

**Universidad Tecnológica Nacional**

Proyecto Final

---

**Sistema de Automatización y Control de Cultivos  
Hidropónicos**

---

*Autores:*

- Gottig, Carla Valeria
- Pérez Buttori, Fabricio

*Director:*

- Martín, Milton Tadeo

*Proyecto final presentado para cumplimentar los requisitos académicos  
para acceder al título de Ingeniero Electrónico*

*en la*

**Facultad Regional Paraná**

Julio 2021

## **Declaración de autoría:**

Nosotros declaramos que el Proyecto Final “Sistema de Automatización y Control de Cultivos Hidropónicos” y el trabajo realizado son propios.

Declaramos:

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero Electrónico, en la Universidad Tecnológica Nacional, Regional Paraná.
- Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
- Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firmas:

•

•

Fecha:

## Agradecimientos:

Un especial agradecimiento para el ingeniero Milton Martín, quien aceptó ser director del presente proyecto, acompañando el desarrollo y la concreción del mismo y al Sr. Carlos Perez, por su colaboración.

Agradecemos a la Universidad Tecnológica Nacional por la excelente formación académica que nos brindó a lo largo de la carrera.

Gottig, Carla Valeria  
Pérez Buttori, Fabricio

Universidad Tecnológica Nacional

*Abstract*

Facultad Regional Paraná

Ingeniería en Electrónica

**Sistema de Automatización y Control de Cultivos Hidropónicos**

Carla Valeria Gottig

Fabrizio Pérez Buttori

**Abstract:**

The project is based on a monitoring system for hydroponic crops, which includes the measurement of different parameters by means of sensors, the control of these variables, the on / off of the system and the action of actuating elements, accessible through a remote graphical interface hosted on a web page.

Was used an Arduino NANO board, whose function was to receive orders, a WEMOS D1 Mini board that is responsible for sending those orders, sensors for measuring temperature, humidity, conductivity, among others, actuators such as solenoid valves, relays and a website, together with an MQTT broker.

Finally, we obtained a control and automation system for hydroponic crops, capable of communicating orders and receiving data remotely through the internet, allowing its visualization in real time.

**Keywords:**

*Internet on Things (IoT), Master, MQTT, Slave*

**Resumen:**

El proyecto consiste en un sistema de monitoreo de cultivos hidropónicos, que incluye la medición de diferentes parámetros por medio de sensores, el control de dichas variables, el encendido/apagado del sistema y la acción de elementos actuadores, accesible a través de una interfaz gráfica remota alojada en una página web.

Se utilizaron una placa Arduino NANO, cuya función era la de recibir órdenes, una placa WEMOS D1 Mini que se encarga del envío de esas órdenes, sensores para medición de temperatura, humedad, conductividad, entre otros, actuadores tales como electroválvulas, relés y una página web junto con un bróker MQTT.

Finalmente, se obtuvo un sistema de control y automatización de cultivos hidropónicos capaz de comunicar órdenes y recibir datos remotamente mediante internet, permitiendo su visualización en tiempo real.

**Palabras Clave:**

Esclavo, Internet de las cosas (IoT), Maestro, MQTT

# Índice:

<b>Capítulo 1: Introducción</b> .....	1
1.1 Surgimiento de la idea.....	1
1.2 Descripción del Proyecto.....	1
1.3 Objetivos generales.....	3
1.4 Competencia.....	3
1.5 Análisis de costos.....	3
<b>Capítulo 2: Desarrollo</b> .....	4
2.1 Diagrama de bloques.....	4
2.2 Lógica de funcionamiento.....	4
2.2.1 Sistema NTF.....	4
2.3 Descripción de componentes.....	6
2.3.1 Arduino NANO.....	6
2.3.2 Placa Wemos D1 Mini.....	7
2.3.3 Sensor de Temperatura DS18B20.....	8
2.3.4 Sensor de Electroconductividad (PPM).....	9
2.3.5 Sensor de Caudal.....	12
2.3.6 Sensor de Temperatura y Humedad Ambiente DHT11.....	13
2.3.7 Actuadores.....	14
2.4 Comunicación entre microcontroladores.....	16
2.5 MQTT.....	18
2.6 Página Web.....	21
2.6.1 Agregado de nutrientes.....	22
2.6.2 Configuraciones de la Bomba.....	22
2.6.3 Comando Manual de la Bomba.....	24
2.6.4 Datos Históricos.....	25
2.6.5 Alertas.....	26
2.6.6 Configuración de estilos.....	26
2.7 Wemos como Web Server (Configuración de wifi).....	27
2.8 Placas de control y potencia.....	31
<b>Capítulo 3: Resultados</b> .....	33
<b>Capítulo 4: Análisis de Costos</b> .....	36
<b>Capítulo 5: Discusión y Conclusión</b> .....	38
<b>Capítulo 6: Literatura Citada</b> .....	39

## Lista de Figuras:

Figura 1 - (Hidroponía Urbana, 2017) Sistema hidropónico NFT (Nutrient Film Technique).....	2
Figura 2 - Diagrama en bloques del sistema.....	4
Figura 3 - Sistema NFT por niveles.....	5
Figura 4 - (Solo Arduino Blog, 2014) Arduino NANO.....	6
Figura 5 - (HWLibre, 2015) Wemos D1 Mini.....	8
Figura 6 - (Mecatronium Chips, 2020) Sensor DS18B20.....	9
Figura 7 - Esquema de conexión al Arduino.....	9
Figura 8 - Sonda diseñada.....	10
Figura 9 - Relación entre la frecuencia y los niveles de PPM.....	11
Figura 10 - Circuito del acondicionador de señal proveniente de la sonda.....	11
Figura 11 - Diseño de la placa.....	12
Figura 12 - (Rodrigo Martel, 2019) Caudalímetro utilizado.....	12
Figura 13 - (Naylamp Mechatronics, 2016) Sensor DHT11.....	13
Figura 14 - Esquema de conexión a Arduino.....	14
Figura 15 - Electroválvula utilizada.....	15
Figura 16 - Comunicación por medio del puerto serie.....	16
Figura 17 - Inicialización del puerto serie.....	17
Figura 18 - Inicialización del puerto serie virtual.....	17
Figura 19 - Conexión entre ESP y Arduino.....	18
Figura 20 - (Luis Llamas, 2019) Protocolo MQTT.....	19
Figura 21 - Suscripción al broker.....	19
Figura 22 - Código para publicación del mensaje.....	19
Figura 23 - Suscripción al tópico “bomba”.....	20
Figura 24 - Función condicional.....	20
Figura 25 - Vista de la sección de datos recibidos.....	21
Figura 26 - Vista de la sección de nutrientes.....	22
Figura 27 - Configuraciones de bomba.....	23
Figura 28 - (Llamas Luis, 2016) Real Time Clock (RTC).....	24
Figura 29 - Encendido y apagado de la bomba.....	25
Figura 30 - Sección de Históricos.....	25
Figura 31 - Alertas que indicarían ausencia de agua. Se muestra que no hay nuevas alertas.....	26
Figura 32 - Configuración de estilos.....	27
Figura 33 - Conexión del ESP8266 como Access Point.....	28
Figura 34 - Imagen de la página del Web Server.....	29
Figura 35 - Configuración de red wifi.....	30
Figura 36 - Código del setup para selección de modo.....	30
Figura 37 - Diseño en Proteus de la placa de control.....	31
Figura 38 - Diseño en Proteus de la placa de potencia.....	32
Figura 40 - Vista frontal.....	33
Figura 39 - Vista del exterior del equipo.....	34
Figura 41 - Vista del interior del equipo.....	35



# Lista de Tablas

Tabla 1: Análisis de Costos.....36

## Lista de Abreviaciones y Símbolos:

NFT	Nutrient Film Technique
PPM	Partes Por Millón
MQTT	MQ Telemetry Transport
IOT	Internet Of Things
SOM	System Of a Module
SOC	System Of a Chip
PWM	Pulse Width Modulation
EC	Electro Conductividad
AWS	Amazon Web Service
RTC	Real Time Clock
GPIO	General Purpose Input Output
USART	Universal Synchronous Asynchronous Receiver Transmitter
I2C	Inter-Integrated Circuit

Dedicado a:

Nuestras familias, por el apoyo incondicional para lograr el objetivo de recibirnos de ingenieros, por habernos forjado como las personas que somos en la actualidad; muchos de nuestros logros se los debemos a ustedes entre los que se incluye este.

## **Capítulo 1: Introducción**

### **1.1 Surgimiento de la idea**

Inicialmente, se plantearon diferentes propuestas para realizar como Proyecto Final, siempre con el objetivo de resolver alguna problemática existente aplicando la electrónica y la programación.

Leyendo artículos en internet, surgió la idea de realizar un dispositivo capaz de monitorear y controlar cultivos hidropónicos, permitiendo que los agricultores o aficionados a esta técnica de agricultura tengan datos en tiempo real acerca del funcionamiento del sistema y a partir de esa información, puedan realizar acciones. Incluso con la posibilidad de configurar ciertos parámetros para que el dispositivo pueda actuar por sí mismo, sin la necesidad de la intervención del usuario. Problemáticas usuales en la Hidroponía, como la falta de agua por evaporación en el cultivo o la obstrucción de cañerías por raíces, son temas que requieren atención constante por parte de quién posee un cultivo de características NFT (técnica que consiste en la circulación de solución nutritiva por cañerías en donde se encuentran las raíces de las plantas). Dichas situaciones pueden fácilmente detectarse con ciertos sensores y generar una respuesta automática, que permita que las plantas no presenten dificultades de crecimiento.

Se investigó sobre el tema y se llegó a la conclusión que es posible realizarlo considerando que aportaría un nivel de tecnología interesante para los pequeños y medianos agricultores hidropónicos, como así también para los aficionados a esta técnica, que ha alcanzado gran popularidad en los últimos años.

### **1.2 Descripción del Proyecto**

En la actualidad, es cada vez más imperiosa la necesidad de alimentarse de manera saludable, respetando el medio ambiente y los recursos naturales. Por dicha razón han cobrado gran importancia los cultivos alternativos como la hidroponía, una técnica agrícola que permite producir diferentes plantas en el agua (con determinados nutrientes), sin utilizar el suelo o la tierra.

La técnica de película nutritiva (NFT) es la más usada dentro de los cultivos hidropónicos y es la más aplicada comercialmente en el mundo, siendo utilizada en más de 67 países. La solución nutritiva en el medio que se encuentra la raíz es renovada constantemente, permitiendo el adecuado suministro de minerales, oxígeno y nutrientes necesarios para el crecimiento de la planta.

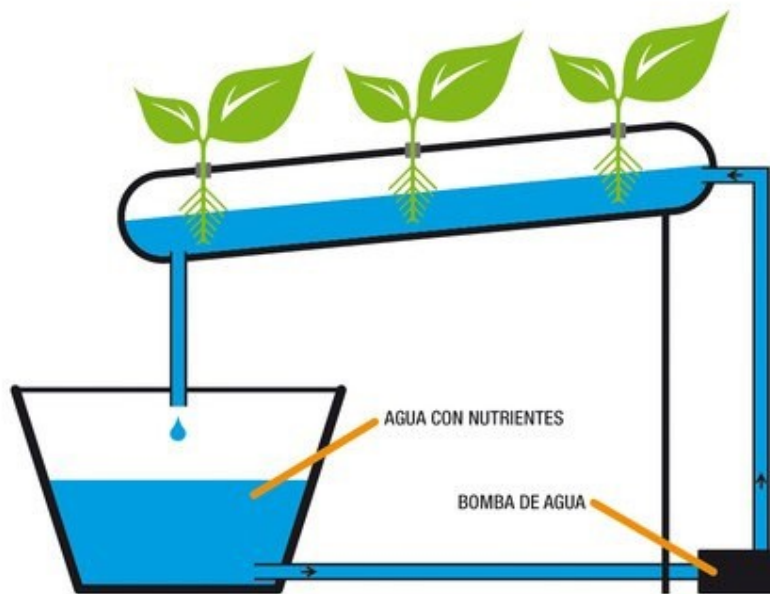


Figura 1 – (Hidroponía Urbana, 2017) Sistema hidropónico NFT (Nutrient Film Technique)

Las ventajas que se presentan en este método son muchas: reduce el gran consumo de agua y fertilizantes, disminuye considerablemente los problemas de enfermedades producidas por patógenos presentes en el suelo, se desarrollan cultivos en menor tiempo, mejorando la calidad de la producción y economizando costos; finalmente, es de destacar que estas plantas están libres de pesticidas.

