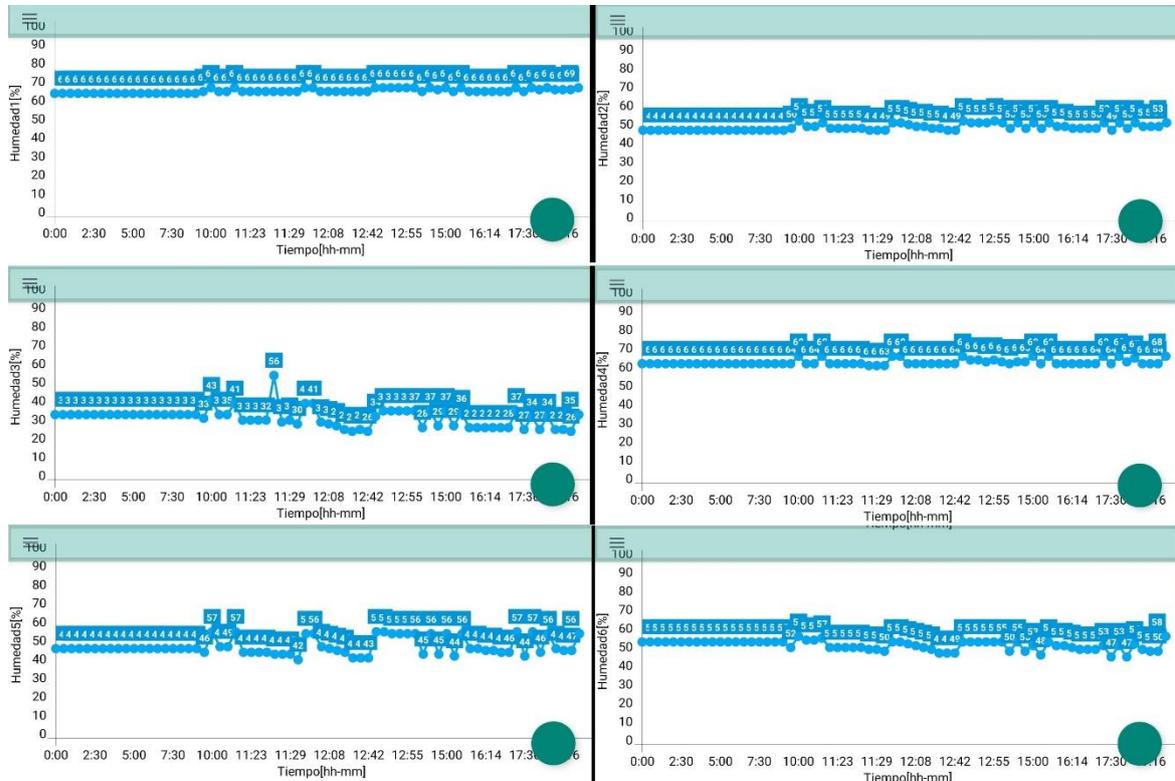
Figura 3.2: Gráfica del sensor de temperatura



Autores: Víctor Yanchapaxi y Bryan Taboada

En la Figura 3.3 se muestran las gráficas de los sensores de humedad colocados en cada tubo del prototipo aeropónico.

Figura 3.3: Gráficas de los sensores de humedad



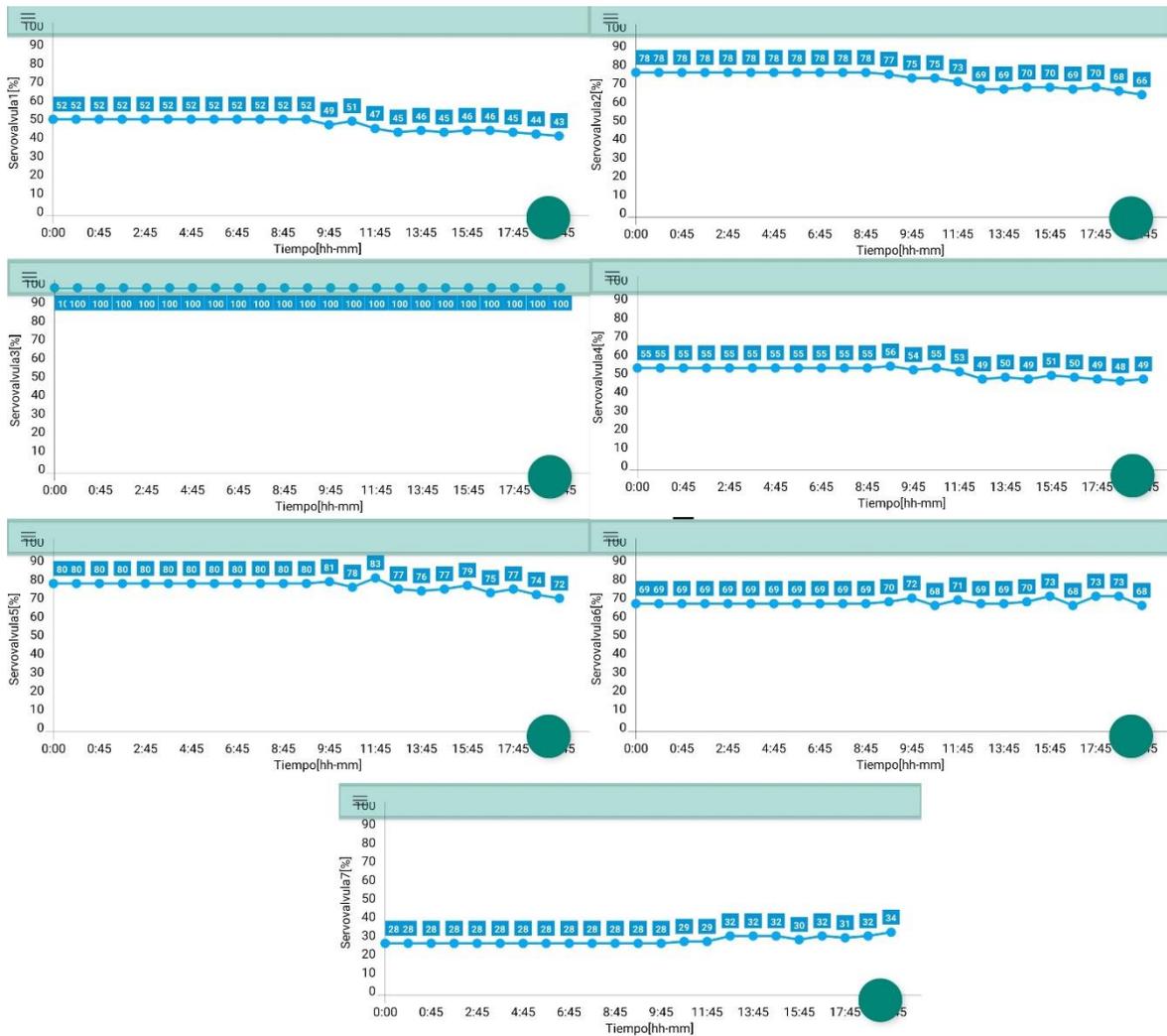
Autores: Víctor Yanchapaxi y Bryan Taboada

