



# Vivero Hidropónico 4.0: Implementación de tecnologías de automatización y digitalización.

## Hydroponic Nursery 4.0: Implementation of Technologies for Automation and Digitalization.

Presentación: 04/07/2025

Aprobación: 08/10/2025

Publicación: 24/10/2025

### **Agustín Aicardi**

Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Grupo de Investigación en Enseñanza de la Ingeniería (GIEDI), Argentina  
[aaicardi@frsf.utn.edu.ar](mailto:aaicardi@frsf.utn.edu.ar)

### **Matías Orué**

Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Grupo de Investigación en Enseñanza de la Ingeniería (GIEDI), Argentina  
[morue@frsf.utn.edu.ar](mailto:morue@frsf.utn.edu.ar)

### **María Laura Caliusco**

Universidad Tecnológica Nacional, Facultad Regional Santa Fe, Grupo de Investigación en Enseñanza de la Ingeniería (GIEDI), Argentina  
[mcaliusco@frsf.utn.edu.ar](mailto:mcaliusco@frsf.utn.edu.ar)

### **Resumen**

Este trabajo presenta el diseño e implementación de un sistema hidropónico inteligente para el cultivo de plantas medicinales en un entorno controlado. Enmarcado en los principios de la Industria 4.0, el proyecto busca automatizar y digitalizar el monitoreo de parámetros ambientales claves, garantizando un crecimiento óptimo de las plantas. Se empleó la metodología en cascada, lo que permitió un desarrollo estructurado y secuencial, desde la selección de componentes hasta la integración final del sistema. El sistema resultante regula automáticamente factores como temperatura, humedad y pH, al tiempo que registra datos en tiempo real mediante herramientas digitales. Este enfoque innovador aporta precisión, eficiencia y sostenibilidad al cultivo hidropónico de plantas medicinales, tradicionalmente realizado de forma manual. Además, el modelo desarrollado es adaptable a distintos entornos y tipos de cultivo, representando una contribución relevante a la modernización de la agricultura mediante tecnologías emergentes.

**Palabras claves:** Hidroponía, Industria 4.0, automatización, digitalización.

## Abstract

This work presents the design and implementation of an intelligent hydroponic system for cultivating medicinal plants in a controlled environment. Aligned with the principles of Industry 4.0, the project aims to automate and digitalize the monitoring of key environmental parameters to ensure optimal plant growth. A waterfall methodology was employed, allowing for a structured and sequential development—from component selection to the final integration of the system. The resulting setup automatically regulates factors such as temperature, humidity, and pH, while recording real-time data through digital tools. This innovative approach enhances precision, efficiency, and sustainability in the hydroponic cultivation of medicinal plants, a process traditionally carried out manually. Furthermore, the developed model is adaptable to various environments and crop types, representing a significant contribution to the modernization of agriculture through emerging technologies.

**Keywords:** Hydroponic, Industry 4.0, automation, digitalization.

## Introducción

La hidroponía constituye un método de cultivo en el cual las plantas se desarrollan en soluciones nutritivas sin la utilización de suelo. En este sistema, las raíces se encuentran en contacto directo con una disolución acuosa que contiene todos los nutrientes esenciales para el crecimiento vegetal (Resh, 2022). Para el soporte físico de las plantas se emplean sustratos inertes como arena, perlita, lana de roca o grava, cuya función es exclusivamente estructural. Entre las principales ventajas de la hidroponía se destaca el uso eficiente del recurso hídrico, ya que consume hasta un 90 % menos agua en comparación con la agricultura convencional (Resh, 2022). Asimismo, el sistema permite un control preciso de variables como la concentración de nutrientes, el pH y la exposición lumínica, optimizando tanto el rendimiento como la calidad del cultivo (Savvas & Gruda, 2018). La eliminación del suelo contribuye también a reducir la incidencia de plagas y enfermedades, disminuyendo la necesidad de pesticidas y mejorando la inocuidad de los alimentos producidos. Las plantas cultivadas mediante hidroponía suelen presentar un crecimiento acelerado y una mayor productividad, atribuibles a la constante disponibilidad de nutrientes en concentraciones óptimas (Urrestarazu, 2013). Adicionalmente, estos sistemas pueden implementarse en configuraciones verticales, favoreciendo el cultivo en espacios urbanos reducidos y promoviendo la agricultura local.

Por un lado, el cultivo hidropónico de plantas medicinales ha cobrado relevancia en los últimos años debido a sus ventajas en términos de sostenibilidad, eficiencia y control de condiciones de crecimiento (Jensen, 1997). En el contexto argentino, el cultivo de plantas medicinales aún se realiza mayoritariamente mediante prácticas tradicionales, sin una adopción sistemática de herramientas tecnológicas. Estudios realizados en comunidades semi-rurales, como en la estepa patagónica, evidencian que la mayoría de estas plantas se cultivan en huertas domésticas. Esta modalidad limita las posibilidades de escalabilidad, estandarización y control de calidad de los cultivos medicinales.

Por otro lado, la mayoría de los sistemas actuales aún dependen de operaciones manuales que limitan la capacidad de monitoreo continuo y la optimización precisa de parámetros ambientales. En respuesta a esta limitación, este trabajo presenta el desarrollo de un sistema automatizado de bajo costo orientado al cultivo hidropónico de plantas medicinales, con

el objetivo de mejorar la calidad del cultivo mediante la integración de tecnologías digitales y principios de la Industria 4.0. La incorporación de tecnologías automatizadas ha transformado radicalmente las prácticas agrícolas contemporáneas, mejorando la eficiencia operativa, el rendimiento productivo y la sostenibilidad. La integración de sensores, actuadores, microcontroladores y algoritmos de control permite una gestión precisa de variables críticas como el riego, el suministro de nutrientes, la temperatura y la iluminación, reduciendo el uso excesivo de insumos y el impacto ambiental.

Actualmente se dispone de un sistema hidropónico funcional (Mattei et al., 2023), pero su operación manual impide una regulación precisa y en tiempo real de variables críticas como temperatura, iluminación y pH. La automatización del proceso busca no sólo optimizar estas condiciones, sino también generar un entorno inteligente capaz de recolectar, analizar y utilizar datos para mejorar el rendimiento del cultivo. Para ello, se propone la selección e integración de sensores económicos, sistemas de adquisición y procesamiento de datos, así como algoritmos de control clásico que permitan la regulación autónoma del entorno de crecimiento.

La digitalización del sistema, junto con su modelización, constituye un paso esencial hacia la implementación de un gemelo digital: una réplica virtual del entorno físico que permitirá simular, analizar y optimizar el sistema antes de realizar modificaciones en el entorno real. Este enfoque facilita la toma de decisiones basadas en datos, permite un análisis continuo del rendimiento, y brinda un entorno seguro para la capacitación técnica. Además, al capturar y almacenar información en un servidor mediante redes de comunicación, se habilita el monitoreo remoto en tiempo real, fortaleciendo la trazabilidad y la capacidad de respuesta ante variaciones ambientales.

En este contexto, el presente trabajo describe la arquitectura del sistema automatizado propuesto, incluyendo la selección de componentes, la lógica de control y la infraestructura de digitalización de datos. Asimismo, se analiza su potencial para generar conocimiento aplicado en el ámbito de la agronomía digital, y se discute su impacto en la mejora de prácticas agrícolas sustentables y en la innovación tecnológica para el cultivo de plantas medicinales.

## Arquitectura del sistema automatizado propuesto

El diseño de la arquitectura del sistema hidropónico se ha realizado teniendo en cuenta la información recolectada en el mapeo sistemático de la literatura (Aicardi, 2022) y el sistema hidropónico autocontenido desarrollado por Mattei et al. (2023). La arquitectura del sistema hidropónico automatizado se exhibe en la Figura 1. Del lado izquierdo se aprecia el gabinete y, del lado derecho, se muestran las principales partes del sistema compuesto por sensores y actuadores que se comunican con la unidad de control implementada en Arduino. El Arduino además de recolectar los datos desde el sistema físico para el control, les da un formato específico y los envía hacia una placa Raspberry Pi donde se aloja un servidor. Este servidor se encargará de darle vida al gemelo digital.