Nuestro proyecto busca aplicar los conocimientos de la Ingeniería Electrónica con el fin de diseñar un sistema de control y automatización de cultivos hidropónicos. El mismo consiste en un sistema capaz de monitorear los datos del cultivo como: temperatura de la solución, humedad ambiental, temperatura ambiental, concentración de sales, entre otros; permitiendo al usuario saber las condiciones en las que se encuentra el cultivo en cada momento para desarrollar la actividad de manera óptima.

Dicho sistema consiste a grandes rasgos en un conjunto de sensores conectados a un microcontrolador, que hace la función de esclavo. A su vez, el esclavo está conectado a un controlador maestro, el cual permite el acceso a los datos y comunica las acciones de manera remota a través de internet.

El “esclavo”, (módulo de medición), le envía al “maestro”, (módulo que hace de central), los valores medidos que permiten al usuario monitorear: fallas en la bomba de agua, valores de PPM fuera de rango normal (esto indica que debe reponer o reemplazar nutrientes en el cultivo), alta o baja temperatura del ambiente o de la solución nutritiva y baja humedad en el ambiente. Asimismo, el esclavo tiene el control sobre del encendido y apagado de la bomba de agua según lo transmita el maestro.

### 1.3 Objetivos generales

Los objetivos del presente proyecto son los siguientes:

- Brindar a los pequeños y medianos productores hidropónicos y aficionados una solución para las posibles problemáticas que se presenten en sus cultivos.
- Conocer el estado del sistema a través del uso de sensores.
- Accionar actuadores, tanto de manera manual como automática, que permitan la sostenibilidad del cultivo.
- Mostrar en la página web todos los datos obtenidos y las posibles acciones.
- Almacenar datos en un historial para su posterior consulta.

### 1.4 Competencia

Argentina es un país que cuenta con excelentes condiciones agropecuarias, pero es castigado por las inclemencias climáticas, lo que abre una puerta a la expansión de la hidroponía. Nuestro objetivo es llegar principalmente a los pequeños y medianos productores hidropónicos que utilizan la técnica NFT (Nutrient Film Technique), que es el sistema más popular para la producción de cultivos en el mundo.

Debido a que, no se halló en el mercado sistemas que permitan monitorear correctamente este tipo de cultivos y la mayor parte de las mediciones son realizadas por una persona que las registra manualmente. Se plantea la necesidad de crear una herramienta tecnológica, que le permita al productor hidropónico mejorar sus cultivos mediante un control de todas las variables que están involucradas en el proceso.

### 1.5 Análisis de Costos

Inicialmente se realizó un análisis de los costos requeridos para la realización del proyecto. Los valores aquí presentes son aproximados y corresponden al precio de compra de una sola unidad. En caso de realizar pedidos de mayor volumen, dichos costos se reducirían.

- Precio aproximado de todos los sensores (temperatura, humedad, etc.): \$3.000
- Precio aprox. de los microcontroladores (considerando un ESP8266 y un Arduino Nano): \$1.500
- Componentes electrónicos (relés, borneras, resistencias, capacitores, etc.): \$3.000
- Materiales eléctricos (cables, conectores, etc.): \$5.000

El precio que se obtuvo corresponde a una estimación de lo que cuesta cada parte que compone el sistema. Esto va a depender de la relación precio calidad y de los productos existentes en plaza, por ejemplo los sensores y controladores que se utilicen.

## Capítulo 2: Desarrollo

### 2.1 Diagrama de Bloques

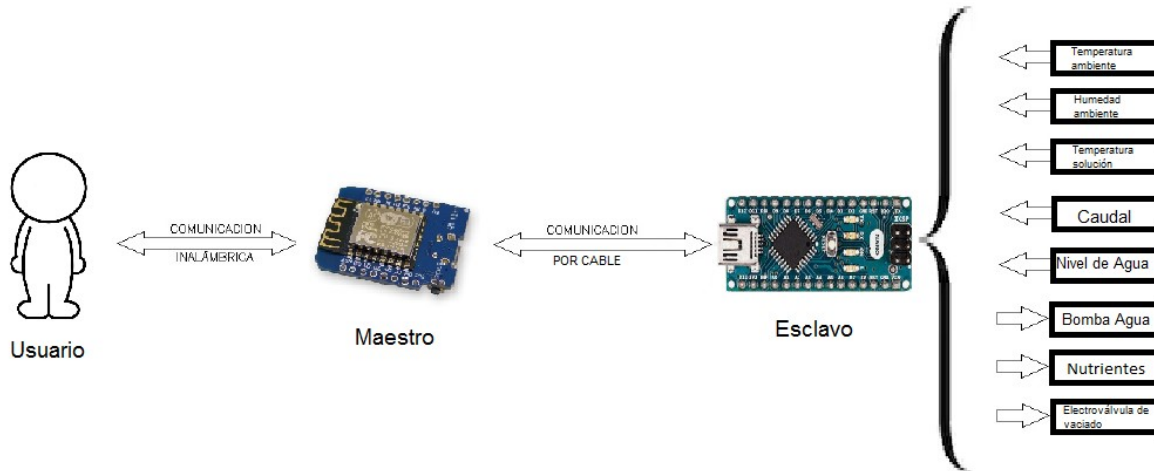


Figura 2 - diagrama en bloques del sistema

### 2.2 Lógica de Funcionamiento

El sistema se separa en tres etapas: la primera corresponde a los sensores que están en contacto con el cultivo hidropónico y el microcontrolador que maneja dichos sensores, al cual se denominó “esclavo”; la segunda etapa consiste en el microcontrolador (al que se llamó “maestro”) que “maneja” al esclavo y hace de conexión entre este y la tercera etapa, que consiste en la interfaz gráfica a la que se conecta el usuario y sobre la que recibe la información de su cultivo.

En la etapa uno, los sensores toman la información de temperatura de la solución, temperatura ambiental, humedad ambiental, caudal de agua que circula por el cultivo y PPM, seguidamente la envían a un microcontrolador esclavo, que en este caso consiste en un Arduino NANO. Este último se conecta mediante la USART al microcontrolador maestro (que consiste en una placa Wemos D1 Mini con ESP8266 integrado). A su vez, el maestro comunica enviando los datos a un broker mediante protocolo MQTT. Dichos valores son mostrados al usuario mediante una página web que además desde la misma puede ejecutar acciones sobre el cultivo (ingresar tiempos de encendido/apagado de la bomba, agregar nutrientes o agua a la solución, entre otros), que son enviadas al maestro que transmite la orden al esclavo para que ejecute las tareas solicitadas.

#### 2.2.1 Sistema NFT

El sistema NFT (Nutrient Film Technique) fue desarrollado en la década del 60 por el Dr. Allan Cooper en Inglaterra y es el sistema más usado para la producción de hortalizas, especialmente especies de hoja verde como son lechuga, espinaca, acelga, entre otras. Presenta grandes ventajas frente a otros sistemas: ahorro significativo en nutrientes y agua, menor tiempo de cosecha y buen aprovechamiento del espacio; ya que se puede cultivar en niveles.



Figura 3 - Sistema NFT por niveles

Esta técnica se basa principalmente en la reducción de espacio y comprende una serie de diseños, en los cuales el principio básico es la circulación continua o intermitente de una fina capa de agua con solución nutritiva, que pasa por una serie de canales de cultivo, éstos pueden ser de PVC, polietileno o poliuretano.



En dichos canales hay aberturas circulares donde se colocan las plantas, las que van dentro de un recipiente contenedor con una salida para las raíces de las mismas, que deben estar en contacto con la película de solución que circula.

Esta recirculación mantiene el sistema radicular irrigado permanente con la solución nutritiva, favoreciendo la oxigenación de las raíces y un suministro adecuado de nutrientes minerales para el desarrollo del cultivo. Como las plantas encuentran fácilmente disponibles los nutrientes, el gasto de energía es mínimo y es aprovechado en otros procesos metabólicos.

## 2.3 Descripción de componentes

### 2.3.1 Arduino NANO

Los Arduinos son plataformas de desarrollo de código abierto basados en placas sencillas con microcontroladores de la línea ATMEL. Al ser de software libre, la comunidad de Arduino brinda a disposición de los usuarios una gran variedad de librerías y componentes que facilitan el desarrollo de proyectos.

Para llevar a cabo este trabajo se optó por la placa NANO, la cual dispone de una cantidad suficiente de entradas y salidas digitales/analógicas. En este caso cumple una doble función, por un lado la de obtener los valores de los sensores y transmitirlos al maestro. Por otra parte la de recibir las órdenes que este último le transmite para comandar los actuadores.

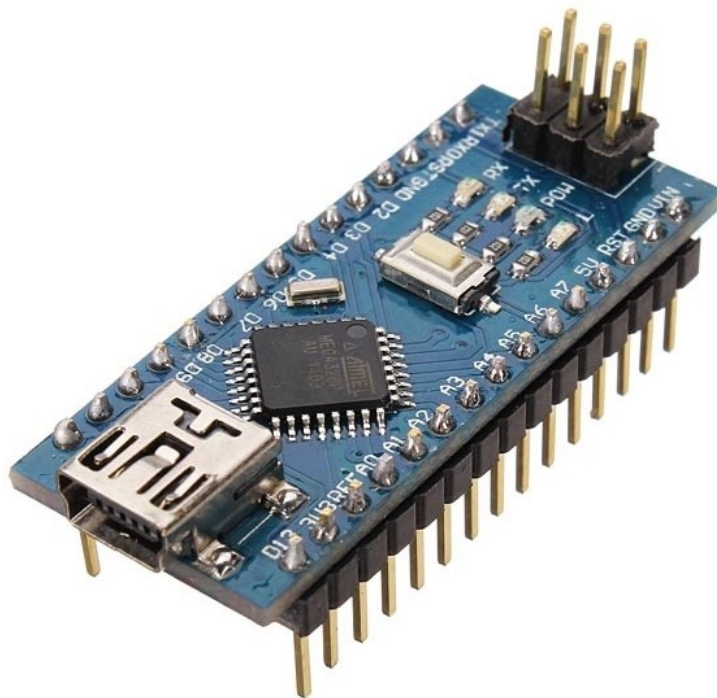


Figura 4 – (Solo Arduino Blog, 2014) Arduino NANO

**Especificaciones técnicas**

- Microcontrolador: ATmega328P
- Voltaje de alimentación: 7V – 12V DC
- Voltaje de I/O: 5V
- Cantidad de pines digitales I/O: 14
- Cantidad de pines PWM: 6
- Cantidad de entradas analógicas: 8
- Corriente máxima de entrada/salida: 40 mA.
- Memoria FLASH: 32KB (2KB usados por el Bootloader)
- Memoria SRAM: 2KB
- Memoria EEPROM: 1KB
- Frecuencia de reloj: 16 MHz
- Dimensiones: 18.5 mm x 43.2 mm

**2.3.2 Placa Wemos D1 Mini**

Wemos es una plataforma de desarrollo similar al Arduino orientada al internet de las cosas (IoT). La placa tiene como núcleo al SoM (System on Module) ESP12E, que a su vez está basado en el SoC (System on a Chip) ESP8266.

El ESP8266 es un chip diseñado para las necesidades actuales de estar todo conectado mediante IoT, integra un microcontrolador con arquitectura de 32 bits y conectividad Wi-Fi. El ESP12 integra en un módulo al ESP8266, una memoria flash de 4MB, un cristal oscilador y una antena impresa en PCB para Wi-Fi.