En la Figura 3.3 se puede ver que todos los valores de humedad tienden a mantenerse constantes durante el día, lo que indica que el riego se está dando de forma correcta; también se puede apreciar que la variación de la humedad en los sensores es similar a como varía la temperatura.

Si bien es cierto las gráficas de la Figura 3.3 no son completamente lineales, esto es debido a que se hizo un mal uso de los sensores de humedad durante el proceso de implementación del prototipo.

En la Figura 3.4 se puede observar el accionamiento de las servoválvulas para regular el caudal en cada tubo:

Figura 3.4: Gráficas de las servoválvulas



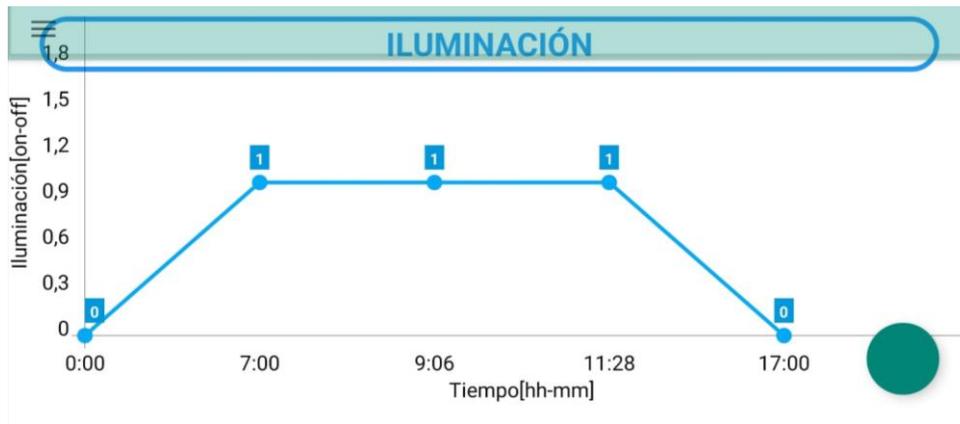
Autores: Víctor Yanchapaxi y Bryan Taboada

La servoválvula del tubo 3 permanece abierta al 100%, ya que el sensor de humedad 3 tiene el valor más bajo de humedad, entonces el sistema prioriza el caudal de la solución nutritiva hacia ese tubo y reduce el caudal para las demás porque tienen un valor de humedad aceptable mayor al 50%. La servoválvula 7 devuelve al tanque parte de la solución nutritiva que no pasa por las demás servoválvulas.

3.2 Prueba del sistema de iluminación

Las gráficas que se pueden analizar en este sistema son del encendido y apagado de los focos (Figura 3.5) y de los valores de luminosidad medidos por el sensor de luz (Figura 3.6).

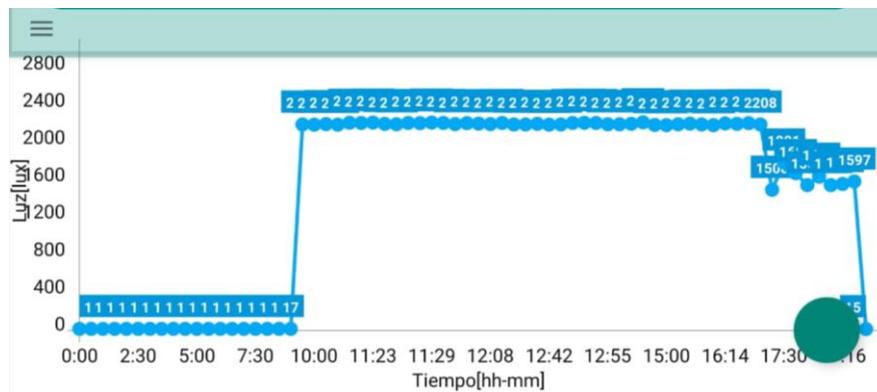
Figura 3.5: Histórico de encendido de los focos



Autores: Víctor Yanchapaxi y Bryan Taboada

Los focos tienen un horario establecido para su encendido desde las 7 am hasta las 5 pm, que se puede ver en la Figura 3.5. Pero, para saber si estos focos se encendieron en el horario mencionado se observa la Figura 3.6 que corresponde a la gráfica del sensor de luz.

Figura 3.6: Gráfica del sensor de luz



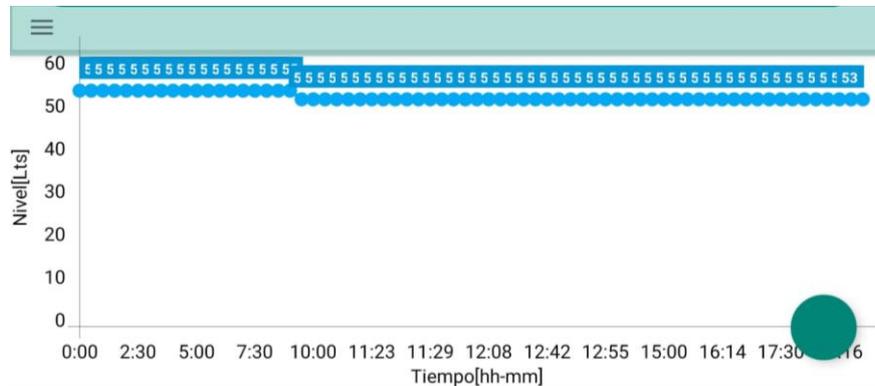
Autores: Víctor Yanchapaxi y Bryan Taboada

Como se ha podido observar, el valor que mide el sensor de luz es de 2200 luxes durante el intervalo de encendido de los focos y menores a 2200 luxes cuando los focos se apagan, con esto se comprueba que el sistema de iluminación se ha estado encendiendo y apagando durante el día según lo requerido. También se puede ver que a partir de las 5 pm el sensor de luz detecta una cantidad de luz cercana a 1500 luxes, esto es debido a la iluminación del lugar donde se encuentra ubicado el prototipo aerónico.