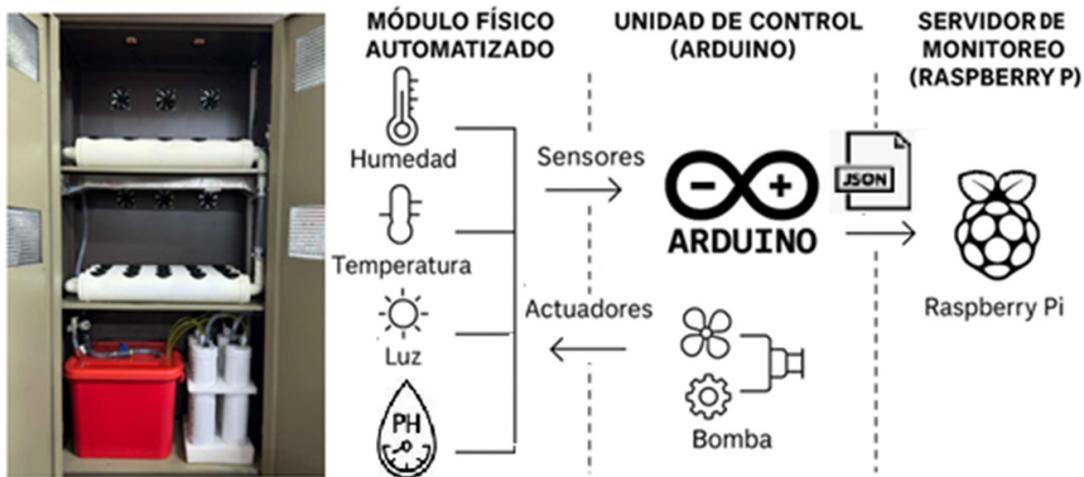


Figura 1 –Arquitectura del sistema hidropónico automatizado.

La placa Arduino seleccionada fue la Mega 2560 por su capacidad para gestionar múltiples dispositivos y sensores simultáneamente, gracias a su microcontrolador ATmega2560, 54 pines digitales, 16 entradas analógicas y amplia memoria (256 KB de flash, 8 KB de SRAM y 4 KB de EEPROM). Sus múltiples puertos UART y conexiones físicas permiten una integración compleja y escalable. Destaca por ser de código abierto, lo que fomenta la innovación y colaboración, y por su costo accesible frente a alternativas similares. La programación se realizó mediante el IDE de Arduino, reconocido por su facilidad de uso y compatibilidad con diversos sistemas operativos, facilitando el desarrollo en C/C++. La biblioteca Serial de Arduino facilita de manera sencilla y eficiente la comunicación en serie para interactuar con computadores y otros dispositivos.

En el contexto de la comunicación entre el Arduino y la Raspberry Pi, el formato JSON resulta una excelente opción para estructurar y transmitir datos de manera eficiente. JSON (JavaScript Object Notation) es un formato de texto ligero para el intercambio de datos. Es fácil de leer y escribir para los humanos y fácil de analizar y generar para las máquinas, lo que lo convierte en un estándar popular para la transmisión de datos entre aplicaciones debido a su simplicidad y flexibilidad. Este formato se basa en pares de clave-valor, donde cada clave está asociada con un valor específico. Las claves son cadenas de texto y deben estar entre comillas dobles. Por ejemplo, el valor de pH medido podría ser representado como:

“pH\_value”:8,

En primer lugar, en el Arduino, los datos que necesitan ser enviados se organizan en un formato de texto definido, como lo es JSON. La biblioteca ArduinoJson facilita la creación de estos objetos JSON, permitiendo estructurar la información en un formato legible y organizado. Una vez que los datos están estructurados, se envían a través del puerto serie del Arduino. La Raspberry Pi, que está escuchando en ese puerto, recibe estos datos y puede procesarlos. Para garantizar un sistema automatizado y eficiente, es clave obtener datos precisos sobre pH, electroconductividad, temperatura y humedad, ya que estos servirán para regular y mantener condiciones óptimas.

La solución debe ser de bajo costo, seleccionando componentes con buena relación costo-beneficio. La combinación de precisión técnica y viabilidad económica asegura un sistema funcional, sostenible y replicable en diversos contextos. A continuación, se detallan los

sensores y actuadores elegidos, destacando su principio de funcionamiento y características técnicas.

## Sensores

### Medición de pH

Se basa en principios electroquímicos. La sonda está compuesta por un electrodo de vidrio, que es sensible a los iones de hidrógeno presentes en la solución, y un electrodo de referencia con un potencial constante. Al sumergirse en una solución, se genera una diferencia de potencial entre ambos electrodos, la cual es proporcional a la concentración de iones de hidrógeno, es decir, al pH de la muestra. Esta diferencia de potencial varía analógicamente, luego se la acondiciona, amplifica y filtra para ser enviada al sistema de microcontrolador (Tabla 1 (A)).

### Medición de electroconductividad

Mediante dos electrodos que, al ser sumergidos en una solución y recibir una diferencia de potencial, permiten la circulación de una corriente eléctrica. La intensidad de esta corriente es proporcional a la concentración de sólidos disueltos: a mayor concentración, mayor conductividad. Esta señal eléctrica es registrada por el sensor y refleja el nivel de sólidos disueltos en la solución. El sensor convierte esta señal en una lectura de TDS (Total de Sólidos Disueltos), expresada en partes por millón (ppm). Para garantizar precisión y estabilidad, el dispositivo incorpora un circuito de calibración y acondicionamiento de señal (Tabla 1 (B)).

### Medición de Caudal

Los caudalímetros son dispositivos encargados de medir la cantidad de soluciones que se agregan al sistema, como nutrientes o reguladores de pH. Se basa en una turbina interna y un sensor de efecto Hall. Cuando el líquido fluye, hace girar la turbina cuyas paletas contienen imanes; estos imanes generan cambios en el campo magnético, que el sensor detecta y traduce en pulsos eléctricos. Cada uno de estos pulsos es proporcional al flujo del líquido, y su frecuencia permite estimar con precisión la velocidad del flujo. Luego solo resta convertir el número de pulsos en  $\text{m}^3/\text{s}$ , lo que permite medir con exactitud el volumen de solución que atraviesa el caudalímetro (Tabla 1 (C)).

### Medición de temperatura y humedad ambiente

El módulo contiene un termistor para detectar la temperatura, cuya resistencia cambia según el calor ambiente, y un polímero que responde a la humedad absorbiendo o liberando agua, lo que altera su conductividad eléctrica. También incorpora un circuito de acondicionamiento de señal y un microcontrolador interno que interpreta y convierte las señales analógicas en datos digitales de temperatura y humedad, que se envían en un protocolo serie (Tabla 1 (D)).

### Medición de temperatura de la solución

Se basa en un termistor de coeficiente negativo (NTC), encapsulado en un paquete resistente al agua y protegido contra interferencias, lo que lo hace adecuado para entornos exigentes. Además, incorpora un convertidor analógico a digital (ADC) y un microcontrolador interno que procesan la señal generada por la variación de resistencia. Este sistema convierte la señal analógica en una lectura digital de temperatura. El sensor transmite esta información mediante el protocolo OneWire (Tabla 1 (E)).