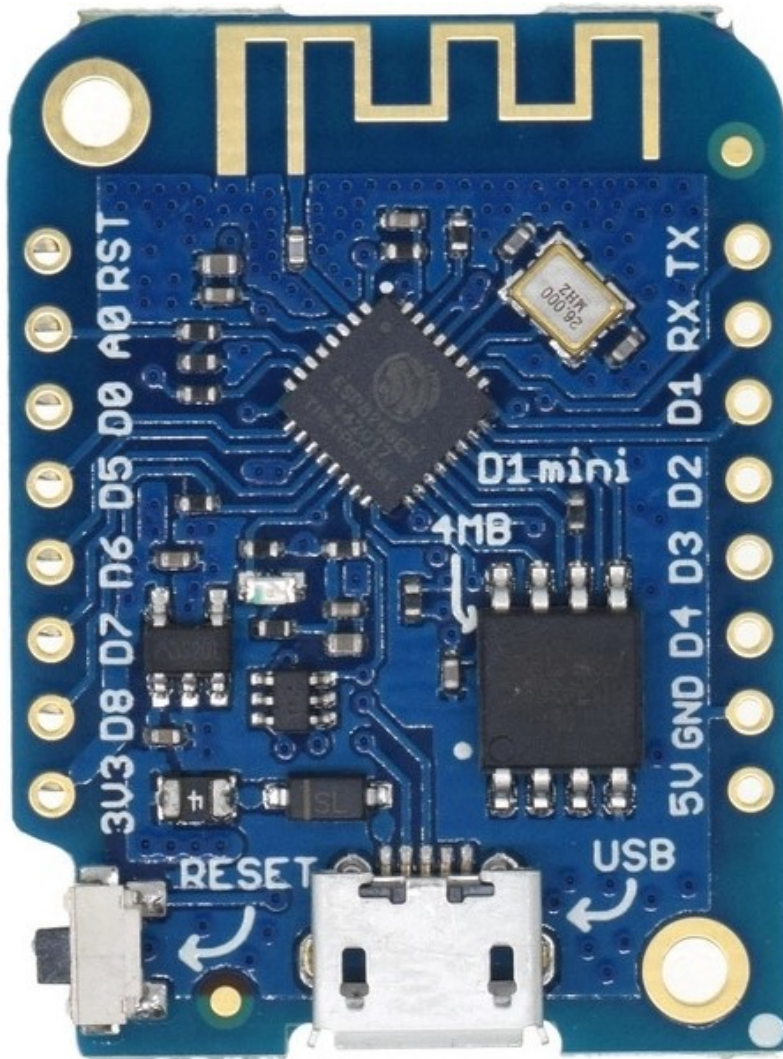


Figura 5 – (HWLibre, 2015) Wemos D1 Mini

### Especificaciones técnicas

- Memoria Flash de 4MB
- Alimentación de 3.3V, con posibilidad de alimentar con 5V mediante un conversor
- 11 GPIO, todos con PWM a excepción del pin D0
- Interrupciones
- Bus I2C
- 1 Entrada Analógica
- Conector Micro USB

### 2.3.3 Sensor de Temperatura DS18B20

Se trata de un sensor digital para medir la temperatura de una solución, capaz de sumergirse en el depósito utilizado en el cultivo. Dispone de 3 pines: Vcc para alimentarlo con 5V, GND y el pin de datos, por donde se recibe la información.



Figura 6 – (Mecatronics Chips, 2020) Sensor DS18B20

Estos sensores tienen un rango de temperatura de entre  $-55^{\circ}\text{C}$  y  $125^{\circ}\text{C}$ , pero como todos los sensores existentes, tienen errores de medición y aquí el error no es el mismo en todo el rango abarcado. De modo que para temperaturas de entre  $-10^{\circ}\text{C}$  y  $85^{\circ}\text{C}$  puede haber un error de  $\pm 0,5^{\circ}\text{C}$  y para el resto de valores de  $\pm 2^{\circ}\text{C}$ . Para este proyecto, si bien se debe tener en cuenta, no es algo demasiado significativo; ya que el sistema no estará expuesto a temperaturas tan extremas.

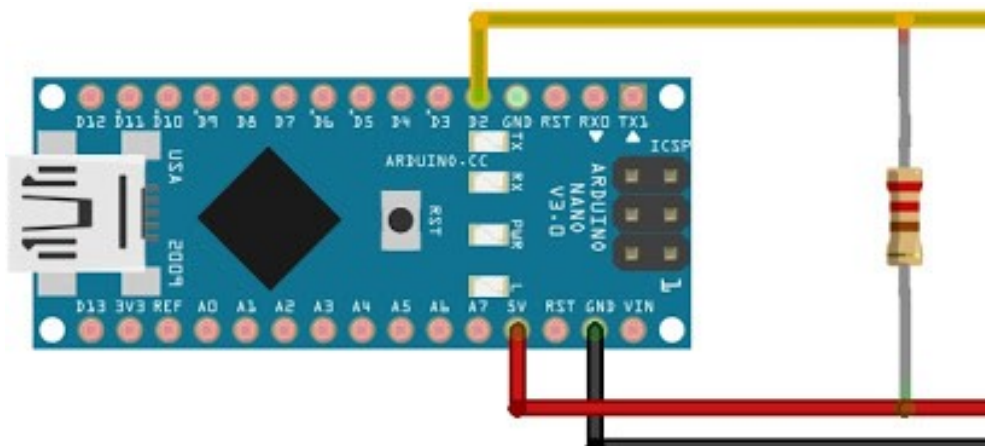


Figura 7 – Esquema de conexión al Arduino

### 2.3.4 Sensor de Electro-conductividad (PPM)

Debido a que los sensores para medición de electro-conductividad existentes en el mercado son muy costosos, se diseñó y elaboró un instrumento capaz de monitorear esta variable en un cultivo hidropónico, permitiendo mantener los niveles de sales dentro de rangos específicos para evitar la muerte de las plantas.

Dicho instrumento mide con fiabilidad los valores de conductividad eléctrica de 1.5 a 3 mS/cm, con incertidumbre del 10%.

La conductividad eléctrica indica la concentración de sales disueltas en el agua. Este parámetro es de suma importancia, pues un valor bajo en el mismo indicaría que la solución tiene deficiencia en nutrientes, mientras que un valor alto mostraría que la solución es tóxica para el cultivo.

La conductividad eléctrica (EC) es una medida de cuánta corriente eléctrica es capaz de transmitir una solución. Además matemáticamente es igual al recíproco de la resistividad. El principio de medición de este parámetro consiste en aplicar una corriente eléctrica a dos electrodos sumergidos en la solución, con el objetivo de medir la tensión resultante.

Se diseñó un circuito generador de onda cuadrada con una sonda que consiste en dos electrodos que son excitados por dicha onda. Al insertar la sonda en un medio salino, se genera en sus extremos una resistividad, la cual permite variar la frecuencia de la señal cuadrada.

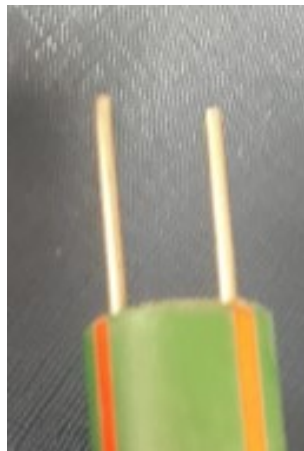


Figura 8 - Sonda diseñada

Sabiendo que a mayor cantidad de sales, la resistividad de la solución disminuye, por ende la frecuencia aumenta, al ser la resistencia inversamente proporcional a esta. Entonces se puede apreciar una relación proporcional entre la conductividad y la frecuencia.

Utilizamos un instrumento patrón para medir la conductividad de la solución, que contrastamos con las mediciones de frecuencia obtenidas por nuestro circuito, seguidamente con los valores resultantes se generó una tabla. Se pudo comprobar que existe una relación lineal entre la EC y la frecuencia; se obtuvo la función cuya variable independiente es EC.

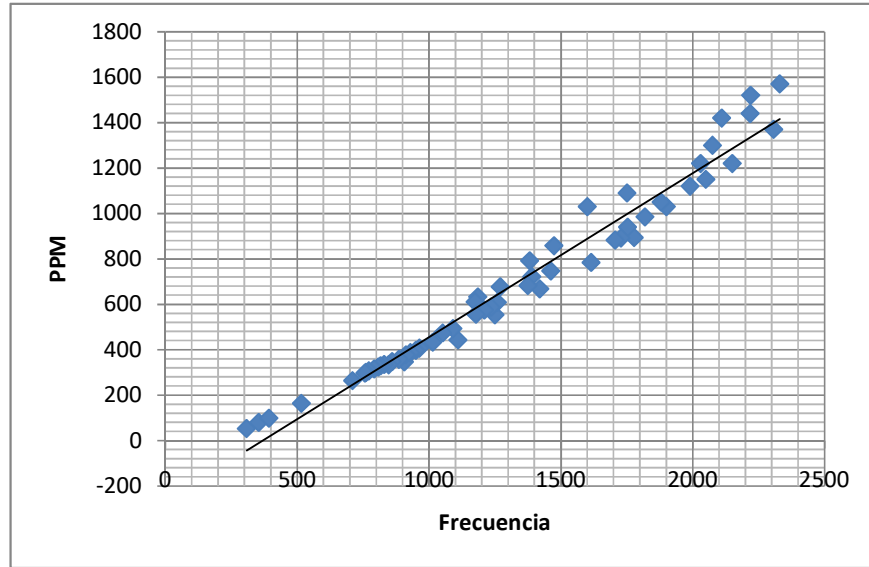


Figura 9 - Relación entre la frecuencia y los niveles de PPM

El circuito consiste en un CI 555 que opera en modo de retroalimentación directa. Aquí, el capacitor C2 (ver figura) se carga a través del resistor R1 que está en paralelo con la resistencia medida en la sonda. El pin 7 del CI555 es enviado a un pin digital del Arduino, el cual incrementa un contador por cada flanco descendente de la señal. Este contador corresponde a la variable dependiente de la función lineal generada previamente, obteniéndose el valor de la EC

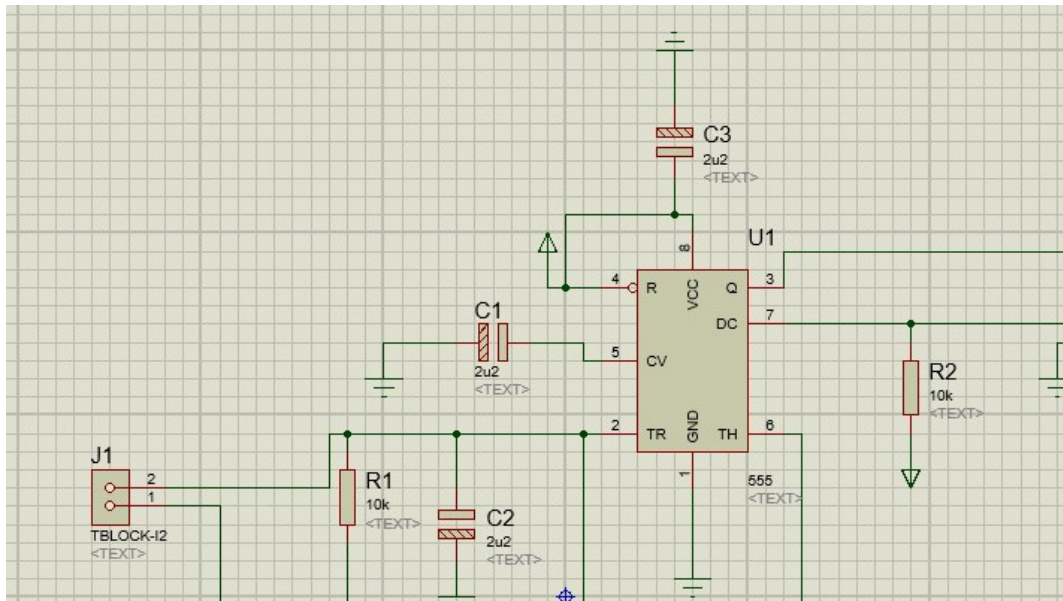


Figura 10 – Circuito del acondicionador de señal proveniente de la sonda



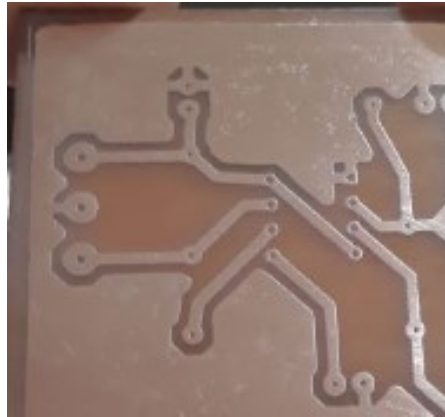


Figura 11 - Diseño de la placa

Ahora bien, la programación de este medidor consiste en generar una interrupción externa por cada flanco descendente. Dicha interrupción ejecuta una función que incrementa un contador, el cual se interpreta como la frecuencia a medir. Utilizando la fórmula ya generada cuya variable independiente es EC, obtenemos dicho valor reemplazando la frecuencia, en consecuencia el valor de las PPM de la solución.

### 2.3.5 Sensor de Caudal

La finalidad de instalar un medidor de caudal es la de controlar que el flujo de agua a la salida del sistema sea constante, pues en caso de que la misma disminuya, es un indicador de un problema a solucionar: ya sea una obstrucción provocada por las raíces, mal funcionamiento de la bomba, etc. Esto puede provocar que el tanque de líquido quede sin suficiente concentración, dejando a las plantas sin nutrientes.