3.3 Prueba del sistema de mezcla de nutrientes

En este sistema se puede comprobar la correcta adición de los nutrientes y del regulador de pH, que debe hacerse en función del nivel del tanque y del día de la semana.

Figura 3.7: Gráfica del sensor de nivel del tanque

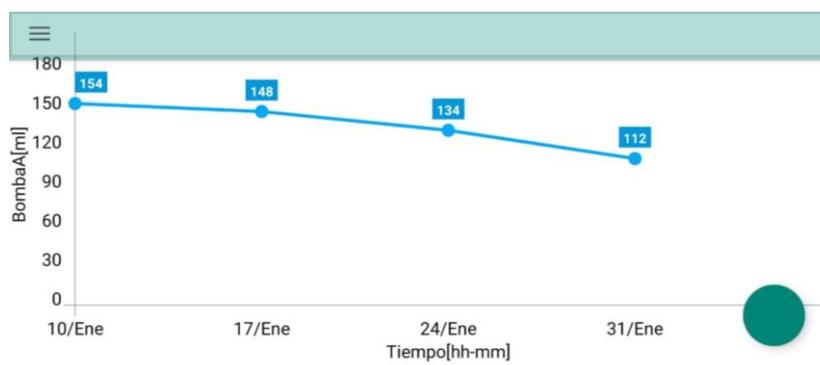


Autores: Víctor Yanchapaxi y Bryan Taboada

En la Figura 3.7 se observan los datos obtenidos del sensor de nivel del tanque durante el día, donde es posible ver que el nivel en litros de esta se va reduciendo debido a la absorción de la solución nutritiva en las raíces de las plantas, aunque también se debe a la evaporación.

Se tomaron datos de la adición de los nutrientes durante el mes de enero, por lo que en la Figura 3.8 se indica la fecha en que se agregó la solución A y la cantidad de esta en mililitros, de esta forma se puede apreciar que la bomba A se activó durante los lunes del mes de enero y con la adición de solución A cada vez menor.

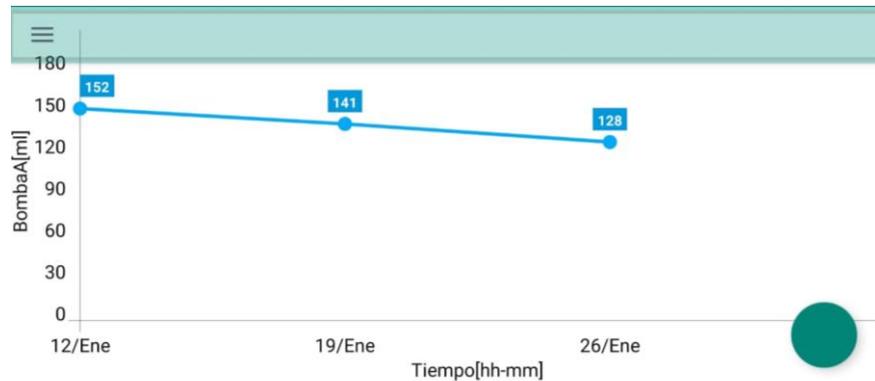
Figura 3.8: Histórico de la bomba de la solución A



Autores: Víctor Yanchapaxi y Bryan Taboada

En el caso de la bomba para la solución B que se muestra en la Figura 3.9, se observa que se ha agregado dicha solución los martes del mes de enero y en las cantidades que se muestran.

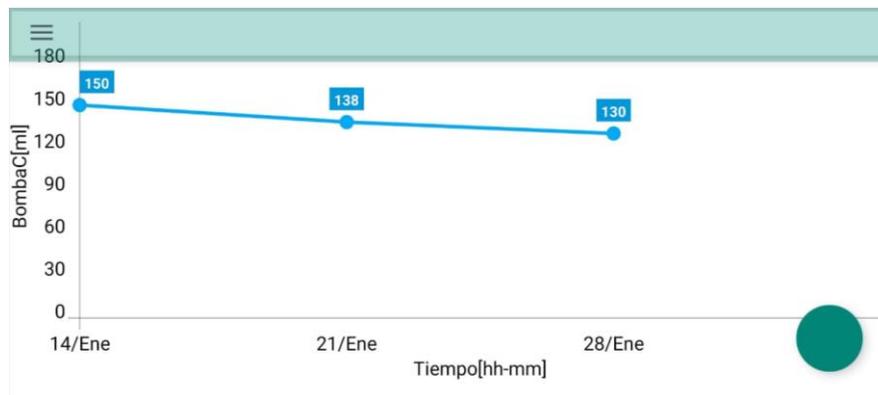
Figura 3.9: Histórico de la bomba de la solución B



Autores: Víctor Yanchapaxi y Bryan Taboada

En la Figura 3.10 es posible ver que la bomba de la solución C se ha activado los miércoles de manera similar a las bombas de la Figura 3.8 y Figura 3.9 en cantidades diferentes.

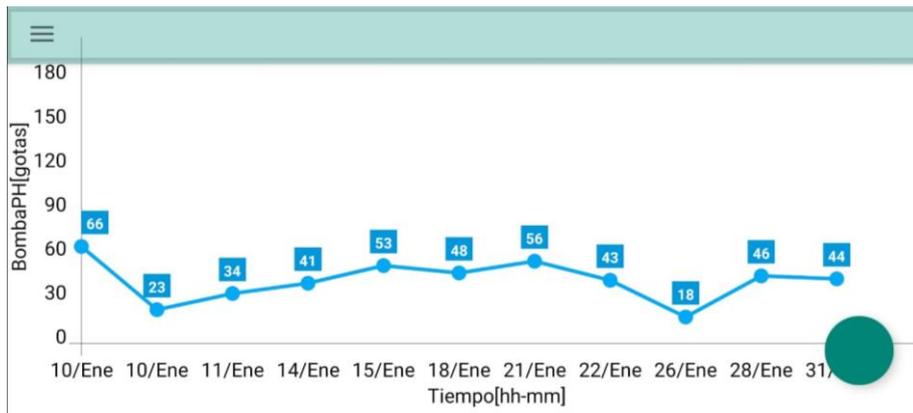
Figura 3.10: Histórico de la bomba de la solución C



Autores: Víctor Yanchapaxi y Bryan Taboada

Respecto a la bomba encargada de suministrar el regulador de pH a la mezcla de nutrientes, se observa en la Figura 3.11 que se activa en varios días del mes de enero.