<b>(A) pH: Sensor PH-4502C + con la sonda E201-C</b>	
<b>Tensión de alimentación:</b> 5 V DC ± 0,2 V <b>Corriente:</b> 5 - 10 mA <b>Rango de pH:</b> 0 - 14 <b>Rango de temperatura:</b> 0 - 80°C <b>Tiempo de respuesta y estabilización:</b> 5s / 60s	
<b>(B) Electroconductividad: Sensor KS0429 Keyestudio TDS Meter V1.0</b>	
<b>Tensión de alimentación:</b> 3,3 - 5,5 V DC <b>Corriente:</b> 3 - 6 mA <b>Rango de TDS:</b> 0 - 1000ppm <b>Precisión:</b> ± 10% F.S. (a 25 °C) <b>Tensión de salida:</b> 0 - 2.3V	
<b>(C) Caudal: caudalímetro YF-S401</b>	
<b>Tensión de alimentación:</b> 5 - 18 V DC <b>Corriente:</b> 10mA (a 5V) <b>Rango de TDS:</b> 0,3 - 6 l/m <b>Precisión:</b> 1,75MPa <b>Temperatura de funcionamiento:</b> -25 - 80 °C	
<b>(D) Temperatura y humedad ambiente: Módulo sensor DHT22</b>	
<b>Tensión de alimentación:</b> 3,3 - 6 V DC <b>Corriente:</b> 2,5mA <b>Rango de temperatura:</b> -40 - 125 °C <b>Precisión temperatura:</b> -40 - 125 °C <b>Precisión humedad:</b> 0,1%	
<b>(E) Temperatura de solución: sensor DS18B20</b>	
<b>Tensión de alimentación:</b> 3 - 5,5 V DC <b>Rango de temperatura:</b> -55 - 125 °C <b>Error (-10°C a 85°C):</b> ±0,5°C <b>Error (-55°C a 125°C):</b> ±0,5°C °C <b>Resolución:</b> 12 bit	

Tabla 1: Sensores utilizados

## Actuadores

### Bomba principal

Es la encargada de movilizar la solución nutritiva hacia las plantas, tomando el fluido desde el tanque principal y elevándolo hacia el estante superior, donde es introducido a los tubos donde se encuentran sumergidas las raíces de las plantas. La solución, al ingresar al primer tubo, recorre los demás por rebalse y efecto de la gravedad, concluyendo su recorrido al retornar al tanque principal. La acción de este actuador estará comandada por períodos de tiempo de encendido y apagado, que son configurados en el código (Tabla 2 (A)).

## Bombas secundarias

Encargadas de movilizar las distintas soluciones concentradas almacenadas en los tanques secundarios hacia el principal. Dichos tanques contienen los nutrientes y los reguladores de pH que permiten acondicionar la solución que luego llegará hacia las raíces de las plantas. La acción de las bombas secundarias rige de forma conjunta con los caudalímetros. En este caso, mediante el código, se calcula la cantidad en m<sup>3</sup>/s de las soluciones (Tabla 2 (B)).

## LEDs

Cumplen una función fundamental en el crecimiento de las plantas. Emiten luz en una longitud de onda violeta del espectro visible, esencial para la fotosíntesis de las plantas. Esta luz estimula el crecimiento vegetativo (fotosíntesis), la formación de clorofila y otros pigmentos necesarios para un desarrollo saludable (Tabla 2 (C)).

## Ventiladores

Son los encargados de la refrigeración del sistema, asegurando mantener la temperatura óptima para el desarrollo de las plantas. Seis de ellos se encuentran en la pared posterior del gabinete, dirigidos de manera tal que extraigan el aire de este, permitiendo que, por presión negativa, ingrese aire a menor temperatura desde el exterior por las ventilaciones ubicadas en las puertas (Tabla 2 (D)).

<b>(A) Bomba principal: RS-932</b>	
Tensión de alimentación: 220 V AC Corriente: 0.082A Potencia: 18W Capacidad: 820 l/hora Altura máxima: 1,8 m	
<b>(B) Bombas secundarias</b>	
Tensión de alimentación: 3 - 5 V DC Corriente: 0.3A Potencia: 1.5W Capacidad: 120 l/hora Altura máxima: 0,4 m	
<b>(C) LEDs ultravioletas: Full Spectrum 50w</b>	
Tensión de alimentación: 220 V AC Corriente: 0.23A Potencia: 50W Longitud de onda: 380 - 840 nm Flujo luminoso: 5000 lm	
<b>(D) Ventiladores o Coolers de extracción de aire</b>	
Tensión de alimentación: 12 V DC Dimensiones: 70 x 70 mm	

Tabla 2: Actuadores utilizados

## Diseño e implementación del sistema de control automatizado

Tanto la automatización como la recolección de datos se basan en un único código programado en la memoria interna de la placa Arduino. A continuación, se presentan las estructuras de algunas de ellas que se definieron siguiendo el proceso de la Figura 2.



Figura 2 – Ciclo de inicialización, configuración y control del Arduino.

## Estructura de funciones de control

La función encargada del control del pH es “pHControl”, la cual se divide en dos grandes secciones, la de medición y la de actuación. En la etapa de medición se obtiene el valor del sensor pH y se evalúa, comparándolo con los valores máximos y mínimos y definiendo si se encuentra por encima o por debajo de los límites establecidos. Esto a su vez modifica un contador, en el cual, si el valor de pH es alto, se incrementa en una unidad, si es bajo se decrementa y si está dentro de los límites, se establece en cero.

En la etapa de actuación se evalúa este contador. Para este ejemplo se eligió, como valor límite del contador, tres. Al superar las tres mediciones de pH por fuera del rango permitido, se actúa, según corresponda, sobre las bombas que agregan una solución acida o alcalina para regularlo.

### Control de temperatura PID

El desarrollo de las plantas está condicionado por múltiples factores ambientales y biológicos, como la luz, el agua, los nutrientes y las interacciones bióticas, cuya combinación afecta directamente su fisiología y productividad. Por ejemplo, deficiencias nutricionales o de luz pueden limitar la fotosíntesis y el crecimiento vegetal. Entre estos factores, la temperatura es particularmente crítica, ya que su desviación del rango óptimo puede inducir estrés térmico, disminuyendo la fotosíntesis, alterando la transpiración y la actividad enzimática, y afectando negativamente el desarrollo y la producción de metabolitos secundarios, esenciales en plantas medicinales por sus propiedades terapéuticas. Además, el estrés térmico puede provocar estrés oxidativo, reduciendo la síntesis de compuestos bioactivos

como alcaloides, flavonoides y aceites esenciales, comprometiendo así la eficacia medicinal de estas plantas (Srivastava et al., 2023; Castroverde & Dina, 2021).

Para mantener la temperatura dentro de los rangos configurados se eligió trabajar con un control Proporcional-Integral-Derivativo (PID). El control PID se compone de tres términos principales que actúan en conjunto para ajustar la salida del sistema:

1. Control Proporcional (P): El término proporcional ajusta la salida proporcionalmente al error actual entre la variable medida y el setpoint. Esta acción es rápida y proporciona una respuesta inmediata a las perturbaciones. Sin embargo, puede resultar en un error de estado estacionario si se utiliza solo.
2. Control Integral (I): El término integral integra el error acumulado a lo largo del tiempo y ajusta la salida en función de esta integral. Esto asegura que los errores de estado estacionario, que son errores constantes a largo plazo, sean corregidos gradualmente.
3. Control Derivativo (D): El término derivativo predice la tasa de cambio del error y ajusta la salida para contrarrestar las fluctuaciones rápidas en la variable medida. Esto mejora la estabilidad del sistema al prever y ajustar la respuesta antes de que ocurran cambios significativos.

En el sistema sobre el cual se trabaja, la iluminación mediante LEDs actúa como una fuente de calor, donde parte de la energía eléctrica utilizada por estos se transforma en calor debido a la ineficiencia en la conversión de energía eléctrica a luz. Aunque el encendido de los LEDs es independiente de la temperatura, el calor generado por estos es un factor que debe ser considerado para mantener las condiciones óptimas en el ambiente de cultivo de plantas medicinales. Teniendo esto en cuenta, se realizó una toma de datos para analizar la respuesta de este parámetro en función del tiempo. Los resultados de las mediciones realizadas cada 30 segundos en un lapso de 2 horas permitieron trazar la curva que se presenta en la Figura 3.

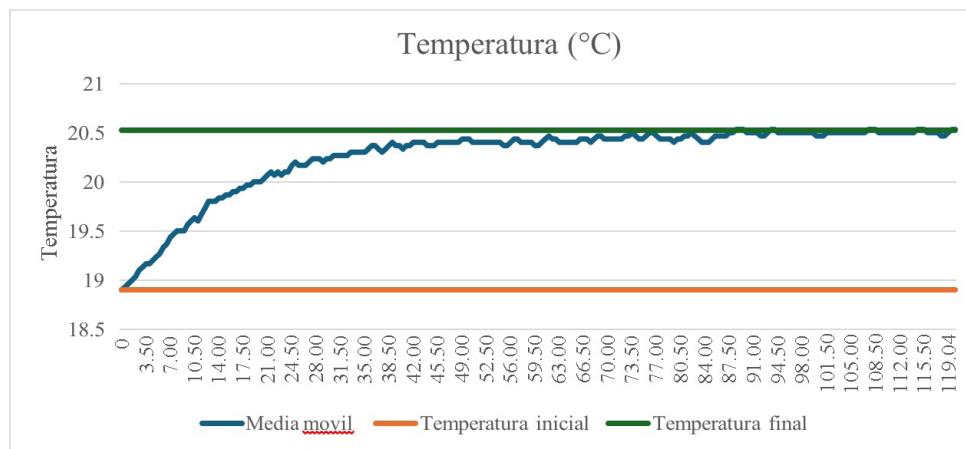


Figura 3 - Evolución de la temperatura en un período de dos horas.