Figura 12 – (Rodrigo Martel, 2019) Caudalímetro utilizado

Para su funcionamiento se utilizan interrupciones externas. Básicamente, por cada detección del sensor, se activa la interrupción INT1, incrementando un contador. Luego, se mide la cantidad de flancos ascendentes en un período de tiempo y se realiza una conversión, obteniéndose el caudal en litros por minuto.

### 2.3.6 Sensor de Temperatura y Humedad Ambiente DHT11

Se trata de un sensor capaz de proporcionar una alta fiabilidad y estabilidad, gracias a que realiza lecturas digitales en lugar de analógicas.

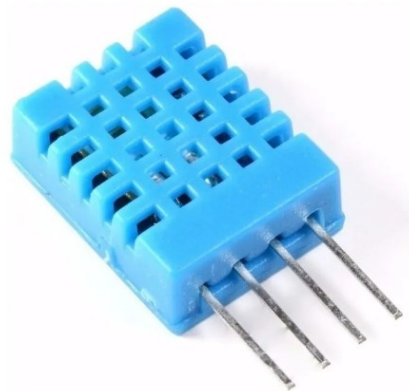


Figura 13 – (Naylamp Mechatronics, 2016) Sensor DHT11

Especificaciones:

- Rango de medición de humedad: 20% - 90%
- Rango de temperatura de medición: 0 – 50 °C
- Precisión en medición de humedad:  $\pm 5\%$  HR
- Precisión en medición de temperatura:  $\pm 2\text{ }^{\circ}\text{C}$
- Tiempo de respuesta: menor a 5 segundos

Como ya se dijo, el sensor envía una señal digital, por lo que debemos conectarlo a un pin de este tipo (en nuestro caso, lo conectamos al pin digital 6). Y si bien presenta estas características, se trata en realidad de un sensor analógico, de manera que es el propio dispositivo el que realiza la conversión de señal analógica a digital, proporcionando una trama de 40 bits correspondientes a la información que el mismo recibe.

Los primeros 8 bits de esa trama conforman la parte entera de la medición de humedad, mientras que los siguientes 8 bits son la parte decimal de dicha medición. Luego siguen los 8 bits de la parte entera de la medición de temperatura y a continuación los 8 bits de la parte decimal. Finalmente, se tienen 8 bits de paridad, para la confirmación de datos.

Este módulo tiene la capacidad de proporcionar la temperatura tanto en grados Celsius como en grados Fahrenheit, tan solo basta con agregar una línea de código. Debido a que no es usada en nuestro país esta última unidad de temperatura, sólo se optó por medir en grados centígrados.



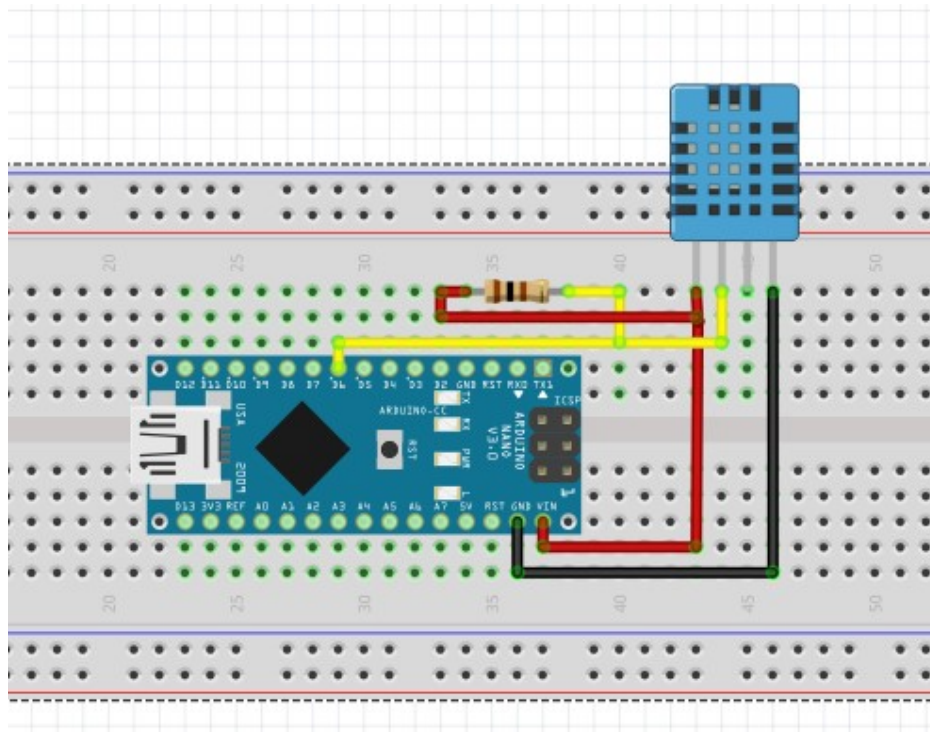


Figura 14 – Esquema de conexión a Arduino

Para la implementación de este sensor se utiliza la librería provista por Adafruit. El código consiste en las lecturas de humedad y temperatura mediante la siguiente secuencia:

```
dht.readHumidity(); // Para la lectura de la humedad
dht.readTemperature(); // Para la lectura de la temperatura
```

Basta con agregar un “true” a la secuencia de lectura de temperatura para obtener la misma en grados Fahrenheit.

### 2.3.7 Actuadores

Los elementos actuadores de nuestro esclavo consisten en 4 electroválvulas que permiten el paso de los nutrientes y agua al depósito. Las mismas constan de una entrada de  $\frac{3}{4}$  pulgada y una salida de  $\frac{1}{4}$  pulgada, la cual se buscó adaptar debido a que su diámetro era demasiado grande y no presenta la precisión requerida, pues las medidas de las concentraciones de sales son muy pequeñas.



Figura 15 –(elagmecatronica.blogspot.com) Electroválvula utilizada

Se realizaron pruebas con diferentes diámetros de salida, a fin de obtener determinados centímetros cúbicos de nutrientes en un período específico de tiempo.

Para una salida de 77 mm de diámetro, se midió lo siguiente:

Apertura de 1 minuto de la electroválvula – salida de 700 cm<sup>3</sup> de nutrientes

Apertura de 15 segundos – salida de 170 cm<sup>3</sup>

Apertura de 5 segundos – salida de 51 cm<sup>3</sup> de nutrientes

Apertura de 3 segundos – salida de 24 cm<sup>3</sup> de nutrientes

Se ve claramente que este diámetro no es óptimo, ya que para 10 litros de agua le corresponden 20 mL de nutrientes, una medida muy pequeña. Al bajar el tiempo a menos de 3 segundos, se generan imprecisiones en las concentraciones. Lo que se buscaba era que la cantidad de nutrientes requerida se obtenga en períodos más largos de tiempo, generando una mayor precisión.

Se redujo la salida a 57 mm de diámetro y se efectuaron nuevamente las mismas mediciones:

Apertura de 15 segundos – salida de 100 cm<sup>3</sup> de nutrientes

Apertura de 5 segundos – salida de 42 cm<sup>3</sup> de nutrientes

Apertura de 3 segundos – salida de 24 cm<sup>3</sup> de nutrientes

Una vez más, se observa que ocurre el mismo efecto que para un diámetro de 57 mm. Se redujeron mucho más los diámetros y a continuación se detalla lo obtenido.

Para una salida de 3 mm:

Apertura de 15 segundos – salida de 52 cm<sup>3</sup>

Apertura de 5 segundos – salida de 24 cm<sup>3</sup> de nutrientes

Apertura de 3 segundos – salida de 14 cm<sup>3</sup> de nutrientes

Para salida de 2 mm:

Apertura de 5 segundos – salida de 15 cm<sup>3</sup>

Apertura de 3 segundos – salida de 10 cm<sup>3</sup> de nutrientes

Apertura de 2 segundos – salida de 6 cm<sup>3</sup> de nutrientes

Para salida de 1,5 mm:

Apertura de 15 segundos – salida de 45 cm<sup>3</sup>

Apertura de 5 segundos – salida de 15 cm<sup>3</sup> de nutrientes

Apertura de 3 segundos – salida de 9,5 cm<sup>3</sup> de nutrientes

Se corroboró que la salida de 1,5 mm era ideal para nuestra aplicación, pues es la más exacta.

Para la electroválvula que ingresa agua al sistema, el diámetro de salida no sufrió modificaciones. Esta se activa cuando el sensor de nivel detecta que hay poca cantidad de líquido (cabe aclarar que el usuario puede seleccionar el nivel mínimo). Frente a una situación de falta de líquido se da una secuencia de pasos: en primer lugar se produce el corte total de la bomba, para evitar que la misma se quemé; luego se abre la válvula y comienza a ingresar agua durante un tiempo determinado. Pasado dicho tiempo, el microcontrolador chequea que el sensor no esté activado (si lo está, es posible que el tanque haya sufrido una rotura y pierda líquido) y si el mismo no lo está, enciende la bomba y continúa su funcionamiento normal.

Cuando se activa este actuador, luego de unos minutos es muy importante hacer un control de los nutrientes, ya que al ingresar agua sin nutrientes, los valores de concentración se ven modificados y la planta puede sufrir la falta de los mismos.

## 2.4 Comunicación entre microcontroladores

El intercambio de datos entre el Arduino NANO y el ESP8266 (Wemos) se realizó por medio físico (cable), utilizando el puerto serie (USART). El módulo central (ESP) le solicita al módulo de medición (Arduino) información sobre el sistema, o bien puede enviarle una orden para modificar el estado de cualquiera de los actuadores, ya sea el encendido o apagado de la bomba, el ingreso de nutrientes o el agregado de agua. A su vez, tiene la función de enviarle los valores de los parámetros de configuración (tema que se tratará más adelante).

USART significa recepción y transmisión síncrona / asíncrona universal y es un protocolo de comunicación simple que permite que los micros se comuniquen mediante el puerto serie.

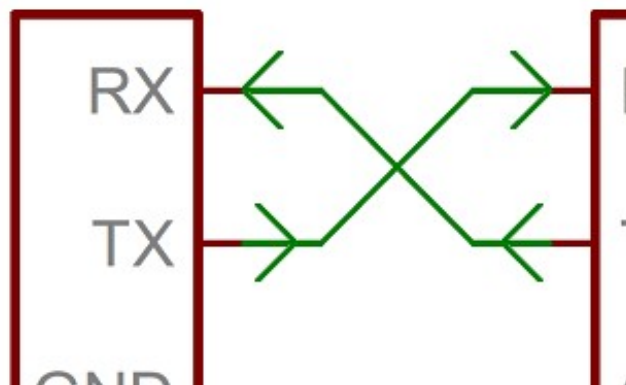


Figura 16 - Comunicación por medio del puerto serie

La comunicación serie envía una cadena de caracteres ASCII, los cuales son convertidos a char para su lectura, usando la sentencia `Serial.read()`

Para enviar datos utiliza la sentencia `Serial.print()`, la cual convierte a ASCII cualquier tipo de dato (float, int, char, etc).

Se debe inicializar primeramente el puerto serie, con la siguiente sentencia:

```
void setup(){
  //iniciamos el puerto serie
  Serial.begin(9600);
}
```

Figura 17 - Inicialización del puerto serie

En este caso se coloca dentro de los paréntesis la velocidad de transmisión en baudios. Luego en el loop se realiza el código de transmisión y recepción de datos.

Para el caso del Arduino, debido a que se buscó poder observar los datos que se enviaban al maestro mediante el uso del monitor serie disponible en el IDE de este, se utilizó el puerto serie virtual, mediante el uso de la librería `SoftwareSerial`.

```
SoftwareSerial mySerial(12,11); //
mySerial.begin(9600)
```

Figura 18 - Inicialización del puerto serie virtual

Las sentencias son las mismas que en el caso del puerto serie y se le puede asignar cualquier pin del Arduino para la transmisión y recepción de datos; en nuestro caso se asignaron los pines digitales 12 para la recepción y 11 para la transmisión.