Figura 3.11: Histórico de la bomba del regulador de pH

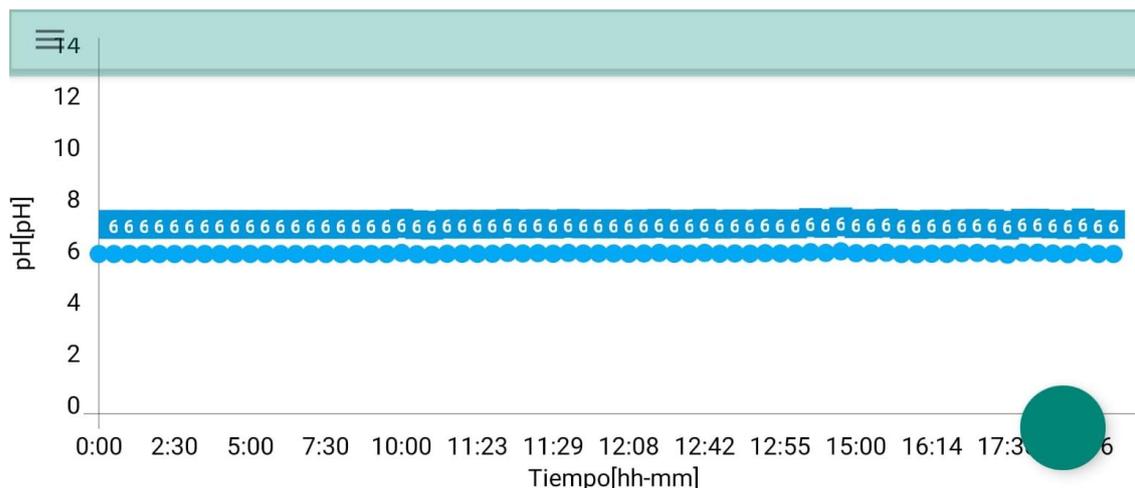


Autores: Víctor Yanchapaxi y Bryan Taboada

De esta prueba se puede determinar que todos los actuadores del sistema de mezcla de nutrientes funcionan como se esperaba, las bombas de las soluciones A, B y C adicionaron los nutrientes respectivos en los días esperados y cantidad cada vez menor debido a que es proporcional a la cantidad de agua en el tanque, mientras que la bomba de regulador de pH se activó en varios días del mes de enero, ya que este proceso control se realiza todos los días.

A continuación, se muestra un gráfico con los datos del sensor de pH en la Figura 3.12 donde se observa que el nivel de acidez se mantiene constante.

Figura 3.12: Gráfica del sensor de pH



Autores: Víctor Yanchapaxi y Bryan Taboada

CONCLUSIONES

Para finalizar, el proyecto técnico cumple con los objetivos planteados, ya que permite monitorear el cultivo de forma remota observando en la aplicación desarrollada en Android Studio los datos entregados por los sensores de humedad, luminosidad, nivel y temperatura, así como también permite medir y controlar de forma automática el pH del agua mezclada con solución nutritiva.

El prototipo aeropónico armado en este proyecto permite sembrar hasta 156 plantas como: lechuga, albahaca, frutilla, cilantro, y en general hortalizas de raíz corta, en un espacio no mayor a 3 m² y de poca iluminación natural, con un sistema que funciona de forma autónoma y que no requiere de la intervención humana más que para supervisión.

Se concluye que en la mayoría de los cultivos aeropónicos automatizados se enfocan en las variables de humedad, temperatura y pH, pero no toman en cuenta la importancia de la cantidad de luz que debe recibir la planta según la estación del año, así como la proporción exacta y de forma automática de nutrientes, ni del debido control de caudal del agua al momento de la irrigación hacia las plántulas.

Las pruebas realizadas en los sensores y actuadores de los sistemas de este prototipo dieron resultados positivos, ya que funcionaron como se esperaban, hubo algunos factores que dificultaron las pruebas como el desgaste de los sensores de humedad, o como la incidencia de luz ajena a la del sistema de iluminación del prototipo.

La aplicación de monitoreo remoto desarrollada en Android Studio es capaz de acceder a los datos de los sensores y actuadores del sistema aeropónico de manera rápida y desde cualquier lugar del mundo, siempre y cuando se tenga acceso a Internet.

La Raspberry pi fue suficientemente basta para llevar a cabo todas las operaciones requeridas en esta implementación e inclusive se cree que se pueden agregar funciones más complejas al sistema actual como por ejemplo Machine Learning para un control aún más eficiente.

Cloud Firestore de Firebase es un buen servicio de base de datos en tiempo real, porque no se ha podido apreciar una latencia significativa en la subida ni en la descarga de la información, ofrecen un gran número de operaciones de lectura y escritura de forma gratuita y un sistema de autenticación que mejora la privacidad de los datos que se almacenan, y que han sido de gran utilidad para este proyecto.

RECOMENDACIONES

Se recomienda adquirir un sensor de conductividad del agua para tener un control más preciso de los nutrientes de la solución nutritiva

Se recomienda utilizar bombas no sumergibles para bombear el líquido de los envases de las soluciones, porque de esa manera las bombas no están en contacto con estas sustancias todo el tiempo, evitando así contaminación de estas.

En el caso de que no sea posible que las plantas del sistema aeropónico reciban luz natural, se recomienda utilizar luminaria LED especializada para cultivos en interior, ya que las plantas crecerán de mejor manera.

Es recomendable colocar anillos de PVC, de diámetro similar al de las canastillas, en los agujeros del tubo PVC de siembra, porque esto evitará que se escapen gotas de solución nutritiva hacia afuera del prototipo

Evitar el uso de sustratos como cáscara de coco si no se tiene un control de la conductividad de la solución nutritiva, porque está va a cambiar enormemente la cantidad de nutrientes en la misma.