Para la obtención de los parámetros necesarios en el control PID se utilizó el primer método de Ziegler-Nichols (Ogata, 2010). Este método es particularmente útil para sistemas con retardo y proporciona reglas empíricas para calcular los parámetros del controlador

PID basándose en la respuesta del sistema a un escalón. En el caso de estudio, la acción de encender las luces LED se decidió representar como un escalón.

En la Figura 4, la curva azul que muestra la evolución de la temperatura a lo largo del tiempo fue aproximada por la curva amarilla para facilitar el análisis. Esta curva tiene una forma de S, caracterizada por dos parámetros clave: el tiempo de retardo ( $L$ ) y la constante de tiempo ( $T$ ). Para determinar estos parámetros de manera gráfica, se dibuja una línea tangente (roja) en el punto de inflexión de la curva en S, y se identifican las intersecciones (rosa) de esta tangente con las líneas que representan los valores iniciales (naranja) y finales (verde).

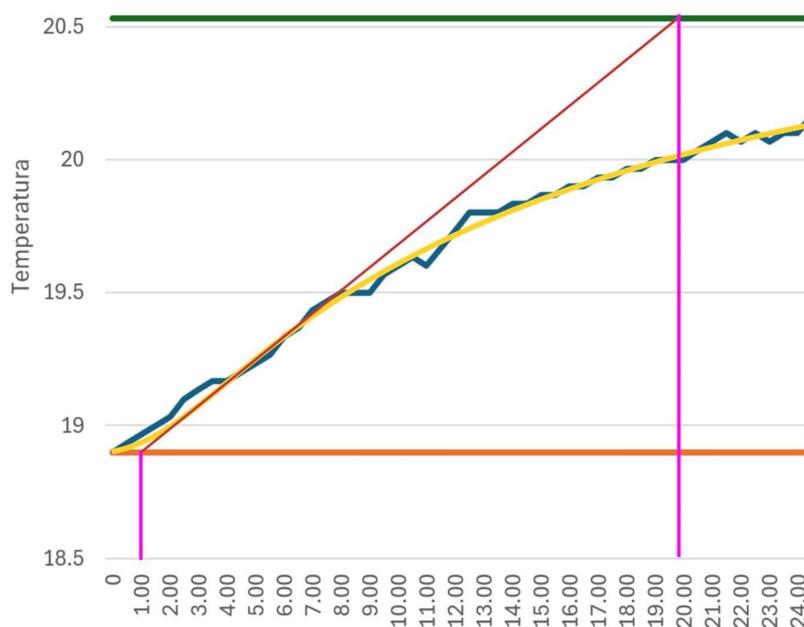


Figura 4 - Aproximación de la curva y obtención de parámetros.

Mediante el método gráfico presentado se obtuvieron los parámetros:

- Retraso ( $L$ ): El retraso o tiempo muerto es el tiempo que tarda la respuesta del sistema en empezar a cambiar después de que se aplique un escalón en la entrada.

$$L=60s$$

- Constante de Tiempo ( $T$ ): la constante de tiempo obtenida gráficamente es igual a 20 minutos, equivalente en segundos a:

$$T=1200s$$

Teniendo en cuenta que la temperatura en este caso no comienza desde cero, se debe tener en cuenta un factor más para el cálculo final.

- Ganancia del Sistema ( $K$ ): La ganancia se calcula como la relación entre el cambio en la salida final y el cambio en la entrada (escalón). Considerando que la temperatura inicial del sistema fue 18.9°C y la final 20.6°C, se calcula:

$$K = \frac{\Delta T}{\Delta u} = \frac{T_f - T_i}{\Delta u}$$

$$K = \frac{20.6^{\circ}\text{C} - 18.9^{\circ}\text{C}}{1} = 1.7^{\circ}\text{C}$$

Con estos valores ya obtenidos se procedió a calcular los parámetros PID, utilizando las fórmulas de Ziegler-Nichols. Dichos parámetros son utilizados posteriormente por el código Arduino para realizar el control.

Proporcional	Integral	Derivado
$K_p = \frac{1.2 \times T}{L \times K} = \frac{1.2 \times 1200}{60 \times 1.7}$ <b><math>K_p = 14.12</math></b>	$K_i = \frac{K_p}{\tau_i} = \frac{K_p}{2 \times L} = \frac{14.12}{2 \times 60}$ <b><math>K_i = 0.12</math></b>	$K_d = K_p \times \tau_d = K_p \times \left( \frac{1}{2} \times L \right)$ $K_d = 14.12 \times \left( \frac{1}{2} \times 60 \right)$ <b><math>K_d = 423.6</math></b>

Tabla 3: Cálculo de parámetros PID.

- En el control PID se utiliza una ecuación que incluye términos proporcionales, integrales y derivativos que se nutren de los parámetros anteriormente calculados para obtener la salida del controlador. La ecuación continua del controlador PID es (1):

$$c_{(t)} = K_p \times e_{(t)} + K_i \times \int_0^t e_{(t)} dt + K_d * \frac{de_{(t)}}{dt} \quad (1)$$

Donde:

- $c_{(t)}$  es la salida del controlador.
- $e_{(t)}$  es el error en el tiempo t (la diferencia entre el valor deseado y el valor medido).

Para implementar este controlador en un sistema digital, es necesario discretizar la ecuación, dado que las computadoras y microcontroladores trabajan con señales discretas en el tiempo. Esto se hace sustituyendo las derivadas e integrales por sus aproximaciones discretas.

El término integral acumula el error a lo largo del tiempo, y en su versión discreta se representa como una suma acumulativa de errores (2):

$$\text{Integral}_{(t)} = \text{Integral}_{(t-\Delta t)} + e_{(t)} \times \Delta t \quad (2)$$

Donde:

- $\Delta t$  es el intervalo de tiempo entre cada actualización

En el caso término derivativo, este mide la tasa de cambio del error. En su forma discreta, se aproxima usando la diferencia finita entre el error actual y el error anterior (3):

$$\text{Derivada}_{(t)} = \frac{e(t) - e(t - \Delta t)}{\Delta t} \quad (3)$$

La salida del controlador  $c(t)$  se usa para actuar sobre un sistema con el fin de minimizar el error. Idealmente, la salida debería aplicarse a un actuador analógico que pueda responder de manera continua al valor de  $c_{(t)}$ . Sin embargo, en este caso particular, se dispone de ventiladores que únicamente pueden encenderse o apagarse, sin capacidad para variar su velocidad de manera continua. Además, se descartó el uso de un control PWM (Modulación por Ancho de Pulso) para simular una respuesta analógica ya que los ventiladores son encendidos a través de relés mecánicos, para los cuales implicaría un gran desgaste.

Esto significa que, aunque el control PID genera una salida continua, en la práctica se toma una decisión digital: si la salida supera un umbral, se encienden los ventiladores; de lo contrario, se mantienen apagados. En ambientes más hostiles, con temperaturas más extremas o cambiantes, podría justificarse el uso de un control analógico para tener una regulación más precisa. Pero, dado que en la ubicación donde se encuentra el sistema, la temperatura se encuentra dentro de valores razonables para el cultivo de plantas, y el encendido de las luces por largos períodos solo aumenta la temperatura en 2 grados Celsius, un control analógico resulta innecesario.

## Verificación de encendido de luces

Dentro de los componentes del sistema se menciona el sensor LDR (Light Dependent Resistor), cuya función es la verificación del encendido efectivo de luces artificiales para desarrollo vegetal. Una demostración de las luces LED encendidas se muestra en la Figura 5.