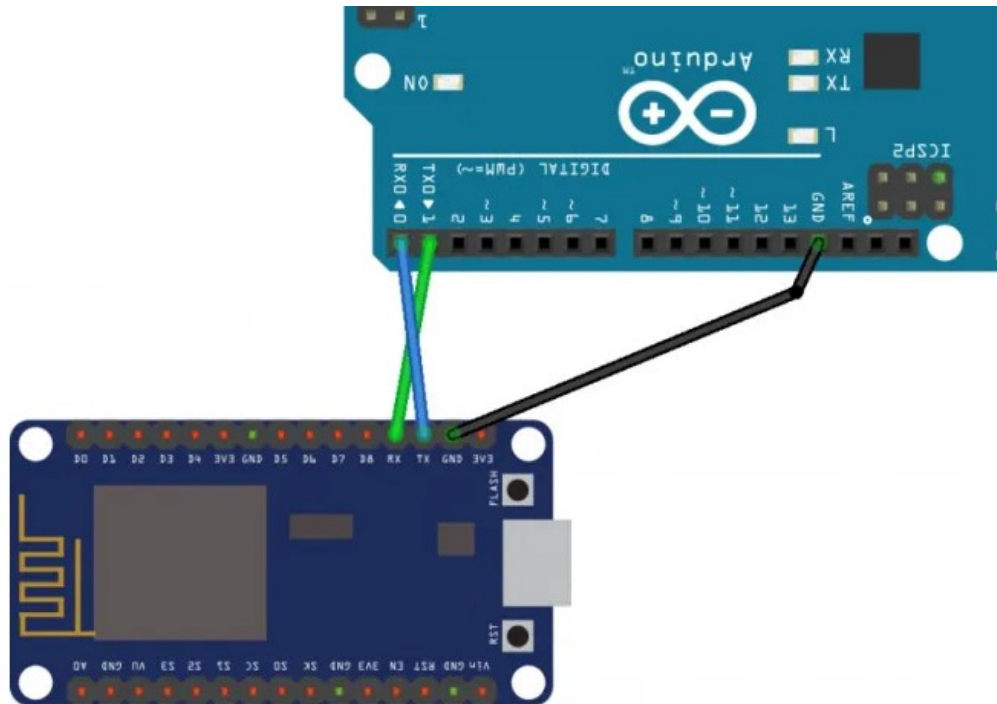


Figura 19 - Conexión entre ESP y Arduino

Debido a que la placa utilizada como maestro (Wemos D1 Mini) trabaja con tensiones de 3.3V, se “adaptó” la tensión de transmisión del Arduino (el cual trabaja a 5V) colocando un divisor resistivo.

Se logró entonces que el maestro le envíe al esclavo una solicitud de alguna medición requerida. Este último la recibe y comprueba que dato se le pide. Obtiene el valor del sensor correspondiente y responde. El valor recibido es enviado a la interfaz gráfica del usuario.

El sistema se diseñó de esta manera, con lógica Maestro – Esclavo con la finalidad de permitir su expansión en un futuro, con múltiples esclavos conectados a un único maestro, los cuales monitorean diferentes cultivos dentro de un mismo espacio.

## 2.5 MQTT

MQTT (Message Queuing Telemetry Transport) es un protocolo de red que transporta mensajes con patrón de publicación/suscripción. En esta estructura, los llamados clientes se conectan a un servidor central llamado Broker. Para filtrar los mensajes que se envían a los clientes, los mensajes se disponen en “tópicos” (topics), los cuales pueden recibir publicaciones de clientes, quienes a su vez pueden suscribirse a dicho tópico y el broker le hará llegar los mensajes del mismo.



Figura 20 – (Luis Llamas, 2019) Protocolo MQTT

Los clientes se comunican mediante protocolo TCP IP al broker, que a su vez lleva un registro de los que se conectan. Esta conexión permanecerá abierta hasta que el cliente la finalice.

En MQTT, el cliente envía un mensaje “connect” con el nombre de usuario, contraseña, ID, etc. y el broker responde con un “connect acknowledge” que informa el resultado de dicha conexión (aceptada o rechazada). Si la conexión es aceptada, el cliente puede publicar y suscribirse al broker, el cual confirmará si dicha suscripción es aceptada o no.

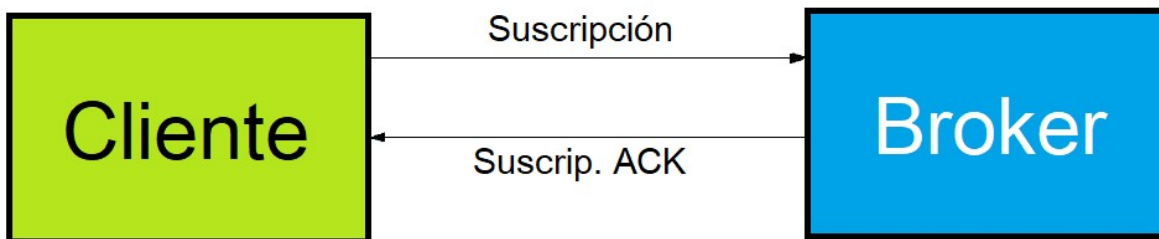


Figura 21 – (Luis Llamas, 2019) Suscripción al broker

En nuestro código, se utiliza la función `client.publish`, que envía al tópic “clientes” el mensaje que se quiere publicar en la página web, en este caso, los valores de los sensores.

```
String to_send = String(tempAmb) + "," + String(tempSol) + "," + String(ppm) + "," + String(caudal) +
to_send.toCharArray(msg, 50);
client.publish("nfhidroponia@gmail.com/clientes", msg);
```

Figura 22 - Código para publicación del mensaje

En la figura vemos que se arma un string con el mensaje, el cual se convierte a char array y finalmente es publicado para que todo aquel que esté suscripto a “clientes” pueda visualizar el mensaje.

Ahora bien, para ejecutar las acciones en el microcontrolador debe en primera instancia suscribirse a los tópicos de los cuales recibirá las acciones correspondientes.

```
client.subscribe("pfhidroponia@gmail.com/i
```

Figura 23 - Suscripción al tópico “bomba”

Como vemos, el cliente se suscribe al tópico “bomba” y luego puede recibir todo lo enviado a través de dicho tópico.

A modo de ejemplo, se observa en la figura siguiente un condicional, donde se analiza que si el tópico es “bomba”, entonces realizará las acciones para el encendido o apagado de la bomba.

En caso de que no sea dicho tópico, el programa continuará su ejecución analizando otros mensajes.

```
if (topico == "pfhidroponia@gmail.com/b
    FlagBomba = 1;
    if ( incoming == "on") {
        Serial.println('7'); //prender bomba
    }
    else if ( incoming == "off") {
        Serial.println('8'); // apagar bomb
    }
}
```

Figura 24 - Función condicional

Inicialmente, se quería utilizar un servidor tal como Amazon Web Services (AWS) o Google Cloud, el cual nos permite alojar todo nuestro sistema en un solo lugar teniendo control total sobre el mismo. El problema de esta opción es que excedía el presupuesto destinado al proyecto, por lo que se optó por utilizar dos servicios separados: por un lado el servidor que contiene la página web y por el otro el broker MQTT. Para la primera se eligió webhost y para el último, inicialmente se probó un broker de instalación local, llamado Mosquitto, que nos permitía no depender de brokers online, teniendo más control sobre el sistema. Esto llevó a diversas complicaciones que derivaron finalmente en la elección de un broker online: MaQiaTTo.

Durante el desarrollo del software del Maestro se descubrió un inconveniente al momento de enviar y recibir datos: el ESP se desconectaba del bróker sin razón aparente. Este error se debía a que por estándar MQTT, el bróker no mantiene la conexión si no registra actividad durante 10 segundos por parte del maestro. Esto se solucionó modificando dicho tiempo para que cumpla con nuestros periodos de ejecución.



## 2.6 Página Web

Al momento de crear la interfaz, se optó por una página web, en lugar de utilizar pantallas LCD, aplicaciones u otros medios. La misma se desarrolló utilizando tecnologías web actuales para el FrontEnd y el BackEnd.

Para el FrontEnd se utilizó HTML, que consiste en lenguaje marcado de hipertexto, el cual nos permite formar la estructura básica de nuestra web.

Por otro lado, para dar el estilo de la página, se usó Bootstrap, el cual es un framework que facilita la tarea de la programación de la interfaz.

Por otra parte, en el BackEnd se utilizó PHP, el cual se encarga de la autenticación de usuarios, toma de información de la Base de Datos, impresión de variables en la página web, entre otras. Asimismo, JavaScript se encarga de la conexión con el broker MQTT, el cual sirve para ver los datos en tiempo real.

Como ya se explicó, el maestro solicita al esclavo los valores registrados, estos datos serán enviados al broker mediante protocolo MQTT. Dichos valores son mostrados en la página web, sobre la cual también se pueden ejecutar acciones en el cultivo (ingresar tiempos de encendido/apagado de la bomba, realizar configuraciones, agregar nutrientes, entre otros), que son enviadas al maestro que transmite la orden al esclavo para que ejecute las tareas solicitadas.



Figura 25 - Vista de la sección de datos recibidos

Como se puede ver en la figura anterior, se muestran allí los valores que obtiene el esclavo y que envía el maestro mediante protocolo MQTT.



### 2.6.1 Agregado de Nutrientes

En la sección de nutrientes es donde se ingresan las cantidades que deben inyectarse de los mismos a la solución. Brinda cuatro opciones, que son las más comunes en cultivos de estas características: 35 mL, 70 mL, 105 mL, 140 mL. Al presionar confirmar, se envía la orden de abrir 3 electroválvulas que contienen los químicos.

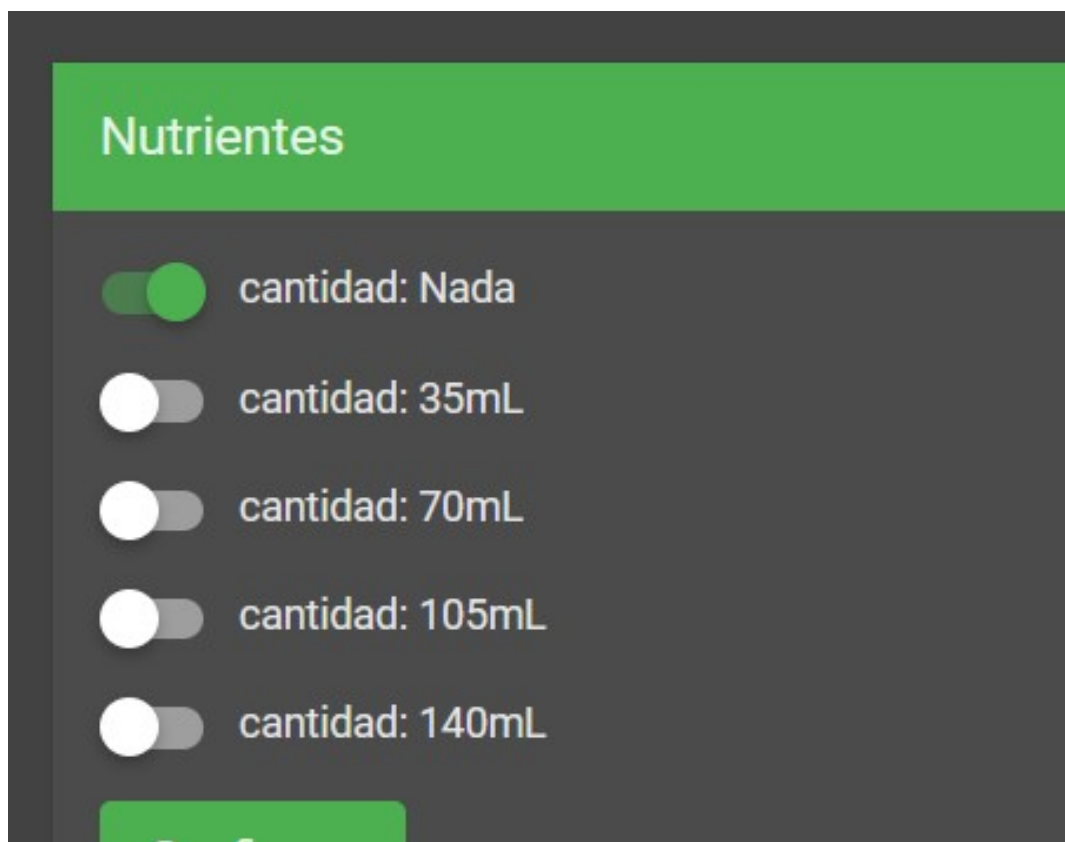


Figura 26 - Vista de la sección de nutrientes

Aquí se envía desde el maestro al esclavo la cantidad de mililitros que se desean agregar y el esclavo responde realizando la acción de abrir las electroválvulas durante un tiempo ya medido previamente (como ya se detalló, se modificaron los diámetros de salida de las electroválvulas para que ingrese la cantidad de nutrientes especificada).