Se recomienda usar agua de lluvia para llenar el tanque del sistema aeropónico, porque su pH es más neutro que el agua potable, y además no contiene sustancias adicionales como el cloro que sí podría tener el agua potable.

Se recomienda utilizar tubería PVC dependiendo de los requerimientos de siembra, por ejemplo, para la siembra de lechuga o de plantas que crecen en hoja es conveniente usar tubería PVC de 4 pulgadas, y para plantas más grandes como fresas, tomate Cherry, etc. Es mejor utilizar tubos de 6 pulgadas ya que el espacio permitirá una mejor oxigenación a sus raíces.

BIBLIOGRAFÍA

- 5MP OV5647. (s/f). Recuperado el 13 de febrero de 2022, de Uctronics website: <https://www.uctronics.com/5mp-ov5647-download-full-datasheet-pdf>
- Apaza, D. B. (2020). *Implementación de un sistema de monitoreo remoto y de control para la mayor eficiencia en un cultivo de tomates cherry por hidroponía en sustrato preparado*. Universidad Nacional de San Agustín de Arequipa, Arequipa.
- Aquino, F. M. (2020). *Diseño y caracterización de sistema aeropónico automatizado para el cultivo de Stevia rebaudiana*.
- Barrera, F. (2020). *Consumo responsable: motivaciones de compra de alimentos orgánicos en hombres y mujeres de 25 a 34 años en la ciudad de Guayaquil derivado del efecto Covid-19*. Universidad Casa Grande, Guayaquil.
- Bedoya, M. A. (2019). *Detallar el desarrollo de un cultivo de cilantro (coriandrum sativum) en un sistema aeropónico automatizado*. Universidad Nacional Abierta y a Distancia (UNAD), Ibagué.
- Bomba sumergible SOBO 4500lph. (s/f). Recuperado el 13 de febrero de 2022, de Mascotamoda.com website: <https://www.mascotamoda.com/acuarios-y-estanques/bombas-sumergibles/bomba-sumergible-sobo-4500lph-detail>
- Cámara Raspberry Pi v2 - 8 Megapixels. (s/f). Recuperado el 23 de febrero de 2022, de Bricogeek.com website: <https://tienda.bricogeek.com/accesorios-raspberry-pi/822-camara-raspberry-pi-v2-8-megapixels.html>
- Castañares, J. (2020). *El ABC de la hidroponía*. Agencia de Extensión Luján, Buenos Aires.
- Castro, R. S., & Zambrano, K. (2020). Comportamiento del mercado local y estrategias de ventas locales frente a la pandemia por Covid-19 en la ciudad de Manta. *Reicomunicar*, 21–37.
- Cómo usar el sensor DHT11. (s/f). Recuperado el 23 de febrero de 2022, de Smelpro.com website: <https://smelpro.com/blog/sensor-dht11/>

- ELECFreaks. (s/f). *Octopus Soil Moisture Sensor Brick*. Recuperado de https://www.electronicoscaldas.com/datasheet/OBSoil-01_ElecFreaks.pdf
- Electrostore. (s/f). MINI BOMBA DE AGUA SUMERGIBLE 5V. Recuperado el 23 de febrero de 2022, de [Grupoelectrostore.com](https://grupoelectrostore.com) website: <https://grupoelectrostore.com/shop/motores/bombas-para-agua/mini-bomba-de-agua-sumergible-5v/>
- Embajadores, E. (s/f). Resistencia LDR 5 x 4 mm - Sensor de Luz. Recuperado el 23 de febrero de 2022, de [Electrónica Embajadores](https://www.electronicaembajadores.com) website: <https://www.electronicaembajadores.com/es/Productos/Detalle/SSLDR34/sensores/sensores-de-luminosidad-color/resistencia-ldr-5-x-4-mm-sensor-de-luz>
- HC-SR04 sensor ultrasonido con atmega-1. (2016, diciembre 26). Recuperado el 23 de febrero de 2022, de [MakerElectronico](https://www.makerelectronico.com) website: <https://www.makerelectronico.com/hc-sr04-sensor-ultrasonido-atmega/hc-sr04-sensor-ultrasonido-con-atmega-1/>
- Hernández, M. T. (2016). *Manual para la producción de cultivos aeropónicos*. Universidad Técnica Nacional, Alajuela.
- Hidroponía vertical. (2019, noviembre 28). Recuperado el 12 de febrero de 2022, de [Arkiplus](https://www.arkiplus.com) website: <https://www.arkiplus.com/hidroponia-vertical/>
- Hunter C. Mitchell, Smith G. Richard, Schipanski E. Meagan, Atwood W. Lesley & Mortensen A. David. (2017). Agriculture in 2050: Recalibrating Targets for Sustainable Intensification. *BioScience*, 386–391.
- López, P. J. (2020). *Proyecto de construcción de un invernadero para cultivo aeropónico con sistemas de control automatizados*. Universidad de Jaén, Linares.
- Micro servo miniatura SG90 1.5kg. (2018, diciembre 25). Recuperado el 23 de febrero de 2022, de [MakerElectronico](https://www.makerelectronico.com) website: <https://www.makerelectronico.com/producto/micro-servo-miniatura-sg90-1-5kg/>

- Mouser Electronics. (s/f). *DHT11 Humidity & Temperature Sensor*. Recuperado de <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>
- Navarro, D., Hernández, J., Posada, J., Nepote, A., & Ramirez, M. (2020). *Huertos urbanos ¿fenómeno pasajero o nuevo estilo de vida ante la pandemia de la COVID-19?* Centro de Investigación Científica de Yucatán, Yucatán.
- Pérez, A., & Romero, R. R. (2019). *Construcción y automatización remota de un invernadero hidropónico tipo N.F.T.* Universidad Autónoma de la ciudad de México, México.
- Pérez, J. M. (2019). *Comparación de producción de tres variedades de lechuga (Lactuca sativa L.) bajo sistema aeropónico vertical automatizado en cantón Daule – Guayas.* Universidad Agraria del Ecuador, Daule.
- PH-4502C Kit sensor de pH económico. (s/f). Recuperado el 23 de febrero de 2022, de Geek Factory website: <https://www.geekfactory.mx/tienda/sensores/ph-4502c-kit-sensor-de-ph-economico/>
- Raspberry Pi Ltd. (s/f). Raspberry pi 4. Recuperado el 12 de febrero de 2022, de Raspberry Pi website: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>
- Rueda, P. E. P., & Pineda, C. A. B. (2012). *Diseño y evaluación de un sistema de producción tipo aeropónico protegido, empleando la especie (Phaseolus vulgaris).* Universidad Francisco de Paula, Santander.
- Salazar, S. S. P. (2021). Aeroponía: cultivo sin tierra. + *Ciencia*, 26–27.
- Scaturro, G. N. (2019). *Evaluación de dos sistemas de producción de lechuga en hidroponía y un cultivo tradicional bajo cubierta.* Universidad Nacional de Luján, Buenos Aires.
- Sensor de nivel de lluvia / agua rojo T1592. (s/f). Recuperado el 23 de febrero de 2022, de ElectroCrea website: <https://electrocrea.com/products/sensor-de-lluvia-agua>
- SparkFun Electronics. (s/f). *Ultrasonic Ranging Module HC - SR04.*