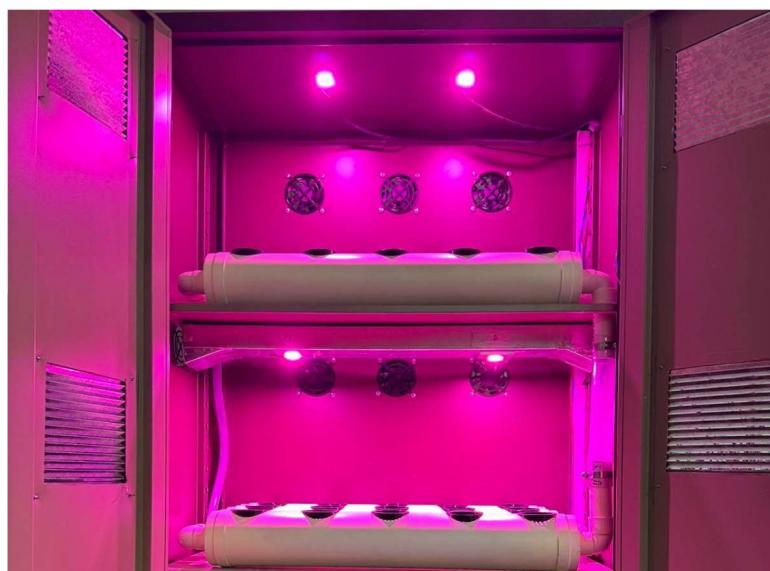


Figura 5 - Demostración de luces LED.

Se sigue el siguiente procedimiento: se verifican 3 escenarios: iluminación total, parcial y oscuridad. Se conecta al pin analógico de placa Arduino, cuyo rango de mapeo es 0-1023 valores digitales y una frecuencia de muestreo de 1 segundo. Los valores fueron tomados cada un segundo y permitieron componer la gráfica que se muestra en la Figura 6.

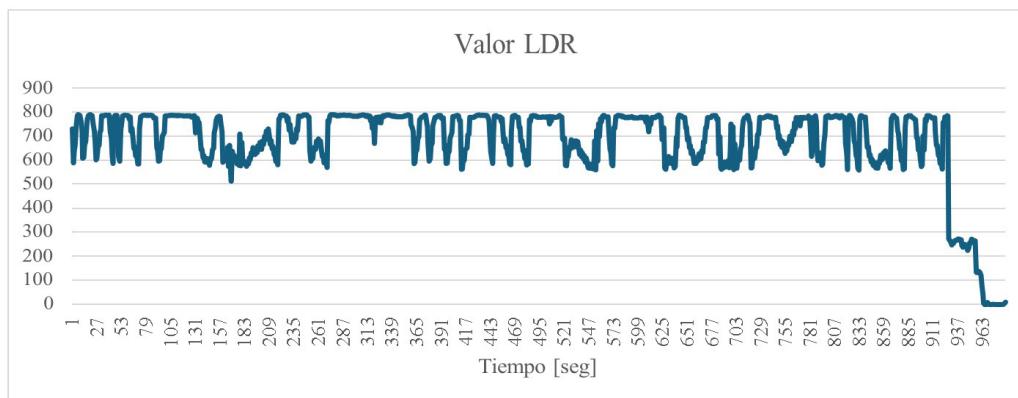


Figura 6 - Valores obtenidos por el LDR frente a distintos estímulos.

Al someter al sensor a estas condiciones se encontraron los siguientes resultados:

- Iluminación total: las luces de ambos estantes se encuentran encendidas. El LDR responde con valores comprendidos entre 450 y 800.
- Oscuridad: las luces de ambos estantes se encuentran apagadas. El LDR responde con valores notablemente inferiores a 100, ubicándose, luego de unos segundos de estabilización, todos ellos por debajo de 10.
- Iluminación parcial: las luces de uno de los estantes se encuentran encendidas, mientras que las del otro se encuentran apagadas. Evaluados los dos posibles escenarios que cumplen esta condición, los valores obtenidos del LDR se sitúan entre 100 y 300, dependiendo del estante que se encuentre encendido.

Con los valores obtenidos experimentalmente se procedió a configurar la etapa de control en el código que ejecuta el Arduino. Además, la instalación del sensor permite que a futuro se puedan usar los valores sensados por el mismo para otros propósitos, como tener un valor certero del nivel de irradiancia que reciben las plantas, luego de calibrar el sensor y adecuar y filtrar la señal.

### Digitalización del sistema y creación de su gemelo digital

La Industria 4.0 promueve la creación de entornos inteligentes capaces de integrar datos, procesos y objetos técnicos en tiempo real, favoreciendo la toma de decisiones automatizada y la mejora continua. Un componente clave es el modelo de referencia RAMI 4.0 (DIN SPEC 91345, 2016), que organiza los activos a lo largo de su ciclo de vida y los representa mediante el Asset Administration Shell (AAS) (Plattform Industrie 4.0 & ECLASS, 2021). El AAS es la implementación del "gemelo digital" para la Industria 4.0. Un gemelo digital se puede definir como un modelo virtual de un sistema físico complejo, habilitado a través de datos y simuladores, para predicciones en tiempo real, optimización, monitoreo, control y toma de decisiones mejoradas (Rasheed et al., 2020).

Para la digitalización se creó una estructura del AAS mediante la utilización del software AASX Package Explorer (n.d.). Este programa permitió organizar y estructurar toda la información relevante en un formato estándar, siguiendo las directrices de la Industria 4.0.

Para ello, se especificó la jerarquía dentro del Sistema Hidropónico Autocontenido, utilizando submodelos y propiedades (Figura 7). Los submodelos son componentes estructurados dentro del AAS que agrupan propiedades relacionadas para describir aspectos específicos del sistema. Cada submodelo tiene un propósito claro y está diseñado para organizar la información de manera lógica y comprensible. Por otro lado, las propiedades dentro de cada submodelo son elementos individuales que contienen datos específicos, como medidas, estados, o configuraciones. Estas propiedades están descritas en detalle, incluyendo su valor, tipo de dato, y una descripción en varios idiomas. Las mismas permiten almacenar y comunicar información precisa sobre las características y el funcionamiento de los componentes del sistema.

Además, se sumó un último bloque con las “Concept Descriptions” del AAS. Éstas son definiciones detalladas que describen el significado y el contexto de las propiedades u otros elementos en el AAS. Están vinculadas a las propiedades mediante referencias semánticas, lo que garantiza que todos los datos asociados con esa propiedad se interpreten según un conjunto estándar y acordado de términos. Estas últimas permiten que las propiedades tengan un significado comúnmente entendido, lo que es crucial cuando se intercambia información entre diferentes sistemas o cuando se necesita asegurar la interoperabilidad, proporcionando una explicación clara del tipo de dato, unidad de medida, y el contexto de uso, lo que ayuda a evitar malentendidos o errores en la interpretación de los datos.

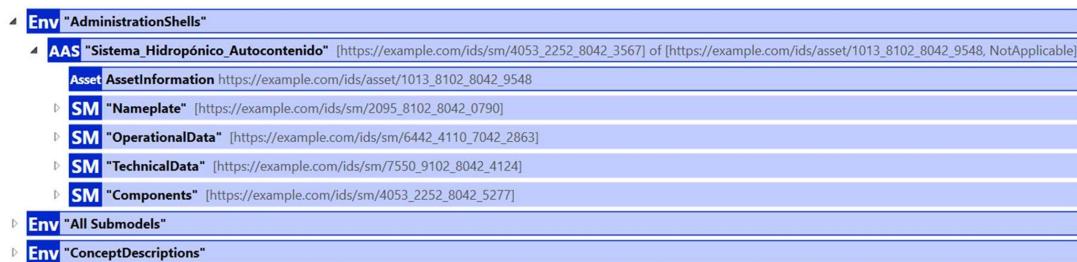


Figura 7 - Vista general del AAS.

A continuación, se presentan capturas de pantalla del software utilizado, donde se aprecian los distintos submodelos con sus propiedades correspondientes.

Submodelo “Nameplate”: Placa de identificación del Sistema Hidropónico Autocontenido.

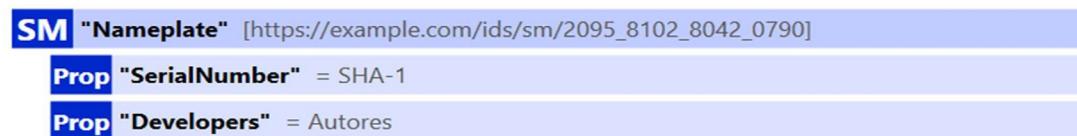


Figura 8 - Submodelo Nameplate.