### 2.6.2 Configuración de la Bomba

Debido a que el cultivo requiere la presencia constante de una lámina de solución nutritiva, el uso de una bomba es aquí fundamental para esta técnica. Pero su uso prolongado, sin interrupciones, puede provocar fallas en la misma o disminuir su vida útil. Y el sistema no requiere la circulación constante de nutrientes, sino que necesita de tiempos de “reposo” (durante la noche se recomienda dejar que las plantas “descansen”).

Dicho esto, se buscó que la bomba encienda y apague en períodos de tiempo que puedan ser determinados por el usuario, según lo desee. Se cuenta con una función para establecer la cantidad de minutos de encendido/apagado y fijar los horarios en los cuales se desea que la bomba cumpla estos regímenes de tiempo. Fuera de estos intervalos establecidos (noche), se puede también configurar lapsos de encendido y apagado pero más distanciados (por ejemplo, durante la noche puede encender la bomba 10 minutos y mantenerla apagada por 4 horas).

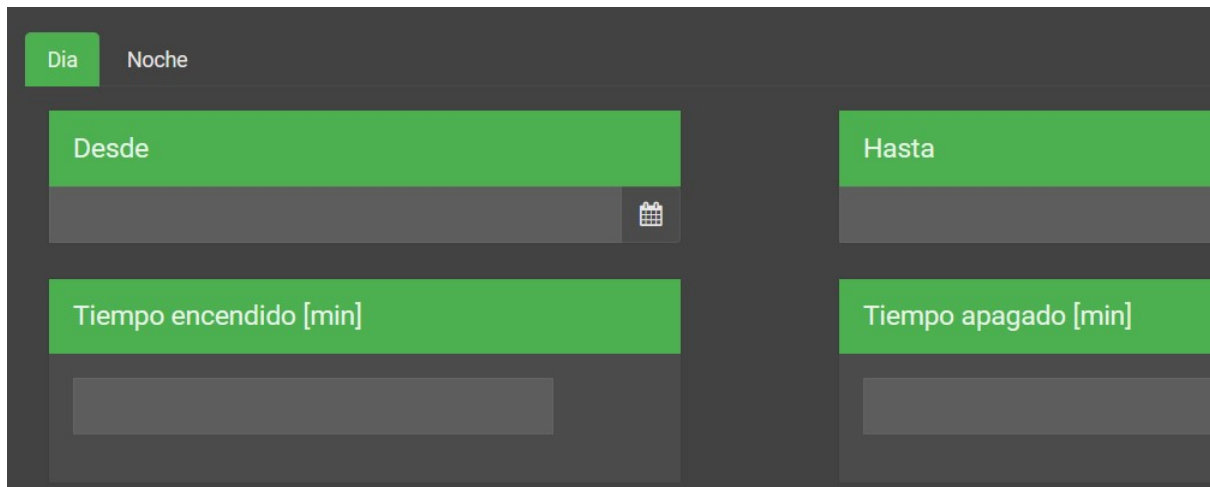
The image shows a user interface for configuring a pump. At the top, there are two tabs: 'Dia' (Day) and 'Noche' (Night). Below the tabs, there are four input fields arranged in a 2x2 grid. The top-left field is labeled 'Desde' (From) and has a calendar icon to its right. The top-right field is labeled 'Hasta' (Until). The bottom-left field is labeled 'Tiempo encendido [min]' (On time [min]). The bottom-right field is labeled 'Tiempo apagado [min]' (Off time [min]). All input fields are currently empty.

Figura 27 - Configuraciones de bomba

A modo de ejemplo, resulta conveniente en épocas de verano que la bomba, de 7 a 20 hs, se encienda durante 20 minutos y luego de transcurrido ese tiempo se apague por 20 minutos. Transcurrido este período, la bomba volverá a funcionar por otros 20 minutos, repitiendo este ciclo por las horas seleccionadas. Ahora, de 20 a 7 hs se considera “horario nocturno”, por lo que el usuario podrá configurar los tiempos de encendido y apagado durante dichas horas.

Para lograr esta configuración de horarios, se recurrió al uso de un RTC (Real Time Clock). Un RTC es un reloj en un circuito integrado, que permite mantener la fecha y hora actual. Están formados por un resonador de cristal integrado, el cual junto a componentes electrónicos permite contabilizar de manera precisa el paso del tiempo.

Como ventajas de estos, se pueden mencionar:

- Bajo consumo de energía (muy importante para nuestra aplicación)
- Permite que el sistema no deba consultar la fecha y hora mediante internet (quitándole tiempo a tareas más críticas o pudiendo no obtener el horario debido a una falla en la conexión wifi)
- Muy preciso en relación a otros métodos.
- Cuenta con una pila integrada, permitiendo mantener el valor del tiempo ante la pérdida de alimentación.

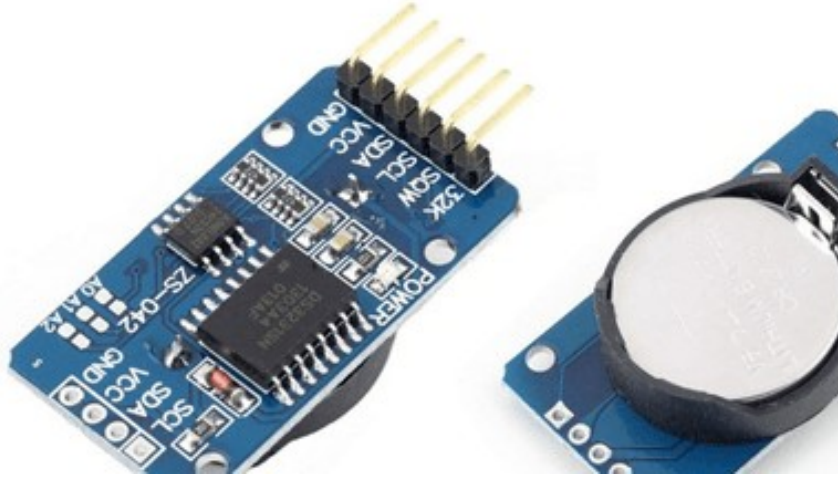


Figura 28 – (Llamas Luis, 2016) Real Time Clock (RTC)

La comunicación entre el microcontrolador y este dispositivo se realiza mediante el bus I2C (Inter-Integrated Circuit), un tipo de transmisión y recepción de datos que requiere únicamente de dos cables para su funcionamiento: uno para la señal de reloj (CLK) y otro para el envío de datos (SDA). Aquí cada dispositivo tiene una dirección dentro del bus, que se emplea para reconocer a los mismos y acceder a ellos, con una arquitectura del tipo maestro-esclavo.

Otro dato a resaltar del I2C es que el bus es síncrono: el maestro da una señal de reloj que mantiene la sincronización en todos los dispositivos conectados al bus, permitiendo que cada esclavo conectado pueda prescindir de un reloj propio.

Para realizar todas estas operaciones con un único cable, el bus I2C emplea una trama, que consta de: 7 bits que son la dirección del esclavo al cual se le enviará la información, un bit restante que indica si se desea enviar o recibir información, un bit para validación, uno o más bytes para los datos que se envían o reciben y un bit para validar los mismos.

Este método de comunicación tiene como principales ventajas su simpleza en cuanto a cantidad de cables a utilizar y que dispone de verificadores que permiten asegurar que la señal ha llegado a destino. Ahora bien, como desventajas se podrían mencionar su baja velocidad, no es full dúplex (es decir que solo dispone de un cable para enviar y recibir, por lo que no puede realizar las dos acciones al mismo tiempo) y no se puede comprobar si el contenido de los datos son correctos.

### 2.6.3 Comando Manual de la Bomba

Como una opción para el usuario, se tienen dos botones que permiten encender o apagar la bomba según se desee. Dichos botones anulan las configuraciones que se hayan realizado en los tiempos de encendido y apagado previamente, por lo que en caso de desear que la bomba funcione automáticamente se deberá configurar nuevamente.

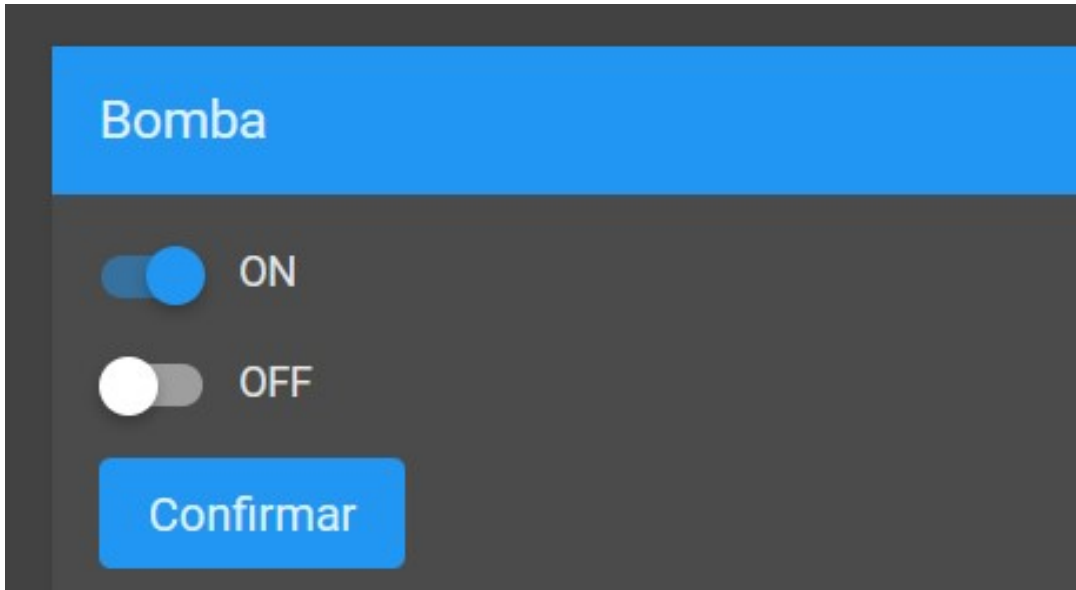


Figura 29 - Encendido y apagado de la bomba

### 2.6.4 Datos Históricos

La página incorpora a su vez una sección de datos históricos de PPM, informando el valor de las mismas, el día que se registró dicho dato y la hora. Esta sección permite al usuario mostrar 10, 25, 50 o 100 datos en la tabla, tiene un apartado de búsqueda en caso de que se desee filtrar por fecha, hora o cantidad de PPM y muestra la número de páginas que tiene la tabla.

También, permite ordenar la tabla, sea por fecha, hora o PPM.

HISTORICOS		
DATOS DE PPM		
Show	10	entries
PPM	FECHA	HORA
1451	28/6/2021	17:54
1456	28/6/2021	17:45
1470	28/6/2021	17:46
1785	8/6/2021	18:52
1786	8/6/2021	18:50
1786	8/6/2021	18:56
1789	8/6/2021	18:43
1793	8/6/2021	18:40

Figura 30 - Sección de Históricos

Se tomó como criterio que cada vez que la bomba se apaga, se envía desde el maestro la información con la fecha, hora y cantidad de PPM registradas en ese instante, a través del método POST. Dichos valores se mostrarán en la tabla una vez recibido el dato.

### 2.6.5 Alertas

Se incorporó una sección de “Alertas”, donde se informa si el sistema se quedó sin agua en algún momento, el día de este acontecimiento y la hora.

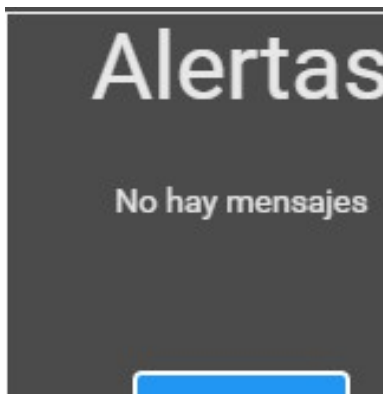


Figura 31 - Alertas que indicarían ausencia de agua. Se muestra que no hay nuevas alertas

Como el sistema resuelve automáticamente la ausencia de agua para no poner en riesgo el funcionamiento de la bomba. Este alerta es sólo para informarle al usuario que en un punto el cultivo, sufrió una baja de agua y puede requerir el agregado de nutrientes para concentrar más la solución.

### 2.6.6 Configuración de estilos

Agregado como un “extra” para la comodidad visual del usuario, la configuración de estilos permite cambiar el color del fondo de pantalla, según se desee.

Incluye cuatro temas: LIGHT (fondo blanco), GREY (fondo gris), BLACK (fondo negro) y DARK (fondo azul oscuro).