STMicroelectronics. (2021). *ULN2001, ULN2002 ULN2003, ULN2004*.

United Nations. (2015). *World Population Prospects: The 2015 Revision, Key Findings and Advance Tables*. Department of Economic and Social Affairs.

United Nations. (2016). *The World's cities in 2016*. Department of Economic and Social Affairs.

Valles, J. (2020). *Sistema de mezcla de soluciones nutritivas y ácido para un cultivo hidropónico basado en la técnica de película de nutrientes (NFT)*. Universidad Técnica del Norte, Ibarra.

Viera, D. N. (2019). *Sistema de monitoreo remoto para mediciones de variables ambientales en cultivos hidropónicos con tecnología inalámbrica zigbee*. Universidad Nacional de Piura, Perú.

ANEXOS

ANEXO I: Ensamble del primer prototipo



ANEXO II: Elaboración del sistema de tuberías



ANEXO III: Sistema de tuberías del primer prototipo



ANEXO IV: Colocación del primer prototipo



ANEXO V: Primer prototipo



ANEXO VI: Fabricación de tubos para la siembra



ANEXO VII: Armado del segundo prototipo



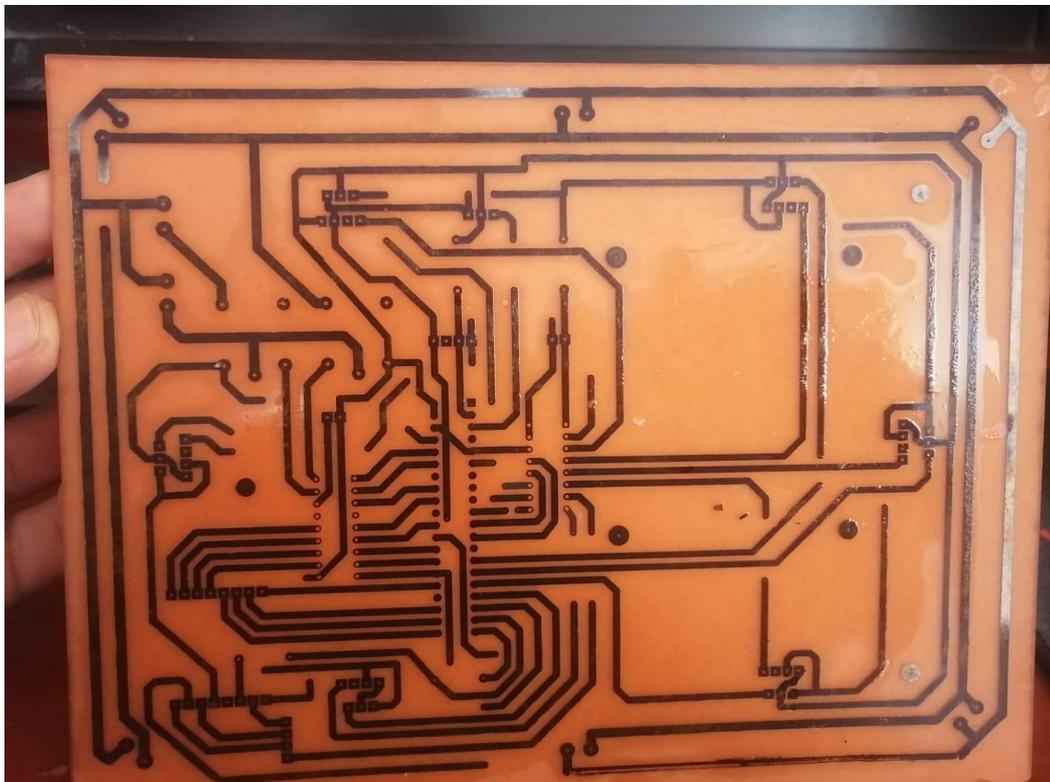
ANEXO VIII: Segundo prototipo



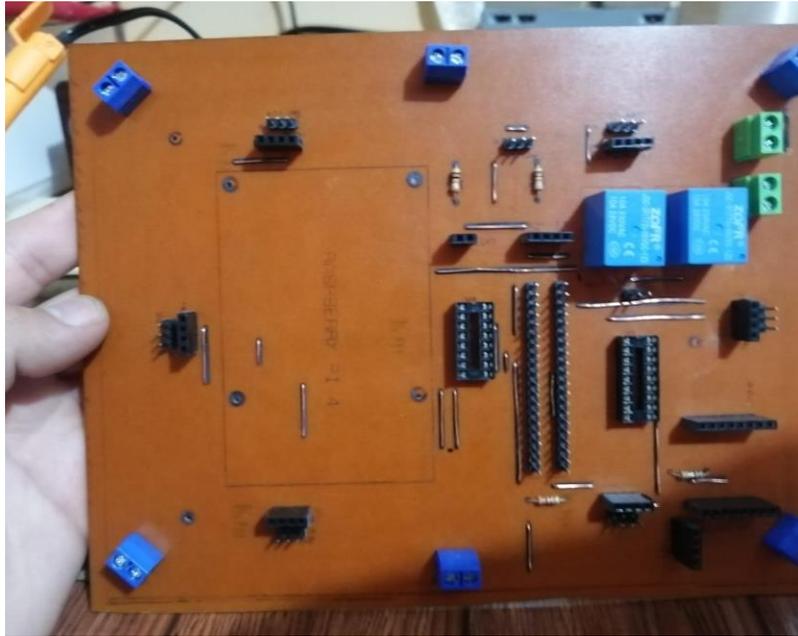
ANEXO IX: Elaboración de PCB por método de planchado