## Submodelo “TechnicalData”: Datos técnicos del Sistema Hidropónico Autocontenido.

<b>SM "TechnicalData"</b> [https://example.com/ids/sm/7550_9102_8042_4124]
Prop "Crop" = menta
Prop "Cultivation" = menta-1
Prop "PotsNumber" = 30
Prop "Tank1Additive" = Regulador pH base
Prop "Tank2Additive" = Regulador pH acido
Prop "Tank3Additive" = Nutriente A
Prop "Tank4Additive" = Nutriente B
Prop "MainTankCapacity" = 0.00 [m3]
Prop "SecondaryTanksCapacity" = 0.00 [m3]
Prop "TargetTemperatureValue" = 0.00 [°C]
Prop "TargetpHValue_min" = 0.00
Prop "TargetpHValue_max" = 0.00
Prop "TargetECValue_min" = 0.00
Prop "TargetECValue_max" = 0.00
Prop "Pump0OnTime" = 0 [min]
Prop "Pump0OffTime" = 0 [min]
Prop "Pump0MinimumOnTime" = 0 [min]
Prop "LEDOnTime" = 0 [min]
Prop "LEDOffTime" = 0 [min]

Figura 9- Submodelo TechnicalData.

Submodelo “OperationalData”: Datos operacionales en tiempo real del Sistema Hidropónico Autocontenido.

<b>SM "OperationalData"</b> [https://example.com/ids/sm/6442_4110_7042_2863]
Prop "IndoorTemperature" = 0.00 [°C]
Prop "OutdoorTemperature" = 0.00 [°C]
Prop "SolutionTemperature" = 0.00 [°C]
Prop "RelativeHumidity" = 0.00 [%]
Prop "pH" = 0.00
Prop "EC" = 0.00
Prop "IlluminationLevel" = 0.00 [%]
Prop "LED1State" = 0
Prop "LED2State" = 0
Prop "Pump0State" = 0
Prop "Pump1State" = 0
Prop "Pump2State" = 0
Prop "Pump3State" = 0
Prop "Pump4State" = 0
Prop "Fan0State" = 0
Prop "Fan1State" = 0
Prop "Fan2State" = 0
Prop "Fan3State" = 0

Figura 10- Submodelo OperationalData.

## Submodelo “Components”: Información de los componentes del Sistema Hidropónico Autocontenido.

SM "Components" [https://example.com/ids/sm/4053_2252_8042_5277]	
Prop	"Dimensions"
Prop	"ProcessingUnit" = Arduino Mega 2560
Prop	"pHSensor" = pH sensor PH-4502C + Measurement Probe E-201-C
Prop	"ECSensor" = KS0429 keyestudio TDS Meter V1.0
Prop	"Flowmeter" = Yf-s401
Prop	"LightLevelSensor" = NORP12 RS
Prop	"Temperature_HumiditySensor" = DHT22
Prop	"SolutionTemperatureSensor" = DS18B20
Prop	"MainPump" = RS-932
Prop	"SecondaryPumps" = Vertical 100L/h 5V
Prop	"Fan" = DITRON SK-COOL2(R.C.F)
Prop	"LEDs" = Full Spectrum 50w

Figura 11 - Submodelo Components.

Por último, se presenta una captura de las Concept Descriptions utilizadas. Las mismas se utilizan para informar las unidades de las distintas variables y parámetros, como [°C] para la temperatura y [m³] para el volumen.

Env "ConceptDescriptions"	
CD	"TemperatureUnit" https://example.com/ids/cd/9353_8102_8042_8986
CD	"VolumeUnit" https://example.com/ids/cd/0472_9102_8042_2778
CD	"Percentage" https://example.com/ids/cd/1372_9102_8042_8933
CD	"TimeUnitMinutes" https://example.com/ids/cd/4185_9102_8042_7905

Figura 12- Concept Descriptions.

Para darle "vida" al AAS, se necesita conectar los elementos virtuales definidos en el AAS con los datos reales del sistema. Para ello, se almacenó la estructura definida del AAS en la placa Raspberry Pi, la cual a su vez estará periódicamente recibiendo los datos en tiempo real enviados por el Arduino.

## Evaluación del funcionamiento y resultados de las pruebas

### Control PID

Una vez obtenidos los parámetros iniciales mediante Ziegler-Nichols, se utilizaron para configurar el control PID dentro del código Arduino. Con el código funcionando, se procedió a realizar pruebas y ajustar los valores manualmente para obtener la calibración final, resultando:

$$K_p = 14.12 \quad K_i = 0.12 \quad K_d = 223.53$$

De esta manera, la temperatura se regula correctamente y permite situarla por debajo de la temperatura marcada como objetivo, la cual a su vez podría ser conceptualmente considerada un límite (Figura 13).

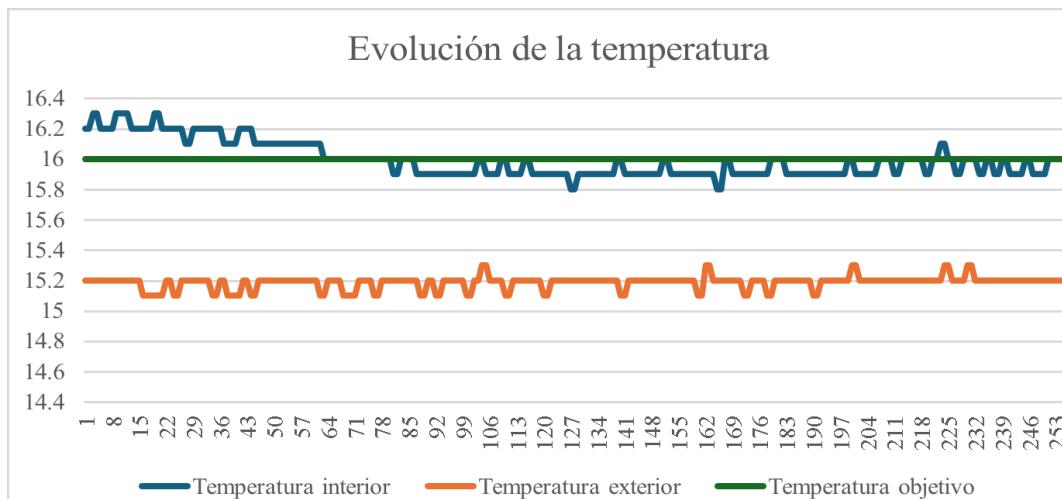


Figura 13 - Evolución de la temperatura al aplicar el control PID.

Se puede observar en la Figura 14 que los ventiladores responden al valor calculado como salida del PID estando encendidos (representado por el valor 10) cuando esta así lo requiere.

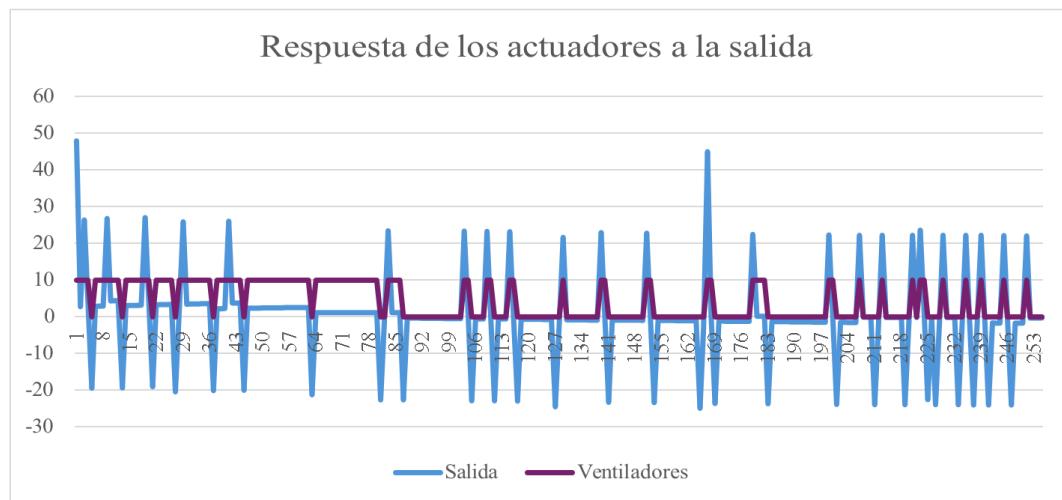


Figura 14 - Respuesta de los ventiladores a la salida del cálculo PID.