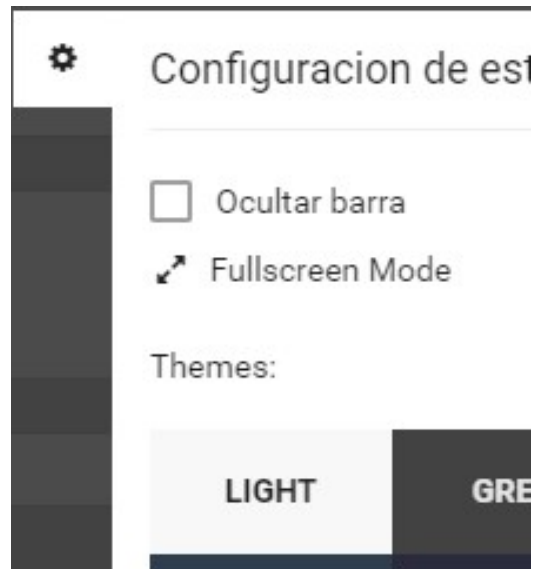


Figura 32 - Configuración de estilos

## 2.7 Wemos como Web Server (Configuración de wifi)

Como ya se explicó previamente, el Wemos se conecta a internet y envía los datos al broker mediante MQTT, el cual recibe los mismos y los muestra en la página web desarrollada. Ahora, para realizar todas estas acciones, el microcontrolador maestro requiere del usuario (SSID) y contraseña de la red wi-fi con la finalidad de comenzar con el funcionamiento

Para que el usuario ingrese los datos de su red, optamos por incluir una “conexión local”, es decir, que el Wemos actúe como servidor web.

Aquí, el maestro actúa como un Access Point (o punto de acceso) , a través del cual los dispositivos pueden conectarse al micro.

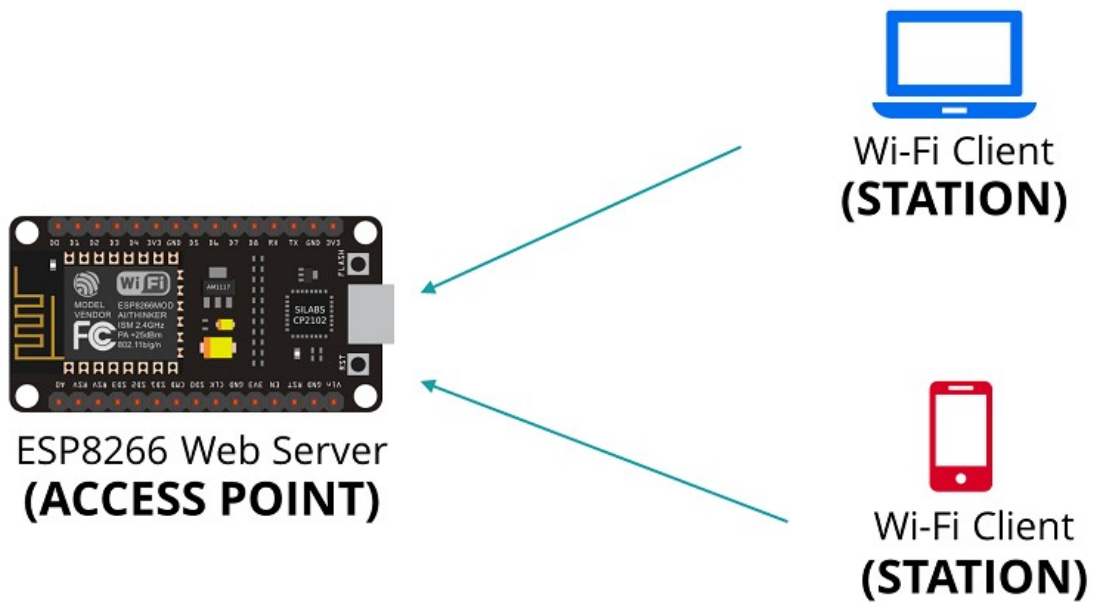


Figura 33 – (gndtovcchome.files.wordpress) Conexión del ESP8266 como Access Point

Para acceder a este modo, se debe colocar la llave selectora ubicada en el exterior de la caja en “Conexión Local” y apagar el equipo. Luego, al encenderlo nuevamente, el pin D5 del Wemos recibe una señal de ALTO, lo que genera la ejecución en su setup del modo web server.

Luego, el usuario tiene que conectarse a la red “PFHidroponía” e ingresar la clave. Una vez conectado, deberá ingresar al navegador y colocar la dirección IP 192.168.4.1, la cual lo llevará a la página alojada en el microcontrolador.

PF HIDROPONIA  
192.168.4.1

SSID:

PASSWORD:

GUARDAR

ESCANEAR

Figura 34 - Imagen de la página del Web Server

Como se observa en la figura anterior, la página permite el escaneo de las redes wi-fi disponibles para conectarse. El usuario ingresa su SSID y contraseña, luego presiona guardar.



Figura 35 - Configuración de red wifi

Al ejecutar esta acción, se guarda en la EEPROM del microcontrolador maestro los datos de la red wi-fi del usuario. Una vez realizada la configuración de la red, se debe volver la llave selectora a “modo remoto” y apagar nuevamente el equipo.

```

    if(digitalRead(Servidor_Activado) == H
    {
        modoconf();
    }
else if(digitalRead(Servidor_Activado) =
    {
        leer(0).toCharArray(ssid, 50);
        leer(50).toCharArray(password, 50);
        setup_wifi();
    }
}

```

Figura 36 - Código del setup para selección de modo

Como observamos en la imagen, si el pin-servidor está activado, se llama a la función “modoconf”, la cual pone en modo escucha para responder a la petición HTTP con un HTML sencillo como el que se mostró previamente. Mediante la interacción del usuario con la página se disparan eventos a dos funciones: una que graba en la memoria EEPROM la SSID y contraseña ingresada y otra que escanea las redes wi-fi a las que puede conectarse el Wemos para acceder a internet.

## 2.8 Placas de control y potencia

Finalmente, realizadas las pruebas en protoboard, con un funcionamiento correcto de los códigos, se procedió al diseño e impresión de las placas finales. Se optó por hacer dos placas separadas: una de control, que incluye a los microcontroladores y otra de potencia, que contiene los relés junto con los transistores y la conexión a la placa de control.

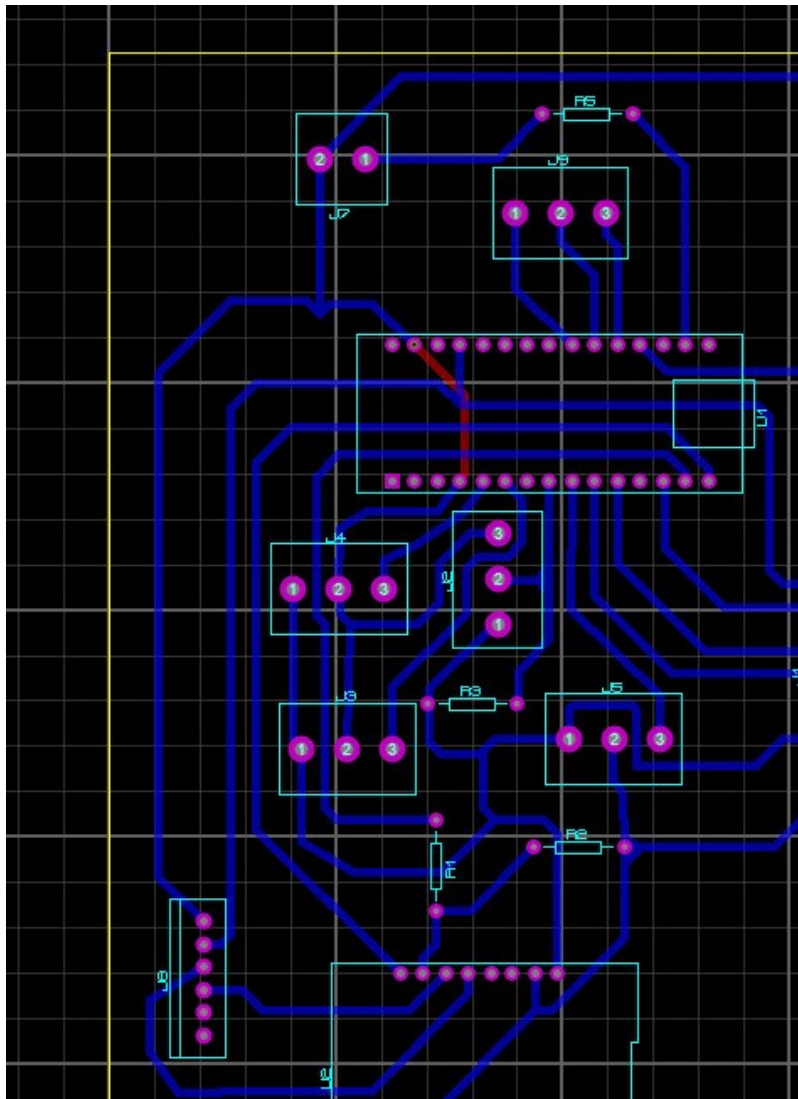


Figura 37 – Diseño en Proteus de la placa de control

Como vemos en la imagen, tenemos en la parte superior al Arduino NANO, por el medio están las borneras donde irán conectados los sensores y debajo el Wemos. El borne J1 es el de conexión con la placa de potencia, que se muestra a continuación.

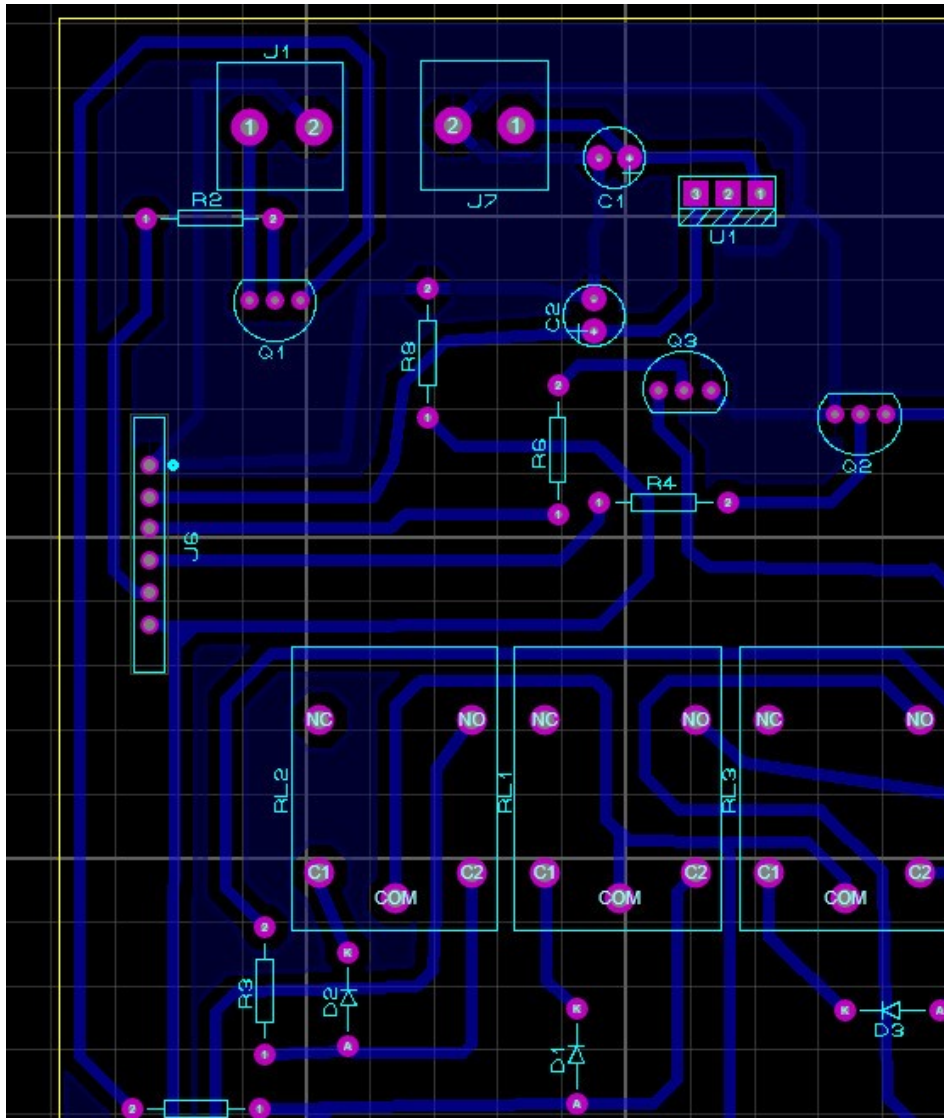


Figura 38 – Diseño en Proteus de la placa de potencia

En la figura anterior J6 es la bornera de conexión con la placa de control, en el centro están los relés que comandan la bomba, la inyección de nutrientes y el agua. Este último es independiente del usuario, sólo se acciona en caso de que el nivel del depósito esté bajo (monitoreado por el sensor de nivel).