ANEXO X: Resultado del método de planchado



ANEXO XI: Soldadura de componentes en el PCB



ANEXO XII: Prototipo final



ANEXO XIII: Librerías importadas

```
import RPi.GPIO as gpio
import time
import adafruit_dht
import firebase_admin
import statistics
import subprocess
from firebase_admin import credentials, firestore
from board import D4
from gpiozero import MCP3008
from flask import Flask, render_template, Response
from camera import VideoCamera
from threading import Thread
```

ANEXO XIV: Configuración de pines

```
#Sensores
sensorh1 = MCP3008(channel=7)
sensorh2 = MCP3008(channel=5)
sensorh3 = MCP3008(channel=4)
sensorh4 = MCP3008(channel=3)
sensorh5 = MCP3008(channel=2)
sensorh6 = MCP3008(channel=1)
sensorluz = MCP3008(channel=0)
sensorph = MCP3008(channel=6)

pinDHT11=D4

PINtrigger=19
PINecho=26

PINbomba=18
PINfocos=15
PINservos=[13,6,5,0,17,14,27]
PINniveles=[21,20,16,12]
PINbombas=[24,25,7,1]

#Configuracion
gpio.setmode(gpio.BCM)
gpio.setup(PINbomba,gpio.OUT)
gpio.setup(PINfocos,gpio.OUT)
gpio.setup(PINservos,gpio.OUT)
gpio.setup(PINbombas,gpio.OUT)
gpio.setup(PINtrigger,gpio.OUT)

gpio.setup(PINEcho,gpio.IN)
gpio.setup(PINniveles,gpio.IN)

dht11 = adafruit_dht.DHT11(pinDHT11, use_pulseio=False)

credenciales=credentials.Certificate('/home/pi/Aeroponico/Credenciales.json')
firebase_admin.initialize_app(credenciales)
db=firestore.client()
```

ANEXO XV: Funciones utilizadas en el programa

```
#Funciones
def readDHT11():
    t=None
    h=None
    while t==None or h==None:
        time.sleep(1)
        try:
            h=dht11.humidity
            t=dht11.temperature
        except RuntimeError as error:
            error=""
    return h, t

def servovalvula(pin, porcentaje):
    duty=-0.00001*porcentaje + 0.0015
    for i in range(100):
        gpio.output(pin, 1)
        time.sleep(duty)
        gpio.output(pin, 0)
        time.sleep(0.02 - duty)

def AjustarValvulas(humedades,servos,pines,fecha,s,servotxt):
    minh=200
    servos[6]=0
    for i in range(6):
        if humedades[i]<minh and s[i]>100:
            minh=humedades[i]
    for i in range(6):
        servos[i]=round(-100*(humedades[i]-minh)/(100-minh)+100)
        if servos[i]<0:
            servos[i]=0
        if servos[i]>100:
            servos[i]=100
        if s[i]<=100:
            servos[i]=s[i]
        servovalvula(pines[i],servos[i])
        keyHora=getStrHora(fecha[3],fecha[4])
        addFirestore(servotxt[i],"Hoy",keyHora,servos[i])
        actuadoresvalue[i]=servos[i]
        print(servotxt[i]," = ", servos[i], "%")
        servos[6]+=servos[i]
    servos[6]=round(100-servos[6]/6)
    if s[6]<=100:
        servos[6]=s[6]
    servovalvula(pines[6],servos[6])
    addFirestore(servotxt[6],"Hoy",keyHora,servos[6])
    actuadoresvalue[6]=servos[6]
    print(servotxt[6]," = ", servos[6], "%")
    return servos
```

ANEXO XVI: Funciones utilizadas en el programa

```
def readNivel():
    mintanque=34
    maxtanque=26
    gpio.output(PINtrigger,False)
    time.sleep(0.2)
    gpio.output(PINtrigger,True)
    time.sleep(0.00001)
    gpio.output(PINtrigger,False)
    while gpio.input(PINEcho)==0:
        Tinicio=time.time()
    while gpio.input(PINEcho)==1:
        Tfin=time.time()
    distanciacm=round((Tfin-Tinicio)*17150,2)
    litros=30*(distanciacm-mintanque)/(maxtanque-mintanque)+30 #min=30lts y max=60lts
    return litros

def getStrHora(hh,mm):
    if mm<10:
        strHora=str(hh)+":0"+str(mm)
    else:
        strHora=str(hh)+":"+str(mm)
    return strHora

def ponerSolucion(pin,t,f,m1,conta):
    bombastxt=["BombaA","BombaB","BombaC","BombaPH"]
    mesestxt=["Ene","Feb","Mar","Abr","May","Jun","Jul","Ago","Sep","Oct","Nov","Dic"]
    keyFecha=str(f[2])+"/"+mesestxt[f[1]-1]+":"+str(conta)
    addFirestore("Solucion",bombastxt[pin],keyFecha,m1)
    print(bombastxt[pin]," = ",m1)
    gpio.output(PINbombas[pin], 1)
    time.sleep(t)
    gpio.output(PINbombas[pin], 0)

def ping(host):
    ret = subprocess.call(['ping', '-c', '1', '-W', '1', host],
        stdout=open('/dev/null', 'w'),
        stderr=open('/dev/null', 'w'))
    return ret == 0

def addFirestore(coll,doc,key,value):
    if ping("8.8.8.8"):
        db.collection(coll).document(doc).set({key:value},merge=True)
    else:
        print("No se ha podido agregar los datos, revise conexion a Internet")
```