Como se mencionó en la sección que describe el cálculo de los parámetros de configuración de este método de control, la actuación debería ser idealmente llevada a cabo por un dispositivo que permita una variación continua y no que trabaje solo con dos valores lógicos, como es el caso de los ventiladores utilizados en este proyecto, los cuales, por acción de un relé, solo pueden encenderse y apagarse.

## Envío de datos

Para el envío de datos, se configuró el código para poder imprimir en formato JSON en el puerto Serie. En primer lugar, se evalúa si el puerto está efectivamente disponible y se aguarda por la llegada de un comando, en este caso, la letra “P”. La placa Raspberry será la encargada de imprimir en el puerto serie este comando para que, una vez que el Arduino lo lea, imprima los datos en formato JSON en el mismo puerto. Al recibirse el comando se comienza a generar el JSON con sus claves (nombres predefinidos en el AAS) y sus valores, los cuales se extraen de las variables del código que los almacenan. Una vez completo el JSON, se crea y se imprime en el puerto serie, donde la placa Raspberry se encarga de leerlo y posteriormente procesarlo.

```
{ "T_int": 18.2, "H_int": 88.3, "T_ext": 17.6, "T_solu": 18.1, "pH_value": 7.0,  
  "EC_value": 50, "LDR_value": 765, "T_setpoint": 18.3, "ph_setpoint_min": 6.0,  
  "ph_setpoint_max": 8.0, "EC_setpoint_min": 55, "EC_setpoint_max": 65,  
  "b0_OnTime": 60, "b0_OffTime": 180, "LED_OnTime": 60, "LED_OffTime": 60,  
  "LED1_st": 1, "LED2_st": 1, "b0_st": 1, "b1_st": 0, "b2_st": 0, "b3_st": 0,  
  "b4_st": 0, "vent0_st": 1, "vent1_st": 0, "vent2_st": 0, "vent3_st": 0  
}
```

## Control pH

Tanto para la función que controla el pH como la que controla la electroconductividad de la solución se utiliza la misma lógica:

1. Se trabaja por intervalos de tiempo. Al comienzo del código se define cada cuanto se debe medir.
2. Se ejecuta la función que lee el valor del sensor (en este caso simulado por un potenciómetro) y se almacena en una variable.
3. Se compara el valor obtenido con los límites mínimos y máximos del rango que se definió como aceptable. En caso de que supere alguno de ellos, se incrementa un contador, positivo para valores superiores y negativo para inferiores. En caso de que la medición se encuentre en valores aceptables, se reinicia el contador.
4. Se chequea si se produjeron más de 3 medidas consecutivas por fuera del rango aceptable, ejecutando las funciones que hacen funcionar las bombas secundarias correspondientes. Si el pH está por encima del rango, la bomba secundaria 1 introduce una solución ácida y, si se encuentra por debajo, la bomba secundaria 2 introduce una solución alcalina.

La evaluación de la respuesta de las bombas a los valores de pH simulados con potenciómetros resultó exitosa y con los datos obtenidos, se trazaron las gráficas que se presentan a continuación. Como se mencionó, el valor de pH fue simulado con un potenciómetro. Se definió a los valores entre 6 y 8 como rango aceptable de pH y se realizó una prueba con valores por encima y por debajo de los límites (Figura 15).

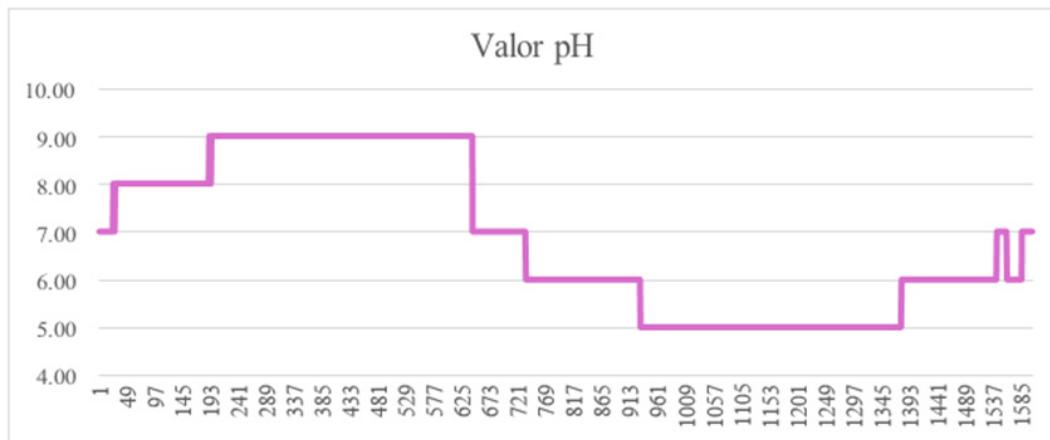


Figura 15 – Valor de pH simulado con un potenciómetro.

Se puede observar en la gráfica “Muestras” (Figura 16) que instantes antes de llegar a las tres muestras positivas se produjeron algunas mediciones por encima del valor aceptado de pH, pero no las suficientes como para obtener tres consecutivas. Este mecanismo se utilizó como filtro para evitar que una medición errónea pueda afectar al sistema. En este caso de prueba se utilizó el valor “3” pero como se aprecia en el código, podría configurarse a otro valor. Solo se actuará si se alcanza dicha cantidad de medidas consecutivas fuera del rango aceptado.



Figura 16 - Respuesta del contador que almacena la cantidad de muestras por fuera de los límites establecidos.

En el caso de las bombas, cuando la lógica del código les indica que deben actuar, éstas se encienden hasta que los caudalímetros detecten que los atravesó el volumen de solución que se brinda como parámetro cuando se ejecuta la función (Figura 17).

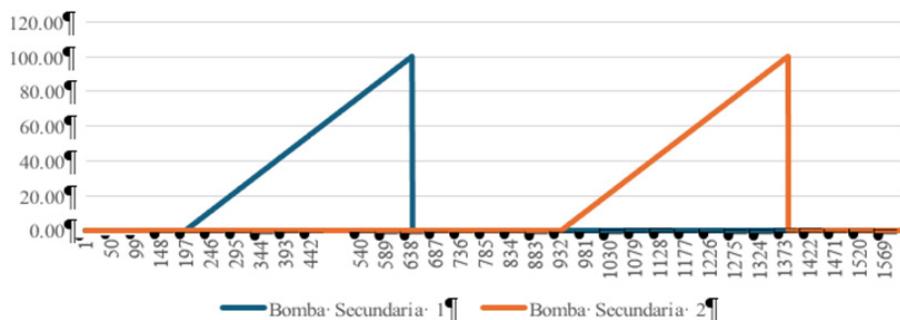


Figura 17 – Volumen desplazado por las bombas secundarias

A continuación, se presentan las gráficas superpuestas para lograr una mayor apreciación de la respuesta del sistema.

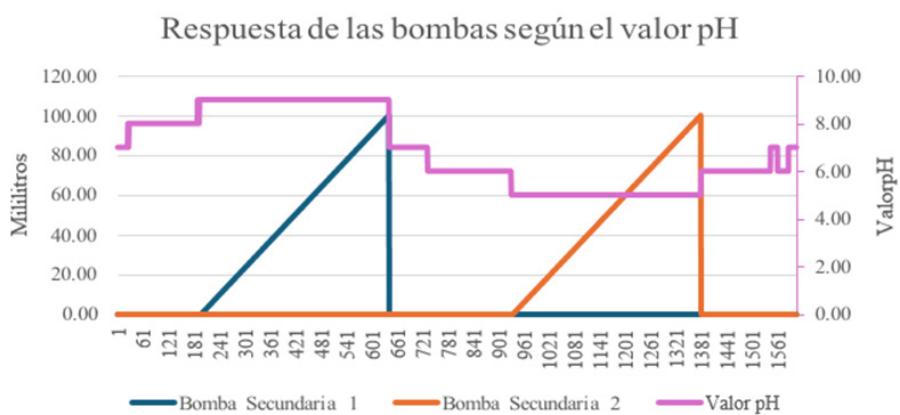


Figura 18 - Superposición del valor de pH y la acción de las bombas secundarias.