### **Capítulo 3: Resultados**

Una vez concluido el dispositivo, se comprobó su correcto funcionamiento en diferentes situaciones que podrían generar fallas. Se observó que el equipo respondía de manera muy confiable a cada problemática, sin poner en riesgo el cultivo.

El producto terminado presenta el siguiente aspecto:



Figura 39 – Vista del exterior del equipo





Figura 40 – Vista frontal

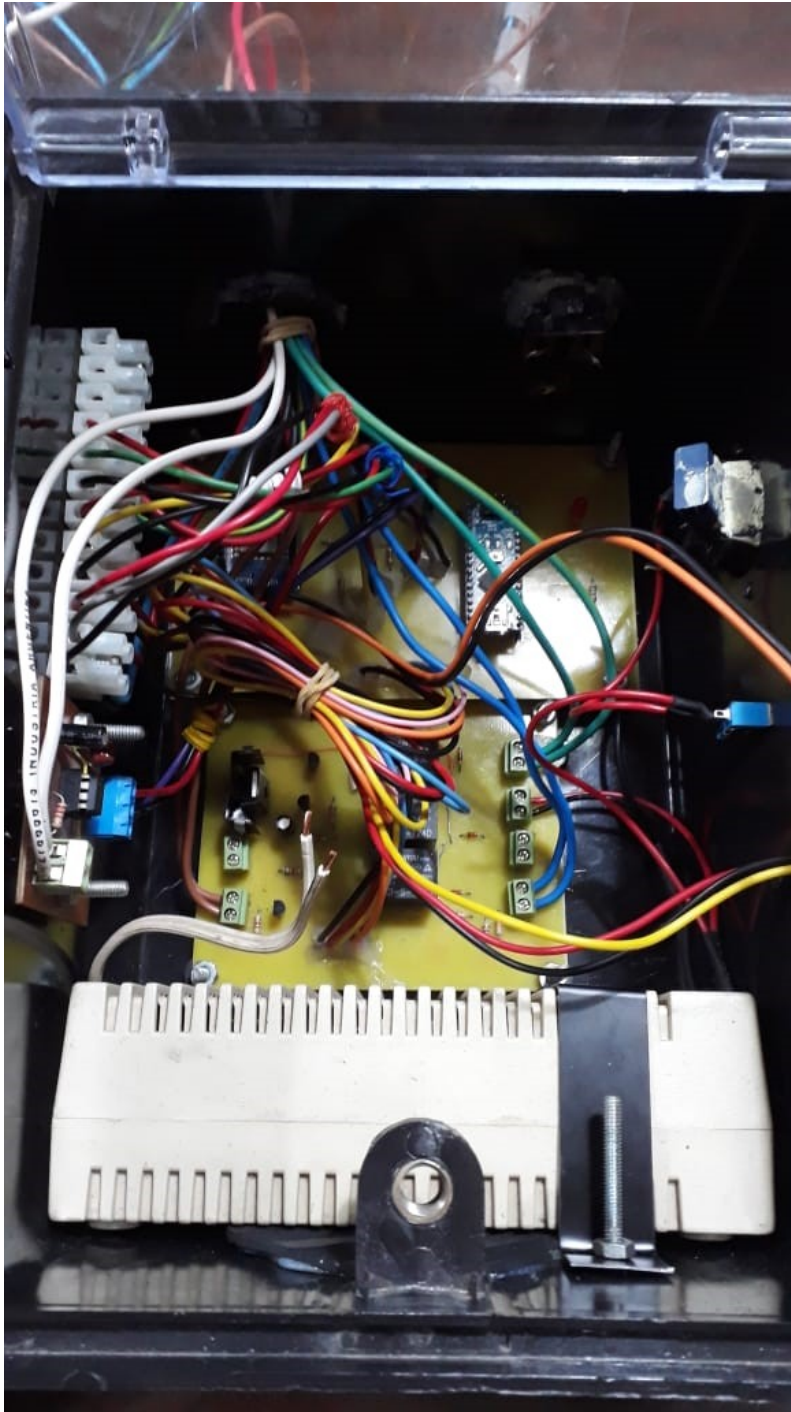


Figura 41 – Vista del interior del equipo

En la imagen anterior se observa el interior del dispositivo, donde se ven claramente las dos placas: la de control arriba y la de potencia debajo. Por debajo de las mismas se encuentra la fuente y a la izquierda la bornera de distribución de cables.

Se obtuvo entonces un dispositivo con la capacidad de medir de manera eficiente los parámetros principales de un cultivo hidropónico, actuando según lo requiera dicho cultivo o el mismo usuario.

## Capítulo 4: Análisis de Costos

En primer lugar, destacamos que el presente proyecto ha requerido un tiempo de desarrollo de las comunicaciones entre dispositivos, la obtención de datos y almacenamiento de ellos, la comunicación por medio de wifi y el desarrollo de una página web donde se muestra lo medido. Es decir, todo lo relacionado al software. A todo eso, se le suma también el tiempo dedicado al hardware: conexiones eléctricas, diseño y fabricación de placas, armado del dispositivo, entre otros.

Teniendo eso en cuenta y sumado a los costos materiales del proyecto, se realizó un presupuesto detallando los componentes utilizados y agregando al costo de estos, el de la mano de obra.

Presupuesto			
Componentes	Costo Unitario	Cantidad	Costo Total
<b>Microcontroladores</b>			
Arduino NANO	\$ 540,00	1	\$ 540,00
Wemos D1 mini	\$ 750,00	1	\$ 750,00
<b>Sensores</b>			
Sensor DHT11	\$ 256,00	1	\$ 256,00
Sensor de PPM	\$ 550,00	1	\$ 550,00
Sonda DS18B20	\$ 600,00	1	\$ 600,00
Sensor de Caudal	\$ 900,00	1	\$ 900,00
Sensor de Nivel	\$ 520,00	1	\$ 520,00
<b>Componentes electrónicos</b>			
Resistencias	\$ 100,00	1	\$ 100,00
Capacitores	\$ 150,00	1	\$ 150,00
Relés	\$ 120,00	4	\$ 480,00
Borneras	\$ 87,00	10	\$ 870,00
Transistores e integrados varios	\$ 60,00	4	\$ 240,00
RTC	\$ 400,00	1	\$ 400,00
Interruptores	\$ 87,00	2	\$ 174,00
Electroválvulas	\$ 265,00	4	\$ 1.060,00
<b>Materiales varios</b>			
Tubos contenedores de nutrientes	\$ 350,00	4	\$ 1.400,00
Caja estanca	\$ 2.500,00	1	\$ 2.500,00
Cables multifilares de 1 mm	\$ 26,50	30	\$ 795,00
<b>Mano de obra</b>			
Mano de obra hardware	\$ 1.980,00	125	\$ 247.500,00
Mano de obra software	\$ 2.130,00	145	\$ 308.850,00
<b>Total</b>			\$ 568.635,00

Tabla 1 – Análisis de Costos

Debido a que este diseño es un prototipo, los costos iniciales son más altos de lo que serían si se fabricara el producto en grandes cantidades. Aquí se observa que el valor del producto se ve muy afectado por el costo de la mano de obra, (el importe de la hora fue obtenido de la página oficial del Colegio de Ingenieros). Este total podría reducirse drásticamente, si se operase el producto en cantidades, pues el número de horas invertidas corresponden en muchos casos a diseños, pruebas, programaciones, que de realizarse a gran escala ya no serían contabilizadas.



## **Capítulo 5: Discusión y Conclusión.**

En base a lo visto durante el desarrollo del presente proyecto se observa que el mismo tiene grandes ventajas en el mercado local. Presenta beneficios en cuanto a su precio (si bien el prototipo tiene un costo elevado, si se produce en serie disminuiría notablemente), disponibilidad (en el mercado argentino no existen empresas dedicadas a la creación de estos dispositivos) y expansión (el equipo puede monitorear mayor cantidad de cultivos mediante la incorporación al mismo maestro de nuevos dispositivos esclavos).

Los resultados logrados por el sistema fueron muy buenos: el mismo responde muy bien ante situaciones imprevistas, mantiene el control ininterrumpido del cultivo (aún sin conexión a internet, ejecuta todas las acciones necesarias para su mantenimiento), actúa de manera independiente ante fallas que podrían generar daños al sistema (como es la falta de agua en depósito con la bomba en funcionamiento, se generaría una rotura de la misma), permite el control del usuario mediante internet y su monitoreo de los valores de PPM registrados en los últimos tiempos.

Si bien presenta grandes prestaciones, el producto es sólo un prototipo sujeto a mejoras sustanciales, tales como:

- Compra de página web y bróker, ya que se utilizan versiones gratuitas que no otorgan mucho control sobre el manejo de estos. Además, está subordinado a que en un futuro se pierdan los datos por una baja en el broker o en la página web.
- Expansión de la página para alojar a múltiples esclavos y permitir el control individual de los mismos. El presente proyecto sólo se realizó para un único esclavo controlado por el maestro.
- Agregado de un sistema de detección de fallas en la red eléctrica, que permita alertar cuando no hay energía que alimente el dispositivo.

Este proyecto se encaró con la idea que pueda ser ampliado para monitorear múltiples cultivos a la vez, mediante un único maestro, que reciba información de diferentes esclavos en tiempo real y actúe según se requiera. Por lo que una mejora también abarcaría el agregado de más esclavos, mayor cantidad de sensores y otras variables a ser monitoreadas mediante dispositivos con acceso a internet.

## Capítulo 6: Literatura Citada

- El Mundo (Agosto 2015). *Agricultores de Altura y Sin Suelo*.  
Enlace: <https://www.elmundo.es/economia/2015/08/16/55c37ceb268e3e184b8b4594.html>
- J. Smith Meza (2002). *Libro de Hidroponía y Nutrición*.  
Enlace: [https://www.academia.edu/4511736/Libro\\_de\\_hidroponia\\_y\\_nutricion](https://www.academia.edu/4511736/Libro_de_hidroponia_y_nutricion)
- Página web oficial de Arduino  
Enlace: <https://www.arduino.cc/>
- Raymundo Barrales Guadarrama. *Medidor de ph, electro-conductividad y temperatura para cultivos hidropónicos*. Pistas Educativas N° 120. México, 2016
- Figura 1. Hidroponía Urbana, 2017. Enlace: <https://hidroponiaurbana.com/sistema-hidroponico/sistema-de-cultivo-hidroponico-nft>
- Figura 4. Mecatronium Chips, 2020. Enlace: <https://soloarduino.blogspot.com/2014/12/arduino-nano.html>
- Figura 5. HWLibre, 2015. Enlace: <https://www.hwlibre.com/wemos/>
- Figura 6. Solo Arduino Blog, 2014. Enlace: <http://chips.mecatronium.com/product/sensor-de-temperatura-ds18b20-digital-tipo-sonda-sumergible-1m/>
- Figura 12. Rodrigo Martel, Tienda Virtual ML, 2016. Enlace: <https://articulo.mercadolibre.com.ar/>
- Figura 13. Naylamp Mechatronics, 2019. Enlace: [https://naylampmechatronics.com/blog/40\\_tutorial-sensor-de-temperatura-y-humedad-dht11-y-dht22.html](https://naylampmechatronics.com/blog/40_tutorial-sensor-de-temperatura-y-humedad-dht11-y-dht22.html)
- Figura 15. elagmecatronica.blogspot.com, 2013. Enlace: <http://elagmecatronica.blogspot.com/2013/05/>
- Figura 20. Llamas Luis, 2019. Enlace: <https://www.luisllamas.es/que-es-mqtt-su-importancia-como-protocolo-iot/>
- Figura 21. Llamas Luis, 2019. Enlace: <https://www.luisllamas.es/wp-content/uploads/2019/04/mqtt-suscribe.png>
- Figura 28. Llamas Luis, 2016. Enlace: <https://www.luisllamas.es/reloj-y-calendario-en-arduino-con-los-rtc-ds1307-y-ds3231/>
- Figura 33. gndtovcchome, 2020. Enlace: <https://gndtovcchome.files.wordpress.com/2020/04/39.3.png>