ANEXO XVII: Programación de la bomba del tanque

```
while(1):
    fecha=time.localtime()
    #Programa de la bomba principal
    if cont2==0: #Si no se esta mezclando la solucion
        if b>1:
            if fecha[4]>=45 and fecha[4]<60 and nivel>17:
                bomba=1
            else:
                bomba=0
        else:
            bomba=b
    if auxbomba!=bomba or auxb==1:
        auxb=0
        cont=0
        auxbomba=bomba
        actuadoresvalue[7]=bomba
        gpio.output(PINbomba,bomba)
        keyHora=getStrHora(fecha[3],fecha[4])
        addFirestore("Bomba","Hoy",keyHora,bomba)
        print("Bomba = ",bomba)

#Programacion para regular el caudal a traves de las servovalvulas
if auxs==1:
    auxs=0
    for i in range(len(servos)):
        if s[i]<=100 and s[i]!=servos[i]:
            servovalvula(PINservos[i],s[i])
            keyHora=getStrHora(fecha[3],fecha[4])
            addFirestore(actuadorestxt[i],"Hoy",keyHora,s[i])
            servos[i]=s[i]
            actuadoresvalue[i]=servos[i]
            print(actuadorestxt[i]," = ", s[i], "%")
    elif bomba==1:
        if cont==0:
            servos=AjustarValvulas(sensoresvalue,servos,PINservos,fecha,s,actuadorestxt)
        elif cont==15:
            cont=0
        cont+=1
```

ANEXO XVIII: Programación de los focos

```
#Programacion de la iluminacion
if f>1:
    if fecha[3]>=7 and fecha[3]<17:
        focos=1
    else:
        focos=0
else:
    focos=f
if auxfocos!=focos or auxf==1:
    auxfocos=focos
    auxf=0
    actuadoresvalue[8]=focos
    gpio.output(PINfocos,focos)
    keyHora=getStrHora(fecha[3],fecha[4])
    addFirestore("Iluminacion","Hoy",keyHora,focos)
    print("Focos = ", focos)
```

ANEXO XIX: Programación de la mezcla de nutrientes

```
#Programacion de la mezcla de soluciones
if fecha[6]==0 and fecha[3]==0 and fecha[4]==0 and solucion==0: #Solucion A todos los 1
    tiempo=30*nivel/89 # 3[ml]*nivel[lts]*10[s]/89[ml]
    ponerSolucion(0,tiempo,fecha,round(3*nivel,2),conta)
    solucion=1
    mezclar=1
    conta+=1
if fecha[6]==2 and fecha[3]==0 and fecha[4]==0 and solucion==0:
    tiempo=30*nivel/100 # 3[ml]*nivel[lts]*10[s]/100[ml]
    ponerSolucion(1,tiempo,fecha,round(3*nivel,2),conta)
    solucion=1
    mezclar=1
    conta+=1
if fecha[6]==4 and fecha[3]==0 and fecha[4]==0 and solucion==0:
    tiempo=30*nivel/98 # 3[ml]*nivel[lts]*10[s]/98[ml]
    ponerSolucion(2,tiempo,fecha,round(3*nivel,2),conta)
    solucion=1
    mezclar=1
    conta+=1
if ph>6.5:
    tiempo=nivel*(ph-6.5)/50 #nivel[lts]*deltaph[s]/(5[gotas]*10)
    ponerSolucion(3,tiempo,fecha,nivel)
    mezclar=1
if fecha[3]==0 and fecha[4]==0 and ph>5 and alcalino==0: #Con 10 gotas se baja 1 de ph
    tiempo=nivel*(ph-5)/4
    ponerSolucion(3,tiempo,fecha,round(tiempo*5),conta)
    alcalino=1
    mezclar=1
if conta==60:
    conta=0
if fecha[6]==1 or fecha[6]==3 or fecha[6]==5:
    solucion=0
if mezclar==1:
    if cont2==0:
        bomba=1
        keyHora=getStrHora(fecha[3],fecha[4])
        addFirestore("Bomba","Hoy",keyHora,bomba)
        print("Mezclando...")
        print("Bomba = ", bomba)
        for i in range(7):
            servovalvula(PINservos[i],0)
            addFirestore(actuadorestxt[i],"Hoy",keyHora,0)
            print(actuadorestxt[i]," = 0%")
            servovalvula(PINservos[6],100)
            addFirestore("Servovalvula7","Hoy",keyHora,100)
            print("Servovalvula7 = 100%")
        cont2+=1
if cont2==10:
    bomba=0
    cont2=0
    mezclar=0
    alcalino=0
    keyHora=getStrHora(fecha[3],fecha[4])
    addFirestore("Bomba","Hoy",keyHora,bomba)
    print("Bomba = ", bomba)
    print("Se ha terminado la mezcla")
```

ANEXO XX: Programación para enviar datos a Firestore

```
#Cargar datos de los sensores en Firebase
if((fecha[4]==0 or fecha[4]==30) and enviado==0) or con==1:
    h1=100-round(sensorh1.value*100)
    h2=100-round(sensorh2.value*100)
    h3=100-round(sensorh3.value*100)
    h4=100-round(sensorh4.value*100)
    h5=100-round(sensorh5.value*100)
    h6=100-round(sensorh6.value*100)
    ph=round(readPh(),2)
    nivel=round(readNivel(),2)
    luz=round(2200*(sensorluz.value*100)/94)
    humedad, temperatura = readDHT11()
    sensoresvalue=[h1,h2,h3,h4,h5,h6,temperatura,humedad,ph,luz,nivel]
    #
    for i in range(6):
        htxt="h"+str(i+1)
        addFirestore("Raspberry","Alarmas",htxt,sensoresvalue[i])

    addFirestore("Raspberry","Alarmas","nivel",nivel)
    #Nivel de los recipientes de las soluciones
    for i in range(len(niveles)):
        #if gpio.input(PINniveles[i])!=niveles[i]:
            if gpio.input(PINniveles[i])==1:
                niveles[i]=1
            else:
                niveles[i]=1
        nivelestxt="n"+str(i+1)
        keyHora=getStrHora(fecha[3],fecha[4])
        addFirestore("Solucion","Alarmas",nivelestxt,niveles[i])
        addFirestore("Raspberry","Alarmas",nivelestxt,niveles[i])
        print("Nivel", (i+1), " = ",niveles[i])
    enviado=1
    keyHora=getStrHora(fecha[3],fecha[4])
    for i in range(len(sensorstxt)):
        addFirestore(sensorstxt[i],"Hoy",keyHora,sensoresvalue[i])
    print("Se subieron los datos a Firestore")
    if con==1:
        con=0
        addFirestore("Android","Configuracion","con",0)
        addFirestore("Raspberry","Alarmas","con",1)

    #Imprime datos en la consola
    print(time.asctime(fecha))
    for i in range(len(sensoresvalue)):
        print(sensorstxt[i]," = ",sensoresvalue[i])

if fecha[4]==2 or fecha[4]==32:
    enviado=0
```