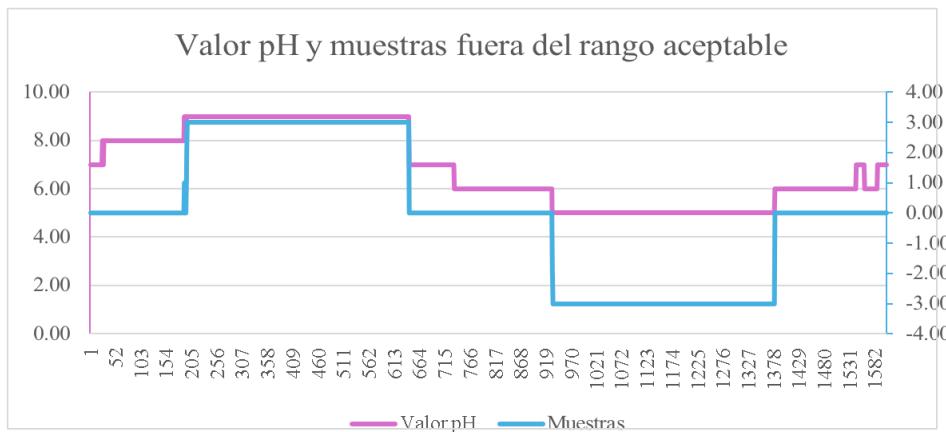


Figura 19 - Superposición del valor de pH y las muestras correspondientes.

## Encendido de bomba principal

La última prueba que se realizó fue la evaluación experimental del tiempo que debe estar encendida la bomba para lograr llevar la solución del tanque principal a todos los tubos del sistema, donde se encontrarán situadas las plantas. Esta prueba se hizo de manera experimental y no matemática ya que el conducto que sale hacia los tubos superiores y por el cual pasa todo el fluido desplazado por la bomba cuenta con un retorno que tiene un doble propósito: evitar un caudal muy grande y agitar el contenido del tanque principal, permitiendo la homogenización de la solución. Este retorno se logra con una válvula de derivación que se regula de forma manual. La ejecución de esta prueba dio como resultado 2 minutos, tiempo mínimo que debe estar encendida la bomba para suministrar el volumen de solución necesario para llegar a todas las plantas.

## Conclusiones

Los objetivos planteados en este trabajo fueron alcanzados satisfactoriamente. En primer lugar, se logró el desarrollo de un sistema automatizado para el cultivo hidropónico auto-contenido de plantas medicinales, culminando en un prototipo funcional. Tras la selección e instalación de sensores y actuadores, el sistema es capaz de monitorear y controlar parámetros ambientales críticos para el crecimiento vegetal. Algunas variables se controlan mediante umbrales simples, mientras que otras, como la temperatura, se regulan mediante un controlador PID ajustado mediante el método de Ziegler-Nichols. Este controlador demostró una regulación efectiva, manteniendo la temperatura dentro de un margen de  $\pm 0.5$  °C respecto al valor de consigna.

Asimismo, se concretó la propuesta de digitalización del sistema en el marco del modelo de referencia RAMI 4.0, permitiendo no solo su representación digital, sino también sentando las bases para la interoperabilidad con plataformas avanzadas de gestión. Esta digitalización fortalece la escalabilidad del sistema, facilitando su integración con otras tecnologías emergentes en el contexto de la Industria 4.0.

El sistema desarrollado representa una primera etapa sólida en la automatización y digitalización de sistemas de cultivo, con potencial de adaptación a otros entornos controlados, como invernaderos, laboratorios o centros de producción agrícola. Además, su diseño flexible permite modificar su configuración y escalarse para otros tipos de cultivos y condiciones ambientales, ampliando así su aplicabilidad.

Durante el desarrollo se identificaron desafíos técnicos, como la limitada resolución de algunos sensores —en particular el DHT22—, lo cual afectó la precisión del sistema de control y demandó ajustes manuales adicionales. Esta limitación evidenció la necesidad de evaluar cuidadosamente la selección de componentes en futuras implementaciones, especialmente cuando se busca una regulación precisa de variables críticas.

Finalmente, este trabajo no solo contribuye a la mejora tecnológica de los sistemas de cultivo hidropónico, sino que también representa un aporte a la investigación aplicada en el ámbito agrícola. Su enfoque sostenible, escalable y tecnológicamente innovador puede tener un impacto positivo en la productividad, la eficiencia y la sustentabilidad del cultivo de plantas medicinales, abriendo nuevas oportunidades para su implementación en contextos educativos, científicos y productivos.

## Referencias

- Aicardi A. (2022), "Hidroponía 4.0: un Mapeo Sistemático de la Literatura". Jornadas de Investigadores Tecnológicos.
- Castroverde, C. D. M., & Dina, D. (2021). Temperature regulation of plant hormone signaling during stress and development. *Journal Of Experimental Botany*. <https://doi.org/10.1093/jxb/erab257>
- DIN SPEC 91345. (2016). Reference Architecture Model Industrie 4.0 (RAMI 4.0). Berlin: Beuth Verlag GmbH. <https://www.dimmedia.de/en/technical-rule/din-spec-91345/250940128>.
- Eclipse AASX Package Explorer (n.d.). GitHub. <https://github.com/eclipse-aaspe/package-explorer>
- Jensen, M. H. (1997). Hydroponics. *HortScience*, 32(6), 1018-1021.
- Mattei, A.; Orué, M.; Caliusco, M. L. (13-15 de setiembre de 2023). Laboratorio autocontenido de hidroponía 4.0. VIII Congreso Argentino de Ingeniería Mecánica y III Congreso Argentino de Ingeniería Ferroviaria (CAIM CAIFE 2023), UTN. Facultad Regional Santa Fe, Argentina.
- Ogata, K. (2010). Ingeniería de Control Moderna (5.a ed.). Pearson Educación.
- Plattform Industrie 4.0 & ECLASS. (2021). Modelling the semantics of data of an asset administration shell with elements of ECLASS [White paper]. Plattform Industrie 4.0 & ECLASS.
- Rasheed, O., y Kvamsdal T. (2020). "Digital Twin: Values, Challenges and Enablers From a Modeling Perspective," in IEEE Access, vol. 8, pp. 21980-22012, <https://doi.org/10.1109/ACCESS.2020.2970143>.
- Resh, H.M. (2022). Hydroponic Food Production: A Definitive Guidebook for the Advanced Home Gardener and the Commercial Hydroponic Grower (8th ed.). CRC Press. <https://doi.org/10.1201/9781003133254>
- Savvas, D., y Gruda, N. (2018). Application of soilless culture technologies in the modern greenhouse industry – A review. *European Journal of Horticultural Science*, 83(5), 280-293. <https://doi.org/10.17660/ejhs.2018/83.5.2>
- Srivastava, K., Singh, S., Singh, A., Jain, T., Datta, R., y Kohli, A. (2023). Effect of Temperature (Cold and Hot) Stress on Medicinal Plants. En Medicinal Plants (pp. 153-168). [https://doi.org/10.1007/978-981-19-5611-9\\_5](https://doi.org/10.1007/978-981-19-5611-9_5)
- Urrestarazu, M. (2013). III International Symposium on Soilless Culture and Hydroponics: Innovation and Advanced Technology for Circular Horticulture | International Society for Horticultural Science. <https://www.ishs.org/symposium/711>

## Contribución de los Autores

Nombres y Apellidos del autor	Colaboración Académica													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Agustín Aicardi	X		X	X	x	X	X	x		X			X	X
María Laura Caliusco	X	X	X	X		X		X	X			X		X
Matías Orué	x		x	x		x		X	x			x		x

1-Administración del proyecto, 2-Adquisición de fondos, 3-Análisis formal, 4-Conceptualización, 5-Curaduría de datos, 6-Escritura - revisión y edición, 7-Investigación, 8-Metodología, 9-Recursos, 10-Redacción - borrador original, 11-Software, 12-Supervisión, 13-Validación, 14-Visualización.